

Contents

Windows security

Zero Trust and Windows

Hardware security

Overview

Trusted Platform Module

Trusted Platform Module Overview

TPM fundamentals

How Windows uses the TPM

TPM Group Policy settings

Back up the TPM recovery information to AD DS

View status, clear, or troubleshoot the TPM

Understanding PCR banks on TPM 2.0 devices

TPM recommendations

Hardware-based root of trust

System Guard Secure Launch and SMM protection

Enable virtualization-based protection of code integrity

Kernel DMA Protection

Windows secured-core devices

Operating system security

Overview

System security

Secure the Windows boot process

Trusted Boot

Cryptography and certificate management

The Windows Security app

Virus & threat protection

Account protection

Firewall & network protection

App & browser control

[Device security](#)

[Device performance & health](#)

[Family options](#)

[Security policy settings](#)

[Security auditing](#)

[Encryption and data protection](#)

[Encrypted Hard Drive](#)

[BitLocker](#)

[Overview of BitLocker Device Encryption in Windows](#)

[BitLocker frequently asked questions \(FAQ\)](#)

[Overview and requirements](#)

[Upgrading](#)

[Deployment and administration](#)

[Key management](#)

[BitLocker To Go](#)

[Active Directory Domain Services](#)

[Security](#)

[BitLocker Network Unlock](#)

[General](#)

[Prepare your organization for BitLocker: Planning and policies](#)

[BitLocker deployment comparison](#)

[BitLocker basic deployment](#)

[Deploy BitLocker on Windows Server 2012 and later](#)

[BitLocker management for enterprises](#)

[Enable Network Unlock with BitLocker](#)

[Use BitLocker Drive Encryption Tools to manage BitLocker](#)

[Use BitLocker Recovery Password Viewer](#)

[BitLocker Group Policy settings](#)

[BCD settings and BitLocker](#)

[BitLocker Recovery Guide](#)

[BitLocker Countermeasures](#)

[Protecting cluster shared volumes and storage area networks with BitLocker](#)

Troubleshoot BitLocker

Troubleshoot BitLocker

BitLocker cannot encrypt a drive: known issues

Enforcing BitLocker policies by using Intune: known issues

BitLocker Network Unlock: known issues

BitLocker recovery: known issues

BitLocker configuration: known issues

Troubleshoot BitLocker and TPM issues

BitLocker cannot encrypt a drive: known TPM issues

BitLocker and TPM: other known issues

Decode Measured Boot logs to track PCR changes

Configure S/MIME for Windows

Network security

VPN technical guide

VPN connection types

VPN routing decisions

VPN authentication options

VPN and conditional access

VPN name resolution

VPN auto-triggered profile options

VPN security features

VPN profile options

How to configure Diffie Hellman protocol over IKEv2 VPN connections

How to use single sign-on (SSO) over VPN and Wi-Fi connections

Optimizing Office 365 traffic with the Windows VPN client

Windows Defender Firewall

Windows security baselines

Security Compliance Toolkit

Get support

Virus & threat protection

Overview

Microsoft Defender Antivirus

Attack surface reduction rules

Tamper protection

Network protection

Controlled folder access

Exploit protection

Microsoft Defender for Endpoint

More Windows security

Override Process Mitigation Options to help enforce app-related security policies

Use Windows Event Forwarding to help with intrusion detection

Block untrusted fonts in an enterprise

Windows Information Protection (WIP)

Create a WIP policy using Microsoft Intune

Create a WIP policy with MDM using the Azure portal for Microsoft Intune

Deploy your WIP policy using the Azure portal for Microsoft Intune

Associate and deploy a VPN policy for WIP using the Azure portal for Microsoft Intune

Create and verify an EFS Data Recovery Agent (DRA) certificate

Determine the Enterprise Context of an app running in WIP

Create a WIP policy using Microsoft Endpoint Configuration Manager

Create and deploy a WIP policy using Microsoft Endpoint Configuration Manager

Create and verify an EFS Data Recovery Agent (DRA) certificate

Determine the Enterprise Context of an app running in WIP

Mandatory tasks and settings required to turn on WIP

Testing scenarios for WIP

Limitations while using WIP

How to collect WIP audit event logs

General guidance and best practices for WIP

Enlightened apps for use with WIP

Unenlightened and enlightened app behavior while using WIP

Recommended Enterprise Cloud Resources and Neutral Resources network settings with WIP

Using Outlook Web Access with WIP

Fine-tune WIP Learning

Application security

Overview

Windows Defender Application Control and virtualization-based protection of code integrity

Windows Defender Application Control

Microsoft Defender Application Guard

Windows Sandbox

Windows Sandbox architecture

Windows Sandbox configuration

Microsoft Defender SmartScreen overview

Configure S/MIME for Windows

Windows Credential Theft Mitigation Guide Abstract

User security and secured identity

Overview

Windows Hello for Business

Windows credential theft mitigation guide

Enterprise Certificate Pinning

Protect derived domain credentials with Credential Guard

How Credential Guard works

Credential Guard Requirements

Manage Credential Guard

Hardware readiness tool

Credential Guard protection limits

Considerations when using Credential Guard

Credential Guard: Additional mitigations

Credential Guard: Known issues

Protect Remote Desktop credentials with Remote Credential Guard

Technical support policy for lost or forgotten passwords

Access Control Overview

Dynamic Access Control Overview

Security identifiers

Security Principals

Local Accounts

Active Directory Accounts

Microsoft Accounts

Service Accounts

Active Directory Security Groups

Special Identities

User Account Control

How User Account Control works

User Account Control security policy settings

User Account Control Group Policy and registry key settings

Smart Cards

How Smart Card Sign-in Works in Windows

Smart Card Architecture

Certificate Requirements and Enumeration

Smart Card and Remote Desktop Services

Smart Cards for Windows Service

Certificate Propagation Service

Smart Card Removal Policy Service

Smart Card Tools and Settings

Smart Cards Debugging Information

Smart Card Group Policy and Registry Settings

Smart Card Events

Virtual Smart Cards

Understanding and Evaluating Virtual Smart Cards

Get Started with Virtual Smart Cards: Walkthrough Guide

Use Virtual Smart Cards

Deploy Virtual Smart Cards

Evaluate Virtual Smart Card Security

Tpmvscmgr

Cloud services

Overview

Mobile device management

[Windows 365 Cloud PCs](#)

[Azure Virtual Desktop](#)

[Security foundations](#)

[Overview](#)

[Microsoft Security Development Lifecycle](#)

[FIPS 140-2 Validation](#)

[Common Criteria Certifications](#)

[Windows Privacy](#)

Zero Trust and Windows device health

7/1/2022 • 4 minutes to read • [Edit Online](#)

Organizations need a security model that more effectively adapts to the complexity of the modern work environment. IT admins need to embrace the hybrid workplace, while protecting people, devices, apps, and data wherever they're located. Implementing a Zero Trust model for security helps address today's complex environments.

The [Zero Trust](#) principles are:

- **Verify explicitly.** Always authenticate and authorize based on all available data points, including user identity, location, device health, service or workload, data classification, and monitor anomalies.
- **Use least-privileged access.** Limit user access with just-in-time and just-enough-access, risk-based adaptive policies, and data protection to help secure data and maintain productivity.
- **Assume breach.** Prevent attackers from obtaining access to minimize potential damage to data and systems. Protect privileged roles, verify end-to-end encryption, use analytics to get visibility, and drive threat detection to improve defenses.

The Zero Trust concept of **verify explicitly** applies to the risks introduced by both devices and users. Windows enables **device health attestation** and **conditional access** capabilities, which are used to grant access to corporate resources.

[Conditional access](#) evaluates identity signals to confirm that users are who they say they are before they are granted access to corporate resources.

Windows 11 supports device health attestation, helping to confirm that devices are in a good state and have not been tampered with. This capability helps users access corporate resources whether they're in the office, at home, or when they're traveling.

Attestation helps verify the identity and status of essential components and that the device, firmware, and boot process have not been altered. Information about the firmware, boot process, and software, is used to validate the security state of the device. This information is cryptographically stored in the security co-processor Trusted Platform Module (TPM). Once the device is attested, it can be granted access to resources.

Device health attestation on Windows

Many security risks can emerge during the boot process as this process can be the most privileged component of the whole system. The verification process uses remote attestation as the secure channel to determine and present the device's health. Remote attestation determines:

- If the device can be trusted
- If the operating system booted correctly
- If the OS has the right set of security features enabled

These determinations are made with the help of a secure root of trust using the Trusted Platform Module (TPM). Devices can attest that the TPM is enabled, and that the device has not been tampered with.

Windows includes many security features to help protect users from malware and attacks. However, trusting the Windows security components can only be achieved if the platform boots as expected and was not tampered with. Windows relies on Unified Extensible Firmware Interface (UEFI) Secure Boot, Early-launch antimalware (ELAM), Dynamic Root of Trust for Measurement (DRTM), Trusted Boot, and other low-level hardware and

firmware security features. When you power on your PC until your anti-malware starts, Windows is backed with the appropriate hardware configuration to help keep you safe. [Measured and Trusted boot](#), implemented by bootloaders and BIOS, verifies and cryptographically records each step of the boot in a chained manner. These events are bound to a security coprocessor (TPM) that acts as the Root of Trust. Remote Attestation is the mechanism by which these events are read and verified by a service to provide a verifiable, unbiased, and tamper resilient report. Remote attestation is the trusted auditor of your system's boot, allowing specific entities to trust the device.

A summary of the steps involved in attestation and Zero Trust on the device side are as follows:

1. During each step of the boot process, such as a file load, update of special variables, and more, information such as file hashes and signature are measured in the TPM PCRs. The measurements are bound by a [Trusted Computing Group specification](#) (TCG) that dictates what events can be recorded and the format of each event.
2. Once Windows has booted, the attester/verifier requests the TPM to fetch the measurements stored in its Platform Configuration Register (PCR) alongside a TCG log. Both of these together form the attestation evidence that is then sent to the attestation service.
3. The TPM is verified by using the keys/cryptographic material available on the chipset with an [Azure Certificate Service](#).
4. This information is then sent to the attestation service in the cloud to verify that the device is safe. Microsoft Endpoint Manger integrates with Microsoft Azure Attestation to review device health comprehensively and connect this information with Azure Active Directory conditional access. This integration is key for Zero Trust solutions that help bind trust to an untrusted device.
5. The attestation service does the following:
 - Verify the integrity of the evidence. This is done by validating the PCRs that match the values recomputed by replaying the TCG log.
 - Verify that the TPM has a valid Attestation Identity Key issued by the authenticated TPM.
 - Verify that the security features are in the expected states.
6. The attestation service returns an attestation report that contains information about the security features based on the policy configured in the attestation service.
7. The device then sends the report to the Microsoft Endpoint Manager cloud to assess the trustworthiness of the platform according to the admin-configured device compliance rules.
8. Conditional access, along with device-compliance state then decides to allow or deny access.

Other Resources

Learn more about Microsoft Zero Trust solutions in the [Zero Trust Guidance Center](#).

Windows hardware security

7/1/2022 • 3 minutes to read • [Edit Online](#)

Modern threats require modern security with a strong alignment between hardware security and software security techniques to keep users, data, and devices protected. The operating system alone cannot protect from the wide range of tools and techniques cybercriminals use to compromise a computer deep inside its silicon. Once inside, intruders can be difficult to detect while engaging in multiple nefarious activities from stealing important data to capturing email addresses and other sensitive pieces of information. These new threats call for computing hardware that is secure down to the very core, including hardware chips and processors. Microsoft and our partners, including chip and device manufacturers, have worked together to integrate powerful security capabilities across software, firmware, and hardware.

SECURITY MEASURES	FEATURES & CAPABILITIES
Trusted Platform Module (TPM)	<p>A Trusted Platform Module (TPM) is designed to provide hardware-based security-related functions and help prevent unwanted tampering. TPMs provide security and privacy benefits for system hardware, platform owners, and users. A TPM chip is a secure crypto-processor that helps with actions such as generating, storing, and limiting the use of cryptographic keys. Many TPMs include multiple physical security mechanisms to make it tamper resistant and prevent malicious software from tampering with the security functions of the TPM.</p> <p>Learn more about the Trusted Platform Module.</p>
Hardware-based root of trust with Windows Defender System Guard	<p>To protect critical resources such as Windows authentication, single sign-on tokens, Windows Hello, and the Virtual Trusted Platform Module, a system's firmware and hardware must be trustworthy. Windows Defender System Guard helps protect and maintain the integrity of the system as it starts up and validate that system integrity has truly been maintained through local and remote attestation.</p> <p>Learn more about How a hardware-based root of trust helps protect Windows and System Guard Secure Launch and SMM protection.</p>

SECURITY MEASURES	FEATURES & CAPABILITIES
<p>Enable virtualization-based protection of code integrity</p>	<p>Hypervisor-protected Code Integrity (HVCI) is a virtualization based security (VBS) feature available in Windows. In the Windows Device Security settings, HVCI is referred to as Memory Integrity.</p> <p>HVCI and VBS improve the threat model of Windows and provide stronger protections against malware trying to exploit the Windows Kernel. VBS uses the Windows Hypervisor to create an isolated virtual environment that becomes the root of trust of the OS that assumes the kernel can be compromised. HVCI is a critical component that protects and hardens this virtual environment by running kernel mode code integrity within it and restricting kernel memory allocations that could be used to compromise the system.</p> <p>Learn more: Enable virtualization-based protection of code integrity.</p>
<p>Kernel Direct Memory Access (DMA) Protection</p>	<p>PCIe hot plug devices such as Thunderbolt, USB4, and CExpress allow users to attach new classes of external peripherals, including graphics cards or other PCI devices, to their PCs with an experience identical to USB. Because PCI hot plug ports are external and easily accessible, PCs are susceptible to drive-by Direct Memory Access (DMA) attacks. Memory access protection (also known as Kernel DMA Protection) protects PCs against drive-by DMA attacks that use PCIe hot plug devices by limiting these external peripherals from being able to directly copy memory when the user has locked their PC.</p> <p>Learn more about Kernel DMA Protection.</p>
<p>Secured-core PCs</p>	<p>Microsoft is working closely with OEM partners and silicon vendors to build Secured-core PCs that feature deeply integrated hardware, firmware, and software to ensure enhanced security for devices, identities, and data.</p> <p>Secured-core PCs provide protections that are useful against sophisticated attacks and can provide increased assurance when handling mission-critical data in some of the most data-sensitive industries, such as healthcare workers that handle medical records and other personally identifiable information (PII), commercial roles that handle high business impact and highly sensitive data, such as a financial controller with earnings data.</p> <p>Learn more about Secured-core PCs.</p>

Trusted Platform Module

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016 and above

Trusted Platform Module (TPM) technology is designed to provide hardware-based, security-related functions. A TPM chip is a secure crypto-processor that helps you with actions such as generating, storing, and limiting the use of cryptographic keys. The following topics provide details.

TOPIC	DESCRIPTION
Trusted Platform Module Overview	Provides an overview of the Trusted Platform Module (TPM) and how Windows uses it for access control and authentication.
TPM fundamentals	Provides background about how a TPM can work with cryptographic keys. Also describes technologies that work with the TPM, such as TPM-based virtual smart cards.
TPM Group Policy settings	Describes TPM services that can be controlled centrally by using Group Policy settings.
Back up the TPM recovery information to AD DS	For Windows 10, version 1511 and Windows 10, version 1507 only, describes how to back up a computer's TPM information to Active Directory Domain Services.
Troubleshoot the TPM	Describes actions you can take through the TPM snap-in, TPM.msc: view TPM status, troubleshoot TPM initialization, and clear keys from the TPM. Also, for TPM 1.2 and Windows 10, version 1507 or 1511, or Windows 11, describes how to turn the TPM on or off.
Understanding PCR banks on TPM 2.0 devices	Provides background about what happens when you switch PCR banks on TPM 2.0 devices.
TPM recommendations	Discusses aspects of TPMs such as the difference between TPM 1.2 and 2.0, and the Windows features for which a TPM is required or recommended.

Trusted Platform Module Technology Overview

7/1/2022 • 4 minutes to read • [Edit Online](#)

Applies to

- Windows 11
- Windows 10
- Windows Server 2016
- Windows Server 2019

This topic for the IT professional describes the Trusted Platform Module (TPM) and how Windows uses it for access control and authentication.

Feature description

[Trusted Platform Module \(TPM\)](#) technology is designed to provide hardware-based, security-related functions. A TPM chip is a secure crypto-processor that is designed to carry out cryptographic operations. The chip includes multiple physical security mechanisms to make it tamper-resistant, and malicious software is unable to tamper with the security functions of the TPM. Some of the key advantages of using TPM technology are that you can:

- Generate, store, and limit the use of cryptographic keys.
- Use TPM technology for platform device authentication by using the TPM's unique RSA key, which is burned into it.
- Help ensure platform integrity by taking and storing security measurements.

The most common TPM functions are used for system integrity measurements and for key creation and use. During the boot process of a system, the boot code that is loaded (including firmware and the operating system components) can be measured and recorded in the TPM. The integrity measurements can be used as evidence for how a system started and to make sure that a TPM-based key was used only when the correct software was used to boot the system.

TPM-based keys can be configured in a variety of ways. One option is to make a TPM-based key unavailable outside the TPM. This is good to mitigate phishing attacks because it prevents the key from being copied and used without the TPM. TPM-based keys can also be configured to require an authorization value to use them. If too many incorrect authorization guesses occur, the TPM will activate its dictionary attack logic and prevent further authorization value guesses.

Different versions of the TPM are defined in specifications by the Trusted Computing Group (TCG). For more information, consult the [TCG Web site](#).

Automatic initialization of the TPM with Windows

Starting with Windows 10 and Windows 11, the operating system automatically initializes and takes ownership of the TPM. This means that in most cases, we recommend that you avoid configuring the TPM through the TPM management console, `TPM.msc`. There are a few exceptions, mostly related to resetting or performing a clean installation on a PC. For more information, see [Clear all the keys from the TPM](#). We're [no longer actively developing the TPM management console](#) beginning with Windows Server 2019 and Windows 10, version 1809.

In certain specific enterprise scenarios limited to Windows 10, versions 1507 and 1511, Group Policy might be used to back up the TPM owner authorization value in Active Directory. Because the TPM state persists across

operating system installations, this TPM information is stored in a location in Active Directory that is separate from computer objects.

Practical applications

Certificates can be installed or created on computers that are using the TPM. After a computer is provisioned, the RSA private key for a certificate is bound to the TPM and cannot be exported. The TPM can also be used as a replacement for smart cards, which reduces the costs associated with creating and disbursing smart cards.

Automated provisioning in the TPM reduces the cost of TPM deployment in an enterprise. New APIs for TPM management can determine if TPM provisioning actions require physical presence of a service technician to approve TPM state change requests during the boot process.

Antimalware software can use the boot measurements of the operating system start state to prove the integrity of a computer running Windows 10 or Windows 11 or Windows Server 2016. These measurements include the launch of Hyper-V to test that datacenters using virtualization are not running untrusted hypervisors. With BitLocker Network Unlock, IT administrators can push an update without concerns that a computer is waiting for PIN entry.

The TPM has several Group Policy settings that might be useful in certain enterprise scenarios. For more info, see [TPM Group Policy Settings](#).

New and changed functionality

For more info on new and changed functionality for Trusted Platform Module in Windows, see [What's new in Trusted Platform Module?](#)

Device health attestation

Device health attestation enables enterprises to establish trust based on hardware and software components of a managed device. With device health attestation, you can configure an MDM server to query a health attestation service that will allow or deny a managed device access to a secure resource.

Some things that you can check on the device are:

- Is Data Execution Prevention supported and enabled?
- Is BitLocker Drive Encryption supported and enabled?
- Is SecureBoot supported and enabled?

NOTE

Windows 11, Windows 10, Windows Server 2016, and Windows Server 2019 support Device Health Attestation with TPM 2.0. Support for TPM 1.2 was added beginning with Windows version 1607 (RS1). TPM 2.0 requires UEFI firmware. A computer with legacy BIOS and TPM 2.0 won't work as expected.

Supported versions for device health attestation

TPM VERSION	WINDOWS 11	WINDOWS 10	WINDOWS SERVER 2016	WINDOWS SERVER 2019
TPM 1.2		>= ver 1607	>= ver 1607	Yes
TPM 2.0	Yes	Yes	Yes	Yes

Related topics

- [Trusted Platform Module](#) (list of topics)
- [Details on the TPM standard](#) (has links to features using TPM)
- [TPM Base Services Portal](#)
- [TPM Base Services API](#)
- [TPM Cmdlets in Windows PowerShell](#)
- [Prepare your organization for BitLocker: Planning and Policies - TPM configurations](#)
- [Azure device provisioning: Identity attestation with TPM](#)
- [Azure device provisioning: A manufacturing timeline for TPM devices](#)
- [Windows 10: Enabling vTPM \(Virtual TPM\)](#)
- [How to Multiboot with Bitlocker, TPM, and a Non-Windows OS](#)

TPM fundamentals

7/1/2022 • 11 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016 and later

This article for the IT professional provides a description of the components of the Trusted Platform Module (TPM 1.2 and TPM 2.0) and explains how they are used to mitigate dictionary attacks.

A Trusted Platform Module (TPM) is a microchip designed to provide basic security-related functions, primarily involving encryption keys. The TPM is installed on the motherboard of a computer, and it communicates with the rest of the system by using a hardware bus.

Computers that incorporate a TPM can create cryptographic keys and encrypt them so that they can only be decrypted by the TPM. This process, often called wrapping or binding a key, can help protect the key from disclosure. Each TPM has a master wrapping key, called the storage root key, which is stored within the TPM itself. The private portion of a storage root key or endorsement key that is created in a TPM is never exposed to any other component, software, process, or user.

You can specify whether encryption keys that are created by the TPM can be migrated or not. If you specify that they can be migrated, the public and private portions of the key can be exposed to other components, software, processes, or users. If you specify that encryption keys cannot be migrated, the private portion of the key is never exposed outside the TPM.

Computers that incorporate a TPM can also create a key that is wrapped and tied to certain platform measurements. This type of key can be unwrapped only when those platform measurements have the same values that they had when the key was created. This process is referred to as "sealing the key to the TPM." Decrypting the key is called unsealing. The TPM can also seal and unseal data that is generated outside the TPM. With this sealed key and software, such as BitLocker Drive Encryption, you can lock data until specific hardware or software conditions are met.

With a TPM, private portions of key pairs are kept separate from the memory that is controlled by the operating system. Keys can be sealed to the TPM, and certain assurances about the state of a system (assurances that define the trustworthiness of a system) can be made before the keys are unsealed and released for use. The TPM uses its own internal firmware and logic circuits to process instructions. Hence, it doesn't rely on the operating system and it isn't exposed to vulnerabilities that might exist in the operating system or application software.

For info about which versions of Windows support which versions of the TPM, see [Trusted Platform Module technology overview](#). The features that are available in the versions are defined in specifications by the Trusted Computing Group (TCG). For more info, see the Trusted Platform Module page on the Trusted Computing Group website: [Trusted Platform Module](#).

The following sections provide an overview of the technologies that support the TPM:

- [Measured Boot with support for attestation](#)
- [TPM-based Virtual Smart Card](#)
- [TPM-based certificate storage](#)
- [TPM Cmdlets](#)

- [Physical presence interface](#)
- [TPM 1.2 states and initialization](#)
- [Endorsement keys](#)
- [TPM Key Attestation](#)
- [Anti-hammering](#)

The following topic describes the TPM Services that can be controlled centrally by using Group Policy settings: [TPM Group Policy Settings](#).

Measured Boot with support for attestation

The Measured Boot feature provides antimalware software with a trusted (resistant to spoofing and tampering) log of all boot components. Antimalware software can use the log to determine whether components that ran before it are trustworthy versus infected with malware. It can also send the Measured Boot logs to a remote server for evaluation. The remote server can start remediation actions by interacting with software on the client or through out-of-band mechanisms, as appropriate.

TPM-based Virtual Smart Card

The Virtual Smart Card emulates the functionality of traditional smart cards. Virtual Smart Cards use the TPM chip that is available on an organization's computers, rather than using a separate physical smart card and reader. This greatly reduces the management and deployment cost of smart cards in an enterprise. To the end user, the Virtual Smart Card is always available on the computer. If a user needs to use more than one computer, a Virtual Smart Card must be issued to the user for each computer. A computer that is shared among multiple users can host multiple Virtual Smart Cards, one for each user.

TPM-based certificate storage

The TPM protects certificates and RSA keys. The TPM key storage provider (KSP) provides easy and convenient use of the TPM as a way of strongly protecting private keys. The TPM KSP generates keys when an organization enrolls for certificates. The KSP is managed by templates in the UI. The TPM also protects certificates that are imported from an outside source. TPM-based certificates are standard certificates. The certificate can never leave the TPM from which the keys are generated. The TPM can now be used for crypto-operations through Cryptography API: Next Generation (CNG). For more info, see [Cryptography API: Next Generation](#).

TPM Cmdlets

You can manage the TPM using Windows PowerShell. For details, see [TPM Cmdlets in Windows PowerShell](#).

Physical presence interface

For TPM 1.2, the TCG specifications for TPMs require physical presence (typically, pressing a key) for turning on the TPM, turning it off, or clearing it. These actions typically cannot be automated with scripts or other automation tools unless the individual OEM supplies them.

TPM 1.2 states and initialization

TPM 1.2 has multiple possible states. Windows automatically initializes the TPM, which brings it to an enabled, activated, and owned state.

Endorsement keys

A trusted application can use TPM only if the TPM contains an endorsement key, which is an RSA key pair. The private half of the key pair is held inside the TPM and it is never revealed or accessible outside the TPM.

Key attestation

TPM key attestation allows a certification authority to verify that a private key is protected by a TPM and that the TPM is one that the certification authority trusts. Endorsement keys proven valid are used to bind the user identity to a device. The user certificate with a TPM attested key provides higher security assurance backed up by the non-exportability, anti-hammering, and isolation of keys provided by a TPM.

Anti-hammering

When a TPM processes a command, it does so in a protected environment, for example, a dedicated microcontroller on a discrete chip or a special hardware-protected mode on the main CPU. A TPM is used to create a cryptographic key that is not disclosed outside the TPM. It is used in the TPM after the correct authorization value is provided.

TPMs have anti-hammering protection that is designed to prevent brute force attacks, or more complex dictionary attacks, that attempt to determine authorization values for using a key. The basic approach is for the TPM to allow only a limited number of authorization failures before it prevents more attempts to use keys and locks. Providing a failure count for individual keys is not technically practical, so TPMs have a global lockout when too many authorization failures occur.

Because many entities can use the TPM, a single authorization success cannot reset the TPM's anti-hammering protection. This prevents an attacker from creating a key with a known authorization value and then using it to reset the TPM's protection. TPMs are designed to forget about authorization failures after a period of time so the TPM does not enter a lockout state unnecessarily. A TPM owner password can be used to reset the TPM's lockout logic.

TPM 2.0 anti-hammering

TPM 2.0 has well defined anti-hammering behavior. This is in contrast to TPM 1.2 for which the anti-hammering protection was implemented by the manufacturer and the logic varied widely throughout the industry.

For systems with TPM 2.0, the TPM is configured by Windows to lock after 32 authorization failures and to forget one authorization failure every 10 minutes. This means that a user could quickly attempt to use a key with the wrong authorization value 32 times. For each of the 32 attempts, the TPM records if the authorization value was correct or not. This inadvertently causes the TPM to enter a locked state after 32 failed attempts.

Attempts to use a key with an authorization value for the next 10 minutes would not return success or failure; instead the response indicates that the TPM is locked. After 10 minutes, one authorization failure is forgotten and the number of authorization failures remembered by the TPM drops to 31, so the TPM leaves the locked state and returns to normal operation. With the correct authorization value, keys could be used normally if no authorization failures occur during the next 10 minutes. If a period of 320 minutes elapses with no authorization failures, the TPM does not remember any authorization failures, and 32 failed attempts could occur again.

Windows 8 Certification does not require TPM 2.0 systems to forget about authorization failures when the system is fully powered off or when the system has hibernated. Windows does require that authorization failures are forgotten when the system is running normally, in a sleep mode, or in low power states other than off. If a Windows system with TPM 2.0 is locked, the TPM leaves lockout mode if the system is left on for 10 minutes.

The anti-hammering protection for TPM 2.0 can be fully reset immediately by sending a reset lockout command to the TPM and providing the TPM owner password. By default, Windows automatically provisions TPM 2.0 and stores the TPM owner password for use by system administrators.

In some enterprise situations, the TPM owner authorization value is configured to be stored centrally in Active

Directory, and it is not stored on the local system. An administrator can launch the TPM MMC and choose to reset the TPM lockout time. If the TPM owner password is stored locally, it is used to reset the lockout time. If the TPM owner password is not available on the local system, the administrator needs to provide it. If an administrator attempts to reset the TPM lockout state with the wrong TPM owner password, the TPM does not allow another attempt to reset the lockout state for 24 hours.

TPM 2.0 allows some keys to be created without an authorization value associated with them. These keys can be used when the TPM is locked. For example, BitLocker with a default TPM-only configuration is able to use a key in the TPM to start Windows, even when the TPM is locked.

Rationale behind the defaults

Originally, BitLocker allowed from 4 to 20 characters for a PIN. Windows Hello has its own PIN for logon, which can be 4 to 127 characters. Both BitLocker and Windows Hello use the TPM to prevent PIN brute-force attacks.

Windows 10, version 1607 and earlier used Dictionary Attack Prevention parameters. The Dictionary Attack Prevention Parameters provide a way to balance security needs with usability. For example, when BitLocker is used with a TPM + PIN configuration, the number of PIN guesses is limited over time. A TPM 2.0 in this example could be configured to allow only 32 PIN guesses immediately, and then only one more guess every two hours. This totals a maximum of about 4415 guesses per year. If the PIN is 4 digits, all 9999 possible PIN combinations could be attempted in a little over two years.

Beginning with Windows 10, version 1703, the minimum length for the BitLocker PIN was increased to 6 characters to better align with other Windows features that leverage TPM 2.0, including Windows Hello. Increasing the PIN length requires a greater number of guesses for an attacker. Therefore, the lockout duration between each guess was shortened to allow legitimate users to retry a failed attempt sooner while maintaining a similar level of protection. In case the legacy parameters for lockout threshold and recovery time need to be used, make sure that GPO is enabled and [configure the system to use legacy Dictionary Attack Prevention Parameters setting for TPM 2.0](#).

TPM-based smart cards

The Windows TPM-based smart card, which is a virtual smart card, can be configured to allow sign in to the system. In contrast with physical smart cards, the sign-in process uses a TPM-based key with an authorization value. The following list shows the advantages of virtual smart cards:

- Physical smart cards can enforce lockout for only the physical smart card PIN, and they can reset the lockout after the correct PIN is entered. With a virtual smart card, the TPM's anti-hammering protection is not reset after a successful authentication. The allowed number of authorization failures before the TPM enters lockout includes many factors.
- Hardware manufacturers and software developers have the option to use the security features of the TPM to meet their requirements.
- The intent of selecting 32 failures as the lock-out threshold is so users rarely lock the TPM (even when learning to type new passwords or if they frequently lock and unlock their computers). If users lock the TPM, they must wait 10 minutes or use some other credential to sign in, such as a user name and password.

Related topics

- [Trusted Platform Module](#) (list of topics)
- [TPM Cmdlets in Windows PowerShell](#)
- [TPM WMI providers](#)
- [Prepare your organization for BitLocker: Planning and Policies - TPM configurations](#)

How Windows uses the Trusted Platform Module

7/1/2022 • 22 minutes to read • [Edit Online](#)

The Windows operating system improves most existing security features in the operating system and adds groundbreaking new security features such as Device Guard and Windows Hello for Business. It places hardware-based security deeper inside the operating system than previous Windows versions had done, maximizing platform security while increasing usability. To achieve many of these security enhancements, Windows makes extensive use of the Trusted Platform Module (TPM). This article offers a brief overview of the TPM, describes how it works, and discusses the benefits that TPM brings to Windows and the cumulative security impact of running Windows on a PC that contains a TPM.

See also:

- [Windows 11 Specifications](#)
- [Windows 10 Specifications](#)
- [TPM Fundamentals](#)
- [TPM Recommendations](#)

TPM Overview

The TPM is a cryptographic module that enhances computer security and privacy. Protecting data through encryption and decryption, protecting authentication credentials, and proving which software is running on a system are basic functionalities associated with computer security. The TPM helps with all these scenarios and more.

Historically, TPMs have been discrete chips soldered to a computer's motherboard. Such implementations allow the computer's original equipment manufacturer (OEM) to evaluate and certify the TPM separate from the rest of the system. Although discrete TPM implementations are still common, they can be problematic for integrated devices that are small or have low power consumption. Some newer TPM implementations integrate TPM functionality into the same chipset as other platform components while still providing logical separation similar to discrete TPM chips.

TPMs are passive: they receive commands and return responses. To realize the full benefit of a TPM, the OEM must carefully integrate system hardware and firmware with the TPM to send it commands and react to its responses. TPMs were originally designed to provide security and privacy benefits to a platform's owner and users, but newer versions can provide security and privacy benefits to the system hardware itself. Before it can be used for advanced scenarios, a TPM must be provisioned. Windows automatically provisions a TPM, but if the user reinstalls the operating system, user may need to tell the operating system to explicitly provision the TPM again before it can use all the TPM's features.

The Trusted Computing Group (TCG) is the nonprofit organization that publishes and maintains the TPM specification. The TCG exists to develop, define, and promote vendor-neutral, global industry standards that support a hardware-based root of trust for interoperable trusted computing platforms. The TCG also publishes the TPM specification as the international standard ISO/IEC 11889, using the Publicly Available Specification Submission Process that the Joint Technical Committee 1 defines between the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC).

OEMs implement the TPM as a component in a trusted computing platform, such as a PC, tablet, or phone. Trusted computing platforms use the TPM to support privacy and security scenarios that software alone cannot achieve. For example, software alone cannot reliably report whether malware is present during the system

startup process. The close integration between TPM and platform increases the transparency of the startup process and supports evaluating device health by enabling reliable measuring and reporting of the software that starts the device. Implementation of a TPM as part of a trusted computing platform provides a hardware root of trust—that is, it behaves in a trusted way. For example, if a key stored in a TPM has properties that disallow exporting the key, that key *truly cannot leave the TPM*.

The TCG designed the TPM as a low-cost, mass-market security solution that addresses the requirements of different customer segments. There are variations in the security properties of different TPM implementations just as there are variations in customer and regulatory requirements for different sectors. In public-sector procurement, for example, some governments have clearly defined security requirements for TPMs, whereas others do not.

Certification programs for TPMs—and technology in general—continue to evolve as the speed of innovation increases. Although having a TPM is clearly better than not having a TPM, Microsoft's best advice is to determine your organization's security needs and research any regulatory requirements associated with procurement for your industry. The result is a balance between scenarios used, assurance level, cost, convenience, and availability.

TPM in Windows

The security features of Windows combined with the benefits of a TPM offer practical security and privacy benefits. The following sections start with major TPM-related security features in Windows and go on to describe how key technologies use the TPM to enable or increase security.

Platform Crypto Provider

Windows includes a cryptography framework called *Cryptographic API: Next Generation* (CNG), the basic approach of which is to implement cryptographic algorithms in different ways but with a common application programming interface (API). Applications that use cryptography can use the common API without knowing the details of how an algorithm is implemented much less the algorithm itself.

Although CNG sounds like a mundane starting point, it illustrates some of the advantages that a TPM provides. Underneath the CNG interface, Windows or third parties supply a cryptographic provider (that is, an implementation of an algorithm) implemented as software libraries alone or in a combination of software and available system hardware or third-party hardware. If implemented through hardware, the cryptographic provider communicates with the hardware behind the software interface of CNG.

The Platform Crypto Provider, introduced in the Windows 8 operating system, exposes the following special TPM properties, which software-only CNG providers cannot offer or cannot offer as effectively:

- **Key protection.** The Platform Crypto Provider can create keys in the TPM with restrictions on their use. The operating system can load and use the keys in the TPM without copying the keys to system memory, where they are vulnerable to malware. The Platform Crypto Provider can also configure keys that a TPM protects so that they are not removable. If a TPM creates a key, the key is unique and resides only in that TPM. If the TPM imports a key, the Platform Crypto Provider can use the key in that TPM, but that TPM is not a source for making more copies of the key or enabling the use of copies elsewhere. In sharp contrast, software solutions that protect keys from copying are subject to reverse-engineering attacks, in which someone figures out how the solution stores keys or makes copies of keys while they are in memory during use.
- **Dictionary attack protection.** Keys that a TPM protects can require an authorization value such as a PIN. With dictionary attack protection, the TPM can prevent attacks that attempt a large number of guesses to determine the PIN. After too many guesses, the TPM simply returns an error saying no more guesses are allowed for a period of time. Software solutions might provide similar features, but they cannot provide the same level of protection, especially if the system restarts, the system clock changes, or files on the hard disk that count failed guesses are rolled back. In addition, with dictionary attack

protection, authorization values such as PINs can be shorter and easier to remember while still providing the same level of protection as more complex values when using software solutions.

These TPM features give Platform Crypto Provider distinct advantages over software-based solutions. A practical way to see these benefits in action is when using certificates on a Windows device. On platforms that include a TPM, Windows can use the Platform Crypto Provider to provide certificate storage. Certificate templates can specify that a TPM use the Platform Crypto Provider to protect the key associated with a certificate. In mixed environments, where some computers might not have a TPM, the certificate template could prefer the Platform Crypto Provider over the standard Windows software provider. If a certificate is configured as not able to be exported, the private key for the certificate is restricted and cannot be exported from the TPM. If the certificate requires a PIN, the PIN gains the TPM's dictionary attack protection automatically.

Virtual Smart Card

Smart cards are highly secure physical devices that typically store a single certificate and the corresponding private key. Users insert a smart card into a built-in or USB card reader and enter a PIN to unlock it. Windows can then access the card's certificate and use the private key for authentication or to unlock BitLocker protected data volumes. Smart cards are popular because they provide two-factor authentication that requires both something the user has (that is, the smart card) and something the user knows (such as the smart card PIN). Smart cards are difficult to use, however, because they require purchase and deployment of both smart cards and smart card readers.

In Windows, the Virtual Smart Card feature allows the TPM to mimic a permanently inserted smart card. The TPM becomes "something the user has" but still requires a PIN. Although physical smart cards limit the number of PIN attempts before locking the card and requiring a reset, a virtual smart card relies on the TPM's dictionary attack protection to prevent too many PIN guesses.

For TPM-based virtual smart cards, the TPM protects the use and storage of the certificate private key so that it cannot be copied when it is in use or stored and used elsewhere. Using a component that is part of the system rather than a separate physical smart card can reduce total cost of ownership because it eliminates "lost card" and "card left at home" scenarios while still delivering the benefits of smart card-based multifactor authentication. For users, virtual smart cards are simple to use, requiring only a PIN to unlock. Virtual smart cards support the same scenarios that physical smart cards support, including signing in to Windows or authenticating for resource access.

Windows Hello for Business

Windows Hello for Business provides authentication methods intended to replace passwords, which can be difficult to remember and easily compromised. In addition, user name - password solutions for authentication often reuse the same user name - password combinations on multiple devices and services; if those credentials are compromised, they are compromised in many places. Windows Hello for Business provisions devices one by one and combines the information provisioned on each device (i.e., the cryptographic key) with additional information to authenticate users. On a system that has a TPM, the TPM can protect the key. If a system does not have a TPM, software-based techniques protect the key. The additional information the user supplies can be a PIN value or, if the system has the necessary hardware, biometric information, such as fingerprint or facial recognition. To protect privacy, the biometric information is used only on the provisioned device to access the provisioned key: it is not shared across devices.

The adoption of new authentication technology requires that identity providers and organizations deploy and use that technology. Windows Hello for Business lets users authenticate with their existing Microsoft account, an Active Directory account, a Microsoft Azure Active Directory account, or even non-Microsoft Identity Provider Services or Relying Party Services that support [Fast ID Online V2.0 authentication](#).

Identity providers have flexibility in how they provision credentials on client devices. For example, an organization might provision only those devices that have a TPM so that the organization knows that a TPM

protects the credentials. The ability to distinguish a TPM from malware acting like a TPM requires the following TPM capabilities (see Figure 1):

- **Endorsement key.** The TPM manufacturer can create a special key in the TPM called an *endorsement key*. An endorsement key certificate, signed by the manufacturer, says that the endorsement key is present in a TPM that the manufacturer made. Solutions can use the certificate with the TPM containing the endorsement key to confirm a scenario really involves a TPM from a specific TPM manufacturer (instead of malware acting like a TPM).
- **Attestation identity key.** To protect privacy, most TPM scenarios do not directly use an actual endorsement key. Instead, they use attestation identity keys, and an identity certificate authority (CA) uses the endorsement key and its certificate to prove that one or more attestation identity keys actually exist in a real TPM. The identity CA issues attestation identity key certificates. More than one identity CA will generally see the same endorsement key certificate that can uniquely identify the TPM, but any number of attestation identity key certificates can be created to limit the information shared in other scenarios.

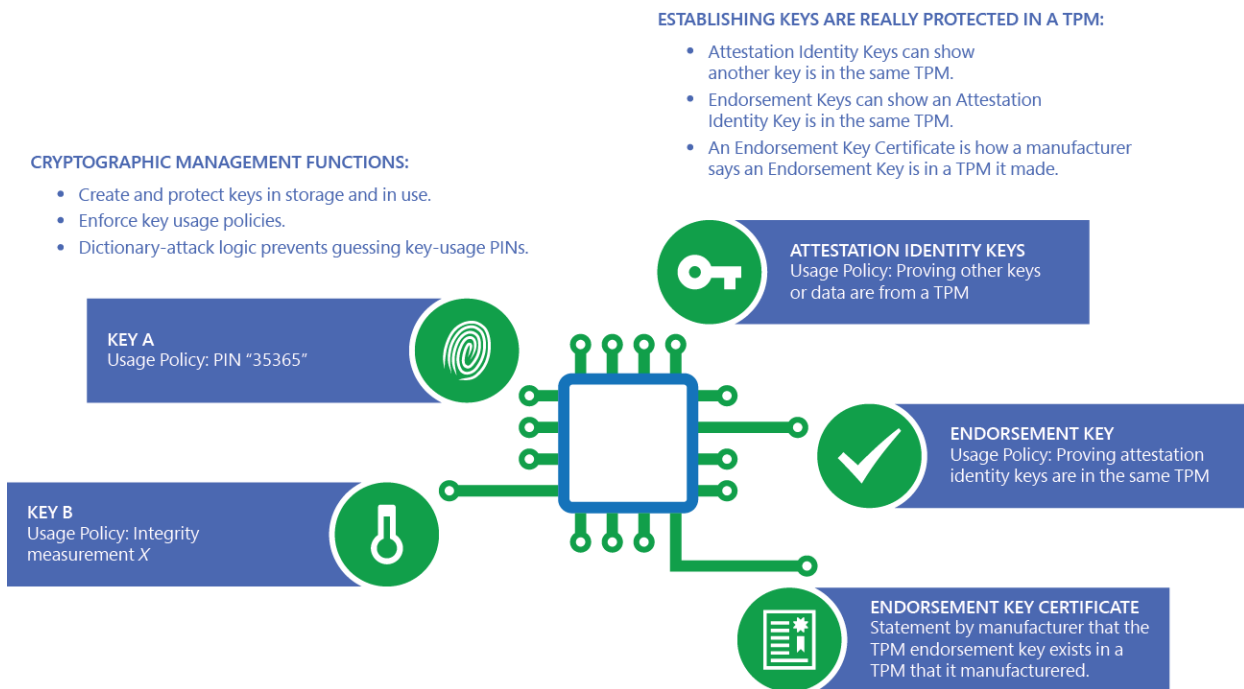


Figure 1: TPM Cryptographic Key Management

For Windows Hello for Business, Microsoft can fill the role of the identity CA. Microsoft services can issue an attestation identity key certificate for each device, user, and identify provider to ensure that privacy is protected and to help identity providers ensure that device TPM requirements are met before Windows Hello for Business credentials are provisioned.

BitLocker Drive Encryption

BitLocker provides full-volume encryption to protect data at rest. The most common device configuration splits the hard drive into several volumes. The operating system and user data reside on one volume that holds confidential information, and other volumes hold public information such as boot components, system information and recovery tools. (These other volumes are used infrequently enough that they do not need to be visible to users.) Without more protections in place, if the volume containing the operating system and user data is not encrypted, someone can boot another operating system and easily bypass the intended operating system's enforcement of file permissions to read any user data.

In the most common configuration, BitLocker encrypts the operating system volume so that if the computer or hard disk is lost or stolen when powered off, the data on the volume remains confidential. When the computer is turned on, starts normally, and proceeds to the Windows logon prompt, the only path forward is for the user to log on with his or her credentials, allowing the operating system to enforce its normal file permissions. If

something about the boot process changes, however—for example, a different operating system is booted from a USB device—the operating system volume and user data cannot be read and are not accessible. The TPM and system firmware collaborate to record measurements of how the system started, including loaded software and configuration details such as whether boot occurred from the hard drive or a USB device. BitLocker relies on the TPM to allow the use of a key only when startup occurs in an expected way. The system firmware and TPM are carefully designed to work together to provide the following capabilities:

- **Hardware root of trust for measurement.** A TPM allows software to send it commands that record measurements of software or configuration information. This information can be calculated using a hash algorithm that essentially transforms a lot of data into a small, statistically unique hash value. The system firmware has a component called the Core Root of Trust for Measurement (CRTM) that is implicitly trusted. The CRTM unconditionally hashes the next software component and records the measurement value by sending a command to the TPM. Successive components, whether system firmware or operating system loaders, continue the process by measuring any software components they load before running them. Because each component's measurement is sent to the TPM before it runs, a component cannot erase its measurement from the TPM. (However, measurements are erased when the system is restarted.) The result is that at each step of the system startup process, the TPM holds measurements of boot software and configuration information. Any changes in boot software or configuration yield different TPM measurements at that step and later steps. Because the system firmware unconditionally starts the measurement chain, it provides a hardware-based root of trust for the TPM measurements. At some point in the startup process, the value of recording all loaded software and configuration information diminishes and the chain of measurements stops. The TPM allows for the creation of keys that can be used only when the platform configuration registers that hold the measurements have specific values.
- **Key used only when boot measurements are accurate.** BitLocker creates a key in the TPM that can be used only when the boot measurements match an expected value. The expected value is calculated for the step in the startup process when Windows Boot Manager runs from the operating system volume on the system hard drive. Windows Boot Manager, which is stored unencrypted on the boot volume, needs to use the TPM key so that it can decrypt data read into memory from the operating system volume and startup can proceed using the encrypted operating system volume. If a different operating system is booted or the configuration is changed, the measurement values in the TPM will be different, the TPM will not let Windows Boot Manager use the key, and the startup process cannot proceed normally because the data on the operating system cannot be decrypted. If someone tries to boot the system with a different operating system or a different device, the software or configuration measurements in the TPM will be wrong and the TPM will not allow use of the key needed to decrypt the operating system volume. As a failsafe, if measurement values change unexpectedly, the user can always use the BitLocker recovery key to access volume data. Organizations can configure BitLocker to store the recovery key in Active Directory Domain Services (AD DS).

Device hardware characteristics are important to BitLocker and its ability to protect data. One consideration is whether the device provides attack vectors when the system is at the logon screen. For example, if the Windows device has a port that allows direct memory access so that someone can plug in hardware and read memory, an attacker can read the operating system volume's decryption key from memory while at the Windows logon screen. To mitigate this risk, organizations can configure BitLocker so that the TPM key requires both the correct software measurements and an authorization value. The system startup process stops at Windows Boot Manager, and the user is prompted to enter the authorization value for the TPM key or insert a USB device with the value. This process stops BitLocker from automatically loading the key into memory where it might be vulnerable, but has a less desirable user experience.

Newer hardware and Windows work better together to disable direct memory access through ports and reduce attack vectors. The result is that organizations can deploy more systems without requiring users to enter additional authorization information during the startup process. The right hardware allows BitLocker to be used with the "TPM-only" configuration giving users a single sign-on experience without having to enter a PIN or USB key during boot.

Device Encryption

Device Encryption is the consumer version of BitLocker, and it uses the same underlying technology. How it works is if a customer logs on with a Microsoft account and the system meets Modern Standby hardware requirements, BitLocker Drive Encryption is enabled automatically in Windows. The recovery key is backed up in the Microsoft cloud and is accessible to the consumer through his or her Microsoft account. The Modern Standby hardware requirements inform Windows that the hardware is appropriate for deploying Device Encryption and allows use of the "TPM-only" configuration for a simple consumer experience. In addition, Modern Standby hardware is designed to reduce the likelihood that measurement values change and prompt the customer for the recovery key.

For software measurements, Device Encryption relies on measurements of the authority providing software components (based on code signing from manufacturers such as OEMs or Microsoft) instead of the precise hashes of the software components themselves. This permits servicing of components without changing the resulting measurement values. For configuration measurements, the values used are based on the boot security policy instead of the numerous other configuration settings recorded during startup. These values also change less frequently. The result is that Device Encryption is enabled on appropriate hardware in a user-friendly way while also protecting data.

Measured Boot

Windows 8 introduced Measured Boot as a way for the operating system to record the chain of measurements of software components and configuration information in the TPM through the initialization of the Windows operating system. In previous Windows versions, the measurement chain stopped at the Windows Boot Manager component itself, and the measurements in the TPM were not helpful for understanding the starting state of Windows.

The Windows boot process happens in stages and often involves third-party drivers to communicate with vendor-specific hardware or implement antimalware solutions. For software, Measured Boot records measurements of the Windows kernel, Early-Launch Anti-Malware drivers, and boot drivers in the TPM. For configuration settings, Measured Boot records security-relevant information such as signature data that antimalware drivers use and configuration data about Windows security features (e.g., whether BitLocker is on or off).

Measured Boot ensures that TPM measurements fully reflect the starting state of Windows software and configuration settings. If security settings and other protections are set up correctly, they can be trusted to maintain the security of the running operating system thereafter. Other scenarios can use the operating system's starting state to determine whether the running operating system should be trusted.

TPM measurements are designed to avoid recording any privacy-sensitive information as a measurement. As an additional privacy protection, Measured Boot stops the measurement chain at the initial starting state of Windows. Therefore, the set of measurements does not include details about which applications are in use or how Windows is being used. Measurement information can be shared with external entities to show that the device is enforcing adequate security policies and did not start with malware.

The TPM provides the following way for scenarios to use the measurements recorded in the TPM during boot:

- **Remote Attestation.** Using an attestation identity key, the TPM can generate and cryptographically sign a statement (*or quote*) of the current measurements in the TPM. Windows can create unique attestation identity keys for various scenarios to prevent separate evaluators from collaborating to track the same device. Additional information in the quote is cryptographically scrambled to limit information sharing and better protect privacy. By sending the quote to a remote entity, a device can attest which software and configuration settings were used to boot the device and initialize the operating system. An attestation identity key certificate can provide further assurance that the quote is coming from a real TPM. Remote attestation is the process of recording measurements in the TPM, generating a quote, and sending the quote information to

another system that evaluates the measurements to establish trust in a device. Figure 2 illustrates this process.

When new security features are added to Windows, Measured Boot adds security-relevant configuration information to the measurements recorded in the TPM. Measured Boot enables remote attestation scenarios that reflect the system firmware and the Windows initialization state.

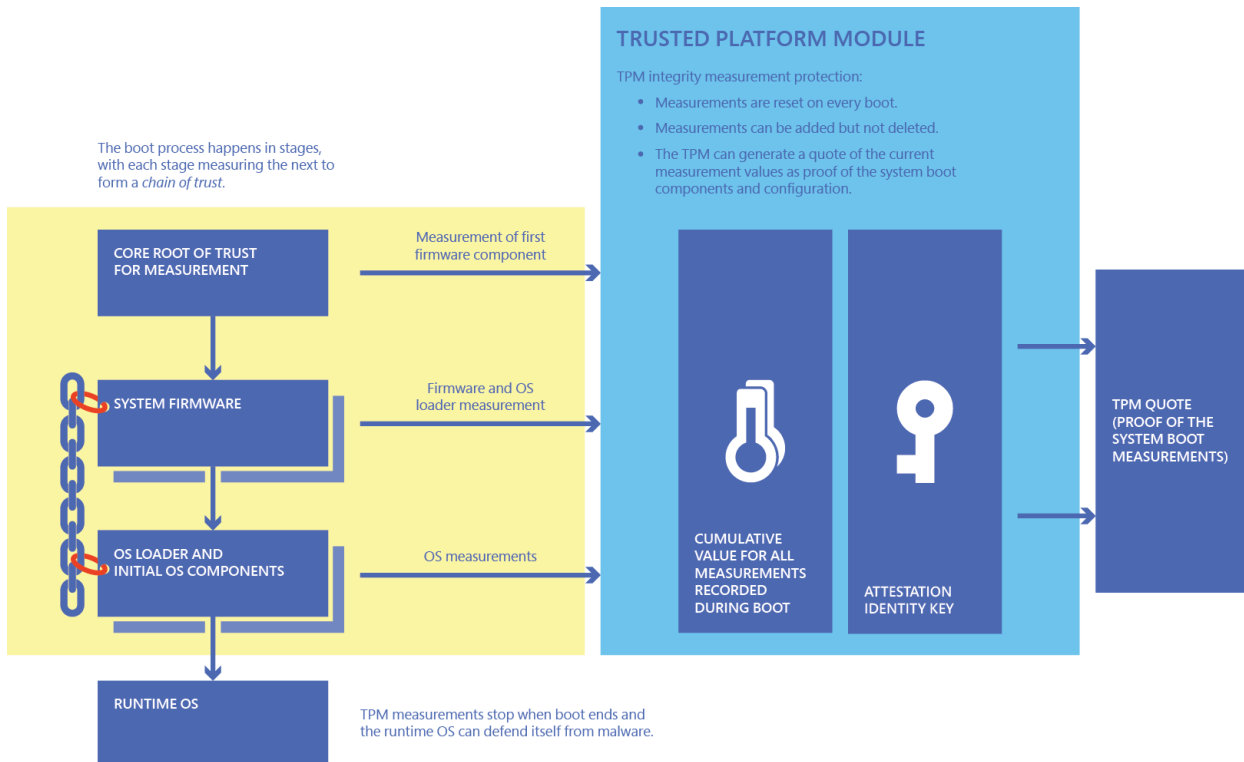


Figure 2: Process used to create evidence of boot software and configuration using a TPM

Health Attestation

Some Windows improvements help security solutions implement remote attestation scenarios. Microsoft provides a Health Attestation service, which can create attestation identity key certificates for TPMs from different manufacturers as well as parse measured boot information to extract simple security assertions, such as whether BitLocker is on or off. The simple security assertions can be used to evaluate device health.

Mobile device management (MDM) solutions can receive simple security assertions from the Microsoft Health Attestation service for a client without having to deal with the complexity of the quote or the detailed TPM measurements. MDM solutions can act on the security information by quarantining unhealthy devices or blocking access to cloud services such as Microsoft Office 365.

Credential Guard

Credential Guard is a new feature in Windows that helps protect Windows credentials in organizations that have deployed AD DS. Historically, a user's credentials (e.g., logon password) were hashed to generate an authorization token. The user employed the token to access resources that he or she was permitted to use. One weakness of the token model is that malware that had access to the operating system kernel could look through the computer's memory and harvest all the access tokens currently in use. The attacker could then use harvested tokens to log on to other machines and collect more credentials. This kind of attack is called a "pass the hash" attack, a malware technique that infects one machine to infect many machines across an organization.

Similar to the way Microsoft Hyper-V keeps virtual machines (VMs) separate from one another, Credential Guard uses virtualization to isolate the process that hashes credentials in a memory area that the operating system kernel cannot access. This isolated memory area is initialized and protected during the boot process so that components in the larger operating system environment cannot tamper with it. Credential Guard uses the

TPM to protect its keys with TPM measurements, so they are accessible only during the boot process step when the separate region is initialized; they are not available for the normal operating system kernel. The local security authority code in the Windows kernel interacts with the isolated memory area by passing in credentials and receiving single-use authorization tokens in return.

The resulting solution provides defense in depth, because even if malware runs in the operating system kernel, it cannot access the secrets inside the isolated memory area that actually generates authorization tokens. The solution does not solve the problem of key loggers because the passwords such loggers capture actually pass through the normal Windows kernel, but when combined with other solutions, such as smart cards for authentication, Credential Guard greatly enhances the protection of credentials in Windows.

Conclusion

The TPM adds hardware-based security benefits to Windows. When installed on hardware that includes a TPM, Windows delivers remarkably improved security benefits. The following table summarizes the key benefits of the TPM's major features.

FEATURE	BENEFITS WHEN USED ON A SYSTEM WITH A TPM
Platform Crypto Provider	<ul style="list-style-type: none">• If the machine is compromised, the private key associated with the certificate cannot be copied off the device.• The TPM's dictionary attack mechanism protects PIN values to use a certificate.
Virtual Smart Card	<ul style="list-style-type: none">• Achieve security similar to that of physical smart cards without deploying physical smart cards or card readers.
Windows Hello for Business	<ul style="list-style-type: none">• Credentials provisioned on a device cannot be copied elsewhere.• Confirm a device's TPM before credentials are provisioned.
BitLocker Drive Encryption	<ul style="list-style-type: none">• Multiple options are available for enterprises to protect data at rest while balancing security requirements with different device hardware.
Device Encryption	<ul style="list-style-type: none">• With a Microsoft account and the right hardware, consumers' devices seamlessly benefit from data-at-rest protection.
Measured Boot	<ul style="list-style-type: none">• A hardware root of trust contains boot measurements that help detect malware during remote attestation.
Health Attestation	<ul style="list-style-type: none">• MDM solutions can easily perform remote attestation and evaluate client health before granting access to resources or cloud services such as Office 365.

FEATURE	BENEFITS WHEN USED ON A SYSTEM WITH A TPM
Credential Guard	<ul style="list-style-type: none">• Defense in depth increases so that even if malware has administrative rights on one machine, it is significantly more difficult to compromise additional machines in an organization.

Although some of the aforementioned features have additional hardware requirements (e.g., virtualization support), the TPM is a cornerstone of Windows security. Microsoft and other industry stakeholders continue to improve the global standards associated with TPM and find more and more applications that use it to provide tangible benefits to customers. Microsoft has included support for most TPM features in its version of Windows for the Internet of Things (IoT) called [Windows IoT Core](#). IoT devices that might be deployed in insecure physical locations and connected to cloud services like [Azure IoT Hub](#) for management can use the TPM in innovative ways to address their emerging security requirements.

TPM Group Policy settings

7/1/2022 • 8 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016 and above

This topic describes the Trusted Platform Module (TPM) Services that can be controlled centrally by using Group Policy settings.

The Group Policy settings for TPM services are located at:

Computer Configuration\Administrative Templates\System\Trusted Platform Module Services

The following Group Policy settings were introduced in Windows.

Configure the level of TPM owner authorization information available to the operating system

IMPORTANT

Beginning with Windows 10 version 1703, the default value is 5. This value is implemented during provisioning so that another Windows component can either delete it or take ownership of it, depending on the system configuration. For TPM 2.0, a value of 5 means keep the lockout authorization. For TPM 1.2, it means discard the Full TPM owner authorization and retain only the Delegated authorization.

This policy setting configured which TPM authorization values are stored in the registry of the local computer. Certain authorization values are required in order to allow Windows to perform certain actions.

TPM 1.2 VALUE	TPM 2.0 VALUE	PURPOSE	KEPT AT LEVEL 0?	KEPT AT LEVEL 2?	KEPT AT LEVEL 4?
OwnerAuthAdmin	StorageOwnerAuth	Create SRK	No	Yes	Yes
OwnerAuthEndorsement	EndorsementAuth	Create or use EK (1.2 only: Create AIK)	No	Yes	Yes
OwnerAuthFull	LockoutAuth	Reset/change Dictionary Attack Protection	No	No	Yes

There are three TPM owner authentication settings that are managed by the Windows operating system. You can choose a value of **Full**, **Delegate**, or **None**.

- **Full** This setting stores the full TPM owner authorization, the TPM administrative delegation blob, and the TPM user delegation blob in the local registry. With this setting, you can use the TPM without requiring remote or external storage of the TPM owner authorization value. This setting is appropriate for

scenarios that do not require you to reset the TPM anti-hammering logic or change the TPM owner authorization value. Some TPM-based applications may require that this setting is changed before features that depend on the TPM anti-hammering logic can be used. Full owner authorization in TPM 1.2 is similar to lockout authorization in TPM 2.0. Owner authorization has a different meaning for TPM 2.0.

- **Delegated** This setting stores only the TPM administrative delegation blob and the TPM user delegation blob in the local registry. This setting is appropriate for use with TPM-based applications that depend on the TPM antihammering logic. This is the default setting in Windows prior to version 1703.
- **None** This setting provides compatibility with previous operating systems and applications. You can also use it for scenarios when TPM owner authorization cannot be stored locally. Using this setting might cause issues with some TPM-based applications.

NOTE

If the operating system managed TPM authentication setting is changed from **Full** to **Delegated**, the full TPM owner authorization value will be regenerated, and any copies of the previously set TPM owner authorization value will be invalid.

Registry information

Registry key: HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\TPM

DWORD: OSManagedAuthLevel

The following table shows the TPM owner authorization values in the registry.

VALUE DATA	SETTING
0	None
2	Delegated
4	Full

If you enable this policy setting, the Windows operating system will store the TPM owner authorization in the registry of the local computer according to the TPM authentication setting you choose.

On Windows 10 prior to version 1607, if you disable or do not configure this policy setting, and the **Turn on TPM backup to Active Directory Domain Services** policy setting is also disabled or not configured, the default setting is to store the full TPM authorization value in the local registry. If this policy is disabled or not configured, and the **Turn on TPM backup to Active Directory Domain Services** policy setting is enabled, only the administrative delegation and the user delegation blobs are stored in the local registry.

Standard User Lockout Duration

This policy setting allows you to manage the duration in minutes for counting standard user authorization failures for Trusted Platform Module (TPM) commands requiring authorization. An authorization failure occurs each time a standard user sends a command to the TPM and receives an error response that indicates an authorization failure occurred. Authorization failures that are older than the duration you set are ignored. If the number of TPM commands with an authorization failure within the lockout duration equals a threshold, a standard user is prevented from sending commands that require authorization to the TPM.

The TPM is designed to protect itself against password guessing attacks by entering a hardware lockout mode when it receives too many commands with an incorrect authorization value. When the TPM enters a lockout mode, it is global for all users (including administrators) and for Windows features such as BitLocker Drive

Encryption.

This setting helps administrators prevent the TPM hardware from entering a lockout mode by slowing the speed at which standard users can send commands that require authorization to the TPM.

For each standard user, two thresholds apply. Exceeding either threshold prevents the user from sending a command that requires authorization to the TPM. Use the following policy settings to set the lockout duration:

- **Standard User Individual Lockout Threshold** This value is the maximum number of authorization failures that each standard user can have before the user is not allowed to send commands that require authorization to the TPM.
- **Standard User Total Lockout Threshold** This value is the maximum total number of authorization failures that all standard users can have before all standard users are not allowed to send commands that require authorization to the TPM.

An administrator with the TPM owner password can fully reset the TPM's hardware lockout logic by using the Windows Defender Security Center. Each time an administrator resets the TPM's hardware lockout logic, all prior standard user TPM authorization failures are ignored. This allows standard users to immediately use the TPM normally.

If you do not configure this policy setting, a default value of 480 minutes (8 hours) is used.

Standard User Individual Lockout Threshold

This policy setting allows you to manage the maximum number of authorization failures for each standard user for the Trusted Platform Module (TPM). This value is the maximum number of authorization failures that each standard user can have before the user is not allowed to send commands that require authorization to the TPM. If the number of authorization failures for the user within the duration that is set for the **Standard User Lockout Duration** policy setting equals this value, the standard user is prevented from sending commands that require authorization to the Trusted Platform Module (TPM).

This setting helps administrators prevent the TPM hardware from entering a lockout mode by slowing the speed at which standard users can send commands that require authorization to the TPM.

An authorization failure occurs each time a standard user sends a command to the TPM and receives an error response indicating an authorization failure occurred. Authorization failures older than the duration are ignored.

An administrator with the TPM owner password can fully reset the TPM's hardware lockout logic by using the Windows Defender Security Center. Each time an administrator resets the TPM's hardware lockout logic, all prior standard user TPM authorization failures are ignored. This allows standard users to immediately use the TPM normally.

If you do not configure this policy setting, a default value of 4 is used. A value of zero means that the operating system will not allow standard users to send commands to the TPM, which might cause an authorization failure.

Standard User Total Lockout Threshold

This policy setting allows you to manage the maximum number of authorization failures for all standard users for the Trusted Platform Module (TPM). If the total number of authorization failures for all standard users within the duration that is set for the **Standard User Lockout Duration** policy equals this value, all standard users are prevented from sending commands that require authorization to the Trusted Platform Module (TPM).

This setting helps administrators prevent the TPM hardware from entering a lockout mode because it slows the speed standard users can send commands requiring authorization to the TPM.

An authorization failure occurs each time a standard user sends a command to the TPM and receives an error

response indicating an authorization failure occurred. Authorization failures older than the duration are ignored.

An administrator with the TPM owner password can fully reset the TPM's hardware lockout logic by using the Windows Defender Security Center. Each time an administrator resets the TPM's hardware lockout logic, all prior standard user TPM authorization failures are ignored. This allows standard users to immediately use the TPM normally.

If you do not configure this policy setting, a default value of 9 is used. A value of zero means that the operating system will not allow standard users to send commands to the TPM, which might cause an authorization failure.

Configure the system to use legacy Dictionary Attack Prevention Parameters setting for TPM 2.0

Introduced in Windows 10, version 1703, this policy setting configures the TPM to use the Dictionary Attack Prevention Parameters (lockout threshold and recovery time) to the values that were used for Windows 10 Version 1607 and below.

IMPORTANT

Setting this policy will take effect only if:

- The TPM was originally prepared using a version of Windows after Windows 10 Version 1607
- The system has a TPM 2.0.

NOTE

Enabling this policy will only take effect after the TPM maintenance task runs (which typically happens after a system restart). Once this policy has been enabled on a system and has taken effect (after a system restart), disabling it will have no impact and the system's TPM will remain configured using the legacy Dictionary Attack Prevention parameters, regardless of the value of this group policy. The only ways for the disabled setting of this policy to take effect on a system where it was once enabled are to either:

- Disable it from group policy
- Clear the TPM on the system

TPM Group Policy settings in the Windows Security app

You can change what users see about TPM in the Windows Security app. The Group Policy settings for the TPM area in the Windows Security app are located at:

Computer Configuration\Administrative Templates\Windows Components\Windows Security\Device security

Disable the Clear TPM button

If you don't want users to be able to click the **Clear TPM** button in the Windows Security app, you can disable it with this Group Policy setting. Select **Enabled** to make the **Clear TPM** button unavailable for use.

Hide the TPM Firmware Update recommendation

If you don't want users to see the recommendation to update TPM firmware, you can disable it with this setting. Select **Enabled** to prevent users from seeing a recommendation to update their TPM firmware when a vulnerable firmware is detected.

Related topics

- [Trusted Platform Module](#)
- [TPM Cmdlets in Windows PowerShell](#)
- [Prepare your organization for BitLocker: Planning and Policies - TPM configurations](#)

Back up the TPM recovery information to AD DS

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016 and above

Does not apply to

- Windows 10, version 1607 or later

With Windows 10, versions 1511 and 1507, or Windows 11, you can back up a computer's Trusted Platform Module (TPM) information to Active Directory Domain Services (AD DS). By doing this, you can use AD DS to administer the TPM from a remote computer. The procedure is the same as it was for Windows 8.1. For more information, see [Backup the TPM Recovery Information to AD DS](#).

Related topics

- [Trusted Platform Module](#) (list of topics)
- [TPM Group Policy settings](#)

Troubleshoot the TPM

7/1/2022 • 7 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016 and above

This article provides information for the IT professional to troubleshoot the Trusted Platform Module (TPM):

- [Troubleshoot TPM initialization](#)
- [Clear all the keys from the TPM](#)

With TPM 1.2 and Windows 10, version 1507 or 1511, or Windows 11, you can also take the following actions:

- [Turn on or turn off the TPM](#)

For information about the TPM cmdlets, see [TPM Cmdlets in Windows PowerShell](#).

About TPM initialization and ownership

Starting with Windows 10 and Windows 11, the operating system automatically initializes and takes ownership of the TPM. This is a change from previous operating systems, where you would initialize the TPM and create an owner password.

Troubleshoot TPM initialization

If you find that Windows is not able to initialize the TPM automatically, review the following information:

- You can try clearing the TPM to the factory default values and allowing Windows to re-initialize it. For important precautions for this process, and instructions for completing it, see [Clear all the keys from the TPM](#), later in this article.
- If the TPM is a TPM 2.0 and is not detected by Windows, verify that your computer hardware contains a Unified Extensible Firmware Interface (UEFI) that is Trusted Computing Group-compliant. Also, ensure that in the UEFI settings, the TPM has not been disabled or hidden from the operating system.
- If you have TPM 1.2 with Windows 10, version 1507 or 1511, or Windows 11, the TPM might be turned off, and need to be turned back on, as described in [Turn on the TPM](#). When it is turned back on, Windows will re-initialize it.
- If you are attempting to set up BitLocker with the TPM, check which TPM driver is installed on the computer. We recommend always using one of the TPM drivers that is provided by Microsoft and is protected with BitLocker. If a non-Microsoft TPM driver is installed, it may prevent the default TPM driver from loading and cause BitLocker to report that a TPM is not present on the computer. If you have a non-Microsoft driver installed, remove it and then allow the operating system to initialize the TPM.

Troubleshoot network connection issues for Windows 10, versions 1507 and 1511, or Windows 11

If you have Windows 10, version 1507 or 1511, or Windows 11, the initialization of the TPM cannot complete when your computer has network connection issues and both of the following conditions exist:

- An administrator has configured your computer to require that TPM recovery information be saved in

Active Directory Domain Services (AD DS). This requirement can be configured through Group Policy.

- A domain controller cannot be reached. This can occur on a computer that is currently disconnected from the network, separated from the domain by a firewall, or experiencing a network component failure (such as an unplugged cable or a faulty network adapter).

If these issues occur, an error message appears, and you cannot complete the initialization process. To avoid this issue, allow Windows to initialize the TPM while you are connected to the corporate network and you can contact a domain controller.

Troubleshoot systems with multiple TPMs

Some systems may have multiple TPMs and the active TPM may be toggled in UEFI. Windows does not support this behavior. If you switch TPMs, Windows might not properly detect or interact with the new TPM. If you plan to switch TPMs you should toggle to the new TPM, clear it, and reinstall Windows. For more information, see [Clear all the keys from the TPM](#), later in this article.

For example, toggling TPMs will cause BitLocker to enter recovery mode. We strongly recommend that, on systems with two TPMs, one TPM is selected to be used and the selection is not changed.

Clear all the keys from the TPM

You can use the Windows Defender Security Center app to clear the TPM as a troubleshooting step, or as a final preparation before a clean installation of a new operating system. Preparing for a clean installation in this way helps ensure that the new operating system can fully deploy any TPM-based functionality that it includes, such as attestation. However, even if the TPM is not cleared before a new operating system is installed, most TPM functionality will probably work correctly.

Clearing the TPM resets it to an unowned state. After you clear the TPM, the Windows operating system will automatically re-initialize it and take ownership again.

WARNING

Clearing the TPM can result in data loss. For more information, see the next section, "Precautions to take before clearing the TPM."

Precautions to take before clearing the TPM

Clearing the TPM can result in data loss. To protect against such loss, review the following precautions:

- Clearing the TPM causes you to lose all created keys associated with the TPM, and data protected by those keys, such as a virtual smart card or a sign in PIN. Make sure that you have a backup and recovery method for any data that is protected or encrypted by the TPM.
- Do not clear the TPM on a device you do not own, such as a work or school PC, without being instructed to do so by your IT administrator.
- If you want to temporarily suspend TPM operations and you have TPM 1.2 with Windows 10, version 1507 or 1511, or Windows 11, you can turn off the TPM. For more information, see [Turn off the TPM](#), later in this article.
- Always use functionality in the operating system (such as TPM.msc) to clear the TPM. Do not clear the TPM directly from UEFI.
- Because your TPM security hardware is a physical part of your computer, before clearing the TPM, you might want to read the manuals or instructions that came with your computer, or search the manufacturer's website.

Membership in the local Administrators group, or equivalent, is the minimum required to complete this procedure.

To clear the TPM

1. Open the Windows Defender Security Center app.
2. Select **Device security**.
3. Select **Security processor details**.
4. Select **Security processor troubleshooting**.
5. Select **Clear TPM**.
6. You will be prompted to restart the computer. During the restart, you might be prompted by the UEFI to press a button to confirm that you wish to clear the TPM.
7. After the PC restarts, your TPM will be automatically prepared for use by Windows.

Turn on or turn off the TPM (available only with TPM 1.2 with Windows 10, version 1507 and higher)

Normally, the TPM is turned on as part of the TPM initialization process. You do not normally need to turn the TPM on or off. However, if necessary you can do so by using the TPM MMC.

Turn on the TPM

If you want to use the TPM after you have turned it off, you can use the following procedure to turn on the TPM.

To turn on the TPM (TPM 1.2 with Windows 10, version 1507 and higher)

1. Open the TPM MMC (tpm.msc).
2. In the **Action** pane, select **Turn TPM On** to display the **Turn on the TPM Security Hardware** page. Read the instructions on this page.
3. Select **Shutdown** (or **Restart**), and then follow the UEFI screen prompts.

After the computer restarts, but before you sign in to Windows, you will be prompted to accept the reconfiguration of the TPM. This ensures that the user has physical access to the computer and that malicious software is not attempting to make changes to the TPM.

Turn off the TPM

If you want to stop using the services that are provided by the TPM, you can use the TPM MMC to turn off the TPM.

To turn off the TPM (TPM 1.2 with Windows 10, version 1507 and higher)

1. Open the TPM MMC (tpm.msc).
2. In the **Action** pane, select **Turn TPM Off** to display the **Turn off the TPM security hardware** page.
3. In the **Turn off the TPM security hardware** dialog box, select a method to enter your owner password and turning off the TPM:
 - If you saved your TPM owner password on a removable storage device, insert it, and then select **I have the owner password file**. In the **Select backup file with the TPM owner password** dialog box, select **Browse** to locate the .tpm file that is saved on your removable storage device, select **Open**, and then select **Turn TPM Off**.
 - If you do not have the removable storage device with your saved TPM owner password, select **I**

want to enter the password. In the **Type your TPM owner password** dialog box, type your password (including hyphens), and then select **Turn TPM Off**.

- If you did not save your TPM owner password or no longer know it, select **I do not have the TPM owner password**, and follow the instructions that are provided in the dialog box and subsequent UEFI screens to turn off the TPM without entering the password.

Use the TPM cmdlets

You can manage the TPM using Windows PowerShell. For details, see [TPM Cmdlets in Windows PowerShell](#).

Related articles

- [Trusted Platform Module](#) (list of articles)

Understanding PCR banks on TPM 2.0 devices

7/1/2022 • 4 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016 and above

For steps on how to switch PCR banks on TPM 2.0 devices on your PC, you should contact your OEM or UEFI vendor. This topic provides background about what happens when you switch PCR banks on TPM 2.0 devices.

A Platform Configuration Register (PCR) is a memory location in the TPM that has some unique properties. The size of the value that can be stored in a PCR is determined by the size of a digest generated by an associated hashing algorithm. A SHA-1 PCR can store 20 bytes – the size of a SHA-1 digest. Multiple PCRs associated with the same hashing algorithm are referred to as a PCR bank.

To store a new value in a PCR, the existing value is extended with a new value as follows: $PCR[N] = \text{HASHalg}(PCR[N] || \text{ArgumentOfExtend})$

The existing value is concatenated with the argument of the TPM Extend operation. The resulting concatenation is then used as input to the associated hashing algorithm, which computes a digest of the input. This computed digest becomes the new value of the PCR.

The [TCG PC Client Platform TPM Profile Specification](#) defines the inclusion of at least one PCR bank with 24 registers. The only way to reset the first 16 PCRs is to reset the TPM itself. This restriction helps ensure that the value of those PCRs can only be modified via the TPM Extend operation.

Some TPM PCRs are used as checksums of log events. The log events are extended in the TPM as the events occur. Later, an auditor can validate the logs by computing the expected PCR values from the log and comparing them to the PCR values of the TPM. Since the first 16 TPM PCRs cannot be modified arbitrarily, a match between an expected PCR value in that range and the actual TPM PCR value provides assurance of an unmodified log.

How does Windows use PCRs?

To bind the use of a TPM based key to a certain state of the PC, the key can be sealed to an expected set of PCR values. For instance, PCRs 0 through 7 have a well-defined value after the boot process – when the OS is loaded. When the hardware, firmware, or boot loader of the machine changes, the change can be detected in the PCR values. Windows uses this capability to make certain cryptographic keys only available at certain times during the boot process. For instance, the BitLocker key can be used at a certain point in the boot, but not before or after.

It is important to note that this binding to PCR values also includes the hashing algorithm used for the PCR. For instance, a key can be bound to a specific value of the SHA-1 PCR[12], if using SHA-256 PCR banks, even with the same system configuration. Otherwise, the PCR values will not match.

What happens when PCR banks are switched?

When the PCR banks are switched, the algorithm used to compute the hashed values stored in the PCRs during extend operations is changed. Each hash algorithm will return a different cryptographic signature for the same inputs.

As a result, if the currently used PCR bank is switched all keys that have been bound to the previous PCR values will no longer work. For example, if you had a key bound to the SHA-1 value of PCR[12] and subsequently changed the PCR banks to SHA-256, the banks wouldn't match, and you would be unable to use that key. The BitLocker key is secured using the PCR banks and Windows will not be able to unseal it if the PCR banks are switched while BitLocker is enabled.

What can I do to switch PCRs when BitLocker is already active?

Before switching PCR banks you should suspend or disable BitLocker – or have your recovery key ready. For steps on how to switch PCR banks on your PC, you should contact your OEM or UEFI vendor.

How can I identify which PCR bank is being used?

A TPM can be configured to have multiple PCR banks active. When BIOS is performing measurements it will do so into all active PCR banks, depending on its capability to make these measurements. BIOS may choose to deactivate PCR banks that it does not support or "cap" PCR banks that it does not support by extending a separator. The following registry value identifies which PCR banks are active.

- Registry key: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\IntegrityServices
- DWORD: TPMActivePCRBanks
- Defines which PCR banks are currently active. (This value should be interpreted as a bitmap for which the bits are defined in the [TCG Algorithm Registry](#) Table 21 of Revision 1.27.)

Windows checks which PCR banks are active and supported by the BIOS. Windows also checks if the measured boot log supports measurements for all active PCR banks. Windows will prefer the use of the SHA-256 bank for measurements and will fall back to SHA1 PCR bank if one of the pre-conditions is not met.

You can identify which PCR bank is currently used by Windows by looking at the registry.

- Registry key: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\IntegrityServices
- DWORD: TPMDigestAlgID
- Algorithm ID of the PCR bank that Windows is currently using. (This value represents an algorithm identifier as defined in the [TCG Algorithm Registry](#) Table 3 of Revision 1.27.)

Windows only uses one PCR bank to continue boot measurements. All other active PCR banks will be extended with a separator to indicate that they are not used by Windows and measurements that appear to be from Windows should not be trusted.

Related topics

- [Trusted Platform Module](#) (list of topics)

TPM recommendations

7/1/2022 • 8 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016 and above

This topic provides recommendations for Trusted Platform Module (TPM) technology for Windows.

For a basic feature description of TPM, see the [Trusted Platform Module Technology Overview](#).

TPM design and implementation

Traditionally, TPMs are discrete chips soldered to a computer's motherboard. Such implementations allow the computer's original equipment manufacturer (OEM) to evaluate and certify the TPM separate from the rest of the system. Discrete TPM implementations are common. However, they can be problematic for integrated devices that are small or have low power consumption. Some newer TPM implementations integrate TPM functionality into the same chipset as other platform components while still providing logical separation similar to discrete TPM chips.

TPMs are passive: they receive commands and return responses. To realize the full benefit of a TPM, the OEM must carefully integrate system hardware and firmware with the TPM to send it commands and react to its responses. TPMs were originally designed to provide security and privacy benefits to a platform's owner and users, but newer versions can provide security and privacy benefits to the system hardware itself. Before it can be used for advanced scenarios, however, a TPM must be provisioned. Windows automatically provisions a TPM, but if the user is planning to reinstall the operating system, he or she may need to clear the TPM before reinstalling so that Windows can take full advantage of the TPM.

The Trusted Computing Group (TCG) is the nonprofit organization that publishes and maintains the TPM specification. The TCG exists to develop, define, and promote vendor-neutral, global industry standards. These standards support a hardware-based root of trust for interoperable trusted computing platforms. The TCG also publishes the TPM specification as the international standard ISO/IEC 11889, using the Publicly Available Specification Submission Process that the Joint Technical Committee 1 defines between the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC).

OEMs implement the TPM as a component in a trusted computing platform, such as a PC, tablet, or phone. Trusted computing platforms use the TPM to support privacy and security scenarios that software alone cannot achieve. For example, software alone cannot reliably report whether malware is present during the system startup process. The close integration between TPM and platform increases the transparency of the startup process and supports evaluating device health by enabling reliable measuring and reporting of the software that starts the device. Implementation of a TPM as part of a trusted computing platform provides a hardware root of trust—that is, it behaves in a trusted way. For example, if a key stored in a TPM has properties that disallow exporting the key, that key truly cannot leave the TPM.

The TCG designed the TPM as a low-cost, mass-market security solution that addresses the requirements of different customer segments. There are variations in the security properties of different TPM implementations just as there are variations in customer and regulatory requirements for different sectors. In public-sector procurement, for example, some governments have clearly defined security requirements for TPMs whereas others do not.

TPM 1.2 vs. 2.0 comparison

From an industry standard, Microsoft has been an industry leader in moving and standardizing on TPM 2.0, which has many key realized benefits across algorithms, crypto, hierarchy, root keys, authorization and NV RAM.

Why TPM 2.0?

TPM 2.0 products and systems have important security advantages over TPM 1.2, including:

- The TPM 1.2 spec only allows for the use of RSA and the SHA-1 hashing algorithm.
- For security reasons, some entities are moving away from SHA-1. Notably, NIST has required many federal agencies to move to SHA-256 as of 2014, and technology leaders, including Microsoft and Google have announced they will remove support for SHA-1 based signing or certificates in 2017.
- TPM 2.0 enables **greater crypto agility** by being more flexible with respect to cryptographic algorithms.
 - TPM 2.0 supports newer algorithms, which can improve drive signing and key generation performance. For the full list of supported algorithms, see the [TCG Algorithm Registry](#). Some TPMs don't support all algorithms.
 - For the list of algorithms that Windows supports in the platform cryptographic storage provider, see [CNG Cryptographic Algorithm Providers](#).
 - TPM 2.0 achieved ISO standardization ([ISO/IEC 11889:2015](#)).
 - Use of TPM 2.0 may help eliminate the need for OEMs to make exception to standard configurations for certain countries and regions.
- TPM 2.0 offers a more **consistent experience** across different implementations.
 - TPM 1.2 implementations vary in policy settings. This may result in support issues as lockout policies vary.
 - TPM 2.0 lockout policy is configured by Windows, ensuring a consistent dictionary attack protection guarantee.
- While TPM 1.2 parts are discrete silicon components, which are typically soldered on the motherboard, TPM 2.0 is available as a **discrete (dTPM)** silicon component in a single semiconductor package, an **integrated** component incorporated in one or more semiconductor packages - alongside other logic units in the same package(s), and as a **firmware (fTPM)** based component running in a trusted execution environment (TEE) on a general purpose SoC.

NOTE

TPM 2.0 is not supported in Legacy and CSM Modes of the BIOS. Devices with TPM 2.0 must have their BIOS mode configured as Native UEFI only. The Legacy and Compatibility Support Module (CSM) options must be disabled. For added security Enable the Secure Boot feature.

Installed Operating System on hardware in legacy mode will stop the OS from booting when the BIOS mode is changed to UEFI. Use the tool [MBR2GPT](#) before changing the BIOS mode which will prepare the OS and the disk to support UEFI.

Discrete, Integrated, or Firmware TPM?

There are three implementation options for TPMs:

- Discrete TPM chip as a separate component in its own semiconductor package

- Integrated TPM solution, using dedicated hardware integrated into one or more semiconductor packages alongside, but logically separate from, other components
- Firmware TPM solution, running the TPM in firmware in a Trusted Execution mode of a general purpose computation unit

Windows uses any compatible TPM in the same way. Microsoft does not take a position on which way a TPM should be implemented and there is a wide ecosystem of available TPM solutions, which should suit all needs.

Is there any importance for TPM for consumers?

For end consumers, TPM is behind the scenes but is still relevant. TPM is used for Windows Hello, Windows Hello for Business and in the future, will be a component of many other key security features in Windows. TPM secures the PIN, helps encrypt passwords, and builds on our overall Windows experience story for security as a critical pillar. Using Windows on a system with a TPM enables a deeper and broader level of security coverage.

TPM 2.0 Compliance for Windows

Windows for desktop editions (Home, Pro, Enterprise, and Education)

- Since July 28, 2016, all new device models, lines, or series (or if you're updating the hardware configuration of an existing model, line, or series with a major update, such as CPU, graphic cards) must implement and enable by default TPM 2.0 (details in section 3.7 of the [Minimum hardware requirements](#) page). The requirement to enable TPM 2.0 only applies to the manufacturing of new devices. For TPM recommendations for specific Windows features, see [TPM and Windows Features](#).

IoT Core

- TPM is optional on IoT Core.

Windows Server 2016

- TPM is optional for Windows Server SKUs unless the SKU meets the other qualification (AQ) criteria for the Host Guardian Services scenario in which case TPM 2.0 is required.

TPM and Windows Features

The following table defines which Windows features require TPM support.

WINDOWS FEATURES	TPM REQUIRED	SUPPORTS TPM 1.2	SUPPORTS TPM 2.0	DETAILS
Measured Boot	Yes	Yes	Yes	Measured Boot requires TPM 1.2 or 2.0 and UEFI Secure Boot. TPM 2.0 is recommended since it supports newer cryptographic algorithms. TPM 1.2 only supports the SHA-1 algorithm which is being deprecated.

WINDOWS FEATURES	TPM REQUIRED	SUPPORTS TPM 1.2	SUPPORTS TPM 2.0	DETAILS
BitLocker	No	Yes	Yes	TPM 1.2 or 2.0 are supported but TPM 2.0 is recommended. Automatic Device Encryption requires Modern Standby including TPM 2.0 support
Device Encryption	Yes	N/A	Yes	Device Encryption requires Modern Standby/Connected Standby certification, which requires TPM 2.0.
Windows Defender Application Control (Device Guard)	No	Yes	Yes	
Windows Defender System Guard (DRTM)	Yes	No	Yes	TPM 2.0 and UEFI firmware is required.
Credential Guard	No	Yes	Yes	Windows 10, version 1507 (End of Life as of May 2017) only supported TPM 2.0 for Credential Guard. Beginning with Windows 10, version 1511, TPM 1.2 and 2.0 are supported. Paired with Windows Defender System Guard, TPM 2.0 provides enhanced security for Credential Guard. Windows 11 requires TPM 2.0 by default to facilitate easier enablement of this enhanced security for customers.
Device Health Attestation	Yes	Yes	Yes	TPM 2.0 is recommended since it supports newer cryptographic algorithms. TPM 1.2 only supports the SHA-1 algorithm which is being deprecated.

WINDOWS FEATURES	TPM REQUIRED	SUPPORTS TPM 1.2	SUPPORTS TPM 2.0	DETAILS
Windows Hello/Windows Hello for Business	No	Yes	Yes	Azure AD join supports both versions of TPM, but requires TPM with keyed-hash message authentication code (HMAC) and Endorsement Key (EK) certificate for key attestation support. TPM 2.0 is recommended over TPM 1.2 for better performance and security. Windows Hello as a FIDO platform authenticator will take advantage of TPM 2.0 for key storage.
UEFI Secure Boot	No	Yes	Yes	
TPM Platform Crypto Provider Key Storage Provider	Yes	Yes	Yes	
Virtual Smart Card	Yes	Yes	Yes	
Certificate storage	No	Yes	Yes	TPM is only required when the certificate is stored in the TPM.
Autopilot	No	N/A	Yes	If you intend to deploy a scenario which requires TPM (such as white glove and self-deploying mode), then TPM 2.0 and UEFI firmware are required.
SecureBIO	Yes	No	Yes	TPM 2.0 and UEFI firmware is required.

OEM Status on TPM 2.0 system availability and certified parts

Government customers and enterprise customers in regulated industries may have acquisition standards that require use of common certified TPM parts. As a result, OEMs, who provide the devices, may be required to use only certified TPM components on their commercial class systems. For more information, contact your OEM or hardware vendor.

Related topics

- [Trusted Platform Module](#) (list of topics)

Windows Defender System Guard: How a hardware-based root of trust helps protect Windows 10

7/1/2022 • 5 minutes to read • [Edit Online](#)

To protect critical resources such as the Windows authentication stack, single sign-on tokens, the Windows Hello biometric stack, and the Virtual Trusted Platform Module, a system's firmware and hardware must be trustworthy.

Windows Defender System Guard reorganizes the existing Windows 10 system integrity features under one roof and sets up the next set of investments in Windows security. It's designed to make these security guarantees:

- Protect and maintain the integrity of the system as it starts up
- Validate that system integrity has truly been maintained through local and remote attestation

Maintaining the integrity of the system as it starts

Static Root of Trust for Measurement (SRTM)

With Windows 7, one of the means attackers would use to persist and evade detection was to install what is often referred to as a bootkit or rootkit on the system. This malicious software would start before Windows started, or during the boot process itself, enabling it to start with the highest level of privilege.

With Windows 10 running on modern hardware (that is, Windows 8-certified or greater) a hardware-based root of trust helps ensure that no unauthorized firmware or software (such as a bootkit) can start before the Windows bootloader. This hardware-based root of trust comes from the device's Secure Boot feature, which is part of the Unified Extensible Firmware Interface (UEFI). This technique of measuring the static early boot UEFI components is called the Static Root of Trust for Measurement (SRTM).

As there are thousands of PC vendors that produce many models with different UEFI BIOS versions, there becomes an incredibly large number of SRTM measurements upon bootup. Two techniques exist to establish trust here—either maintain a list of known 'bad' SRTM measurements (also known as a blocklist), or a list of known 'good' SRTM measurements (also known as an allowlist).

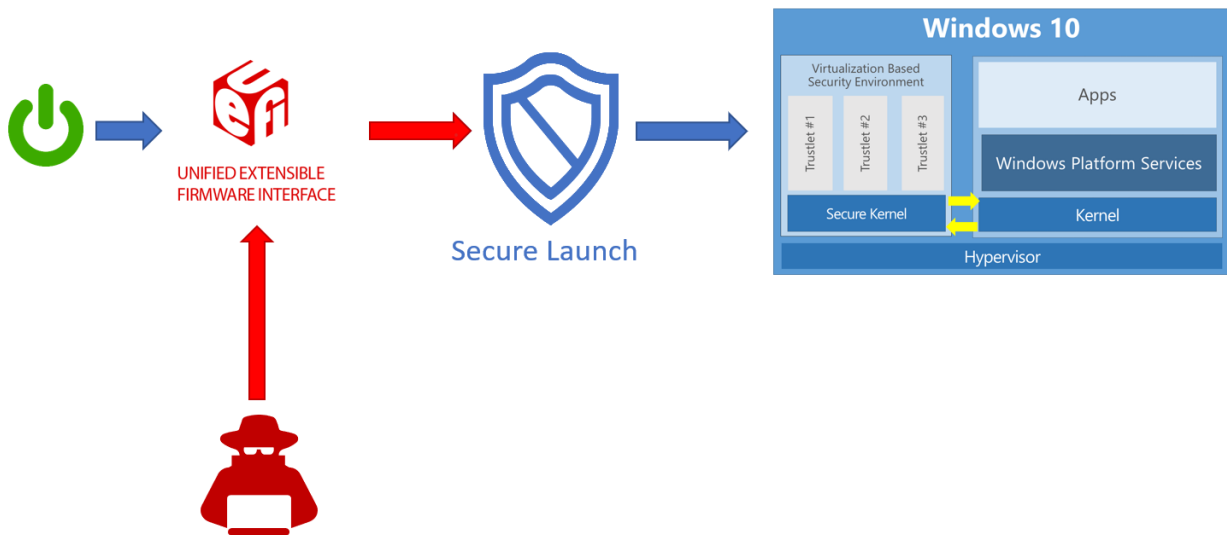
Each option has a drawback:

- A list of known 'bad' SRTM measurements allows a hacker to change just 1 bit in a component to create an entirely new SRTM hash that needs to be listed. This means that the SRTM flow is inherently brittle - a minor change can invalidate the entire chain of trust.
- A list of known 'good' SRTM measurements requires each new BIOS/PC combination measurement to be carefully added, which is slow. Also, a bug fix for UEFI code can take a long time to design, build, retest, validate, and redeploy.

Secure Launch—the Dynamic Root of Trust for Measurement (DRTM)

[Windows Defender System Guard Secure Launch](#), first introduced in Windows 10 version 1809, aims to alleviate these issues by leveraging a technology known as the Dynamic Root of Trust for Measurement (DRTM). DRTM lets the system freely boot into untrusted code initially, but shortly after launches the system into a trusted state by taking control of all CPUs and forcing them down a well-known and measured code path. This has the benefit of allowing untrusted early UEFI code to boot the system, but then being able to securely transition into a

trusted and measured state.



Secure Launch simplifies management of SRTM measurements because the launch code is now unrelated to a specific hardware configuration. This means the number of valid code measurements is small, and future updates can be deployed more widely and quickly.

System Management Mode (SMM) protection

System Management Mode (SMM) is a special-purpose CPU mode in x86 microcontrollers that handles power management, hardware configuration, thermal monitoring, and anything else the manufacturer deems useful. Whenever one of these system operations is requested, a non-maskable interrupt (SMI) is invoked at runtime, which executes SMM code installed by the BIOS. SMM code executes in the highest privilege level and is invisible to the OS, which makes it an attractive target for malicious activity. Even if System Guard Secure Launch is used to late launch, SMM code can potentially access hypervisor memory and change the hypervisor.

To defend against this, two techniques are used:

- Paging protection to prevent inappropriate access to code and data
- SMM hardware supervision and attestation

Paging protection can be implemented to lock certain code tables to be read-only to prevent tampering. This prevents access to any memory that hasn't been assigned.

A hardware-enforced processor feature known as a supervisor SMI handler can monitor the SMM and make sure it doesn't access any part of the address space that it isn't supposed to.

SMM protection is built on top of the Secure Launch technology and requires it to function. In the future, Windows 10 will also measure this SMI Handler's behavior and attest that no OS-owned memory has been tampered with.

Validating platform integrity after Windows is running (run time)

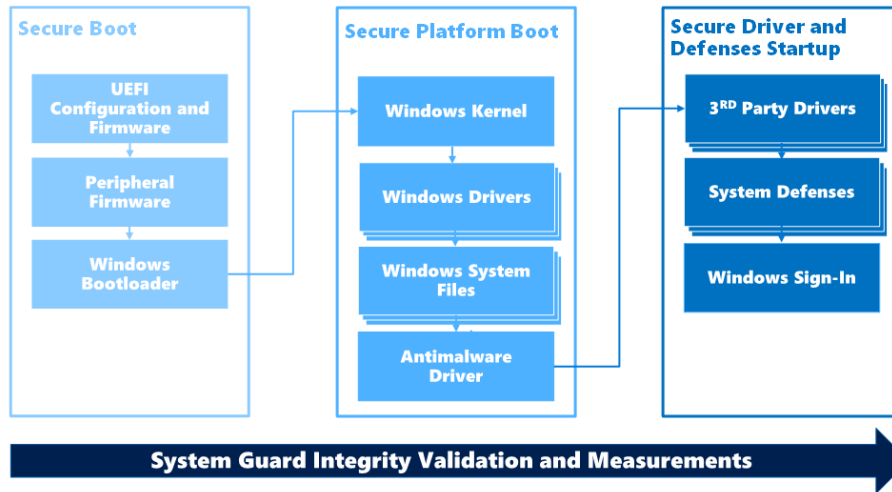
While Windows Defender System Guard provides advanced protection that will help protect and maintain the integrity of the platform during boot and at run time, the reality is that we must apply an "assume breach" mentality to even our most sophisticated security technologies. We can trust that the technologies are successfully doing their jobs, but we also need the ability to verify that they were successful in achieving their goals. For platform integrity, we can't just trust the platform, which potentially could be compromised, to self-attest to its security state. So Windows Defender System Guard includes a series of technologies that enable remote analysis of the device's integrity.

As Windows 10 boots, a series of integrity measurements are taken by Windows Defender System Guard using the device's Trusted Platform Module 2.0 (TPM 2.0). System Guard Secure Launch won't support earlier TPM

versions, such as TPM 1.2. This process and data are hardware-isolated away from Windows to help ensure that the measurement data isn't subject to the type of tampering that could happen if the platform was compromised. From here, the measurements can be used to determine the integrity of the device's firmware, hardware configuration state, and Windows boot-related components, just to name a few.

WINDOWS DEFENDER SYSTEM GUARD

BOOT TIME INTEGRITY PROTECTION



After the system boots, Windows Defender System Guard signs and seals these measurements using the TPM. Upon request, a management system like Intune or Microsoft Endpoint Configuration Manager can acquire them for remote analysis. If Windows Defender System Guard indicates that the device lacks integrity, the management system can take a series of actions, such as denying the device access to resources.

System Guard Secure Launch and SMM protection

7/1/2022 • 6 minutes to read • [Edit Online](#)

Applies to:

- Windows 11
- Windows 10

This topic explains how to configure [System Guard Secure Launch and System Management Mode \(SMM\) protection](#) to improve the startup security of Windows 10 and Windows 11 devices. The information below is presented from a client perspective.

How to enable System Guard Secure Launch

You can enable System Guard Secure Launch by using any of these options:

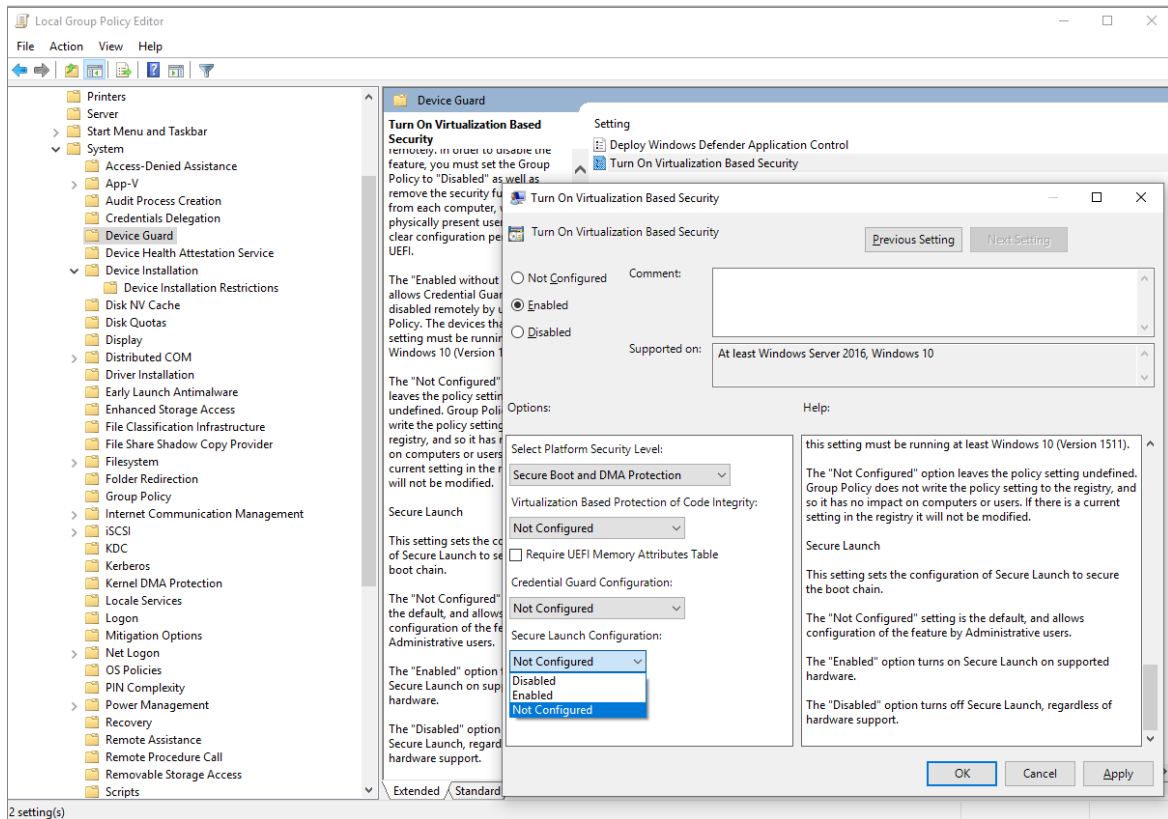
- [Mobile Device Management \(MDM\)](#)
- [Group Policy](#)
- [Windows Security app](#)
- [Registry](#)

Mobile Device Management

System Guard Secure Launch can be configured for Mobile Device Management (MDM) by using DeviceGuard policies in the Policy CSP, [DeviceGuard/ConfigureSystemGuardLaunch](#).

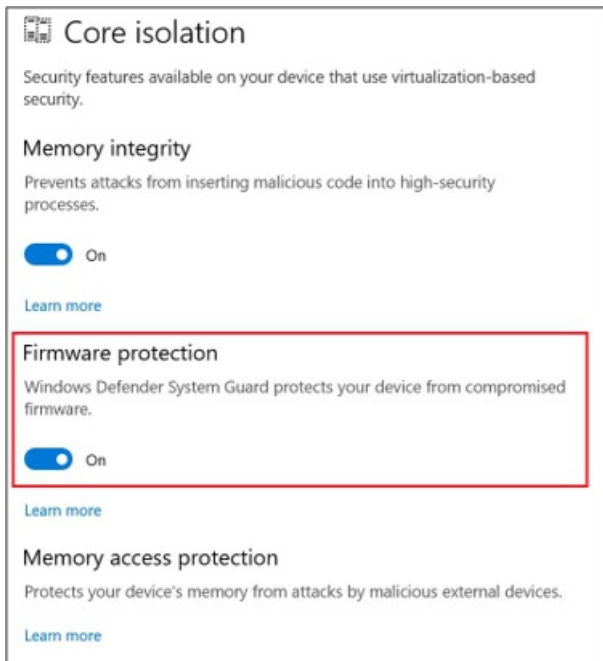
Group Policy

1. Click **Start** > type and then click **Edit group policy**.
2. Click **Computer Configuration** > **Administrative Templates** > **System** > **Device Guard** > **Turn On Virtualization Based Security** > **Secure Launch Configuration**.



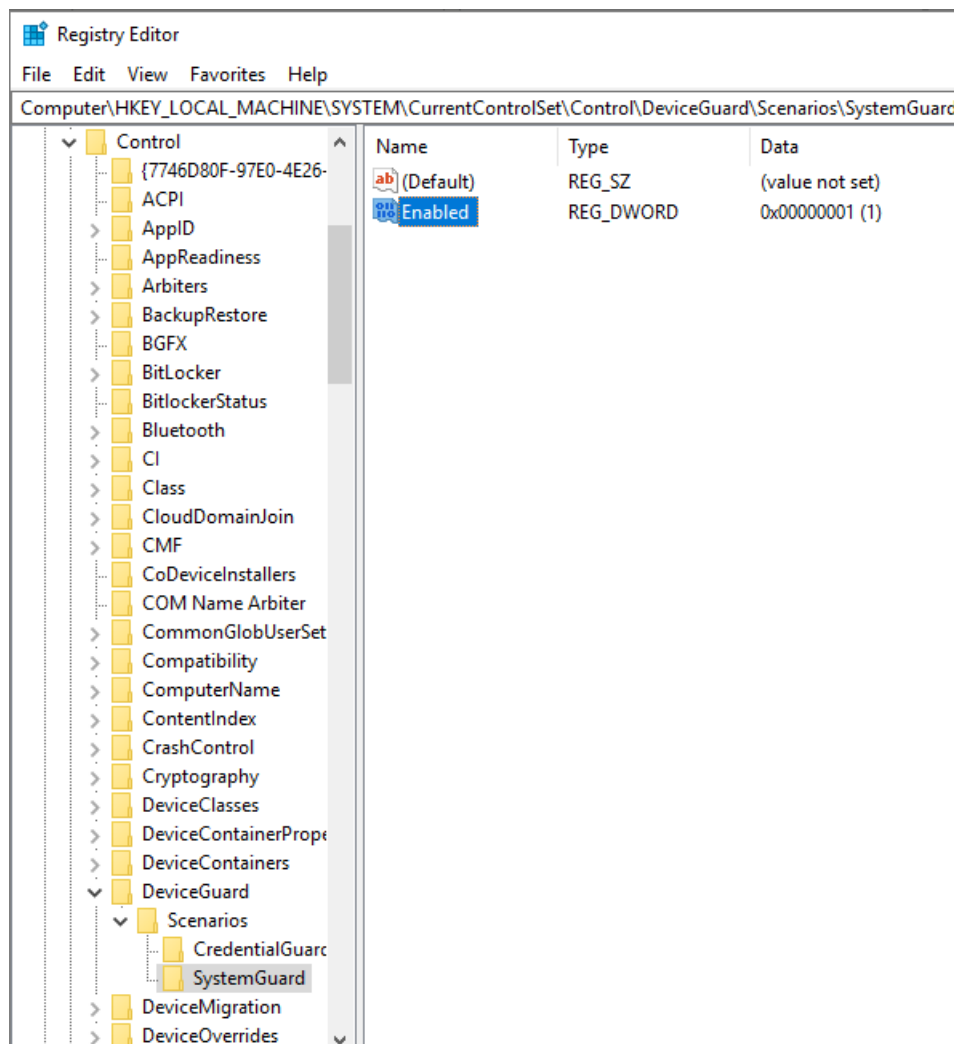
Windows Security app

Click Start > Settings > Update & Security > Windows Security > Open Windows Security > Device security > Core isolation > Firmware protection.



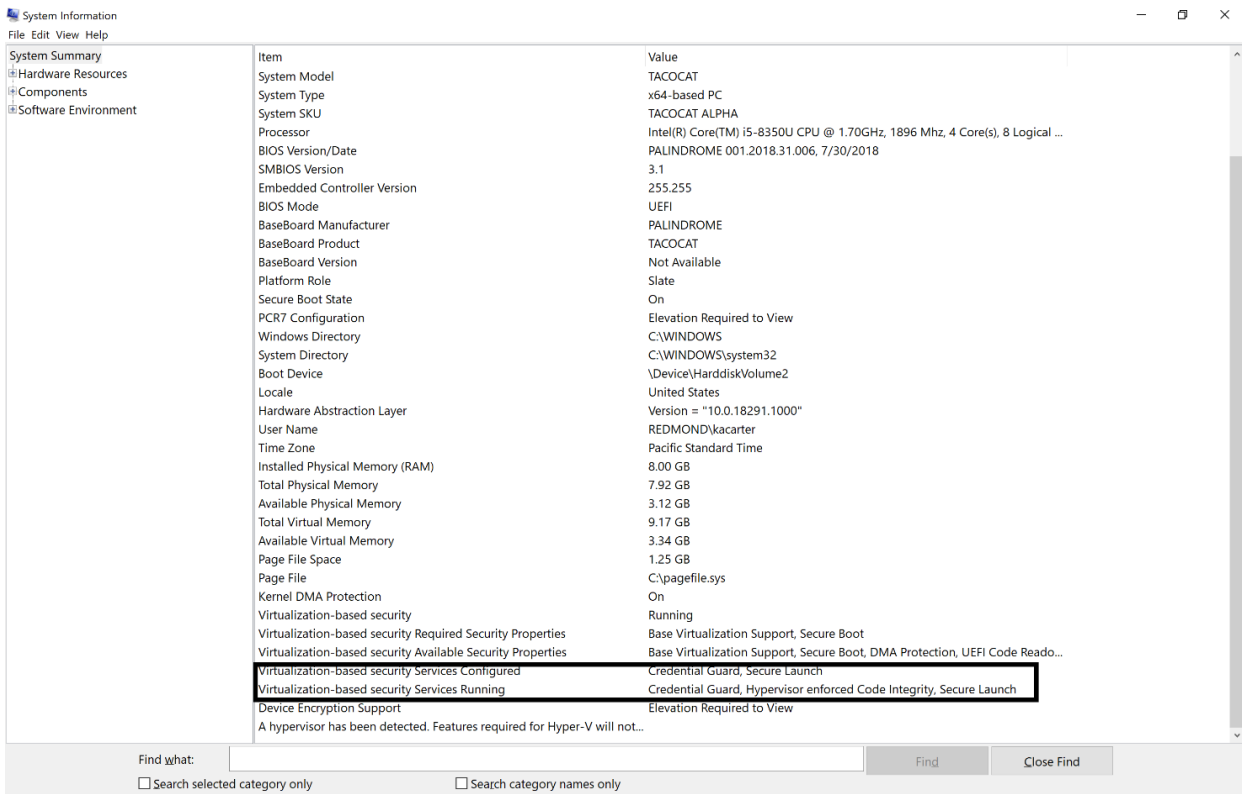
Registry

1. Open Registry editor.
2. Click HKEY_LOCAL_MACHINE > SYSTEM > CurrentControlSet > Control > DeviceGuard > Scenarios.
3. Right-click Scenarios > New > Key and name the new key SystemGuard.
4. Right-click SystemGuard > New > DWORD (32-bit) Value and name the new DWORD Enabled.
5. Double-click Enabled, change the value to 1, and click OK.



How to verify System Guard Secure Launch is configured and running

To verify that Secure Launch is running, use System Information (MSInfo32). Click **Start**, search for **System Information**, and look under **Virtualization-based Security Services Running** and **Virtualization-based Security Services Configured**.



NOTE

To enable System Guard Secure launch, the platform must meet all the baseline requirements for [Device Guard](#), [Credential Guard](#), and [Virtualization Based Security](#).

System requirements for System Guard

FOR INTEL® VPRO™ PROCESSORS STARTING WITH INTEL® COFFEELAKE, WHISKEYLAKE, OR LATER SILICON	DESCRIPTION
64-bit CPU	A 64-bit computer with minimum four cores (logical processors) is required for hypervisor and virtualization-based security (VBS). For more information about Hyper-V, see Hyper-V on Windows Server 2016 or Introduction to Hyper-V on Windows 10 . For more information about hypervisor, see Hypervisor Specifications .
Trusted Platform Module (TPM) 2.0	Platforms must support a discrete TPM 2.0. Integrated/firmware TPMs aren't supported, except Intel chips that support Platform Trust Technology (PTT), which is a type of integrated hardware TPM that meets the TPM 2.0 spec.
Windows DMA Protection	Platforms must meet the Windows DMA Protection Specification (all external DMA ports must be off by default until the OS explicitly powers them).
SMM communication buffers	All SMM communication buffers must be implemented in EfiRuntimeServicesData, EfiRuntimeServicesCode, EfiACPIMemoryNVS, or EfiReservedMemoryType memory types.

FOR INTEL® VPRO™ PROCESSORS STARTING WITH INTEL® COFFEELAKE, WHISKEYLAKE, OR LATER SILICON	DESCRIPTION
SMM Page Tables	<p>Must NOT contain any mappings to EfiConventionalMemory (for example no OS/VMM owned memory).</p> <p>Must NOT contain any mappings to code sections within EfiRuntimeServicesCode.</p> <p>Must NOT have execute and write permissions for the same page</p> <p>Must allow ONLY that TSEG pages can be marked executable and the memory map must report TSEG EfiReservedMemoryType.</p> <p>BIOS SMI handler must be implemented such that SMM page tables are locked on every SMM entry.</p>
Modern/Connected Standby	Platforms must support Modern/Connected Standby.
TPM AUX Index	<p>Platform must set up a AUX index with index, attributes, and policy that exactly corresponds to the AUX index specified in the TXT DG with a data size of exactly 104 bytes (for SHA256 AUX data). (NameAlg = SHA256)</p> <p>Platforms must set up a PS (Platform Supplier) index with:</p> <ul style="list-style-type: none"> • Exactly the "TXT PS2" style Attributes on creation as follows: <ul style="list-style-type: none"> ◦ AuthWrite ◦ PolicyDelete ◦ WriteLocked ◦ WriteDefine ◦ AuthRead ◦ WriteDefine ◦ NoDa ◦ Written ◦ PlatformCreate • A policy of exactly PolicyCommandCode(CC = TPM2_CC_UndefineSpaceSpecial) (SHA256 NameAlg and Policy) • Size of exactly 70 bytes • NameAlg = SHA256 • Also, it must have been initialized and locked (TPMA_NV_WRITTEN = 1, TPMA_NV_WRITELOCKED = 1) at time of OS launch. <p>PS index data DataRevocationCounters, SINITMinVersion, and PolicyControl must all be 0x00</p>
AUX Policy	<p>The required AUX policy must be as follows:</p> <ul style="list-style-type: none"> • A = TPM2_PolicyLocality (Locality 3 & Locality 4) • B = TPM2_PolicyCommandCode (TPM_CC_NV_UndefineSpecial) • authPolicy = {A} OR {{A} AND {B}} • authPolicy digest = 0xef, 0x9a, 0x26, 0xfc, 0x22, 0xd1, 0xae, 0x8c, 0xec, 0xff, 0x59, 0xe9, 0x48, 0x1a, 0xc1, 0xec, 0x53, 0x3d, 0xbe, 0x22, 0x8b, 0xec, 0x6d, 0x17, 0x93, 0x0f, 0x4c, 0xb2, 0xcc, 0x5b, 0x97, 0x24

FOR INTEL® VPRO™ PROCESSORS STARTING WITH INTEL® COFFEELAKE, WHISKEYLAKE, OR LATER SILICON	DESCRIPTION
TPM NV Index	<p>Platform firmware must set up a TPM NV index for use by the OS with:</p> <ul style="list-style-type: none"> • Handle: 0x01C101C0 • Attributes: <ul style="list-style-type: none"> ◦ TPMA_NV_POLICYWRITE ◦ TPMA_NV_PPREAD ◦ TPMA_NV_OWNERREAD ◦ TPMA_NV_AUTHREAD ◦ TPMA_NV_POLICYREAD ◦ TPMA_NV_NO_DA ◦ TPMA_NV_PLATFORMCREATE ◦ TPMA_NV_POLICY_DELETE • A policy of: • A = TPM2_PolicyAuthorize(MSFT_DRTM_AUTH_BLOB_SigningKey) • B = TPM2_PolicyCommandCode(TPM_CC_NV_UndefineSpaceSpecial) • authPolicy = {A} OR {{A} AND {B}} • Digest value of 0xcb, 0x45, 0xc8, 0x1f, 0xf3, 0x4b, 0xcf, 0x0a, 0xfb, 0x9e, 0x1a, 0x80, 0x29, 0xfa, 0x23, 0x1c, 0x87, 0x27, 0x30, 0x3c, 0x09, 0x22, 0xdc, 0xce, 0x68, 0x4b, 0xe3, 0xdb, 0x81, 0x7c, 0x20, 0xe1
Platform firmware	<p>Platform firmware must carry all code required to execute an Intel® Trusted Execution Technology secure launch:</p> <ul style="list-style-type: none"> • Intel® SINIT ACM must be carried in the OEM BIOS • Platforms must ship with a production ACM signed by the correct production Intel® ACM signer for the platform
Platform firmware update	System firmware is recommended to be updated via UpdateCapsule in Windows Update.
FOR AMD® PROCESSORS STARTING WITH ZEN2 OR LATER SILICON	DESCRIPTION
64-bit CPU	A 64-bit computer with minimum four cores (logical processors) is required for hypervisor and virtualization-based security (VBS). For more information about Hyper-V, see Hyper-V on Windows Server 2016 or Introduction to Hyper-V on Windows 10 . For more information about hypervisor, see Hypervisor Specifications .
Trusted Platform Module (TPM) 2.0	Platforms must support a discrete TPM 2.0 OR Microsoft Pluton TPM.
Windows DMA Protection	Platforms must meet the Windows DMA Protection Specification (all external DMA ports must be off by default until the OS explicitly powers them).

FOR AMD [®] PROCESSORS STARTING WITH ZEN2 OR LATER SILICON	DESCRIPTION
SMM communication buffers	All SMM communication buffers must be implemented in EfiRuntimeServicesData, EfiRuntimeServicesCode, EfiACPIMemoryNVS, or EfiReservedMemoryType memory types.
SMM Page Tables	<p>Must NOT contain any mappings to EfiConventionalMemory (for example no OS/VMM owned memory).</p> <p>Must NOT contain any mappings to code sections within EfiRuntimeServicesCode.</p> <p>Must NOT have execute and write permissions for the same page</p> <p>BIOS SMI handler must be implemented such that SMM page tables are locked on every SMM entry.</p>
Modern/Connected Standby	Platforms must support Modern/Connected Standby.
TPM NV Index	<p>Platform firmware must set up a TPM NV index for use by the OS with:</p> <ul style="list-style-type: none"> • Handle: 0x01C101C0 • Attributes: <ul style="list-style-type: none"> ◦ TPMA_NV_POLICYWRITE ◦ TPMA_NV_PPREAD ◦ TPMA_NV_OWNERREAD ◦ TPMA_NV_AUTHREAD ◦ TPMA_NV_POLICYREAD ◦ TPMA_NV_NO_DA ◦ TPMA_NV_PLATFORMCREATE ◦ TPMA_NV_POLICY_DELETE • A policy of: • A = TPM2_PolicyAuthorize(MSFT_DRTM_AUTH_BLOB_SigningKey) • B = TPM2_PolicyCommandCode(TPM_CC_NV_UndefineSpaceSpecial) • authPolicy = {A} OR {{A} AND {B}} • Digest value of 0xcb, 0x45, 0xc8, 0x1f, 0xf3, 0x4b, 0xcf, 0x0a, 0xfb, 0x9e, 0x1a, 0x80, 0x29, 0xfa, 0x23, 0x1c, 0x87, 0x27, 0x30, 0x3c, 0x09, 0x22, 0xdc, 0xce, 0x68, 0x4b, 0xe3, 0xdb, 0x81, 0x7c, 0x20, 0xe1
Platform firmware	<p>Platform firmware must carry all code required to execute Secure Launch:</p> <ul style="list-style-type: none"> • AMD[®] Secure Launch platforms must ship with AMD[®] DRTM driver devnode exposed and the AMD[®] DRTM driver installed <p>Platform must have AMD[®] Secure Processor Firmware Anti-Rollback protection enabled</p> <p>Platform must have AMD[®] Memory Guard enabled.</p>
Platform firmware update	System firmware is recommended to be updated via UpdateCapsule in Windows Update.

FOR QUALCOMM® PROCESSORS WITH SD850 OR LATER CHIPSETS	DESCRIPTION
Monitor Mode Communication	All Monitor Mode communication buffers must be implemented in either EfiRuntimeServicesData (recommended), data sections of EfiRuntimeServicesCode as described by the Memory Attributes Table, EfiACPIMemoryNVS, or EfiReservedMemoryType memory types
Monitor Mode Page Tables	<p>All Monitor Mode page tables must:</p> <ul style="list-style-type: none"> • NOT contain any mappings to EfiConventionalMemory (for example no OS/VMM owned memory) • They must NOT have execute and write permissions for the same page • Platforms must only allow Monitor Mode pages marked as executable • The memory map must report Monitor Mode as EfiReservedMemoryType • Platforms must provide mechanism to protect the Monitor Mode page tables from modification
Modern/Connected Standby	Platforms must support Modern/Connected Standby.
Platform firmware	Platform firmware must carry all code required to launch.
Platform firmware update	System firmware is recommended to be updated via UpdateCapsule in Windows Update.

NOTE

For more information around AMD processors, see [Microsoft Security Blog: Force firmware code to be measured and attested by Secure Launch on Windows 10](#).

Enable virtualization-based protection of code integrity

7/1/2022 • 9 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11

This topic covers different ways to enable Hypervisor-protected code integrity (HVCI) on Windows 10 and Windows 11. Some applications, including device drivers, may be incompatible with HVCI. This can cause devices or software to malfunction and in rare cases may result in a blue screen. Such issues may occur after HVCI has been turned on or during the enablement process itself. If this happens, see [Troubleshooting](#) for remediation steps.

NOTE

Because it makes use of *Mode Based Execution Control*, HVCI works better with Intel Kaby Lake or AMD Zen 2 CPUs and newer. Processors without MBEC will rely on an emulation of this feature, called *Restricted User Mode*, which has a bigger impact on performance.

HVCI Features

- HVCI protects modification of the Control Flow Guard (CFG) bitmap.
- HVCI also ensures that your other trusted processes, like Credential Guard, have got a valid certificate.
- Modern device drivers must also have an EV (Extended Validation) certificate and should support HVCI.

How to turn on HVCI in Windows 10 and Windows 11

To enable HVCI on Windows 10 and Windows 11 devices with supporting hardware throughout an enterprise, use any of these options:

- [Windows Security app](#)
- [Microsoft Intune \(or another MDM provider\)](#)
- [Group Policy](#)
- [Microsoft Endpoint Configuration Manager](#)
- [Registry](#)

Windows Security app

HVCI is labeled **Memory integrity** in the Windows Security app and it can be accessed via **Settings > Update & Security > Windows Security > Device security > Core isolation details > Memory integrity**. For more information, see [KB4096339](#).

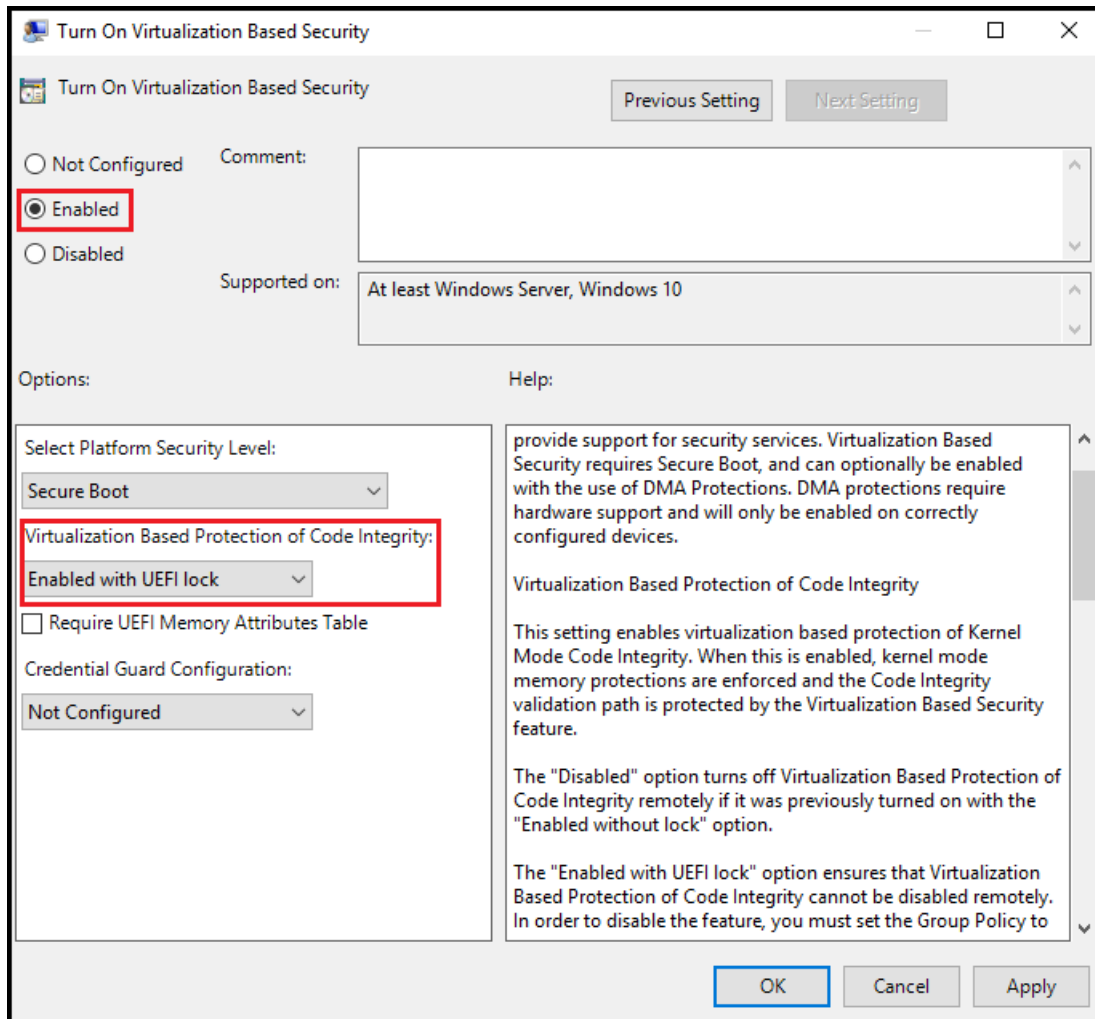
Enable HVCI using Intune

Enabling in Intune requires using the Code Integrity node in the [AppLocker CSP](#).

Enable HVCI using Group Policy

1. Use Group Policy Editor (gpedit.msc) to either edit an existing GPO or create a new one.

2. Navigate to **Computer Configuration > Administrative Templates > System > Device Guard**.
3. Double-click **Turn on Virtualization Based Security**.
4. Click **Enabled** and under **Virtualization Based Protection of Code Integrity**, select **Enabled with UEFI lock** to ensure HVCI cannot be disabled remotely or select **Enabled without UEFI lock**.



5. Click **Ok** to close the editor.

To apply the new policy on a domain-joined computer, either restart or run `gpupdate /force` in an elevated command prompt.

Use registry keys to enable virtualization-based protection of code integrity

Set the following registry keys to enable HVCI. This provides exactly the same set of configuration options provided by Group Policy.

IMPORTANT

- Among the commands that follow, you can choose settings for **Secure Boot** and **Secure Boot with DMA**. In most situations, we recommend that you choose **Secure Boot**. This option provides Secure Boot with as much protection as is supported by a given computer's hardware. A computer with input/output memory management units (IOMMUs) will have Secure Boot with DMA protection. A computer without IOMMUs will simply have Secure Boot enabled.
- In contrast, with **Secure Boot with DMA**, the setting will enable Secure Boot—and VBS itself—only on a computer that supports DMA, that is, a computer with IOMMUs. With this setting, any computer without IOMMUs will not have VBS or HVCI protection, although it can still have Windows Defender Application Control enabled.
- All drivers on the system must be compatible with virtualization-based protection of code integrity; otherwise, your system may fail. We recommend that you enable these features on a group of test computers before you enable them on users' computers.

For Windows 10 version 1607 and later and for Windows 11 version 21H2

Recommended settings (to enable virtualization-based protection of Code Integrity policies, without UEFI Lock):

```
reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v "EnableVirtualizationBasedSecurity" /t REG_DWORD /d 1 /f

reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v "RequirePlatformSecurityFeatures" /t REG_DWORD /d 1 /f

reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v "Locked" /t REG_DWORD /d 0 /f

reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Scenarios\HypervisorEnforcedCodeIntegrity" /v "Enabled" /t REG_DWORD /d 1 /f

reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Scenarios\HypervisorEnforcedCodeIntegrity" /v "Locked" /t REG_DWORD /d 0 /f
```

If you want to customize the preceding recommended settings, use the following settings.

To enable VBS

```
reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v "EnableVirtualizationBasedSecurity" /t REG_DWORD /d 1 /f
```

To enable VBS and require Secure boot only (value 1)

```
reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v "RequirePlatformSecurityFeatures" /t REG_DWORD /d 1 /f
```

To enable VBS with Secure Boot and DMA (value 3)

```
reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v "RequirePlatformSecurityFeatures" /t REG_DWORD /d 3 /f
```

To enable VBS without UEFI lock (value 0)

```
reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v "Locked" /t REG_DWORD /d 0 /f
```


To enable VBS with UEFI lock (value 1)

```
reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v "Locked" /t REG_DWORD /d 1 /f
```

To enable virtualization-based protection of Code Integrity policies

```
reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Scenarios\HypervisorEnforcedCodeIntegrity" /v "Enabled" /t REG_DWORD /d 1 /f
```

To enable virtualization-based protection of Code Integrity policies without UEFI lock (value 0)

```
reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Scenarios\HypervisorEnforcedCodeIntegrity" /v "Locked" /t REG_DWORD /d 0 /f
```

To enable virtualization-based protection of Code Integrity policies with UEFI lock (value 1)

```
reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Scenarios\HypervisorEnforcedCodeIntegrity" /v "Locked" /t REG_DWORD /d 1 /f
```

For Windows 10 version 1511 and earlier

Recommended settings (to enable virtualization-based protection of Code Integrity policies, without UEFI Lock):

```
reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v "EnableVirtualizationBasedSecurity" /t REG_DWORD /d 1 /f

reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v "RequirePlatformSecurityFeatures" /t REG_DWORD /d 1 /f

reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v "HypervisorEnforcedCodeIntegrity" /t REG_DWORD /d 1 /f

reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v "Unlocked" /t REG_DWORD /d 1 /f
```

If you want to customize the preceding recommended settings, use the following settings.

To enable VBS (it is always locked to UEFI)

```
reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v "EnableVirtualizationBasedSecurity" /t REG_DWORD /d 1 /f
```

To enable VBS and require Secure boot only (value 1)

```
reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v "RequirePlatformSecurityFeatures" /t REG_DWORD /d 1 /f
```

To enable VBS with Secure Boot and DMA (value 3)

```
reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v "RequirePlatformSecurityFeatures" /t REG_DWORD /d 3 /f
```

To enable virtualization-based protection of Code Integrity policies (with the default, UEFI lock)

```
reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v "HypervisorEnforcedCodeIntegrity" /t REG_DWORD /d 1 /f
```

To enable virtualization-based protection of Code Integrity policies without UEFI lock

```
reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v "Unlocked" /t REG_DWORD /d 1 /f
```

Validate enabled Windows Defender Device Guard hardware-based security features

Windows 10, Windows 11, and Windows Server 2016 have a WMI class for related properties and features: *Win32_DeviceGuard*. This class can be queried from an elevated Windows PowerShell session by using the following command:

```
Get-CimInstance -ClassName Win32_DeviceGuard -Namespace root\Microsoft\Windows\DeviceGuard
```

NOTE

The *Win32_DeviceGuard* WMI class is only available on the Enterprise edition of Windows 10 and Windows 11.

NOTE

Mode Based Execution Control property will only be listed as available starting with Windows 10 version 1803 and Windows 11 version 21H2.

The output of this command provides details of the available hardware-based security features as well as those features that are currently enabled.

AvailableSecurityProperties

This field helps to enumerate and report state on the relevant security properties for Windows Defender Device Guard.

VALUE	DESCRIPTION
0.	If present, no relevant properties exist on the device.
1.	If present, hypervisor support is available.
2.	If present, Secure Boot is available.
3.	If present, DMA protection is available.
4.	If present, Secure Memory Overwrite is available.
5.	If present, NX protections are available.
6.	If present, SMM mitigations are available.
7.	If present, Mode Based Execution Control is available.
8.	If present, APIC virtualization is available.

InstanceIdentifier

A string that is unique to a particular device. Valid values are determined by WMI.

RequiredSecurityProperties

This field describes the required security properties to enable virtualization-based security.

VALUE	DESCRIPTION
0.	Nothing is required.
1.	If present, hypervisor support is needed.
2.	If present, Secure Boot is needed.
3.	If present, DMA protection is needed.
4.	If present, Secure Memory Overwrite is needed.
5.	If present, NX protections are needed.
6.	If present, SMM mitigations are needed.
7.	If present, Mode Based Execution Control is needed.

SecurityServicesConfigured

This field indicates whether the Windows Defender Credential Guard or HVCI service has been configured.

VALUE	DESCRIPTION
0.	No services configured.
1.	If present, Windows Defender Credential Guard is configured.
2.	If present, HVCI is configured.
3.	If present, System Guard Secure Launch is configured.
4.	If present, SMM Firmware Measurement is configured.

SecurityServicesRunning

This field indicates whether the Windows Defender Credential Guard or HVCI service is running.

VALUE	DESCRIPTION
0.	No services running.
1.	If present, Windows Defender Credential Guard is running.
2.	If present, HVCI is running.
3.	If present, System Guard Secure Launch is running.
4.	If present, SMM Firmware Measurement is running.

Version

This field lists the version of this WMI class. The only valid value now is **1.0**.

VirtualizationBasedSecurityStatus

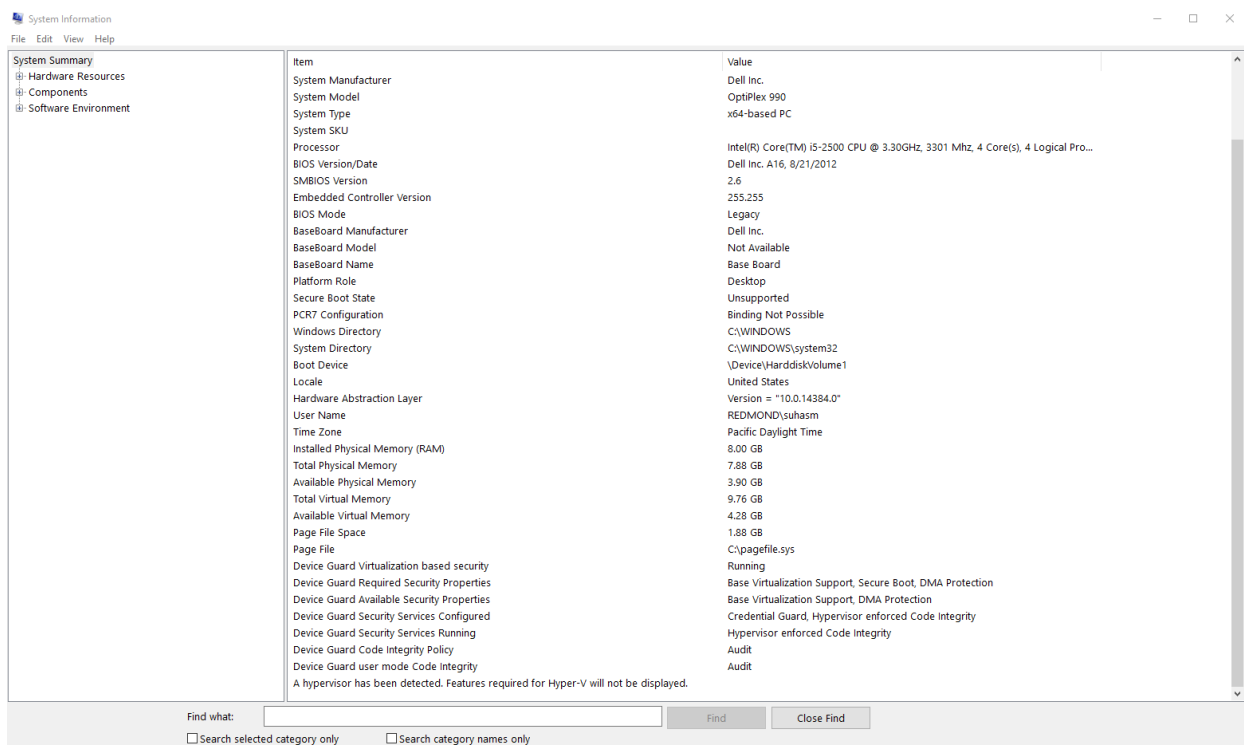
This field indicates whether VBS is enabled and running.

VALUE	DESCRIPTION
0.	VBS is not enabled.
1.	VBS is enabled but not running.
2.	VBS is enabled and running.

PSComputerName

This field lists the computer name. All valid values for computer name.

Another method to determine the available and enabled Windows Defender Device Guard features is to run `msinfo32.exe` from an elevated PowerShell session. When you run this program, the Windows Defender Device Guard properties are displayed at the bottom of the **System Summary** section.



The screenshot shows the Windows System Information application. The 'System Summary' section is expanded, displaying various system properties. At the bottom of the list, Device Guard properties are shown, including 'Device Guard Virtualization based security' (Running), 'Device Guard Required Security Properties' (Base Virtualization Support, Secure Boot, DMA Protection), 'Device Guard Available Security Properties' (Base Virtualization Support, DMA Protection), 'Device Guard Security Services Configured' (Credential Guard, Hypervisor enforced Code Integrity), 'Device Guard Security Services Running' (Hypervisor enforced Code Integrity), 'Device Guard Code Integrity Policy' (Audit), and 'Device Guard user mode Code Integrity' (Audit). A note at the bottom states: 'A hypervisor has been detected. Features required for Hyper-V will not be displayed.'

Troubleshooting

A. If a device driver fails to load or crashes at runtime, you may be able to update the driver using **Device Manager**.

B. If you experience software or device malfunction after using the above procedure to turn on HVCI, but you are able to log in to Windows, you can turn off HVCI by renaming or deleting the `SIPolicy.p7b` file from

`<OS Volume>\Windows\System32\CodeIntegrity\` and then restart your device.

C. If you experience a critical error during boot or your system is unstable after using the above procedure to turn on HVCI, you can recover using the Windows Recovery Environment (Windows RE). To boot to Windows RE, see [Windows RE Technical Reference](#). After logging in to Windows RE, you can turn off HVCI by renaming or deleting the `SIPolicy.p7b` file from `<OS Volume>\Windows\System32\CodeIntegrity\` and then restart your device.

How to turn off HVCI

1. Run the following command from an elevated prompt to set the HVCI registry key to off:

```
reg add "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Scenarios\HypervisorEnforcedCodeIntegrity"  
/v "Enabled" /t REG_DWORD /d 0 /f
```

2. Restart the device.
3. To confirm HVCI has been successfully disabled, open System Information and check **Virtualization-based security Services Running**, which should now have no value displayed.

HVCI deployment in virtual machines

HVCI can protect a Hyper-V virtual machine, just as it would a physical machine. The steps to enable Windows Defender Application Control are the same from within the virtual machine.

WDAC protects against malware running in the guest virtual machine. It does not provide additional protection from the host administrator. From the host, you can disable WDAC for a virtual machine:

```
Set-VMSecurity -VMName <VMName> -VirtualizationBasedSecurityOptOut $true
```

Requirements for running HVCI in Hyper-V virtual machines

- The Hyper-V host must run at least Windows Server 2016 or Windows 10 version 1607.
- The Hyper-V virtual machine must be Generation 2, and running at least Windows Server 2016 or Windows 10.
- HVCI and [nested virtualization](#) can be enabled at the same time. To enable the HyperV role on the virtual machine, you must first install the HyperV role in a Windows nested virtualization environment.
- Virtual Fibre Channel adapters are not compatible with HVCI. Before attaching a virtual Fibre Channel Adapter to a virtual machine, you must first opt out of virtualization-based security using `Set-VMSecurity`.
- The AllowFullSCSICommandSet option for pass-through disks is not compatible with HVCI. Before configuring a pass-through disk with AllowFullSCSICommandSet, you must first opt out of virtualization-based security using `Set-VMSecurity`.

Kernel DMA Protection

7/1/2022 • 7 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11

In Windows 10 version 1803, Microsoft introduced a new feature called Kernel DMA Protection to protect PCs against drive-by Direct Memory Access (DMA) attacks using PCI hot plug devices connected to externally accessible PCIe ports (for example, Thunderbolt™ 3 ports and CFexpress). In Windows 10 version 1903, Microsoft expanded the Kernel DMA Protection support to cover internal PCIe ports (for example, M.2 slots)

Drive-by DMA attacks can lead to disclosure of sensitive information residing on a PC, or even injection of malware that allows attackers to bypass the lock screen or control PCs remotely.

This feature doesn't protect against DMA attacks via 1394/FireWire, PCMCIA, CardBus, ExpressCard, and so on.

Background

PCI devices are DMA-capable, which allows them to read and write to system memory at will, without having to engage the system processor in these operations. The DMA capability is what makes PCI devices the highest performing devices available today. These devices have historically existed only inside the PC chassis, either connected as a card or soldered on the motherboard. Access to these devices required the user to turn off power to the system and disassemble the chassis.

Today, this is no longer the case with hot plug PCIe ports (for example, Thunderbolt™ and CFexpress).

Hot plug PCIe ports such as Thunderbolt™ technology have provided modern PCs with extensibility that wasn't available before for PCs. It allows users to attach new classes of external peripherals, such as graphics cards or other PCI devices, to their PCs with a hot plug experience identical to USB. Having PCI hot plug ports externally and easily accessible makes PCs susceptible to drive-by DMA attacks.

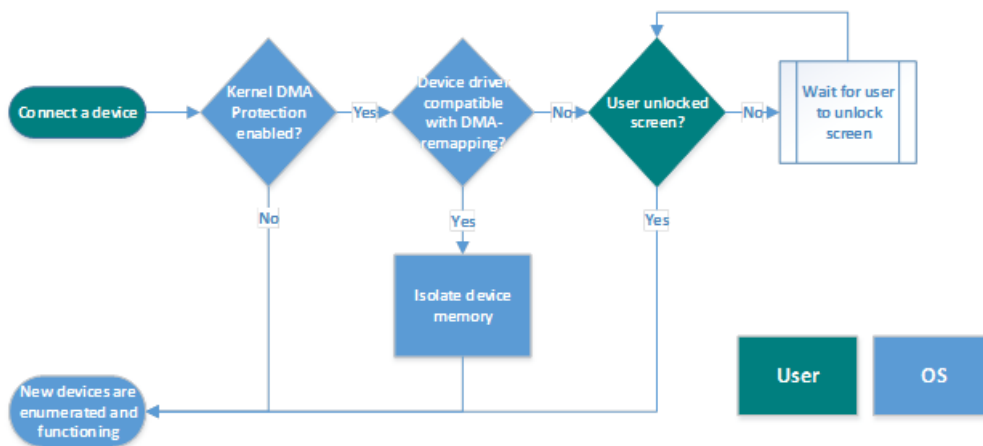
Drive-by DMA attacks are attacks that occur while the owner of the system is not present and usually take less than 10 minutes, with simple to moderate attacking tools (affordable, off-the-shelf hardware and software) that do not require the disassembly of the PC. A simple example would be a PC owner leaves the PC for a quick coffee break, and within the break, an attacker steps in, plugs in a USB-like device and walks away with all the secrets on the machine, or injects a malware that allows them to have full control over the PC remotely.

How Windows protects against DMA drive-by attacks

Windows leverages the system Input/Output Memory Management Unit (IOMMU) to block external peripherals from starting and performing DMA unless the drivers for these peripherals support memory isolation (such as DMA-remapping). Peripherals with [DMA Remapping compatible drivers](#) will be automatically enumerated, started, and allowed to perform DMA to their assigned memory regions.

By default, peripherals with DMA Remapping incompatible drivers will be blocked from starting and performing DMA until an authorized user signs into the system or unlocks the screen. IT administrators can modify the default behavior applied to devices with DMA Remapping incompatible drivers using the [DmaGuard MDM policies](#).

User experience



By default, peripherals with DMA remapping compatible device drivers will be automatically enumerated and started. Peripherals with DMA Remapping incompatible drivers will be blocked from starting if the peripheral was plugged in before an authorized user logs in, or while the screen is locked. Once the system is unlocked, the peripheral driver will be started by the OS, and the peripheral will continue to function normally until the system is rebooted, or the peripheral is unplugged. The peripheral will continue to function normally if the user locks the screen or logs out of the system.

System compatibility

Kernel DMA Protection requires new UEFI firmware support. This support is anticipated only on newly introduced, Intel-based systems shipping with Windows 10 version 1803 (not all systems). Virtualization-based Security (VBS) is not required.

To see if a system supports Kernel DMA Protection, check the System Information desktop app (MSINFO32). Systems released prior to Windows 10 version 1803 do not support Kernel DMA Protection, but they can leverage other DMA attack mitigations as described in [BitLocker countermeasures](#).

NOTE

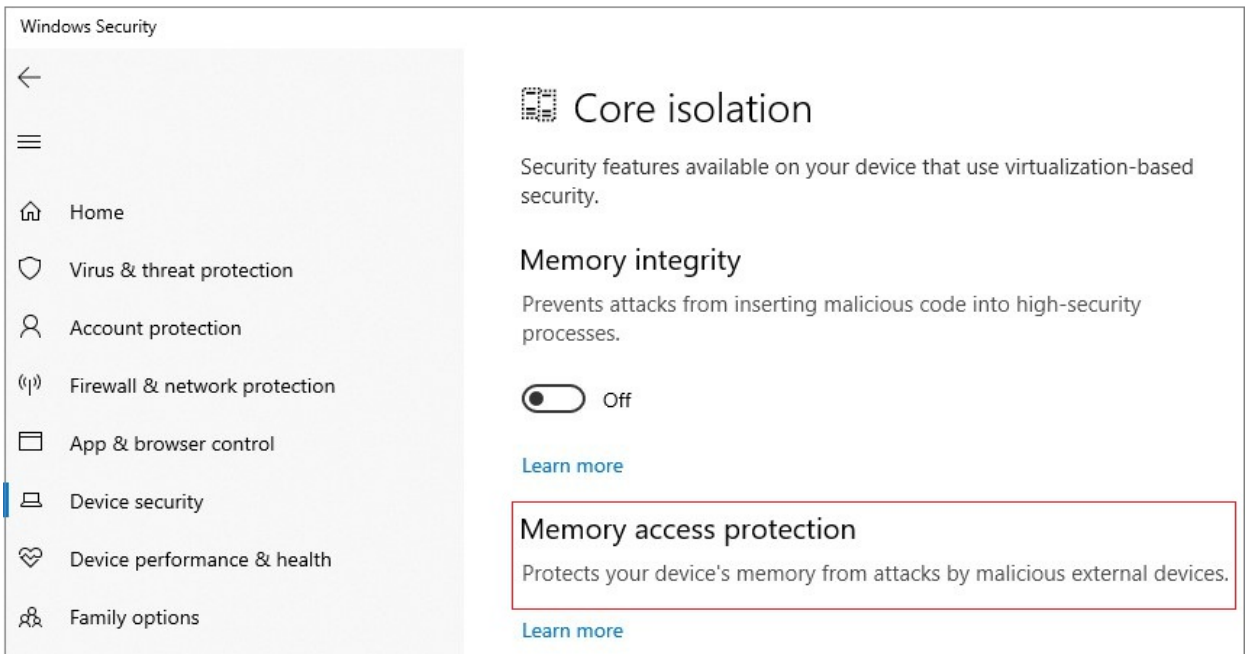
Kernel DMA Protection is not compatible with other BitLocker DMA attacks countermeasures. It is recommended to disable the BitLocker DMA attacks countermeasures if the system supports Kernel DMA Protection. Kernel DMA Protection provides higher security bar for the system over the BitLocker DMA attack countermeasures, while maintaining usability of external peripherals.

How to check if Kernel DMA Protection is enabled

Systems running Windows 10 version 1803 that do support Kernel DMA Protection do have this security feature enabled automatically by the OS with no user or IT admin configuration required.

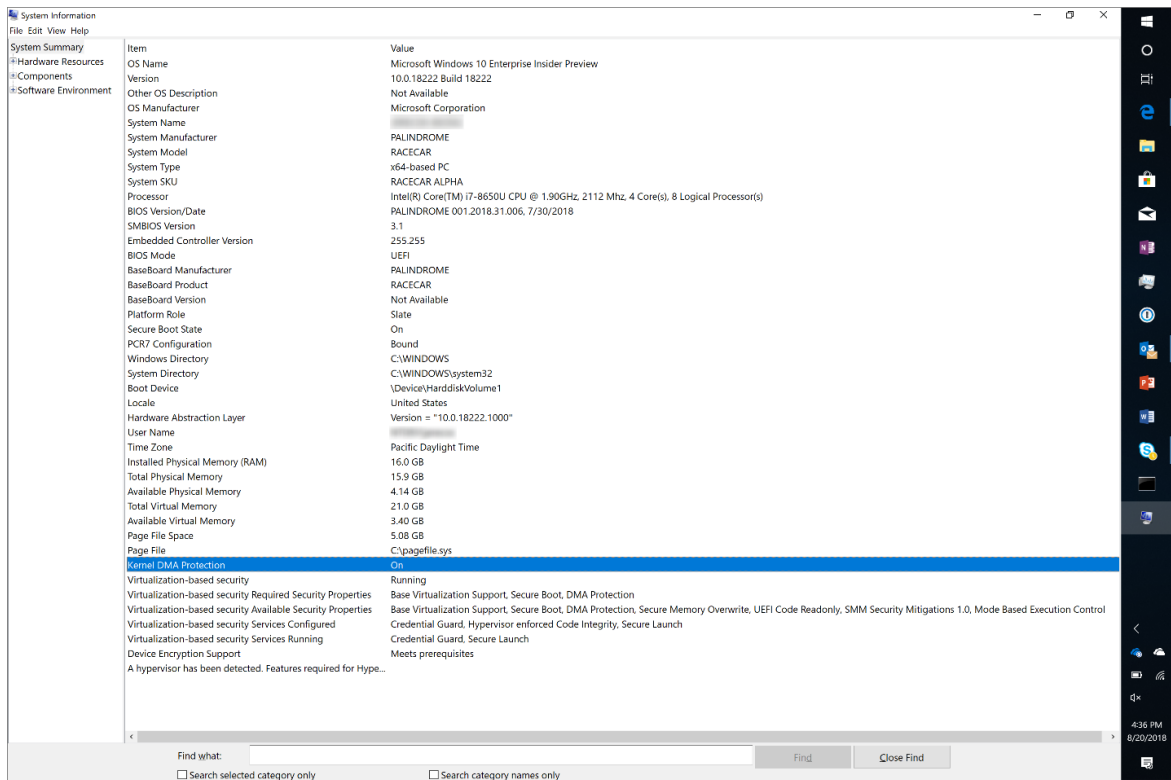
Using the Windows Security app

Beginning with Windows 10 version 1809, you can use the Windows Security app to check if Kernel DMA Protection is enabled. Click **Start > Settings > Update & Security > Windows Security > Open Windows Security > Device security > Core isolation details > Memory access protection**.



Using System information

1. Launch MSINFO32.exe in a command prompt, or in the Windows search bar.
2. Check the value of Kernel DMA Protection.



3. If the current state of Kernel DMA Protection is OFF and Hyper-V - Virtualization Enabled in Firmware is NO:

- Reboot into BIOS settings
- Turn on Intel Virtualization Technology.
- Turn on Intel Virtualization Technology for I/O (VT-d). In Windows 10 version 1803, only Intel VT-d is supported. Other platforms can use DMA attack mitigations described in [BitLocker countermeasures](#).
- Reboot system into Windows.

NOTE

Hyper-V - Virtualization Enabled in Firmware is not available when **A hypervisor has been detected**. **Features required for Hyper-V will not be displayed**. is displayed. This means that **Hyper-V - Virtualization Enabled in Firmware** is set to Yes and the **Hyper-V Windows** feature is enabled. Enabling Hyper-V virtualization in Firmware (IOMMU) is required to enable **Kernel DMA Protection**, even when the firmware has the flag of "ACPI Kernel DMA Protection Indicators" described in [Kernel DMA Protection \(Memory Access Protection\) for OEMs](#).

4. If the state of **Kernel DMA Protection** remains Off, then the system does not support this feature.

For systems that do not support Kernel DMA Protection, please refer to the [BitLocker countermeasures](#) or [Thunderbolt™ 3 and Security on Microsoft Windows® 10 Operating system](#) for other means of DMA protection.

Frequently asked questions

Do in-market systems support Kernel DMA Protection for Thunderbolt™ 3?

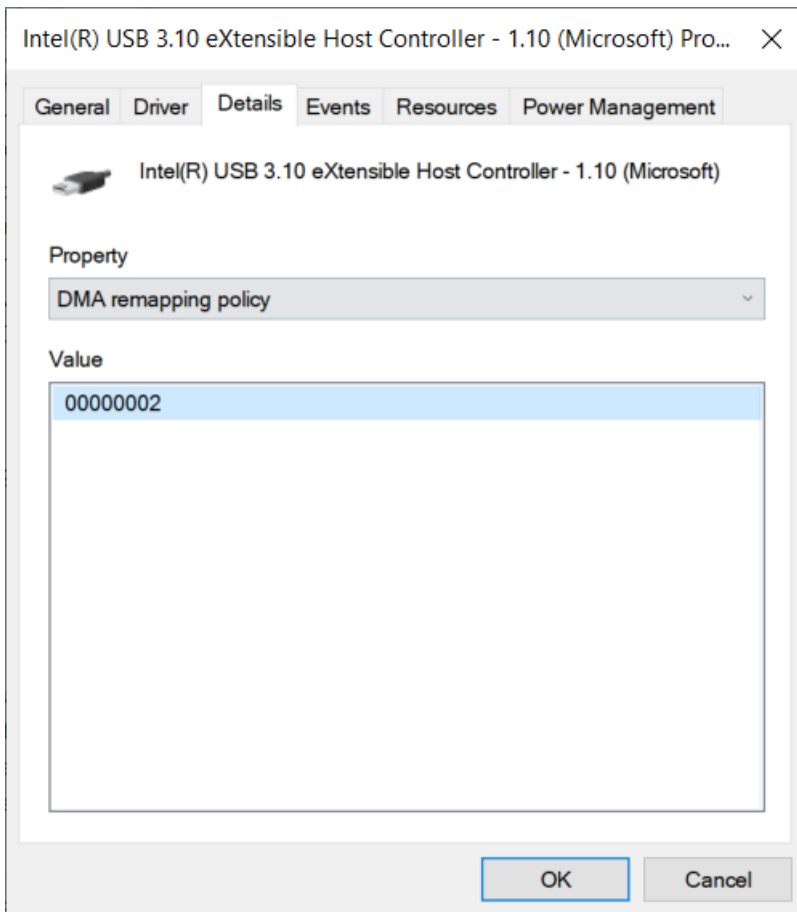
In-market systems, released with Windows 10 version 1709 or earlier, will not support Kernel DMA Protection for Thunderbolt™ 3 after upgrading to Windows 10 version 1803, as this feature requires the BIOS/platform firmware changes and guarantees that cannot be backported to previously released devices. For these systems, please refer to the [BitLocker countermeasures](#) or [Thunderbolt™ 3 and Security on Microsoft Windows® 10 Operating system](#) for other means of DMA protection.

Does Kernel DMA Protection prevent drive-by DMA attacks during Boot?

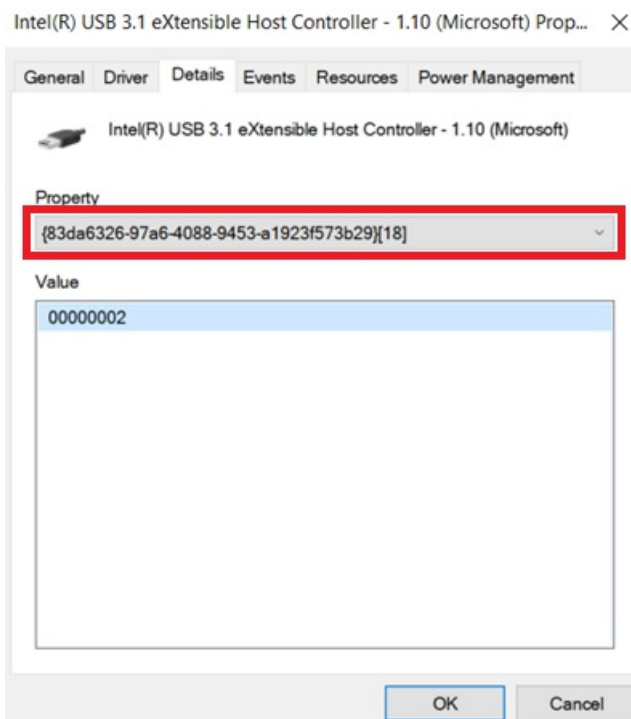
No, Kernel DMA Protection only protects against drive-by DMA attacks after the OS is loaded. It is the responsibility of the system firmware/BIOS to protect against attacks via the Thunderbolt™ 3 ports during boot.

How can I check if a certain driver supports DMA-remapping?

DMA-remapping is supported for specific device drivers, and is not universally supported by all devices and drivers on a platform. To check if a specific driver is opted into DMA-remapping, check the values corresponding to the DMA Remapping Policy property in the Details tab of a device in Device Manager*. A value of 0 or 1 means that the device driver does not support DMA-remapping. A value of two means that the device driver supports DMA-remapping. If the property is not available, then the policy is not set by the device driver (that is, the device driver does not support DMA-remapping). Check the driver instance for the device you are testing. Some drivers may have varying values depending on the location of the device (internal vs. external).



*For Windows 10 versions 1803 and 1809, the property field in Device Manager uses a GUID, as highlighted in the following image.



When the drivers for PCI or Thunderbolt™ 3 peripherals do not support DMA-remapping?

If the peripherals do have class drivers provided by Windows, use these drivers on your systems. If there are no class drivers provided by Windows for your peripherals, contact your peripheral vendor/driver vendor to update the driver to support [DMA Remapping](#).

My system's Kernel DMA Protection is off. Can DMA-remapping for a specific device be turned on?

Yes. DMA remapping for a specific device can be turned on independent from Kernel DMA Protection. For

example, if the driver opts in and VT-d (Virtualization Technology for Directed I/O) is turned on, then DMA remapping will be enabled for the devices driver even if Kernel DMA Protection is turned off.

Kernel DMA Protection is a policy that allows or blocks devices to perform DMA, based on their remapping state and capabilities.

Do Microsoft drivers support DMA-remapping?

In Windows 10 1803 and beyond, the Microsoft inbox drivers for USB XHCI (3.x) Controllers, Storage AHCI/SATA Controllers, and Storage NVMe Controllers support DMA Remapping.

Do drivers for non-PCI devices need to be compatible with DMA-remapping?

No. Devices for non-PCI peripherals, such as USB devices, do not perform DMA, thus no need for the driver to be compatible with DMA Remapping.

How can an enterprise enable the External device enumeration policy?

The External device enumeration policy controls whether to enumerate external peripherals that are not compatible with DMA-remapping. Peripherals that are compatible with DMA-remapping are always enumerated. Peripherals that aren't, can be blocked, allowed, or allowed only after the user signs in (default).

The policy can be enabled by using:

- Group Policy: Administrative Templates\System\Kernel DMA Protection\Enumeration policy for external devices incompatible with Kernel DMA Protection
- Mobile Device Management (MDM): [DmaGuard policies](#)

Related topics

- [BitLocker countermeasures](#)
- [DmaGuard MDM policies](#)

Windows operating system security

7/1/2022 • 6 minutes to read • [Edit Online](#)

Security and privacy depend on an operating system that guards your system and information from the moment it starts up, providing fundamental chip-to-cloud protection. Windows 11 is the most secure Windows yet with extensive security measures designed to help keep you safe. These measures include built-in advanced encryption and data protection, robust network and system security, and intelligent safeguards against ever-evolving threats.

Watch the latest [Microsoft Mechanics Windows 11 security](#) video that shows off some of the latest Windows 11 security technology.

Use the links in the following table to learn more about the operating system security features and capabilities in Windows 11.

SECURITY MEASURES	FEATURES & CAPABILITIES
Secure Boot and Trusted Boot	<p>Secure Boot and Trusted Boot help prevent malware and corrupted components from loading when a Windows device is starting. Secure Boot starts with initial boot-up protection, and then Trusted Boot picks up the process. Together, Secure Boot and Trusted Boot help to ensure your Windows system boots up safely and securely.</p> <p>Learn more Secure Boot and Trusted Boot.</p>
Cryptography and certificate management	<p>Cryptography uses code to convert data so that only a specific recipient can read it by using a key. Cryptography enforces privacy to prevent anyone except the intended recipient from reading data, integrity to ensure data is free of tampering, and authentication that verifies identity to ensure that communication is secure.</p> <p>Learn more about Cryptography and certificate management.</p>
Windows Security app	<p>The Windows built-in security application found in settings provides an at-a-glance view of the security status and health of your device. These insights help you identify issues and take action to make sure you're protected. You can quickly see the status of your virus and threat protection, firewall and network security, device security controls, and more.</p> <p>Learn more about the Windows Security app.</p>
Encryption and data protection	<p>Wherever confidential data is stored, it must be protected against unauthorized access, whether through physical device theft or from malicious applications. Windows provides strong at-rest data-protection solutions that guard against nefarious attackers.</p> <p>Learn more about Encryption.</p>

SECURITY MEASURES	FEATURES & CAPABILITIES
BitLocker	<p>BitLocker Drive Encryption is a data protection feature that integrates with the operating system and addresses the threats of data theft or exposure from lost, stolen, or inappropriately decommissioned computers. BitLocker provides the most protection when used with a Trusted Platform Module (TPM) version 1.2 or later.</p> <p>Learn more about BitLocker.</p>
Encrypted Hard Drive	<p>Encrypted Hard Drive uses the rapid encryption that is provided by BitLocker Drive Encryption to enhance data security and management.</p> <p>By offloading the cryptographic operations to hardware, Encrypted Hard Drives increase BitLocker performance and reduce CPU usage and power consumption. Because Encrypted Hard Drives encrypt data quickly, enterprise devices can expand BitLocker deployment with minimal impact on productivity.</p> <p>Learn more about Encrypted Hard Drives.</p>
Security baselines	<p>A security baseline is a group of Microsoft-recommended configuration settings that explains their security impact. These settings are based on feedback from Microsoft security engineering teams, product groups, partners, and customers.</p> <p>Security baselines are included in the Security Compliance Toolkit that you can download from the Microsoft Download Center.</p> <p>Learn more about security baselines.</p>
Virtual Private Network	<p>Virtual private networks (VPNs) are point-to-point connections across a private or public network, such as the Internet. A VPN client uses special TCP/IP or UDP-based protocols, called tunneling protocols, to make a virtual call to a virtual port on a VPN server.</p> <p>Learn more about Virtual Private Networks.</p>
Windows Defender Firewall	<p>Windows Defender Firewall is a stateful host firewall that helps secure the device by allowing you to create rules that determine which network traffic is permitted to enter the device from the network and which network traffic the device is allowed to send to the network. Windows Defender Firewall also supports Internet Protocol security (IPsec), which you can use to require authentication from any device that is attempting to communicate with your device.</p> <p>Learn more about Windows Defender Firewall with advanced security.</p>

SECURITY MEASURES	FEATURES & CAPABILITIES
Antivirus & antimalware protection	<p>Microsoft Defender Antivirus is included in all versions of Windows 10, Windows Server 2016 and later, and Windows 11. If you have another antivirus app installed and turned on, Microsoft Defender Antivirus will turn off automatically. If you uninstall the other app, Microsoft Defender Antivirus will turn back on.</p> <p>From the moment you boot Windows, Microsoft Defender Antivirus continually monitors for malware, viruses, and security threats. Updates are downloaded automatically to help protect your device from threats. Microsoft Defender Antivirus continually scans for malware and threats, and also detects and blocks potentially unwanted applications (applications that can negatively impact your device even though they are not considered malware).</p> <p>Microsoft Defender Antivirus integrates with cloud-delivered protection, which helps ensure near-instant detection and blocking of new and emerging threats.</p> <p>Learn more about next-generation protection and Microsoft Defender Antivirus.</p>
Attack surface reduction rules	<p>Your attack surfaces are the places and ways you are vulnerable to a cyber attack. Attack surface reduction rules are built into Windows and Windows Server to prevent and block certain behaviors that are often abused to compromise your device or network. Such behaviors can include launching scripts or executables that attempt to download or run other files, running suspicious scripts, or performing other behaviors that apps don't typically initiate during normal work. You can configure your attack surface reduction rules to protect against these risky behaviors.</p> <p>Learn more about Attack surface reduction rules</p>
Anti-tampering protection	<p>During cyber attacks (like ransomware attempts), bad actors attempt to disable security features, such as antivirus protection on targeted devices. Bad actors like to disable security features to get easier access to user's data, to install malware, or to otherwise exploit user's data, identity, and devices without fear of being blocked. Tamper protection helps prevent these kinds of activities.</p> <p>With tamper protection, malware is prevented from taking actions such as:</p> <ul style="list-style-type: none"> - Disabling virus and threat protection - Disabling real-time protection - Turning off behavior monitoring - Disabling antivirus (such as IOfficeAntivirus (IOAV)) - Disabling cloud-delivered protection - Removing security intelligence updates <p>Learn more about Tamper protection.</p>

SECURITY MEASURES	FEATURES & CAPABILITIES
Network protection	<p>Network protection in Windows helps prevent users from accessing dangerous IP addresses and domains that may host phishing scams, exploits, and other malicious content on the Internet. Network protection is part of attack surface reduction and helps provide an extra layer of protection for a user. Using reputation-based services, network protection blocks access to potentially harmful, low-reputation based domains and IP addresses.</p> <p>In enterprise environments, network protection works best with Microsoft Defender for Endpoint, which provides detailed reporting into protection events as part of larger investigation scenarios.</p> <p>Learn more about Network protection.</p>
Controlled folder access	<p>With controlled folder access, you can protect your valuable information in specific folders by managing apps' access to specific folders. Only trusted apps can access protected folders, which are specified when controlled folder access is configured. Typically, commonly used folders, such as those used for documents, pictures, downloads, are included in the list of controlled folders. Controlled folder access helps protect valuable data from malicious apps and threats, such as ransomware.</p> <p>Learn more about Controlled folder access.</p>
Exploit protection	<p>Exploit protection, available in Windows 10, version 1709 and later, automatically applies several exploit mitigation techniques to operating system processes and apps. Exploit protection works best with Microsoft Defender for Endpoint, which gives organizations detailed reporting into exploit protection events and blocks as part of typical alert investigation scenarios.</p> <p>You can enable exploit protection on an individual device, and then use Group Policy to distribute the XML file to multiple devices simultaneously. When a mitigation is encountered on the device, a notification will be displayed from the Action Center. You can customize the notification with your company details and contact information. You can also enable the rules individually to customize which techniques the feature monitors.</p> <p>Learn more about Exploit protection.</p>

SECURITY MEASURES	FEATURES & CAPABILITIES
Microsoft Defender for Endpoint	<p>Windows E5 customers benefit from Microsoft Defender for Endpoint, an enterprise endpoint detection and response capability that helps enterprise security teams detect, investigate, and respond to advanced threats. With rich event data and attack insights, Defender for Endpoint enables your security team to investigate incidents and take remediation actions effectively and efficiently.</p> <p>Defender for Endpoint also is part of Microsoft 365 Defender, a unified pre- and post-breach enterprise defense suite that natively coordinates detection, prevention, investigation, and response across endpoints, identities, email, and applications to provide integrated protection against sophisticated attacks.</p> <p>Learn more about Microsoft Defender for Endpoint and Microsoft 365 Defender.</p>

Secure the Windows boot process

7/1/2022 • 8 minutes to read • [Edit Online](#)

Applies to:

- Windows 11
- Windows 10
- Windows 8.1

The Windows OS has many features to help protect you from malware, and it does an amazingly good job. Except for apps that businesses develop and use internally, all Microsoft Store apps must meet a series of requirements to be certified and included in the Microsoft Store. This certification process examines several criteria, including security, and is an effective means of preventing malware from entering the Microsoft Store. Even if a malicious app does get through, the Windows 10 OS includes a series of security features that can mitigate the effect. For instance, Microsoft Store apps are sandboxed and lack the privileges necessary to access user data or change system settings.

Windows has multiple levels of protection for desktop apps and data, too. Windows Defender Antivirus uses cloud-powered real-time detection to identify and quarantine apps that are known to be malicious. Windows Defender SmartScreen warns users before allowing them to run an untrustworthy app, even if it's recognized as malware. Before an app can change system settings, the user would have to grant the app administrative privileges by using User Account Control.

Those components are just some of the ways that Windows protects you from malware. However, those security features protect you only after Windows starts. Modern malware, and bootkits specifically, are capable of starting before Windows, completely bypassing OS security, and remaining hidden.

When you run Windows 10 or Windows 11 on a PC or any PC that supports Unified Extensible Firmware Interface (UEFI), Trusted Boot protects your PC from malware from the moment you power on your PC until your anti-malware starts. In the unlikely event that malware does infect a PC, it can't remain hidden; Trusted Boot can prove the system's integrity to your infrastructure in a way that malware can't disguise. Even on PCs without UEFI, Windows provides even better startup security than previous versions of Windows.

First, let's examine what rootkits are and how they work. Then, we'll show you how Windows can protect you.

The threat: rootkits

Rootkits are a sophisticated and dangerous type of malware. They run in kernel mode, using the same privileges as the OS. Because rootkits have the same rights as the OS and start before it, they can completely hide themselves and other applications. Often, rootkits are part of an entire suite of malware that can bypass local logins, record passwords and keystrokes, transfer private files, and capture cryptographic data.

Different types of rootkits load during different phases of the startup process:

- **Firmware rootkits.** These kits overwrite the firmware of the PC's basic input/output system or other hardware so the rootkit can start before Windows.
- **Bootkits.** These kits replace the OS's bootloader (the small piece of software that starts the OS) so that the PC loads the bootkit before the OS.
- **Kernel rootkits.** These kits replace a portion of the OS kernel so the rootkit can start automatically when the OS loads.
- **Driver rootkits.** These kits pretend to be one of the trusted drivers that Windows uses to communicate with

the PC hardware.

The countermeasures

Windows supports four features to help prevent rootkits and bootkits from loading during the startup process:

- **Secure Boot.** PCs with UEFI firmware and a Trusted Platform Module (TPM) can be configured to load only trusted OS bootloaders.
- **Trusted Boot.** Windows checks the integrity of every component of the startup process before loading it.
- **Early Launch Anti-Malware (ELAM).** ELAM tests all drivers before they load and prevents unapproved drivers from loading.
- **Measured Boot.** The PC's firmware logs the boot process, and Windows can send it to a trusted server that can objectively assess the PC's health.

Figure 1 shows the Windows startup process.

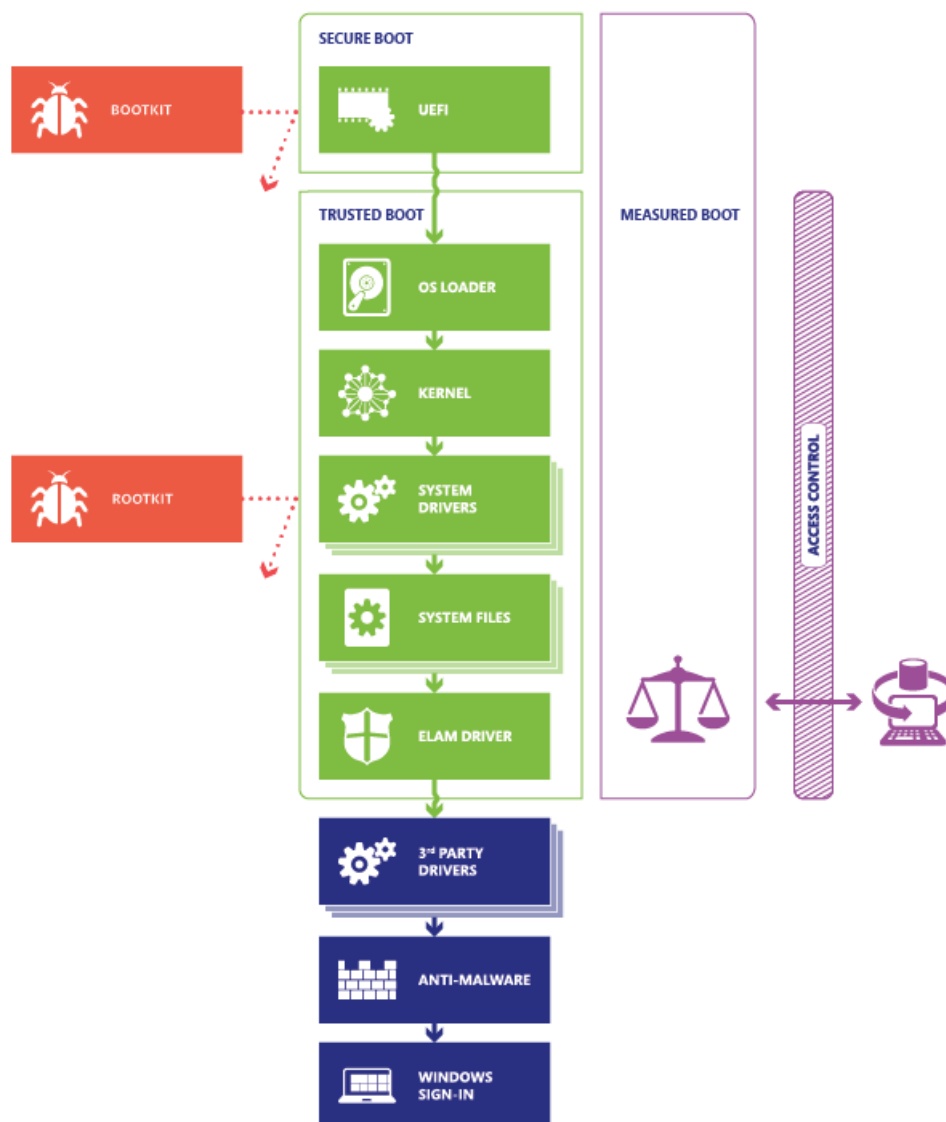


Figure 1. Secure Boot, Trusted Boot, and Measured Boot block malware at every stage

Secure Boot and Measured Boot are only possible on PCs with UEFI 2.3.1 and a TPM chip. Fortunately, all Windows 10 and Windows 11 PCs that meet Windows Hardware Compatibility Program requirements have these components, and many PCs designed for earlier versions of Windows have them as well.

The sections that follow describe Secure Boot, Trusted Boot, ELAM, and Measured Boot.

Secure Boot

When a PC starts, it first finds the OS bootloader. PCs without Secure Boot run whatever bootloader is on the PC's hard drive. There's no way for the PC to tell whether it's a trusted OS or a rootkit.

When a PC equipped with UEFI starts, the PC first verifies that the firmware is digitally signed, reducing the risk of firmware rootkits. If Secure Boot is enabled, the firmware examines the bootloader's digital signature to verify that it hasn't been modified. If the bootloader is intact, the firmware starts the bootloader only if one of the following conditions is true:

- **The bootloader was signed using a trusted certificate.** For PCs certified for Windows, the Microsoft certificate is trusted.
- **The user has manually approved the bootloader's digital signature.** This action allows the user to load non-Microsoft operating systems.

All x86-based Certified For Windows PCs must meet several requirements related to Secure Boot:

- They must have Secure Boot enabled by default.
- They must trust Microsoft's certificate (and thus any bootloader Microsoft has signed).
- They must allow the user to configure Secure Boot to trust other bootloaders.
- They must allow the user to completely disable Secure Boot.

These requirements help protect you from rootkits while allowing you to run any OS you want. You have three options for running non-Microsoft operating systems:

- **Use an OS with a certified bootloader.** Because all Certified For Windows PCs must trust Microsoft's certificate, Microsoft offers a service to analyze and sign any non-Microsoft bootloader so that it will be trusted by all Certified For Windows PCs. In fact, an [open source bootloader](#) capable of loading Linux is already available. To begin the process of obtaining a certificate, go to <https://partner.microsoft.com/dashboard>.
- **Configure UEFI to trust your custom bootloader.** All Certified For Windows PCs allow you to trust a non-certified bootloader by adding a signature to the UEFI database, allowing you to run any OS, including homemade operating systems.
- **Turn off Secure Boot.** All *Certified For Windows* PCs allow you to turn off Secure Boot so that you can run any software. This action doesn't help protect you from bootkits, however.

To prevent malware from abusing these options, the user must manually configure the UEFI firmware to trust a non-certified bootloader or to turn off Secure Boot. Software can't change the Secure Boot settings.

Like most mobile devices, ARM-based Certified For Windows RT devices, such as the Microsoft Surface RT device, are designed to run only Windows 8.1. Therefore, Secure Boot can't be turned off, and you can't load a different OS. Fortunately, there's a large market of ARM processor devices designed to run other operating systems.

Trusted Boot

Trusted Boot takes over where Secure Boot ends. The bootloader verifies the digital signature of the Windows 10 kernel before loading it. The Windows 10 kernel, in turn, verifies every other component of the Windows startup process, including the boot drivers, startup files, and ELAM. If a file has been modified, the bootloader detects the problem and refuses to load the corrupted component. Often, Windows can automatically repair the corrupted component, restoring the integrity of Windows and allowing the PC to start normally.

Early Launch Anti-Malware

Because Secure Boot has protected the bootloader and Trusted Boot has protected the Windows kernel, the next opportunity for malware to start is by infecting a non-Microsoft boot driver. Traditional anti-malware apps don't start until after the boot drivers have been loaded, giving a rootkit disguised as a driver the opportunity to work.

Early Launch Anti-Malware (ELAM) can load a Microsoft or non-Microsoft anti-malware driver before all non-Microsoft boot drivers and applications, thus continuing the chain of trust established by Secure Boot and Trusted Boot. Because the OS hasn't started yet, and because Windows needs to boot as quickly as possible, ELAM has a simple task: examine every boot driver and determine whether it is on the list of trusted drivers. If it's not trusted, Windows won't load it.

An ELAM driver isn't a full-featured anti-malware solution; that loads later in the boot process. Windows Defender (included with Windows) supports ELAM, as does several non-Microsoft anti-malware apps.

Measured Boot

If a PC in your organization does become infected with a rootkit, you need to know about it. Enterprise anti-malware apps can report malware infections to the IT department, but that doesn't work with rootkits that hide their presence. In other words, you can't trust the client to tell you whether it's healthy.

As a result, PCs infected with rootkits appear to be healthy, even with anti-malware running. Infected PCs continue to connect to the enterprise network, giving the rootkit access to vast amounts of confidential data and potentially allowing the rootkit to spread across the internal network.

Measured Boot works with the TPM and non-Microsoft software in Windows. It allows a trusted server on the network to verify the integrity of the Windows startup process. Measured Boot uses the following process:

1. The PC's UEFI firmware stores in the TPM a hash of the firmware, bootloader, boot drivers, and everything that will be loaded before the anti-malware app.
2. At the end of the startup process, Windows starts the non-Microsoft remote attestation client. The trusted attestation server sends the client a unique key.
3. The TPM uses the unique key to digitally sign the log recorded by the UEFI.
4. The client sends the log to the server, possibly with other security information.

Depending on the implementation and configuration, the server can now determine whether the client is healthy. It can grant the client access to either a limited quarantine network or to the full network.

Figure 2 illustrates the Measured Boot and remote attestation process.

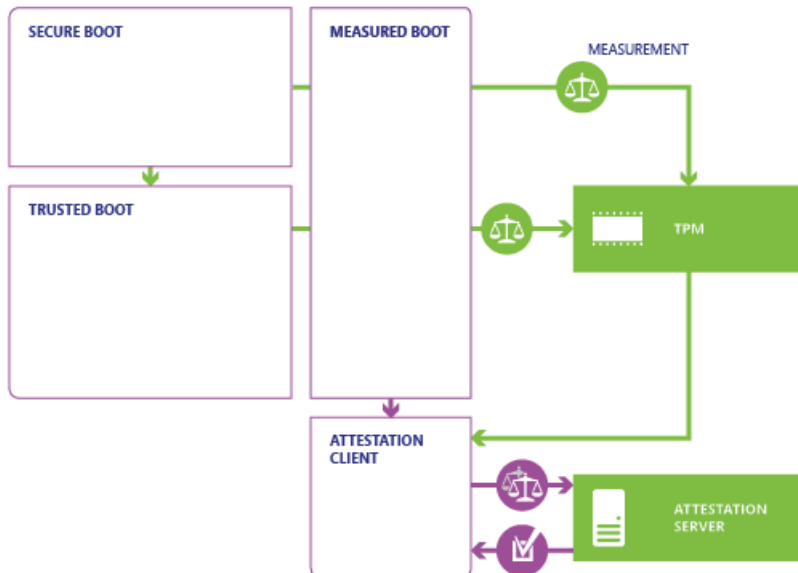


Figure 2. Measured Boot proves the PC's health to a remote server

Windows includes the application programming interfaces to support Measured Boot, but you'll need non-Microsoft tools to implement a remote attestation client and trusted attestation server to take advantage of it. For example, see the following tools from Microsoft Research:

- [TPM Platform Crypto-Provider Toolkit](#)
- [TSS.MSR](#)

Measured Boot uses the power of UEFI, TPM, and Windows to give you a way to confidently assess the trustworthiness of a client PC across the network.

Summary

Secure Boot, Trusted Boot, and Measured Boot create an architecture that is fundamentally resistant to bootkits and rootkits. In Windows, these features have the potential to eliminate kernel-level malware from your network. With Windows, you can trust the integrity of your OS.

Secure Boot and Trusted Boot

7/1/2022 • 2 minutes to read • [Edit Online](#)

This article describes Secure Boot and Trusted Boot, security measures built into Windows 11.

Secure Boot and Trusted Boot help prevent malware and corrupted components from loading when a Windows 11 device is starting. Secure Boot starts with initial boot-up protection, and then Trusted Boot picks up the process. Together, Secure Boot and Trusted Boot help to ensure your Windows 11 system boots up safely and securely.

Secure Boot

The first step in protecting the operating system is to ensure that it boots securely after the initial hardware and firmware boot sequences have safely finished their early boot sequences. Secure Boot makes a safe and trusted path from the Unified Extensible Firmware Interface (UEFI) through the Windows kernel's Trusted Boot sequence. Malware attacks on the Windows boot sequence are blocked by the signature-enforcement handshakes throughout the boot sequence between the UEFI, bootloader, kernel, and application environments.

As the PC begins the boot process, it will first verify that the firmware is digitally signed, reducing the risk of firmware rootkits. Secure Boot then checks all code that runs before the operating system and checks the OS bootloader's digital signature to ensure that it is trusted by the Secure Boot policy and hasn't been tampered with.

Trusted Boot

Trusted Boot picks up the process that started with Secure Boot. The Windows bootloader verifies the digital signature of the Windows kernel before loading it. The Windows kernel, in turn, verifies every other component of the Windows startup process, including boot drivers, startup files, and your antimalware product's early-launch antimalware (ELAM) driver. If any of these files were tampered, the bootloader detects the problem and refuses to load the corrupted component. Tampering or malware attacks on the Windows boot sequence are blocked by the signature-enforcement handshakes between the UEFI, bootloader, kernel, and application environments.

Often, Windows can automatically repair the corrupted component, restoring the integrity of Windows and allowing the Windows 11 device to start normally.

See also

[Secure the Windows boot process](#)

Cryptography and Certificate Management

7/1/2022 • 2 minutes to read • [Edit Online](#)

Cryptography

Cryptography uses code to convert data so that only a specific recipient can read it by using a key. Cryptography enforces privacy to prevent anyone except the intended recipient from reading data, integrity to ensure data is free of tampering, and authentication that verifies identity to ensure that communication is secure. The cryptography stack in Windows extends from the chip to the cloud enabling Windows, applications, and services protect system and user secrets.

Cryptography in Windows is Federal Information Processing Standards (FIPS) 140 certified. FIPS 140 certification ensures that US government approved algorithms are being used (RSA for signing, ECDH with NIST curves for key agreement, AES for symmetric encryption, and SHA2 for hashing), tests module integrity to prove that no tampering has occurred and proves the randomness for entropy sources.

Windows cryptographic modules provide low-level primitives such as:

- Random number generators (RNG)
- Symmetric and asymmetric encryption (support for AES 128/256 and RSA 512 to 16384, in 64-bit increments and ECDSA over NIST-standard prime curves P-256, P-384, P-521)
- Hashing (support for SHA-256, SHA-384, and SHA-512)
- Signing and verification (padding support for OAEP, PSS, PKCS1)
- Key agreement and key derivation (support for ECDH over NIST-standard prime curves P-256, P-384, P-521, and HKDF)

These modules are natively exposed on Windows through the Crypto API (CAPI) and the Cryptography Next Generation API (CNG) which is powered by Microsoft's open-source cryptographic library SymCrypt. Application developers can use these APIs to perform low-level cryptographic operations (BCrypt), key storage operations (NCrypt), protect static data (DPAPI), and securely share secrets (DPAPI-NG).

Certificate management

Windows offers several APIs to operate and manage certificates. Certificates are crucial to public key infrastructure (PKI) as they provide the means for safeguarding and authenticating information. Certificates are electronic documents used to claim ownership of a public key. Public keys are used to prove server and client identity, validate code integrity, and used in secure emails. Windows offers users the ability to auto-enroll and renew certificates in Active Directory with Group Policy to reduce the risk of potential outages due to certificate expiration or misconfiguration. Windows validates certificates through an automatic update mechanism that downloads certificate trust lists (CTL) daily. Trusted root certificates are used by applications as a reference for trustworthy PKI hierarchies and digital certificates. The list of trusted and untrusted certificates are stored in the CTL and can be updated by administrators. In the case of certificate revocation, a certificate is added as an untrusted certificate in the CTL causing it to be revoked globally across user devices immediately.

Windows also offers enterprise certificate pinning to help reduce man-in-the-middle attacks by enabling users to protect their internal domain names from chaining to unwanted certificates. A web application's server authentication certificate chain is checked to ensure it matches a restricted set of certificates. Any web application triggering a name mismatch will start event logging and prevent user access from Edge or Internet Explorer.

The Windows Security app

7/1/2022 • 3 minutes to read • [Edit Online](#)

Applies to

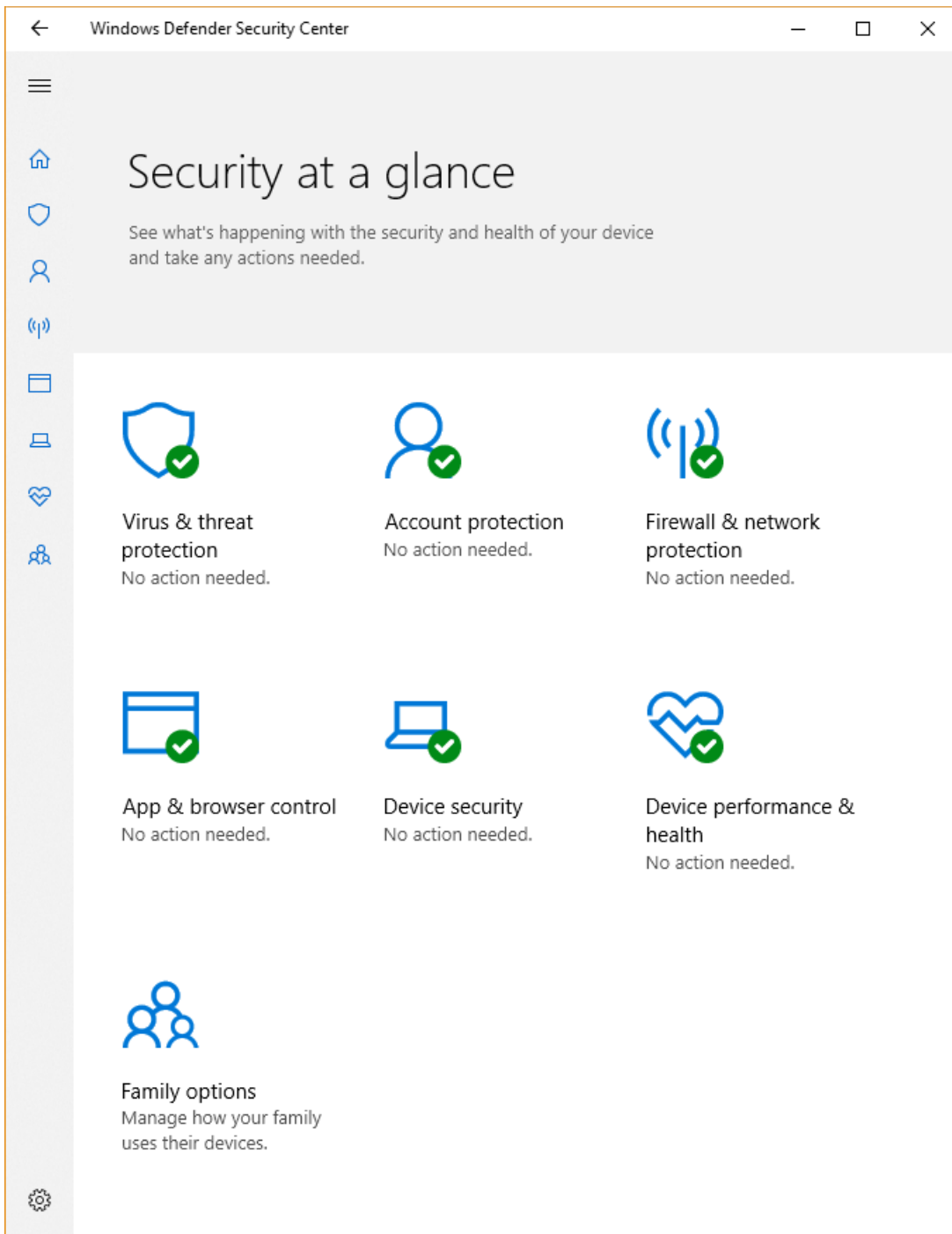
- Windows 10
- Windows 11

This library describes the Windows Security app, and provides information on configuring certain features, including:

- [Showing and customizing contact information on the app and in notifications](#)
- [Hiding notifications](#)

In Windows 10, version 1709 and later, the app also shows information from third-party antivirus and firewall apps.

In Windows 10, version 1803, the app has two new areas: **Account protection** and **Device security**.



NOTE

The Windows Security app is a client interface on Windows 10, version 1703 and later. It is not the Microsoft Defender Security Center web portal console that is used to review and manage [Microsoft Defender for Endpoint](#).

You can't uninstall the Windows Security app, but you can do one of the following:

- Disable the interface on Windows Server 2016. See [Microsoft Defender Antivirus on Windows Server](#).
- Hide all of the sections on client computers (see below).
- Disable Microsoft Defender Antivirus, if needed. See [Enable and configure Microsoft Defender AV always-on protection and monitoring](#).

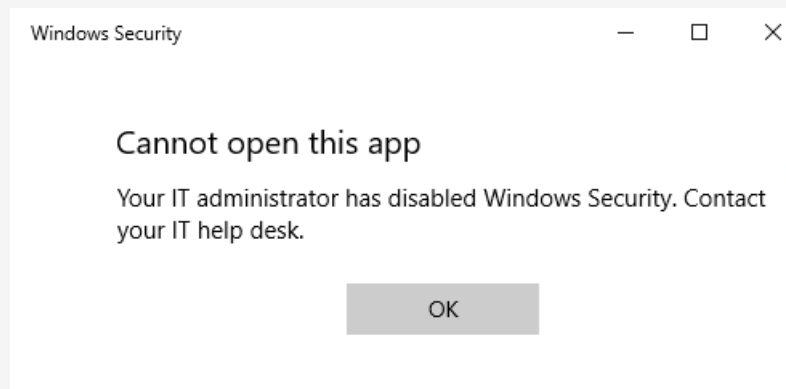
You can find more information about each section, including options for configuring the sections - such as hiding each of the sections - at the following topics:

- [Virus & threat protection](#), which has information and access to antivirus ransomware protection settings and notifications, including Controlled folder access, and sign-in to Microsoft OneDrive.

- [Account protection](#), which has information and access to sign-in and account protection settings.
- [Firewall & network protection](#), which has information and access to firewall settings, including Windows Defender Firewall.
- [App & browser control](#), covering Windows Defender SmartScreen settings and Exploit protection mitigations.
- [Device security](#), which provides access to built-in device security settings.
- [Device performance & health](#), which has information about drivers, storage space, and general Windows Update issues.
- [Family options](#), which includes access to parental controls along with tips and information for keeping kids safe online.

NOTE

If you hide all sections then the app will show a restricted interface, as in the following screenshot:

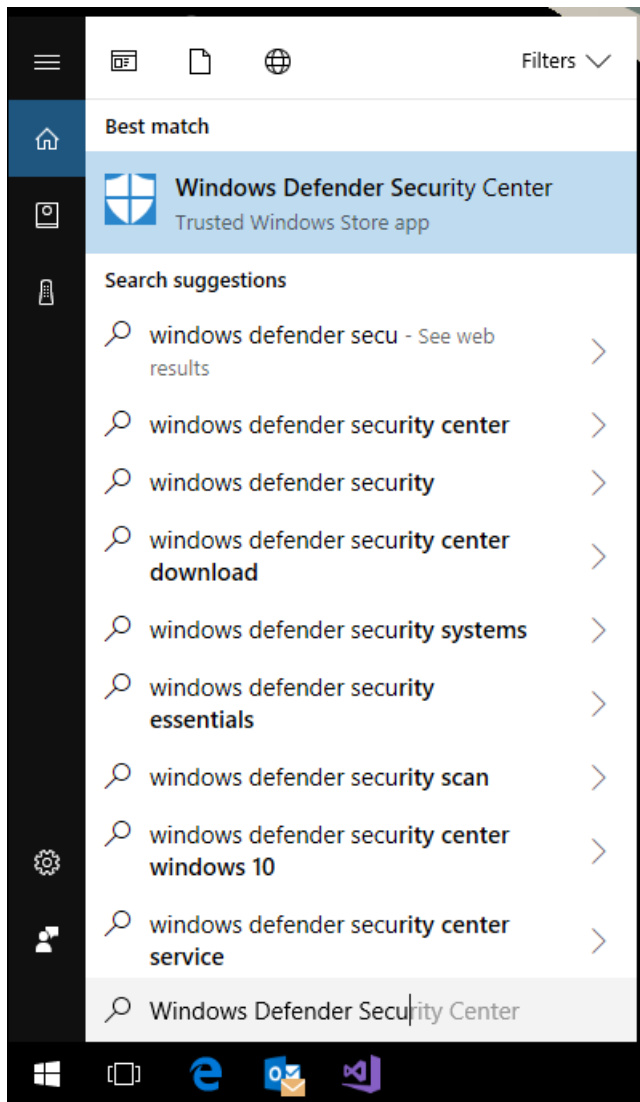


Open the Windows Security app

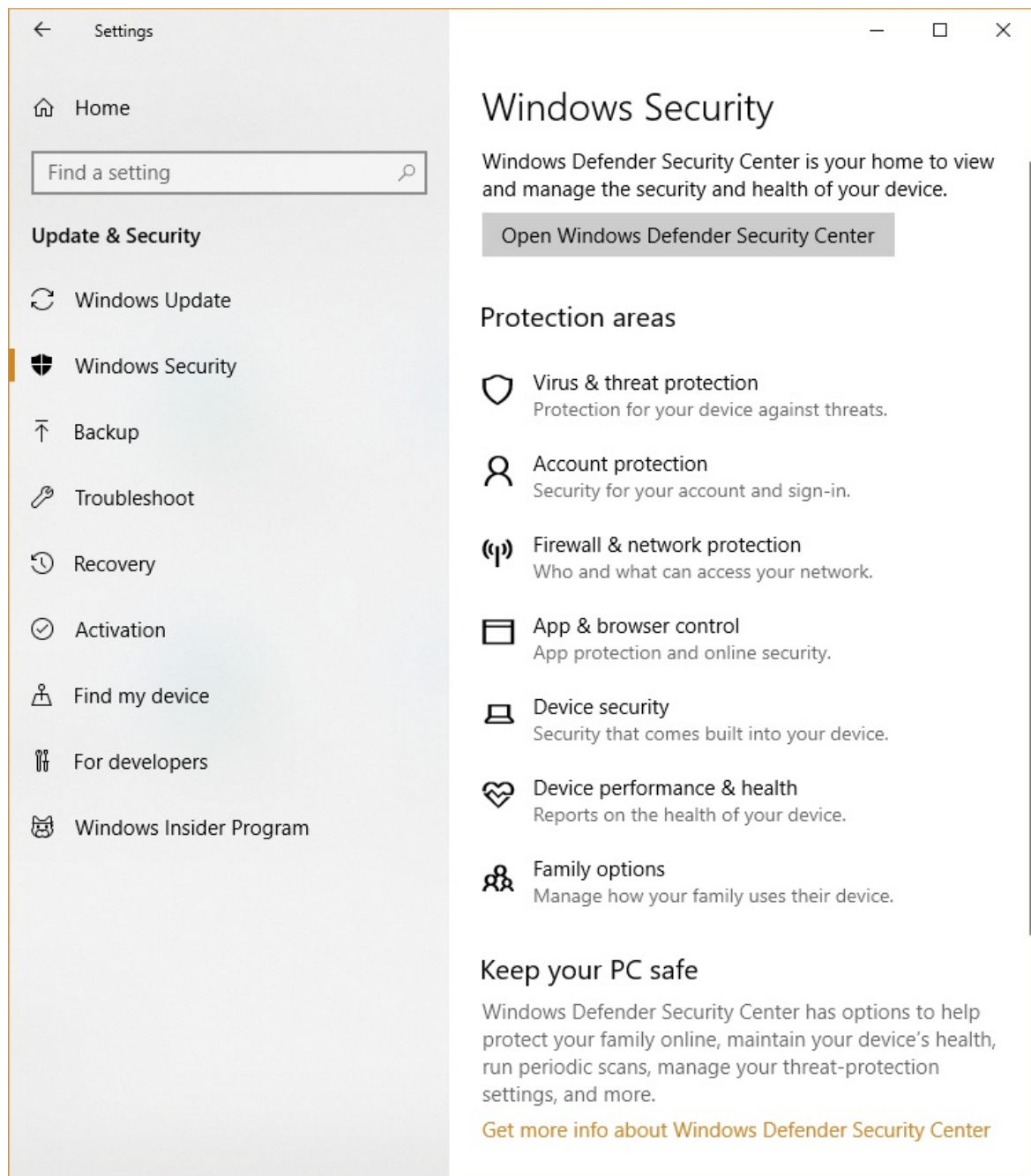
- Click the icon in the notification area on the taskbar.



- Search the Start menu for **Windows Security**.



- Open an area from Windows Settings.



NOTE

Settings configured with management tools, such as Group Policy, Microsoft Intune, or Microsoft Endpoint Configuration Manager, will generally take precedence over the settings in the Windows Security. See the topics for each of the sections for links to configuring the associated features or products.

How the Windows Security app works with Windows security features

IMPORTANT

Microsoft Defender Antivirus and the Windows Security app use similarly named services for specific purposes.

The Windows Security app uses the Windows Security Service (*SecurityHealthService* or *Windows Security Health Service*), which in turn utilizes the Windows Security Center Service (*wscsvc*) to ensure the app provides the most up-to-date information about the protection status on the endpoint, including protection offered by third-party antivirus products, Windows Defender Firewall, third-party firewalls, and other security protection.

These services do not affect the state of Microsoft Defender Antivirus. Disabling or modifying these services will not disable Microsoft Defender Antivirus, and will lead to a lowered protection state on the endpoint, even if you are using a third-party antivirus product.

Microsoft Defender Antivirus will be [disabled automatically when a third-party antivirus product is installed and kept up to date](#).

Disabling the Windows Security Center Service will not disable Microsoft Defender Antivirus or [Windows Defender Firewall](#).

WARNING

If you disable the Windows Security Center Service, or configure its associated Group Policy settings to prevent it from starting or running, the Windows Security app may display stale or inaccurate information about any antivirus or firewall products you have installed on the device.

It may also prevent Microsoft Defender Antivirus from enabling itself if you have an old or outdated third-party antivirus, or if you uninstall any third-party antivirus products you may have previously installed.

This will significantly lower the protection of your device and could lead to malware infection.

The Windows Security app operates as a separate app or process from each of the individual features, and will display notifications through the Action Center.

It acts as a collector or single place to see the status and perform some configuration for each of the features.

Disabling any of the individual features (through Group Policy or other management tools, such as Microsoft Endpoint Configuration Manager) will prevent that feature from reporting its status in the Windows Security app. The Windows Security app itself will still run and show status for the other security features.

IMPORTANT

Individually disabling any of the services will not disable the other services or the Windows Security app.

For example, [using a third-party antivirus will disable Microsoft Defender Antivirus](#). However, the Windows Security app will still run, show its icon in the taskbar, and display information about the other features, such as Windows Defender SmartScreen and Windows Defender Firewall.

Virus and threat protection

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11

The **Virus & threat protection** section contains information and settings for antivirus protection from Microsoft Defender Antivirus and third-party AV products.

In Windows 10, version 1803, this section also contains information and settings for ransomware protection and recovery. This includes Controlled folder access settings to prevent unknown apps from changing files in protected folders, plus Microsoft OneDrive configuration to help you recover from a ransomware attack. This area also notifies users and provides recovery instructions in case of a ransomware attack.

IT administrators and IT pros can get more configuration information from these articles:

- [Microsoft Defender Antivirus in the Windows Security app](#)
- [Microsoft Defender Antivirus documentation library](#)
- [Protect important folders with Controlled folder access](#)
- [Defend yourself from cybercrime with new Office 365 capabilities](#)
- [Microsoft Defender for Office 365](#)
- [Ransomware detection and recovering your files](#)

You can hide the **Virus & threat protection** section or the **Ransomware protection** area from users of the machine. This can be useful if you don't want employees in your organization to see or have access to user-configured options for these features.

Hide the Virus & threat protection section

You can choose to hide the entire section by using Group Policy. The section will not appear on the home page of the Windows Security app, and its icon will not be shown on the navigation bar on the side of the app.

This can only be done in Group Policy.

IMPORTANT

Requirements

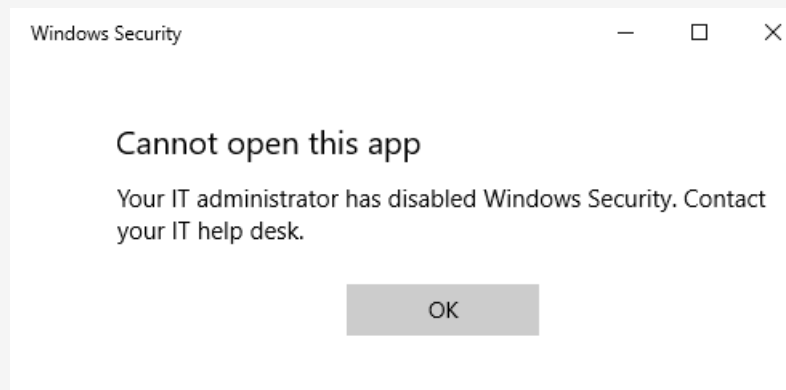
You must have Windows 10, version 1709 or later. The ADMX/ADML template files for earlier versions of Windows do not include these Group Policy settings.

1. On your Group Policy management machine, open the [Group Policy Management Console](#), right-click the Group Policy Object you want to configure and click **Edit**.
2. In the **Group Policy Management Editor** go to **Computer configuration** and click **Administrative templates**.
3. Expand the tree to **Windows components > Windows Security > Virus and threat protection**.
4. Open the **Hide the Virus and threat protection area** setting and set it to **Enabled**. Click **OK**.

5. [Deploy the updated GPO as you normally do.](#)

NOTE

If you hide all sections then the app will show a restricted interface, as in the following screenshot:



Hide the Ransomware protection area

You can choose to hide the **Ransomware protection** area by using Group Policy. The area will not appear on the **Virus & threat protection** section of the Windows Security app.

This can only be done in Group Policy.

IMPORTANT

Requirements

You must have Windows 10, version 1709 or later. The ADMX/ADML template files for earlier versions of Windows do not include these Group Policy settings.

1. On your Group Policy management machine, open the [Group Policy Management Console](#), right-click the Group Policy Object you want to configure and click **Edit**.
2. In the **Group Policy Management Editor** go to **Computer configuration** and click **Administrative templates**.
3. Expand the tree to **Windows components > Windows Security > Virus and threat protection**.
4. Open the **Hide the Ransomware data recovery area** setting and set it to **Enabled**. Click **OK**.
5. [Deploy the updated GPO as you normally do.](#)

Account protection

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11

The **Account protection** section contains information and settings for account protection and sign-in. You can get more information about these capabilities from the following list:

- [Microsoft Account](#)
- [Windows Hello for Business](#)
- [Lock your Windows 10 PC automatically when you step away from it](#)

You can also choose to hide the section from users of the device. This is useful if you don't want your employees to access or view user-configured options for these features.

Hide the Account protection section

You can choose to hide the entire section by using Group Policy. The section won't appear on the home page of the Windows Security app, and its icon won't be shown on the navigation bar on the side of the app.

You can only configure these settings by using Group Policy.

IMPORTANT

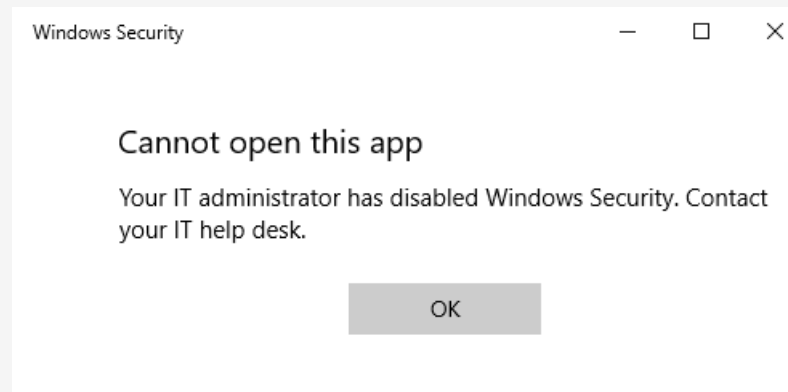
Requirements

You must have Windows 10, version 1803 or later. The ADMX/ADML template files for earlier versions of Windows do not include these Group Policy settings.

1. On your Group Policy management machine, open the [Group Policy Management Console](#), right-click the Group Policy Object you want to configure and select **Edit**.
2. In the **Group Policy Management Editor** go to **Computer configuration** and select **Administrative templates**.
3. Expand the tree to **Windows components > Windows Security > Account protection**.
4. Open the **Hide the Account protection area** setting and set it to **Enabled**. Select **OK**.
5. [Deploy the updated GPO as you normally do](#).

NOTE

If you hide all sections then the app will show a restricted interface, as in the following screenshot:



Firewall and network protection

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11

The **Firewall & network protection** section contains information about the firewalls and network connections used by the machine, including the status of Windows Defender Firewall and any other third-party firewalls. IT administrators and IT pros can get configuration guidance from the [Windows Defender Firewall with Advanced Security documentation library](#).

In Windows 10, version 1709 and later, the section can be hidden from users of the machine. This information is useful if you don't want employees in your organization to see or have access to user-configured options for the features shown in the section.

Hide the Firewall & network protection section

You can choose to hide the entire section by using Group Policy. The section will not appear on the home page of the Windows Security app, and its icon will not be shown on the navigation bar on the side of the app.

This can only be done in Group Policy.

IMPORTANT

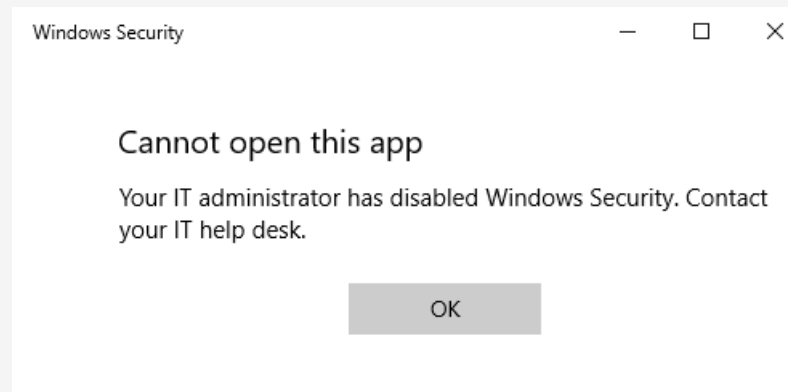
Requirements

You must have Windows 10, version 1709 or later. The ADMX/ADML template files for earlier versions of Windows do not include these Group Policy settings.

1. On your Group Policy management machine, open the Group Policy Management Console, right-click the Group Policy Object you want to configure and click **Edit**.
2. In the **Group Policy Management Editor** go to **Computer configuration** and click **Administrative templates**.
3. Expand the tree to **Windows components > Windows Security > Firewall and network protection**.
4. Open the **Hide the Firewall and network protection area** setting and set it to **Enabled**. Click **OK**.
5. Deploy the updated GPO as you normally do.

NOTE

If you hide all sections then the app will show a restricted interface, as in the following screenshot:



App and browser control

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11

The **App and browser control** section contains information and settings for Windows Defender SmartScreen. IT administrators and IT pros can get configuration guidance from the [Windows Defender SmartScreen documentation library](#).

In Windows 10, version 1709 and later, the section also provides configuration options for Exploit protection. You can prevent users from modifying these specific options with Group Policy. IT administrators can get more information at [Exploit protection](#).

You can also choose to hide the section from users of the machine. This can be useful if you don't want employees in your organization to see or have access to user-configured options for the features shown in the section.

Prevent users from making changes to the Exploit protection area in the App & browser control section

You can prevent users from modifying settings in the Exploit protection area. The settings will be either greyed out or not appear if you enable this setting. Users will still have access to other settings in the App & browser control section, such as those for Windows Defender SmartScreen, unless those options have been configured separately.

You can only prevent users from modifying Exploit protection settings by using Group Policy.

IMPORTANT

You must have Windows 10, version 1709 or later. The ADMX/ADML template files for earlier versions of Windows do not include these Group Policy settings.

1. On your Group Policy management machine, open the [Group Policy Management Console](#), right-click the Group Policy Object you want to configure and click **Edit**.
2. In the **Group Policy Management Editor** go to **Computer configuration**, select **Policies** and then **Administrative templates**.
3. Expand the tree to **Windows components > Windows Security > App and browser protection**.
4. Open the **Prevent users from modifying settings** setting and set it to **Enabled**. Click **OK**.
5. [Deploy the updated GPO as you normally do](#).

Hide the App & browser control section

You can choose to hide the entire section by using Group Policy. The section will not appear on the home page of the Windows Security app, and its icon will not be shown on the navigation bar on the side of the app.

This can only be done in Group Policy.

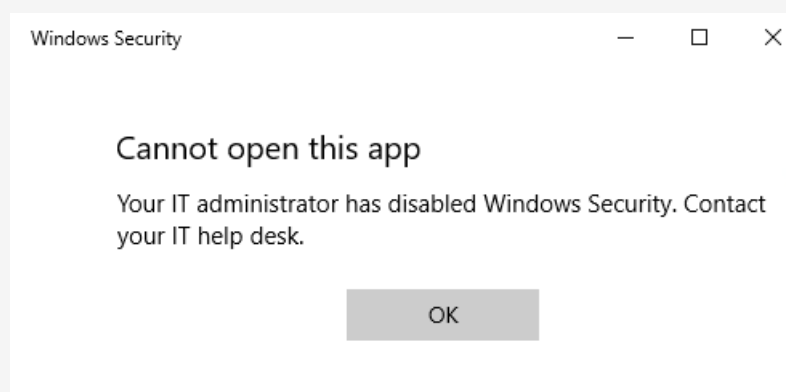
IMPORTANT

You must have Windows 10, version 1709 or later. The ADMX/ADML template files for earlier versions of Windows do not include these Group Policy settings.

1. On your Group Policy management machine, open the [Group Policy Management Console](#), right-click the Group Policy Object you want to configure and click **Edit**.
2. In the **Group Policy Management Editor** go to **Computer configuration**, select **Policies** and then **Administrative templates**.
3. Expand the tree to **Windows components > Windows Security > App and browser protection**.
4. Open the **Hide the App and browser protection area** setting and set it to **Enabled**. Click **OK**.
5. [Deploy the updated GPO as you normally do](#).

NOTE

If you hide all sections then the app will show a restricted interface, as in the following screenshot:



Device security

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11

The **Device security** section contains information and settings for built-in device security.

You can choose to hide the section from users of the machine. This can be useful if you don't want employees in your organization to see or have access to user-configured options for the features shown in the section.

Hide the Device security section

You can choose to hide the entire section by using Group Policy. The section will not appear on the home page of the Windows Security app, and its icon will not be shown on the navigation bar on the side of the app. You can hide the device security section by using Group Policy only.

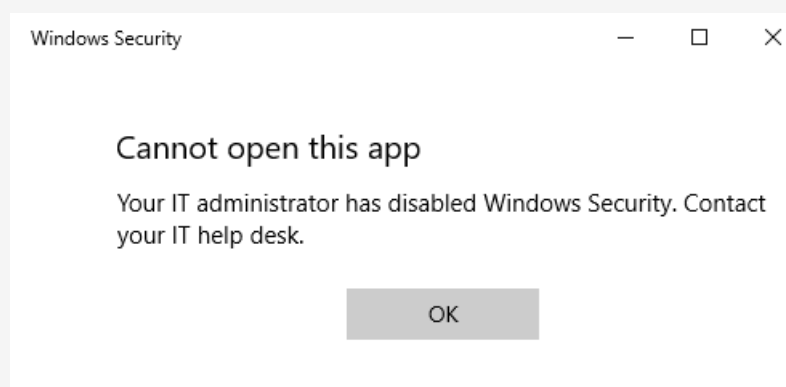
IMPORTANT

You must have Windows 10, version 1803 or later. The ADMX/ADML template files for earlier versions of Windows do not include these Group Policy settings.

1. On your Group Policy management machine, open the [Group Policy Management Console](#), right-click the Group Policy Object you want to configure and click **Edit**.
2. In the **Group Policy Management Editor** go to **Computer configuration** and then select **Administrative templates**.
3. Expand the tree to **Windows components > Windows Security > Device security**.
4. Open the **Hide the Device security area** setting and set it to **Enabled**. Select **OK**.
5. [Deploy the updated GPO as you normally do](#).

NOTE

If you hide all sections then the app will show a restricted interface, as in the following screenshot:



Disable the Clear TPM button

If you don't want users to be able to click the **Clear TPM** button in the Windows Security app, you can disable it.

IMPORTANT

You must have Windows 10, version 1809 or later. The ADMX/ADML template files for earlier versions of Windows do not include these Group Policy settings.

1. On your Group Policy management computer, open the [Group Policy Management Console](#), right-click the Group Policy Object you want to configure and click **Edit**.
2. In the **Group Policy Management Editor** go to **Computer configuration** and then select **Administrative templates**.
3. Expand the tree to **Windows components > Windows Security > Device security**.
4. Open the **Disable the Clear TPM button** setting and set it to **Enabled**. Select **OK**.
5. [Deploy the updated GPO as you normally do](#).

Hide the TPM Firmware Update recommendation

If you don't want users to see the recommendation to update TPM firmware, you can disable it.

1. On your Group Policy management computer, open the [Group Policy Management Console](#), right-click the Group Policy Object you want to configure and click **Edit**.
2. In the **Group Policy Management Editor** go to **Computer configuration** and then select **Administrative templates**.
3. Expand the tree to **Windows components > Windows Security > Device security**.
4. Open the **Hide the TPM Firmware Update recommendation** setting and set it to **Enabled**. Select **OK**.
5. [Deploy the updated GPO as you normally do](#).

Device performance and health

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11

The **Device performance & health** section contains information about hardware, devices, and drivers related to the machine. IT administrators and IT pros should reference the appropriate documentation library for the issues they are seeing, such as the [configure the Load and unload device drivers security policy setting](#) and how to [deploy drivers during Windows 10 deployment using Microsoft Endpoint Configuration Manager](#).

The [Windows 10 IT pro troubleshooting topic](#), and the main [Windows 10 documentation library](#) can also be helpful for resolving issues.

In Windows 10, version 1709 and later, the section can be hidden from users of the machine. This can be useful if you don't want employees in your organization to see or have access to user-configured options for the features shown in the section.

Hide the Device performance & health section

You can choose to hide the entire section by using Group Policy. The section will not appear on the home page of the Windows Security app, and its icon will not be shown on the navigation bar on the side of the app.

This can only be done in Group Policy.

IMPORTANT

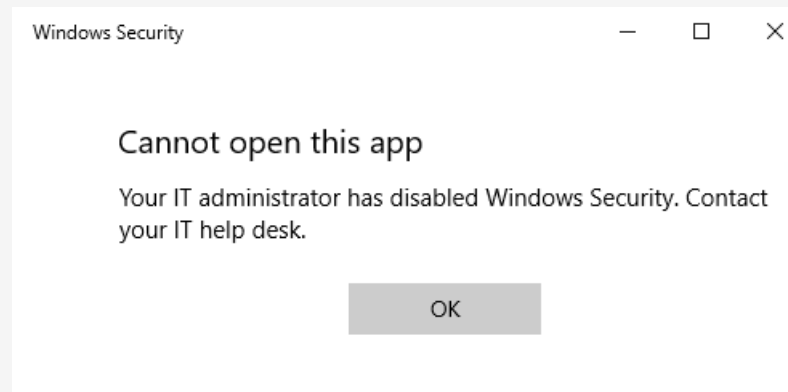
Requirements

You must have Windows 10, version 1709 or later. The ADMX/ADML template files for earlier versions of Windows do not include these Group Policy settings.

1. On your Group Policy management machine, open the [Group Policy Management Console](#), right-click the Group Policy Object you want to configure and click **Edit**.
2. In the **Group Policy Management Editor** go to **Computer configuration** and click **Administrative templates**.
3. Expand the tree to **Windows components > Windows Security > Device performance and health**.
4. Open the **Hide the Device performance and health area** setting and set it to **Enabled**. Click **OK**.
5. [Deploy the updated GPO as you normally do](#).

NOTE

If you hide all sections then the app will show a restricted interface, as in the following screenshot:



Family options

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11

The **Family options** section contains links to settings and further information for parents of a Windows 10 PC. It is not generally intended for enterprise or business environments.

Home users can learn more at the [Help protection your family online in Windows Security topic at support.microsoft.com](#)

In Windows 10, version 1709, the section can be hidden from users of the machine. This can be useful if you don't want employees in your organization to see or have access to this section.

Hide the Family options section

You can choose to hide the entire section by using Group Policy. The section will not appear on the home page of the Windows Security app, and its icon will not be shown on the navigation bar on the side of the app.

This can only be done in Group Policy.

IMPORTANT

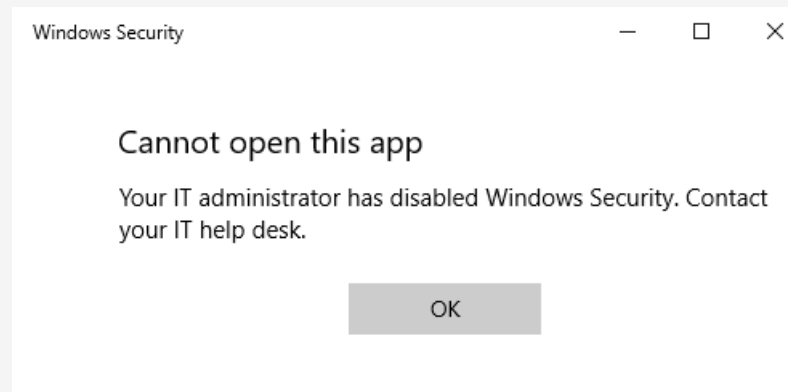
Requirements

You must have Windows 10, version 1709 or later. The ADMX/ADML template files for earlier versions of Windows do not include these Group Policy settings.

1. On your Group Policy management machine, open the [Group Policy Management Console](#), right-click the Group Policy Object you want to configure and click **Edit**.
2. In the **Group Policy Management Editor** go to **Computer configuration** and click **Administrative templates**.
3. Expand the tree to **Windows components > Windows Security > Family options**.
4. Open the **Hide the Family options area** setting and set it to **Enabled**. Click **OK**.
5. [Deploy the updated GPO as you normally do](#).

NOTE

If you hide all sections then the app will show a restricted interface, as in the following screenshot:



Security policy settings

7/1/2022 • 22 minutes to read • [Edit Online](#)

Applies to

- Windows 10

This reference topic describes the common scenarios, architecture, and processes for security settings.

Security policy settings are rules that administrators configure on a computer or multiple devices for the purpose of protecting resources on a device or network. The Security Settings extension of the Local Group Policy Editor snap-in allows you to define security configurations as part of a Group Policy Object (GPO). The GPOs are linked to Active Directory containers such as sites, domains, or organizational units, and they enable you to manage security settings for multiple devices from any device joined to the domain. Security settings policies are used as part of your overall security implementation to help secure domain controllers, servers, clients, and other resources in your organization.

Security settings can control:

- User authentication to a network or device.
- The resources that users are permitted to access.
- Whether to record a user's or group's actions in the event log.
- Membership in a group.

To manage security configurations for multiple devices, you can use one of the following options:

- Edit specific security settings in a GPO.
- Use the Security Templates snap-in to create a security template that contains the security policies you want to apply, and then import the security template into a Group Policy Object. A security template is a file that represents a security configuration, and it can be imported to a GPO, applied to a local device, or used to analyze security.

For more info about managing security configurations, see [Administer security policy settings](#).

The Security Settings extension of the Local Group Policy Editor includes the following types of security policies:

- **Account Policies.** These policies are defined on devices; they affect how user accounts can interact with the computer or domain. Account policies include the following types of policies:
 - **Password Policy.** These policies determine settings for passwords, such as enforcement and lifetimes. Password policies are used for domain accounts.
 - **Account Lockout Policy.** These policies determine the conditions and length of time that an account will be locked out of the system. Account lockout policies are used for domain or local user accounts.
 - **Kerberos Policy.** These policies are used for domain user accounts; they determine Kerberos-related settings, such as ticket lifetimes and enforcement.
- **Local Policies.** These policies apply to a computer and include the following types of policy settings:
 - **Audit Policy.** Specify security settings that control the logging of security events into the Security log on the computer, and specifies what types of security events to log (success, failure, or both).

NOTE

For devices running Windows 7 and later, we recommend to use the settings under Advanced Audit Policy Configuration rather than the Audit Policy settings under Local Policies.

- **User Rights Assignment.** Specify the users or groups that have logon rights or privileges on a device
- **Security Options.** Specify security settings for the computer, such as Administrator and Guest Account names; access to floppy disk drives and CD-ROM drives; installation of drivers; logon prompts; and so on.
- **Windows Firewall with Advanced Security.** Specify settings to protect the device on your network by using a stateful firewall that allows you to determine which network traffic is permitted to pass between your device and the network.
- **Network List Manager Policies.** Specify settings that you can use to configure different aspects of how networks are listed and displayed on one device or on many devices.
- **Public Key Policies.** Specify settings to control Encrypting File System, Data Protection, and BitLocker Drive Encryption in addition to certain certificate paths and services settings.
- **Software Restriction Policies.** Specify settings to identify software and to control its ability to run on your local device, organizational unit, domain, or site.
- **Application Control Policies.** Specify settings to control which users or groups can run particular applications in your organization based on unique identities of files.
- **IP Security Policies on Local Computer.** Specify settings to ensure private, secure communications over IP networks through the use of cryptographic security services. IPsec establishes trust and security from a source IP address to a destination IP address.
- **Advanced Audit Policy Configuration.** Specify settings that control the logging of security events into the security log on the device. The settings under Advanced Audit Policy Configuration provide finer control over which activities to monitor as opposed to the Audit Policy settings under Local Policies.

Policy-based security settings management

The Security Settings extension to Group Policy provides an integrated policy-based management infrastructure to help you manage and enforce your security policies.

You can define and apply security settings policies to users, groups, and network servers and clients through Group Policy and Active Directory Domain Services (AD DS). A group of servers with the same functionality can be created (for example, a Microsoft Web (IIS) server), and then Group Policy Objects can be used to apply common security settings to the group. If more servers are added to this group later, many of the common security settings are automatically applied, reducing deployment and administrative labor.

Common scenarios for using security settings policies

Security settings policies are used to manage the following aspects of security: accounts policy, local policy, user rights assignment, registry values, file and registry Access Control Lists (ACLs), service startup modes, and more.

As part of your security strategy, you can create GPOs with security settings policies configured specifically for the various roles in your organization, such as domain controllers, file servers, member servers, clients, and so on.

You can create an organizational unit (OU) structure that groups devices according to their roles. Using OUs is

the best method for separating specific security requirements for the different roles in your network. This approach also allows you to apply customized security templates to each class of server or computer. After creating the security templates, you create a new GPO for each of the OUs, and then import the security template (.inf file) into the new GPO.

Importing a security template to a GPO ensures that any accounts to which the GPO is applied automatically receive the template's security settings when the Group Policy settings are refreshed. On a workstation or server, the security settings are refreshed at regular intervals (with a random offset of at most 30 minutes), and, on a domain controller, this process occurs every few minutes if changes have occurred in any of the GPO settings that apply. The settings are also refreshed every 16 hours, whether or not any changes have occurred.

NOTE

These refresh settings vary between versions of the operating system and can be configured.

By using Group Policy–based security configurations in conjunction with the delegation of administration, you can ensure that specific security settings, rights, and behavior are applied to all servers and computers within an OU. This approach makes it simple to update a number of servers with any additional changes required in the future.

Dependencies on other operating system technologies

For devices that are members of a Windows Server 2008 or later domain, security settings policies depend on the following technologies:

- **Active Directory Domain Services (AD DS)**

The Windows-based directory service, AD DS, stores information about objects on a network and makes this information available to administrators and users. By using AD DS, you can view and manage network objects on the network from a single location, and users can access permitted network resources by using a single logon.

- **Group Policy**

The infrastructure within AD DS that enables directory-based configuration management of user and computer settings on devices running Windows Server. By using Group Policy, you can define configurations for groups of users and computers, including policy settings, registry-based policies, software installation, scripts, folder redirection, Remote Installation Services, Internet Explorer maintenance, and security.

- **Domain Name System (DNS)**

A hierarchical naming system used for locating domain names on the Internet and on private TCP/IP networks. DNS provides a service for mapping DNS domain names to IP addresses, and IP addresses to domain names. This allows users, computers, and applications to query DNS to specify remote systems by fully qualified domain names rather than by IP addresses.

- **Winlogon**

A part of the Windows operating system that provides interactive logon support. Winlogon is designed around an interactive logon model that consists of three components: the Winlogon executable, a credential provider, and any number of network providers.

- **Setup**

Security configuration interacts with the operating system setup process during a clean installation or upgrade from earlier versions of Windows Server.

- **Security Accounts Manager (SAM)**

A Windows service used during the logon process. SAM maintains user account information, including groups to which a user belongs.

- **Local Security Authority (LSA)**

A protected subsystem that authenticates and logs users onto the local system. LSA also maintains information about all aspects of local security on a system, collectively known as the Local Security Policy of the system.

- **Windows Management Instrumentation (WMI)**

A feature of the Microsoft Windows operating system, WMI is the Microsoft implementation of Web-Based Enterprise Management (WBEM), which is an industry initiative to develop a standard technology for accessing management information in an enterprise environment. WMI provides access to information about objects in a managed environment. Through WMI and the WMI application programming interface (API), applications can query for and make changes to static information in the Common Information Model (CIM) repository and dynamic information maintained by the various types of providers.

- **Resultant Set of Policy (RSOP)**

An enhanced Group Policy infrastructure that uses WMI in order to make it easier to plan and debug policy settings. RSOP provides public methods that expose what an extension to Group Policy would do in a what-if situation, and what the extension has done in an actual situation. This allows administrators to easily determine the combination of policy settings that apply to, or will apply to, a user or device.

- **Service Control Manager (SCM)**

Used for configuration of service startup modes and security.

- **Registry**

Used for configuration of registry values and security.

- **File system**

Used for configuration of security.

- **File system conversions**

Security is set when an administrator converts a file system from FAT to NTFS.

- **Microsoft Management Console (MMC)**

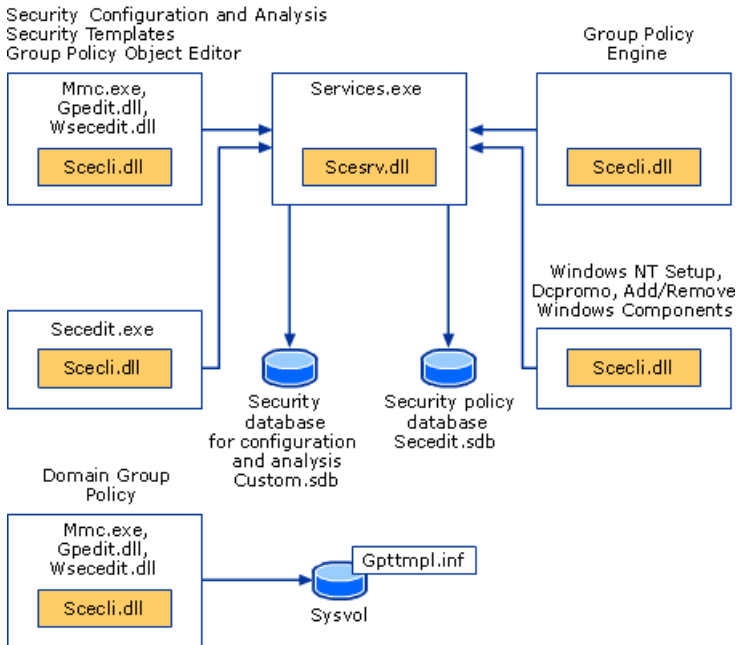
The user interface for the Security Settings tool is an extension of the Local Group Policy Editor MMC snap-in.

Security settings policies and Group Policy

The Security Settings extension of the Local Group Policy Editor is part of the Security Configuration Manager tool set. The following components are associated with Security Settings: a configuration engine; an analysis engine; a template and database interface layer; setup integration logic; and the secdit.exe command-line tool. The security configuration engine is responsible for handling security configuration editor-related security requests for the system on which it runs. The analysis engine analyzes system security for a given configuration and saves the result. The template and database interface layer handles reading and writing requests from and to the template or database (for internal storage). The Security Settings extension of the Local Group Policy Editor handles Group Policy from a domain-based or local device. The security configuration logic integrates with setup and manages system security for a clean installation or upgrade to a more recent Windows operating system. Security information is stored in templates (.inf files) or in the Secedit.sdb database.

The following diagram shows Security Settings and related features.

Security Settings Policies and Related Features



- **Scesrv.dll**

Provides the core security engine functionality.

- **Scecli.dll**

Provides the client-side interfaces to the security configuration engine and provides data to Resultant Set of Policy (RSOP).

- **Wsecedit.dll**

The Security Settings extension of Local Group Policy Editor. scecli.dll is loaded into wsecedit.dll to support the Security Settings user interface.

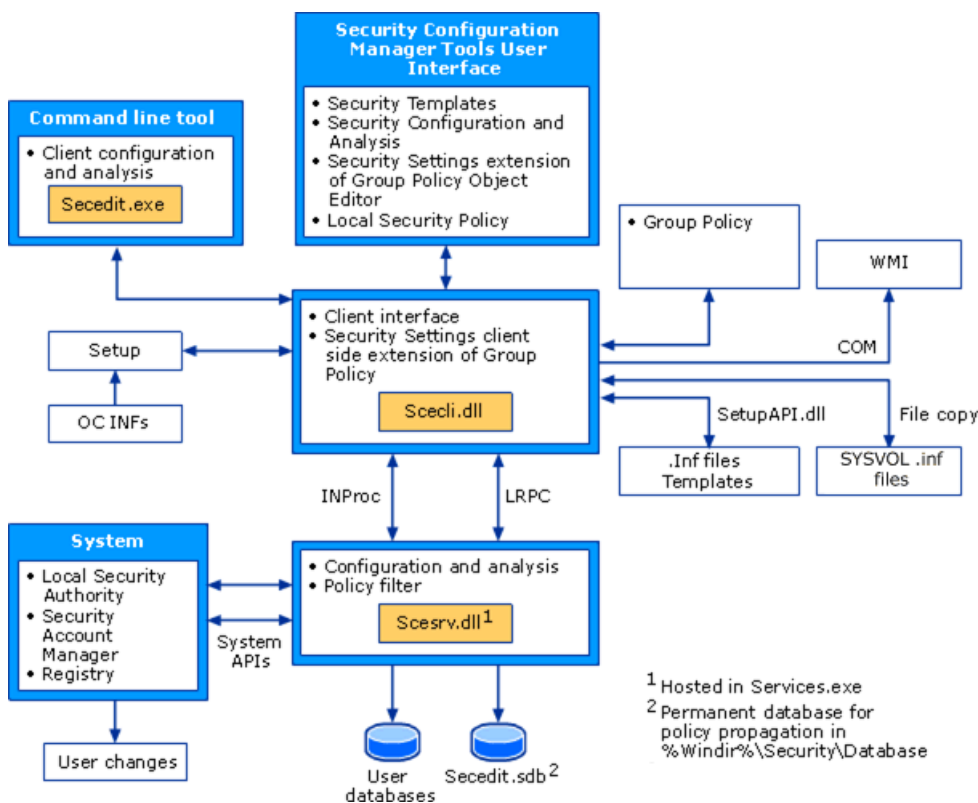
- **Gpedit.dll**

The Local Group Policy Editor MMC snap-in.

Security Settings extension architecture

The Security Settings extension of the Local Group Policy Editor is part of the Security Configuration Manager tools, as shown in the following diagram.

Security Settings Architecture



The security settings configuration and analysis tools include a security configuration engine, which provides local computer (non-domain member) and Group Policy–based configuration and analysis of security settings policies. The security configuration engine also supports the creation of security policy files. The primary features of the security configuration engine are scecli.dll and scesrv.dll.

The following list describes these primary features of the security configuration engine and other Security Settings–related features.

- **scesrv.dll**

This .dll is hosted in services.exe and runs under local system context. scesrv.dll provides core Security Configuration Manager functionality, such as import, configure, analyze, and policy propagation.

Scesrv.dll performs configuration and analysis of various security-related system parameters by calling corresponding system APIs, including LSA, SAM, and the registry.

Scesrv.dll exposes APIs such as import, export, configure, and analyze. It checks that the request is made over LRPC (Windows XP) and fails the call if it is not.

Communication between parts of the Security Settings extension occurs by using the following methods:

- Component Object Model (COM) calls
- Local Remote Procedure Call (LRPC)
- Lightweight Directory Access Protocol (LDAP)
- Active Directory Service Interfaces (ADSI)
- Server Message Block (SMB)
- Win32 APIs
- Windows Management Instrumentation (WMI) calls

On domain controllers, scesrv.dll receives notifications of changes made to SAM and the LSA that need to be synchronized across domain controllers. Scesrv.dll incorporates those changes into the Default Domain Controller Policy GPO by using in-process scecli.dll template modification APIs. Scesrv.dll also performs configuration and analysis operations.

- **Scecli.dll**

This is the client-side interface or wrapper to scesrv.dll. scecli.dll is loaded into Wsecedit.dll to support MMC snap-ins. It is used by Setup to configure default system security and security of files, registry keys, and services installed by the Setup API .inf files.

The command-line version of the security configuration and analysis user interfaces, secedit.exe, uses scecli.dll.

Scecli.dll implements the client-side extension for Group Policy.

Scesrv.dll uses scecli.dll to download applicable Group Policy files from SYSVOL in order to apply Group Policy security settings to the local device.

Scecli.dll logs application of security policy into WMI (RSOP).

Scesrv.dll policy filter uses scecli.dll to update Default Domain Controller Policy GPO when changes are made to SAM and LSA.

- **Wsecedit.dll**

The Security Settings extension of the Group Policy Object Editor snap-in. You use this tool to configure security settings in a Group Policy Object for a site, domain, or organizational unit. You can also use Security Settings to import security templates to a GPO.

- **Secedit.sdb**

This is a permanent system database used for policy propagation including a table of persistent settings for rollback purposes.

- **User databases**

A user database is any database other than the system database created by administrators for the purposes of configuration or analysis of security.

- **.Inf Templates**

These are text files that contain declarative security settings. They are loaded into a database before configuration or analysis. Group Policy security policies are stored in .inf files on the SYSVOL folder of domain controllers, where they are downloaded (by using file copy) and merged into the system database during policy propagation.

Security settings policy processes and interactions

For a domain-joined device, where Group Policy is administered, security settings are processed in conjunction with Group Policy. Not all settings are configurable.

Group Policy processing

When a computer starts and a user logs on, computer policy and user policy are applied according to the following sequence:

1. The network starts. Remote Procedure Call System Service (RPCSS) and Multiple Universal Naming Convention Provider (MUP) start.
2. An ordered list of Group Policy Objects is obtained for the device. The list might depend on these factors:
 - Whether the device is part of a domain and, therefore, subject to Group Policy through Active Directory.
 - The location of the device in Active Directory.
 - Whether the list of Group Policy Objects has changed. If the list of Group Policy Objects has not changed, no processing is done.

3. Computer policy is applied. These are the settings under Computer Configuration from the gathered list. This is a synchronous process by default and occurs in the following order: local, site, domain, organizational unit, child organizational unit, and so on. No user interface appears while computer policies are processed.
4. Startup scripts run. This is hidden and synchronous by default; each script must complete or time out before the next one starts. The default time-out is 600 seconds. You can use several policy settings to modify this behavior.
5. The user presses CTRL+ALT+DEL to log on.
6. After the user is validated, the user profile loads; it is governed by the policy settings that are in effect.
7. An ordered list of Group Policy Objects is obtained for the user. The list might depend on these factors:
 - Whether the user is part of a domain and, therefore, subject to Group Policy through Active Directory.
 - Whether loopback policy processing is enabled, and if so, the state (Merge or Replace) of the loopback policy setting.
 - The location of the user in Active Directory.
 - Whether the list of Group Policy Objects has changed. If the list of Group Policy Objects has not changed, no processing is done.
8. User policy is applied. These are the settings under User Configuration from the gathered list. This is synchronous by default and in the following order: local, site, domain, organizational unit, child organizational unit, and so on. No user interface appears while user policies are processed.
9. Logon scripts run. Group Policy–based logon scripts are hidden and asynchronous by default. The user object script runs last.
10. The operating system user interface that is prescribed by Group Policy appears.

Group Policy Objects storage

A Group Policy Object (GPO) is a virtual object that is identified by a Globally Unique Identifier (GUID) and stored at the domain level. The policy setting information of a GPO is stored in the following two locations:

- **Group Policy containers in Active Directory.**

The Group Policy container is an Active Directory container that contains GPO properties, such as version information, GPO status, plus a list of other component settings.

- **Group Policy templates in a domain's system volume folder (SYSVOL).**

The Group Policy template is a file system folder that includes policy data specified by .admx files, security settings, script files, and information about applications that are available for installation. The Group Policy template is located in the SYSVOL folder in the <domain>\Policies subfolder.

The `GROUP_POLICY_OBJECT` structure provides information about a GPO in a GPO list, including the version number of the GPO, a pointer to a string that indicates the Active Directory portion of the GPO, and a pointer to a string that specifies the path to the file system portion of the GPO.

Group Policy processing order

Group Policy settings are processed in the following order:

1. **Local Group Policy Object.**

Each device running a Windows operating system beginning with Windows XP has exactly one Group Policy Object that is stored locally.

2. **Site.**

Any Group Policy Objects that have been linked to the site are processed next. Processing is synchronous and in an order that you specify.

3. Domain.

Processing of multiple domain-linked Group Policy Objects is synchronous and in an order you specify.

4. Organizational units.

Group Policy Objects that are linked to the organizational unit that is highest in the Active Directory hierarchy are processed first, then Group Policy Objects that are linked to its child organizational unit, and so on. Finally, the Group Policy Objects that are linked to the organizational unit that contains the user or device are processed.

At the level of each organizational unit in the Active Directory hierarchy, one, many, or no Group Policy Objects can be linked. If several Group Policy Objects are linked to an organizational unit, their processing is synchronous and in an order that you specify.

This order means that the local Group Policy Object is processed first, and Group Policy Objects that are linked to the organizational unit of which the computer or user is a direct member are processed last, which overwrites the earlier Group Policy Objects.

This is the default processing order and administrators can specify exceptions to this order. A Group Policy Object that is linked to a site, domain, or organizational unit (not a local Group Policy Object) can be set to **Enforced** with respect to that site, domain, or organizational unit, so that none of its policy settings can be overridden. At any site, domain, or organizational unit, you can mark Group Policy inheritance selectively as **Block Inheritance**. Group Policy Object links that are set to **Enforced** are always applied, however, and they cannot be blocked. For more information see [Group Policy Basics – Part 2: Understanding Which GPOs to Apply](#).

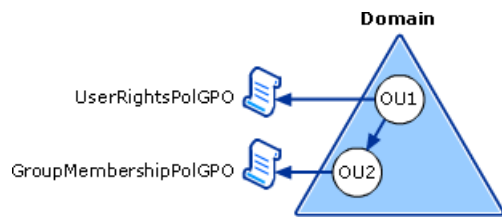
Security settings policy processing

In the context of Group Policy processing, security settings policy is processed in the following order.

1. During Group Policy processing, the Group Policy engine determines which security settings policies to apply.
2. If security settings policies exist in a GPO, Group Policy invokes the Security Settings client-side extension.
3. The Security Settings extension downloads the policy from the appropriate location such as a specific domain controller.
4. The Security Settings extension merges all security settings policies according to precedence rules. The processing is according to the Group Policy processing order of local, site, domain, and organizational unit (OU), as described earlier in the "Group Policy processing order" section. If multiple GPOs are in effect for a given device and there are no conflicting policies, then the policies are cumulative and are merged.

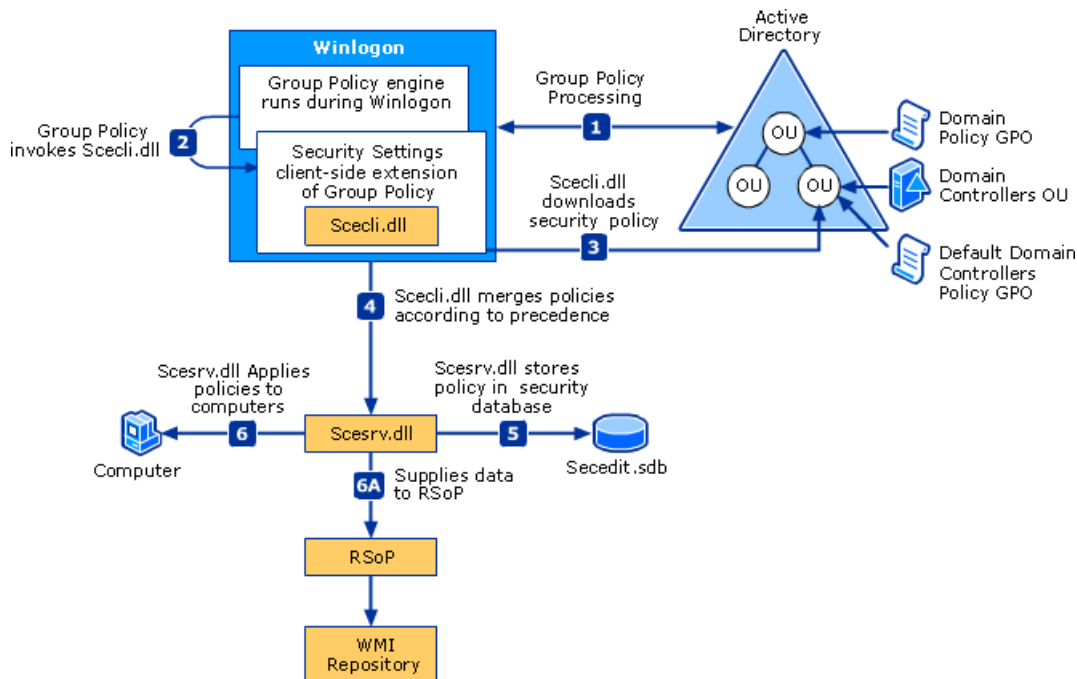
This example uses the Active Directory structure shown in the following figure. A given computer is a member of OU2, to which the **GroupMembershipPolGPO** GPO is linked. This computer is also subject to the **UserRightsPolGPO** GPO, which is linked to OU1, higher in the hierarchy. In this case, no conflicting policies exist so the device receives all of the policies contained in both the **UserRightsPolGPO** and the **GroupMembershipPolGPO** GPOs.

Multiple GPOs and Merging of Security Policy



- The resultant security policies are stored in `secdit.sdb`, the security settings database. The security engine gets the security template files and imports them to `secdit.sdb`.
- The security settings policies are applied to devices. The following figure illustrates the security settings policy processing.

Security Settings Policy Processing



Merging of security policies on domain controllers

Password policies, Kerberos, and some security options are only merged from GPOs that are linked at the root level on the domain. This is done to keep those settings synchronized across all domain controllers in the domain. The following security options are merged:

- Network Security: Force logoff when logon hours expire
- Accounts: Administrator account status
- Accounts: Guest account status
- Accounts: Rename administrator account
- Accounts: Rename guest account

Another mechanism exists that allows security policy changes made by administrators by using net accounts to be merged into the Default Domain Policy GPO. User rights changes that are made by using Local Security Authority (LSA) APIs are filtered into the Default Domain Controllers Policy GPO.

Special considerations for domain controllers

If an application is installed on a primary domain controller (PDC) with operations master role (also known as flexible single master operations or FSMO) and the application makes changes to user rights or password policy, these changes must be communicated to ensure that synchronization across domain controllers occurs. `Scesrv.dll` receives a notification of any changes made to the security account manager (SAM) and LSA that need to be synchronized across domain controllers and then incorporates the changes into the Default Domain Controller Policy GPO by using `scecli.dll` template modification APIs.

When security settings are applied

After you have edited the security settings policies, the settings are refreshed on the computers in the organizational unit linked to your Group Policy Object in the following instances:

- When a device is restarted.
- Every 90 minutes on a workstation or server and every 5 minutes on a domain controller. This refresh interval is configurable.
- By default, Security policy settings delivered by Group Policy are also applied every 16 hours (960 minutes) even if a GPO has not changed.

Persistence of security settings policy

Security settings can persist even if a setting is no longer defined in the policy that originally applied it.

Security settings might persist in the following cases:

- The setting has not been previously defined for the device.
- The setting is for a registry security object.
- The settings are for a file system security object.

All settings applied through local policy or through a Group Policy Object are stored in a local database on your computer. Whenever a security setting is modified, the computer saves the security setting value to the local database, which retains a history of all the settings that have been applied to the computer. If a policy first defines a security setting and then no longer defines that setting, then the setting takes on the previous value in the database. If a previous value does not exist in the database then the setting does not revert to anything and remains defined as is. This behavior is sometimes referred to as "tattooing".

Registry and file security settings will maintain the values applied through Group Policy until that setting is set to other values.

Permissions required for policy to apply

Both Apply Group Policy and Read permissions are required to have the settings from a Group Policy Object apply to users or groups, and computers.

Filtering security policy

By default, all GPOs have Read and Apply Group Policy both Allowed for the Authenticated Users group. The Authenticated Users group includes both users and computers. Security settings policies are computer-based. To specify which client computers will or will not have a Group Policy Object applied to them, you can deny them either the Apply Group Policy or Read permission on that Group Policy Object. Changing these permissions allows you to limit the scope of the GPO to a specific set of computers within a site, domain, or OU.

NOTE

Do not use security policy filtering on a domain controller as this would prevent security policy from applying to it.

Migration of GPOs containing security settings

In some situations, you might want to migrate GPOs from one domain environment to another environment. The two most common scenarios are test-to-production migration, and production-to-production migration. The GPO copying process has implications for some types of security settings.

Data for a single GPO is stored in multiple locations and in various formats; some data is contained in Active Directory and other data is stored on the SYSVOL share on the domain controllers. Certain policy data might be valid in one domain but might be invalid in the domain to which the GPO is being copied. For example, Security Identifiers (SIDs) stored in security policy settings are often domain-specific. So copying GPOs is not as simple as taking a folder and copying it from one device to another.

The following security policies can contain security principals and might require some additional work to successfully move them from one domain to another.

- User rights assignment
- Restricted groups
- Services
- File system
- Registry
- The GPO DACL, if you choose to preserve it during a copy operation

To ensure that data is copied correctly, you can use Group Policy Management Console (GPMC). When migrating a GPO from one domain to another, GPMC ensures that all relevant data is properly copied. GPMC also offers migration tables, which can be used to update domain-specific data to new values as part of the migration process. GPMC hides much of the complexity involved in the migrating GPO operations, and it provides simple and reliable mechanisms for performing operations such as copy and backup of GPOs.

In this section

TOPIC	DESCRIPTION
Administer security policy settings	This article discusses different methods to administer security policy settings on a local device or throughout a small- or medium-sized organization.
Configure security policy settings	Describes steps to configure a security policy setting on the local device, on a domain-joined device, and on a domain controller.
Security policy settings reference	This reference of security settings provides information about how to implement and manage security policies, including setting options and security considerations.

Security auditing

7/1/2022 • 2 minutes to read • [Edit Online](#)

Topics in this section are for IT professionals and describes the security auditing features in Windows and how your organization can benefit from using these technologies to enhance the security and manageability of your network.

Security auditing is one of the most powerful tools that you can use to maintain the integrity of your system. As part of your overall security strategy, you should determine the level of auditing that is appropriate for your environment. Auditing should identify attacks (successful or not) that pose a threat to your network, and attacks against resources that you have determined to be valuable in your risk assessment.

In this section

TOPIC	DESCRIPTION
Basic security audit policies	Before you implement auditing, you must decide on an auditing policy. A basic audit policy specifies categories of security-related events that you want to audit. When this version of Windows is first installed, all auditing categories are disabled. By enabling various auditing event categories, you can implement an auditing policy that suits the security needs of your organization.
Advanced security audit policies	Advanced security audit policy settings are found in Security Settings\Advanced Audit Policy Configuration\System Audit Policies and appear to overlap with basic security audit policies, but they are recorded and applied differently.

Encryption and data protection in Windows client

7/1/2022 • 2 minutes to read • [Edit Online](#)

When people travel with their computers and devices, their confidential information travels with them. Wherever confidential data is stored, it must be protected against unauthorized access, whether through physical device theft or from malicious applications. Encryption and data protection features include:

- Encrypted Hard Drive
- BitLocker

Encrypted Hard Drive

Encrypted Hard Drive uses the rapid encryption provided by BitLocker Drive Encryption to enhance data security and management. By offloading the cryptographic operations to hardware, encrypted hard drives increase BitLocker performance and reduce CPU usage and power consumption. Because encrypted hard drives encrypt data quickly, enterprise devices can expand BitLocker deployment with minimal impact on productivity.

Encrypted hard drives provide:

- Better performance: Encryption hardware, integrated into the drive controller, allows the drive to operate at full data rate with no performance degradation.
- Strong security based in hardware: Encryption is always "on" and the keys for encryption never leave the hard drive. User authentication is performed by the drive before it will unlock, independently of the operating system.
- Ease of use: Encryption is transparent to the user, and the user does not need to enable it. Encrypted hard drives are easily erased using on-board encryption key; there is no need to re-encrypt data on the drive.
- Lower cost of ownership: There is no need for new infrastructure to manage encryption keys, since BitLocker uses your existing infrastructure to store recovery information. Your device operates more efficiently because processor cycles do not need to be used for the encryption process.

Encrypted hard drives are a new class of hard drives that are self-encrypted at a hardware level and allow for full disk hardware encryption.

BitLocker

BitLocker Drive Encryption is a data protection feature that integrates with the operating system and addresses the threats of data theft or exposure from lost, stolen, or inappropriately decommissioned computers.

BitLocker provides encryption for the operating system, fixed data, and removable data drives, using technologies like hardware security test interface (HSTI), Modern Standby, UEFI Secure Boot, and TPM.

Windows consistently improves data protection by improving existing options and providing new strategies.

See also

- [Encrypted Hard Drive](#)
- [BitLocker](#)

Encrypted Hard Drive

7/1/2022 • 6 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2022
- Windows Server 2019
- Windows Server 2016
- Azure Stack HCI

Encrypted hard drive uses the rapid encryption that is provided by BitLocker drive encryption to enhance data security and management.

By offloading the cryptographic operations to a hardware, Encrypted hard drives increase BitLocker performance and reduce CPU usage and power consumption. Because Encrypted hard drives encrypt data quickly, enterprise devices can expand BitLocker deployment with minimal impact on productivity.

Encrypted hard drives are a new class of hard drives that are self-encrypting at a hardware level and allow for full disk hardware encryption. You can install Windows to encrypted hard drives without additional modification, beginning with Windows 8 and Windows Server 2012.

Encrypted hard drives provide:

- **Better performance:** Encryption hardware, integrated into the drive controller, allows the drive to operate at full data rate with no performance degradation.
- **Strong security based in hardware:** Encryption is always "on" and the keys for encryption never leave the hard drive. User authentication is performed by the drive before it will unlock, independently of the operating system
- **Ease of use:** Encryption is transparent to the user, and the user doesn't need to enable it. Encrypted Hard Drives are easily erased using on-board encryption key; there's no need to re-encrypt data on the drive.
- **Lower cost of ownership:** There's no need for new infrastructure to manage encryption keys, since BitLocker leverages your existing infrastructure to store recovery information. Your device operates more efficiently because processor cycles don't need to be used for the encryption process.

Encrypted hard drives are supported natively in the operating system through the following mechanisms:

- **Identification:** The operating system identifies that the drive is an Encrypted hard drive device type.
- **Activation:** The operating system disk management utility activates, creates and maps volumes to ranges/bands as appropriate.
- **Configuration:** The operating system creates and maps volumes to ranges/bands as appropriate.
- **API:** API support for applications to manage Encrypted hard drives independent of BitLocker drive encryption (BDE).
- **BitLocker support:** Integration with the BitLocker Control Panel provides a seamless BitLocker end-user experience.

WARNING

Self-encrypting hard drives and encrypted hard drives for Windows are not the same type of devices. Encrypted hard drives for Windows require compliance for specific TCG protocols as well as IEEE 1667 compliance; Self-encrypting hard drives do not have these requirements. It is important to confirm that the device type is an encrypted hard drive for Windows when planning for deployment.

If you are a storage device vendor who is looking for more info on how to implement Encrypted Hard Drive, see the [Encrypted Hard Drive Device Guide](#).

System Requirements

To use encrypted hard drives, the following system requirements apply:

For an encrypted hard drive used as a **data drive**:

- The drive must be in an uninitialized state.
- The drive must be in a security inactive state.

For an encrypted hard drive used as a **startup drive**:

- The drive must be in an uninitialized state.
- The drive must be in a security inactive state.
- The computer must be UEFI 2.3.1 based and have the `EFI_STORAGE_SECURITY_COMMAND_PROTOCOL` defined. (This protocol is used to allow programs running in the EFI boot services environment to send security protocol commands to the drive).
- The computer must have the compatibility support module (CSM) disabled in UEFI.
- The computer must always boot natively from UEFI.

WARNING

All encrypted hard drives must be attached to non-RAID controllers to function properly.

Technical overview

Rapid encryption in BitLocker directly addresses the security needs of enterprises while offering significantly improved performance. In versions of Windows earlier than Windows Server 2012, BitLocker required a two-step process to complete read/write requests. In Windows Server 2012, Windows 8, or later versions, encrypted hard drives offload the cryptographic operations to the drive controller for much greater efficiency. When the operating system identifies an encrypted hard drive, it activates the security mode. This activation lets the drive controller generate a media key for every volume that the host computer creates. This media key, which is never exposed outside the disk, is used to rapidly encrypt or decrypt every byte of data that is sent or received from the disk.

Configuring encrypted hard drives as startup drives

Configuration of encrypted hard drives as startup drives is done using the same methods as standard hard drives. These methods include:

- **Deploy from media:** Configuration of Encrypted Hard Drives happens automatically through the installation process.
- **Deploy from network:** This deployment method involves booting a Windows PE environment and using imaging tools to apply a Windows image from a network share. Using this method, the Enhanced Storage

optional component needs to be included in the Windows PE image. You can enable this component using Server Manager, Windows PowerShell, or the DISM command line tool. If this component isn't present, configuration of Encrypted Hard Drives won't work.

- **Deploy from server:** This deployment method involves PXE booting a client with Encrypted Hard Drives present. Configuration of Encrypted Hard Drives happens automatically in this environment when the Enhanced Storage component is added to the PXE boot image. During deployment, the [TCGSecurityActivationDisabled](#) setting in unattend.xml controls the encryption behavior of Encrypted Hard Drives.
- **Disk Duplication:** This deployment method involves use of a previously configured device and disk duplication tools to apply a Windows image to an Encrypted Hard Drive. Disks must be partitioned using at least Windows 8 or Windows Server 2012 for this configuration to work. Images made using disk duplicators won't work.

Configuring hardware-based encryption with group policy

There are three related Group Policy settings that help you manage how BitLocker uses hardware-based encryption and which encryption algorithms to use. If these settings aren't configured or disabled on systems that are equipped with encrypted drives, BitLocker uses software-based encryption:

- [Configure use of hardware-based encryption for fixed data drives](#)
- [Configure use of hardware-based encryption for removable data drives](#)
- [Configure use of hardware-based encryption for operating system drives](#)

Encrypted hard drive architecture

Encrypted hard drives utilize two encryption keys on the device to control the locking and unlocking of data on the drive. These are the data encryption key (DEK) and the authentication key (AK).

The Data Encryption Key is the key used to encrypt all of the data on the drive. The drive generates the DEK and it never leaves the device. It's stored in an encrypted format at a random location on the drive. If the DEK is changed or erased, data encrypted using the DEK is irrecoverable.

The AK is the key used to unlock data on the drive. A hash of the key is stored on the drive and requires confirmation to decrypt the DEK.

When a computer with an encrypted hard drive is in a powered-off state, the drive locks automatically. As a computer powers on, the device remains in a locked state and is only unlocked after the AK decrypts the DEK. Once the AK decrypts the DEK, read-write operations can take place on the device.

When writing data to the drive, it passes through an encryption engine before the write operation completes. Likewise, reading data from the drive requires the encryption engine to decrypt the data before passing that data back to the user. In the event that the DEK needs to be changed or erased, the data on the drive doesn't need to be re-encrypted. A new Authentication Key needs to be created and it will re-encrypt the DEK. Once completed, the DEK can now be unlocked using the new AK and read-writes to the volume can continue.

Re-configuring encrypted hard drives

Many encrypted hard drive devices come pre-configured for use. If reconfiguration of the drive is required, use the following procedure after removing all available volumes and reverting the drive to an uninitialized state:

1. Open Disk Management (diskmgmt.msc)
2. Initialize the disk and select the appropriate partition style (MBR or GPT)
3. Create one or more volumes on the disk.
4. Use the BitLocker setup wizard to enable BitLocker on the volume.

BitLocker

7/1/2022 • 7 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016 and above

This topic provides a high-level overview of BitLocker, including a list of system requirements, practical applications, and deprecated features.

BitLocker overview

BitLocker Drive Encryption is a data protection feature that integrates with the operating system and addresses the threats of data theft or exposure from lost, stolen, or inappropriately decommissioned computers.

BitLocker provides the most protection when used with a Trusted Platform Module (TPM) version 1.2 or later. The TPM is a hardware component installed in many newer computers by the computer manufacturers. It works with BitLocker to help protect user data and to ensure that a computer has not been tampered with while the system was offline.

On computers that do not have a TPM version 1.2 or later, you can still use BitLocker to encrypt the Windows operating system drive. However, this implementation will require the user to insert a USB startup key to start the computer or resume from hibernation. Starting with Windows 8, you can use an operating system volume password to protect the operating system volume on a computer without TPM. Both options do not provide the pre-startup system integrity verification offered by BitLocker with a TPM.

In addition to the TPM, BitLocker offers the option to lock the normal startup process until the user supplies a personal identification number (PIN) or inserts a removable device, such as a USB flash drive, that contains a startup key. These additional security measures provide multifactor authentication and assurance that the computer will not start or resume from hibernation until the correct PIN or startup key is presented.

Practical applications

Data on a lost or stolen computer is vulnerable to unauthorized access, either by running a software-attack tool against it or by transferring the computer's hard disk to a different computer. BitLocker helps mitigate unauthorized data access by enhancing file and system protections. BitLocker also helps render data inaccessible when BitLocker-protected computers are decommissioned or recycled.

There are two additional tools in the Remote Server Administration Tools, which you can use to manage BitLocker.

- **BitLocker Recovery Password Viewer.** The BitLocker Recovery Password Viewer enables you to locate and view BitLocker Drive Encryption recovery passwords that have been backed up to Active Directory Domain Services (AD DS). You can use this tool to help recover data that is stored on a drive that has been encrypted by using BitLocker. The BitLocker Recovery Password Viewer tool is an extension for the Active Directory Users and Computers Microsoft Management Console (MMC) snap-in. By using this tool, you can examine a computer object's **Properties** dialog box to view the corresponding BitLocker recovery passwords. Additionally, you can right-click a domain container and then search for a BitLocker recovery password across all the domains in the Active Directory forest. To view recovery passwords, you must be a domain administrator, or you must have been delegated permissions by a domain

administrator.

- **BitLocker Drive Encryption Tools.** BitLocker Drive Encryption Tools include the command-line tools, manage-bde and repair-bde, and the BitLocker cmdlets for Windows PowerShell. Both manage-bde and the BitLocker cmdlets can be used to perform any task that can be accomplished through the BitLocker control panel, and they are appropriate to use for automated deployments and other scripting scenarios. Repair-bde is provided for disaster recovery scenarios in which a BitLocker protected drive cannot be unlocked normally or by using the recovery console.

New and changed functionality

To find out what's new in BitLocker for Windows, such as support for the XTS-AES encryption algorithm, see the [BitLocker](#) section in "What's new in Windows 10."

System requirements

BitLocker has the following hardware requirements:

For BitLocker to use the system integrity check provided by a Trusted Platform Module (TPM), the computer must have TPM 1.2 or later. If your computer does not have a TPM, enabling BitLocker requires that you save a startup key on a removable device, such as a USB flash drive.

A computer with a TPM must also have a Trusted Computing Group (TCG)-compliant BIOS or UEFI firmware. The BIOS or UEFI firmware establishes a chain of trust for the pre-operating system startup, and it must include support for TCG-specified Static Root of Trust Measurement. A computer without a TPM does not require TCG-compliant firmware.

The system BIOS or UEFI firmware (for TPM and non-TPM computers) must support the USB mass storage device class, including reading small files on a USB flash drive in the pre-operating system environment.

IMPORTANT

From Windows 7, you can encrypt an OS drive without a TPM and USB flash drive. For this procedure, see [Tip of the Day: Bitlocker without TPM or USB](#).

NOTE

TPM 2.0 is not supported in Legacy and CSM Modes of the BIOS. Devices with TPM 2.0 must have their BIOS mode configured as Native UEFI only. The Legacy and Compatibility Support Module (CSM) options must be disabled. For added security Enable the Secure Boot feature.

Installed Operating System on hardware in legacy mode will stop the OS from booting when the BIOS mode is changed to UEFI. Use the tool [MBR2GPT](#) before changing the BIOS mode which will prepare the OS and the disk to support UEFI.

The hard disk must be partitioned with at least two drives:

- The operating system drive (or boot drive) contains the operating system and its support files. It must be formatted with the NTFS file system.
- The system drive contains the files that are needed to load Windows after the firmware has prepared the system hardware. BitLocker is not enabled on this drive. For BitLocker to work, the system drive must not be encrypted, must differ from the operating system drive, and must be formatted with the FAT32 file system on computers that use UEFI-based firmware or with the NTFS file system on computers that use BIOS firmware. We recommend that system drive be approximately 350 MB in size. After BitLocker is turned on it should have approximately 250 MB of free space.

A partition subject to encryption cannot be marked as an active partition (this applies to the operating system, fixed data, and removable data drives).

When installed on a new computer, Windows will automatically create the partitions that are required for BitLocker.

When installing the BitLocker optional component on a server you will also need to install the Enhanced Storage feature, which is used to support hardware encrypted drives.

In this section

TOPIC	DESCRIPTION
Overview of BitLocker Device Encryption in Windows	This topic for the IT professional provides an overview of the ways that BitLocker Device Encryption can help protect data on devices running Windows.
BitLocker frequently asked questions (FAQ)	This topic for the IT professional answers frequently asked questions concerning the requirements to use, upgrade, deploy and administer, and key management policies for BitLocker.
Prepare your organization for BitLocker: Planning and policies	This topic for the IT professional explains how can you plan your BitLocker deployment.
BitLocker basic deployment	This topic for the IT professional explains how BitLocker features can be used to protect your data through drive encryption.
BitLocker: How to deploy on Windows Server	This topic for the IT professional explains how to deploy BitLocker on Windows Server.
BitLocker: How to enable Network Unlock	This topic for the IT professional describes how BitLocker Network Unlock works and how to configure it.
BitLocker: Use BitLocker Drive Encryption Tools to manage BitLocker	This topic for the IT professional describes how to use tools to manage BitLocker.
BitLocker: Use BitLocker Recovery Password Viewer	This topic for the IT professional describes how to use the BitLocker Recovery Password Viewer.
BitLocker Group Policy settings	This topic for IT professionals describes the function, location, and effect of each Group Policy setting that is used to manage BitLocker.
BCD settings and BitLocker	This topic for IT professionals describes the BCD settings that are used by BitLocker.
BitLocker Recovery Guide	This topic for IT professionals describes how to recover BitLocker keys from AD DS.
Protect BitLocker from pre-boot attacks	This detailed guide will help you understand the circumstances under which the use of pre-boot authentication is recommended for devices running Windows 11, Windows 10, Windows 8.1, Windows 8, or Windows 7; and when it can be safely omitted from a device's configuration.

TOPIC	DESCRIPTION
Troubleshoot BitLocker	This guide describes the resources that can help you troubleshoot BitLocker issues, and provides solutions for several common BitLocker issues.
Protecting cluster shared volumes and storage area networks with BitLocker	This topic for IT pros describes how to protect CSVs and SANs with BitLocker.
Enabling Secure Boot and BitLocker Device Encryption on Windows IoT Core	This topic covers how to use BitLocker with Windows IoT Core

Overview of BitLocker Device Encryption in Windows

7/1/2022 • 13 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016 and above

This article explains how BitLocker Device Encryption can help protect data on devices running Windows. For a general overview and list of articles about BitLocker, see [BitLocker](#).

When users travel, their organization's confidential data goes with them. Wherever confidential data is stored, it must be protected against unauthorized access. Windows has a long history of providing at-rest data-protection solutions that guard against nefarious attackers, beginning with the Encrypting File System in the Windows 2000 operating system. More recently, BitLocker has provided encryption for full drives and portable drives. Windows consistently improves data protection by improving existing options and providing new strategies.

Table 2 lists specific data-protection concerns and how they're addressed in Windows 11, Windows 10, and Windows 7.

Table 2. Data Protection in Windows 11, Windows 10, and Windows 7

WINDOWS 7	WINDOWS 11 AND WINDOWS 10
When BitLocker is used with a PIN to protect startup, PCs such as kiosks can't be restarted remotely.	Modern Windows devices are increasingly protected with BitLocker Device Encryption out of the box and support SSO to seamlessly protect the BitLocker encryption keys from cold boot attacks. Network Unlock allows PCs to start automatically when connected to the internal network.
When BitLocker is enabled, the provisioning process can take several hours.	BitLocker pre-provisioning, encrypting hard drives, and Used Space Only encryption allow administrators to enable BitLocker quickly on new computers.
There's no support for using BitLocker with self-encrypting drives (SEDs).	BitLocker supports offloading encryption to encrypted hard drives.
Administrators have to use separate tools to manage encrypted hard drives.	BitLocker supports encrypted hard drives with onboard encryption hardware built in, which allows administrators to use the familiar BitLocker administrative tools to manage them.
Encrypting a new flash drive can take more than 20 minutes.	Used Space Only encryption in BitLocker To Go allows users to encrypt removable data drives in seconds.
BitLocker could require users to enter a recovery key when system configuration changes occur.	BitLocker requires the user to enter a recovery key only when disk corruption occurs or when you lose the PIN or password.

WINDOWS 7	WINDOWS 11 AND WINDOWS 10
Users need to enter a PIN to start the PC, and then their password to sign in to Windows.	Modern Windows devices are increasingly protected with BitLocker Device Encryption out of the box and support SSO to help protect the BitLocker encryption keys from cold boot attacks.

Prepare for drive and file encryption

The best type of security measures is transparent to the user during implementation and use. Every time there's a possible delay or difficulty because of a security feature, there's strong likelihood that users will try to bypass security. This situation is especially true for data protection, and that's a scenario that organizations need to avoid. Whether you're planning to encrypt entire volumes, removable devices, or individual files, Windows 11 and Windows 10 meet your needs by providing streamlined, usable solutions. In fact, you can take several steps in advance to prepare for data encryption and make the deployment quick and smooth.

TPM pre-provisioning

In Windows 7, preparing the TPM for use offered a couple of challenges:

- You can turn on the TPM in the BIOS, which requires someone to either go into the BIOS settings to turn it on or to install a driver to turn it on from within Windows.
- When you enable the TPM, it may require one or more restarts.

Basically, it was a big hassle. If IT staff were provisioning new PCs, they could handle all of this, but if you wanted to add BitLocker to devices that were already in users' hands, those users would have struggled with the technical challenges and would either call IT for support or simply leave BitLocker disabled.

Microsoft includes instrumentation in Windows 11 and Windows 10 that enable the operating system to fully manage the TPM. There's no need to go into the BIOS, and all scenarios that required a restart have been eliminated.

Deploy hard drive encryption

BitLocker is capable of encrypting entire hard drives, including both system and data drives. BitLocker pre-provisioning can drastically reduce the time required to provision new PCs with BitLocker enabled. With Windows 11 and Windows 10, administrators can turn on BitLocker and the TPM from within the Windows Pre-installation Environment before they install Windows or as part of an automated deployment task sequence without any user interaction. Combined with Used Disk Space Only encryption and a mostly empty drive (because Windows isn't yet installed), it takes only a few seconds to enable BitLocker.

With earlier versions of Windows, administrators had to enable BitLocker after Windows had been installed. Although this process could be automated, BitLocker would need to encrypt the entire drive, a process that could take anywhere from several hours to more than a day depending on drive size and performance, which delayed deployment. Microsoft has improved this process through multiple features in Windows 11 and Windows 10.

BitLocker device encryption

Beginning in Windows 8.1, Windows automatically enables BitLocker Device Encryption on devices that support Modern Standby. With Windows 11 and Windows 10, Microsoft offers BitLocker Device Encryption support on a much broader range of devices, including those that are Modern Standby, and devices that run Windows 10 Home edition or Windows 11.

Microsoft expects that most devices in the future will pass the testing requirements, which makes BitLocker device encryption pervasive across modern Windows devices. BitLocker device encryption further protects the

system by transparently implementing device-wide data encryption.

Unlike a standard BitLocker implementation, BitLocker device encryption is enabled automatically so that the device is always protected. The following list outlines how this happens:

- When a clean installation of Windows 11 or Windows 10 is completed and the out-of-box experience is finished, the computer is prepared for first use. As part of this preparation, BitLocker Device Encryption is initialized on the operating system drive and fixed data drives on the computer with a clear key (this is the equivalent of standard BitLocker suspended state). In this state, the drive is shown with a warning icon in Windows Explorer. The yellow warning icon is removed after the TPM protector is created and the recovery key is backed up, as explained in the following bullet points.
- If the device isn't domain joined, a Microsoft account that has been granted administrative privileges on the device is required. When the administrator uses a Microsoft account to sign in, the clear key is removed, a recovery key is uploaded to the online Microsoft account, and a TPM protector is created. Should a device require the recovery key, the user will be guided to use an alternate device and navigate to a recovery key access URL to retrieve the recovery key by using his or her Microsoft account credentials.
- If the user uses a domain account to sign in, the clear key isn't removed until the user joins the device to a domain and the recovery key is successfully backed up to Active Directory Domain Services (AD DS). You must enable the **Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Operating System Drives** Group Policy setting, and select the **Do not enable BitLocker until recovery information is stored in AD DS for operating system drives** option. With this configuration, the recovery password is created automatically when the computer joins the domain, and then the recovery key is backed up to AD DS, the TPM protector is created, and the clear key is removed.
- Similar to signing in with a domain account, the clear key is removed when the user logs on to an Azure AD account on the device. As described in the bullet point above, the recovery password is created automatically when the user authenticates to Azure AD. Then, the recovery key is backed up to Azure AD, the TPM protector is created, and the clear key is removed.

Microsoft recommends that BitLocker Device Encryption be enabled on any systems that support it, but the automatic BitLocker Device Encryption process can be prevented by changing the following registry setting:

- **Subkey:** HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\BitLocker
- **Value:** PreventDeviceEncryption equal to True (1)
- **Type:** REG_DWORD

Administrators can manage domain-joined devices that have BitLocker device encryption enabled through Microsoft BitLocker Administration and Monitoring (MBAM). In this case, BitLocker device encryption automatically makes additional BitLocker options available. No conversion or encryption is required, and MBAM can manage the full BitLocker policy set if any configuration changes are required.

NOTE

BitLocker Device Encryption uses the XTS-AES 128-bit encryption method. In case you need to use a different encryption method and/or cipher strength, the device must be configured and decrypted (if already encrypted) first. After that, different BitLocker settings can be applied.

Used Disk Space Only encryption

BitLocker in earlier Windows versions could take a long time to encrypt a drive, because it encrypted every byte on the volume (including parts that didn't have data). That is still the most secure way to encrypt a drive, especially if a drive has previously contained confidential data that has since been moved or deleted. In that case, traces of the confidential data could remain on portions of the drive marked as unused. But why encrypt a

new drive when you can simply encrypt the data as it is being written? To reduce encryption time, BitLocker in Windows 11 and Windows 10 let users choose to encrypt just their data. Depending on the amount of data on the drive, this option can reduce encryption time by more than 99 percent. Exercise caution when encrypting only used space on an existing volume on which confidential data may have already been stored in an unencrypted state, however, because those sectors can be recovered through disk-recovery tools until they're overwritten by new encrypted data. In contrast, encrypting only used space on a brand-new volume can significantly decrease deployment time without the security risk because all new data will be encrypted as it's written to the disk.

Encrypted hard drive support

SEDs have been available for years, but Microsoft couldn't support their use with some earlier versions of Windows because the drives lacked important key management features. Microsoft worked with storage vendors to improve the hardware capabilities, and now BitLocker supports the next generation of SEDs, which are called encrypted hard drives. Encrypted hard drives provide onboard cryptographic capabilities to encrypt data on drives, which improves both drive and system performance by offloading cryptographic calculations from the PC's processor to the drive itself and rapidly encrypting the drive by using dedicated, purpose-built hardware. If you plan to use, whole-drive encryption with Windows 11 or Windows 10, Microsoft recommends that you investigate hard drive manufacturers and models to determine whether any of their encrypted hard drives meet your security and budget requirements. For more information about encrypted hard drives, see [Encrypted Hard Drive](#).

Preboot information protection

An effective implementation of information protection, like most security controls, considers usability and security. Users typically prefer a simple security experience. In fact, the more transparent a security solution becomes, the more likely users are to conform to it. It's crucial that organizations protect information on their PCs regardless of the state of the computer or the intent of users. This protection shouldn't be cumbersome to users. One undesirable and previously commonplace situation is when the user is prompted for input during preboot, and then again during Windows sign-in. Challenging users for input more than once should be avoided. Windows 11 and Windows 10 can enable a true SSO experience from the preboot environment on modern devices and in some cases even on older devices when robust information protection configurations are in place. The TPM in isolation is able to securely protect the BitLocker encryption key while it is at rest, and it can securely unlock the operating system drive. When the key is in use and thus in memory, a combination of hardware and Windows capabilities can secure the key and prevent unauthorized access through cold-boot attacks. Although other countermeasures like PIN-based unlock are available, they aren't as user-friendly; depending on the devices' configuration they may not offer additional security when it comes to key protection. For more information, see [BitLocker Countermeasures](#).

Manage passwords and PINs

When BitLocker is enabled on a system drive and the PC has a TPM, you can choose to require that users type a PIN before BitLocker will unlock the drive. Such a PIN requirement can prevent an attacker who has physical access to a PC from even getting to the Windows sign-in, which makes it virtually impossible for the attacker to access or modify user data and system files.

Requiring a PIN at startup is a useful security feature because it acts as a second authentication factor (a second "something you know"). This configuration comes with some costs, however. One of the most significant is the need to change the PIN regularly. In enterprises that used BitLocker with Windows 7 and the Windows Vista operating system, users had to contact systems administrators to update their BitLocker PIN or password. This requirement not only increased management costs but made users less willing to change their BitLocker PIN or password regularly. Windows 11 and Windows 10 users can update their BitLocker PINs and passwords themselves, without administrator credentials. Not only will this feature reduce support costs, but it could

improve security, too, because it encourages users to change their PINs and passwords more often. In addition, Modern Standby devices don't require a PIN for startup: They're designed to start infrequently and have other mitigations in place that further reduce the attack surface of the system. For more information about how startup security works and the countermeasures that Windows 11 and Windows 10 provide, see [Protect BitLocker from pre-boot attacks](#).

Configure Network Unlock

Some organizations have location-specific data security requirements. This is most common in environments where high-value data is stored on PCs. The network environment may provide crucial data protection and enforce mandatory authentication; therefore, policy states that those PCs shouldn't leave the building or be disconnected from the corporate network. Safeguards like physical security locks and geofencing may help enforce this policy as reactive controls. Beyond these, a proactive security control that grants data access only when the PC is connected to the corporate network is necessary.

Network Unlock enables BitLocker-protected PCs to start automatically when connected to a wired corporate network on which Windows Deployment Services runs. Anytime the PC isn't connected to the corporate network, a user must type a PIN to unlock the drive (if PIN-based unlock is enabled). Network Unlock requires the following infrastructure:

- Client PCs that have Unified Extensible Firmware Interface (UEFI) firmware version 2.3.1 or later, which supports Dynamic Host Configuration Protocol (DHCP)
- A server running at least Windows Server 2012 with the Windows deployment services role
- A server with the DHCP server role installed

For more information about how to configure Network unlock feature, see [BitLocker: How to enable Network Unlock](#).

Microsoft BitLocker administration and monitoring

Part of the Microsoft Desktop Optimization Pack, Microsoft BitLocker Administration and Monitoring (MBAM) makes it easier to manage and support BitLocker and BitLocker To Go. MBAM 2.5 with Service Pack 1, the latest version, has the following key features:

- Enables administrators to automate the process of encrypting volumes on client computers across the enterprise.
- Enables security officers to quickly determine the compliance state of individual computers or even of the enterprise itself.
- Provides centralized reporting and hardware management with Microsoft Endpoint Configuration Manager.
- Reduces the workload on the help desk to assist end users with BitLocker recovery requests.
- Enables end users to recover encrypted devices independently by using the Self-Service Portal.
- Enables security officers to easily audit access to recovery key information.
- Empowers Windows Enterprise users to continue working anywhere with the assurance that their corporate data is protected.
- Enforces the BitLocker encryption policy options that you set for your enterprise.
- Integrates with existing management tools, such as Microsoft Endpoint Configuration Manager.
- Offers an IT-customizable recovery user experience.
- Supports Windows 11 and Windows 10.

IMPORTANT

Enterprises could use MBAM to manage client computers with BitLocker that are domain-joined on-premises until mainstream support ended in July 2019, or they could receive extended support until April 2026.

Going forward, the functionality of MBAM will be incorporated into Configuration Manager. For more information, see [Features in Configuration Manager technical preview version 1909](#).

Enterprises not using Configuration Manager can use the built-in features of Azure AD and Microsoft Intune in Microsoft Endpoint Manager for administration and monitoring. For more information, see [Monitor device encryption with Intune](#).

Prepare your organization for BitLocker: Planning and policies

7/1/2022 • 13 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016 and above

This topic for the IT professional explains how can you plan your BitLocker deployment.

When you design your BitLocker deployment strategy, define the appropriate policies and configuration requirements based on the business requirements of your organization. The following sections will help you collect information. Use this information to help with your decision-making process about deploying and managing BitLocker systems.

Audit your environment

To plan your BitLocker deployment, understand your current environment. Do an informal audit to define your current policies, procedures, and hardware environment. Review your existing disk encryption software corporate security policies. If your organization isn't using disk encryption software, then none of these policies will exist. If you use disk encryption software, then you might need to change your organization's policies to use the BitLocker features.

To help you document your organization's current disk encryption security policies, answer the following questions:

1. Are there policies to determine which computers will use BitLocker and which computers won't use BitLocker?
2. What policies exist to control recovery password and recovery key storage?
3. What are the policies for validating the user identities that need to run BitLocker recovery?
4. What policies exist to control who in the organization has access to recovery data?
5. What policies exist to control computer decommissioning or retirement?

Encryption keys and authentication

BitLocker helps prevent unauthorized access to data on lost or stolen computers by:

- Encrypting the entire Windows operating system volume on the hard disk.
- Verifying the boot process integrity.

The trusted platform module (TPM) is a hardware component installed in many newer computers by the computer manufacturers. It works with BitLocker to help protect user data. And, help make sure a computer hasn't been tampered with while the system was offline.

Also, BitLocker can lock the normal startup process until the user supplies a personal identification number (PIN) or inserts a removable USB device, such as a flash drive, that contains a startup key. These extra security measures provide multifactor authentication. They also make sure that the computer won't start or resume from hibernation until the correct PIN or startup key is presented.

On computers that don't have a TPM version 1.2 or higher, you can still use BitLocker to encrypt the Windows operating system volume. However, this implementation requires the user to insert a USB startup key to start the computer or resume from hibernation. It doesn't provide the pre-startup system integrity verification offered by BitLocker working with a TPM.

BitLocker key protectors

KEY PROTECTOR	DESCRIPTION
TPM	A hardware device used to help establish a secure root-of-trust. BitLocker only supports TPM version 1.2 or higher.
PIN	A user-entered numeric key protector that can only be used in addition to the TPM.
Enhanced PIN	A user-entered alphanumeric key protector that can only be used in addition to the TPM.
Startup key	An encryption key that can be stored on most removable media. This key protector can be used alone on non-TPM computers, or with a TPM for added security.
Recovery password	A 48-digit number used to unlock a volume when it is in recovery mode. Numbers can often be typed on a regular keyboard, if the numbers on the normal keyboard are not responding you can always use the function keys (F1-F10) to input the numbers.
Recovery key	An encryption key stored on removable media that can be used for recovering data encrypted on a BitLocker volume.

BitLocker authentication methods

AUTHENTICATION METHOD	REQUIRES USER INTERACTION	DESCRIPTION
TPM only	No	TPM validates early boot components.
TPM + PIN	Yes	TPM validates early boot components. The user must enter the correct PIN before the start-up process can continue, and before the drive can be unlocked. The TPM will enter lockout if the incorrect PIN is entered repeatedly to protect the PIN from brute force attacks. The number of repeated attempts that will trigger a lockout is variable.
TPM + Network key	No	The TPM successfully validates early boot components, and a valid encrypted network key has been provided from the WDS server. This authentication method provides automatic unlock of operating system volumes at system reboot while still maintaining multifactor authentication.

AUTHENTICATION METHOD	REQUIRES USER INTERACTION	DESCRIPTION
TPM + startup key	Yes	The TPM successfully validates early boot components, and a USB flash drive containing the startup key has been inserted.
Startup key only	Yes	The user is prompted for the USB flash drive that has the recovery key and/or startup key, and then reboot the computer.

Will you support computers without TPM version 1.2 or higher?

Determine if you're support computers that don't have a TPM version 1.2 or higher. If you support BitLocker on this type of computer, a user must use a USB startup key to boot the system. This startup key requires extra support processes similar to multifactor authentication.

What areas of your organization need a baseline level of data protection?

The TPM-only authentication method will provide the most transparent user experience for organizations that need a baseline level of data protection to meet security policies. It has the lowest total cost of ownership. TPM-only might also be more appropriate for computers that are unattended or that must reboot unattended.

However, TPM-only authentication method offers the lowest level of data protection. This authentication method protects against attacks that modify early boot components. But, the level of protection can be affected by potential weaknesses in hardware or in the early boot components. BitLocker's multifactor authentication methods significantly increase the overall level of data protection.

What areas of your organization need a more secure level of data protection?

If there are user computers with highly sensitive data, then deploy BitLocker with multifactor authentication on those systems. Requiring the user to input a PIN significantly increases the level of protection for the system. You can also use BitLocker Network Unlock to allow these computers to automatically unlock when connected to a trusted wired network that can provide the Network Unlock key.

What multifactor authentication method does your organization prefer?

The protection differences provided by multifactor authentication methods can't be easily quantified. Consider each authentication method's impact on Helpdesk support, user education, user productivity, and any automated systems management processes.

TPM hardware configurations

In your deployment plan, identify what TPM-based hardware platforms will be supported. Document the hardware models from an OEM of your choice, so that their configurations can be tested and supported. TPM hardware requires special consideration during all aspects of planning and deployment.

TPM 1.2 states and initialization

For TPM 1.2, there are multiple possible states. Windows automatically initializes the TPM, which brings it to an enabled, activated, and owned state. This state is the state that BitLocker requires before it can use the TPM.

Endorsement keys

For a TPM to be usable by BitLocker, it must contain an endorsement key, which is an RSA key pair. The private half of the key pair is held inside the TPM and is never revealed or accessible outside the TPM. If the TPM doesn't have an endorsement key, BitLocker will force the TPM to generate one automatically as part of BitLocker setup.

An endorsement key can be created at various points in the TPM's lifecycle, but needs to be created only once

for the lifetime of the TPM. If an endorsement key doesn't exist for the TPM, it must be created before TPM ownership can be taken.

For more information about the TPM and the TCG, see the Trusted Computing Group: Trusted Platform Module (TPM) Specifications (<https://go.microsoft.com/fwlink/p/?linkid=69584>).

Non-TPM hardware configurations

Devices that don't include a TPM can still be protected by drive encryption. Windows To Go workspaces can be BitLocker protected using a startup password and PCs without a TPM can use a startup key.

Use the following questions to identify issues that might affect your deployment in a non-TPM configuration:

- Are password complexity rules in place?
- Do you have budget for USB flash drives for each of these computers?
- Do your existing non-TPM devices support USB devices at boot time?

Test your individual hardware platforms with the BitLocker system check option while you're enabling BitLocker. The system check makes sure that BitLocker can read the recovery information from a USB device and encryption keys correctly before it encrypts the volume. CD and DVD drives can't act as a block storage device and can't be used to store the BitLocker recovery material.

Disk configuration considerations

To function correctly, BitLocker requires a specific disk configuration. BitLocker requires two partitions that meet the following requirements:

- The operating system partition contains the operating system and its support files; it must be formatted with the NTFS file system
- The system partition (or boot partition) includes the files needed to load Windows after the BIOS or UEFI firmware has prepared the system hardware. BitLocker isn't enabled on this partition. For BitLocker to work, the system partition must not be encrypted, and must be on a different partition than the operating system. On UEFI platforms, the system partition must be formatted with the FAT 32-file system. On BIOS platforms, the system partition must be formatted with the NTFS file system. It should be at least 350 MB in size.

Windows setup will automatically configure the disk drives of your computer to support BitLocker encryption.

Windows Recovery Environment (Windows RE) is an extensible recovery platform that is based on Windows Pre-installation Environment (Windows PE). When the computer fails to start, Windows automatically transitions into this environment, and the Startup Repair tool in Windows RE automates the diagnosis and repair of an unbootable Windows installation. Windows RE also contains the drivers and tools that are needed to unlock a volume protected by BitLocker by providing a recovery key or recovery password. To use Windows RE with BitLocker, the Windows RE boot image must be on a volume that isn't protected by BitLocker.

Windows RE can also be used from boot media other than the local hard disk. If you don't install Windows RE on the local hard disk of BitLocker-enabled computers, then you can use different boot methods. For example, you can use Windows Deployment Services, CD-ROM, or USB flash drive for recovery.

BitLocker provisioning

In Windows Vista and Windows 7, BitLocker was provisioned after the installation for system and data volumes. It used the `manage-bde` command line interface or the Control Panel user interface. With newer operating systems, BitLocker can be provisioned before the operating system is installed. Preprovisioning requires the computer have a TPM.

To check the BitLocker status of a particular volume, administrators can look at the drive status in the BitLocker

control panel applet or Windows Explorer. The "Waiting For Activation" status with a yellow exclamation icon means that the drive was preprovisioned for BitLocker. This status means that there was only a clear protector used when encrypting the volume. In this case, the volume isn't protected, and needs to have a secure key added to the volume before the drive is considered fully protected. Administrators can use the control panel options, `manage-bde` tool, or WMI APIs to add an appropriate key protector. The volume status will be updated.

When using the control panel options, administrators can choose to **Turn on BitLocker** and follow the steps in the wizard to add a protector, such as a PIN for an operating system volume (or a password if no TPM exists), or a password or smart card protector to a data volume. Then the drive security window is presented before changing the volume status.

Administrators can enable BitLocker before to operating system deployment from the Windows Pre-installation Environment (WinPE). This step is done with a randomly generated clear key protector applied to the formatted volume. It encrypts the volume before running the Windows setup process. If the encryption uses the Used Disk Space Only option, then this step takes only a few seconds. And, it incorporates into the regular deployment processes.

Used Disk Space Only encryption

The BitLocker Setup wizard provides administrators the ability to choose the Used Disk Space Only or Full encryption method when enabling BitLocker for a volume. Administrators can use the new BitLocker Group Policy setting to enforce either Used Disk Space Only or Full disk encryption.

Launching the BitLocker Setup wizard prompts for the authentication method to be used (password and smart card are available for data volumes). Once the method is chosen and the recovery key is saved, you're asked to choose the drive encryption type. Select Used Disk Space Only or Full drive encryption.

With Used Disk Space Only, only the portion of the drive that contains data will be encrypted. Unused space will remain unencrypted. This behavior causes the encryption process to be much faster, especially for new PCs and data drives. When BitLocker is enabled with this method, as data is added to the drive, the portion of the drive used is encrypted. So, there's never unencrypted data stored on the drive.

With Full drive encryption, the entire drive is encrypted, whether data is stored on it or not. This option is useful for drives that have been repurposed, and may contain data remnants from their previous use.

Active Directory Domain Services considerations

BitLocker integrates with Active Directory Domain Services (AD DS) to provide centralized key management. By default, no recovery information is backed up to Active Directory. Administrators can configure the following Group Policy setting for each drive type to enable backup of BitLocker recovery information:

Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption*drive type*\Choose how BitLocker protected drives can be recovered.

By default, only Domain Admins have access to BitLocker recovery information, but [access can be delegated to others](#).

The following recovery data is saved for each computer object:

- **Recovery password**

A 48-digit recovery password used to recover a BitLocker-protected volume. Users enter this password to unlock a volume when BitLocker enters recovery mode.

- **Key package data**

With this key package and the recovery password, you will be able decrypt portions of a BitLocker-protected volume if the disk is severely damaged. Each key package will only work with the volume it was

created on, which can be identified by the corresponding volume ID.

FIPS support for recovery password protector

Functionality introduced in Windows Server 2012 R2 and Windows 8.1, allows BitLocker to be fully functional in FIPS mode.

NOTE

The United States Federal Information Processing Standard (FIPS) defines security and interoperability requirements for computer systems that are used by the U.S. federal government. The FIPS 140 standard defines approved cryptographic algorithms. The FIPS 140 standard also sets forth requirements for key generation and for key management. The National Institute of Standards and Technology (NIST) uses the Cryptographic Module Validation Program (CMVP) to determine whether a particular implementation of a cryptographic algorithm is compliant with the FIPS 140 standard. An implementation of a cryptographic algorithm is considered FIPS 140-compliant only if it has been submitted for and has passed NIST validation. An algorithm that hasn't been submitted can't be considered FIPS-compliant, even if the implementation produces identical data as a validated implementation of the same algorithm.

Before these supported versions of Windows, when Windows was in FIPS mode, BitLocker prevented the creation or use of recovery passwords and instead forced the user to use recovery keys. For more information about these issues, see the support article [kb947249](#).

But on computers running these supported systems with BitLocker enabled:

- FIPS-compliant recovery password protectors can be created when Windows is in FIPS mode. These protectors use the FIPS 140 NIST SP800-132 algorithm.
- Recovery passwords created in FIPS mode on Windows 8.1 can be distinguished from recovery passwords created on other systems.
- Recovery unlock using the FIPS-compliant algorithm based recovery password protector work in all cases that currently work for recovery passwords.
- When FIPS-compliant recovery passwords unlock volumes, the volume is unlocked to allow read/write access even while in FIPS mode.
- FIPS-compliant recovery password protectors can be exported and stored in AD a while in FIPS mode.

The BitLocker Group Policy settings for recovery passwords work the same for all Windows versions that support BitLocker, whether in FIPS mode or not.

On Windows Server 2012 R2 and Windows 8.1 and older, you can't use recovery passwords generated on a system in FIPS mode. Recovery passwords created on Windows Server 2012 R2 and Windows 8.1 are incompatible with BitLocker on operating systems older than Windows Server 2012 R2 and Windows 8.1. So, recovery keys should be used instead.

More information

- [Trusted Platform Module](#)
- [TPM Group Policy settings](#)
- [BitLocker frequently asked questions \(FAQ\)](#)
- [BitLocker](#)
- [BitLocker Group Policy settings](#)
- [BitLocker basic deployment](#)

BitLocker deployment comparison

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016 and above

This article depicts the BitLocker deployment comparison chart.

BitLocker deployment comparison chart

REQUIREMENTS	MICROSOFT INTUNE	MICROSOFT ENDPOINT CONFIGURATION MANAGER	MICROSOFT BITLOCKER ADMINISTRATION AND MONITORING (MBAM)
Minimum client operating system version	Windows 11 and Windows 10	Windows 11, Windows 10, and Windows 8.1	Windows 7, Windows 8, Windows 8.1, Windows 10, Windows 10 IoT, and Windows 11
Supported Windows SKUs	Enterprise, Pro, Education	Enterprise, Pro, Education	Enterprise
Minimum Windows version	1909	None	None
Supported domain-joined status	Microsoft Azure Active Directory (Azure AD) joined, hybrid Azure AD joined	Active Directory-joined, hybrid Azure AD joined	Active Directory-joined
Permissions required to manage policies	Endpoint security manager or custom	Full administrator or custom	Domain Admin or Delegated GPO access
Cloud or on premises	Cloud	On premises	On premises
Server components required?		✓	✓
Additional agent required?	No (device enrollment only)	Configuration Manager client	MBAM client
Administrative plane	Microsoft Endpoint Manager admin center	Configuration Manager console	Group Policy Management Console and MBAM sites
Administrative portal installation required		✓	✓
Compliance reporting capabilities	✓	✓	✓
Force encryption	✓	✓	✓

REQUIREMENTS	MICROSOFT INTUNE	MICROSOFT ENDPOINT CONFIGURATION MANAGER	MICROSOFT BITLOCKER ADMINISTRATION AND MONITORING (MBAM)
Encryption for storage cards (mobile)	✓	✓	
Allow recovery password	✓	✓	✓
Manage startup authentication	✓	✓	✓
Select cipher strength and algorithms for fixed drives	✓	✓	✓
Select cipher strength and algorithms for removable drives	✓	✓	✓
Select cipher strength and algorithms for operating environment drives	✓	✓	✓
Standard recovery password storage location	Azure AD or Active Directory	Configuration Manager site database	MBAM database
Store recovery password for operating system and fixed drives to Azure AD or Active Directory	Yes (Active Directory and Azure AD)	Yes (Active Directory only)	Yes (Active Directory only)
Customize preboot message and recovery link	✓	✓	✓
Allow/deny key file creation	✓	✓	✓
Deny Write permission to unprotected drives	✓	✓	✓
Can be administered outside company network	✓	✓	
Support for organization unique IDs		✓	✓
Self-service recovery	Yes (through Azure AD or Company Portal app)	✓	✓
Recovery password rotation for fixed and operating environment drives	Yes (Windows 10, version 1909 and later or Windows 11)	✓	✓
Wait to complete encryption until recovery information is backed up to Azure AD	✓		

REQUIREMENTS	MICROSOFT INTUNE	MICROSOFT ENDPOINT CONFIGURATION MANAGER	MICROSOFT BITLOCKER ADMINISTRATION AND MONITORING (MBAM)
Wait to complete encryption until recovery information is backed up to Active Directory		✓	✓
Allow or deny Data Recovery Agent	✓	✓	✓
Unlock a volume using certificate with custom object identifier		✓	✓
Prevent memory overwrite on restart		✓	✓
Configure custom Trusted Platform Module Platform Configuration Register profiles			✓
Manage auto-unlock functionality		✓	✓

BitLocker basic deployment

7/1/2022 • 21 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016 and above

This article for the IT professional explains how BitLocker features can be used to protect your data through drive encryption.

Using BitLocker to encrypt volumes

BitLocker provides full volume encryption (FVE) for operating system volumes, and fixed and removable data drives. To support fully encrypted operating system drives, BitLocker uses an unencrypted system partition for the files required to boot, decrypt, and load the operating system. This volume is automatically created during a new installation of both client and server operating systems.

If the drive was prepared as a single contiguous space, BitLocker requires a new volume to hold the boot files. BdeHdCfg.exe can create these volumes.

NOTE

For more info about using this tool, see [Bdehdcfg](#) in the Command-Line Reference.

BitLocker encryption can be done using the following methods:

- BitLocker control panel
- Windows Explorer
- `manage-bde` command-line interface
- BitLocker Windows PowerShell cmdlets

Encrypting volumes using the BitLocker control panel

Encrypting volumes with the BitLocker control panel (select **Start**, type *BitLocker*, select **Manage BitLocker**) is how many users will use BitLocker. The name of the BitLocker control panel is BitLocker Drive Encryption. The BitLocker control panel supports encrypting operating system, fixed data, and removable data volumes. The BitLocker control panel will organize available drives in the appropriate category based on how the device reports itself to Windows. Only formatted volumes with assigned drive letters will appear properly in the BitLocker control panel applet.

To start encryption for a volume, select **Turn on BitLocker** for the appropriate drive to initialize the BitLocker Drive Encryption Wizard. BitLocker Drive Encryption Wizard options vary based on volume type (operating system volume or data volume).

Operating system volume

When the BitLocker Drive Encryption Wizard launches, it verifies the computer meets the BitLocker system requirements for encrypting an operating system volume. By default, the system requirements are:

REQUIREMENT	DESCRIPTION
Hardware configuration	The computer must meet the minimum requirements for the supported Windows versions.
Operating system	BitLocker is an optional feature that can be installed by Server Manager on Windows Server 2012 and later.
Hardware TPM	TPM version 1.2 or 2.0. A TPM isn't required for BitLocker; however, only a computer with a TPM can provide the additional security of pre-startup system integrity verification and multifactor authentication.
BIOS configuration	<ul style="list-style-type: none"> • A Trusted Computing Group (TCG)-compliant BIOS or UEFI firmware. • The boot order must be set to start first from the hard disk, and not the USB or CD drives. • The firmware must be able to read from a USB flash drive during startup.
File system	<p>For computers that boot natively with UEFI firmware, at least one FAT32 partition for the system drive and one NTFS partition for the operating system drive.</p> <p>For computers with legacy BIOS firmware, at least two NTFS disk partitions, one for the system drive and one for the operating system drive.</p> <p>For either firmware, the system drive partition must be at least 350 megabytes (MB) and set as the active partition.</p>
Hardware encrypted drive prerequisites (optional)	To use a hardware encrypted drive as the boot drive, the drive must be in the uninitialized state and in the security inactive state. In addition, the system must always boot with native UEFI version 2.3.1 or higher and the CSM (if any) disabled.

Upon passing the initial configuration, users are required to enter a password for the volume. If the volume doesn't pass the initial configuration for BitLocker, the user is presented with an error dialog describing the appropriate actions to be taken. Once a strong password has been created for the volume, a recovery key will be generated. The BitLocker Drive Encryption Wizard will prompt for a location to save this key. A BitLocker recovery key is a special key that you can create when you turn on BitLocker Drive Encryption for the first time on each drive that you encrypt. You can use the recovery key to gain access to your computer if the drive that Windows is installed on (the operating system drive) is encrypted using BitLocker Drive Encryption and BitLocker detects a condition that prevents it from unlocking the drive when the computer is starting up. A recovery key can also be used to gain access to your files and folders on a removable data drive (such as an external hard drive or USB flash drive) that is encrypted using BitLocker To Go, if for some reason you forget the password or your computer can't access the drive.

You should store the recovery key by printing it, saving it on removable media, or saving it as a file in a network folder or on your OneDrive, or on another drive of your computer that you aren't encrypting. You can't save the recovery key to the root directory of a non-removable drive and can't be stored on the encrypted volume. You can't save the recovery key for a removable data drive (such as a USB flash drive) on removable media. Ideally, you should store the recovery key separate from your computer. After you create a recovery key, you can use the BitLocker control panel to make additional copies.

- Encrypt used disk space only - Encrypts only disk space that contains data

- Encrypt entire drive - Encrypts the entire volume including free space

It's recommended that drives with little to no data use the **used disk space only** encryption option and that drives with data or an operating system use the **encrypt entire drive** option.

NOTE

Deleted files appear as free space to the file system, which isn't encrypted by **used disk space only**. Until they are wiped or overwritten, deleted files hold information that could be recovered with common data forensic tools.

Selecting an encryption type and choosing **Next** will give the user the option of running a BitLocker system check (selected by default) which will ensure that BitLocker can properly access the recovery and encryption keys before the volume encryption begins. We recommend running this system check before starting the encryption process. If the system check isn't run and a problem is encountered when the operating system attempts to start, the user will need to provide the recovery key to start Windows.

After completing the system check (if selected), the BitLocker Drive Encryption Wizard restarts the computer to begin encryption. Upon reboot, users are required to enter the password chosen to boot into the operating system volume. Users can check encryption status by checking the system notification area or the BitLocker control panel.

Until encryption is completed, the only available options for managing BitLocker involve manipulation of the password protecting the operating system volume, backing up the recovery key, and turning off BitLocker.

Data volume

Encrypting data volumes using the BitLocker control panel interface works in a similar fashion to encryption of the operating system volumes. Users select **Turn on BitLocker** within the control panel to begin the BitLocker Drive Encryption wizard. Unlike for operating system volumes, data volumes aren't required to pass any configuration tests for the wizard to proceed. Upon launching the wizard, a choice of authentication methods to unlock the drive appears. The available options are **password** and **smart card** and **automatically unlock this drive on this computer**. Disabled by default, the latter option will unlock the data volume without user input when the operating system volume is unlocked.

After selecting the desired authentication method and choosing **Next**, the wizard presents options for storage of the recovery key. These options are the same as for operating system volumes. With the recovery key saved, selecting **Next** in the wizard will show available options for encryption. These options are the same as for operating system volumes; **used disk space only** and **full drive encryption**. If the volume being encrypted is new or empty, it's recommended that used space only encryption is selected.

With an encryption method chosen, a final confirmation screen is displayed before the encryption process begins. Selecting **Start encrypting** begins encryption.

Encryption status displays in the notification area or within the BitLocker control panel.

OneDrive option

There's a new option for storing the BitLocker recovery key using the OneDrive. This option requires that computers aren't members of a domain and that the user is using a Microsoft Account. Local accounts don't give the option to use OneDrive. Using the OneDrive option is the default, recommended recovery key storage method for computers that aren't joined to a domain.

Users can verify whether the recovery key was saved properly by checking their OneDrive for the BitLocker folder which is created automatically during the save process. The folder will contain two files, a readme.txt and the recovery key. For users storing more than one recovery password on their OneDrive, they can identify the required recovery key by looking at the file name. The recovery key ID is appended to the end of the file name.

Using BitLocker within Windows Explorer

Windows Explorer allows users to launch the BitLocker Drive Encryption wizard by right-clicking a volume and selecting **Turn On BitLocker**. This option is available on client computers by default. On servers, you must first install the BitLocker and Desktop-Experience features for this option to be available. After selecting **Turn on BitLocker**, the wizard works exactly as it does when launched using the BitLocker control panel.

Down-level compatibility

The following table shows the compatibility matrix for systems that have been BitLocker-enabled and then presented to a different version of Windows.

Table 1: Cross compatibility for Windows 11, Windows 10, Windows 8.1, Windows 8, and Windows 7 encrypted volumes

ENCRYPTION TYPE	WINDOWS 11, WINDOWS 10, AND WINDOWS 8.1	WINDOWS 8	WINDOWS 7
Fully encrypted on Windows 8	Presents as fully encrypted	N/A	Presented as fully encrypted
Used Disk Space Only encrypted on Windows 8	Presents as encrypt on write	N/A	Presented as fully encrypted
Fully encrypted volume from Windows 7	Presents as fully encrypted	Presented as fully encrypted	N/A
Partially encrypted volume from Windows 7	Windows 11, Windows 10, and Windows 8.1 will complete encryption regardless of policy	Windows 8 will complete encryption regardless of policy	N/A

Encrypting volumes using the manage-bde command-line interface

Manage-bde is a command-line utility that can be used for scripting BitLocker operations. Manage-bde offers additional options not displayed in the BitLocker control panel. For a complete list of the options, see [Manage-bde](#).

Manage-bde offers a multitude of wider options for configuring BitLocker. So using the command syntax may require care and possibly later customization by the user. For example, using just the `manage-bde -on` command on a data volume will fully encrypt the volume without any authenticating protectors. A volume encrypted in this manner still requires user interaction to turn on BitLocker protection, even though the command successfully completed because an authentication method needs to be added to the volume for it to be fully protected.

Command-line users need to determine the appropriate syntax for a given situation. The following section covers general encryption for operating system volumes and data volumes.

Operating system volume

Listed below are examples of basic valid commands for operating system volumes. In general, using only the `manage-bde -on <drive letter>` command encrypts the operating system volume with a TPM-only protector and no recovery key. However, many environments require more secure protectors such as passwords or PIN and expect to be able to recover information with a recovery key.

Determining volume status

A good practice when using manage-bde is to determine the volume status on the target system. Use the following command to determine volume status:

```
manage-bde -status
```

This command returns the volumes on the target, current encryption status, and volume type (operating system or data) for each volume. Using this information, users can determine the best encryption method for their environment.

Enabling BitLocker without a TPM

For example, suppose that you want to enable BitLocker on a computer without a TPM chip. To properly enable BitLocker for the operating system volume, you'll need to use a USB flash drive as a startup key to boot (in this example, the drive letter E). You would first create the startup key needed for BitLocker using the `-protectors` option and save it to the USB drive on E: and then begin the encryption process. You'll need to reboot the computer when prompted to complete the encryption process.

```
manage-bde -protectors -add C: -startupkey E:  
manage-bde -on C:
```

Enabling BitLocker with a TPM only

It's possible to encrypt the operating system volume without any defined protectors by using `manage-bde`. Use this command:

```
manage-bde -on C:
```

This will encrypt the drive using the TPM as the protector. If users are unsure of the protector for a volume, they can use the `-protectors` option in `manage-bde` to list this information by executing the following command:

```
manage-bde -protectors -get <volume>
```

Provisioning BitLocker with two protectors

Another example is a user on a non-TPM hardware who wishes to add a password and SID-based protector to the operating system volume. In this instance, the user adds the protectors first. This is done with the command:

```
manage-bde -protectors -add C: -pw -sid <user or group>
```

This command requires the user to enter and then confirm the password protectors before adding them to the volume. With the protectors enabled on the volume, the user just needs to turn BitLocker on.

Data volume

Data volumes use the same syntax for encryption as operating system volumes but they don't require protectors for the operation to complete. Encrypting data volumes can be done using the base command:

```
manage-bde -on <drive letter>
```

or users can choose to add protectors to the volume. We recommend that you add at least one primary protector and a recovery protector to a data volume.

Enabling BitLocker with a password

A common protector for a data volume is the password protector. In the example below, we add a password protector to the volume and turn on BitLocker.

```
manage-bde -protectors -add -pw C:  
manage-bde -on C:
```

Encrypting volumes using the BitLocker Windows PowerShell cmdlets

Windows PowerShell cmdlets provide an alternative way to work with BitLocker. Using Windows PowerShell's scripting capabilities, administrators can integrate BitLocker options into existing scripts with ease. The list below

displays the available BitLocker cmdlets.

NAME	PARAMETERS
Add-BitLockerKeyProtector	<ul style="list-style-type: none"> • ADAccountOrGroup • ADAccountOrGroupProtector • Confirm • MountPoint • Password • PasswordProtector • Pin • RecoveryKeyPath • RecoveryKeyProtector • RecoveryPassword • RecoveryPasswordProtector • Service • StartupKeyPath • StartupKeyProtector • TpmAndPinAndStartupKeyProtector • TpmAndPinProtector • TpmAndStartupKeyProtector • TpmProtector • WhatIf
Backup-BitLockerKeyProtector	<ul style="list-style-type: none"> • Confirm • KeyProtectorId • MountPoint • WhatIf
Disable-BitLocker	<ul style="list-style-type: none"> • Confirm • MountPoint • WhatIf
Disable-BitLockerAutoUnlock	<ul style="list-style-type: none"> • Confirm • MountPoint • WhatIf
Enable-BitLocker	<ul style="list-style-type: none"> • AdAccountOrGroup • AdAccountOrGroupProtector • Confirm • EncryptionMethod • HardwareEncryption • Password • PasswordProtector • Pin • RecoveryKeyPath • RecoveryKeyProtector • RecoveryPassword • RecoveryPasswordProtector • Service • SkipHardwareTest • StartupKeyPath • StartupKeyProtector • TpmAndPinAndStartupKeyProtector • TpmAndPinProtector • TpmAndStartupKeyProtector • TpmProtector • UsedSpaceOnly • WhatIf

NAME	PARAMETERS
Enable-BitLockerAutoUnlock	<ul style="list-style-type: none"> • Confirm • MountPoint • WhatIf
Get-BitLockerVolume	<ul style="list-style-type: none"> • MountPoint
Lock-BitLocker	<ul style="list-style-type: none"> • Confirm • ForceDismount • MountPoint • WhatIf
Remove-BitLockerKeyProtector	<ul style="list-style-type: none"> • Confirm • KeyProtectorId • MountPoint • WhatIf
Resume-BitLocker	<ul style="list-style-type: none"> • Confirm • MountPoint • WhatIf
Suspend-BitLocker	<ul style="list-style-type: none"> • Confirm • MountPoint • RebootCount • WhatIf
Unlock-BitLocker	<ul style="list-style-type: none"> • AdAccountOrGroup • Confirm • MountPoint • Password • RecoveryKeyPath • RecoveryPassword • RecoveryPassword • WhatIf

Similar to manage-bde, the Windows PowerShell cmdlets allow configuration beyond the options offered in the control panel. As with manage-bde, users need to consider the specific needs of the volume they're encrypting prior to running Windows PowerShell cmdlets.

A good initial step is to determine the current state of the volume(s) on the computer. You can do this using the `Get-BitLocker` volume cmdlet. The output from this cmdlet displays information on the volume type, protectors, protection status, and other useful information.

Occasionally, all protectors may not be shown when using `Get-BitLockerVolume` due to lack of space in the output display. If you don't see all of the protectors for a volume, you can use the Windows PowerShell pipe command (`|`) to format a listing of the protectors.

NOTE

In the event that there are more than four protectors for a volume, the pipe command may run out of display space. For volumes with more than four protectors, use the method described in the section below to generate a listing of all protectors with protector ID.

```
Get-BitLockerVolume C: | fl
```


If you want to remove the existing protectors prior to provisioning BitLocker on the volume, you can utilize the `Remove-BitLockerKeyProtector` cmdlet. Accomplishing this requires the GUID associated with the protector to be removed. A simple script can pipe out the values of each `Get-BitLockerVolume` return to another variable as seen below:

```
$vol = Get-BitLockerVolume
$keyprotectors = $vol.KeyProtector
```

Using this script, we can display the information in the `$keyprotectors` variable to determine the GUID for each protector. Using this information, we can then remove the key protector for a specific volume using the command:

```
Remove-BitLockerKeyProtector <volume>: -KeyProtectorID "{GUID}"
```

NOTE

The BitLocker cmdlet requires the key protector GUID (enclosed in quotation marks) to execute. Ensure the entire GUID, with braces, is included in the command.

Operating system volume

Using the BitLocker Windows PowerShell cmdlets is similar to working with the `manage-bde` tool for encrypting operating system volumes. Windows PowerShell offers users a lot of flexibility. For example, users can add the desired protector as part command for encrypting the volume. Below are examples of common user scenarios and steps to accomplish them using the BitLocker cmdlets for Windows PowerShell.

To enable BitLocker with just the TPM protector, use this command:

```
Enable-BitLocker C:
```

The example below adds one additional protector, the StartupKey protectors, and chooses to skip the BitLocker hardware test. In this example, encryption starts immediately without the need for a reboot.

```
Enable-BitLocker C: -StartupKeyProtector -StartupKeyPath <path> -SkipHardwareTest
```

Data volume

Data volume encryption using Windows PowerShell is the same as for operating system volumes. You should add the desired protectors prior to encrypting the volume. The following example adds a password protector to the E: volume using the variable `$pw` as the password. The `$pw` variable is held as a `SecureString` value to store the user-defined password. Last, encryption begins.

```
$pw = Read-Host -AsSecureString
<user inputs password>
Enable-BitLockerKeyProtector E: -PasswordProtector -Password $pw
```

Using an SID-based protector in Windows PowerShell

The `ADAccountOrGroup` protector is an Active Directory SID-based protector. This protector can be added to both operating system and data volumes, although it doesn't unlock operating system volumes in the pre-boot environment. The protector requires the SID for the domain account or group to link with the protector. BitLocker can protect a cluster-aware disk by adding an SID-based protector for the Cluster Name Object (CNO) that lets the disk properly failover and be unlocked to any member computer of the cluster.

WARNING

The SID-based protector requires the use of an additional protector (such as TPM, PIN, recovery key, etc.) when used on operating system volumes.

To add an ADAccountOrGroup protector to a volume, you need either the actual domain SID or the group name preceded by the domain and a backslash. In the example below, the CONTOSO\Administrator account is added as a protector to the data volume G.

```
Enable-BitLocker G: -AdAccountOrGroupProtector -AdAccountOrGroup CONTOSO\Administrator
```

For users who wish to use the SID for the account or group, the first step is to determine the SID associated with the account. To get the specific SID for a user account in Windows PowerShell, use the following command:

```
Get-ADUser -filter {samaccountname -eq "administrator"}
```

NOTE

Use of this command requires the RSAT-AD-PowerShell feature.

TIP

In addition to the Windows PowerShell command above, information about the locally logged on user and group membership can be found using: WHOAMI /ALL. This doesn't require the use of additional features.

In the example below, the user wishes to add a domain SID-based protector to the previously encrypted operating system volume. The user knows the SID for the user account or group they wish to add and uses the following command:

```
Add-BitLockerKeyProtector C: -ADAccountOrGroupProtector -ADAccountOrGroup "<SID>"
```

NOTE

Active Directory-based protectors are normally used to unlock Failover Cluster-enabled volumes.

Checking BitLocker status

To check the BitLocker status of a particular volume, administrators can look at the status of the drive in the BitLocker control panel applet, Windows Explorer, manage-bde command-line tool, or Windows PowerShell cmdlets. Each option offers different levels of detail and ease of use. We'll look at each of the available methods in the following section.

Checking BitLocker status with the control panel

Checking BitLocker status with the control panel is the most common method used by most users. Once opened, the status for each volume is displayed next to the volume description and drive letter. Available status return values with the control panel include:

STATUS	DESCRIPTION
On	BitLocker is enabled for the volume
Off	BitLocker isn't enabled for the volume
Suspended	BitLocker is suspended and not actively protecting the volume
Waiting for Activation	BitLocker is enabled with a clear protector key and requires further action to be fully protected

If a drive is pre-provisioned with BitLocker, a status of "Waiting for Activation" displays with a yellow exclamation icon on the volume. This status means that there was only a clear protector used when encrypting the volume. In this case, the volume isn't in a protected state and needs to have a secure key added to the volume before the drive is fully protected. Administrators can use the control panel, manage-bde tool, or WMI APIs to add an appropriate key protector. Once complete, the control panel will update to reflect the new status.

Using the control panel, administrators can choose **Turn on BitLocker** to start the BitLocker Drive Encryption wizard and add a protector, like PIN for an operating system volume (or password if no TPM exists), or a password or smart card protector to a data volume. The drive security window displays prior to changing the volume status. Selecting **Activate BitLocker** will complete the encryption process.

Once BitLocker protector activation is completed, the completion notice is displayed.

Checking BitLocker status with manage-bde

Administrators who prefer a command-line interface can utilize manage-bde to check volume status. Manage-bde is capable of returning more information about the volume than the graphical user interface tools in the control panel. For example, manage-bde can display the BitLocker version in use, the encryption type, and the protectors associated with a volume.

To check the status of a volume using manage-bde, use the following command:

```
manage-bde -status <volume>
```

NOTE

If no volume letter is associated with the -status command, all volumes on the computer display their status.

Checking BitLocker status with Windows PowerShell

Windows PowerShell commands offer another way to query BitLocker status for volumes. Like manage-bde, Windows PowerShell includes the advantage of being able to check the status of a volume on a remote computer.

Using the Get-BitLockerVolume cmdlet, each volume on the system displays its current BitLocker status. To get information that is more detailed on a specific volume, use the following command:

```
Get-BitLockerVolume <volume> -Verbose | fl
```

This command displays information about the encryption method, volume type, key protectors, etc.

Provisioning BitLocker during operating system deployment

Administrators can enable BitLocker prior to operating system deployment from the Windows Pre-installation

environment. This is done with a randomly generated clear key protector applied to the formatted volume and by encrypting the volume prior to running the Windows setup process. If the encryption uses the **Used Disk Space Only** option described later in this document, this step takes only a few seconds and incorporates well into regular deployment processes.

Decrypting BitLocker volumes

Decrypting volumes removes BitLocker and any associated protectors from the volumes. Decryption should occur when protection is no longer required. BitLocker decryption shouldn't occur as a troubleshooting step. BitLocker can be removed from a volume using the BitLocker control panel applet, `manage-bde`, or Windows PowerShell cmdlets. We'll discuss each method further below.

Decrypting volumes using the BitLocker control panel applet

BitLocker decryption using the control panel is done using a wizard. The control panel can be called from Windows Explorer or by opening it directly. After opening the BitLocker control panel, users will select the **Turn off BitLocker** option to begin the process. After selecting the **Turn off BitLocker** option, the user chooses to continue by clicking the confirmation dialog. With **Turn off BitLocker** confirmed, the drive decryption process begins and reports status to the control panel.

The control panel doesn't report decryption progress but displays it in the notification area of the task bar. Selecting the notification area icon will open a modal dialog with progress.

Once decryption is complete, the drive updates its status in the control panel and becomes available for encryption.

Decrypting volumes using the `manage-bde` command-line interface

Decrypting volumes using `manage-bde` is straightforward. Decryption with `manage-bde` offers the advantage of not requiring user confirmation to start the process. `manage-bde` uses the `-off` command to start the decryption process. A sample command for decryption is:

```
manage-bde -off C:
```

This command disables protectors while it decrypts the volume and removes all protectors when decryption is complete. If users wish to check the status of the decryption, they can use the following command:

```
manage-bde -status C:
```

Decrypting volumes using the BitLocker Windows PowerShell cmdlets

Decryption with Windows PowerShell cmdlets is straightforward, similar to `manage-bde`. Windows PowerShell offers the ability to decrypt multiple drives in one pass. In the example below, the user has three encrypted volumes, which they wish to decrypt.

Using the `Disable-BitLocker` command, they can remove all protectors and encryption at the same time without the need for more commands. An example of this command is:

```
Disable-BitLocker
```

If a user didn't want to input each mount point individually, using the `-MountPoint` parameter in an array can sequence the same command into one line without requiring additional user input. An example command is:

```
Disable-BitLocker -MountPoint E:,F:,G:
```

See also

- [Prepare your organization for BitLocker: Planning and policies](#)
- [BitLocker recovery guide](#)
- [BitLocker: How to enable Network Unlock](#)
- [BitLocker overview](#)

BitLocker: How to deploy on Windows Server 2012 and later

7/1/2022 • 4 minutes to read • [Edit Online](#)

Applies to: Windows Server 2012, Windows Server 2012 R2, Windows Server 2016, Windows Server 2019

This article explains how to deploy BitLocker on Windows Server 2012 and later versions. For all Windows Server editions, BitLocker can be installed using Server Manager or Windows PowerShell cmdlets. BitLocker requires administrator privileges on the server on which it's to be installed.

Installing BitLocker

To install BitLocker using server manager

1. Open server manager by selecting the server manager icon or running servermanager.exe.
2. Select **Manage** from the **Server Manager Navigation** bar and select **Add Roles and Features** to start the **Add Roles and Features Wizard**.
3. With the **Add Roles and Features** wizard open, select **Next** at the **Before you begin** pane (if shown).
4. Select **Role-based or feature-based installation** on the **Installation type** pane of the **Add Roles and Features** wizard and select **Next** to continue.
5. Select the **Select a server from the server pool** option in the **Server Selection** pane and confirm the server on which the BitLocker feature is to be installed.
6. Select **Next** on the **Server Roles** pane of the **Add Roles and Features** wizard to proceed to the **Features** pane. **Note:** Server roles and features are installed by using the same wizard in Server Manager.
7. Select the check box next to **BitLocker Drive Encryption** within the **Features** pane of the **Add Roles and Features** wizard. The wizard shows the extra management features available for BitLocker. If you don't want to install these features, deselect the ****Include management tools**** and select **Add Features**. Once optional features selection is complete, select **Next** to proceed in the wizard.

Note: The **Enhanced Storage** feature is a required feature for enabling BitLocker. This feature enables support for encrypted hard drives on capable systems.

8. Select **Install** on the **Confirmation** pane of the **Add Roles and Features** wizard to begin BitLocker feature installation. The BitLocker feature requires a restart for its installation to be complete. Selecting the **Restart the destination server automatically if required** option in the **Confirmation** pane forces a restart of the computer after installation is complete.
9. If the **Restart the destination server automatically if required** check box isn't selected, the **Results** pane of the **Add Roles and Features** wizard displays the success or failure of the BitLocker feature installation. If necessary, a notification of other action necessary to complete the feature installation, such as the restart of the computer, will be displayed in the results text.

To install BitLocker using Windows PowerShell

Windows PowerShell offers administrators another option for BitLocker feature installation. Windows

PowerShell installs features using the `servermanager` or `dism` module; however, the `servermanager` and `dism` modules don't always share feature name parity. Because of this, it's advisable to confirm the feature or role name prior to installation.

Note: You must restart the server to complete the installation of BitLocker.

Using the servermanager module to install BitLocker

The `servermanager` Windows PowerShell module can use either the `Install-WindowsFeature` or `Add-WindowsFeature` to install the BitLocker feature. The `Add-WindowsFeature` cmdlet is merely a stub to the `Install-WindowsFeature`. This example uses the `Install-WindowsFeature` cmdlet. The feature name for BitLocker in the `servermanager` module is `BitLocker`.

By default, installation of features in Windows PowerShell doesn't include optional sub-features or management tools as part of the installation process. This can be seen using the `-WhatIf` option in Windows PowerShell.

```
Install-WindowsFeature BitLocker -WhatIf
```

The results of this command show that only the BitLocker Drive Encryption feature is installed using this command.

To see what would be installed with the BitLocker feature, including all available management tools and sub-features, use the following command:

```
Install-WindowsFeature BitLocker -IncludeAllSubFeature -IncludeManagementTools -WhatIf | fl
```

The result of this command displays the following list of all the administration tools for BitLocker, which would be installed along with the feature, including tools for use with Active Directory Domain Services (AD DS) and Active Directory Lightweight Directory Services (AD LDS).

- BitLocker Drive Encryption
- BitLocker Drive Encryption Tools
- BitLocker Drive Encryption Administration Utilities
- BitLocker Recovery Password Viewer
- AD DS Snap-Ins and Command-Line Tools
- AD DS Tools
- AD DS and AD LDS Tools

The command to complete a full installation of the BitLocker feature with all available sub-features and then to reboot the server at completion is:

```
Install-WindowsFeature BitLocker -IncludeAllSubFeature -IncludeManagementTools -Restart
```

Important: Installing the BitLocker feature using Windows PowerShell does not install the Enhanced Storage feature. Administrators wishing to support Encrypted Hard Drives in their environment will need to install the Enhanced Storage feature separately.

Using the dism module to install BitLocker

The `dism` Windows PowerShell module uses the `Enable-WindowsOptionalFeature` cmdlet to install features. The BitLocker feature name for BitLocker is `BitLocker`. The `dism` module doesn't support wildcards when searching for feature names. To list feature names for the `dism` module, use the `Get-WindowsOptionalFeatures` cmdlet. The following command will list all of the optional features in an online (running) operating system.

```
Get-WindowsOptionalFeature -Online | ft
```

From this output, we can see that there are three BitLocker-related optional feature names: BitLocker, BitLocker-Utilities and BitLocker-NetworkUnlock. To install the BitLocker feature, the BitLocker and BitLocker-Utilities features are the only required items.

To install BitLocker using the `dism` module, use the following command:

```
Enable-WindowsOptionalFeature -Online -FeatureName BitLocker -All
```

This command prompts the user for a reboot. The `Enable-WindowsOptionalFeature` cmdlet doesn't offer support for forcing a reboot of the computer. This command doesn't include installation of the management tools for BitLocker. For a complete installation of BitLocker and all available management tools, use the following command:

```
Enable-WindowsOptionalFeature -Online -FeatureName BitLocker, BitLocker-Utilities -All
```

More information

- [BitLocker overview](#)
- [BitLocker frequently asked questions \(FAQ\)](#)
- [Prepare your organization for BitLocker: Planning and policies](#)
- [BitLocker: How to enable Network Unlock](#)

BitLocker management for enterprises

7/1/2022 • 5 minutes to read • [Edit Online](#)

The ideal solution for BitLocker management is to eliminate the need for IT administrators to set management policies using tools or other mechanisms by having Windows perform tasks that are more practical to automate. This vision leverages modern hardware developments. The growth of TPM 2.0, secure boot, and other hardware improvements, for example, have helped to alleviate the support burden on the helpdesk, and we are seeing a consequent decrease in support-call volumes, yielding improved user satisfaction. Windows continues to be the focus for new features and improvements for built-in encryption management, such as automatically enabling encryption on devices that support Modern Standby beginning with Windows 8.1.

Though much Windows BitLocker [documentation](#) has been published, customers frequently ask for recommendations and pointers to specific, task-oriented documentation that is both easy to digest and focused on how to deploy and manage BitLocker. This article links to relevant documentation, products, and services to help answer this and other related frequently asked questions, and also provides BitLocker recommendations for different types of computers.

IMPORTANT

Microsoft BitLocker Administration and Monitoring (MBAM) capabilities will be offered from [ConfigMgr in on-prem scenarios](#) in the future.

Managing domain-joined computers and moving to cloud

Companies that image their own computers using Configuration Manager can use an existing task sequence to [pre-provision BitLocker](#) encryption while in Windows Preinstallation Environment (WinPE) and can then [enable protection](#). This can help ensure that computers are encrypted from the start, even before users receive them. As part of the imaging process, a company could also decide to use Configuration Manager to pre-set any desired [BitLocker Group Policy](#).

Enterprises can use [Microsoft BitLocker Administration and Monitoring \(MBAM\)](#) to manage client computers with BitLocker that are domain-joined on-premises until [mainstream support ends in July 2019](#) or they can receive extended support until April 2026. Thus, over the next few years, a good strategy for enterprises will be to plan and move to cloud-based management for BitLocker. Refer to the [PowerShell examples](#) to see how to store recovery keys in Azure Active Directory (Azure AD).

Managing devices joined to Azure Active Directory

Devices joined to Azure AD are managed using Mobile Device Management (MDM) policy from an MDM solution such as Microsoft Intune. Without Windows 10, version 1809, or Windows 11, only local administrators can enable BitLocker via Intune policy. Starting with Windows 10, version 1809, or Windows 11, Intune can enable BitLocker for standard users. [BitLocker Device Encryption](#) status can be queried from managed machines via the [Policy Configuration Settings Provider \(CSP\)](#), which reports on whether BitLocker Device Encryption is enabled on the device. Compliance with BitLocker Device Encryption policy can be a requirement for [Conditional Access](#) to services like Exchange Online and SharePoint Online.

Starting with Windows 10 version 1703 (also known as the Windows Creators Update), or Windows 11, the enablement of BitLocker can be triggered over MDM either by the [Policy CSP](#) or the [BitLocker CSP](#). The BitLocker CSP adds policy options that go beyond ensuring that encryption has occurred, and is available on computers that run Windows 11, Windows 10, and on Windows phones.

For hardware that is compliant with Modern Standby and HSTI, when using either of these features, [BitLocker Device Encryption](#) is automatically turned on whenever the user joins a device to Azure AD. Azure AD provides a portal where recovery keys are also backed up, so users can retrieve their own recovery key for self-service, if required. For older devices that are not yet encrypted, beginning with Windows 10 version 1703 (the Windows 10 Creators Update), or Windows 11, admins can use the [BitLocker CSP](#) to trigger encryption and store the recovery key in Azure AD.

This is applicable to Azure Hybrid AD as well.

Managing workplace-joined PCs and phones

For Windows PCs and Windows Phones that are enrolled using **Connect to work or school account**, BitLocker Device Encryption is managed over MDM, the same as devices joined to Azure AD.

Managing servers

Servers are often installed, configured, and deployed using PowerShell; therefore, the recommendation is to also use [PowerShell to enable BitLocker on a server](#), ideally as part of the initial setup. BitLocker is an Optional Component (OC) in Windows Server; therefore, follow the directions in [BitLocker: How to deploy on Windows Server 2012 and later](#) to add the BitLocker OC.

The Minimal Server Interface is a prerequisite for some of the BitLocker administration tools. On a [Server Core](#) installation, you must add the necessary GUI components first. The steps to add shell components to Server Core are described in [Using Features on Demand with Updated Systems and Patched Images](#) and [How to update local source media to add roles and features](#).

If you are installing a server manually, such as a stand-alone server, then choosing [Server with Desktop Experience](#) is the easiest path because you can avoid performing the steps to add a GUI to Server Core.

Additionally, lights-out data centers can take advantage of the enhanced security of a second factor while avoiding the need for user intervention during reboots by optionally using a combination of BitLocker (TPM+PIN) and BitLocker Network Unlock. BitLocker Network Unlock brings together the best of hardware protection, location dependence, and automatic unlock, while in the trusted location. For the configuration steps, see [BitLocker: How to enable Network Unlock](#).

For more information, see the Bitlocker FAQs article and other useful links in [Related Articles](#).

PowerShell examples

For Azure AD-joined computers, including virtual machines, the recovery password should be stored in Azure AD.

Example: Use PowerShell to add a recovery password and back it up to Azure AD before enabling BitLocker

```
Add-BitLockerKeyProtector -MountPoint "C:" -RecoveryPasswordProtector  
  
$BLV = Get-BitLockerVolume -MountPoint "C:"  
  
BackupToAAD-BitLockerKeyProtector -MountPoint "C:" -KeyProtectorId $BLV.KeyProtector[0].KeyProtectorId
```

For domain-joined computers, including servers, the recovery password should be stored in Active Directory Domain Services (AD DS).

Example: Use PowerShell to add a recovery password and back it up to AD DS before enabling BitLocker

```
Add-BitLockerKeyProtector -MountPoint "C:" -RecoveryPasswordProtector  
  
$BLV = Get-BitLockerVolume -MountPoint "C:"  
  
Backup-BitLockerKeyProtector -MountPoint "C:" -KeyProtectorId $BLV.KeyProtector[0].KeyProtectorId
```

Subsequently, you can use PowerShell to enable BitLocker.

Example: Use PowerShell to enable BitLocker with a TPM protector

```
Enable-BitLocker -MountPoint "D:" -EncryptionMethod XtsAes256 -UsedSpaceOnly -TpmProtector
```

Example: Use PowerShell to enable BitLocker with a TPM+PIN protector, in this case with a PIN set to 123456

```
$SecureString = ConvertTo-SecureString "123456" -AsPlainText -Force  
  
Enable-BitLocker -MountPoint "C:" -EncryptionMethod XtsAes256 -UsedSpaceOnly -Pin $SecureString -  
TPMandPinProtector
```

Related Articles

[BitLocker: FAQs](#)

[Microsoft BitLocker Administration and Management \(MBAM\)](#)

[Overview of BitLocker Device Encryption in Windows](#)

[BitLocker Group Policy Reference](#)

[Microsoft Intune \(Overview\)](#)

[Configuration Settings Providers \(Policy CSP: See *Security-RequireDeviceEncryption*\)](#)

[BitLocker CSP](#)

Windows Server setup tools

[Windows Server Installation Options](#)

[How to update local source media to add roles and features](#)

[How to add or remove optional components on Server Core \(*Features on Demand*\)](#)

[BitLocker: How to deploy on Windows Server 2012 and newer](#)

[BitLocker: How to enable Network Unlock](#)

[Shielded VMs and Guarded Fabric](#)

PowerShell

[BitLocker cmdlets for Windows PowerShell](#)

[Surface Pro Specifications](#)

BitLocker: How to enable network unlock

7/1/2022 • 21 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016 and above

This topic describes how BitLocker network unlock works and how to configure it.

Network Unlock was introduced in Windows 8 and Windows Server 2012 as a BitLocker protector option for operating system volumes. Network unlock enables easier management for BitLocker-enabled desktops and servers in a domain environment by providing automatic unlock of operating system volumes at system reboot when connected to a wired corporate network. This feature requires the client hardware to have a DHCP driver implemented in its UEFI firmware. Without Network Unlock, operating system volumes protected by TPM+PIN protectors require a PIN to be entered when a computer reboots or resumes from hibernation (for example, by Wake on LAN). This can make it difficult to enterprises to roll out software patches to unattended desktops and remotely administered servers.

Network unlock allows BitLocker-enabled systems that have a TPM+PIN and that meet the hardware requirements to boot into Windows without user intervention. Network unlock works in a similar fashion to the TPM+StartupKey at boot. Rather than needing to read the StartupKey from USB media, however, the Network Unlock feature needs the key to be composed from a key stored in the TPM and an encrypted network key that is sent to the server, decrypted and returned to the client in a secure session.

Network unlock core requirements

Network Unlock must meet mandatory hardware and software requirements before the feature can automatically unlock domain-joined systems. These requirements include:

- Windows 8 or Windows Server 2012 as the current operating system.
- Any supported operating system with UEFI DHCP drivers that can serve as Network Unlock clients.
- Network Unlock clients with a TPM chip and at least one TPM protector.
- A server running the Windows Deployment Services (WDS) role on any supported server operating system.
- BitLocker Network Unlock optional feature installed on any supported server operating system.
- A DHCP server, separate from the WDS server.
- Properly configured public/private key pairing.
- Network Unlock group policy settings configured.

The network stack must be enabled to use the Network Unlock feature. Equipment manufacturers deliver their products in various states and with different BIOS menus; therefore, you need to confirm that the network stack has been enabled in the BIOS before starting the computer.

NOTE

To properly support DHCP within UEFI, the UEFI-based system should be in native mode and shouldn't have a compatibility support module (CSM) enabled.

On computers that run Windows 8 and later, the first network adapter on the computer, usually the onboard

adapter, must be configured to support DHCP. This adapter must be used for Network Unlock.

For network unlock to work reliably on computers running Windows 8 and later versions, the first network adapter on the computer, usually the onboard adapter, must be configured to support DHCP and must be used for Network Unlock. This is especially worth noting when you have multiple adapters, and you wish to configure one without DHCP, such as for a lights-out management protocol. This configuration is necessary because network unlock stops enumerating adapters when it reaches one with a DHCP port failure for any reason. Thus, if the first enumerated adapter does not support DHCP, is not plugged into the network, or fails to report availability of the DHCP port for any reason, then Network Unlock fails.

The Network Unlock server component is installed on supported versions of Windows Server 2012 and later as a Windows feature that uses Server Manager or Windows PowerShell cmdlets. The feature name is BitLocker Network Unlock in Server Manager and BitLocker-NetworkUnlock in Windows PowerShell. This feature is a core requirement.

Network unlock requires Windows Deployment Services (WDS) in the environment where the feature will be utilized. Configuration of the WDS installation is not required; however, the WDS service must be running on the server.

The network key is stored on the system drive along with an AES 256 session key and encrypted with the 2048-bit RSA public key of the Unlock server certificate. The network key is decrypted with the help of a provider on a supported version of Windows Server running WDS, and returned encrypted with its corresponding session key.

Network Unlock sequence

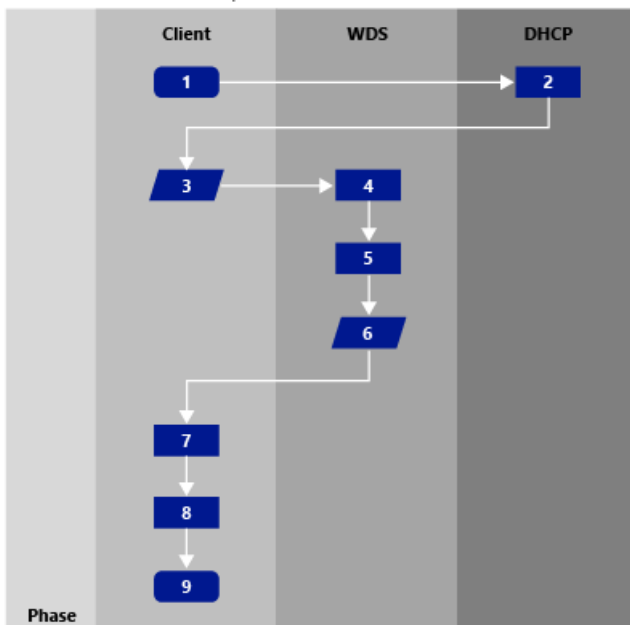
The unlock sequence starts on the client side when the Windows boot manager detects the existence of network unlock protector. It leverages the DHCP driver in UEFI to obtain an IP address for IPv4 and then broadcasts a vendor-specific DHCP request that contains the network key and a session key for the reply, all encrypted by the server's Network Unlock certificate, as described above. The Network Unlock provider on the supported WDS server recognizes the vendor-specific request, decrypts it with the RSA private key, and returns the network key encrypted with the session key via its own vendor-specific DHCP reply.

On the server side, the WDS server role has an optional plugin component, like a PXE provider, which is what handles the incoming network unlock requests. You can also configure the provider with subnet restrictions, which would require that the IP address provided by the client in the network unlock request belong to a permitted subnet to release the network key to the client. In instances where the Network Unlock provider is unavailable, BitLocker fails over to the next available protector to unlock the drive. In a typical configuration, this means the standard TPM+PIN unlock screen is presented to unlock the drive.

The server side configuration to enable Network Unlock also requires provisioning a 2048-bit RSA public/private key pair in the form of an X.509 certificate, and distributing the public key certificate to the clients. This certificate must be managed and deployed through the Group Policy editor directly on a domain controller with at least a Domain Functional Level of Windows Server 2012. This certificate is the public key that encrypts the intermediate network key (which is one of the two secrets required to unlock the drive; the other secret is stored in the TPM).

Manage and deploy this certificate through the Group Policy editor directly on a domain controller that has a domain functional level of at least Windows Server 2012. This certificate is the public key that encrypts the intermediate network key. The intermediate network key is one of the two secrets that are required to unlock the drive; the other secret is stored in the TPM.

BitLocker Network Unlock process



The Network Unlock process follows these phases:

1. The Windows boot manager detects a Network Unlock protector in the BitLocker configuration.
2. The client computer uses its DHCP driver in the UEFI to get a valid IPv4 IP address.
3. The client computer broadcasts a vendor-specific DHCP request that contains:
 - a. A network key (a 256-bit intermediate key) that is encrypted by using the 2048-bit RSA Public Key of the network unlock certificate from the WDS server.
 - b. An AES-256 session key for the reply.
4. The Network Unlock provider on the WDS server recognizes the vendor-specific request.
5. The provider decrypts the request by using the WDS server's BitLocker Network Unlock certificate RSA private key.
6. The WDS provider returns the network key encrypted with the session key by using its own vendor-specific DHCP reply to the client computer. This key is an intermediate key.
7. The returned intermediate key is combined with another local 256-bit intermediate key. This key can be decrypted only by the TPM.
8. This combined key is used to create an AES-256 key that unlocks the volume.
9. Windows continues the boot sequence.

Configure network unlock

The following steps allow an administrator to configure network unlock in a domain where the Domain Functional Level is at least Windows Server 2012.

Install the WDS server role

The BitLocker network unlock feature installs the WDS role if it is not already installed. If you want to install it separately before you install BitLocker network unlock, you can use Server Manager or Windows PowerShell. To install the role using Server Manager, select the **Windows Deployment Services** role in Server Manager.

To install the role by using Windows PowerShell, use the following command:

```
Install-WindowsFeature WDS-Deployment
```

You must configure the WDS server so that it can communicate with DHCP (and optionally AD DS) and the client computer. You can configure using the WDS management tool, `wdsmgmt.msc`, which starts the Windows

Deployment Services Configuration wizard.

Confirm the WDS service is running

To confirm that the WDS service is running, use the Services Management Console or Windows PowerShell. To confirm that the service is running in Services Management Console, open the console using `services.msc` and check the status of the Windows Deployment Services service.

To confirm that the service is running using Windows PowerShell, use the following command:

```
Get-Service WDSserver
```

Install the Network Unlock feature

To install the network unlock feature, use Server Manager or Windows PowerShell. To install the feature using Server Manager, select the **BitLocker Network Unlock** feature in the Server Manager console.

To install the feature by using Windows PowerShell, use the following command:

```
Install-WindowsFeature BitLocker-NetworkUnlock
```

Create the certificate template for Network Unlock

A properly configured Active Directory Services Certification Authority can use this certificate template to create and issue Network Unlock certificates.

1. Open the Certificates Template snap-in (certtmpl.msc).
2. Locate the User template, right-click the template name and select **Duplicate Template**.
3. On the **Compatibility** tab, change the **Certification Authority** and **Certificate recipient** fields to Windows Server 2012 and Windows 8, respectively. Ensure that the **Show resulting changes** dialog box is selected.
4. Select the **General** tab of the template. The **Template display name** and **Template name** should clearly identify that the template will be used for Network Unlock. Clear the check box for the **Publish certificate in Active Directory** option.
5. Select the **Request Handling** tab. Select **Encryption** from the **Purpose** drop-down menu. Ensure that the **Allow private key to be exported** option is selected.
6. Select the **Cryptography** tab. Set the **Minimum key size** to 2048. (Any Microsoft cryptographic provider that supports RSA can be used for this template, but for simplicity and forward compatibility, we recommend using **Microsoft Software Key Storage Provider**.)
7. Select the **Requests must use one of the following providers** option and clear all options except for the cryptography provider you selected, such as **Microsoft Software Key Storage Provider**.
8. Select the **Subject Name** tab. Select **Supply in the request**. Click **OK** if the certificate templates pop-up dialog appears.
9. Select the **Issuance Requirements** tab. Select both **CA certificate manager approval** and **Valid existing certificate** options.
10. Select the **Extensions** tab. Select **Application Policies** and choose **Edit...**
11. In the **Edit Application Policies Extension** options dialog box, select **Client Authentication**, **Encrypting File System**, and **Secure Email** and choose **Remove**.
12. On the **Edit Application Policies Extension** dialog box, select **Add**.

13. On the **Add Application Policy** dialog box, select **New**. In the **New Application Policy** dialog box, enter the following information in the space provided and then click **OK** to create the BitLocker Network Unlock application policy:
 - **Name:** BitLocker Network Unlock
 - **Object Identifier:** 1.3.6.1.4.1.311.67.1.1
14. Select the newly created **BitLocker Network Unlock** application policy and click **OK**.
15. With the **Extensions** tab still open, select the **Edit Key Usage Extension** dialog. Select the **Allow key exchange only with key encryption (key encipherment)** option. Select the **Make this extension critical** option.
16. Select the **Security** tab. Confirm that the **Domain Admins** group has been granted **Enroll** permission.
17. Click **OK** to complete configuration of the template.

To add the Network Unlock template to the certificate authority, open the certificate authority snap-in (`certsrv.msc`). Right-click **Certificate Templates**, and then choose **New, Certificate Template to issue**. Select the previously created BitLocker Network Unlock certificate.

After you add the Network Unlock template to the certificate authority, you can use this certificate to configure BitLocker Network Unlock.

Create the Network Unlock certificate

Network Unlock can use imported certificates from an existing public key infrastructure (PKI). Or it can use a self-signed certificate.

To enroll a certificate from an existing certificate authority:

1. On the WDS server, open Certificate Manager by using `certmgr.msc`.
2. Under **Certificates - Current User**, right-click **Personal**.
3. Select **All Tasks > Request New Certificate**.
4. When the Certificate Enrollment wizard opens, select **Next**.
5. Select **Active Directory Enrollment Policy**.
6. Choose the certificate template that was created for Network Unlock on the domain controller. Then select **Enroll**.
7. When you're prompted for more information, select **Subject Name** and provide a friendly name value. Your friendly name should include information for the domain or organizational unit for the certificate. Here's an example: *BitLocker Network Unlock Certificate for Contoso domain*.
8. Create the certificate. Ensure the certificate appears in the **Personal** folder.
9. Export the public key certificate for Network Unlock:
 - a. Create a .cer file by right-clicking the previously created certificate, selecting **All Tasks**, and then selecting **Export**.
 - b. Select **No, do not export the private key**.
 - c. Select **DER encoded binary X.509** and complete exporting the certificate to a file.
 - d. Give the file a name such as BitLocker-NetworkUnlock.cer.
10. Export the public key with a private key for Network Unlock.
 - a. Create a .pfx file by right-clicking the previously created certificate, selecting **All Tasks**, and then selecting **Export**.

- b. Select **Yes, export the private key**.
- c. Complete the steps to create the *.pfx* file.

To create a self-signed certificate, either use the `New-SelfSignedCertificate` cmdlet in Windows PowerShell or use `certreq`.

Here's a Windows PowerShell example:

```
New-SelfSignedCertificate -CertStoreLocation Cert:\LocalMachine\My -Subject "CN=BitLocker Network Unlock certificate" -Provider "Microsoft Software Key Storage Provider" -KeyUsage KeyEncipherment -KeyUsageProperty Decrypt,Sign -KeyLength 2048 -HashAlgorithm sha512 -TextExtension @("1.3.6.1.4.1.311.21.10={text}0ID=1.3.6.1.4.1.311.67.1.1", "2.5.29.37={text}1.3.6.1.4.1.311.67.1.1")
```

Here's a `certreq` example:

1. Create a text file with an *.inf* extension, for example, `notepad.exe BitLocker-NetworkUnlock.inf`.
2. Add the following contents to the previously created file:

```
[NewRequest]
Subject="CN=BitLocker Network Unlock certificate"
ProviderType=0
MachineKeySet=True
Exportable=true
RequestType=Cert
KeyUsage="CERT_KEY_ENCIPHERMENT_KEY_USAGE"
KeyUsageProperty="NCRYPT_ALLOW_DECRYPT_FLAG | NCRYPT_ALLOW_SIGNING_FLAG"
KeyLength=2048
SMIME=FALSE
HashAlgorithm=sha512
[Extensions]
1.3.6.1.4.1.311.21.10 = "{text}"
_continue_ = "0ID=1.3.6.1.4.1.311.67.1.1"
2.5.29.37 = "{text}"
_continue_ = "1.3.6.1.4.1.311.67.1.1"
```

3. Open an elevated command prompt and use the `certreq` tool to create a new certificate. Use the following command, specifying the full path to the file that you created previously. Also specify the file name.

```
certreq -new BitLocker-NetworkUnlock.inf BitLocker-NetworkUnlock.cer
```

4. Verify that certificate was properly created by the previous command by confirming that the *.cer* file exists.
5. Launch Certificates - Local Machine by running `certlm.msc`.
6. Create a *.pfx* file by opening the **Certificates – Local Computer\Personal\Certificates** path in the navigation pane, right-clicking the previously imported certificate, selecting **All Tasks**, and then selecting **Export**. Follow through the wizard to create the *.pfx* file.

Deploy the private key and certificate to the WDS server

Now that you've created the certificate and key, deploy them to the infrastructure to properly unlock systems. To deploy the certificates:

1. On the WDS server, open a new MMC and add the certificates snap-in. Select the computer account and local computer when given the options.
2. Right-click the Certificates (Local Computer) - BitLocker Drive Encryption Network Unlock item -, select **All**

Tasks, and then select **Import**.

3. In the **File to Import** dialog, choose the .pfx file created previously.
4. Enter the password used to create the .pfx and complete the wizard.

Configure group policy settings for network unlock

With certificate and key deployed to the WDS server for Network Unlock, the final step is to use group policy settings to deploy the public key certificate to computers that you want to be able to unlock using the Network Unlock key. Group policy settings for BitLocker can be found under **\Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption** using the Local Group Policy Editor or the Microsoft Management Console.

The following steps describe how to enable the group policy setting that is a requirement for configuring network unlock.

1. Open Group Policy Management Console (`gpmc.msc`).
2. Enable the policy **Require additional authentication at startup**, and then select **Require startup PIN with TPM** or **Allow startup PIN with TPM**.
3. Turn on BitLocker with TPM+PIN protectors on all domain-joined computers.

The following steps describe how to deploy the required group policy setting:

NOTE

The group policy settings **Allow network unlock at startup** and **Add Network Unlock Certificate** were introduced in Windows Server 2012.

1. Copy the .cer file that you created for Network Unlock to the domain controller.
2. On the domain controller, open Group Policy Management Console (`gpmc.msc`).
3. Create a new Group Policy Object or modify an existing object to enable the **Allow network unlock at startup** setting.
4. Deploy the public certificate to clients:
 - a. Within group policy management console, navigate to the following location: **Computer Configuration\Policies\Windows Settings\Security Settings\Public Key Policies\BitLocker Drive Encryption Network Unlock Certificate**.
 - b. Right-click the folder and select **Add Network Unlock Certificate**.
 - c. Follow the wizard steps and import the .cer file that was copied earlier.

NOTE

Only one network unlock certificate can be available at a time. If you need a new certificate, delete the current certificate before you deploy a new one. The Network Unlock certificate is located in the `HKEY_LOCAL_MACHINE\Software\Policies\Microsoft\SystemCertificates\FVE_NKP` key on the client computer.

5. Reboot the clients after you deploy the Group Policy.

NOTE

The **Network (Certificate Based)** protector will be added only after a reboot, with the policy enabled and a valid certificate present in the FVE_NKP store.

Subnet policy configuration files on the WDS server (optional)

By default, all clients with the correct network unlock certificate and valid Network Unlock protectors that have wired access to a network unlock-enabled WDS server via DHCP are unlocked by the server. A subnet policy configuration file on the WDS server can be created to limit which are the subnet(s) the network unlock clients can use to unlock.

The configuration file, called `bde-network-unlock.ini`, must be located in the same directory as the network unlock provider DLL (`%windir%\System32\Nkpprov.dll`) and it applies to both IPv6 and IPv4 DHCP implementations. If the subnet configuration policy becomes corrupted, the provider fails and stops responding to requests.

The subnet policy configuration file must use a "[SUBNETS]" section to identify the specific subnets. The named subnets may then be used to specify restrictions in certificate subsections. Subnets are defined as simple name-value pairs, in the common INI format, where each subnet has its own line, with the name on the left of the equal-sign, and the subnet identified on the right of the equal-sign as a Classless Inter-Domain Routing (CIDR) address or range. The key word "ENABLED" is disallowed for subnet names.

```
[SUBNETS]
SUBNET1=10.185.250.0/24 ; a comment about this subrange could be here, after the semicolon
SUBNET2=10.185.252.200/28
SUBNET3= 2001:4898:a:2::/64 ; an IPv6 subnet
SUBNET4=2001:4898:a:3::/64; in production, the admin would likely give more useful names, like BUILDING9-EXCEPT-RECEP.
```

Following the [SUBNETS] section, there can be sections for each Network Unlock certificate, identified by the certificate thumbprint formatted without any spaces, which define the subnets clients that can be unlocked from that certificate.

NOTE

When specifying the certificate thumbprint, do not include any spaces. If spaces are included in the thumbprint, the subnet configuration fails because the thumbprint will not be recognized as valid.

Subnet restrictions are defined within each certificate section by denoting the allowed list of permitted subnets. If any subnets are listed in a certificate section, then only those subnets are permitted for that certificate. If no subnet is listed in a certificate section, then all subnets are permitted for that certificate. If a certificate does not have a section in the subnet policy configuration file, then no subnet restrictions are applied for unlocking with that certificate. This means for restrictions to apply to every certificate, there must be a certificate section for every network unlock certificate on the server, and an explicit allowed list set for each certificate section. Subnet lists are created by putting the name of a subnet from the [SUBNETS] section on its own line below the certificate section header. Then, the server will only unlock clients with this certificate on the subnet(s) specified as in the list. For troubleshooting, a subnet can be quickly excluded without deleting it from the section by simply commenting it out with a prepended semi-colon.

```
[2158a767e1c14e88e27a4c0aee111d2de2eafe60]
;Comments could be added here to indicate when the cert was issued, which Group Policy should get it, and so on.
;This list shows this cert is allowed to unlock clients only on the SUBNET1 and SUBNET3 subnets. In this example, SUBNET2 is commented out.
SUBNET1
;SUBNET2
SUBNET3
```

To disallow the use of a certificate altogether, add a `DISABLED` line to its subnet list.

Turn off Network Unlock

To turn off the unlock server, the PXE provider can be unregistered from the WDS server or uninstalled altogether. However, to stop clients from creating network unlock protectors, the **Allow Network Unlock at startup** group policy setting should be disabled. When this policy setting is updated to **disabled** on client computers, any Network Unlock key protector on the computer is deleted. Alternatively, the BitLocker network unlock certificate policy can be deleted on the domain controller to accomplish the same task for an entire domain.

NOTE

Removing the FVE_NKP certificate store that contains the network unlock certificate and key on the WDS server will also effectively disable the server's ability to respond to unlock requests for that certificate. However, this is seen as an error condition and is not a supported or recommended method for turning off the network unlock server.

Update Network Unlock certificates

To update the certificates used by network unlock, administrators need to import or generate the new certificate for the server and then update the network unlock certificate group policy setting on the domain controller.

NOTE

Servers that don't receive the Group Policy Object (GPO) will require a PIN when they boot. In such cases, find out why the server didn't receive the GPO to update the certificate.

Troubleshoot Network Unlock

Troubleshooting network unlock issues begins by verifying the environment. Many times, a small configuration issue can be the root cause of the failure. Items to verify include:

- Verify that the client hardware is UEFI-based and is on firmware version 2.3.1 and that the UEFI firmware is in native mode without a Compatibility Support Module (CSM) for BIOS mode enabled. Do this by checking that the firmware does not have an option enabled such as "Legacy mode" or "Compatibility mode" or that the firmware does not appear to be in a BIOS-like mode.
- All required roles and services are installed and started.
- Public and private certificates have been published and are in the proper certificate containers. The presence of the network unlock certificate can be verified in the Microsoft Management Console (MMC.exe) on the WDS server with the certificate snap-ins for the local computer enabled. The client certificate can be verified by checking the registry key `HKEY_LOCAL_MACHINE\Software\Policies\Microsoft\SystemCertificates\FVE_NKP` on the client computer.
- Group policy for network unlock is enabled and linked to the appropriate domains.
- Verify whether group policy is reaching the clients properly. This can be done using the GPRESULT.exe or RSOP.msc utilities.
- Verify whether the clients were rebooted after applying the policy.
- Verify whether the **Network (Certificate Based)** protector is listed on the client. This can be done using either manage-bde or Windows PowerShell cmdlets. For example, the following command will list the key protectors currently configured on the C: drive of the local computer:

```
manage-bde -protectors -get C:
```

NOTE

Use the output of `manage-bde` along with the WDS debug log to determine whether the proper certificate thumbprint is being used for Network Unlock.

Gather the following files to troubleshoot BitLocker Network Unlock.

- The Windows event logs. Specifically, get the BitLocker event logs and the Microsoft-Windows-Deployment-Services-Diagnostics-Debug log.

Debug logging is turned off by default for the WDS server role, so you need to enable it before you can retrieve it. Use either of the following two methods to turn on WDS debug logging.

- Start an elevated command prompt, and then run the following command:

```
weventutil s1 Microsoft-Windows-Deployment-Services-Diagnostics/Debug /e:true
```

- Open Event Viewer on the WDS server:

1. In the left pane, select **Applications and Services Logs > Microsoft > Windows > Deployment-Services-Diagnostics > Debug**.
2. In the right pane, select **Enable Log**.

- The DHCP subnet configuration file (if one exists).
- The output of the BitLocker status on the volume. Gather this output into a text file by using `manage-bde -status`. Or in Windows PowerShell, use `Get-BitLockerVolume`.
- The Network Monitor capture on the server that hosts the WDS role, filtered by client IP address.

Configure Network Unlock Group Policy settings on earlier versions

Network Unlock and the accompanying Group Policy settings were introduced in Windows Server 2012. But you can deploy them by using operating systems that run Windows Server 2008 R2 and Windows Server 2008.

Your system must meet these requirements:

- The server that hosts WDS must be running a server operating system that's designated in the "Applies to" list at the beginning of this article.
- Client computers must be running a client operating system that's designated in the "Applies to" list at the beginning of this article.

Follow these steps to configure Network Unlock on these older systems.

1. [Install the WDS Server role](#)
2. [Confirm the WDS Service is running](#)
3. [Install the Network Unlock feature](#)
4. [Create the Network Unlock certificate](#)
5. [Deploy the private key and certificate to the WDS server](#)
6. Configure registry settings for network unlock:

Apply the registry settings by running the following `certutil` script (assuming your Network Unlock certificate file is called *BitLocker-NetworkUnlock.cer*) on each computer that runs a client operating system that's designated in the "Applies to" list at the beginning of this article.

```
certutil -f -grouppolicy -addstore FVE_NKP BitLocker-NetworkUnlock.cer
reg add "HKLM\SOFTWARE\Policies\Microsoft\FVE" /v OSManageNKP /t REG_DWORD /d 1 /f
reg add "HKLM\SOFTWARE\Policies\Microsoft\FVE" /v UseAdvancedStartup /t REG_DWORD /d 1 /f
reg add "HKLM\SOFTWARE\Policies\Microsoft\FVE" /v UsePIN /t REG_DWORD /d 2 /f
reg add "HKLM\SOFTWARE\Policies\Microsoft\FVE" /v UseTPMPIN /t REG_DWORD /d 2 /f
reg add "HKLM\SOFTWARE\Policies\Microsoft\FVE" /v UseTPM /t REG_DWORD /d 2 /f
reg add "HKLM\SOFTWARE\Policies\Microsoft\FVE" /v UseTPMKey /t REG_DWORD /d 2 /f
reg add "HKLM\SOFTWARE\Policies\Microsoft\FVE" /v UseTPMKeyPIN /t REG_DWORD /d 2 /f
```

7. Set up a TPM protector on the clients.
8. Reboot the clients to add the Network (certificate based) protector.

See also

- [BitLocker overview](#)
- [BitLocker frequently asked questions \(FAQ\)](#)
- [Prepare your organization for BitLocker: Planning and policies](#)

BitLocker: Use BitLocker Drive Encryption Tools to manage BitLocker

7/1/2022 • 10 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016 and above

This article for the IT professional describes how to use tools to manage BitLocker.

BitLocker Drive Encryption Tools include the command-line tools `manage-bde` and `repair-bde` and the BitLocker cmdlets for Windows PowerShell.

Both `manage-bde` and the BitLocker cmdlets can be used to perform any task that can be accomplished through the BitLocker control panel and are appropriate to use for automated deployments and other scripting scenarios.

`Repair-bde` is a special circumstance tool that is provided for disaster recovery scenarios in which a BitLocker protected drive cannot be unlocked normally or using the recovery console.

1. [Manage-bde](#)
2. [Repair-bde](#)
3. [BitLocker cmdlets for Windows PowerShell](#)

Manage-bde

`Manage-bde` is a command-line tool that can be used for scripting BitLocker operations. `Manage-bde` offers additional options not displayed in the BitLocker control panel. For a complete list of the `manage-bde` options, see the [Manage-bde](#) command-line reference.

`Manage-bde` includes fewer default settings and requires greater customization for configuring BitLocker. For example, using just the `manage-bde -on` command on a data volume will fully encrypt the volume without any authenticating protectors. A volume encrypted in this manner still requires user interaction to turn on BitLocker protection, even though the command successfully completed because an authentication method needs to be added to the volume for it to be fully protected. The following sections provide examples of common usage scenarios for `manage-bde`.

Using `manage-bde` with operating system volumes

Listed below are examples of basic valid commands for operating system volumes. In general, using only the `manage-bde -on <drive letter>` command will encrypt the operating system volume with a TPM-only protector and no recovery key. However, many environments require more secure protectors such as passwords or PIN and expect to be able to recover information with a recovery key. We recommend that you add at least one primary protector and a recovery protector to an operating system volume.

A good practice when using `manage-bde` is to determine the volume status on the target system. Use the following command to determine volume status:

```
manage-bde -status
```

This command returns the volumes on the target, current encryption status, encryption method, and volume type (operating system or data) for each volume:

```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\windows\system32>manage-bde -status
BitLocker Drive Encryption: Configuration Tool version 10.0.15063
Copyright (C) 2013 Microsoft Corporation. All rights reserved.

Disk volumes that can be protected with
BitLocker Drive Encryption:
Volume C: [OSDisk]
[OS Volume]

Size:                465.27 GB
BitLocker Version:   2.0
Conversion Status:   Fully Encrypted
Percentage Encrypted: 100.0%
Encryption Method:   XTS-AES 128
Protection Status:   Protection On
Lock Status:         Unlocked
Identification Field: Unknown
Key Protectors:
    Numerical Password
    TPM
```

The following example illustrates enabling BitLocker on a computer without a TPM chip. Before beginning the encryption process, you must create the startup key needed for BitLocker and save it to the USB drive. When BitLocker is enabled for the operating system volume, the BitLocker will need to access the USB flash drive to obtain the encryption key (in this example, the drive letter E represents the USB drive). You will be prompted to reboot to complete the encryption process.

```
manage-bde -protectors -add C: -startupkey E:
manage-bde -on C:
```

NOTE

After the encryption is completed, the USB startup key must be inserted before the operating system can be started.

An alternative to the startup key protector on non-TPM hardware is to use a password and an **ADaccountorgroup** protector to protect the operating system volume. In this scenario, you would add the protectors first. To add them, use this command:

```
manage-bde -protectors -add C: -pw -sid <user or group>
```

This command will require you to enter and then confirm the password protector before adding them to the volume. With the protectors enabled on the volume, you can then turn on BitLocker.

On computers with a TPM, it is possible to encrypt the operating system volume without any defined protectors using `manage-bde`. Use this command:

```
manage-bde -on C:
```

This command encrypts the drive using the TPM as the default protector. If you are not sure if a TPM protector is available, to list the protectors available for a volume, run the following command:


```
manage-bde -protectors -get <volume>
```

Using manage-bde with data volumes

Data volumes use the same syntax for encryption as operating system volumes but they do not require protectors for the operation to complete. Encrypting data volumes can be done using the base command:

```
manage-bde -on <drive letter>
```

 or you can choose to add additional protectors to the volume first. We recommend that you add at least one primary protector and a recovery protector to a data volume.

A common protector for a data volume is the password protector. In the example below, we add a password protector to the volume and turn on BitLocker.

```
manage-bde -protectors -add -pw C:  
manage-bde -on C:
```

Repair-bde

You may experience a problem that damages an area of a hard disk on which BitLocker stores critical information. This kind of problem may be caused by a hard disk failure or if Windows exits unexpectedly.

The BitLocker Repair Tool (Repair-bde) can be used to access encrypted data on a severely damaged hard disk if the drive was encrypted by using BitLocker. Repair-bde can reconstruct critical parts of the drive and salvage recoverable data as long as a valid recovery password or recovery key is used to decrypt the data. If the BitLocker metadata data on the drive has become corrupt, you must be able to supply a backup key package in addition to the recovery password or recovery key. This key package is backed up in Active Directory Domain Services (AD DS) if you used the default setting for AD DS backup. With this key package and either the recovery password or recovery key, you can decrypt portions of a BitLocker-protected drive if the disk is corrupted. Each key package will work only for a drive that has the corresponding drive identifier. You can use the BitLocker Recovery Password Viewer to obtain this key package from AD DS.

TIP

If you are not backing up recovery information to AD DS or if you want to save key packages alternatively, you can use the command `manage-bde -KeyPackage` to generate a key package for a volume.

The Repair-bde command-line tool is intended for use when the operating system does not start or when you cannot start the BitLocker Recovery Console. Use Repair-bde if the following conditions are true:

- You have encrypted the drive by using BitLocker Drive Encryption.
- Windows does not start, or you cannot start the BitLocker recovery console.
- You do not have a copy of the data that is contained on the encrypted drive.

NOTE

Damage to the drive may not be related to BitLocker. Therefore, we recommend that you try other tools to help diagnose and resolve the problem with the drive before you use the BitLocker Repair Tool. The Windows Recovery Environment (Windows RE) provides additional options to repair computers.

The following limitations exist for Repair-bde:

- The Repair-bde command-line tool cannot repair a drive that failed during the encryption or decryption process.

- The Repair-bde command-line tool assumes that if the drive has any encryption, then the drive has been fully encrypted.

For more information about using repair-bde, see [Repair-bde](#).

BitLocker cmdlets for Windows PowerShell

Windows PowerShell cmdlets provide a new way for administrators to use when working with BitLocker. Using Windows PowerShell's scripting capabilities, administrators can integrate BitLocker options into existing scripts with ease. The list below displays the available BitLocker cmdlets.

NAME	PARAMETERS
Add-BitLockerKeyProtector	<ul style="list-style-type: none"> • ADAccountOrGroup • ADAccountOrGroupProtector • Confirm • MountPoint • Password • PasswordProtector • Pin • RecoveryKeyPath • RecoveryKeyProtector • RecoveryPassword • RecoveryPasswordProtector • Service • StartupKeyPath • StartupKeyProtector • TpmAndPinAndStartupKeyProtector • TpmAndPinProtector • TpmAndStartupKeyProtector • TpmProtector • WhatIf
Backup-BitLockerKeyProtector	<ul style="list-style-type: none"> • Confirm • KeyProtectorId • MountPoint • WhatIf
Disable-BitLocker	<ul style="list-style-type: none"> • Confirm • MountPoint • WhatIf
Disable-BitLockerAutoUnlock	<ul style="list-style-type: none"> • Confirm • MountPoint • WhatIf

NAME	PARAMETERS
Enable-BitLocker	<ul style="list-style-type: none"> • AdAccountOrGroup • AdAccountOrGroupProtector • Confirm • EncryptionMethod • HardwareEncryption • Password • PasswordProtector • Pin • RecoveryKeyPath • RecoveryKeyProtector • RecoveryPassword • RecoveryPasswordProtector • Service • SkipHardwareTest • StartupKeyPath • StartupKeyProtector • TpmAndPinAndStartupKeyProtector • TpmAndPinProtector • TpmAndStartupKeyProtector • TpmProtector • UsedSpaceOnly • WhatIf
Enable-BitLockerAutoUnlock	<ul style="list-style-type: none"> • Confirm • MountPoint • WhatIf
Get-BitLockerVolume	<ul style="list-style-type: none"> • MountPoint
Lock-BitLocker	<ul style="list-style-type: none"> • Confirm • ForceDismount • MountPoint • WhatIf
Remove-BitLockerKeyProtector	<ul style="list-style-type: none"> • Confirm • KeyProtectorId • MountPoint • WhatIf
Resume-BitLocker	<ul style="list-style-type: none"> • Confirm • MountPoint • WhatIf
Suspend-BitLocker	<ul style="list-style-type: none"> • Confirm • MountPoint • RebootCount • WhatIf
Unlock-BitLocker	<ul style="list-style-type: none"> • AdAccountOrGroup • Confirm • MountPoint • Password • RecoveryKeyPath • RecoveryPassword • RecoveryPassword • WhatIf

Similar to manage-bde, the Windows PowerShell cmdlets allow configuration beyond the options offered in the

control panel. As with manage-bde, users need to consider the specific needs of the volume they are encrypting prior to running Windows PowerShell cmdlets.

A good initial step is to determine the current state of the volume(s) on the computer. You can do this using the `Get-BitLockerVolume` cmdlet.

The `Get-BitLockerVolume` cmdlet output gives information on the volume type, protectors, protection status, and other details.

TIP

Occasionally, all protectors may not be shown when using `Get-BitLockerVolume` due to lack of space in the output display. If you do not see all of the protectors for a volume, you can use the Windows PowerShell pipe command (`|`) to format a full listing of the protectors. `Get-BitLockerVolume C: | fl`

If you want to remove the existing protectors prior to provisioning BitLocker on the volume, you could use the `Remove-BitLockerKeyProtector` cmdlet. Accomplishing this requires the GUID associated with the protector to be removed.

A simple script can pipe the values of each `Get-BitLockerVolume` return out to another variable as seen below:

```
$vol = Get-BitLockerVolume
$keyprotectors = $vol.KeyProtector
```

By using this script, you can display the information in the `$keyprotectors` variable to determine the GUID for each protector.

By using this information, you can then remove the key protector for a specific volume using the command:

```
Remove-BitLockerKeyProtector <volume>: -KeyProtectorID "{GUID}"
```

NOTE

The BitLocker cmdlet requires the key protector GUID enclosed in quotation marks to execute. Ensure the entire GUID, with braces, is included in the command.

Using the BitLocker Windows PowerShell cmdlets with operating system volumes

Using the BitLocker Windows PowerShell cmdlets is similar to working with the manage-bde tool for encrypting operating system volumes. Windows PowerShell offers users a lot of flexibility. For example, users can add the desired protector as part command for encrypting the volume. Below are examples of common user scenarios and steps to accomplish them in BitLocker Windows PowerShell.

The following example shows how to enable BitLocker on an operating system drive using only the TPM protector:

```
Enable-BitLocker C:
```

In the example below, adds one additional protector, the StartupKey protector and chooses to skip the BitLocker hardware test. In this example, encryption starts immediately without the need for a reboot.

```
Enable-BitLocker C: -StartupKeyProtector -StartupKeyPath <path> -SkipHardwareTest
```

Using the BitLocker Windows PowerShell cmdlets with data volumes

Data volume encryption using Windows PowerShell is the same as for operating system volumes. Add the desired protectors prior to encrypting the volume. The following example adds a password protector to the E: volume using the variable \$pw as the password. The \$pw variable is held as a SecureString value to store the user-defined password.

```
$pw = Read-Host -AsSecureString  
<user inputs password>  
Enable-BitLockerKeyProtector E: -PasswordProtector -Password $pw
```

Using an AD Account or Group protector in Windows PowerShell

The **ADAccountOrGroup** protector, introduced in Windows 8 and Windows Server 2012, is an Active Directory SID-based protector. This protector can be added to both operating system and data volumes, although it does not unlock operating system volumes in the pre-boot environment. The protector requires the SID for the domain account or group to link with the protector. BitLocker can protect a cluster-aware disk by adding a SID-based protector for the Cluster Name Object (CNO) that lets the disk properly fail over to and be unlocked by any member computer of the cluster.

WARNING

The **ADAccountOrGroup** protector requires the use of an additional protector for use (such as TPM, PIN, or recovery key) when used on operating system volumes

To add an **ADAccountOrGroup** protector to a volume, use either the actual domain SID or the group name preceded by the domain and a backslash. In the example below, the CONTOSO\Administrator account is added as a protector to the data volume G.

```
Enable-BitLocker G: -AdAccountOrGroupProtector -AdAccountOrGroup CONTOSO\Administrator
```

For users who wish to use the SID for the account or group, the first step is to determine the SID associated with the account. To get the specific SID for a user account in Windows PowerShell, use the following command:

NOTE

Use of this command requires the RSAT-AD-PowerShell feature.

```
get-aduser -filter {samaccountname -eq "administrator"}
```

TIP

In addition to the PowerShell command above, information about the locally logged on user and group membership can be found using: WHOAMI /ALL. This does not require the use of additional features.

The following example adds an **ADAccountOrGroup** protector to the previously encrypted operating system volume using the SID of the account:

```
Add-BitLockerKeyProtector C: -ADAccountOrGroupProtector -ADAccountOrGroup S-1-5-21-3651336348-8937238915-291003330-500
```

NOTE

Active Directory-based protectors are normally used to unlock Failover Cluster enabled volumes.

More information

- [BitLocker overview](#)
- [BitLocker frequently asked questions \(FAQ\)](#)
- [Prepare your organization for BitLocker: Planning and policies](#)
- [BitLocker: How to enable Network Unlock](#)
- [BitLocker: How to deploy on Windows Server 2012](#)

BitLocker: Use BitLocker Recovery Password Viewer

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016 and above

This topic describes how to use the BitLocker Recovery Password Viewer.

The BitLocker Recovery Password Viewer tool is an optional tool included with the Remote Server Administration Tools (RSAT). It lets you locate and view BitLocker recovery passwords that are stored in Active Directory Domain Services (AD DS). You can use this tool to help recover data that is stored on a drive that has been encrypted by using BitLocker. The BitLocker Active Directory Recovery Password Viewer tool is an extension for the Active Directory Users and Computers Microsoft Management Console (MMC) snap-in. Using this tool, you can examine a computer object's **Properties** dialog box to view the corresponding BitLocker recovery passwords. Additionally you can right-click a domain container and then search for a BitLocker recovery password across all the domains in the Active Directory forest. You can also search for a password by password identifier (ID).

Before you start

To complete the procedures in this scenario:

- You must have domain administrator credentials.
- Your test computers must be joined to the domain.
- On the domain-joined test computers, BitLocker must have been turned on.

The following procedures describe the most common tasks performed by using the BitLocker Recovery Password Viewer.

To view the recovery passwords for a computer

1. In **Active Directory Users and Computers**, locate and then click the container in which the computer is located.
2. Right-click the computer object, and then click **Properties**.
3. In the **Properties** dialog box, click the **BitLocker Recovery** tab to view the BitLocker recovery passwords that are associated with the computer.

To copy the recovery passwords for a computer

1. Follow the steps in the previous procedure to view the BitLocker recovery passwords.
2. On the **BitLocker Recovery** tab of the **Properties** dialog box, right-click the BitLocker recovery password that you want to copy, and then click **Copy Details**.
3. Press CTRL+V to paste the copied text to a destination location, such as a text file or spreadsheet.

To locate a recovery password by using a password ID

1. In Active Directory Users and Computers, right-click the domain container, and then click **Find BitLocker Recovery Password**.
2. In the **Find BitLocker Recovery Password** dialog box, type the first eight characters of the recovery

password in the **Password ID (first 8 characters)** box, and then click **Search**. By completing the procedures in this scenario, you have viewed and copied the recovery passwords for a computer and used a password ID to locate a recovery password.

More information

- [BitLocker Overview](#)
- [BitLocker frequently asked questions \(FAQ\)](#)
- [Prepare your organization for BitLocker: Planning and policies](#)
- [BitLocker: How to deploy on Windows Server 2012](#)
- [BitLocker: Use BitLocker Drive Encryption Tools to manage BitLocker](#)

BitLocker group policy settings

7/1/2022 • 79 minutes to read • [Edit Online](#)

Applies to:

- Windows 10, Windows 11, Windows Server 2019, Windows Server 2016, Windows 8.1, and Windows Server 2012 R2

This article for IT professionals describes the function, location, and effect of each Group Policy setting that is used to manage BitLocker Drive Encryption.

To control the drive encryption tasks the user can perform from the Windows Control Panel or to modify other configuration options, you can use Group Policy administrative templates or local computer policy settings. How you configure these policy settings depends on how you implement BitLocker and what level of user interaction will be allowed.

NOTE

A separate set of Group Policy settings supports the use of the Trusted Platform Module (TPM). For details about those settings, see [Trusted Platform Module Group Policy settings](#).

BitLocker Group Policy settings can be accessed using the Local Group Policy Editor and the Group Policy Management Console (GPMC) under **Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption**. Most of the BitLocker Group Policy settings are applied when BitLocker is initially turned on for a drive. If a computer isn't compliant with existing Group Policy settings, BitLocker may not be turned on or modified until the computer is in a compliant state. When a drive is out of compliance with Group Policy settings (for example, if a Group Policy setting was changed after the initial BitLocker deployment in your organization, and then the setting was applied to previously encrypted drives), no change can be made to the BitLocker configuration of that drive except a change that will bring it into compliance.

If multiple changes are necessary to bring the drive into compliance, you must suspend BitLocker protection, make the necessary changes, and then resume protection. This situation could occur, for example, if a removable drive is initially configured to be unlocked with a password and then Group Policy settings are changed to disallow passwords and require smart cards. In this situation, you need to suspend BitLocker protection by using the [Manage-bde](#) command-line tool, delete the password unlock method, and add the smart card method. After this is complete, BitLocker is compliant with the Group Policy setting and BitLocker protection on the drive can be resumed.

BitLocker group policy settings

NOTE

For more details about Active Directory configuration related to BitLocker enablement, please see [Set up MDT for BitLocker](#).

The following sections provide a comprehensive list of BitLocker group policy settings that are organized by usage. BitLocker group policy settings include settings for specific drive types (operating system drives, fixed data drives, and removable data drives) and settings that are applied to all drives.

The following policy settings can be used to determine how a BitLocker-protected drive can be unlocked.

- Allow devices with Secure Boot and protected DMA ports to opt out of preboot PIN
- Allow network unlock at startup
- Require additional authentication at startup
- Allow enhanced PINs for startup
- Configure minimum PIN length for startup
- Disable new DMA devices when this computer is locked
- Disallow standard users from changing the PIN or password
- Configure use of passwords for operating system drives
- Require additional authentication at startup (Windows Server 2008 and Windows Vista)
- Configure use of smart cards on fixed data drives
- Configure use of passwords on fixed data drives
- Configure use of smart cards on removable data drives
- Configure use of passwords on removable data drives
- Validate smart card certificate usage rule compliance
- Enable use of BitLocker authentication requiring preboot keyboard input on slates

The following policy settings are used to control how users can access drives and how they can use BitLocker on their computers.

- Deny write access to fixed drives not protected by BitLocker
- Deny write access to removable drives not protected by BitLocker
- Control use of BitLocker on removable drives

The following policy settings determine the encryption methods and encryption types that are used with BitLocker.

- Choose drive encryption method and cipher strength
- Configure use of hardware-based encryption for fixed data drives
- Configure use of hardware-based encryption for operating system drives
- Configure use of hardware-based encryption for removable data drives
- Enforce drive encryption type on fixed data drives
- Enforce drive encryption type on operating system drives
- Enforce drive encryption type on removable data drives

The following policy settings define the recovery methods that can be used to restore access to a BitLocker-protected drive if an authentication method fails or is unable to be used.

- Choose how BitLocker-protected operating system drives can be recovered
- Choose how users can recover BitLocker-protected drives (Windows Server 2008 and Windows Vista)
- Store BitLocker recovery information in Active Directory Domain Services (Windows Server 2008 and Windows Vista)
- Choose default folder for recovery password
- Choose how BitLocker-protected fixed drives can be recovered
- Choose how BitLocker-protected removable drives can be recovered
- Configure the pre-boot recovery message and URL

The following policies are used to support customized deployment scenarios in your organization.

- Allow Secure Boot for integrity validation
- Provide the unique identifiers for your organization

- [Prevent memory overwrite on restart](#)
- [Configure TPM platform validation profile for BIOS-based firmware configurations](#)
- [Configure TPM platform validation profile \(Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2\)](#)
- [Configure TPM platform validation profile for native UEFI firmware configurations](#)
- [Reset platform validation data after BitLocker recovery](#)
- [Use enhanced Boot Configuration Data validation profile](#)
- [Allow access to BitLocker-protected fixed data drives from earlier versions of Windows](#)
- [Allow access to BitLocker-protected removable data drives from earlier versions of Windows](#)

Allow devices with secure boot and protected DMA ports to opt out of preboot PIN

Policy description	With this policy setting, you can allow TPM-only protection for newer, more secure devices, such as devices that support Modern Standby or HSTI, while requiring PIN on older devices.
Introduced	Windows 10, version 1703, or Windows 11
Drive type	Operating system drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Operating System Drives
Conflicts	This setting overrides the Require startup PIN with TPM option of the Require additional authentication at startup policy on compliant hardware.
When enabled	Users on Modern Standby and HSTI compliant devices will have the choice to turn on BitLocker without preboot authentication.
When disabled or not configured	The options of the Require additional authentication at startup policy apply.

Reference

The preboot authentication option **Require startup PIN with TPM** of the [Require additional authentication at startup](#) policy is often enabled to help ensure security for older devices that don't support Modern Standby. But visually impaired users have no audible way to know when to enter a PIN. This setting enables an exception to the PIN-required policy on secure hardware.

Allow network unlock at startup

This policy controls a portion of the behavior of the Network Unlock feature in BitLocker. This policy is required to enable BitLocker Network Unlock on a network because it allows clients running BitLocker to create the necessary network key protector during encryption.

This policy is used with the BitLocker Drive Encryption Network Unlock Certificate security policy (located in the **Public Key Policies** folder of Local Computer Policy) to allow systems that are connected to a trusted network to properly utilize the Network Unlock feature.

Policy description	With this policy setting, you can control whether a BitLocker-protected computer that is connected to a trusted local area network and joined to a domain can create and use network key protectors on TPM-enabled computers to automatically unlock the operating system drive when the computer is started.
Introduced	Windows Server 2012 and Windows 8
Drive type	Operating system drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Operating System Drives
Conflicts	None
When enabled	Clients configured with a BitLocker Network Unlock certificate can create and use Network Key Protectors.
When disabled or not configured	Clients can't create and use Network Key Protectors

Reference

To use a network key protector to unlock the computer, the computer and the server that hosts BitLocker Drive Encryption Network Unlock must be provisioned with a Network Unlock certificate. The Network Unlock certificate is used to create a network key protector and to protect the information exchange with the server to unlock the computer. You can use the Group Policy setting **Computer Configuration\Windows Settings\Security Settings\Public Key Policies\BitLocker Drive Encryption Network Unlock Certificate** on the domain controller to distribute this certificate to computers in your organization. This unlock method uses the TPM on the computer, so computers that don't have a TPM can't create network key protectors to automatically unlock by using Network Unlock.

NOTE

For reliability and security, computers should also have a TPM startup PIN that can be used when the computer is disconnected from the wired network or can't connect to the domain controller at startup.

For more information about Network Unlock feature, see [BitLocker: How to enable Network Unlock](#).

Require additional authentication at startup

This policy setting is used to control which unlock options are available for operating system drives.

Policy description	With this policy setting, you can configure whether BitLocker requires additional authentication each time the computer starts and whether you are using BitLocker with a Trusted Platform Module (TPM). This policy setting is applied when you turn on BitLocker.
Introduced	Windows Server 2008 R2 and Windows 7

Drive type	Operating system drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Operating System Drives
Conflicts	If one authentication method is required, the other methods can't be allowed. Use of BitLocker with a TPM startup key or with a TPM startup key and a PIN must be disallowed if the Deny write access to removable drives not protected by BitLocker policy setting is enabled.
When enabled	Users can configure advanced startup options in the BitLocker Setup Wizard.
When disabled or not configured	Users can configure only basic options on computers with a TPM. Only one of the additional authentication options can be required at startup; otherwise, a policy error occurs.

Reference

If you want to use BitLocker on a computer without a TPM, select **Allow BitLocker without a compatible TPM**. In this mode, a password or USB drive is required for startup. The USB drive stores the startup key that is used to encrypt the drive. When the USB drive is inserted, the startup key is authenticated and the operating system drive is accessible. If the USB drive is lost or unavailable, BitLocker recovery is required to access the drive.

On a computer with a compatible TPM, additional authentication methods can be used at startup to improve protection for encrypted data. When the computer starts, it can use:

- Only the TPM
- Insertion of a USB flash drive containing the startup key
- The entry of a 4-digit to 20-digit personal identification number (PIN)
- A combination of the PIN and the USB flash drive

There are four options for TPM-enabled computers or devices:

- Configure TPM startup
 - Allow TPM
 - Require TPM
 - Do not allow TPM
- Configure TPM startup PIN
 - Allow startup PIN with TPM
 - Require startup PIN with TPM
 - Do not allow startup PIN with TPM
- Configure TPM startup key
 - Allow startup key with TPM
 - Require startup key with TPM
 - Do not allow startup key with TPM

- Configure TPM startup key and PIN
 - Allow TPM startup key with PIN
 - Require startup key and PIN with TPM
 - Do not allow TPM startup key with PIN

Allow enhanced PINs for startup

This policy setting permits the use of enhanced PINs when you use an unlock method that includes a PIN.

Policy description	With this policy setting, you can configure whether enhanced startup PINs are used with BitLocker.
Introduced	Windows Server 2008 R2 and Windows 7
Drive type	Operating system drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Operating System Drives
Conflicts	None
When enabled	All new BitLocker startup PINs that are set will be enhanced PINs. Existing drives that were protected by using standard startup PINs aren't affected.
When disabled or not configured	Enhanced PINs will not be used.

Reference

Enhanced startup PINs permit the use of characters (including uppercase and lowercase letters, symbols, numbers, and spaces). This policy setting is applied when you turn on BitLocker.

IMPORTANT

Not all computers support enhanced PIN characters in the preboot environment. It's strongly recommended that users perform a system check during the BitLocker setup to verify that enhanced PIN characters can be used.

Configure minimum PIN length for startup

This policy setting is used to set a minimum PIN length when you use an unlock method that includes a PIN.

Policy description	With this policy setting, you can configure a minimum length for a TPM startup PIN. This policy setting is applied when you turn on BitLocker. The startup PIN must have a minimum length of four digits, and it can have a maximum length of 20 digits. By default, the minimum PIN length is 6.
Introduced	Windows Server 2008 R2 and Windows 7
Drive type	Operating system drives

Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Operating System Drives
Conflicts	None
When enabled	You can require that startup PINs set by users must have a minimum length you choose that is between 4 and 20 digits.
When disabled or not configured	Users can configure a startup PIN of any length between 6 and 20 digits.

Reference

This policy setting is applied when you turn on BitLocker. The startup PIN must have a minimum length of four digits and can have a maximum length of 20 digits.

Originally, BitLocker allowed a length from 4 to 20 characters for a PIN. Windows Hello has its own PIN for logon, length of which can be 4 to 127 characters. Both BitLocker and Windows Hello use the TPM to prevent PIN brute-force attacks.

The TPM can be configured to use Dictionary Attack Prevention parameters ([lockout threshold and lockout duration](#)) to control how many failed authorization attempts are allowed before the TPM is locked out, and how much time must elapse before another attempt can be made.

The Dictionary Attack Prevention Parameters provide a way to balance security needs with usability. For example, when BitLocker is used with a TPM + PIN configuration, the number of PIN guesses is limited over time. A TPM 2.0 in this example could be configured to allow only 32 PIN guesses immediately, and then only one more guess every two hours. This totals a maximum of about 4415 guesses per year. If the PIN is four digits, all 9999 possible PIN combinations could be attempted in a little over two years.

Increasing the PIN length requires a greater number of guesses for an attacker. In that case, the lockout duration between each guess can be shortened to allow legitimate users to retry a failed attempt sooner, while maintaining a similar level of protection.

Beginning with Windows 10, version 1703, or Windows 11, the minimum length for the BitLocker PIN was increased to six characters to better align with other Windows features that use TPM 2.0, including Windows Hello. To help organizations with the transition, beginning with Windows 10, version 1709 and Windows 10, version 1703 with the October 2017, or Windows 11 [cumulative update](#) installed, the BitLocker PIN length is six characters by default, but it can be reduced to four characters. If the minimum PIN length is reduced from the default of six characters, then the TPM 2.0 lockout period will be extended.

Disable new DMA devices when this computer is locked

This policy setting allows you to block direct memory access (DMA) for all hot pluggable PCI ports until a user signs in to Windows.

Policy description	This setting helps prevent attacks that use external PCI-based devices to access BitLocker keys.
Introduced	Windows 10, version 1703, or Windows 11

Drive type	Operating system drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption
Conflicts	None
When enabled	Every time the user locks the screen, DMA will be blocked on hot pluggable PCI ports until the user signs in again.
When disabled or not configured	DMA is available on hot pluggable PCI devices if the device is turned on, regardless of whether a user is signed in.

Reference

This policy setting is only enforced when BitLocker or device encryption is enabled. As explained in the [Microsoft Security Guidance blog](#), in some cases when this setting is enabled, internal, PCI-based peripherals can fail, including wireless network drivers and input and audio peripherals. This problem is fixed in the [April 2018 quality update](#).

Disallow standard users from changing the PIN or password

This policy setting allows you to configure whether standard users are allowed to change the PIN or password that is used to protect the operating system drive.

Policy description	With this policy setting, you can configure whether standard users are allowed to change the PIN or password used to protect the operating system drive.
Introduced	Windows Server 2012 and Windows 8
Drive type	Operating system drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Operating System Drives
Conflicts	None
When enabled	Standard users aren't allowed to change BitLocker PINs or passwords.
When disabled or not configured	Standard users are permitted to change BitLocker PINs or passwords.

Reference

To change the PIN or password, the user must be able to provide the current PIN or password. This policy setting is applied when you turn on BitLocker.

Configure use of passwords for operating system drives

This policy controls how non-TPM based systems utilize the password protector. Used with the **Password must meet complexity requirements** policy, this policy allows administrators to require password length and

complexity for using the password protector. By default, passwords must be eight characters in length. Complexity configuration options determine how important domain connectivity is for the client. For the strongest password security, administrators should choose **Require password complexity** because it requires domain connectivity, and it requires that the BitLocker password meets the same password complexity requirements as domain sign-in passwords.

Policy description	With this policy setting, you can specify the constraints for passwords that are used to unlock operating system drives that are protected with BitLocker.
Introduced	Windows Server 2012 and Windows 8
Drive type	Operating system drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Operating System Drives
Conflicts	<p>Passwords can't be used if FIPS-compliance is enabled.</p> <p>NOTE: The System cryptography: Use FIPS-compliant algorithms for encryption, hashing, and signing policy setting, which is located at Computer Configuration\Windows Settings\Security Settings\Local Policies\Security Options specifies whether FIPS-compliance is enabled.</p>
When enabled	Users can configure a password that meets the requirements you define. To enforce complexity requirements for the password, select Require complexity .
When disabled or not configured	The default length constraint of eight characters will apply to operating system drive passwords and no complexity checks will occur.

Reference

If non-TPM protectors are allowed on operating system drives, you can provision a password, enforce complexity requirements on the password, and configure a minimum length for the password. For the complexity requirement setting to be effective, the group policy setting **Password must meet complexity requirements**, which is located at **Computer Configuration\Windows Settings\Security Settings\Account Policies>Password Policy**, must be also enabled.

NOTE

These settings are enforced when turning on BitLocker, not when unlocking a volume. BitLocker allows unlocking a drive with any of the protectors that are available on the drive.

When set to **Require complexity**, a connection to a domain controller is necessary when BitLocker is enabled to validate the complexity the password. When set to **Allow complexity**, a connection to a domain controller is attempted to validate that the complexity adheres to the rules set by the policy. If no domain controllers are found, the password will be accepted regardless of actual password complexity, and the drive will be encrypted by using that password as a protector. When set to **Do not allow complexity**, there is no password complexity

validation. Passwords must be at least eight characters. To configure a greater minimum length for the password, enter the desired number of characters in the **Minimum password length** box.

When this policy setting is enabled, you can set the option **Configure password complexity for operating system drives** to:

- Allow password complexity
- Deny password complexity
- Require password complexity

Require additional authentication at startup (Windows Server 2008 and Windows Vista)

This policy setting is used to control what unlock options are available for computers running Windows Server 2008 or Windows Vista.

Policy description	With this policy setting, you can control whether the BitLocker Setup Wizard on computers running Windows Vista or Windows Server 2008 can set up an additional authentication method that is required each time the computer starts.
Introduced	Windows Server 2008 and Windows Vista
Drive type	Operating system drives (Windows Server 2008 and Windows Vista)
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Operating System Drives
Conflicts	If you choose to require an additional authentication method, other authentication methods can't be allowed.
When enabled	The BitLocker Setup Wizard displays the page that allows the user to configure advanced startup options for BitLocker. You can further configure setting options for computers with or without a TPM.
When disabled or not configured	The BitLocker Setup Wizard displays basic steps that allow users to enable BitLocker on computers with a TPM. In this basic wizard, no additional startup key or startup PIN can be configured.

Reference

On a computer with a compatible TPM, two authentication methods can be used at startup to provide added protection for encrypted data. When the computer starts, it can prompt users to insert a USB drive that contains a startup key. It can also prompt users to enter a startup PIN with a length between 6 and 20 digits.

A USB drive that contains a startup key is needed on computers without a compatible TPM. Without a TPM, BitLocker-encrypted data is protected solely by the key material that is on this USB drive.

There are two options for TPM-enabled computers or devices:

- Configure TPM startup PIN
 - Allow startup PIN with TPM

- Require startup PIN with TPM
- Do not allow startup PIN with TPM
- Configure TPM startup key
 - Allow startup key with TPM
 - Require startup key with TPM
 - Do not allow startup key with TPM

These options are mutually exclusive. If you require the startup key, you must not allow the startup PIN. If you require the startup PIN, you must not allow the startup key. Otherwise, a policy error will occur.

To hide the advanced page on a TPM-enabled computer or device, set these options to **Do not allow** for the startup key and for the startup PIN.

Configure use of smart cards on fixed data drives

This policy setting is used to require, allow, or deny the use of smart cards with fixed data drives.

Policy description	With this policy setting, you can specify whether smart cards can be used to authenticate user access to the BitLocker-protected fixed data drives on a computer.
Introduced	Windows Server 2008 R2 and Windows 7
Drive type	Fixed data drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Fixed Data Drives
Conflicts	To use smart cards with BitLocker, you may also need to modify the object identifier setting in the Computer Configuration\Administrative Templates\BitLocker Drive Encryption\Validate smart card certificate usage rule compliance policy setting to match the object identifier of your smart card certificates.
When enabled	Smart cards can be used to authenticate user access to the drive. You can require smart card authentication by selecting the Require use of smart cards on fixed data drives check box.
When disabled	Users can't use smart cards to authenticate their access to BitLocker-protected fixed data drives.
When not configured	Smart cards can be used to authenticate user access to a BitLocker-protected drive.

Reference

NOTE

These settings are enforced when turning on BitLocker, not when unlocking a drive. BitLocker allows unlocking a drive by using any of the protectors that are available on the drive.

Configure use of passwords on fixed data drives

This policy setting is used to require, allow, or deny the use of passwords with fixed data drives.

Policy description	With this policy setting, you can specify whether a password is required to unlock BitLocker-protected fixed data drives.
Introduced	Windows Server 2008 R2 and Windows 7
Drive type	Fixed data drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Fixed Data Drives
Conflicts	To use password complexity, the Computer Configuration\Windows Settings\Security Settings\Account Policies>Password Policy>Password must meet complexity requirements policy setting must also be enabled.
When enabled	Users can configure a password that meets the requirements you define. To require the use of a password, select Require password for fixed data drive . To enforce complexity requirements on the password, select Require complexity .
When disabled	The user isn't allowed to use a password.
When not configured	Passwords are supported with the default settings, which don't include password complexity requirements and require only eight characters.

Reference

When set to **Require complexity**, a connection to a domain controller is necessary to validate the complexity of the password when BitLocker is enabled.

When set to **Allow complexity**, a connection to a domain controller is attempted to validate that the complexity adheres to the rules set by the policy. However, if no domain controllers are found, the password is accepted regardless of the actual password complexity, and the drive is encrypted by using that password as a protector.

When set to **Do not allow complexity**, no password complexity validation is performed.

Passwords must be at least eight characters. To configure a greater minimum length for the password, enter the desired number of characters in the **Minimum password length** box.

NOTE

These settings are enforced when turning on BitLocker, not when unlocking a drive. BitLocker allows unlocking a drive with any of the protectors that are available on the drive.

For the complexity requirement setting to be effective, the Group Policy setting **Computer Configuration\Windows Settings\Security Settings\Account Policies>Password Policy>Password must meet complexity requirements** must also be enabled. This policy setting is configured on a per-computer basis. This means that it applies to local user accounts and domain user accounts. Because the password filter that's used to validate password complexity is located on the domain controllers, local user

accounts can't access the password filter because they're not authenticated for domain access. When this policy setting is enabled, if you sign in with a local user account, and you attempt to encrypt a drive or change a password on an existing BitLocker-protected drive, an "Access denied" error message is displayed. In this situation, the password key protector can't be added to the drive.

Enabling this policy setting requires that connectivity to a domain be established before adding a password key protector to a BitLocker-protected drive. Users who work remotely and have periods of time in which they can't connect to the domain should be made aware of this requirement so that they can schedule a time when they will be connected to the domain to turn on BitLocker or to change a password on a BitLocker-protected data drive.

IMPORTANT

Passwords can't be used if FIPS compliance is enabled. The **System cryptography: Use FIPS-compliant algorithms for encryption, hashing, and signing** policy setting in **Computer Configuration\Windows Settings\Security Settings\Local Policies\Security Options** specifies whether FIPS compliance is enabled.

Configure use of smart cards on removable data drives

This policy setting is used to require, allow, or deny the use of smart cards with removable data drives.

Policy description	With this policy setting, you can specify whether smart cards can be used to authenticate user access to BitLocker-protected removable data drives on a computer.
Introduced	Windows Server 2008 R2 and Windows 7
Drive type	Removable data drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Removable Data Drives
Conflicts	To use smart cards with BitLocker, you may also need to modify the object identifier setting in the Computer Configuration\Administrative Templates\BitLocker Drive Encryption\Validate smart card certificate usage rule compliance policy setting to match the object identifier of your smart card certificates.
When enabled	Smart cards can be used to authenticate user access to the drive. You can require smart card authentication by selecting the Require use of smart cards on removable data drives check box.
When disabled or not configured	Users aren't allowed to use smart cards to authenticate their access to BitLocker-protected removable data drives.
When not configured	Smart cards are available to authenticate user access to a BitLocker-protected removable data drive.

Reference

NOTE

These settings are enforced when turning on BitLocker, not when unlocking a drive. BitLocker allows unlocking a drive with any of the protectors that are available on the drive.

Configure use of passwords on removable data drives

This policy setting is used to require, allow, or deny the use of passwords with removable data drives.

Policy description	With this policy setting, you can specify whether a password is required to unlock BitLocker-protected removable data drives.
Introduced	Windows Server 2008 R2 and Windows 7
Drive type	Removable data drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Removable Data Drives
Conflicts	To use password complexity, the Password must meet complexity requirements policy setting, which is located at Computer Configuration\Windows Settings\Security Settings\Account Policies>Password Policy must also be enabled.
When enabled	Users can configure a password that meets the requirements you define. To require the use of a password, select Require password for removable data drive . To enforce complexity requirements on the password, select Require complexity .
When disabled	The user isn't allowed to use a password.
When not configured	Passwords are supported with the default settings, which don't include password complexity requirements and require only eight characters.

Reference

If you choose to allow the use of a password, you can require a password to be used, enforce complexity requirements, and configure a minimum length. For the complexity requirement setting to be effective, the group policy setting **Password must meet complexity requirements**, which is located at **Computer Configuration\Windows Settings\Security Settings\Account Policies>Password Policy**, must also be enabled.

NOTE

These settings are enforced when turning on BitLocker, not when unlocking a drive. BitLocker allows unlocking a drive with any of the protectors that are available on the drive.

Passwords must be at least eight characters. To configure a greater minimum length for the password, enter the wanted number of characters in the **Minimum password length** box.

When set to **Require complexity**, a connection to a domain controller is necessary when BitLocker is enabled to validate the complexity of the password.

When set to **Allow complexity**, a connection to a domain controller is attempted to validate that the complexity adheres to the rules set by the policy. However, if no domain controllers are found, the password is still accepted regardless of actual password complexity and the drive is encrypted by using that password as a protector.

When set to **Do not allow complexity**, no password complexity validation is done.

NOTE

Passwords can't be used if FIPS compliance is enabled. The **System cryptography: Use FIPS-compliant algorithms for encryption, hashing, and signing** policy setting in **Computer Configuration\Windows Settings\Security Settings\Local Policies\Security Options** specifies whether FIPS compliance is enabled.

For information about this setting, see [System cryptography: Use FIPS-compliant algorithms for encryption, hashing, and signing](#).

Validate smart card certificate usage rule compliance

This policy setting is used to determine what certificate to use with BitLocker.

Policy description	With this policy setting, you can associate an object identifier from a smart card certificate to a BitLocker-protected drive.
Introduced	Windows Server 2008 R2 and Windows 7
Drive type	Fixed and removable data drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption
Conflicts	None
When enabled	The object identifier that is specified in the Object identifier setting must match the object identifier in the smart card certificate.
When disabled or not configured	The default object identifier is used.

Reference

This policy setting is applied when you turn on BitLocker.

The object identifier is specified in the enhanced key usage (EKU) of a certificate. BitLocker can identify which certificates can be used to authenticate a user certificate to a BitLocker-protected drive by matching the object identifier in the certificate with the object identifier that is defined by this policy setting.

The default object identifier is 1.3.6.1.4.1.311.67.1.1.

NOTE

BitLocker doesn't require that a certificate have an EKU attribute; however, if one is configured for the certificate, it must be set to an object identifier that matches the object identifier configured for BitLocker.

Enable use of BitLocker authentication requiring preboot keyboard input on slates**Enable use of BitLocker authentication requiring pre-boot keyboard input on slates**

Policy description	With this policy setting, you can allow users to enable authentication options that require user input from the preboot environment, even if the platform indicates a lack of preboot input capability.
Introduced	Windows Server 2012 and Windows 8
Drive type	Operating system drive
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Operating System Drive
Conflicts	None
When enabled	Devices must have an alternative means of preboot input (such as an attached USB keyboard).
When disabled or not configured	The Windows Recovery Environment must be enabled on tablets to support entering the BitLocker recovery password.

Reference

The Windows touch keyboard (such as used by tablets) isn't available in the preboot environment where BitLocker requires additional information, such as a PIN or password.

It's recommended that administrators enable this policy only for devices that are verified to have an alternative means of preboot input, such as attaching a USB keyboard.

When the Windows Recovery Environment isn't enabled and this policy isn't enabled, you can't turn on BitLocker on a device that uses the Windows touch keyboard.

If you don't enable this policy setting, the following options in the **Require additional authentication at startup** policy might not be available:

- Configure TPM startup PIN: Required and Allowed
- Configure TPM startup key and PIN: Required and Allowed
- Configure use of passwords for operating system drives

Deny write access to fixed drives not protected by BitLocker

This policy setting is used to require encryption of fixed drives prior to granting Write access.

--	--

Policy description	With this policy setting, you can set whether BitLocker protection is required for fixed data drives to be writable on a computer.
Introduced	Windows Server 2008 R2 and Windows 7
Drive type	Fixed data drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Fixed Data Drives
Conflicts	See the Reference section for a description of conflicts.
When enabled	All fixed data drives that aren't BitLocker-protected are mounted as Read-only. If the drive is protected by BitLocker, it's mounted with Read and Write access.
When disabled or not configured	All fixed data drives on the computer are mounted with Read and Write access.

Reference

This policy setting is applied when you turn on BitLocker.

Conflict considerations include:

- When this policy setting is enabled, users receive "Access denied" error messages when they try to save data to unencrypted fixed data drives. See the Reference section for additional conflicts.
- If BdeHdCfg.exe is run on a computer when this policy setting is enabled, you could encounter the following issues:
 - If you attempted to shrink the drive and create the system drive, the drive size is successfully reduced and a raw partition is created. However, the raw partition isn't formatted. The following error message is displayed: "The new active drive cannot be formatted. You may need to manually prepare your drive for BitLocker."
 - If you attempt to use unallocated space to create the system drive, a raw partition will be created. However, the raw partition will not be formatted. The following error message is displayed: "The new active drive cannot be formatted. You may need to manually prepare your drive for BitLocker."
 - If you attempt to merge an existing drive into the system drive, the tool fails to copy the required boot file onto the target drive to create the system drive. The following error message is displayed: "BitLocker setup failed to copy boot files. You may need to manually prepare your drive for BitLocker."
- If this policy setting is enforced, a hard drive can't be repartitioned because the drive is protected. If you are upgrading computers in your organization from a previous version of Windows, and those computers were configured with a single partition, you should create the required BitLocker system partition before you apply this policy setting to the computers.

Deny write access to removable drives not protected by BitLocker

This policy setting is used to require that removable drives are encrypted prior to granting Write access, and to control whether BitLocker-protected removable drives that were configured in another organization can be opened with Write access.

Policy description	With this policy setting, you can configure whether BitLocker protection is required for a computer to be able to write data to a removable data drive.
Introduced	Windows Server 2008 R2 and Windows 7
Drive type	Removable data drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Removable Data Drives
Conflicts	See the Reference section for a description of conflicts.
When enabled	All removable data drives that aren't BitLocker-protected are mounted as Read-only. If the drive is protected by BitLocker, it's mounted with Read and Write access.
When disabled or not configured	All removable data drives on the computer are mounted with Read and Write access.

Reference

If the **Deny write access to devices configured in another organization** option is selected, only drives with identification fields that match the computer's identification fields are given Write access. When a removable data drive is accessed, it's checked for a valid identification field and allowed identification fields. These fields are defined by the **Provide the unique identifiers for your organization** policy setting.

NOTE

You can override this policy setting with the policy settings under **User Configuration\Administrative Templates\System\Removable Storage Access**. If the **Removable Disks: Deny write access** policy setting is enabled, this policy setting will be ignored.

Conflict considerations include:

1. Use of BitLocker with the TPM plus a startup key or with the TPM plus a PIN and startup key must be disallowed if the **Deny write access to removable drives not protected by BitLocker** policy setting is enabled.
2. Use of recovery keys must be disallowed if the **Deny write access to removable drives not protected by BitLocker** policy setting is enabled.
3. You must enable the **Provide the unique identifiers for your organization** policy setting if you want to deny Write access to drives that were configured in another organization.

Control use of BitLocker on removable drives

This policy setting is used to prevent users from turning BitLocker on or off on removable data drives.

Policy description	With this policy setting, you can control the use of BitLocker on removable data drives.

Introduced	Windows Server 2008 R2 and Windows 7
Drive type	Removable data drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Removable Data Drives
Conflicts	None
When enabled	You can select property settings that control how users can configure BitLocker.
When disabled	Users can't use BitLocker on removable data drives.
When not configured	Users can use BitLocker on removable data drives.

Reference

This policy setting is applied when you turn on BitLocker.

For information about suspending BitLocker protection, see [BitLocker Basic Deployment](#).

The options for choosing property settings that control how users can configure BitLocker are:

- **Allow users to apply BitLocker protection on removable data drives** Enables the user to run the BitLocker Setup Wizard on a removable data drive.
- **Allow users to suspend and decrypt BitLocker on removable data drives** Enables the user to remove BitLocker from the drive or to suspend the encryption while performing maintenance.

Choose drive encryption method and cipher strength

This policy setting is used to control the encryption method and cipher strength.

Policy description	With this policy setting, you can control the encryption method and strength for drives.
Introduced	Windows Server 2012 and Windows 8
Drive type	All drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption
Conflicts	None
When enabled	You can choose an encryption algorithm and key cipher strength for BitLocker to use to encrypt drives.
When disabled or not configured	Beginning with Windows 10, version 1511, or Windows 11, BitLocker uses the default encryption method of XTS-AES 128-bit or the encryption method that is specified by the setup script.

Reference

The values of this policy determine the strength of the cipher that BitLocker uses for encryption. Enterprises may want to control the encryption level for increased security (AES-256 is stronger than AES-128).

If you enable this setting, you can configure an encryption algorithm and key cipher strength for fixed data drives, operating system drives, and removable data drives individually. For fixed and operating system drives, we recommend that you use the XTS-AES algorithm. For removable drives, you should use AES-CBC 128-bit or AES-CBC 256-bit if the drive will be used in other devices that aren't running Windows 10, version 1511 or later, or Windows 11.

Changing the encryption method has no effect if the drive is already encrypted or if encryption is in progress. In these cases, this policy setting is ignored.

WARNING

This policy doesn't apply to encrypted drives. Encrypted drives utilize their own algorithm, which is set by the drive during partitioning.

When this policy setting is disabled or not configured, BitLocker will use the default encryption method of XTS-AES 128-bit or the encryption method that is specified in the setup script.

Configure use of hardware-based encryption for fixed data drives

This policy controls how BitLocker reacts to systems that are equipped with encrypted drives when they're used as fixed data volumes. Using hardware-based encryption can improve the performance of drive operations that involve frequent reading or writing of data to the drive.

Policy description	With this policy setting, you can manage BitLocker's use of hardware-based encryption on fixed data drives and to specify which encryption algorithms BitLocker can use with hardware-based encryption.
Introduced	Windows Server 2012 and Windows 8
Drive type	Fixed data drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Fixed Data Drives
Conflicts	None

When enabled	You can specify additional options that control whether BitLocker software-based encryption is used instead of hardware-based encryption on computers that don't support hardware-based encryption. You can also specify whether you want to restrict the encryption algorithms and cipher suites that are used with hardware-based encryption.
When disabled	BitLocker can't use hardware-based encryption with fixed data drives, and BitLocker software-based encryption is used by default when the drive is encrypted.
When not configured	BitLocker software-based encryption is used irrespective of hardware-based encryption ability.

Reference

NOTE

The **Choose drive encryption method and cipher strength** policy setting doesn't apply to hardware-based encryption.

The encryption algorithm that is used by hardware-based encryption is set when the drive is partitioned. By default, BitLocker uses the algorithm that is configured on the drive to encrypt the drive. The **Restrict encryption algorithms and cipher suites allowed for hardware-based encryption** option of this setting enables you to restrict the encryption algorithms that BitLocker can use with hardware encryption. If the algorithm that is set for the drive isn't available, BitLocker disables the use of hardware-based encryption. Encryption algorithms are specified by object identifiers (OID), for example:

- Advanced Encryption Standard (AES) 128 in Cipher Block Chaining (CBC) mode OID: 2.16.840.1.101.3.4.1.2
- AES 256 in CBC mode OID: 2.16.840.1.101.3.4.1.42

Configure use of hardware-based encryption for operating system drives

This policy controls how BitLocker reacts when encrypted drives are used as operating system drives. Using hardware-based encryption can improve the performance of drive operations that involve frequent reading or writing of data to the drive.

Policy description	With this policy setting, you can manage BitLocker's use of hardware-based encryption on operating system drives and specify which encryption algorithms it can use with hardware-based encryption.
Introduced	Windows Server 2012 and Windows 8
Drive type	Operating system drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Operating System Drives
Conflicts	None

When enabled	You can specify additional options that control whether BitLocker software-based encryption is used instead of hardware-based encryption on computers that don't support hardware-based encryption. You can also specify whether you want to restrict the encryption algorithms and cipher suites that are used with hardware-based encryption.
When disabled	BitLocker can't use hardware-based encryption with operating system drives, and BitLocker software-based encryption is used by default when the drive is encrypted.
When not configured	BitLocker software-based encryption is used irrespective of hardware-based encryption ability.

Reference

If hardware-based encryption isn't available, BitLocker software-based encryption is used instead.

NOTE

The **Choose drive encryption method and cipher strength** policy setting doesn't apply to hardware-based encryption.

The encryption algorithm that is used by hardware-based encryption is set when the drive is partitioned. By default, BitLocker uses the algorithm that is configured on the drive to encrypt the drive. The **Restrict encryption algorithms and cipher suites allowed for hardware-based encryption** option of this setting enables you to restrict the encryption algorithms that BitLocker can use with hardware encryption. If the algorithm that is set for the drive isn't available, BitLocker disables the use of hardware-based encryption. Encryption algorithms are specified by object identifiers (OID), for example:

- Advanced Encryption Standard (AES) 128 in Cipher Block Chaining (CBC) mode OID: 2.16.840.1.101.3.4.1.2
- AES 256 in CBC mode OID: 2.16.840.1.101.3.4.1.42

Configure use of hardware-based encryption for removable data drives

This policy controls how BitLocker reacts to encrypted drives when they're used as removable data drives. Using hardware-based encryption can improve the performance of drive operations that involve frequent reading or writing of data to the drive.

Policy description	With this policy setting, you can manage BitLocker's use of hardware-based encryption on removable data drives and specify which encryption algorithms it can use with hardware-based encryption.
Introduced	Windows Server 2012 and Windows 8
Drive type	Removable data drive
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Removable Data Drives

Conflicts	None
When enabled	You can specify additional options that control whether BitLocker software-based encryption is used instead of hardware-based encryption on computers that don't support hardware-based encryption. You can also specify whether you want to restrict the encryption algorithms and cipher suites that are used with hardware-based encryption.
When disabled	BitLocker can't use hardware-based encryption with removable data drives, and BitLocker software-based encryption is used by default when the drive is encrypted.
When not configured	BitLocker software-based encryption is used irrespective of hardware-based encryption ability.

Reference

If hardware-based encryption isn't available, BitLocker software-based encryption is used instead.

NOTE

The **Choose drive encryption method and cipher strength** policy setting doesn't apply to hardware-based encryption.

The encryption algorithm that is used by hardware-based encryption is set when the drive is partitioned. By default, BitLocker uses the algorithm that is configured on the drive to encrypt the drive. The **Restrict encryption algorithms and cipher suites allowed for hardware-based encryption** option of this setting enables you to restrict the encryption algorithms that BitLocker can use with hardware encryption. If the algorithm that is set for the drive isn't available, BitLocker disables the use of hardware-based encryption. Encryption algorithms are specified by object identifiers (OID), for example:

- Advanced Encryption Standard (AES) 128 in Cipher Block Chaining (CBC) mode OID: 2.16.840.1.101.3.4.1.2
- AES 256 in CBC mode OID: 2.16.840.1.101.3.4.1.42

Enforce drive encryption type on fixed data drives

This policy controls whether fixed data drives utilize Used Space Only encryption or Full encryption. Setting this policy also causes the BitLocker Setup Wizard to skip the encryption options page so no encryption selection displays to the user.

Policy description	With this policy setting, you can configure the encryption type that is used by BitLocker.
Introduced	Windows Server 2012 and Windows 8
Drive type	Fixed data drive
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Fixed Data Drives
Conflicts	None

When enabled	This policy defines the encryption type that BitLocker uses to encrypt drives, and the encryption type option isn't presented in the BitLocker Setup Wizard.
When disabled or not configured	The BitLocker Setup Wizard asks the user to select the encryption type before turning on BitLocker.

Reference

This policy setting is applied when you turn on BitLocker. Changing the encryption type has no effect if the drive is already encrypted or if encryption is in progress. Choose Full encryption to make it mandatory for the entire drive to be encrypted when BitLocker is turned on. Choose Used Space Only encryption to make it mandatory to encrypt only that portion of the drive that is used to store data when BitLocker is turned on.

NOTE

This policy is ignored when you are shrinking or expanding a volume and the BitLocker driver uses the current encryption method. For example, when a drive that is using Used Space Only encryption is expanded, the new free space isn't wiped as it would be for a drive that is using Full encryption. The user could wipe the free space on a Used Space Only drive by using the following command: `manage-bde -w`. If the volume is shrunk, no action is taken for the new free space.

For more information about the tool to manage BitLocker, see [Manage-bde](#).

Enforce drive encryption type on operating system drives

This policy controls whether operating system drives utilize Full encryption or Used Space Only encryption. Setting this policy also causes the BitLocker Setup Wizard to skip the encryption options page, so no encryption selection displays to the user.

Policy description	With this policy setting, you can configure the encryption type that is used by BitLocker.
Introduced	Windows Server 2012 and Windows 8
Drive type	Operating system drive
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Operating System Drives
Conflicts	None
When enabled	The encryption type that BitLocker uses to encrypt drives is defined by this policy, and the encryption type option isn't presented in the BitLocker Setup Wizard.
When disabled or not configured	The BitLocker Setup Wizard asks the user to select the encryption type before turning on BitLocker.

Reference

This policy setting is applied when you turn on BitLocker. Changing the encryption type has no effect if the drive

is already encrypted or if encryption is in progress. Choose Full encryption to make it mandatory for the entire drive to be encrypted when BitLocker is turned on. Choose Used Space Only encryption to make it mandatory to encrypt only that portion of the drive that is used to store data when BitLocker is turned on.

NOTE

This policy is ignored when shrinking or expanding a volume, and the BitLocker driver uses the current encryption method. For example, when a drive that is using Used Space Only encryption is expanded, the new free space isn't wiped as it would be for a drive that uses Full encryption. The user could wipe the free space on a Used Space Only drive by using the following command: `manage-bde -w`. If the volume is shrunk, no action is taken for the new free space.

For more information about the tool to manage BitLocker, see [Manage-bde](#).

Enforce drive encryption type on removable data drives

This policy controls whether fixed data drives utilize Full encryption or Used Space Only encryption. Setting this policy also causes the BitLocker Setup Wizard to skip the encryption options page, so no encryption selection displays to the user.

Policy description	With this policy setting, you can configure the encryption type that is used by BitLocker.
Introduced	Windows Server 2012 and Windows 8
Drive type	Removable data drive
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Removable Data Drives
Conflicts	None
When enabled	The encryption type that BitLocker uses to encrypt drives is defined by this policy, and the encryption type option isn't presented in the BitLocker Setup Wizard.
When disabled or not configured	The BitLocker Setup Wizard asks the user to select the encryption type before turning on BitLocker.

Reference

This policy setting is applied when you turn on BitLocker. Changing the encryption type has no effect if the drive is already encrypted or if encryption is in progress. Choose Full encryption to make it mandatory for the entire drive to be encrypted when BitLocker is turned on. Choose Used Space Only encryption to make it mandatory to encrypt only that portion of the drive that is used to store data when BitLocker is turned on.

NOTE

This policy is ignored when shrinking or expanding a volume, and the BitLocker driver uses the current encryption method. For example, when a drive that is using Used Space Only encryption is expanded, the new free space isn't wiped as it would be for a drive that is using Full Encryption. The user could wipe the free space on a Used Space Only drive by using the following command: `manage-bde -w`. If the volume is shrunk, no action is taken for the new free space.

For more information about the tool to manage BitLocker, see [Manage-bde](#).

Choose how BitLocker-protected operating system drives can be recovered

This policy setting is used to configure recovery methods for operating system drives.

Policy description	With this policy setting, you can control how BitLocker-protected operating system drives are recovered in the absence of the required startup key information.
Introduced	Windows Server 2008 R2 and Windows 7
Drive type	Operating system drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Operating System Drives
Conflicts	<p>You must disallow the use of recovery keys if the Deny write access to removable drives not protected by BitLocker policy setting is enabled.</p> <p>When using data recovery agents, you must enable the Provide the unique identifiers for your organization policy setting.</p>
When enabled	You can control the methods that are available to users to recover data from BitLocker-protected operating system drives.
When disabled or not configured	The default recovery options are supported for BitLocker recovery. By default, a data recovery agent is allowed, the recovery options can be specified by the user (including the recovery password and recovery key), and recovery information isn't backed up to AD DS.

Reference

This policy setting is applied when you turn on BitLocker.

The **Allow data recovery agent** check box is used to specify whether a data recovery agent can be used with BitLocker-protected operating system drives. Before a data recovery agent can be used, it must be added from **Public Key Policies**, which is located in the Group Policy Management Console (GPMC) or in the Local Group Policy Editor.

For more information about adding data recovery agents, see [BitLocker basic deployment](#).

In **Configure user storage of BitLocker recovery information**, select whether users are allowed, required, or not allowed to generate a 48-digit recovery password.

Select **Omit recovery options from the BitLocker setup wizard** to prevent users from specifying recovery options when they enable BitLocker on a drive. This means that you can't specify which recovery option to use when you enable BitLocker. Instead, BitLocker recovery options for the drive are determined by the policy setting.

In **Save BitLocker recovery information to Active Directory Domain Services**, choose which BitLocker recovery information to store in Active Directory Domain Services (AD DS) for operating system drives. If you

select **Store recovery password and key packages**, the BitLocker recovery password and the key package are stored in AD DS. Storing the key package supports the recovery of data from a drive that is physically corrupted. If you select **Store recovery password only**, only the recovery password is stored in AD DS.

Select the **Do not enable BitLocker until recovery information is stored in AD DS for operating system drives** check box if you want to prevent users from enabling BitLocker unless the computer is connected to the domain and the backup of BitLocker recovery information to AD DS succeeds.

NOTE

If the **Do not enable BitLocker until recovery information is stored in AD DS for operating system drives** check box is selected, a recovery password is automatically generated.

Choose how users can recover BitLocker-protected drives (Windows Server 2008 and Windows Vista)

This policy setting is used to configure recovery methods for BitLocker-protected drives on computers running Windows Server 2008 or Windows Vista.

Policy description	With this policy setting, you can control whether the BitLocker Setup Wizard can display and specify BitLocker recovery options.
Introduced	Windows Server 2008 and Windows Vista
Drive type	Operating system drives and fixed data drives on computers running Windows Server 2008 and Windows Vista
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption
Conflicts	This policy setting provides an administrative method of recovering data that is encrypted by BitLocker to prevent data loss due to lack of key information. If you choose the Do not allow option for both user recovery options, you must enable the Store BitLocker recovery information in Active Directory Domain Services (Windows Server 2008 and Windows Vista) policy setting to prevent a policy error.
When enabled	You can configure the options that the BitLocker Setup Wizard displays to users for recovering BitLocker encrypted data.
When disabled or not configured	The BitLocker Setup Wizard presents users with ways to store recovery options.

Reference

This policy is only applicable to computers running Windows Server 2008 or Windows Vista. This policy setting is applied when you turn on BitLocker.

Two recovery options can be used to unlock BitLocker-encrypted data in the absence of the required startup key information. Users can type a 48-digit numerical recovery password, or they can insert a USB drive that contains a 256-bit recovery key.

Saving the recovery password to a USB drive stores the 48-digit recovery password as a text file and the 256-bit

recovery key as a hidden file. Saving the recovery password to a folder stores the 48-digit recovery password as a text file. Printing the recovery password sends the 48-digit recovery password to the default printer. For example, not allowing the 48-digit recovery password prevents users from printing or saving recovery information to a folder.

IMPORTANT

If TPM initialization is performed during the BitLocker setup, TPM owner information is saved or printed with the BitLocker recovery information. The 48-digit recovery password isn't available in FIPS-compliance mode.

IMPORTANT

To prevent data loss, you must have a way to recover BitLocker encryption keys. If you don't allow both recovery options, you must enable the backup of BitLocker recovery information to AD DS. Otherwise, a policy error occurs.

Store BitLocker recovery information in Active Directory Domain Services (Windows Server 2008 and Windows Vista)

This policy setting is used to configure the storage of BitLocker recovery information in AD DS. This provides an administrative method of recovering data that is encrypted by BitLocker to prevent data loss due to lack of key information.

Policy description	With this policy setting, you can manage the AD DS backup of BitLocker Drive Encryption recovery information.
Introduced	Windows Server 2008 and Windows Vista
Drive type	Operating system drives and fixed data drives on computers running Windows Server 2008 and Windows Vista.
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption
Conflicts	None
When enabled	BitLocker recovery information is automatically and silently backed up to AD DS when BitLocker is turned on for a computer.
When disabled or not configured	BitLocker recovery information isn't backed up to AD DS.

Reference

This policy is only applicable to computers running Windows Server 2008 or Windows Vista.

This policy setting is applied when you turn on BitLocker.

BitLocker recovery information includes the recovery password and unique identifier data. You can also include a package that contains an encryption key for a BitLocker-protected drive. This key package is secured by one or more recovery passwords, and it can help perform specialized recovery when the disk is damaged or corrupted.

If you select **Require BitLocker backup to AD DS**, BitLocker can't be turned on unless the computer is connected to the domain and the backup of BitLocker recovery information to AD DS succeeds. This option is selected by default to help ensure that BitLocker recovery is possible.

A recovery password is a 48-digit number that unlocks access to a BitLocker-protected drive. A key package contains a drive's BitLocker encryption key, which is secured by one or more recovery passwords. Key packages may help perform specialized recovery when the disk is damaged or corrupted.

If the **Require BitLocker backup to AD DS** option isn't selected, AD DS backup is attempted, but network or other backup failures don't prevent the BitLocker setup. The Backup process isn't automatically retried, and the recovery password might not be stored in AD DS during BitLocker setup. TPM initialization might be needed during the BitLocker setup. Enable the **Turn on TPM backup to Active Directory Domain Services** policy setting in **Computer Configuration\Administrative Templates\System\Trusted Platform Module Services** to ensure that TPM information is also backed up.

For more information about this setting, see [TPM Group Policy settings](#).

Choose default folder for recovery password

This policy setting is used to configure the default folder for recovery passwords.

Policy description	With this policy setting, you can specify the default path that is displayed when the BitLocker Setup Wizard prompts the user to enter the location of a folder in which to save the recovery password.
Introduced	Windows Vista
Drive type	All drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption
Conflicts	None
When enabled	You can specify the path that will be used as the default folder location when the user chooses the option to save the recovery password in a folder. You can specify a fully qualified path or include the target computer's environment variables in the path. If the path isn't valid, the BitLocker Setup Wizard displays the computer's top-level folder view.
When disabled or not configured	The BitLocker Setup Wizard displays the computer's top-level folder view when the user chooses the option to save the recovery password in a folder.

Reference

This policy setting is applied when you turn on BitLocker.

NOTE

This policy setting doesn't prevent the user from saving the recovery password in another folder.

Choose how BitLocker-protected fixed drives can be recovered

This policy setting is used to configure recovery methods for fixed data drives.

Policy description	With this policy setting, you can control how BitLocker-protected fixed data drives are recovered in the absence of the required credentials.
Introduced	Windows Server 2008 R2 and Windows 7
Drive type	Fixed data drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Fixed Data Drives
Conflicts	<p>You must disallow the use of recovery keys if the Deny write access to removable drives not protected by BitLocker policy setting is enabled.</p> <p>When using data recovery agents, you must enable and configure the Provide the unique identifiers for your organization policy setting.</p>
When enabled	You can control the methods that are available to users to recover data from BitLocker-protected fixed data drives.
When disabled or not configured	The default recovery options are supported for BitLocker recovery. By default, a data recovery agent is allowed, the recovery options can be specified by the user (including the recovery password and recovery key), and recovery information isn't backed up to AD DS.

Reference

This policy setting is applied when you turn on BitLocker.

The **Allow data recovery agent** check box is used to specify whether a data recovery agent can be used with BitLocker-protected fixed data drives. Before a data recovery agent can be used, it must be added from **Public Key Policies**, which is located in the Group Policy Management Console (GPMC) or in the Local Group Policy Editor.

In **Configure user storage of BitLocker recovery information**, select whether users can be allowed, required, or not allowed to generate a 48-digit recovery password or a 256-bit recovery key.

Select **Omit recovery options from the BitLocker setup wizard** to prevent users from specifying recovery options when they enable BitLocker on a drive. This means that you can't specify which recovery option to use when you enable BitLocker. Instead, BitLocker recovery options for the drive are determined by the policy setting.

In **Save BitLocker recovery information to Active Directory Domain Services**, choose which BitLocker recovery information to store in AD DS for fixed data drives. If you select **Backup recovery password and key package**, the BitLocker recovery password and the key package are stored in AD DS. Storing the key package supports recovering data from a drive that has been physically corrupted. To recover this data, you can use the `Repair-bde` command-line tool. If you select **Backup recovery password only**, only the recovery password is stored in AD DS.

For more information about the BitLocker repair tool, see [Repair-bde](#).

Select the **Do not enable BitLocker until recovery information is stored in AD DS for fixed data drives** check box if you want to prevent users from enabling BitLocker unless the computer is connected to the

domain and the backup of BitLocker recovery information to AD DS succeeds.

NOTE

If the **Do not enable BitLocker until recovery information is stored in AD DS for fixed data drives** check box is selected, a recovery password is automatically generated.

Choose how BitLocker-protected removable drives can be recovered

This policy setting is used to configure recovery methods for removable data drives.

Policy description	With this policy setting, you can control how BitLocker-protected removable data drives are recovered in the absence of the required credentials.
Introduced	Windows Server 2008 R2 and Windows 7
Drive type	Removable data drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Removable Data Drives
Conflicts	<p>You must disallow the use of recovery keys if the Deny write access to removable drives not protected by BitLocker policy setting is enabled.</p> <p>When using data recovery agents, you must enable and configure the Provide the unique identifiers for your organization policy setting.</p>
When enabled	You can control the methods that are available to users to recover data from BitLocker-protected removable data drives.
When disabled or not configured	The default recovery options are supported for BitLocker recovery. By default, a data recovery agent is allowed, the recovery options can be specified by the user (including the recovery password and recovery key), and recovery information isn't backed up to AD DS.

Reference

This policy setting is applied when you turn on BitLocker.

The **Allow data recovery agent** check box is used to specify whether a data recovery agent can be used with BitLocker-protected removable data drives. Before a data recovery agent can be used, it must be added from **Public Key Policies**, which is accessed using the GPMC or the Local Group Policy Editor.

In **Configure user storage of BitLocker recovery information**, select whether users can be allowed, required, or not allowed to generate a 48-digit recovery password.

Select **Omit recovery options from the BitLocker setup wizard** to prevent users from specifying recovery options when they enable BitLocker on a drive. This means that you can't specify which recovery option to use when you enable BitLocker. Instead, BitLocker recovery options for the drive are determined by the policy setting.

In **Save BitLocker recovery information to Active Directory Domain Services**, choose which BitLocker recovery information is to be stored in AD DS for removable data drives. If you select **Backup recovery password and key package**, the BitLocker recovery password and the key package are stored in AD DS. If you select **Backup recovery password only**, only the recovery password is stored in AD DS.

Select the **Do not enable BitLocker until recovery information is stored in AD DS for removable data drives** check box if you want to prevent users from enabling BitLocker unless the computer is connected to the domain and the backup of BitLocker recovery information to AD DS succeeds.

NOTE

If the **Do not enable BitLocker until recovery information is stored in AD DS for fixed data drives** check box is selected, a recovery password is automatically generated.

Configure the pre-boot recovery message and URL

This policy setting is used to configure the entire recovery message and to replace the existing URL that is displayed on the pre-boot recovery screen when the operating system drive is locked.

Policy description	With this policy setting, you can configure the BitLocker recovery screen to display a customized message and URL.
Introduced	Windows
Drive type	Operating system drives
Policy path	Computer Configuration \ Administrative Templates \ Windows Components \ BitLocker Drive Encryption \ Operating System Drives \ Configure pre-boot recovery message and URL
Conflicts	None
When enabled	The customized message and URL are displayed on the pre-boot recovery screen. If you have previously enabled a custom recovery message and URL and want to revert to the default message and URL, you must keep the policy setting enabled and select the Use default recovery message and URL option.
When disabled or not configured	If the setting hasn't been previously enabled, then the default pre-boot recovery screen is displayed for BitLocker recovery. If the setting previously was enabled and is later disabled, then the last message in Boot Configuration Data (BCD) is displayed whether it was the default recovery message or the custom message.

Reference

Enabling the **Configure the pre-boot recovery message and URL** policy setting allows you to customize the default recovery screen message and URL to assist customers in recovering their key.

Once you enable the setting, you have three options:

- If you select the **Use default recovery message and URL** option, the default BitLocker recovery message and URL will be displayed on the pre-boot recovery screen.

- If you select the **Use custom recovery message** option, type the custom message in the **Custom recovery message option** text box. The message that you type in the **Custom recovery message option** text box is displayed on the pre-boot recovery screen. If a recovery URL is available, include it in the message.
- If you select the **Use custom recovery URL** option, type the custom message URL in the **Custom recovery URL option** text box. The URL that you type in the **Custom recovery URL option** text box replaces the default URL in the default recovery message, which is displayed on the pre-boot recovery screen.

IMPORTANT

Not all characters and languages are supported in the pre-boot environment. We strongly recommended that you verify the correct appearance of the characters that you use for the custom message and URL on the pre-boot recovery screen.

IMPORTANT

Because you can alter the BCDEdit commands manually before you have set Group Policy settings, you can't return the policy setting to the default setting by selecting the **Not Configured** option after you have configured this policy setting. To return to the default pre-boot recovery screen leave the policy setting enabled and select the **Use default message** options from the **Choose an option for the pre-boot recovery message** drop-down list box.

Allow Secure Boot for integrity validation

This policy controls how BitLocker-enabled system volumes are handled with the Secure Boot feature. Enabling this feature forces Secure Boot validation during the boot process and verifies Boot Configuration Data (BCD) settings according to the Secure Boot policy.

Policy description	With this policy setting, you can configure whether Secure Boot will be allowed as the platform integrity provider for BitLocker operating system drives.
Introduced	Windows Server 2012 and Windows 8
Drive type	All drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Operating System Drives
Conflicts	If you enable Allow Secure Boot for integrity validation , make sure the Configure TPM platform validation profile for native UEFI firmware configurations Group Policy setting isn't enabled or include PCR 7 to allow BitLocker to use Secure Boot for platform or BCD integrity validation. For more information about PCR 7, see Platform Configuration Register (PCR) in this article.
When enabled or not configured	BitLocker uses Secure Boot for platform integrity if the platform is capable of Secure Boot-based integrity validation.

When disabled	BitLocker uses legacy platform integrity validation, even on systems that are capable of Secure Boot-based integrity validation.

Reference

Secure boot ensures that the computer's pre-boot environment loads only firmware that is digitally signed by authorized software publishers. Secure boot also started providing more flexibility for managing pre-boot configurations than BitLocker integrity checks prior to Windows Server 2012 and Windows 8. When this policy is enabled and the hardware is capable of using secure boot for BitLocker scenarios, the **Use enhanced Boot Configuration Data validation profile** group policy setting is ignored, and secure boot verifies BCD settings according to the secure boot policy setting, which is configured separately from BitLocker.

WARNING

Disabling this policy might result in BitLocker recovery when manufacturer-specific firmware is updated. If you disable this policy, suspend BitLocker prior to applying firmware updates.

Provide the unique identifiers for your organization

This policy setting is used to establish an identifier that is applied to all drives that are encrypted in your organization.

Policy description	With this policy setting, you can associate unique organizational identifiers to a new drive that is enabled with BitLocker.
Introduced	Windows Server 2008 R2 and Windows 7
Drive type	All drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption
Conflicts	Identification fields are required to manage certificate-based data recovery agents on BitLocker-protected drives. BitLocker manages and updates certificate-based data recovery agents only when the identification field is present on a drive and it's identical to the value that is configured on the computer.
When enabled	You can configure the identification field on the BitLocker-protected drive and any allowed identification field that is used by your organization.
When disabled or not configured	The identification field isn't required.

Reference

These identifiers are stored as the identification field and the allowed identification field. The identification field allows you to associate a unique organizational identifier to BitLocker-protected drives. This identifier is automatically added to new BitLocker-protected drives, and it can be updated on existing BitLocker-protected

drives by using the [Manage-bde](#) command-line tool.

An identification field is required to manage certificate-based data recovery agents on BitLocker-protected drives and for potential updates to the BitLocker To Go Reader. BitLocker manages and updates data recovery agents only when the identification field on the drive matches the value that is configured in the identification field. In a similar manner, BitLocker updates the BitLocker To Go Reader only when the identification field's value on the drive matches the value that is configured for the identification field.

For more information about the tool to manage BitLocker, see [Manage-bde](#).

The allowed identification field is used in combination with the **Deny write access to removable drives not protected by BitLocker** policy setting to help control the use of removable drives in your organization. It's a comma-separated list of identification fields from your organization or external organizations.

You can configure the identification fields on existing drives by using the [Manage-bde](#) command-line tool.

When a BitLocker-protected drive is mounted on another BitLocker-enabled computer, the identification field and the allowed identification field are used to determine whether the drive is from an external organization.

Multiple values separated by commas can be entered in the identification and allowed identification fields. The identification field can be any value upto 260 characters.

Prevent memory overwrite on restart

This policy setting is used to control whether the computer's memory will be overwritten the next time the computer is restarted.

Policy description	With this policy setting, you can control computer restart performance at the risk of exposing BitLocker secrets.
Introduced	Windows Vista
Drive type	All drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption
Conflicts	None
When enabled	The computer will not overwrite memory when it restarts. Preventing memory overwrite may improve restart performance, but it increases the risk of exposing BitLocker secrets.
When disabled or not configured	BitLocker secrets are removed from memory when the computer restarts.

Reference

This policy setting is applied when you turn on BitLocker. BitLocker secrets include key material that is used to encrypt data. This policy setting applies only when BitLocker protection is enabled.

Configure TPM platform validation profile for BIOS-based firmware configurations

This policy setting determines what values the TPM measures when it validates early boot components before it unlocks an operating system drive on a computer with a BIOS configuration or with UEFI firmware that has the Compatibility Support Module (CSM) enabled.

Policy description	With this policy setting, you can configure how the computer's TPM security hardware secures the BitLocker encryption key.
Introduced	Windows Server 2012 and Windows 8
Drive type	Operating system drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Operating System Drives
Conflicts	None
When enabled	You can configure the boot components that the TPM validates before unlocking access to the BitLocker-encrypted operating system drive. If any of these components change while BitLocker protection is in effect, then the TPM doesn't release the encryption key to unlock the drive. Instead, the computer displays the BitLocker Recovery console and requires that the recovery password or the recovery key is provided to unlock the drive.
When disabled or not configured	The TPM uses the default platform validation profile or the platform validation profile that is specified by the setup script.

Reference

This policy setting doesn't apply if the computer doesn't have a compatible TPM or if BitLocker has already been turned on with TPM protection.

IMPORTANT

This Group Policy setting only applies to computers with BIOS configurations or to computers with UEFI firmware with the CSM enabled. Computers that use a native UEFI firmware configuration store different values in the Platform Configuration Registers (PCRs). Use the **Configure TPM platform validation profile for native UEFI firmware configurations** Group Policy setting to configure the TPM PCR profile for computers that use native UEFI firmware.

A platform validation profile consists of a set of PCR indices that range from 0 to 23. The default platform validation profile secures the encryption key against changes to the following:

- Core Root of Trust of Measurement (CRTM), BIOS, and Platform Extensions (PCR 0)
- Option ROM Code (PCR 2)
- Master Boot Record (MBR) Code (PCR 4)
- NTFS Boot Sector (PCR 8)
- NTFS Boot Block (PCR 9)
- Boot Manager (PCR 10)
- BitLocker Access Control (PCR 11)

NOTE

Changing from the default platform validation profile affects the security and manageability of your computer. BitLocker's sensitivity to platform modifications (malicious or authorized) is increased or decreased depending on inclusion or exclusion (respectively) of the PCRs.

The following list identifies all of the available PCRs:

- PCR 0: Core root-of-trust for measurement, BIOS, and platform extensions
- PCR 1: Platform and motherboard configuration and data.
- PCR 2: Option ROM code
- PCR 3: Option ROM data and configuration
- PCR 4: Master Boot Record (MBR) code
- PCR 5: Master Boot Record (MBR) partition table
- PCR 6: State transition and wake events
- PCR 7: Computer manufacturer-specific
- PCR 8: NTFS boot sector
- PCR 9: NTFS boot block
- PCR 10: Boot manager
- PCR 11: BitLocker access control
- PCR 12-23: Reserved for future use

Configure TPM platform validation profile (Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2)

This policy setting determines what values the TPM measures when it validates early boot components before unlocking a drive on a computer running Windows Vista, Windows Server 2008, or Windows 7.

Policy description	With this policy setting, you can configure how the computer's TPM security hardware secures the BitLocker encryption key.
Introduced	Windows Server 2008 and Windows Vista
Drive type	Operating system drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Operating System Drives
Conflicts	None
When enabled	You can configure the boot components that the TPM validates before unlocking access to the BitLocker-encrypted operating system drive. If any of these components change while BitLocker protection is in effect, the TPM doesn't release the encryption key to unlock the drive. Instead, the computer displays the BitLocker Recovery console and requires that the recovery password or the recovery key is provided to unlock the drive.

When disabled or not configured	The TPM uses the default platform validation profile or the platform validation profile that is specified by the setup script.

Reference

This policy setting doesn't apply if the computer doesn't have a compatible TPM or if BitLocker is already turned on with TPM protection.

A platform validation profile consists of a set of PCR indices that range from 0 to 23. The default platform validation profile secures the encryption key against changes to the following:

- Core Root of Trust of Measurement (CRTM), BIOS, and Platform Extensions (PCR 0)
- Option ROM Code (PCR 2)
- Master Boot Record (MBR) Code (PCR 4)
- NTFS Boot Sector (PCR 8)
- NTFS Boot Block (PCR 9)
- Boot Manager (PCR 10)
- BitLocker Access Control (PCR 11)

NOTE

The default TPM validation profile PCR settings for computers that use an Extensible Firmware Interface (EFI) are the PCRs 0, 2, 4, and 11 only.

The following list identifies all of the available PCRs:

- PCR 0: Core root-of-trust for measurement, EFI boot and run-time services, EFI drivers embedded in system ROM, ACPI static tables, embedded SMM code, and BIOS code
- PCR 1: Platform and motherboard configuration and data. Hand-off tables and EFI variables that affect system configuration
- PCR 2: Option ROM code
- PCR 3: Option ROM data and configuration
- PCR 4: Master Boot Record (MBR) code or code from other boot devices
- PCR 5: Master Boot Record (MBR) partition table. Various EFI variables and the GPT table
- PCR 6: State transition and wake events
- PCR 7: Computer manufacturer-specific
- PCR 8: NTFS boot sector
- PCR 9: NTFS boot block
- PCR 10: Boot manager
- PCR 11: BitLocker access control
- PCR 12 - 23: Reserved for future use

WARNING

Changing from the default platform validation profile affects the security and manageability of your computer. BitLocker's sensitivity to platform modifications (malicious or authorized) is increased or decreased depending on inclusion or exclusion (respectively) of the PCRs.

Configure TPM platform validation profile for native UEFI firmware configurations

This policy setting determines what values the TPM measures when it validates early boot components before unlocking an operating system drive on a computer with native UEFI firmware configurations.

Policy description	With this policy setting, you can configure how the computer's Trusted Platform Module (TPM) security hardware secures the BitLocker encryption key.
Introduced	Windows Server 2012 and Windows 8
Drive type	Operating system drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Operating System Drives
Conflicts	<p>Setting this policy with PCR 7 omitted, overrides the Allow Secure Boot for integrity validation Group Policy setting, and it prevents BitLocker from using Secure Boot for platform or Boot Configuration Data (BCD) integrity validation.</p> <p>If your environments use TPM and Secure Boot for platform integrity checks, this policy is configured.</p> <p>For more information about PCR 7, see Platform Configuration Register (PCR) in this article.</p>
When enabled	Before you turn on BitLocker, you can configure the boot components that the TPM validates before it unlocks access to the BitLocker-encrypted operating system drive. If any of these components change while BitLocker protection is in effect, the TPM doesn't release the encryption key to unlock the drive. Instead, the computer displays the BitLocker Recovery console and requires that the recovery password or the recovery key is provided to unlock the drive.
When disabled or not configured	BitLocker uses the default platform validation profile or the platform validation profile that is specified by the setup script.

Reference

This policy setting doesn't apply if the computer doesn't have a compatible TPM or if BitLocker is already turned on with TPM protection.

IMPORTANT

This group policy setting only applies to computers with a native UEFI firmware configuration. Computers with BIOS or UEFI firmware with a Compatibility Support Module (CSM) enabled store different values in the Platform Configuration Registers (PCRs). Use the **Configure TPM platform validation profile for BIOS-based firmware configurations** Group Policy setting to configure the TPM PCR profile for computers with BIOS configurations or for computers with UEFI firmware with a CSM enabled.

A platform validation profile consists of a set of PCR indices ranging from 0 to 23. The default platform validation profile secures the encryption key against changes to the core system firmware executable code (PCR 0), extended or pluggable executable code (PCR 2), boot manager (PCR 4), and the BitLocker access control (PCR

11).

The following list identifies all of the available PCRs:

- PCR 0: Core System Firmware executable code
- PCR 1: Core System Firmware data
- PCR 2: Extended or pluggable executable code
- PCR 3: Extended or pluggable firmware data
- PCR 4: Boot Manager
- PCR 5: GPT/Partition Table
- PCR 6: Resume from S4 and S5 Power State Events
- PCR 7: Secure Boot State

For more information about this PCR, see [Platform Configuration Register \(PCR\)](#) in this article.

- PCR 8: Initialized to 0 with no Extends (reserved for future use)
- PCR 9: Initialized to 0 with no Extends (reserved for future use)
- PCR 10: Initialized to 0 with no Extends (reserved for future use)
- PCR 11: BitLocker access control
- PCR 12: Data events and highly volatile events
- PCR 13: Boot Module Details
- PCR 14: Boot Authorities
- PCR 15 – 23: Reserved for future use

WARNING

Changing from the default platform validation profile affects the security and manageability of your computer. BitLocker's sensitivity to platform modifications (malicious or authorized) is increased or decreased depending on inclusion or exclusion (respectively) of the PCRs.

Reset platform validation data after BitLocker recovery

This policy setting determines if you want platform validation data to refresh when Windows is started following a BitLocker recovery. A platform validation data profile consists of the values in a set of Platform Configuration Register (PCR) indices that range from 0 to 23.

Policy description	With this policy setting, you can control whether platform validation data is refreshed when Windows is started following a BitLocker recovery.
Introduced	Windows Server 2012 and Windows 8
Drive type	Operating system drives

Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Operating System Drives
Conflicts	None
When enabled	Platform validation data is refreshed when Windows is started following a BitLocker recovery.
When disabled	Platform validation data isn't refreshed when Windows is started following a BitLocker recovery.
When not configured	Platform validation data is refreshed when Windows is started following a BitLocker recovery.

Reference

For more information about the recovery process, see the [BitLocker recovery guide](#).

Use enhanced Boot Configuration Data validation profile

This policy setting determines specific Boot Configuration Data (BCD) settings to verify during platform validation. A platform validation uses the data in the platform validation profile, which consists of a set of Platform Configuration Register (PCR) indices that range from 0 to 23.

Policy description	With this policy setting, you can specify Boot Configuration Data (BCD) settings to verify during platform validation.
Introduced	Windows Server 2012 and Windows 8
Drive type	Operating system drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Operating System Drives
Conflicts	When BitLocker is using Secure Boot for platform and Boot Configuration Data integrity validation, the Use enhanced Boot Configuration Data validation profile Group Policy setting is ignored (as defined by the Allow Secure Boot for integrity validation Group Policy setting).
When enabled	You can add additional BCD settings, exclude the BCD settings you specify, or combine inclusion and exclusion lists to create a customized BCD validation profile, which gives you the ability to verify those BCD settings.
When disabled	The computer reverts to a BCD profile validation similar to the default BCD profile that is used by Windows 7.
When not configured	The computer verifies the default BCD settings in Windows.

Reference

NOTE

The setting that controls boot debugging (0x16000010) is always validated, and it has no effect if it's included in the inclusion or the exclusion list.

Allow access to BitLocker-protected fixed data drives from earlier versions of Windows

This policy setting is used to control whether access to drives is allowed by using the BitLocker To Go Reader, and whether BitLocker To Go Reader can be installed on the drive.

Policy description	With this policy setting, you can configure whether fixed data drives that are formatted with the FAT file system can be unlocked and viewed on computers running Windows Vista, Windows XP with Service Pack 3 (SP3), or Windows XP with Service Pack 2 (SP2).
Introduced	Windows Server 2008 R2 and Windows 7
Drive type	Fixed data drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Fixed Data Drives
Conflicts	None
When enabled and When not configured	Fixed data drives that are formatted with the FAT file system can be unlocked on computers running Windows Server 2008, Windows Vista, Windows XP with SP3, or Windows XP with SP2, and their content can be viewed. These operating systems have Read-only access to BitLocker-protected drives.
When disabled	Fixed data drives that are formatted with the FAT file system and are BitLocker-protected can't be unlocked on computers running Windows Vista, Windows XP with SP3, or Windows XP with SP2. BitLocker To Go Reader (bitlockertogo.exe) isn't installed.

Reference**NOTE**

This policy setting doesn't apply to drives that are formatted with the NTFS file system.

When this policy setting is enabled, select the **Do not install BitLocker To Go Reader on FAT formatted fixed drives** check box to help prevent users from running BitLocker To Go Reader from their fixed drives. If BitLocker To Go Reader (bitlockertogo.exe) is present on a drive that doesn't have an identification field specified, or if the drive has the same identification field as specified in the **Provide unique identifiers for your organization** policy setting, the user is prompted to update BitLocker, and BitLocker To Go Reader is deleted from the drive. In this situation, for the fixed drive to be unlocked on computers running Windows Vista, Windows XP with SP3, or Windows XP with SP2, BitLocker To Go Reader must be installed on the computer. If this check box isn't selected, then BitLocker To Go Reader will be installed on the fixed drive to enable users to unlock the drive on computers running Windows Vista, Windows XP with SP3, or Windows XP with SP2.

Allow access to BitLocker-protected removable data drives from earlier versions of Windows

This policy setting controls access to removable data drives that are using the BitLocker To Go Reader and whether the BitLocker To Go Reader can be installed on the drive.

Policy description	With this policy setting, you can configure whether removable data drives that are formatted with the FAT file system can be unlocked and viewed on computers running Windows Vista, Windows XP with SP3, or Windows XP with SP2.
Introduced	Windows Server 2008 R2 and Windows 7
Drive type	Removable data drives
Policy path	Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Removable Data Drives
Conflicts	None
When enabled and When not configured	Removable data drives that are formatted with the FAT file system can be unlocked on computers running Windows Vista, Windows XP with SP3, or Windows XP with SP2, and their content can be viewed. These operating systems have Read-only access to BitLocker-protected drives.
When disabled	Removable data drives that are formatted with the FAT file system that are BitLocker-protected can't be unlocked on computers running Windows Vista, Windows XP with SP3, or Windows XP with SP2. BitLocker To Go Reader (bitlockertogo.exe) isn't installed.

Reference

NOTE

This policy setting doesn't apply to drives that are formatted with the NTFS file system.

When this policy setting is enabled, select the **Do not install BitLocker To Go Reader on FAT formatted removable drives** check box to help prevent users from running BitLocker To Go Reader from their removable drives. If BitLocker To Go Reader (bitlockertogo.exe) is present on a drive that doesn't have an identification field specified, or if the drive has the same identification field as specified in the **Provide unique identifiers for your organization** policy setting, the user will be prompted to update BitLocker, and BitLocker To Go Reader is deleted from the drive. In this situation, for the removable drive to be unlocked on computers running Windows Vista, Windows XP with SP3, or Windows XP with SP2, BitLocker To Go Reader must be installed on the computer. If this check box isn't selected, then BitLocker To Go Reader will be installed on the removable drive to enable users to unlock the drive on computers running Windows Vista, Windows XP with SP3, or Windows XP with SP2 that don't have BitLocker To Go Reader installed.

FIPS setting

You can configure the Federal Information Processing Standard (FIPS) setting for FIPS compliance. As an effect of FIPS compliance, users can't create or save a BitLocker password for recovery or as a key protector. The use of

a recovery key is permitted.

Policy description	Notes
Introduced	Windows Server 2003 with SP1
Drive type	System-wide
Policy path	Local Policies\Security Options\System cryptography: Use FIPS compliant algorithms for encryption, hashing, and signing
Conflicts	Some applications, such as Terminal Services, don't support FIPS-140 on all operating systems.
When enabled	Users will be unable to save a recovery password to any location. This includes AD DS and network folders. Also, you can't use WMI or the BitLocker Drive Encryption Setup wizard to create a recovery password.
When disabled or not configured	No BitLocker encryption key is generated

Reference

This policy must be enabled before any encryption key is generated for BitLocker. When this policy is enabled, BitLocker prevents creating or using recovery passwords, so recovery keys should be used instead.

You can save the optional recovery key to a USB drive. Because recovery passwords can't be saved to AD DS when FIPS is enabled, an error is caused if AD DS backup is required by Group Policy.

You can edit the FIPS setting by using the Security Policy Editor (Secpol.msc) or by editing the Windows registry. You must be an administrator to perform these procedures.

For more information about setting this policy, see [System cryptography: Use FIPS compliant algorithms for encryption, hashing, and signing](#).

Power management group policy settings: Sleep and Hibernate

PCs default power settings for a computer will cause the computer to enter Sleep mode frequently to conserve power when idle and to help extend the system's battery life. When a computer transitions to Sleep, open programs and documents are persisted in memory. When a computer resumes from Sleep, users aren't required to reauthenticate with a PIN or USB startup key to access encrypted data. This might lead to conditions where data security is compromised.

However, when a computer hibernates the drive is locked, and when it resumes from hibernation the drive is unlocked, which means that users will need to provide a PIN or a startup key if using multifactor authentication with BitLocker. Therefore, organizations that use BitLocker may want to use Hibernate instead of Sleep for improved security. This setting doesn't have an impact on TPM-only mode, because it provides a transparent user experience at startup and when resuming from the Hibernate states.

You can disable the following Group Policy settings, which are located in **Computer Configuration\Administrative Templates\System\Power Management** to disable all available sleep states:

- Allow Standby States (S1-S3) When Sleeping (Plugged In)

- [Allow Standby States \(S1-S3\) When Sleeping \(Battery\)](#)

About the Platform Configuration Register (PCR)

A platform validation profile consists of a set of PCR indices that range from 0 to 23. The scope of the values can be specific to the version of the operating system.

Changing from the default platform validation profile affects the security and manageability of your computer. BitLocker's sensitivity to platform modifications (malicious or authorized) is increased or decreased depending on inclusion or exclusion (respectively) of the PCRs.

About PCR 7

PCR 7 measures the state of Secure Boot. With PCR 7, BitLocker can use Secure Boot for integrity validation. Secure Boot ensures that the computer's preboot environment loads only firmware that is digitally signed by authorized software publishers. PCR 7 measurements indicate whether Secure Boot is on and which keys are trusted on the platform. If Secure Boot is on and the firmware measures PCR 7 correctly per the UEFI specification, BitLocker can bind to this information rather than to PCRs 0, 2, and 4, which have the measurements of the exact firmware and Bootmgr images loaded. This reduces the likelihood of BitLocker starting in recovery mode as a result of firmware and image updates, and it provides you with greater flexibility to manage the preboot configuration.

PCR 7 measurements must follow the guidance that is described in [Appendix A Trusted Execution Environment EFI Protocol](#).

PCR 7 measurements are a mandatory logo requirement for systems that support Modern Standby (also known as Always On, Always Connected PCs), such as the Microsoft Surface RT. On such systems, if the TPM with PCR 7 measurement and secure boot are correctly configured, BitLocker binds to PCR 7 and PCR 11 by default.

See also

- [Trusted Platform Module](#)
- [TPM Group Policy settings](#)
- [BitLocker frequently asked questions \(FAQ\)](#)
- [BitLocker overview](#)
- [Prepare your organization for BitLocker: Planning and policies](#)

Boot Configuration Data settings and BitLocker

7/1/2022 • 6 minutes to read • [Edit Online](#)

Applies to

This topic for IT professionals describes the Boot Configuration Data (BCD) settings that are used by BitLocker.

When protecting data at rest on an operating system volume, during the boot process BitLocker verifies that the security sensitive BCD settings have not changed since BitLocker was last enabled, resumed, or recovered.

BitLocker and BCD Settings

In Windows 7 and Windows Server 2008 R2, BitLocker validated BCD settings with the winload, winresume, and memtest prefixes to a large degree. However, this high degree of validation caused BitLocker to go into recovery mode for benign setting changes, for example, when applying a language pack, BitLocker would enter recovery mode.

In Windows 8, Windows Server 2012, and later operating systems, BitLocker narrows the set of BCD settings validated to reduce the chance of benign changes causing a BCD validation problem. If you believe that there is a risk in excluding a particular BCD setting from the validation profile, include that BCD setting in the BCD validation coverage to suit your validation preferences. If a default BCD setting is found to persistently trigger a recovery for benign changes, exclude that BCD setting from the validation coverage.

When secure boot is enabled

Computers with UEFI firmware can use secure boot to provide enhanced boot security. When BitLocker is able to use secure boot for platform and BCD integrity validation, as defined by the **Allow Secure Boot for integrity validation** group policy setting, the **Use enhanced Boot Configuration Data validation profile** group policy is ignored.

One of the benefits of using secure boot is that it can correct BCD settings during boot without triggering recovery events. Secure boot enforces the same BCD settings as BitLocker. Secure boot BCD enforcement is not configurable from within the operating system.

Customizing BCD validation settings

To modify the BCD settings that are validated by BitLocker, the administrator will add or exclude BCD settings from the platform validation profile by enabling and configuring the **Use enhanced Boot Configuration Data validation profile** group policy setting.

For the purposes of BitLocker validation, BCD settings are associated with a specific set of Microsoft boot applications. These BCD settings can also be applied to the other Microsoft boot applications that are not part of the set to which the BCD settings are already applicable to. This can be done by attaching any of the following prefixes to the BCD settings which are being entered in the group policy settings dialog:

- winload
- winresume
- memtest
- all of the above

All BCD settings are specified by combining the prefix value with either a hexadecimal (hex) value or a "friendly name."

The BCD setting hex value is reported when BitLocker enters recovery mode and is stored in the event log (event ID 523). The hex value uniquely identifies the BCD setting that caused the recovery event.

You can quickly obtain the friendly name for the BCD settings on your computer by using the command "`bcdedit.exe /enum all`".

Not all BCD settings have friendly names; for those settings without a friendly name, the hex value is the only way to configure an exclusion policy.

When specifying BCD values in the **Use enhanced Boot Configuration Data validation profile** group policy setting, use the following syntax:

- Prefix the setting with the boot application prefix
- Append a colon ':'
- Append either the hex value or the friendly name
- If entering more than one BCD setting, you will need to enter each BCD setting on a new line

For example, either "`winload:hypervisordebugport`" or "`winload:0x250000f4`" yields the same value.

A setting that applies to all boot applications may be applied only to an individual application; however, the reverse is not true. For example, one can specify either "`all:locale`" or "`winresume:locale`", but as the BCD setting "`win-pe`" does not apply to all boot applications, "`winload:winpe`" is valid, but "`all:winpe`" is not valid. The setting that controls boot debugging ("`bootdebug`" or 0x16000010) will always be validated and will have no effect if it is included in the provided fields.

NOTE

Take care when configuring BCD entries in the Group Policy setting. The Local Group Policy Editor does not validate the correctness of the BCD entry. BitLocker will fail to be enabled if the Group Policy setting specified is invalid.

Default BCD validation profile

The following table contains the default BCD validation profile used by BitLocker in Windows 8, Windows Server 2012, and subsequent versions:

HEX VALUE	PREFIX	FRIENDLY NAME
0x11000001	all	device
0x12000002	all	path
0x12000030	all	loadoptions
0x16000010	all	bootdebug
0x16000040	all	advancedoptions
0x16000041	all	optionsedit
0x16000048	all	nointegritychecks
0x16000049	all	testsigning
0x16000060	all	isolatedcontext

HEX VALUE	PREFIX	FRIENDLY NAME
0x1600007b	all	forcefipscrypto
0x22000002	winload	systemroot
0x22000011	winload	kernel
0x22000012	winload	hal
0x22000053	winload	evstore
0x25000020	winload	nx
0x25000052	winload	restrictapiccluster
0x26000022	winload	winpe
0x26000025	winload	lastknowngood
0x26000081	winload	safebootalternateshell
0x260000a0	winload	debug
0x260000f2	winload	hypervisordebug
0x26000116	winload	hypervisorusevapic
0x21000001	winresume	filedevice
0x22000002	winresume	filepath
0x26000006	winresume	debugoptionenabled

Full list of friendly names for ignored BCD settings

This following is a full list of BCD settings with friendly names, which are ignored by default. These settings are not part of the default BitLocker validation profile, but can be added if you see a need to validate any of these settings before allowing a BitLocker-protected operating system drive to be unlocked.

NOTE

Additional BCD settings exist that have hex values but do not have friendly names. These settings are not included in this list.

HEX VALUE	PREFIX	FRIENDLY NAME
0x12000004	all	description
0x12000005	all	locale
0x12000016	all	targetname

HEX VALUE	PREFIX	FRIENDLY NAME
0x12000019	all	busparams
0x1200001d	all	key
0x1200004a	all	fontpath
0x14000006	all	inherit
0x14000008	all	recoverysquence
0x15000007	all	truncatememory
0x1500000c	all	firstmegabytepolicy
0x1500000d	all	relocatephysical
0x1500000e	all	avoidlowmemory
0x15000011	all	debugtype
0x15000012	all	debugaddress
0x15000013	all	debugport
0x15000014	all	baudrate
0x15000015	all	channel
0x15000018	all	debugstart
0x1500001a	all	hostip
0x1500001b	all	port
0x15000022	all	emSPORT
0x15000023	all	emSBaudrate
0x15000042	all	keyringaddress
0x15000047	all	configaccesspolicy
0x1500004b	all	integrityservices
0x1500004c	all	volumebandid
0x15000051	all	initialconsoleinput
0x15000052	all	graphicsresolution

HEX VALUE	PREFIX	FRIENDLY NAME
0x15000065	all	displaymessage
0x15000066	all	displaymessageoverride
0x15000081	all	logcontrol
0x16000009	all	recoveryenabled
0x1600000b	all	badmemoryaccess
0x1600000f	all	traditionalkseg
0x16000017	all	noumex
0x1600001c	all	dhcp
0x1600001e	all	vm
0x16000020	all	bootems
0x16000046	all	graphicsmodedisabled
0x16000050	all	extendedinput
0x16000053	all	restartonfailure
0x16000054	all	highestmode
0x1600006c	all	bootuxdisabled
0x16000072	all	nokeyboard
0x16000074	all	bootshutdowndisabled
0x1700000a	all	badmemorylist
0x17000077	all	allowedinmemorysettings
0x22000040	all	fvrecoveryurl
0x22000041	all	fvrecoverymessage
0x31000003	all	ramdisksdidevice
0x32000004	all	ramdisksdipath
0x35000001	all	ramdiskimageoffset
0x35000002	all	ramdiskftppclientport

HEX VALUE	PREFIX	FRIENDLY NAME
0x35000005	all	ramdiskimagelength
0x35000007	all	ramdisktftpblocksize
0x35000008	all	ramdisktftpwindowsize
0x36000006	all	exportasc
0x36000009	all	ramdiskmcenabled
0x3600000a	all	ramdiskmctftpfallback
0x3600000b	all	ramdisktftpvarwindow
0x21000001	winload	osdevice
0x22000013	winload	dbgtransport
0x220000f9	winload	hypervisorbusparams
0x22000110	winload	hypervisorusekey
0x23000003	winload	resumeobject
0x25000021	winload	pae
0x25000031	winload	removememory
0x25000032	winload	increaseuserva
0x25000033	winload	perfmem
0x25000050	winload	clustermodeaddressing
0x25000055	winload	x2apicpolicy
0x25000061	winload	numproc
0x25000063	winload	configflags
0x25000066	winload	groupsize
0x25000071	winload	msi
0x25000072	winload	pciexpress
0x25000080	winload	safeboot
0x250000a6	winload	tcsyncpolicy

HEX VALUE	PREFIX	FRIENDLY NAME
0x250000c1	winload	driverloadfailurepolicy
0x250000c2	winload	bootmenupolicy
0x250000e0	winload	bootstatuspolicy
0x250000f0	winload	hypervisorlaunchtype
0x250000f3	winload	hypervisordebugtype
0x250000f4	winload	hypervisordebugport
0x250000f5	winload	hypervisorbaudrate
0x250000f6	winload	hypervisorchannel
0x250000f7	winload	bootux
0x250000fa	winload	hypervisornumproc
0x250000fb	winload	hypervisorrootprocpnode
0x250000fd	winload	hypervisorhostip
0x250000fe	winload	hypervisorhostport
0x25000100	winload	tpmbootentropy
0x25000113	winload	hypervisorrootproc
0x25000115	winload	hypervisoriommpolicy
0x25000120	winload	xsavepolicy
0x25000121	winload	xsaveaddfeature0
0x25000122	winload	xsaveaddfeature1
0x25000123	winload	xsaveaddfeature2
0x25000124	winload	xsaveaddfeature3
0x25000125	winload	xsaveaddfeature4
0x25000126	winload	xsaveaddfeature5
0x25000127	winload	xsaveaddfeature6
0x25000128	winload	xsaveaddfeature7

HEX VALUE	PREFIX	FRIENDLY NAME
0x25000129	winload	xsaveremovefeature
0x2500012a	winload	xsaveprocessorsmask
0x2500012b	winload	xsavedisable
0x25000130	winload	claimedtpmcounter
0x26000004	winload	stampdisks
0x26000010	winload	detecthal
0x26000024	winload	nocrashautoreboot
0x26000030	winload	nolowmem
0x26000040	winload	vga
0x26000041	winload	quietboot
0x26000042	winload	novesa
0x26000043	winload	novga
0x26000051	winload	usephysicaldestination
0x26000054	winload	uselegacyapicmode
0x26000060	winload	onecpu
0x26000062	winload	maxproc
0x26000064	winload	maxgroup
0x26000065	winload	groupaware
0x26000070	winload	usefirmwarepcsettings
0x26000090	winload	bootlog
0x26000091	winload	sos
0x260000a1	winload	halbreakpoint
0x260000a2	winload	useplatformclock
0x260000a3	winload	forcelegacyplatform
0x260000a4	winload	useplatformtick

HEX VALUE	PREFIX	FRIENDLY NAME
0x260000a5	winload	disabledynamictick
0x260000b0	winload	ems
0x260000c3	winload	onetimeadvancedoptions
0x260000c4	winload	onetimeoptionsedit
0x260000e1	winload	disableelamdriivers
0x260000f8	winload	hypervisordisableslat
0x260000fc	winload	hypervisoruselargevtlb
0x26000114	winload	hypervisordhcp
0x21000005	winresume	associatedosdevice
0x25000007	winresume	bootux
0x25000008	winresume	bootmenupolicy
0x26000003	winresume	customsettings
0x26000004	winresume	pae
0x25000001	memtest	passcount
0x25000002	memtest	testmix
0x25000005	memtest	stridefailcount
0x25000006	memtest	invcfailcount
0x25000007	memtest	matsfailcount
0x25000008	memtest	randfailcount
0x25000009	memtest	chckrfailcount
0x26000003	memtest	cacheenable
0x26000004	memtest	failuresenabled

BitLocker recovery guide

7/1/2022 • 35 minutes to read • [Edit Online](#)

Applies to:

- Windows 10
- Windows 11
- Windows Server 2016 and above

This article for IT professionals describes how to recover BitLocker keys from AD DS.

Organizations can use BitLocker recovery information saved in Active Directory Domain Services (AD DS) to access BitLocker-protected data. Creating a recovery model for BitLocker while you are planning your BitLocker deployment is recommended.

This article assumes that you understand how to set up AD DS to back up BitLocker recovery information automatically, and what types of recovery information are saved to AD DS.

This article does not detail how to configure AD DS to store the BitLocker recovery information.

What is BitLocker recovery?

BitLocker recovery is the process by which you can restore access to a BitLocker-protected drive in the event that you cannot unlock the drive normally. In a recovery scenario, you have the following options to restore access to the drive:

- The user can supply the recovery password. If your organization allows users to print or store recovery passwords, the user can type in the 48-digit recovery password that they printed or stored on a USB drive or with your Microsoft Account online. (Saving a recovery password with your Microsoft Account online is only allowed when BitLocker is used on a PC that is not a member of a domain).
- A data recovery agent can use their credentials to unlock the drive. If the drive is an operating system drive, the drive must be mounted as a data drive on another computer for the data recovery agent to unlock it.
- A domain administrator can obtain the recovery password from AD DS and use it to unlock the drive. Storing recovery passwords in AD DS is recommended to provide a way for IT professionals to be able to obtain recovery passwords for drives in their organization if needed. This method requires that you have enabled this recovery method in the BitLocker Group Policy setting **Choose how BitLocker-protected operating system drives can be recovered** located at **Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Operating System Drives** in the Local Group Policy Editor. For more information, see [BitLocker Group Policy settings](#).

What causes BitLocker recovery?

The following list provides examples of specific events that will cause BitLocker to enter recovery mode when attempting to start the operating system drive:

- On PCs that use BitLocker Drive Encryption, or on devices such as tablets or phones that use [BitLocker Device Encryption](#) only, when an attack is detected, the device will immediately reboot and enter into BitLocker recovery mode. To take advantage of this functionality, administrators can set the **Interactive logon: Machine account lockout threshold** Group Policy setting located in **\Computer Configuration\Windows Settings\Security Settings\Local Policies\Security Options** in the Local Group Policy Editor. Or they can use the **MaxFailedPasswordAttempts** policy of [Exchange ActiveSync](#) (also configurable through [Microsoft Intune](#)), to limit the number of failed password attempts

before the device goes into Device Lockout.

- On devices with TPM 1.2, changing the BIOS or firmware boot device order causes BitLocker recovery. However, devices with TPM 2.0 do not start BitLocker recovery in this case. TPM 2.0 does not consider a firmware change of boot device order as a security threat because the OS Boot Loader is not compromised.
- Having the CD or DVD drive before the hard drive in the BIOS boot order and then inserting or removing a CD or DVD.
- Failing to boot from a network drive before booting from the hard drive.
- Docking or undocking a portable computer. In some instances (depending on the computer manufacturer and the BIOS), the docking condition of the portable computer is part of the system measurement and must be consistent to validate the system status and unlock BitLocker. So if a portable computer is connected to its docking station when BitLocker is turned on, then it might also need to be connected to the docking station when it is unlocked. Conversely, if a portable computer is not connected to its docking station when BitLocker is turned on, then it might need to be disconnected from the docking station when it is unlocked.
- Changes to the NTFS partition table on the disk including creating, deleting, or resizing a primary partition.
- Entering the personal identification number (PIN) incorrectly too many times so that the anti-hammering logic of the TPM is activated. Anti-hammering logic is software or hardware methods that increase the difficulty and cost of a brute force attack on a PIN by not accepting PIN entries until after a certain amount of time has passed.
- Turning off the support for reading the USB device in the pre-boot environment from the BIOS or UEFI firmware if you are using USB-based keys instead of a TPM.
- Turning off, disabling, deactivating, or clearing the TPM.
- Upgrading critical early startup components, such as a BIOS or UEFI firmware upgrade, causing the related boot measurements to change.
- Forgetting the PIN when PIN authentication has been enabled.
- Updating option ROM firmware.
- Upgrading TPM firmware.
- Adding or removing hardware; for example, inserting a new card in the computer, including some PCMCIA wireless cards.
- Removing, inserting, or completely depleting the charge on a smart battery on a portable computer.
- Changes to the master boot record on the disk.
- Changes to the boot manager on the disk.
- Hiding the TPM from the operating system. Some BIOS or UEFI settings can be used to prevent the enumeration of the TPM to the operating system. When implemented, this option can make the TPM hidden from the operating system. When the TPM is hidden, BIOS and UEFI secure startup are disabled, and the TPM does not respond to commands from any software.
- Using a different keyboard that does not correctly enter the PIN or whose keyboard map does not match the keyboard map assumed by the pre-boot environment. This problem can prevent the entry of enhanced PINs.

- Modifying the Platform Configuration Registers (PCRs) used by the TPM validation profile. For example, including PCR[1] would result in BitLocker measuring most changes to BIOS settings, causing BitLocker to enter recovery mode even when non-boot critical BIOS settings change.

NOTE

Some computers have BIOS settings that skip measurements to certain PCRs, such as PCR[2]. Changing this setting in the BIOS would cause BitLocker to enter recovery mode because the PCR measurement will be different.

- Moving the BitLocker-protected drive into a new computer.
- Upgrading the motherboard to a new one with a new TPM.
- Losing the USB flash drive containing the startup key when startup key authentication has been enabled.
- Failing the TPM self-test.
- Having a BIOS, UEFI firmware, or an option ROM component that is not compliant with the relevant Trusted Computing Group standards for a client computer. For example, a non-compliant implementation may record volatile data (such as time) in the TPM measurements, causing different measurements on each startup and causing BitLocker to start in recovery mode.
- Changing the usage authorization for the storage root key of the TPM to a non-zero value.

NOTE

The BitLocker TPM initialization process sets the usage authorization value to zero, so another user or process must explicitly have changed this value.

- Disabling the code integrity check or enabling test signing on Windows Boot Manager (Bootmgr).
- Pressing the F8 or F10 key during the boot process.
- Adding or removing add-in cards (such as video or network cards), or upgrading firmware on add-in cards.
- Using a BIOS hot key during the boot process to change the boot order to something other than the hard drive.

NOTE

Before you begin recovery, we recommend that you determine what caused recovery. This might help prevent the problem from occurring again in the future. For instance, if you determine that an attacker has modified your computer by obtaining physical access, you can create new security policies for tracking who has physical presence. After the recovery password has been used to recover access to the PC, BitLocker will reseal the encryption key to the current values of the measured components.

For planned scenarios, such as a known hardware or firmware upgrades, you can avoid initiating recovery by temporarily suspending BitLocker protection. Because suspending BitLocker leaves the drive fully encrypted, the administrator can quickly resume BitLocker protection after the planned task has been completed. Using suspend and resume also reseals the encryption key without requiring the entry of the recovery key.

NOTE

If suspended BitLocker will automatically resume protection when the PC is rebooted, unless a reboot count is specified using the `manage-bde` command line tool.

If software maintenance requires the computer to be restarted and you are using two-factor authentication, you can enable BitLocker Network Unlock to provide the secondary authentication factor when the computers do not have an on-premises user to provide the additional authentication method.

Recovery has been described within the context of unplanned or undesired behavior, but you can also cause recovery as an intended production scenario, in order to manage access control. For example, when you redeploy desktop or laptop computers to other departments or employees in your enterprise, you can force BitLocker into recovery before the computer is given to a new user.

Testing recovery

Before you create a thorough BitLocker recovery process, we recommend that you test how the recovery process works for both end users (people who call your helpdesk for the recovery password) and administrators (people who help the end user get the recovery password). The `-forcerecovery` command of `manage-bde` is an easy way for you to step through the recovery process before your users encounter a recovery situation.

To force a recovery for the local computer:

1. Select the **Start** button, type `cmd` in the **Start Search** box, right-click `cmd.exe`, and then select **Run as administrator**.
2. At the command prompt, type the following command and then press **Enter**:

```
manage-bde -forcerecovery <BitLockerVolume>
```

To force recovery for a remote computer:

1. On the Start screen, type `cmd.exe`, and then select **Run as administrator**.
2. At the command prompt, type the following command and then press **ENTER**:

```
manage-bde -ComputerName <RemoteComputerName> -forcerecovery <BitLockerVolume>
```

NOTE

Recovery triggered by `-forcerecovery` persists for multiple restarts until a TPM protector is added or protection is suspended by the user. When using Modern Standby devices (such as Surface devices), the `-forcerecovery` option is not recommended because BitLocker will have to be unlocked and disabled manually from the WinRE environment before the OS can boot up again. For more information, see [BitLocker Troubleshooting: Continuous reboot loop with BitLocker recovery on a slate device](#).

Planning your recovery process

When planning the BitLocker recovery process, first consult your organization's current best practices for recovering sensitive information. For example: How does your enterprise handle lost Windows passwords? How does your organization perform smart card PIN resets? You can use these best practices and related resources (people and tools) to help formulate a BitLocker recovery model.

Organizations that rely on BitLocker Drive Encryption and BitLocker To Go to protect data on a large number of computers and removable drives running the Windows 11, Windows 10, Windows 8, or Windows 7 operating systems and Windows to Go should consider using the Microsoft BitLocker Administration and Monitoring (MBAM) Tool version 2.0, which is included in the Microsoft Desktop Optimization Pack (MDOP) for Microsoft

Software Assurance. MBAM makes BitLocker implementations easier to deploy and manage and allows administrators to provision and monitor encryption for operating system and fixed drives. MBAM prompts the user before encrypting fixed drives. MBAM also manages recovery keys for fixed and removable drives, making recovery easier to manage. MBAM can be used as part of a Microsoft System Center deployment or as a stand-alone solution. For more info, see [Microsoft BitLocker Administration and Monitoring](#).

After a BitLocker recovery has been initiated, users can use a recovery password to unlock access to encrypted data. Consider both self-recovery and recovery password retrieval methods for your organization.

When you determine your recovery process, you should:

- Become familiar with how you can retrieve the recovery password. See:
 - [Self-recovery](#)
 - [Recovery password retrieval](#)
- Determine a series of steps for post-recovery, including analyzing why the recovery occurred and resetting the recovery password. See:
 - [Post-recovery analysis](#)

Self-recovery

In some cases, users might have the recovery password in a printout or a USB flash drive and can perform self-recovery. We recommend that your organization create a policy for self-recovery. If self-recovery includes using a password or recovery key stored on a USB flash drive, the users should be warned not to store the USB flash drive in the same place as the PC, especially during travel, for example if both the PC and the recovery items are in the same bag, then it's easy for an unauthorized user to access the PC. Another policy to consider is having users contact the Helpdesk before or after performing self-recovery so that the root cause can be identified.

Recovery password retrieval

If the user does not have a recovery password in a printout or on a USB flash drive, the user will need to be able to retrieve the recovery password from an online source. If the PC is a member of a domain, the recovery password can be backed up to AD DS. However, this does not happen by default. You must have configured the appropriate Group Policy settings before BitLocker was enabled on the PC. BitLocker Group Policy settings can be found in the Local Group Policy Editor or the Group Policy Management Console (GPMC) under **Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption**. The following policy settings define the recovery methods that can be used to restore access to a BitLocker-protected drive if an authentication method fails or is unable to be used.

- **Choose how BitLocker-protected operating system drives can be recovered**
- **Choose how BitLocker-protected fixed drives can be recovered**
- **Choose how BitLocker-protected removable drives can be recovered**

In each of these policies, select **Save BitLocker recovery information to Active Directory Domain Services** and then choose which BitLocker recovery information to store in Active Directory Domain Services (AD DS). Select the **Do not enable BitLocker until recovery information is stored in AD DS** check box if you want to prevent users from enabling BitLocker unless the computer is connected to the domain and the backup of BitLocker recovery information for the drive to AD DS succeeds.

NOTE

If the PCs are part of a workgroup, users should be advised to save their BitLocker recovery password with their Microsoft Account online. Having an online copy of your BitLocker recovery password is recommended to help ensure that you do not lose access to your data in the event that recovery is required.

The BitLocker Recovery Password Viewer for Active Directory Users and Computers tool allows domain

administrators to view BitLocker recovery passwords for specific computer objects in Active Directory.

You can use the following list as a template for creating your own recovery process for recovery password retrieval. This sample process uses the BitLocker Recovery Password Viewer for Active Directory Users and Computers tool.

- [Record the name of the user's computer](#)
- [Verify the user's identity](#)
- [Locate the recovery password in AD DS](#)
- [Gather information to determine why recovery occurred](#)
- [Give the user the recovery password](#)

Record the name of the user's computer

You can use the name of the user's computer to locate the recovery password in AD DS. If the user does not know the name of the computer, ask the user to read the first word of the **Drive Label** in the **BitLocker Drive Encryption Password Entry** user interface. This is the computer name when BitLocker was enabled and is probably the current name of the computer.

Verify the user's identity

Verify that the person that is asking for the recovery password is truly the authorized user of that computer. You might also want to verify that the computer with the name the user provided belongs to the user.

Locate the recovery password in AD DS

Locate the Computer object with the matching name in AD DS. Because Computer object names are listed in the AD DS global catalog, you should be able to locate the object even if you have a multi-domain forest.

Multiple recovery passwords

If multiple recovery passwords are stored under a computer object in AD DS, the name of the BitLocker recovery information object includes the date that the password was created.

If at any time you are unsure what password to provide, or if you think you might be providing the incorrect password, ask the user to read the eight character password ID that is displayed in the recovery console.

Since the password ID is a unique value that is associated with each recovery password stored in AD DS, running a query using this ID will find the correct password to unlock the encrypted volume.

Gather information to determine why recovery occurred

Before you give the user the recovery password, you should gather any information that will help determine why the recovery was needed, in order to analyze the root cause during the post-recovery analysis. For more info about post-recovery analysis, see [Post-recovery analysis](#).

Give the user the recovery password

Because the recovery password is 48 digits long, the user might need to record the password by writing it down or typing it on a different computer. If you are using MBAM, the recovery password will be regenerated after it is recovered from the MBAM database to avoid the security risks associated with an uncontrolled password.

NOTE

Because the 48-digit recovery password is long and contains a combination of digits, the user might mishear or mistype the password. The boot-time recovery console uses built-in checksum numbers to detect input errors in each 6-digit block of the 48-digit recovery password, and offers the user the opportunity to correct such errors.

Post-recovery analysis

When a volume is unlocked using a recovery password, an event is written to the event log and the platform validation measurements are reset in the TPM to match the current configuration. Unlocking the volume means

that the encryption key has been released and is ready for on-the-fly encryption when data is written to the volume, and on-the-fly decryption when data is read from the volume. After the volume is unlocked, BitLocker behaves the same way, regardless of how the access was granted.

If you notice that a computer is having repeated recovery password unlocks, you might want to have an administrator perform post-recovery analysis to determine the root cause of the recovery and refresh BitLocker platform validation so that the user no longer needs to enter a recovery password each time that the computer starts up. See:

- [Determine the root cause of the recovery](#)
- [Refresh BitLocker protection](#)

Determine the root cause of the recovery

If a user needed to recover the drive, it is important to determine the root cause that initiated the recovery as soon as possible. Properly analyzing the state of the computer and detecting tampering may reveal threats that have broader implications for enterprise security.

While an administrator can remotely investigate the cause of recovery in some cases, the end user might need to bring the computer that contains the recovered drive on site to analyze the root cause further.

Review and answer the following questions for your organization:

1. What BitLocker protection mode is in effect (TPM, TPM + PIN, TPM + startup key, startup key only)? Which PCR profile is in use on the PC?
2. Did the user merely forget the PIN or lose the startup key? If a token was lost, where might the token be?
3. If TPM mode was in effect, was recovery caused by a boot file change?
4. If recovery was caused by a boot file change, was the change an intended user action (for example, BIOS upgrade), or was it caused by malicious software?
5. When was the user last able to start the computer successfully, and what might have happened to the computer since then?
6. Might the user have encountered malicious software or left the computer unattended since the last successful startup?

To help you answer these questions, use the BitLocker command-line tool to view the current configuration and protection mode (for example, `manage-bde -status`). Scan the event log to find events that help indicate why recovery was initiated (for example, if the boot file changed). Both of these capabilities can be performed remotely.

Resolve the root cause

After you have identified what caused recovery, you can reset BitLocker protection and avoid recovery on every startup.

The details of this reset can vary according to the root cause of the recovery. If you cannot determine the root cause, or if malicious software or a rootkit might have infected the computer, Helpdesk should apply best-practice virus policies to react appropriately.

NOTE

You can perform a BitLocker validation profile reset by suspending and resuming BitLocker.

- [Unknown PIN](#)
- [Lost startup key](#)
- [Changes to boot files](#)

Unknown PIN

If a user has forgotten the PIN, you must reset the PIN while you are logged on to the computer in order to prevent BitLocker from initiating recovery each time the computer is restarted.

To prevent continued recovery due to an unknown PIN

1. Unlock the computer using the recovery password.
2. Reset the PIN:
 - a. Right-click the drive and then select **Change PIN**.
 - b. In the BitLocker Drive Encryption dialog, select **Reset a forgotten PIN**. If you are not logged in with an administrator account, provide administrative credentials at this time.
 - c. In the PIN reset dialog, provide and confirm the new PIN to use and then select **Finish**.
3. You will use the new PIN the next time you unlock the drive.

Lost startup key

If you have lost the USB flash drive that contains the startup key, then you must unlock the drive by using the recovery key and then create a new startup key.

To prevent continued recovery due to a lost startup key

1. Log on as an administrator to the computer that has the lost startup key.
2. Open Manage BitLocker.
3. Select **Duplicate start up key**, insert the clean USB drive on which you are going to write the key and then select **Save**.

Changes to boot files

This error might occur if you updated the firmware. As a best practice, you should suspend BitLocker before making changes to the firmware and then resume protection after the update has completed. This action prevents the computer from going into recovery mode. However if changes were made when BitLocker protection was on, then log on to the computer using the recovery password, and the platform validation profile will be updated so that recovery will not occur the next time.

Windows RE and BitLocker Device Encryption

Windows Recovery Environment (RE) can be used to recover access to a drive protected by [BitLocker Device Encryption](#). If a PC is unable to boot after two failures, Startup Repair will automatically start. When Startup Repair is launched automatically due to boot failures, it will only execute operating system and driver file repairs, provided that the boot logs or any available crash dump point to a specific corrupted file. In Windows 8.1 and later, devices that include firmware to support specific TPM measurements for PCR[7] the TPM can validate that Windows RE is a trusted operating environment and will unlock any BitLocker-protected drives if Windows RE has not been modified. If the Windows RE environment has been modified, for example the TPM has been disabled, the drives will stay locked until the BitLocker recovery key is provided. If Startup Repair can't run automatically from the PC and instead Windows RE is manually started from a repair disk, then the BitLocker recovery key must be provided to unlock the BitLocker-protected drives.

BitLocker recovery screen

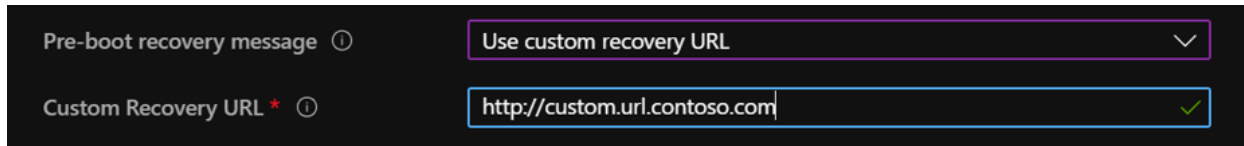
During BitLocker recovery, Windows can display a custom recovery message and hints that identify where a key can be retrieved from. These improvements can help a user during BitLocker recovery.

Custom recovery message

BitLocker Group Policy settings in Windows 10, version 1511, or Windows 11, let you configure a custom recovery message and URL on the BitLocker recovery screen, which can include the address of the BitLocker self-service recovery portal, the IT internal website, or a phone number for support.

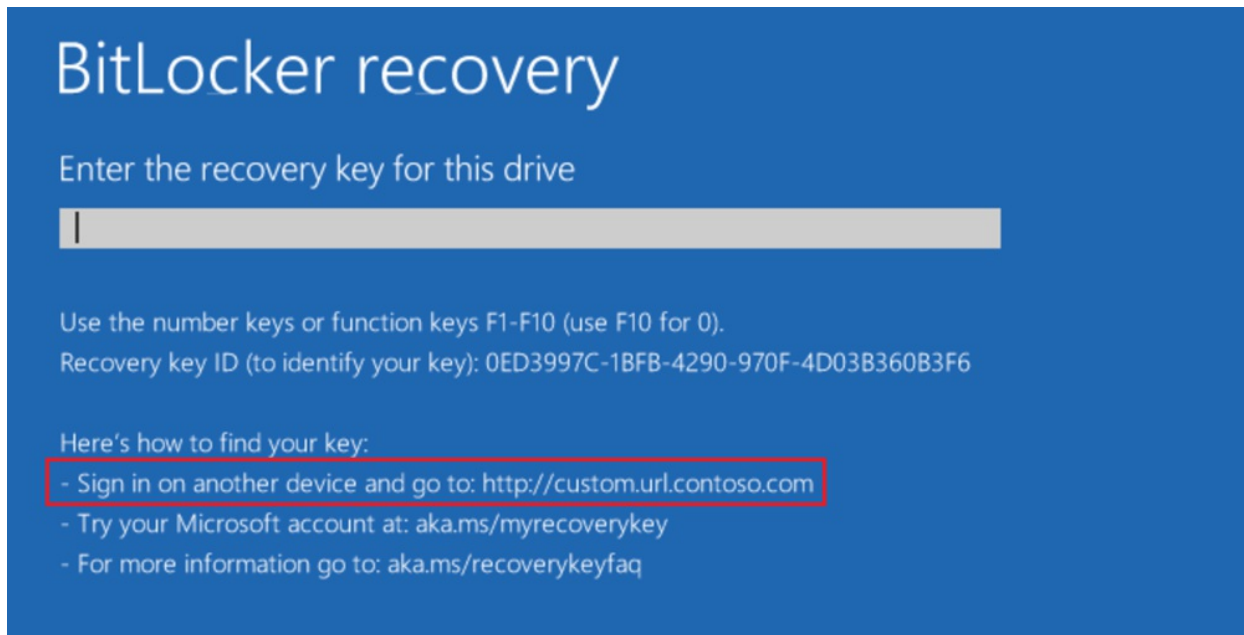
This policy can be configured using GPO under **Computer Configuration > Administrative Templates > Windows Components > BitLocker Drive Encryption > Operating System Drives > Configure pre-boot recovery message and URL.**

It can also be configured using Intune mobile device management (MDM) in the BitLocker CSP:
<LocURI>./Device/Vendor/MSFT/BitLocker/SystemDrivesRecoveryMessage</LocURI>



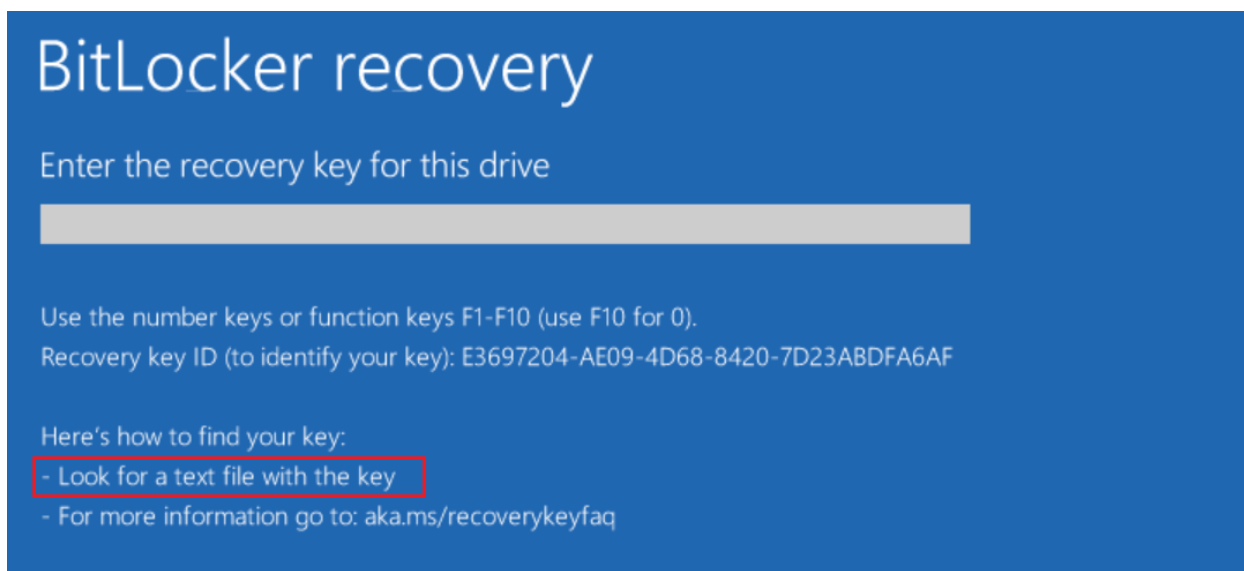
Pre-boot recovery message ⓘ	Use custom recovery URL ▾
Custom Recovery URL * ⓘ	http://custom.url.contoso.com ✓

Example of customized recovery screen:



BitLocker recovery key hints

BitLocker metadata has been enhanced in Windows 10, version 1903 or Windows 11 to include information about when and where the BitLocker recovery key was backed up. This information is not exposed through the UI or any public API. It is used solely by the BitLocker recovery screen in the form of hints to help a user locate a volume's recovery key. Hints are displayed on the recovery screen and refer to the location where the key has been saved. Hints are displayed on both the modern (blue) and legacy (black) recovery screen. This applies to both the boot manager recovery screen and the WinRE unlock screen.



IMPORTANT

We don't recommend printing recovery keys or saving them to a file. Instead, use Active Directory backup or a cloud-based backup. Cloud-based backup includes Azure Active Directory (Azure AD) and Microsoft Account.

There are rules governing which hint is shown during the recovery (in order of processing):

1. Always display custom recovery message if it has been configured (using GPO or MDM).
2. Always display generic hint: "For more information, go to <https://aka.ms/recoverykeyfaq>".
3. If multiple recovery keys exist on the volume, prioritize the last created (and successfully backed up) recovery key.
4. Prioritize keys with successful backup over keys that have never been backed up.
5. Prioritize backup hints in the following order for remote backup locations: **Microsoft Account > Azure AD > Active Directory**.
6. If a key has been printed and saved to file, display a combined hint, "Look for a printout or a text file with the key," instead of two separate hints.
7. If multiple backups of the same type (remove vs. local) have been performed for the same recovery key, prioritize backup info with latest backed up date.
8. There is no specific hint for keys saved to an on-premises Active Directory. In this case, a custom message (if configured) or a generic message, "Contact your organization's help desk," will be displayed.
9. If two recovery keys are present on the disk, but only one has been successfully backed up, the system will ask for a key that has been backed up, even if another key is newer.

Example 1 (single recovery key with single backup)

CUSTOM URL	YES
Saved to Microsoft Account	Yes
Saved to Azure AD	No
Saved to Active Directory	No
Printed	No
Saved to file	No

Result: The hint for the Microsoft Account and the custom URL are displayed.

BitLocker recovery

Enter the recovery key for this drive

Use the number keys or function keys F1-F10 (use F10 for 0).

Recovery key ID (to identify your key): ABD09F3E-C04C-4C8F-B2AE-CF0253006F7B

Here's how to find your key:

- Sign in on another device and go to: <http://custom.url.contoso.com>
- Try your Microsoft account at: aka.ms/myrecoverykey
- For more information go to: aka.ms/recoverykeyfaq

Example 2 (single recovery key with single backup)

CUSTOM URL	YES
Saved to Microsoft Account	No
Saved to Azure AD	No
Saved to Active Directory	Yes
Printed	No
Saved to file	No

Result: Only the custom URL is displayed.

BitLocker recovery

Enter the recovery key for this drive

Use the number keys or function keys F1-F10 (use F10 for 0).

Recovery key ID (to identify your key): 3D181897-89C4-46A2-8148-1D225418BEEA

Here's how to find your key:

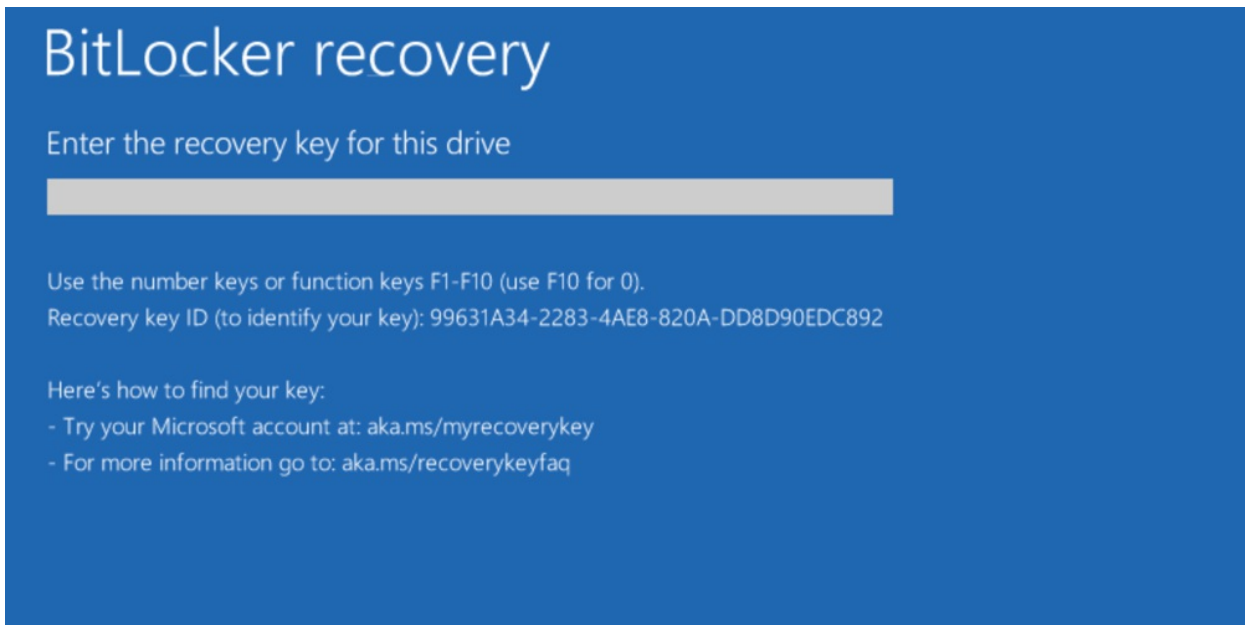
- Sign in on another device and go to: [Http://custom.url.contoso.com](http://custom.url.contoso.com)
- For more information go to: aka.ms/recoverykeyfaq

Example 3 (single recovery key with multiple backups)

CUSTOM URL	NO
Saved to Microsoft Account	Yes

CUSTOM URL	NO
Saved to Azure AD	Yes
Saved to Active Directory	No
Printed	Yes
Saved to file	Yes

Result: Only the Microsoft Account hint is displayed.



Example 4 (multiple recovery passwords)

CUSTOM URL	NO
Saved to Microsoft Account	No
Saved to Azure AD	No
Saved to Active Directory	No
Printed	No
Saved to file	Yes
Creation time	1 PM
Key ID	A564F193

CUSTOM URL	NO
Saved to Microsoft Account	No
Saved to Azure AD	No

CUSTOM URL	NO
Saved to Active Directory	No
Printed	No
Saved to file	No
Creation time	3PM
Key ID	T4521ER5

Result: Only the hint for a successfully backed up key is displayed, even if it isn't the most recent key.

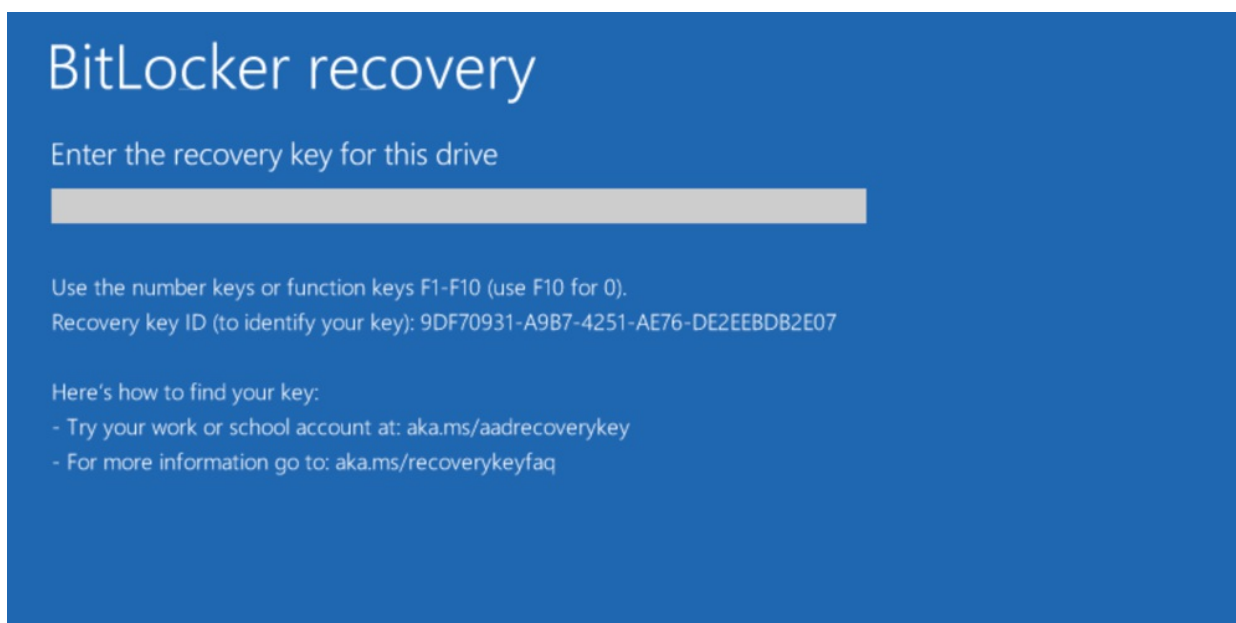
Example 5 (multiple recovery passwords)

CUSTOM URL	NO
Saved to Microsoft Account	Yes
Saved to Azure AD	Yes
Saved to Active Directory	No
Printed	No
Saved to file	No
Creation time	1PM
Key ID	99631A34

CUSTOM URL	NO
Saved to Microsoft Account	No

CUSTOM URL	NO
Saved to Azure AD	Yes
Saved to Active Directory	No
Printed	No
Saved to file	No
Creation time	3PM
Key ID	9DF70931

Result: The hint for the most recent key is displayed.



Using additional recovery information

Besides the 48-digit BitLocker recovery password, other types of recovery information are stored in Active Directory. This section describes how this additional information can be used.

BitLocker key package

If the recovery methods discussed earlier in this document do not unlock the volume, you can use the BitLocker Repair tool to decrypt the volume at the block level. The tool uses the BitLocker key package to help recover encrypted data from severely damaged drives. You can then use this recovered data to salvage encrypted data, even after the correct recovery password has failed to unlock the damaged volume. We recommend that you still save the recovery password. A key package cannot be used without the corresponding recovery password.

NOTE

You must use the BitLocker Repair tool **repair-bde** to use the BitLocker key package.

The BitLocker key package is not saved by default. To save the package along with the recovery password in AD DS, you must select the **Backup recovery password and key package** option in the Group Policy settings that control the recovery method. You can also export the key package from a working volume. For more details about how to export key packages, see [Retrieving the BitLocker Key Package](#).

Resetting recovery passwords

Invalidate a recovery password after it has been provided and used. It should also be done when you intentionally want to invalidate an existing recovery password for any reason.

You can reset the recovery password in two ways:

- **Use manage-bde:** You can use manage-bde to remove the old recovery password and add a new recovery password. The procedure identifies the command and the syntax for this method.
- **Run a script:** You can run a script to reset the password without decrypting the volume. The sample script in the procedure illustrates this functionality. The sample script creates a new recovery password and invalidates all other passwords.

To reset a recovery password using manage-bde:

1. Remove the previous recovery password

```
Manage-bde -protectors -delete C: -type RecoveryPassword
```

2. Add the new recovery password

```
Manage-bde -protectors -add C: -RecoveryPassword
```

3. Get the ID of the new recovery password. From the screen, copy the ID of the recovery password.

```
Manage-bde -protectors -get C: -Type RecoveryPassword
```

4. Back up the new recovery password to AD DS.

```
Manage-bde -protectors -adbackup C: -id {EXAMPLE6-5507-4924-AA9E-AFB2EB003692}
```

WARNING

You must include the braces in the ID string.

To run the sample recovery password script:

1. Save the following sample script in a VBScript file. For example: ResetPassword.vbs.
2. At the command prompt, type a command similar to the following sample script:

```
cscript ResetPassword.vbs
```

IMPORTANT

This sample script is configured to work only for the C volume. You must customize the script to match the volume where you want to test password reset.

NOTE

To manage a remote computer, you can specify the remote computer name rather than the local computer name.

You can use the following sample script to create a VBScript file to reset the recovery passwords:

```
' Target drive letter
strDriveLetter = "c:"
' Target computer name
' Use "." to connect to the local computer
strComputerName = "."
' -----
' Connect to the BitLocker WMI provider class
' -----
strConnectionStr = "winmgmts:" _
    & "{impersonationLevel=impersonate,authenticationLevel=pktPrivacy}!\" _
    & strComputerName _
    & "\root\cimv2\Security\MicrosoftVolumeEncryption"

On Error Resume Next 'handle permission errors
Set objWMIService = GetObject(strConnectionStr)
If Err.Number <> 0 Then
    WScript.Echo "Failed to connect to the BitLocker interface (Error 0x" & Hex(Err.Number) & ")."
    WScript.Echo "Ensure that you are running with administrative privileges."
    WScript.Quit -1
End If
On Error GoTo 0
strQuery = "Select * from Win32_EncryptableVolume where DriveLetter='" & strDriveLetter & "'"
Set colTargetVolumes = objWMIService.ExecQuery(strQuery)
If colTargetVolumes.Count = 0 Then
    WScript.Echo "FAILURE: Unable to find BitLocker-capable drive " & strDriveLetter & " on computer " &
strComputerName & "."
    WScript.Quit -1
End If
' there should only be one volume found
For Each objFoundVolume in colTargetVolumes
    set objVolume = objFoundVolume
Next
' objVolume is now our found BitLocker-capable disk volume
' -----
' Perform BitLocker WMI provider functionality
' -----
' Add a new recovery password, keeping the ID around so it doesn't get deleted later
' -----
nRC = objVolume.ProtectKeyWithNumericalPassword("Recovery Password Refreshed By Script", ,
sNewKeyProtectorID)
If nRC <> 0 Then
WScript.Echo "FAILURE: ProtectKeyWithNumericalPassword failed with return code 0x" & Hex(nRC)
WScript.Quit -1
End If
' Removes the other, "stale", recovery passwords
' -----
nKeyProtectorTypeIn = 3 ' type associated with "Numerical Password" protector
nRC = objVolume.GetKeyProtectors(nKeyProtectorTypeIn, aKeyProtectorIDs)
If nRC <> 0 Then
WScript.Echo "FAILURE: GetKeyProtectors failed with return code 0x" & Hex(nRC)
WScript.Quit -1
End If
' Delete those key protectors other than the one we just added.
For Each sKeyProtectorID In aKeyProtectorIDs
If sKeyProtectorID <> sNewKeyProtectorID Then
nRC = objVolume.DeleteKeyProtector(sKeyProtectorID)
If nRC <> 0 Then
WScript.Echo "FAILURE: DeleteKeyProtector on ID " & sKeyProtectorID & " failed with return code 0x" &
Hex(nRC)
WScript.Quit -1
Else
' no output
'WScript.Echo "SUCCESS: Key protector with ID " & sKeyProtectorID & " deleted"
End If
End If
```

```

Next
WScript.Echo "A new recovery password has been added. Old passwords have been removed."
' - some advanced output (hidden)
'WScript.Echo ""
'WScript.Echo "Type ""manage-bde -protectors -get " & strDriveLetter & " -type recoverypassword"" to view
existing passwords."

```

Retrieving the BitLocker key package

You can use two methods to retrieve the key package, as described in [Using Additional Recovery Information](#):

- **Export a previously saved key package from AD DS.** You must have Read access to BitLocker recovery passwords that are stored in AD DS.
- **Export a new key package from an unlocked, BitLocker-protected volume.** You must have local administrator access to the working volume, before any damage has occurred.

The following sample script exports all previously saved key packages from AD DS.

To run the sample key package retrieval script:

1. Save the following sample script in a VBScript file. For example: GetBitLockerKeyPackageADDS.vbs.
2. At the command prompt, type a command similar to the following sample script:

```
cscript GetBitLockerKeyPackageADDS.vbs -?
```

You can use the following sample script to create a VBScript file to retrieve the BitLocker key package from AD DS:

```

' -----
' Usage
' -----
Sub ShowUsage
    Wscript.Echo "USAGE: GetBitLockerKeyPackageADDS [Path To Save Key Package] [Optional Computer Name]"
    Wscript.Echo "If no computer name is specified, the local computer is assumed."
    Wscript.Echo
    Wscript.Echo "Example: GetBitLockerKeyPackageADDS E:\bitlocker-ad-key-package mycomputer"
    WScript.Quit
End Sub
' -----
' Parse Arguments
' -----
Set args = WScript.Arguments
Select Case args.Count
    Case 1
        If args(0) = "/"? Or args(0) = "-?" Then
            ShowUsage
        Else
            strFilePath = args(0)
            ' Get the name of the local computer
            Set objNetwork = CreateObject("WScript.Network")
            strComputerName = objNetwork.ComputerName
        End If

    Case 2
        If args(0) = "/"? Or args(0) = "-?" Then
            ShowUsage
        Else
            strFilePath = args(0)
            strComputerName = args(1)
        End If
    Case Else
        ShowUsage
End Select

```

```

' -----
' Get path to Active Directory computer object associated with the computer name
' -----
Function GetStrPathToComputer(strComputerName)
    ' Uses the global catalog to find the computer in the forest
    ' Search also includes deleted computers in the tombstone
    Set objRootLDAP = GetObject("LDAP://rootDSE")
    namingContext = objRootLDAP.Get("defaultNamingContext") ' e.g. string dc=fabrikam,dc=com
    strBase = "<GC://" & namingContext & ">"

    Set objConnection = CreateObject("ADODB.Connection")
    Set objCommand = CreateObject("ADODB.Command")
    objConnection.Provider = "ADsDSOObject"
    objConnection.Open "Active Directory Provider"
    Set objCommand.ActiveConnection = objConnection
    strFilter = "(&(objectCategory=Computer)(cn=" & strComputerName & "))"
    strQuery = strBase & ";" & strFilter & ";distinguishedName;subtree"
    objCommand.CommandText = strQuery
    objCommand.Properties("Page Size") = 100
    objCommand.Properties("Timeout") = 100
    objCommand.Properties("Cache Results") = False
    ' Enumerate all objects found.
    Set objRecordSet = objCommand.Execute
    If objRecordSet.EOF Then
        WScript.Echo "The computer name '" & strComputerName & "' cannot be found."
        WScript.Quit 1
    End If
    ' Found object matching name
    Do Until objRecordSet.EOF
        dnFound = objRecordSet.Fields("distinguishedName")
        GetStrPathToComputer = "LDAP://" & dnFound
        objRecordSet.MoveNext
    Loop
    ' Clean up.
    Set objConnection = Nothing
    Set objCommand = Nothing
    Set objRecordSet = Nothing
End Function
' -----
' Securely access the Active Directory computer object using Kerberos
' -----

Set objDSO = GetObject("LDAP:")
strPathToComputer = GetStrPathToComputer(strComputerName)
WScript.Echo "Accessing object: " + strPathToComputer
Const ADS_SECURE_AUTHENTICATION = 1
Const ADS_USE_SEALING = 64 '0x40
Const ADS_USE_SIGNING = 128 '0x80
' -----
' Get all BitLocker recovery information from the Active Directory computer object
' -----

' Get all the recovery information child objects of the computer object
Set objFveInfos = objDSO.OpenDSObject(strPathToComputer, vbNullString, vbNullString, _
    ADS_SECURE_AUTHENTICATION + ADS_USE_SEALING + ADS_USE_SIGNING)
objFveInfos.Filter = Array("msFVE-RecoveryInformation")
' Iterate through each recovery information object and saves any existing key packages
nCount = 1
strFilePathCurrent = strFilePath & nCount
For Each objFveInfo in objFveInfos
    strName = objFveInfo.Get("name")
    strRecoveryPassword = objFveInfo.Get("msFVE-RecoveryPassword")
    strKeyPackage = objFveInfo.Get("msFVE-KeyPackage")
    WScript.Echo
    WScript.Echo "Recovery Object Name: " + strName
    WScript.Echo "Recovery Password: " + strRecoveryPassword
    ' Validate file path
    Set fso = CreateObject("Scripting.FileSystemObject")
    If (fso.FileExists(strFilePathCurrent)) Then
        WScript.Echo "The file " & strFilePathCurrent & " already exists. Please use a different path."
    End If
WScript.Quit -1

```



```

WScript.Quit -1
End If
' Save binary data to the file
SaveBinaryDataText strFilePathCurrent, strKeyPackage

WScript.Echo "Related key package successfully saved to " + strFilePathCurrent
' Update next file path using base name
nCount = nCount + 1
strFilePathCurrent = strFilePath & nCount
Next
'-----
' Utility functions to save binary data
'-----
Function SaveBinaryDataText(FileName, ByteArray)
'Create FileSystemObject object
Dim FS: Set FS = CreateObject("Scripting.FileSystemObject")

'Create text stream object
Dim TextStream
Set TextStream = FS.CreateTextFile(FileName)

'Convert binary data To text And write them To the file
TextStream.Write BinaryToString(ByteArray)
End Function
Function BinaryToString(Binary)
Dim I, S
For I = 1 To LenB(Binary)
S = S & Chr(AscB(MidB(Binary, I, 1)))
Next
BinaryToString = S
End Function
WScript.Quit

```

The following sample script exports a new key package from an unlocked, encrypted volume.

To run the sample key package retrieval script:

1. Save the following sample script in a VBScript file. For example: GetBitLockerKeyPackage.vbs
2. Open an administrator command prompt, and then type a command similar to the following sample script:

cscript GetBitLockerKeyPackage.vbs -?

```

'-----
' Usage
'-----
Sub ShowUsage
Wscript.Echo "USAGE: GetBitLockerKeyPackage [VolumeLetter/DriveLetter:] [Path To Save Key Package]"
Wscript.Echo
Wscript.Echo "Example: GetBitLockerKeyPackage C: E:\bitlocker-backup-key-package"
WScript.Quit
End Sub
'-----
' Parse Arguments
'-----
Set args = WScript.Arguments
Select Case args.Count
Case 2
If args(0) = "/"? Or args(0) = "-?" Then
ShowUsage
Else
strDriveLetter = args(0)
strFilePath = args(1)
End If
Case Else
ShowUsage

```

```

End Select
' -----
' Other Inputs
' -----
' Target computer name
' Use "." to connect to the local computer
strComputerName = "."
' Default key protector ID to use. Specify "" to let the script choose.
strDefaultKeyProtectorID = ""
' strDefaultKeyProtectorID = "{001298E0-870E-4BA0-A2FF-FC74758D5720}" ' sample
' -----
' Connect to the BitLocker WMI provider class
' -----
strConnectionStr = "winmgmts:" _
    & "{impersonationLevel=impersonate,authenticationLevel=pktPrivacy}!\" _
    & strComputerName _
    & "\root\cimv2\Security\MicrosoftVolumeEncryption"

On Error Resume Next 'handle permission errors
Set objWMIService = GetObject(strConnectionStr)
If Err.Number <> 0 Then
    WScript.Echo "Failed to connect to the BitLocker interface (Error 0x" & Hex(Err.Number) & ")."
    WScript.Echo "Ensure that you are running with administrative privileges."
    WScript.Quit -1
End If
On Error GoTo 0
strQuery = "Select * from Win32_EncryptableVolume where DriveLetter='" & strDriveLetter & "'"
Set colTargetVolumes = objWMIService.ExecQuery(strQuery)
If colTargetVolumes.Count = 0 Then
    WScript.Echo "FAILURE: Unable to find BitLocker-capable drive " & strDriveLetter & " on computer " &
strComputerName & "."
    WScript.Quit -1
End If
' there should only be one volume found
For Each objFoundVolume in colTargetVolumes
    set objVolume = objFoundVolume
Next
' objVolume is now our found BitLocker-capable disk volume
' -----
' Perform BitLocker WMI provider functionality
' -----
' Collect all possible valid key protector ID's that can be used to get the package
' -----
nNumericalKeyProtectorType = 3 ' type associated with "Numerical Password" protector
nRC = objVolume.GetKeyProtectors(nNumericalKeyProtectorType, aNumericalKeyProtectorIDs)
If nRC <> 0 Then
    WScript.Echo "FAILURE: GetKeyProtectors failed with return code 0x" & Hex(nRC)
    WScript.Quit -1
End If
nExternalKeyProtectorType = 2 ' type associated with "External Key" protector
nRC = objVolume.GetKeyProtectors(nExternalKeyProtectorType, aExternalKeyProtectorIDs)
If nRC <> 0 Then
    WScript.Echo "FAILURE: GetKeyProtectors failed with return code 0x" & Hex(nRC)
    WScript.Quit -1
End If
' Get first key protector of the type "Numerical Password" or "External Key", if any
' -----
if strDefaultKeyProtectorID = "" Then
' Save first numerical password, if exists
If UBound(aNumericalKeyProtectorIDs) <> -1 Then
strDefaultKeyProtectorID = aNumericalKeyProtectorIDs(0)
End If
' No numerical passwords exist, save the first external key
If strDefaultKeyProtectorID = "" and UBound(aExternalKeyProtectorIDs) <> -1 Then
strDefaultKeyProtectorID = aExternalKeyProtectorIDs(0)
End If
' Fail case: no recovery key protectors exist.
If strDefaultKeyProtectorID = "" Then

```

```

WScript.Echo "FAILURE: Cannot create backup key package because no recovery passwords or recovery keys
exist. Check that BitLocker protection is on for this drive."
WScript.Echo "For help adding recovery passwords or recovery keys, type ""manage-bde -protectors -add -?""."
WScript.Quit -1
End If
End If
' Get some information about the chosen key protector ID
' -----
' is the type valid?
nRC = objVolume.GetKeyProtectorType(strDefaultKeyProtectorID, nDefaultKeyProtectorType)
If Hex(nRC) = "80070057" Then
WScript.Echo "The key protector ID " & strDefaultKeyProtectorID & " is not valid."
WScript.Echo "This ID value may have been provided by the script writer."
ElseIf nRC <> 0 Then
WScript.Echo "FAILURE: GetKeyProtectorType failed with return code 0x" & Hex(nRC)
WScript.Quit -1
End If
' what's a string that can be used to describe it?
strDefaultKeyProtectorType = ""
Select Case nDefaultKeyProtectorType
Case nNumericalKeyProtectorType
strDefaultKeyProtectorType = "recovery password"
Case nExternalKeyProtectorType
strDefaultKeyProtectorType = "recovery key"
Case Else
WScript.Echo "The key protector ID " & strDefaultKeyProtectorID & " does not refer to a valid recovery
password or recovery key."
WScript.Echo "This ID value may have been provided by the script writer."
End Select
' Save the backup key package using the chosen key protector ID
' -----
nRC = objVolume.GetKeyPackage(strDefaultKeyProtectorID, oKeyPackage)
If nRC <> 0 Then
WScript.Echo "FAILURE: GetKeyPackage failed with return code 0x" & Hex(nRC)
WScript.Quit -1
End If
' Validate file path
Set fso = CreateObject("Scripting.FileSystemObject")
If (fso.FileExists(strFilePath)) Then
WScript.Echo "The file " & strFilePath & " already exists. Please use a different path."
WScript.Quit -1
End If
Dim oKeyPackageByte, bKeyPackage
For Each oKeyPackageByte in oKeyPackage
'WScript.echo "key package byte: " & oKeyPackageByte
bKeyPackage = bKeyPackage & ChrB(oKeyPackageByte)
Next
' Save binary data to the file
SaveBinaryDataText strFilePath, bKeyPackage
' Display helpful information
' -----
WScript.Echo "The backup key package has been saved to " & strFilePath & "."
WScript.Echo "IMPORTANT: To use this key package, the " & strDefaultKeyProtectorType & " must also be
saved."
' Display the recovery password or a note about saving the recovery key file
If nDefaultKeyProtectorType = nNumericalKeyProtectorType Then
nRC = objVolume.GetKeyProtectorNumericalPassword(strDefaultKeyProtectorID, sNumericalPassword)
If nRC <> 0 Then
WScript.Echo "FAILURE: GetKeyProtectorNumericalPassword failed with return code 0x" & Hex(nRC)
WScript.Quit -1
End If
WScript.Echo "Save this recovery password: " & sNumericalPassword
ElseIf nDefaultKeyProtectorType = nExternalKeyProtectorType Then
WScript.Echo "The saved key file is named " & strDefaultKeyProtectorID & ".BEK"
WScript.Echo "For help re-saving this external key file, type ""manage-bde -protectors -get -?""."
End If
' -----
' Utility functions to save binary data
' -----

```

```
Function SaveBinaryDataText(FileName, ByteArray)
    'Create FileSystemObject object
    Dim FS: Set FS = CreateObject("Scripting.FileSystemObject")

    'Create text stream object
    Dim TextStream
    Set TextStream = FS.CreateTextFile(FileName)

    'Convert binary data To text And write them To the file
    TextStream.Write BinaryToString(ByteArray)
End Function
Function BinaryToString(Binary)
    Dim I, S
    For I = 1 To LenB(Binary)
        S = S & Chr(AscB(MidB(Binary, I, 1)))
    Next
    BinaryToString = S
End Function
```

See also

- [BitLocker overview](#)

BitLocker Countermeasures

7/1/2022 • 10 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016 and above

Windows uses technologies including trusted platform module (TPM), secure boot, and measured boot to help protect BitLocker encryption keys against attacks. BitLocker is part of a strategic approach to securing data against offline attacks through encryption technology. Data on a lost or stolen computer is vulnerable. For example, there could be unauthorized access, either by running a software attack tool against the computer or by transferring the computer's hard disk to a different computer.

BitLocker helps mitigate unauthorized data access on lost or stolen computers before the authorized operating system is started. This mitigation is done by:

- **Encrypting volumes on your computer.** For example, you can turn on BitLocker for your operating system volume, or a volume on a fixed or removable data drive (such as a USB flash drive, SD card, and so on). Turning on BitLocker for your operating system volume encrypts all system files on the volume, including the paging files and hibernation files. The only exception is for the System partition, which includes the Windows Boot Manager and minimal boot collateral required for decryption of the operating system volume after the key is unsealed.
- **Ensuring the integrity of early boot components and boot configuration data.** On devices that have a TPM version 1.2 or higher, BitLocker uses the enhanced security capabilities of the TPM to make data accessible only if the computer's BIOS firmware code and configuration, original boot sequence, boot components, and BCD configuration all appear unaltered and the encrypted disk is located in the original computer. On systems that leverage TPM PCR[7], BCD setting changes deemed safe are permitted to improve usability.

The next sections provide more details about how Windows protects against various attacks on the BitLocker encryption keys in Windows 11, Windows 10, Windows 8.1, and Windows 8.

For more information about how to enable the best overall security configuration for devices beginning with Windows 10 version 1803 or Windows 11, see [Standards for a highly secure Windows device](#).

Protection before startup

Before Windows starts, you must rely on security features implemented as part of the device hardware and firmware, including TPM and secure boot. Fortunately, many modern computers feature a TPM and secure boot.

Trusted Platform Module

A trusted platform module (TPM) is a microchip designed to provide basic security-related functions, primarily involving encryption keys. On some platforms, TPM can alternatively be implemented as a part of secure firmware. BitLocker binds encryption keys with the TPM to ensure that a computer hasn't been tampered with while the system was offline. For more info about TPM, see [Trusted Platform Module](#).

UEFI and secure boot

Unified Extensible Firmware Interface (UEFI) is a programmable boot environment that initializes devices and starts the operating system's bootloader.

The UEFI specification defines a firmware execution authentication process called [Secure Boot](#). Secure Boot blocks untrusted firmware and bootloaders (signed or unsigned) from being able to start on the system.

By default, BitLocker provides integrity protection for Secure Boot by utilizing the TPM PCR[7] measurement. An unauthorized EFI firmware, EFI boot application, or bootloader can't run and acquire the BitLocker key.

BitLocker and reset attacks

To defend against malicious reset attacks, BitLocker leverages the TCG Reset Attack Mitigation, also known as MOR bit (Memory Overwrite Request), before extracting keys into memory.

NOTE

This does not protect against physical attacks where an attacker opens the case and attacks the hardware.

Security policies

The next sections cover pre-boot authentication and DMA policies that can provide additional protection for BitLocker.

Pre-boot authentication

Pre-boot authentication with BitLocker is a policy setting that requires the use of either user input, such as a PIN, a startup key, or both to authenticate prior to making the contents of the system drive accessible. The Group Policy setting is [Require additional authentication at startup](#) and the corresponding setting in the [BitLocker CSP](#) is SystemDrivesRequireStartupAuthentication.

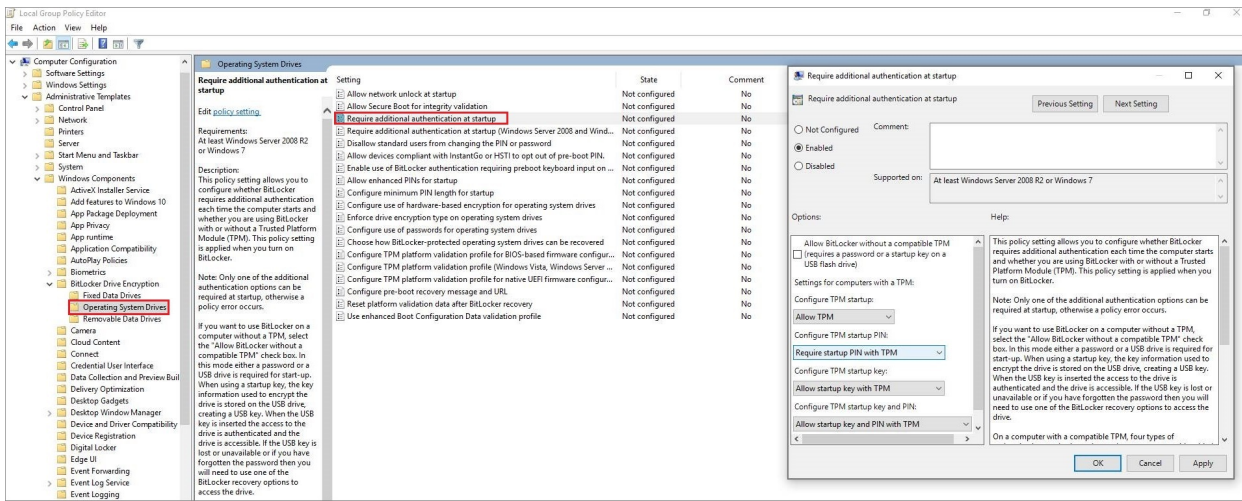
BitLocker accesses and stores the encryption keys in memory only after pre-boot authentication is completed. If Windows can't access the encryption keys, the device can't read or edit the files on the system drive. The only option for bypassing pre-boot authentication is entering the recovery key.

Pre-boot authentication is designed to prevent the encryption keys from being loaded to system memory without the trusted user supplying another authentication factor such as a PIN or startup key. This helps mitigate DMA and memory remanence attacks.

On computers with a compatible TPM, operating system drives that are BitLocker-protected can be unlocked in four ways:

- **TPM-only.** Using TPM-only validation doesn't require any interaction with the user to unlock and provide access to the drive. If the TPM validation succeeds, the user sign-in experience is the same as a standard sign in. If the TPM is missing or changed or if BitLocker detects changes to the BIOS or UEFI code or configuration, critical operating system startup files, or the boot configuration, BitLocker enters recovery mode, and the user must enter a recovery password to regain access to the data. This option is more convenient for sign-in but less secure than the other options, which require an additional authentication factor.
- **TPM with startup key.** In addition to the protection that the TPM-only provides, part of the encryption key is stored on a USB flash drive, referred to as a startup key. Data on the encrypted volume can't be accessed without the startup key.
- **TPM with PIN.** In addition to the protection that the TPM provides, BitLocker requires that the user enter a PIN. Data on the encrypted volume can't be accessed without entering the PIN. TPMs also have [anti-hammering protection](#) that is designed to prevent brute force attacks that attempt to determine the PIN.
- **TPM with startup key and PIN.** In addition to the core component protection that the TPM-only provides, part of the encryption key is stored on a USB flash drive, and a PIN is required to authenticate the user to the TPM. This configuration provides multifactor authentication so that if the USB key is lost or stolen, it can't be used for access to the drive, because the correct PIN is also required.

In the following group policy example, TPM + PIN is required to unlock an operating system drive:



Pre-boot authentication with a PIN can mitigate an attack vector for devices that use a bootable eDrive because an exposed eDrive bus can allow an attacker to capture the BitLocker encryption key during startup. Pre-boot authentication with a PIN can also mitigate DMA port attacks during the window of time between when BitLocker unlocks the drive and Windows boots to the point that Windows can set any port-related policies that have been configured.

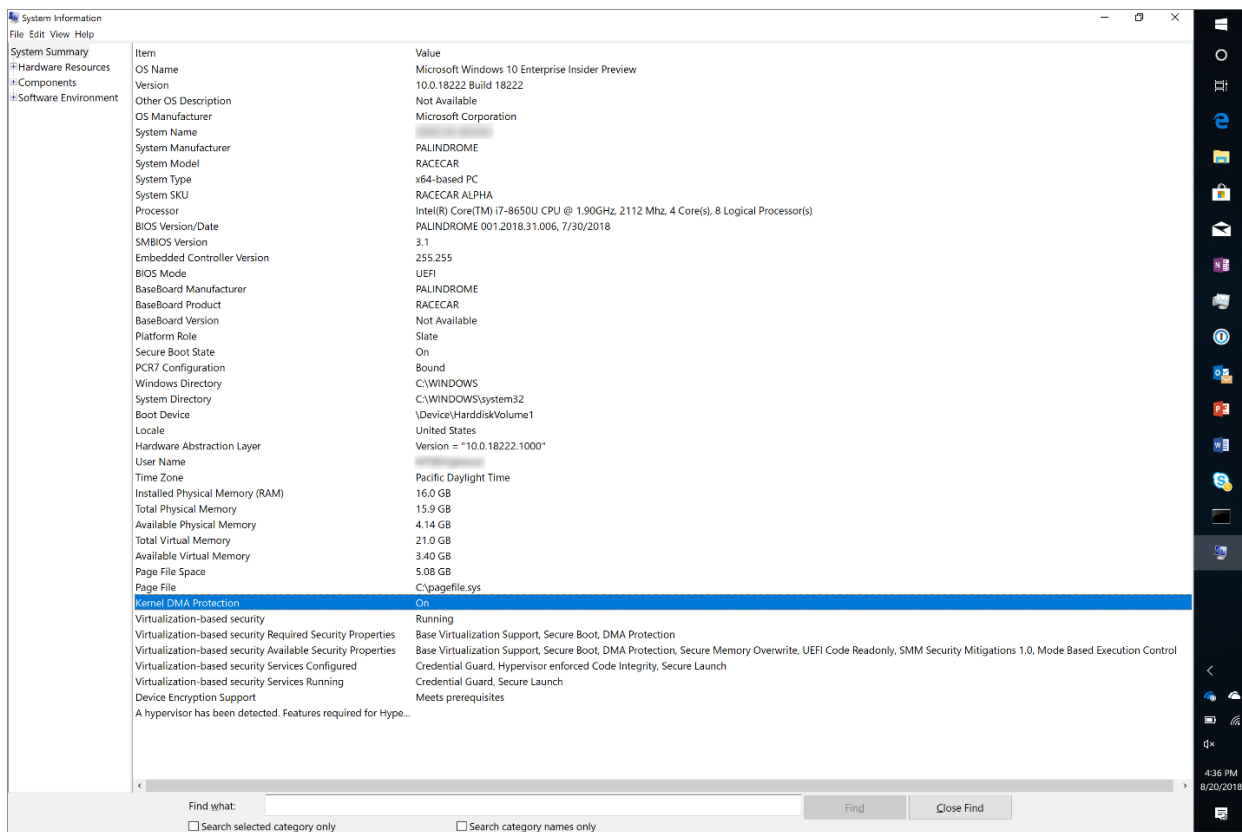
On the other hand, Pre-boot authentication-prompts can be inconvenient to users. In addition, users who forget their PIN or lose their startup key are denied access to their data until they can contact their organization's support team to obtain a recovery key. Pre-boot authentication can also make it more difficult to update unattended desktops and remotely administered servers because a PIN needs to be entered when a computer reboots or resumes from hibernation.

To address these issues, you can deploy [BitLocker Network Unlock](#). Network Unlock allows systems within the physical enterprise security perimeter that meet the hardware requirements and have BitLocker enabled with TPM+PIN to boot into Windows without user intervention. It requires direct ethernet connectivity to an enterprise Windows Deployment Services (WDS) server.

Protecting Thunderbolt and other DMA ports

There are a few different options to protect DMA ports, such as Thunderbolt™ 3. Beginning with Windows 10 version 1803 or Windows 11, new Intel-based devices have kernel protection against DMA attacks via Thunderbolt™ 3 ports enabled by default. This Kernel DMA Protection is available only for new systems beginning with Windows 10 version 1803 or Windows 11, as it requires changes in the system firmware and/or BIOS.

You can use the System Information desktop app (MSINFO32) to check if a device has kernel DMA protection enabled:



If kernel DMA protection is *not* enabled, follow these steps to protect Thunderbolt™ 3-enabled ports:

1. Require a password for BIOS changes
2. Intel Thunderbolt Security must be set to User Authorization in BIOS settings. Refer to [Intel Thunderbolt™ 3 and Security on Microsoft Windows® 10 Operating System documentation](#)
3. Additional DMA security may be added by deploying policy (beginning with Windows 10 version 1607 or Windows 11):
 - MDM: [DataProtection/AllowDirectMemoryAccess](#) policy
 - Group Policy: [Disable new DMA devices when this computer is locked](#) (This setting isn't configured by default.)

For Thunderbolt v1 and v2 (DisplayPort Connector), refer to the "Thunderbolt Mitigation" section in [KB 2516445](#). For SBP-2 and 1394 (a.k.a. Firewire), refer to the "SBP-2 Mitigation" section in [KB 2516445](#).

Attack countermeasures

This section covers countermeasures for specific types of attacks.

Bootkits and rootkits

A physically-present attacker might attempt to install a bootkit or rootkit-like piece of software into the boot chain in an attempt to steal the BitLocker keys. The TPM should observe this installation via PCR measurements, and the BitLocker key won't be released.

This is the default configuration.

A BIOS password is recommended for defense-in-depth in case a BIOS exposes settings that may weaken the BitLocker security promise. Intel Boot Guard and AMD Hardware Verified Boot support stronger implementations of Secure Boot that provide additional resilience against malware and physical attacks. Intel Boot Guard and AMD Hardware Verified Boot are part of platform boot verification [standards for a highly secure Windows device](#).

Brute force attacks against a PIN

Require TPM + PIN for anti-hammering protection.

DMA attacks

See [Protecting Thunderbolt and other DMA ports](#) earlier in this article.

Paging file, crash dump, and Hyberfil.sys attacks

These files are secured on an encrypted volume by default when BitLocker is enabled on OS drives. It also blocks automatic or manual attempts to move the paging file.

Memory remanence

Enable secure boot and mandatorily prompt a password to change BIOS settings. For customers requiring protection against these advanced attacks, configure a TPM+PIN protector, disable Standby power management, and shut down or hibernate the device before it leaves the control of an authorized user.

Attacker countermeasures

The following sections cover mitigations for different types of attackers.

Attacker without much skill or with limited physical access

Physical access may be limited by a form factor that doesn't expose buses and memory. For example, there are no external DMA-capable ports, no exposed screws to open the chassis, and memory is soldered to the mainboard. This attacker of opportunity doesn't use destructive methods or sophisticated forensics hardware/software.

Mitigation:

- Pre-boot authentication set to TPM only (the default)

Attacker with skill and lengthy physical access

Targeted attack with plenty of time; this attacker will open the case, will solder, and will use sophisticated hardware or software.

Mitigation:

- Pre-boot authentication set to TPM with a PIN protector (with a sophisticated alphanumeric PIN [enhanced pin] to help the TPM anti-hammering mitigation).
- And-
- Disable Standby power management and shut down or hibernate the device before it leaves the control of an authorized user. This can be set using Group Policy:
 - Computer Configuration\Policies\Administrative Templates\Windows Components\File Explorer>Show hibernate in the power options menu
 - Computer Configuration\Policies\Administrative Templates\System\Power Management\Sleep Settings\Allow standby states (S1-S3) when sleeping (plugged in)
 - Computer Configuration\Policies\Administrative Templates\System\Power Management\Sleep Settings\Allow standby states (S1-S3) when sleeping (on battery)

These settings are **Not configured** by default.

For some systems, bypassing TPM-only may require opening the case, and may require soldering, but could possibly be done for a reasonable cost. Bypassing a TPM with a PIN protector would cost much more, and require brute forcing the PIN. With a sophisticated enhanced PIN, it could be nearly impossible. The Group Policy setting for [enhanced PIN](#) is:

Computer Configuration\Administrative Templates\Windows Components\BitLocker Drive Encryption\Operating System Drives\Allow enhanced PINs for startup

This setting is **Not configured** by default.

For secure administrative workstations, Microsoft recommends a TPM with PIN protector and to disable Standby power management and shut down or hibernate the device.

See also

- [Blocking the SBP-2 driver and Thunderbolt controllers to reduce 1394 DMA and Thunderbolt DMA threats to BitLocker](#)
- [BitLocker Group Policy settings](#)
- [BitLocker CSP](#)
- [Winlogon automatic restart sign-on \(ARSO\)](#)

Protecting cluster shared volumes and storage area networks with BitLocker

7/1/2022 • 8 minutes to read • [Edit Online](#)

Applies to

- Windows Server 2016

This article for IT pros describes how to protect CSVs and SANs with BitLocker.

BitLocker can protect both physical disk resources and cluster shared volumes version 2.0 (CSV2.0). BitLocker on clustered volumes allows for an additional layer of protection for administrators wishing to protect sensitive, highly available data. By adding additional protectors to the clustered volume, administrators can also add an additional barrier of security to resources within an organization by allowing only certain user accounts access to unlock the BitLocker volume.

Configuring BitLocker on Cluster Shared Volumes

Using BitLocker with Clustered Volumes

BitLocker on volumes within a cluster are managed based on how the cluster service "views" the volume to be protected. The volume can be a physical disk resource such as a logical unit number (LUN) on a storage area network (SAN) or network attached storage (NAS).

IMPORTANT

SANs used with BitLocker must have obtained Windows Hardware Certification. For more info, see [Windows Hardware Lab Kit](#).

Alternatively, the volume can be a cluster-shared volume, a shared namespace, within the cluster. Windows Server 2012 expanded the CSV architecture, now known as CSV2.0, to enable support for BitLocker. When using BitLocker with volumes designated for a cluster, the volume will need to turn on BitLocker before its addition to the storage pool within cluster or put the resource into maintenance mode before BitLocker operations will complete.

Windows PowerShell or the manage-bde command-line interface is the preferred method to manage BitLocker on CSV2.0 volumes. This method is recommended over the BitLocker Control Panel item because CSV2.0 volumes are mount points. Mount points are an NTFS object that is used to provide an entry point to other volumes. Mount points do not require the use of a drive letter. Volumes that lack drive letters do not appear in the BitLocker Control Panel item. Additionally, the new Active Directory-based protector option required for cluster disk resource or CSV2.0 resources is not available in the Control Panel item.

NOTE

Mount points can be used to support remote mount points on SMB based network shares. This type of share is not supported for BitLocker encryption.

For thinly provisioned storage, such as a Dynamic Virtual Hard Disk (VHD), BitLocker runs in Used Disk Space Only encryption mode. You cannot use the **manage-bde -WipeFreeSpace** command to transition the volume to full-volume encryption on these types of volumes. This action is blocked in order to avoid expanding thinly

provisioned volumes to occupy the entire backing store while wiping the unoccupied (free) space.

Active Directory-based protector

You can also use an Active Directory Domain Services (AD DS) protector for protecting clustered volumes held within your AD DS infrastructure. The **ADAccountOrGroup** protector is a domain security identifier (SID)-based protector that can be bound to a user account, machine account, or group. When an unlock request is made for a protected volume, the BitLocker service interrupts the request and uses the BitLocker protect/unprotect APIs to unlock or deny the request. BitLocker will unlock protected volumes without user intervention by attempting protectors in the following order:

1. Clear key
2. Driver-based auto-unlock key
3. ADAccountOrGroup protector
 - a. Service context protector
 - b. User protector
4. Registry-based auto-unlock key

NOTE

A Windows Server 2012 or later domain controller is required for this feature to work properly.

Turning on BitLocker before adding disks to a cluster using Windows PowerShell

BitLocker encryption is available for disks before or after addition to a cluster storage pool. The advantage of encrypting volumes prior to adding them to a cluster is that the disk resource does not require suspending the resource to complete the operation. To turn on BitLocker for a disk before adding it to a cluster:

1. Install the BitLocker Drive Encryption feature if it is not already installed.
2. Ensure the disk is formatted NTFS and has a drive letter assigned to it.
3. Identify the name of the cluster with Windows PowerShell.

```
Get-Cluster
```

4. Enable BitLocker on the volume of your choice with an **ADAccountOrGroup** protector, using the cluster name. For example, use a command such as:

```
Enable-BitLocker E: -ADAccountOrGroupProtector -ADAccountOrGroup CLUSTER$
```

WARNING

You must configure an **ADAccountOrGroup** protector using the cluster CNO for a BitLocker enabled volume to either be shared in a Cluster Shared Volume or to fail over properly in a traditional failover cluster.

5. Repeat the preceding steps for each disk in the cluster.
6. Add the volume(s) to the cluster.

Turning on BitLocker for a clustered disk using Windows PowerShell

When the cluster service owns a disk resource already, it needs to be set into maintenance mode before BitLocker can be enabled. Use the following steps for turning on BitLocker for a clustered disk:

1. Install the BitLocker Drive Encryption feature if it is not already installed.
2. Check the status of the cluster disk using Windows PowerShell.

```
Get-ClusterResource "Cluster Disk 1"
```

3. Put the physical disk resource into maintenance mode using Windows PowerShell.

```
Get-ClusterResource "Cluster Disk 1" | Suspend-ClusterResource
```

4. Identify the name of the cluster with Windows PowerShell.

```
Get-Cluster
```

5. Enable BitLocker on the volume of your choice with an **ADAccountOrGroup** protector, using the cluster name. For example, use a command such as:

```
Enable-BitLocker E: -ADAccountOrGroupProtector -ADAccountOrGroup CLUSTER$
```

WARNING

You must configure an **ADAccountOrGroup** protector using the cluster CNO for a BitLocker enabled volume to either be shared in a Cluster Shared Volume or to fail over properly in a traditional failover cluster.

6. Use **Resume-ClusterResource** to take the physical disk resource back out of maintenance mode:

```
Get-ClusterResource "Cluster Disk 1" | Resume-ClusterResource
```

7. Repeat the preceding steps for each disk in the cluster.

Adding BitLocker encrypted volumes to a cluster using manage-bde

You can also use `manage-bde` to enable BitLocker on clustered volumes. Follow these steps to add a physical disk resource or CSV2.0 volume to an existing cluster:

1. Verify the BitLocker Drive Encryption feature is installed on the computer.
2. Ensure new storage is formatted as NTFS.
3. Encrypt the volume, add a recovery key, and add the cluster administrator as a protector key by using the `manage-bde` command-line interface (see example):

- `Manage-bde -on -used <drive letter> -RP -sid domain\CNO$ -sync`

- a. BitLocker will check to see if the disk is already part of a cluster. If it is, administrators will encounter a hard block. Otherwise, the encryption will continue.
 - b. Using the `-sync` parameter is optional. Using it ensures the command waits until the encryption for the volume is completed before releasing the volume for use in the cluster storage pool.
4. Open the Failover Cluster Manager snap-in or cluster PowerShell cmdlets to enable the disk to be clustered
 - Once the disk is clustered, it can also be enabled for CSV.
 5. During the resource online operation, cluster will check to see if the disk is BitLocker encrypted.

- a. If the volume is not BitLocker enabled, traditional cluster online operations occur.
 - b. If the volume is BitLocker enabled, the following check occurs:
 - If volume is **locked**, BitLocker will impersonate the CNO and unlock the volume using the CNO protector. If this operation fails, an event will be logged that the volume could not be unlocked and the online operation will fail.
6. Once the disk is online in the storage pool, it can be added to a CSV by right-clicking the disk resource and choosing **Add to cluster shared volumes**.

CSVs can include both encrypted and unencrypted volumes. To check the status of a particular volume for BitLocker encryption, administrators can utilize the `manage-bde -status` command with a path to the volume inside the CSV namespace as seen in the example command line below.

```
manage-bde -status "C:\ClusterStorage\volume1"
```

Physical Disk Resources

Unlike CSV2.0 volumes, physical disk resources can only be accessed by one cluster node at a time. So operations such as encrypting, decrypting, locking, or unlocking volumes require context to perform. For example, you cannot unlock or decrypt a physical disk resource if you are not administering the cluster node that owns the disk resource because the disk resource is not available.

Restrictions on BitLocker actions with cluster volumes

The following table contains information about both Physical Disk Resources (that is, traditional failover cluster volumes) and Cluster Shared Volumes (CSV) and the actions that are allowed by BitLocker in each situation.

ACTION	ON OWNER NODE OF FAILOVER VOLUME	ON METADATA SERVER (MDS) OF CSV	ON (DATA SERVER) DS OF CSV	MAINTENANCE MODE
<code>Manage-bde -on</code>	Blocked	Blocked	Blocked	Allowed
<code>Manage-bde -off</code>	Blocked	Blocked	Blocked	Allowed
<code>Manage-bde Pause/Resume</code>	Blocked	Blocked**	Blocked	Allowed
<code>Manage-bde -lock</code>	Blocked	Blocked	Blocked	Allowed
<code>manage-bde -wipe</code>	Blocked	Blocked	Blocked	Allowed
Unlock	Automatic via cluster service	Automatic via cluster service	Automatic via cluster service	Allowed
<code>manage-bde -protector -add</code>	Allowed	Allowed	Blocked	Allowed
<code>manage-bde -protector -delete</code>	Allowed	Allowed	Blocked	Allowed
<code>manage-bde -autounlock</code>	Allowed (not recommended)	Allowed (not recommended)	Blocked	Allowed (not recommended)

ACTION	ON OWNER NODE OF FAILOVER VOLUME	ON METADATA SERVER (MDS) OF CSV	ON (DATA SERVER) DS OF CSV	MAINTENANCE MODE
Manage-bde - upgrade	Allowed	Allowed	Blocked	Allowed
Shrink	Allowed	Allowed	Blocked	Allowed
Extend	Allowed	Allowed	Blocked	Allowed

NOTE

Although the `manage-bde -pause` command is Blocked in clusters, the cluster service will automatically resume a paused encryption or decryption from the MDS node

In the case where a physical disk resource experiences a failover event during conversion, the new owning node will detect the conversion is not complete and will complete the conversion process.

Other considerations when using BitLocker on CSV2.0

Also take these considerations into account for BitLocker on clustered storage:

- BitLocker volumes have to be initialized and beginning encryption before they are available to add to a CSV2.0 volume.
- If an administrator needs to decrypt a CSV volume, remove the volume from the cluster or put into disk maintenance mode. You can add the CSV back to the cluster while waiting for decryption to complete.
- If an administrator needs to start encrypting a CSV volume, remove the volume from the cluster or put it in maintenance mode.
- If conversion is paused with encryption in progress and the CSV volume is offline from the cluster, the cluster thread (health check) will automatically resume conversion when the volume is online to the cluster.
- If conversion is paused with encryption in progress and a physical disk resource volume is offline from the cluster, the BitLocker driver will automatically resume conversion when the volume is online to the cluster.
- If conversion is paused with encryption in progress, while the CSV volume is in maintenance mode, the cluster thread (health check) will automatically resume conversion when moving the volume back from maintenance.
- If conversion is paused with encryption in progress, while the disk resource volume is in maintenance mode, the BitLocker driver will automatically resume conversion when the volume is moved back from maintenance mode.

Guidelines for troubleshooting BitLocker

7/1/2022 • 4 minutes to read • [Edit Online](#)

This article addresses common issues in BitLocker and provides guidelines to troubleshoot these issues. This article also provides information such as what data to collect and what settings to check. This information makes your troubleshooting process much easier.

Review the event logs

Open Event Viewer and review the following logs under Applications and Services logs\Microsoft\Windows:

- **BitLocker-API**. Review the management log, the operational log, and any other logs that are generated in this folder. The default logs have the following unique names:
 - Microsoft-Windows-BitLocker-API/BitLocker Operational
 - Microsoft-Windows-BitLocker-API/BitLocker Management
- **BitLocker-DrivePreparationTool**. Review the admin log, the operational log, and any other logs that are generated in this folder. The default logs have the following unique names:
 - Microsoft-Windows-BitLocker-DrivePreparationTool/Operational
 - Microsoft-Windows-BitLocker-DrivePreparationTool/Admin

Additionally, review the Windows logs\System log for events that were produced by the TPM and TPM-WMI event sources.

To filter and display or export logs, you can use the [wevtutil.exe](#) command-line tool or the [Get-WinEvent](#) cmdlet.

For example, to use wevtutil to export the contents of the operational log from the BitLocker-API folder to a text file that is named BitLockerAPIOpsLog.txt, open a Command Prompt window, and run the following command:

```
wevtutil qe "Microsoft-Windows-BitLocker/BitLocker Operational" /f:text > BitLockerAPIOpsLog.txt
```

To use the **Get-WinEvent** cmdlet to export the same log to a comma-separated text file, open a Windows Powershell window and run the following command:

```
Get-WinEvent -logname "Microsoft-Windows-BitLocker/BitLocker Operational" | Export-Csv -Path Bitlocker-Operational.csv
```

You can use Get-WinEvent in an elevated PowerShell window to display filtered information from the system or application log by using the following syntax:

- To display BitLocker-related information:

```
Get-WinEvent -FilterHashtable @{LogName='System'} | Where-Object -Property Message -Match 'BitLocker'  
| fl
```

The output of such a command resembles the following.


```
PS C:\> Get-WinEvent -FilterHashtable @{LogName='System'} | Where-Object -Property Message -Match 'BitLocker' | fl
TimeCreated      : 9/22/2019 8:44:00 PM
ProviderName     : Microsoft-Windows-BitLocker-Driver
Id               : 24667
Message          : BitLocker finalization sweep completed for volume H:.
TimeCreated      : 9/22/2019 8:44:00 PM
ProviderName     : Microsoft-Windows-BitLocker-Driver
Id               : 24665
Message          : BitLocker finalization sweep paused for volume H:.
```

- To export BitLocker-related information:

```
Get-WinEvent -FilterHashtable @{LogName='System'} | Where-Object -Property Message -Match 'BitLocker'
| Export-Csv -Path System-BitLocker.csv
```

- To display TPM-related information:

```
Get-WinEvent -FilterHashtable @{LogName='System'} | Where-Object -Property Message -Match 'TPM' | fl
```

- To export TPM-related information:

```
Get-WinEvent -FilterHashtable @{LogName='System'} | Where-Object -Property Message -Match 'TPM' |
Export-Csv -Path System-TPM.csv
```

The output of such a command resembles the following.

```
PS C:\> Get-WinEvent -FilterHashtable @{LogName='System'} | Where-Object -Property Message -Match 'TPM' | fl
TimeCreated      : 10/12/2019 3:17:47 PM
ProviderName     : Microsoft-Windows-TPM-WMI
Id               : 1025
Message          : The TPM was successfully provisioned and is now ready for use.
TimeCreated      : 10/12/2019 3:17:44 PM
ProviderName     : Microsoft-Windows-TPM-WMI
Id               : 1025
Message          : The TPM was successfully provisioned and is now ready for use.
```

NOTE

If you intend to contact Microsoft Support, we recommend that you export the logs listed in this section.

Gather status information from the BitLocker technologies

Open an elevated Windows PowerShell window, and run each of the following commands.

COMMAND	NOTES
<code>get-tpm > C:\TPM.txt</code>	Exports information about the local computer's Trusted Platform Module (TPM). This cmdlet shows different values depending on whether the TPM chip is version 1.2 or 2.0. This cmdlet is not supported in Windows 7.
<code>manage-bde -status > C:\BDEStatus.txt</code>	Exports information about the general encryption status of all drives on the computer.
<code>manage-bde c: -protectors -get > C:\Protectors</code>	Exports information about the protection methods that are used for the BitLocker encryption key.

COMMAND	NOTES
<code>reagentc /info > C:\reagent.txt</code>	Exports information about an online or offline image about the current status of the Windows Recovery Environment (WindowsRE) and any available recovery image.
<code>get-BitLockerVolume fl</code>	Gets information about volumes that BitLocker Drive Encryption can protect.

Review the configuration information

1. Open an elevated Command Prompt window, and run the following commands.

COMMAND	NOTES
<code>gpresult /h <Filename></code>	Exports the Resultant Set of Policy information, and saves the information as an HTML file.
<code>msinfo /report <Path> /computer <ComputerName></code>	Exports comprehensive information about the hardware, system components, and software environment on the local computer. The <code>/report</code> option saves the information as a .txt file.

2. Open Registry Editor, and export the entries in the following subkeys:

- HKLM\SOFTWARE\Policies\Microsoft\FVE
- HKLM\SYSTEM\CurrentControlSet\Services\TPM\

Check the BitLocker prerequisites

Common settings that can cause issues for BitLocker include the following scenarios:

- The TPM must be unlocked. You can check the output of the `get-tpm` command for the status of the TPM.
- Windows RE must be enabled. You can check the output of the `reagentc` command for the status of WindowsRE.
- The system-reserved partition must use the correct format.
 - On Unified Extensible Firmware Interface (UEFI) computers, the system-reserved partition must be formatted as FAT32.
 - On legacy computers, the system-reserved partition must be formatted as NTFS.
- If the device that you are troubleshooting is a slate or tablet PC, use <https://gpsearch.azurewebsites.net/#8153> to verify the status of the **Enable use of BitLocker authentication requiring preboot keyboard input on slates** option.

For more information about the BitLocker prerequisites, see [BitLocker basic deployment: Using BitLocker to encrypt volumes](#)

Next steps

If the information that you have examined so far indicates a specific issue (for example, WindowsRE is not enabled), the issue may have a straightforward fix.

Resolving issues that do not have obvious causes depends on exactly which components are involved and what

behavior you see. The information that you have gathered helps you narrow down the areas to investigate.

- If you are working on a device that is managed by Microsoft Intune, see [Enforcing BitLocker policies by using Intune: known issues](#).
- If BitLocker does not start or cannot encrypt a drive and you notice errors or events that are related to the TPM, see [BitLocker cannot encrypt a drive: known TPM issues](#).
- If BitLocker does not start or cannot encrypt a drive, see [BitLocker cannot encrypt a drive: known issues](#).
- If BitLocker Network Unlock does not behave as expected, see [BitLocker Network Unlock: known issues](#).
- If BitLocker does not behave as expected when you recover an encrypted drive, or if you did not expect BitLocker to recover the drive, see [BitLocker recovery: known issues](#).
- If BitLocker or the encrypted drive does not behave as expected, and you notice errors or events that are related to the TPM, see [BitLocker and TPM: other known issues](#).
- If BitLocker or the encrypted drive does not behave as expected, see [BitLocker configuration: known issues](#).

We recommend that you keep the information that you have gathered handy in case you decide to contact Microsoft Support for help to resolve your issue.

BitLocker cannot encrypt a drive: known issues

7/1/2022 • 2 minutes to read • [Edit Online](#)

This article describes common issues that prevent BitLocker from encrypting a drive. This article also provides guidance to address these issues.

NOTE

If you have determined that your BitLocker issue involves the trusted platform module (TPM), see [BitLocker cannot encrypt a drive: known TPM issues](#).

Error 0x80310059: BitLocker drive encryption is already performing an operation on this drive

When you turn on BitLocker Drive Encryption on a computer that is running Windows 10 Professional or Windows 11, you receive a message that resembles the following:

ERROR: An error occurred (code 0x80310059):BitLocker Drive Encryption is already performing an operation on this drive. Please complete all operations before continuing.**NOTE:** If the -on switch has failed to add key protectors or start encryption,you may need to call manage-bde -off before attempting -on again.

Cause

This issue may be caused by settings that are controlled by group policy objects (GPOs).

Resolution

IMPORTANT

Follow the steps in this section carefully. Serious problems might occur if you modify the registry incorrectly. Before you modify it, [back up the registry for restoration](#) in case problems occur.

To resolve this issue, follow these steps:

1. Start Registry Editor, and navigate to the following subkey:

HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\FVE

2. Delete the following entries:

- OSPlatformValidation_BIOS
- OSPlatformValidation_UEFI
- PlatformValidation

3. Exit registry editor, and turn on BitLocker drive encryption again.

"Access is denied" message when you try to encrypt removable drives

You have a computer that is running Windows 10, version 1709 or version 1607, or Windows 11. You try to encrypt a USB drive by following these steps:

1. In Windows Explorer, right-click the USB drive and select **Turn on BitLocker**.
2. On the **Choose how you want to unlock this drive** page, select **Use a password to unlock the drive**.
3. Follow the instructions on the page to enter your password.
4. On the **Are you ready to encrypt this drive?** page, select **Start encrypting**.
5. The **Starting encryption** page displays the message "Access is denied."

You receive this message on any computer that runs Windows 10 version 1709 or version 1607, or Windows 11, when you use any USB drive.

Cause

The security descriptor of the BitLocker drive encryption service (BDESvc) has an incorrect entry. Instead of NT AUTHORITY\Authenticated Users, the security descriptor uses NT AUTHORITY\INTERACTIVE.

To verify that this issue has occurred, follow these steps:

1. On an affected computer, open an elevated Command Prompt window and an elevated PowerShell window.
2. At the command prompt, enter the following command:

```
C:\>sc sdshow bdesvc
```

The output of this command resembles the following:

```
D:(A;;CCDCLCSWRPWPDTLORCWDWO;;;SY)(A;;CCDCLCSWRPWPDTLORCWDWO;;;BA)(A;;CCLCSWRPLORC;;;BU)(A;;CCLCSWRPLORC;;;AU)S:(AU;FA;CCDCLCSWRPWPDTLOSDRCWDWO;;;WD)
```

3. Copy this output, and use it as part of the **ConvertFrom-SddlString** command in the PowerShell window, as follows.

```
PS C:\WINDOWS\system32> ConvertFrom-SddlString -Sddl "D:(A;;CCDCLCSWRPWPDTLOCRSDRCWDWO;;;BA)(A;;CCDCLCSWRPWPDTLOCRSDRCWDWO;;;SY)(A;;CCLCSWLOCRRC;;;IU)S:(AU;FA;CCDCLCSWRPWPDTLOCRSDRCWDWO;;;WD)" -Type ActiveDirectoryRights

Owner           :
Group           :
DiscretionaryAcl : {
    NT AUTHORITY\INTERACTIVE :      AccessAllowed (CreateChild, ExtendedRight, GenericExecute, ListChildren, ListObject, Read, ReadControl, Self),
    NT AUTHORITY\SYSTEM:      AccessAllowed (CreateChild, Delete, DeleteChild, DeleteTree, ExecuteKey, ExtendedRight, FullControl, GenericAll, GenericExecute, GenericRead, GenericWrite, ListChildren, ListObject, Modify, Read, ReadAndExecute, ReadControl, ReadProperty, Self, Write, WriteDacl, WriteKey, WriteOwner, WriteProperty),
    BUILTIN\Administrators:   AccessAllowed (CreateChild, Delete, DeleteChild, DeleteTree, ExecuteKey, ExtendedRight, FullControl, GenericAll, GenericExecute, GenericRead, GenericWrite, ListChildren, ListObject, Modify, Read, ReadAndExecute, ReadControl, ReadProperty, Self, Write, WriteDacl, WriteKey, WriteOwner, WriteProperty)}

SystemAcl       : {Everyone: SystemAudit FailedAccess (CreateChild, Delete, DeleteChild, DeleteTree, ExecuteKey, ExtendedRight, FullControl, GenericAll, GenericExecute, GenericRead, GenericWrite, ListChildren, ListObject, Modify, Read, ReadAndExecute, ReadControl, ReadProperty, Self, Write, WriteDacl, WriteKey, WriteOwner, WriteProperty)}
RawDescriptor   : System.Security.AccessControl.CommonSecurityDescriptor
```

If you see NT AUTHORITY\INTERACTIVE (as highlighted) in the output of this command, this is the cause of the issue. Under typical conditions, the output should resemble the following:

```

PS C:\WINDOWS\system32> ConvertFrom-SddlString -Sddl "D:(A;;CCDCLCSWRPWPDTLORCWDWO;;;SY)
(A;;CCDCLCSWRPWPDTLORCWDWO;;;BA)(A;;CCLCSWRPLORC;;;BU)(A;;CCLCSWRPLORC;;;AU)S:
(AU;FA;CCDCLCSWRPWPDTLORCWDWO;;;WD)" -Type ActiveDirectoryRights

Owner          :
Group          :
DiscretionaryAcl : {
    NT AUTHORITY\Authenticated Users : AccessAllowed (CreateChild, ExecuteKey,
    GenericExecute, GenericRead, ListChildren, ListObject, Read, ReadControl, ReadProperty,
    Self),
    NT AUTHORITY\SYSTEM:              AccessAllowed (CreateChild, DeleteChild,
    DeleteTree, ExecuteKey, GenericExecute, GenericRead, GenericWrite, ListChildren, ListObject,
    Read, ReadAndExecute, ReadControl, ReadProperty, Self, WriteDacl, WriteKey, WriteOwner,
    WriteProperty),
    BUILTIN\Administrators:           AccessAllowed (CreateChild, DeleteChild,
    DeleteTree, ExecuteKey, GenericExecute, GenericRead, GenericWrite, ListChildren, ListObject,
    Read, ReadAndExecute, ReadControl, ReadProperty, Self, WriteDacl, WriteKey, WriteOwner,
    WriteProperty), BUILTIN\Users: AccessAllowed (CreateChild, ExecuteKey, GenericExecute,
    GenericRead, ListChildren, ListObject, Read, ReadControl, ReadProperty, Self)}

SystemAcl      : {Everyone: SystemAudit FailedAccess (CreateChild, Delete, DeleteChild,
    DeleteTree, ExecuteKey, FullControl, GenericExecute, GenericRead, GenericWrite,
    ListChildren, ListObject, Read, ReadAndExecute, ReadControl, ReadProperty, Self, WriteDacl,
    WriteKey, WriteOwner, WriteProperty)}
RawDescriptor  : System.Security.AccessControl.CommonSecurityDescriptor

```

NOTE

GPOs that change the security descriptors of services have been known to cause this issue.

Resolution

1. To repair the security descriptor of BDESvc, open an elevated PowerShell window and enter the following command:

```

sc sdset bdesvc D:(A;;CCDCLCSWRPWPDTLORCWDWO;;;SY)(A;;CCDCLCSWRPWPDTLORCWDWO;;;BA)
(A;;CCLCSWRPLORC;;;BU)(A;;CCLCSWRPLORC;;;AU)S:(AU;FA;CCDCLCSWRPWPDTLORCWDWO;;;WD)

```

2. Restart the computer.

The issue should now be resolved.

Enforcing BitLocker policies by using Intune: known issues

7/1/2022 • 11 minutes to read • [Edit Online](#)

This article helps you troubleshoot issues that you may experience if you use Microsoft Intune policy to manage silent BitLocker encryption on devices. The Intune portal indicates whether BitLocker has failed to encrypt one or more managed devices.

BitLockerFixedDrivesRecoveryOptions	✔ Succeeded	
Warning for other disk encryption	✔ Succeeded	
Encrypt devices	❌ Error	-2016281112 (Remediation failed)
Allow standard users to enable encryption during Azure A...	✔ Succeeded	
BitLockerFixedDrivesRequireEncryption	✔ Succeeded	

To start narrowing down the cause of the problem, review the event logs as described in [Troubleshoot BitLocker](#). Concentrate on the Management and Operations logs in the **Applications and Services logs\Microsoft\Windows\BitLocker-API** folder. The following sections provide more information about how to resolve the indicated events and error messages:

- [Event ID 853: Error: A compatible Trusted Platform Module \(TPM\) Security Device cannot be found on this computer](#)
- [Event ID 853: Error: BitLocker Drive Encryption detected bootable media \(CD or DVD\) in the computer](#)
- [Event ID 854: WinRE is not configured](#)
- [Event ID 851: Contact manufacturer for BIOS upgrade](#)
- [Error message: The UEFI variable 'SecureBoot' could not be read](#)
- [Event ID 846, 778, and 851: Error 0x80072f9a](#)
- [Error message: Conflicting Group Policy settings for recovery options on operating system drives](#)

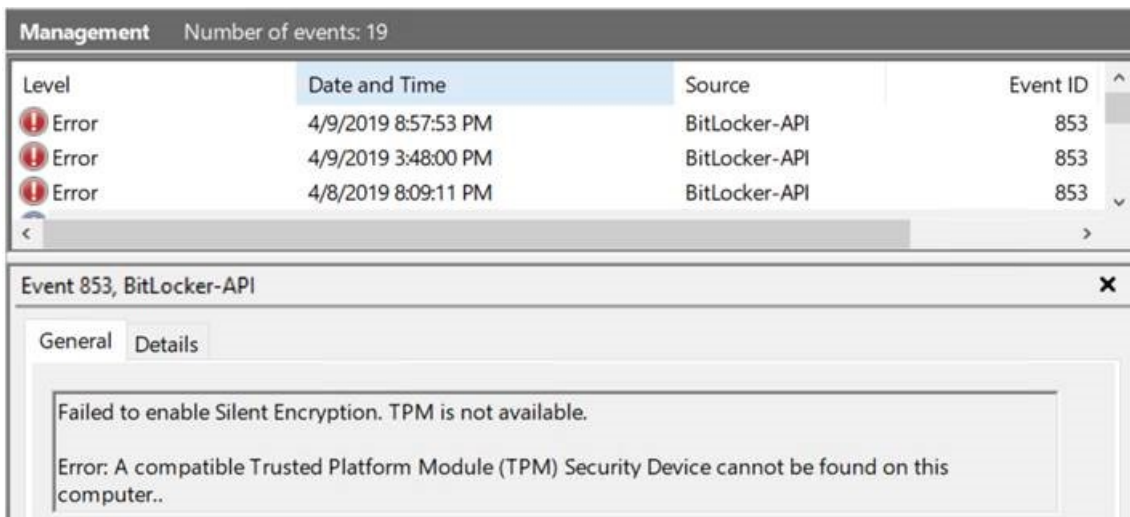
If you do not have a clear trail of events or error messages to follow, other areas to investigate include the following:

- [Review the hardware requirements for using Intune to manage BitLocker on devices](#)
- [Review your BitLocker policy configuration](#)

For information about the procedure to verify whether Intune policies are enforcing BitLocker correctly, see [Verifying that BitLocker is operating correctly](#).

Event ID 853: Error: A compatible Trusted Platform Module (TPM) Security Device cannot be found on this computer

Event ID 853 can carry different error messages, depending on the context. In this case, the Event ID 853 error message indicates that the device does not appear to have a TPM. The event information resembles the following:



Cause

The device that you are trying to secure may not have a TPM chip, or the device BIOS might have been configured to disable the TPM.

Resolution

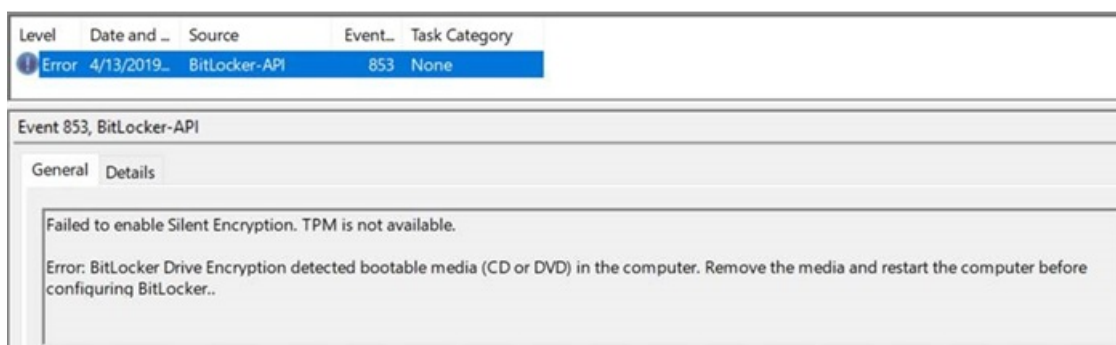
To resolve this issue, verify the following:

- The TPM is enabled in the device BIOS.
- The TPM status in the TPM management console resembles the following:
 - Ready (TPM 2.0)
 - Initialized (TPM 1.2)

For more information, see [Troubleshoot the TPM](#).

Event ID 853: Error: BitLocker Drive Encryption detected bootable media (CD or DVD) in the computer

In this case, you see event ID 853, and the error message in the event indicates that bootable media is available to the device. The event information resembles the following.



Cause

During the provisioning process, BitLocker drive encryption records the configuration of the device to establish a baseline. If the device configuration changes later (for example, if you remove the media), BitLocker recovery mode automatically starts.

To avoid this situation, the provisioning process stops if it detects a removable bootable media.

Resolution

Remove the bootable media, and restart the device. After the device restarts, verify the encryption status.

Event ID 854: WinRE is not configured

The event information resembles the following:

Failed to enable Silent Encryption. WinRe is not configured.

Error: This PC cannot support device encryption because WinRE is not properly configured.

Cause

Windows Recovery Environment (WinRE) is a minimal Windows operating system that is based on Windows Preinstallation Environment (Windows PE). WinRE includes several tools that an administrator can use to recover or reset Windows and diagnose Windows issues. If a device cannot start the regular Windows operating system, the device tries to start WinRE.

The provisioning process enables BitLocker drive encryption on the operating system drive during the Windows PE phase of provisioning. This action makes sure that the drive is protected before the full operating system is installed. The provisioning process also creates a system partition for WinRE to use if the system crashes.

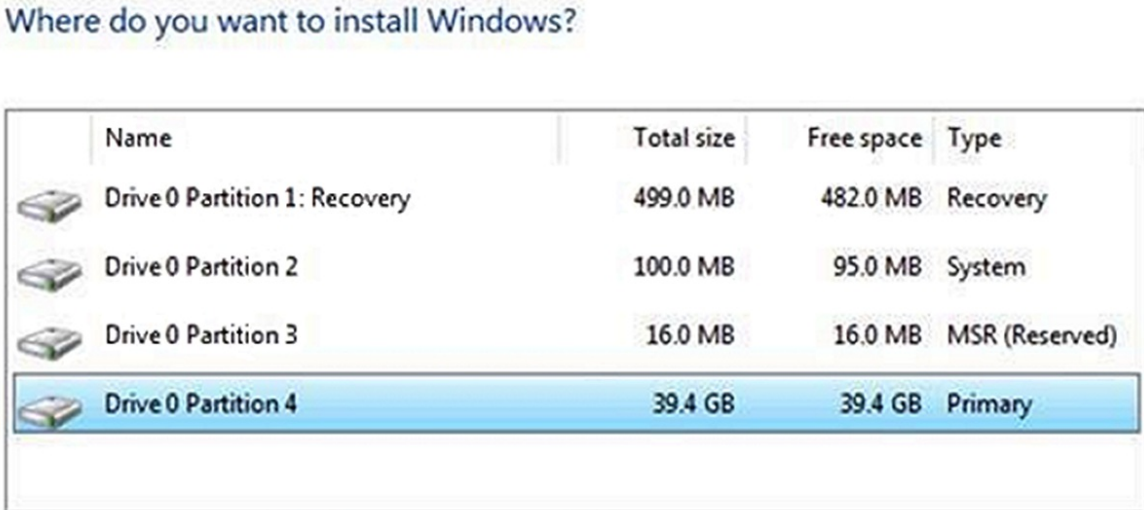
If WinRE is not available on the device, provisioning stops.

Resolution

You can resolve this issue by verifying the configuration of the disk partitions, the status of WinRE, and the Windows Boot Loader configuration. To do this, follow these steps.

Step 1: Verify the configuration of the disk partitions

The procedures described in this section depend on the default disk partitions that Windows configures during installation. Windows 11 and Windows 10 automatically create a recovery partition that contains the Winre.wim file. The partition configuration resembles the following.



The screenshot shows the 'Where do you want to install Windows?' window. It displays a table of disk partitions for Drive 0. The table has columns for Name, Total size, Free space, and Type. The partitions are:

Name	Total size	Free space	Type
Drive 0 Partition 1: Recovery	499.0 MB	482.0 MB	Recovery
Drive 0 Partition 2	100.0 MB	95.0 MB	System
Drive 0 Partition 3	16.0 MB	16.0 MB	MSR (Reserved)
Drive 0 Partition 4	39.4 GB	39.4 GB	Primary

Below the table are several action buttons: Refresh, Delete, Format, New, Load driver, and Extend.

To verify the configuration of the disk partitions, open an elevated Command Prompt window and run the following commands:

```
diskpart
list volume
```

```
C:\WINDOWS\system32>diskpart

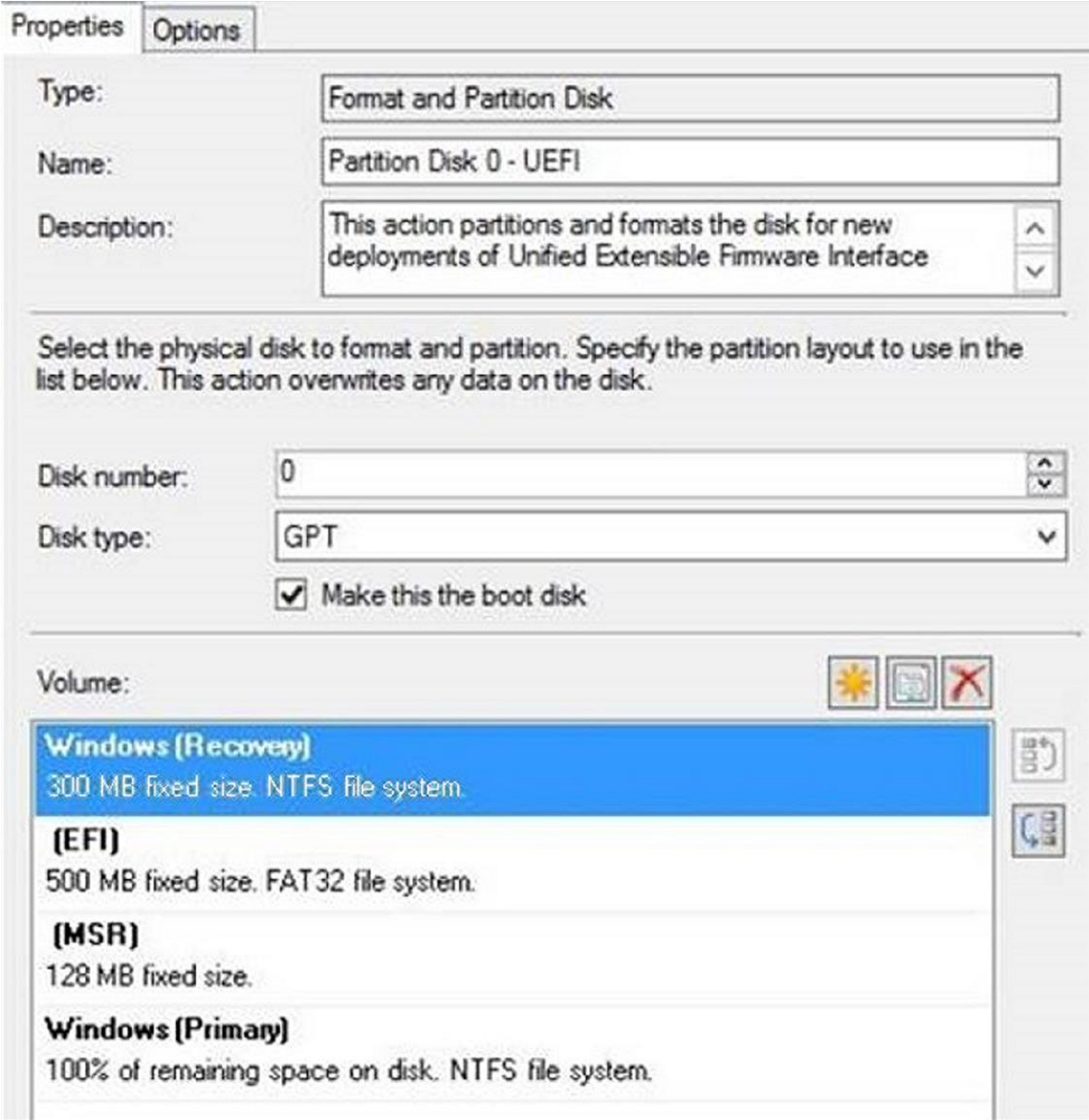
Microsoft DiskPart version 10.0.17763.1

Copyright (C) Microsoft Corporation.
On computer: MININT-8L1MASA

DISKPART> list vol

Volume ### Ltr Label          Fs          Type          Size         Status       Info
-----
Volume 0    E             DVD-ROM      0 B          No Media
Volume 1    System        NTFS         Partition     499 MB       Healthy      System
Volume 2    C    OSDisk        NTFS         Partition     216 GB       Healthy      Boot
Volume 3    D    New Volume    NTFS         Partition     248 GB       Healthy
```

If the status of any of the volumes is not healthy or if the recovery partition is missing, you may have to reinstall Windows. Before you do this, check the configuration of the Windows image that you are using for provisioning. Make sure that the image uses the correct disk configuration. The image configuration should resemble the following (this example is from Microsoft Endpoint Configuration Manager):



Step 2: Verify the status of WinRE

To verify the status of WinRE on the device, open an elevated Command Prompt window and run the following command:

```
reagentc /info
```

The output of this command resembles the following.

```
C:\>reagentc /info
Windows Recovery Environment (Windows RE) and system reset configuration
Information:

    Windows RE status:          Enabled
    Windows RE location:        \\?\GLOBALROOT\device\harddisk0\partition3\Recovery\WindowsRE
    Boot Configuration Data (BCD) identifier: 795594e5-c7d0-11e8-b0ac-af33c6ecd1ec
    Recovery image location:
    Recovery image index:       0
    Custom image location:
    Custom image index:         0

REAGENTC.EXE: Operation Successful.
```

If the Windows RE status is not Enabled, run the following command to enable it:

```
reagentc /enable
```

Step 3: Verify the Windows Boot Loader configuration

If the partition status is healthy, but the `reagentc /enable` command results in an error, verify whether the Windows Boot Loader contains the recovery sequence GUID. To do this, run the following command in an elevated Command Prompt window:

```
bcdedit /enum all
```

The output of this command resembles the following:

```
C:\WINDOWS\system32>bcdedit /enum all

Windows Boot Manager
-----
identifier           {bootmgr}
device               partition=\Device\HarddiskVolume1
description           Windows Boot Manager
locale               en-US
inherit              {globalsettings}
fvrecoverymessage    To retrieve this key, go to https://aka.ms/bitlockerrecovery from another device with access to
the corporate network, or to https://bitlockerrecovery.microsoft.com for registered, non-domain joined machines.
default              {current}
resumeobject         {4b2bb885-8418-11e8-b279-001f29044af6}
displayorder         {current}
toolsdisplayorder    {memdiag}
timeout              30

Windows Boot Loader
-----
identifier           {current}
device               partition=C:
path                 \WINDOWS\system32\winload.exe
description           Windows 10
locale               en-US
inherit              {bootloadersettings}
recoverysequence     {795594e5-c7d0-11e8-b0ac-af33c6ecd1ec}
displaymessageoverride Recovery
recoveryenabled       Yes
allowedinmemorysettings 0x15000075
osdevice             partition=C:
systemroot           \WINDOWS
resumeobject         {4b2bb885-8418-11e8-b279-001f29044af6}
nx                   OptIn
bootmenupolicy       Standard
hypervisorlaunchtype Auto
```

In the output, locate the **Windows Boot Loader** section that includes the line `identifier={current}`. In that section, locate the `recoverysequence` attribute. The value of this attribute should be a GUID value, not a string of zeros.

Event ID 851: Contact the manufacturer for BIOS upgrade instructions

The event information resembles the following:

Failed to enable Silent Encryption.

Error: BitLocker Drive Encryption cannot be enabled on the operating system drive. Contact the computer manufacturer for BIOS upgrade instructions.

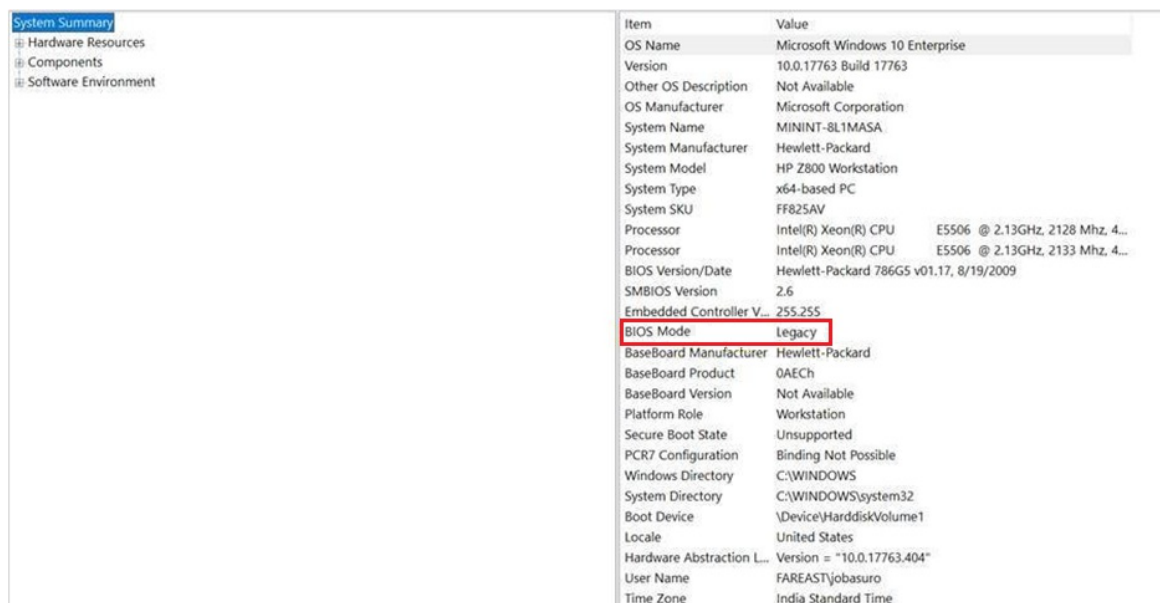
Cause

The device must have Unified Extensible Firmware Interface (UEFI) BIOS. Silent BitLocker drive encryption does not support legacy BIOS.

Resolution

To verify the BIOS mode, use the System Information application. To do this, follow these steps:

1. Select **Start**, and enter `msinfo32` in the **Search** box.
2. Verify that the **BIOS Mode** setting is **UEFI** and not **Legacy**.



The screenshot shows the System Information application window. The left sidebar is expanded to 'Software Environment'. The main pane displays a list of system information items and their values. The 'BIOS Mode' item is highlighted with a red box, and its value is 'Legacy'.

Item	Value
OS Name	Microsoft Windows 10 Enterprise
Version	10.0.17763 Build 17763
Other OS Description	Not Available
OS Manufacturer	Microsoft Corporation
System Name	MININT-8L1MASA
System Manufacturer	Hewlett-Packard
System Model	HP Z800 Workstation
System Type	x64-based PC
System SKU	FF825AV
Processor	Intel(R) Xeon(R) CPU E5506 @ 2.13GHz 2128 Mhz, 4...
Processor	Intel(R) Xeon(R) CPU E5506 @ 2.13GHz 2133 Mhz, 4...
BIOS Version/Date	Hewlett-Packard 786G5 v01.17, 8/19/2009
SMBIOS Version	2.6
Embedded Controller V...	255.255
BIOS Mode	Legacy
BaseBoard Manufacturer	Hewlett-Packard
BaseBoard Product	0AEC
BaseBoard Version	Not Available
Platform Role	Workstation
Secure Boot State	Unsupported
PCR7 Configuration	Binding Not Possible
Windows Directory	C:\WINDOWS
System Directory	C:\WINDOWS\system32
Boot Device	\Device\HarddiskVolume1
Locale	United States
Hardware Abstraction L...	Version = "10.0.17763.404"
User Name	FAREAST\jobasuro
Time Zone	India Standard Time

3. If the **BIOS Mode** setting is **Legacy**, you have to switch the BIOS into **UEFI** or **EFI** mode. The steps for doing this are specific to the device.

NOTE

If the device supports only Legacy mode, you cannot use Intune to manage BitLocker device encryption on the device.

Error message: The UEFI variable 'SecureBoot' could not be read

You receive an error message that resembles the following:

Error: BitLocker cannot use Secure Boot for integrity because the UEFI variable 'SecureBoot' could not be read. A required privilege is not held by the client.

Cause

A platform configuration register (PCR) is a memory location in the TPM. In particular, PCR 7 measures the state of secure boot. Silent BitLocker drive encryption requires the secure boot to be turned on.

Resolution

You can resolve this issue by verifying the PCR validation profile of the TPM and the secure boot state. To do this, follow these steps:

Step 1: Verify the PCR validation profile of the TPM

To verify that PCR 7 is in use, open an elevated Command Prompt window and run the following command:

```
Manage-bde -protectors -get %systemdrive%
```

In the TPM section of the output of this command, verify whether the **PCR Validation Profile** setting includes 7, as follows:

```
C:\windows\system32>manage-bde -protectors -get %systemdrive%
BitLocker Drive Encryption: Configuration Tool version 10.0.17763
Copyright (C) 2013 Microsoft Corporation. All rights reserved.

Volume C: [OSDisk]
All Key Protectors

    Numerical Password:
        ID: {DFBA5848-A9B1-4D5E-AAC5-011DEF09F705}
        Password:

    TPM:
        ID: {1B527682-77DA-475F-A2A7-EB81B73D1A88}
        PCR Validation Profile:
            7, 11
            (Uses Secure Boot for integrity validation)
```

If PCR Validation Profile doesn't include 7 (for example, the values include 0, 2, 4, and 11, but not 7), then secure boot is not turned on.

```
C:\WINDOWS\system32>manage-bde -protectors -get %systemdrive%
BitLocker Drive Encryption: Configuration Tool version 10.0.17763
Copyright (C) 2013 Microsoft Corporation. All rights reserved.

Volume C: [OSDisk]
All Key Protectors

    Numerical Password:
        ID: {4856F8A7-2751-4668-89F8-BDF0D519AD4D}
        Password:

    TPM:
        ID: {90A0BB72-ABA6-4F08-AC89-84C4884491BD}
        PCR Validation Profile:
            0, 2, 4, 8, 9, 10, 11
```

2. Verify the secure boot state

To verify the secure boot state, use the System Information application. To do this, follow these steps:

1. Select **Start**, and enter **msinfo32** in the **Search** box.
2. Verify that the **Secure Boot State** setting is **On**, as follows:

Item	Value
BIOS Version/Date	HP Q78 Ver. 01.04.00, 9/12/2018
SMBIOS Version	3.1
Embedded Controll...	4.83
BIOS Mode	UEFI
BaseBoard Manufact...	HP
BaseBoard Product	83B2
BaseBoard Version	KBC Version 04.53.00
Platform Role	Mobile
Secure Boot State	On
PCR7 Configuration	Elevation Required to View
Windows Directory	C:\windows
System Directory	C:\windows\system32
Boot Device	\Device\HarddiskVolume2
Locale	United States

3. If the **Secure Boot State** setting is **Unsupported**, you cannot use Silent BitLocker Encryption on this device.

Item	Value
Other OS Description	Not Available
OS Manufacturer	Microsoft Corporation
System Name	MININT-8L1MASA
System Manufacturer	Hewlett-Packard
System Model	HP Z800 Workstation
System Type	x64-based PC
System SKU	FF825AV
Processor	Intel(R) Xeon(R) CPU E550
Processor	Intel(R) Xeon(R) CPU E550
BIOS Version/Date	Hewlett-Packard 786G5 v01.17, 8,
SMBIOS Version	2.6
Embedded Controller V...	255.255
BIOS Mode	Legacy
BaseBoard Manufacturer	Hewlett-Packard
BaseBoard Product	0AEC
BaseBoard Version	Not Available
Platform Role	Workstation
Secure Boot State	Unsupported
PCR7 Configuration	Binding Not Possible

NOTE

You can also use the [Confirm-SecureBootUEFI](#) cmdlet to verify the Secure Boot state. To do this, open an elevated PowerShell window and run the following command:

```
PS C:\> Confirm-SecureBootUEFI
```

If the computer supports Secure Boot and Secure Boot is enabled, this cmdlet returns "True."

If the computer supports secure boot and secure boot is disabled, this cmdlet returns "False."

If the computer does not support Secure Boot or is a BIOS (non-UEFI) computer, this cmdlet returns "Cmdlet not supported on this platform."

Event ID 846, 778, and 851: Error 0x80072f9a

In this case, you are deploying Intune policy to encrypt a Windows 11, Windows 10, version 1809 device, and store the recovery password in Azure Active Directory (Azure AD). As part of the policy configuration, you have selected the **Allow standard users to enable encryption during Azure AD Join** option.

The policy deployment fails and the failure generates the following events (visible in Event Viewer in the

Applications and Services Logs\Microsoft\Windows\BitLocker API folder):

Event ID:846

Event: Failed to backup BitLocker Drive Encryption recovery information for volume C: to your Azure AD.

Traceld: {cbac2b6f-1434-4faa-a9c3-597b17c1dfa3} Error: Unknown HRESULT Error code: 0x80072f9a

Event ID:778

Event: The BitLocker volume C: was reverted to an unprotected state.

Event ID: 851

Event: Failed to enable Silent Encryption.

Error: Unknown HRESULT Error code: 0x80072f9a.

These events refer to Error code 0x80072f9a.

Cause

These events indicate that the signed-in user does not have permission to read the private key on the certificate that is generated as part of the provisioning and enrollment process. Therefore, the BitLocker MDM policy refresh fails.

The issue affects Windows 11 and Windows 10 version 1809.

Resolution

To resolve this issue, install the [May 21, 2019](#) update.

Error message: There are conflicting group policy settings for recovery options on operating system drives

You receive a message that resembles the following:

Error: BitLocker Drive Encryption cannot be applied to this drive because there are conflicting Group Policy settings for recovery options on operating system drives. Storing recovery information to Active Directory Domain Services cannot be required when the generation of recovery passwords is not permitted. Please have your system administrator resolve these policy conflicts before attempting to enable BitLocker...

Resolution

To resolve this issue, review your group policy object (GPO) settings for conflicts. For further guidance, see the next section, [Review your BitLocker policy configuration](#).

For more information about GPOs and BitLocker, see [BitLocker Group Policy Reference](#).

Review your BitLocker policy configuration

For information about the procedure to use policy together with BitLocker and Intune, see the following resources:

- [BitLocker management for enterprises: Managing devices joined to Azure Active Directory](#)
- [BitLocker Group Policy Reference](#)
- [Configuration service provider reference](#)
- [Policy CSP – BitLocker](#)

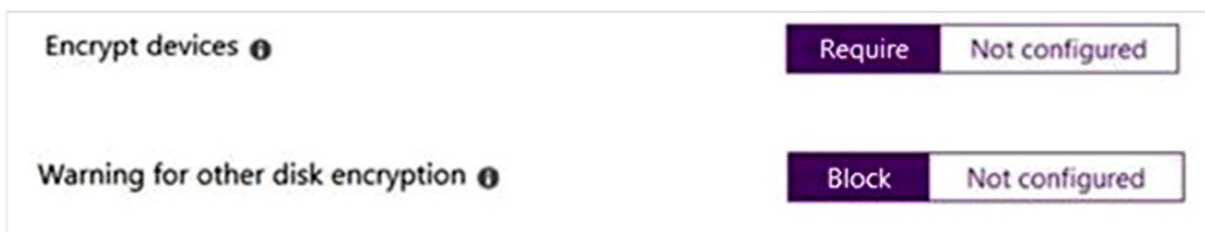
- [BitLocker CSP](#)
- [Enable ADMX-backed policies in MDM](#)
- [gpresult](#)

Intune offers the following enforcement types for BitLocker:

- **Automatic** (Enforced when the device joins Azure AD during the provisioning process. This option is available in Windows 10 version 1703 and later, or Windows 11.)
- **Silent** (Endpoint protection policy. This option is available in Windows 10 version 1803 and later, or Windows 11.)
- **Interactive** (Endpoint policy for Windows versions that are older than Windows 10 version 1803, or Windows 11.)

If your device runs Windows 10 version 1703 or later, or Windows 11, supports Modern Standby (also known as Instant Go) and is HSTI-compliant, joining the device to Azure AD triggers automatic device encryption. A separate endpoint protection policy is not required to enforce device encryption.

If your device is HSTI-compliant but does not support Modern Standby, you have to configure an endpoint protection policy to enforce silent BitLocker drive encryption. The settings for this policy should resemble the following:



The OMA-URI references for these settings are as follows:

- OMA-URI: `./Device/Vendor/MSFT/BitLocker/RequireDeviceEncryption`
Value Type: **Integer**
Value: **1** (1 = Require, 0 = Not Configured)
- OMA-URI: `./Device/Vendor/MSFT/BitLocker/AllowWarningForOtherDiskEncryption`
Value Type: **Integer**
Value: **0** (0 = Blocked, 1 = Allowed)

NOTE

Because of an update to the BitLocker Policy CSP, if the device uses Windows 10 version 1809 or later, or Windows 11, you can use an endpoint protection policy to enforce silent BitLocker Device Encryption even if the device is not HSTI-compliant.

NOTE

If the **Warning for other disk encryption** setting is set to **Not configured**, you have to manually start the BitLocker drive encryption wizard.

If the device does not support Modern Standby but is HSTI-compliant, and it uses a version of Windows that is earlier than Windows 10, version 1803, or Windows 11, an endpoint protection policy that has the settings that are described in this article delivers the policy configuration to the device. However, Windows then notifies the user to manually enable BitLocker Drive Encryption. To do this, the user selects the notification. This action starts the BitLocker Drive Encryption wizard.

The Intune 1901 release provides settings that you can use to configure automatic device encryption for Autopilot devices for standard users. Each device must meet the following requirements:

- Be HSTI-compliant
- Support Modern Standby
- Use Windows 10 version 1803 or later, or Windows 11

Allow standard users to enable encryption during Azure AD Join ⓘ

Allow

Not configured

The OMA-URI references for these settings are as follows:

- OMA-URI: `./Device/Vendor/MSFT/BitLocker/AllowStandardUserEncryption`
Value Type: **Integer** Value: 1

NOTE

This node works together with the `RequireDeviceEncryption` and `AllowWarningForOtherDiskEncryption` nodes. For this reason, when you set `RequireDeviceEncryption` to 1, `AllowStandardUserEncryption` to 1, and `AllowWarningForOtherDiskEncryption` to 0, Intune enforces silent BitLocker encryption for Autopilot devices that have standard user profiles.

Verifying that BitLocker is operating correctly

During regular operations, BitLocker drive encryption generates events such as Event ID 796 and Event ID 845.

Level	Date and ...	Source	Event...	Task Category
Information	4/13/2019...	BitLocker-API	775	None
Information	4/13/2019...	BitLocker-API	796	None
Error	4/13/2019...	BitLocker-API	853	None

Event 796, BitLocker-API

General Details

BitLocker Drive Encryption is using software-based encryption to protect volume C:.

Level	Date and ...	Source	Event...	Task Category
Information	4/13/2019...	BitLocker-API	845	None
Information	4/13/2019...	BitLocker-API	775	None
Information	4/13/2019...	BitLocker-API	796	None

Event 845, BitLocker-API

General Details

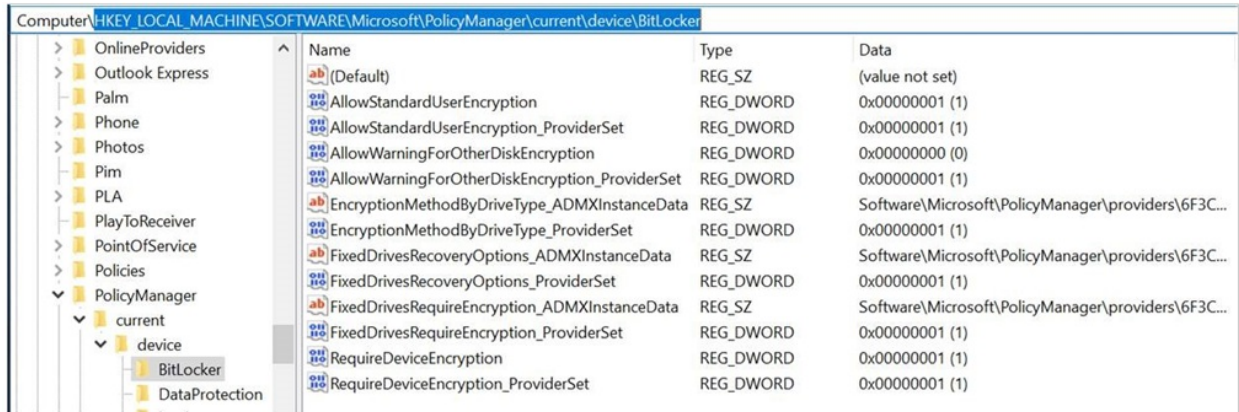
BitLocker Drive Encryption recovery information for volume C: was backed up successfully to your Azure AD. Protector GUID: {dda4f062-803e-4efb-bd1e-5f46e0c52a24}. Tracelid: {e3c6b9f3-dc9b-4bc9-9ace-0c98e818db69}

You can also determine whether the BitLocker recovery password has been uploaded to Azure AD by checking the device details in the Azure AD Devices section.

BITLOCKER KEY ID	BITLOCKER RECOVERY KEY	DRIVE TYPE
dda4f062-803e-4efb-bd1e-...	003289-619993-483340-53...	Operating system drive

On the device, check the Registry Editor to verify the policy settings on the device. Verify the entries under the following subkeys:

- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\PolicyManager\current\device\BitLocker
- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\PolicyManager\current\device



BitLocker network unlock: known issues

7/1/2022 • 4 minutes to read • [Edit Online](#)

By using the BitLocker network unlock feature, you can manage computers remotely without having to enter a BitLocker PIN when each computer starts up. To configure this behavior, your environment needs to meet the following requirements:

- Each computer belongs to a domain.
- Each computer has a wired connection to the internal network.
- The internal network uses DHCP to manage IP addresses.
- Each computer has a DHCP driver implemented in its Unified Extensible Firmware Interface (UEFI) firmware.

For general guidelines about how to troubleshoot network unlock, see [How to enable network unlock: Troubleshoot network unlock](#).

This article describes several known issues that you may encounter when you use network unlock, and provides guidance to address these issues.

Tip: Detect whether BitLocker network unlock is enabled on a specific computer

You can use the following steps on computers with either x64 or x32 UEFI firmware. You can also script these commands.

1. Open an elevated command prompt window and run the following command:

```
manage-bde -protectors -get <Drive>
```

```
manage-bde -protectors -get C:
```

Where `<Drive>` is the drive letter, followed by a colon (`:`), of the bootable drive. If the output of this command includes a key protector of type **TpmCertificate (9)**, the configuration is correct for BitLocker network unlock.

2. Start Registry Editor, and verify the following settings:

- Entry `HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\FVE: OSManageNKP` is set to `1`.
- Subkey `HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\SystemCertificates\FVE_NKP\Certificates` has an entry whose name matches the name of the certificate thumbprint of the network unlock key protector that you found in step 1.

1. On a Surface Pro 4 device, BitLocker network unlock doesn't work because the UEFI network stack is incorrectly configured

You've configured BitLocker network unlock as described in [BitLocker: How to enable network unlock](#). You've configured the UEFI of the device to use DHCP. However, when you restart the device, it still prompts you for the BitLocker PIN.

You test another device, such as a different type of tablet or laptop PC that's configured to use the same

infrastructure. The device restarts as expected, without prompting for the BitLocker PIN. You conclude that the infrastructure is correctly configured, and the issue is specific to the device.

Cause of issue 1

The UEFI network stack on the device was incorrectly configured.

Resolution for issue 1

To correctly configure the UEFI network stack of the Surface Pro 4, you have to use Microsoft Surface Enterprise Management Mode (SEMM). For information about SEMM, see [Enroll and configure Surface devices with SEMM](#).

NOTE

If you cannot use SEMM, you may be able to configure the Surface Pro 4 to use BitLocker network unlock by configuring the device to use the network as its first boot option.

2. Unable to use BitLocker network unlock feature on a Windows client computer

You have configured BitLocker network unlock as described in [BitLocker: How to enable network unlock](#). You have a Windows 8 client computer that is connected to the internal network with an ethernet cable. However, when you restart the computer, it still prompts you for the BitLocker PIN.

Cause of issue 2

A Windows 8-based or Windows Server 2012-based client computer sometimes doesn't receive or use the network unlock protector, depending on whether the client receives unrelated BOOTP replies from a DHCP server or WDS server.

DHCP servers may send any DHCP options to a BOOTP client as allowed by the DHCP options and BOOTP vendor extensions. This behavior means that because a DHCP server supports BOOTP clients, the DHCP server replies to BOOTP requests.

The manner in which a DHCP server handles an incoming message depends in part on whether the message uses the Message Type option:

- The first two messages that the BitLocker network unlock client sends are DHCP DISCOVER\REQUEST messages. They use the Message Type option, so the DHCP server treats them as DHCP messages.
- The third message that the BitLocker network unlock client sends doesn't have the Message Type option. The DHCP server treats the message as a BOOTP request.

A DHCP server that supports BOOTP clients must interact with those clients according to the BOOTP protocol. The server must create a BOOTP BOOTREPLY message instead of a DHCP DHCPOFFER message. (In other words, the server must not include the DHCP message option type and must not exceed the size limit for BOOTREPLY messages.) After the server sends the BOOTP BOOTREPLY message, the server marks a binding for a BOOTP client as BOUND. A non-DHCP client doesn't send a DHCPREQUEST message, nor does that client expect a DHCPACK message.

If a DHCP server that isn't configured to support BOOTP clients receives a BOOTREQUEST message from a BOOTP client, that server silently discards the BOOTREQUEST message.

For more information about DHCP and BitLocker network unlock, see [BitLocker: How to enable network unlock: network unlock sequence](#).

Resolution for issue 2

To resolve this issue, change the configuration of the DHCP server by changing the DHCP option from DHCP and BOOTP to DHCP.

BitLocker recovery: known issues

7/1/2022 • 11 minutes to read • [Edit Online](#)

This article describes common issues that may prevent BitLocker from behaving as expected when you recover a drive, or that may cause BitLocker to start recovery unexpectedly. The article also provides guidance to address these issues.

NOTE

In this article, "recovery password" refers to the 48-digit recovery password and "recovery key" refers to 32-digit recovery key. For more information, see [BitLocker key protectors](#).

Windows prompts for a non-existing BitLocker recovery password

Windows prompts you for a BitLocker recovery password. However, you did not configure a BitLocker recovery password.

Resolution

The BitLocker and Active Directory Domain Services (AD DS) FAQ address situations that may produce this symptom, and provides information about the procedure to resolve the issue:

- [What if BitLocker is enabled on a computer before the computer has joined the domain?](#)
- [What happens if the backup initially fails? Will BitLocker retry the backup?](#)

The recovery password for a laptop was not backed up, and the laptop is locked

You have a Windows 11 or Windows 10 Home-based laptop, and you have to recover its hard disk. The disk was encrypted by using BitLocker Drive Encryption. However, the BitLocker recovery password was not backed up, and the usual user of the laptop is not available to provide the password.

Resolution

You can use either of the following methods to manually back up or synchronize an online client's existing recovery information:

- Create a Windows Management Instrumentation (WMI) script that backs up the information. For more information, see [BitLocker Drive Encryption Provider](#).
- In an elevated Command Prompt window, use the `manage-bde` command to back up the information.

For example, to back up all of the recovery information for the C: drive to AD DS, open an elevated Command Prompt window and run the following command:

```
manage-bde -protectors -adbackup C:
```

NOTE

BitLocker does not automatically manage this backup process.

Tablet devices do not support using Manage-bde -forcerecovery to test recovery mode

You have a tablet or slate device, and you try to test BitLocker recovery by running the following command:

```
Manage-bde -forcerecovery
```

However, after you enter the recovery password, the device cannot start.

Cause

IMPORTANT

Tablet devices do not support the **manage-bde -forcerecovery** command.

This issue occurs because the Windows Boot Manager cannot process touch-input during the pre-boot phase of startup. If Boot Manager detects that the device is a tablet, it redirects the startup process to the Windows Recovery Environment (WinRE), which can process touch-input.

If WindowsRE detects the TPM protector on the hard disk, it does a PCR reseal. However, the **manage-bde -forcerecovery** command deletes the TPM protectors on the hard disk. Therefore, WinRE cannot reseal the PCRs. This failure triggers an infinite BitLocker recovery cycle and prevents Windows from starting.

This behavior is by design for all versions of Windows.

Workaround

To resolve the restart loop, follow these steps:

1. On the BitLocker Recovery screen, select **Skip this drive**.
2. Select **Troubleshoot > Advanced Options > Command Prompt**.
3. In the Command Prompt window, run the following commands:

```
manage-bde -unlock C: -rp <48-digit BitLocker recovery password>  
manage-bde -protectors -disable C:
```

4. Close the Command Prompt window.
5. Shut down the device.
6. Start the device. Windows should start as usual.

After you install UEFI or TPM firmware updates on Surface, BitLocker prompts for the recovery password

You have a Surface device that has BitLocker drive encryption turned on. You update the firmware of the device TPM or install an update that changes the signature of the system firmware. For example, you install the Surface TPM (IFX) update.

You experience one or more of the following symptoms on the Surface device:

- At startup, you are prompted for your BitLocker recovery password. You enter the correct recovery password, but Windows doesn't start up.
- Startup progresses directly into the Surface Unified Extensible Firmware Interface (UEFI) settings.

- The Surface device appears to be in an infinite restart loop.

Cause

This issue occurs if the Surface device TPM is configured to use Platform Configuration Register (PCR) values other than the default values of PCR 7 and PCR 11. For example, the following settings can configure the TPM this way:

- Secure boot is turned off.
- PCR values have been explicitly defined, such as by group policy.

Devices that support Connected Standby (also known as *InstantGO* or *Always On, Always Connected PCs*), including Surface devices, must use PCR 7 of the TPM. In its default configuration on such systems, BitLocker binds to PCR 7 and PCR 11 if PCR 7 and Secure Boot are correctly configured. For more information, see "About the Platform Configuration Register (PCR)" at [BitLocker Group Policy Settings](#).

Resolution

To verify the PCR values that are in use on a device, open an elevated Command Prompt window and run the following command:

```
manage-bde.exe -protectors -get <OSDriveLetter>:
```

In this command, *<OSDriveLetter>* represents the drive letter of the operating system drive.

To resolve this issue and repair the device, follow these steps.

Step 1: Disable the TPM protectors on the boot drive

If you have installed a TPM or UEFI update and your device cannot start, even if you enter the correct BitLocker recovery password, you can restore the ability to start by using the BitLocker recovery password and a Surface recovery image to remove the TPM protectors from the boot drive.

To do this, follow these steps:

1. Obtain your BitLocker recovery password from [your Microsoft.com account](#). If BitLocker is managed by a different method, such as Microsoft BitLocker Administration and Monitoring (MBAM), contact your administrator for help.
2. Use another computer to download the Surface recovery image from [Download a recovery image for your Surface](#). Use the downloaded image to create a USB recovery drive.
3. Insert the USB Surface recovery image drive into the Surface device, and start the device.
4. When you are prompted, select the following items:
 - a. Your operating system language.
 - b. Your keyboard layout.
5. Select **Troubleshoot > Advanced Options > Command Prompt**.
6. In the Command Prompt window, run the following commands:

```
manage-bde -unlock -recoverypassword <Password> <DriveLetter>:  
manage-bde -protectors -disable <DriveLetter>:
```

In these commands, *<Password>* is the BitLocker recovery password that you obtained in step 1, and *<DriveLetter>* is the drive letter that is assigned to your operating system drive.

NOTE

For more information about how to use this command, see [manage-bde: unlock](#).

- Restart the computer.
- When you are prompted, enter the BitLocker recovery password that you obtained in step 1.

NOTE

After you disable the TPM protectors, BitLocker drive encryption no longer protects your device. To re-enable BitLocker drive encryption, select **Start**, type **Manage BitLocker**, and then press Enter. Follow the steps to encrypt your drive.

Step 2: Use Surface BMR to recover data and reset your device

To recover data from your Surface device if you cannot start Windows, follow steps 1 through 5 of [Step 1](#) to return to the Command Prompt window, and then follow these steps:

- At the command prompt, run the following command:

```
manage-bde -unlock -recoverypassword <Password> <DriveLetter>:
```

In this command, *<Password>* is the BitLocker recovery password that you obtained in step 1 of [Step 1](#), and *<DriveLetter>* is the drive letter that is assigned to your operating system drive.

- After the drive is unlocked, use the **copy** or **xcopy** command to copy the user data to another drive.

NOTE

For more information about these commands, see the [Windows commands](#).

- To reset your device by using a Surface recovery image, follow the instructions in the "How to reset your Surface using your USB recovery drive" section in [Creating and using a USB recovery drive](#).

Step 3: Restore the default PCR values

To prevent this issue from recurring, we strongly recommend that you restore the default configuration of secure boot and the PCR values.

To enable secure boot on a Surface device, follow these steps:

- Suspend BitLocker. To do this, open an elevated Windows PowerShell window, and run the following cmdlet:

```
Suspend-BitLocker -MountPoint "<DriveLetter>:" -RebootCount 0
```

In this command, *<DriveLetter>* is the letter that is assigned to your drive.

- Restart the device, and then edit the BIOS to set the **Secure Boot** option to **Microsoft Only**.
- Restart the device.
- Open an elevated PowerShell window, and run the following cmdlet:

```
Resume-BitLocker -MountPoint "<DriveLetter>:"
```


To reset the PCR settings on the TPM, follow these steps:

1. Disable any Group Policy Objects that configure the PCR settings, or remove the device from any groups that enforce such policies.

For more information, see [BitLocker Group Policy settings](#).

2. Suspend BitLocker. To do this, open an elevated Windows PowerShell window, and run the following cmdlet:

```
Suspend-BitLocker -MountPoint "<DriveLetter>:" -RebootCount 0
```

where *<DriveLetter>* is the letter assigned to your drive.

3. Run the following cmdlet:

```
Resume-BitLocker -MountPoint "<DriveLetter>:"
```

Step 4: Suspend BitLocker during TPM or UEFI firmware updates

You can avoid this scenario when you install updates to system firmware or TPM firmware by temporarily suspending BitLocker before you apply such updates.

IMPORTANT

TPM and UEFI firmware updates may require multiple restarts while they install. To keep BitLocker suspended during this process, you must use [Suspend-BitLocker](#) and set the **Reboot Count** parameter to either of the following values:

- 2 or greater: This value sets the number of times the device can restart before BitLocker Device Encryption resumes.
- 0: This value suspends BitLocker Drive Encryption indefinitely, until you use [Resume-BitLocker](#) or another mechanism to resume protection.

To suspend BitLocker while you install TPM or UEFI firmware updates:

1. Open an elevated Windows PowerShell window, and run the following cmdlet:

```
Suspend-BitLocker -MountPoint "<DriveLetter>:" -RebootCount 0
```

In this cmdlet *<DriveLetter>* is the letter that is assigned to your drive.

2. Install the Surface device driver and firmware updates.
3. After you install the firmware updates, restart the computer, open an elevated PowerShell window, and then run the following cmdlet:

```
Resume-BitLocker -MountPoint "<DriveLetter>:"
```

To re-enable BitLocker drive encryption, select **Start**, type **Manage BitLocker**, and then press Enter. Follow the steps to encrypt your drive.

After you install an update to a Hyper V-enabled computer, BitLocker prompts for the recovery password and returns error 0xC0210000

You have a device that runs Windows 11, Windows 10, version 1703, Windows 10, version 1607, or Windows

Server 2016. Also, Hyper-V is enabled on the device. After you install an affected update and restart the device, the device enters BitLocker Recovery mode and you see error code 0xC0210000.

Workaround

If your device is already in this state, you can successfully start Windows after suspending BitLocker from the Windows Recovery Environment (WinRE). To do this, follow these steps:

1. Retrieve the 48-digit BitLocker recovery password for the operating system drive from your organization's portal or from wherever the password was stored when BitLocker Drive Encryption was first turned on.
2. On the Recovery screen, press Enter. When you are prompted, enter the recovery password.
3. If your device starts in the (WinRE) and prompts you for the recovery password again, select **Skip the drive**.
4. Select **Advanced options > Troubleshoot > Advanced options > Command Prompt**.
5. In the Command Prompt window, run the following commands:

```
Manage-bde -unlock c: -rp <48 digit numerical recovery password separated by "-" in 6 digit group>
Manage-bde -protectors -disable c:
exit
```

These commands unlock the drive and then suspend BitLocker by disabling the TPM protectors on the drive. The final command closes the Command Prompt window.

NOTE

These commands suspend BitLocker for one restart of the device. The `-rc 1` option works only inside the operating system and does not work in the recovery environment.

6. Select **Continue**. Windows should start.
7. After Windows has started, open an elevated Command Prompt window and run the following command:

```
Manage-bde -protectors -enable c:
```

IMPORTANT

Unless you suspend BitLocker before you start the device, this issue recurs.

To temporarily suspend BitLocker just before you restart the device, open an elevated Command Prompt window and run the following command:

```
Manage-bde -protectors -disable c: -rc 1
```

Resolution

To resolve this issue, install the appropriate update on the affected device:

- For Windows 10, version 1703, or Windows 11: [July 9, 2019—KB4507450 \(OS Build 15063.1928\)](#)
- For Windows 11, Windows 10, version 1607 and Windows Server 2016: [July 9, 2019—KB4507460 \(OS Build](#)

Credential Guard/Device Guard on TPM 1.2: At every restart, BitLocker prompts for the recovery password and returns error 0xC0210000

You have a device that uses TPM 1.2 and runs Windows 10, version 1809, or Windows 11. Also, the device uses [Virtualization-based Security](#) features such as [Device Guard and Credential Guard](#). Every time that you start the device, the device enters BitLocker Recovery mode and you see error code 0xc0210000, and a message that resembles the following.

Recovery

Your PC/Device needs to be repaired. A required file couldn't be accessed because your BitLocker key wasn't loaded correctly.

Error code 0xc0210000

You'll need to use recovery tools. If you don't have any installation media (like a disc or USB device), contact your PC administrator or PC/Device manufacturer.

Cause

TPM 1.2 does not support Secure Launch. For more information, see [System Guard Secure Launch and SMM protection: Requirements Met by System Guard Enabled Machines](#)

For more information about this technology, see [Windows Defender System Guard: How a hardware-based root of trust helps protect Windows](#)

Resolution

To resolve this issue, do one of the following:

- Remove any device that uses TPM 1.2 from any group that is subject to GPOs that enforce secure launch.
- Edit the **Turn On Virtualization Based Security** GPO to set **Secure Launch Configuration** to **Disabled**.

BitLocker configuration: known issues

7/1/2022 • 6 minutes to read • [Edit Online](#)

This article describes common issues that affect your BitLocker's configuration and general functionality. This article also provides guidance to address these issues.

BitLocker encryption is slower in Windows 10 and Windows 11

In both Windows 11, Windows 10, and Windows 7, BitLocker runs in the background to encrypt drives. However, in Windows 11 and Windows 10, BitLocker is less aggressive about requesting resources. This behavior reduces the chance that BitLocker will affect the computer's performance.

To compensate for these changes, BitLocker uses a new conversion model. This model, (referred to as Encrypt-On-Write), makes sure that any new disk writes on all client SKUs and that any internal drives are always encrypted *as soon as you turn on BitLocker*.

IMPORTANT

To preserve backward compatibility, BitLocker uses the previous conversion model to encrypt removable drives.

Benefits of using the new conversion model

By using the previous conversion model, you cannot consider an internal drive to be protected (and compliant with data protection standards) until the BitLocker conversion is 100 percent complete. Before the process finishes, the data that existed on the drive before encryption began—that is, potentially compromised data—can still be read and written without encryption. Therefore, you must wait for the encryption process to finish before you store sensitive data on the drive. Depending on the size of the drive, this delay can be substantial.

By using the new conversion model, you can safely store sensitive data on the drive as soon as you turn on BitLocker. You don't have to wait for the encryption process to finish, and encryption does not adversely affect performance. The tradeoff is that the encryption process for pre-existing data takes more time.

Other BitLocker enhancements

After Windows 7 was released, several other areas of BitLocker were improved:

- **New encryption algorithm, XTS-AES.** The new algorithm provides additional protection from a class of attacks on encrypted data that rely on manipulating cipher text to cause predictable changes in plain text.

By default, this algorithm complies with the Federal Information Processing Standards (FIPS). FIPS is a United States Government standard that provides a benchmark for implementing cryptographic software.

- **Improved administration features.** You can manage BitLocker on PCs or other devices by using the following interfaces:
 - BitLocker Wizard
 - manage-bde
 - Group Policy Objects (GPOs)
 - Mobile Device Management (MDM) policy
 - Windows PowerShell
 - Windows Management Interface (WMI)

- **Integration with Azure Active Directory (Azure AD).** BitLocker can store recovery information in Azure AD to make it easier to recover.
- **Direct memory access (DMA) Port Protection.** By using MDM policies to manage BitLocker, you can block a device's DMA ports and secure the device during its startup.
- **BitLocker Network Unlock.** If your BitLocker-enabled desktop or server computer is connected to a wired corporate network in a domain environment, you can automatically unlock its operating system volume during a system restart.
- **Support for Encrypted Hard Drives.** Encrypted Hard Drives are a new class of hard drives that are self-encrypting at a hardware level and allow for full disk hardware encryption. By taking on that workload, Encrypted Hard Drives increase BitLocker performance and reduce CPU usage and power consumption.
- **Support for classes of HDD/SSD hybrid disks.** BitLocker can encrypt a disk that uses a small SSD as a non-volatile cache in front of the HDD, such as Intel Rapid Storage Technology.

Hyper-V Gen 2 VM: Cannot access the volume after BitLocker encryption

Consider the following scenario:

1. You turn on BitLocker on a generation-2 virtual machine (VM) that runs on Hyper-V.
2. You add data to the data disk as it encrypts.
3. You restart the VM, and observe the following:
 - The system volume is not encrypted.
 - The encrypted volume is not accessible, and the computer lists the volume's file system as "Unknown."
 - You see a message that resembles: "You need to format the disk in <x> drive before you can use it"

Cause

This issue occurs because the third-party filter driver Stcvsm.sys (from StorageCraft) is installed on the VM.

Resolution

To resolve this issue, remove the third-party software.

Production snapshots fail for virtualized domain controllers that use BitLocker-encrypted disks

You have a Windows Server 2019 or 2016 Hyper-V Server that is hosting VMs (guests) that are configured as Windows domain controllers. BitLocker has encrypted the disks that store the Active Directory database and log files. When you run a "production snapshot" of the domain controller guests, the Volume Snap-Shot (VSS) service does not correctly process the backup.

This issue occurs regardless of any of the following variations in the environment:

- How the domain controller volumes are unlocked.
- Whether the VMs are generation 1 or generation 2.
- Whether the guest operating system is Windows Server 2019, 2016 or 2012 R2.

In the domain controller application log, the VSS event source records event ID 8229:

ID: 8229

Level: Warning

Source: VSS

Message: A VSS writer has rejected an event with error 0x800423f4. The writer experienced a non-transient error. If the backup process is retried, the error is likely to reoccur.

Changes that the writer made to the writer components while handling the event will not be available to the requester.

Check the event log for related events from the application hosting the VSS writer.

Operation:

PostSnapshot Event

Context:

Execution Context: Writer Writer Class Id: {b2014c9e-8711-4c5c-a5a9-3cf384484757}

Writer Name: NTDS

Writer Instance ID: {d170b355-a523-47ba-a5c8-732244f70e75} Command Line:

C:\Windows\system32\lsass.exe

Process ID: 680

In the domain controller Directory Services event log, you see an event that resembles the following:

Error Microsoft-Windows-ActiveDirectory_DomainService 1168

Internal Processing Internal error: An Active Directory Domain Services error has occurred.

Additional Data

Error value (decimal): -1022

Error value (hex): fffffc02

Internal ID: 160207d9

NOTE

The internal ID of this event may differ based on your operating system release and path level.

After this issue occurs, if you run the **VSSADMIN list writers** command, you see output that resembles the following for the Active Directory Domain Services (NTDS) VSS Writer:

Writer name: 'NTDS'

Writer Id: {b2014c9e-8711-4c5c-a5a9-3cf384484757}

Writer Instance Id: {08321e53-4032-44dc-9b03-7a1a15ad3eb8}

State: [11] Failed

Last error: Non-retryable error

Additionally, you cannot back up the VMs until you restart them.

Cause

After VSS creates a snapshot of a volume, the VSS writer takes "post snapshot" actions. In the case of a "production snapshot," which you initiate from the host server, Hyper-V tries to mount the snapshotted volume. However, it cannot unlock the volume for unencrypted access. BitLocker on the Hyper-V server does not recognize the volume. Therefore, the access attempt fails and then the snapshot operation fails.

This behavior is by design.

Workaround

There is one supported way to perform backup and restore of a virtualized domain controller:

- Run Windows Server Backup in the guest operating system.

If you have to take a production snapshot of a virtualized domain controller, you can suspend BitLocker in the guest operating system before you start the production snapshot. However, this approach is not recommended.

For more information and recommendations about backing up virtualized domain controllers, see [Virtualizing Domain Controllers using Hyper-V: Backup and Restore Considerations for Virtualized Domain Controllers](#)

More information

When the VSS NTDS writer requests access to the encrypted drive, the Local Security Authority Subsystem Service (LSASS) generates an error entry that resembles the following:

```
\# for hex 0xc0210000 / decimal -1071579136
STATUS\_FVE\_LOCKED\_VOLUME ntstatus.h
\# This volume is locked by BitLocker Drive Encryption.
```

The operation produces the following call stack:

```
\# Child-SP RetAddr Call Site
00 00000086`b357a800 00007ffc`ea6e7a4c KERNELBASE!FindFirstFileExW+0x1ba \
[d:\rs1\minkernel\kernelbase\filefind.c @ 872\]
01 00000086`b357abd0 00007ffc`e824accb KERNELBASE!FindFirstFileW+0x1c \
[d:\rs1\minkernel\kernelbase\filefind.c @ 208\]
02 00000086`b357ac10 00007ffc`e824afa1 ESENT!COSFileFind::ErrInit+0x10b \
[d:\rs1\onecore\ds\esent\src\os\osfs.cxx @ 2476\]
03 00000086`b357b700 00007ffc`e827bf02 ESENT!COSFileSystem::ErrFileFind+0xa1 \
[d:\rs1\onecore\ds\esent\src\os\osfs.cxx @ 1443\]
04 00000086`b357b960 00007ffc`e82882a9 ESENT!JetGetDatabaseFileInfoEx+0xa2 \
[d:\rs1\onecore\ds\esent\src\ese\jetapi.cxx @ 11503\]
05 00000086`b357c260 00007ffc`e8288166 ESENT!JetGetDatabaseFileInfoExA+0x59 \
[d:\rs1\onecore\ds\esent\src\ese\jetapi.cxx @ 11759\]
06 00000086`b357c390 00007ffc`e84c64fb ESENT!JetGetDatabaseFileInfoA+0x46 \
[d:\rs1\onecore\ds\esent\src\ese\jetapi.cxx @ 12076\]
07 00000086`b357c3f0 00007ffc`e84c5f23 ntdsbsrv!CVssJetWriterLocal::RecoverJetDB+0x12f \
[d:\rs1\ds\ds\src\jetback\snapshot.cxx @ 2009\]
08 00000086`b357c710 00007ffc`e80339e0 ntdsbsrv!CVssJetWriterLocal::OnPostSnapshot+0x293 \
[d:\rs1\ds\ds\src\jetback\snapshot.cxx @ 2190\]
09 00000086`b357cad0 00007ffc`e801fe6d VSSAPI!CVssIJetWriter::OnPostSnapshot+0x300 \
[d:\rs1\base\stor\vss\modules\jetwriter\ijetwriter.cpp @ 1704\]
0a 00000086`b357ccc0 00007ffc`e8022193 VSSAPI!CVssWriterImpl::OnPostSnapshotGuard+0x1d \
[d:\rs1\base\stor\vss\modules\vswriter\vswrtemp.cpp @ 5228\]
0b 00000086`b357ccf0 00007ffc`e80214f0 VSSAPI!CVssWriterImpl::PostSnapshotInternal+0xc3b \
[d:\rs1\base\stor\vss\modules\vswriter\vswrtemp.cpp @ 3552\]
```

BitLocker cannot encrypt a drive: known TPM issues

7/1/2022 • 4 minutes to read • [Edit Online](#)

This article describes common issues that affect the Trusted Platform Module (TPM) that might prevent BitLocker from encrypting a drive. This article also provides guidance to address these issues.

NOTE

If you have determined that your BitLocker issue does not involve the TPM, see [BitLocker cannot encrypt a drive: known issues](#).

The TPM is locked and you see "The TPM is defending against dictionary attacks and is in a time-out period"

When you turn on BitLocker drive encryption, it does not start. Instead, you receive a message that resembles "The TPM is defending against dictionary attacks and is in a time-out period."

Cause

The TPM is locked out.

Resolution

To resolve this issue, follow these steps:

1. Open an elevated PowerShell window and run the following script:

```
$Tpm = Get-WmiObject -class Win32_Tpm -namespace "root\CIMv2\Security\MicrosoftTpm"  
$ConfirmationStatus = $Tpm.GetPhysicalPresenceConfirmationStatus(22).ConfirmationStatus  
if($ConfirmationStatus -ne 4) {$Tpm.SetPhysicalPresenceRequest(22)}
```

2. Restart the computer. If you are prompted at the restart screen, press F12 to agree.⁸
3. Retry starting BitLocker drive encryption.

You cannot prepare the TPM, and you see "The TPM is defending against dictionary attacks and is in a time-out period"

You cannot turn on BitLocker drive encryption on a device. You use the TPM management console (tpm.msc) to prepare the TPM on a device. The operation fails and you receive a message that resembles "The TPM is defending against dictionary attacks and is in a time-out period."

Cause

The TPM is locked out.

Resolution

To resolve this issue, disable and re-enable the TPM. To do this, follow these steps:

1. Restart the device, and change the BIOS configuration to disable the TPM.
2. Restart the device again, and return to the TPM management console. Following message is displayed:

Compatible Trusted Platform Module (TPM) cannot be found on this computer. Verify that this

computer has 1.2 TPM and it is turned on in the BIOS.

- Restart the device, and change the BIOS configuration to enable the TPM.
- Restart the device, and return to the TPM management console.

If you still cannot prepare the TPM, clear the existing TPM keys. To do this, follow the instructions in [Troubleshoot the TPM: Clear all the keys from the TPM](#).

WARNING

Clearing the TPM can cause data loss.

Access Denied: Failed to backup TPM Owner Authorization information to Active Directory Domain Services. Errorcode: 0x80070005

You have an environment that enforces the **Do not enable BitLocker until recovery information is stored in AD DS** policy. You try to turn on BitLocker drive encryption on a computer that runs Windows 7, but the operation fails. You receive a message that resembles "Access Denied" or "Insufficient Rights."

Cause

The TPM did not have sufficient permissions on the TPM devices container in Active Directory Domain Services (AD DS). Therefore, the BitLocker recovery information could not be backed up to AD DS, and BitLocker drive encryption could not run.

This issue appears to be limited to computers that run versions of Windows that are earlier than Windows 10.

Resolution

To verify that you have correctly identified this issue, use one of the following methods:

- Disable the policy or remove the computer from the domain. Then try to turn on BitLocker drive encryption again. The operation should now succeed.
- Use LDAP and network trace tools to examine the LDAP exchanges between the client and the AD DS domain controller to identify the cause of the "Access Denied" or "Insufficient Rights" error. In this case, you should see the error when the client tries to access its object in the "CN=TPM Devices,DC= <domain>,DC=com" container.

- To review the TPM information for the affected computer, open an elevated Windows PowerShell window and run the following command:

```
Get-ADComputer -Filter {Name -like "ComputerName"} -Property * | Format-Table name,msTPM-TPMInformationForComputer
```

In this command, *ComputerName* is the name of the affected computer.

- To resolve the issue, use a tool such as dscls.exe to ensure that the access control list of msTPM-TPMInformationForComputer grants both Read and Write permissions to NTAUTHORITY/SELF.

Cannot prepare the TPM, error 0x80072030: "There is no such object on the server"

Your domain controllers were upgraded from Windows Server 2008 R2 to Windows Server 2012 R2. A group

policy object (GPO) enforces the **Do not enable BitLocker until recovery information is stored in AD DS** policy.

You cannot turn on BitLocker drive encryption on a device. You use the TPM management console (tpm.msc) to prepare the TPM on a device. The operation fails and you see a message that resembles the following:

```
0x80072030 There is no such object on the server when a policy to back up TPM information to active directory is enabled
```

You have confirmed that the **ms-TPM-OwnerInformation** and **msTPM-TpmInformationForComputer** attributes are present.

Cause

The domain and forest functional level of the environment may still be set to Windows 2008 R2. Additionally, the permissions in AD DS might not be correctly set.

Resolution

To resolve this issue, follow these steps:

1. Upgrade the functional level of the domain and forest to Windows Server 2012 R2.
2. Download [Add-TPMSelfWriteACE.vbs](#).
3. In the script, modify the value of **strPathToDomain** to your domain name.
4. Open an elevated PowerShell window, and run the following command:

```
cscript <Path>Add-TPMSelfWriteACE.vbs
```

In this command *<Path>* is the path to the script file.

For more information, see the following articles:

- [Back up the TPM recovery information to AD DS](#)
- [Prepare your organization for BitLocker: Planning and policies](#)

BitLocker and TPM: other known issues

7/1/2022 • 4 minutes to read • [Edit Online](#)

This article describes common issues that relate directly to the trusted platform module (TPM), and provides guidance to address these issues.

Azure AD: Windows Hello for Business and single sign-on don't work

You have an Azure Active Directory (Azure AD)-joined client computer that can't authenticate correctly. You experience one or more of the following symptoms:

- Windows Hello for Business doesn't work.
- Conditional access fails.
- Single sign-on (SSO) doesn't work.

Additionally, the computer logs the following entry for Event ID 1026:

```
Log Name: System
Source: Microsoft-Windows-TPM-WMI
Date: <Date and Time>
Event ID: 1026
Task Category: None
Level: Information
Keywords:
User: SYSTEM
Computer: <Computer name>
Description:
The Trusted Platform Module (TPM) hardware on this computer cannot be provisioned for use automatically. To set up the TPM interactively use the TPM management console (Start->tpm.msc) and use the action to make the TPM ready.
Error: The TPM is defending against dictionary attacks and is in a time-out period.
Additional Information: 0x840000
```

Cause

This event indicates that the TPM isn't ready or has some setting that prevents access to the TPM keys.

Additionally, the behavior indicates that the client computer can't obtain a [Primary Refresh Token \(PRT\)](#).

Resolution

To verify the status of the PRT, use the [dsregcmd /status command](#) to collect information. In the tool output, verify that either **User state** or **SSO state** contains the **AzureAdPrt** attribute. If the value of this attribute is **No**, the PRT wasn't issued. This may indicate that the computer couldn't present its certificate for authentication.

To resolve this issue, follow these steps to troubleshoot the TPM:

1. Open the TPM management console (tpm.msc). To do this, select **Start**, and enter **tpm.msc** in the **Search** box.
2. If you see a notice to either unlock the TPM or reset the lockout, follow those instructions.
3. If you don't see such a notice, review the BIOS settings of the computer for any setting that you can use to reset or disable the lockout.

4. Contact the hardware vendor to determine whether there's a known fix for the issue.
5. If you still can't resolve the issue, clear and reinitialize the TPM. To do this, follow the instructions in [Troubleshoot the TPM: Clear all the keys from the TPM](#).

WARNING

Clearing the TPM can cause data loss.

TPM 1.2 Error: Loading the management console failed. The device that is required by the cryptographic provider isn't ready for use

You have a Windows 11 or Windows 10 version 1703-based computer that uses TPM version 1.2. When you try to open the TPM management console, you receive the following message:

```
Loading the management console failed. The device that is required by the cryptographic provider is not ready for use.  
HRESULT 0x800900300x80090030 - NTE_DEVICE_NOT_READY  
The device that is required by this cryptographic provider is not ready for use.  
TPM Spec version: TPM v1.2
```

On a different device that is running the same version of Windows, you can open the TPM management console.

Cause (suspected)

These symptoms indicate that the TPM has hardware or firmware issues.

Resolution

To resolve this issue, switch the TPM operating mode from version 1.2 to version 2.0.

If this doesn't resolve the issue, consider replacing the device motherboard. After you replace the motherboard, switch the TPM operating mode from version 1.2 to version 2.0.

Devices don't join hybrid Azure AD because of a TPM issue

You have a device that you're trying to join to a hybrid Azure AD. However, the join operation appears to fail.

To verify that the join succeeded, use the [dsregcmd /status command](#). In the tool output, the following attributes indicate that the join succeeded:

- **AzureAdJoined:** YES
- **DomainName:** <on-prem Domain name>

If the value of **AzureADJoined** is **No**, the join operation failed.

Causes and Resolutions

This issue may occur when the Windows operating system isn't the owner of the TPM. The specific fix for this issue depends on which errors or events you experience, as shown in the following table:

MESSAGE	REASON	RESOLUTION
NTE_BAD_KEYSET (0x80090016/-2146893802)	TPM operation failed or was invalid	This issue was probably caused by a corrupted sysprep image. Make sure that you create the sysprep image by using a computer that isn't joined to or registered in Azure AD or hybrid Azure AD.

MESSAGE	REASON	RESOLUTION
---------	--------	------------

TPM_E_PCP_INTERNAL_ERROR (0x80290407/-2144795641)	Generic TPM error.	If the device returns this error, disable its TPM. Windows 10, version 1809 and later versions, or Windows 11 automatically detect TPM failures and finish the hybrid Azure AD join without using the TPM.
TPM_E_NOTFIPS (0x80280036/-2144862154)	The FIPS mode of the TPM is currently not supported.	If the device gives this error, disable its TPM. Windows 10, version 1809 and later versions, or Windows 11 automatically detect TPM failures and finish the hybrid Azure AD join without using the TPM.
NTE_AUTHENTICATION_IGNORED (0x80090031/-2146893775)	The TPM is locked out.	This error is transient. Wait for the cooldown period, and then retry the join operation.

For more information about TPM issues, see the following articles:

- [TPM fundamentals: Anti-hammering](#)
- [Troubleshooting hybrid Azure Active Directory-joined devices](#)
- [Troubleshoot the TPM](#)

Decode Measured Boot logs to track PCR changes

7/1/2022 • 3 minutes to read • [Edit Online](#)

Platform Configuration Registers (PCRs) are memory locations in the Trusted Platform Module (TPM). BitLocker and its related technologies depend on specific PCR configurations. Additionally, specific change in PCRs can cause a device or computer to enter BitLocker recovery mode.

By tracking changes in the PCRs, and identifying when they changed, you can gain insight into issues that occur or learn why a device or computer entered BitLocker recovery mode. The Measured Boot logs record PCR changes and other information. These logs are located in the C:\Windows\Logs\MeasuredBoot\ folder.

This article describes tools that you can use to decode these logs: TBSLogGenerator and PCPTool.

For more information about Measured Boot and PCRs, see the following articles:

- [TPM fundamentals: Measured Boot with support for attestation](#)
- [Understanding PCR banks on TPM 2.0 devices](#)

Use TBSLogGenerator to decode Measured Boot logs

Use TBSLogGenerator to decode Measured Boot logs that you have collected from Windows 11, Windows 10, and earlier versions. You can install this tool on the following systems:

- A computer that is running Windows Server 2016 and that has a TPM enabled
- A Gen 2 virtual machine (running on Hyper-V) that is running Windows Server 2016 (you can use the virtual TPM)

To install the tool, follow these steps:

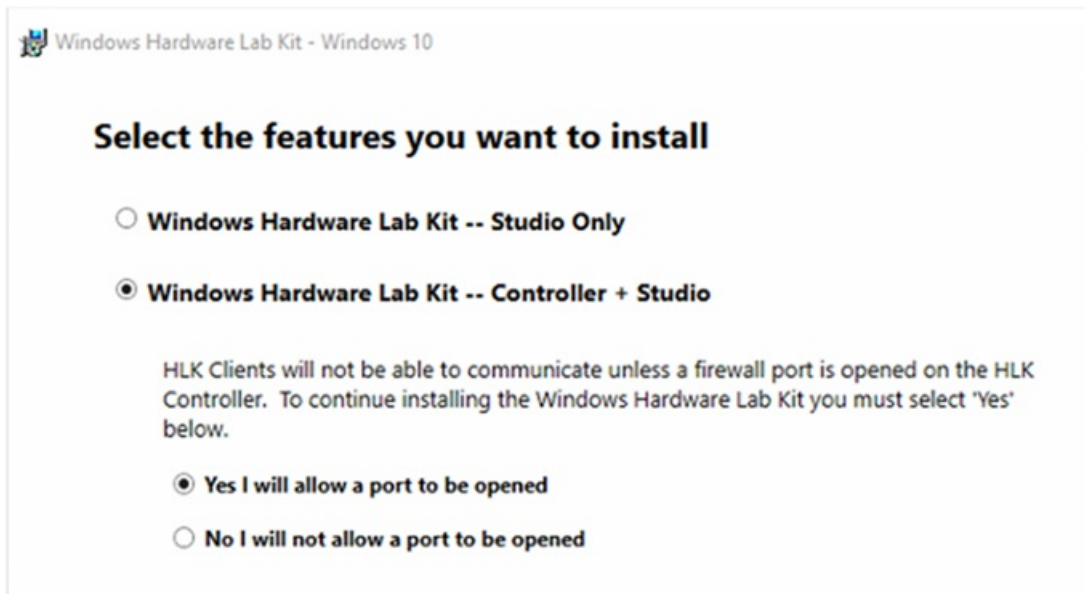
1. Download the Windows Hardware Lab Kit from one of the following locations:

- [Windows Hardware Lab Kit](#)
- Direct download link for Windows Server 2016: [Windows HLK, version 1607](#)

2. Accept the default installation path.



3. Under **Select the features you want to install**, select **Windows Hardware Lab Kit—Controller + Studio**.



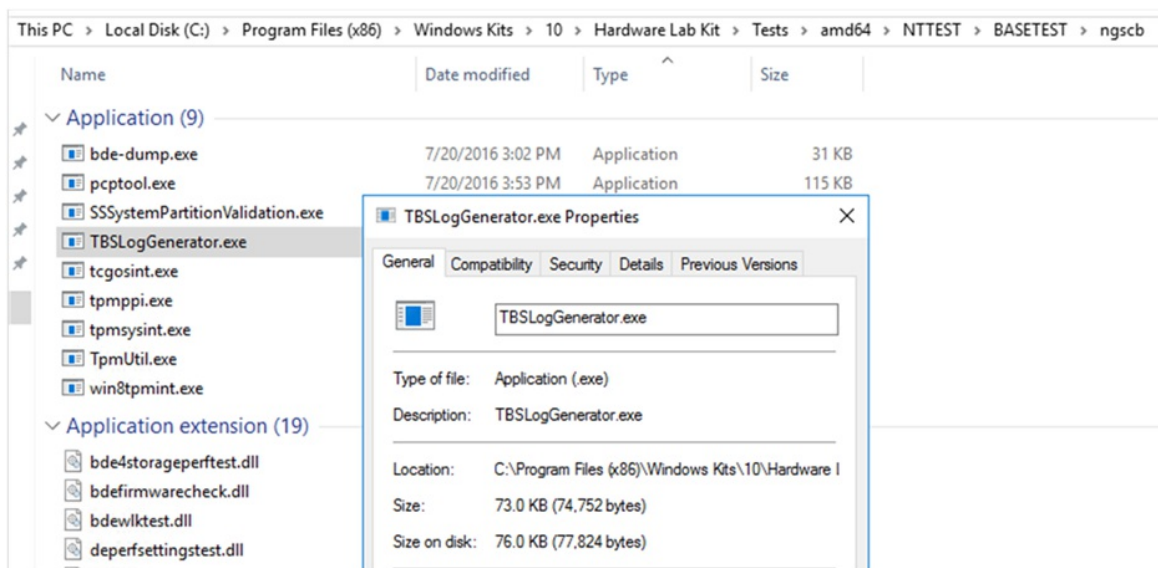
4. Finish the installation.

To use TBSLogGenerator, follow these steps:

1. After the installation finishes, open an elevated Command Prompt window and navigate to the following folder:

C:\Program Files (x86)\Windows Kits\10\Hardware Lab Kit\Tests\amd64\NTTEST\BASETEST\ngscb

This folder contains the TBSLogGenerator.exe file.



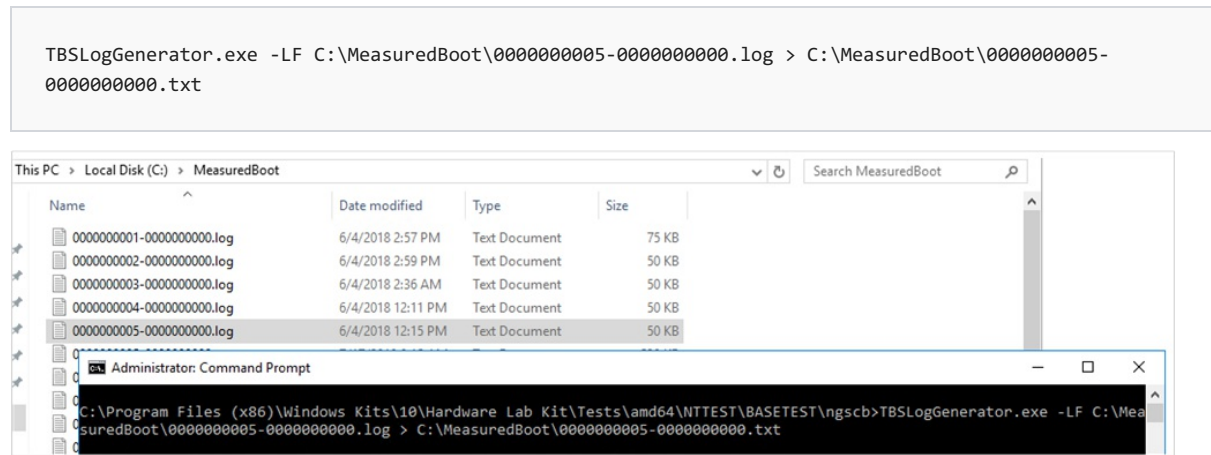
2. Run the following command:

```
TBSLogGenerator.exe -LF <LogFolderName>\<LogFileName>.log > <DestinationFolderName>\<DecodedFileName>.txt
```

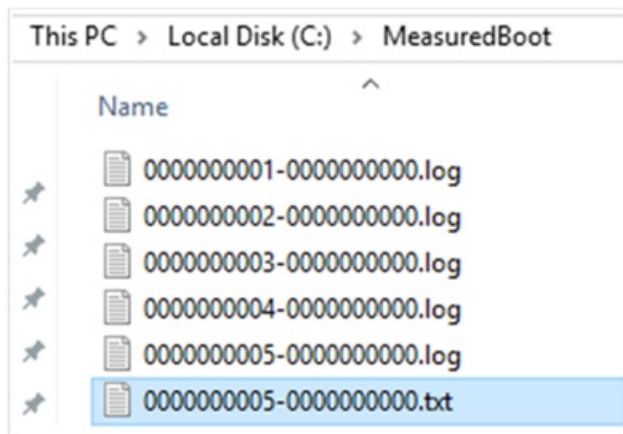
where the variables represent the following values:

- *<LogFolderName>* = the name of the folder that contains the file to be decoded
- *<LogFileName>* = the name of the file to be decoded
- *<DestinationFolderName>* = the name of the folder for the decoded text file
- *<DecodedFileName>* = the name of the decoded text file

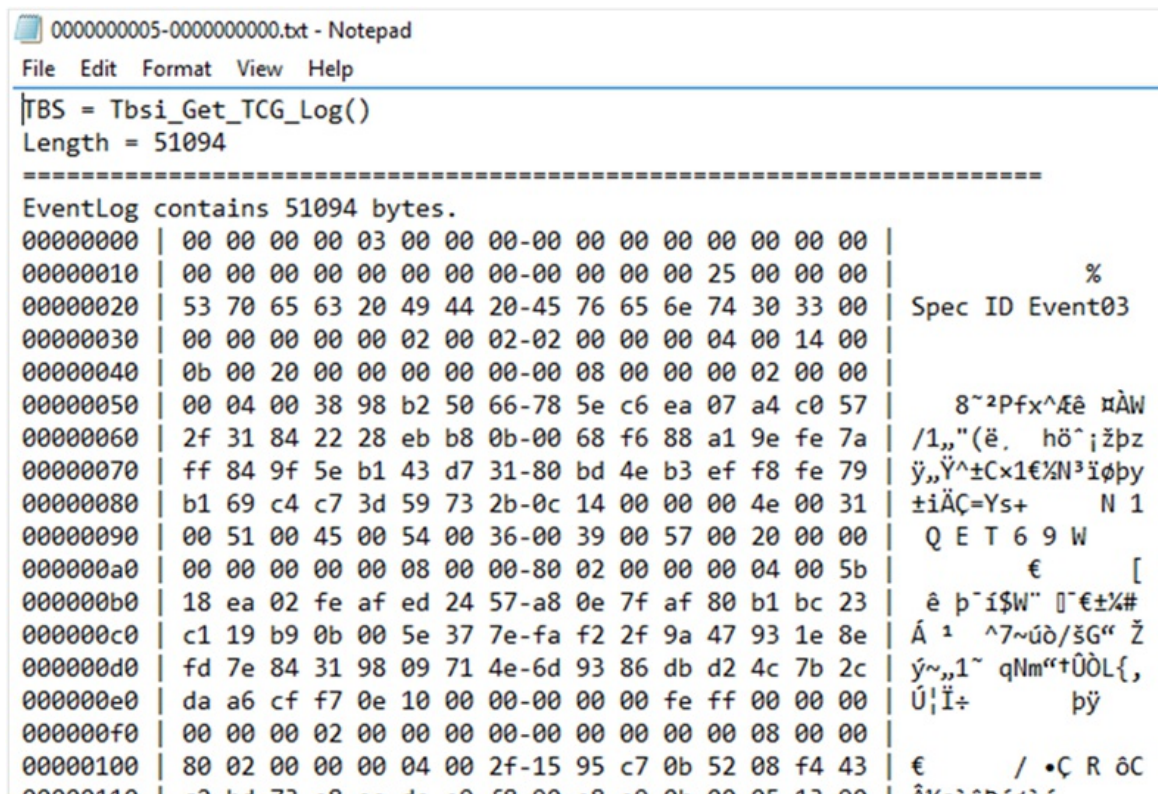
For example, the following figure shows Measured Boot logs that were collected from a Windows 10 computer and put into the C:\MeasuredBoot\ folder. The figure also shows a Command Prompt window and the command to decode the 000000005-000000000.log file:



The command produces a text file that uses the specified name. In the case of the example, the file is 000000005-000000000.txt. The file is located in the same folder as the original .log file.



The content of this text file resembles the following.



To find the PCR information, go to the end of the file.

```
000000e0 | f7 ac 1c 56 35 fd 5a 81-32 cd 75 81 2d f2 71 7a | +- V5ýZ2Íu-ðqz
000000f0 | 3e fa 4c 1a dc 91 84 10-99 dd 29 04 83 c0 a8 9c | >úL Ü´„ ”Ý) fÀ"œ
END AGGREGATION

-----
| PCR[00] | 100ffab3d44a795eb3415d9432ce9e478905853cb8a7c017ad31809b4ac22245 | INVALID! |
| TPM | [c1a83c2d518990c532797f9fc544587827173ff72ab56a182e700c60448ffc64] |
| PCR[01] | dd8b8ec0d7e6544dc79061c8ff89d1c4b2959a2b889abaea57cceb7e4e4b8c43 | INVALID! |
| TPM | [7f3842880eb9a9d050d4a71facda7897e868acea29ef707eb60216ae13bc5360] |
| PCR[02] | 50058edb288dae0f72e4fd6b6d3de351b78893e701ae98973cbe96b98e10e43 | INVALID! |
| TPM | [3d458cfe55cc03ea1f443f1562beec8df51c75e14a9fcf9a7234a13f198e7969] |
| PCR[03] | 3d458cfe55cc03ea1f443f1562beec8df51c75e14a9fcf9a7234a13f198e7969 | MATCH! |
| PCR[04] | 86ad8aaaecb8e2fe1d1a4c63a7d757989375045e4763f5a042e03dfe464587d | INVALID! |
| TPM | [18d808fa43a0a3ed530e174e6d4e3a228a90bac4ee665fc38de99bd37a355d61] |
| PCR[05] | 49ceb793c870db3dab47968291ef77f9ad80ad7cf952761154e2e3a1899c5811 | INVALID! |
| TPM | [675daac0ff10dc82aa862329c53a6906492c88057a0aa5d228b687e33971e8d4] |
| PCR[06] | 3d458cfe55cc03ea1f443f1562beec8df51c75e14a9fcf9a7234a13f198e7969 | MATCH! |
| PCR[07] | 730777cfa2b4c2cf67a54ce7c80d7d15ceb0a443d1bc320e43fe338812ea67b | INVALID! |
| TPM | [6dee7716b8b2bcd8d36ac7a3e37eed3f08c30a4af1890928ba48836ad640d19] |
| PCR[08] | 00000000000000000000000000000000000000000000000000000000000000 | MATCH! |
| PCR[09] | 00000000000000000000000000000000000000000000000000000000000000 | MATCH! |
| PCR[10] | 00000000000000000000000000000000000000000000000000000000000000 | MATCH! |
| PCR[11] | 0fe6e8f2110d5d53935c9e7d6f6bf722598b550595aabdc6e4fd2ecdf310f980 | MATCH! |
```

Use PCPTool to decode Measured Boot logs

NOTE

PCPTool is a Visual Studio solution, but you need to build the executable before you can start using this tool.

PCPTool is part of the [TPM Platform Crypto-Provider Toolkit](#). The tool decodes a Measured Boot log file and converts it into an XML file.

To download and install PCPTool, go to the Toolkit page, select **Download**, and follow the instructions.

To decode a log, run the following command:

```
PCPTool.exe decodeLog <LogFolderPath>\<LogFileName>.log > <DestinationFolderName>\<DecodedFileName>.xml
```

where the variables represent the following values:

- *<LogFolderPath>* = the path to the folder that contains the file to be decoded
- *<LogFileName>* = the name of the file to be decoded
- *<DestinationFolderName>* = the name of the folder for the decoded text file
- *<DecodedFileName>* = the name of the decoded text file

The content of the XML file resembles the following.

```
> .\PCPTool.exe decode log C:\Windows\Logs\MeasuredBoot\000000001-000000000.log > TPM_log.xml
```

```
129 | </EV_Event_Tag>
130 | <EV_Event_Tag PCR="13" EventDigest="3db51a2ab6859f67510849226e545d7a193b66eb" Size="760">
131 |   <Trustboundary Size="752">
132 |     <SipaEvent Type="0x00040002" Size="46">
133 |       00ad31d5660ed00120000000b00f4247831deed0fc371f8e8a6b0746ae7b508c3c924e5b1c485c14631afa33879
134 |       <!-- ..l.f.....xl....q....tj.....Fl..8y -->
135 |     </SipaEvent>
136 |     <LoadedModule_Aggregation Size="354">
137 |       <FilePath Size="86">\EFI\Microsoft\Boot\en-GB\bootmgfw.efi.MUI</FilePath>
138 |       <ImageSize Size="8">69632<!-- 0x0000000000011000 --></ImageSize>
139 |       <HashAlgorithmId Size="4">SHA-256</HashAlgorithmId>
140 |       <AuthenticodeHash Size="32">
141 |         alcd73aee5d6f2c87e57573f0e50b8ab7f512dcf7c544df8ae231d492d5835b8
142 |         <!-- ..s.....WW..P...Q...TM....I.X5. -->
143 |       </AuthenticodeHash>
144 |       <ImageValidated Size="1">TRUE</ImageValidated>
145 |       <AuthorityIssuer Size="76">Microsoft Windows Production PCA 2011</AuthorityIssuer>
146 |       <AuthorityPublisher Size="36">Microsoft Windows</AuthorityPublisher>
```

Configure S/MIME for Windows

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11

S/MIME stands for Secure/Multipurpose Internet Mail Extensions, and provides an added layer of security for email sent to and from an Exchange ActiveSync (EAS) account. S/MIME lets users encrypt outgoing messages and attachments so that only intended recipients who have a digital identification (ID), also known as a certificate, can read them. Users can digitally sign a message, which provides the recipients with a way to verify the identity of the sender and that the message hasn't been tampered with.

About message encryption

Users can send encrypted message to people in their organization and people outside their organization if they have their encryption certificates. However, users using Windows Mail app can only read encrypted messages if the message is received on their Exchange account and they have corresponding decryption keys.

Encrypted messages can be read only by recipients who have a certificate. If you try to send an encrypted message to recipient(s) whose encryption certificate are not available, the app will prompt you to remove these recipients before sending the email.

About digital signatures

A digitally signed message reassures the recipient that the message hasn't been tampered with and verifies the identity of the sender. Recipients can only verify the digital signature if they're using an email client that supports S/MIME.

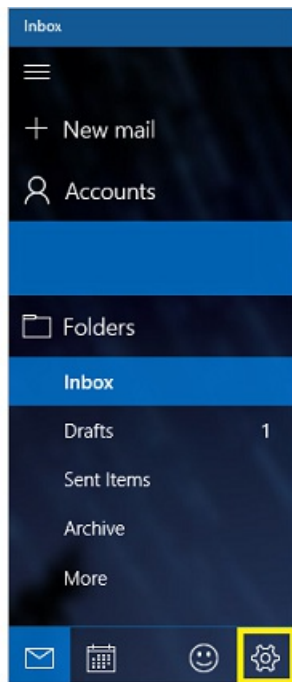
Prerequisites

- [S/MIME is enabled for Exchange accounts](#) (on-premises and Office 365). Users can't use S/MIME signing and encryption with a personal account such as Outlook.com.
- Valid Personal Information Exchange (PFX) certificates are installed on the device.
 - [How to Create PFX Certificate Profiles in Configuration Manager](#)
 - [Enable access to company resources using certificate profiles with Microsoft Intune](#)

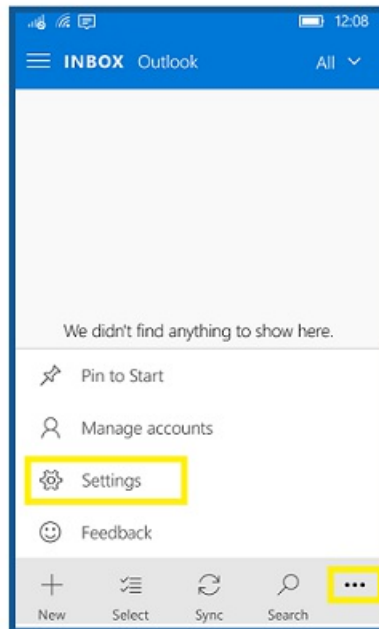
Choose S/MIME settings

On the device, perform the following steps: (add select certificate)

1. Open the Mail app.
2. Open **Settings** by tapping the gear icon on a PC, or the ellipsis (...) and then the gear icon on a phone.

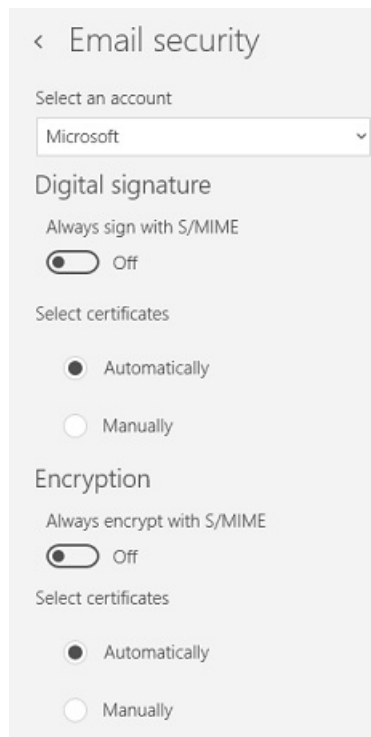


Desktop app



Phone app

3. Tap **Email security**.



4. In **Select an account**, select the account for which you want to configure S/MIME options.

5. Make a certificate selection for digital signature and encryption.

- Select **Automatically** to let the app choose the certificate.
- Select **Manually** to specify the certificate yourself from the list of valid certificates on the device.

6. (Optional) Select **Always sign with S/MIME**, **Always encrypt with S/MIME**, or both, to automatically digitally sign or encrypt all outgoing messages.

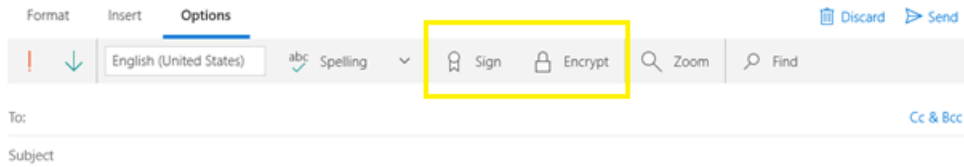
NOTE

The option to sign or encrypt can be changed for individual messages, unless EAS policies prevent it.

7. Tap the back arrow.

Encrypt or sign individual messages

1. While composing a message, choose **Options** from the ribbon. On phone, **Options** can be accessed by tapping the ellipsis (...).
2. Use **Sign** and **Encrypt** icons to turn on digital signature and encryption for this message.



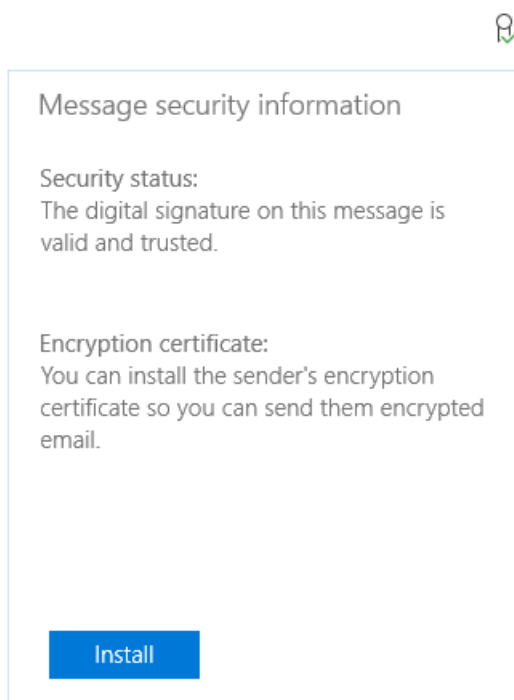
Read signed or encrypted messages

When you receive an encrypted message, the mail app will check whether there is a certificate available on your computer. If there is a certificate available, the message will be decrypted when you open it. If your certificate is stored on a smartcard, you will be prompted to insert the smartcard to read the message. Your smartcard may also require a PIN to access the certificate.

Install certificates from a received message

When you receive a signed email, the app provide feature to install corresponding encryption certificate on your device if the certificate is available. This certificate can then be used to send encrypted email to this person.

1. Open a signed email.
2. Tap or click the digital signature icon in the reading pane.
3. Tap **Install**.



Windows VPN technical guide

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11

This guide will walk you through the decisions you will make for Windows 10 or Windows 11 clients in your enterprise VPN solution and how to configure your deployment. This guide references the [VPNv2 Configuration Service Provider \(CSP\)](#) and provides mobile device management (MDM) configuration instructions using Microsoft Intune and the VPN Profile template for Windows 10 and Windows 11.

To create a Windows 10 VPN device configuration profile see: [Windows 10 and Windows Holographic device settings to add VPN connections using Intune](#).

NOTE

This guide does not explain server deployment.

In this guide

ARTICLE	DESCRIPTION
VPN connection types	Select a VPN client and tunneling protocol
VPN routing decisions	Choose between split tunnel and force tunnel configuration
VPN authentication options	Select a method for Extensible Authentication Protocol (EAP) authentication.
VPN and conditional access	Use Azure Active Directory policy evaluation to set access policies for VPN connections.
VPN name resolution	Decide how name resolution should work
VPN auto-triggered profile options	Set a VPN profile to connect automatically by app or by name, to be "always on", and to not trigger VPN on trusted networks
VPN security features	Configure traffic filtering, connect a VPN profile to Windows Information Protection (WIP), and more
VPN profile options	Combine settings into single VPN profile using XML

Learn more

- [Create VPN profiles to connect to VPN servers in Intune](#)

VPN connection types

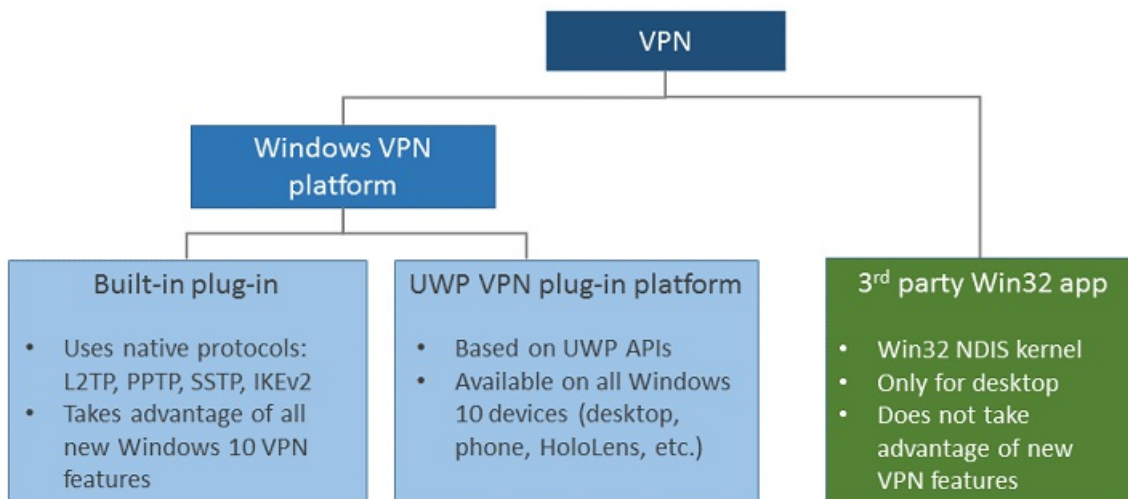
7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11

Virtual private networks (VPNs) are point-to-point connections across a private or public network, such as the Internet. A VPN client uses special TCP/IP or UDP-based protocols, called *tunneling protocols*, to make a virtual call to a virtual port on a VPN server. In a typical VPN deployment, a client initiates a virtual point-to-point connection to a remote access server over the Internet. The remote access server answers the call, authenticates the caller, and transfers data between the VPN client and the organization's private network.

There are many options for VPN clients. In Windows 10 and Windows 11, the built-in plug-in and the Universal Windows Platform (UWP) VPN plug-in platform are built on top of the Windows VPN platform. This guide focuses on the Windows VPN platform clients and the features that can be configured.



Built-in VPN client

- Tunneling protocols
 - [Internet Key Exchange version 2 \(IKEv2\)](#)

Configure the IPsec/IKE tunnel cryptographic properties using the **Cryptography Suite** setting in the [VPNv2 Configuration Service Provider \(CSP\)](#).

- [L2TP](#)

L2TP with pre-shared key (PSK) authentication can be configured using the **L2tpPsk** setting in the [VPNv2 CSP](#).

- [PPTP](#)
- [SSTP](#)

SSTP is supported for Windows desktop editions only. SSTP cannot be configured using mobile device management (MDM), but it is one of the protocols attempted in the **Automatic** option.

NOTE

When a VPN plug-in is used, the adapter will be listed as an SSTP adapter, even though the VPN protocol used is the plug-in's protocol.

- Automatic

The **Automatic** option means that the device will try each of the built-in tunneling protocols until one succeeds. It will attempt from most secure to least secure.

Configure **Automatic** for the **NativeProtocolType** setting in the [VPNv2 CSP](#).

Universal Windows Platform VPN plug-in

The Universal Windows Platform (UWP) VPN plug-ins were introduced in Windows 10 and Windows 11, although there was originally separate version available for the Windows 8.1 PC platform. Using the UWP platform, third-party VPN providers can create app-containerized plug-ins using WinRT APIs, eliminating the complexity and problems often associated with writing to system-level drivers.

There are a number of Universal Windows Platform VPN applications, such as Pulse Secure, Cisco AnyConnect, F5 Access, Sonicwall Mobile Connect, and Check Point Capsule. If you want to use a UWP VPN plug-in, work with your vendor for any custom settings needed to configure your VPN solution.

Configure connection type

See [VPN profile options](#) and [VPNv2 CSP](#) for XML configuration.

The following image shows connection options in a VPN Profile configuration policy using Microsoft Intune:

The screenshot displays the Microsoft Endpoint Manager admin center interface for configuring a VPN profile. The page is titled "VPN" and is for "Windows 10 and later". The "Configuration settings" tab is active, showing various configuration options. The "Connection type" dropdown menu is open, showing a list of available VPN plug-ins. The "Automatic" option is highlighted in blue, indicating it is the selected connection type. Other options in the list include Pulse Secure, F5 Access, SonicWall Mobile Connect, Check Point Capsule VPN, Citrix, Palo Alto Networks GlobalProtect, IKEv2, L2TP, and PPTP. The "Register IP addresses with internal DNS" option is set to "Disable".

In Intune, you can also include custom XML for third-party plug-in profiles:

Add Row

OMA-URI Settings


Name *

Description

OMA-URI *

Data type *

Custom XML *

Select a file 

1	<div style="border: 1px solid #ccc; height: 200px;"></div>
---	--

Related topics

- [VPN technical guide](#)
- [VPN routing decisions](#)
- [VPN authentication options](#)
- [VPN and conditional access](#)
- [VPN name resolution](#)
- [VPN auto-triggered profile options](#)
- [VPN security features](#)
- [VPN profile options](#)

VPN routing decisions

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11

Network routes are required for the stack to understand which interface to use for outbound traffic. One of the most important decision points for VPN configuration is whether you want to send all the data through VPN (*force tunnel*) or only some data through the VPN (*split tunnel*). This decision impacts the configuration and the capacity planning, as well as security expectations from the connection.

Split tunnel configuration

In a split tunnel configuration, routes can be specified to go over VPN and all other traffic will go over the physical interface.

Routes can be configured using the `VPNv2/ProfileName/RouteList` setting in the [VPNv2 Configuration Service Provider \(CSP\)](#).

For each route item in the list, the following can be specified:

- **Address:** `VPNv2/ProfileName/RouteList/routeRowId/Address`
- **Prefix size:** `VPNv2/ProfileName/RouteList/routeRowId/Prefix`
- **Exclusion route:** `VPNv2/ProfileName/RouteList/routeRowId/ExclusionRoute`

Windows VPN platform now supports the ability to specify exclusion routes that specifically should not go over the physical interface.

Routes can also be added at connect time through the server for UWP VPN apps.

Force tunnel configuration

In a force tunnel configuration, all traffic will go over VPN. This is the default configuration and takes effect if no routes are specified.

The only implication of this setting is the manipulation of routing entries. In the case of a force tunnel, VPN V4 and V6 default routes (for example. 0.0.0.0/0) are added to the routing table with a lower metric than ones for other interfaces. This sends traffic through the VPN as long as there isn't a specific route on the physical interface itself.

For built-in VPN, this decision is controlled using the MDM setting `VPNv2/ProfileName/NativeProfile/RoutingPolicyType`.

For a UWP VPN plug-in, this property is directly controlled by the app. If the VPN plug-in indicates the default route for IPv4 and IPv6 as the only two Inclusion routes, the VPN platform marks the connection as Force Tunneled.

Configure routing

See [VPN profile options](#) and [VPNv2 CSP](#) for XML configuration.

When you configure a VPN profile in Microsoft Intune, you select a checkbox to enable split tunnel configuration.

VPN Settings

Connection

* VPN connection name (displayed to users):

Connection type:

* Server list:

Server description	IP address or FQDN
--------------------	--------------------

Default server:

Enable split tunneling

Next, in **Corporate Boundaries**, you add the routes that should use the VPN connection.

Corporate Boundaries

Configure rules that specify the type of network traffic that can use this connection:

Rule name	Rule scope
-----------	------------

Specify the routes for this VPN connection (optional for third-party providers):

Destination Prefix	Prefix Size
--------------------	-------------

Add or edit VPN route

* Destination prefix (IPv4/v6 addresses):

* Prefix size:

Enter a valid route

Related topics

- [VPN technical guide](#)
- [VPN connection types](#)
- [VPN authentication options](#)
- [VPN and conditional access](#)
- [VPN name resolution](#)
- [VPN auto-triggered profile options](#)

- VPN security features
- VPN profile options

VPN authentication options

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11

In addition to older and less-secure password-based authentication methods (which should be avoided), the built-in VPN solution uses Extensible Authentication Protocol (EAP) to provide secure authentication using both user name and password, and certificate-based methods. You can only configure EAP-based authentication if you select a built-in VPN type (IKEv2, L2TP, PPTP or Automatic).

Windows supports a number of EAP authentication methods.

- EAP-Microsoft Challenge Handshake Authentication Protocol version 2 (EAP-MSCHAPv2):
 - User name and password authentication
 - Winlogon credentials - can specify authentication with computer sign-in credentials
- EAP-Transport Layer Security (EAP-TLS):
 - Supports the following types of certificate authentication:
 - Certificate with keys in the software Key Storage Provider (KSP)
 - Certificate with keys in Trusted Platform Module (TPM) KSP
 - Smart card certificates
 - Windows Hello for Business certificate
 - Certificate filtering:
 - Certificate filtering can be enabled to search for a particular certificate to use to authenticate with
 - Filtering can be Issuer-based or Enhanced Key Usage (EKU)-based
 - Server validation - with TLS, server validation can be toggled on or off:
 - Server name - specify the server to validate
 - Server certificate - trusted root certificate to validate the server
 - Notification - specify if the user should get a notification asking whether to trust the server or not
- [Protected Extensible Authentication Protocol \(PEAP\)](#):
 - Server validation - with PEAP, server validation can be toggled on or off:
 - Server name - specify the server to validate
 - Server certificate - trusted root certificate to validate the server
 - Notification - specify if the user should get a notification asking whether to trust the server or not
 - Inner method - the outer method creates a secure tunnel inside while the inner method is used to complete the authentication:
 - EAP-MSCHAPv2
 - EAP-TLS

- Fast Reconnect: reduces the delay between an authentication request by a client and the response by the Network Policy Server (NPS) or other Remote Authentication Dial-in User Service (RADIUS) server. This reduces resource requirements for both client and server, and minimizes the number of times that users are prompted for credentials.
- **Cryptobinding**: By deriving and exchanging values from the PEAP phase 1 key material (**Tunnel Key**) and from the PEAP phase 2 inner EAP method key material (**Inner Session Key**), it is possible to prove that the two authentications terminate at the same two entities (PEAP peer and PEAP server). This process, termed "cryptobinding", is used to protect the PEAP negotiation against "Man in the Middle" attacks.
- Tunneled Transport Layer Security (TTLS)
 - Inner method
 - Non-EAP
 - Password Authentication Protocol (PAP)
 - CHAP
 - MSCHAP
 - MSCHAPv2
 - EAP
 - MSCHAPv2
 - TLS
 - Server validation: in TTLS, the server must be validated. The following can be configured:
 - Server name
 - Trusted root certificate for server certificate
 - Whether there should be a server validation notification

For a UWP VPN plug-in, the app vendor controls the authentication method to be used. The following credential types can be used:

- Smart card
- Certificate
- Windows Hello for Business
- User name and password
- One-time password
- Custom credential type

Configure authentication

See [EAP configuration](#) for EAP XML configuration.

NOTE

To configure Windows Hello for Business authentication, follow the steps in [EAP configuration](#) to create a smart card certificate. [Learn more about Windows Hello for Business.](#)

The following image shows the field for EAP XML in a Microsoft Intune VPN profile. The EAP XML field only appears when you select a built-in connection type (automatic, IKEv2, L2TP, PPTP).

Authentication

Authentication method:
Certificates

Remember the user credentials at each logon

* Select a client certificate for client authentication (Identity Certificate):
 Select...

Enable conditional access for this VPN connection

* EAP XML:

Related topics

- [VPN technical guide](#)
- [VPN connection types](#)
- [VPN routing decisions](#)
- [VPN and conditional access](#)
- [VPN name resolution](#)
- [VPN auto-triggered profile options](#)
- [VPN security features](#)
- [VPN profile options](#)

VPN and conditional access

7/1/2022 • 5 minutes to read • [Edit Online](#)

Applies to: Windows 10 and Windows 11

The VPN client is now able to integrate with the cloud-based Conditional Access Platform to provide a device compliance option for remote clients. Conditional Access is a policy-based evaluation engine that lets you create access rules for any Azure Active Directory (Azure AD) connected application.

NOTE

Conditional Access is an Azure AD Premium feature.

Conditional Access Platform components used for Device Compliance include the following cloud-based services:

- [Conditional Access Framework](#)
- [Azure AD Connect Health](#)
- [Windows Health Attestation Service](#) (optional)
- Azure AD Certificate Authority - It is a requirement that the client certificate used for the cloud-based device compliance solution be issued by an Azure Active Directory-based Certificate Authority (CA). An Azure AD CA is essentially a mini-CA cloud tenant in Azure. The Azure AD CA cannot be configured as part of an on-premises Enterprise CA. See also [Always On VPN deployment for Windows Server and Windows 10](#).
- Azure AD-issued short-lived certificates - When a VPN connection attempt is made, the Azure AD Token Broker on the local device communicates with Azure Active Directory, which then checks for health based on compliance rules. If compliant, Azure AD sends back a short-lived certificate that is used to authenticate the VPN. Note that certificate authentication methods such as EAP-TLS can be used. When that certificate expires, the client will again check with Azure AD for health validation before a new certificate is issued.
- [Microsoft Intune device compliance policies](#) - Cloud-based device compliance leverages Microsoft Intune Compliance Policies, which are capable of querying the device state and define compliance rules for the following, among other things.
 - Antivirus status
 - Auto-update status and update compliance
 - Password policy compliance
 - Encryption compliance
 - Device health attestation state (validated against attestation service after query)

The following client-side components are also required:

- [HealthAttestation Configuration Service Provider \(CSP\)](#)
- [VPNv2 CSP](#) DeviceCompliance node settings
- Trusted Platform Module (TPM)

VPN device compliance

At this time, the Azure AD certificates issued to users do not contain a CRL Distribution Point (CDP) and are not suitable for Key Distribution Centers (KDCs) to issue Kerberos tokens. For users to gain access to on-premises resources such as files on a network share, client authentication certificates must be deployed to the Windows profiles of the users, and their VPNv2 profiles must contain the <SSO> section.

Server-side infrastructure requirements to support VPN device compliance include:

- The VPN server should be configured for certificate authentication.
- The VPN server should trust the tenant-specific Azure AD CA.
- For client access using Kerberos/NTLM, a domain-trusted certificate is deployed to the client device and is configured to be used for single sign-on (SSO).

After the server side is set up, VPN admins can add the policy settings for conditional access to the VPN profile using the VPNv2 DeviceCompliance node.

Two client-side configuration service providers are leveraged for VPN device compliance.

- VPNv2 CSP DeviceCompliance settings:
 - **Enabled**: enables the Device Compliance flow from the client. If marked as **true**, the VPN client attempts to communicate with Azure AD to get a certificate to use for authentication. The VPN should be set up to use certificate authentication and the VPN server must trust the server returned by Azure AD.
 - **Sso**: entries under SSO should be used to direct the VPN client to use a certificate other than the VPN authentication certificate when accessing resources that require Kerberos authentication.
 - **Sso/Enabled**: if this field is set to **true**, the VPN client looks for a separate certificate for Kerberos authentication.
 - **Sso/IssuerHash**: hashes for the VPN client to look for the correct certificate for Kerberos authentication.
 - **Sso/Eku**: comma-separated list of Enhanced Key Usage (EKU) extensions for the VPN client to look for the correct certificate for Kerberos authentication.
- HealthAttestation CSP (not a requirement) - functions performed by the HealthAttestation CSP include:
 - Collects TPM data used to verify health states
 - Forwards the data to the Health Attestation Service (HAS)
 - Provisions the Health Attestation Certificate received from the HAS
 - Upon request, forward the Health Attestation Certificate (received from HAS) and related runtime information to the MDM server for verification

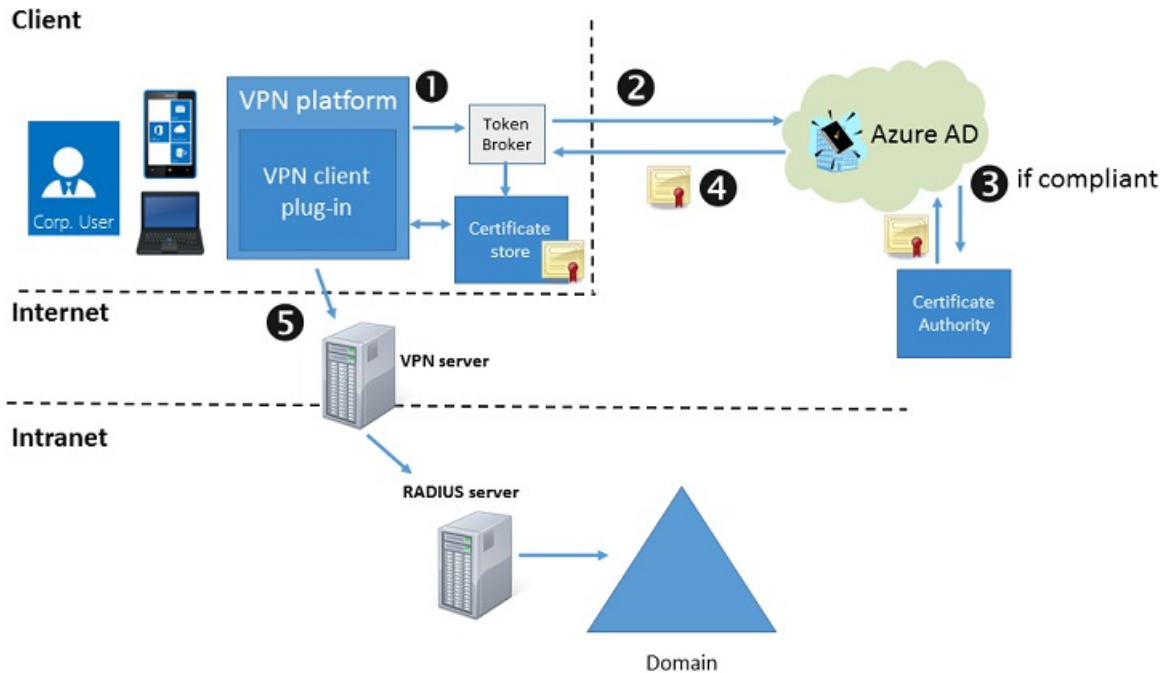
NOTE

Currently, it is required that certificates used for obtaining Kerberos tickets must be issued from an on-premises CA, and that SSO must be enabled in the user's VPN profile. This will enable the user to access on-premises resources.

In the case of AzureAD-only joined devices (not hybrid joined devices), if the user certificate issued by the on-premises CA has the user UPN from AzureAD in Subject and SAN (Subject Alternative Name), the VPN profile must be modified to ensure that the client does not cache the credentials used for VPN authentication. To do this, after deploying the VPN profile to the client, modify the *Rasphone.pbk* on the client by changing the entry **UseRasCredentials** from 1 (default) to 0 (zero).

Client connection flow

The VPN client side connection flow works as follows:



When a VPNv2 Profile is configured with `<DeviceCompliance> <Enabled>true</Enabled>` the VPN client uses this connection flow:

1. The VPN client calls into Windows 10's or Windows 11's Azure AD Token Broker, identifying itself as a VPN client.
2. The Azure AD Token Broker authenticates to Azure AD and provides it with information about the device trying to connect. The Azure AD Server checks if the device is in compliance with the policies.
3. If compliant, Azure AD requests a short-lived certificate.
4. Azure AD pushes down a short-lived certificate to the Certificate Store via the Token Broker. The Token Broker then returns control back over to the VPN client for further connection processing.
5. The VPN client uses the Azure AD-issued certificate to authenticate with the VPN server.

Configure conditional access

See [VPN profile options](#) and [VPNv2 CSP](#) for XML configuration.

Learn more about Conditional Access and Azure AD Health

- [Azure Active Directory conditional access](#)
- [Getting started with Azure Active Directory Conditional Access](#)
- [Control the health of Windows 10-based devices](#)
- [Control the health of Windows 11-based devices](#)
- [Tip of the Day: The Conditional Access Framework and Device Compliance for VPN \(Part 1\)](#)
- [Tip of the Day: The Conditional Access Framework and Device Compliance for VPN \(Part 2\)](#)
- [Tip of the Day: The Conditional Access Framework and Device Compliance for VPN \(Part 3\)](#)
- [Tip of the Day: The Conditional Access Framework and Device Compliance for VPN \(Part 4\)](#)

Related topics

- [VPN technical guide](#)
- [VPN connection types](#)
- [VPN routing decisions](#)
- [VPN authentication options](#)
- [VPN name resolution](#)
- [VPN auto-triggered profile options](#)
- [VPN security features](#)
- [VPN profile options](#)

VPN name resolution

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11

When the VPN client connects to the VPN server, the VPN client receives the client IP address. The client may also receive the IP address of the Domain Name System (DNS) server and the IP address of the Windows Internet Name Service (WINS) server.

The name resolution setting in the VPN profile configures how name resolution should work on the system when VPN is connected. The networking stack first looks at the Name Resolution Policy table (NRPT) for any matches and tries a resolution in the case of a match. If no match is found, the DNS suffix on the most preferred interface based on the interface metric is appended to the name (in the case of a short name) and a DNS query is sent out on the preferred interface. If the query times out, the DNS suffix search list is used in order and DNS queries are sent on all interfaces.

Name Resolution Policy table (NRPT)

The NRPT is a table of namespaces that determines the DNS client's behavior when issuing name resolution queries and processing responses. It is the first place that the stack will look after the DNSCache.

There are 3 types of name matches that can set up for NRPT:

- Fully qualified domain name (FQDN) that can be used for direct matching to a name
- Suffix match results in either a comparison of suffixes (for FQDN resolution) or the appending of the suffix (in case of a short name)
- Any resolution should attempt to first resolve with the proxy server/DNS server with this entry

NRPT is set using the `VPNv2/ProfileName/DomainNameInformationList` node of the [VPNv2 CSP](#). This node also configures Web proxy server or domain name servers.

[Learn more about NRPT](#)

DNS suffix

This setting is used to configure the primary DNS suffix for the VPN interface and the suffix search list after the VPN connection is established.

Primary DNS suffix is set using the `VPNv2/ProfileName/DnsSuffix` node.

[Learn more about primaryDNS suffix](#)

Persistent

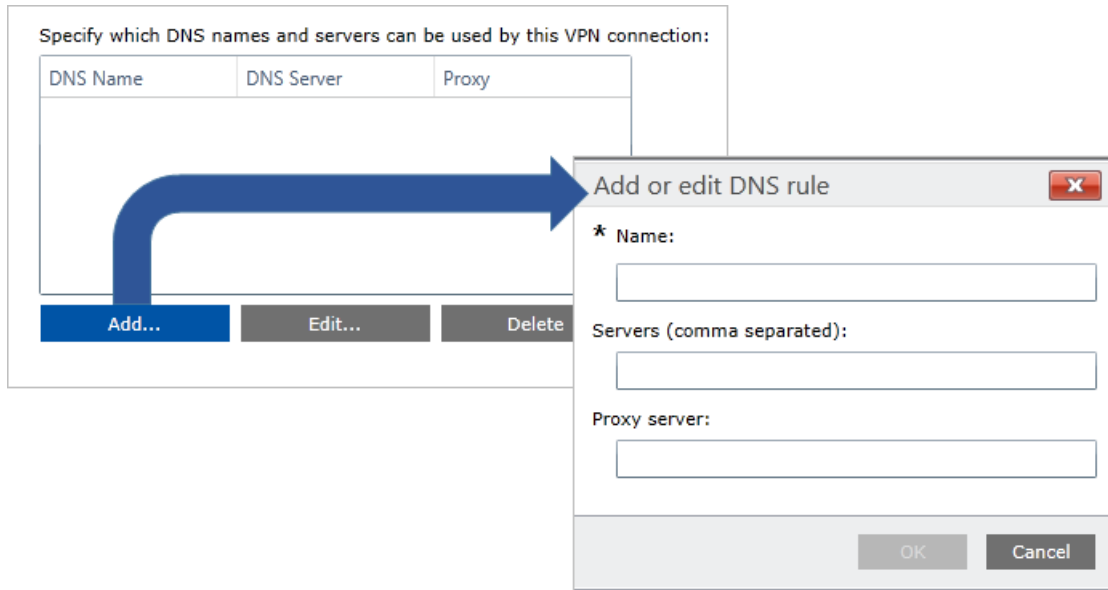
You can also configure *persistent* name resolution rules. Name resolution for specified items will only be performed over the VPN.

Persistent name resolution is set using the `VPNv2/ProfileName/DomainNameInformationList//dniRowId/Persistent` node.

Configure name resolution

See [VPN profile options](#) and [VPNv2 CSP](#) for XML configuration.

The following image shows name resolution options in a VPN Profile configuration policy using Microsoft Intune.



The fields in **Add or edit DNS rule** in the Intune profile correspond to the XML settings shown in the following table.

FIELD	XML
Name	<code>VPNv2/ProfileName/DomainNameInformationList/dn iRowId/DomainName</code>
Servers (comma separated)	<code>VPNv2/ProfileName/DomainNameInformationList/dn iRowId/DnsServers</code>
Proxy server	<code>VPNv2/ProfileName/DomainNameInformationList/dn iRowId/WebServers</code>

Related topics

- [VPN technical guide](#)
- [VPN connection types](#)
- [VPN routing decisions](#)
- [VPN authentication options](#)
- [VPN and conditional access](#)
- [VPN auto-triggered profile options](#)
- [VPN security features](#)
- [VPN profile options](#)

VPN auto-triggered profile options

7/1/2022 • 3 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11

In Windows 10 and Windows 11, a number of features have been added to auto-trigger VPN so users won't have to manually connect when VPN is needed to access necessary resources. There are three different types of auto-trigger rules:

- App trigger
- Name-based trigger
- Always On

NOTE

Auto-triggered VPN connections will not work if Folder Redirection for AppData is enabled. Either Folder Redirection for AppData must be disabled or the auto-triggered VPN profile must be deployed in system context, which changes the path to where the rasphone.pbk file is stored.

App trigger

VPN profiles in Windows 10 or Windows 11 can be configured to connect automatically on the launch of a specified set of applications. You can configure desktop or Universal Windows Platform (UWP) apps to trigger a VPN connection. You can also configure per-app VPN and specify traffic rules for each app. See [Traffic filters](#) for more details.

The app identifier for a desktop app is a file path. The app identifier for a UWP app is a package family name.

[Find a package family name \(PFN\) for per-app VPN configuration](#)

Name-based trigger

You can configure a domain name-based rule so that a specific domain name triggers the VPN connection.

Name-based auto-trigger can be configured using the `VPNv2/ProfileName/DomainNameInformationList/dniRowId/AutoTrigger` setting in the [VPNv2 Configuration Service Provider \(CSP\)](#).

There are four types of name-based triggers:

- Short name: for example, if **HRweb** is configured as a trigger and the stack sees a DNS resolution request for **HRweb**, the VPN will be triggered.
- Fully-qualified domain name (FQDN): for example, if **HRweb.corp.contoso.com** is configured as a trigger and the stack sees a DNS resolution request for **HRweb.corp.contoso.com**, the VPN will be triggered.
- Suffix: for example, if **.corp.contoso.com** is configured as a trigger and the stack sees a DNS resolution request with a matching suffix (such as **HRweb.corp.contoso.com**), the VPN will be triggered. For any short name resolution, VPN will be triggered and the DNS server will be queried for the *ShortName.corp.contoso.com*.

- All: if used, all DNS resolution should trigger VPN.

Always On

Always On is a feature in Windows 10 and Windows 11 which enables the active VPN profile to connect automatically on the following triggers:

- User sign-in
- Network change
- Device screen on

When the trigger occurs, VPN tries to connect. If an error occurs or any user input is needed, the user is shown a toast notification for additional interaction.

When a device has multiple profiles with Always On triggers, the user can specify the active profile in **Settings > Network & Internet > VPN > VPN profile** by selecting the **Let apps automatically use this VPN connection** checkbox. By default, the first MDM-configured profile is marked as **Active**. Devices with multiple users have the same restriction: only one profile and therefore only one user will be able to use the Always On triggers.

Preserving user Always On preference

Windows has a feature to preserve a user's AlwaysOn preference. In the event that a user manually unchecks the "Connect automatically" checkbox, Windows will remember this user preference for this profile name by adding the profile name to the value **AutoTriggerDisabledProfilesList**.

Should a management tool remove or add the same profile name back and set **AlwaysOn** to **true**, Windows will not check the box if the profile name exists in the following registry value in order to preserve user preference.

Key: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\RasMan\Config

Value: AutoTriggerDisabledProfilesList

Type: REG_MULTI_SZ

Trusted network detection

This feature configures the VPN such that it would not get triggered if a user is on a trusted corporate network. The value of this setting is a list of DNS suffixes. The VPN stack will look at the network name of the physical interface connection profile and if it matches any in the configured list and the network is private or provisioned by MDM, then VPN will not get triggered.

Trusted network detection can be configured using the `VPNv2/ProfileName/TrustedNetworkDetection` setting in the [VPNv2 CSP](#).

Configure app-triggered VPN

See [VPN profile options](#) and [VPNv2 CSP](#) for XML configuration.

The following image shows associating an app to a VPN connection in a VPN Profile configuration policy using Microsoft Intune.

Associated Apps

The following apps automatically use this VPN connection:

Name	Type
Microsoft.WindowsAlarms_8wekyb3d8bbwe!App	Universal

Add... **Edit...** **Delete**

Only these apps can use this VPN connection (per-app VPN):

Add or edit an App for the VPN connection ✕

* Choose app type:

Universal

* Enter the app identifier:

Microsoft.Windows.Cortana_cw5n1h2tyewy!Cortana

OK **Cancel**

After you add an associated app, if you select the **Only these apps can use this VPN connection (per-app VPN)** checkbox, the app becomes available in **Corporate Boundaries**, where you can configure rules for the app. See [Traffic filters](#) for more details.

Associated Apps

The following apps automatically use this VPN connection:

Name	Type
Microsoft.Windows.Cortana_cw5n1h2tyewy!Cortana	Universal

Add... **Edit...** **Delete**

Only these apps can use this VPN connection (per-app VPN):

Corporate Boundaries

Configure rules that specify the type of network traffic that can use this connection:

Rule name	Rule scope
App Rule: Microsoft.Windows.Co	App

Add... **Edit...** **Delete**

Related topics

- [VPN technical guide](#)
- [VPN connection types](#)
- [VPN routing decisions](#)
- [VPN authentication options](#)
- [VPN and conditional access](#)
- [VPN name resolution](#)
- [VPN security features](#)
- [VPN profile options](#)

VPN security features

7/1/2022 • 3 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11

Windows Information Protection (WIP) integration with VPN

Windows Information Protection provides capabilities allowing the separation and protection of enterprise data against disclosure across both company and personally owned devices, without requiring additional changes to the environments or the apps themselves. Additionally, when used with Rights Management Services (RMS), WIP can help to protect enterprise data locally.

The **EdpModelId** node in the [VPNv2 Configuration Service Provider \(CSP\)](#) allows a Windows 10 or Windows 11 VPN client to integrate with WIP, extending its functionality to remote devices. Use case scenarios for WIP include:

- Core functionality: File encryption and file access blocking
- UX policy enforcement: Restricting copy/paste, drag/drop, and sharing operations
- WIP network policy enforcement: Protecting intranet resources over the corporate network and VPN
- Network policy enforcement: Protecting SMB and Internet cloud resources over the corporate network and VPN

The value of the **EdpModelId** is an Enterprise ID. The networking stack will look for this ID in the app token to determine whether VPN should be triggered for that particular app.

Additionally, when connecting with WIP, the admin does not have to specify `AppTriggerList` and `TrafficFilterList` rules separately in this profile (unless more advanced configuration is needed) because the WIP policies and App lists automatically take effect.

[Learn more about Windows Information Protection](#)

Traffic Filters

Traffic Filters give enterprises the ability to decide what traffic is allowed into the corporate network based on policy. Network admins can use Traffic Filters to effectively add interface specific firewall rules on the VPN Interface. There are two types of Traffic Filter rules:

- App-based rules. With app-based rules, a list of applications can be marked to allow only traffic originating from these apps to go over the VPN interface.
- Traffic-based rules. Traffic-based rules are 5-tuple policies (ports, addresses, protocol) that can be specified to allow only traffic matching these rules to go over the VPN interface.

There can be many sets of rules which are linked by OR. Within each set, there can be app-based rules and traffic-based rules; all the properties within the set will be linked by AND. In addition, these rules can be applied at a per-app level or a per-device level.

For example, an admin could define rules that specify:

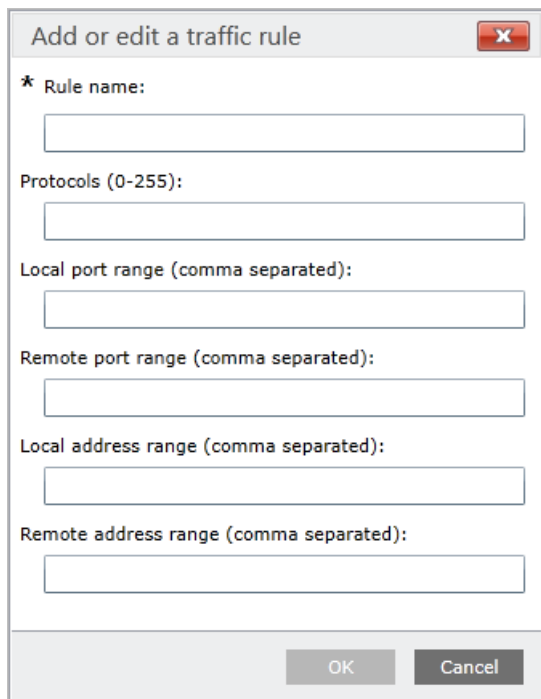
- The Contoso HR App must be allowed to go through the VPN and only access port 4545.

- The Contoso finance apps are allowed to go over the VPN and only access the Remote IP ranges of 10.10.0.40 - 10.10.0.201 on port 5889.
- All other apps on the device should be able to access only ports 80 or 443.

Configure traffic filters

See [VPN profile options](#) and [VPNv2 CSP](#) for XML configuration.

The following image shows the interface to configure traffic rules in a VPN Profile configuration policy, using Microsoft Intune.



The screenshot shows a dialog box titled "Add or edit a traffic rule" with a close button (X) in the top right corner. The dialog contains the following fields and labels:

- * Rule name:** A text input field.
- Protocols (0-255):** A text input field.
- Local port range (comma separated):** A text input field.
- Remote port range (comma separated):** A text input field.
- Local address range (comma separated):** A text input field.
- Remote address range (comma separated):** A text input field.

At the bottom of the dialog, there are two buttons: "OK" and "Cancel".

LockDown VPN

A VPN profile configured with LockDown secures the device to only allow network traffic over the VPN interface. It has the following features:

- The system attempts to keep the VPN connected at all times.
- The user cannot disconnect the VPN connection.
- The user cannot delete or modify the VPN profile.
- The VPN LockDown profile uses forced tunnel connection.
- If the VPN connection is not available, outbound network traffic is blocked.
- Only one VPN LockDown profile is allowed on a device.

NOTE

For built-in VPN, LockDown VPN is only available for the Internet Key Exchange version 2 (IKEv2) connection type.

Deploy this feature with caution, as the resultant connection will not be able to send or receive any network traffic without the VPN being connected.

Related topics

- [VPN technical guide](#)
- [VPN connection types](#)

- VPN routing decisions
- VPN authentication options
- VPN and conditional access
- VPN name resolution
- VPN auto-triggered profile options
- VPN profile options

VPN profile options

7/1/2022 • 5 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11

Most of the VPN settings in Windows 10 and Windows 11 can be configured in VPN profiles using Microsoft Intune or Microsoft Endpoint Configuration Manager. All VPN settings in Windows 10 and Windows 11 can be configured using the **ProfileXML** node in the [VPNv2 configuration service provider \(CSP\)](#).

NOTE

If you're not familiar with CSPs, read [Introduction to configuration service providers \(CSPs\)](#) first.

The following table lists the VPN settings and whether the setting can be configured in Intune and Configuration Manager, or can only be configured using **ProfileXML**.

PROFILE SETTING	CAN BE CONFIGURED IN INTUNE AND CONFIGURATION MANAGER
Connection type	Yes
Routing: split-tunnel routes	Yes, except exclusion routes
Routing: forced-tunnel	Yes
Authentication (EAP)	Yes, if connection type is built in
Conditional access	Yes
Name resolution: NRPT	Yes
Name resolution: DNS suffix	No
Name resolution: persistent	No
Auto-trigger: app trigger	Yes
Auto-trigger: name trigger	Yes
Auto-trigger: Always On	Yes
Auto-trigger: trusted network detection	No
LockDown	No
Windows Information Protection (WIP)	Yes

PROFILE SETTING	CAN BE CONFIGURED IN INTUNE AND CONFIGURATION MANAGER
Traffic filters	Yes
Proxy settings	Yes, by PAC/WPAD file or server and port

NOTE

VPN proxy settings are only used on Force Tunnel Connections. On Split Tunnel Connections, the general proxy settings are used.

The ProfileXML node was added to the VPNv2 CSP to allow users to deploy VPN profile as a single blob. This node is useful for deploying profiles with features that aren't yet supported by MDMs. You can get more examples in the [ProfileXML XSD](#) article.

Sample Native VPN profile

The following sample is a sample Native VPN profile. This blob would fall under the ProfileXML node.

```
<VPNProfile>
  <ProfileName>TestVpnProfile</ProfileName>
  <NativeProfile>
    <Servers>testServer.VPN.com</Servers>
    <NativeProtocolType>IKEv2</NativeProtocolType>

    <!--Sample EAP profile (PEAP)-->
    <Authentication>
      <UserMethod>Eap</UserMethod>
      <Eap>
        <Configuration>
          <EapHostConfig xmlns="http://www.microsoft.com/provisioning/EapHostConfig">
            <EapMethod>
              <Type xmlns="http://www.microsoft.com/provisioning/EapCommon">25</Type>
              <VendorId xmlns="http://www.microsoft.com/provisioning/EapCommon">0</VendorId>
              <VendorType xmlns="http://www.microsoft.com/provisioning/EapCommon">0</VendorType>
              <AuthorId xmlns="http://www.microsoft.com/provisioning/EapCommon">0</AuthorId>
            </EapMethod>
            <Config xmlns="http://www.microsoft.com/provisioning/EapHostConfig">
              <Eap xmlns="http://www.microsoft.com/provisioning/BaseEapConnectionPropertiesV1">
                <Type>25</Type>
                <EapType xmlns="http://www.microsoft.com/provisioning/MsPeapConnectionPropertiesV1">
                  <ServerValidation>
                    <DisableUserPromptForServerValidation>true</DisableUserPromptForServerValidation>
                    <ServerNames></ServerNames>
                    <TrustedRootCA>d2 d3 8e ba 60 ca a1 c1 20 55 a2 e1 c8 3b 15 ad 45 01 10 c2
</TrustedRootCA>
                    <TrustedRootCA>d1 76 97 cc 20 6e d2 6e 1a 51 f5 bb 96 e9 35 6d 6d 61 0b 74
</TrustedRootCA>
                  </ServerValidation>
                  <FastReconnect>true</FastReconnect>
                  <InnerEapOptional>false</InnerEapOptional>
                <Eap xmlns="http://www.microsoft.com/provisioning/BaseEapConnectionPropertiesV1">
                  <Type>13</Type>
                  <EapType xmlns="http://www.microsoft.com/provisioning/EapTlsConnectionPropertiesV1">
                    <CredentialsSource>
                      <CertificateStore>
                        <SimpleCertSelection>true</SimpleCertSelection>
                      </CertificateStore>
                    </CredentialsSource>
                    <ServerValidation>
                      <DisableUserPromptForServerValidation>true</DisableUserPromptForServerValidation>

```

```

        <ServerNames></ServerNames>
        <TrustedRootCA>d2 d3 8e ba 60 ca a1 c1 20 55 a2 e1 c8 3b 15 ad 45 01 10 c2
</TrustedRootCA>
        <TrustedRootCA>d1 76 97 cc 20 6e d2 6e 1a 51 f5 bb 96 e9 35 6d 6d 61 0b 74
</TrustedRootCA>
        </ServerValidation>
        <DifferentUsername>>false</DifferentUsername>
        <PerformServerValidation
xmlns="http://www.microsoft.com/provisioning/EapTlsConnectionPropertiesV2">true</PerformServerValidation>
        <AcceptServerName
xmlns="http://www.microsoft.com/provisioning/EapTlsConnectionPropertiesV2">>false</AcceptServerName>
        <TLSExtensions
xmlns="http://www.microsoft.com/provisioning/EapTlsConnectionPropertiesV2">
        <FilteringInfo
xmlns="http://www.microsoft.com/provisioning/EapTlsConnectionPropertiesV3">
            <EKUMapping>
                <EKUMap>
                    <EKUName>AAD Conditional Access</EKUName>
                    <EKUOID>1.3.6.1.4.1.311.87</EKUOID>
                </EKUMap>
            </EKUMapping>
            <ClientAuthEKUList Enabled="true">
                <EKUMapInList>
                    <EKUName>AAD Conditional Access</EKUName>
                </EKUMapInList>
            </ClientAuthEKUList>
        </FilteringInfo>
        </TLSExtensions>
    </EapType>
</Eap>
    <EnableQuarantineChecks>>false</EnableQuarantineChecks>
    <RequireCryptoBinding>true</RequireCryptoBinding>
    <PeapExtensions>
        <PerformServerValidation
xmlns="http://www.microsoft.com/provisioning/MsPeapConnectionPropertiesV2">true</PerformServerValidation>
        <AcceptServerName
xmlns="http://www.microsoft.com/provisioning/MsPeapConnectionPropertiesV2">>false</AcceptServerName>
    </PeapExtensions>
</EapType>
</Eap>
</Config>
</EapHostConfig>
</Configuration>
</Eap>
</Authentication>

    <!--Sample routing policy: in this case, this is a split tunnel configuration with two routes
configured-->
    <RoutingPolicyType>SplitTunnel</RoutingPolicyType>
    <DisableClassBasedDefaultRoute>true</DisableClassBasedDefaultRoute>
</NativeProfile>
    <Route>
        <Address>192.168.0.0</Address>
        <PrefixSize>24</PrefixSize>
    </Route>
    <Route>
        <Address>10.10.0.0</Address>
        <PrefixSize>16</PrefixSize>
    </Route>

    <!--VPN will be triggered for the two apps specified here-->
    <AppTrigger>
        <App>
            <Id>Microsoft.MicrosoftEdge_8wekyb3d8bbwe</Id>
        </App>
    </AppTrigger>
    <AppTrigger>
        <App>
            <Id>C:\windows\system32\ping.exe</Id>

```

```

</App>
</AppTrigger>

<!--Example of per-app VPN. This configures traffic filtering rules for two apps. Internet Explorer is
configured for force tunnel, meaning that all traffic allowed through this app must go over VPN. Microsoft
Edge is configured as split tunnel, so whether data goes over VPN or the physical interface is dictated by
the routing configuration.-->
<TrafficFilter>
  <App>
    <Id>%ProgramFiles%\Internet Explorer\iexplore.exe</Id>
  </App>
  <Protocol>6</Protocol>
  <LocalPortRanges>10,20-50,100-200</LocalPortRanges>
  <RemotePortRanges>20-50,100-200,300</RemotePortRanges>
  <RemoteAddressRanges>30.30.0.0/16,10.10.10.10-20.20.20.20</RemoteAddressRanges>
  <RoutingPolicyType>ForceTunnel</RoutingPolicyType>
</TrafficFilter>
<TrafficFilter>
  <App>
    <Id>Microsoft.MicrosoftEdge_8wekyb3d8bbwe</Id>
  </App>
  <LocalAddressRanges>3.3.3.3/32,1.1.1.1-2.2.2.2</LocalAddressRanges>
</TrafficFilter>

<!--Name resolution configuration. The AutoTrigger node configures name-based triggering. In this profile,
the domain "hrsite.corporate.contoso.com" triggers VPN.-->
<DomainNameInformation>
  <DomainName>hrsite.corporate.contoso.com</DomainName>
  <DnsServers>1.2.3.4,5.6.7.8</DnsServers>
  <WebProxyServers>5.5.5.5</WebProxyServers>
  <AutoTrigger>true</AutoTrigger>
</DomainNameInformation>
<DomainNameInformation>
  <DomainName>.corp.contoso.com</DomainName>
  <DnsServers>10.10.10.10,20.20.20.20</DnsServers>
  <WebProxyServers>100.100.100.100</WebProxyServers>
</DomainNameInformation>

<!--EDPMode is turned on for the enterprise ID "corp.contoso.com". When a user accesses an app with that
ID, VPN will be triggered.-->
<EdpModeId>corp.contoso.com</EdpModeId>
<RememberCredentials>true</RememberCredentials>

<!--Always On is turned off, and triggering VPN for the apps and domain name specified earlier in the
profile will not occur if the user is connected to the trusted network "contoso.com".-->
<AlwaysOn>false</AlwaysOn>
<DnsSuffix>corp.contoso.com</DnsSuffix>
<TrustedNetworkDetection>contoso.com</TrustedNetworkDetection>
<Proxy>
  <Manual>
    <Server>HelloServer</Server>
  </Manual>
  <AutoConfigUrl>Helloworld.Com</AutoConfigUrl>
</Proxy>

<!--Device compliance is enabled and an alternate certificate is specified for domain resource
authentication.-->
<DeviceCompliance>
  <Enabled>true</Enabled>
  <Sso>
    <Enabled>true</Enabled>
    <Eku>This is my Eku</Eku>
    <IssuerHash>This is my issuer hash</IssuerHash>
  </Sso>
</DeviceCompliance>
</VPNProfile>

```

Sample plug-in VPN profile

The following sample is a sample plug-in VPN profile. This blob would fall under the ProfileXML node.

```
<VPNProfile>
  <ProfileName>TestVpnProfile</ProfileName>
  <PluginProfile>
    <ServerUrlList>testserver1.contoso.com;testserver2.contoso..com</ServerUrlList>
    <PluginPackageFamilyName>JuniperNetworks.JunosPulseVpn_cw5n1h2txyewy</PluginPackageFamilyName>
    <CustomConfiguration>&lt;pulse-
schema&gt;&lt;isSingleSignOnCredential&gt;true&lt;/isSingleSignOnCredential&gt;&lt;/pulse-schema&gt;
</CustomConfiguration>
  </PluginProfile>
  <Route>
    <Address>192.168.0.0</Address>
    <PrefixSize>24</PrefixSize>
  </Route>
  <Route>
    <Address>10.10.0.0</Address>
    <PrefixSize>16</PrefixSize>
  </Route>
  <AppTrigger>
    <App>
      <Id>Microsoft.MicrosoftEdge_8wekyb3d8bbwe</Id>
    </App>
  </AppTrigger>
  <AppTrigger>
    <App>
      <Id>%ProgramFiles%\Internet Explorer\iexplore.exe</Id>
    </App>
  </AppTrigger>
  <TrafficFilter>
    <App>
      <Id>%ProgramFiles%\Internet Explorer\iexplore.exe</Id>
    </App>
    <Protocol>6</Protocol>
    <LocalPortRanges>10,20-50,100-200</LocalPortRanges>
    <RemotePortRanges>20-50,100-200,300</RemotePortRanges>
    <RemoteAddressRanges>30.30.0.0/16,10.10.10.10-20.20.20.20</RemoteAddressRanges>
    <!--<RoutingPolicyType>ForceTunnel</RoutingPolicyType>-->
  </TrafficFilter>
  <TrafficFilter>
    <App>
      <Id>Microsoft.MicrosoftEdge_8wekyb3d8bbwe</Id>
    </App>
    <LocalAddressRanges>3.3.3.3/32,1.1.1.1-2.2.2.2</LocalAddressRanges>
  </TrafficFilter>
  <TrafficFilter>
    <App>
      <Id>Microsoft.MicrosoftEdge_8wekyb3d8bbwe</Id>
    </App>
    <Claims>0:SYG:SYD:(A;;CC;;;AU)</Claims>
    <!--<RoutingPolicyType>SplitTunnel</RoutingPolicyType>-->
  </TrafficFilter>
  <DomainNameInformation>
    <DomainName>corp.contoso.com</DomainName>
    <DnsServers>1.2.3.4,5.6.7.8</DnsServers>
    <WebProxyServers>5.5.5.5</WebProxyServers>
    <AutoTrigger>false</AutoTrigger>
  </DomainNameInformation>
  <DomainNameInformation>
    <DomainName>corp.contoso.com</DomainName>
    <DnsServers>10.10.10.10,20.20.20.20</DnsServers>
    <WebProxyServers>100.100.100.100</WebProxyServers>
  </DomainNameInformation>
  <!--<EdpModeId>corp.contoso.com</EdpModeId>-->
  <RememberCredentials>true</RememberCredentials>

```



```
<AlwaysOn>false</AlwaysOn>
<DnsSuffix>corp.contoso.com</DnsSuffix>
<TrustedNetworkDetection>contoso.com,test.corp.contoso.com</TrustedNetworkDetection>
<Proxy>
  <Manual>
    <Server>HelloServer</Server>
  </Manual>
  <AutoConfigUrl>Helloworld.Com</AutoConfigUrl>
</Proxy>
</VPNProfile>
```

Apply ProfileXML using Intune

After you configure the settings that you want using ProfileXML, you can create a custom profile in the [Microsoft Endpoint Manager admin center](#). After it's created, you deploy this profile to your devices.

1. Sign in to the [Microsoft Endpoint Manager admin center](#).
2. Select **Devices > Configuration profiles > Create profile**.
3. Enter the following properties:
 - **Platform:** Select **Windows 10 and later**
 - **Profile:** Select **Templates > Custom**.
4. Select **Create**.
5. In **Basics**, enter the following properties:
 - **Name:** Enter a descriptive name for the profile. Name your profiles so you can easily identify them later.
 - **Description:** Enter a description for the profile. This setting is optional, but recommended.
6. Select **Next**.
7. In **Configuration settings**, enter the following properties:
 - **OMA-URI:** Enter `./user/vendor/MSFT/VPNv2/Your_VPN_profile_name_/ProfileXML`.
 - **Data type:** Select `String (XML file)`.
 - **Value:** Browse to, and select your XML file.For more information on these settings, see [Use custom settings for Windows devices in Intune](#).
8. Select **Next**, and continue configuring the policy. For the specific steps and recommendations, see [Create a profile with custom settings in Intune](#).

Learn more

- [Create VPN profiles to connect to VPN servers in Intune](#)
- [VPNv2 configuration service provider \(CSP\) reference](#)
- [How to Create VPN Profiles in Configuration Manager](#)

Related articles

- [VPN technical guide](#)
- [VPN connection types](#)
- [VPN routing decisions](#)
- [VPN authentication options](#)
- [VPN and conditional access](#)

- VPN name resolution
- VPN auto-triggered profile options
- VPN security features

How to configure Diffie Hellman protocol over IKEv2 VPN connections

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies To: Windows Server (General Availability Channel), Windows Server 2016, Windows 10, Windows 11

In IKEv2 VPN connections, the default configuration for Diffie Hellman group is Group 2, which is not secure for IKE exchanges.

To secure the connections, update the configuration of VPN servers and clients by running VPN cmdlets.

VPN server

For VPN servers that run Windows Server 2012 R2 or later, you need to run [Set-VpnServerConfiguration](#) to configure the tunnel type. This makes all IKE exchanges on IKEv2 tunnel use the secure configuration.

```
Set-VpnServerConfiguration -TunnelType IKEv2 -CustomPolicy
```

On an earlier version of Windows Server, run [Set-VpnServerIPsecConfiguration](#). Since

`Set-VpnServerIPsecConfiguration` doesn't have `-TunnelType`, the configuration applies to all tunnel types on the server.

```
Set-VpnServerIPsecConfiguration -CustomPolicy
```

VPN client

For VPN client, you need to configure each VPN connection. For example, run [Set-VpnConnectionIPsecConfiguration \(version 4.0\)](#) and specify the name of the connection:

```
Set-VpnConnectionIPsecConfiguration -ConnectionName <String>
```

How to use Single Sign-On (SSO) over VPN and Wi-Fi connections

7/1/2022 • 4 minutes to read • [Edit Online](#)

This article explains requirements to enable Single Sign-On (SSO) to on-premises domain resources over Wi-Fi or VPN connections. The following scenarios are typically used:

- Connecting to a network using Wi-Fi or VPN.
- Use credentials for Wi-Fi or VPN authentication to also authenticate requests to access a domain resource without being prompted for your domain credentials.

For example, you want to connect to a corporate network and access an internal website that requires Windows integrated authentication.

The credentials that are used for the connection authentication are placed in Credential Manager as the default credentials for the logon session. Credential Manager stores credentials that can be used for specific domain resources. These are based on the target name of the resource:

- For VPN, the VPN stack saves its credential as the session default.
- For Wi-Fi, Extensible Authentication Protocol (EAP) provides support.

The credentials are placed in Credential Manager as a "*Session" credential. A "*Session" credential implies that it is valid for the current user session. The credentials are also cleaned up when the Wi-Fi or VPN connection is disconnected.

NOTE

In Windows 10, version 21h2 and later, the "*Session" credential is not visible in Credential Manager.

For example, if someone using Microsoft Edge tries to access a domain resource, Microsoft Edge has the right Enterprise Authentication capability. This allows [WinInet](#) to release the credentials that it gets from the Credential Manager to the SSP that is requesting it. For more information about the Enterprise Authentication capability, see [App capability declarations](#).

The local security authority will look at the device application to determine if it has the right capability. This includes items such as a Universal Windows Platform (UWP) application. If the app isn't a UWP, it doesn't matter. But if the application is a UWP app, it will evaluate at the device capability for Enterprise Authentication. If it does have that capability and if the resource that you're trying to access is in the Intranet zone in the Internet Options (ZoneMap), then the credential will be released. This behavior helps prevent credentials from being misused by untrusted third parties.

Intranet zone

For the Intranet zone, by default it only allows single-label names, such as [Http://finance](#). If the resource that needs to be accessed has multiple domain labels, then the workaround is to use the [Registry CSP](#).

Setting the ZoneMap

The ZoneMap is controlled using a registry that can be set through MDM. By default, single-label names such as [http://finance](#) are already in the intranet zone. For multi-label names, such as [http://finance.net](#), the ZoneMap needs to be updated.

MDM Policy

OMA URI example:

```
./Vendor/MSFT/Registry/HKU/S-1-5-21-2702878673-795188819-444038987-2781/Software/Microsoft/Windows/CurrentVersion/Internet%20Settings/ZoneMap/Domains/<domain name>/*
```

as an Integer Value of 1 for each of the domains that you want to SSO into from your device. This adds the specified domains to the Intranet Zone of the Microsoft Edge browser.

Credential requirements

For VPN, the following types of credentials will be added to credential manager after authentication:

- Username and password
- Certificate-based authentication:
 - TPM Key Storage Provider (KSP) Certificate
 - Software Key Storage Provider (KSP) Certificates
 - Smart Card Certificate
 - Windows Hello for Business Certificate

The username should also include a domain that can be reached over the connection (VPN or WiFi).

User certificate templates

If the credentials are certificate-based, then the elements in the following table need to be configured for the certificate templates to ensure they can also be used for Kerberos client authentication.

TEMPLATE ELEMENT	CONFIGURATION
SubjectName	The user's distinguished name (DN) where the domain components of the distinguished name reflect the internal DNS namespace when the SubjectAlternativeName does not have the fully qualified UPN required to find the domain controller. This requirement is relevant in multi-forest environments as it ensures a domain controller can be located.
SubjectAlternativeName	The user's fully qualified UPN where a domain name component of the user's UPN matches the organizations internal domain's DNS namespace. This requirement is relevant in multi-forest environments as it ensures a domain controller can be located when the SubjectName does not have the DN required to find the domain controller.
Key Storage Provider (KSP)	If the device is joined to Azure AD, a discrete SSO certificate is used.
EnhancedKeyUsage	One or more of the following EKUs is required: <ul style="list-style-type: none">- Client Authentication (for the VPN)- EAP Filtering OID (for Windows Hello for Business)- SmartCardLogon (for Azure AD-joined devices) If the domain controllers require smart card EKU either: <ul style="list-style-type: none">- SmartCardLogon- id-pkinit-KPClientAuth (1.3.6.1.5.2.3.4) Otherwise: <ul style="list-style-type: none">- TLS/SSL Client Authentication (1.3.6.1.5.5.7.3.2)

NDES server configuration

The NDES server is required to be configured so that incoming SCEP requests can be mapped to the correct template to be used. For more information, see [Configure certificate infrastructure for SCEP](#).

Active Directory requirements

You need IP connectivity to a DNS server and domain controller over the network interface so that authentication can succeed as well.

Domain controllers must have appropriate KDC certificates for the client to trust them as domain controllers. Because phones are not domain-joined, the root CA of the KDC's certificate must be in the Third-Party Root CA or Smart Card Trusted Roots store.

Domain controllers must be using certificates based on the updated KDC certificate template Kerberos Authentication. This requires that all authenticating domain controllers run Windows Server 2016, or you'll need to enable strict KDC validation on domain controllers that run previous versions of Windows Server.

For more information, see [Enabling Strict KDC Validation in Windows Kerberos](#).

Optimizing Office 365 traffic for remote workers with the native Windows 10 and Windows 11 VPN client

7/1/2022 • 14 minutes to read • [Edit Online](#)

This article describes how to configure the recommendations in the article [Optimize Office 365 connectivity for remote users using VPN split tunneling](#) for the *native Windows 10 and Windows 11 VPN client*. This guidance enables VPN administrators to optimize Office 365 usage while still ensuring that all other traffic goes over the VPN connection and through existing security gateways and tooling.

This can be achieved for the native/built-in Windows 10 and Windows 11 VPN client using a *Force Tunneling with Exclusions* approach. This allows you to define IP-based exclusions *even when using force tunneling* in order to "split" certain traffic to use the physical interface while still forcing all other traffic via the VPN interface. Traffic addressed to specifically defined destinations (like those listed in the Office 365 optimize categories) will therefore follow a much more direct and efficient path, without the need to traverse or "hairpin" via the VPN tunnel and back out of the corporate network. For cloud-services like Office 365, this makes a huge difference in performance and usability for remote users.

NOTE

The term *force tunneling with exclusions* is sometimes confusingly called "split tunnels" by other vendors and in some online documentation. For Windows 10 and Windows 11 VPN, the term *split tunneling* is defined differently as described in the article [VPN routing decisions](#).

Solution Overview

The solution is based upon the use of a VPN Configuration Service Provider Reference profile ([VPNv2 CSP](#)) and the embedded [ProfileXML](#). These are used to configure the VPN profile on the device. Various provisioning approaches can be used to create and deploy the VPN profile as discussed in the article [Step 6. Configure Windows 10 client Always On VPN connections](#).

Typically, these VPN profiles are distributed using a Mobile Device Management solution like Intune, as described in [VPN profile options](#) and [Configure the VPN client by using Intune](#).

To enable the use of force tunneling in Windows 10 or Windows 11 VPN, the `<RoutingPolicyType>` setting is typically configured with a value of *ForceTunnel* in your existing Profile XML (or script) by way of the following entry, under the `<NativeProfile></NativeProfile>` section:

```
<RoutingPolicyType>ForceTunnel</RoutingPolicyType>
```

In order to define specific force tunnel exclusions, you then need to add the following lines to your existing Profile XML (or script) for each required exclusion, and place them outside of the

`<NativeProfile></NativeProfile>` section as follows:

```
<Route>
  <Address>[IP addresses or subnet]</Address>
  <PrefixSize>[IP Prefix]</PrefixSize>
  <ExclusionRoute>true</ExclusionRoute>
</Route>
```

Entries defined by the `[IP Addresses or Subnet]` and `[IP Prefix]` references will consequently be added to the routing table as *more specific route entries* that will use the Internet-connected interface as the default gateway, as opposed to using the VPN interface. You will need to define a unique and separate `<Route></Route>` section for each required exclusion.

An example of a correctly formatted Profile XML configuration for force tunnel with exclusions is shown below:

```
<VPNProfile>
  <NativeProfile>
    <RoutingPolicyType>ForceTunnel</RoutingPolicyType>
  </NativeProfile>
  <Route>
    <Address>203.0.113.0</Address>
    <PrefixSize>24</PrefixSize>
    <ExclusionRoute>true</ExclusionRoute>
  </Route>
  <Route>
    <Address>198.51.100.0</Address>
    <PrefixSize>22</PrefixSize>
    <ExclusionRoute>true</ExclusionRoute>
  </Route>
</VPNProfile>
```

NOTE

The IP addresses and prefix size values in this example are used purely as examples only and should not be used.

Solution Deployment

For Office 365, it is therefore necessary to add exclusions for all IP addresses documented within the optimize categories described in [Office 365 URLs and IP address ranges](#) to ensure that they are excluded from VPN force tunneling.

This can be achieved manually by adding the IP addresses defined within the *optimize* category entries to an existing Profile XML (or script) file, or alternatively the following script can be used which dynamically adds the required entries to an existing PowerShell script, or XML file, based upon directly querying the REST-based web service to ensure the correct IP address ranges are always used.

An example of a PowerShell script that can be used to update a force tunnel VPN connection with Office 365 exclusions is provided below.

```
# Copyright (c) Microsoft Corporation. All rights reserved.
#
# THIS SAMPLE CODE AND INFORMATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND,
# WHETHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE IMPLIED
# WARRANTIES OF MERCHANTABILITY AND/OR FITNESS FOR A PARTICULAR PURPOSE.
# IF THIS CODE AND INFORMATION IS MODIFIED, THE ENTIRE RISK OF USE OR RESULTS IN
# CONNECTION WITH THE USE OF THIS CODE AND INFORMATION REMAINS WITH THE USER.

<#
.SYNOPSIS
    Applies or updates recommended Office 365 optimize IP address exclusions to an existing force tunnel
```


Windows 10 and Windows 11 VPN profile

.DESCRIPTION

Connects to the Office 365 worldwide commercial service instance endpoints to obtain the latest published IP address ranges

Compares the optimized IP addresses with those contained in the supplied VPN Profile (PowerShell or XML file)

Adds or updates IP addresses as necessary and saves the resultant file with "-NEW" appended to the file name

.PARAMETERS

Filename and path for a supplied Windows 10 or Windows 11 VPN profile file in either PowerShell or XML format

.NOTES

Requires at least Windows 10 Version 1803 with KB4493437, 1809 with KB4490481, or later

.VERSION

1.0

#>

```
param (  
    [string]$VPNprofilefile  
)
```

\$usage="@"

This script uses the following parameters:

VPNprofilefile - The full path and name of the VPN profile PowerShell script or XML file

EXAMPLES

To check a VPN profile PowerShell script file:

```
Update-VPN-Profile-Office365-Exclusion-Routes.ps1 -VPNprofilefile [FULLPATH AND NAME OF POWERSHELL SCRIPT FILE]
```

To check a VPN profile XML file:

```
Update-VPN-Profile-Office365-Exclusion-Routes.ps1 -VPNprofilefile [FULLPATH AND NAME OF XML FILE]
```

"@

```
# Check if filename has been provided #
```

```
if ($VPNprofilefile -eq "")
```

```
{
```

```
    Write-Host "`nWARNING: You must specify either a PowerShell script or XML filename!" -ForegroundColor Red
```

```
    $usage
```

```
    exit
```

```
}
```

```
$FileExtension = [System.IO.Path]::GetExtension($VPNprofilefile)
```

```
# Check if XML file exists and is a valid XML file #
```

```
if ( $VPNprofilefile -ne "" -and $FileExtension -eq ".xml")
```

```
{
```

```
    if ( Test-Path $VPNprofilefile )
```

```
    {
```

```
        $xml = New-Object System.Xml.XmlDocument
```

```
        try
```

```
        {
```

```
            $xml.Load((Get-ChildItem -Path $VPNprofilefile).FullName)
```

```
        }
```

```
        catch [System.Xml.XmlException]
```

```
        {
```

```
            Write-Verbose "$VPNprofilefile : $($_.toString())"
```

```
            Write-Host "`nWARNING: The VPN profile XML file is not a valid xml file or incorrectly
```

```
formatted!" -ForegroundColor Red
```

```
            $usage
```

```
            exit
```

```

    }
  }else
  {
    Write-Host "`nWARNING: VPN profile XML file does not exist or cannot be found!" -ForegroundColor Red
    $usage
    exit
  }
}

# Check if VPN profile PowerShell script file exists and contains a VPNPROFILE XML section #
if ( $VPNprofilefile -ne "" -and $FileExtension -eq ".ps1")
{
  if ( (Test-Path $VPNprofilefile) )
  {
    if (-Not $(Select-String -Path $VPNprofilefile -Pattern "<VPNPROFILE>"))
    {
      Write-Host "`nWARNING: PowerShell script file does not contain a valid VPN profile XML section
or is incorrectly formatted!" -ForegroundColor Red
      $usage
      exit
    }
  }else
  {
    Write-Host "`nWARNING: PowerShell script file does not exist or cannot be found!"-ForegroundColor
Red
    $usage
    exit
  }
}

# Define Office 365 endpoints and service URLs #
$ws = "https://endpoints.office.com"
$baseServiceUrl = "https://endpoints.office.com"

# Path where client ID and latest version number will be stored #
$datapath = $Env:TEMP + "\endpoints_clientid_latestversion.txt"

# Fetch client ID and version if data file exists; otherwise create new file #
if (Test-Path $datapath)
{
  $content = Get-Content $datapath
  $clientRequestId = $content[0]
  $lastVersion = $content[1]
}
else
{
  $clientRequestId = [GUID]::NewGuid().Guid
  $lastVersion = "0000000000"
  @($clientRequestId, $lastVersion) | Out-File $datapath
}

# Call version method to check the latest version, and pull new data if version number is different #
$version = Invoke-RestMethod -Uri ($ws + "/version?clientRequestId=" + $clientRequestId)

if ($version[0].latest -gt $lastVersion)
{
  Write-Host
  Write-Host "A new version of Office 365 worldwide commercial service instance endpoints has been
detected!" -ForegroundColor Cyan

  # Write the new version number to the data file #
  @($clientRequestId, $version[0].latest) | Out-File $datapath
}

# Invoke endpoints method to get the new data #
$uri = "$baseServiceUrl" + "/endpoints/worldwide?clientRequestId=$clientRequestId"

# Invoke endpoints method to get the data for the VPN profile comparison #

```

```

$endpointSets = Invoke-RestMethod -Uri ($uri)
$Optimize = $endpointSets | Where-Object { $_.category -eq "Optimize" }
$optimizeIpsv4 = $Optimize.ips | Where-Object { ($_.contains(".")) } | Sort-Object -Unique

# Temporarily include additional IP address until Teams client update is released
$optimizeIpsv4 += "13.107.60.1/32"

# Process PowerShell script file start #
if ($VPNprofilefile -ne "" -and $FileExtension -eq ".ps1")
{
    Write-host "`nStarting PowerShell script exclusion route check...`n" -ForegroundColor Cyan

    # Clear Variables to allow re-run testing #

    $ARRVPN=$null          # Array to hold VPN addresses from VPN profile PowerShell file #
    $In_Opt_Only=$null     # Variable to hold IP addresses that only appear in the optimize list #
    $In_VPN_Only=$null     # Variable to hold IP addresses that only appear in the VPN profile
PowerShell file #

    # Extract the Profile XML from the ps1 file #

    $regex = '(?sm).**.<VPNProfile>\r?\n(?:\r?\n</VPNProfile>.*'

    # Create xml format variable to compare with the optimize list #

    $xmlbody=(Get-Content -Raw $VPNprofilefile) -replace $regex, '$1'
    [xml]$VPNprofilexml="<VPNProfile>"+$xmlbody+"</VPNProfile>"

    # Loop through each address found in VPNPROFILE XML section #
    foreach ($Route in $VPNprofilexml.VPNProfile.Route)
    {
        $VPNIP=$Route.Address+"/"+$Route.PrefixSize
        [array]$ARRVPN=$ARRVPN+$VPNIP
    }

    # In optimize address list only #
    $In_Opt_Only= $optimizeIpsv4 | Where {$ARRVPN -NotContains $_}

    # In VPN list only #
    $In_VPN_Only = $ARRVPN | Where {$optimizeIpsv4 -NotContains $_}
    [array]$Inpfile = get-content $VPNprofilefile

    if ($In_Opt_Only.Count -gt 0 )
    {
        Write-Host "Exclusion route IP addresses are unknown, missing, or need to be updated in the VPN
profile`n" -ForegroundColor Red

        [int32]$insline=0

        for ($i=0; $i -lt $Inpfile.count; $i++)
        {
            if ($Inpfile[$i] -match "</NativeProfile>")
            {
                $insline += $i # Record the position of the line after the NativeProfile section ends #
            }
        }
        $OFS = "`r`n"
        foreach ($NewIP in $In_Opt_Only)
        {
            # Add the missing IP address(es) #
            $IPInfo=$NewIP.Split("/")
            $Inpfile[$insline] += $OFS+ "    <Route>"
            $Inpfile[$insline] += $OFS+ "        <Address>"+$IPInfo[0].Trim()+"</Address>"
            $Inpfile[$insline] += $OFS+ "        <PrefixSize>"+$IPInfo[1].Trim()+"</PrefixSize>"
            $Inpfile[$insline] += $OFS+ "        <ExclusionRoute>true</ExclusionRoute>"
            $Inpfile[$insline] += $OFS+ "    </Route>"
        }
        # Update fileName and write new PowerShell file #
        $NewFileName=(Get-Item $VPNprofilefile).Basename + "-NEW.ps1"
    }
}

```

```

        $OutFile=$(Split-Path $VPNprofilefile -Parent)+"\"+$NewFileName
        $InpFile | Set-Content $OutFile
        Write-Host "Exclusion routes have been added to VPN profile and output to a separate PowerShell
script file; the original file has not been modified`n" -ForegroundColor Green
    }else
    {
        Write-Host "Exclusion route IP addresses are correct and up to date in the VPN profile`n" -
ForegroundColor Green
        $OutFile=$VPNprofilefile
    }

if ( $In_VPN_Only.Count -gt 0 )
{
    Write-Host "Unknown exclusion route IP addresses have been found in the VPN profile`n" -ForegroundColor
Yellow

    foreach ($OldIP in $In_VPN_Only)
    {
        [array]$Inpfile = get-content $Outfile
        $IPInfo=$OldIP.Split("/")
        Write-Host "Unknown exclusion route IP address"$IPInfo[0]"has been found in the VPN profile - Do
you wish to remove it? (Y/N)`n" -ForegroundColor Yellow
        $matchstr="<Address>"+$IPInfo[0].Trim()+"</Address>"
        $DelAns=Read-host
        if ($DelAns.ToUpper() -eq "Y")
        {
            [int32]$insline=0
            for ($i=0; $i -lt $Inpfile.count; $i++)
            {
                if ($Inpfile[$i] -match $matchstr)
                {
                    $insline += $i # Record the position of the line for the string match #
                }
            }
            # Remove entries from XML #
            $InpFile[$insline-1]="REMOVETHISLINE"
            $InpFile[$insline]="REMOVETHISLINE"
            $InpFile[$insline+1]="REMOVETHISLINE"
            $InpFile[$insline+2]="REMOVETHISLINE"
            $InpFile[$insline+3]="REMOVETHISLINE"
            $InpFile=$InpFile | Where-Object {$_ -ne "REMOVETHISLINE"}

            # Update filename and write new PowerShell file #
            $NewFileName=(Get-Item $VPNprofilefile).Basename + "-NEW.xml"
            $OutFile=$(Split-Path $VPNprofilefile -Parent)+"\"+$NewFileName
            $Inpfile | Set-content $OutFile
            Write-Host "`nAddress"$IPInfo[0]"exclusion route has been removed from the VPN
profile and output to a separate PowerShell script file; the original file has not been modified`n" -
ForegroundColor Green

        }else
        {
            Write-Host "`nExclusion route IP address has *NOT* been removed from the VPN profile`n"
-ForegroundColor Green
        }
    }
}

# Process XML file start #
if ($VPNprofilefile -ne "" -and $FileExtension -eq ".xml")
{
    Write-host "`nStarting XML file exclusion route check...`n" -ForegroundColor Cyan

    # Clear variables to allow re-run testing #
    $ARRVPN=$null # Array to hold VPN addresses from the XML file #
    $In_Opt_Only=$null # Variable to hold IP Addresses that only appear in optimize list #
    $In_VPN_Only=$null # Variable to hold IP Addresses that only appear in the VPN profile XML file
#

```

```

# Extract the Profile XML from the XML file #
$regex = '(?sm).*<VPNPProfile>\r?\n(?:\r?\n</VPNPProfile>).*'

# Create xml format variable to compare with optimize list #
$xmlbody=(Get-Content -Raw $VPNprofilefile) -replace $regex, '$1'
$xml]$VPNRulesxml="$xmlbody"

# Loop through each address found in VPNPROFILE file #
foreach ($Route in $VPNRulesxml.VPNProfile.Route)
{
    $VPNIP=$Route.Address+"/"+$Route.PrefixSize
    [array]$ARRVPN=$ARRVPN+$VPNIP
}

# In optimize address list only #
$In_Opt_Only= $optimizeIpsv4 | Where {$ARRVPN -NotContains $_}

# In VPN list only #
$In_VPN_only =$ARRVPN | Where {$optimizeIpsv4 -NotContains $_}
[System.Collections.ArrayList]$Inpfile = get-content $VPNprofilefile

if ($In_Opt_Only.Count -gt 0 )
{
    Write-Host "Exclusion route IP addresses are unknown, missing, or need to be updated in the VPN
profile`n" -ForegroundColor Red

    foreach ($NewIP in $In_Opt_Only)
    {
        # Add the missing IP address(es) #
        $IPInfo=$NewIP.Split("/")
        $routes += "<Route>`n"+`t<Address>"+$IPInfo[0].Trim()+
</Address>`n"+`t<PrefixSize>"+$IPInfo[1].Trim()+
</PrefixSize>`n"+`t<ExclusionRoute>true</ExclusionRoute>`n"+`t</Route>`n"
    }
    $inspoint = $Inpfile.IndexOf("</VPNPProfile>")
    $Inpfile.Insert($inspoint,$routes)

    # Update filename and write new XML file #
    $NewFileName=(Get-Item $VPNprofilefile).Basename + "-NEW.xml"
    $OutFile=$(Split-Path $VPNprofilefile -Parent)+"\"+$NewFileName
    $InpFile | Set-Content $OutFile
    Write-Host "Exclusion routes have been added to VPN profile and output to a separate XML file;
the original file has not been modified`n`n" -ForegroundColor Green

}
else
{
    Write-Host "Exclusion route IP addresses are correct and up to date in the VPN profile`n" -
ForegroundColor Green
    $OutFile=$VPNprofilefile
}

if ( $In_VPN_Only.Count -gt 0 )
{
    Write-Host "Unknown exclusion route IP addresses found in the VPN profile`n" -ForegroundColor
Yellow

    foreach ($OldIP in $In_VPN_Only)
    {
        [array]$Inpfile = get-content $OutFile
        $IPInfo=$OldIP.Split("/")
        Write-Host "Unknown exclusion route IP address"$IPInfo[0]"has been found in the VPN profile
- Do you wish to remove it? (Y/N)`n" -ForegroundColor Yellow
        $matchstr="<Route>"+<Address>"+$IPInfo[0].Trim()+</Address>"+
<PrefixSize>"+$IPInfo[1].Trim()+</PrefixSize>"+<ExclusionRoute>true</ExclusionRoute>"+</Route>"
        $DelAns=Read-host
        if ($DelAns.ToUpper() -eq "Y")
        {
            # Remove unknown IP address(es) #

```

```

# Remove unknown IP address(es) #
$inspoint = $Inpfile[0].IndexOf($matchstr)
$Inpfile[0] = $Inpfile[0].Replace($matchstr,"")

# Update filename and write new XML file #
$NewFileName=(Get-Item $VPNprofilefile).Basename + "-NEW.xml"
$OutFile=$(Split-Path $VPNprofilefile -Parent)+"\"+$NewFileName
$Inpfile | Set-content $OutFile
Write-Host "`nAddress"$IPInfo[0]"exclusion route has been removed from the VPN
profile and output to a separate XML file; the original file has not been modified`n" -ForegroundColor Green

    }else
    {
        Write-Host "`nExclusion route IP address has *NOT* been removed from the VPN
profile`n" -ForegroundColor Green
    }
}
}
}

```

Version Support

This solution is supported with the following versions of Windows:

- Windows 11
- Windows 10 1903/1909 and newer: Included, no action needed
- Windows 10 1809: At least [KB4490481](#)
- Windows 10 1803: At least [KB4493437](#)
- Windows 10 1709 and lower: Exclusion routes are not supported
- Windows 10 Enterprise 2019 LTSC: At least [KB4490481](#)
- Windows 10 Enterprise 2016 LTSC: Exclusion routes are not supported
- Windows 10 Enterprise 2015 LTSC: Exclusion routes are not supported

Microsoft strongly recommends that the latest available Windows 10 cumulative update always be applied.

Other Considerations

You should also be able to adapt this approach to include necessary exclusions for other cloud-services that can be defined by known/static IP addresses; exclusions required for [Cisco WebEx](#) or [Zoom](#) are good examples.

Examples

An example of a PowerShell script that can be used to create a force tunnel VPN connection with Office 365 exclusions is provided below, or refer to the guidance in [Create the ProfileXML configuration files](#) to create the initial PowerShell script:

```

# Copyright (c) Microsoft Corporation. All rights reserved.
#
# THIS SAMPLE CODE AND INFORMATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND,
# WHETHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE IMPLIED
# WARRANTIES OF MERCHANTABILITY AND/OR FITNESS FOR A PARTICULAR PURPOSE.
# IF THIS CODE AND INFORMATION IS MODIFIED, THE ENTIRE RISK OF USE OR RESULTS IN
# CONNECTION WITH THE USE OF THIS CODE AND INFORMATION REMAINS WITH THE USER.

<#
.SYNOPSIS

```

Configures an AlwaysOn IKEv2 VPN Connection using a basic script

.DESCRIPTION

Configures an AlwaysOn IKEv2 VPN Connection with proxy PAC information and force tunneling

.PARAMETERS

Parameters are defined in a ProfileXML object within the script itself

.NOTES

Requires at least Windows 10 Version 1803 with KB4493437, 1809 with KB4490481, or later

.VERSION

1.0

#>

```
<!-- Define Key VPN Profile Parameters -->
```

```
$ProfileName = 'Contoso VPN with Office 365 Exclusions'
```

```
$ProfileNameEscaped = $ProfileName -replace ' ', '%20'
```

```
<!-- Define VPN ProfileXML -->
```

```
$ProfileXML = '<VPNProfile>
```

```
  <RememberCredentials>true</RememberCredentials>
```

```
  <DnsSuffix>corp.contoso.com</DnsSuffix>
```

```
  <AlwaysOn>true</AlwaysOn>
```

```
  <TrustedNetworkDetection>corp.contoso.com</TrustedNetworkDetection>
```

```
<NativeProfile>
```

```
  <Servers>edge1.contoso.com</Servers>
```

```
  <RoutingPolicyType>ForceTunnel</RoutingPolicyType>
```

```
  <NativeProtocolType>IKEv2</NativeProtocolType>
```

```
  <Authentication>
```

```
    <MachineMethod>Certificate</MachineMethod>
```

```
  </Authentication>
```

```
</NativeProfile>
```

```
<Route>
```

```
  <Address>13.107.6.152</Address>
```

```
  <PrefixSize>31</PrefixSize>
```

```
  <ExclusionRoute>true</ExclusionRoute>
```

```
</Route>
```

```
<Route>
```

```
  <Address>13.107.18.10</Address>
```

```
  <PrefixSize>31</PrefixSize>
```

```
  <ExclusionRoute>true</ExclusionRoute>
```

```
</Route>
```

```
<Route>
```

```
  <Address>13.107.128.0</Address>
```

```
  <PrefixSize>22</PrefixSize>
```

```
  <ExclusionRoute>true</ExclusionRoute>
```

```
</Route>
```

```
<Route>
```

```
  <Address>23.103.160.0</Address>
```

```
  <PrefixSize>20</PrefixSize>
```

```
  <ExclusionRoute>true</ExclusionRoute>
```

```
</Route>
```

```
<Route>
```

```
  <Address>40.96.0.0</Address>
```

```
  <PrefixSize>13</PrefixSize>
```

```
  <ExclusionRoute>true</ExclusionRoute>
```

```
</Route>
```

```
<Route>
```

```
  <Address>40.104.0.0</Address>
```

```
  <PrefixSize>15</PrefixSize>
```

```
  <ExclusionRoute>true</ExclusionRoute>
```

```
</Route>
```

```
<Route>
```

```
  <Address>52.96.0.0</Address>
```

```
  <PrefixSize>14</PrefixSize>
```

```
  <ExclusionRoute>true</ExclusionRoute>
```

```
</Route>
```

```
<Route>
```

```
  <Address>131.253.33.215</Address>
```

```
  <PrefixSize>32</PrefixSize>
```

```
  <ExclusionRoute>true</ExclusionRoute>
```

```
</Route>
```

```
<Route>
  <Address>132.245.0.0</Address>
  <PrefixSize>16</PrefixSize>
  <ExclusionRoute>true</ExclusionRoute>
</Route>
<Route>
  <Address>150.171.32.0</Address>
  <PrefixSize>22</PrefixSize>
  <ExclusionRoute>true</ExclusionRoute>
</Route>
<Route>
  <Address>191.234.140.0</Address>
  <PrefixSize>22</PrefixSize>
  <ExclusionRoute>true</ExclusionRoute>
</Route>
<Route>
  <Address>204.79.197.215</Address>
  <PrefixSize>32</PrefixSize>
  <ExclusionRoute>true</ExclusionRoute>
</Route>
<Route>
  <Address>13.107.136.0</Address>
  <PrefixSize>22</PrefixSize>
  <ExclusionRoute>true</ExclusionRoute>
</Route>
<Route>
  <Address>40.108.128.0</Address>
  <PrefixSize>17</PrefixSize>
  <ExclusionRoute>true</ExclusionRoute>
</Route>
<Route>
  <Address>52.104.0.0</Address>
  <PrefixSize>14</PrefixSize>
  <ExclusionRoute>true</ExclusionRoute>
</Route>
<Route>
  <Address>104.146.128.0</Address>
  <PrefixSize>17</PrefixSize>
  <ExclusionRoute>true</ExclusionRoute>
</Route>
<Route>
  <Address>150.171.40.0</Address>
  <PrefixSize>22</PrefixSize>
  <ExclusionRoute>true</ExclusionRoute>
</Route>
<Route>
  <Address>13.107.60.1</Address>
  <PrefixSize>32</PrefixSize>
  <ExclusionRoute>true</ExclusionRoute>
</Route>
<Route>
  <Address>13.107.64.0</Address>
  <PrefixSize>18</PrefixSize>
  <ExclusionRoute>true</ExclusionRoute>
</Route>
<Route>
  <Address>52.112.0.0</Address>
  <PrefixSize>14</PrefixSize>
  <ExclusionRoute>true</ExclusionRoute>
</Route>
<Route>
  <Address>52.120.0.0</Address>
  <PrefixSize>14</PrefixSize>
  <ExclusionRoute>true</ExclusionRoute>
</Route>
<Proxy>
  <AutoConfigUrl>http://webproxy.corp.contoso.com/proxy.pac</AutoConfigUrl>
</Proxy>
</VPNProfile>
```



```

<!-- Convert ProfileXML to Escaped Format -->
$ProfileXML = $ProfileXML -replace '<', '&lt;';
$ProfileXML = $ProfileXML -replace '>', '&gt;';
$ProfileXML = $ProfileXML -replace '"', '&quot;';

<!-- Define WMI-to-CSP Bridge Properties -->
$nodeCSPURI = './Vendor/MSFT/VPNv2'
$namespaceName = "root\cimv2\mdm\dmmap"
$class_name = "MDM_VPNv2_01"

<!-- Define WMI Session -->
$session = New-CimSession

<!-- Detect and Delete Previous VPN Profile -->
try
{
    $deleteInstances = $session.EnumerateInstances($namespaceName, $className, $options)
    foreach ($deleteInstance in $deleteInstances)
    {
        $InstanceId = $deleteInstance.InstanceID
        if ("$InstanceId" -eq "$ProfileNameEscaped")
        {
            $session.DeleteInstance($namespaceName, $deleteInstance, $options)
            $Message = "Removed $ProfileName profile $InstanceId"
            Write-Host "$Message"
        } else {
            $Message = "Ignoring existing VPN profile $InstanceId"
            Write-Host "$Message"
        }
    }
}
catch [Exception]
{
    $Message = "Unable to remove existing outdated instance(s) of $ProfileName profile: $_"
    Write-Host "$Message"
    exit
}

<!-- Create VPN Profile -->
try
{
    $newInstance = New-Object Microsoft.Management.Infrastructure.CimInstance $className, $namespaceName
    $property = [Microsoft.Management.Infrastructure.CimProperty]::Create("ParentID", "$nodeCSPURI",
'String', 'Key')
    $newInstance.CimInstanceProperties.Add($property)
    $property = [Microsoft.Management.Infrastructure.CimProperty]::Create("InstanceID",
"$ProfileNameEscaped", 'String', 'Key')
    $newInstance.CimInstanceProperties.Add($property)
    $property = [Microsoft.Management.Infrastructure.CimProperty]::Create("ProfileXML", "$ProfileXML",
'String', 'Property')
    $newInstance.CimInstanceProperties.Add($property)

    $session.CreateInstance($namespaceName, $newInstance, $options)
    $Message = "Created $ProfileName profile."
    Write-Host "$Message"
    Write-Host "$ProfileName profile summary:"
    $session.EnumerateInstances($namespaceName, $className, $options)
}
catch [Exception]
{
    $Message = "Unable to create $ProfileName profile: $_"
    Write-Host "$Message"
    exit
}

$Message = "Script Complete"
Write-Host "$Message"

```

An example of an [Intune-ready XML file](#) that can be used to create a force tunnel VPN connection with Office 365 exclusions is provided below, or refer to the guidance in [Create the ProfileXML configuration files](#) to create the initial XML file.

NOTE

This XML is formatted for use with Intune and cannot contain any carriage returns or whitespace.

```
<VPNProfile><RememberCredentials>true</RememberCredentials><DnsSuffix>corp.contoso.com</DnsSuffix>
<AlwaysOn>true</AlwaysOn><TrustedNetworkDetection>corp.contoso.com</TrustedNetworkDetection><NativeProfile>
<Servers>edge1.contoso.com</Servers><RoutingPolicyType>ForceTunnel</RoutingPolicyType>
<NativeProtocolType>IKEv2</NativeProtocolType><Authentication><MachineMethod>Certificate</MachineMethod>
</Authentication></NativeProfile><Route><Address>13.107.6.152</Address><PrefixSize>31</PrefixSize>
<ExclusionRoute>true</ExclusionRoute></Route><Route><Address>13.107.18.10</Address>
<PrefixSize>31</PrefixSize><ExclusionRoute>true</ExclusionRoute></Route><Route>
<Address>13.107.128.0</Address><PrefixSize>22</PrefixSize><ExclusionRoute>true</ExclusionRoute></Route>
<Route><Address>23.103.160.0</Address><PrefixSize>20</PrefixSize><ExclusionRoute>true</ExclusionRoute>
</Route><Route><Address>40.96.0.0</Address><PrefixSize>13</PrefixSize><ExclusionRoute>true</ExclusionRoute>
</Route><Route><Address>40.104.0.0</Address><PrefixSize>15</PrefixSize><ExclusionRoute>true</ExclusionRoute>
</Route><Route><Address>52.96.0.0</Address><PrefixSize>14</PrefixSize><ExclusionRoute>true</ExclusionRoute>
</Route><Route><Address>131.253.33.215</Address><PrefixSize>32</PrefixSize>
<ExclusionRoute>true</ExclusionRoute></Route><Route><Address>132.245.0.0</Address>
<PrefixSize>16</PrefixSize><ExclusionRoute>true</ExclusionRoute></Route><Route>
<Address>150.171.32.0</Address><PrefixSize>22</PrefixSize><ExclusionRoute>true</ExclusionRoute></Route>
<Route><Address>191.234.140.0</Address><PrefixSize>22</PrefixSize><ExclusionRoute>true</ExclusionRoute>
</Route><Route><Address>204.79.197.215</Address><PrefixSize>32</PrefixSize>
<ExclusionRoute>true</ExclusionRoute></Route><Route><Address>13.107.136.0</Address>
<PrefixSize>22</PrefixSize><ExclusionRoute>true</ExclusionRoute></Route><Route>
<Address>40.108.128.0</Address><PrefixSize>17</PrefixSize><ExclusionRoute>true</ExclusionRoute></Route>
<Route><Address>52.104.0.0</Address><PrefixSize>14</PrefixSize><ExclusionRoute>true</ExclusionRoute></Route>
<Route><Address>104.146.128.0</Address><PrefixSize>17</PrefixSize><ExclusionRoute>true</ExclusionRoute>
</Route><Route><Address>150.171.40.0</Address><PrefixSize>22</PrefixSize>
<ExclusionRoute>true</ExclusionRoute></Route><Route><Address>13.107.60.1</Address>
<PrefixSize>32</PrefixSize><ExclusionRoute>true</ExclusionRoute></Route><Route>
<Address>13.107.64.0</Address><PrefixSize>18</PrefixSize><ExclusionRoute>true</ExclusionRoute></Route>
<Route><Address>52.112.0.0</Address><PrefixSize>14</PrefixSize><ExclusionRoute>true</ExclusionRoute></Route>
<Route><Address>52.120.0.0</Address><PrefixSize>14</PrefixSize><ExclusionRoute>true</ExclusionRoute></Route>
<Proxy><AutoConfigUrl>http://webproxy.corp.contoso.com/proxy.pac</AutoConfigUrl></Proxy></VPNProfile>
```

Windows Defender Firewall with Advanced Security

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016 and above

This is an overview of the Windows Defender Firewall with Advanced Security (WFAS) and Internet Protocol security (IPsec) features.

Overview of Windows Defender Firewall with Advanced Security

Windows Defender Firewall in Windows 8, Windows 7, Windows Vista, Windows Server 2012, Windows Server 2008, and Windows Server 2008 R2 is a stateful host firewall that helps secure the device by allowing you to create rules that determine which network traffic is permitted to enter the device from the network and which network traffic the device is allowed to send to the network. Windows Defender Firewall also supports Internet Protocol security (IPsec), which you can use to require authentication from any device that is attempting to communicate with your device. When authentication is required, devices that cannot be authenticated as a trusted device cannot communicate with your device. You can also use IPsec to require that certain network traffic is encrypted to prevent it from being read by network packet analyzers that could be attached to the network by a malicious user.

The Windows Defender Firewall with Advanced Security MMC snap-in is more flexible and provides much more functionality than the consumer-friendly Windows Defender Firewall interface found in the Control Panel. Both interfaces interact with the same underlying services, but provide different levels of control over those services. While the Windows Defender Firewall Control Panel program can protect a single device in a home environment, it does not provide enough centralized management or security features to help secure more complex network traffic found in a typical business enterprise environment.

Feature description

Windows Defender Firewall with Advanced Security is an important part of a layered security model. By providing host-based, two-way network traffic filtering for a device, Windows Defender Firewall blocks unauthorized network traffic flowing into or out of the local device. Windows Defender Firewall also works with Network Awareness so that it can apply security settings appropriate to the types of networks to which the device is connected. Windows Defender Firewall and Internet Protocol Security (IPsec) configuration settings are integrated into a single Microsoft Management Console (MMC) named Windows Defender Firewall, so Windows Defender Firewall is also an important part of your network's isolation strategy.

Practical applications

To help address your organizational network security challenges, Windows Defender Firewall offers the following benefits:

- **Reduces the risk of network security threats.** Windows Defender Firewall reduces the attack surface of a device, providing an additional layer to the defense-in-depth model. Reducing the attack surface of a device increases manageability and decreases the likelihood of a successful attack.
- **Safeguards sensitive data and intellectual property.** With its integration with IPsec, Windows

Defender Firewall provides a simple way to enforce authenticated, end-to-end network communications. It provides scalable, tiered access to trusted network resources, helping to enforce integrity of the data, and optionally helping to protect the confidentiality of the data.

- **Extends the value of existing investments.** Because Windows Defender Firewall is a host-based firewall that is included with the operating system, there is no additional hardware or software required. Windows Defender Firewall is also designed to complement existing non-Microsoft network security solutions through a documented application programming interface (API).

Security baselines

7/1/2022 • 3 minutes to read • [Edit Online](#)

Using security baselines in your organization

Microsoft is dedicated to providing its customers with secure operating systems, such as Windows and Windows Server, and secure apps, such as Microsoft 365 apps for enterprise and Microsoft Edge. In addition to the security assurance of its products, Microsoft also enables you to have fine control over your environments by providing various configuration capabilities.

Even though Windows and Windows Server are designed to be secure out-of-the-box, many organizations still want more granular control over their security configurations. To navigate the large number of controls, organizations need guidance on configuring various security features. Microsoft provides this guidance in the form of security baselines.

We recommend that you implement an industry-standard configuration that is broadly known and well-tested, such as Microsoft security baselines, as opposed to creating a baseline yourself. This helps increase flexibility and reduce costs.

Here is a good blog about [Sticking with Well-Known and Proven Solutions](#).

What are security baselines?

Every organization faces security threats. However, the types of security threats that are of most concern to one organization can be completely different from another organization. For example, an e-commerce company may focus on protecting its Internet-facing web apps, while a hospital may focus on protecting confidential patient information. The one thing that all organizations have in common is a need to keep their apps and devices secure. These devices must be compliant with the security standards (or security baselines) defined by the organization.

A security baseline is a group of Microsoft-recommended configuration settings that explains their security impact. These settings are based on feedback from Microsoft security engineering teams, product groups, partners, and customers.

Why are security baselines needed?

Security baselines are an essential benefit to customers because they bring together expert knowledge from Microsoft, partners, and customers.

For example, there are over 3,000 Group Policy settings for Windows 10, which does not include over 1,800 Internet Explorer 11 settings. Of these 4,800 settings, only some are security-related. Although Microsoft provides extensive guidance on different security features, exploring each one can take a long time. You would have to determine the security impact of each setting on your own. Then, you would still need to determine the appropriate value for each setting.

In modern organizations, the security threat landscape is constantly evolving, and IT pros and policy-makers must keep up with security threats and make required changes to security settings to help mitigate these threats. To enable faster deployments and make managing Microsoft products easier, Microsoft provides customers with security baselines that are available in consumable formats, such as Group Policy Objects Backups.

Baseline principles

Our recommendations follow a streamlined and efficient approach to baseline definitions. The foundation of that approach is essentially:

- The baselines are designed for well-managed, security-conscious organizations in which standard end users do not have administrative rights.
- A baseline enforces a setting only if it mitigates a contemporary security threat and does not cause operational issues that are worse than the risks they mitigate.
- A baseline enforces a default only if it is otherwise likely to be set to an insecure state by an authorized user:
 - If a non-administrator can set an insecure state, enforce the default.
 - If setting an insecure state requires administrative rights, enforce the default only if it is likely that a misinformed administrator will otherwise choose poorly.

How can you use security baselines?

You can use security baselines to:

- Ensure that user and device configuration settings are compliant with the baseline.
- Set configuration settings. For example, you can use Group Policy, Microsoft Endpoint Configuration Manager, or Microsoft Intune to configure a device with the setting values specified in the baseline.

Where can I get the security baselines?

There are several ways to get and use security baselines:

1. You can download the security baselines from the [Microsoft Download Center](#). This download page is for the Security Compliance Toolkit (SCT), which comprises tools that can assist admins in managing baselines in addition to the security baselines. The security baselines are included in the [Security Compliance Toolkit \(SCT\)](#), which can be downloaded from the Microsoft Download Center. The SCT also includes tools to help admins manage the security baselines. You can also [Get Support for the security baselines](#)
2. [MDM \(Mobile Device Management\) security baselines](#) function like the Microsoft group policy-based security baselines and can easily integrate this into an existing MDM management tool.
3. MDM Security baselines can easily be configured in Microsoft Endpoint Manager on devices that run Windows 10 and 11. The following article provides the detail steps: [Windows MDM \(Mobile Device Management\) baselines](#).

Community



Related Videos

You may also be interested in this msdn channel 9 video:

- [Defrag Tools](#)

See Also

- [Microsoft Endpoint Configuration Manager](#)
- [Azure Monitor](#)
- [Microsoft Security Guidance Blog](#)
- [Microsoft Security Compliance Toolkit Download](#)
- [Microsoft Download Center](#)

Microsoft Security Compliance Toolkit 1.0 - How to use

7/1/2022 • 3 minutes to read • [Edit Online](#)

What is the Security Compliance Toolkit (SCT)?

The Security Compliance Toolkit (SCT) is a set of tools that allows enterprise security administrators to download, analyze, test, edit, and store Microsoft-recommended security configuration baselines for Windows and other Microsoft products.

The SCT enables administrators to effectively manage their enterprise's Group Policy Objects (GPOs). Using the toolkit, administrators can compare their current GPOs with Microsoft-recommended GPO baselines or other baselines, edit them, store them in GPO backup file format, and apply them broadly through Active Directory or individually through local policy.

The Security Compliance Toolkit consists of:

- Windows 11 security baseline
- Windows 10 security baselines
 - Windows 10 Version 21H2
 - Windows 10 Version 21H1
 - Windows 10 Version 20H2
 - Windows 10 Version 1809
 - Windows 10 Version 1607
 - Windows 10 Version 1507
- Windows Server security baselines
 - Windows Server 2022
 - Windows Server 2019
 - Windows Server 2016
 - Windows Server 2012 R2
- Microsoft Office security baseline
 - Office 2016
 - Microsoft 365 Apps for Enterprise Version 2206
- Microsoft Edge security baseline
 - Edge version 98
- Tools
 - Policy Analyzer
 - Local Group Policy Object (LGPO)
 - Set Object Security
 - GPO to Policy Rules

You can [download the tools](#) along with the baselines for the relevant Windows versions. For more details about security baseline recommendations, see the [Microsoft Security Guidance blog](#).

What is the Policy Analyzer tool?

The Policy Analyzer is a utility for analyzing and comparing sets of Group Policy Objects (GPOs). Its main features include:

- Highlight when a set of Group Policies has redundant settings or internal inconsistencies
- Highlight the differences between versions or sets of Group Policies
- Compare GPOs against current local policy and local registry settings
- Export results to a Microsoft Excel spreadsheet

Policy Analyzer lets you treat a set of GPOs as a single unit. This makes it easy to determine whether particular settings are duplicated across the GPOs or are set to conflicting values. Policy Analyzer also lets you capture a baseline and then compare it to a snapshot taken at a later time to identify changes anywhere across the set.

More information on the Policy Analyzer tool can be found on the [Microsoft Security Guidance blog](#) or by [downloading the tool](#).

What is the Local Group Policy Object (LGPO) tool?

LGPO.exe is a command-line utility that is designed to help automate management of Local Group Policy. Using local policy gives administrators a simple way to verify the effects of Group Policy settings, and is also useful for managing non-domain-joined systems. LGPO.exe can import and apply settings from Registry Policy (Registry.pol) files, security templates, Advanced Auditing backup files, as well as from formatted "LGPO text" files. It can export local policy to a GPO backup. It can export the contents of a Registry Policy file to the "LGPO text" format that can then be edited, and can build a Registry Policy file from an LGPO text file.

Documentation for the LGPO tool can be found on the [Microsoft Security Guidance blog](#) or by [downloading the tool](#).

What is the Set Object Security tool?

SetObjectSecurity.exe enables you to set the security descriptor for just about any type of Windows securable object, such as files, directories, registry keys, event logs, services, and SMB shares. For file system and registry objects, you can choose whether to apply inheritance rules. You can also choose to output the security descriptor in a .reg-file-compatible representation of the security descriptor for a REG_BINARY registry value.

Documentation for the Set Object Security tool can be found on the [Microsoft Security Baselines blog](#) or by [downloading the tool](#).

What is the GPO to Policy Rules tool?

Automate the conversion of GPO backups to Policy Analyzer .PolicyRules files and skip the GUI. GPO2PolicyRules is a command-line tool that is included with the Policy Analyzer download.

Documentation for the GPO to PolicyRules tool can be found on the [Microsoft Security Baselines blog](#) or by [downloading the tool](#).

Get Support

7/1/2022 • 2 minutes to read • [Edit Online](#)

What is the Microsoft Security Compliance Manager (SCM)?

The Security Compliance Manager (SCM) is now retired and is no longer supported. The reason is that SCM was an incredibly complex and large program that needed to be updated for every Windows release. It has been replaced by the Security Compliance Toolkit (SCT). To provide a better service for our customers, we have moved to SCT with which we can publish baselines through the Microsoft Download Center in a lightweight .zip file that contains GPO Backups, GPO reports, Excel spreadsheets, WMI filters, and scripts to apply the settings to local policy.

More information about this change can be found on the [Microsoft Security Guidance blog](#).

Where can I get an older version of a Windows baseline?

Any version of Windows baseline before Windows 10 1703 can still be downloaded using SCM. Any future versions of Windows baseline will be available through SCT. See the version matrix in this article to see if your version of Windows baseline is available on SCT.

- [SCM 4.0 Download](#)
- [SCM Frequently Asked Questions \(FAQ\)](#)
- [SCM Release Notes](#)
- [SCM baseline download help](#)

What file formats are supported by the new SCT?

The toolkit supports formats created by the Windows GPO backup feature (.pol, .inf, and .csv). Policy Analyzer saves its data in XML files with a .PolicyRules file extension. LGPO also supports its own LGPO text file format as a text-based analog for the binary registry.pol file format. See the LGPO documentation for more information. Keep in mind that SCM's .cab files are no longer supported.

Does SCT support Desired State Configuration (DSC) file format?

No. PowerShell-based DSC is rapidly gaining popularity, and more DSC tools are coming online to convert GPOs and DSC and to validate system configuration.

Does SCT support the creation of Microsoft Endpoint Manager DCM packs?

No. A potential alternative is Desired State Configuration (DSC), a feature of the [Windows Management Framework](#). A tool that supports conversion of GPO Backups to DSC format can be found [here](#).

Does SCT support the creation of Security Content Automation Protocol (SCAP)-format policies?

No. SCM supported only SCAP 1.0, which was not updated as SCAP evolved. The new toolkit likewise does not include SCAP support.

Version Matrix

Client Versions

NAME	BUILD	BASELINE RELEASE DATE	SECURITY TOOLS
Windows 11	Windows 11	October 2021	SCT 1.0
Windows 10	21H2 21H1 20H2 1809 1607 1507	December 2021 May 2021 December 2020 October 2018 October 2016 January 2016	SCT 1.0
Windows 8.1	9600 (April Update)	October 2013	SCM 4.0

Server Versions

NAME	BUILD	BASELINE RELEASE DATE	SECURITY TOOLS
Windows Server 2022	SecGuide	September 2021	SCT 1.0
Windows Server 2019	SecGuide	November 2018	SCT 1.0
Windows Server 2016	SecGuide	October 2016	SCT 1.0
Windows Server 2012 R2	SecGuide	August 2014	SCT 1.0

Microsoft Products

NAME	DETAILS	SECURITY TOOLS
Microsoft 365 Apps for enterprise, version 2206	SecGuide	SCT 1.0
Microsoft Edge, version 98	SecGuide	SCT 1.0

See also

[Windows security baselines](#)

Windows threat protection

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to:

- Windows 10
- Windows 11

In Windows client, hardware and software work together to help protect you from new and emerging threats. Expanded security protections in Windows 11 help boost security from the chip, to the cloud.

Windows threat protection

See the following articles to learn more about the different areas of Windows threat protection:

- [Application Control](#)
- [Attack Surface Reduction Rules](#)
- [Controlled Folder Access](#)
- [Exploit Protection](#)
- [Microsoft Defender Application Guard](#)
- [Microsoft Defender Device Guard](#)
- [Microsoft Defender SmartScreen](#)
- [Network Protection](#)
- [Virtualization-Based Protection of Code Integrity](#)
- [Web Protection](#)
- [Windows Firewall](#)
- [Windows Sandbox](#)

Next-generation protection

Next-generation protection is designed to identify and block new and emerging threats. Powered by the cloud and machine learning, Microsoft Defender Antivirus can help stop attacks in real-time.

- [Automated sandbox service](#)
- [Behavior monitoring](#)
- [Cloud-based protection](#)
- [Machine learning](#)
- [URL Protection](#)

Override Process Mitigation Options to help enforce app-related security policies

7/1/2022 • 3 minutes to read • [Edit Online](#)

Applies to:

- Windows 10, version 1607
- Windows Server 2016

Windows 10 includes Group Policy-configurable "Process Mitigation Options" that add advanced protections against memory-based attacks, that is, attacks where malware manipulates memory to gain control of a system. For example, malware might attempt to use buffer overruns to inject malicious executable code into memory, but Process Mitigation Options can prevent the running of the malicious code.

IMPORTANT

We recommend trying these mitigations in a test lab before deploying to your organization, to determine if they interfere with your organization's required apps.

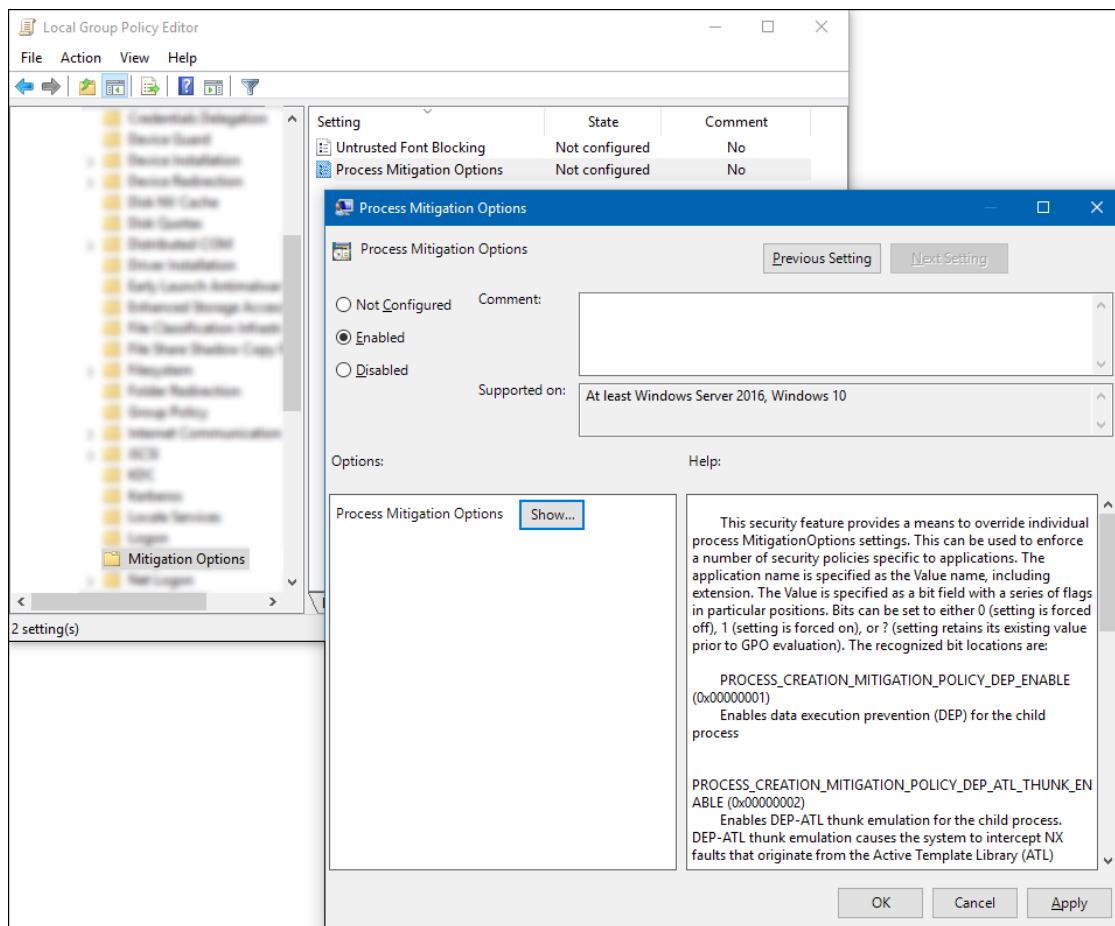
The Group Policy settings in this topic are related to three types of process mitigations. In Windows 10, all three types are on by default for 64-bit applications, but by using the Group Policy settings described in this topic, you can configure additional protections. The types of process mitigations are:

- **Data Execution Prevention (DEP)** is a system-level memory protection feature that enables the operating system to mark one or more pages of memory as non-executable, preventing code from being run from that region of memory, to help prevent exploitation of buffer overruns. DEP helps prevent code from being run from data pages such as the default heap, stacks, and memory pools. For more information, see [Data Execution Prevention](#).
- **Structured Exception Handling Overwrite Protection (SEHOP)** is designed to block exploits that use the Structured Exception Handler (SEH) overwrite technique. Because this protection mechanism is provided at run-time, it helps to protect apps regardless of whether they have been compiled with the latest improvements. For more information, see [Structured Exception Handling Overwrite Protection](#).
- **Address Space Layout Randomization (ASLR)** loads DLLs into random memory addresses at boot time to mitigate against malware that's designed to attack specific memory locations, where specific DLLs are expected to be loaded. For more information, see [Address Space Layout Randomization](#). To find additional ASLR protections in the table below, look for `IMAGES` or `ASLR`.

The following procedure describes how to use Group Policy to override individual **Process Mitigation Options** settings.

To modify Process Mitigation Options

1. Open your Group Policy editor and go to the **Administrative Templates\System\Mitigation Options\Process Mitigation Options** setting.



- Click **Enabled**, and then in the **Options** area, click **Show** to open the **Show Contents** box, where you'll be able to add your apps and the appropriate bit flag values, as shown in the [Setting the bit field](#) and [Example](#) sections of this topic.

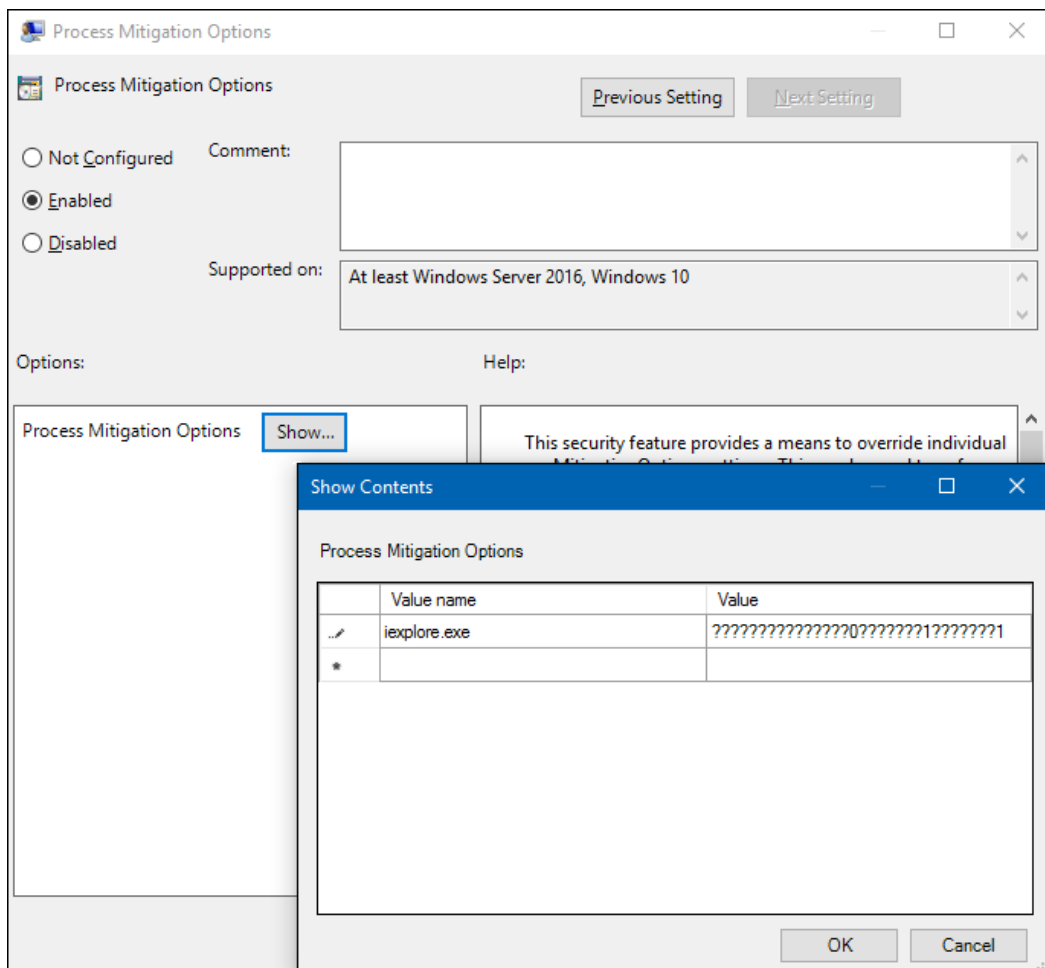
Important

For each app you want to include, you must include:

- **Value name.** The app file name, including the extension. For example, iexplore.exe.
- **Value.** A bit field with a series of bit flags in particular positions. Bits can be set to 0 (where the setting is forced off), 1 (where the setting is forced on), or ? (where the setting retains the previous, existing value).

Note

Setting bit flags in positions not specified here to anything other than ? might cause undefined behavior.



Setting the bit field

Here’s a visual representation of the bit flag locations for the various Process Mitigation Options settings:

	3								2							1														0	
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
?	?	?	?	?	?	?	?	?	?	?	?	F	E	?	?	?	?	?	?	?	?	D	?	?	?	?	?	C	B	A	

Where the bit flags are read from right to left and are defined as:

FLAG	BIT LOCATION	SETTING	DETAILS
A	0	PROCESS_CREATION_MITIGATION_TRANSMISSION_DATA_EXECUTION_PREVENTION (0x00000001)	Turns on Data Execution Prevention (DEP) for child processes.
B	1	PROCESS_CREATION_MITIGATION_TRANSMISSION_DATA_EXECUTION_EMULATION (0x00000002)	Turns on DEP-ATL thunk emulation for child processes. DEP-ATL thunk emulation lets the system intercept non-executable (NX) faults that originate from the Active Template Library (ATL) thunk layer, and then emulate and handle the instructions so the process can continue to run.

FLAG	BIT LOCATION	SETTING	DETAILS
C	2	PROCESS_CREATION_MITIGATION_POLICY_STRUCTURED_EXCEPTION_HANDLER_ALWAYS_ON (0x00000004)	Turns on Structured Exception Handler Overwrite Protection (SEHOP) for child processes. SEHOP helps to block exploits that use the Structured Exception Handler (SEH) overwrite technique.
D	8	PROCESS_CREATION_MITIGATION_POLICY_FORCE_RELOCATE_IMAGES_ALWAYS_ON (0x00001000)	Uses the Force Address Space Layout Randomization (ASLR) setting to act as though an image base collision happened at load time, forcibly rebasing images that aren't dynamic base compatible. Images without the base relocation section won't be loaded if relocations are required.
E	15	PROCESS_CREATION_MITIGATION_POLICY_BOTTOM_UP_ASLR_ALWAYS_ON (0x00010000)	Turns on the bottom-up randomization policy, which includes stack randomization options and causes a random location to be used as the lowest user address.
F	16	PROCESS_CREATION_MITIGATION_POLICY_BOTTOM_UP_ASLR_ALWAYS_OFF (0x00020000)	Turns off the bottom-up randomization policy, which includes stack randomization options and causes a random location to be used as the lowest user address.

Example

If you want to turn on the **PROCESS_CREATION_MITIGATION_POLICY_DEP_ENABLE** and **PROCESS_CREATION_MITIGATION_POLICY_FORCE_RELOCATE_IMAGES_ALWAYS_ON** settings, turn off the **PROCESS_CREATION_MITIGATION_POLICY_BOTTOM_UP_ASLR_ALWAYS_OFF** setting, and leave everything else as the default values, you'd want to type a value of `????????????0?????1?????1`.

Use Windows Event Forwarding to help with intrusion detection

7/1/2022 • 25 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows Server

Learn about an approach to collect events from devices in your organization. This article talks about events in both normal operations and when an intrusion is suspected.

Windows Event Forwarding (WEF) reads any operational or administrative event log on a device in your organization and forwards the events you choose to a Windows Event Collector (WEC) server.

To accomplish this functionality, there are two different subscriptions published to client devices - the Baseline subscription and the suspect subscription. The Baseline subscription enrolls all devices in your organization, and a Suspect subscription only includes devices that have been added by you. The Suspect subscription collects more events to help build context for system activity and can quickly be updated to accommodate new events and/or scenarios as needed without impacting baseline operations.

This implementation helps differentiate where events are ultimately stored. Baseline events can be sent to devices with online analytical capability, such as Security Event Manager (SEM), while also sending events to a MapReduce system, such as HDInsight or Hadoop, for long-term storage and deeper analysis. Events from the Suspect subscription are sent directly to a MapReduce system due to volume and lower signal/noise ratio, they are largely used for host forensic analysis.

An SEM's strength lies in being able to inspect, correlate events, and generate alerts for known patterns manner and alert security staff at machine speed.

A MapReduce system has a longer retention time (years versus months for an SEM), larger ingress ability (hundreds of terabytes per day), and the ability to perform more complex operations on the data like statistical and trend analysis, pattern clustering analysis, or apply Machine Learning algorithms.

Here's an approximate scaling guide for WEF events:

EVENTS/SECOND RANGE	DATA STORE
0 - 5,000	SQL or SEM
5,000 - 50,000	SEM
50,000+	Hadoop/HDInsight/Data Lake

Event generation on a device must be enabled either separately or as part of the GPO for the baseline WEF implementation, including enabling of disabled event logs and setting channel permissions. For more info, see [Appendix C - Event channel settings \(enable and channel access\) methods](#). This condition is because WEF is a passive system regarding the event log. It cannot change the size of event log files, enable disabled event channels, change channel permissions, or adjust a security audit policy. WEF only queries event channels for existing events. Additionally, having event generation already occurring on a device allows for more complete event collection building a complete history of system activity. Otherwise, you'll be limited to the speed of GPO

and WEF subscription refresh cycles to make changes to what is being generated on the device. On modern devices, enabling additional event channels and expanding the size of event log files hasn't resulted in noticeable performance differences.

For the minimum recommended audit policy and registry system ACL settings, see [Appendix A - Minimum recommended minimum audit policy](#) and [Appendix B - Recommended minimum registry system ACL policy](#).

Note: These are only minimum values need to meet what the WEF subscription selects.

From a WEF subscription management perspective, the event queries provided should be used in two separate subscriptions for ease of maintenance; only machines meeting specific criteria would be allowed access to the targeted subscription, this access would be determined by an algorithm or an analysts' direction. All devices should have access to the Baseline subscription.

This system of dual subscription means you would create two base subscriptions:

- **Baseline WEF subscription.** Events collected from all hosts; these events include some role-specific events, which will only be emitted by those machines.
- **Targeted WEF subscription.** Events collected from a limited set of hosts due to unusual activity and/or heightened awareness for those systems.

Each using the respective event query below. For the Targeted subscription enabling the "read existing events" option should be set to true to allow collection of existing events from systems. By default, WEF subscriptions will only forward events generated after the WEF subscription was received by the client.

In [Appendix E – Annotated Baseline Subscription Event Query](#) and [Appendix F – Annotated Suspect Subscription Event Query](#), the event query XML is included when creating WEF subscriptions. These subscriptions are annotated for query purpose and clarity. Individual <Query> element can be removed or edited without affecting the rest of the query.

Common WEF questions

This section addresses common questions from IT pros and customers.

Will the user notice if their machine is enabled for WEF or if WEF encounters an error?

The short answer is: No.

The longer answer is: The **Eventlog-forwardingPlugin/Operational** event channel logs the success, warning, and error events related to WEF subscriptions present on the device. Unless the user opens Event Viewer and navigates to that channel, they won't notice WEF either through resource consumption or Graphical User Interface pop-ups. Even if there is an issue with the WEF subscription, there is no user interaction or performance degradation. All success, warning, and failure events are logged to this operational event channel.

Is WEF Push or Pull?

A WEF subscription can be configured to be push or pull, but not both. The simplest, most flexible IT deployment with the greatest scalability can be achieved by using a push, or source initiated, subscription. WEF clients are configured by using a GPO and the built-in forwarding client is activated. For pull, collector initiated, the subscription on the WEC server is pre-configured with the names of the WEF Client devices from which events are to be selected. Those clients are to be configured ahead of time to allow the credentials used in the subscription to access their event logs remotely (normally by adding the credential to the **Event Log Readers** built-in local security group.) A useful scenario: closely monitoring a specific set of machines.

Will WEF work over VPN or RAS?

WEF handles VPN, RAS, and DirectAccess scenarios well and will reconnect and send any accumulated backlog of events when the connection to the WEF Collector is re-established.

How is client progress tracked?

The WEC server maintains in its registry the bookmark information and last heartbeat time for each event source for each WEF subscription. When an event source reconnects to a WEC server, the last bookmark position is sent to the device to use as a starting point to resume forwarding events. If a WEF client has no events to send, the WEF client will connect periodically to send a Heartbeat to the WEC server to indicate it is active. This heartbeat value can be individually configured for each subscription.

Will WEF work in an IPv4, IPv6, or mixed IPv4/IPv6 environment?

Yes. WEF is transport agnostic and will work over IPv4 or IPv6.

Are WEF events encrypted? I see an HTTP/HTTPS option!

In a domain setting, the connection used to transmit WEF events is encrypted using Kerberos, by default (with NTLM as a fallback option, which can be disabled by using a GPO). Only the WEF collector can decrypt the connection. Additionally, the connection between WEF client and WEC server is mutually authenticated regardless of authentication type (Kerberos or NTLM.) There are GPO options to force Authentication to use Kerberos Only.

This authentication and encryption is performed regardless if HTTP or HTTPS is selected.

The HTTPS option is available if certificate based authentication is used, in cases where the Kerberos based mutual authentication isn't an option. The SSL certificate and provisioned client certificates are used to provide mutual authentication.

Do WEF Clients have a separate buffer for events?

The WEF client machines local event log is the buffer for WEF for when the connection to the WEC server is lost. To increase the "buffer size", increase the maximum file size of the specific event log file where events are being selected. For more info, see [Appendix C – Event Channel Settings \(enable and Channel Access\) methods](#).

When the event log overwrites existing events (resulting in data loss if the device isn't connected to the Event Collector), there is no notification sent to the WEF collector that events are lost from the client. Neither is there an indicator that there was a gap encountered in the event stream.

What format is used for forwarded events?

WEF has two modes for forwarded events. The default is "Rendered Text" which includes the textual description of the event as you would see it in Event Viewer. This means that the event size is effectively doubled or tripled depending on the size of the rendered description. The alternative mode is "Events" (also sometimes referred to as "Binary" format) – which is just the event XML itself sent in binary XML format (as it would be written to the evtx file.) This is very compact and can more than double the event volume a single WEC server can accommodate.

A subscription "testSubscription" can be configured to use the Events format through the WECUTIL utility:

```
@rem required to set the DeliveryMaxItems or DeliveryMaxLatencyTime
Wecutil ss "testSubscription" /cf:Events
```

How frequently are WEF events delivered?

Event delivery options are part of the WEF subscription configuration parameters – There are three built-in subscription delivery options: Normal, Minimize Bandwidth, and Minimize Latency. A fourth, catch-all called "Custom" is available but cannot be selected or configured through the WEF UI by using Event Viewer. The Custom delivery option must be selected and configured using the WECUTIL.EXE command-line application. All subscription options define a maximum event count and maximum event age, if either limit is exceeded then the accumulated events are sent to the event collector.

This table outlines the built-in delivery options:

EVENT DELIVERY OPTIMIZATION OPTIONS	DESCRIPTION
Normal	This option ensures reliable delivery of events and doesn't attempt to conserve bandwidth. It is the appropriate choice unless you need tighter control over bandwidth usage or need forwarded events delivered as quickly as possible. It uses pull delivery mode, batches 5 items at a time and sets a batch timeout of 15 minutes.
Minimize bandwidth	This option ensures that the use of network bandwidth for event delivery is strictly controlled. It is an appropriate choice if you want to limit the frequency of network connections made to deliver events. It uses push delivery mode and sets a batch timeout of 6 hours. In addition, it uses a heartbeat interval of 6 hours.
Minimize latency	This option ensures that events are delivered with minimal delay. It is an appropriate choice if you are collecting alerts or critical events. It uses push delivery mode and sets a batch timeout of 30 seconds.

For more info about delivery options, see [Configure Advanced Subscription Settings](#).

The primary difference is in the latency which events are sent from the client. If none of the built-in options meet your requirements you can set Custom event delivery options for a given subscription from an elevated command prompt:

```
@rem required to set the DeliveryMaxItems or DeliveryMaxLatencyTime
Wecutil ss "SubscriptionNameGoesHere" /cm:Custom
@rem set DeliveryMaxItems to 1 event
Wecutil ss "SubscriptionNameGoesHere" /dmi:1
@rem set DeliveryMaxLatencyTime to 10 ms
Wecutil ss "SubscriptionNameGoesHere" /dmlt:10
```

How do I control which devices have access to a WEF Subscription?

For source initiated subscriptions: Each WEF subscription on a WEC server has its own ACL for machine accounts or security groups containing machine accounts (not user accounts) that are explicitly allowed to participate in that subscription or are explicitly denied access. This ACL applies to only a single WEF subscription (since there can be multiple WEF subscriptions on a given WEC server), other WEF Subscriptions have their own separate ACL.

For collector initiated subscriptions: The subscription contains the list of machines from which the WEC server is to collect events. This list is managed at the WEC server, and the credentials used for the subscription must have access to read event logs from the WEF Clients – the credentials can be either the machine account or a domain account.

Can a client communicate to multiple WEF Event Collectors?

Yes. If you desire a High-Availability environment, simply configure multiple WEC servers with the same subscription configuration and publish both WEC Server URIs to WEF clients. WEF Clients will forward events simultaneously to the configured subscriptions on the WEC servers, if they have the appropriate access.

What are the WEC server's limitations?

There are three factors that limit the scalability of WEC servers. The general rule for a stable WEC server on commodity hardware is planning for a total of 3,000 events per second on average for all configured subscriptions.

- **Disk I/O.** The WEC server doesn't process or validate the received event, but rather buffers the received

event and then logs it to a local event log file (EVTX file). The speed of logging to the EVTX file is limited by the disk write speed. Isolating the EVTX file to its own array or using high speed disks can increase the number of events per second that a single WEC server can receive.

- **Network Connections.** While a WEF source doesn't maintain a permanent, persistent connection to the WEC server, it doesn't immediately disconnect after sending its events. This means that the number of WEF sources that can simultaneously connect to the WEC server is limited to the open TCP ports available on the WEC server.
- **Registry size.** For each unique device that connects to a WEF subscription, there is a registry key (corresponding to the FQDN of the WEF Client) created to store bookmark and source heartbeat information. If this isn't pruned to remove inactive clients this set of registry keys can grow to an unmanageable size over time.
 - When a subscription has >1000 WEF sources connect to it over its operational lifetime, also known as lifetime WEF sources, Event Viewer can become unresponsive for a few minutes when selecting the **Subscriptions** node in the left-navigation, but will function normally afterwards.
 - At >50,000 lifetime WEF sources, Event Viewer is no longer an option and wecutil.exe (included with Windows) must be used to configure and manage subscriptions.
 - At >100,000 lifetime WEF sources, the registry won't be readable and the WEC server will likely have to be rebuilt.

Subscription information

Below lists all of the items that each subscription collects, the actual subscription XML is available in an Appendix. These are separated out into Baseline and Targeted. The intent is to subscribe all hosts to Baseline, and then enroll (and remove) hosts on an as needed basis to the Targeted subscription.

Baseline subscription

While this appears to be the largest subscription, it really is the lowest volume on a per-device basis. (Exceptions should be allowed for unusual devices – a device performing complex developer related tasks can be expected to create an unusually high volume of process create and AppLocker events.) This subscription doesn't require special configuration on client devices to enable event channels or modify channel permissions.

The subscription is essentially a collection of query statements applied to the Event Log. This means that it is modular in nature and a given query statement can be removed or changed without impacting other query statement in the subscription. Additionally, suppress statements which filter out specific events, only apply within that query statement and aren't to the entire subscription.

Baseline subscription requirements

To gain the most value out of the baseline subscription we recommend to have the following requirements set on the device to ensure that the clients are already generating the required events to be forwarded off the system.

- Apply a security audit policy that is a super-set of the recommended minimum audit policy. For more info, see [Appendix A – Minimum Recommended minimum Audit Policy](#). This ensures that the security event log is generating the required events.
- Apply at least an Audit-Only AppLocker policy to devices.
 - If you are already allowing or restricting events by using AppLocker, then this requirement is met.
 - AppLocker events contain extremely useful information, such as file hash and digital signature information for executables and scripts.
- Enable disabled event channels and set the minimum size for modern event files.
- Currently, there is no GPO template for enabling or setting the maximum size for the modern event files.

This must be done by using a GPO. For more info, see [Appendix C – Event Channel Settings \(enable and Channel Access\) methods](#).

The annotated event query can be found in the following. For more info, see [Appendix F – Annotated Suspect Subscription Event Query](#).

- Anti-malware events from Microsoft Antimalware or Windows Defender. This can be configured for any given anti-malware product easily if it writes to the Windows event log.
- Security event log Process Create events.
- AppLocker Process Create events (EXE, script, packaged App installation and execution).
- Registry modification events. For more info, see [Appendix B – Recommended minimum Registry System ACL Policy](#).
- OS startup and shutdown
 - Startup events include operating system version, service pack level, QFE version, and boot mode.
- Service install
 - Includes what the name of the service, the image path, and who installed the service.
- Certificate Authority audit events
 - This is only applicable on systems with the Certificate Authority role installed.
 - Logs certificate requests and responses.
- User profile events
 - Use of a temporary profile or unable to create a user profile may indicate an intruder is interactively logging into a device but not wanting to leave a persistent profile behind.
- Service start failure
 - Failure codes are localized, so you have to check the message DLL for values.
- Network share access events
 - Filter out IPC\$ and /NetLogon file shares, which are expected and noisy.
- System shutdown initiate requests
 - Find out what initiated the restart of a device.
- User initiated interactive logoff event
- Remote Desktop Services sessions connect, reconnect, or disconnect.
- EMET events, if EMET is installed.
- Event forwarding plugin events
 - For monitoring WEF subscription operations, particularly Partial Success events. This is useful for diagnosing deployment issues.
- Network share creation and deletion
 - Enables detection of unauthorized share creation.

Note: All shares are re-created when the device starts.

- Logon sessions
 - Logon success for interactive (local and Remote Interactive/Remote Desktop)
 - Logon success for services for non-built-in accounts, such as LocalSystem, LocalNetwork, and so on.

- Logon success for batch sessions
- Logon session close, which is logoff events for non-network sessions.
- Windows Error Reporting (Application crash events only)
 - This can help detect early signs of intruder not familiar with enterprise environment using targeted malware.
- Event log service events
 - Errors, start events, and stop events for the Windows Event Log service.
- Event log cleared (including the Security Event Log)
 - This could indicate an intruder that is covering their tracks.
- Special privileges assigned to new logon
 - This indicates that at the time of logon a user is either an Administrator or has the sufficient access to make themselves Administrator.
- Outbound Remote Desktop Services session attempts
 - Visibility into potential beachhead for intruder
- System time changed
- SMB Client (mapped drive connections)
- Account credential validation
 - Local accounts or domain accounts on domain controllers
- A user was added or removed from the local Administrators security group.
- Crypto API private key accessed
 - Associated with signing objects using the locally stored private key.
- Task Scheduler task creation and delete
 - Task Scheduler allows intruders to run code at specified times as LocalSystem.
- Logon with explicit credentials
 - Detect credential use changes by intruders to access more resources.
- Smartcard card holder verification events
 - This detects when a smartcard is being used.

Suspect subscription

This adds some possible intruder-related activity to help analyst further refine their determinations about the state of the device.

- Logon session creation for network sessions
 - Enables time-series analysis of network graphs.
- RADIUS and VPN events
 - Useful if you use a Microsoft IAS RADIUS/VPN implementation. It shows user-> IP address assignment with remote IP address connecting to the enterprise.
- Crypto API X509 object and build chain events
 - Detects known bad certificate, CA, or sub-CA
 - Detects unusual process use of CAPI
- Groups assigned to local logon

- Gives visibility to groups which enable account-wide access
- Allows better planning for remediation efforts
- Excludes well known, built-in system accounts.
- Logon session exit
 - Specific for network logon sessions.
- Client DNS lookup events
 - Returns what process performed a DNS query and the results returned from the DNS server.
- Process exit
 - Enables checking for processes terminating unexpectedly.
- Local credential validation or logon with explicit credentials
 - Generated when the local SAM is authoritative for the account credentials being authenticated.
 - Noisy on domain controllers
 - On client devices this is only generated when local accounts log on.
- Registry modification audit events
 - Only when a registry value is being created, modified, or deleted.
- Wireless 802.1x authentication
 - Detect wireless connection with a peer MAC address
- Windows PowerShell logging
 - Covers Windows PowerShell 2.0 and later and includes the Windows PowerShell 5.0 logging improvements for in-memory attacks using Windows PowerShell.
 - Includes Windows PowerShell remoting logging
- User Mode Driver Framework "Driver Loaded" event
 - Can possibly detect a USB device loading multiple device drivers. For example, a USB_STOR device loading the keyboard or network driver.

Appendix A - Minimum recommended minimum audit policy

If your organizational audit policy enables more auditing to meet its needs, that is fine. The policy below is the minimum audit policy settings needed to enable events collected by both baseline and targeted subscriptions.

CATEGORY	SUBCATEGORY	AUDIT SETTINGS
Account Logon	Credential Validation	Success and Failure
Account Management	Security Group Management	Success
Account Management	User Account Management	Success and Failure
Account Management	Computer Account Management	Success and Failure
Account Management	Other Account Management Events	Success and Failure
Detailed Tracking	Process Creation	Success
Detailed Tracking	Process Termination	Success

CATEGORY	SUBCATEGORY	AUDIT SETTINGS
Logon/Logoff	User/Device Claims	Not configured
Logon/Logoff	IPsec Extended Mode	Not configured
Logon/Logoff	IPsec Quick Mode	Not configured
Logon/Logoff	Logon	Success and Failure
Logon/Logoff	Logoff	Success
Logon/Logoff	Other Logon/Logoff Events	Success and Failure
Logon/Logoff	Special Logon	Success and Failure
Logon/Logoff	Account Lockout	Success
Object Access	Application Generated	Not configured
Object Access	File Share	Success
Object Access	File System	Not configured
Object Access	Other Object Access Events	Not configured
Object Access	Registry	Not configured
Object Access	Removable Storage	Success
Policy Change	Audit Policy Change	Success and Failure
Policy Change	MPSSVC Rule-Level Policy Change	Success and Failure
Policy Change	Other Policy Change Events	Success and Failure
Policy Change	Authentication Policy Change	Success and Failure
Policy Change	Authorization Policy Change	Success and Failure
Privilege Use	Sensitive Privilege Use	Not configured
System	Security State Change	Success and Failure
System	Security System Extension	Success and Failure
System	System Integrity	Success and Failure

Appendix B - Recommended minimum registry system ACL policy

The Run and RunOnce keys are useful for intruders and malware persistence. It allows code to be run (or run only once then removed, respectively) when a user logs into the system.

This can easily be extended to other Auto-Execution Start Points keys in the registry.

Use the following figures to see how you can configure those registry keys.

MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run			
Configure this key then: Propagate inheritable permissions to all subkeys			
Owner			
Permissions			
Type	Name	Permission	Apply To
Allow	BUILTIN\Administrators	Full control	This key and subkeys
Allow	CREATOR OWNER	Full control	Subkeys only
Allow	NT AUTHORITY\SYSTEM	Full control	This key and subkeys
Allow	BUILTIN\Users	Read	This key and subkeys
Allow	APPLICATION PACKAGE AUTHORITY\ALL APPLICATION PACKAGES	Read	This key and subkeys
Allow inheritable permissions from the parent to propagate to this object and all child objects		Disabled	
Auditing			
Type	Name	Access	Apply To
Success	NT AUTHORITY\Authenticated Users	Set Value, Create Subkey	This key and subkeys
Allow inheritable auditing entries from the parent to propagate to this object and all child objects		Enabled	

MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnce			
Configure this key then: Propagate inheritable permissions to all subkeys			
Owner			
Permissions			
Type	Name	Permission	Apply To
Allow	BUILTIN\Administrators	Full control	This key and subkeys
Allow	CREATOR OWNER	Full control	Subkeys only
Allow	NT AUTHORITY\SYSTEM	Full control	This key and subkeys
Allow	BUILTIN\Users	Read	This key and subkeys
Allow	APPLICATION PACKAGE AUTHORITY\ALL APPLICATION PACKAGES	Read	This key and subkeys
Allow inheritable permissions from the parent to propagate to this object and all child objects		Disabled	
Auditing			
Type	Name	Access	Apply To
Success	NT AUTHORITY\Authenticated Users	Set Value, Create Subkey, Delete	This key and subkeys
Allow inheritable auditing entries from the parent to propagate to this object and all child objects		Enabled	

Appendix C - Event channel settings (enable and channel access) methods

Some channels are disabled by default and have to be enabled. Others, such as Microsoft-Windows-CAPI2/Operational must have the channel access modified to allow the Event Log Readers built-in security group to read from it.

The recommended and most effective way to do this is configuring the baseline GPO to run a scheduled task to configure the event channels (enable, set maximum size, and adjust channel access.) This will take effect at the next GPO refresh cycle and has minimal impact on the client device.

The following GPO snippet performs the following:

- Enables the **Microsoft-Windows-Capi2/Operational** event channel.
- Sets the maximum file size for **Microsoft-Windows-Capi2/Operational** to 100MB.
- Sets the maximum file size for **Microsoft-Windows-AppLocker/EXE and DLL** to 100MB.
- Sets the maximum channel access for **Microsoft-Windows-Capi2/Operational** to include the built-in Event Log Readers security group.
- Enables the **Microsoft-Windows-DriverFrameworks-UserMode/Operational** event channel.
- Sets the maximum file size for **Microsoft-Windows-DriverFrameworks-UserMode/Operational** to 50MB.

Scheduled Tasks		hide
Immediate Task (At least Windows 7) (Name: Enable configure event channels)		hide
Enable configure event channels (Order: 1)		hide
General		hide
Task		
Name	Enable configure event channels	
Author	[REDACTED]	
Description		
Run only when user is logged on		
GroupId	NT AUTHORITY\SYSTEM	
Run with highest privileges	HighestAvailable	
Hidden	Yes	
Configure for	1.2	
Enabled	Yes	
Actions		
1. Start a program	Program/script	%SystemRoot%\System32\Wevtutil.exe
	Arguments	sl Microsoft-Windows-Capi2/Operational /e.true
2. Start a program	Program/script	%systemroot%\system32\wevtutil.exe
	Arguments	sl Microsoft-Windows-Capi2/Operational /ms:102432768
3. Start a program	Program/script	%SystemRoot%\System32\wevtutil.exe
	Arguments	sl "Microsoft-Windows-AppLocker/EXE and DLL" /ms:102432768
4. Start a program	Program/script	%SystemRoot%\System32\Wevtutil.exe
	Arguments	sl Microsoft-Windows-Capi2/Operational /ca:"O:BAG:SYD:(A;;0x7;;;BA)(A;;0x2;;;AU)(A;;0x1;;;S-1-5-32-573)"
5. Start a program	Program/script	%systemroot%\system32\wevtutil.exe
	Arguments	sl "Microsoft-Windows-DriverFrameworks-UserMode/Operational" /e.true
6. Start a program	Program/script	%SystemRoot%\system32\wevtutil.exe
	Arguments	sl "Microsoft-Windows-DriverFrameworks-UserMode/Operational" /ms:52432896

Appendix D - Minimum GPO for WEF Client configuration

Here are the minimum steps for WEF to operate:

1. Configure the collector URI(s).
2. Start the WinRM service.
3. Add the Network Service account to the built-in Event Log Readers security group. This allows reading from secured event channel, such as the security event channel.

Computer Configuration (Enabled) [hide](#)

Policies [hide](#)

Windows Settings [hide](#)

Security Settings [hide](#)

System Services [hide](#)

Windows Remote Management (WS-Management) (Startup Mode: Automatic) [hide](#)

Permissions
No permissions specified

Auditing
No auditing specified

Administrative Templates [hide](#)

Policy definitions (ADMX files) retrieved from the local computer.

Windows Components/Event Forwarding [hide](#)

Policy	Setting	Comment
Configure target Subscription Manager	Enabled	

SubscriptionManagers
 Server=http://WEFCollector01.corp.contoso.com:5985/wsman/SubscriptionManager/WEC
 Server=http://WEFCollector02.corp.contoso.com:5985/wsman/SubscriptionManager/WEC
 Server=http://WEFCollector03.corp.contoso.com:5985/wsman/SubscriptionManager/WEC

Preferences [hide](#)

Control Panel Settings [hide](#)

Local Users and Groups [hide](#)

Group (Name: Event Log Readers (built-in)) [hide](#)

Event Log Readers (built-in) (Order: 1) [hide](#)

Local Group [hide](#)

Action	Update
Properties	
Group name	Event Log Readers (built-in)
Delete all member users	Disabled
Delete all member groups	Disabled
Add members	
BUILTIN\NETWORK SERVICE	S-1-5-20
Common hide	
Options	
Stop processing items on this extension if an error occurs on this item	No
Remove this item when it is no longer applied	No
Apply once and do not reapply	No

User Configuration (Disabled) [hide](#)

No settings defined.

Appendix E – Annotated baseline subscription event query

```

<QueryList>
  <Query Id="0" Path="System">
    <!-- Anti-malware *old* events, but only detect events (cuts down noise) -->
    <Select Path="System">*[System[Provider[@Name='Microsoft Antimalware'] and (EventID &gt;= 1116 and
EventID &lt;= 1119)]]</Select>
  </Query>
  <!-- AppLocker EXE events or Script events -->
  <Query Id="1" Path="Microsoft-Windows-AppLocker/EXE and DLL">
    <Select Path="Microsoft-Windows-AppLocker/EXE and DLL">*[UserData[RuleAndFileData[PolicyName="EXE"]]]
  </Select>
    <Select Path="Microsoft-Windows-AppLocker/MSI and Script">*</Select>
  </Query>
  <Query Id="2" Path="Security">
    <!-- Wireless Lan 802.1x authentication events with Peer MAC address -->
    <Select Path="Security">*[System[(EventID=5632)]]</Select>
  </Query>
  <Query Id="3" Path="Microsoft-Windows-TaskScheduler/Operational">
    <!-- Task scheduler Task Registered (106), Task Registration Deleted (141), Task Deleted (142) -->
    <Select Path="Microsoft-Windows-TaskScheduler/Operational">*[System[Provider[@Name='Microsoft-Windows-
TaskScheduler'] and (EventID=106 or EventID=141 or EventID=142 )]]</Select>
    <Select Path="System">*[System[Provider[@Name='Microsoft-Windows-TaskScheduler'] and (EventID=106 or
EventID=141 or EventID=142 )]]</Select>
  </Query>
  <Query Id="4" Path="System">
    <!-- System startup (12 - includes OS/SP/Version) and shutdown -->
    <Select Path="System">*[System[Provider[@Name='Microsoft-Windows-Kernel-General'] and (EventID=12 or
EventID=13)]]</Select>
  </Query>
  <Query Id="5" Path="System">

```

```

    <!-- Service Install (7000), service start failure (7045), new service (4697) -->
    <Select Path="System">*[System[Provider[@Name='Service Control Manager'] and (EventID = 7000 or
EventID=7045)]]</Select>
<Select Path="Security">*[System[(EventID=4697)]]</Select>
</Query>
<Query Id="6" Path="Security">
    <!-- TS Session reconnect (4778), TS Session disconnect (4779) -->
    <Select Path="Security">*[System[(EventID=4778 or EventID=4779)]]</Select>
</Query>
<Query Id="7" Path="Security">
    <!-- Network share object access without IPC$ and Netlogon shares -->
    <Select Path="Security">*[System[(EventID=5140)]] and (*
[EventData[Data[@Name="ShareName"]!="\\*\IPC$]]) and (*
[EventData[Data[@Name="ShareName"]!="\\*\NetLogon"]])</Select>
</Query>
<Query Id="8" Path="Security">
    <!-- System Time Change (4616) -->
    <Select Path="Security">*[System[(EventID=4616)]]</Select>
</Query>
<Query Id="9" Path="System">
    <!-- Shutdown initiate requests, with user, process and reason (if supplied) -->
    <Select Path="System">*[System[Provider[@Name='USER32'] and (EventID=1074)]]</Select>
</Query>
<!-- AppLocker packaged (Modern UI) app execution -->
<Query Id="10" Path="Microsoft-Windows-AppLocker/Packaged app-Execution">
    <Select Path="Microsoft-Windows-AppLocker/Packaged app-Execution">*</Select>
</Query>
<!-- AppLocker packaged (Modern UI) app installation -->
<Query Id="11" Path="Microsoft-Windows-AppLocker/Packaged app-Deployment">
    <Select Path="Microsoft-Windows-AppLocker/Packaged app-Deployment">*</Select>
</Query>
<Query Id="12" Path="Application">
    <!-- EMET events -->
    <Select Path="Application">*[System[Provider[@Name='EMET']]]</Select>
</Query>
<Query Id="13" Path="System">
    <!-- Event log service events -->
    <Select Path="System">*[System[Provider[@Name='Microsoft-Windows-Eventlog']]]</Select>
</Query>
<Query Id="14" Path="Security">
    <!-- Local logons without network or service events -->
    <Select Path="Security">*[System[(EventID=4624)]] and (*[EventData[Data[@Name="LogonType"]!="3"]]) and
(*[EventData[Data[@Name="LogonType"]!="5"]])</Select>
</Query>
<Query Id="15" Path="Application">
    <!-- WER events for application crashes only -->
    <Select Path="Application">*[System[Provider[@Name='Windows Error Reporting']]] and (*[EventData[Data[3]
="APPCRASH"]])</Select>
</Query>
<Query Id="16" Path="Security">
    <!-- Security Log cleared events (1102), EventLog Service shutdown (1100)-->
    <Select Path="Security">*[System[(EventID=1102 or EventID = 1100)]]</Select>
</Query>
<Query Id="17" Path="System">
    <!-- Other Log cleared events (104)-->
    <Select Path="System">*[System[(EventID=104)]]</Select>
</Query>
<Query Id="18" Path="Security">
    <!-- user initiated logoff -->
    <Select Path="Security">*[System[(EventID=4647)]]</Select>
</Query>
<Query Id="19" Path="Security">
    <!-- user logoff for all non-network logon sessions-->
    <Select Path="Security">*[System[(EventID=4634)]] and (*[EventData[Data[@Name="LogonType"] != "3"]])
</Select>
</Query>
<Query Id="20" Path="Security">
    <!-- Service logon events if the user account isn't LocalSystem, NetworkService, LocalService -->
    <Select Path="Security">*[System[(EventID=4624)]] and (*[EventData[Data[@Name="LogonType"]="5"]]) and (*

```

```

[EventData[Data[@Name="TargetUserSid"] != "S-1-5-18"]] and (*[EventData[Data[@Name="TargetUserSid"] != "S-
1-5-19"]]) and (*[EventData[Data[@Name="TargetUserSid"] != "S-1-5-20"]])</Select>
</Query>
<Query Id="21" Path="Security">
  <!-- Network Share create (5142), Network Share Delete (5144) -->
  <Select Path="Security">*[System[(EventID=5142 or EventID=5144)]]</Select>
</Query>
<Query Id="22" Path="Security">
  <!-- Process Create (4688) -->
  <Select Path="Security">*[System[EventID=4688]]</Select>
</Query>
<Query Id="23" Path="Security">
  <!-- Event log service events specific to Security channel -->
  <Select Path="Security">*[System[Provider[@Name='Microsoft-Windows-Eventlog']]]</Select>
</Query>
<Query Id="26" Path="Security">
  <!-- Special Privileges (Admin-equivalent Access) assigned to new logon, excluding LocalSystem-->
  <Select Path="Security">*[System[(EventID=4672)]]</Select>
  <Suppress Path="Security">*[EventData[Data[1]="S-1-5-18"]]</Suppress>
</Query>
<Query Id="27" Path="Security">
  <!-- New user added to local security group-->
  <Select Path="Security">*[System[(EventID=4732)]]</Select>
</Query>
<Query Id="28" Path="Security">
  <!-- New user added to global security group-->
  <Select Path="Security">*[System[(EventID=4728)]]</Select>
</Query>
<Query Id="29" Path="Security">
  <!-- New user added to universal security group-->
  <Select Path="Security">*[System[(EventID=4756)]]</Select>
</Query>
<Query Id="30" Path="Security">
  <!-- User removed from local Administrators group-->
  <Select Path="Security">*[System[(EventID=4733)]] and (*
[EventData[Data[@Name="TargetUserName"]="Administrators"]])</Select>
</Query>
<Query Id="31" Path="Microsoft-Windows-TerminalServices-RDPClient/Operational">
  <!-- Log attempted TS connect to remote server -->
  <Select Path="Microsoft-Windows-TerminalServices-RDPClient/Operational">*[System[(EventID=1024)]]
</Select>
</Query>
<Query Id="32" Path="Security">
  <!-- Certificate Services received certificate request (4886), Approved and Certificate issued (4887),
Denied request (4888) -->
  <Select Path="Security">*[System[(EventID=4886 or EventID=4887 or EventID=4888)]]</Select>
</Query>
<Query Id="34" Path="Security">
  <!-- New User Account Created(4720), User Account Enabled (4722), User Account Disabled (4725), User
Account Deleted (4726) -->
  <Select Path="Security">*[System[(EventID=4720 or EventID=4722 or EventID=4725 or EventID=4726)]]
</Select>
</Query>
<Query Id="35" Path="Microsoft-Windows-SmartCard-Audit/Authentication">
  <!-- Gets all Smart-card Card-Holder Verification (CHV) events (success and failure) performed on the
host. -->
  <Select Path="Microsoft-Windows-SmartCard-Audit/Authentication">*</Select>
</Query>
<Query Id="36" Path="Microsoft-Windows-SMBClient/Operational">
  <!-- get all UNC/mapped drive successful connection -->
  <Select Path="Microsoft-Windows-SMBClient/Operational">*[System[(EventID=30622 or EventID=30624)]]
</Select>
</Query>
<Query Id="37" Path="Application">
  <!-- User logging on with Temporary profile (1511), cannot create profile, using temporary profile
(1518)-->
  <Select Path="Application">*[System[Provider[@Name='Microsoft-Windows-User Profiles Service'] and
(EventID=1511 or EventID=1518)]]</Select>
</Query>

```

```

<Query Id="39" Path="Microsoft-Windows-Sysmon/Operational">
  <!-- Modern SysMon event provider-->
  <Select Path="Microsoft-Windows-Sysmon/Operational">*</Select>
</Query>
<Query Id="40" Path="Application">
  <!-- Application crash/hang events, similar to WER/1001. These include full path to faulting
  EXE/Module.-->
  <Select Path="Application">*[System[Provider[@Name='Application Error'] and (EventID=1000)]]</Select>
  <Select Path="Application">*[System[Provider[@Name='Application Hang'] and (EventID=1002)]]</Select>
</Query>
<Query Id="41" Path="Microsoft-Windows-Windows Defender/Operational">
  <!-- Modern Windows Defender event provider Detection events (1006-1009) and (1116-1119) -->
  <Select Path="Microsoft-Windows-Windows Defender/Operational">*[System[( (EventID &gt;= 1006 and EventID
  &lt;= 1009) )]]</Select>
  <Select Path="Microsoft-Windows-Windows Defender/Operational">*[System[( (EventID &gt;= 1116 and EventID
  &lt;= 1119) )]]</Select>
</Query>
<Query Id="42" Path="Security">
  <!-- An account Failed to Log on events -->
  <Select Path="Security">*[System[(EventID=4625)]] and (*[EventData[Data[@Name="LogonType"]!="2"]])
</Select>
</Query>

</QueryList>

```

Appendix F – Annotated Suspect Subscription Event Query

```

<QueryList>
  <Query Id="0" Path="Security">
    <!-- Network logon events-->
    <Select Path="Security">*[System[(EventID=4624)]] and (*[EventData[Data[@Name="LogonType"]="3"]])
  </Select>
  </Query>
  <Query Id="1" Path="System">
    <!-- RADIUS authentication events User Assigned IP address (20274), User successfully authenticated
    (20250), User Disconnected (20275) -->
    <Select Path="System">*[System[Provider[@Name='RemoteAccess'] and (EventID=20274 or EventID=20250 or
    EventID=20275)]]</Select>
  </Query>
  <Query Id="2" Path="Microsoft-Windows-CAPI2/Operational">
    <!-- CAPI events Build Chain (11), Private Key accessed (70), X509 object (90)-->
    <Select Path="Microsoft-Windows-CAPI2/Operational">*[System[(EventID=11 or EventID=70 or EventID=90)]]
  </Select>
  </Query>
  <Query Id="3" Path="Security">
    <!-- CA stop/Start events CA Service Stopped (4880), CA Service Started (4881), CA DB row(s) deleted
    (4896), CA Template loaded (4898) -->
    <Select Path="Security">*[System[(EventID=4880 or EventID = 4881 or EventID = 4896 or EventID = 4898)]]
  </Select>
  </Query>
  <Query Id="4" Path="Microsoft-Windows-LSA/Operational">
    <!-- Groups assigned to new login (except for well known, built-in accounts)-->
    <Select Path="Microsoft-Windows-LSA/Operational">*[System[(EventID=300)]] and (*
  [EventData[Data[@Name="TargetUserSid"] != "S-1-5-20"]]) and (*[EventData[Data[@Name="TargetUserSid"] != "S-
  1-5-18"]]) and (*[EventData[Data[@Name="TargetUserSid"] != "S-1-5-19"]])</Select>
  </Query>
  <Query Id="5" Path="Security">
    <!-- Logoff events - for Network Logon events-->
    <Select Path="Security">*[System[(EventID=4634)]] and (*[EventData[Data[@Name="LogonType"] = "3"]])
  </Select>
  </Query>
  <Query Id="6" Path="Security">
    <!-- RRAS events - only generated on Microsoft IAS server -->
    <Select Path="Security">*[System[( (EventID &gt;= 6272 and EventID &lt;= 6280) )]]</Select>
  </Query>
  <Query Id="7" Path="Microsoft-Windows-DNS-Client/Operational">

```

```

    <!-- DNS Client events Query Completed (3008) -->
    <Select Path="Microsoft-Windows-DNS-Client/Operational">*[System[(EventID=3008)]]</Select>
<!-- suppresses local machine name resolution events -->
<Suppress Path="Microsoft-Windows-DNS-Client/Operational">*
[EventData[Data[@Name="QueryOptions"]="140737488355328"]]</Suppress>
<!-- suppresses empty name resolution events -->
<Suppress Path="Microsoft-Windows-DNS-Client/Operational">*[EventData[Data[@Name="QueryResults"]=""]]
</Suppress>
</Query>
<Query Id="8" Path="Security">
    <!-- Process Terminate (4689) -->
    <Select Path="Security">*[System[(EventID = 4689)]]</Select>
</Query>
<Query Id="9" Path="Security">
    <!-- Local credential authentication events (4776), Logon with explicit credentials (4648) -->
    <Select Path="Security">*[System[(EventID=4776 or EventID=4648)]]</Select>
</Query>
<Query Id="10" Path="Security">
    <!-- Registry modified events for Operations: New Registry Value created (%%1904), Existing Registry
Value modified (%%1905), Registry Value Deleted (%%1906) -->
    <Select Path="Security">*[System[(EventID=4657)]] and ((*[EventData[Data[@Name="OperationType"] =
"%%1904"]]) or (*[EventData[Data[@Name="OperationType"] = "%%1905"]]) or (*
[EventData[Data[@Name="OperationType"] = "%%1906"]]))</Select>
</Query>
<Query Id="11" Path="Security">
    <!-- Request made to authenticate to Wireless network (including Peer MAC (5632) -->
    <Select Path="Security">*[System[(EventID=5632)]]</Select>
</Query>
<Query Id="12" Path="Microsoft-Windows-PowerShell/Operational">
    <!-- PowerShell execute block activity (4103), Remote Command(4104), Start Command(4105), Stop
Command(4106) -->
    <Select Path="Microsoft-Windows-PowerShell/Operational">*[System[(EventID=4103 or EventID=4104 or
EventID=4105 or EventID=4106)]]</Select>
</Query>
<Query Id="13" Path="Microsoft-Windows-DriverFrameworks-UserMode/Operational">
    <!-- Detect User-Mode drivers loaded - for potential BadUSB detection. -->
    <Select Path="Microsoft-Windows-DriverFrameworks-UserMode/Operational">*[System[(EventID=2004)]]
</Select>
</Query>
<Query Id="14" Path="Windows PowerShell">
    <!-- Legacy PowerShell pipeline execution details (800) -->
    <Select Path="Windows PowerShell">*[System[(EventID=800)]]</Select>
</Query>
</QueryList>

```

Appendix G - Online resources

You can get more info with the following links:

- [Event Selection](#)
- [Event Queries and Event XML](#)
- [Event Query Schema](#)
- [Windows Event Collector](#)
- [4625\(F\): An account failed to log on](#)

Block untrusted fonts in an enterprise

7/1/2022 • 5 minutes to read • [Edit Online](#)

Applies to:

- Windows 10

Learn more about what features and functionality are supported in each Windows edition at [Compare Windows 10 Editions](#).

To help protect your company from attacks which may originate from untrusted or attacker-controlled font files, we've created the Blocking Untrusted Fonts feature. Using this feature, you can turn on a global setting that stops your employees from loading untrusted fonts processed using the Graphics Device Interface (GDI) onto your network. Untrusted fonts are any font installed outside of the `%windir%/Fonts` directory. Blocking untrusted fonts helps prevent both remote (web-based or email-based) and local EOP attacks that can happen during the font file-parsing process.

What does this mean for me?

Blocking untrusted fonts helps improve your network and employee protection against font-processing-related attacks. By default, this feature is not turned on.

How does this feature work?

There are 3 ways to use this feature:

- **On.** Helps stop any font processed using GDI from loading outside of the `%windir%/Fonts` directory. It also turns on event logging.
- **Audit.** Turns on event logging, but doesn't block fonts from loading, regardless of location. The name of the apps that use untrusted fonts appear in your event log.

NOTE

If you aren't quite ready to deploy this feature into your organization, you can run it in Audit mode to see if not loading untrusted fonts causes any usability or compatibility issues.

- **Exclude apps to load untrusted fonts.** You can exclude specific apps, allowing them to load untrusted fonts, even while this feature is turned on. For instructions, see [Fix apps having problems because of blocked fonts](#).

Potential reductions in functionality

After you turn this feature on, your employees might experience reduced functionality when:

- Sending a print job to a remote printer server that uses this feature and where the spooler process hasn't been specifically excluded. In this situation, any fonts that aren't already available in the server's `%windir%/Fonts` folder won't be used.
- Printing using fonts provided by the installed printer's graphics .dll file, outside of the `%windir%/Fonts` folder. For more information, see [Introduction to Printer Graphics DLLs](#).

- Using first or third-party apps that use memory-based fonts.
- Using Internet Explorer to look at websites that use embedded fonts. In this situation, the feature blocks the embedded font, causing the website to use a default font. However, not all fonts have all of the characters, so the website might render differently.
- Using desktop Office to look at documents with embedded fonts. In this situation, content shows up using a default font picked by Office.

Turn on and use the Blocking Untrusted Fonts feature

Use Group Policy or the registry to turn this feature on, off, or to use audit mode.

To turn on and use the Blocking Untrusted Fonts feature through Group Policy

1. Open the Group Policy editor (gpedit.msc) and go to

`Computer Configuration\Administrative Templates\System\Mitigation Options\Untrusted Font Blocking`.

2. Click **Enabled** to turn the feature on, and then click one of the following **Mitigation Options**:

- **Block untrusted fonts and log events.** Turns the feature on, blocking untrusted fonts and logging installation attempts to the event log.
- **Do not block untrusted fonts.** Turns the feature on, but doesn't block untrusted fonts nor does it log installation attempts to the event log.
- **Log events without blocking untrusted fonts.** Turns the feature on, logging installation attempts to the event log, but not blocking untrusted fonts.

3. Click **OK**.

To turn on and use the Blocking Untrusted Fonts feature through the registry To turn this feature on, off, or to use audit mode:

1. Open the registry editor (regedit.exe) and go to

`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\Kernel`.

2. If the **MitigationOptions** key isn't there, right-click and add a new **QWORD (64-bit) Value**, renaming it to **MitigationOptions**.
3. Right click on the **MitigationOptions** key, and then click **Modify**.

The **Edit QWORD (64-bit) Value** box opens.

4. Make sure the **Base** option is **Hexadecimal**, and then update the **Value data**, making sure you keep your existing value, like in the important note below:
 - **To turn this feature on.** Type `1000000000000`.
 - **To turn this feature off.** Type `2000000000000`.
 - **To audit with this feature.** Type `3000000000000`.

IMPORTANT

Your existing **MitigationOptions** values should be saved during your update. For example, if the current value is `1000`, your updated value should be `1000000001000`.

5. Restart your computer.

View the event log

After you turn this feature on, or start using Audit mode, you can look at your event logs for details.

To look at your event log

1. Open the event viewer (eventvwr.exe) and go to **Application and Service Logs/Microsoft/Windows/Win32k/Operational**.
2. Scroll down to **EventID: 260** and review the relevant events.

Event Example 1 - MS Word

WINWORD.EXE attempted loading a font that is restricted by font-loading policy.

FontType: Memory

FontPath:

Blocked: true

NOTE

Because the **FontType** is *Memory*, there's no associated **FontPath**.

Event Example 2 - Winlogon

Winlogon.exe attempted loading a font that is restricted by font-loading policy.

FontType: File

FontPath: \??\C:\PROGRAM FILES (X86)\COMMON FILES\MICROSOFT SHARED\EQUATION\MTEXTRA.TTF

Blocked: true

NOTE

Because the **FontType** is *File*, there's also an associated **FontPath**.

Event Example 3 - Internet Explorer running in Audit mode

explore.exe attempted loading a font that is restricted by font-loading policy.

FontType: Memory

FontPath:

Blocked: false

NOTE

In Audit mode, the problem is recorded, but the font isn't blocked.

Fix apps having problems because of blocked fonts

Your company may still need apps that are having problems because of blocked fonts, so we suggest that you first run this feature in Audit mode to determine which fonts are causing the problems.

After you figure out the problematic fonts, you can try to fix your apps in 2 ways: by directly installing the fonts into the %windir%/Fonts directory or by excluding the underlying processes and letting the fonts load. As the default solution, we highly recommend that you install the problematic font. Installing fonts is safer than excluding apps because excluded apps can load any font, trusted or untrusted.

To fix your apps by installing the problematic fonts (recommended)

- On each computer with the app installed, right-click on the font name and click **Install**.

The font should automatically install into your `%windir%/Fonts` directory. If it doesn't, you'll need to manually copy the font files into the **Fonts** directory and run the installation from there.

To fix your apps by excluding processes

1. On each computer with the app installed, open `regedit.exe` and go to

```
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\  
<process_image_name>
```

.

For example, if you want to exclude Microsoft Word processes, you'd use

```
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Image File Execution  
Options\Winword.exe
```

.

2. Add any additional processes that need to be excluded here, and then turn the Blocking untrusted fonts feature on, using the steps in [Turn on and use the Blocking Untrusted Fonts feature](#), earlier in this article.

Related content

- [Dropping the "Untrusted Font Blocking" setting](#)

Protect your enterprise data using Windows Information Protection (WIP)

7/1/2022 • 13 minutes to read • [Edit Online](#)

Applies to:

- Windows 10, version 1607 and later

Learn more about what features and functionality are supported in each Windows edition at [Compare Windows 10 Editions](#).

With the increase of employee-owned devices in the enterprise, there's also an increasing risk of accidental data leak through apps and services, like email, social media, and the public cloud, which are outside of the enterprise's control. For example, when an employee sends the latest engineering pictures from their personal email account, copies and pastes product info into a tweet, or saves an in-progress sales report to their public cloud storage.

Windows Information Protection (WIP), previously known as enterprise data protection (EDP), helps to protect against this potential data leakage without otherwise interfering with the employee experience. WIP also helps to protect enterprise apps and data against accidental data leak on enterprise-owned devices and personal devices that employees bring to work without requiring changes to your environment or other apps. Finally, another data protection technology, Azure Rights Management also works alongside WIP to extend data protection for data that leaves the device, such as when email attachments are sent from an enterprise aware version of a rights management mail client.

IMPORTANT

While Windows Information Protection can stop accidental data leaks from honest employees, it is not intended to stop malicious insiders from removing enterprise data. For more details about the benefits WIP provides, see [Why use WIP?](#) later in this topic.

Video: Protect enterprise data from being accidentally copied to the wrong place

Prerequisites

You'll need this software to run Windows Information Protection in your enterprise:

OPERATING SYSTEM

MANAGEMENT SOLUTION

OPERATING SYSTEM	MANAGEMENT SOLUTION
Windows 10, version 1607 or later	Microsoft Intune -OR- Microsoft Endpoint Configuration Manager -OR- Your current company-wide 3rd party mobile device management (MDM) solution. For info about 3rd party MDM solutions, see the documentation that came with your product. If your 3rd party MDM does not have UI support for the policies, refer to the EnterpriseDataProtection CSP documentation.

What is enterprise data control?

Effective collaboration means that you need to share data with others in your enterprise. This sharing can be from one extreme where everyone has access to everything without any security, all the way to the other extreme where people can't share anything and it's all highly secured. Most enterprises fall somewhere in between the two extremes, where success is balanced between providing the necessary access with the potential for improper data disclosure.

As an admin, you can address the question of who gets access to your data by using access controls, such as employee credentials. However, just because someone has the right to access your data doesn't guarantee that the data will remain within the secured locations of the enterprise. This means that while access controls are a great start, they're not enough.

In the end, all of these security measures have one thing in common: employees will tolerate only so much inconvenience before looking for ways around the security restrictions. For example, if you don't allow employees to share files through a protected system, employees will turn to an outside app that more than likely lacks security controls.

Using data loss prevention systems

To help address this security insufficiency, companies developed data loss prevention (also known as DLP) systems. Data loss prevention systems require:

- **A set of rules about how the system can identify and categorize the data that needs to be protected.** For example, a rule set might contain a rule that identifies credit card numbers and another rule that identifies Social Security numbers.
- **A way to scan company data to see whether it matches any of your defined rules.** Currently, Microsoft Exchange Server and Exchange Online provide this service for email in transit, while Microsoft SharePoint and SharePoint Online provide this service for content stored in document libraries.
- **The ability to specify what happens when data matches a rule, including whether employees can bypass enforcement.** For example, in Microsoft SharePoint and SharePoint Online, the Microsoft Purview data loss prevention system lets you warn your employees that shared data includes sensitive info, and to share it anyway (with an optional audit log entry).

Unfortunately, data loss prevention systems have their own problems. For example, the less detailed the rule set, the more false positives are created, leading employees to believe that the rules slow down their work and need to be bypassed in order to remain productive, potentially leading to data being incorrectly blocked or improperly released. Another major problem is that data loss prevention systems must be widely implemented to be effective. For example, if your company uses a data loss prevention system for email, but not for file shares

or document storage, you might find that your data leaks through the unprotected channels. But perhaps the biggest problem with data loss prevention systems is that it provides a jarring experience that interrupts the employees' natural workflow by stopping some operations (such as sending a message with an attachment that the system tags as sensitive) while allowing others, often according to subtle rules that the employee doesn't see and can't understand.

Using information rights management systems

To help address the potential data loss prevention system problems, companies developed information rights management (also known as IRM) systems. Information rights management systems embed protection directly into documents, so that when an employee creates a document, he or she determines what kind of protection to apply. For example, an employee can choose to stop the document from being forwarded, printed, shared outside of the organization, and so on.

After the type of protection is set, the creating app encrypts the document so that only authorized people can open it, and even then, only in compatible apps. After an employee opens the document, the app becomes responsible for enforcing the specified protections. Because protection travels with the document, if an authorized person sends it to an unauthorized person, the unauthorized person won't be able to read or change it. However, for this to work effectively information rights management systems require you to deploy and set up both a server and client environment. And, because only compatible clients can work with protected documents, an employees' work might be unexpectedly interrupted if he or she attempts to use a non-compatible app.

And what about when an employee leaves the company or unenrolls a device?

Finally, there's the risk of data leaking from your company when an employee leaves or unenrolls a device. Previously, you would simply erase all of the corporate data from the device, along with any other personal data on the device.

Benefits of WIP

Windows Information Protection provides:

- Obvious separation between personal and corporate data, without requiring employees to switch environments or apps.
- Additional data protection for existing line-of-business apps without a need to update the apps.
- Ability to wipe corporate data from Intune MDM enrolled devices while leaving personal data alone.
- Use of audit reports for tracking issues and remedial actions.
- Integration with your existing management system (Microsoft Intune, Microsoft Endpoint Configuration Manager, or your current mobile device management (MDM) system) to configure, deploy, and manage Windows Information Protection for your company.

Why use WIP?

Windows Information Protection is the mobile application management (MAM) mechanism on Windows 10. WIP gives you a new way to manage data policy enforcement for apps and documents on Windows 10 desktop operating systems, along with the ability to remove access to enterprise data from both enterprise and personal devices (after enrollment in an enterprise management solution, like Intune).

- **Change the way you think about data policy enforcement.** As an enterprise admin, you need to maintain compliance in your data policy and data access. Windows Information Protection helps protect enterprise on both corporate and employee-owned devices, even when the employee isn't using the device. When employees create content on an enterprise-protected device, they can choose to save it as a work document. If it's a work document, it becomes locally-maintained as enterprise data.

- **Manage your enterprise documents, apps, and encryption modes.**

- **Copying or downloading enterprise data.** When an employee or an app downloads content from a location like SharePoint, a network share, or an enterprise web location, while using a WIP-protected device, WIP encrypts the data on the device.
- **Using protected apps.** Managed apps (apps that you've included on the **Protected apps** list in your WIP policy) are allowed to access your enterprise data and will interact differently when used with unallowed, non-enterprise aware, or personal-only apps. For example, if WIP management is set to **Block**, your employees can copy and paste from one protected app to another protected app, but not to personal apps. Imagine an HR person wants to copy a job description from a protected app to the internal career website, an enterprise-protected location, but makes a mistake and tries to paste into a personal app instead. The paste action fails and a notification pops up, saying that the app couldn't paste because of a policy restriction. The HR person then correctly pastes to the career website without a problem.
- **Managed apps and restrictions.** With WIP you can control which apps can access and use your enterprise data. After adding an app to your protected apps list, the app is trusted with enterprise data. All apps not on this list are stopped from accessing your enterprise data, depending on your WIP management-mode.

You don't have to modify line-of-business apps that never touch personal data to list them as protected apps; just include them in the protected apps list.

- **Deciding your level of data access.** WIP lets you block, allow overrides, or audit employees' data sharing actions. Hiding overrides stops the action immediately. Allowing overrides lets the employee know there's a risk, but lets him or her continue to share the data while recording and auditing the action. Silent just logs the action without stopping anything that the employee could've overridden while using that setting; collecting info that can help you to see patterns of inappropriate sharing so you can take educative action or find apps that should be added to your protected apps list. For info about how to collect your audit log files, see [How to collect Windows Information Protection \(WIP\) audit event logs](#).
- **Data encryption at rest.** Windows Information Protection helps protect enterprise data on local files and on removable media.

Apps such as Microsoft Word work with WIP to help continue your data protection across local files and removable media. These apps are being referred to as, enterprise aware. For example, if an employee opens WIP-encrypted content from Word, edits the content, and then tries to save the edited version with a different name, Word automatically applies Windows Information Protection to the new document.

- **Helping prevent accidental data disclosure to public spaces.** Windows Information Protection helps protect your enterprise data from being accidentally shared to public spaces, such as public cloud storage. For example, if Dropbox™ isn't on your protected apps list, employees won't be able to sync encrypted files to their personal cloud storage. Instead, if the employee stores the content to an app on your protected apps list, like Microsoft OneDrive for Business, the encrypted files can sync freely to the business cloud, while maintaining the encryption locally.
- **Helping prevent accidental data disclosure to removable media.** Windows Information Protection helps prevent enterprise data from leaking when it's copied or transferred to removable media. For example, if an employee puts enterprise data on a Universal Serial Bus (USB) drive that also has personal data, the enterprise data remains encrypted while the personal data doesn't.

- **Remove access to enterprise data from enterprise-protected devices.** Windows Information Protection gives admins the ability to revoke enterprise data from one or many MDM-enrolled devices,

while leaving personal data alone. This is a benefit when an employee leaves your company, or in the case of a stolen device. After determining that the data access needs to be removed, you can use Microsoft Intune to unenroll the device so when it connects to the network, the user's encryption key for the device is revoked and the enterprise data becomes unreadable.

NOTE

For management of Surface devices it is recommended that you use the Current Branch of Microsoft Endpoint Configuration Manager.

Microsoft Endpoint Manager also allows you to revoke enterprise data. However, it does it by performing a factory reset of the device.

How WIP works

Windows Information Protection helps address your everyday challenges in the enterprise. Including:

- Helping to prevent enterprise data leaks, even on employee-owned devices that can't be locked down.
- Reducing employee frustrations because of restrictive data management policies on enterprise-owned devices.
- Helping to maintain the ownership and control of your enterprise data.
- Helping control the network and data access and data sharing for apps that aren't enterprise aware

Enterprise scenarios

Windows Information Protection currently addresses these enterprise scenarios:

- You can encrypt enterprise data on employee-owned and corporate-owned devices.
- You can remotely wipe enterprise data off managed computers, including employee-owned computers, without affecting the personal data.
- You can protect specific apps that can access enterprise data that are clearly recognizable to employees. You can also stop non-protected apps from accessing enterprise data.
- Your employees won't have their work otherwise interrupted while switching between personal and enterprise apps while the enterprise policies are in place. Switching environments or signing in multiple times isn't required.

WIP-protection modes

Enterprise data is automatically encrypted after it's loaded on a device from an enterprise source or if an employee marks the data as corporate. Then, when the enterprise data is written to disk, Windows Information Protection uses the Windows-provided Encrypting File System (EFS) to protect it and associate it with your enterprise identity.

Your Windows Information Protection policy includes a list of trusted apps that are protected to access and process corporate data. This list of apps is implemented through the [AppLocker](#) functionality, controlling what apps are allowed to run and letting the Windows operating system know that the apps can edit corporate data. Apps included on this list don't have to be modified to open corporate data because their presence on the list allows Windows to determine whether to grant them access. However, new for Windows 10, app developers can use a new set of application programming interfaces (APIs) to create *enlightened* apps that can use and edit both enterprise and personal data. A huge benefit to working with enlightened apps is that dual-use apps, like Microsoft Word, can be used with less concern about encrypting personal data by mistake because the APIs allow the app to determine whether data is owned by the enterprise or if it's personally owned.

NOTE

For info about how to collect your audit log files, see [How to collect Windows Information Protection \(WIP\) audit event logs](#).

You can set your Windows Information Protection policy to use 1 of 4 protection and management modes:

MODE	DESCRIPTION
Block	Windows Information Protection looks for inappropriate data sharing practices and stops the employee from completing the action. This can include sharing enterprise data to non-enterprise-protected apps in addition to sharing enterprise data between apps or attempting to share outside of your organization's network.
Allow overrides	Windows Information Protection looks for inappropriate data sharing, warning employees if they do something deemed potentially unsafe. However, this management mode lets the employee override the policy and share the data, logging the action to your audit log.
Silent	Windows Information Protection runs silently, logging inappropriate data sharing, without stopping anything that would've been prompted for employee interaction while in Allow overrides mode. Unallowed actions, like apps inappropriately trying to access a network resource or WIP-protected data, are still stopped.
Off	Windows Information Protection is turned off and doesn't help to protect or audit your data. After you turn off WIP, an attempt is made to decrypt any WIP-tagged files on the locally attached drives. Be aware that your previous decryption and policy info isn't automatically reapplied if you turn Windows Information Protection back on.

Turn off WIP

You can turn off all Windows Information Protection and restrictions, decrypting all devices managed by WIP and reverting to where you were pre-WIP, with no data loss. However, this isn't recommended. If you choose to turn WIP off, you can always turn it back on, but your decryption and policy info won't be automatically reapplied.

Next steps

After deciding to use WIP in your enterprise, you need to:

- [Create a Windows Information Protection \(WIP\) policy](#)

NOTE

Help to make this topic better by providing us with edits, additions, and feedback. For info about how to contribute to this topic, see [Editing Windows IT professional documentation](#).

Create a Windows Information Protection (WIP) policy using Microsoft Intune

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to:

- Windows 10, version 1607 and later

Microsoft Intune helps you create and deploy your enterprise data protection (WIP) policy, including letting you choose your protected apps, your WIP-protection level, and how to find enterprise data on the network.

In this section

TOPIC	DESCRIPTION
Create a Windows Information Protection (WIP) policy using the Azure portal for Microsoft Intune	Details about how to use the Azure portal for Microsoft Intune to create and deploy your WIP policy with MDM (Mobile Device Management), including letting you choose your protected apps, your WIP-protection level, and how to find enterprise data on the network.
Create and verify an Encrypting File System (EFS) Data Recovery Agent (DRA) certificate	Steps to create, verify, and perform a quick recovery using a Encrypting File System (EFS) Data Recovery Agent (DRA) certificate.
Determine the Enterprise Context of an app running in Windows Information Protection (WIP)	Use the Task Manager to determine whether an app is considered work, personal or exempt by Windows Information Protection (WIP).

Create a Windows Information Protection (WIP) policy using the Azure portal for Microsoft Intune

7/1/2022 • 20 minutes to read • [Edit Online](#)

Applies to:

- Windows 10, version 1607 and later

Microsoft Intune has an easy way to create and deploy a Windows Information Protection (WIP) policy. You can choose which apps to protect, the level of protection, and how to find enterprise data on the network. The devices can be fully managed by Mobile Device Management (MDM), or managed by Mobile Application Management (MAM), where Intune manages only the apps on a user's personal device.

Differences between MDM and MAM for WIP

You can create an app protection policy in Intune either with device enrollment for MDM or without device enrollment for MAM. The process to create either policy is similar, but there are important differences:

- MAM has more **Access** settings for Windows Hello for Business.
- MAM can [selectively wipe company data](#) from a user's personal device.
- MAM requires an [Azure Active Directory \(Azure AD\) Premium license](#).
- An Azure AD Premium license is also required for WIP auto-recovery, where a device can re-enroll and regain access to protected data. WIP auto-recovery depends on Azure AD registration to back up the encryption keys, which requires device auto-enrollment with MDM.
- MAM supports only one user per device.
- MAM can only manage [enlightened apps](#).
- Only MDM can use [BitLocker CSP](#) policies.
- If the same user and device are targeted for both MDM and MAM, the MDM policy will be applied to devices joined to Azure AD. For personal devices that are workplace-joined (that is, added by using **Settings > Email & accounts > Add a work or school account**), the MAM-only policy will be preferred but it's possible to upgrade the device management to MDM in **Settings**. Windows Home edition only supports WIP for MAM-only; upgrading to MDM policy on Home edition will revoke WIP-protected data access.

Prerequisites

Before you can create a WIP policy using Intune, you need to configure an MDM or MAM provider in Azure Active Directory (Azure AD). MAM requires an [Azure Active Directory \(Azure AD\) Premium license](#). An Azure AD Premium license is also required for WIP auto-recovery, where a device can re-enroll and regain access to protected data. WIP auto-recovery relies on Azure AD registration to back up the encryption keys, which requires device auto-enrollment with MDM.

Configure the MDM or MAM provider

1. Sign in to the Azure portal.
2. Select **Azure Active Directory > Mobility (MDM and MAM) > Microsoft Intune**.
3. Select **Restore Default URLs** or enter the settings for MDM or MAM user scope and select **Save**:

Home > Contoso - Mobility (MDM and MAM) > Configure

Configure

Microsoft Intune

Save Discard Delete

MDM user scope **i**

MDM terms of use URL **i**

MDM discovery URL **i**

MDM compliance URL **i**

[Restore default MDM URLs](#)

MAM User scope **i**

MAM Terms of use URL **i**

MAM Discovery URL **i**

MAM Compliance URL **i**

[Restore default MAM URLs](#)

Create a WIP policy

1. Sign in to the [Microsoft Endpoint Manager](#).
2. Open Microsoft Intune and select **Apps > App protection policies > Create policy**.

Home > Microsoft Intune > Client apps - App protection policies > Create policy

Microsoft Intune

Client apps - App protection policies

Microsoft Intune

Search (Ctrl+/) Create policy Refresh Columns Export

Filter by Policy Name...

POLICY	DEPLOYED
Android apps data leak prevention po...	Yes
iOS apps data leak prevention policy	Yes

3. In the **App policy** screen, select **Add a policy**, and then fill out the fields:

- **Name.** Type a name (required) for your new policy.
- **Description.** Type an optional description.
- **Platform.** Choose **Windows 10**.
- **Enrollment state.** Choose **Without enrollment** for MAM or **With enrollment** for MDM.

Home > Microsoft Intune > Mobile apps - App protection

Add a policy

* Name
Win10Policy ✓

Description
Windows Information Protection policy for Windows 10 computers ✓

Platform
Windows 10 ✓

Enrollment state
With enrollment ✓

Protected apps ⓘ >

Exempt apps ⓘ >

Required settings
Configure required settings >

Advanced settings
Configure advanced settings >

4. Select **Protected apps** and then select **Add apps**.

Home > Microsoft Intune > Mobile apps - App protection

Protected apps

Add apps Import apps

NAME	PRODUCT...	TYPE
No apps selected.		

You can add these types of apps:

- Recommended apps
- Store apps
- Desktop apps

NOTE

An application might return access denied errors after removing it from the list of protected apps. Rather than remove it from the list, uninstall and reinstall the application or exempt it from WIP policy.

Add recommended apps

Select **Recommended apps** and select each app you want to access your enterprise data or select them all, and select OK.

Add apps

Add recommended Microsoft apps, or manually add store or desktop apps to be allowed in this policy.

Recommended apps

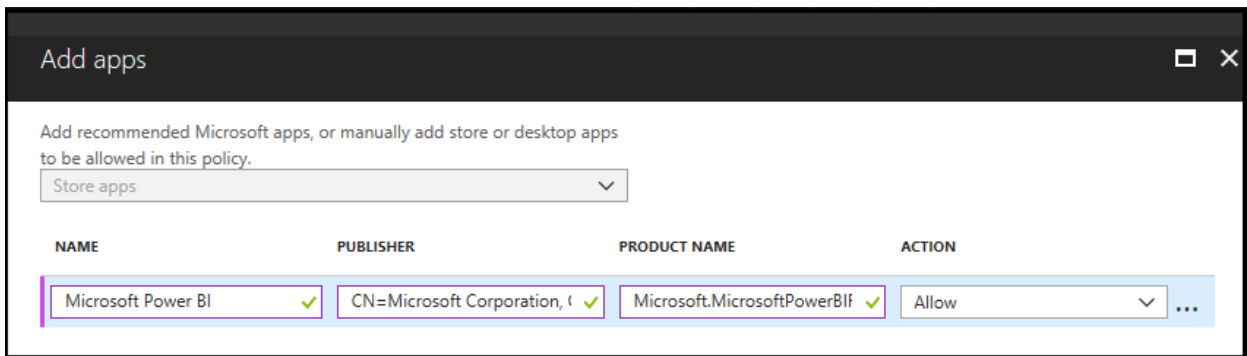
<input checked="" type="checkbox"/>	NAME	PRODUCT ...	TYPE	PUBLISHER	FILE	MIN VERSI...	MAX VERSI...	ACTION
<input checked="" type="checkbox"/>	Microsoft Edge	Microsoft.Micr...	Store	CN=Microsoft...				Allow
<input checked="" type="checkbox"/>	Microsoft Peo...	Microsoft.Peo...	Store	CN=Microsoft...				Allow
<input checked="" type="checkbox"/>	Word Mobile	Microsoft.Offi...	Store	CN=Microsoft...				Allow
<input checked="" type="checkbox"/>	Excel Mobile	Microsoft.Offi...	Store	CN=Microsoft...				Allow
<input checked="" type="checkbox"/>	PowerPoint M...	Microsoft.Offi...	Store	CN=Microsoft...				Allow
<input checked="" type="checkbox"/>	OneDrive App	Microsoft.Micr...	Store	CN=Microsoft...				Allow
<input checked="" type="checkbox"/>	OneNote	Microsoft.Offi...	Store	CN=Microsoft...				Allow
<input checked="" type="checkbox"/>	Mail and Cale...	microsoft.win...	Store	CN=Microsoft...				Allow
<input checked="" type="checkbox"/>	Microsoft Pho...	Microsoft.Win...	Store	CN=Microsoft...				Allow
<input checked="" type="checkbox"/>	Groove Music	Microsoft.Zun...	Store	CN=Microsoft...				Allow
<input checked="" type="checkbox"/>	Microsoft Mo...	Microsoft.Zun...	Store	CN=Microsoft...				Allow
<input checked="" type="checkbox"/>	Microsoft Mes...	Microsoft.Mes...	Store	CN=Microsoft...				Allow
<input checked="" type="checkbox"/>	Company Portal	Microsoft.Co...	Store	CN=Microsoft...				Allow
<input checked="" type="checkbox"/>	IE11	*	Desktop	O=Microsoft ...	iexplore.exe	*	*	Allow
<input checked="" type="checkbox"/>	Microsoft One...	*	Desktop	O=Microsoft ...	onedrive.exe	*	*	Allow
<input checked="" type="checkbox"/>	Notepad	*	Desktop	O=Microsoft ...	notepad.exe	*	*	Allow
<input checked="" type="checkbox"/>	Microsoft Paint	*	Desktop	O=Microsoft ...	mspaint.exe	*	*	Allow
<input checked="" type="checkbox"/>	Microsoft Re...	*	Desktop	O=Microsoft ...	mstsc.exe	*	*	Allow
<input checked="" type="checkbox"/>	Microsoft Tea...	*	Desktop	O=Microsoft ...	teams.exe	*	*	Allow
<input checked="" type="checkbox"/>	Microsoft Azu...	*	Desktop	O=Microsoft ...	msip.viewer.exe	*	*	Allow
<input checked="" type="checkbox"/>	Office-365-Pr...		AppLocker File					
<input checked="" type="checkbox"/>	Recommende...		AppLocker File					

OK

Add Store apps

Select Store apps, type the app product name and publisher, and select OK. For example, to add the Power BI Mobile App from the Store, type the following:

- **Name:** Microsoft Power BI
- **Publisher:** CN=Microsoft Corporation, O=Microsoft Corporation, L=Redmond, S=Washington, C=US
- **Product Name:** Microsoft.MicrosoftPowerBIForWindows



To add multiple Store apps, select the ellipsis .

If you don't know the Store app publisher or product name, you can find them by following these steps.

1. Go to the [Microsoft Store for Business](#) website, and find your app. For example, *Power BI Mobile App*.

2. Copy the ID value from the app URL. For example, the Power BI Mobile App ID URL is

`https://www.microsoft.com/store/p/microsoft-power-bi/9nb1gggz1xn1`, and you'd copy the ID value, `9nb1gggz1xn1`.

3. In a browser, run the Store for Business portal web API, to return a JavaScript Object Notation (JSON) file that includes the publisher and product name values. For example, run

`https://bspmts.mp.microsoft.com/v1/public/catalog/Retail/Products/9nb1gggz1xn1/applockerdata`, where `9nb1gggz1xn1` is replaced with your ID value.

The API runs and opens a text editor with the app details.

```
{
  "packageIdentityName": "Microsoft.MicrosoftPowerBIForWindows",
  "publisherCertificateName": "CN=Microsoft Corporation, O=Microsoft Corporation, L=Redmond, S=Washington, C=US"
}
```

4. Copy the `publisherCertificateName` value into the **Publisher** box and copy the `packageIdentityName` value into the **Name** box of Intune.

IMPORTANT

The JSON file might also return a `windowsPhoneLegacyId` value for both the **Publisher Name** and **Product Name** boxes. This means that you have an app that's using a XAP package and that you must set the **Product Name** as `windowsPhoneLegacyId`, and set the **Publisher Name** as `CN=` followed by the `windowsPhoneLegacyId`.

For example:

```
{
  "windowsPhoneLegacyId": "ca05b3ab-f157-450c-8c49-a1f127f5e71d",
}
```

Add Desktop apps

To add Desktop apps, complete the following fields, based on what results you want returned.

FIELD	MANAGES
-------	---------

FIELD	MANAGES
All fields marked as "*"	All files signed by any publisher. (Not recommended and may not work)
Publisher only	If you only fill out this field, you'll get all files signed by the named publisher. This might be useful if your company is the publisher and signer of internal line-of-business apps.
Publisher and Name only	If you only fill out these fields, you'll get all files for the specified product, signed by the named publisher.
Publisher, Name, and File only	If you only fill out these fields, you'll get any version of the named file or package for the specified product, signed by the named publisher.
Publisher, Name, File, and Min version only	If you only fill out these fields, you'll get the specified version or newer releases of the named file or package for the specified product, signed by the named publisher. This option is recommended for enlightened apps that weren't previously enlightened.
Publisher, Name, File, and Max version only	If you only fill out these fields, you'll get the specified version or older releases of the named file or package for the specified product, signed by the named publisher.
All fields completed	If you fill out all fields, you'll get the specified version of the named file or package for the specified product, signed by the named publisher.

To add another Desktop app, select the ellipsis After you've entered the info into the fields, select OK.

The screenshot shows the 'Add apps' dialog box. At the top, it says 'Add recommended Microsoft apps, or manually add store or desktop apps to be allowed in this policy.' Below this is a dropdown menu set to 'Desktop apps'. A table with the following columns is displayed: NAME, PUBLISHER, PRODUCT NAME, FILE, MIN VERSION, and MAX VERSION. The first row contains: 'Microsoft Word' (with a green checkmark), 'O=MICROSOFT' (with a green checkmark), 'Microsoft Word' (with a green checkmark), 'WORDPAD.EXE' (with a green checkmark), an empty text box, and another empty text box followed by an ellipsis button. Below the table, there is a checkbox labeled 'Pin to dashboard' which is unchecked, and a blue 'OK' button.

If you're unsure about what to include for the publisher, you can run this PowerShell command:

```
Get-AppLockerFileInformation -Path "<path_of_the_exe>"
```

Where "<path_of_the_exe>" goes to the location of the app on the device. For example:

```
Get-AppLockerFileInformation -Path "C:\Program Files\Windows NT\Accessories\wordpad.exe"
```

In this example, you'd get the following info:

Path	Publisher
-----	-----
%PROGRAMFILES%\WINDOWS NT\ACCESSORIES\WORDPAD.EXE	O=MICROSOFT CORPORATION, L=REDMOND, S=WASHINGTON, C=US

Where `O=MICROSOFT CORPORATION, L=REDMOND, S=WASHINGTON, C=US` is the **Publisher** name and `WORDPAD.EXE` is the **File** name.

Regarding to how to get the Product Name for the Apps you wish to Add, contact the Windows Support Team to request the guidelines

Import a list of apps

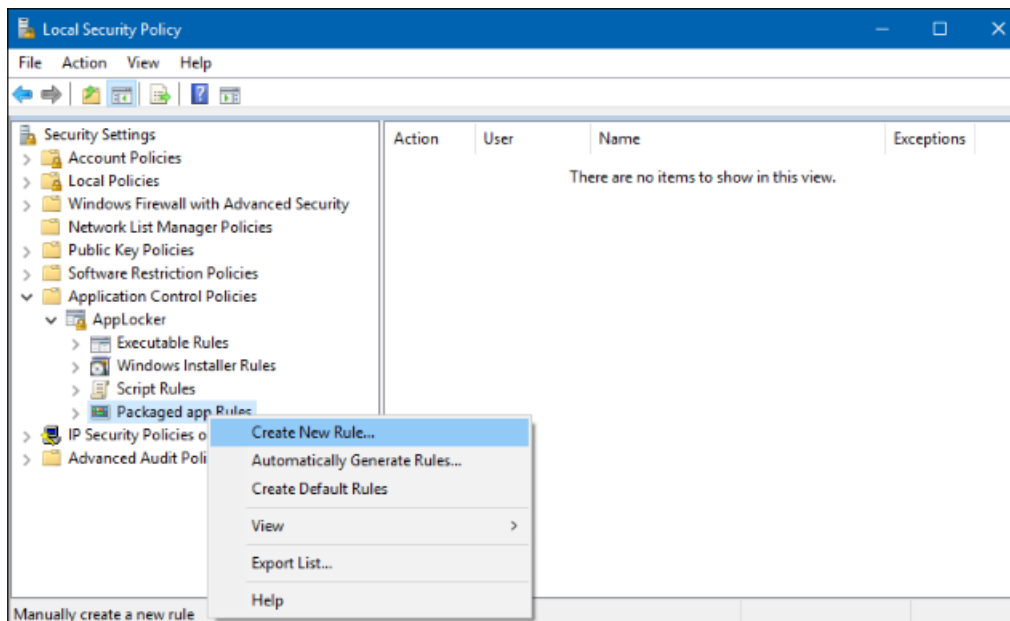
This section covers two examples of using an AppLocker XML file to the **Protected apps** list. You'll use this option if you want to add multiple apps at the same time.

- [Create a Packaged App rule for Store apps](#)
- [Create an Executable rule for unsigned apps](#)

For more info about AppLocker, see the [AppLocker](#) content.

Create a Packaged App rule for Store apps

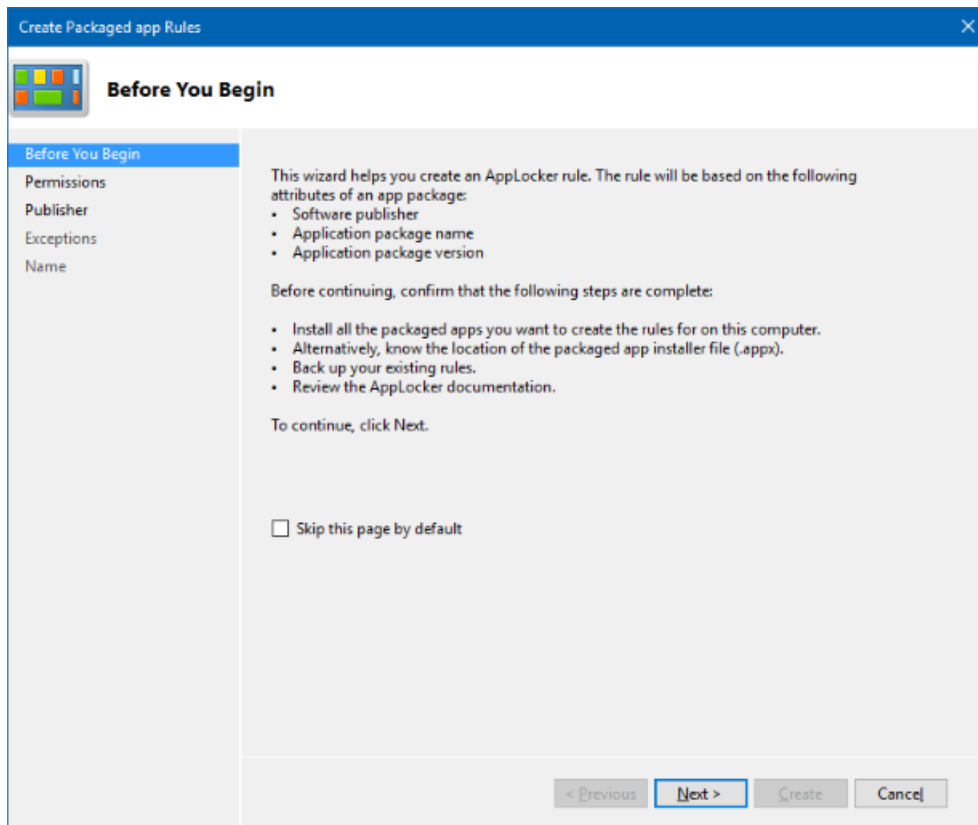
1. Open the Local Security Policy snap-in (SecPol.msc).
2. Expand **Application Control Policies**, expand **AppLocker**, and then select **Packaged App Rules**.



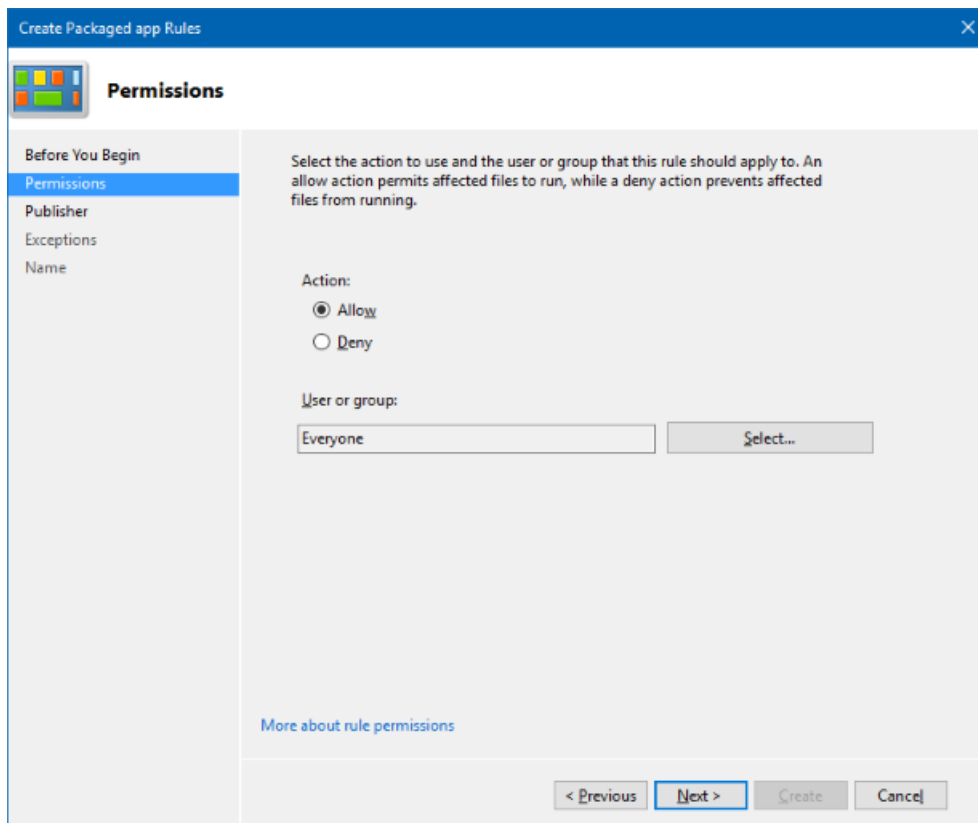
3. Right-click in the right side, and then select **Create New Rule**.

The **Create Packaged app Rules** wizard appears.

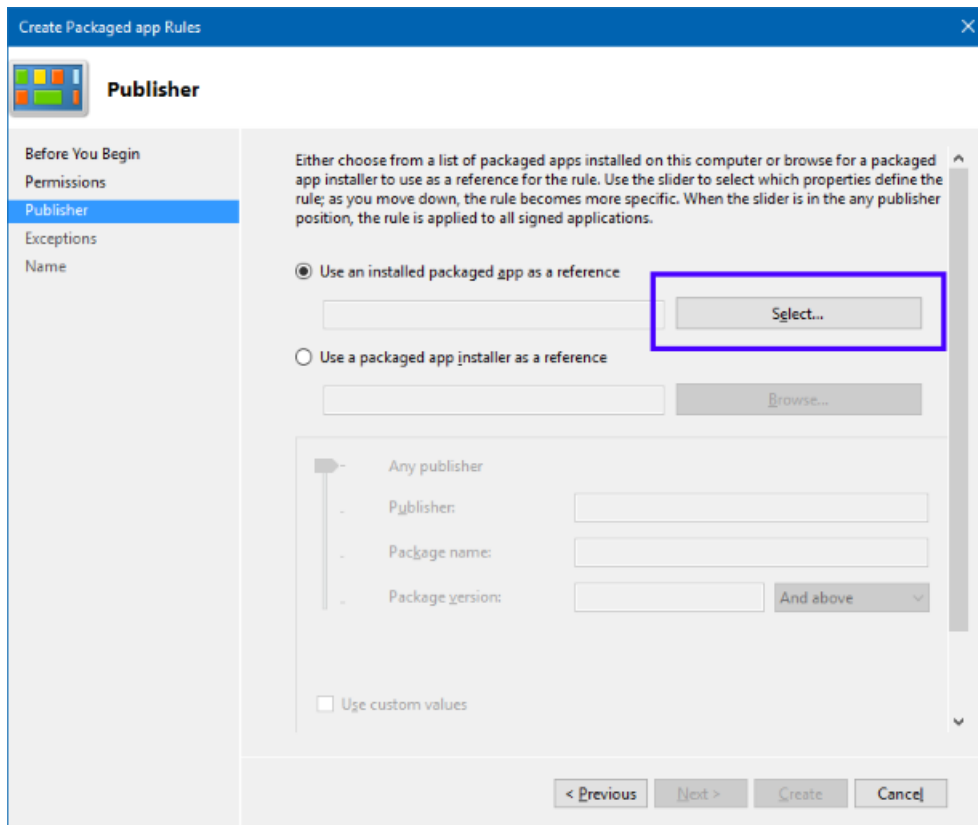
4. On the **Before You Begin** page, select **Next**.



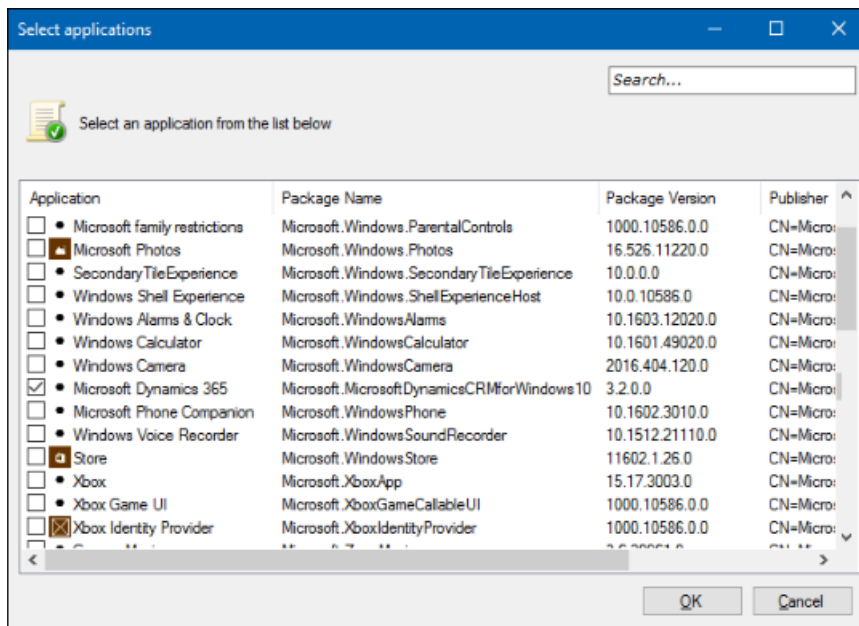
5. On the **Permissions** page, make sure the **Action** is set to **Allow** and the **User or group** is set to **Everyone**, and then select **Next**.



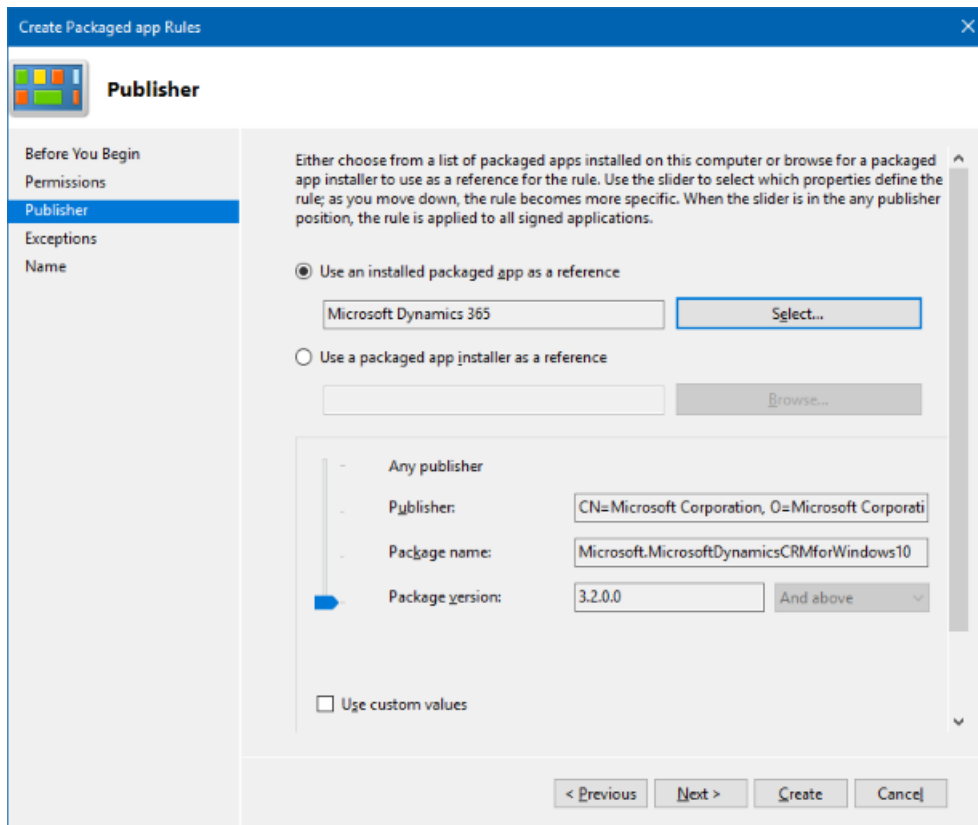
6. On the **Publisher** page, choose **Select** from the **Use an installed packaged app as a reference area**.



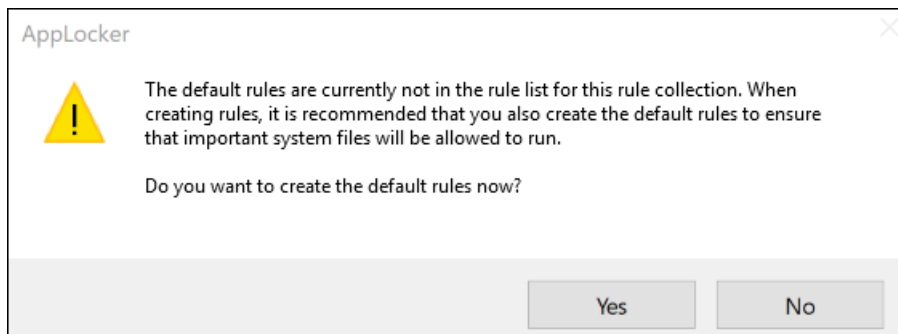
7. In the **Select applications** box, pick the app that you want to use as the reference for your rule, and then select **OK**. For this example, we're using Microsoft Dynamics 365.



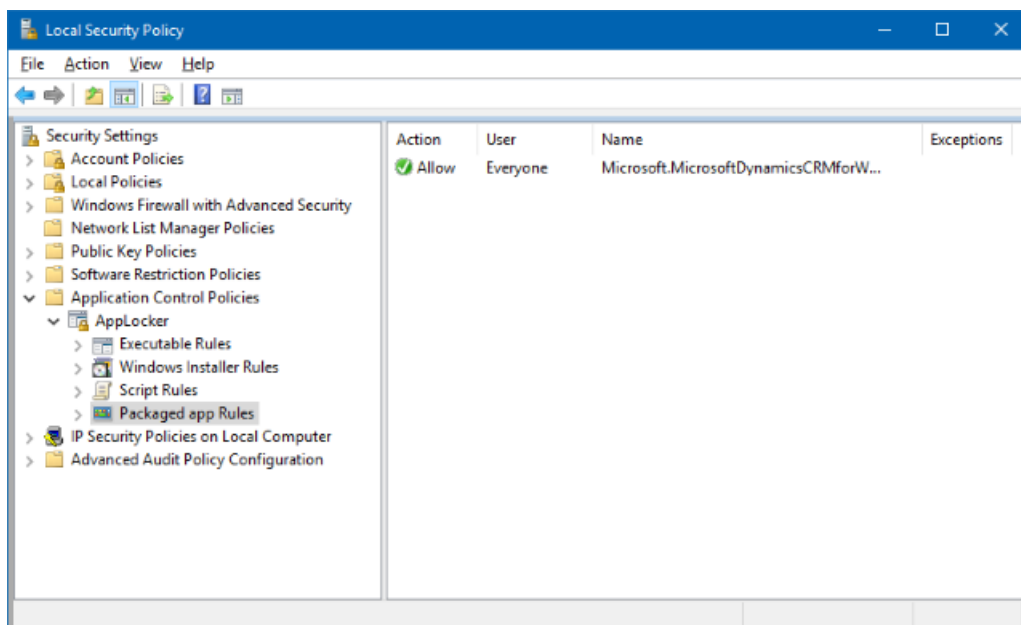
8. On the updated **Publisher** page, select **Create**.



9. Select **No** in the dialog box that appears, asking if you want to create the default rules. Don't create default rules for your WIP policy.

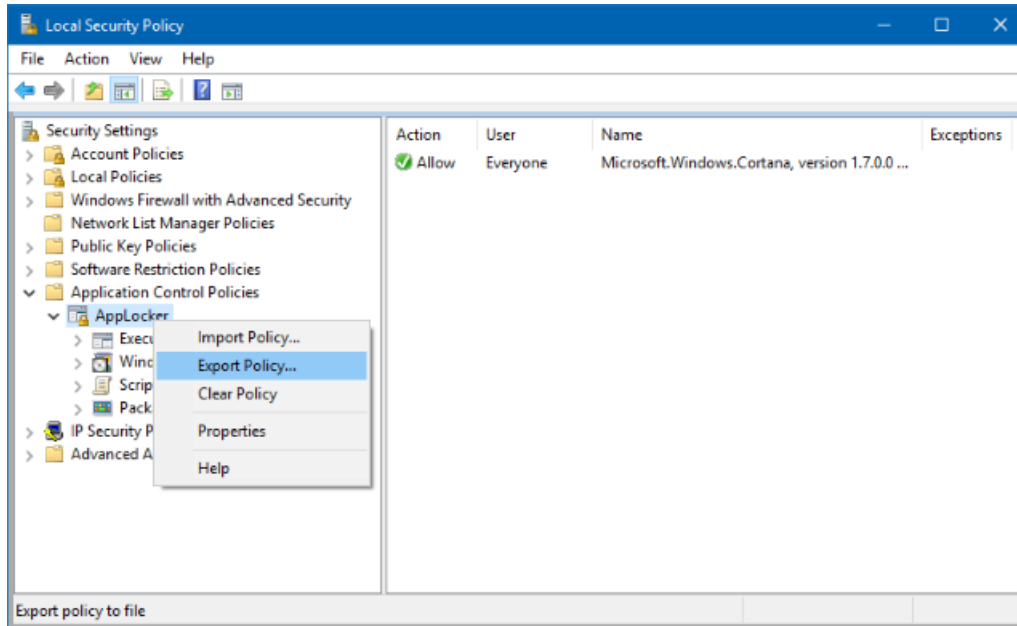


10. Review the Local Security Policy snap-in to make sure your rule is correct.



11. On the left, right-click on **AppLocker**, and then select **Export policy**.

The **Export policy** box opens, letting you export and save your new policy as XML.



12. In the **Export policy** box, browse to where the policy should be stored, give the policy a name, and then select **Save**.

The policy is saved and you'll see a message that says one rule was exported from the policy.

Example XML file

This is the XML file that AppLocker creates for Microsoft Dynamics 365.

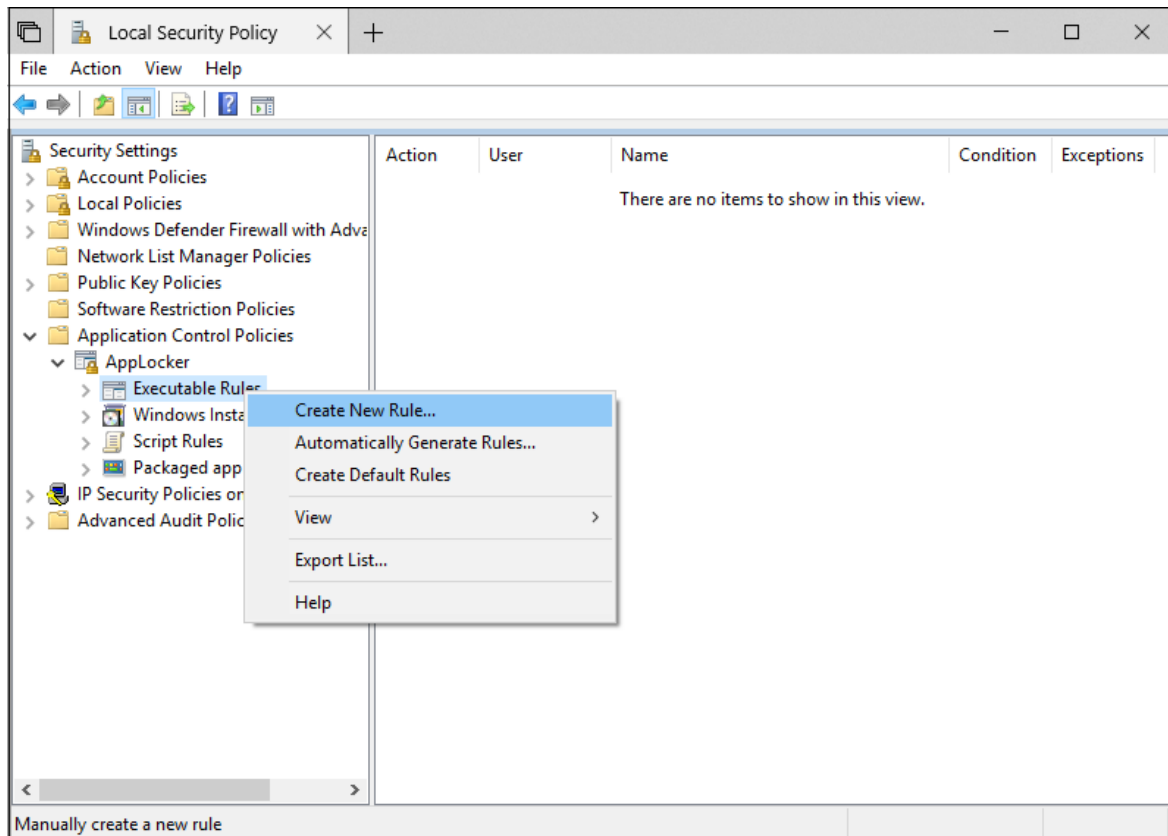
```
<?xml version="1.0"?>
<AppLockerPolicy Version="1">
  <RuleCollection EnforcementMode="NotConfigured" Type="Appx">
    <FilePublisherRule Action="Allow" UserOrGroupSid="S-1-1-0" Description=""
Name="Microsoft.MicrosoftDynamicsCRMforWindows10, version 3.2.0.0 and above, from Microsoft
Corporation" Id="3da34ed9-aec6-4239-88ba-0afdce252ab4">
      <Conditions>
        <FilePublisherCondition BinaryName="*"
ProductName="Microsoft.MicrosoftDynamicsCRMforWindows10" PublisherName="CN=Microsoft Corporation,
O=Microsoft Corporation, L=Redmond, S=Washington, C=US">
          <BinaryVersionRange HighSection="*" LowSection="3.2.0.0"/>
        </FilePublisherCondition>
      </Conditions>
    </FilePublisherRule>
  </RuleCollection>
  <RuleCollection EnforcementMode="NotConfigured" Type="Dll"/>
  <RuleCollection EnforcementMode="NotConfigured" Type="Exe"/>
  <RuleCollection EnforcementMode="NotConfigured" Type="Msi"/>
  <RuleCollection EnforcementMode="NotConfigured" Type="Script"/>
</AppLockerPolicy>
```

13. After you've created your XML file, you need to import it by using Microsoft Intune.

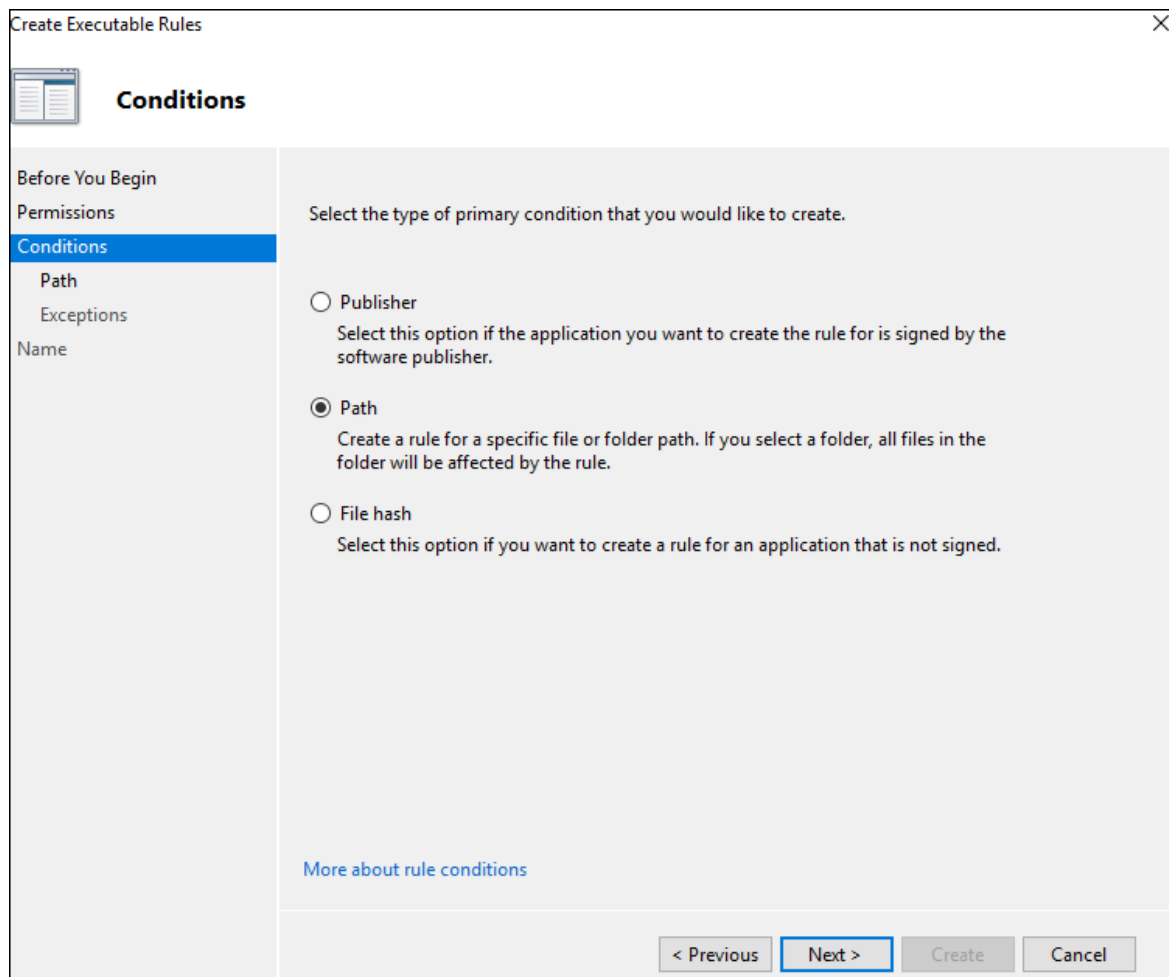
Create an Executable rule for unsigned apps

The executable rule helps to create an AppLocker rule to sign any unsigned apps. It enables adding the file path or the app publisher contained in the file's digital signature needed for the WIP policy to be applied.

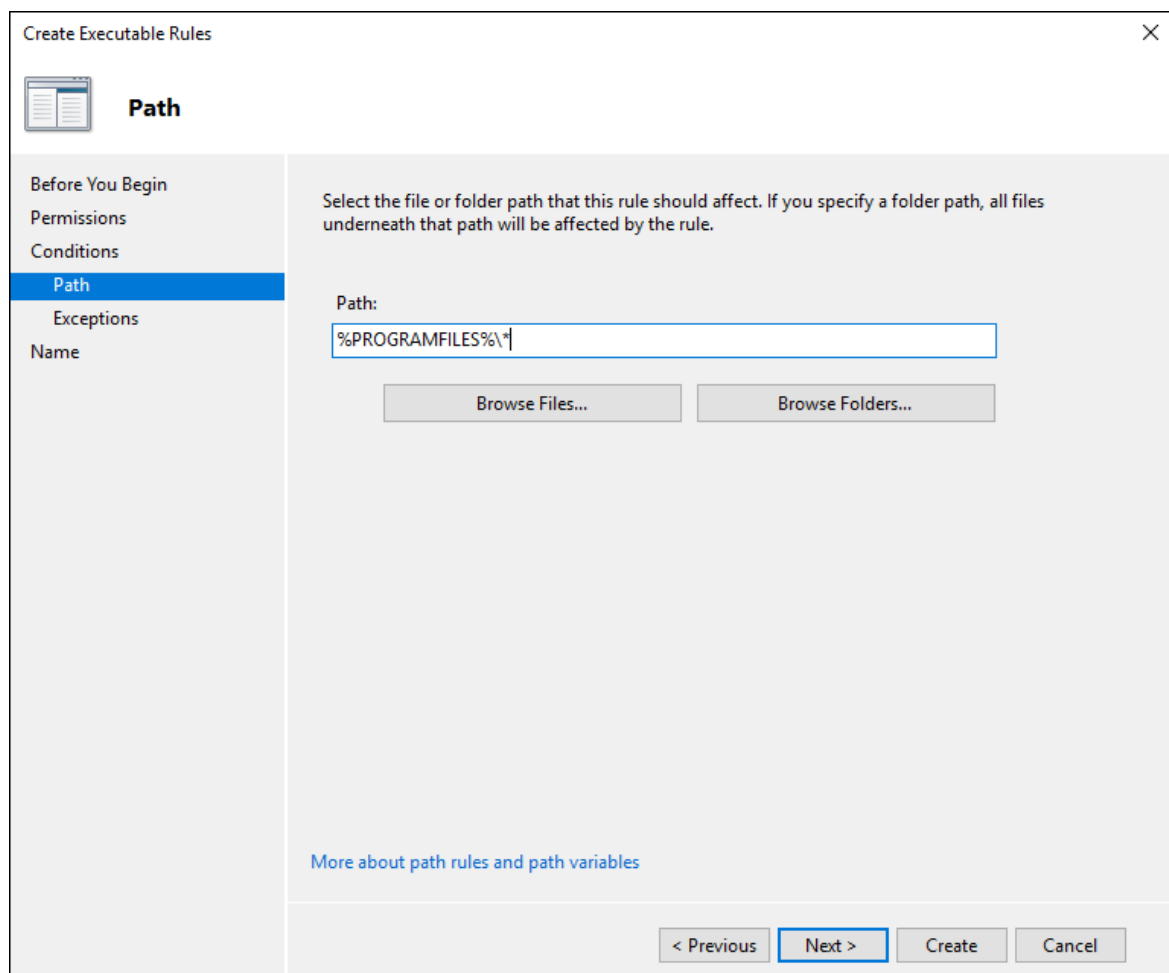
1. Open the Local Security Policy snap-in (SecPol.msc).
2. In the left pane, select **Application Control Policies > AppLocker > Executable Rules**.
3. Right-click **Executable Rules > Create New Rule**.



4. On the **Before You Begin** page, select **Next**.
5. On the **Permissions** page, make sure the **Action** is set to **Allow** and the **User or group** is set to **Everyone**, and then select **Next**.
6. On the **Conditions** page, select **Path** and then select **Next**.



7. Select **Browse Folders...** and select the path for the unsigned apps. For this example, we're using "C:\Program Files".



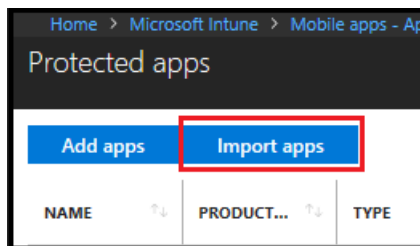
8. On the **Exceptions** page, add any exceptions and then select **Next**.
9. On the **Name** page, type a name and description for the rule and then select **Create**.
10. In the left pane, right-click **AppLocker** > **Export policy**.
11. In the **Export policy** box, browse to where the policy should be stored, give the policy a name, and then select **Save**.

The policy is saved and you'll see a message that says one rule was exported from the policy.

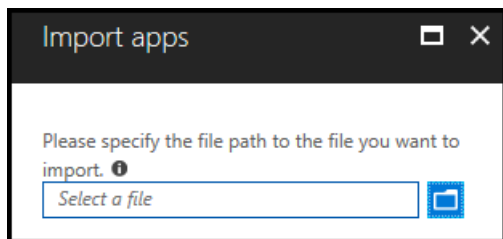
12. After you've created your XML file, you need to import it by using Microsoft Intune.

To import a list of protected apps using Microsoft Intune

1. In **Protected apps**, select **Import apps**.



Then import your file.



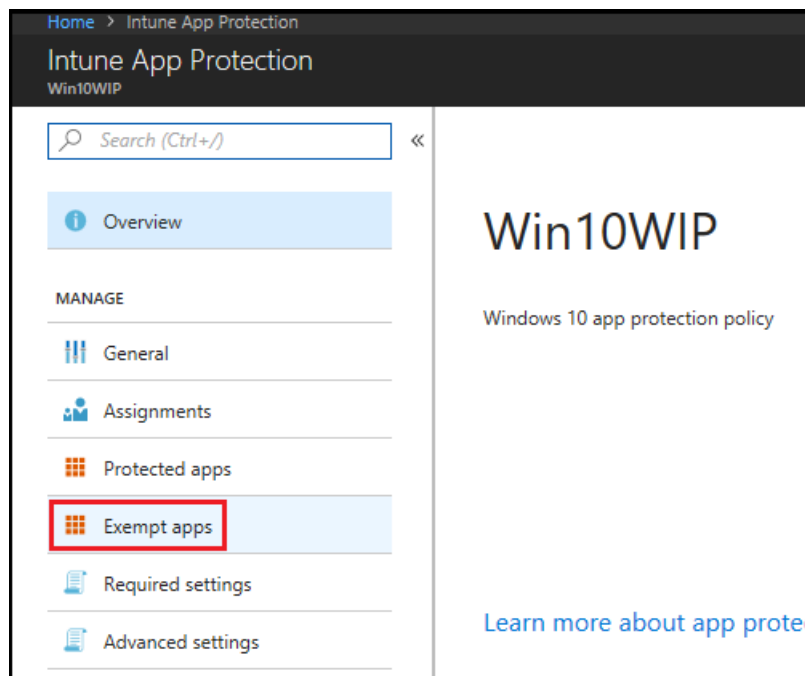
2. Browse to your exported AppLocker policy file, and then select **Open**.

The file imports and the apps are added to your **Protected apps** list.

Exempt apps from a WIP policy

If your app is incompatible with WIP, but still needs to be used with enterprise data, then you can exempt the app from the WIP restrictions. This means that your apps won't include auto-encryption or tagging and won't honor your network restrictions. It also means that your exempted apps might leak.

1. In **Client apps - App protection policies**, select **Exempt apps**.



2. In **Exempt apps**, select **Add apps**.

When you exempt apps, they're allowed to bypass the WIP restrictions and access your corporate data.

3. Fill out the rest of the app info, based on the type of app you're adding:

- [Add Recommended apps](#)
- [Add Store apps](#)
- [Add Desktop apps](#)
- [Import apps](#)

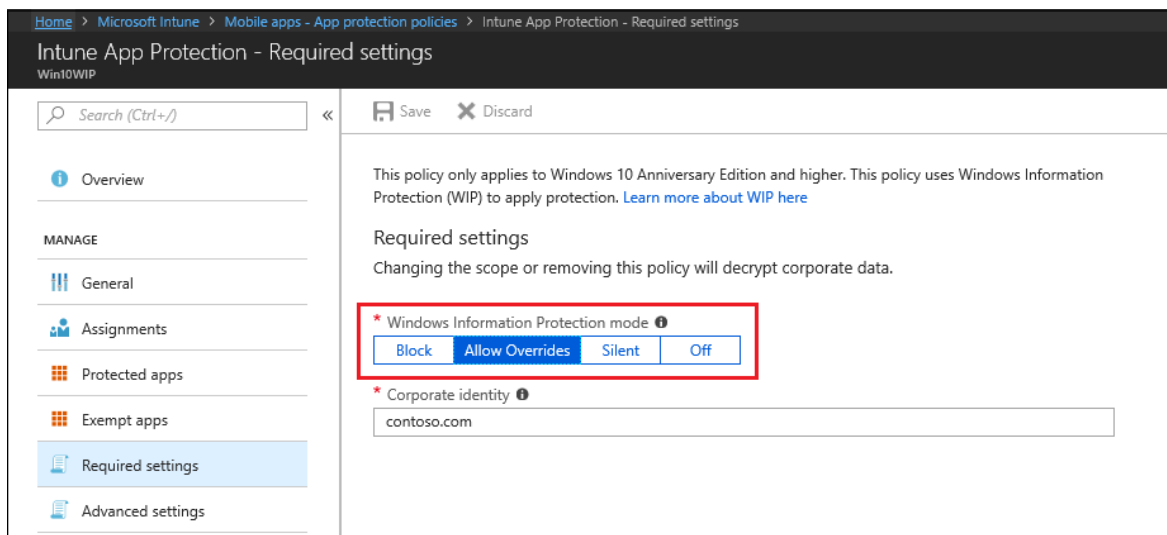
4. Select **OK**.

Manage the WIP protection mode for your enterprise data

After you've added the apps you want to protect with WIP, you'll need to apply a management and protection mode.

We recommend that you start with **Silent** or **Allow Overrides** while verifying with a small group that you have the right apps on your protected apps list. After you're done, you can change to your final enforcement policy, **Block**.

1. From **App protection policy**, select the name of your policy, and then select **Required settings**.



MODE	DESCRIPTION
Block	WIP looks for inappropriate data sharing practices and stops the employee from completing the action. This can include sharing info across non-enterprise-protected apps in addition to sharing enterprise data between other people and devices outside of your enterprise.
Allow Overrides	WIP looks for inappropriate data sharing, warning employees if they do something deemed potentially unsafe. However, this management mode lets the employee override the policy and share the data, logging the action to your audit log. For info about how to collect your audit log files, see How to collect Windows Information Protection (WIP) audit event logs .
Silent	WIP runs silently, logging inappropriate data sharing, without blocking anything that would have been prompted for employee interaction while in Allow Override mode. Unallowed actions, like apps inappropriately trying to access a network resource or WIP-protected data, are still stopped.
Off (not recommended)	WIP is turned off and doesn't help to protect or audit your data. After you turn off WIP, an attempt is made to decrypt any WIP-tagged files on the locally attached drives. Your previous decryption and policy info isn't automatically reapplied if you turn WIP protection back on.

2. Select **Save**.

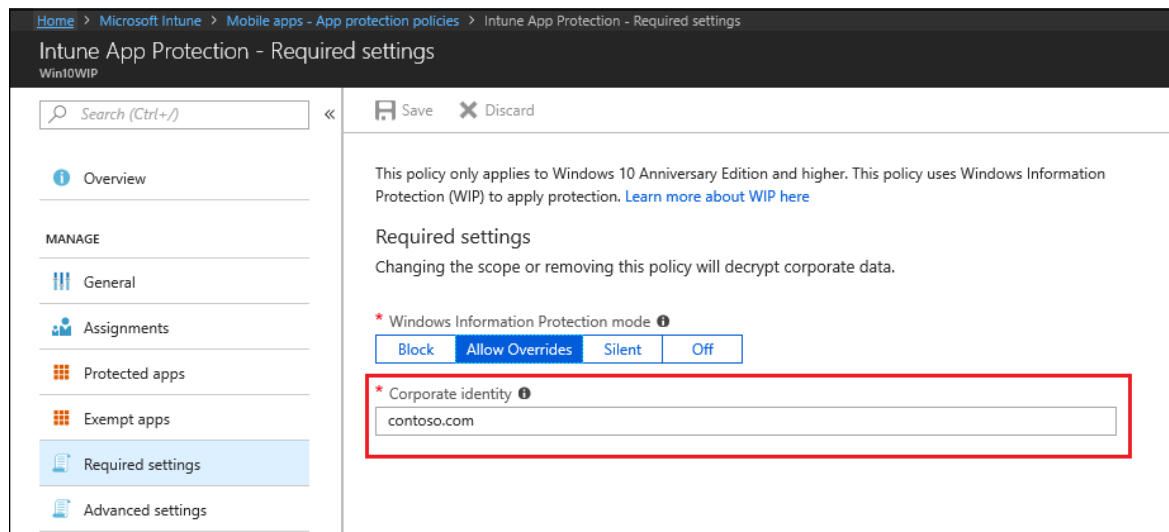
Define your enterprise-managed corporate identity

Corporate identity, typically expressed as your primary Internet domain (for example, contoso.com), helps to identify and tag your corporate data from apps you've marked as protected by WIP. For example, emails using contoso.com are identified as being corporate and are restricted by your Windows Information Protection policies.

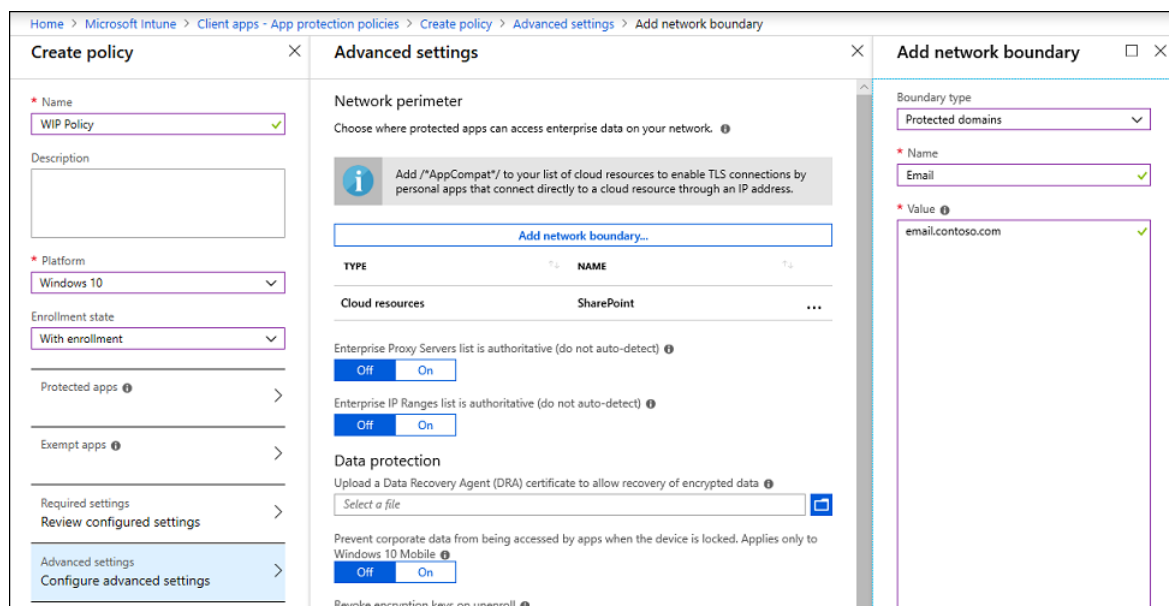
Starting with Windows 10, version 1703, Intune automatically determines your corporate identity and adds it to the **Corporate identity** field.

To change your corporate identity

1. From **App policy**, select the name of your policy, and then select **Required settings**.
2. If the auto-defined identity isn't correct, you can change the info in the **Corporate identity** field.



3. To add domains, such your email domain names, select **Configure Advanced settings > Add network boundary** and select **Protected domains**.

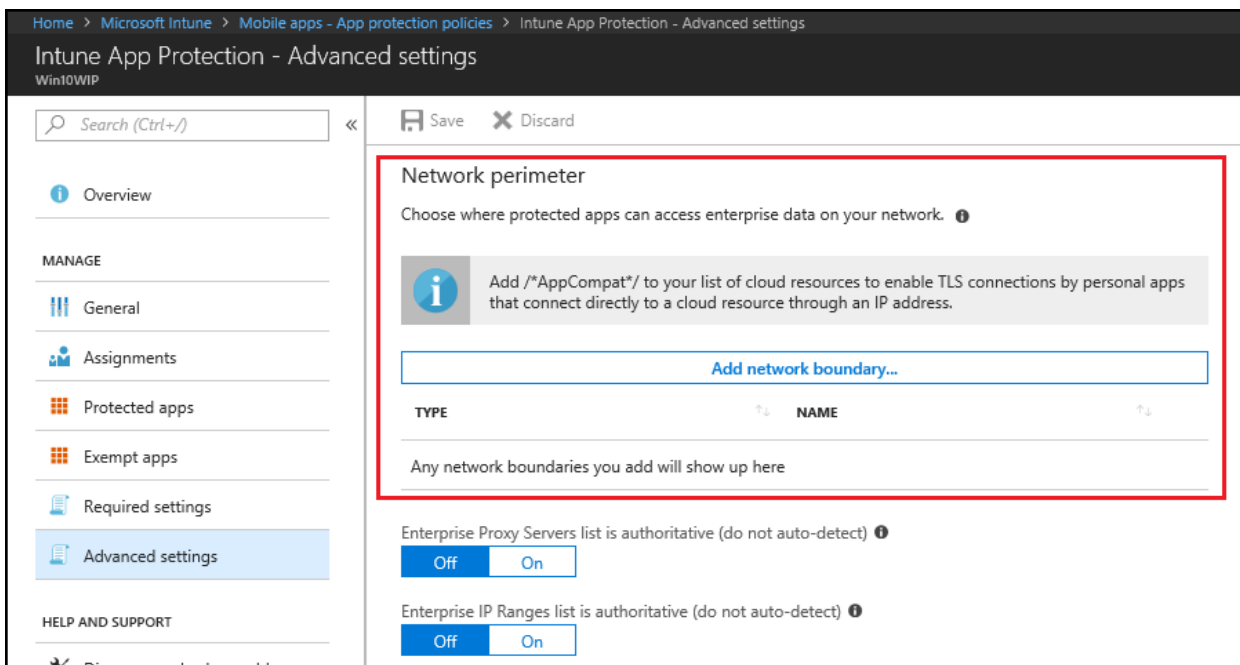


Choose where apps can access enterprise data

After you've added a protection mode to your apps, you'll need to decide where those apps can access enterprise data on your network. Every WIP policy should include your enterprise network locations.

There are no default locations included with WIP, you must add each of your network locations. This area applies to any network endpoint device that gets an IP address in your enterprise's range and is also bound to one of your enterprise domains, including SMB shares. Local file system locations should just maintain encryption (for example, on local NTFS, FAT, ExFAT).

To define the network boundaries, select **App policy > the name of your policy > Advanced settings > Add network boundary**.



Select the type of network boundary to add from the **Boundary type** box. Type a name for your boundary into the **Name** box, add your values to the **Value** box, based on the options covered in the following subsections, and then select **OK**.

Cloud resources

Specify the cloud resources to be treated as corporate and protected by WIP. For each cloud resource, you may also optionally specify a proxy server from your Internal proxy servers list to route traffic for this cloud resource. All traffic routed through your Internal proxy servers is considered enterprise.

Separate multiple resources with the "|" delimiter. For example:

```
URL <,proxy>|URL <,proxy>
```

Personal applications can access a cloud resource that has a blank space or an invalid character, such as a trailing dot in the URL.

To add a subdomain for a cloud resource, use a period (.) instead of an asterisk (*). For example, to add all subdomains within Office.com, use ".office.com" (without the quotation marks).

In some cases, such as when an app connects directly to a cloud resource through an IP address, Windows can't tell whether it's attempting to connect to an enterprise cloud resource or to a personal site. In this case, Windows blocks the connection by default. To stop Windows from automatically blocking these connections, you can add the `/*AppCompat*/` string to the setting. For example:

```
URL <,proxy>|URL <,proxy>/*AppCompat*/
```

When you use this string, we recommend that you also turn on [Azure Active Directory Conditional Access](#), using the **Domain joined or marked as compliant** option, which blocks apps from accessing any enterprise cloud resources that are protected by conditional access.

Value format with proxy:

```
contoso.sharepoint.com,contoso.internalproxy1.com|contoso.visualstudio.com,contoso.internalproxy2.com
```

Value format without proxy:

```
contoso.sharepoint.com|contoso.visualstudio.com|contoso.onedrive.com,
```

Protected domains

Specify the domains used for identities in your environment. All traffic to the fully qualified domains appearing in this list will be protected. Separate multiple domains with the "|" delimiter.

```
exchange.contoso.com|contoso.com|region.contoso.com
```

Network domains

Specify the DNS suffixes used in your environment. All traffic to the fully qualified domains appearing in this list will be protected. Separate multiple resources with the "," delimiter.

```
corp.contoso.com,region.contoso.com
```

Proxy servers

Specify the proxy servers your devices will go through to reach your cloud resources. Using this server type indicates that the cloud resources you're connecting to are enterprise resources.

This list shouldn't include any servers listed in your Internal proxy servers list. Proxy servers must be used only for non-WIP-protected (non-enterprise) traffic. Separate multiple resources with the ";" delimiter.

```
proxy.contoso.com:80;proxy2.contoso.com:443
```

Internal proxy servers

Specify the internal proxy servers your devices will go through to reach your cloud resources. Using this server type indicates that the cloud resources you're connecting to are enterprise resources.

This list shouldn't include any servers listed in your Proxy servers list. Internal proxy servers must be used only for WIP-protected (enterprise) traffic. Separate multiple resources with the ";" delimiter.

```
contoso.internalproxy1.com;contoso.internalproxy2.com
```

IPv4 ranges

Specify the addresses for a valid IPv4 value range within your intranet. These addresses, used with your Network domain names, define your corporate network boundaries. Classless Inter-Domain Routing (CIDR) notation isn't supported.

Separate multiple ranges with the ";" delimiter.

Starting IPv4 Address: 3.4.0.1

Ending IPv4 Address: 3.4.255.254

Custom URI: 3.4.0.1-3.4.255.254,
10.0.0.1-10.255.255.254

IPv6 ranges

Starting with Windows 10, version 1703, this field is optional.

Specify the addresses for a valid IPv6 value range within your intranet. These addresses, used with your network domain names, define your corporate network boundaries. Classless Inter-Domain Routing (CIDR) notation isn't supported.

Separate multiple ranges with the ";" delimiter.

Starting IPv6 Address: 2a01:110::

Ending IPv6 Address: 2a01:110:7fff:ffff:ffff:ffff:ffff:ffff

Custom URI: 2a01:110:7fff:ffff:ffff:ffff:ffff:ffff,
fd00::-fdff:ffff:ffff:ffff:ffff:ffff:ffff:ffff

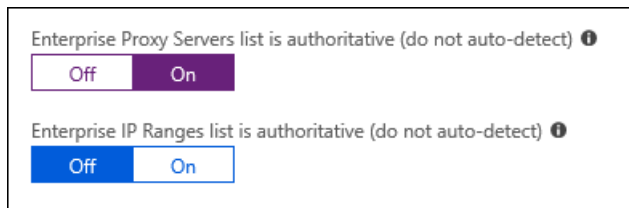
Neutral resources

Specify your authentication redirection endpoints for your company. These locations are considered enterprise or personal, based on the context of the connection before the redirection. Separate multiple resources with the ";" delimiter.

```
sts.contoso.com,sts.contoso2.com
```

Decide if you want Windows to look for more network settings:

- **Enterprise Proxy Servers list is authoritative (do not auto-detect).** Turn on if you want Windows to treat the proxy servers you specified in the network boundary definition as the complete list of proxy servers available on your network. If you turn this off, Windows will search for more proxy servers in your immediate network.
- **Enterprise IP Ranges list is authoritative (do not auto-detect).** Turn on if you want Windows to treat the IP ranges you specified in the network boundary definition as the complete list of IP ranges available on your network. If you turn this off, Windows will search for more IP ranges on any domain-joined devices connected to your network.



Upload your Data Recovery Agent (DRA) certificate

After you create and deploy your WIP policy to your employees, Windows begins to encrypt your corporate data on the employees' local device drive. If somehow the employees' local encryption keys get lost or revoked, the encrypted data can become unrecoverable. To help avoid this possibility, the Data Recovery Agent (DRA) certificate lets Windows use an included public key to encrypt the local data while you maintain the private key that can unencrypt the data.

IMPORTANT

Using a DRA certificate isn't mandatory. However, we strongly recommend it. For more info about how to find and export your data recovery certificate, see [Data Recovery and Encrypting File System \(EFS\)](#). For more info about creating and verifying your EFS DRA certificate, see [Create and verify an Encrypting File System \(EFS\) Data Recovery Agent \(DRA\) certificate](#).

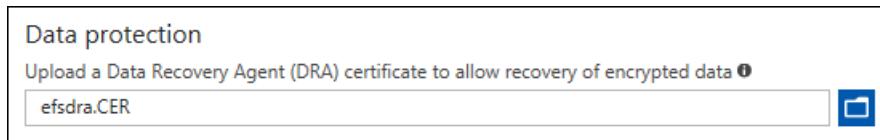
To upload your DRA certificate

1. From **App policy**, select the name of your policy, and then select **Advanced settings** from the menu that appears.

Advanced settings shows.

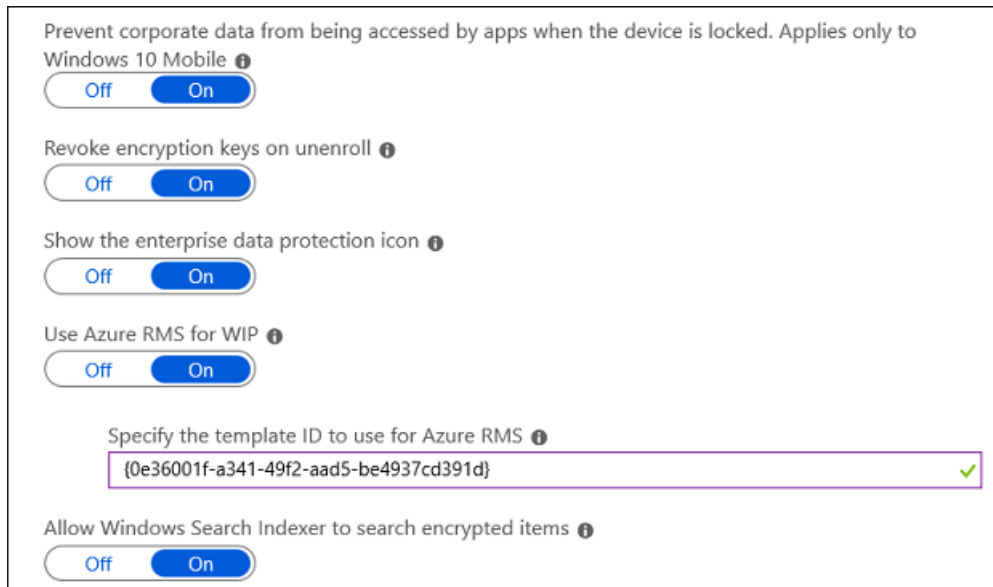
2. In the **Upload a Data Recovery Agent (DRA) certificate to allow recovery of encrypted data**

box, select **Browse** to add a data recovery certificate for your policy.



Choose your optional WIP-related settings

After you've decided where your protected apps can access enterprise data on your network, you can choose optional settings.



Revoke encryption keys on unenroll. Determines whether to revoke a user's local encryption keys from a device when it's unenrolled from Windows Information Protection. If the encryption keys are revoked, a user no longer has access to encrypted corporate data. The options are:

- **On, or not configured (recommended).** Revokes local encryption keys from a device during unenrollment.
- **Off.** Stop local encryption keys from being revoked from a device during unenrollment. For example, if you're migrating between Mobile Device Management (MDM) solutions.

Show the enterprise data protection icon. Determines whether the Windows Information Protection icon overlay appears on corporate files in the Save As and File Explorer views. The options are:

- **On.** Allows the Windows Information Protection icon overlay to appear on corporate files in the Save As and File Explorer views. Also, for unenlightened but protected apps, the icon overlay also appears on the app tile and with Managed text on the app name in the **Start** menu.
- **Off, or not configured (recommended).** Stops the Windows Information Protection icon overlay from appearing on corporate files or unenlightened, but protected apps. Not configured is the default option.

Use Azure RMS for WIP. Determines whether WIP uses [Microsoft Azure Rights Management](#) to apply EFS encryption to files that are copied from Windows 10 to USB or other removable drives so they can be securely shared with employees. In other words, WIP uses Azure Rights Management "machinery" to apply EFS encryption to files when they're copied to removable drives. You must already have Azure Rights Management set up. The EFS file encryption key is protected by the RMS template's license. Only users with permission to that template can read it from the removable drive. WIP can also integrate with Azure RMS by using the **AllowAzureRMSForEDP** and the **RMSTemplateIDForEDP** MDM settings in the [EnterpriseDataProtection CSP](#).

- **On.** Protects files that are copied to a removable drive. You can enter a TemplateID GUID to specify who can access the Azure Rights Management protected files, and for how long. The RMS template is only applied to the files on removable media, and is only used for access control—it doesn't actually apply Azure Information Protection to the files.

If you don't specify an [RMS template](#), it's a regular EFS file using a default RMS template that all users can access.

- **Off, or not configured.** Stops WIP from encrypting Azure Rights Management files that are copied to a removable drive.

NOTE

Regardless of this setting, all files in OneDrive for Business will be encrypted, including moved Known Folders.

Allow Windows Search Indexer to search encrypted files. Determines whether to allow the Windows Search Indexer to index items that are encrypted, such as WIP protected files.

- **On.** Starts Windows Search Indexer to index encrypted files.
- **Off, or not configured.** Stops Windows Search Indexer from indexing encrypted files.

Encrypted file extensions

You can restrict which files are protected by WIP when they're downloaded from an SMB share within your enterprise network locations. If this setting is configured, only files with the extensions in the list will be encrypted. If this setting is not specified, the existing auto-encryption behavior is applied.

Add encrypted file extensions		
NAME	ENCRYPTED FILE EXTENSIONS	
PowerPoint	pptx	...
Word	docx	...
Excel	xlsx	...
Text	txt	...

Related articles

- [How to collect Windows Information Protection \(WIP\) audit event logs](#)
- [General guidance and best practices for Windows Information Protection \(WIP\)](#)
- [What is Azure Rights Management?](#)
- [Create a Windows Information Protection \(WIP\) protection policy using Microsoft Intune](#)
- [Intune MAM Without Enrollment](#)
- [Azure RMS Documentation Update for May 2016](#)

NOTE

Help to make this topic better by providing us with edits, additions, and feedback. For info about how to contribute to this topic, see [Editing Windows IT professional documentation](#).

Deploy your Windows Information Protection (WIP) policy using the Azure portal for Microsoft Intune

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to:

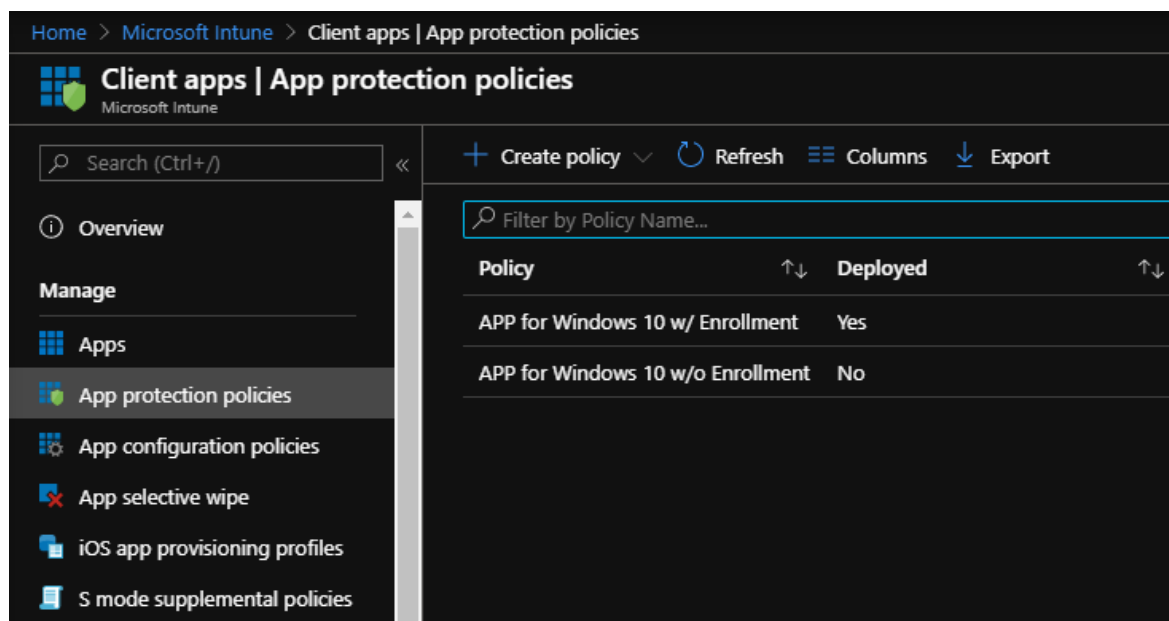
- Windows 10, version 1607 and later

After you've created your Windows Information Protection (WIP) policy, you'll need to deploy it to your organization's enrolled devices. Enrollment can be done for business or personal devices, allowing the devices to use your managed apps and to sync with your managed content and information.

To deploy your WIP policy

1. On the **App protection policies** pane, click your newly-created policy, click **Assignments**, and then select groups to include or exclude from the policy.
2. Choose the group you want your policy to apply to, and then click **Select** to deploy the policy.

The policy is deployed to the selected users' devices.



The screenshot shows the Microsoft Intune Azure portal interface for managing App protection policies. The breadcrumb navigation is 'Home > Microsoft Intune > Client apps | App protection policies'. The main heading is 'Client apps | App protection policies' with the Microsoft Intune logo. A search bar is present with the text 'Search (Ctrl+ /)'. On the left, there is a navigation pane with 'Overview' and a 'Manage' section containing 'Apps', 'App protection policies' (selected), 'App configuration policies', 'App selective wipe', 'iOS app provisioning profiles', and 'S mode supplemental policies'. The main content area has a toolbar with '+ Create policy', 'Refresh', 'Columns', and 'Export'. Below the toolbar is a search filter 'Filter by Policy Name...'. A table lists two policies:

Policy	Deployed
APP for Windows 10 w/ Enrollment	Yes
APP for Windows 10 w/o Enrollment	No

NOTE

Help to make this topic better by providing us with edits, additions, and feedback. For info about how to contribute to this topic, see [Editing Windows IT professional documentation](#).

Related topics

- [General guidance and best practices for Windows Information Protection \(WIP\)](#)

Associate and deploy a VPN policy for Windows Information Protection (WIP) using Endpoint Manager

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to:

- Windows 10, version 1607 and later

After you've created and deployed your Windows Information Protection (WIP) policy, you can use Microsoft Intune to associate and deploy your Virtual Private Network (VPN) policy, linking it to your WIP policy.

Associate your WIP policy to your VPN policy using Endpoint Manager

To associate your WIP policy with your organization's existing VPN policy, use the following steps:

1. Sign in to the [Microsoft Endpoint Manager admin center](#).
2. Select **Devices > Configuration profiles > Create profile**.
3. Enter the following properties:
 - **Platform:** Select **Windows 10 and later**
 - **Profile:** Select **Templates > Custom**.
4. Select **Create**.
5. In **Basics**, enter the following properties:
 - **Name:** Enter a descriptive name for the profile. Name your profiles so you can easily identify them later.
 - **Description:** Enter a description for the profile. This setting is optional, but recommended.
6. Select **Next**.
7. In **Configuration settings**, enter the following properties:
 - **Name:** Enter a name for your setting. For example, enter `EDPModeID`.
 - **OMA-URI:** Enter `./Vendor/MSFT/VPNv2/YourVPNProfileName/EDPModeId`.
 - **Data type:** Select `String`.
 - **Value:** Type your fully-qualified domain that should be used by the OMA-URI setting. For example, enter `corp.contoso.com`.

For more information on these settings, see [Use custom settings for Windows devices in Intune](#).
8. Select **Next**, and continue configuring the policy. For the specific steps and recommendations, see [Create a profile with custom settings in Intune](#).

Deploy your VPN policy using Microsoft Intune

After you've created your VPN policy, you'll need to deploy it to the same group you deployed your Windows Information Protection (WIP) policy.

1. On the **App policy** blade, click your newly-created policy, click **User groups** from the menu that appears, and then click **Add user group**.

A list of user groups, made up of all of the security groups in your Azure Active Directory, appear in the **Add user group** blade.

2. Choose the group you want your policy to apply to, and then click **Select** to deploy the policy.

The policy is deployed to the selected users' devices.

Home > Microsoft Intune > Client apps | App protection policies

Client apps | App protection policies
Microsoft Intune

Search (Ctrl+/) << + Create policy Refresh Columns Export

Filter by Policy Name...

Policy	Deployed
APP for Windows 10 w/ Enrollment	Yes
APP for Windows 10 w/o Enrollment	No

Overview
Manage
Apps
App protection policies
App configuration policies
App selective wipe
iOS app provisioning profiles
S mode supplemental policies

NOTE

Help to make this topic better by providing us with edits, additions, and feedback. For info about how to contribute to this topic, see [Editing Windows IT professional documentation](#).

Create and verify an Encrypting File System (EFS) Data Recovery Agent (DRA) certificate

7/1/2022 • 5 minutes to read • [Edit Online](#)

Applies to:

- Windows 10, version 1607 and later

If you don't already have an EFS DRA certificate, you'll need to create and extract one from your system before you can use Windows Information Protection (WIP), formerly known as enterprise data protection (EDP), in your organization. For the purposes of this section, we'll use the file name EFSDRA; however, this name can be replaced with anything that makes sense to you.

IMPORTANT

If you already have an EFS DRA certificate for your organization, you can skip creating a new one. Just use your current EFS DRA certificate in your policy. For more info about when to use a PKI and the general strategy you should use to deploy DRA certificates, see the [Security Watch Deploying EFS: Part 1](#) article on TechNet. For more general info about EFS protection, see [Protecting Data by Using EFS to Encrypt Hard Drives](#).

If your DRA certificate has expired, you won't be able to encrypt your files with it. To fix this, you'll need to create a new certificate, using the steps in this topic, and then deploy it through policy.

Manually create an EFS DRA certificate

1. On a computer without an EFS DRA certificate installed, open a command prompt with elevated rights, and then navigate to where you want to store the certificate.
2. Run this command:

```
cipher /r:EFSDRA
```

Where *EFSDRA* is the name of the `.cer` and `.pfx` files that you want to create.

3. When prompted, type and confirm a password to help protect your new Personal Information Exchange (.pfx) file.

The EFSDRA.cer and EFSDRA.pfx files are created in the location you specified in Step 1.

IMPORTANT

Because the private keys in your DRA .pfx files can be used to decrypt any WIP file, you must protect them accordingly. We highly recommend storing these files offline, keeping copies on a smart card with strong protection for normal use and master copies in a secured physical location.

4. Add your EFS DRA certificate to your WIP policy using a deployment tool, such as [Microsoft Intune](#) or [Microsoft Endpoint Configuration Manager](#).

NOTE

This certificate can be used in Intune for policies both *with* device enrollment (MDM) and *without* device enrollment (MAM).

Verify your data recovery certificate is correctly set up on a WIP client computer

1. Find or create a file that's encrypted using Windows Information Protection. For example, you could open an app on your allowed app list, and then create and save a file so it's encrypted by WIP.
2. Open an app on your protected app list, and then create and save a file so that it's encrypted by WIP.
3. Open a command prompt with elevated rights, navigate to where you stored the file you just created, and then run this command:

```
cipher /c filename
```

Where *filename* is the name of the file you created in Step 1.

4. Make sure that your data recovery certificate is listed in the **Recovery Certificates** list.

Recover your data using the EFS DRA certificate in a test environment

1. Copy your WIP-encrypted file to a location where you have admin access.
2. Install the EFS DRA.pfx file, using its password.
3. Open a command prompt with elevated rights, navigate to the encrypted file, and then run this command:

```
cipher /d encryptedfile.extension
```

Where *encryptedfile.extension* is the name of your encrypted file. For example, `corporatedata.docx`.

Recover WIP-protected after unenrollment

It's possible that you might revoke data from an unenrolled device only to later want to restore it all. This can happen in the case of a missing device being returned or if an unenrolled employee enrolls again. If the employee enrolls again using the original user profile, and the revoked key store is still on the device, all of the revoked data can be restored at once.

IMPORTANT

To maintain control over your enterprise data, and to be able to revoke again in the future, you must only perform this process after the employee has re-enrolled the device.

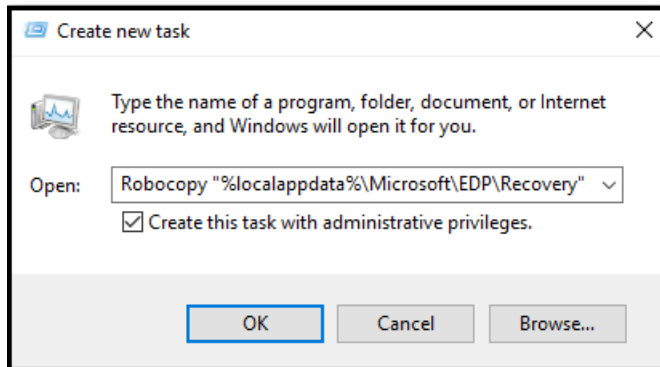
1. Have the employee sign in to the unenrolled device, open an elevated command prompt, and type:

```
Robocopy "%localappdata%\Microsoft\EDP\Recovery" "new_location" * /EFSRAW
```

Where *"new_location"* is in a different directory. This can be on the employee's device or on a shared

folder on a computer that runs Windows 8 or Windows Server 2012 or newer and can be accessed while you're logged in as a data recovery agent.

To start Robocopy in S mode, open Task Manager. Click **File > Run new task**, type the command, and click **Create this task with administrative privileges**.



If the employee performed a clean installation and there is no user profile, you need to recover the keys from the System Volume folder in each drive. Type:

```
Robocopy "drive_letter:\System Volume Information\EDP\Recovery\" "new_location" * /EFSRAW
```

2. Sign in to a different device with administrator credentials that have access to your organization's DRA certificate, and perform the file decryption and recovery by typing:

```
cipher.exe /D "new_location"
```

3. Have your employee sign in to the unenrolled device, and type:

```
Robocopy "new_location" "%localappdata%\Microsoft\EDP\Recovery\Input"
```

4. Ask the employee to lock and unlock the device.

The Windows Credential service automatically recovers the employee's previously revoked keys from the `Recovery\Input` location.

Auto-recovery of encryption keys

Starting with Windows 10, version 1709, WIP includes a data recovery feature that lets your employees auto-recover access to work files if the encryption key is lost and the files are no longer accessible. This typically happens if an employee reimages the operating system partition, removing the WIP key info, or if a device is reported as lost and you mistakenly target the wrong device for unenrollment.

To help make sure employees can always access files, WIP creates an auto-recovery key that's backed up to their Azure Active Directory (Azure AD) identity.

The employee experience is based on sign in with an Azure AD work account. The employee can either:

- Add a work account through the **Windows Settings > Accounts > Access work or school > Connect** menu.

-OR-

- Open **Windows Settings > Accounts > Access work or school > Connect** and choose the **Join this device to Azure Active Directory** link, under **Alternate actions**.

NOTE

To perform an Azure AD Domain Join from the Settings page, the employee must have administrator privileges to the device.

After signing in, the necessary WIP key info is automatically downloaded and employees are able to access the files again.

To test what the employee sees during the WIP key recovery process

1. Attempt to open a work file on an unenrolled device.

The **Connect to Work to access work files** box appears.

2. Click **Connect**.

The **Access work or school settings** page appears.

3. Sign-in to Azure AD as the employee and verify that the files now open

Related topics

- [Security Watch Deploying EFS: Part 1](#)
- [Protecting Data by Using EFS to Encrypt Hard Drives](#)
- [Create a Windows Information Protection \(WIP\) policy using Microsoft Intune](#)
- [Create a Windows Information Protection \(WIP\) policy using Microsoft Endpoint Configuration Manager](#)
- [Creating a Domain-Based Recovery Agent](#)

NOTE

Help to make this topic better by providing us with edits, additions, and feedback. For info about how to contribute to this topic, see [Contributing to this article](#).

Determine the Enterprise Context of an app running in Windows Information Protection (WIP)

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to:

- Windows 10, version 1607 and later

Learn more about what features and functionality are supported in each Windows edition at [Compare Windows 10 Editions](#).

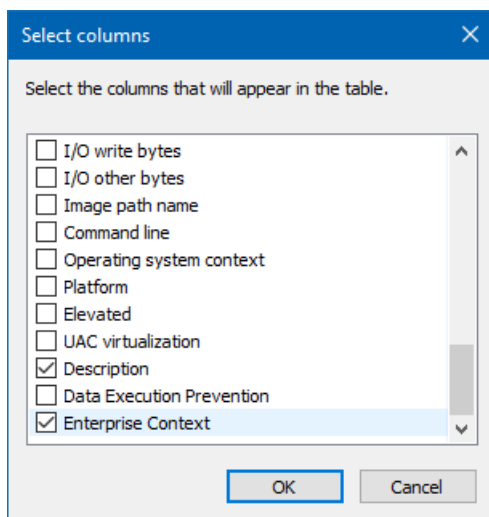
Use Task Manager to check the context of your apps while running in Windows Information Protection (WIP) to make sure that your organization's policies are applied and running correctly.

Viewing the Enterprise Context column in Task Manager

You need to add the Enterprise Context column to the **Details** tab of the Task Manager.

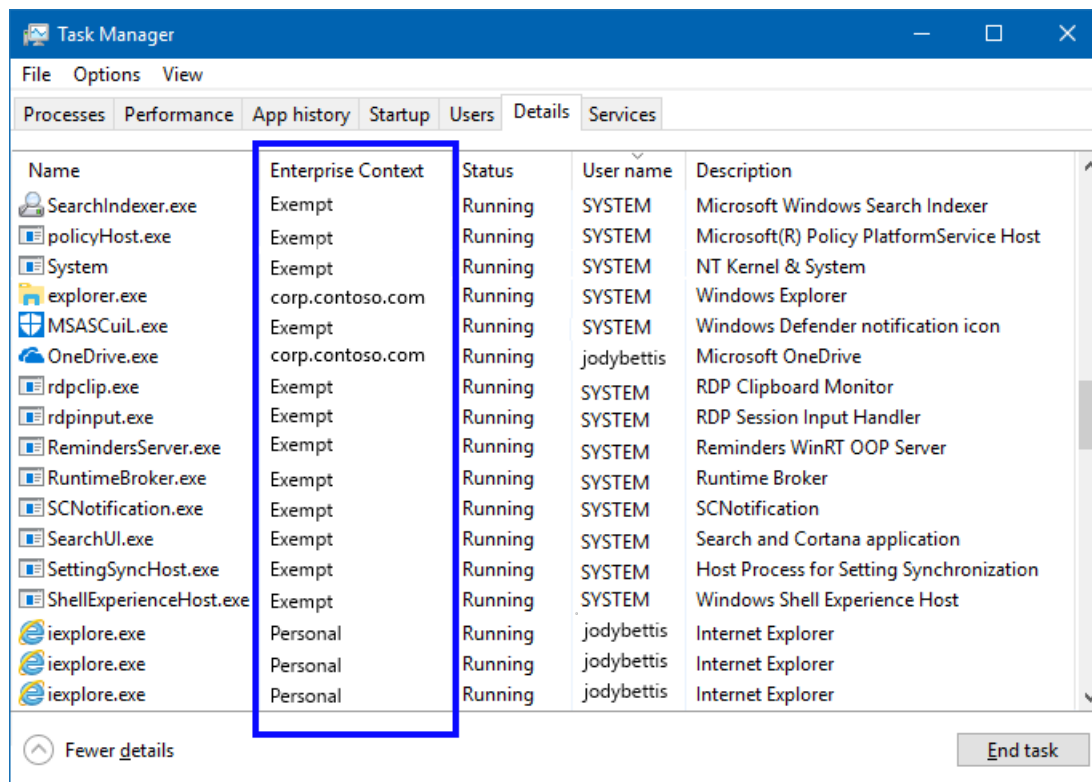
1. Make sure that you have an active Windows Information Protection policy deployed and turned on in your organization.
2. Open the Task Manager (taskmgr.exe), click the **Details** tab, right-click in the column heading area, and click **Select columns**.

The **Select columns** box appears.



3. Scroll down and check the **Enterprise Context** option, and then click **OK** to close the box.

The **Enterprise Context** column should now be available in Task Manager.



Review the Enterprise Context

The **Enterprise Context** column shows you what each app can do with your enterprise data:

- **Domain.** Shows the employee's work domain (such as, corp.contoso.com). This app is considered work-related and can freely touch and open work data and resources.
- **Personal.** Shows the text, *Personal*. This app is considered non-work-related and can't touch any work data or resources.
- **Exempt.** Shows the text, *Exempt*. Windows Information Protection policies don't apply to these apps (such as, system components).

IMPORTANT

Enlightened apps can change between Work and Personal, depending on the data being touched. For example, Microsoft Word 2016 shows as **Personal** when an employee opens a personal letter, but changes to **Work** when that same employee opens the company financials.

Create a Windows Information Protection (WIP) policy using Microsoft Endpoint Configuration Manager

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to:

- Windows 10, version 1607 and later

Microsoft Endpoint Manager helps you create and deploy your enterprise data protection (WIP) policy, including letting you choose your protected apps, your WIP-protection level, and how to find enterprise data on the network.

In this section

TOPIC	DESCRIPTION
Create and deploy a Windows Information Protection (WIP) policy using Microsoft Endpoint Configuration Manager	Microsoft Endpoint Manager helps you create and deploy your WIP policy, including letting you choose your protected apps, your WIP-protection level, and how to find enterprise data on the network.
Create and verify an Encrypting File System (EFS) Data Recovery Agent (DRA) certificate	Steps to create, verify, and perform a quick recovery using a Encrypting File System (EFS) Data Recovery Agent (DRA) certificate.
Determine the Enterprise Context of an app running in Windows Information Protection (WIP)	Use the Task Manager to determine whether an app is considered work, personal or exempt by Windows Information Protection (WIP).

Create and deploy a Windows Information Protection (WIP) policy using Microsoft Endpoint Configuration Manager

7/1/2022 • 20 minutes to read • [Edit Online](#)

Applies to:

- Windows 10, version 1607 and later
- Microsoft Endpoint Configuration Manager

Configuration Manager helps you create and deploy your Windows Information Protection (WIP) policy, including letting you choose your protected apps, your WIP-protection mode, and how to find enterprise data on the network.

Add a WIP policy

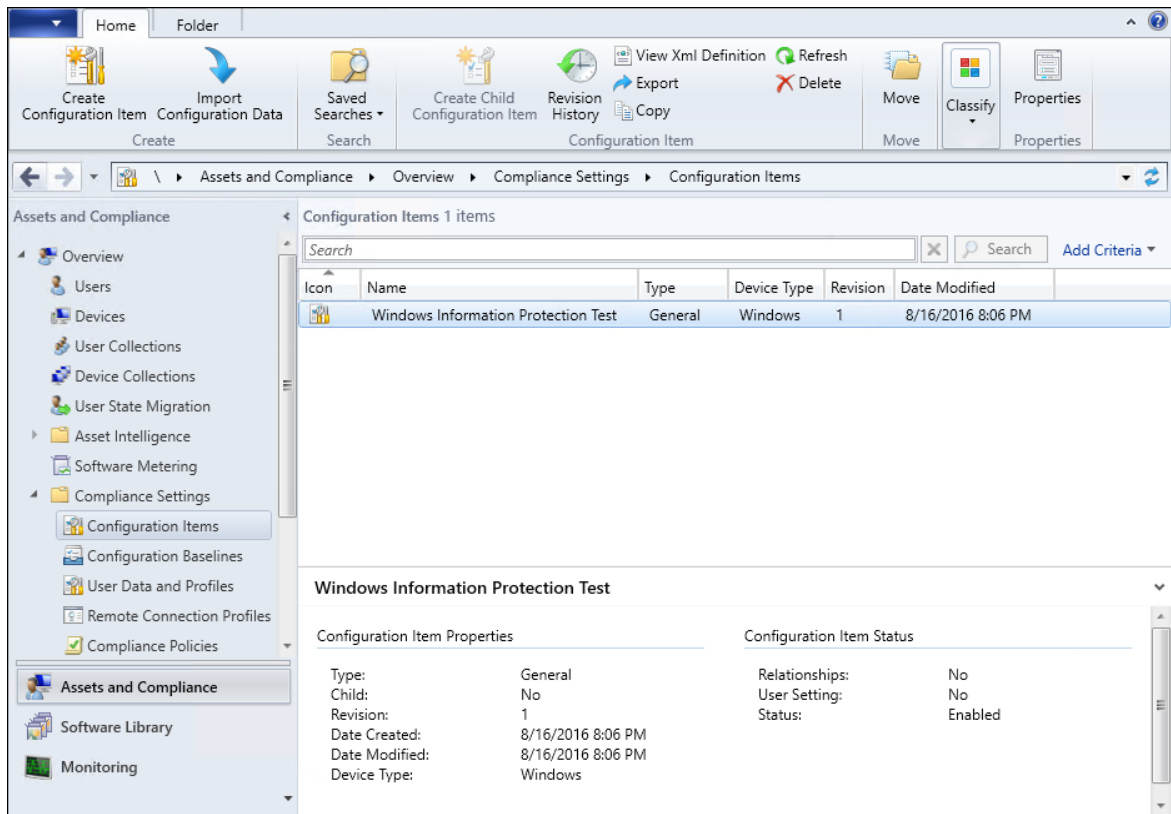
After you've installed and set up Configuration Manager for your organization, you must create a configuration item for WIP, which in turn becomes your WIP policy.

TIP

Review the [Limitations while using Windows Information Protection \(WIP\)](#) article before creating a new configuration item to avoid common issues.

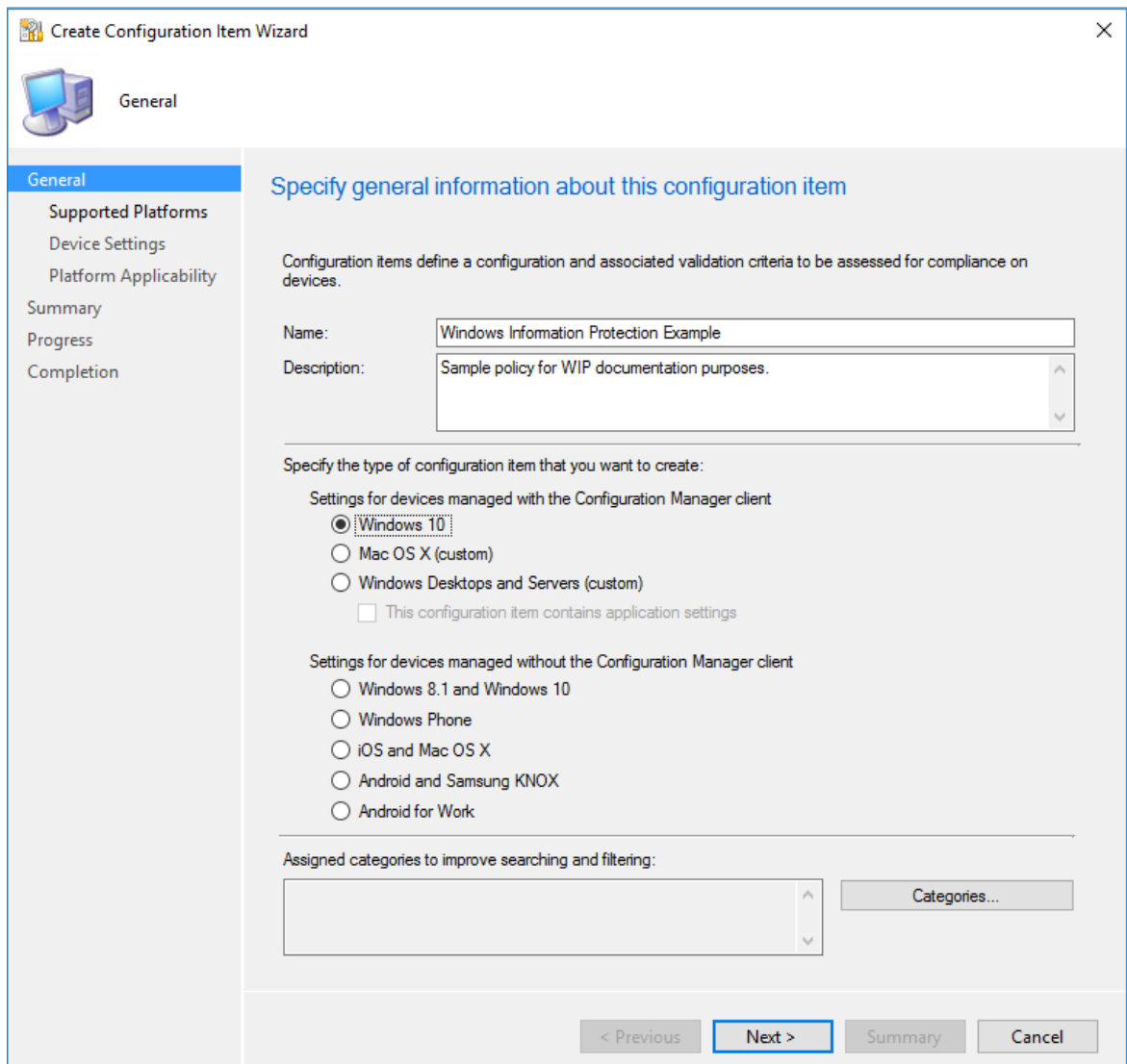
To create a configuration item for WIP

1. Open the Configuration Manager console, click the **Assets and Compliance** node, expand the **Overview** node, expand the **Compliance Settings** node, and then expand the **Configuration Items** node.

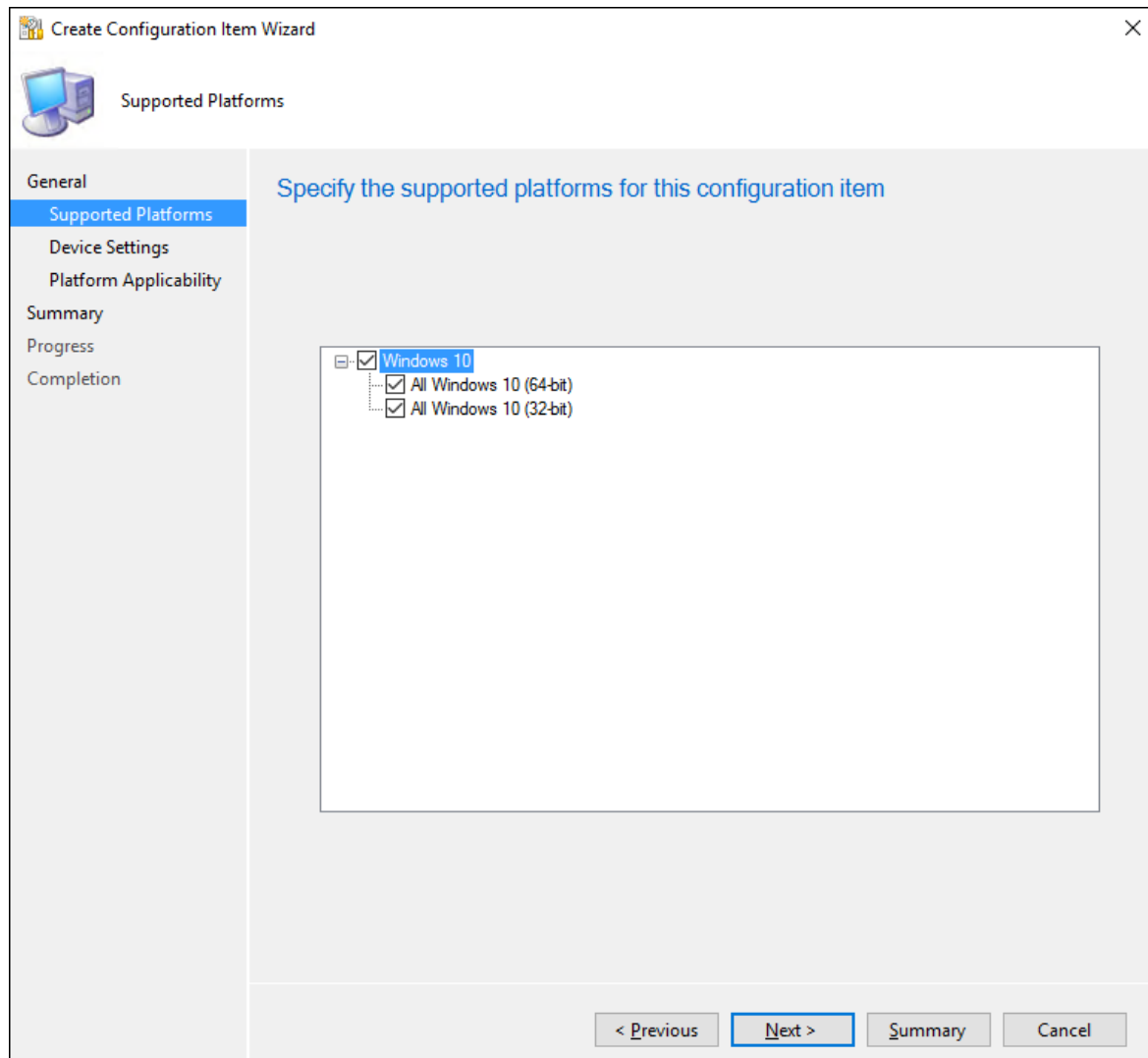


2. Click the Create Configuration Item button.

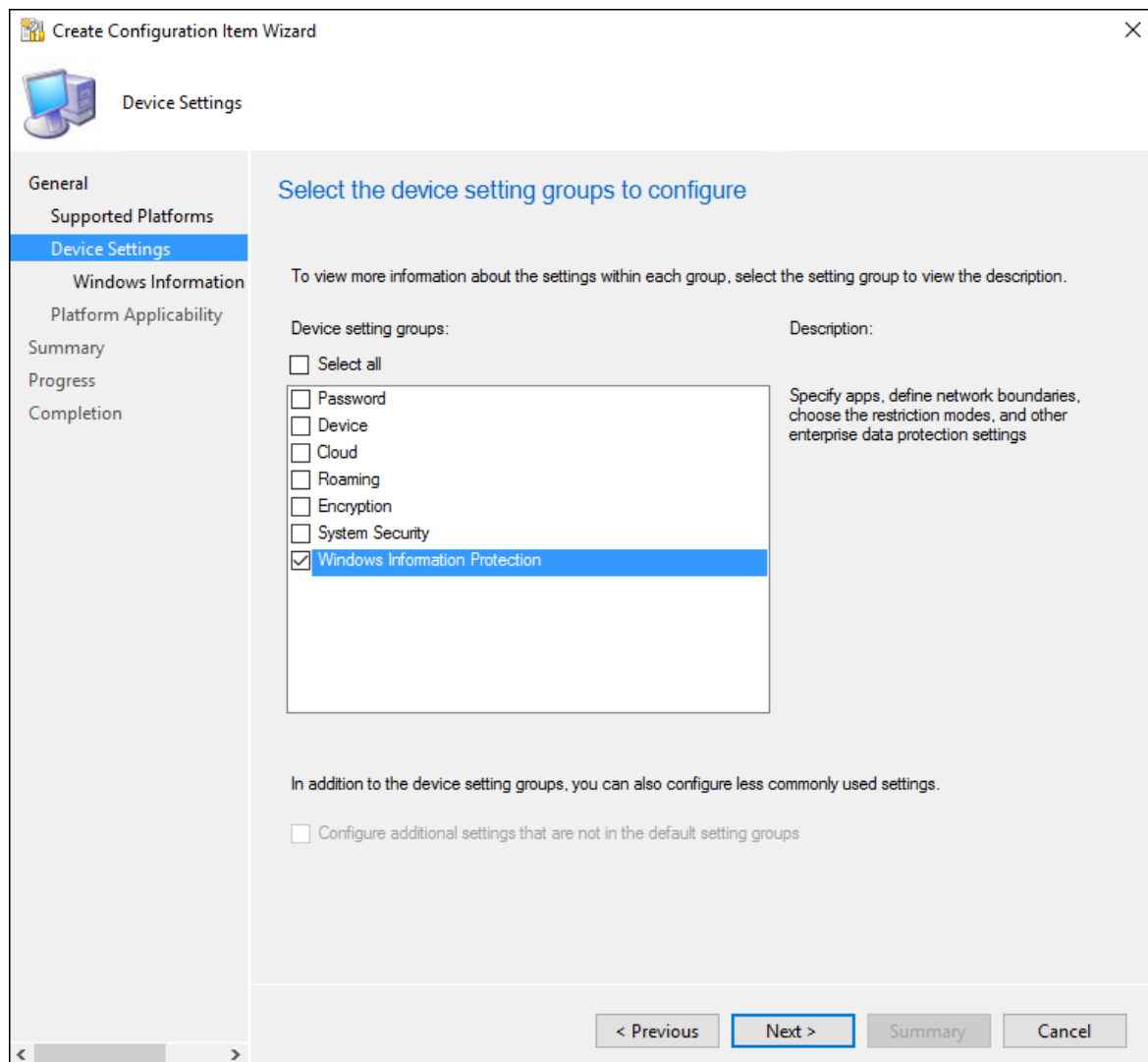
The Create Configuration Item Wizard starts.



3. On the **General Information** screen, type a name (required) and an optional description for your policy into the **Name** and **Description** boxes.
4. In the **Specify the type of configuration item you want to create** area, pick the option that represents whether you use Configuration Manager for device management, and then click **Next**.
 - **Settings for devices managed with the Configuration Manager client: Windows 10**
 - OR-
 - **Settings for devices managed without the Configuration Manager client: Windows 8.1 and Windows 10**
5. On the **Supported Platforms** screen, click the **Windows 10** box, and then click **Next**.



6. On the **Device Settings** screen, click **Windows Information Protection**, and then click **Next**.



The **Configure Windows Information Protection settings** page appears, where you'll configure your policy for your organization.

Add app rules to your policy

During the policy-creation process in Configuration Manager, you can choose the apps you want to give access to your enterprise data through Windows Information Protection. Apps included in this list can protect data on behalf of the enterprise and are restricted from copying or moving enterprise data to unprotected apps.

The steps to add your app rules are based on the type of rule template being applied. You can add a store app (also known as a Universal Windows Platform (UWP) app), a signed Windows desktop app, or an AppLocker policy file.

IMPORTANT

Enlightened apps are expected to prevent enterprise data from going to unprotected network locations and to avoid encrypting personal data. On the other hand, WIP-unaware apps might not respect the corporate network boundary, and WIP-unaware apps will encrypt all files they create or modify. This means that they could encrypt personal data and cause data loss during the revocation process.

Care must be taken to get a support statement from the software provider that their app is safe with Windows Information Protection before adding it to your **App rules** list. If you don't get this statement, it's possible that you could experience app compat issues due to an app losing the ability to access a necessary file after revocation.

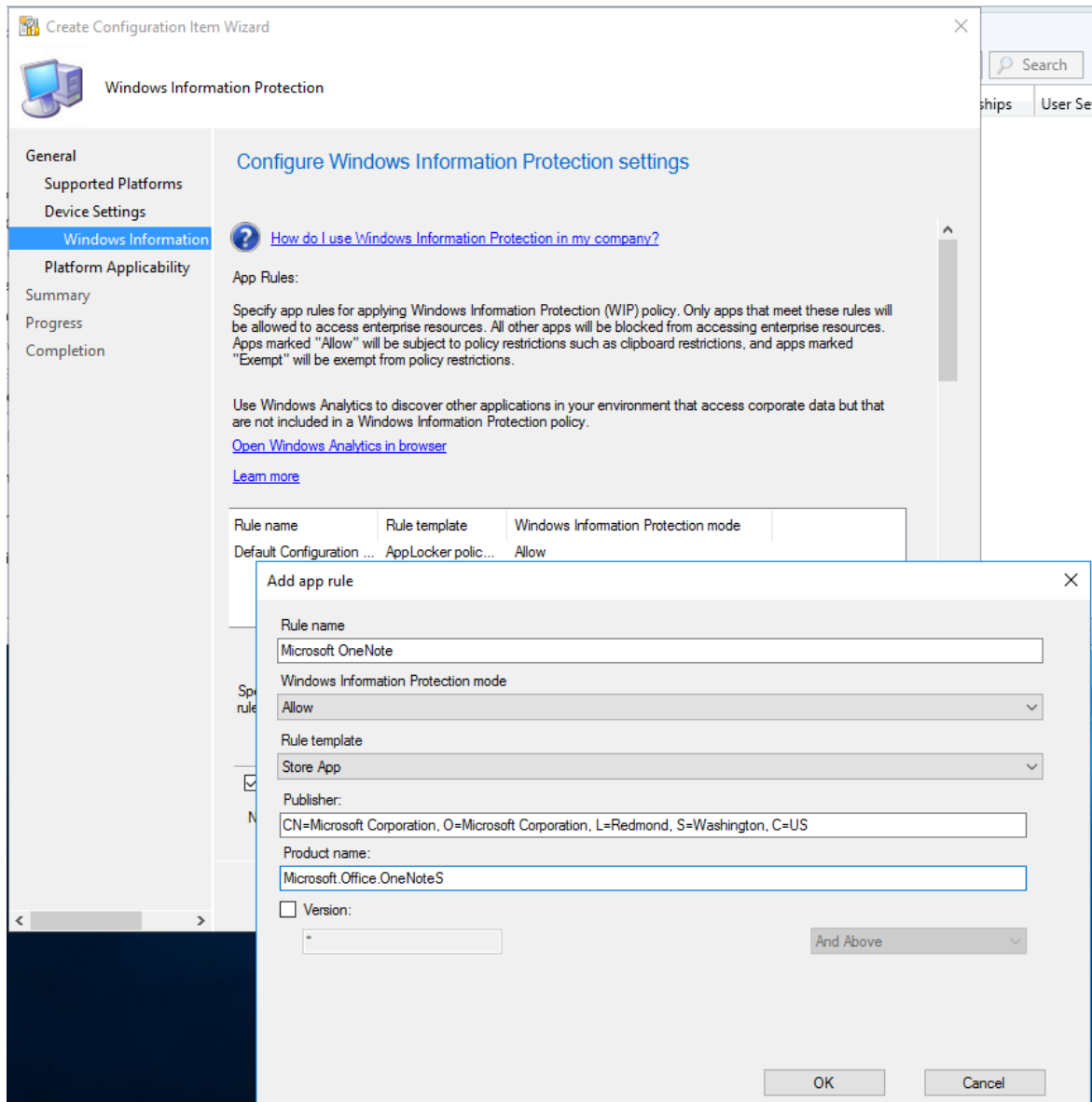
Add a store app rule to your policy

For this example, we're going to add Microsoft OneNote, a store app, to the **App Rules** list.

To add a store app

1. From the App rules area, click **Add**.

The **Add app rule** box appears.



2. Add a friendly name for your app into the **Title** box. In this example, it's *Microsoft OneNote*.
3. Click **Allow** from the **Windows Information Protection mode** drop-down list.

Allow turns on WIP, helping to protect that app's corporate data through the enforcement of WIP restrictions. If you want to exempt an app, you can follow the steps in the [Exempt apps from WIP restrictions](#) section.

4. Pick **Store App** from the **Rule template** drop-down list.

The box changes to show the store app rule options.

5. Type the name of the app and the name of its publisher, and then click **OK**. For this UWP app example, the **Publisher** is `CN=Microsoft Corporation, O=Microsoft Corporation, L=Redmond, S=Washington, C=US` and the **Product name** is `Microsoft.Office.OneNoteS`.

If you don't know the publisher or product name, you can find them for both desktop devices by following these steps.

To find the Publisher and Product Name values for Store apps without installing them

1. Go to the [Microsoft Store for Business](#) website, and find your app. For example, Microsoft OneNote.

NOTE

If your app is already installed on desktop devices, you can use the AppLocker local security policy MMC snap-in to gather the info for adding the app to the protected apps list. For info about how to do this, see the steps in [Add an AppLocker policy file](#) in this article.

2. Copy the ID value from the app URL. For example, Microsoft OneNote's ID URL is <https://www.microsoft.com/store/apps/onenote/9wzdncrfhvjl>, and you'd copy the ID value, `9wzdncrfhvjl`.
3. In a browser, run the Store for Business portal web API, to return a JavaScript Object Notation (JSON) file that includes the publisher and product name values. For example, run <https://bspmts.mp.microsoft.com/v1/public/catalog/Retail/Products/9wzdncrfhvjl/applockerdata>, where `9wzdncrfhvjl` is replaced with your ID value.

The API runs and opens a text editor with the app details.

```
{
  "packageIdentityName": "Microsoft.Office.OneNote",
  "publisherCertificateName": "CN=Microsoft Corporation, O=Microsoft Corporation, L=Redmond, S=Washington, C=US"
}
```

4. Copy the `publisherCertificateName` value and paste them into the **Publisher Name** box, copy the `packageIdentityName` value into the **Product Name** box of Intune.

IMPORTANT

The JSON file might also return a `windowsPhoneLegacyId` value for both the **Publisher Name** and **Product Name** boxes. This means that you have an app that's using a XAP package and that you must set the **Product Name** as `windowsPhoneLegacyId`, and set the **Publisher Name** as "CN=" followed by the `windowsPhoneLegacyId`.

For example:

```
{
  "windowsPhoneLegacyId": "ca05b3ab-f157-450c-8c49-a1f127f5e71d",
}
```

Add a desktop app rule to your policy

For this example, we're going to add Internet Explorer, a desktop app, to the **App Rules** list.

To add a desktop app to your policy

1. From the **App rules** area, click **Add**.

The **Add app rule** box appears.

2. Add a friendly name for your app into the **Title** box. In this example, it's *Internet Explorer*.

3. Click **Allow** from the **Windows Information Protection mode** drop-down list.

Allow turns on WIP, helping to protect that app's corporate data through the enforcement of WIP restrictions. If you want to exempt an app, you can follow the steps in the [Exempt apps from WIP restrictions](#) section.

4. Pick **Desktop App** from the **Rule template** drop-down list.

The box changes to show the desktop app rule options.

5. Pick the options you want to include for the app rule (see table), and then click **OK**.

OPTION	MANAGES
All fields left as "*"	All files signed by any publisher. (Not recommended.)
Publisher selected	All files signed by the named publisher. This might be useful if your company is the publisher and signer of internal line-of-business apps.
Publisher and Product Name selected	All files for the specified product, signed by the named publisher.
Publisher , Product Name , and Binary name selected	Any version of the named file or package for the specified product, signed by the named publisher.
Publisher , Product Name , Binary name , and File Version , and above , selected	Specified version or newer releases of the named file or package for the specified product, signed by the named publisher. This option is recommended for enlightened apps that weren't previously enlightened.
Publisher , Product Name , Binary name , and File Version , and below selected	Specified version or older releases of the named file or package for the specified product, signed by the named publisher.

OPTION	MANAGES
Publisher, Product Name, Binary name, and File Version, Exactly selected	Specified version of the named file or package for the specified product, signed by the named publisher.

If you're unsure about what to include for the publisher, you can run this PowerShell command:

```
Get-AppLockerFileInformation -Path "<path of the exe>"
```

Where "<path of the exe>" goes to the location of the app on the device. For example,

```
Get-AppLockerFileInformation -Path "C:\Program Files\Internet Explorer\iexplore.exe"
```

In this example, you'd get the following info:

```
Path                Publisher
----                -
%PROGRAMFILES%\INTERNET EXPLORER\IEXPLORE.EXE O=MICROSOFT CORPORATION, L=REDMOND, S=WASHINGTON, C=US\INTERNET EXPLOR...
```

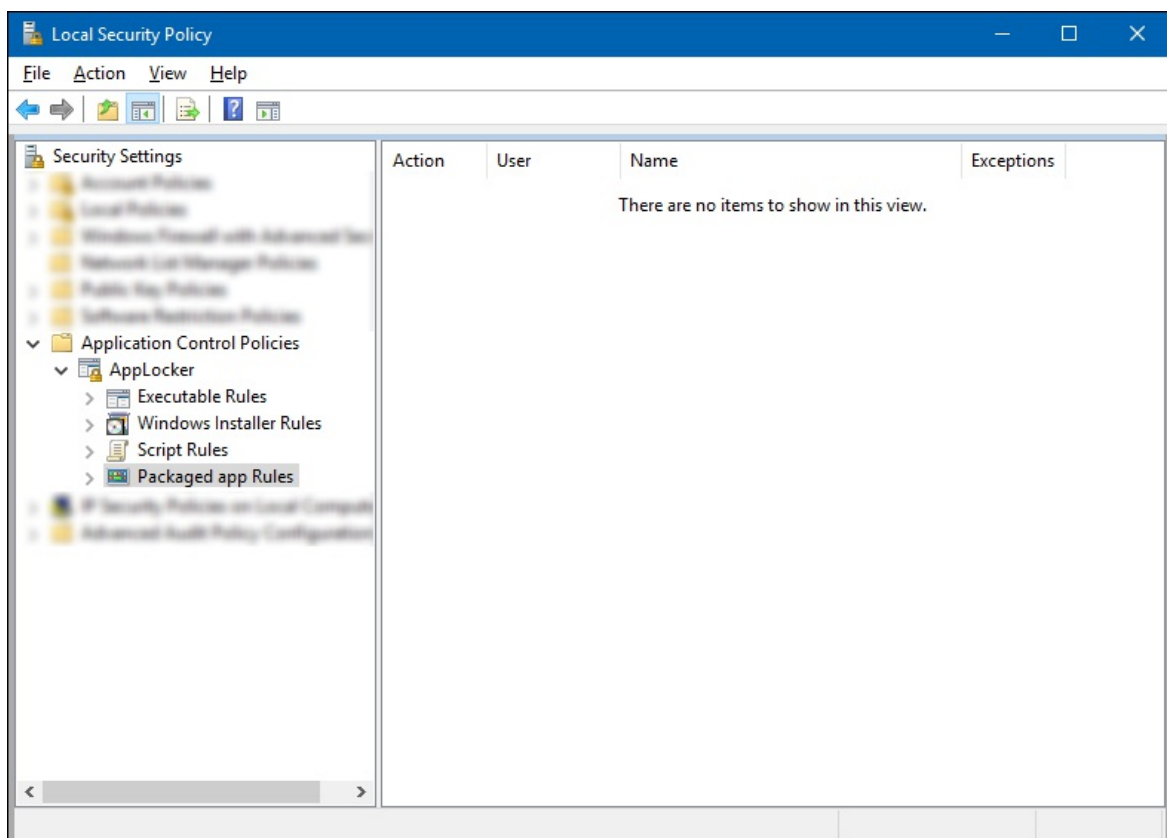
Where the text, `O=MICROSOFT CORPORATION, L=REDMOND, S=WASHINGTON, C=US` is the publisher name to enter in the **Publisher Name** box.

Add an AppLocker policy file

For this example, we're going to add an AppLocker XML file to the **App Rules** list. You'll use this option if you want to add multiple apps at the same time. For more info about AppLocker, see the [AppLocker](#) content.

To create an app rule and xml file using the AppLocker tool

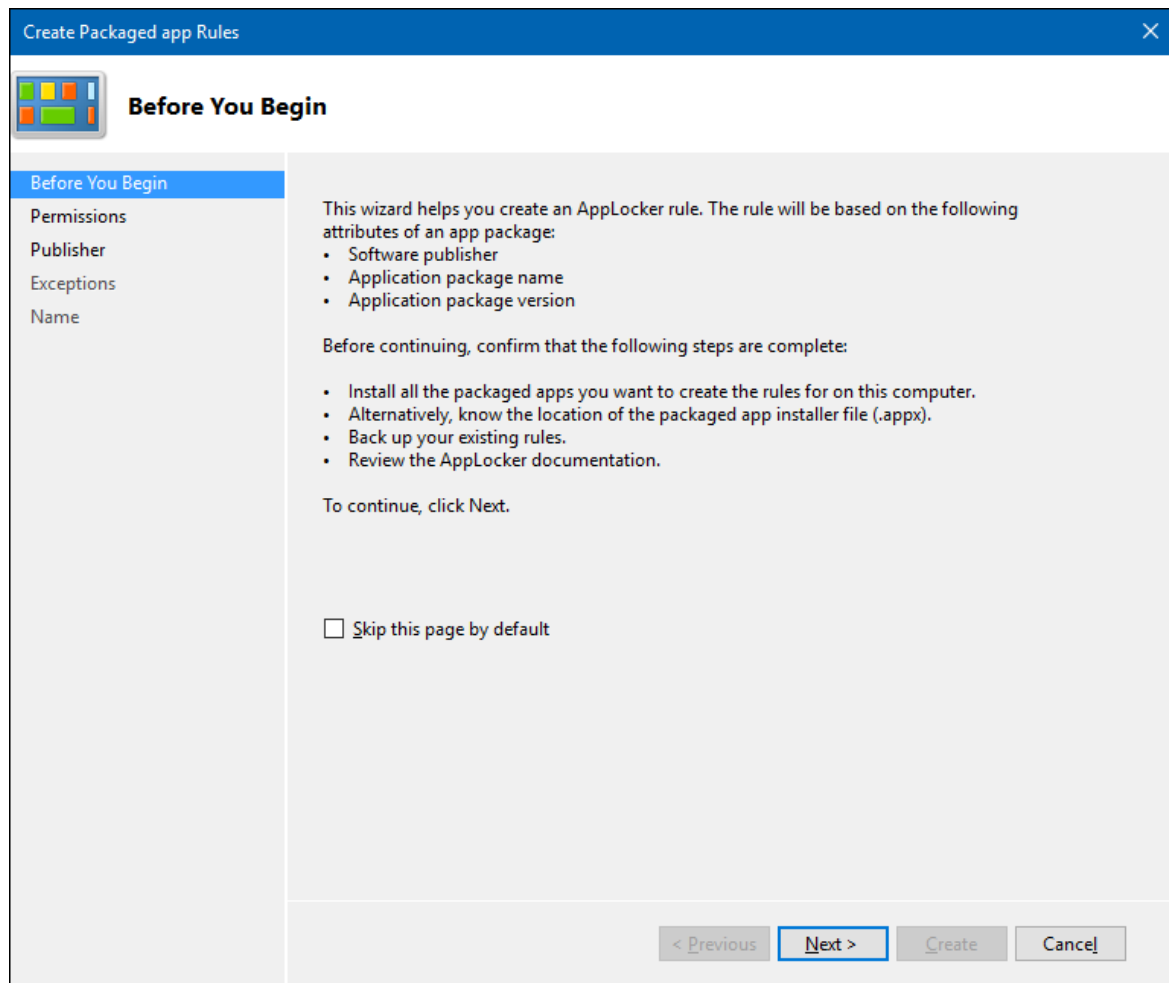
1. Open the Local Security Policy snap-in (SecPol.msc).
2. In the left pane, expand **Application Control Policies**, expand **AppLocker**, and then click **Packaged App Rules**.



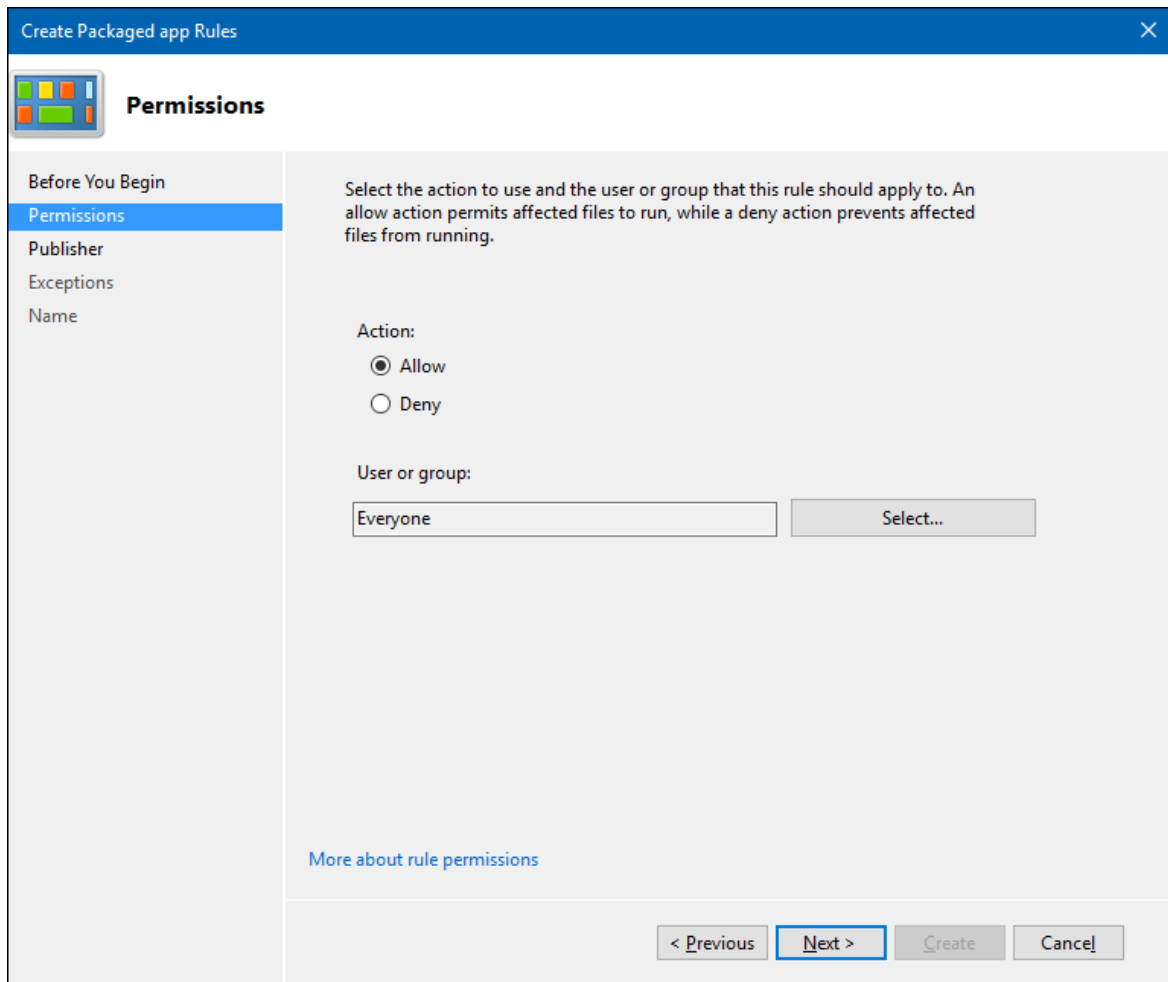
3. Right-click in the right-hand pane, and then click **Create New Rule**.

The **Create Packaged app Rules** wizard appears.

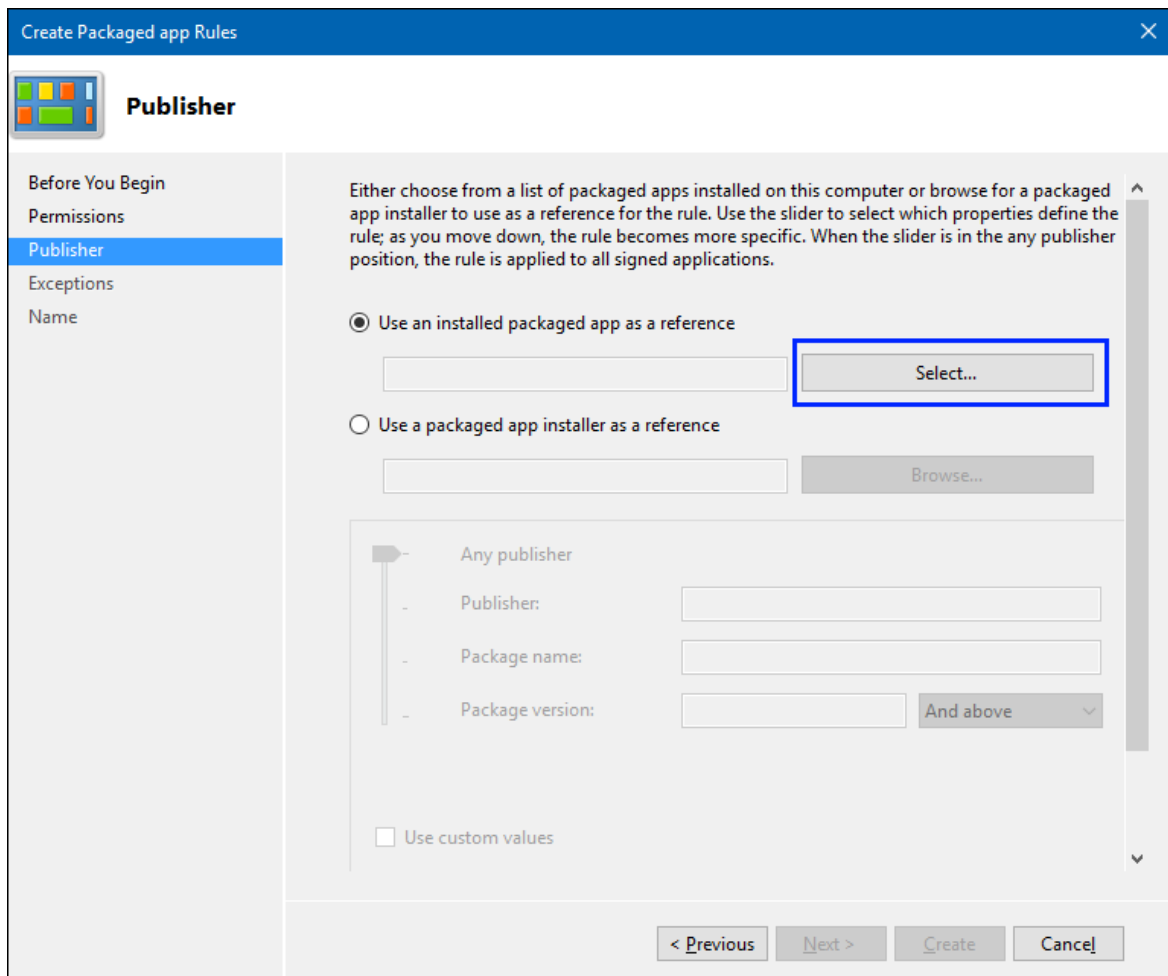
4. On the **Before You Begin** page, click **Next**.



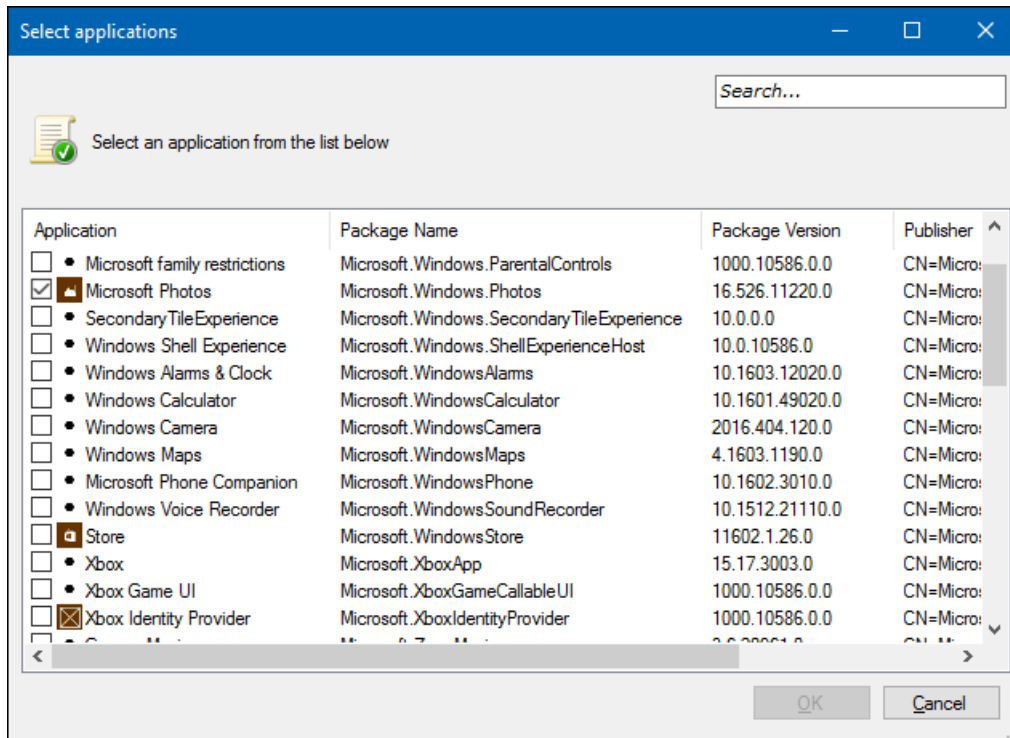
5. On the **Permissions** page, make sure the **Action** is set to **Allow** and the **User or group** is set to **Everyone**, and then click **Next**.



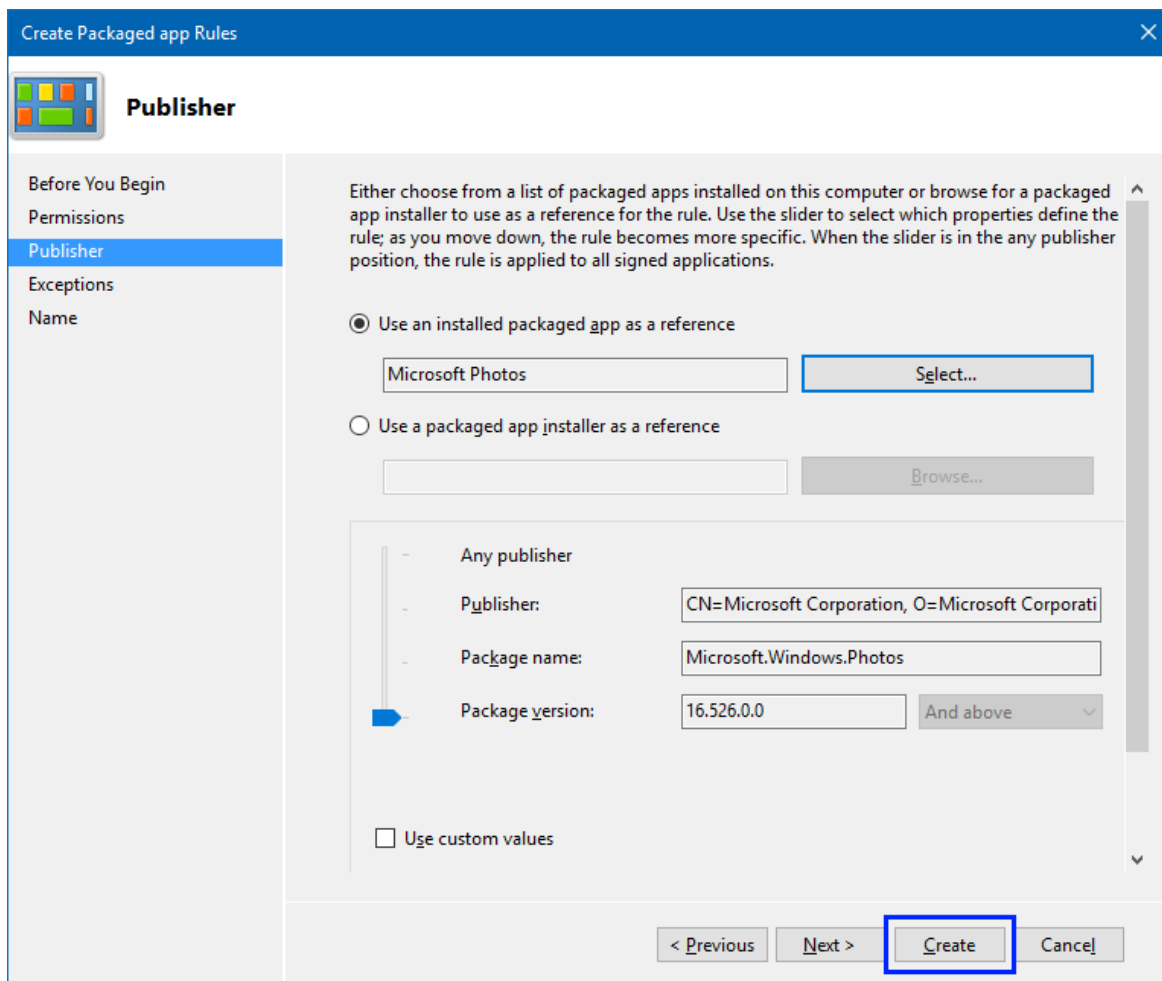
6. On the **Publisher** page, click **Select** from the **Use an installed packaged app as a reference** area.



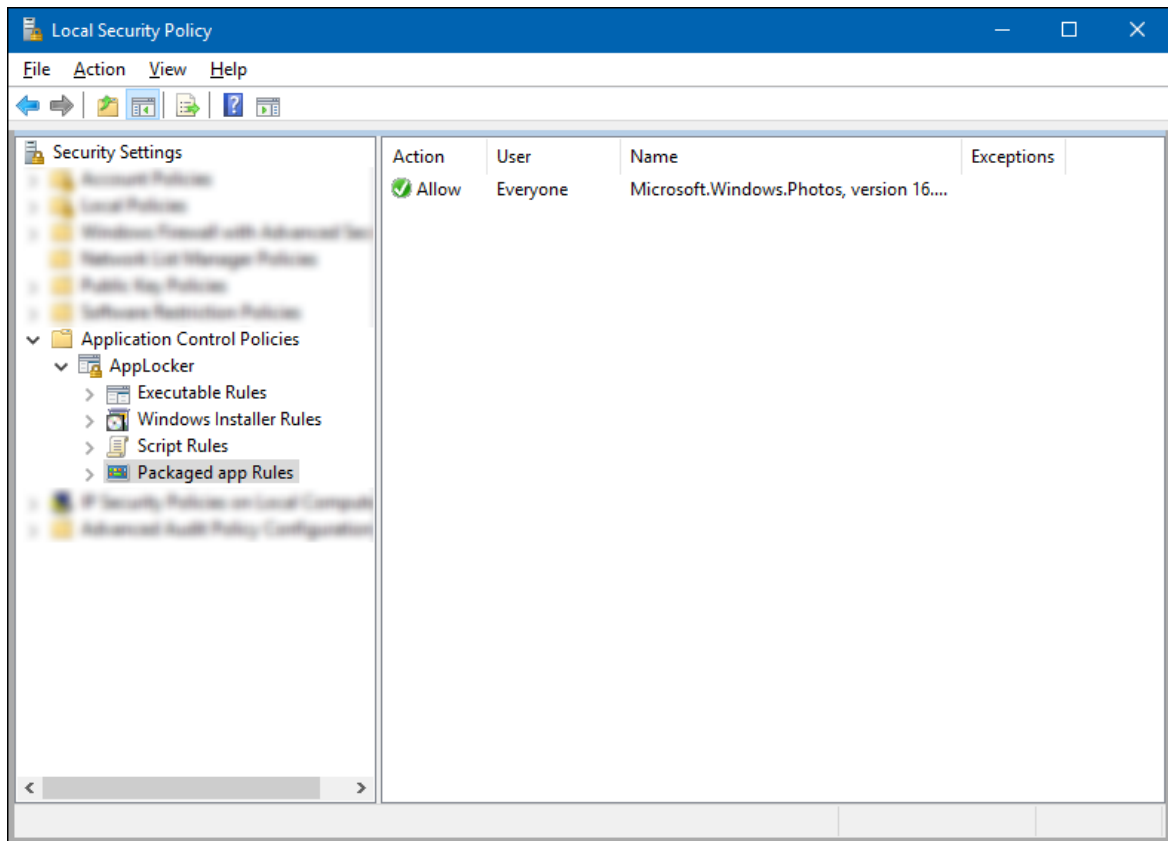
7. In the **Select applications** box, pick the app that you want to use as the reference for your rule, and then click OK. For this example, we're using Microsoft Photos.



8. On the updated **Publisher** page, click **Create**.

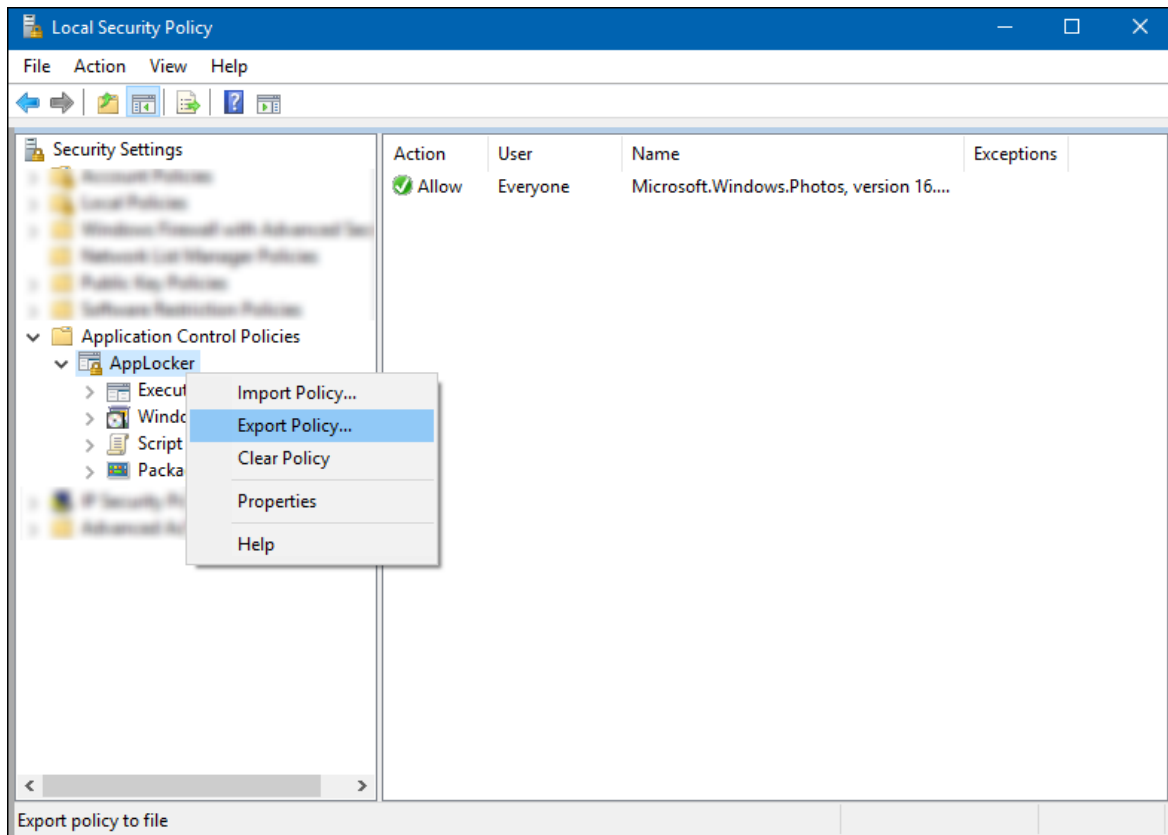


9. Review the Local Security Policy snap-in to make sure your rule is correct.



10. In the left pane, right-click on **AppLocker**, and then click **Export policy**.

The **Export policy** box opens, letting you export and save your new policy as XML.



11. In the **Export policy** box, browse to where the policy should be stored, give the policy a name, and then click **Save**.

The policy is saved and you'll see a message that says 1 rule was exported from the policy.

Example XML file

This is the XML file that AppLocker creates for Microsoft Photos.

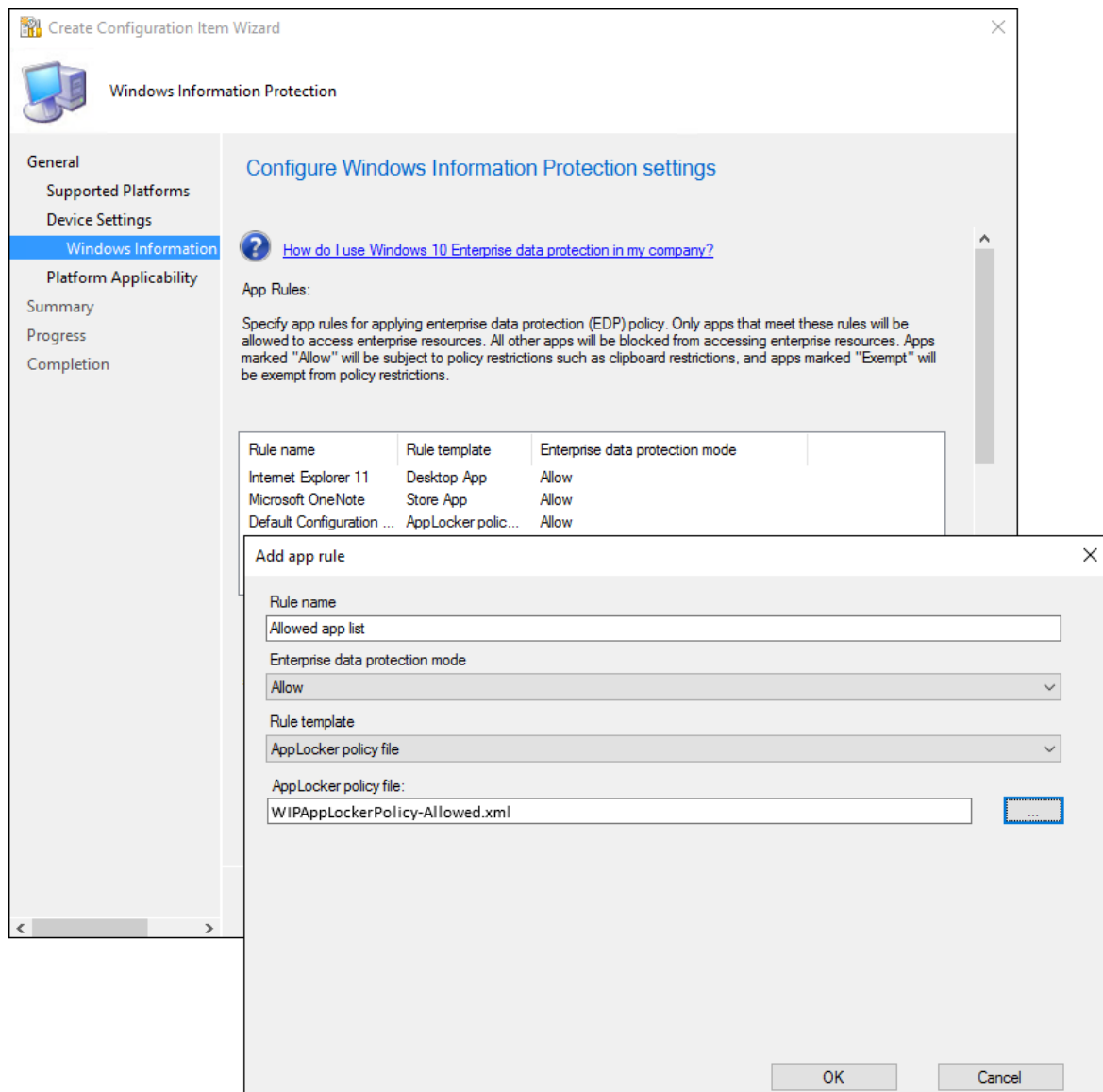
```
<AppLockerPolicy Version="1">
  <RuleCollection Type="Exe" EnforcementMode="NotConfigured" />
  <RuleCollection Type="Msi" EnforcementMode="NotConfigured" />
  <RuleCollection Type="Script" EnforcementMode="NotConfigured" />
  <RuleCollection Type="Dll" EnforcementMode="NotConfigured" />
  <RuleCollection Type="Appx" EnforcementMode="NotConfigured">
    <FilePublisherRule Id="5e0c752b-5921-4f72-8146-80ad5f582110" Name="Microsoft.Windows.Photos,
version 16.526.0.0 and above, from Microsoft Corporation" Description="" UserOrGroupSid="S-1-1-0"
Action="Allow">
      <Conditions>
        <FilePublisherCondition PublisherName="CN=Microsoft Corporation, O=Microsoft
Corporation, L=Redmond, S=Washington, C=US" ProductName="Microsoft.Windows.Photos" BinaryName="*">
          <BinaryVersionRange LowSection="16.526.0.0" HighSection="*" />
        </FilePublisherCondition>
      </Conditions>
    </FilePublisherRule>
  </RuleCollection>
</AppLockerPolicy>
```

12. After you've created your XML file, you need to import it by using Configuration Manager.

To import your AppLocker policy file app rule using Configuration Manager

1. From the App rules area, click **Add**.

The **Add app rule** box appears.



2. Add a friendly name for your app into the Title box. In this example, it's *Allowed app list*.

3. Click **ALLOW** from the **Windows Information Protection mode** drop-down list.

Allow turns on WIP, helping to protect that app's corporate data through the enforcement of WIP restrictions. If you want to exempt an app, you can follow the steps in the [Exempt apps from WIP restrictions](#) section.

4. Pick the **AppLocker policy file** from the **Rule template** drop-down list.

The box changes to let you import your AppLocker XML policy file.

5. Click the ellipsis (...) to browse for your AppLocker XML file, click **Open**, and then click **OK** to close the **Add app rule** box.

The file is imported and the apps are added to your **App Rules** list.

Exempt apps from WIP restrictions

If you're running into compatibility issues where your app is incompatible with Windows Information Protection (WIP), but still needs to be used with enterprise data, you can exempt the app from the WIP restrictions. This means that your apps won't include auto-encryption or tagging and won't honor your network restrictions. It also means that your exempted apps might leak.

To exempt a store app, a desktop app, or an AppLocker policy file app rule

1. From the **App rules** area, click **Add**.

The **Add app rule** box appears.

2. Add a friendly name for your app into the **Title** box. In this example, it's *Exempt apps list*.
3. Click **Exempt** from the **Windows Information Protection mode** drop-down list.

Be aware that when you exempt apps, they're allowed to bypass the WIP restrictions and access your corporate data. To allow apps, see [Add app rules to your policy](#) in this article.

4. Fill out the rest of the app rule info, based on the type of rule you're adding:
 - **Store app.** Follow the **Publisher** and **Product name** instructions in the [Add a store app rule to your policy](#) section of this topic.
 - **Desktop app.** Follow the **Publisher**, **Product name**, **Binary name**, and **Version** instructions in the [Add a desktop app rule to your policy](#) section of this topic.
 - **AppLocker policy file.** Follow the **Import** instructions in the [Add an AppLocker policy file](#) section of this topic, using a list of exempted apps.
5. Click **OK**.

Manage the WIP-protection level for your enterprise data

After you've added the apps you want to protect with WIP, you'll need to apply a management and protection mode.

We recommend that you start with **Silent** or **Override** while verifying with a small group that you have the right apps on your protected apps list. After you're done, you can change to your final enforcement policy, either **Override** or **Block**.

NOTE

For info about how to collect your audit log files, see [How to collect Windows Information Protection \(WIP\) audit event logs](#).

MODE	DESCRIPTION
Block	WIP looks for inappropriate data sharing practices and stops the employee from completing the action. This can include sharing info across non-enterprise-protected apps in addition to sharing enterprise data between other people and devices outside of your enterprise.
Override	WIP looks for inappropriate data sharing, warning employees if they do something deemed potentially unsafe. However, this management mode lets the employee override the policy and share the data, logging the action to your audit log.
Silent	WIP runs silently, logging inappropriate data sharing, without blocking anything that would've been prompted for employee interaction while in Override mode. Unallowed actions, like apps inappropriately trying to access a network resource or WIP-protected data, are still blocked.

MODE	DESCRIPTION
Off (not recommended)	<p>WIP is turned off and doesn't help to protect or audit your data.</p> <p>After you turn off WIP, an attempt is made to decrypt any WIP-tagged files on the locally attached drives. Be aware that your previous decryption and policy info isn't automatically reapplied if you turn WIP protection back on.</p>

The screenshot shows the 'Create Configuration Item Wizard' window for 'Windows Information Protection'. The 'Windows Information' tab is selected in the left-hand navigation pane. The main area is titled 'Configure Windows Information Protection settings' and contains the following elements:

- General** (selected in the left pane)
- Supported Platforms**
- Device Settings**
- Windows Information** (selected)
- Platform Applicability**
- Summary**
- Progress**
- Completion**

The main configuration area includes:

- A heading: **Configure Windows Information Protection settings**
- Text: Specify the paste/drop/share restriction mode for apps that meet the app criteria defined in the "App rules" section
- Four radio button options:
 - Block: Blocks paste/drop/share actions when attempting to move data out of enterprise locations and apps.
 - Override: Blocks paste/drop/share actions and displays a prompt to the user allowing them to override the block when attempting to move data out of enterprise locations and apps. Override actions are logged for audit.
 - Silent: Allows paste/drop/share actions when attempting to move data out of enterprise locations and apps. These actions are logged for audit.
 - Off: Turns off Windows Information Protection.
- Corporate identity (required): [Text input field]
- Corporate network definition:
 - Text: Define your corporate network boundary to be protected by Windows Information Protection. Access to these network locations will be restricted to only the apps that meet the app criteria defined in the "App rules" section.
 - Table with columns: Name, Network element, Network element definition. The table is empty and contains the message: "There are no items to show in this view."
- Remediate noncompliant settings
- Noncompliance severity for reports: [Dropdown menu set to 'None']
- Navigation buttons: < Previous, Next >, Summary, Cancel

Define your enterprise-managed identity domains

Corporate identity, usually expressed as your primary internet domain (for example, contoso.com), helps to identify and tag your corporate data from apps you've marked as protected by WIP. For example, emails using contoso.com are identified as being corporate and are restricted by your Windows Information Protection policies.

You can specify multiple domains owned by your enterprise by separating them with the "|" character. For example, (contoso.com|newcontoso.com). With multiple domains, the first one is designated as your corporate identity and all of the additional ones as being owned by the first one. We strongly recommend that you include all of your email address domains in this list.

To add your corporate identity

- Type the name of your corporate identity into the **Corporate identity** field. For example, `contoso.com` or `contoso.com|newcontoso.com`.

Corporate identity (required):

Choose where apps can access enterprise data

After you've added a protection mode to your apps, you'll need to decide where those apps can access enterprise data on your network.

There are no default locations included with WIP, you must add each of your network locations. This area applies to any network endpoint device that gets an IP address in your enterprise's range and is also bound to one of your enterprise domains, including SMB shares. Local file system locations should just maintain encryption (for example, on local NTFS, FAT, ExFAT).

IMPORTANT

Every WIP policy should include policy that defines your enterprise network locations. Classless Inter-Domain Routing (CIDR) notation isn't supported for WIP configurations.

To define where your protected apps can find and send enterprise data on you network

1. Add additional network locations your apps can access by clicking **Add**.

The **Add or edit corporate network definition** box appears.

2. Type a name for your corporate network element into the **Name** box, and then pick what type of network element it is, from the **Network element** drop-down box. This can include any of the options in the following table.

Add or Edit corporate network definition ✕

Specify network definitions using one of the available network types. "Enterprise Network Domain Names" and "Enterprise IP Ranges" are required fields.

Name:

Network element:

Enterprise Network Domain Names definition:

Specify the DNS names that form your corporate network. These are used in conjunction with the IP ranges that you specify to define your corporate network boundary. Multiple values can be specified by separating individual entries with a comma.

This setting is required to have enterprise data protection enabled.

For example: corp.contoso.com,region.contoso.com

- **Enterprise Cloud Resources:** Specify the cloud resources to be treated as corporate and protected by WIP.

For each cloud resource, you may also optionally specify a proxy server from your Internal proxy servers list to route traffic for this cloud resource. Be aware that all traffic routed through your Internal proxy servers is considered enterprise.

If you have multiple resources, you must separate them using the `|` delimiter. If you don't use proxy servers, you must also include the `,` delimiter just before the `|`. For example: URL `<,proxy>|URL <,proxy>`.

Format examples:

- **With proxy:**

`contoso.sharepoint.com,contoso.internalproxy1.com|contoso.visualstudio.com,contoso.internalproxy2.com`

- **Without proxy:** `contoso.sharepoint.com|contoso.visualstudio.com`

IMPORTANT

In some cases, such as when an app connects directly to a cloud resource through an IP address, Windows can't tell whether it's attempting to connect to an enterprise cloud resource or to a personal site. In this case, Windows blocks the connection by default. To stop Windows from automatically blocking these connections, you can add the `/AppCompat/` string to the setting. For example: URL `<,proxy>|URL <,proxy>/AppCompat/`.

- **Enterprise Network Domain Names (Required):** Specify the DNS suffixes used in your environment. All traffic to the fully-qualified domains appearing in this list will be protected.

This setting works with the IP ranges settings to detect whether a network endpoint is enterprise or personal on private networks.

If you have multiple resources, you must separate them using the `,` delimiter.

Format examples: `corp.contoso.com,region.contoso.com`

- **Proxy servers:** Specify the proxy servers your devices will go through to reach your cloud resources. Using this server type indicates that the cloud resources you're connecting to are enterprise resources.

This list shouldn't include any servers listed in your Internal proxy servers list. Internal proxy servers must be used only for WIP-protected (enterprise) traffic.

If you have multiple resources, you must separate them using the `;` delimiter.

Format examples: `proxy.contoso.com:80;proxy2.contoso.com:443`

- **Internal proxy servers:** Specify the internal proxy servers your devices will go through to reach your cloud resources. Using this server type indicates that the cloud resources you're connecting to are enterprise resources.

This list shouldn't include any servers listed in your Proxy servers list. Proxy servers must be used only for non-WIP-protected (non-enterprise) traffic.

If you have multiple resources, you must separate them using the `;` delimiter.

Format examples: `contoso.internalproxy1.com;contoso.internalproxy2.com`

- **Enterprise IPv4 Range (Required):** Specify the addresses for a valid IPv4 value range within your intranet. These addresses, used with your Enterprise Network Domain Names, define your corporate network boundaries.

If you have multiple ranges, you must separate them using the "," delimiter.

Format examples:

- **Starting IPv4 Address:** 3.4.0.1
- **Ending IPv4 Address:** 3.4.255.254
- **Custom URI:** 3.4.0.1-3.4.255.254, 10.0.0.1-10.255.255.254
- **Enterprise IPv6 Range:** Specify the addresses for a valid IPv6 value range within your intranet. These addresses, used with your Enterprise Network Domain Names, define your corporate network boundaries.

If you have multiple ranges, you must separate them using the "," delimiter.

Format examples:

- **Starting IPv6 Address:** 2a01:110::
- **Ending IPv6 Address:** 2a01:110:7fff:ffff:ffff:ffff:ffff:ffff
- **Custom URI:**
2a01:110:7fff:ffff:ffff:ffff:ffff:ffff, fd00::-fdff:ffff:ffff:ffff:ffff:ffff:ffff:ffff
- **Neutral Resources:** Specify your authentication redirection endpoints for your company. These locations are considered enterprise or personal, based on the context of the connection before the redirection.

If you have multiple resources, you must separate them using the "," delimiter.

Format examples: sts.contoso.com,sts.contoso2.com

3. Add as many locations as you need, and then click **OK**.

The **Add or edit corporate network definition** box closes.

4. Decide if you want to Windows to look for additional network settings and if you want to show the WIP icon on your corporate files while in File Explorer.



Windows Information Protection

General

Supported Platforms

Device Settings

Windows Information

Platform Applicability

Summary

Progress

Completion

Configure Windows Information Protection settings

Corporate identity (required):

Corporate network definition:

Define your corporate network boundary to be protected by Windows Information Protection. Access to these network locations will be restricted to only the apps that meet the app criteria defined in the "App rules" section.

Name	Network element	Network element definition
Enterprise Netwo...	Enterprise Network Domain Names	corp.contoso.com,region.cont...
Enterprise IPv4 r...	Enterprise IPv4 Ranges	3.4.0.1-3.4.255.254,10.0.0.1-...

Enterprise Proxy Servers list is authoritative (do not auto-detect)

Enterprise IP Ranges list is authoritative (do not auto-detect)

Remediate noncompliant settings

Noncompliance severity for reports:

- **Enterprise Proxy Servers list is authoritative (do not auto-detect).** Click this box if you want Windows to treat the proxy servers you specified in the network boundary definition as the complete list of proxy servers available on your network. If you clear this box, Windows will search for additional proxy servers in your immediate network. Not configured is the default option.
 - **Enterprise IP Ranges list is authoritative (do not auto-detect).** Click this box if you want Windows to treat the IP ranges you specified in the network boundary definition as the complete list of IP ranges available on your network. If you clear this box, Windows will search for additional IP ranges on any domain-joined devices connected to your network. Not configured is the default option.
 - **Show the Windows Information Protection icon overlay on your allowed apps that are WIP-unaware on corporate files in the File Explorer.** Click this box if you want the Windows Information Protection icon overlay to appear on corporate files in the Save As and File Explorer views. Additionally, for unenlightened but allowed apps, the icon overlay also appears on the app tile and with *Managed* text on the app name in the Start menu. Not configured is the default option.
5. In the required **Upload a Data Recovery Agent (DRA) certificate to allow recovery of encrypted data** box, click **Browse** to add a data recovery certificate for your policy.

Upload a DRA (Data Recovery Agent) certificate to allow recovery of encrypted data (required):

After you create and deploy your WIP policy to your employees, Windows will begin to encrypt your corporate data on the employees' local device drive. If somehow the employees' local encryption keys get lost or revoked, the encrypted data can become unrecoverable. To help avoid this possibility, the DRA certificate lets Windows use an included public key to encrypt the local data, while you maintain the private key that can unencrypt the data.

For more info about how to find and export your data recovery certificate, see [Data Recovery and Encrypting File System \(EFS\)](#). For more info about creating and verifying your EFS DRA certificate, see [Create and verify an Encrypting File System \(EFS\) Data Recovery Agent \(DRA\) certificate](#).

Choose your optional WIP-related settings

After you've decided where your protected apps can access enterprise data on your network, you'll be asked to decide if you want to add any optional WIP settings.

The screenshot shows the 'Create Configuration Item Wizard' window for 'Windows Information Protection'. The 'Windows Information' tab is selected in the left-hand navigation pane. The main area is titled 'Configure Windows Information Protection settings' and contains several configuration options:

- file icons in the File Explorer.** (This option is currently disabled/grayed out.)
- Upload a DRA (Data Recovery Agent) certificate to allow recovery of encrypted data (required):** A text box contains 'EFSRA.CER' and a 'Browse...' button is to its right.
- Prevent corporate data from being accessed by apps when the device is locked. (Applies only to Windows 10 Mobile)** A dropdown menu is set to 'Not Configured'.
- Allow Windows Search to search encrypted corporate data and Store apps.** A dropdown menu is set to 'Not Configured'.
- Revoke encryption keys on un-enroll (This setting only applies to devices managed without the Configuration Manager client)** A dropdown menu is set to 'Not Configured'.
- Allow Azure RMS** A dropdown menu is set to 'Not Configured'.
- Optionally provide the Azure RMS template GUID:** An empty text box is provided.
- Remediate noncompliant settings**
- Noncompliance severity for reports:** A dropdown menu is set to 'None'.

At the bottom of the wizard, there are four buttons: '< Previous', 'Next >', 'Summary', and 'Cancel'.

To set your optional settings

1. Choose to set any or all of the optional settings:

- **Allow Windows Search to search encrypted corporate data and Store apps.** Determines whether Windows Search can search and index encrypted corporate data and Store apps. The options are:
 - **Yes.** Allows Windows Search to search and index encrypted corporate data and Store apps.

- **No, or not configured (recommended).** Stops Windows Search from searching and indexing encrypted corporate data and Store apps.
- **Revoke local encryption keys during the unenrollment process.** Determines whether to revoke a user's local encryption keys from a device when it's unenrolled from Windows Information Protection. If the encryption keys are revoked, a user no longer has access to encrypted corporate data. The options are:
 - **Yes, or not configured (recommended).** Revokes local encryption keys from a device during unenrollment.
 - **No.** Stop local encryption keys from being revoked from a device during unenrollment. For example, if you're migrating between Mobile Device Management (MDM) solutions.
- **Allow Azure RMS.** Enables secure sharing of files by using removable media such as USB drives. For more information about how RMS works with WIP, see [Create a WIP policy using Intune](#). To confirm what templates your tenant has, run [Get-AadrmTemplate](#) from the [AADRM PowerShell module](#). If you don't specify a template, WIP uses a key from a default RMS template that everyone in the tenant will have access to.

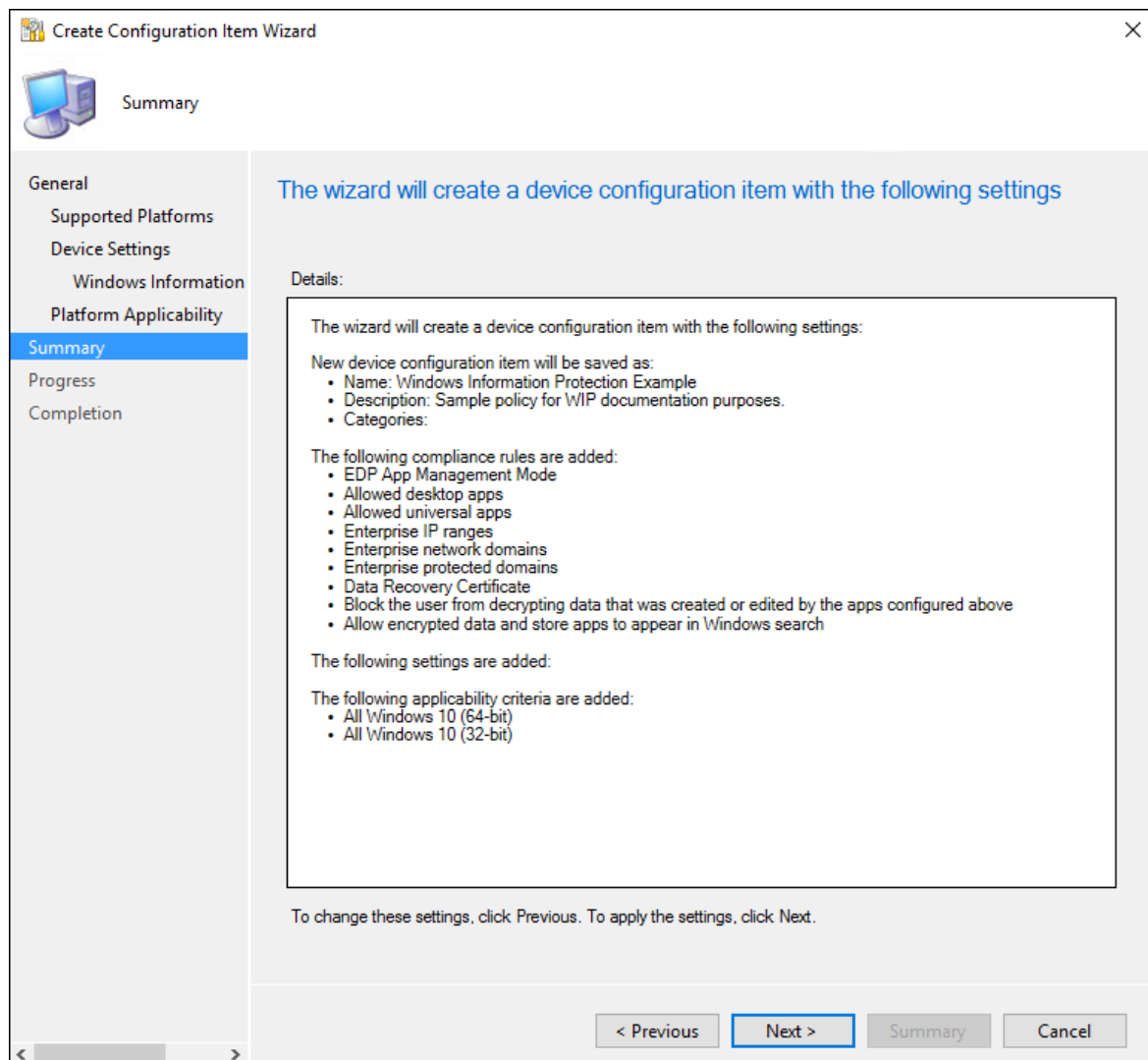
2. After you pick all of the settings you want to include, click **Summary**.

Review your configuration choices in the Summary screen

After you've finished configuring your policy, you can review all of your info on the **Summary** screen.

To view the Summary screen

- Click the **Summary** button to review your policy choices, and then click **Next** to finish and to save your policy.



A progress bar appears, showing you progress for your policy. After it's done, click **Close** to return to the **Configuration Items** page.

Deploy the WIP policy

After you've created your WIP policy, you'll need to deploy it to your organization's devices. For info about your deployment options, see these topics:

- [Operations and Maintenance for Compliance Settings in Configuration Manager](#)
- [How to Create Configuration Baselines for Compliance Settings in Configuration Manager](#)
- [How to Deploy Configuration Baselines in Configuration Manager](#)

Related topics

- [How to collect Windows Information Protection \(WIP\) audit event logs](#)
- [General guidance and best practices for Windows Information Protection \(WIP\)](#)
- [Limitations while using Windows Information Protection \(WIP\)](#)

Create and verify an Encrypting File System (EFS) Data Recovery Agent (DRA) certificate

7/1/2022 • 5 minutes to read • [Edit Online](#)

Applies to:

- Windows 10, version 1607 and later

If you don't already have an EFS DRA certificate, you'll need to create and extract one from your system before you can use Windows Information Protection (WIP), formerly known as enterprise data protection (EDP), in your organization. For the purposes of this section, we'll use the file name EFSDRA; however, this name can be replaced with anything that makes sense to you.

IMPORTANT

If you already have an EFS DRA certificate for your organization, you can skip creating a new one. Just use your current EFS DRA certificate in your policy. For more info about when to use a PKI and the general strategy you should use to deploy DRA certificates, see the [Security Watch Deploying EFS: Part 1](#) article on TechNet. For more general info about EFS protection, see [Protecting Data by Using EFS to Encrypt Hard Drives](#).

If your DRA certificate has expired, you won't be able to encrypt your files with it. To fix this, you'll need to create a new certificate, using the steps in this topic, and then deploy it through policy.

Manually create an EFS DRA certificate

1. On a computer without an EFS DRA certificate installed, open a command prompt with elevated rights, and then navigate to where you want to store the certificate.
2. Run this command:

```
cipher /r:EFSDRA
```

Where *EFSDRA* is the name of the `.cer` and `.pfx` files that you want to create.

3. When prompted, type and confirm a password to help protect your new Personal Information Exchange (.pfx) file.

The EFSDRA.cer and EFSDRA.pfx files are created in the location you specified in Step 1.

IMPORTANT

Because the private keys in your DRA .pfx files can be used to decrypt any WIP file, you must protect them accordingly. We highly recommend storing these files offline, keeping copies on a smart card with strong protection for normal use and master copies in a secured physical location.

4. Add your EFS DRA certificate to your WIP policy using a deployment tool, such as [Microsoft Intune](#) or [Microsoft Endpoint Configuration Manager](#).

NOTE

This certificate can be used in Intune for policies both *with* device enrollment (MDM) and *without* device enrollment (MAM).

Verify your data recovery certificate is correctly set up on a WIP client computer

1. Find or create a file that's encrypted using Windows Information Protection. For example, you could open an app on your allowed app list, and then create and save a file so it's encrypted by WIP.
2. Open an app on your protected app list, and then create and save a file so that it's encrypted by WIP.
3. Open a command prompt with elevated rights, navigate to where you stored the file you just created, and then run this command:

```
cipher /c filename
```

Where *filename* is the name of the file you created in Step 1.

4. Make sure that your data recovery certificate is listed in the **Recovery Certificates** list.

Recover your data using the EFS DRA certificate in a test environment

1. Copy your WIP-encrypted file to a location where you have admin access.
2. Install the EFS DRA.pfx file, using its password.
3. Open a command prompt with elevated rights, navigate to the encrypted file, and then run this command:

```
cipher /d encryptedfile.extension
```

Where *encryptedfile.extension* is the name of your encrypted file. For example, `corporatedata.docx`.

Recover WIP-protected after unenrollment

It's possible that you might revoke data from an unenrolled device only to later want to restore it all. This can happen in the case of a missing device being returned or if an unenrolled employee enrolls again. If the employee enrolls again using the original user profile, and the revoked key store is still on the device, all of the revoked data can be restored at once.

IMPORTANT

To maintain control over your enterprise data, and to be able to revoke again in the future, you must only perform this process after the employee has re-enrolled the device.

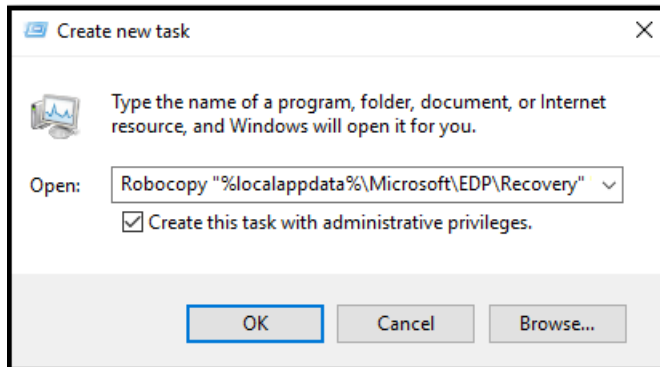
1. Have the employee sign in to the unenrolled device, open an elevated command prompt, and type:

```
Robocopy "%localappdata%\Microsoft\EDP\Recovery" "new_location" * /EFSRAW
```

Where *"new_location"* is in a different directory. This can be on the employee's device or on a shared

folder on a computer that runs Windows 8 or Windows Server 2012 or newer and can be accessed while you're logged in as a data recovery agent.

To start Robocopy in S mode, open Task Manager. Click **File > Run new task**, type the command, and click **Create this task with administrative privileges**.



If the employee performed a clean installation and there is no user profile, you need to recover the keys from the System Volume folder in each drive. Type:

```
Robocopy "drive_letter:\System Volume Information\EDP\Recovery\" "new_location" * /EFSRAW
```

2. Sign in to a different device with administrator credentials that have access to your organization's DRA certificate, and perform the file decryption and recovery by typing:

```
cipher.exe /D "new_location"
```

3. Have your employee sign in to the unenrolled device, and type:

```
Robocopy "new_location" "%localappdata%\Microsoft\EDP\Recovery\Input"
```

4. Ask the employee to lock and unlock the device.

The Windows Credential service automatically recovers the employee's previously revoked keys from the `Recovery\Input` location.

Auto-recovery of encryption keys

Starting with Windows 10, version 1709, WIP includes a data recovery feature that lets your employees auto-recover access to work files if the encryption key is lost and the files are no longer accessible. This typically happens if an employee reimages the operating system partition, removing the WIP key info, or if a device is reported as lost and you mistakenly target the wrong device for unenrollment.

To help make sure employees can always access files, WIP creates an auto-recovery key that's backed up to their Azure Active Directory (Azure AD) identity.

The employee experience is based on sign in with an Azure AD work account. The employee can either:

- Add a work account through the **Windows Settings > Accounts > Access work or school > Connect** menu.

-OR-

- Open **Windows Settings > Accounts > Access work or school > Connect** and choose the **Join this device to Azure Active Directory** link, under **Alternate actions**.

NOTE

To perform an Azure AD Domain Join from the Settings page, the employee must have administrator privileges to the device.

After signing in, the necessary WIP key info is automatically downloaded and employees are able to access the files again.

To test what the employee sees during the WIP key recovery process

1. Attempt to open a work file on an unenrolled device.

The **Connect to Work to access work files** box appears.

2. Click **Connect**.

The **Access work or school settings** page appears.

3. Sign-in to Azure AD as the employee and verify that the files now open

Related topics

- [Security Watch Deploying EFS: Part 1](#)
- [Protecting Data by Using EFS to Encrypt Hard Drives](#)
- [Create a Windows Information Protection \(WIP\) policy using Microsoft Intune](#)
- [Create a Windows Information Protection \(WIP\) policy using Microsoft Endpoint Configuration Manager](#)
- [Creating a Domain-Based Recovery Agent](#)

NOTE

Help to make this topic better by providing us with edits, additions, and feedback. For info about how to contribute to this topic, see [Contributing to this article](#).

Determine the Enterprise Context of an app running in Windows Information Protection (WIP)

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to:

- Windows 10, version 1607 and later

Learn more about what features and functionality are supported in each Windows edition at [Compare Windows 10 Editions](#).

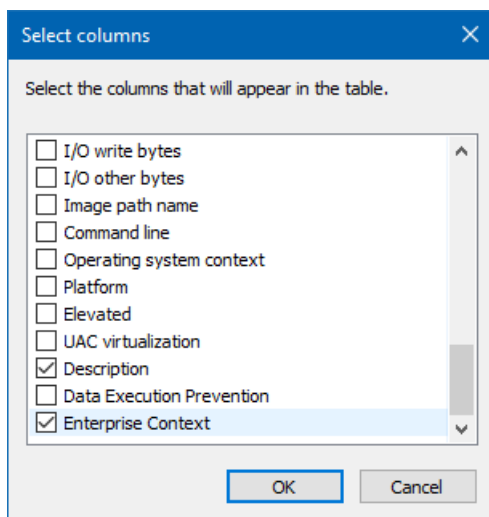
Use Task Manager to check the context of your apps while running in Windows Information Protection (WIP) to make sure that your organization's policies are applied and running correctly.

Viewing the Enterprise Context column in Task Manager

You need to add the Enterprise Context column to the **Details** tab of the Task Manager.

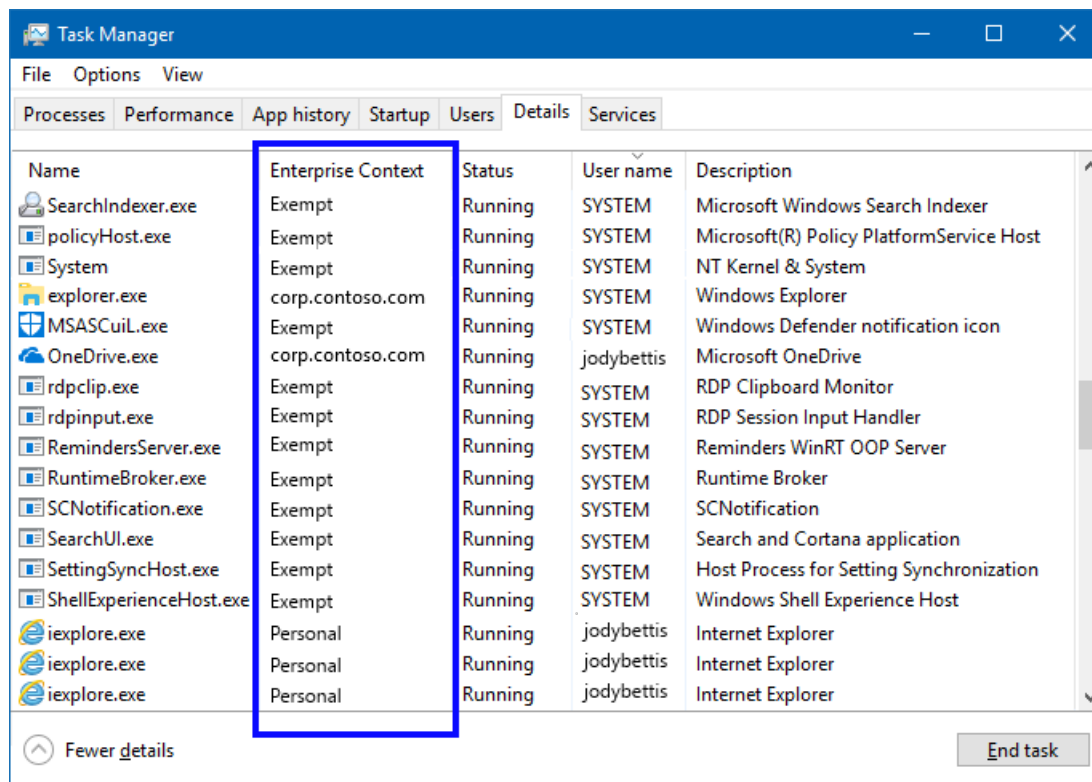
1. Make sure that you have an active Windows Information Protection policy deployed and turned on in your organization.
2. Open the Task Manager (taskmgr.exe), click the **Details** tab, right-click in the column heading area, and click **Select columns**.

The **Select columns** box appears.



3. Scroll down and check the **Enterprise Context** option, and then click **OK** to close the box.

The **Enterprise Context** column should now be available in Task Manager.



Review the Enterprise Context

The **Enterprise Context** column shows you what each app can do with your enterprise data:

- **Domain.** Shows the employee's work domain (such as, corp.contoso.com). This app is considered work-related and can freely touch and open work data and resources.
- **Personal.** Shows the text, *Personal*. This app is considered non-work-related and can't touch any work data or resources.
- **Exempt.** Shows the text, *Exempt*. Windows Information Protection policies don't apply to these apps (such as, system components).

IMPORTANT

Enlightened apps can change between Work and Personal, depending on the data being touched. For example, Microsoft Word 2016 shows as **Personal** when an employee opens a personal letter, but changes to **Work** when that same employee opens the company financials.

Mandatory tasks and settings required to turn on Windows Information Protection (WIP)

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to:

- Windows 10, version 1607 and later

This list provides all of the tasks and settings that are required for the operating system to turn on Windows Information Protection (WIP), formerly known as enterprise data protection (EDP), in your enterprise.

TASK	DESCRIPTION
Add at least one app of each type (Store and Desktop) to the Protected apps list in your WIP policy.	You must have at least one Store app and one Desktop app added to your Protected apps list. For more info about where this area is and how to add apps, see the Add apps to your Protected apps list section of the policy creation topics.
Choose your Windows Information Protection protection level.	You must choose the level of protection you want to apply to your WIP-protected content, including Allow Overrides , Silent , or Block . For more info about where this area is and how to decide on your protection level, see the Manage Windows Information Protection mode for your enterprise data section of the policy creation topics. For info about how to collect your audit log files, see How to collect Windows Information Protection (WIP) audit event logs .
Specify your corporate identity.	This field is automatically filled out for you by Microsoft Intune. However, you must manually correct it if it's incorrect or if you need to add additional domains. For more info about where this area is and what it means, see the Define your enterprise-managed corporate identity section of the policy creation topics.
Specify your network domain names.	Starting with Windows 10, version 1703, this field is optional. Specify the DNS suffixes used in your environment. All traffic to the fully-qualified domains appearing in this list will be protected. For more info about where this area is and how to add your suffixes, see the table that appears in the Choose where apps can access enterprise data section of the policy creation topics.
Specify your enterprise IPv4 or IPv6 ranges.	Starting with Windows 10, version 1703, this field is optional. Specify the addresses for a valid IPv4 or IPv6 value range within your intranet. These addresses, used with your Network domain names, define your corporate network boundaries. For more info about where this area is and what it means, see the table that appears in the Define your enterprise-managed corporate identity section of the policy creation topics.

TASK	DESCRIPTION
Include your Data Recovery Agent (DRA) certificate.	<p>Starting with Windows 10, version 1703, this field is optional. But we strongly recommend that you add a certificate.</p> <p>This certificate makes sure that any of your WIP-encrypted data can be decrypted, even if the security keys are lost. For more info about where this area is and what it means, see the Create and verify an Encrypting File System (EFS) Data Recovery Agent (DRA) certificate topic.</p>

NOTE

Help to make this topic better by providing us with edits, additions, and feedback. For info about how to contribute to this topic, see [Editing Windows IT professional documentation](#).

Testing scenarios for Windows Information Protection (WIP)

7/1/2022 • 6 minutes to read • [Edit Online](#)

Applies to:

- Windows 10, version 1607 and later

We've come up with a list of suggested testing scenarios that you can use to test Windows Information Protection (WIP) in your company.

Testing scenarios

You can try any of the processes included in these scenarios, but you should focus on the ones that you might encounter in your organization.

IMPORTANT

If any of these scenarios does not work, first take note of whether WIP has been revoked. If it has, unenlightened apps will have to be uninstalled and re-installed since their settings files will remain encrypted.

- **Encrypt and decrypt files using File Explorer:**

1. Open File Explorer, right-click a work document, and then click **Work** from the **File Ownership** menu.

Make sure the file is encrypted by right-clicking the file again, clicking **Advanced** from the **General** tab, and then clicking **Details** from the **Compress or Encrypt attributes** area. The file should show up under the heading, **This enterprise domain can remove or revoke access:**

`*<your_enterprise_identity>*`. For example, `contoso.com`.

2. In File Explorer, right-click the same document, and then click **Personal** from the **File Ownership** menu.

Make sure the file is decrypted by right-clicking the file again, clicking **Advanced** from the **General** tab, and then verifying that the **Details** button is unavailable.

- **Create work documents in enterprise-allowed apps:** Start an unenlightened but allowed app, such as a line-of-business app, and then create a new document, saving your changes.

Make sure the document is encrypted to your Enterprise Identity. This might take a few minutes and require you to close and re-open the file.

IMPORTANT

Certain file types like `.exe` and `.dll`, along with certain file paths, such as `%windir%` and `%programfiles%` are excluded from automatic encryption.

For more info about your Enterprise Identity and adding apps to your allowed apps list, see either [Create a Windows Information Protection \(WIP\) policy using Microsoft Intune](#) or [Create a Windows Information Protection \(WIP\) policy using Microsoft Endpoint Configuration Manager](#), based on your deployment

system.

- **Block enterprise data from non-enterprise apps:**

1. Start an app that doesn't appear on your allowed apps list, and then try to open a work-encrypted file.

The app shouldn't be able to access the file.

2. Try double-clicking or tapping on the work-encrypted file. If your default app association is an app not on your allowed apps list, you should get an **Access Denied** error message.

- **Copy and paste from enterprise apps to non-enterprise apps:**

1. Copy (CTRL+C) content from an app on your allowed apps list, and then try to paste (CTRL+V) the content into an app that doesn't appear on your allowed apps list.

You should see a WIP-related warning box, asking you to click either **Change to personal** or **Keep at work**.

2. Click **Keep at work**. The content isn't pasted into the non-enterprise app.

3. Repeat Step 1, but this time click **Change to personal**, and try to paste the content again.

The content is pasted into the non-enterprise app.

4. Try copying and pasting content between apps on your allowed apps list. The content should copy and paste between apps without any warning messages.

- **Drag and drop from enterprise apps to non-enterprise apps:**

1. Drag content from an app on your allowed apps list, and then try to drop the content into an app that doesn't appear on your allowed apps list.

You should see a WIP-related warning box, asking you to click either **Keep at work** or **Change to personal**.

2. Click **Keep at work**. The content isn't dropped into the non-enterprise app.

3. Repeat Step 1, but this time click **Change to personal**, and try to drop the content again.

The content is dropped into the non-enterprise app.

4. Try dragging and dropping content between apps on your allowed apps list. The content should move between the apps without any warning messages.

- **Share between enterprise apps and non-enterprise apps:**

1. Open an app on your allowed apps list, like Microsoft Photos, and try to share content with an app that doesn't appear on your allowed apps list, like Facebook.

You should see a WIP-related warning box, asking you to click either **Keep at work** or **Change to personal**.

2. Click **Keep at work**. The content isn't shared into Facebook.

3. Repeat Step 1, but this time click **Change to personal**, and try to share the content again.

The content is shared into Facebook.

4. Try sharing content between apps on your allowed apps list. The content should share between the apps without any warning messages.

- **Verify that Windows system components can use WIP:**

1. Start Windows Journal and Internet Explorer 11, creating, editing, and saving files in both apps.

Make sure that all of the files you worked with are encrypted to your configured Enterprise Identity. In some cases, you might need to close the file and wait a few moments for it to be automatically encrypted.

2. Open File Explorer and make sure your modified files are appearing with a **Lock** icon.
3. Try copying and pasting, dragging and dropping, and sharing using these apps with other apps that appear both on and off the allowed apps list.

NOTE

Most Windows-signed components like File Explorer (when running in the user's context), should have access to enterprise data.

A few notable exceptions include some of the user-facing in-box apps, like Wordpad, Notepad, and Microsoft Paint. These apps don't have access by default, but can be added to your allowed apps list.

- **Use WIP on NTFS, FAT, and exFAT systems:**

1. Start an app that uses the FAT or exFAT file system (for example a SD card or USB flash drive), and appears on your allowed apps list.
2. Create, edit, write, save, copy, and move files. Basic file and folder operations like copy, move, rename, delete, and so on, should work properly on encrypted files.

- **Verify your shared files can use WIP:**

1. Download a file from a protected file share, making sure the file is encrypted by locating the **Briefcase** icon next to the file name.
2. Open the same file, make a change, save it and then try to upload it back to the file share. Again, this should work without any warnings.
3. Open an app that doesn't appear on your allowed apps list and attempt to access a file on the WIP-enabled file share.

The app shouldn't be able to access the file share.

- **Verify your cloud resources can use WIP:**

1. Add both Internet Explorer 11 and Microsoft Edge to your allowed apps list.
2. Open SharePoint (or another cloud resource that's part of your policy) and access a WIP-enabled resource by using both IE11 and Microsoft Edge.

Both browsers should respect the enterprise and personal boundary.

3. Remove Internet Explorer 11 from your allowed app list and then try to access an intranet site or enterprise-related cloud resource.

IE11 shouldn't be able to access the sites.

NOTE

Any file downloaded from your work SharePoint site, or any other WIP-enabled cloud resource, is automatically marked as **Work**.

- **Verify your Virtual Private Network (VPN) can be auto-triggered:**

1. Set up your VPN network to start based on the **WIPModelID** setting. For specific info, see [Create and deploy a VPN policy for Windows Information Protection \(WIP\) using Microsoft Intune](#).
2. Start an app from your allowed apps list. The VPN network should automatically start.
3. Disconnect from your network and then start an app that isn't on your allowed apps list.

The VPN shouldn't start and the app shouldn't be able to access your enterprise network.

- **Unenroll client devices from WIP:** Unenroll a device from WIP by going to **Settings**, click **Accounts**, click **Work**, click the name of the device you want to unenroll, and then click **Remove**.

The device should be removed and all of the enterprise content for that managed account should be gone.

IMPORTANT

On client devices, the data isn't removed and can be recovered. So, you must make sure the content is marked as **Revoked** and that access is denied for the employee.

NOTE

Help to make this topic better by providing us with edits, additions, and feedback. For info about how to contribute, see [Editing Windows IT professional documentation](#).

Limitations while using Windows Information Protection (WIP)

7/1/2022 • 8 minutes to read • [Edit Online](#)

Applies to:

- Windows 10, version 1607 and later

This following list provides info about the most common problems you might encounter while running Windows Information Protection in your organization.

- **Limitation:** Your enterprise data on USB drives might be tied to the device it was protected on, based on your Azure RMS configuration.

- **How it appears:**

- If you're using Azure RMS: Authenticated users can open enterprise data on USB drives, on computers running Windows 10, version 1703.
- If you're not using Azure RMS: Data in the new location remains encrypted, but becomes inaccessible on other devices and for other users. For example, the file won't open or the file opens, but doesn't contain readable text.

- **Workaround:** Share files with fellow employees through enterprise file servers or enterprise cloud locations. If data must be shared via USB, employees can decrypt protected files, but it will be audited.

We strongly recommend educating employees about how to limit or eliminate the need for this decryption.

- **Limitation:** Direct Access is incompatible with Windows Information Protection.

- **How it appears:** Direct Access might experience problems with how Windows Information Protection enforces app behavior and data movement because of how WIP determines what is and isn't a corporate network resource.

- **Workaround:** We recommend that you use VPN for client access to your intranet resources.

NOTE

VPN is optional and isn't required by Windows Information Protection.

- **Limitation: NetworkIsolation** Group Policy setting takes precedence over MDM Policy settings.

- **How it appears:** The **NetworkIsolation** Group Policy setting can configure network settings that can also be configured by using MDM. WIP relies on these policies being correctly configured.

- **Workaround:** If you use both Group Policy and MDM to configure your **NetworkIsolation** settings, you must make sure that those same settings are deployed to your organization using both Group Policy and MDM.

- **Limitation:** Cortana can potentially allow data leakage if it's on the allowed apps list.

- **How it appears:** If Cortana is on the allowed list, some files might become unexpectedly encrypted after an employee performs a search using Cortana. Your employees will still be able to use Cortana to search and provide results on enterprise documents and locations, but results might be sent to

Microsoft.

- **Workaround:** We don't recommend adding Cortana to your allowed apps list. However, if you wish to use Cortana and don't mind whether the results potentially go to Microsoft, you can make Cortana an Exempt app.
- **Limitation:** Windows Information Protection is designed for use by a single user per device.
 - **How it appears:** A secondary user on a device might experience app compatibility issues when unenlightened apps start to automatically encrypt for all users. Additionally, only the initial, enrolled user's content can be revoked during the unenrollment process.
 - **Workaround:** We recommend only having one user per managed device.
- **Limitation:** Installers copied from an enterprise network file share might not work properly.
 - **How it appears:** An app might fail to properly install because it can't read a necessary configuration or data file, such as a .cab or .xml file needed for installation, which was protected by the copy action.
 - **Workaround:** To fix this, you can:
 - Start the installer directly from the file share.
 - OR
 - Decrypt the locally copied files needed by the installer.
 - OR
 - Mark the file share with the installation media as "personal". To do this, you'll need to set the Enterprise IP ranges as **Authoritative** and then exclude the IP address of the file server, or you'll need to put the file server on the Enterprise Proxy Server list.
- **Limitation:** Changing your primary Corporate Identity isn't supported.
 - **How it appears:** You might experience various instabilities, including but not limited to network and file access failures, and potentially granting incorrect access.
 - **Workaround:** Turn off Windows Information Protection for all devices before changing the primary Corporate Identity (first entry in the list), restarting, and finally redeploying.
- **Limitation:** Redirected folders with Client-Side Caching are not compatible with Windows Information Protection.
 - **How it appears:** Apps might encounter access errors while attempting to read a cached, offline file.
 - **Workaround:** Migrate to use another file synchronization method, such as Work Folders or OneDrive for Business.

NOTE

For more info about Work Folders and Offline Files, see the [Work Folders and Offline Files support for Windows Information Protection blog](#). If you're having trouble opening files offline while using Offline Files and Windows Information Protection, see [Can't open files offline when you use Offline Files and Windows Information Protection](#).

- **Limitation:** An unmanaged device can use Remote Desktop Protocol (RDP) to connect to a WIP-managed device.
 - **How it appears:**
 - Data copied from the WIP-managed device is marked as **Work**.
 - Data copied to the WIP-managed device is not marked as **Work**.

- Local **Work** data copied to the WIP-managed device remains **Work** data.
- **Work** data that is copied between two apps in the same session remains ****** data.
- **Workaround**: Disable RDP to prevent access because there is no way to restrict access to only devices managed by Windows Information Protection. RDP is disabled by default.
- **Limitation**: You can't upload an enterprise file to a personal location using Microsoft Edge or Internet Explorer.
 - **How it appears**: A message appears stating that the content is marked as **Work** and the user isn't given an option to override to **Personal**.
 - **Workaround**: Open File Explorer and change the file ownership to **Personal** before you upload.
- **Limitation**: ActiveX controls should be used with caution.
 - **How it appears**: Webpages that use ActiveX controls can potentially communicate with other outside processes that aren't protected by using Windows Information Protection.
 - **Workaround**: We recommend that you switch to using Microsoft Edge, the more secure and safer browser that prevents the use of ActiveX controls. We also recommend that you limit the usage of Internet Explorer 11 to only those line-of-business apps that require legacy technology.

For more info, see [Out-of-date ActiveX control blocking](#).

- **Limitation**: Resilient File System (ReFS) isn't currently supported with Windows Information Protection.
 - **How it appears**: Trying to save or transfer Windows Information Protection files to ReFS will fail.
 - **Workaround**: Format drive for NTFS, or use a different drive.
- **Limitation**: Windows Information Protection isn't turned on if any of the following folders have the **MakeFolderAvailableOfflineDisabled** option set to **False**:
 - AppDataRoaming
 - Desktop
 - StartMenu
 - Documents
 - Pictures
 - Music
 - Videos
 - Favorites
 - Contacts
 - Downloads
 - Links
 - Searches
 - SavedGames
 - **How it appears**: Windows Information Protection isn't turned on for employees in your organization. Error code 0x807c0008 will result if Windows Information Protection is deployed by using Microsoft Endpoint Configuration Manager.
 - **Workaround**: Don't set the **MakeFolderAvailableOfflineDisabled** option to **False** for any of the specified folders. You can configure this parameter, as described [Disable Offline Files on individual redirected folders](#).

If you currently use redirected folders, we recommend that you migrate to a file synchronization solution that supports Windows Information Protection, such as Work Folders or OneDrive for Business. Additionally, if you apply redirected folders after Windows Information Protection is

already in place, you might be unable to open your files offline.

For more info about these potential access errors, see [Can't open files offline when you use Offline Files and Windows Information Protection](#).

- **Limitation:** Only enlightened apps can be managed without device enrollment
 - **How it appears:** If a user enrolls a device for Mobile Application Management (MAM) without device enrollment, only enlightened apps will be managed. This is by design to prevent personal files from being unintentionally encrypted by unenlightened apps.

Unenlightened apps that need to access work using MAM need to be re-compiled as LOB apps or managed by using MDM with device enrollment.
 - **Workaround:** If all apps need to be managed, enroll the device for MDM.
- **Limitation:** By design, files in the Windows directory (%windir% or C:/Windows) cannot be encrypted because they need to be accessed by any user. If a file in the Windows directory gets encrypted by one user, other users can't access it.
 - **How it appears:** Any attempt to encrypt a file in the Windows directory will return a file access denied error. But if you copy or drag and drop an encrypted file to the Windows directory, it will retain encryption to honor the intent of the owner.
 - **Workaround:** If you need to save an encrypted file in the Windows directory, create and encrypt the file in a different directory and copy it.
- **Limitation:** OneNote notebooks on OneDrive for Business must be properly configured to work with Windows Information Protection.
 - **How it appears:** OneNote might encounter errors syncing a OneDrive for Business notebook and suggest changing the file ownership to Personal. Attempting to view the notebook in OneNote Online in the browser will show an error and unable to view it.
 - **Workaround:** OneNote notebooks that are newly copied into the OneDrive for Business folder from File Explorer should get fixed automatically. To do this, follow these steps:
 1. Close the notebook in OneNote.
 2. Move the notebook folder via File Explorer out of the OneDrive for Business folder to another location, such as the Desktop.
 3. Copy the notebook folder and Paste it back into the OneDrive for Business folder.Wait a few minutes to allow OneDrive to finish syncing & upgrading the notebook, and the folder should automatically convert to an Internet Shortcut. Opening the shortcut will open the notebook in the browser, which can then be opened in the OneNote client by using the "Open in app" button.
- **Limitation:** Microsoft Office Outlook offline data files (PST and OST files) are not marked as **Work** files, and are therefore not protected.
 - **How it appears:** If Microsoft Office Outlook is set to work in cached mode (default setting), or if some emails are stored in a local PST file, the data is unprotected.
 - **Workaround:** It is recommended to use Microsoft Office Outlook in Online mode, or to use encryption to protect OST and PST files manually.

NOTE

- When corporate data is written to disk, Windows Information Protection uses the Windows-provided Encrypting File System (EFS) to protect it and associate it with your enterprise identity. One caveat to keep in mind is that the Preview Pane in File Explorer will not work for encrypted files.
- Help to make this topic better by providing us with edits, additions, and feedback. For info about how to contribute to this topic, see [Contributing to our content](#).

How to collect Windows Information Protection (WIP) audit event logs

7/1/2022 • 6 minutes to read • [Edit Online](#)

Applies to:

- Windows 10, version 1607 and later

Windows Information Protection (WIP) creates audit events in the following situations:

- If an employee changes the File ownership for a file from **Work** to **Personal**.
- If data is marked as **Work**, but shared to a personal app or webpage. For example, through copying and pasting, dragging and dropping, sharing a contact, uploading to a personal webpage, or if the user grants a personal app provides temporary access to a work file.
- If an app has custom audit events.

Collect WIP audit logs by using the Reporting configuration service provider (CSP)

Collect the WIP audit logs from your employee's devices by following the guidance provided by the [Reporting configuration service provider \(CSP\)](#) documentation. This topic provides info about the actual audit events.

NOTE

The **Data** element in the response includes the requested audit logs in an XML-encoded format.

User element and attributes

This table includes all available attributes for the **User** element.

ATTRIBUTE	VALUE TYPE	DESCRIPTION
UserID	String	The security identifier (SID) of the user corresponding to this audit report.
EnterpriseID	String	The enterprise ID corresponding to this audit report.

Log element and attributes

This table includes all available attributes/elements for the **Log** element. The response can contain zero (0) or more **Log** elements.

ATTRIBUTE/ELEMENT	VALUE TYPE	DESCRIPTION
ProviderType	String	This is always EDPAudit .

ATTRIBUTE/ELEMENT	VALUE TYPE	DESCRIPTION
LogType	String	<p>Includes:</p> <ul style="list-style-type: none"> • DataCopied. Work data is copied or shared to a personal location. • ProtectionRemoved. Windows Information Protection is removed from a Work-defined file. • ApplicationGenerated. A custom audit log provided by an app.
TimeStamp	Int	<p>Uses the FILETIME structure to represent the time that the event happened.</p>
Policy	String	<p>How the work data was shared to the personal location:</p> <ul style="list-style-type: none"> • CopyPaste. Work data was pasted into a personal location or app. • ProtectionRemoved. Work data was changed to be unprotected. • DragDrop. Work data was dropped into a personal location or app. • Share. Work data was shared with a personal location or app. • NULL. Any other way work data could be made personal beyond the options above. For example, when a work file is opened using a personal application (also known as, temporary access).
Justification	String	<p>Not implemented. This will always be either blank or NULL.</p> <p>Note Reserved for future use to collect the user justification for changing from Work to Personal.</p>
Object	String	<p>A description of the shared work data. For example, if an employee opens a work file by using a personal app, this would be the file path.</p>

ATTRIBUTE/ELEMENT	VALUE TYPE	DESCRIPTION
DataInfo	String	<p>Any additional info about how the work file changed:</p> <ul style="list-style-type: none"> • A file path. If an employee uploads a work file to a personal website by using Microsoft Edge or Internet Explorer, the file path is included here. • Clipboard data types. If an employee pastes work data into a personal app, the list of clipboard data types provided by the work app are included here. For more info, see the Examples section of this topic.
Action	Int	<p>Provides info about what happened when the work data was shared to personal, including:</p> <ul style="list-style-type: none"> • 1. File decrypt. • 2. Copy to location. • 3. Send to recipient. • 4. Other.
FilePath	String	The file path to the file specified in the audit event. For example, the location of a file that's been decrypted by an employee or uploaded to a personal website.
SourceApplicationName	String	The source app or website. For the source app, this is the AppLocker identity. For the source website, this is the hostname.
SourceName	String	A string provided by the app that's logging the event. It's intended to describe the source of the work data.
DestinationEnterpriseID	String	<p>The enterprise ID value for the app or website where the employee is sharing the data.</p> <p>NULL, Personal, or blank means there's no enterprise ID because the work data was shared to a personal location. Because we don't currently support multiple enrollments, you'll always see one of these values.</p>
DestinationApplicationName	String	The destination app or website. For the destination app, this is the AppLocker identity. For the destination website, this is the hostname.

ATTRIBUTE/ELEMENT	VALUE TYPE	DESCRIPTION
DestinationName	String	A string provided by the app that's logging the event. It's intended to describe the destination of the work data.
Application	String	The AppLocker identity for the app where the audit event happened.

Examples

Here are a few examples of responses from the Reporting CSP.

File ownership on a file is changed from work to personal

```
<SyncML><SyncHdr></SyncHdr><SyncBody><Status><CmdID>1</CmdID><MsgRef>1</MsgRef><CmdRef>0</CmdRef><Cmd>SyncHdr</Cmd>
<Data>200</Data></Status><Status><CmdID>2</CmdID><MsgRef>1</MsgRef><CmdRef>2</CmdRef><Cmd>Replace</Cmd>
<Data>200</Data></Status><Status><CmdID>3</CmdID><MsgRef>1</MsgRef><CmdRef>4</CmdRef><Cmd>Get</Cmd>
<Data>200</Data></Status><Results><CmdID>4</CmdID><MsgRef>1</MsgRef><CmdRef>4</CmdRef><Item><Source>
<LocURI>./Vendor/MSFT/Reporting/EnterpriseDataProtection/RetrieveByTimeRange/Logs</LocURI></Source><Meta>
<Format xmlns="syncml:metinf">xml</Format></Meta><Data><?xml version="1.0" encoding="utf-8"?>
<Reporting Version="com.contoso/2.0/MDM/Reporting">
  <User UserID="S-1-12-1-1111111111-1111111111-1111111111-1111111111" EnterpriseID="corp.contoso.com">
    <Log ProviderType="EDPAudit" LogType="ProtectionRemoved" TimeStamp="131357166318347527">
      <Policy>Protection removed</Policy>
      <Justification>NULL</Justification>
      <FilePath>C:\Users\TestUser\Desktop\tmp\demo\Work document.docx</FilePath>
    </Log>
  </User>
</Reporting></Data></Item></Results><Final/></SyncBody></SyncML>
```

A work file is uploaded to a personal webpage in Edge

```
<SyncML><SyncHdr></SyncHdr><SyncBody><Status><CmdID>1</CmdID><MsgRef>1</MsgRef><CmdRef>0</CmdRef><Cmd>SyncHdr</Cmd>
<Data>200</Data></Status><Status><CmdID>2</CmdID><MsgRef>1</MsgRef><CmdRef>2</CmdRef><Cmd>Replace</Cmd>
<Data>200</Data></Status><Status><CmdID>3</CmdID><MsgRef>1</MsgRef><CmdRef>4</CmdRef><Cmd>Get</Cmd>
<Data>200</Data></Status><Results><CmdID>4</CmdID><MsgRef>1</MsgRef><CmdRef>4</CmdRef><Item><Source>
<LocURI>./Vendor/MSFT/Reporting/EnterpriseDataProtection/RetrieveByTimeRange/Logs</LocURI></Source><Meta>
<Format xmlns="syncml:metinf">xml</Format></Meta><Data><?xml version="1.0" encoding="utf-8"?>
<Reporting Version="com.contoso/2.0/MDM/Reporting">
  <User UserID="S-1-12-1-1111111111-1111111111-1111111111-1111111111" EnterpriseID="corp.contoso.com">
    <Log ProviderType="EDPAudit" LogType="DataCopied" TimeStamp="131357192409318534">
      <Policy>CopyPaste</Policy>
      <Justification>NULL</Justification>
      <SourceApplicationName>NULL</SourceApplicationName>
      <DestinationEnterpriseID>NULL</DestinationEnterpriseID>
      <DestinationApplicationName>mail.contoso.com</DestinationApplicationName>
      <DataInfo>C:\Users\TestUser\Desktop\tmp\demo\Work document.docx</DataInfo>
    </Log>
  </User>
</Reporting></Data></Item></Results><Final/></SyncBody></SyncML>
```

Work data is pasted into a personal webpage

```

<SyncML><SyncHdr></SyncBody><Status><CmdID>1</CmdID><MsgRef>1</MsgRef><CmdRef>0</CmdRef><Cmd>SyncHdr</Cmd>
<Data>200</Data></Status><Status><CmdID>2</CmdID><MsgRef>1</MsgRef><CmdRef>2</CmdRef><Cmd>Replace</Cmd>
<Data>200</Data></Status><Status><CmdID>3</CmdID><MsgRef>1</MsgRef><CmdRef>4</CmdRef><Cmd>Get</Cmd>
<Data>200</Data></Status><Results><CmdID>4</CmdID><MsgRef>1</MsgRef><CmdRef>4</CmdRef><Item><Source>
<LocURI>./Vendor/MSFT/Reporting/EnterpriseDataProtection/RetrieveByTimeRange/Logs</LocURI></Source><Meta>
<Format xmlns="syncml:metinf">xml</Format></Meta><Data><?xml version="1.0" encoding="utf-8"?>
<Reporting Version="com.contoso/2.0/MDM/Reporting">
  <User UserID="S-1-12-1-1111111111-1111111111-1111111111-1111111111" EnterpriseID="corp.contoso.com">
    <Log ProviderType="EDPAudit" LogType="DataCopied" TimeStamp="131357193734179782">
      <Policy>CopyPaste</Policy>
      <Justification>NULL</Justification>
      <SourceApplicationName>O=MICROSOFT CORPORATION, L=REDMOND, S=WASHINGTON, C=US\MICROSOFT OFFICE
2016\WINWORD.EXE\16.0.8027.1000</SourceApplicationName>
      <DestinationEnterpriseID>NULL</DestinationEnterpriseID>
      <DestinationApplicationName>mail.contoso.com</DestinationApplicationName>
      <DataInfo>EnterpriseDataProtectionId|Object Descriptor|Rich Text Format|HTML
Format|AnsiText|Text|EnhancedMetafile|Embed Source|Link Source|Link Source
Descriptor|ObjectLink|Hyperlink</DataInfo>
    </Log>
  </User>
</Reporting></Data></Item></Results><Final/></SyncBody></SyncML>

```

A work file is opened with a personal application

```

<SyncML><SyncHdr></SyncBody><Status><CmdID>1</CmdID><MsgRef>1</MsgRef><CmdRef>0</CmdRef><Cmd>SyncHdr</Cmd>
<Data>200</Data></Status><Status><CmdID>2</CmdID><MsgRef>1</MsgRef><CmdRef>2</CmdRef><Cmd>Replace</Cmd>
<Data>200</Data></Status><Status><CmdID>3</CmdID><MsgRef>1</MsgRef><CmdRef>4</CmdRef><Cmd>Get</Cmd>
<Data>200</Data></Status><Results><CmdID>4</CmdID><MsgRef>1</MsgRef><CmdRef>4</CmdRef><Item><Source>
<LocURI>./Vendor/MSFT/Reporting/EnterpriseDataProtection/RetrieveByTimeRange/Logs</LocURI></Source><Meta>
<Format xmlns="syncml:metinf">xml</Format></Meta><Data><?xml version="1.0" encoding="utf-8"?>
<Reporting Version="com.contoso/2.0/MDM/Reporting">
  <User UserID="S-1-12-1-1111111111-1111111111-1111111111-1111111111" EnterpriseID="corp.contoso.com">
    <Log ProviderType="EDPAudit" LogType="ApplicationGenerated" TimeStamp="131357194991209469">
      <Policy>NULL</Policy>
      <Justification></Justification>
      <Object>C:\Users\TestUser\Desktop\tmp\demo\Work document.docx</Object>
      <Action>1</Action>
      <SourceName>O=MICROSOFT CORPORATION, L=REDMOND, S=WASHINGTON, C=US\MICROSOFT® WINDOWS® OPERATING
SYSTEM\WORDPAD.EXE\10.0.15063.2</SourceName>
      <DestinationEnterpriseID>Personal</DestinationEnterpriseID>
      <DestinationName>O=MICROSOFT CORPORATION, L=REDMOND, S=WASHINGTON, C=US\MICROSOFT® WINDOWS® OPERATING
SYSTEM\WORDPAD.EXE\10.0.15063.2</DestinationName>
      <Application>O=MICROSOFT CORPORATION, L=REDMOND, S=WASHINGTON, C=US\MICROSOFT® WINDOWS® OPERATING
SYSTEM\WORDPAD.EXE\10.0.15063.2</Application>
    </Log>
  </User>
</Reporting></Data></Item></Results><Final/></SyncBody></SyncML>

```

Work data is pasted into a personal application


```

<SyncML><SyncHdr><SyncBody><Status><CmdID>1</CmdID><MsgRef>1</MsgRef><CmdRef>0</CmdRef><Cmd>SyncHdr</Cmd>
<Data>200</Data></Status><Status><CmdID>2</CmdID><MsgRef>1</MsgRef><CmdRef>2</CmdRef><Cmd>Replace</Cmd>
<Data>200</Data></Status><Status><CmdID>3</CmdID><MsgRef>1</MsgRef><CmdRef>4</CmdRef><Cmd>Get</Cmd>
<Data>200</Data></Status><Results><CmdID>4</CmdID><MsgRef>1</MsgRef><CmdRef>4</CmdRef><Item><Source>
<LocURI>./Vendor/MSFT/Reporting/EnterpriseDataProtection/RetrieveByTimeRange/Logs</LocURI></Source><Meta>
<Format xmlns="syncml:metinf">xml</Format></Meta><Data><?xml version="1.0" encoding="utf-8"?>
<Reporting Version="com.contoso/2.0/MDM/Reporting">
  <User UserID="S-1-12-1-1111111111-1111111111-1111111111-1111111111" EnterpriseID="corp.contoso.com">
    <Log ProviderType="EDPAudit" LogType="DataCopied" TimeStamp="131357196076537270">
      <Policy>CopyPaste</Policy>
      <Justification>NULL</Justification>
      <SourceApplicationName>O=MICROSOFT CORPORATION, L=REDMOND, S=WASHINGTON, C=US\MICROSOFT OFFICE
2016\WINWORD.EXE\16.0.8027.1000</SourceApplicationName>
      <DestinationEnterpriseID>NULL</DestinationEnterpriseID>
      <DestinationApplicationName></DestinationApplicationName>
      <DataInfo>EnterpriseDataProtectionId|Object Descriptor|Rich Text Format|HTML
Format|AnsiText|Text|EnhancedMetafile|Embed Source|Link Source|Link Source
Descriptor|ObjectLink|Hyperlink</DataInfo>
    </Log>
  </User>
</Reporting></Data></Item></Results><Final/></SyncBody></SyncML>

```

Collect WIP audit logs by using Windows Event Forwarding (for Windows desktop domain-joined devices only)

Use Windows Event Forwarding to collect and aggregate your Windows Information Protection audit events. You can view your audit events in the Event Viewer.

To view the WIP events in the Event Viewer

1. Open Event Viewer.
2. In the console tree under **Application and Services Logs\Microsoft\Windows**, click **EDP-Audit-Regular** and **EDP-Audit-TCB**.

Collect WIP audit logs using Azure Monitor

You can collect audit logs using Azure Monitor. See [Windows event log data sources in Azure Monitor](#).

To view the WIP events in Azure Monitor

1. Use an existing or create a new Log Analytics workspace.
2. In **Log Analytics > Advanced Settings**, select **Data**. In Windows Event Logs, add logs to receive:

```

Microsoft-Windows-EDP-Application-Learning/Admin
Microsoft-Windows-EDP-Audit-TCB/Admin

```

NOTE

If using Windows Events Logs, the event log names can be found under Properties of the event in the Events folder (Application and Services Logs\Microsoft\Windows, click EDP-Audit-Regular and EDP-Audit-TCB).

3. Download Microsoft [Monitoring Agent](#).
4. To get MSI for Intune installation as stated in the Azure Monitor article, extract: `MMASetup-.exe /c /t:`

Install Microsoft Monitoring Agent to WIP devices using Workspace ID and Primary key. [More](#)

information on Workspace ID and Primary key can be found in **Log Analytics > Advanced Settings**.

5. To deploy MSI via Intune, in installation parameters add:

```
/q /norestart NOAPM=1 ADD_OPINSIGHTS_WORKSPACE=1 OPINSIGHTS_WORKSPACE_AZURE_CLOUD_TYPE=0  
OPINSIGHTS_WORKSPACE_ID=<WORKSPACE_ID> OPINSIGHTS_WORKSPACE_KEY=<WORKSPACE_KEY>  
AcceptEndUserLicenseAgreement=1
```

NOTE

Replace <WORKSPACE_ID> & <WORKSPACE_KEY> received from step 5. In installation parameters, don't place <WORKSPACE_ID> & <WORKSPACE_KEY> in quotes (" or ").

6. After the agent is deployed, data will be received within approximately 10 minutes.

7. To search for logs, go to **Log Analytics workspace > Logs**, and type **Event** in search.

Example

```
Event | where EventLog == "Microsoft-Windows-EDP-Audit-TCB/Admin"
```

Additional resources

- [How to deploy app via Intune](#)
- [How to create Log workspace](#)
- [How to use Microsoft Monitoring Agents for Windows](#)

General guidance and best practices for Windows Information Protection (WIP)

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to:

- Windows 10, version 1607 and later

This section includes info about the enlightened Microsoft apps, including how to add them to your allowed apps list in Microsoft Intune. It also includes some testing scenarios that we recommend running through with Windows Information Protection (WIP).

In this section

TOPIC	DESCRIPTION
Enlightened apps for use with Windows Information Protection (WIP)	Learn the difference between enlightened and unenlightened apps, and then review the list of enlightened apps provided by Microsoft along with the text you will need to use to add them to your allowed apps list.
Unenlightened and enlightened app behavior while using Windows Information Protection (WIP)	Learn the difference between enlightened and unenlightened app behaviors.
Recommended Enterprise Cloud Resources and Neutral Resources network settings with Windows Information Protection (WIP)	Recommended additions for the Enterprise Cloud Resources and Neutral Resources network settings, when used with Windows Information Protection (WIP).
Using Outlook on the web with Windows Information Protection (WIP)	Options for using Outlook on the web with Windows Information Protection (WIP).

NOTE

Help to make this topic better by providing us with edits, additions, and feedback. For info about how to contribute to this topic, see [Editing Windows IT professional documentation](#).

List of enlightened Microsoft apps for use with Windows Information Protection (WIP)

7/1/2022 • 4 minutes to read • [Edit Online](#)

Applies to:

- Windows 10, version 1607 and later

Learn the difference between enlightened and unenlightened apps, and then review the list of enlightened apps provided by Microsoft along with the text you will need to use to add them to your allowed apps list.

Enlightened versus unenlightened apps

Apps can be enlightened or unenlightened:

- **Enlightened apps** can differentiate between corporate and personal data, correctly determining which to protect, based on your policies.
- **Unenlightened apps** consider all data corporate and encrypt everything. Typically, you can tell an unenlightened app because:
 - Windows Desktop shows it as always running in enterprise mode.
 - Windows **Save As** experiences only allow you to save your files as enterprise.
- **Windows Information Protection-work only apps** are unenlightened line-of-business apps that have been tested and deemed safe for use in an enterprise with WIP and Mobile App Management (MAM) solutions without device enrollment. Unenlightened apps that are targeted by WIP without enrollment run under personal mode.

List of enlightened Microsoft apps

Microsoft has made a concerted effort to enlighten several of our more popular apps, including the following:

- Microsoft 3D Viewer
- Microsoft Edge
- Internet Explorer 11
- Microsoft People
- Mobile Office apps, including Word, Excel, PowerPoint, OneNote, and Outlook Mail and Calendar
- Microsoft 365 Apps for enterprise apps, including Word, Excel, PowerPoint, OneNote, and Outlook
- OneDrive app
- OneDrive sync client (OneDrive.exe, the next generation sync client)
- Microsoft Photos
- Groove Music
- Notepad

- Microsoft Paint
- Microsoft Movies & TV
- Microsoft Messaging
- Microsoft Remote Desktop
- Microsoft To Do

NOTE

Microsoft Visio, Microsoft Office Access, Microsoft Project, and Microsoft Publisher are not enlightened apps and need to be exempted from Windows Information Protection policy. If they are allowed, there is a risk of data loss. For example, if a device is workplace-joined and managed and the user leaves the company, metadata files that the apps rely on remain encrypted and the apps stop functioning.

List of WIP-work only apps from Microsoft

Microsoft still has apps that are unenlightened, but which have been tested and deemed safe for use in an enterprise with Windows Information Protection and MAM solutions.

- Skype for Business
- Microsoft Teams (build 1.3.00.12058 and later)

Adding enlightened Microsoft apps to the allowed apps list

NOTE

As of January 2019 it is no longer necessary to add Intune Company Portal as an exempt app since it is now included in the default list of protected apps.

You can add any or all of the enlightened Microsoft apps to your allowed apps list. Included here is the **Publisher name, Product or File name, and App Type** info for both Microsoft Intune and Microsoft Endpoint Configuration Manager.

PRODUCT NAME	APP INFO
Microsoft 3D Viewer	<p>Publisher: CN=Microsoft Corporation, O=Microsoft Corporation, L=Redmond, S=Washington, C=US</p> <p>Product Name: Microsoft.Microsoft3DViewer</p> <p>App Type: Universal app</p>
Microsoft Edge	<p>Publisher: CN=Microsoft Corporation, O=Microsoft Corporation, L=Redmond, S=Washington, C=US</p> <p>Product Name: Microsoft.MicrosoftEdge</p> <p>App Type: Universal app</p>
Microsoft People	<p>Publisher: CN=Microsoft Corporation, O=Microsoft Corporation, L=Redmond, S=Washington, C=US</p> <p>Product Name: Microsoft.People</p> <p>App Type: Universal app</p>

PRODUCT NAME	APP INFO
Word Mobile	<p>Publisher: CN=Microsoft Corporation, O=Microsoft Corporation, L=Redmond, S=Washington, C=US</p> <p>Product Name: Microsoft.Office.Word</p> <p>App Type: Universal app</p>
Excel Mobile	<p>Publisher: CN=Microsoft Corporation, O=Microsoft Corporation, L=Redmond, S=Washington, C=US</p> <p>Product Name: Microsoft.Office.Excel</p> <p>App Type: Universal app</p>
PowerPoint Mobile	<p>Publisher: CN=Microsoft Corporation, O=Microsoft Corporation, L=Redmond, S=Washington, C=US</p> <p>Product Name: Microsoft.Office.PowerPoint</p> <p>App Type: Universal app</p>
OneNote	<p>Publisher: CN=Microsoft Corporation, O=Microsoft Corporation, L=Redmond, S=Washington, C=US</p> <p>Product Name: Microsoft.Office.OneNote</p> <p>App Type: Universal app</p>
Outlook Mail and Calendar	<p>Publisher: CN=Microsoft Corporation, O=Microsoft Corporation, L=Redmond, S=Washington, C=US</p> <p>Product Name: microsoft.windowscommunicationsapps</p> <p>App Type: Universal app</p>
Microsoft 365 Apps for enterprise and Office 2019 Professional Plus	<p>Microsoft 365 Apps for enterprise and Office 2019 Professional Plus apps are set up as a suite. You must use the O365 ProPlus - Allow and Exempt AppLocker policy files (zip files) to turn the suite on for Windows Information Protection.</p> <p>We don't recommend setting up Office by using individual paths or publisher rules.</p>
Microsoft Photos	<p>Publisher: CN=Microsoft Corporation, O=Microsoft Corporation, L=Redmond, S=Washington, C=US</p> <p>Product Name: Microsoft.Windows.Photos</p> <p>App Type: Universal app</p>
Groove Music	<p>Publisher: CN=Microsoft Corporation, O=Microsoft Corporation, L=Redmond, S=Washington, C=US</p> <p>Product Name: Microsoft.ZuneMusic</p> <p>App Type: Universal app</p>
Microsoft Movies & TV	<p>Publisher: CN=Microsoft Corporation, O=Microsoft Corporation, L=Redmond, S=Washington, C=US</p> <p>Product Name: Microsoft.ZuneVideo</p> <p>App Type: Universal app</p>

PRODUCT NAME	APP INFO
Microsoft Messaging	Publisher: CN=Microsoft Corporation, O=Microsoft Corporation, L=Redmond, S=Washington, C=US Product Name: Microsoft.Messaging App Type: Universal app
IE11	Publisher: O=Microsoft Corporation, L=Redmond, S=Washington, C=US Binary Name: iexplore.exe App Type: Desktop app
OneDrive Sync Client	Publisher: O=Microsoft Corporation, L=Redmond, S=Washington, C=US Binary Name: onedrive.exe App Type: Desktop app
OneDrive app	Publisher: CN=Microsoft Corporation, O=Microsoft Corporation, L=Redmond, S=Washington, C=US Product Name: Microsoft.Microsoftskydrive Product Version: Product version: 17.21.0.0 (and later) App Type: Universal app
Notepad	Publisher: O=Microsoft Corporation, L=Redmond, S=Washington, C=US Binary Name: notepad.exe App Type: Desktop app
Microsoft Paint	Publisher: O=Microsoft Corporation, L=Redmond, S=Washington, C=US Binary Name: mspaint.exe App Type: Desktop app
Microsoft Remote Desktop	Publisher: O=Microsoft Corporation, L=Redmond, S=Washington, C=US Binary Name: mstsc.exe App Type: Desktop app
Microsoft MAPI Repair Tool	Publisher: O=Microsoft Corporation, L=Redmond, S=Washington, C=US Binary Name: fixmapi.exe App Type: Desktop app
Microsoft To Do	Publisher: O=Microsoft Corporation, L=Redmond, S=Washington, C=US Product Name: Microsoft.Todos App Type: Store app

NOTE

Help to make this topic better by providing us with edits, additions, and feedback. For info about how to contribute to this topic, see [Editing Windows IT professional documentation](#).

Unenlightened and enlightened app behavior while using Windows Information Protection (WIP)

7/1/2022 • 3 minutes to read • [Edit Online](#)

Applies to:

- Windows 10, version 1607 and later

Windows Information Protection (WIP) classifies apps into two categories: enlightened and unenlightened. Enlightened apps can differentiate between corporate and personal data, correctly determining which to protect based on internal policies. Corporate data is encrypted on the managed device and attempts to copy/paste or share this information with non-corporate apps or people will fail. Unenlightened apps, when marked as corporate-managed, consider all data corporate and encrypt everything by default.

To avoid the automatic encryption of data, developers can enlighten apps by adding and compiling code using the Windows Information Protection application programming interfaces. The most likely candidates for enlightenment are apps that:

- Don't use common controls for saving files.
- Don't use common controls for text boxes.
- Simultaneously work on personal and corporate data (for example, contact apps that display personal and corporate data in a single view or a browser that displays personal and corporate web pages on tabs within a single instance).

We strongly suggest that the only unenlightened apps you add to your allowed apps list are Line-of-Business (LOB) apps.

IMPORTANT

After revoking WIP, unenlightened apps will have to be uninstalled and re-installed since their settings files will remain encrypted. For more info about creating enlightened apps, see the [Windows Information Protection \(WIP\)](#) topic in the Windows Dev Center.

Unenlightened app behavior

This table includes info about how unenlightened apps might behave, based on your Windows Information Protection (WIP) networking policies, your app configuration, and potentially whether the app connects to network resources directly by using IP addresses or by using hostnames.

APP RULE SETTING	NETWORKING POLICY CONFIGURATION
------------------	---------------------------------

APP RULE SETTING	NETWORKING POLICY CONFIGURATION
<p>Not required. App connects to enterprise cloud resources directly, using an IP address.</p>	<p>Name-based policies, without the <code>/*AppCompat*/</code> string:</p> <ul style="list-style-type: none"> • App is entirely blocked from both personal and enterprise cloud resources. • No encryption is applied. • App can't access local Work files. <p>Name-based policies, using the <code>/*AppCompat*/</code> string or proxy-based policies:</p> <ul style="list-style-type: none"> • App can access both personal and enterprise cloud resources. However, you might encounter apps using policies that restrict access to enterprise cloud resources. • No encryption is applied. • App can't access local Work files.
<p>Not required. App connects to enterprise cloud resources, using a hostname.</p>	<ul style="list-style-type: none"> • App is blocked from accessing enterprise cloud resources, but can access other network resources. • No encryption is applied. • App can't access local Work files.
<p>Allow. App connects to enterprise cloud resources, using an IP address or a hostname.</p>	<ul style="list-style-type: none"> • App can access both personal and enterprise cloud resources. • Auto-encryption is applied. • App can access local Work files.
<p>Exempt. App connects to enterprise cloud resources, using an IP address or a hostname.</p>	<ul style="list-style-type: none"> • App can access both personal and enterprise cloud resources. • No encryption is applied. • App can access local Work files.

Enlightened app behavior

This table includes info about how enlightened apps might behave, based on your Windows Information Protection (WIP) networking policies, your app configuration, and potentially whether the app connects to network resources directly by using IP addresses or by using hostnames.

APP RULE SETTING	NETWORKING POLICY CONFIGURATION FOR NAME-BASED POLICIES, POSSIBLY USING THE /*APPCOMPAT*/ STRING, OR PROXY-BASED POLICIES
<p>Not required. App connects to enterprise cloud resources, using an IP address or a hostname.</p>	<ul style="list-style-type: none"> • App is blocked from accessing enterprise cloud resources, but can access other network resources. • No encryption is applied. • App can't access local Work files.
<p>Allow. App connects to enterprise cloud resources, using an IP address or a hostname.</p>	<ul style="list-style-type: none"> • App can access both personal and enterprise cloud resources. • App protects work data and leaves personal data unprotected. • App can access local Work files.
<p>Exempt. App connects to enterprise cloud resources, using an IP address or a hostname.</p>	<ul style="list-style-type: none"> • App can access both personal and enterprise cloud resources. • App protects work data and leaves personal data unprotected. • App can access local Work files.

NOTE

Help to make this topic better by providing us with edits, additions, and feedback. For info about how to contribute to this topic, see [Editing Windows IT professional documentation](#).

Recommended Enterprise Cloud Resources and Neutral Resources network settings with Windows Information Protection (WIP)

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to:

- Windows 10, version 1607 and later

Learn more about what features and functionality are supported in each Windows edition at [Compare Windows 10 Editions](#).

We recommend that you add the following URLs to the Enterprise Cloud Resources and Neutral Resources network settings when you create a Windows Information Protection policy. If you are using Intune, the SharePoint entries may be added automatically.

Recommended Enterprise Cloud Resources

This table includes the recommended URLs to add to your Enterprise Cloud Resources network setting, based on the apps you use in your organization.

IF YOUR ORGANIZATION USES...	ADD THESE ENTRIES TO YOUR ENTERPRISE CLOUD RESOURCES NETWORK SETTING (REPLACE "CONTOSO" WITH YOUR DOMAIN NAME(S))
Sharepoint Online	<ul style="list-style-type: none">- <code>contoso.sharepoint.com</code>- <code>contoso-my.sharepoint.com</code>- <code>contoso-files.sharepoint.com</code>
Yammer	<ul style="list-style-type: none">- <code>www.yammer.com</code>- <code>yammer.com</code>- <code>persona.yammer.com</code>
Outlook Web Access (OWA)	<ul style="list-style-type: none">- <code>outlook.office.com</code>- <code>outlook.office365.com</code>- <code>attachments.office.net</code>
Microsoft Dynamics	<code>contoso.crm.dynamics.com</code>
Visual Studio Online	<code>contoso.visualstudio.com</code>
Power BI	<code>contoso.powerbi.com</code>
Microsoft Teams	<code>teams.microsoft.com</code>

IF YOUR ORGANIZATION USES...	ADD THESE ENTRIES TO YOUR ENTERPRISE CLOUD RESOURCES NETWORK SETTING (REPLACE "CONTOSO" WITH YOUR DOMAIN NAME(S))
Other Office 365 services	<ul style="list-style-type: none">- tasks.office.com- protection.office.com- meet.lync.com- project.microsoft.com

You can add other work-only apps to the Cloud Resource list, or you can create a packaged app rule for the .exe file to protect every file the app creates or modifies. Depending on how the app is accessed, you might want to add both.

For Office 365 endpoints, see [Office 365 URLs and IP address ranges](#). Office 365 endpoints are updated monthly. Allow the domains listed in section number 46 Allow Required and add also add the apps. Note that apps from officeapps.live.com can also store personal data.

When multiple files are selected from SharePoint Online or OneDrive, the files are aggregated and the URL can change. In this case, add a entry for a second-level domain and use a wildcard such as .svc.ms.

Recommended Neutral Resources

We recommended adding these URLs if you use the Neutral Resources network setting with Windows Information Protection (WIP).

- login.microsoftonline.com
- login.windows.net

Using Outlook on the web with Windows Information Protection (WIP)

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to:

- Windows 10, version 1607 and later

Learn more about what features and functionality are supported in each Windows edition at [Compare Windows 10 Editions](#).

Because Outlook on the web can be used both personally and as part of your organization, you have the following options to configure it with Windows Information Protection (WIP):

OPTION	OUTLOOK ON THE WEB BEHAVIOR
Disable Outlook on the web. Employees can only use Microsoft Outlook 2016 or the Mail for Windows 10 app.	Disabled.
Don't configure outlook.office.com in any of your networking settings.	All mailboxes are automatically marked as personal. This means employees attempting to copy work content into Outlook on the web receive prompts and that files downloaded from Outlook on the web aren't automatically protected as corporate data.
Add outlook.office.com and outlook.office365.com to the Cloud resources network element in your WIP policy.	All mailboxes are automatically marked as corporate. This means any personal inboxes hosted on Office 365 are also automatically marked as corporate data.

NOTE

These limitations don't apply to Outlook 2016, the Mail for Windows 10 app, or the Calendar for Windows 10 app. These apps will work properly, marking an employee's mailbox as corporate data, regardless of how you've configured outlook.office.com in your network settings.

Fine-tune Windows Information Protection (WIP) with WIP Learning

7/1/2022 • 3 minutes to read • [Edit Online](#)

Applies to:

- Windows 10, version 1703 and later

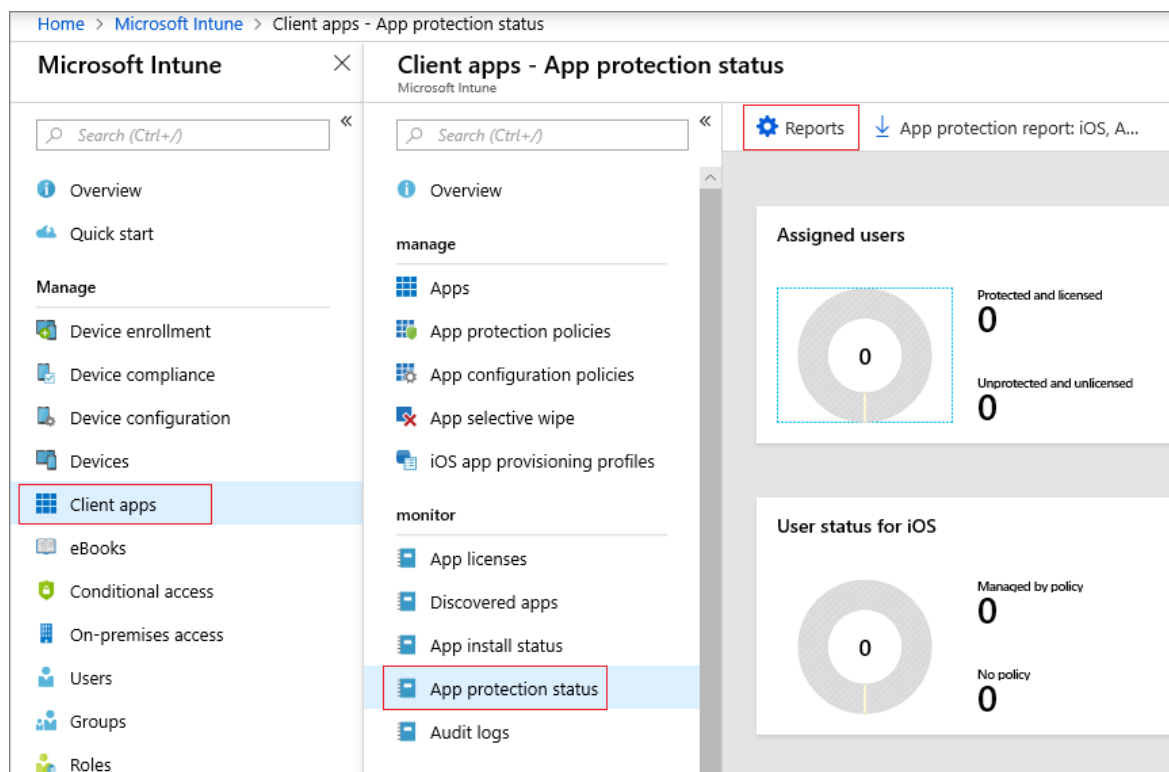
With WIP Learning, you can intelligently tune which apps and websites are included in your WIP policy to help reduce disruptive prompts and keep it accurate and relevant. WIP Learning generates two reports: The **App learning report** and the **Website learning report**. Both reports can be accessed from Microsoft Azure Intune.

The **App learning report** monitors your apps, not in policy, that attempt to access work data. You can identify these apps using the report and add them to your WIP policies to avoid productivity disruption before fully enforcing WIP with “Block” mode. Frequent monitoring of the report will help you continuously identify access attempts so you can update your policy accordingly.

In the **Website learning report**, you can view a summary of the devices that have shared work data with websites. You can use this information to determine which websites should be added to group and user WIP policies. The summary shows which website URLs are accessed by WIP-enabled apps so you can decide which ones are cloud or personal, and add them to the resource list.

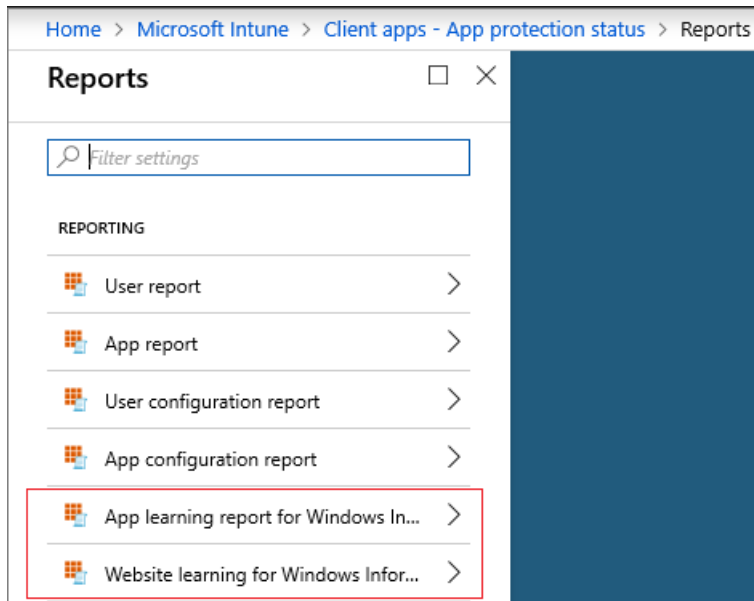
Access the WIP Learning reports

1. Sign in to the [Microsoft Endpoint Manager admin center](#).
2. Select **Apps > Monitor > App protection status > Reports**.



3. Select either **App learning report for Windows Information Protection** or **Website learning**

report for Windows Information Protection.



Once you have the apps and websites showing up in the WIP Learning logging reports, you can decide whether to add them to your app protection policies.

Use the WIP section of Device Health

You can use Device Health to adjust your WIP protection policy. See [Using Device Health](#) to learn more.

If you want to configure your environment for Windows Analytics: Device Health, see [Get Started with Device Health](#) for more information.

Once you have WIP policies in place, by using the WIP section of Device Health, you can:

- Reduce disruptive prompts by adding rules to allow data sharing from approved apps.
- Tune WIP rules by confirming that certain apps are allowed or denied by current policy.

Use Device Health and Intune to adjust WIP protection policy

The information needed for the following steps can be found using Device Health, which you will first have to set up. Learn more about how you can [Monitor the health of devices with Device Health](#).

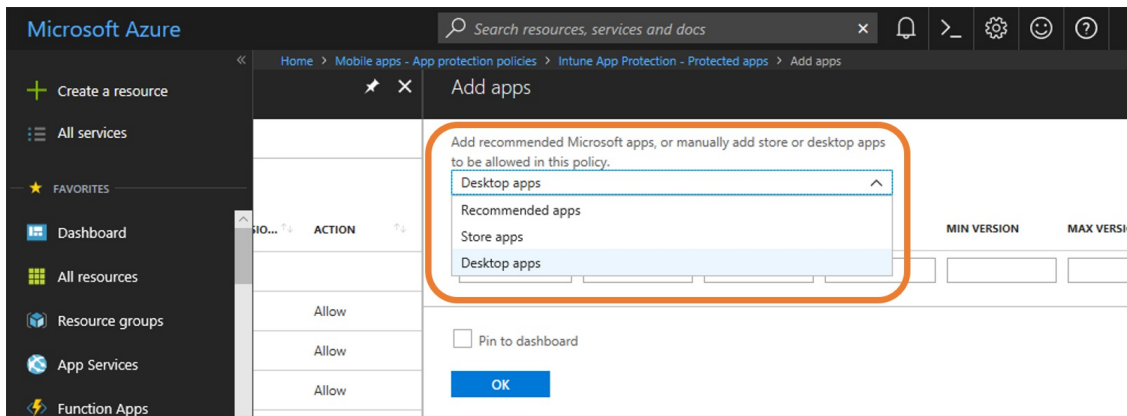
1. In **Device Health** click the app you want to add to your policy and copy the **WipAppId**.

For example, if the app is Google Chrome, the WipAppId is:

```
O=GOOGLE LLC, L=MOUNTAIN VIEW, S=CA, C=US\GOOGLE CHROME\CHROME.EXE\74.0.3729.108
```

In the steps below, you separate the WipAppId by back slashes into the **PUBLISHER**, **PRODUCT NAME**, and **FILE** fields.

2. In Intune, click **App protection policies** and then choose the app policy you want to add an application to.
3. Click **Protected apps**, and then click **Add Apps**.
4. In the **Recommended apps** drop down menu, choose either **Store apps** or **Desktop apps**, depending on the app you've chosen (for example, an executable (EXE) is a desktop app).



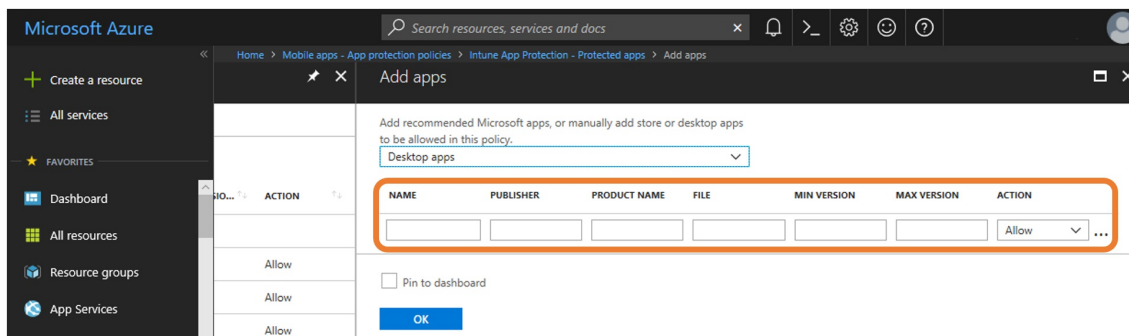
- In **NAME** (optional), type the name of the app, and then in **PUBLISHER** (required), paste the publisher information that you copied in step 1 above.

For example, if the WipAppld is

```
O=GOOGLE LLC, L=MOUNTAIN VIEW, S=CA, C=US\GOOGLE CHROME\CHROME.EXE\74.0.3729.108
```

the text before the first back slash is the publisher:

```
O=GOOGLE LLC, L=MOUNTAIN VIEW, S=CA, C=US
```



- Type the name of the product in **PRODUCT NAME** (required) (this will probably be the same as what you typed for **NAME**).

For example, if the WipAppld is

```
O=GOOGLE LLC, L=MOUNTAIN VIEW, S=CA, C=US\GOOGLE CHROME\CHROME.EXE\74.0.3729.108
```

the text between the first and second back slashes is the product name:

```
GOOGLE CHROME
```

- Copy the name of the executable (for example, snippingtool.exe) and paste it in **FILE** (required).

For example, if the WipAppld is

```
O=GOOGLE LLC, L=MOUNTAIN VIEW, S=CA, C=US\GOOGLE CHROME\CHROME.EXE\74.0.3729.108
```

the text between the second and third back slashes is the file:

```
CHROME.EXE
```

- Type the version number of the app into **MIN VERSION** in Intune (alternately, you can specify the max version, but one or the other is required), and then select the **ACTION: Allow** or **Deny**

When working with WIP-enabled apps and WIP-unknown apps, it is recommended that you start with **Silent** or **Allow overrides** while verifying with a small group that you have the right apps on your allowed apps list.

After you're done, you can change to your final enforcement policy, **Block**. For more information about WIP modes, see: [Protect enterprise data using WIP: WIP-modes](#)

NOTE

Help to make this topic better by providing us with edits, additions, and feedback. For info about how to contribute to this topic, see [Editing Windows IT professional documentation](#).

Windows application security

7/1/2022 • 2 minutes to read • [Edit Online](#)

Cyber-criminals regularly gain access to valuable data by hacking applications. This can include “code injection” attacks, in which attackers insert malicious code that can tamper with data, or even destroy it. An application may have its security misconfigured, leaving open doors for hackers. Or vital customer and corporate information may leave sensitive data exposed. Windows protects your valuable data with layers of application security.

The following table summarizes the Windows security features and capabilities for apps:

SECURITY MEASURES	FEATURES & CAPABILITIES
Windows Defender Application Control	Application control is one of the most effective security controls to prevent unwanted or malicious code from running. It moves away from an application trust model where all code is assumed trustworthy to one where apps must earn trust to run. Learn more: Application Control for Windows
Microsoft Defender Application Guard	Application Guard uses chip-based hardware isolation to isolate untrusted websites and untrusted Office files, seamlessly running untrusted websites and files in an isolated Hyper-V-based container, separate from the desktop operating system, and making sure that anything that happens within the container remains isolated from the desktop. Learn more Microsoft Defender Application Guard overview .
Windows Sandbox	Windows Sandbox provides a lightweight desktop environment to safely run applications in isolation. Software installed inside the Windows Sandbox environment remains “sandboxed” and runs separately from the host machine. A sandbox is temporary. When it’s closed, all the software and files and the state are deleted. You get a brand-new instance of the sandbox every time you open the application. Learn more: Windows Sandbox
Email Security	With Windows S/MIME email security, users can encrypt outgoing messages and attachments, so only intended recipients with digital identification (ID)—also called a certificate—can read them. Users can digitally sign a message, which verifies the identity of the sender and ensures the message has not been tampered with. Configure S/MIME for Windows 10
Microsoft Defender SmartScreen	Microsoft Defender SmartScreen protects against phishing or malware websites and applications, and the downloading of potentially malicious files. Learn more: Microsoft Defender SmartScreen overview

Windows Defender Application Control and virtualization-based protection of code integrity

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows Server 2016

Windows 10 includes a set of hardware and OS technologies that, when configured together, allow enterprises to "lock down" Windows 10 systems so they behave more like mobile devices. In this configuration, Windows Defender Application Control (WDAC) is used to restrict devices to run only approved apps, while the OS is hardened against kernel memory attacks using hypervisor-protected code integrity (HVCI).

WDAC policies and HVCI are powerful protections that can be used separately. However, when these two technologies are configured to work together, they present a strong protection capability for Windows 10 devices.

Using Windows Defender Application Control to restrict devices to only authorized apps has these advantages over other solutions:

1. WDAC policy is enforced by the Windows kernel itself, and the policy takes effect early in the boot sequence before nearly all other OS code and before traditional antivirus solutions run.
2. WDAC lets you set application control policy for code that runs in user mode, kernel mode hardware and software drivers, and even code that runs as part of Windows.
3. Customers can protect the WDAC policy even from local administrator tampering by digitally signing the policy. To change signed policy requires both administrative privilege and access to the organization's digital signing process. This makes it difficult for an attacker, including one who has managed to gain administrative privilege, to tamper with WDAC policy.
4. You can protect the entire WDAC enforcement mechanism with HVCI. Even if a vulnerability exists in kernel mode code, HVCI greatly reduces the likelihood that an attacker could successfully exploit it. This is important because an attacker that compromises the kernel could normally disable most system defenses, including those enforced by WDAC or any other application control solution.

Why we no longer use the Device Guard brand

When we originally promoted Device Guard, we did so with a specific security promise in mind. Although there were no direct dependencies between WDAC and HVCI, we intentionally focused our discussion around the lockdown state achieved when using them together. However, since HVCI relies on Windows virtualization-based security, it has hardware, firmware, and kernel driver compatibility requirements that some older systems can't meet. This misled many people to assume that if systems couldn't use HVCI, they couldn't use WDAC either.

WDAC has no specific hardware or software requirements other than running Windows 10, which means customers were denied the benefits of this powerful application control capability due to Device Guard confusion.

Since the initial release of Windows 10, the world has witnessed numerous hacking and malware attacks where application control alone could have prevented the attack altogether. With this in mind, we now discuss and document Windows Defender Application Control as an independent technology within our security stack and

gave it a name of its own: [Windows Defender Application Control](#). We hope this change will help us better communicate options for adopting application control within your organizations.

Related articles

- [Windows Defender Application Control](#)
- [Dropping the Hammer Down on Malware Threats with Windows 10's Windows Defender](#)
- [Driver compatibility with Windows Defender in Windows 10](#)
- [Code integrity](#)

Application Control for Windows

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to:

- Windows 10
- Windows 11
- Windows Server 2016 and above

NOTE

Some capabilities of Windows Defender Application Control are only available on specific Windows versions. Learn more about the [Windows Defender Application Control feature availability](#).

With thousands of new malicious files created every day, using traditional methods like antivirus solutions—signature-based detection to fight against malware—provides an inadequate defense against new attacks.

In most organizations, information is the most valuable asset, and ensuring that only approved users have access to that information is imperative. However, when a user runs a process, that process has the same level of access to data that the user has. As a result, sensitive information could easily be deleted or transmitted out of the organization if a user knowingly or unknowingly runs malicious software.

Application control can help mitigate these types of security threats by restricting the applications that users are allowed to run and the code that runs in the System Core (kernel). Application control policies can also block unsigned scripts and MSIs, and restrict Windows PowerShell to run in [Constrained Language Mode](#).

Application control is a crucial line of defense for protecting enterprises given today's threat landscape, and it has an inherent advantage over traditional antivirus solutions. Specifically, application control moves away from an application trust model where all applications are assumed trustworthy to one where applications must earn trust in order to run. Many organizations, like the Australian Signals Directorate, understand this and frequently cite application control as one of the most effective means for addressing the threat of executable file-based malware (.exe, .dll, etc.).

NOTE

Although application control can significantly harden your computers against malicious code, we recommend that you continue to maintain an enterprise antivirus solution for a well-rounded enterprise security portfolio.

Windows 10 and Windows 11 include two technologies that can be used for application control depending on your organization's specific scenarios and requirements:

- **Windows Defender Application Control (WDAC)**; and
- **AppLocker**

In this section

ARTICLE	DESCRIPTION
---------	-------------

ARTICLE	DESCRIPTION
WDAC and AppLocker Overview	This article describes the decisions you need to make to establish the processes for managing and maintaining WDAC policies.
WDAC and AppLocker Feature Availability	This article lists the design questions, possible answers, and ramifications of the decisions when you plan a deployment of application control policies.

Related articles

- [WDAC design guide](#)
- [WDAC deployment guide](#)
- [AppLocker overview](#)

Microsoft Defender Application Guard overview

7/1/2022 • 3 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11

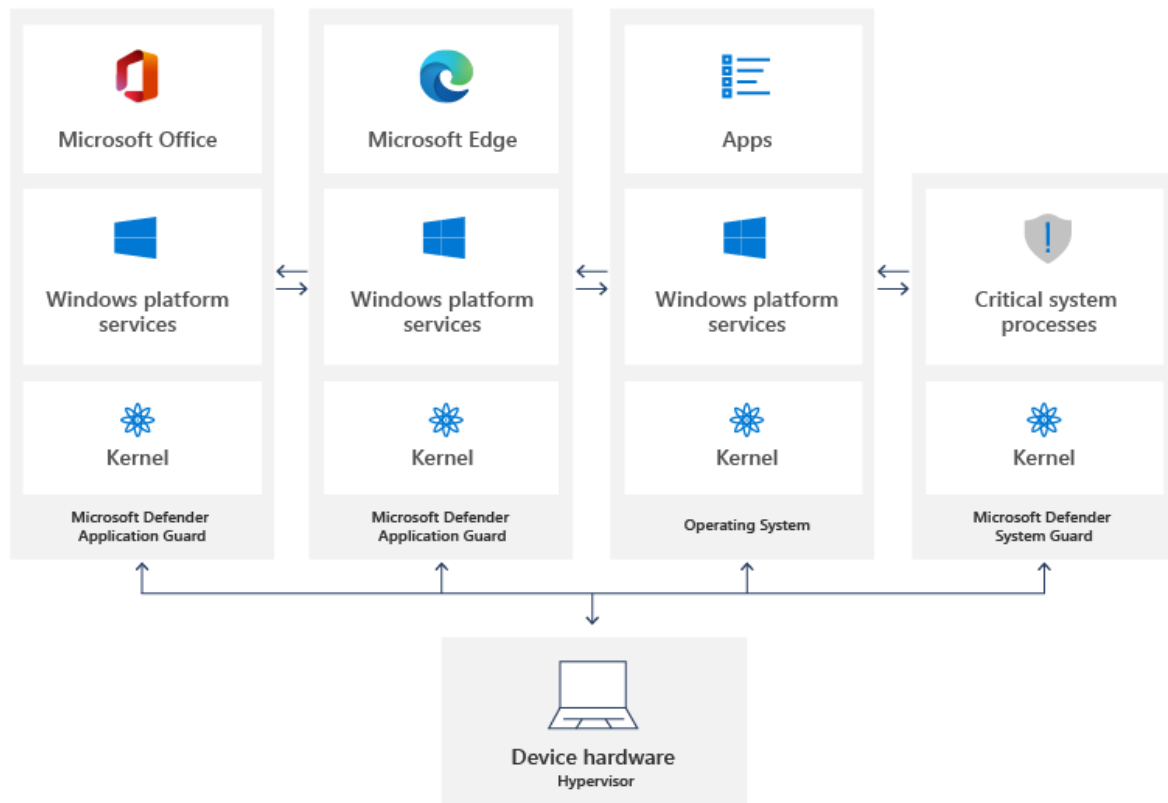
Microsoft Defender Application Guard (Application Guard) is designed to help prevent old and newly emerging attacks to help keep employees productive. Using our unique hardware isolation approach, our goal is to destroy the playbook that attackers use by making current attack methods obsolete.

What is Application Guard and how does it work?

For Microsoft Edge, Application Guard helps to isolate enterprise-defined untrusted sites, protecting your company while your employees browse the Internet. As an enterprise administrator, you define what is among trusted web sites, cloud resources, and internal networks. Everything not on your list is considered untrusted. If an employee goes to an untrusted site through either Microsoft Edge or Internet Explorer, Microsoft Edge opens the site in an isolated Hyper-V-enabled container.

For Microsoft Office, Application Guard helps prevent untrusted Word, PowerPoint and Excel files from accessing trusted resources. Application Guard opens untrusted files in an isolated Hyper-V-enabled container. The isolated Hyper-V container is separate from the host operating system. This container isolation means that if the untrusted site or file turns out to be malicious, the host device is protected, and the attacker can't get to your enterprise data. For example, this approach makes the isolated container anonymous, so an attacker can't get to your employee's enterprise credentials.

Hardware isolation of Microsoft Edge & Microsoft Office with Microsoft Defender Application Guard



What types of devices should use Application Guard?

Application Guard has been created to target several types of devices:

- **Enterprise desktops.** These desktops are domain-joined and managed by your organization. Configuration management is primarily done through Microsoft Endpoint Manager or Microsoft Intune. Employees typically have Standard User privileges and use a high-bandwidth, wired, corporate network.
- **Enterprise mobile laptops.** These laptops are domain-joined and managed by your organization. Configuration management is primarily done through Microsoft Endpoint Manager or Microsoft Intune. Employees typically have Standard User privileges and use a high-bandwidth, wireless, corporate network.
- **Bring your own device (BYOD) mobile laptops.** These personally-owned laptops are not domain-joined, but are managed by your organization through tools, such as Microsoft Intune. The employee is typically an admin on the device and uses a high-bandwidth wireless corporate network while at work and a comparable personal network while at home.
- **Personal devices.** These personally-owned desktops or mobile laptops are not domain-joined or managed by an organization. The user is an admin on the device and uses a high-bandwidth wireless personal network while at home or a comparable public network while outside.

Related articles

ARTICLE	DESCRIPTION
System requirements for Microsoft Defender Application Guard	Specifies the prerequisites necessary to install and use Application Guard.
Prepare and install Microsoft Defender Application Guard	Provides instructions about determining which mode to use, either Standalone or Enterprise-managed, and how to install Application Guard in your organization.
Configure the Group Policy settings for Microsoft Defender Application Guard	Provides info about the available Group Policy and MDM settings.
Testing scenarios using Microsoft Defender Application Guard in your business or organization	Provides a list of suggested testing scenarios that you can use to test Application Guard in your organization.
Microsoft Defender Application Guard Extension for web browsers	Describes the Application Guard extension for Chrome and Firefox, including known issues, and a troubleshooting guide
Microsoft Defender Application Guard for Microsoft Office	Describes Application Guard for Microsoft Office, including minimum hardware requirements, configuration, and a troubleshooting guide
Frequently asked questions - Microsoft Defender Application Guard	Provides answers to frequently asked questions about Application Guard features, integration with the Windows operating system, and general configuration.
Use a network boundary to add trusted sites on Windows devices in Microsoft Intune	Network boundary, a feature that helps you protect your environment from sites that aren't trusted by your organization.

Windows Sandbox

7/1/2022 • 2 minutes to read • [Edit Online](#)

Windows Sandbox provides a lightweight desktop environment to safely run applications in isolation. Software installed inside the Windows Sandbox environment remains "sandboxed" and runs separately from the host machine.

A sandbox is temporary. When it's closed, all the software and files and the state are deleted. You get a brand-new instance of the sandbox every time you open the application. Note, however, that as of [Windows 11 Build 22509](#), your data will persist through a restart initiated from inside the virtualized environment—useful for installing applications that require the OS to reboot.

Software and applications installed on the host aren't directly available in the sandbox. If you need specific applications available inside the Windows Sandbox environment, they must be explicitly installed within the environment.

Windows Sandbox has the following properties:

- **Part of Windows:** Everything required for this feature is included in Windows 10 Pro and Enterprise. There's no need to download a VHD.
- **Pristine:** Every time Windows Sandbox runs, it's as clean as a brand-new installation of Windows.
- **Disposable:** Nothing persists on the device. Everything is discarded when the user closes the application.
- **Secure:** Uses hardware-based virtualization for kernel isolation. It relies on the Microsoft hypervisor to run a separate kernel that isolates Windows Sandbox from the host.
- **Efficient:** Uses the integrated kernel scheduler, smart memory management, and virtual GPU.

IMPORTANT

Windows Sandbox enables network connection by default. It can be disabled using the [Windows Sandbox configuration file](#).

The following video provides an overview of Windows Sandbox.

Prerequisites

- Windows 10 Pro, Enterprise or Education build 18305 or Windows 11 (*Windows Sandbox is currently not supported on Windows Home edition*)
- AMD64 or (as of [Windows 11 Build 22483](#)) ARM64 architecture
- Virtualization capabilities enabled in BIOS
- At least 4 GB of RAM (8 GB recommended)
- At least 1 GB of free disk space (SSD recommended)
- At least two CPU cores (four cores with hyperthreading recommended)

Installation

1. Ensure that your machine is using Windows 10 Pro or Enterprise, build version 18305 or Windows 11.

2. Enable virtualization on the machine.

- If you're using a physical machine, make sure virtualization capabilities are enabled in the BIOS.
- If you're using a virtual machine, run the following PowerShell command to enable nested virtualization:

```
Set-VMProcessor -VMName \<VMName> -ExposeVirtualizationExtensions $true
```

3. Use the search bar on the task bar and type **Turn Windows Features on or off** to access the Windows Optional Features tool. Select **Windows Sandbox** and then **OK**. Restart the computer if you're prompted.

If the **Windows Sandbox** option is unavailable, your computer doesn't meet the requirements to run Windows Sandbox. If you think this is incorrect, review the prerequisite list as well as steps 1 and 2.

NOTE

To enable Sandbox using PowerShell, open PowerShell as Administrator and run **Enable-WindowsOptionalFeature -FeatureName "Containers-DisposableClientVM" -All -Online**.

4. Locate and select **Windows Sandbox** on the Start menu to run it for the first time.

Usage

1. Copy an executable file (and any other files needed to run the application) from the host and paste them into the **Windows Sandbox** window.
2. Run the executable file or installer inside the sandbox.
3. When you're finished experimenting, close the sandbox. A dialog box will state that all sandbox content will be discarded and permanently deleted. Select **Ok**.
4. Confirm that your host machine doesn't exhibit any of the modifications that you made in Windows Sandbox.

Windows Sandbox architecture

7/1/2022 • 2 minutes to read • [Edit Online](#)

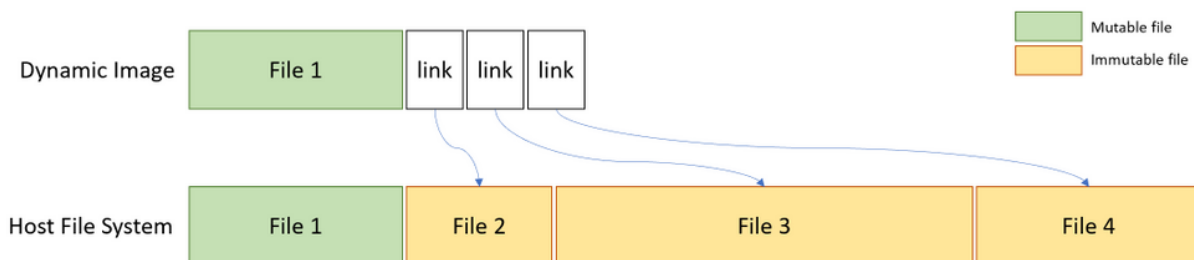
Windows Sandbox benefits from new container technology in Windows to achieve a combination of security, density, and performance that isn't available in traditional VMs.

Dynamically generated image

Rather than requiring a separate copy of Windows to boot the sandbox, Dynamic Base Image technology leverages the copy of Windows already installed on the host.

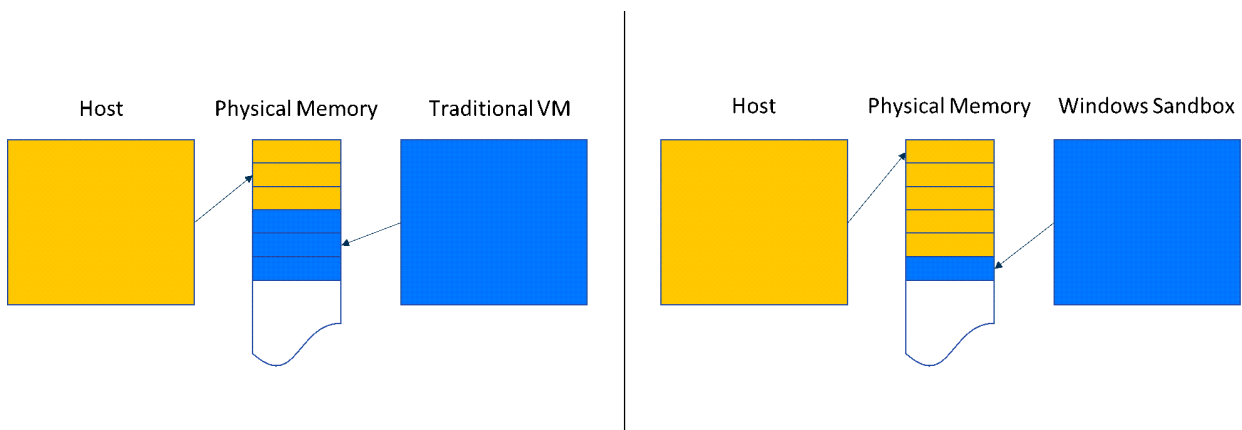
Most OS files are immutable and can be freely shared with Windows Sandbox. A small subset of operating system files are mutable and cannot be shared, so the sandbox base image contains pristine copies of them. A complete Windows image can be constructed from a combination of the sharable immutable files on the host and the pristine copies of the mutable files. By using this scheme, Windows Sandbox has a full Windows installation to boot from without needing to download or store an additional copy of Windows.

Before Windows Sandbox is installed, the dynamic base image package is stored as a compressed 30-MB package. Once it's installed, the dynamic base image occupies about 500 MB of disk space.



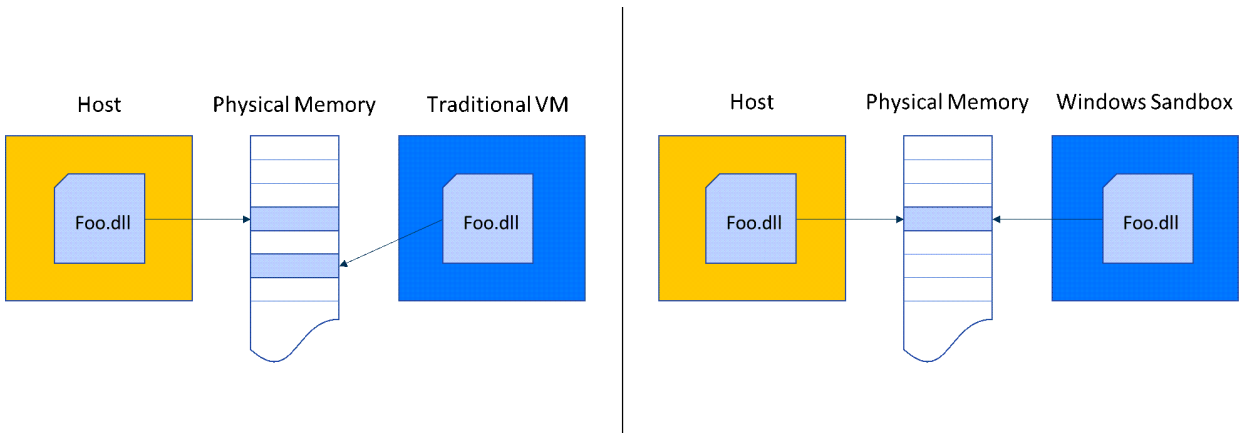
Memory management

Traditional VMs apportion statically sized allocations of host memory. When resource needs change, classic VMs have limited mechanisms for adjusting their resource needs. On the other hand, containers collaborate with the host to dynamically determine how host resources are allocated. This method is similar to how processes normally compete for memory on the host. If the host is under memory pressure, it can reclaim memory from the container much like it would with a process.



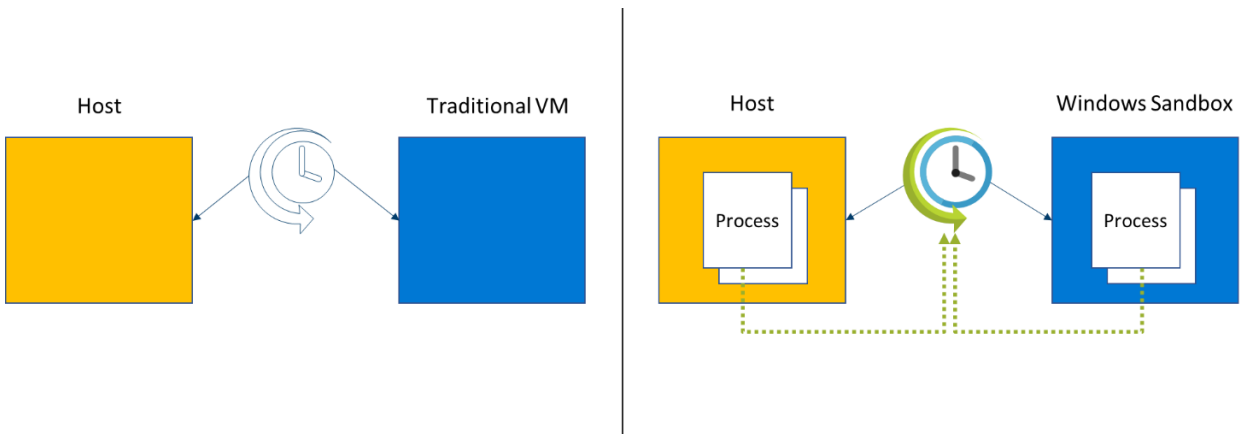
Memory sharing

Because Windows Sandbox runs the same operating system image as the host, it has been enhanced to use the same physical memory pages as the host for operating system binaries via a technology referred to as "direct map." For example, when *ntdll.dll* is loaded into memory in the sandbox, it uses the same physical pages as those of the binary when loaded on the host. Memory sharing between the host and the sandbox results in a smaller memory footprint when compared to traditional VMs, without compromising valuable host secrets.



Integrated kernel scheduler

With ordinary virtual machines, the Microsoft hypervisor controls the scheduling of the virtual processors running in the VMs. Windows Sandbox uses a new technology called "integrated scheduling," which allows the host scheduler to decide when the sandbox gets CPU cycles.

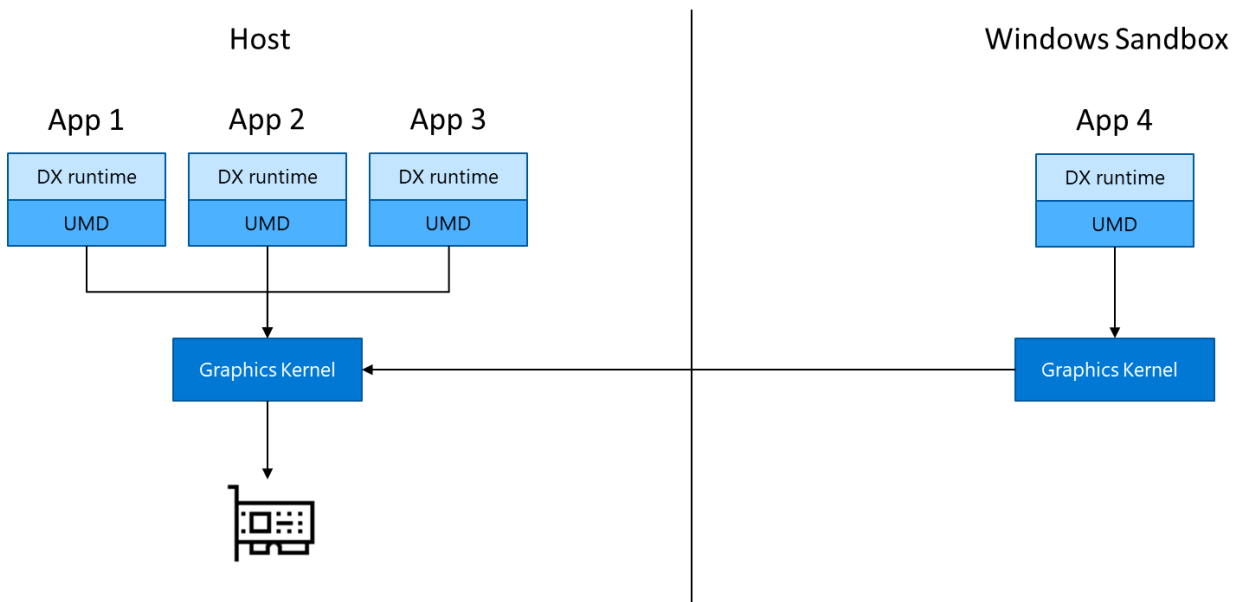


Windows Sandbox employs a unique policy that allows the virtual processors of the Sandbox to be scheduled like host threads. Under this scheme, high-priority tasks on the host can preempt less important work in the Sandbox. This means that the most important work will be prioritized, whether it's on the host or in the container.

WDDM GPU virtualization

Hardware accelerated rendering is key to a smooth and responsive user experience, especially for graphics-intensive use cases. Microsoft works with its graphics ecosystem partners to integrate modern graphics virtualization capabilities directly into DirectX and Windows Display Driver Model (WDDM), the driver model used by Windows.

This feature allows programs running inside the sandbox to compete for GPU resources with applications that are running on the host.



To take advantage of these benefits, a system with a compatible GPU and graphics drivers (WDDM 2.5 or newer) is required. Incompatible systems will render apps in Windows Sandbox with Microsoft's CPU-based rendering technology, Windows Advanced Rasterization Platform (WARP).

Battery pass-through

Windows Sandbox is also aware of the host's battery state, which allows it to optimize its power consumption. This functionality is critical for technology that is used on laptops, where battery life is often critical.

Windows Sandbox configuration

7/1/2022 • 6 minutes to read • [Edit Online](#)

Windows Sandbox supports simple configuration files, which provide a minimal set of customization parameters for Sandbox. This feature can be used with Windows 10 build 18342 or Windows 11. Windows Sandbox configuration files are formatted as XML and are associated with Sandbox via the `.wsb` file extension.

A configuration file enables the user to control the following aspects of Windows Sandbox:

- **vGPU (virtualized GPU):** Enable or disable the virtualized GPU. If vGPU is disabled, the sandbox will use Windows Advanced Rasterization Platform (WARP).
- **Networking:** Enable or disable network access within the sandbox.
- **Mapped folders:** Share folders from the host with *read* or *write* permissions. Note that exposing host directories may allow malicious software to affect the system or steal data.
- **Logon command:** A command that's executed when Windows Sandbox starts.
- **Audio input:** Shares the host's microphone input into the sandbox.
- **Video input:** Shares the host's webcam input into the sandbox.
- **Protected client:** Places increased security settings on the RDP session to the sandbox.
- **Printer redirection:** Shares printers from the host into the sandbox.
- **Clipboard redirection:** Shares the host clipboard with the sandbox so that text and files can be pasted back and forth.
- **Memory in MB:** The amount of memory, in megabytes, to assign to the sandbox.

Creating a configuration file

To create a simple configuration file:

1. Open a plain text editor or source code editor (e.g. Notepad, Visual Studio Code, etc.)
2. Insert the following lines:

```
<Configuration>
</Configuration>
```

3. Add appropriate configuration text between the two lines. For details, see the correct syntax and the examples below.
4. Save the file with the desired name, but make sure its filename extension is `.wsb`. In Notepad, you should enclose the filename and the extension inside double quotation marks, e.g. `"My config file.wsb"`.

Using a configuration file

To use a configuration file, double-click it to start Windows Sandbox according to its settings. You can also invoke it via the command line as shown here:

```
C:\Temp> MyConfigFile.wsb
```

Keywords, values, and limits

vGPU

Enables or disables GPU sharing.

```
<vGPU>value</vGPU>
```

Supported values:

- *Enable*: Enables vGPU support in the sandbox.
- *Disable*: Disables vGPU support in the sandbox. If this value is set, the sandbox will use software rendering, which may be slower than virtualized GPU.
- *Default*: This is the default value for vGPU support. Currently this means vGPU is disabled.

NOTE

Enabling virtualized GPU can potentially increase the attack surface of the sandbox.

Networking

Enables or disables networking in the sandbox. You can disable network access to decrease the attack surface exposed by the sandbox.

```
<Networking>value</Networking>
```

Supported values:

- *Disable*: Disables networking in the sandbox.
- *Default*: This is the default value for networking support. This value enables networking by creating a virtual switch on the host and connects the sandbox to it via a virtual NIC.

NOTE

Enabling networking can expose untrusted applications to the internal network.

Mapped folders

An array of folders, each representing a location on the host machine that will be shared into the sandbox at the specified path. At this time, relative paths are not supported. If no path is specified, the folder will be mapped to the container user's desktop.

```
<MappedFolders>
  <MappedFolder>
    <HostFolder>absolute path to the host folder</HostFolder>
    <SandboxFolder>absolute path to the sandbox folder</SandboxFolder>
    <ReadOnly>value</ReadOnly>
  </MappedFolder>
  <MappedFolder>
    ...
  </MappedFolder>
</MappedFolders>
```

HostFolder: Specifies the folder on the host machine to share into the sandbox. Note that the folder must already exist on the host, or the container will fail to start.

SandboxFolder: Specifies the destination in the sandbox to map the folder to. If the folder doesn't exist, it will be created. If no sandbox folder is specified, the folder will be mapped to the container desktop.

ReadOnly: If *true*, enforces read-only access to the shared folder from within the container. Supported values: *true/false*. Defaults to *false*.

NOTE

Files and folders mapped in from the host can be compromised by apps in the sandbox or potentially affect the host.

Logon command

Specifies a single command that will be invoked automatically after the sandbox logs on. Apps in the sandbox are run under the container user account.

```
<LogonCommand>  
  <Command>command to be invoked</Command>  
</LogonCommand>
```

Command: A path to an executable or script inside the container that will be executed after login.

NOTE

Although very simple commands will work (such as launching an executable or script), more complicated scenarios involving multiple steps should be placed into a script file. This script file may be mapped into the container via a shared folder, and then executed via the *LogonCommand* directive.

Audio input

Enables or disables audio input to the sandbox.

```
<AudioInput>value</AudioInput>
```

Supported values:

- *Enable:* Enables audio input in the sandbox. If this value is set, the sandbox will be able to receive audio input from the user. Applications that use a microphone may require this capability.
- *Disable:* Disables audio input in the sandbox. If this value is set, the sandbox can't receive audio input from the user. Applications that use a microphone may not function properly with this setting.
- *Default:* This is the default value for audio input support. Currently this means audio input is enabled.

NOTE

There may be security implications of exposing host audio input to the container.

Video input

Enables or disables video input to the sandbox.

```
<VideoInput>value</VideoInput>
```

Supported values:

- *Enable:* Enables video input in the sandbox.
- *Disable:* Disables video input in the sandbox. Applications that use video input may not function properly in the sandbox.
- *Default:* This is the default value for video input support. Currently this means video input is disabled. Applications that use video input may not function properly in the sandbox.

NOTE

There may be security implications of exposing host video input to the container.

Protected client

Applies additional security settings to the sandbox Remote Desktop client, decreasing its attack surface.

```
<ProtectedClient>value</ProtectedClient>
```

Supported values:

- *Enable*: Runs Windows sandbox in Protected Client mode. If this value is set, the sandbox runs with extra security mitigations enabled.
- *Disable*: Runs the sandbox in standard mode without extra security mitigations.
- *Default*: This is the default value for Protected Client mode. Currently, this means the sandbox doesn't run in Protected Client mode.

NOTE

This setting may restrict the user's ability to copy/paste files in and out of the sandbox.

Printer redirection

Enables or disables printer sharing from the host into the sandbox.

```
<PrinterRedirection>value</PrinterRedirection>
```

Supported values:

- *Enable*: Enables sharing of host printers into the sandbox.
- *Disable*: Disables printer redirection in the sandbox. If this value is set, the sandbox can't view printers from the host.
- *Default*: This is the default value for printer redirection support. Currently this means printer redirection is disabled.

Clipboard redirection

Enables or disables sharing of the host clipboard with the sandbox.

```
<ClipboardRedirection>value</ClipboardRedirection>
```

Supported values:

- *Disable*: Disables clipboard redirection in the sandbox. If this value is set, copy/paste in and out of the sandbox will be restricted.
- *Default*: This is the default value for clipboard redirection. Currently copy/paste between the host and sandbox are permitted under *Default*.

Memory in MB

Specifies the amount of memory that the sandbox can use in megabytes (MB).

```
<MemoryInMB>value</MemoryInMB>
```

If the memory value specified is insufficient to boot a sandbox, it will be automatically increased to the required minimum amount.

Example 1

The following config file can be used to easily test downloaded files inside the sandbox. To achieve this, networking and vGPU are disabled, and the sandbox is allowed read-only access to the shared downloads folder. For convenience, the logon command opens the downloads folder inside the sandbox when it's started.

Downloads.wsb

```
<Configuration>
  <VGpu>Disable</VGpu>
  <Networking>Disable</Networking>
  <MappedFolders>
    <MappedFolder>
      <HostFolder>C:\Users\Public\Downloads</HostFolder>
      <SandboxFolder>C:\Users\WDAGUtilityAccount\Downloads</SandboxFolder>
      <ReadOnly>true</ReadOnly>
    </MappedFolder>
  </MappedFolders>
  <LogonCommand>
    <Command>explorer.exe C:\users\WDAGUtilityAccount\Downloads</Command>
  </LogonCommand>
</Configuration>
```

Example 2

The following config file installs Visual Studio Code in the sandbox, which requires a slightly more complicated LogonCommand setup.

Two folders are mapped into the sandbox; the first (SandboxScripts) contains VSCodeInstall.cmd, which will install and run Visual Studio Code. The second folder (CodingProjects) is assumed to contain project files that the developer wants to modify using Visual Studio Code.

With the Visual Studio Code installer script already mapped into the sandbox, the LogonCommand can reference it.

VSCodeInstall.cmd

```
REM Download Visual Studio Code
curl -L "https://update.code.visualstudio.com/latest/win32-x64-user/stable" --output
C:\users\WDAGUtilityAccount\Desktop\vscode.exe

REM Install and run Visual Studio Code
C:\users\WDAGUtilityAccount\Desktop\vscode.exe /verysilent /suppressmsgboxes
```

VSCode.wsb

```
<Configuration>
  <MappedFolders>
    <MappedFolder>
      <HostFolder>C:\SandboxScripts</HostFolder>
      <ReadOnly>true</ReadOnly>
    </MappedFolder>
    <MappedFolder>
      <HostFolder>C:\CodingProjects</HostFolder>
      <ReadOnly>>false</ReadOnly>
    </MappedFolder>
  </MappedFolders>
  <LogonCommand>
    <Command>C:\Users\WDAGUtilityAccount\Desktop\SandboxScripts\VSCodeInstall.cmd</Command>
  </LogonCommand>
</Configuration>
```

Microsoft Defender SmartScreen

7/1/2022 • 3 minutes to read • [Edit Online](#)

Applies to:

- Windows 10
- Windows 11
- Microsoft Edge

Microsoft Defender SmartScreen protects against phishing or malware websites and applications, and the downloading of potentially malicious files.

Microsoft Defender SmartScreen determines whether a site is potentially malicious by:

- Analyzing visited webpages and looking for indications of suspicious behavior. If Microsoft Defender SmartScreen determines that a page is suspicious, it will show a warning page to advise caution.
- Checking the visited sites against a dynamic list of reported phishing sites and malicious software sites. If it finds a match, Microsoft Defender SmartScreen shows a warning to let the user know that the site might be malicious.

Microsoft Defender SmartScreen determines whether a downloaded app or app installer is potentially malicious by:

- Checking downloaded files against a list of reported malicious software sites and programs known to be unsafe. If it finds a match, Microsoft Defender SmartScreen shows a warning to let the user know that the site might be malicious.
- Checking downloaded files against a list of files that are well known and downloaded by many Windows users. If the file isn't on that list, Microsoft Defender SmartScreen shows a warning, advising caution.

Benefits of Microsoft Defender SmartScreen

Microsoft Defender SmartScreen provide an early warning system against websites that might engage in phishing attacks or attempt to distribute malware through a socially engineered attack. The primary benefits are:

- **Anti-phishing and anti-malware support:** Microsoft Defender SmartScreen helps to protect users from sites that are reported to host phishing attacks or attempt to distribute malicious software. It can also help protect against deceptive advertisements, scam sites, and drive-by attacks. Drive-by attacks are web-based attacks that tend to start on a trusted site, targeting security vulnerabilities in commonly used software. Because drive-by attacks can happen even if the user does not click or download anything on the page, the danger often goes unnoticed. For more information about drive-by attacks, see [Evolving Microsoft Defender SmartScreen to protect you from drive-by attacks](#)
- **Reputation-based URL and app protection:** Microsoft Defender SmartScreen evaluates a website's URLs to determine if they're known to distribute or host unsafe content. It also provides reputation checks for apps, checking downloaded programs and the digital signature used to sign a file. If a URL, a file, an app, or a certificate has an established reputation, users won't see any warnings. If, however, there's no reputation, the item is marked as a higher risk and presents a warning to the user.
- **Operating system integration:** Microsoft Defender SmartScreen is integrated into the Windows 10 operating system. It checks any files an app (including 3rd-party browsers and email clients) that

attempts to download and run.

- **Improved heuristics and diagnostic data:** Microsoft Defender SmartScreen is constantly learning and endeavoring to stay up to date, so it can help to protect you against potentially malicious sites and files.
- **Management through Group Policy and Microsoft Intune:** Microsoft Defender SmartScreen supports using both Group Policy and Microsoft Intune settings. For more info about all available settings, see [Available Microsoft Defender SmartScreen Group Policy and mobile device management \(MDM\) settings](#).
- **Blocking URLs associated with potentially unwanted applications:** In Microsoft Edge (based on Chromium), SmartScreen blocks URLs associated with potentially unwanted applications, or PUAs. For more information on blocking URLs associated with PUAs, see [Detect and block potentially unwanted applications](#).

IMPORTANT

SmartScreen protects against malicious files from the internet. It does not protect against malicious files on internal locations or network shares, such as shared folders with UNC paths or SMB/CIFS shares.

Submit files to Microsoft Defender SmartScreen for review

If you believe a warning or block was incorrectly shown for a file or application, or if you believe an undetected file is malware, you can [submit a file](#) to Microsoft for review. For more information, see [Submit files for analysis](#).

When submitting Microsoft Defender SmartScreen products, make sure to select **Microsoft Defender SmartScreen** from the product menu.

The screenshot shows a web form titled "Submit a file for malware analysis". Below the title, it says "Specify the file and provide information that will help us to efficiently handle your case. Required fields are marked with an asterisk (*)." The form has a section "Select the Microsoft security product used to scan the file *" with a dropdown menu. The dropdown is open, showing options: "Microsoft Defender Antivirus (Windows 10)", "Microsoft Defender Smartscreen", "Microsoft Security Essentials", "Windows Defender (Windows 8)", and "Windows 10X Device". Below this is a text input field with a "Select" button. At the bottom, there is a note: "Maximum file size is 500 MB. Use the password 'infected' to encrypt ZIP or RAR archives." and another note: "NOTE: Submit only the specific files you want analyzed. Submitting an installer package or an archive with a large number of files may delay the analysis and cause your submission to be deprioritized."

Viewing Microsoft Defender SmartScreen anti-phishing events

NOTE

No SmartScreen events will be logged when using Microsoft Edge version 77 or later.

When Microsoft Defender SmartScreen warns or blocks a user from a website, it's logged as [Event 1035 - Anti-Phishing](#).

Viewing Windows event logs for Microsoft Defender SmartScreen

Microsoft Defender SmartScreen events appear in the Microsoft-Windows-SmartScreen/Debug log, in the Event Viewer.

Windows event log for SmartScreen is disabled by default, users can use Event Viewer UI to enable the log or use the command line to enable it:

```
wevtutil sl Microsoft-Windows-SmartScreen/Debug /e:true
```

NOTE

For information on how to use the Event Viewer, see [Windows Event Viewer](#).

EVENTID	DESCRIPTION
1000	Application Windows Defender SmartScreen Event
1001	Uri Windows Defender SmartScreen Event
1002	User Decision Windows Defender SmartScreen Event

Related topics

- [SmartScreen Frequently Asked Questions](#)
- [Threat protection](#)
- [Available Microsoft Defender SmartScreen Group Policy and mobile device management \(MDM\) settings](#)
- [Configuration service provider reference](#)

Configure S/MIME for Windows

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11

S/MIME stands for Secure/Multipurpose Internet Mail Extensions, and provides an added layer of security for email sent to and from an Exchange ActiveSync (EAS) account. S/MIME lets users encrypt outgoing messages and attachments so that only intended recipients who have a digital identification (ID), also known as a certificate, can read them. Users can digitally sign a message, which provides the recipients with a way to verify the identity of the sender and that the message hasn't been tampered with.

About message encryption

Users can send encrypted message to people in their organization and people outside their organization if they have their encryption certificates. However, users using Windows Mail app can only read encrypted messages if the message is received on their Exchange account and they have corresponding decryption keys.

Encrypted messages can be read only by recipients who have a certificate. If you try to send an encrypted message to recipient(s) whose encryption certificate are not available, the app will prompt you to remove these recipients before sending the email.

About digital signatures

A digitally signed message reassures the recipient that the message hasn't been tampered with and verifies the identity of the sender. Recipients can only verify the digital signature if they're using an email client that supports S/MIME.

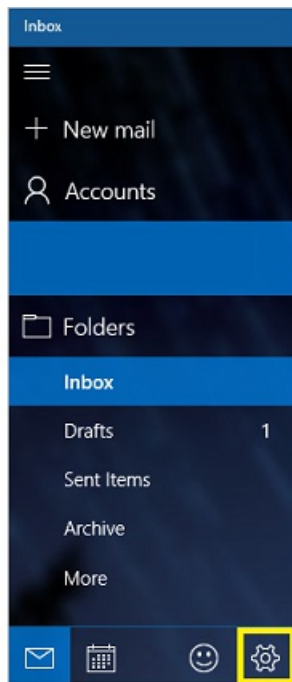
Prerequisites

- [S/MIME is enabled for Exchange accounts](#) (on-premises and Office 365). Users can't use S/MIME signing and encryption with a personal account such as Outlook.com.
- Valid Personal Information Exchange (PFX) certificates are installed on the device.
 - [How to Create PFX Certificate Profiles in Configuration Manager](#)
 - [Enable access to company resources using certificate profiles with Microsoft Intune](#)

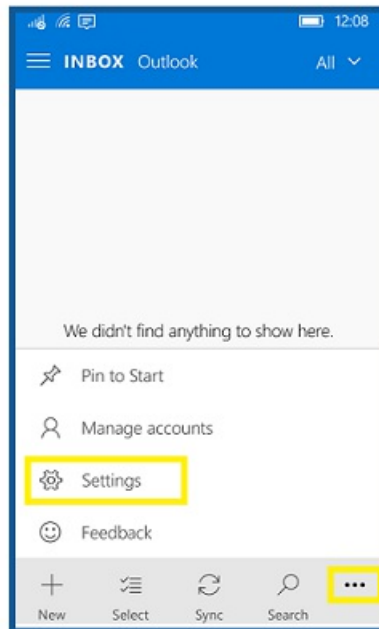
Choose S/MIME settings

On the device, perform the following steps: (add select certificate)

1. Open the Mail app.
2. Open **Settings** by tapping the gear icon on a PC, or the ellipsis (...) and then the gear icon on a phone.

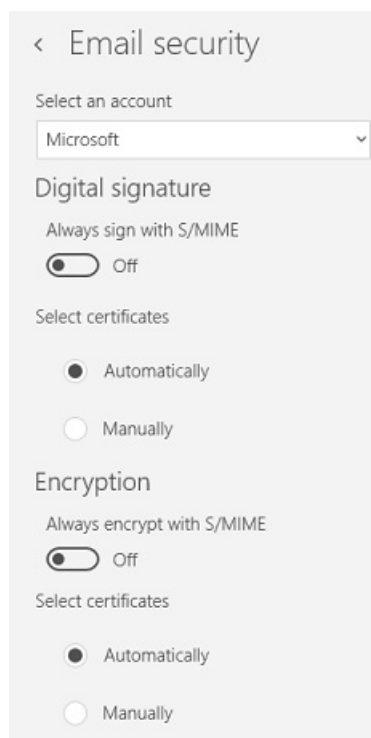


Desktop app



Phone app

3. Tap **Email security**.



4. In **Select an account**, select the account for which you want to configure S/MIME options.

5. Make a certificate selection for digital signature and encryption.

- Select **Automatically** to let the app choose the certificate.
- Select **Manually** to specify the certificate yourself from the list of valid certificates on the device.

6. (Optional) Select **Always sign with S/MIME**, **Always encrypt with S/MIME**, or both, to automatically digitally sign or encrypt all outgoing messages.

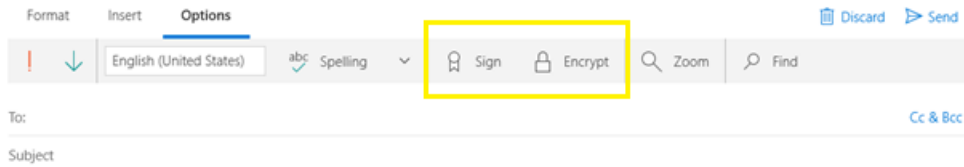
NOTE

The option to sign or encrypt can be changed for individual messages, unless EAS policies prevent it.

7. Tap the back arrow.

Encrypt or sign individual messages

1. While composing a message, choose **Options** from the ribbon. On phone, **Options** can be accessed by tapping the ellipsis (...).
2. Use **Sign** and **Encrypt** icons to turn on digital signature and encryption for this message.



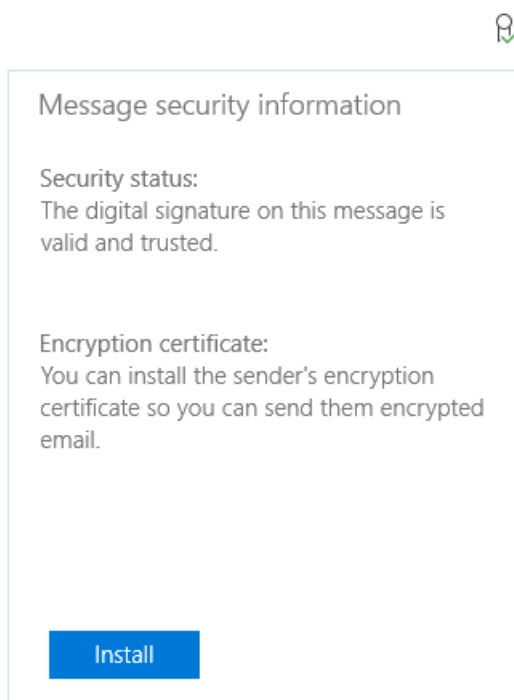
Read signed or encrypted messages

When you receive an encrypted message, the mail app will check whether there is a certificate available on your computer. If there is a certificate available, the message will be decrypted when you open it. If your certificate is stored on a smartcard, you will be prompted to insert the smartcard to read the message. Your smartcard may also require a PIN to access the certificate.

Install certificates from a received message

When you receive a signed email, the app provide feature to install corresponding encryption certificate on your device if the certificate is available. This certificate can then be used to send encrypted email to this person.

1. Open a signed email.
2. Tap or click the digital signature icon in the reading pane.
3. Tap **Install**.



Windows Credential Theft Mitigation Guide

Abstract

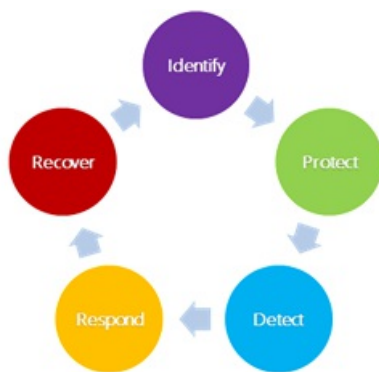
7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10

This topic provides a summary of the Windows credential theft mitigation guide, which can be downloaded from the [Microsoft Download Center](#). This guide explains how credential theft attacks occur and the strategies and countermeasures you can implement to mitigate them, following these security stages:

- Identify high-value assets
- Protect against known and unknown threats
- Detect pass-the-hash and related attacks
- Respond to suspicious activity
- Recover from a breach



Attacks that steal credentials

Learn about the different types of attacks that are used to steal credentials, and the factors that can place your organization at risk. The types of attacks that are covered include:

- Pass the hash
- Kerberos pass the ticket
- Kerberos golden ticket and silver ticket
- Key loggers
- Shoulder surfing

Credential protection strategies

This part of the guide helps you consider the mindset of the attacker, with prescriptive guidance about how to prioritize high-value accounts and computers. You'll learn how to architect a defense against credential theft:

- Establish a containment model for account privileges
- Harden and restrict administrative hosts
- Ensure that security configurations and best practices are implemented

Technical countermeasures for credential theft

Objectives and expected outcomes are covered for each of these countermeasures:

- Use Windows 10 with Credential Guard
- Restrict and protect high-privilege domain accounts
- Restrict and protect local accounts with administrative privileges
- Restrict inbound network traffic

Many other countermeasures are also covered, such as using Microsoft Passport and Windows Hello, or multifactor authentication.

Detecting credential attacks

This sections covers how to detect the use of stolen credentials and how to collect computer events to help you detect credential theft.

Responding to suspicious activity

Learn Microsoft's recommendations for responding to incidents, including how to recover control of compromised accounts, how to investigate attacks, and how to recover from a breach.

Windows identity and privacy

7/1/2022 • 2 minutes to read • [Edit Online](#)

Malicious actors launch millions of password attacks every day. Weak passwords, password spraying, and phishing are the entry point for many attacks. Knowing that the right user is accessing the right device and the right data is critical to keeping your business, family, and self, safe and secure. Windows Hello, Windows Hello for Business, and Credential Guard enable customers to move to passwordless multifactor authentication (MFA). MFA can reduce the risk of compromise in organizations.

SECURITY CAPABILITIES	DESCRIPTION
Securing user identity with Windows Hello	Windows Hello and Windows Hello for Business replace password-based authentication with a stronger authentication model to sign into your device using a passcode (PIN) or other biometric based authentication. This PIN or biometric based authentication is only valid on the device that you registered it for and cannot be used on another device. Learn more: Windows Hello for Business
Windows Defender Credential Guard and Remote Credential Guard	Windows Defender Credential Guard helps protect your systems from credential theft attack techniques (pass-the-hash or pass-the-ticket) as well as helping prevent malware from accessing system secrets even if the process is running with admin privileges. Windows Defender Remote Credential Guard helps you protect your credentials over a Remote Desktop connection by redirecting Kerberos requests back to the device that's requesting the connection. It also provides single sign-on experiences for Remote Desktop sessions. Learn more: Protect derived domain credentials with Windows Defender Credential Guard and Protect Remote Desktop credentials with Windows Defender Remote Credential Guard
FIDO Alliance	Fast Identity Online (FIDO) defined protocols are becoming the open standard for providing strong authentication that helps prevent phishing and are user-friendly and privacy-respecting. Windows 11 supports the use of device sign-in with FIDO 2 security keys, and with Microsoft Edge or other modern browsers, supports the use of secure FIDO-backed credentials to keep user accounts protected. Learn more about the FIDO Alliance .
Microsoft Authenticator	The Microsoft Authenticator app is a perfect companion to help keep secure with Windows 11. It allows easy, secure sign-ins for all your online accounts using multi-factor authentication, passwordless phone sign-in, or password autofill. You also have additional account management options for your Microsoft personal, work, or school accounts. Microsoft Authenticator can be used to set up multi-factor authentication for your users. Learn more: Enable passwordless sign-in with the Microsoft Authenticator app .

SECURITY CAPABILITIES	DESCRIPTION
Smart Cards	Smart cards are tamper-resistant portable storage devices that can enhance the security of tasks in Windows, such as authenticating clients, signing code, securing e-mail, and signing in with Windows domain accounts. Learn more about Smart Cards .
Access Control	Access control is the process of authorizing users, groups, and computers to access objects and assets on a network or computer. Computers can control the use of system and network resources through the interrelated mechanisms of authentication and authorization. Learn more: Access Control .

Windows Credential Theft Mitigation Guide

Abstract

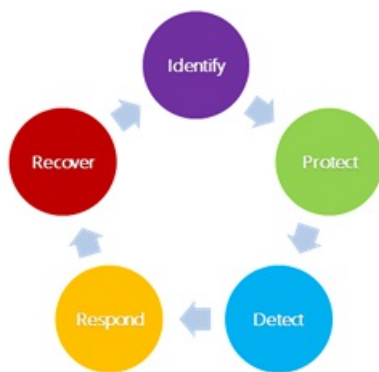
7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10

This topic provides a summary of the Windows credential theft mitigation guide, which can be downloaded from the [Microsoft Download Center](#). This guide explains how credential theft attacks occur and the strategies and countermeasures you can implement to mitigate them, following these security stages:

- Identify high-value assets
- Protect against known and unknown threats
- Detect pass-the-hash and related attacks
- Respond to suspicious activity
- Recover from a breach



Attacks that steal credentials

Learn about the different types of attacks that are used to steal credentials, and the factors that can place your organization at risk. The types of attacks that are covered include:

- Pass the hash
- Kerberos pass the ticket
- Kerberos golden ticket and silver ticket
- Key loggers
- Shoulder surfing

Credential protection strategies

This part of the guide helps you consider the mindset of the attacker, with prescriptive guidance about how to prioritize high-value accounts and computers. You'll learn how to architect a defense against credential theft:

- Establish a containment model for account privileges
- Harden and restrict administrative hosts
- Ensure that security configurations and best practices are implemented

Technical countermeasures for credential theft

Objectives and expected outcomes are covered for each of these countermeasures:

- Use Windows 10 with Credential Guard
- Restrict and protect high-privilege domain accounts
- Restrict and protect local accounts with administrative privileges
- Restrict inbound network traffic

Many other countermeasures are also covered, such as using Microsoft Passport and Windows Hello, or multifactor authentication.

Detecting credential attacks

This sections covers how to detect the use of stolen credentials and how to collect computer events to help you detect credential theft.

Responding to suspicious activity

Learn Microsoft's recommendations for responding to incidents, including how to recover control of compromised accounts, how to investigate attacks, and how to recover from a breach.

Enterprise Certificate Pinning

7/1/2022 • 12 minutes to read • [Edit Online](#)

Applies to

- Windows 10

Enterprise certificate pinning is a Windows feature for remembering, or pinning a root issuing certificate authority or end entity certificate to a given domain name. Enterprise certificate pinning helps reduce man-in-the-middle attacks by enabling you to protect your internal domain names from chaining to unwanted certificates or to fraudulently issued certificates.

NOTE

External domain names, where the certificate issued to these domains is issued by a public certificate authority, are not ideal for enterprise certificate pinning.

Windows Certificate APIs (CertVerifyCertificateChainPolicy and WinVerifyTrust) are updated to check if the site's chain that authenticates servers matches a restricted set of certificates. These restrictions are encapsulated in a Pin Rules Certificate Trust List (CTL) that is configured and deployed to Windows 10 computers. Any site certificate that triggers a name mismatch causes Windows to write an event to the CAPI2 event log and prevents the user from navigating to the web site using Microsoft Edge or Internet Explorer.

NOTE

Enterprise Certificate Pinning feature triggering doesn't cause clients other than Microsoft Edge or Internet Explorer to block the connection.

Deployment

To deploy enterprise certificate pinning, you need to:

- Create a well-formatted certificate pinning rule XML file
- Create a pin rules certificate trust list file from the XML file
- Apply the pin rules certificate trust list file to a reference administrative computer
- Deploy the registry configuration on the reference computer using Group Policy Management Console (GPMC), which is included in the [Remote Server Administration Tools \(RSAT\)](#).

Create a Pin Rules XML file

The XML-based pin rules file consists of a sequence of PinRule elements. Each PinRule element contains a sequence of one or more Site elements and a sequence of zero or more Certificate elements.

```

<PinRules ListIdentifier="PinRulesExample" Duration="P28D">

  <PinRule Name="AllCertificateAttributes" Error="None" Log="true">
    <Certificate File="Single.cer"/>
    <Certificate File="Multiple.p7b"/>
    <Certificate File="Multiple.sst"/>
    <Certificate Directory="Multiple"/>
    <Certificate Base64="MIIBY ... QFzuM"/>
    <Certificate File="WillExpire.cer" EndDate="2015-05-12T00:00:00Z"/>
    <Site Domain="xyz.com"/>
  </PinRule>

  <PinRule Name="MultipleSites" Log="false">
    <Certificate File="Root.cer"/>
    <Site Domain="xyz.com"/>
    <Site Domain=".xyz.com"/>
    <Site Domain="*.abc.xyz.com" AllSubdomains="true"/>
    <Site Domain="WillNormalize.com"/>
  </PinRule>

</PinRules>

```

PinRules Element

The PinRules element can have the following attributes. For help with formatting Pin Rules, see [Representing a Date in XML](#) or [Representing a Duration in XML](#).

ATTRIBUTE	DESCRIPTION	REQUIRED
Duration or NextUpdate	Specifies when the Pin Rules will expire. Either is required. NextUpdate takes precedence if both are specified. Duration , represented as an XML TimeSpan data type, doesn't allow years and months. You represent the NextUpdate attribute as an XML DateTime data type in UTC.	Required? Yes. At least one is required.
LogDuration or LogEndDate	Configures auditing only to extend beyond the expiration of enforcing the Pin Rules. LogEndDate , represented as an XML DateTime data type in UTC, takes precedence if both are specified. You represent LogDuration as an XML TimeSpan data type, which doesn't allow years and months. If none of the attributes are specified, auditing expiration uses Duration or NextUpdate attributes.	No.
ListIdentifier	Provides a friendly name for the list of pin rules. Windows doesn't use this attribute for certificate pinning enforcement; however, it's included when the pin rules are converted to a certificate trust list (CTL).	No.

PinRule Element

The PinRule element can have the following attributes.

ATTRIBUTE	DESCRIPTION	REQUIRED
Name	Uniquely identifies the PinRule . Windows uses this attribute to identify the element for a parsing error or for verbose output. The attribute isn't included in the generated certificate trust list (CTL).	Yes.
Error	Describes the action Windows performs when it encounters a PIN mismatch. You can choose from the following string values: <ul style="list-style-type: none"> - Revoked - Windows reports the certificate protecting the site as if it was revoked. This typically prevents the user from accessing the site. - InvalidName - Windows reports the certificate protecting the site as if the name on the certificate doesn't match the name of the site. This typically results in prompting the user before accessing the site. - None - The default value. No error is returned. You can use this setting to audit the pin rules without introducing any user friction. 	No.
Log	A Boolean value represents a string that equals true or false . By default, logging is enabled (true).	No.

Certificate element

The **Certificate** element can have the following attributes.

ATTRIBUTE	DESCRIPTION	REQUIRED
File	Path to a file containing one or more certificates. Where the certificate(s) can be encoded as: <ul style="list-style-type: none"> - single certificate - p7b - sst These files can also be Base64 formatted. All Site elements included in the same PinRule element can match any of these certificates.	Yes (File, Directory, or Base64 must be present).
Directory	Path to a directory containing one or more of the above certificate files. Skips any files not containing any certificates.	Yes (File, Directory, or Base64 must be present).

ATTRIBUTE	DESCRIPTION	REQUIRED
Base64	<p>Base64 encoded certificate(s). Where the certificate(s) can be encoded as:</p> <ul style="list-style-type: none"> - single certificate - p7b - sst <p>This allows the certificates to be included in the XML file without a file directory dependency.</p> <p>Note:</p> <p>You can use certutil -encode to convert a .cer file into base64. You can then use Notepad to copy and paste the base64 encoded certificate into the pin rule.</p>	Yes (File, Directory, or Base64 must be present).
EndDate	<p>Enables you to configure an expiration date for when the certificate is no longer valid in the pin rule.</p> <p>If you are in the process of switching to a new root or CA, you can set the EndDate to allow matching of this element's certificates.</p> <p>If the current time is past the EndDate, then, when creating the certificate trust list (CTL), the parser outputs a warning message and exclude the certificate(s) from the Pin Rule in the generated CTL.</p> <p>For help with formatting Pin Rules, see Representing a Date in XML.</p>	No.

Site element

The Site element can have the following attributes.

ATTRIBUTE	DESCRIPTION	REQUIRED
Domain	<p>Contains the DNS name to be matched for this pin rule. When creating the certificate trust list, the parser normalizes the input name string value as follows:</p> <ul style="list-style-type: none"> - If the DNS name has a leading "*", it's removed. - Non-ASCII DNS name is converted to ASCII Puny Code. - Upper case ASCII characters are converted to lower case. <p>If the normalized name has a leading ".", then, wildcard left-hand label matching is enabled. For example, ".xyz.com" would match "abc.xyz.com".</p>	Yes.

ATTRIBUTE	DESCRIPTION	REQUIRED
AllSubdomains	By default, wildcard left-hand label matching is restricted to a single left-hand label. This attribute can be set to "true" to enable wildcard matching of all of the left-hand labels. For example, setting this attribute would also match "123.abc.xyz.com" for the ".xyz.com" domain value.	No.

Create a Pin Rules Certificate Trust List

The command line utility, **Certutil.exe**, includes the **generatePinRulesCTL** argument to parse the XML file and generate the encoded certificate trust list (CTL) that you add to your reference Windows 10 version 1703 computer and subsequently deploy. The usage syntax is:

```
CertUtil [Options] -generatePinRulesCTL XMLFile CTLFile [SSTFile]
Generate Pin Rules CTL
  XMLFile -- input XML file to be parsed.
  CTLFile -- output CTL file to be generated.
  SSTFile -- optional .sst file to be created.
             The .sst file contains all of the certificates
             used for pinning.

Options:
  -f           -- Force overwrite
  -v           -- Verbose operation
```

The same certificate(s) can occur in multiple **PinRule** elements. The same domain can occur in multiple **PinRule** elements. Certutil coalesces these in the resultant pin rules certificate trust list.

Certutil.exe doesn't strictly enforce the XML schema definition. It does perform the following to enable other tools to add/consume their own specific elements and attributes:

- Skips elements before and after the **PinRules** element.
- Skips any element not matching **Certificate** or **Site** within the **PinRules** element.
- Skips any attributes not matching the above names for each element type.

Use the **certutil** command with the **generatePinRulesCTL** argument along with your XML file that contains your certificate pinning rules. Lastly, provide the name of an output file that will include your certificate pinning rules in the form of a certificate trust list.

```
certutil -generatePinRulesCTL certPinRules.xml pinrules.stl
```

Applying Certificate Pinning Rules to a Reference Computer

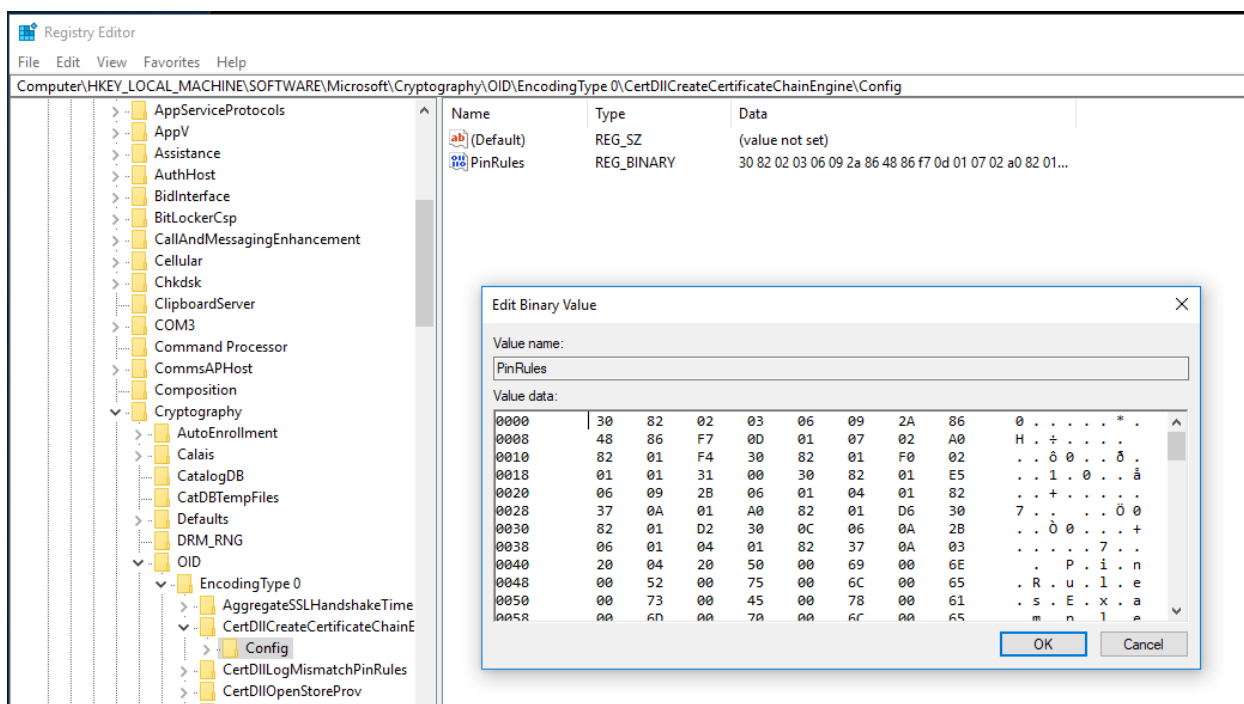
Now that your certificate pinning rules are in the certificate trust list format, you need to apply the settings to a reference computer as a prerequisite to deploying the setting to your enterprise. To simplify the deployment configuration, it's best to apply your certificate pinning rules to a computer that has the Group Policy Management Console (GPMC) included in the Remote Server Administration Tools (RSAT).

Use **certutil.exe** to apply your certificate pinning rules to your reference computer using the **setreg** argument. The **setreg** argument takes a secondary argument that determines the location of where certutil writes the certificate pinning rules. This secondary argument is **chain\PinRules**. The last argument you provide is the name of file that contains your certificate pinning rules in certificate trust list format (.stl). You'll pass the name of the file as the last argument; however, you need to prefix the file name with the '@' symbol as shown in the following example. You need to perform this command from an elevated command prompt.

```
Certutil -setreg chain\PinRules @pinrules.stl
```

Certutil writes the binary information to the following registration location:

NAME	VALUE
Key	HKLM\SOFTWARE\Microsoft\Cryptography\OID\EncodingType0\CertDIICreateCertificateChainEngine\Config
Name	PinRules
Value	Binary contents from the certificate pin rules certificate trust list file
Data type	REG_BINARY



Deploying Enterprise Pin Rule Settings using Group Policy

You've successfully created a certificate pinning rules XML file. From the XML file you've created a certificate pinning trust list file, and you've applied the contents of that file to your reference computer from which you can run the Group Policy Management Console. Now you need to configure a Group Policy object to include the applied certificate pin rule settings and deploy it to your environment.

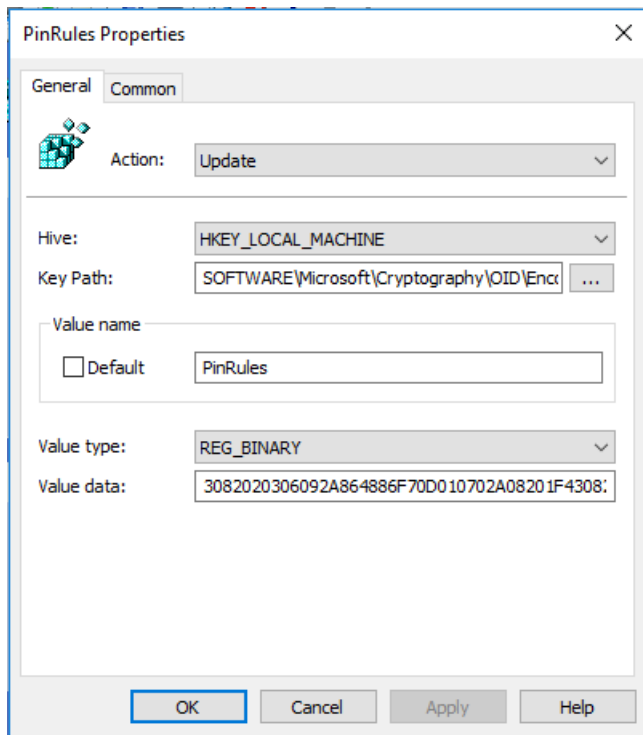
Sign-in to the reference computer using domain administrator equivalent credentials.

1. Start the **Group Policy Management Console** (`gpmc.msc`)
2. In the navigation pane, expand the forest node and then expand the domain node.
3. Expand the node that contains your Active Directory's domain name
4. Select the **Group Policy objects** node. Right-click the **Group Policy objects** node and click **New**.
5. In the **New GPO** dialog box, type *Enterprise Certificate Pinning Rules* in the **Name** text box and click **OK**.
6. In the content pane, right-click the **Enterprise Certificate Pinning Rules** Group Policy object and click **Edit**.

- In the **Group Policy Management Editor**, in the navigation pane, expand the **Preferences** node under **Computer Configuration**. Expand **Windows Settings**.
- Right-click the **Registry** node and click **New**.
- In the **New Registry Properties** dialog box, select **Update** from the **Action** list. Select **HKEY_LOCAL_MACHINE** from the **Hive** list.
- For the **Key Path**, click ... to launch the **Registry Item Browser**. Navigate to the following registry key and select the **PinRules** registry value name:

HKLM\SOFTWARE\Microsoft\Cryptography\OID\EncodingType0\CertDllCreateCertificateChainEngine\Config

Click **Select** to close the **Registry Item Browser**.
- The **Key Path** should contain the selected registry key. The **Value name** configuration should contain the registry value name *PinRules*. **Value type** should read **REG_BINARY** and **Value data** should contain a long series of numbers from 0-9 and letters ranging from A-F (hexadecimal). Click **OK** to save your settings and close the dialog box.



- Close the **Group Policy Management Editor** to save your settings.
- Link the **Enterprise Certificate Pinning Rules** Group Policy object to apply to computers that run Windows 10, version 1703 in your enterprise. When these domain-joined computers apply Group Policy, the registry information configured in the Group Policy object is applied to the computer.

Additional Pin Rules Logging

To assist in constructing certificate pinning rules, you can configure the **PinRulesLogDir** setting under the certificate chain configuration registry key to include a parent directory to log pin rules.

NAME	VALUE
Key	HKLM\SOFTWARE\Microsoft\Cryptography\OID\EncodingType0\CertDllCreateCertificateChainEngine\Config

NAME	VALUE
Name	PinRulesLogDir
Value	The Parent directory where Windows should write the additional pin rule logs
Data type	REG_SZ

Permission for the Pin Rule Log Folder

The folder in which Windows writes the additional pin rule logs must have permissions so that all users and applications have full access. You can run the following commands from an elevated command prompt to achieve the proper permissions.

```
set PinRulesLogDir=c:\PinRulesLog
mkdir %PinRulesLogDir%
icacls %PinRulesLogDir% /grant *S-1-15-2-1:(OI)(CI)(F)
icacls %PinRulesLogDir% /grant *S-1-1-0:(OI)(CI)(F)
icacls %PinRulesLogDir% /grant *S-1-5-12:(OI)(CI)(F)
icacls %PinRulesLogDir% /inheritance:e /setintegritylevel (OI)(CI)L
```

Whenever an application verifies a TLS/SSL certificate chain that contains a server name matching a DNS name in the server certificate, Windows writes a .p7b file consisting of all the certificates in the server's chain to one of three child folders:

- AdminPinRules Matched a site in the enterprise certificate pinning rules.
- AutoUpdatePinRules Matched a site in the certificate pinning rules managed by Microsoft.
- NoPinRules Didn't match any site in the certificate pin rules.

The output file name consists of the leading eight ASCII hex digits of the root's SHA1 thumbprint followed by the server name. For example:

- D4DE20D0_xsi.outlook.com.p7b
- DE28F4A4_www.yammer.com.p7b

If there's either an enterprise certificate pin rule or a Microsoft certificate pin rule mismatch, then Windows writes the .p7b file to the **MismatchPinRules** child folder. If the pin rules have expired, then Windows writes the .p7b to the **ExpiredPinRules** child folder.

Representing a Date in XML

Many attributes within the pin rules xml file are dates.

These dates must be properly formatted and represented in UTC.

You can use Windows PowerShell to format these dates.

You can then copy and paste the output of the cmdlet into the XML file.

```
Windows PowerShell
PS C:\>
PS C:\>
PS C:\>
PS C:\> $xmlDate = get-date -Month 5 -Day 11 -Year 2015 -Hour 0 -Minute 0 -Second 0
PS C:\> $xmlDate
Monday, May 11, 2015 12:00:00 AM
PS C:\> [system.xml.xmlconvert]::ToString($xmlDate, [system.xml.XmlDateTimeSerializationMode]::Utc)
2015-05-11T07:00:00.2655691Z
PS C:\>
```

For simplicity, you can truncate decimal point (.) and the numbers after it. However, be certain to append the uppercase "Z" to the end of the XML date string.

```
2015-05-11T07:00:00.2655691Z  
2015-05-11T07:00:00Z
```

Converting an XML Date

You can also use Windows PowerShell to validate and convert an XML date into a human readable date to validate it's the correct date.

```
Windows PowerShell  
PS C:\>  
PS C:\>  
PS C:\>  
PS C:\> [system.xml.xml]convert>::ToDateTime("2015-05-11T07:00:00Z", [system.xml.xml]DateTimeSerializationMode)::Local)  
Monday, May 11, 2015 12:00:00 AM  
PS C:\>
```

Representing a Duration in XML

Some elements may be configured to use a duration rather than a date. You must represent the duration as an XML timespan data type. You can use Windows PowerShell to properly format and validate durations (timespans) and copy and paste them into your XML file.

Windows PowerShell

```
PS C:\>  
PS C:\>  
PS C:\>  
PS C:\> $ts = New-TimeSpan -Days 45  
PS C:\> $ts  
  
Days : 45  
Hours : 0  
Minutes : 0  
Seconds : 0  
Milliseconds : 0  
Ticks : 3888000000000  
TotalDays : 45  
TotalHours : 1080  
TotalMinutes : 64800  
TotalSeconds : 3888000  
TotalMilliseconds : 3888000000  
  
PS C:\> [system.xml.xml]convert>::ToString($ts)  
P45D  
PS C:\>
```

Converting an XML Duration

You can convert an XML formatted timespan into a timespan variable that you can read.

```
Windows PowerShell
PS C:\>
PS C:\>
PS C:\>
PS C:\> [system.xml.xmlconvert]::ToTimeSpan("P45D")

Days           : 45
Hours          : 0
Minutes        : 0
Seconds        : 0
Milliseconds   : 0
Ticks          : 3888000000000000
TotalDays      : 45
TotalHours     : 1080
TotalMinutes   : 64800
TotalSeconds   : 3888000
TotalMilliseconds : 3888000000
```

Certificate Trust List XML Schema Definition (XSD)

```

<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="PinRules">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="PinRule" maxOccurs="unbounded" minOccurs="1">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="Certificate" maxOccurs="unbounded" minOccurs="0">
                <xs:complexType>
                  <xs:simpleContent>
                    <xs:extension base="xs:string">
                      <xs:attribute type="xs:dateTime" name="EndDate" use="optional"/>
                      <xs:attribute type="xs:string" name="File" use="optional"/>
                      <xs:attribute type="xs:string" name="Directory" use="optional"/>
                      <xs:attribute type="xs:base64Binary" name="Base64" use="optional"/>
                    </xs:extension>
                  </xs:simpleContent>
                </xs:complexType>
              </xs:element>
              <xs:element name="Site" maxOccurs="unbounded" minOccurs="1">
                <xs:complexType>
                  <xs:simpleContent>
                    <xs:extension base="xs:string">
                      <xs:attribute type="xs:string" name="Domain"/>
                      <xs:attribute type="xs:boolean" name="AllSubdomains" use="optional" default="false"/>
                    </xs:extension>
                  </xs:simpleContent>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
            <xs:attribute type="xs:string" name="Name"/>
            <xs:attribute name="Error" use="optional" default="None">
              <xs:simpleType>
                <xs:restriction base="xs:string">
                  <xs:enumeration value="Revoked"/>
                  <xs:enumeration value="InvalidName"/>
                  <xs:enumeration value="None"/>
                </xs:restriction>
              </xs:simpleType>
            </xs:attribute>
            <xs:attribute type="xs:boolean" name="Log" use="optional" default="true"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
      <xs:attribute type="xs:duration" name="Duration" use="optional"/>
      <xs:attribute type="xs:duration" name="LogDuration" use="optional"/>
      <xs:attribute type="xs:dateTime" name="NextUpdate" use="optional"/>
      <xs:attribute type="xs:dateTime" name="LogEndDate" use="optional"/>
      <xs:attribute type="xs:string" name="ListIdentifier" use="optional"/>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Protect derived domain credentials with Windows Defender Credential Guard

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016
- Windows Server 2019

Introduced in Windows 10 Enterprise and Windows Server 2016, Windows Defender Credential Guard uses virtualization-based security to isolate secrets so that only privileged system software can access them. Unauthorized access to these secrets can lead to credential theft attacks, such as Pass-the-Hash or Pass-The-Ticket. Windows Defender Credential Guard prevents these attacks by protecting NTLM password hashes, Kerberos Ticket Granting Tickets, and credentials stored by applications as domain credentials.

By enabling Windows Defender Credential Guard, the following features and solutions are provided:

- **Hardware security** NTLM, Kerberos, and Credential Manager take advantage of platform security features, including Secure Boot and virtualization, to protect credentials.
- **Virtualization-based security** Windows NTLM and Kerberos derived credentials and other secrets run in a protected environment that is isolated from the running operating system.
- **Better protection against advanced persistent threats** When Credential Manager domain credentials, NTLM, and Kerberos derived credentials are protected using virtualization-based security, the credential theft attack techniques and tools used in many targeted attacks are blocked. Malware running in the operating system with administrative privileges cannot extract secrets that are protected by virtualization-based security. While Windows Defender Credential Guard is a powerful mitigation, persistent threat attacks will likely shift to new attack techniques and you should also incorporate other security strategies and architectures.

Related topics

- [Protecting network passwords with Windows Defender Credential Guard](#)
- [Enabling Strict KDC Validation in Windows Kerberos](#)
- [What's New in Kerberos Authentication for Windows Server 2012](#)
- [Authentication Mechanism Assurance for AD DS in Windows Server 2008 R2 Step-by-Step Guide](#)
- [Trusted Platform Module](#)

How Windows Defender Credential Guard works

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016
- Windows Server 2019

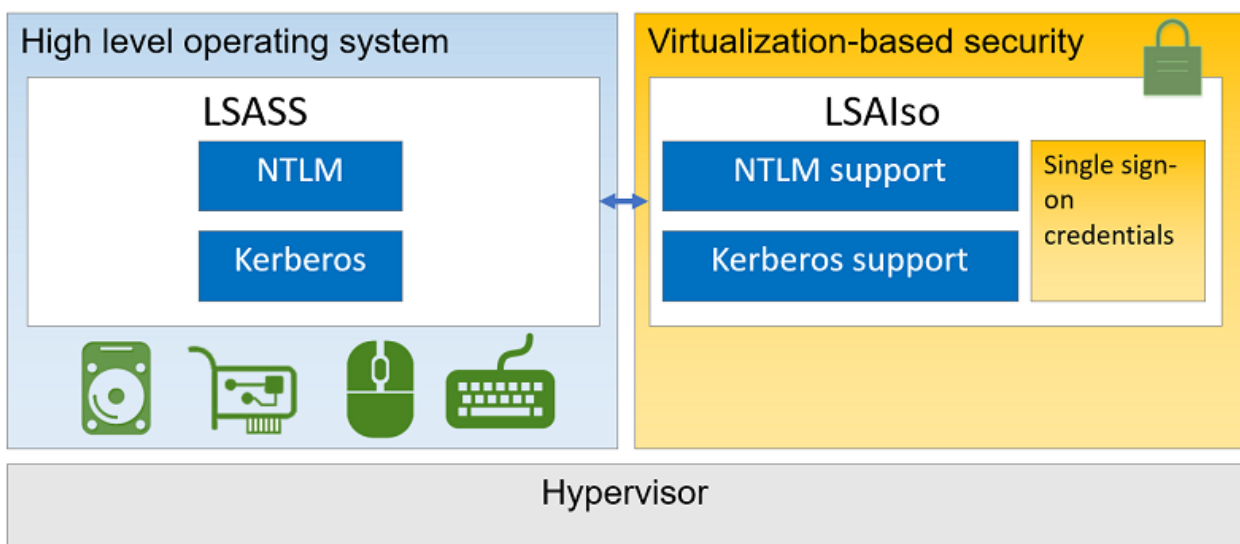
Kerberos, NTLM, and Credential manager isolate secrets by using virtualization-based security. Previous versions of Windows stored secrets in the Local Security Authority (LSA). Prior to Windows 10, the LSA stored secrets used by the operating system in its process memory. With Windows Defender Credential Guard enabled, the LSA process in the operating system talks to a new component called the isolated LSA process that stores and protects those secrets. Data stored by the isolated LSA process is protected using Virtualization-based security and is not accessible to the rest of the operating system. LSA uses remote procedure calls to communicate with the isolated LSA process.

For security reasons, the isolated LSA process doesn't host any device drivers. Instead, it only hosts a small subset of operating system binaries that are needed for security and nothing else. All of these binaries are signed with a certificate that is trusted by virtualization-based security and these signatures are validated before launching the file in the protected environment.

When Windows Defender Credential Guard is enabled, NTLMv1, MS-CHAPv2, Digest, and CredSSP cannot use the signed-in credentials. Thus, single sign-on does not work with these protocols. However, applications can prompt for credentials or use credentials stored in the Windows Vault, which are not protected by Windows Defender Credential Guard with any of these protocols. It is recommended that valuable credentials, such as the sign-in credentials, are not to be used with any of these protocols. If these protocols must be used by domain or Azure AD users, secondary credentials should be provisioned for these use cases.

When Windows Defender Credential Guard is enabled, Kerberos does not allow unconstrained Kerberos delegation or DES encryption, not only for signed-in credentials, but also prompted or saved credentials.

Here's a high-level overview on how the LSA is isolated by using Virtualization-based security:



See also

Related videos

[What is Virtualization-based security?](#)

Windows Defender Credential Guard: Requirements

7/1/2022 • 8 minutes to read • [Edit Online](#)

Applies to

- Windows 11
- Windows 10
- Windows Server 2019
- Windows Server 2016

For Windows Defender Credential Guard to provide protection, the computers you are protecting must meet certain baseline hardware, firmware, and software requirements, which we will refer to as [Hardware and software requirements](#). Additionally, Windows Defender Credential Guard blocks specific authentication capabilities, so applications that require such capabilities will break. We will refer to these requirements as [Application requirements](#). Beyond these requirements, computers can meet additional hardware and firmware qualifications, and receive additional protections. Those computers will be more hardened against certain threats. For detailed information on baseline protections, plus protections for improved security that are associated with hardware and firmware options available in 2015, 2016, and 2017, refer to the tables in [Security Considerations](#).

Hardware and software requirements

To provide basic protections against OS level attempts to read Credential Manager domain credentials, NTLM and Kerberos derived credentials, Windows Defender Credential Guard uses:

- Support for Virtualization-based security (required)
- Secure boot (required)
- Trusted Platform Module (TPM, preferred - provides binding to hardware) versions 1.2 and 2.0 are supported, either discrete or firmware
- UEFI lock (preferred - prevents attacker from disabling with a simple registry key change)

The Virtualization-based security requires:

- 64-bit CPU
- CPU virtualization extensions plus extended page tables
- Windows hypervisor (does not require Hyper-V Windows Feature to be installed)

Windows Defender Credential Guard deployment in virtual machines

Credential Guard can protect secrets in a Hyper-V virtual machine, just as it would on a physical machine. When Credential Guard is deployed on a VM, secrets are protected from attacks inside the VM. Credential Guard does not provide additional protection from privileged system attacks originating from the host.

Requirements for running Windows Defender Credential Guard in Hyper-V virtual machines

- The Hyper-V host must have an IOMMU, and run at least Windows Server 2016 or Windows 10 version 1607.
- The Hyper-V virtual machine must be Generation 2, have an enabled virtual TPM, and be running at least Windows Server 2016 or Windows 10.
 - TPM is not a requirement, but we recommend that you implement TPM.

For information about other host platforms, see [Enabling Windows Server 2016 and Hyper-V virtualization](#)

[based security features on other platforms.](#)

For information about Windows Defender Remote Credential Guard hardware and software requirements, see [Windows Defender Remote Credential Guard requirements.](#)

Application requirements

When Windows Defender Credential Guard is enabled, specific authentication capabilities are blocked, so applications that require such capabilities will break. Applications should be tested prior to deployment to ensure compatibility with the reduced functionality.

WARNING

Enabling Windows Defender Credential Guard on domain controllers is not supported. The domain controller hosts authentication services which integrate with processes isolated when Windows Defender Credential Guard is enabled, causing crashes.

NOTE

Windows Defender Credential Guard does not provide protections for the Active Directory database or the Security Accounts Manager (SAM). The credentials protected by Kerberos and NTLM when Windows Defender Credential Guard is enabled are also in the Active Directory database (on domain controllers) and the SAM (for local accounts).

Applications will break if they require:

- Kerberos DES encryption support
- Kerberos unconstrained delegation
- Extracting the Kerberos TGT
- NTLMv1

Applications will prompt and expose credentials to risk if they require:

- Digest authentication
- Credential delegation
- MS-CHAPv2

Applications may cause performance issues when they attempt to hook the isolated Windows Defender Credential Guard process.

Services or protocols that rely on Kerberos, such as file shares, remote desktop, or BranchCache, continue to work and are not affected by Windows Defender Credential Guard.

Security considerations

All computers that meet baseline protections for hardware, firmware, and software can use Windows Defender Credential Guard. Computers that meet additional qualifications can provide additional protections to further reduce the attack surface. The following tables describe baseline protections, plus protections for improved security that are associated with hardware and firmware options available in 2015, 2016, and 2017.

NOTE

Beginning with Windows 10, version 1607, Trusted Platform Module (TPM 2.0) must be enabled by default on new shipping computers.

If you are an OEM, see [PC OEM requirements for Windows Defender Credential Guard](#).

Baseline protections

BASELINE PROTECTIONS	DESCRIPTION	SECURITY BENEFITS
Hardware: 64-bit CPU	A 64-bit computer is required for the Windows hypervisor to provide VBS.	
Hardware: CPU virtualization extensions, plus extended page tables	Requirements: - These hardware features are required for VBS: One of the following virtualization extensions: - VT-x (Intel) or - AMD-V And: - Extended page tables, also called Second Level Address Translation (SLAT).	VBS provides isolation of secure kernel from normal operating system. Vulnerabilities and Day 0s in normal operating system cannot be exploited because of this isolation.
Hardware: Trusted Platform Module (TPM)	Requirement: - TPM 1.2 or TPM 2.0, either discrete or firmware. TPM recommendations	A TPM provides protection for VBS encryption keys that are stored in the firmware. TPM helps protect against attacks involving a physically present user with BIOS access.
Firmware: UEFI firmware version 2.3.1.c or higher with UEFI Secure Boot	Requirements: - See the following Windows Hardware Compatibility Program requirement: System.Fundamentals.Firmware.UEFIsecureBoot	UEFI Secure Boot helps ensure that the device boots only authorized code, and can prevent boot kits and root kits from installing and persisting across reboots.
Firmware: Secure firmware update process	Requirements: - UEFI firmware must support secure firmware update found under the following Windows Hardware Compatibility Program requirement: System.Fundamentals.Firmware.UEFIsecureBoot.	UEFI firmware just like software can have security vulnerabilities that, when found, need to be patched through firmware updates. Patching helps prevent root kits from getting installed.
Software: Qualified Windows operating system	Requirement: - At least Windows 10 Enterprise or Windows Server 2016.	Support for VBS and for management features that simplify configuration of Windows Defender Credential Guard.

IMPORTANT

Windows Server 2016 running as a domain controller does not support Windows Defender Credential Guard.

IMPORTANT

The following tables list additional qualifications for improved security. We strongly recommend meeting the additional qualifications to significantly strengthen the level of security that Windows Defender Credential Guard can provide.

Technical Preview 4

PROTECTIONS FOR IMPROVED SECURITY	DESCRIPTION
Hardware: IOMMU (input/output memory management unit)	<p>Requirement:</p> <ul style="list-style-type: none"> - VT-D or AMD Vi IOMMU <p>Security benefits:</p> <ul style="list-style-type: none"> - An IOMMU can enhance system resiliency against memory attacks. For more information, see Advanced Configuration and Power Interface (ACPI) description tables
Firmware: Securing Boot Configuration and Management	<p>Requirements:</p> <ul style="list-style-type: none"> - BIOS password or stronger authentication must be supported. - In the BIOS configuration, BIOS authentication must be set. - There must be support for protected BIOS option to configure list of permitted boot devices (for example, "Boot only from internal hard drive") and boot device order, overriding BOOTORDER modification made by operating system. - In the BIOS configuration, BIOS options related to security and boot options (list of permitted boot devices, boot order) must be secured to prevent other operating systems from starting and to prevent changes to the BIOS settings.
Firmware: Secure MOR, revision 2 implementation	<p>Requirement:</p> <ul style="list-style-type: none"> - Secure MOR, revision 2 implementation

2016 Additional security qualifications starting with Windows 10, version 1607, and Windows Server 2016

IMPORTANT

The following tables list additional qualifications for improved security. Systems that meet these additional qualifications can provide more protections.

PROTECTIONS FOR IMPROVED SECURITY	DESCRIPTION	SECURITY BENEFITS
Firmware: Hardware Rooted Trust Platform Secure Boot	<p>Requirements:</p> <ul style="list-style-type: none"> - Boot Integrity (Platform Secure Boot) must be supported. See the Windows Hardware Compatibility Program requirements under System.Fundamentals.Firmware.CS.UEFI SecureBoot.ConnectedStandby - The Hardware Security Test Interface (HSTI) must be implemented. See Hardware Security Testability Specification. 	<p>Boot Integrity (Platform Secure Boot) from Power-On provides protections against physically present attackers, and defense-in-depth against malware.</p> <ul style="list-style-type: none"> - HSTI provides additional security assurance for correctly secured silicon and platform.
Firmware: Firmware Update through Windows Update	<p>Requirements:</p> <ul style="list-style-type: none"> - Firmware must support field updates through Windows Update and UEFI encapsulation update. 	<p>Helps ensure that firmware updates are fast, secure, and reliable.</p>

PROTECTIONS FOR IMPROVED SECURITY	DESCRIPTION	SECURITY BENEFITS
Firmware: Securing Boot Configuration and Management	Requirements: <ul style="list-style-type: none"> - Required BIOS capabilities: Ability of OEM to add ISV, OEM, or Enterprise Certificate in Secure Boot DB at manufacturing time. - Required configurations: Microsoft UEFI CA must be removed from Secure Boot DB. Support for 3rd-party UEFI modules is permitted but should leverage ISV-provided certificates or OEM certificate for the specific UEFI software. 	<ul style="list-style-type: none"> - Enterprises can choose to allow proprietary EFI drivers/applications to run. - Removing Microsoft UEFI CA from Secure Boot DB provides full control to enterprises over software that runs before the operating system boots.

2017 Additional security qualifications starting with Windows 10, version 1703

The following table lists qualifications for Windows 10, version 1703, which are in addition to all preceding qualifications.

PROTECTIONS FOR IMPROVED SECURITY	DESCRIPTION	SECURITY BENEFITS
Firmware: VBS enablement of No-Execute (NX) protection for UEFI runtime services	Requirements: <ul style="list-style-type: none"> - VBS will enable NX protection on UEFI runtime service code and data memory regions. UEFI runtime service code must support read-only page protections, and UEFI runtime service data must not be executable. UEFI runtime service must meet these requirements: <ul style="list-style-type: none"> - Implement UEFI 2.6 EFI_MEMORY_ATTRIBUTES_TABLE. All UEFI runtime service memory (code and data) must be described by this table. - PE sections must be page-aligned in memory (not required for in non-volatile storage). - The Memory Attributes Table needs to correctly mark code and data as RO/NX for configuration by the OS: <ul style="list-style-type: none"> - All entries must include attributes EFI_MEMORY_RO, EFI_MEMORY_XP, or both. - No entries may be left with neither of the above attributes, indicating memory that is both executable and writable. Memory must be either readable and executable or writable and non-executable. <p>(SEE IMPORTANT INFORMATION AFTER THIS TABLE)</p>	<p>Vulnerabilities in UEFI runtime, if any, will be blocked from compromising VBS (such as in functions like UpdateCapsule and SetVariable)</p> <ul style="list-style-type: none"> - Reduces the attack surface to VBS from system firmware.

PROTECTIONS FOR IMPROVED SECURITY	DESCRIPTION	SECURITY BENEFITS
Firmware: Firmware support for SMM protection	Requirements: - The Windows SMM Security Mitigations Table (WSMT) specification contains details of an ACPI table that was created for use with Windows operating systems that support Windows virtualization-based security (VBS) features.	<ul style="list-style-type: none"> - Protects against potential vulnerabilities in UEFI runtime services, if any, will be blocked from compromising VBS (such as in functions like UpdateCapsule and SetVariable) - Reduces the attack surface to VBS from system firmware. - Blocks additional security attacks against SMM.

IMPORTANT

Regarding VBS enablement of NX protection for UEFI runtime services:

- This only applies to UEFI runtime service memory, and not UEFI boot service memory.
- This protection is applied by VBS on OS page tables.

Please also note the following:

- Do not use sections that are both writable and executable
- Do not attempt to directly modify executable system memory
- Do not use dynamic code

Manage Windows Defender Credential Guard

7/1/2022 • 9 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016
- Windows Server 2019
- Windows Server 2022

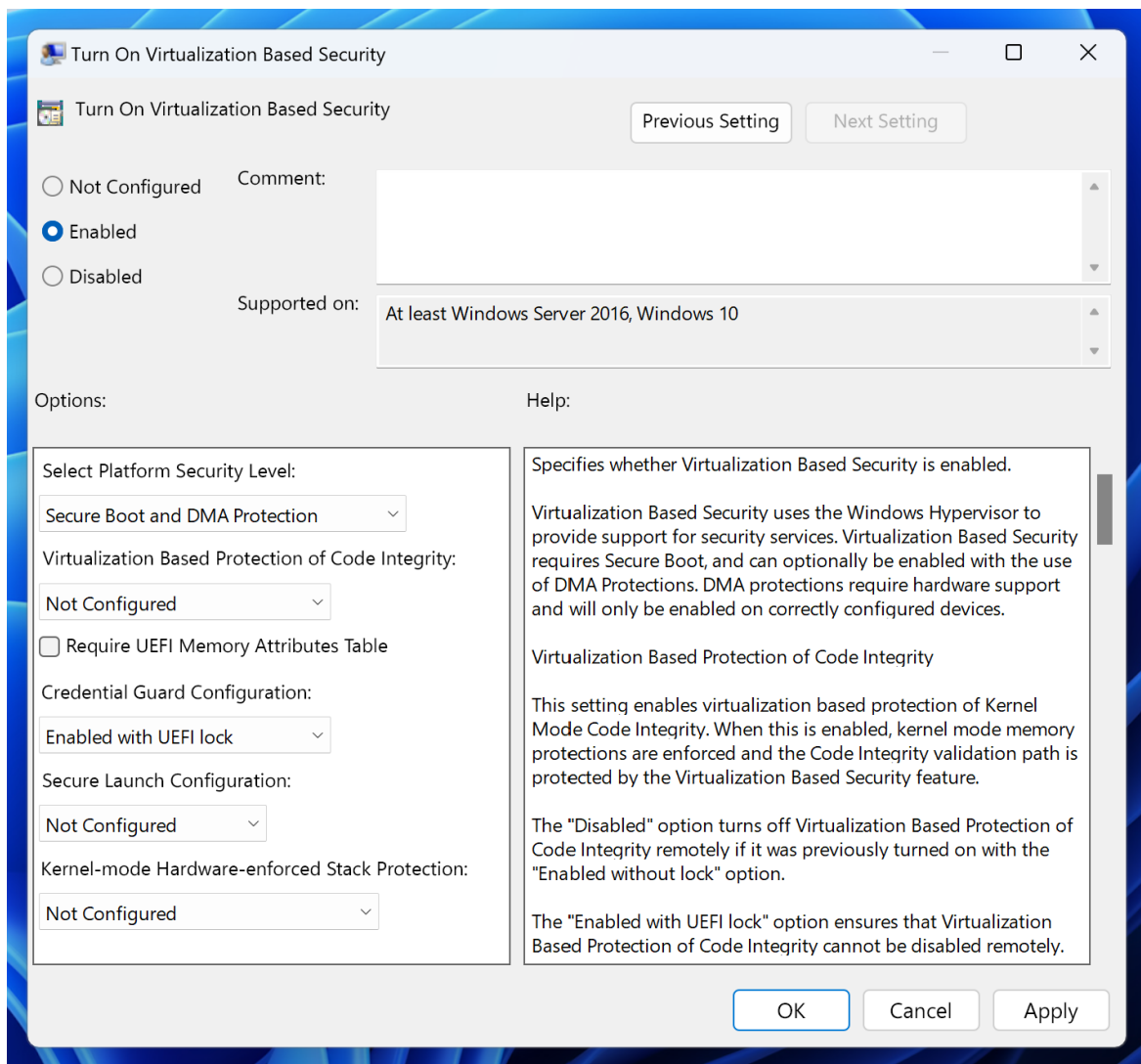
Enable Windows Defender Credential Guard

Windows Defender Credential Guard can be enabled either by using [Group Policy](#), the [registry](#), or the [Hypervisor-Protected Code Integrity \(HVCI\) and Windows Defender Credential Guard hardware readiness tool](#). Windows Defender Credential Guard can also protect secrets in a Hyper-V virtual machine, just as it would on a physical machine. The same set of procedures used to enable Windows Defender Credential Guard on physical machines applies also to virtual machines.

Enable Windows Defender Credential Guard by using Group Policy

You can use Group Policy to enable Windows Defender Credential Guard. This will add and enable the virtualization-based security features for you if needed.

1. From the Group Policy Management Console, go to **Computer Configuration > Administrative Templates > System > Device Guard**.
2. Select **Turn On Virtualization Based Security**, and then select the **Enabled** option.
3. In the **Select Platform Security Level** box, choose **Secure Boot** or **Secure Boot and DMA Protection**.
4. In the **Credential Guard Configuration** box, select **Enabled with UEFI lock**. If you want to be able to turn off Windows Defender Credential Guard remotely, choose **Enabled without lock**.
5. In the **Secure Launch Configuration** box, choose **Not Configured**, **Enabled** or **Disabled**. For more information, see [System Guard Secure Launch and SMM protection](#).



6. Select **OK**, and then close the Group Policy Management Console.

To enforce processing of the group policy, you can run `gpupdate /force`.

Enable Windows Defender Credential Guard by using Intune

1. From **Home**, select **Microsoft Intune**.
2. Select **Device configuration**.
3. Select **Profiles > Create Profile > Endpoint protection > Windows Defender Credential Guard**.

NOTE

It will enable VBS and Secure Boot and you can do it with or without UEFI Lock. If you will need to disable Credential Guard remotely, enable it without UEFI lock.

TIP

You can also configure Credential Guard by using an account protection profile in endpoint security. For more information, see [Account protection policy settings for endpoint security in Intune](#).

Enable Windows Defender Credential Guard by using the registry

If you don't use Group Policy, you can enable Windows Defender Credential Guard by using the registry. Windows Defender Credential Guard uses virtualization-based security features which have to be enabled first on some operating systems.

Add the virtualization-based security features

Starting with Windows 10, version 1607 and Windows Server 2016, enabling Windows features to use virtualization-based security is not necessary and this step can be skipped.

If you are using Windows 10, version 1507 (RTM) or Windows 10, version 1511, Windows features have to be enabled to use virtualization-based security. You can do this by using either the Control Panel or the Deployment Image Servicing and Management tool (DISM).

NOTE

If you enable Windows Defender Credential Guard by using Group Policy, the steps to enable Windows features through Control Panel or DISM are not required. Group Policy will install Windows features for you.

Add the virtualization-based security features by using Programs and Features

1. Open the Programs and Features control panel.
2. Select **Turn Windows feature on or off**.
3. Go to **Hyper-V > Hyper-V Platform**, and then select the **Hyper-V Hypervisor** check box.
4. Select the **Isolated User Mode** check box at the top level of the feature selection.
5. Select **OK**.

Add the virtualization-based security features to an offline image by using DISM

1. Open an elevated command prompt.
2. Add the Hyper-V Hypervisor by running the following command:

```
dism /image:<WIM file name> /Enable-Feature /FeatureName:Microsoft-Hyper-V-Hypervisor /all
```

3. Add the Isolated User Mode feature by running the following command:

```
dism /image:<WIM file name> /Enable-Feature /FeatureName:IsolatedUserMode
```

NOTE

In Windows 10, version 1607 and later, the Isolated User Mode feature has been integrated into the core operating system. Running the command in step 3 above is therefore no longer required.

TIP

You can also add these features to an online image by using either DISM or Configuration Manager.

Enable virtualization-based security and Windows Defender Credential Guard

1. Open Registry Editor.
2. Enable virtualization-based security:
 - a. Go to `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\DeviceGuard`.
 - b. Add a new DWORD value named **EnableVirtualizationBasedSecurity**. Set the value of this registry setting to 1 to enable virtualization-based security and set it to 0 to disable it.
 - c. Add a new DWORD value named **RequirePlatformSecurityFeatures**. Set the value of this registry setting to 1 to use **Secure Boot** only or set it to 3 to use **Secure Boot and DMA**

protection.

3. Enable Windows Defender Credential Guard:

- a. Go to `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa`.
- b. Add a new DWORD value named **LsaCfgFlags**. Set the value of this registry setting to 1 to enable Windows Defender Credential Guard with UEFI lock, set it to 2 to enable Windows Defender Credential Guard without lock, and set it to 0 to disable it.

4. Close Registry Editor.

NOTE

You can also enable Windows Defender Credential Guard by setting the registry entries in the [FirstLogonCommands](#) unattend setting.

Enable Windows Defender Credential Guard by using the HVCI and Windows Defender Credential Guard hardware readiness tool

You can also enable Windows Defender Credential Guard by using the [HVCI and Windows Defender Credential Guard hardware readiness tool](#).

```
DG_Readiness_Tool.ps1 -Enable -AutoReboot
```

IMPORTANT

When running the HVCI and Windows Defender Credential Guard hardware readiness tool on a non-English operating system, within the script, change `$OSArch = $(gwmi win32_operatingsystem).OSArchitecture` to be `$OSArch = $($gwmi win32_operatingsystem).OSArchitecture.ToLower()` instead, in order for the tool to work.

This is a known issue.

Review Windows Defender Credential Guard performance

Is Windows Defender Credential Guard running?

You can view System Information to check that Windows Defender Credential Guard is running on a PC.

1. Select **Start**, type `msinfo32.exe`, and then select **System Information**.
2. Select **System Summary**.
3. Confirm that **Credential Guard** is shown next to **Virtualization-based security Services Running**.

Virtualization-based security	Running
Virtualization-based security Required Security Properties	Base Virtualization Support, Secure Boot, DMA Protection
Virtualization-based security Available Security Properties	Base Virtualization Support, Secure Boot, DMA Protection, Mode Based ...
Virtualization-based security Services Configured	Credential Guard, Hypervisor enforced Code Integrity
Virtualization-based security Services Running	Credential Guard, Hypervisor enforced Code Integrity

You can also check that Windows Defender Credential Guard is running by using the [HVCI and Windows Defender Credential Guard hardware readiness tool](#).

```
DG_Readiness_Tool_v3.6.ps1 -Ready
```

IMPORTANT

When running the HVCI and Windows Defender Credential Guard hardware readiness tool on a non-English operating system, within the script, change `*$OSArch = $(gwmi win32_operatingsystem).OSArchitecture` to be `$OSArch = $($((gwmi win32_operatingsystem).OSArchitecture).tolower())` instead, in order for the tool to work.

This is a known issue.

NOTE

For client machines that are running Windows 10 1703, Lsalso.exe is running whenever virtualization-based security is enabled for other features.

- We recommend enabling Windows Defender Credential Guard before a device is joined to a domain. If Windows Defender Credential Guard is enabled after domain join, the user and device secrets may already be compromised. In other words, enabling Credential Guard will not help to secure a device or identity that has already been compromised, which is why we recommend turning on Credential Guard as early as possible.
- You should perform regular reviews of the PCs that have Windows Defender Credential Guard enabled. This can be done with security audit policies or WMI queries. Here's a list of WinInit event IDs to look for:
 - **Event ID 13** Windows Defender Credential Guard (Lsalso.exe) was started and will protect LSA credentials.
 - **Event ID 14** Windows Defender Credential Guard (Lsalso.exe) configuration: [0x0 | 0x1 | 0x2], 0
 - The first variable: **0x1** or **0x2** means that Windows Defender Credential Guard is configured to run. **0x0** means that it's not configured to run.
 - The second variable: **0** means that it's configured to run in protect mode. **1** means that it's configured to run in test mode. This variable should always be **0**.
 - **Event ID 15** Windows Defender Credential Guard (Lsalso.exe) is configured but the secure kernel is not running; continuing without Windows Defender Credential Guard.
 - **Event ID 16** Windows Defender Credential Guard (Lsalso.exe) failed to launch: [error code]
 - **Event ID 17** Error reading Windows Defender Credential Guard (Lsalso.exe) UEFI configuration: [error code]
- You can also verify that TPM is being used for key protection by checking **Event ID 51** in *Applications and Services logs > Microsoft > Windows > Kernel-Boot* event log. The full event text will read like this:

```
VSM Master Encryption Key Provisioning. Using cached copy status: 0x0. Unsealing cached copy status: 0x1. New key generation status: 0x1. Sealing status: 0x1. TPM PCR mask: 0x0.
```

If you are running with a TPM, the TPM PCR mask value will be something other than 0.
- You can use Windows PowerShell to determine whether credential guard is running on a client computer. On the computer in question, open an elevated PowerShell window and run the following command:

```
(Get-CimInstance -ClassName Win32_DeviceGuard -Namespace root\Microsoft\Windows\DeviceGuard).SecurityServicesRunning
```

This command generates the following output:

- **0**: Windows Defender Credential Guard is disabled (not running)

- o 1: Windows Defender Credential Guard is enabled (running)

NOTE

Checking the task list or Task Manager to see if LSAISO.exe is running is not a recommended method for determining whether Windows Defender Credential Guard is running.

Disable Windows Defender Credential Guard

To disable Windows Defender Credential Guard, you can use the following set of procedures or the [HVCI and Windows Defender Credential Guard hardware readiness tool](#). If Credential Guard was enabled with UEFI Lock then you must use the following procedure as the settings are persisted in EFI (firmware) variables and it will require physical presence at the machine to press a function key to accept the change. If Credential Guard was enabled without UEFI Lock then you can turn it off by using Group Policy.

1. If you used Group Policy, disable the Group Policy setting that you used to enable Windows Defender Credential Guard (**Computer Configuration > Administrative Templates > System > Device Guard > Turn on Virtualization Based Security**).

2. Delete the following registry settings:

- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\LsaCfgFlags
- HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\DeviceGuard\LsaCfgFlags

3. If you also wish to disable virtualization-based security delete the following registry settings:

- HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\DeviceGuard\EnableVirtualizationBasedSecurity
- HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\DeviceGuard\RequirePlatformSecurityFeatures

IMPORTANT

If you manually remove these registry settings, make sure to delete them all. If you don't remove them all, the device might go into BitLocker recovery.

4. Delete the Windows Defender Credential Guard EFI variables by using bcdedit. From an elevated command prompt, type the following commands:

```
mountvol X: /s
copy %WINDIR%\System32\SecConfig.efi X:\EFI\Microsoft\Boot\SecConfig.efi /Y
bcdedit /create {0cb3b571-2f2e-4343-a879-d86a476d7215} /d "DebugTool" /application osloader
bcdedit /set {0cb3b571-2f2e-4343-a879-d86a476d7215} path "\EFI\Microsoft\Boot\SecConfig.efi"
bcdedit /set {bootmgr} bootsequence {0cb3b571-2f2e-4343-a879-d86a476d7215}
bcdedit /set {0cb3b571-2f2e-4343-a879-d86a476d7215} loadoptions DISABLE-LSA-ISO
bcdedit /set {0cb3b571-2f2e-4343-a879-d86a476d7215} device partition=X:
mountvol X: /d
```

5. Restart the PC.
6. Accept the prompt to disable Windows Defender Credential Guard.
7. Alternatively, you can disable the virtualization-based security features to turn off Windows Defender Credential Guard.

NOTE

The PC must have one-time access to a domain controller to decrypt content, such as files that were encrypted with EFS. If you want to turn off both Windows Defender Credential Guard and virtualization-based security, run the following bcdedit commands after turning off all virtualization-based security Group Policy and registry settings:

```
bcdedit /set {0cb3b571-2f2e-4343-a879-d86a476d7215} loadoptions DISABLE-LSA-ISO,DISABLE-VBS  
bcdedit /set vsmlaunchtype off
```

For more info on virtualization-based security and HVCI, see [Enable virtualization-based protection of code integrity](#).

NOTE

Credential Guard and Device Guard are not supported when using Azure Gen 1 VMs. These options are available with Gen 2 VMs only.

Disable Windows Defender Credential Guard by using the HVCI and Windows Defender Credential Guard hardware readiness tool

You can also disable Windows Defender Credential Guard by using the [HVCI and Windows Defender Credential Guard hardware readiness tool](#).

```
DG_Readiness_Tool_v3.6.ps1 -Disable -AutoReboot
```

IMPORTANT

When running the HVCI and Windows Defender Credential Guard hardware readiness tool on a non-English operating system, within the script, change `*$OSArch = $(gwmi win32_operatingsystem).OSArchitecture` to be `$OSArch = $(gwmi win32_operatingsystem).OSArchitecture.ToLower()` instead, in order for the tool to work.

This is a known issue.

Disable Windows Defender Credential Guard for a virtual machine

From the host, you can disable Windows Defender Credential Guard for a virtual machine:

```
Set-VMSecurity -VMName <VMName> -VirtualizationBasedSecurityOptOut $true
```

Windows Defender Device Guard and Windows Defender Credential Guard hardware readiness tool

7/1/2022 • 19 minutes to read • [Edit Online](#)

Applies to:

- Windows 10
- Windows 11
- Windows Server 2016
- Windows Server 2019
- Windows Server 2022

```
# Script to find out if a machine is Device Guard compliant.
# The script requires a driver verifier present on the system.

param([switch]$Capable, [switch]$Ready, [switch]$Enable, [switch]$Disable, $SIPolicyPath,
[switch]$AutoReboot, [switch]$DG, [switch]$CG, [switch]$HVCI, [switch]$HLK, [switch]$Clear,
[switch]$ResetVerifier)

$path = "C:\DGLogs\"
$logFile = $path + "DeviceGuardCheckLog.txt"

$CompatibleModules = New-Object System.Text.StringBuilder
$FailingModules = New-Object System.Text.StringBuilder
$FailingExecuteWriteCheck = New-Object System.Text.StringBuilder

$DGVerifyCrit = New-Object System.Text.StringBuilder
$DGVerifyWarn = New-Object System.Text.StringBuilder
$DGVerifySuccess = New-Object System.Text.StringBuilder

$Sys32Path = "$env:windir\system32"
$DriverPath = "$env:windir\system32\drivers"

#generated by certutil -encode
$SIPolicy_Encoded = "BQAAAA43RKLJRAZMtVH2AW5WMHbk9wcuTBkgTbfJb0SmxaI0BACNkAgAAAAAAAA
HQAAAAIAAAAAAAAAAKAEAAAAAMAAAAQorBgEEAYI3CgMGDAAAAEKWYBBAGC
NwoDBQwAAAAABcIsGAQQBgc9BAEMAAAAQorBgEEAYI3PQUBDAAAAEKWYBBAGC
NwoDFQwAAAAABcIsGAQQBgdMAwEMAAAAQorBgEEAYI3TAUBDAAAAEKWYBBAGC
N0wLAQEAAAAGAAAAQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AQAAAAIAAAABAAAAAgAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
BgAAAAEAAAADAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAEAAAAGAAAA
AQAAAAEAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAQAAAAUAAAAABAAA
AQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAABAAAABAAAAEAAAAABAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAEAAAAGAAAAQAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAABAAAABAAAAgAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAABAAAABgAAAEAAAADAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAEAAAAGAAAAQAAAEAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAQAAAUAAAABAAAAQAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAABAAAADgAAAEAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAEAAAADAAAAIAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AQAAAA4AAAAABAAAABAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
DgAAAAEAAAABAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAEAAAAOAAAA
AQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAQAAAA4AAAAABAAA
AgAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAABAAAADgAAAEAAAADAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAEAAAADAAAAQAAAAEAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAQAAAAQAAAAABAAAAQAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAIAAAPye3j3MoJGGst0/m30KIFDLGLVN
otvtV8/cu4Xchn4A0AAAUAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
```



```

else
{
    Log $item.FullName + "Not-exempted"
    Log $cert.ToString()
    return 0
}
}

function CheckExemption($_ModName)
{
    $mod1 = Get-ChildItem $Sys32Path $_ModName
    $mod2 = Get-ChildItem $DriverPath $_ModName
    if($mod1)
    {
        Log "NonDriver module" + $mod1.FullName
        return IsExempted($mod1)
    }
    elseif($mod2)
    {
        Log "Driver Module" + $mod2.FullName
        return IsExempted($mod2)
    }
}

function CheckFailedDriver($_ModName, $CIStats)
{
    Log "Module: " $_ModName.Trim()
    if(CheckExemption($_ModName.Trim()) - eq 1)
    {
        $CompatibleModules.AppendLine("Windows Signed: " + $_ModName.Trim()) | Out-Null
        return
    }
    $index = $CIStats.IndexOf("execute pool type count:".ToLower())
    if($index -eq -1)
    {
        return
    }
    $_tempStr = $CIStats.Substring($index)
    $Result = "PASS"
    $separator = "`r`n", ""
    $option = [System.StringSplitOptions]::RemoveEmptyEntries
    $stats = $_tempStr.Split($separator, $option)
    Log $stats.Count

    $FailingStat = ""
    foreach( $stat in $stats)
    {
        $_t = $stat.Split(":")
        if($_t.Count -eq 2 -and $_t[1].trim() -ne "0")
        {
            $Result = "FAIL"
            $FailingStat = $stat
            break
        }
    }
    if($Result.Contains("PASS"))
    {
        $CompatibleModules.AppendLine($_ModName.Trim()) | Out-Null
    }
    elseif($FailingStat.Trim().Contains("execute-write"))
    {
        $FailingExecuteWriteCheck.AppendLine("Module: "+ $_ModName.Trim() + "`r`n`tReason: " +
        $FailingStat.Trim() ) | Out-Null
    }
    else
    {
        $FailingModules.AppendLine("Module: "+ $_ModName.Trim() + "`r`n`tReason: " + $FailingStat.Trim() ) |
        Out-Null
    }
}

```

```

    }
    Log "Result: " $Result
}

function ListCIStats($_ModName, $str1)
{
    $i1 = $str1.IndexOf("Code Integrity Statistics:".ToLower())
    if($i1 -eq -1 )
    {
        Log "String := " $str1
        Log "Warning! CI Stats are missing for " $_ModName
        return
    }
    $temp_str1 = $str1.Substring($i1)
    $CIStats = $temp_str1.Substring(0).Trim()

    CheckFailedDriver $_ModName $CIStats
}

function ListDrivers($str)
{
    $_tempStr= $str

    $separator = "module:", ""
    $option = [System.StringSplitOptions]::RemoveEmptyEntries
    $index1 = $_tempStr.IndexOf("MODULE:".ToLower())
    if($index1 -lt 0)
    {
        return
    }
    $_tempStr = $_tempStr.Substring($Index1)
    $_SplitStr = $_tempStr.Split($separator,$option)

    Log $_SplitStr.Count
    LogAndConsole "Verifying each module please wait ... "
    foreach($ModuleDetail in $_Splitstr)
    {
        #LogAndConsole $Module
        $Index2 = $ModuleDetail.IndexOf("(")
        if($Index2 -eq -1)
        {
            "Skipping .."
            continue
        }
        $ModName = $ModuleDetail.Substring(0,$Index2-1)
        Log "Driver: " $ModName
        Log "Processing module: " $ModName
        ListCIStats $ModName $ModuleDetail
    }

    $DriverScanCompletedMessage = "Completed scan. List of Compatible Modules can be found at " + $LogFile
    LogAndConsole $DriverScanCompletedMessage

    if($FailingModules.Length -gt 0 -or $FailingExecuteWriteCheck.Length -gt 0 )
    {
        $WarningMessage = "Incompatible HVCI Kernel Driver Modules found"
        if($HLK)
        {
            LogAndConsoleError $WarningMessage
        }
        else
        {
            LogAndConsoleWarning $WarningMessage
        }

        LogAndConsoleError $FailingExecuteWriteCheck.ToString()
        if($HLK)
        {

```

```

        LogAndConsoleError $FailingModules.ToString()
    }
    else
    {
        LogAndConsoleWarning $FailingModules.ToString()
    }
    if($FailingModules.Length -ne 0 -or $FailingExecuteWriteCheck.Length -ne 0 )
    {
        if($HLK)
        {
            $DGVerifyCrit.AppendLine($WarningMessage) | Out-Null
        }
        else
        {
            $DGVerifyWarn.AppendLine($WarningMessage) | Out-Null
        }
    }
}
}
else
{
    LogAndConsoleSuccess "No Incompatible Drivers found"
}
}

function ListSummary()
{
    if($DGVerifyCrit.Length -ne 0 )
    {
        LogAndConsoleError "Machine is not Device Guard / Credential Guard compatible because of the
following:"
        LogAndConsoleError $DGVerifyCrit.ToString()
        LogAndConsoleWarning $DGVerifyWarn.ToString()
        if(!$HVCI -and !$DG)
        {
            ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\'
/v "CG_Capable" /t REG_DWORD /d 0 /f '
        }
        if(!$CG)
        {
            ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\'
/v "DG_Capable" /t REG_DWORD /d 0 /f '
            ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\'
/v "HVCI_Capable" /t REG_DWORD /d 0 /f '
        }
    }
    elseif ($DGVerifyWarn.Length -ne 0 )
    {
        LogAndConsoleSuccess "Device Guard / Credential Guard can be enabled on this machine.`n"
        LogAndConsoleWarning "The following additional qualifications, if present, can enhance the security
of Device Guard / Credential Guard on this system:"
        LogAndConsoleWarning $DGVerifyWarn.ToString()
        if(!$HVCI -and !$DG)
        {
            ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\'
/v "CG_Capable" /t REG_DWORD /d 1 /f '
        }
        if(!$CG)
        {
            ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\'
/v "DG_Capable" /t REG_DWORD /d 1 /f '
            ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\'
/v "HVCI_Capable" /t REG_DWORD /d 1 /f '
        }
    }
    else
    {
        LogAndConsoleSuccess "Machine is Device Guard / Credential Guard Ready.`n"
        if(!$HVCI -and !$DG)
    }
}

```

```

    {
        ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\"
/v "CG_Capable" /t REG_DWORD /d 2 /f '
    }
    if(!$CG)
    {
        ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\"
/v "DG_Capable" /t REG_DWORD /d 2 /f '
        ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\"
/v "HVCI_Capable" /t REG_DWORD /d 2 /f '
    }
}
}

```

```
function Instantiate-Kernel32 {
```

```
    try
```

```
    {
```

```
        Add-Type -TypeDefinition @"
```

```
using System;
```

```
using System.Diagnostics;
```

```
using System.Runtime.InteropServices;
```

```
public static class Kernel32
```

```
{
```

```
    [DllImport("kernel32", SetLastError=true, CharSet = CharSet.Ansi)]
```

```
    public static extern IntPtr LoadLibrary(
        [MarshalAs(UnmanagedType.LPStr)]string lpFileName);
```

```
    [DllImport("kernel32", CharSet=CharSet.Ansi, ExactSpelling=true, SetLastError=true)]
```

```
    public static extern IntPtr GetProcAddress(
        IntPtr hModule,
        string procName);
```

```
}
```

```
@"
```

```
    }
```

```
catch
```

```
{
```

```
    Log $_.Exception.Message
```

```
    LogAndConsole "Instantiate-Kernel32 failed"
```

```
}
```

```
}
```

```
function Instantiate-HSTI {
```

```
    try
```

```
    {
```

```
        Add-Type -TypeDefinition @"
```

```
using System;
```

```
using System.Diagnostics;
```

```
using System.Runtime.InteropServices;
```

```
using System.Net;
```

```
public static class HstiTest3
```

```
{
```

```
    [DllImport("hstitest.dll", CharSet = CharSet.Unicode)]
```

```
    public static extern int QueryHSTIdetails(
        ref HstiOverallError pHstiOverallError,
        [In, Out] HstiProviderErrorDuple[] pHstiProviderErrors,
        ref uint pHstiProviderErrorsCount,
        byte[] hstiPlatformSecurityBlob,
        ref uint pHstiPlatformSecurityBlobBytes);
```

```
    [DllImport("hstitest.dll", CharSet = CharSet.Unicode)]
```

```
    public static extern int QueryHSTI(ref bool Pass);
```

```
    [StructLayout(LayoutKind.Sequential, CharSet = CharSet.Unicode)]
```

```
    public struct HstiProviderErrorDuple
```

```
{
```

```

        internal uint protocolError;
        internal uint role;
        internal HstiProviderErrors providerError;
        [MarshalAs(UnmanagedType.ByValTStr, SizeConst = 256)]
        internal string ID;
        [MarshalAs(UnmanagedType.ByValTStr, SizeConst = 4096)]
        internal string ErrorString;
    }

    [FlagsAttribute]
    public enum HstiProviderErrors : int
    {
        None = 0x00000000,
        VersionMismatch = 0x00000001,
        RoleUnknown = 0x00000002,
        RoleDuplicated = 0x00000004,
        SecurityFeatureSizeMismatch = 0x00000008,
        SizeTooSmall = 0x00000010,
        VerifiedMoreThanImplemented = 0x00000020,
        VerifiedNotMatchImplemented = 0x00000040
    }

    [FlagsAttribute]
    public enum HstiOverallError : int
    {
        None = 0x00000000,
        RoleTooManyPlatformReference = 0x00000001,
        RoleTooManyIbv = 0x00000002,
        RoleTooManyOem = 0x00000004,
        RoleTooManyOdm = 0x00000008,
        RoleMissingPlatformReference = 0x00000010,
        VerifiedIncomplete = 0x00000020,
        ProtocolErrors = 0x00000040,
        BlobVersionMismatch = 0x00000080,
        PlatformSecurityVersionMismatch = 0x00000100,
        ProviderError = 0x00000200
    }
}

"@

$LibHandle = [Kernel32]::LoadLibrary("C:\Windows\System32\hstitest.dll")
$FuncHandle = [Kernel32]::GetProcAddress($LibHandle, "QueryHSTIdetails")
$FuncHandle2 = [Kernel32]::GetProcAddress($LibHandle, "QueryHSTI")

if ([System.IntPtr]::Size -eq 8)
{
    #assuming 64 bit
    Log "`nKernel32::LoadLibrary 64bit --> 0x$("{0:X16}" -f $LibHandle.ToInt64())"
    Log "HstiTest2::QueryHSTIdetails 64bit --> 0x$("{0:X16}" -f $FuncHandle.ToInt64())"
}
else
{
    return
}

$overallError = New-Object HstiTest3+HstiOverallError
$providerErrorDupleCount = New-Object int
$blobByteSize = New-Object int
$hr = [HstiTest3]::QueryHSTIdetails([ref] $overallError, $null, [ref] $providerErrorDupleCount,
    $null, [ref] $blobByteSize)

[byte[]]$blob = New-Object byte[] $blobByteSize
[HstiTest3+HstiProviderErrorDuple[]]$providerErrors = New-Object HstiTest3+HstiProviderErrorDuple[]
$providerErrorDupleCount

$hr = [HstiTest3]::QueryHSTIdetails([ref] $overallError, $providerErrors, [ref]
$providerErrorDupleCount, $blob, [ref] $blobByteSize)
$string = $null
$blob | foreach { $string = $string + $_.ToString("X2")+"," }

```

```

$hstiStatus = New-Object bool
$hr = [HstiTest3]::QueryHSTI([ref] $hstiStatus)

LogAndConsole "HSTI Duple Count: $providerErrorDupleCount"
LogAndConsole "HSTI Blob size: $blobByteSize"
LogAndConsole "String: $string"
LogAndConsole "HSTIStatus: $hstiStatus"
if(($blobByteSize -gt 512) -and ($providerErrorDupleCount -gt 0) -and $hstiStatus)
{
    LogAndConsoleSuccess "HSTI validation successful"
}
elseif(($providerErrorDupleCount -eq 0) -or ($blobByteSize -le 512))
{
    LogAndConsoleWarning "HSTI is absent"
    $DGVerifyWarn.AppendLine("HSTI is absent") | Out-Null
}
else
{
    $ErrorMessage = "HSTI validation failed"
    if($HLK)
    {
        LogAndConsoleError $ErrorMessage
        $DGVerifyCrit.AppendLine($ErrorMessage) | Out-Null
    }
    else
    {
        LogAndConsoleWarning $ErrorMessage
        $DGVerifyWarn.AppendLine("HSTI is absent") | Out-Null
    }
}
}
}
catch
{
    LogAndConsoleError $_.Exception.Message
    LogAndConsoleError "Instantiate-HSTI failed"
}
}

function CheckDGRunning($_val)
{
    $DGObj = Get-CimInstance -classname Win32_DeviceGuard -namespace root\Microsoft\Windows\DeviceGuard
    for($i=0; $i -lt $DGObj.SecurityServicesRunning.length; $i++)
    {
        if($DGObj.SecurityServicesRunning[$i] -eq $_val)
        {
            return 1
        }
    }
    return 0
}

function CheckDGFeatures($_val)
{
    $DGObj = Get-CimInstance -classname Win32_DeviceGuard -namespace root\Microsoft\Windows\DeviceGuard
    Log "DG_obj $DG_obj"
    Log "DG_obj.AvailableSecurityProperties.length $DG_obj.AvailableSecurityProperties.length"
    for($i=0; $i -lt $DGObj.AvailableSecurityProperties.length; $i++)
    {
        if($DGObj.AvailableSecurityProperties[$i] -eq $_val)
        {
            return 1
        }
    }
    return 0
}

```

```

}

function PrintConfigCIDetails($_ConfigCIState)
{
    $_ConfigCIRunning = "Config-CI is enabled and running."
    $_ConfigCIDisabled = "Config-CI is not running."
    $_ConfigCIMode = "Not Enabled"
    switch ($_ConfigCIState)
    {
        0 { $_ConfigCIMode = "Not Enabled" }
        1 { $_ConfigCIMode = "Audit mode" }
        2 { $_ConfigCIMode = "Enforced mode" }
        default { $_ConfigCIMode = "Not Enabled" }
    }

    if($_ConfigCIState -ge 1)
    {
        LogAndConsoleSuccess "$_ConfigCIRunning ($_ConfigCIMode)"
    }
    else
    {
        LogAndConsoleWarning "$_ConfigCIDisabled ($_ConfigCIMode)"
    }
}

function PrintHVCIDetails($_HVCIState)
{
    $_HvciRunning = "HVCI is enabled and running."
    $_HvciDisabled = "HVCI is not running."

    if($_HVCIState)
    {
        LogAndConsoleSuccess $_HvciRunning
    }
    else
    {
        LogAndConsoleWarning $_HvciDisabled
    }
}

function PrintCGDetails ($_CGState)
{
    $_CGRunning = "Credential-Guard is enabled and running."
    $_CGDisabled = "Credential-Guard is not running."

    if($_CGState)
    {
        LogAndConsoleSuccess $_CGRunning
    }
    else
    {
        LogAndConsoleWarning $_CGDisabled
    }
}

if(![IO.Directory]::Exists($path))
{
    New-Item -ItemType directory -Path $path
}
else
{
    #Do Nothing!!
}

function IsRedstone
{
    $_osVersion = [environment]::OSVersion.Version
    Log $_osVersion
    #Check if build Major is Windows 10
}

```

```

if($_osVersion.Major -lt 10)
{
    return 0
}
#Check if the build is post Threshold2 (1511 release) => Redstone
if($_osVersion.Build -gt 10586)
{
    return 1
}
#default return False
return 0
}

function ExecuteCommandAndLog($_cmd)
{
    try
    {
        Log "Executing: $_cmd"
        $CmdOutput = Invoke-Expression $_cmd | Out-String
        Log "Output: $CmdOutput"
    }
    catch
    {
        Log "Exception while executing $_cmd"
        Log $_.Exception.Message
    }
}

}

function PrintRebootWarning
{
    LogAndConsoleWarning "Please reboot the machine, for settings to be applied."
}

function AutoRebootHelper
{
    if($AutoReboot)
    {
        LogAndConsole "PC will restart in 30 seconds"
        ExecuteCommandAndLog 'shutdown /r /t 30'
    }
    else
    {
        PrintRebootWarning
    }
}

}

function VerifierReset
{
    $verifier_state = verifier /query | Out-String
    if(!$verifier_state.ToString().Contains("No drivers are currently verified.))
    {
        ExecuteCommandAndLog 'verifier.exe /reset'
    }
    AutoRebootHelper
}

}

function PrintHardwareReq
{
    LogAndConsole "#####"
    LogAndConsole "OS and Hardware requirements for enabling Device Guard and Credential Guard"
    LogAndConsole " 1. OS SKUs: Available only on these OS Skus - Enterprise, Server, Education and Enterprise IoT"
    LogAndConsole " 2. Hardware: Recent hardware that supports virtualization extension with SLAT"
    LogAndConsole "To learn more please visit: https://aka.ms/dgwhcr"
    LogAndConsole "##### `n"
}
}

```



```

function CheckDriverCompat
{
    $_HVCIState = CheckDGRunning(2)
    if($_HVCIState)
    {
        LogAndConsoleWarning "HVCI is already enabled on this machine, driver compat list might not be complete."
        LogAndConsoleWarning "Please disable HVCI and run the script again..."
    }
    $verifier_state = verifier /query | Out-String
    if($verifier_state.ToString().Contains("No drivers are currently verified."))
    {
        LogAndConsole "Enabling Driver verifier"
        verifier.exe /flags 0x02000000 /all /bootmode oneboot /log.code_integrity

        LogAndConsole "Enabling Driver Verifier and Rebooting system"
        Log $verifier_state
        LogAndConsole "Please re-execute this script after reboot..."
        if($AutoReboot)
        {
            LogAndConsole "PC will restart in 30 seconds"
            ExecuteCommandAndLog 'shutdown /r /t 30'
        }
        else
        {
            LogAndConsole "Please reboot manually and run the script again..."
        }
        exit
    }
    else
    {
        LogAndConsole "Driver verifier already enabled"
        Log $verifier_state
        ListDrivers($verifier_state.Trim().ToLowerInvariant())
    }
}

function IsDomainController
{
    $_isDC = 0
    $CompConfig = Get-WmiObject Win32_ComputerSystem
    foreach ($ObjItem in $CompConfig)
    {
        $Role = $ObjItem.DomainRole
        Log "Role=$Role"
        Switch ($Role)
        {
            0 { Log "Standalone Workstation" }
            1 { Log "Member Workstation" }
            2 { Log "Standalone Server" }
            3 { Log "Member Server" }
            4
            {
                Log "Backup Domain Controller"
                $_isDC=1
                break
            }
            5
            {
                Log "Primary Domain Controller"
                $_isDC=1
                break
            }
            default { Log "Unknown Domain Role" }
        }
    }
    return $_isDC
}

```

```

function CheckOSSKU
{
    $osname = $(Get-ComputerInfo).WindowsProductName.ToLower()
    $_SKUSupported = 0
    Log "OSNAME:$osname"
    $SKUarray = @("Enterprise", "Education", "IoT", "Windows Server")
    $HLKAllowed = @("windows 10 pro")
    foreach ($SKUent in $SKUarray)
    {
        if($osname.ToString().Contains($SKUent.ToLower()))
        {
            $_SKUSupported = 1
            break
        }
    }

    # For running HLK tests only, professional SKU's are marked as supported.
    if($HLK)
    {
        if($osname.ToString().Contains($HLKAllowed.ToLower()))
        {
            $_SKUSupported = 1
        }
    }
    $_isDomainController = IsDomainController
    if($_SKUSupported)
    {
        LogAndConsoleSuccess "This PC edition is Supported for DeviceGuard";
        if(($_isDomainController -eq 1) -and !$HVCI -and !$DG)
        {
            LogAndConsoleError "This PC is configured as a Domain Controller, Credential Guard is not supported on DC."
        }
        ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\" /v "OSSKU" /t REG_DWORD /d 2 /f '
    }
    else
    {
        LogAndConsoleError "This PC edition is Unsupported for Device Guard"
        $DGVerifyCrit.AppendLine("OS SKU unsupported") | Out-Null
        ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\" /v "OSSKU" /t REG_DWORD /d 0 /f '
    }
}

function CheckOSArchitecture
{
    $OSArch = $(Get-WmiObject win32_operatingsystem).OSArchitecture.ToLower()
    Log $OSArch
    if($OSArch -match ("^64\-\s?bit"))
    {
        LogAndConsoleSuccess "64 bit architecture"
    }
    elseif($OSArch -match ("^32\-\s?bit"))
    {
        LogAndConsoleError "32 bit architecture"
        $DGVerifyCrit.AppendLine("32 Bit OS, OS Architecture failure.") | Out-Null
    }
    else
    {
        LogAndConsoleError "Unknown architecture"
        $DGVerifyCrit.AppendLine("Unknown OS, OS Architecture failure.") | Out-Null
    }
}

function CheckSecureBootState
{
    $_secureBoot = Confirm-SecureBootUEFI
    Log $ secureBoot
}

```

```

if($secureBoot)
{
    LogAndConsoleSuccess "Secure Boot is present"
    ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\" /v
"SecureBoot" /t REG_DWORD /d 2 /f '
}
else
{
    LogAndConsoleError "Secure Boot is absent / not enabled."
    LogAndConsoleError "If Secure Boot is supported on the system, enable Secure Boot in the BIOS and
run the script again."
    ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\" /v
"SecureBoot" /t REG_DWORD /d 0 /f '
    $DGVerifyCrit.AppendLine("Secure boot validation failed.") | Out-Null
}
}

function CheckVirtualization
{
    $_vmmExtension = $(Get-WMIObject -Class Win32_processor).VMMonitorModeExtensions
    $_vmFirmwareExtension = $(Get-WMIObject -Class Win32_processor).VirtualizationFirmwareEnabled
    $_vmHyperVPresent = (Get-CimInstance -Class Win32_ComputerSystem).HypervisorPresent
    Log "VMMonitorModeExtensions $_vmmExtension"
    Log "VirtualizationFirmwareEnabled $_vmFirmwareExtension"
    Log "HyperVisorPresent $_vmHyperVPresent"

    #success if either processor supports and enabled or if hyper-v is present
    if(($_vmmExtension -and $_vmFirmwareExtension) -or $_vmHyperVPresent )
    {
        LogAndConsoleSuccess "Virtualization firmware check passed"
        ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\" /v
"Virtualization" /t REG_DWORD /d 2 /f '
    }
    else
    {
        LogAndConsoleError "Virtualization firmware check failed."
        LogAndConsoleError "If Virtualization extensions are supported on the system, enable hardware
virtualization (Intel Virtualization Technology, Intel VT-x, Virtualization Extensions, or similar) in the
BIOS and run the script again."
        ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\" /v
"Virtualization" /t REG_DWORD /d 0 /f '
        $DGVerifyCrit.AppendLine("Virtualization firmware check failed.") | Out-Null
    }
}

function CheckTPM
{
    $TPMLockout = $(get-tpm).LockoutCount

    if($TPMLockout)
    {
        if($TPMLockout.ToString().Contains("Not Supported for TPM 1.2"))
        {
            if($HLK)
            {
                LogAndConsoleSuccess "TPM 1.2 is present."
            }
            else
            {
                $WarningMsg = "TPM 1.2 is Present. TPM 2.0 is Preferred."
                LogAndConsoleWarning $WarningMsg
                $DGVerifyWarn.AppendLine($WarningMsg) | Out-Null
            }
        }
        else
        {
            LogAndConsoleSuccess "TPM 2.0 is present."
        }
    }
}

```

```

    }
    ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\" /v
"TPM" /t REG_DWORD /d 2 /f '
}
else
{
    $WarningMsg = "TPM is absent or not ready for use"
    if($HLK)
    {
        LogAndConsoleError $WarningMsg
        $DGVerifyCrit.AppendLine($WarningMsg) | Out-Null
    }
    else
    {
        LogAndConsoleWarning $WarningMsg
        $DGVerifyWarn.AppendLine($WarningMsg) | Out-Null
    }
    ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\" /v
"TPM" /t REG_DWORD /d 0 /f '
}
}

function CheckSecureMOR
{
    $isSecureMOR = CheckDGFeatures(4)
    Log "isSecureMOR= $isSecureMOR "
    if($isSecureMOR -eq 1)
    {
        LogAndConsoleSuccess "Secure MOR is available"
        ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\" /v
"SecureMOR" /t REG_DWORD /d 2 /f '
    }
    else
    {
        $WarningMsg = "Secure MOR is absent"
        if($HLK)
        {
            LogAndConsoleError $WarningMsg
            $DGVerifyCrit.AppendLine($WarningMsg) | Out-Null
        }
        else
        {
            LogAndConsoleWarning $WarningMsg
            $DGVerifyWarn.AppendLine($WarningMsg) | Out-Null
        }
        ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\" /v
"SecureMOR" /t REG_DWORD /d 0 /f '
    }
}

function CheckNXProtection
{
    $isNXProtected = CheckDGFeatures(5)
    Log "isNXProtected= $isNXProtected "
    if($isNXProtected -eq 1)
    {
        LogAndConsoleSuccess "NX Protector is available"
        ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\" /v
"UEFINX" /t REG_DWORD /d 2 /f '
    }
    else
    {
        LogAndConsoleWarning "NX Protector is absent"
        $DGVerifyWarn.AppendLine("NX Protector is absent") | Out-Null
        ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\" /v
"UEFINX" /t REG_DWORD /d 0 /f '
    }
}

```

```

function ChecksMMPProtection
{
    $isSMMMitigated = CheckDGFeatures(6)
    Log "isSMMMitigated= $isSMMMitigated "
    if($isSMMMitigated -eq 1)
    {
        LogAndConsoleSuccess "SMM Mitigation is available"
        ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\" /v
"SMMProtections" /t REG_DWORD /d 2 /f '
    }
    else
    {
        LogAndConsoleWarning "SMM Mitigation is absent"
        $DGVerifyWarn.AppendLine("SMM Mitigation is absent") | Out-Null
        ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\" /v
"SMMProtections" /t REG_DWORD /d 0 /f '
    }
}

function CheckHSTI
{
    LogAndConsole "Copying HSTITest.dll"
    try
    {
        $HSTITest_Decoded = [System.Convert]::FromBase64String($HSTITest_Encoded)
        [System.IO.File]::WriteAllBytes("$env:windir\System32\hstitest.dll",$HSTITest_Decoded)
    }
    catch
    {
        LogAndConsole $_.Exception.Message
        LogAndConsole "Copying and loading HSTITest.dll failed"
    }

    Instantiate-Kernel32
    Instantiate-HSTI
}

function PrintToolVersion
{
    LogAndConsole ""
    LogAndConsole "#####"
    LogAndConsole ""
    LogAndConsole "Readiness Tool Version 3.7.2 Release. `nTool to check if your device is capable to run
Device Guard and Credential Guard."
    LogAndConsole ""
    LogAndConsole "#####"
    LogAndConsole ""
}

PrintToolVersion

if(!$Ready) -and !$Capable) -and !$Enable) -and !$Disable) -and !$Clear) -and !$ResetVerifier))
{
    #Print Usage if none of the options are specified
    LogAndConsoleWarning "How to read the output:"
    LogAndConsoleWarning ""
    LogAndConsoleWarning " 1. Red Errors: Basic things are missing that will prevent enabling and using
DG/CG"
    LogAndConsoleWarning " 2. Yellow Warnings: This device can be used to enable and use DG/CG, but `n
additional security benefits will be absent. To learn more please go through: https://aka.ms/dgwhcr"
    LogAndConsoleWarning " 3. Green Messages: This device is fully compliant with DG/CG requirements`n"

    LogAndConsoleWarning "#####"
    LogAndConsoleWarning ""
    LogAndConsoleWarning "Hardware requirements for enabling Device Guard and Credential Guard"
    LogAndConsoleWarning " 1. Hardware: Recent hardware that supports virtualization extension with SLAT"
    LogAndConsoleWarning ""
}

```

```

LogAndConsoleWarning "##### `n"

LogAndConsoleWarning "Usage: DG_Readiness.ps1 -[Capable/Ready/Enable/Disable/Clear] -[DG/CG/HVCI] -
[AutoReboot] -Path"
LogAndConsoleWarning "Log file with details is found here: C:\DGLogs `n"

LogAndConsoleWarning "To Enable DG/CG. If you have a custom SIPolicy.p7b then use the -Path parameter
else the hardcoded default policy is used"
LogAndConsoleWarning "Usage: DG_Readiness.ps1 -Enable OR DG_Readiness.ps1 -Enable -Path <full path to
the SIPolicy.p7b> `n"

LogAndConsoleWarning "To Enable only HVCI"
LogAndConsoleWarning "Usage: DG_Readiness.ps1 -Enable -HVCI `n"

LogAndConsoleWarning "To Enable only CG"
LogAndConsoleWarning "Usage: DG_Readiness.ps1 -Enable -CG `n"

LogAndConsoleWarning "To Verify if DG/CG is enabled"
LogAndConsoleWarning "Usage: DG_Readiness.ps1 -Ready `n"

LogAndConsoleWarning "To Disable DG/CG."
LogAndConsoleWarning "Usage: DG_Readiness.ps1 -Disable `n"

LogAndConsoleWarning "To Verify if DG/CG is disabled"
LogAndConsoleWarning "Usage: DG_Readiness.ps1 -Ready `n"

LogAndConsoleWarning "To Verify if this device is DG/CG Capable"
LogAndConsoleWarning "Usage: DG_Readiness.ps1 -Capable`n"

LogAndConsoleWarning "To Verify if this device is HVCI Capable"
LogAndConsoleWarning "Usage: DG_Readiness.ps1 -Capable -HVCI`n"

LogAndConsoleWarning "To Auto reboot with each option"
LogAndConsoleWarning "Usage: DG_Readiness.ps1 -[Capable/Enable/Disable] -AutoReboot`n"
LogAndConsoleWarning "#####"
LogAndConsoleWarning ""
LogAndConsoleWarning "When the Readiness Tool with '-capable' is run the following RegKey values are
set:"
LogAndConsoleWarning ""
LogAndConsoleWarning "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities"
LogAndConsoleWarning "CG_Capable"
LogAndConsoleWarning "DG_Capable"
LogAndConsoleWarning "HVCI_Capable"
LogAndConsoleWarning ""
LogAndConsoleWarning "Value 0 = not possible to enable DG/CG/HVCI on this device"
LogAndConsoleWarning "Value 1 = not fully compatible but has sufficient firmware/hardware/software
features to enable DG/CG/HVCI"
LogAndConsoleWarning "Value 2 = fully compatible for DG/CG/HVCI"
LogAndConsoleWarning ""
LogAndConsoleWarning "##### `n"
}

$user = [Security.Principal.WindowsIdentity]::GetCurrent();
$TestForAdmin = (New-Object Security.Principal.WindowsPrincipal
$user).IsInRole([Security.Principal.WindowsBuiltinRole]::Administrator)

if(!$TestForAdmin)
{
    LogAndConsoleError "This script requires local administrator privileges. Please execute this script as a
local administrator."
    exit
}

$isRunningOnVM = (Get-WmiObject win32_computersystem).model
if($isRunningOnVM.Contains("Virtual"))
{
    LogAndConsoleWarning "Running on a Virtual Machine. DG/CG is supported only if both guest VM and host
machine are running with Windows 10, version 1703 or later with English localization."
}

```

```

<# Check the DG status if enabled or disabled, meaning if the device is ready or not #>
if($Ready)
{
    PrintHardwareReq

    $DGRunning = $(Get-CimInstance -classname Win32_DeviceGuard -namespace
root\Microsoft\Windows\DeviceGuard).SecurityServicesRunning
    $_ConfigCIState = $(Get-CimInstance -classname Win32_DeviceGuard -namespace
root\Microsoft\Windows\DeviceGuard).CodeIntegrityPolicyEnforcementStatus
    Log "Current DGRunning = $DGRunning, ConfigCI= $_ConfigCIState"
    $_HVCIState = CheckDGRunning(2)
    $_CGState = CheckDGRunning(1)

    if($HVCI)
    {
        Log "_HVCIState: $_HVCIState"
        PrintHVCIDetails $_HVCIState
    }
    elseif($CG)
    {
        Log "_CGState: $_CGState"
        PrintCGDetails $_CGState

        if($_CGState)
        {
            ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\'
/v "CG_Running" /t REG_DWORD /d 1 /f'
        }
        else
        {
            ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\'
/v "CG_Running" /t REG_DWORD /d 0 /f'
        }
    }
    elseif($DG)
    {
        Log "_HVCIState: $_HVCIState, _ConfigCIState: $_ConfigCIState"

        PrintHVCIDetails $_HVCIState
        PrintConfigCIDetails $_ConfigCIState

        if($_ConfigCIState -and $_HVCIState)
        {
            LogAndConsoleSuccess "HVCI, and Config-CI are enabled and running."

            ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\'
/v "DG_Running" /t REG_DWORD /d 1 /f'
        }
        else
        {
            LogAndConsoleWarning "Not all services are running."

            ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\'
/v "DG_Running" /t REG_DWORD /d 0 /f'
        }
    }
    else
    {
        Log "_CGState: $_CGState, _HVCIState: $_HVCIState, _ConfigCIState: $_ConfigCIState"

        PrintCGDetails $_CGState
        PrintHVCIDetails $_HVCIState
        PrintConfigCIDetails $_ConfigCIState

        if(($DGRunning.Length -ge 2) -and ($_CGState) -and ($_HVCIState) -and ($_ConfigCIState -ge 1))
        {
            LogAndConsoleSuccess "HVCI, Credential Guard, and Config CI are enabled and running."
        }
    }
}

```

```

    }
    else
    {
        LogAndConsoleWarning "Not all services are running."
    }
}

<# Enable and Disable #>
if($Enable)
{
    PrintHardwareReq

    LogAndConsole "Enabling Device Guard and Credential Guard"
    LogAndConsole "Setting RegKeys to enable DG/CG"

    ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v
"EnableVirtualizationBasedSecurity" /t REG_DWORD /d 1 /f'
    #Only SecureBoot is required as part of RequirePlatformSecurityFeatures
    ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v
"RequirePlatformSecurityFeatures" /t REG_DWORD /d 1 /f'

    $_isRedstone = IsRedstone
    if(!$isRedstone)
    {
        ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v "Unlocked" /t
REG_DWORD /d 1 /f'
    }
    else
    {
        ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v "Locked" /t
REG_DWORD /d 0 /f'
    }

    if(!$HVCI -and !$DG)
    {
        # value is 2 for both Th2 and RS1
        ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\Lsa" /v "LsaCfgFlags" /t
REG_DWORD /d 2 /f'
    }
    if(!$CG)
    {
        if(!$isRedstone)
        {
            ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v
"HypervisorEnforcedCodeIntegrity" /t REG_DWORD /d 1 /f'
        }
        else
        {
            ExecuteCommandAndLog 'REG ADD
"HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Scenarios\HypervisorEnforcedCodeIntegrity" /v "Enabled"
/t REG_DWORD /d 1 /f'
            ExecuteCommandAndLog 'REG ADD
"HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Scenarios\HypervisorEnforcedCodeIntegrity" /v "Locked" /t
REG_DWORD /d 0 /f'
        }
    }

    try
    {
        if(!$HVCI -and !$CG)
        {
            if(!$SIPolicyPath)
            {
                Log "Writing Decoded SIPolicy.p7b"
                $SIPolicy_Decoded = [System.Convert]::FromBase64String($SIPolicy_Encoded)
                [System.IO.File]::WriteAllBytes("$env:windir\System32\CodeIntegrity\SIPolicy.p7b",$SIPolicy_Decoded)
            }
        }
    }
}

```



```

    else
    {
        LogAndConsole "Copying user provided SIpolicy.p7b"
        $CmdOutput = Copy-Item $SIPolicyPath "$env:windir\System32\CodeIntegrity\SIpolicy.p7b" |
Out-String
        Log $CmdOutput
    }
}
}
catch
{
    LogAndConsole "Writing SIpolicy.p7b file failed"
}

LogAndConsole "Enabling Hyper-V and IOMMU"
$_isRedstone = IsRedstone
if(!$isRedstone)
{
    LogAndConsole "OS Not Redstone, enabling IsolatedUserMode separately"
    #Enable/Disable IOMMU separately
    ExecuteCommandAndLog 'DISM.EXE /Online /Enable-Feature:IsolatedUserMode /NoRestart'
}
$CmdOutput = DISM.EXE /Online /Enable-Feature:Microsoft-Hyper-V-Hypervisor /All /NoRestart | Out-String
if(!$CmdOutput.Contains("The operation completed successfully."))
{
    $CmdOutput = DISM.EXE /Online /Enable-Feature:Microsoft-Hyper-V-Online /All /NoRestart | Out-String
}

Log $CmdOutput
if($CmdOutput.Contains("The operation completed successfully."))
{
    LogAndConsoleSuccess "Enabling Hyper-V and IOMMU successful"
    #Reg key for HLK validation of DISM.EXE step
    ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\" /v
"HyperVEnabled" /t REG_DWORD /d 1 /f'
}
else
{
    LogAndConsoleWarning "Enabling Hyper-V failed please check the log file"
    #Reg key for HLK validation of DISM.EXE step
    ExecuteCommandAndLog 'REG ADD "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities\" /v
"HyperVEnabled" /t REG_DWORD /d 0 /f'
}
    AutoRebootHelper
}

if($Disable)
{
    LogAndConsole "Disabling Device Guard and Credential Guard"
    LogAndConsole "Deleting RegKeys to disable DG/CG"

    ExecuteCommandAndLog 'REG DELETE "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v
"EnableVirtualizationBasedSecurity" /f'
    ExecuteCommandAndLog 'REG DELETE "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v
"RequirePlatformSecurityFeatures" /f'

    $_isRedstone = IsRedstone
    if(!$isRedstone)
    {
        ExecuteCommandAndLog 'REG DELETE "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v "NoLock" /f'
    }
    else
    {
        ExecuteCommandAndLog 'REG DELETE "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v "Locked" /f'
    }

    if(!$CG)
    {
        ExecuteCommandAndLog 'REG DELETE "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard" /v

```

```

ExecuteCommandAndLog 'REG DELETE "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\ /v
"HypervisorEnforcedCodeIntegrity" /f'
    if($_isRedstone)
    {
        ExecuteCommandAndLog 'REG DELETE
"HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Scenarios\HypervisorEnforcedCodeIntegrity" /f'
    }
}

if(!$HVCI -and !$DG)
{
    ExecuteCommandAndLog 'REG DELETE "HKLM\SYSTEM\CurrentControlSet\Control\Lsa" /v "LsaCfgFlags" /f'
}

if(!$HVCI -and !$CG)
{
    ExecuteCommandAndLog 'del "$env:windir\System32\CodeIntegrity\SIPolicy.p7b"'
}

if(!$HVCI -and !$DG -and !$CG)
{
    LogAndConsole "Disabling Hyper-V and IOMMU"
    $_isRedstone = IsRedstone
    if(!$isRedstone)
    {
        LogAndConsole "OS Not Redstone, disabling IsolatedUserMode separately"
        #Enable/Disable IOMMU separately
        ExecuteCommandAndLog 'DISM.EXE /Online /disable-Feature /FeatureName:IsolatedUserMode
/NoRestart'
    }
    $CmdOutput = DISM.EXE /Online /disable-Feature /FeatureName:Microsoft-Hyper-V-Hypervisor /NoRestart
| Out-String
    if(!$CmdOutput.Contains("The operation completed successfully."))
    {
        $CmdOutput = DISM.EXE /Online /disable-Feature /FeatureName:Microsoft-Hyper-V-Online /NoRestart
| Out-String
    }
    Log $CmdOutput
    if($CmdOutput.Contains("The operation completed successfully."))
    {
        LogAndConsoleSuccess "Disabling Hyper-V and IOMMU successful"
    }
    else
    {
        LogAndConsoleWarning "Disabling Hyper-V failed please check the log file"
    }

    #set of commands to run SecConfig.efi to delete UEFI variables if were set in pre OS
    #these steps can be performed even if the UEFI variables were not set - if not set it will lead to
No-Op but this can be run in general always
    #this requires a reboot and accepting the prompt in the Pre-OS which is self explanatory in the
message that is displayed in pre-OS
    $FreeDrive = ls function:[s-z]: -n | ?{ !(test-path $_) } | random
    Log "FreeDrive=$FreeDrive"
    ExecuteCommandAndLog 'mountvol $FreeDrive /s'
    $CmdOutput = Copy-Item "$env:windir\System32\SecConfig.efi"
$FreeDrive\EFI\Microsoft\Boot\SecConfig.efi -Force | Out-String
    LogAndConsole $CmdOutput
    ExecuteCommandAndLog 'bcdedit /create "{0cb3b571-2f2e-4343-a879-d86a476d7215}" /d DGOptOut
/application osloader'
    ExecuteCommandAndLog 'bcdedit /set "{0cb3b571-2f2e-4343-a879-d86a476d7215}" path
\EFI\Microsoft\Boot\SecConfig.efi'
    ExecuteCommandAndLog 'bcdedit /set "{bootmgr}" bootsequence "{0cb3b571-2f2e-4343-a879-
d86a476d7215}"'
    ExecuteCommandAndLog 'bcdedit /set "{0cb3b571-2f2e-4343-a879-d86a476d7215}" loadoptions DISABLE-LSA-
ISO,DISABLE-VBS'
    ExecuteCommandAndLog 'bcdedit /set "{0cb3b571-2f2e-4343-a879-d86a476d7215}" device
partition=$FreeDrive'
    ExecuteCommandAndLog 'mountvol $FreeDrive /d'
    #these steps can be performed even if the UEFI variables were not set - if not set it will lead to
No-Op but this can be run in general always

```

```

#steps complete

}
AutoRebootHelper
}

if($Clear)
{
    ExecuteCommandAndLog 'REG DELETE "HKLM\SYSTEM\CurrentControlSet\Control\DeviceGuard\Capabilities" /f'
    VerifierReset
}

if($ResetVerifier)
{
    VerifierReset
}

<# Is machine Device Guard / Cred Guard Capable and Verify #>
if($Capable)
{
    PrintHardwareReq

    LogAndConsole "Checking if the device is DG/CG Capable"

    $_isRedstone = IsRedstone
    if(!$isRedstone)
    {
        LogAndConsoleWarning "Capable is currently fully supported in Redstone only.."
    }
    $_StepCount = 1
    if(!$CG)
    {
        LogAndConsole " ===== Step $_StepCount Driver Compat ===== "
        $_StepCount++
        CheckDriverCompat
    }

    LogAndConsole " ===== Step $_StepCount Secure boot present ===== "
    $_StepCount++
    CheckSecureBootState

    if(!$HVCI -and !$DG -and !$CG)
    {
        #check only if sub-options are absent
        LogAndConsole " ===== Step $_StepCount MS UEFI HSTI tests ===== "
        $_StepCount++
        CheckHSTI
    }

    LogAndConsole " ===== Step $_StepCount OS Architecture ===== "
    $_StepCount++
    CheckOSArchitecture

    LogAndConsole " ===== Step $_StepCount Supported OS SKU ===== "
    $_StepCount++
    CheckOSSKU

    LogAndConsole " ===== Step $_StepCount Virtualization Firmware ===== "
    $_StepCount++
    CheckVirtualization

    if(!$HVCI -and !$DG)
    {
        LogAndConsole " ===== Step $_StepCount TPM version ===== "
        $_StepCount++
        CheckTPM

        LogAndConsole " ===== Step $_StepCount Secure MOR ===== "
        $_StepCount++
    }
}

```

```
    CheckSecureMOR
}

LogAndConsole " ===== Step $_StepCount NX Protector ===== "
$_StepCount++
CheckNXProtection

LogAndConsole " ===== Step $_StepCount SMM Mitigation ===== "
$_StepCount++
CheckSMPProtection

LogAndConsole " ===== End Check ===== "

LogAndConsole " ===== Summary ===== "
ListSummary
LogAndConsole "To learn more about required hardware and software please visit: https://aka.ms/dgwhcr"
}

# SIG # Begin signature block
## REPLACE
# SIG # End signature block
```

Windows Defender Credential Guard protection limits

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016
- Windows Server 2019

Some ways to store credentials are not protected by Windows Defender Credential Guard, including:

- Software that manages credentials outside of Windows feature protection
- Local accounts and Microsoft Accounts
- Windows Defender Credential Guard does not protect the Active Directory database running on Windows Server 2016 domain controllers. It also does not protect credential input pipelines, such as Windows Server 2016 servers running Remote Desktop Gateway. If you're using a Windows Server 2016 server as a client PC, it will get the same protection as it would when running Windows 10 Enterprise.
- Key loggers
- Physical attacks
- Does not prevent an attacker with malware on the PC from using the privileges associated with any credential. We recommend using dedicated PCs for high value accounts, such as IT Pros and users with access to high value assets in your organization.
- Third-party security packages
- Digest and CredSSP credentials
 - When Windows Defender Credential Guard is enabled, neither Digest nor CredSSP have access to users' logon credentials. This implies no Single Sign-On use for these protocols.
- Supplied credentials for NTLM authentication are not protected. If a user is prompted for and enters credentials for NTLM authentication, these credentials are vulnerable to be read from LSASS memory. Note that these same credentials are vulnerable to key loggers as well.-
- Kerberos service tickets are not protected by Credential Guard, but the Kerberos Ticket Granting Ticket (TGT) is.
- When Windows Defender Credential Guard is deployed on a VM, Windows Defender Credential Guard protects secrets from attacks inside the VM. However, it does not provide additional protection from privileged system attacks originating from the host.
- Windows logon cached password verifiers (commonly called "cached credentials") do not qualify as credentials because they cannot be presented to another computer for authentication, and can only be used locally to verify credentials. They are stored in the registry on the local computer and provide validation for credentials when a domain-joined computer cannot connect to AD DS during user logon. These "cached logons", or more specifically, cached domain account information, can be managed using the security policy setting **Interactive logon: Number of previous logons to cache** if a domain controller is not available.

See also

Deep Dive into Windows Defender Credential Guard: [Related videos](#)

[Microsoft Cybersecurity Stack: Advanced Identity and Endpoint Protection: Manage Credential Guard](#)

NOTE

- Note: Requires [LinkedIn Learning subscription](#) to view the full video

Considerations when using Windows Defender Credential Guard

7/1/2022 • 6 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016
- Windows Server 2019

Passwords are still weak. We recommend that in addition to deploying Windows Defender Credential Guard, organizations move away from passwords to other authentication methods, such as physical smart cards, virtual smart cards, or Windows Hello for Business.

Windows Defender Credential Guard uses hardware security, so some features such as Windows To Go, are not supported.

Wi-fi and VPN Considerations

When you enable Windows Defender Credential Guard, you can no longer use NTLM classic authentication for Single Sign-On. You will be forced to enter your credentials to use these protocols and cannot save the credentials for future use. If you are using WiFi and VPN endpoints that are based on MS-CHAPv2, they are subject to similar attacks as for NTLMv1. For WiFi and VPN connections, Microsoft recommends that organizations move from MSCHAPv2-based connections such as PEAP-MSCHAPv2 and EAP-MSCHAPv2, to certificate-based authentication such as PEAP-TLS or EAP-TLS.

Kerberos Considerations

When you enable Windows Defender Credential Guard, you can no longer use Kerberos unconstrained delegation or DES encryption. Unconstrained delegation could allow attackers to extract Kerberos keys from the isolated LSA process. Use constrained or resource-based Kerberos delegation instead.

3rd Party Security Support Providers Considerations

Some 3rd party Security Support Providers (SSPs and APs) might not be compatible with Windows Defender Credential Guard because it does not allow third-party SSPs to ask for password hashes from LSA. However, SSPs and APs still get notified of the password when a user logs on and/or changes their password. Any use of undocumented APIs within custom SSPs and APs are not supported. We recommend that custom implementations of SSPs/APs are tested with Windows Defender Credential Guard. SSPs and APs that depend on any undocumented or unsupported behaviors fail. For example, using the `KerbQuerySupplementalCredentialsMessage` API is not supported. Replacing the NTLM or Kerberos SSPs with custom SSPs and APs. For more info, see [Restrictions around Registering and Installing a Security Package](#) on MSDN.

Upgrade Considerations

As the depth and breadth of protections provided by Windows Defender Credential Guard are increased, subsequent releases of Windows 10 with Windows Defender Credential Guard running may impact scenarios

that were working in the past. For example, Windows Defender Credential Guard may block the use of a particular type of credential or a particular component to prevent malware from taking advantage of vulnerabilities. Test scenarios required for operations in an organization before upgrading a device using Windows Defender Credential Guard.

Saved Windows Credentials Protected

Starting with Windows 10, version 1511, domain credentials that are stored with Credential Manager are protected with Windows Defender Credential Guard. Credential Manager allows you to store three types of credentials: Windows credentials, certificate-based credentials, and generic credentials. Generic credentials such as user names and passwords that you use to log on to websites are not protected since the applications require your cleartext password. If the application does not need a copy of the password, they can save domain credentials as Windows credentials that are protected. Windows credentials are used to connect to other computers on a network. The following considerations apply to the Windows Defender Credential Guard protections for Credential Manager:

- Windows credentials saved by Remote Desktop Client cannot be sent to a remote host. Attempts to use saved Windows credentials fail, displaying the error message "Logon attempt failed."
- Applications that extract Windows credentials fail.
- When credentials are backed up from a PC that has Windows Defender Credential Guard enabled, the Windows credentials cannot be restored. If you need to back up your credentials, you must do this before you enable Windows Defender Credential Guard. Otherwise, you cannot restore those credentials.

Clearing TPM Considerations

Virtualization-based Security (VBS) uses the TPM to protect its key. So when the TPM is cleared then the TPM protected key used to encrypt VBS secrets is lost.

WARNING

Clearing the TPM results in loss of protected data for all features that use VBS to protect data.

When a TPM is cleared ALL features, which use VBS to protect data can no longer decrypt their protected data.

As a result Credential Guard can no longer decrypt protected data. VBS creates a new TPM protected key for Credential Guard. Credential Guard uses the new key to protect new data. However, the previously protected data is lost forever.

NOTE

Credential Guard obtains the key during initialization. So the data loss will only impact persistent data and occur after the next system startup.

Windows credentials saved to Credential Manager

Since Credential Manager cannot decrypt saved Windows Credentials, they are deleted. Applications should prompt for credentials that were previously saved. If saved again, then Windows credentials are protected Credential Guard.

Domain-joined device's automatically provisioned public key

Beginning with Windows 10 and Windows Server 2016, domain-devices automatically provision a bound public key, for more information about automatic public key provisioning, see [Domain-joined Device Public Key Authentication](#).

Since Credential Guard cannot decrypt the protected private key, Windows uses the domain-joined computer's password for authentication to the domain. Unless additional policies are deployed, there should not be a loss of

functionality. If a device is configured to only use public key, then it cannot authenticate with password until that policy is disabled. For more information on Configuring devices to only use public key, see [Domain-joined Device Public Key Authentication](#).

Also if any access control checks including authentication policies require devices to have either the KEY TRUST IDENTITY (S-1-18-4) or FRESH PUBLIC KEY IDENTITY (S-1-18-3) well-known SIDs, then those access checks fail. For more information about authentication policies, see [Authentication Policies and Authentication Policy Silos](#). For more information about well-known SIDs, see [\[MS-DTYP\] Section 2.4.2.4 Well-known SID Structures](#).

Breaking DPAPI on domain-joined devices

On domain-joined devices, DPAPI can recover user keys using a domain controller from the user's domain. If a domain-joined device has no connectivity to a domain controller, then recovery is not possible.

IMPORTANT

Best practice when clearing a TPM on a domain-joined device is to be on a network with connectivity to domain controllers. This ensures DPAPI functions and the user does not experience strange behavior.

Auto VPN configuration is protected with user DPAPI. User may not be able to use VPN to connect to domain controllers since the VPN configurations are lost.

If you must clear the TPM on a domain-joined device without connectivity to domain controllers, then you should consider the following.

Domain user sign-in on a domain-joined device after clearing a TPM for as long as there is no connectivity to a domain controller:

CREDENTIAL TYPE	WINDOWS VERSION	BEHAVIOR
Certificate (smart card or Windows Hello for Business)	All	All data protected with user DPAPI is unusable and user DPAPI does not work at all.
Password	Windows 10 v1709 or later	If the user signed-in with a certificate or password prior to clearing the TPM, then they can sign-in with password and user DPAPI is unaffected.
Password	Windows 10 v1703	If the user signed-in with a password prior to clearing the TPM, then they can sign-in with that password and are unaffected.
Password	Windows 10 v1607 or earlier	Existing user DPAPI protected data is unusable. User DPAPI is able to protect new data.

Once the device has connectivity to the domain controllers, DPAPI recovers the user's key and data protected prior to clearing the TPM can be decrypted.

Impact of DPAPI failures on Windows Information Protection

When data protected with user DPAPI is unusable, then the user loses access to all work data protected by Windows Information Protection. The impact includes: Outlook 2016 is unable to start and work protected documents cannot be opened. If DPAPI is working, then newly created work data is protected and can be accessed.

Workaround: Users can resolve the problem by connecting their device to the domain and rebooting or using their Encrypting File System Data Recovery Agent certificate. For more information about Encrypting File

System Data Recovery Agent certificate, see [Create and verify an Encrypting File System \(EFS\) Data Recovery Agent \(DRA\) certificate](#).

See also

Related videos

[What is virtualization-based security?](#)

Additional mitigations

7/1/2022 • 18 minutes to read • [Edit Online](#)

Windows Defender Credential Guard can provide mitigation against attacks on derived credentials and prevent the use of stolen credentials elsewhere. However, PCs can still be vulnerable to certain attacks, even if the derived credentials are protected by Windows Defender Credential Guard. These attacks can include abusing privileges and use of derived credentials directly from a compromised device, re-using previously stolen credentials prior to Windows Defender Credential Guard, and abuse of management tools and weak application configurations. Because of this, additional mitigation also must be deployed to make the domain environment more robust.

Restricting domain users to specific domain-joined devices

Credential theft attacks allow the attacker to steal secrets from one device and use them from another device. If a user can sign on to multiple devices then any device could be used to steal credentials. How do you ensure that users only sign on using devices that have Windows Defender Credential Guard enabled? By deploying authentication policies that restrict them to specific domain-joined devices that have been configured with Windows Defender Credential Guard. For the domain controller to know what device a user is signing on from, Kerberos armoring must be used.

Kerberos armoring

Kerberos armoring is part of RFC 6113. When a device supports Kerberos armoring, its TGT is used to protect the user's proof of possession which can mitigate offline dictionary attacks. Kerberos armoring also provides the additional benefit of signed KDC errors this mitigates tampering which can result in things such as downgrade attacks.

To enable Kerberos armoring for restricting domain users to specific domain-joined devices

- Users need to be in domains that are running Windows Server 2012 R2 or higher
- All the domain controllers in these domains must be configured to support Kerberos armoring. Set the **KDC support for claims, compound authentication, and Kerberos armoring** Group Policy setting to either **Supported** or **Always provide claims**.
- All the devices with Windows Defender Credential Guard that the users will be restricted to must be configured to support Kerberos armoring. Enable the **Kerberos client support for claims, compound authentication and Kerberos armoring** Group Policy settings under **Computer Configuration -> Administrative Templates -> System -> Kerberos**.

Protecting domain-joined device secrets

Since domain-joined devices also use shared secrets for authentication, attackers can steal those secrets as well. By deploying device certificates with Windows Defender Credential Guard, the private key can be protected. Then authentication policies can require that users sign on devices that authenticate using those certificates. This prevents shared secrets stolen from the device to be used with stolen user credentials to sign on as the user.

Domain-joined device certificate authentication has the following requirements:

- Devices' accounts are in Windows Server 2012 domain functional level or higher.
- All domain controllers in those domains have KDC certificates which satisfy strict KDC validation certificate requirements:
 - KDC EKU present
 - DNS domain name matches the DNSName field of the SubjectAltName (SAN) extension

- Windows devices have the CA issuing the domain controller certificates in the enterprise store.
- A process is established to ensure the identity and trustworthiness of the device in a similar manner as you would establish the identity and trustworthiness of a user before issuing them a smartcard.

Deploying domain-joined device certificates

To guarantee that certificates with the required issuance policy are only installed on the devices these users must use, they must be deployed manually on each device. The same security procedures used for issuing smart cards to users should be applied to device certificates.

For example, let's say you wanted to use the High Assurance policy only on these devices. Using a Windows Server Enterprise certificate authority, you would create a new template.

Creating a new certificate template

1. From the Certificate Manager console, right-click **Certificate Templates**, and then click **Manage**.
2. Right-click **Workstation Authentication**, and then click **Duplicate Template**.
3. Right-click the new template, and then click **Properties**.
4. On the **Extensions** tab, click **Application Policies**, and then click **Edit**.
5. Click **Client Authentication**, and then click **Remove**.
6. Add the ID-PKInit-KPClientAuth EKU. Click **Add**, click **New**, and then specify the following values:
 - Name: Kerberos Client Auth
 - Object Identifier: 1.3.6.1.5.2.3.4
7. On the **Extensions** tab, click **Issuance Policies**, and then click **Edit**.
8. Under **Issuance Policies**, click **High Assurance**.
9. On the **Subject name** tab, clear the **DNS name** check box, and then select the **User Principal Name (UPN)** check box.

Then on the devices that are running Windows Defender Credential Guard, enroll the devices using the certificate you just created.

Enrolling devices in a certificate

Run the following command:

```
CertReq -EnrollCredGuardCert MachineAuthentication
```

NOTE

You must restart the device after enrolling the machine authentication certificate.

How a certificate issuance policy can be used for access control

Beginning with the Windows Server 2008 R2 domain functional level, domain controllers support for authentication mechanism assurance provides a way to map certificate issuance policy OIDs to universal security groups. Windows Server 2012 domain controllers with claim support can map them to claims. To learn more about authentication mechanism assurance, see [Authentication Mechanism Assurance for AD DS in Windows Server 2008 R2 Step-by-Step Guide](#) on TechNet.

To see the issuance policies available

- The [get-IssuancePolicy.ps1](#) shows all of the issuance policies that are available on the certificate authority. From a Windows PowerShell command prompt, run the following command:

```
.\get-IssuancePolicy.ps1 -LinkedToGroup:All
```

To link an issuance policy to a universal security group

- The [set-IssuancePolicyToGroupLink.ps1](#) creates a Universal security group, creates an organizational unit, and links the issuance policy to that Universal security group. From a Windows PowerShell command prompt, run the following command:

```
.\set-IssuancePolicyToGroupLink.ps1 -IssuancePolicyName:"<name of issuance policy>" -groupOU:"<Name of OU to create>" -groupName:"<name of Universal security group to create>"
```

Restricting user sign on

So we now have completed the following:

- Created a special certificate issuance policy to identify devices that meet the deployment criteria required for the user to be able to sign on
- Mapped that policy to a universal security group or claim
- Provided a way for domain controllers to get the device authorization data during user sign on using Kerberos armoring. Now what is left to do is to configure the access check on the domain controllers. This is done using authentication policies.

Authentication policies have the following requirements:

- User accounts are in a Windows Server 2012 domain functional level or higher domain.

Creating an authentication policy restricting users to the specific universal security group

1. Open Active Directory Administrative Center.
2. Click **Authentication**, click **New**, and then click **Authentication Policy**.
3. In the **Display name** box, enter a name for this authentication policy.
4. Under the **Accounts** heading, click **Add**.
5. In the **Select Users, Computers, or Service Accounts** dialog box, type the name of the user account you wish to restrict, and then click **OK**.
6. Under the **User Sign On** heading, click the **Edit** button.
7. Click **Add a condition**.
8. In the **Edit Access Control Conditions** box, ensure that it reads **User > Group > Member of each > Value**, and then click **Add items**.
9. In the **Select Users, Computers, or Service Accounts** dialog box, type the name of the universal security group that you created with the set-IssuancePolicyToGroupLink script, and then click **OK**.
10. Click **OK** to close the **Edit Access Control Conditions** box.
11. Click **OK** to create the authentication policy.
12. Close Active Directory Administrative Center.

NOTE

When the authentication policy enforces policy restrictions, users will not be able to sign on using devices that do not have a certificate with the appropriate issuance policy deployed. This applies to both local and remote sign on scenarios. Therefore, it is strongly recommended to first only audit policy restrictions to ensure you don't have unexpected failures.

Discovering authentication failures due to authentication policies

To make tracking authentication failures due to authentication policies easier, an operational log exists with just those events. To enable the logs on the domain controllers, in Event Viewer, navigate to **Applications and Services Logs\Microsoft\Windows\Authentication**, right-click **AuthenticationPolicyFailures-DomainController**, and then click **Enable Log**.

To learn more about authentication policy events, see [Authentication Policies and Authentication Policy Silos](#).

Appendix: Scripts

Here is a list of scripts mentioned in this topic.

Get the available issuance policies on the certificate authority

Save this script file as get-IssuancePolicy.ps1.

```
#####
## Parameters to be defined ##
## by the user ##
#####
Param (
$Identity,
$LinkedToGroup
)
#####
## Strings definitions ##
#####
Data getIP_strings {
# culture="en-US"
ConvertFrom-StringData -stringdata @"
help1 = This command can be used to retrieve all available Issuance Policies in a forest. The forest of the
currently logged on user is targeted.
help2 = Usage:
help3 = The following parameter is mandatory:
help4 = -LinkedToGroup:<yes|no|all>
help5 = "yes" will return only Issuance Policies that are linked to groups. Checks that the linked Issuance
Policies are linked to valid groups.
help6 = "no" will return only Issuance Policies that are not currently linked to any group.
help7 = "all" will return all Issuance Policies defined in the forest. Checks that the linked Issuance
policies are linked to valid groups.
help8 = The following parameter is optional:
help9 = -Identity:<Name, Distinguished Name or Display Name of the Issuance Policy that you want to
retrieve>. If you specify an identity, the option specified in the "-LinkedToGroup" parameter is ignored.
help10 = Output: This script returns the Issuance Policy objects meeting the criteria defined by the above
parameters.
help11 = Examples:
errorIPNotFound = Error: no Issuance Policy could be found with Identity "{0}"
ErrorNotSecurity = Error: Issuance Policy "{0}" is linked to group "{1}" which is not of type "Security".
ErrorNotUniversal = Error: Issuance Policy "{0}" is linked to group "{1}" whose scope is not "Universal".
ErrorHasMembers = Error: Issuance Policy "{0}" is linked to group "{1}" which has a non-empty membership.
The group has the following members:
LinkedIPs = The following Issuance Policies are linked to groups:
displayName = displayName : {0}
Name = Name : {0}
dn = distinguishedName : {0}
InfoName = Linked Group Name: {0}
InfoDN = Linked Group DN: {0}
NonLinkedIPs = The following Issuance Policies are NOT linked to groups:
"@
}
##Import-LocalizedData getIP_strings
import-module ActiveDirectory
#####
## Help ##
#####
function Display-Help {
""
$getIP_strings.help1
""
$getIP_strings.help2
""
$getIP_strings.help3
" " + $getIP_strings.help4
```

```

"          " + $getIP_strings.help5
"          " + $getIP_strings.help6
"          " + $getIP_strings.help7
""
$getIP_strings.help8
"          " + $getIP_strings.help9
""
    $getIP_strings.help10
""
""
$getIP_strings.help11
"          " + '$' + "myIPs = .\get-IssuancePolicy.ps1 -LinkedToGroup:All"
"          " + '$' + "myLinkedIPs = .\get-IssuancePolicy.ps1 -LinkedToGroup:yes"
"          " + '$' + "myIP = .\get-IssuancePolicy.ps1 -Identity:""Medium Assurance""
""
}
$root = get-adrootdse
$domain = get-addomain -current loggedonuser
$configNCDN = [String]$root.configurationNamingContext
if ( !($Identity) -and !($LinkedToGroup) ) {
display-Help
break
}
if ($Identity) {
    $OIDs = get-adobject -Filter {(objectclass -eq "msPKI-Enterprise-Oid") -and ((name -eq $Identity) -or
(displayname -eq $Identity) -or (distinguishedName -like $Identity)) } -searchBase $configNCDN -properties *
    if ($OIDs -eq $null) {
$errormsg = $getIP_strings.ErrorIPNotFound -f $Identity
write-host $errmsg -ForegroundColor Red
    }
    foreach ($OID in $OIDs) {
        if ($OID."msDS-OIDToGroupLink") {
# In case the Issuance Policy is linked to a group, it is good to check whether there is any problem with
the mapping.
            $groupDN = $OID."msDS-OIDToGroupLink"
            $group = get-adgroup -Identity $groupDN
            $groupName = $group.Name
# Analyze the group
            if ($group.groupCategory -ne "Security") {
$errormsg = $getIP_strings.ErrorNotSecurity -f $Identity, $groupName
write-host $errmsg -ForegroundColor Red
            }
            if ($group.groupScope -ne "Universal") {
$errormsg = $getIP_strings.ErrorNotUniversal -f $Identity, $groupName
write-host $errmsg -ForegroundColor Red
            }
            $members = Get-ADGroupMember -Identity $group
            if ($members) {
$errormsg = $getIP_strings.ErrorHasMembers -f $Identity, $groupName
write-host $errmsg -ForegroundColor Red
                foreach ($member in $members) {
write-host "          " $member -ForegroundColor Red
                }
            }
        }
    }
    return $OIDs
break
}
}
if (($LinkedToGroup -eq "yes") -or ($LinkedToGroup -eq "all")) {
$LDAPFilter = "(&(objectClass=msPKI-Enterprise-Oid)(msDS-OIDToGroupLink=*)(flags=2))"
$LinkedOIDs = get-adobject -searchBase $configNCDN -LDAPFilter $LDAPFilter -properties *
write-host ""
write-host "*****"
write-host $getIP_strings.LinkedIPs
write-host "*****"
write-host ""
if ($LinkedOIDs -ne $null){
    foreach ($OID in $LinkedOIDs) {

```

```

# Display basic information about the Issuance Policies
""
$getIP_strings.displayName -f $OID.displayName
$getIP_strings.Name -f $OID.Name
$getIP_strings.dn -f $OID.distinguishedName
# Get the linked group.
$groupDN = $OID."msDS-OIDToGroupLink"
$group = get-adgroup -Identity $groupDN
$getIP_strings.InfoName -f $group.Name
$getIP_strings.InfoDN -f $groupDN
# Analyze the group
$OIDName = $OID.displayName
$groupName = $group.Name
if ($group.groupCategory -ne "Security") {
$errormsg = $getIP_strings.ErrorNotSecurity -f $OIDName, $groupName
write-host $errmsg -ForegroundColor Red
}
if ($group.groupScope -ne "Universal") {
$errormsg = $getIP_strings.ErrorNotUniversal -f $OIDName, $groupName
write-host $errmsg -ForegroundColor Red
}
$members = Get-ADGroupMember -Identity $group
if ($members) {
$errormsg = $getIP_strings.ErrorHasMembers -f $OIDName, $groupName
write-host $errmsg -ForegroundColor Red
    foreach ($member in $members) {
        write-host "        " $member -ForegroundColor Red
    }
}
write-host ""
}
}else{
write-host "There are no issuance policies that are mapped to a group"
}
if ($LinkedToGroup -eq "yes") {
return $LinkedOIDs
break
}
}
if (($LinkedToGroup -eq "no") -or ($LinkedToGroup -eq "all")) {
$LDAPFilter = "(&(objectClass=msPKI-Enterprise-Oid)!(msDS-OIDToGroupLink=*))&(flags=2)"
$NonLinkedOIDs = get-adobject -searchBase $configNCDN -LDAPFilter $LDAPFilter -properties *
write-host ""
write-host "*****"
write-host $getIP_strings.NonLinkedIPs
write-host "*****"
write-host ""
if ($NonLinkedOIDs -ne $null) {
foreach ($OID in $NonLinkedOIDs) {
# Display basic information about the Issuance Policies
write-host ""
$getIP_strings.displayName -f $OID.displayName
$getIP_strings.Name -f $OID.Name
$getIP_strings.dn -f $OID.distinguishedName
write-host ""
}
}else{
write-host "There are no issuance policies which are not mapped to groups"
}
if ($LinkedToGroup -eq "no") {
return $NonLinkedOIDs
break
}
}
}

```


NOTE

If you're having trouble running this script, try replacing the single quote after the ConvertFrom-StringData parameter.

Link an issuance policy to a group

Save the script file as set-IssuancePolicyToGroupLink.ps1.

```
#####
## Parameters to be defined ##
## by the user ##
#####
Param (
$IssuancePolicyName,
$groupOU,
$groupName
)
#####
## Strings definitions ##
#####
Data ErrorMsg {
# culture="en-US"
ConvertFrom-StringData -stringdata @'
help1 = This command can be used to set the link between a certificate issuance policy and a universal
security group.
help2 = Usage:
help3 = The following parameters are required:
help4 = -IssuancePolicyName:<name or display name of the issuance policy that you want to link to a group>
help5 = -groupName:<name of the group you want to link the issuance policy to>. If no name is specified, any
existing link to a group is removed from the Issuance Policy.
help6 = The following parameter is optional:
help7 = -groupOU:<Name of the Organizational Unit dedicated to the groups which are linked to issuance
policies>. If this parameter is not specified, the group is looked for or created in the Users container.
help8 = Examples:
help9 = This command will link the issuance policy whose display name is "High Assurance" to the group
"HighAssuranceGroup" in the Organizational Unit "OU_FOR_IPol_linked_groups". If the group or the
Organizational Unit do not exist, you will be prompted to create them.
help10 = This command will unlink the issuance policy whose name is "402.164959C40F4A5C12C6302E31D5476062"
from any group.
MultipleIPs = Error: Multiple Issuance Policies with name or display name "{0}" were found in the subtree of
"{1}"
NoIP = Error: no issuance policy with name or display name "{0}" could be found in the subtree of "{1}".
IPFound = An Issuance Policy with name or display name "{0}" was successfully found: {1}
MultipleOUs = Error: more than 1 Organizational Unit with name "{0}" could be found in the subtree of "{1}".
confirmOUcreation = Warning: The Organizational Unit that you specified does not exist. Do you want to
create it?
OUCreationSuccess = Organizational Unit "{0}" successfully created.
OUcreationError = Error: Organizational Unit "{0}" could not be created.
OUFoundSuccess = Organizational Unit "{0}" was successfully found.
multipleGroups = Error: More than one group with name "{0}" was found in Organizational Unit "{1}".
confirmGroupCreation = Warning: The group that you specified does not exist. Do you want to create it?
groupCreationSuccess = Universal Security group "{0}" successfully created.
groupCreationError = Error: Universal Security group "{0}" could not be created.
GroupFound = Group "{0}" was successfully found.
confirmLinkDeletion = Warning: The Issuance Policy "{0}" is currently linked to group "{1}". Do you really
want to remove the link?
UnlinkSuccess = Certificate issuance policy successfully unlinked from any group.
UnlinkError = Removing the link failed.
UnlinkExit = Exiting without removing the link from the issuance policy to the group.
IPNotLinked = The Certificate issuance policy is not currently linked to any group. If you want to link it
to a group, you should specify the -groupName option when starting this script.
ErrorNotSecurity = Error: You cannot link issuance Policy "{0}" to group "{1}" because this group is not of
type "Security".
ErrorNotUniversal = Error: You cannot link issuance Policy "{0}" to group "{1}" because the scope of this
group is not "Universal".
ErrorHasMembers = Error: You cannot link issuance Policy "{0}" to group "{1}" because it has a non-empty
```

```

membership. The group has the following members:
ConfirmLinkReplacement = Warning: The Issuance Policy "{0}" is currently linked to group "{1}". Do you
really want to update the link to point to group "{2}"?
LinkSuccess = The certificate issuance policy was successfully linked to the specified group.
LinkError = The certificate issuance policy could not be linked to the specified group.
ExitNoLinkReplacement = Exiting without setting the new link.
'@
}
# import-localizeddata ErrorMsg
function Display-Help {
""
write-host $ErrorMsg.help1
""
write-host $ErrorMsg.help2
""
write-host $ErrorMsg.help3
write-host "`t" $ErrorMsg.help4
write-host "`t" $ErrorMsg.help5
""
write-host $ErrorMsg.help6
write-host "`t" $ErrorMsg.help7
""
""
write-host $ErrorMsg.help8
""
write-host $ErrorMsg.help9
".\Set-IssuancePolicyToGroupMapping.ps1 -IssuancePolicyName "High Assurance" -groupOU
"OU_FOR_IPol_linked_groups" -groupName "HighAssuranceGroup" "
""
write-host $ErrorMsg.help10
'.\Set-IssuancePolicyToGroupMapping.ps1 -IssuancePolicyName "402.164959C40F4A5C12C6302E31D5476062" -
groupName $null '
""
}
# Assumption: The group to which the Issuance Policy is going
#             to be linked is (or is going to be created) in
#             the domain the user running this script is a member of.
import-module ActiveDirectory
$root = get-adrootdse
$domain = get-addomain -current loggedonuser
if ( !($IssuancePolicyName) ) {
display-Help
break
}
#####
## Find the OID object ##
## (aka Issuance Policy) ##
#####
$searchBase = [String]$root.configurationnamingcontext
$OID = get-adobject -searchBase $searchBase -Filter { ((displayname -eq $IssuancePolicyName) -or (name -eq
$IssuancePolicyName)) -and (objectClass -eq "msPKI-Enterprise-Oid")} -properties *
if ($OID -eq $null) {
$tmp = $ErrorMsg.NoIP -f $IssuancePolicyName, $searchBase
write-host $tmp -ForegroundColor Red
break;
}
elseif ($OID.GetType().IsArray) {
$tmp = $ErrorMsg.MultipleIPs -f $IssuancePolicyName, $searchBase
write-host $tmp -ForegroundColor Red
break;
}
else {
$tmp = $ErrorMsg.IPFound -f $IssuancePolicyName, $OID.distinguishedName
write-host $tmp -ForegroundColor Green
}
#####
## Find the container of the group ##
#####
if ($groupOU -eq $null) {

```

```

# default to the Users container
$groupContainer = $domain.UsersContainer
}
else {
$searchBase = [string]$domain.DistinguishedName
$groupContainer = get-adobject -searchBase $searchBase -Filter { (Name -eq $groupOU) -and (objectClass -eq
"organizationalUnit")}
if ($groupContainer.count -gt 1) {
$tmp = $ErrorMsg.MultipleOUs -f $groupOU, $searchBase
write-host $tmp -ForegroundColor Red
break;
}
elseif ($groupContainer -eq $null) {
$tmp = $ErrorMsg.confirmOUcreation
write-host $tmp " ( (y)es / (n)o )" -ForegroundColor Yellow -nonewline
$userChoice = read-host
if ( ($userChoice -eq "y") -or ($userChoice -eq "yes") ) {
new-adobject -Name $groupOU -displayName $groupOU -Type "organizationalUnit" -
ProtectedFromAccidentalDeletion $true -path $domain.distinguishedName
if ($?){
$tmp = $ErrorMsg.OUCreationSuccess -f $groupOU
write-host $tmp -ForegroundColor Green
}
else{
$tmp = $ErrorMsg.OUCreationError -f $groupOU
write-host $tmp -ForegroundColor Red
break;
}
$groupContainer = get-adobject -searchBase $searchBase -Filter { (Name -eq $groupOU) -and (objectClass -eq
"organizationalUnit")}
}
else {
break;
}
}
else {
$tmp = $ErrorMsg.OUFoundSuccess -f $groupContainer.name
write-host $tmp -ForegroundColor Green
}
}
#####
## Find the group ##
#####
if (($groupName -ne $null) -and ($groupName -ne "")){
##$searchBase = [String]$groupContainer.DistinguishedName
$searchBase = $groupContainer
$group = get-adgroup -Filter { (Name -eq $groupName) -and (objectClass -eq "group") } -searchBase
$searchBase
if ($group -ne $null -and $group.gettype().isArray) {
$tmp = $ErrorMsg.multipleGroups -f $groupName, $searchBase
write-host $tmp -ForegroundColor Red
break;
}
elseif ($group -eq $null) {
$tmp = $ErrorMsg.confirmGroupCreation
write-host $tmp " ( (y)es / (n)o )" -ForegroundColor Yellow -nonewline
$userChoice = read-host
if ( ($userChoice -eq "y") -or ($userChoice -eq "yes") ) {
new-adgroup -samAccountName $groupName -path $groupContainer.distinguishedName -GroupScope "Universal" -
GroupCategory "Security"
if ($?){
$tmp = $ErrorMsg.GroupCreationSuccess -f $groupName
write-host $tmp -ForegroundColor Green
}
else{
$tmp = $ErrorMsg.groupCreationError -f $groupName
write-host $tmp -ForegroundColor Red
break
}
}
$group = get-adgroup -Filter { (Name -eq $groupName) -and (objectClass -eq "group") } -searchBase

```

```

$searchBase
}
else {
break;
}
}
else {
$tmp = $ErrorMsg.GroupFound -f $group.Name
write-host $tmp -ForegroundColor Green
}
}
else {
#####
## If the group is not specified, we should remove the link if any exists
#####
if ($OID."msDS-OIDToGroupLink" -ne $null) {
$tmp = $ErrorMsg.confirmLinkDeletion -f $IssuancePolicyName, $OID."msDS-OIDToGroupLink"
write-host $tmp " ( (y)es / (n)o )" -ForegroundColor Yellow -nonewline
$userChoice = read-host
if ( ($userChoice -eq "y") -or ($userChoice -eq "yes") ) {
set-adobject -Identity $OID -Clear "msDS-OIDToGroupLink"
if ($?) {
$tmp = $ErrorMsg.UnlinkSuccess
write-host $tmp -ForegroundColor Green
}else{
$tmp = $ErrorMsg.UnlinkError
write-host $tmp -ForegroundColor Red
}
}
else {
$tmp = $ErrorMsg.UnlinkExit
write-host $tmp
break
}
}
else {
$tmp = $ErrorMsg.IPNotLinked
write-host $tmp -ForegroundColor Yellow
}
break;
}
#####
## Verify that the group is      ##
## Universal, Security, and      ##
## has no members                ##
#####
if ($group.GroupScope -ne "Universal") {
$tmp = $ErrorMsg.ErrorNotUniversal -f $IssuancePolicyName, $groupName
write-host $tmp -ForegroundColor Red
break;
}
if ($group.GroupCategory -ne "Security") {
$tmp = $ErrorMsg.ErrorNotSecurity -f $IssuancePolicyName, $groupName
write-host $tmp -ForegroundColor Red
break;
}
$members = Get-ADGroupMember -Identity $group
if ($members -ne $null) {
$tmp = $ErrorMsg.ErrorHasMembers -f $IssuancePolicyName, $groupName
write-host $tmp -ForegroundColor Red
foreach ($member in $members) {write-host "  $member.name" -ForegroundColor Red}
break;
}
#####
## We have verified everything. We ##
## can create the link from the    ##
## Issuance Policy to the group.   ##
#####
if ($OID."msDS-OIDToGroupLink" -ne $null) {

```

```
$tmp = $ErrorMsg.ConfirmLinkReplacement -f $IssuancePolicyName, $OID."msDS-OIDToGroupLink",
$group.distinguishedName
write-host $tmp "( (y)es / (n)o )" -ForegroundColor Yellow -nonewline
$userChoice = read-host
if ( ($userChoice -eq "y") -or ($userChoice -eq "yes") ) {
$tmp = @{'msDS-OIDToGroupLink'= $group.DistinguishedName}
set-adobject -Identity $OID -Replace $tmp
if ($?) {
$tmp = $ErrorMsg.LinkSuccess
write-host $tmp -ForegroundColor Green
}else{
$tmp = $ErrorMsg.LinkError
write-host $tmp -ForegroundColor Red
}
} else {
$tmp = $ErrorMsg.ExitNoLinkReplacement
write-host $tmp
break
}
}
else {
$tmp = @{'msDS-OIDToGroupLink'= $group.DistinguishedName}
set-adobject -Identity $OID -Add $tmp
if ($?) {
$tmp = $ErrorMsg.LinkSuccess
write-host $tmp -ForegroundColor Green
}else{
$tmp = $ErrorMsg.LinkError
write-host $tmp -ForegroundColor Red
}
}
}
```

NOTE

If you're having trouble running this script, try replacing the single quote after the ConvertFrom-StringData parameter.

Windows Defender Credential Guard: Known issues

7/1/2022 • 4 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016
- Windows Server 2019

Windows Defender Credential Guard has certain application requirements. Windows Defender Credential Guard blocks specific authentication capabilities. So applications that require such capabilities won't function when it's enabled. For more information, see [Application requirements](#).

The following known issue has been fixed in the [Cumulative Security Update for November 2017](#):

- Scheduled tasks with domain user-stored credentials fail to run when Credential Guard is enabled. The task fails and reports Event ID 104 with the following message:
"Task Scheduler failed to log on '\Test'.
Failure occurred in 'LogonUserExEx'.
User Action: Ensure the credentials for the task are correctly specified.
Additional Data: Error Value: 2147943726. 2147943726: ERROR_LOGON_FAILURE (The user name or password is incorrect)."
- When enabling NTLM audit on the domain controller, an Event ID 8004 with an indecipherable username format is logged. You also get a similar user name in a user logon failure event 4625 with error 0xC0000064 on the machine itself. For example:

```
Log Name: Microsoft-Windows-NTLM/Operational
Source: Microsoft-Windows-Security-Netlogon
Event ID: 8004
Task Category: Auditing NTLM
Level: Information
Description:
Domain Controller Blocked Audit: Audit NTLM authentication to this domain controller.
Secure Channel name: <Secure Channel Name>
User name:
@@@CyBAAAUBQYAMHArBwUAMGAoBQZAQGA1BAbAUGAyBgOAFhBwcAsGA6AweAgDA2AQQ
AMEAwAANAgDA1AQLAIEADBQRAADAtAANAYEA1AwQA0CA5AAOAMEAyAQLAYDaxAwQAEDAEB
wMAMEAwAgMAMDACBgRA0HA
Domain name: NULL
```

- This event stems from a scheduled task running under local user context with the [Cumulative Security Update for November 2017](#) or later and happens when Credential Guard is enabled.
- The username appears in an unusual format because local accounts aren't protected by Credential Guard. The task also fails to execute.
- As a workaround, run the scheduled task under a domain user or the computer's SYSTEM account.

The following known issues have been fixed by servicing releases made available in the Cumulative Security Updates for April 2017:

- [KB4015217 Windows Defender Credential Guard generates double bad password count on Active Directory domain-joined Windows machines](#)

This issue can potentially lead to unexpected account lockouts. See also Microsoft® Knowledge Base articles [KB4015219](#) and [KB4015221](#)

Known issues involving third-party applications

The following issue affects MSCHAPv2:

- [Credential guard doesn't work with MSCHAPv2 configurations, of which Cisco ISE is a very popular enterprise implementation.](#)

The following issue affects the Java GSS API. See the following Oracle bug database article:

- [JDK-8161921: Windows Defender Credential Guard doesn't allow sharing of TGT with Java](#)

When Windows Defender Credential Guard is enabled on Windows, the Java GSS API won't authenticate. This is expected behavior because Windows Defender Credential Guard blocks specific application authentication capabilities and won't provide the TGT session key to applications regardless of registry key settings. For more information, see [Application requirements](#).

The following issue affects Cisco AnyConnect Secure Mobility Client:

- [Blue screen on Windows computers running Hypervisor-Protected Code Integrity and Windows Defender Credential Guard with Cisco Anyconnect 4.3.04027 *](#)

*Registration required to access this article.

The following issue affects McAfee Application and Change Control (MACC):

- [KB88869 Windows machines exhibit high CPU usage with McAfee Application and Change Control \(MACC\) installed when Windows Defender Credential Guard is enabled ^{\[1\]}](#)

The following issue affects AppSense Environment Manager. For more information, see the following Knowledge Base article:

- [Installing AppSense Environment Manager on Windows machines causes LSAISO.exe to exhibit high CPU usage when Windows Defender Credential Guard is enabled ^{\[1\]} **](#)

The following issue affects Citrix applications:

- [Windows machines exhibit high CPU usage with Citrix applications installed when Windows Defender Credential Guard is enabled. ^{\[1\]}](#)

^[1] Products that connect to Virtualization Based Security (VBS) protected processes can cause Windows Defender Credential Guard-enabled Windows 10, Windows 11, Windows Server 2016, or Windows Server 2019 machines to exhibit high CPU usage. For technical and troubleshooting information, see the following Microsoft Knowledge Base article:

- [KB4032786 High CPU usage in the LSAISO process on Windows](#)

For further technical information on LSAISO.exe, see the MSDN article: [Isolated User Mode \(IUM\) Processes](#)

** Registration is required to access this article.

Vendor support

See the following article on Citrix support for Secure Boot:

- [Citrix Support for Secure Boot](#)

Windows Defender Credential Guard isn't supported by either these products, products versions, computer systems, or Windows 10 versions:

- For Windows Defender Credential Guard on Windows with McAfee Encryption products, see: [Support for Hypervisor-Protected Code Integrity and Windows Defender Credential Guard on Windows with McAfee encryption products](#)
- For Windows Defender Credential Guard on Windows with Check Point Endpoint Security Client, see: [Check Point Endpoint Security Client support for Microsoft Windows Defender Credential Guard and Hypervisor-Protected Code Integrity features](#)
- For Windows Defender Credential Guard on Windows with VMWare Workstation [Windows host fails when running VMWare Workstation when Windows Defender Credential Guard is enabled](#)
- For Windows Defender Credential Guard on Windows with specific versions of the Lenovo ThinkPad [ThinkPad support for Hypervisor-Protected Code Integrity and Windows Defender Credential Guard in Microsoft Windows – ThinkPad](#)
- For Windows Defender Credential Guard on Windows with Symantec Endpoint Protection [Windows devices with Windows Defender Credential Guard and Symantec Endpoint Protection 12.1](#)

This isn't a comprehensive list. Check whether your product vendor, product version, or computer system, supports Windows Defender Credential Guard on systems that run Windows or specific versions of Windows. Specific computer system models may be incompatible with Windows Defender Credential Guard.

Microsoft encourages third-party vendors to contribute to this page by providing relevant product support information and by adding links to their own product support statements.

Protect Remote Desktop credentials with Windows Defender Remote Credential Guard

7/1/2022 • 8 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows Server 2016

Introduced in Windows 10, version 1607, Windows Defender Remote Credential Guard helps you protect your credentials over a Remote Desktop connection by redirecting Kerberos requests back to the device that's requesting the connection. It also provides single sign-on experiences for Remote Desktop sessions.

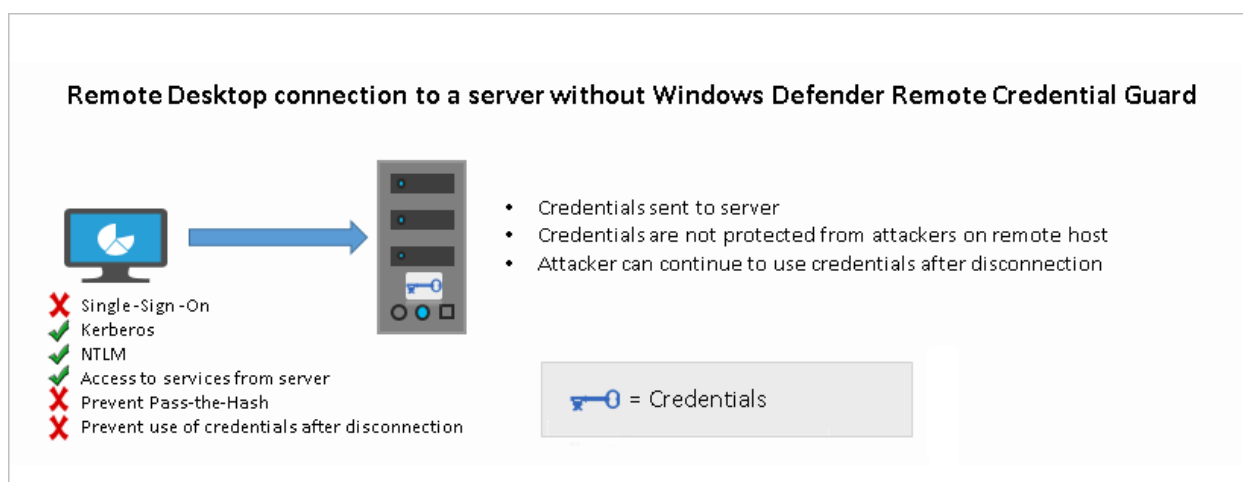
Administrator credentials are highly privileged and must be protected. By using Windows Defender Remote Credential Guard to connect during Remote Desktop sessions, if the target device is compromised, your credentials are not exposed because both credential and credential derivatives are never passed over the network to the target device.

IMPORTANT

For information on Remote Desktop connection scenarios involving helpdesk support, see [Remote Desktop connections and helpdesk support scenarios](#) in this article.

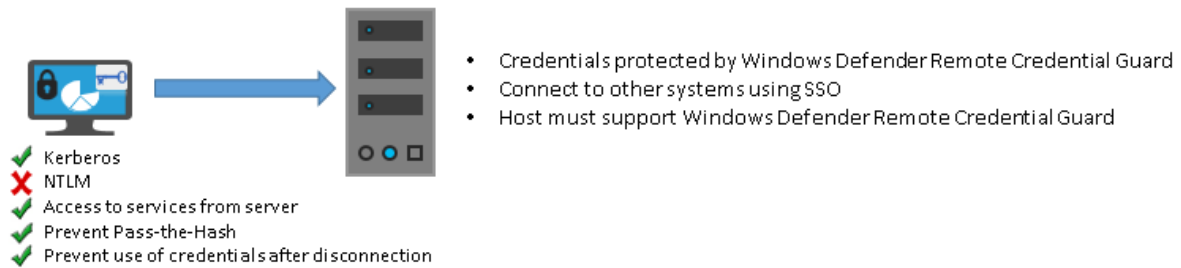
Comparing Windows Defender Remote Credential Guard with other Remote Desktop connection options

The following diagram helps you to understand how a standard Remote Desktop session to a server without Windows Defender Remote Credential Guard works:

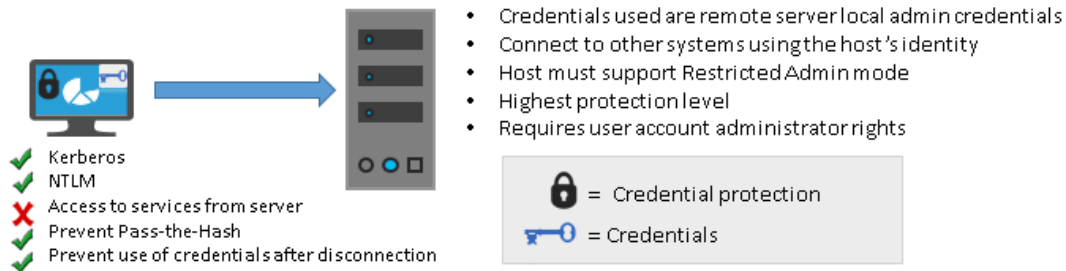


The following diagram helps you to understand how Windows Defender Remote Credential Guard works, what it helps to protect against, and compares it with the [Restricted Admin mode](#) option:

Windows Defender Remote Credential Guard



Restricted Admin Mode



As illustrated, Windows Defender Remote Credential Guard blocks NTLM (allowing only Kerberos), prevents Pass-the-Hash (PtH) attacks, and also prevents use of credentials after disconnection.

Use the following table to compare different Remote Desktop connection security options:

FEATURE	REMOTE DESKTOP	WINDOWS DEFENDER REMOTE CREDENTIAL GUARD	RESTRICTED ADMIN MODE
Protection benefits	Credentials on the server are not protected from Pass-the-Hash attacks.	User credentials remain on the client. An attacker can act on behalf of the user <i>only</i> when the session is ongoing	User logs on to the server as local administrator, so an attacker cannot act on behalf of the "domain user". Any attack is local to the server
Version support	The remote computer can run any Windows operating system	Both the client and the remote computer must be running at least Windows 10, version 1607, or Windows Server 2016 .	The remote computer must be running at least patched Windows 7 or patched Windows Server 2008 R2 . For more information about patches (software updates) related to Restricted Admin mode, see Microsoft Security Advisory 2871997 .
Helps prevent	N/A	<ul style="list-style-type: none"> • Pass-the-Hash • Use of a credential after disconnection 	<ul style="list-style-type: none"> • Pass-the-Hash • Use of domain identity during connection

FEATURE	REMOTE DESKTOP	WINDOWS DEFENDER REMOTE CREDENTIAL GUARD	RESTRICTED ADMIN MODE
Credentials supported from the remote desktop client device	<ul style="list-style-type: none"> • Signed on credentials • Supplied credentials • Saved credentials 	<ul style="list-style-type: none"> • Signed on credentials only 	<ul style="list-style-type: none"> • Signed on credentials • Supplied credentials • Saved credentials
Access	Users allowed, that is, members of Remote Desktop Users group of remote host.	Users allowed, that is, members of Remote Desktop Users of remote host.	Administrators only, that is, only members of Administrators group of remote host.
Network identity	Remote Desktop session connects to other resources as signed-in user.	Remote Desktop session connects to other resources as signed-in user.	Remote Desktop session connects to other resources as remote host's identity.
Multi-hop	From the remote desktop, you can connect through Remote Desktop to another computer	From the remote desktop, you can connect through Remote Desktop to another computer.	Not allowed for user as the session is running as a local host account
Supported authentication	Any negotiable protocol.	Kerberos only.	Any negotiable protocol

For further technical information, see [Remote Desktop Protocol](#) and [How Kerberos works](#).

Remote Desktop connections and helpdesk support scenarios

For helpdesk support scenarios in which personnel require administrative access to provide remote assistance to computer users via Remote Desktop sessions, Microsoft recommends that Windows Defender Remote Credential Guard should not be used in that context. This is because if an RDP session is initiated to a compromised client that an attacker already controls, the attacker could use that open channel to create sessions on the user's behalf (without compromising credentials) to access any of the user's resources for a limited time (a few hours) after the session disconnects.

Therefore, we recommend instead that you use the Restricted Admin mode option. For helpdesk support scenarios, RDP connections should only be initiated using the `/RestrictedAdmin` switch. This helps ensure that credentials and other user resources are not exposed to compromised remote hosts. For more information, see [Mitigating Pass-the-Hash and Other Credential Theft v2](#).

To further harden security, we also recommend that you implement Local Administrator Password Solution (LAPS), a Group Policy client-side extension (CSE) introduced in Windows 8.1 that automates local administrator password management. LAPS mitigates the risk of lateral escalation and other cyberattacks facilitated when customers use the same administrative local account and password combination on all their computers. You can download and install LAPS [here](#).

For further information on LAPS, see [Microsoft Security Advisory 3062591](#).

Remote Credential Guard requirements

To use Windows Defender Remote Credential Guard, the Remote Desktop client and remote host must meet the following requirements:

The Remote Desktop client device:

- Must be running at least Windows 10, version 1703 to be able to supply credentials, which is sent to the remote device. This allows users to run as different users without having to send credentials to the remote machine.
- Must be running at least Windows 10, version 1607 or Windows Server 2016 to use the user's signed-in credentials. This requires the user's account be able to sign in to both the client device and the remote host.
- Must be running the Remote Desktop Classic Windows application. The Remote Desktop Universal Windows Platform application doesn't support Windows Defender Remote Credential Guard.
- Must use Kerberos authentication to connect to the remote host. If the client cannot connect to a domain controller, then RDP attempts to fall back to NTLM. Windows Defender Remote Credential Guard does not allow NTLM fallback because this would expose credentials to risk.

The Remote Desktop remote host:

- Must be running at least Windows 10, version 1607 or Windows Server 2016.
- Must allow Restricted Admin connections.
- Must allow the client's domain user to access Remote Desktop connections.
- Must allow delegation of non-exportable credentials.

There are no hardware requirements for Windows Defender Remote Credential Guard.

NOTE

Remote Desktop client devices running earlier versions, at minimum Windows 10 version 1607, only support signed-in credentials, so the client device must also be joined to an Active Directory domain. Both Remote Desktop client and server must either be joined to the same domain, or the Remote Desktop server can be joined to a domain that has a trust relationship to the client device's domain.

GPO [Remote host allows delegation of non-exportable credentials](#) should be enabled for delegation of non-exportable credentials.

- For Windows Defender Remote Credential Guard to be supported, the user must authenticate to the remote host using Kerberos authentication.
- The remote host must be running at least Windows 10 version 1607, or Windows Server 2016.
- The Remote Desktop classic Windows app is required. The Remote Desktop Universal Windows Platform app doesn't support Windows Defender Remote Credential Guard.

Enable Windows Defender Remote Credential Guard

You must enable Restricted Admin or Windows Defender Remote Credential Guard on the remote host by using the Registry.

1. Open Registry Editor on the remote host.
2. Enable Restricted Admin and Windows Defender Remote Credential Guard:

- Go to HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa.
- Add a new DWORD value named **DisableRestrictedAdmin**.
- To turn on Restricted Admin and Windows Defender Remote Credential Guard, set the value of this registry setting to 0 to turn on Windows Defender Remote Credential Guard.

3. Close Registry Editor.

You can add this by running the following command from an elevated command prompt:

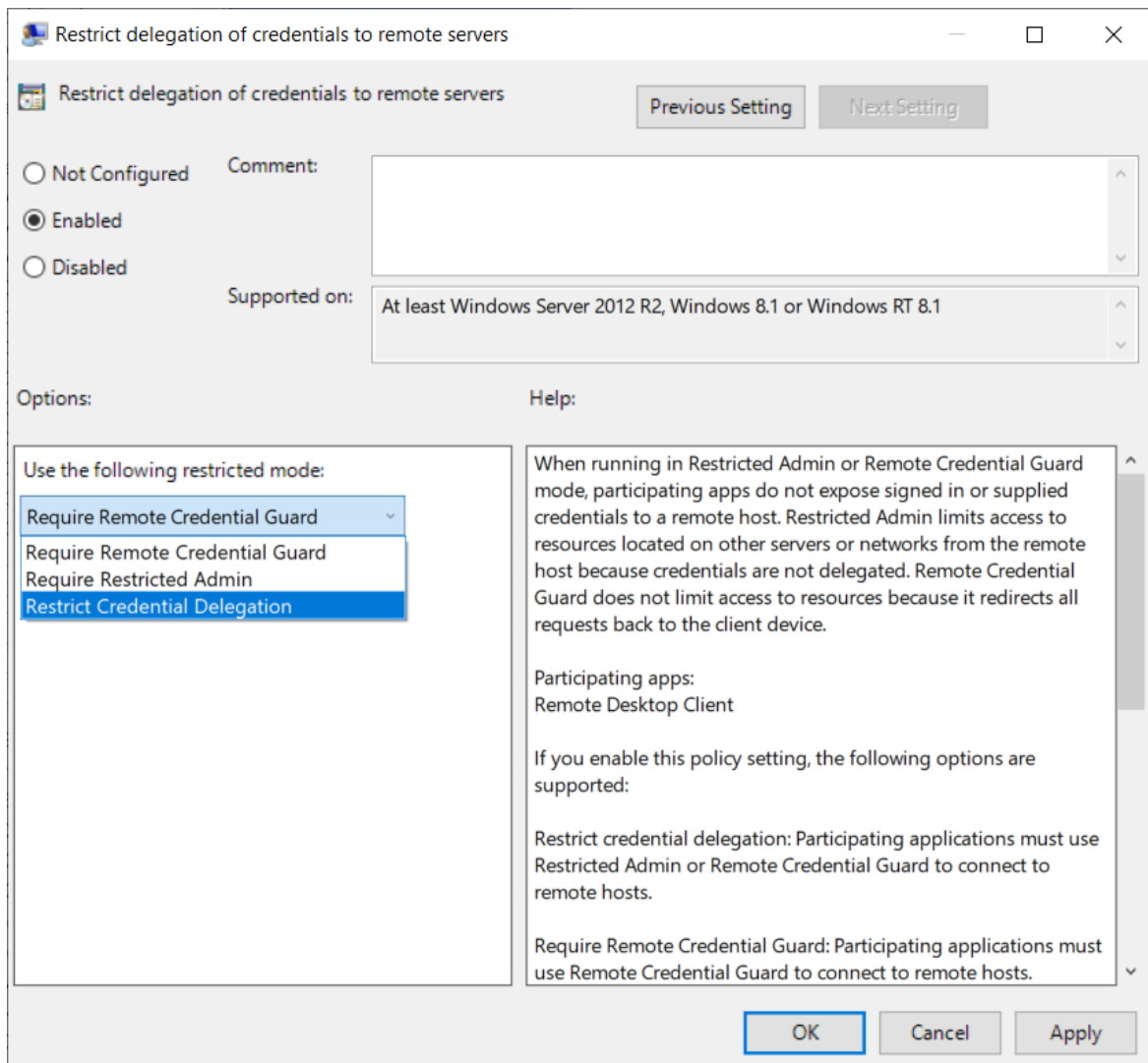
```
reg add HKLM\SYSTEM\CurrentControlSet\Control\Lsa /v DisableRestrictedAdmin /d 0 /t REG_DWORD
```

Using Windows Defender Remote Credential Guard

Beginning with Windows 10 version 1703, you can enable Windows Defender Remote Credential Guard on the client device either by using Group Policy or by using a parameter with the Remote Desktop Connection.

Turn on Windows Defender Remote Credential Guard by using Group Policy

1. From the Group Policy Management Console, go to **Computer Configuration -> Administrative Templates -> System -> Credentials Delegation**.
2. Double-click **Restrict delegation of credentials to remote servers**.



3. Under **Use the following restricted mode**:

- If you want to require either [Restricted Admin mode](#) or Windows Defender Remote Credential

Guard, choose **Restrict Credential Delegation**. In this configuration, Windows Defender Remote Credential Guard is preferred, but it will use Restricted Admin mode (if supported) when Windows Defender Remote Credential Guard cannot be used.

NOTE

Neither Windows Defender Remote Credential Guard nor Restricted Admin mode will send credentials in clear text to the Remote Desktop server.

- If you want to require Windows Defender Remote Credential Guard, choose **Require Remote Credential Guard**. With this setting, a Remote Desktop connection will succeed only if the remote computer meets the [requirements](#) listed earlier in this topic.
- If you want to require Restricted Admin mode, choose **Require Restricted Admin**. For information about Restricted Admin mode, see the table in [Comparing Windows Defender Remote Credential Guard with other Remote Desktop connection options](#), earlier in this topic.

4. Click OK.

5. Close the Group Policy Management Console.

6. From a command prompt, run `gpupdate.exe /force` to ensure that the Group Policy object is applied.

Use Windows Defender Remote Credential Guard with a parameter to Remote Desktop Connection

If you don't use Group Policy in your organization, or if not all your remote hosts support Remote Credential Guard, you can add the `remoteGuard` parameter when you start Remote Desktop Connection to turn on Windows Defender Remote Credential Guard for that connection.

```
mstsc.exe /remoteGuard
```

NOTE

The user must be authorized to connect to the remote server using Remote Desktop Protocol, for example by being a member of the Remote Desktop Users local group on the remote computer.

Considerations when using Windows Defender Remote Credential Guard

- Windows Defender Remote Credential Guard does not support compound authentication. For example, if you're trying to access a file server from a remote host that requires a device claim, access will be denied.
- Windows Defender Remote Credential Guard can be used only when connecting to a device that is joined to a Windows Server Active Directory domain, including AD domain-joined servers that run as Azure virtual machines (VMs). Windows Defender Remote Credential Guard cannot be used when connecting to remote devices joined to Azure Active Directory.
- Remote Desktop Credential Guard only works with the RDP protocol.
- No credentials are sent to the target device, but the target device still acquires Kerberos Service Tickets on its own.
- The server and client must authenticate using Kerberos.

Technical support policy for lost or forgotten passwords

7/1/2022 • 2 minutes to read • [Edit Online](#)

Microsoft takes security seriously. This is for your protection. Microsoft accounts, the Windows operating system, and other Microsoft products include passwords to help secure your information. This article provides some options that you can use to reset or recover your password if you forget it. Be aware that, if these options don't work, Microsoft support engineers can't help you retrieve or circumvent a lost or forgotten password.

If you lose or forget a password, you can use the links in this article to find published support information that will help you reset the password.

How to reset a password for a domain account

If you lose or forget the password for a domain account, contact your IT administrator or Helpdesk. For more information, see [Change or reset your Windows password](#).

How to reset a password for a Microsoft account

If you lose or forget the password for your Microsoft Account, use the [Recover your account](#) wizard.

This wizard requests your security proofs. If you have forgotten your security proofs, or no longer have access to them, select **I no longer have these anymore**. After you select this option, fill out a form for the Microsoft Account team. Provide as much information as you can on this form. The Microsoft Account team reviews the information that you provide to determine whether you are the account holder. This decision is final. Microsoft does not influence the team's choice of action.

How to reset a password for a local account on a Windows device

Local accounts on a device include the device's Administrator account.

Windows 10

If you lose or forget the password for a local account on a device that runs Windows 10, see [Reset your Windows 10 local account password](#).

Windows 8.1 or Windows 7

If you lose or forget the password for a local account on a device that runs Windows 8.1 or Windows 7, see [Change or reset your Windows password](#). In that article, you can select your operating system version from the **Select Product Version** menu.

How to reset a hardware BIOS password

If you lose or forget the password for the hardware BIOS of a device, contact the device manufacturer for help and support. If you do contact the manufacturer online, make sure that you visit the manufacturer website and not the website of some third party.

How to reset a password for an individual file

Some applications let you password-protect individual files. If you lose or forget such a password, you can rely on that application only to reset or recover it. Microsoft support engineers cannot help you reset, retrieve, or

circumvent such passwords.

Using third-party password tools

Some third-party companies claim to be able to circumvent passwords that have been applied to files and features that Microsoft programs use. For legal reasons, we cannot recommend or endorse any one of these companies. If you want help to circumvent or reset a password, you can locate and contact a third party for this help. However, you use such third-party products and services at your own risk.

Access Control Overview

7/1/2022 • 6 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows Server 2016

This topic for the IT professional describes access control in Windows, which is the process of authorizing users, groups, and computers to access objects on the network or computer. Key concepts that make up access control are permissions, ownership of objects, inheritance of permissions, user rights, and object auditing.

Feature description

Computers that are running a supported version of Windows can control the use of system and network resources through the interrelated mechanisms of authentication and authorization. After a user is authenticated, the Windows operating system uses built-in authorization and access control technologies to implement the second phase of protecting resources: determining if an authenticated user has the correct permissions to access a resource.

Shared resources are available to users and groups other than the resource's owner, and they need to be protected from unauthorized use. In the access control model, users and groups (also referred to as security principals) are represented by unique security identifiers (SIDs). They are assigned rights and permissions that inform the operating system what each user and group can do. Each resource has an owner who grants permissions to security principals. During the access control check, these permissions are examined to determine which security principals can access the resource and how they can access it.

Security principals perform actions (which include Read, Write, Modify, or Full control) on objects. Objects include files, folders, printers, registry keys, and Active Directory Domain Services (AD DS) objects. Shared resources use access control lists (ACLs) to assign permissions. This enables resource managers to enforce access control in the following ways:

- Deny access to unauthorized users and groups
- Set well-defined limits on the access that is provided to authorized users and groups

Object owners generally grant permissions to security groups rather than to individual users. Users and computers that are added to existing groups assume the permissions of that group. If an object (such as a folder) can hold other objects (such as subfolders and files), it is called a container. In a hierarchy of objects, the relationship between a container and its content is expressed by referring to the container as the parent. An object in the container is referred to as the child, and the child inherits the access control settings of the parent. Object owners often define permissions for container objects, rather than individual child objects, to ease access control management.

This content set contains:

- [Dynamic Access Control Overview](#)
- [Security identifiers](#)
- [Security Principals](#)
 - [Local Accounts](#)

- [Active Directory Accounts](#)
- [Microsoft Accounts](#)
- [Service Accounts](#)
- [Active Directory Security Groups](#)

Practical applications

Administrators who use the supported version of Windows can refine the application and management of access control to objects and subjects to provide the following security:

- Protect a greater number and variety of network resources from misuse.
- Provision users to access resources in a manner that is consistent with organizational policies and the requirements of their jobs.
- Enable users to access resources from a variety of devices in numerous locations.
- Update users' ability to access resources on a regular basis as an organization's policies change or as users' jobs change.
- Account for a growing number of use scenarios (such as access from remote locations or from a rapidly expanding variety of devices, such as tablet computers and mobile phones).
- Identify and resolve access issues when legitimate users are unable to access resources that they need to perform their jobs.

Permissions

Permissions define the type of access that is granted to a user or group for an object or object property. For example, the Finance group can be granted Read and Write permissions for a file named Payroll.dat.

By using the access control user interface, you can set NTFS permissions for objects such as files, Active Directory objects, registry objects, or system objects such as processes. Permissions can be granted to any user, group, or computer. It is a good practice to assign permissions to groups because it improves system performance when verifying access to an object.

For any object, you can grant permissions to:

- Groups, users, and other objects with security identifiers in the domain.
- Groups and users in that domain and any trusted domains.
- Local groups and users on the computer where the object resides.

The permissions attached to an object depend on the type of object. For example, the permissions that can be attached to a file are different from those that can be attached to a registry key. Some permissions, however, are common to most types of objects. These common permissions are:

- Read
- Modify
- Change owner
- Delete

When you set permissions, you specify the level of access for groups and users. For example, you can let one user read the contents of a file, let another user make changes to the file, and prevent all other users from

accessing the file. You can set similar permissions on printers so that certain users can configure the printer and other users can only print.

When you need to change the permissions on a file, you can run Windows Explorer, right-click the file name, and click **Properties**. On the **Security** tab, you can change permissions on the file. For more information, see [Managing Permissions](#).

Note Another kind of permissions, called share permissions, is set on the Sharing tab of a folder's **Properties** page or by using the Shared Folder Wizard. For more information see [Share and NTFS Permissions on a File Server](#).

Ownership of objects

An owner is assigned to an object when that object is created. By default, the owner is the creator of the object. No matter what permissions are set on an object, the owner of the object can always change the permissions. For more information, see [Manage Object Ownership](#).

Inheritance of permissions

Inheritance allows administrators to easily assign and manage permissions. This feature automatically causes objects within a container to inherit all the inheritable permissions of that container. For example, the files within a folder inherit the permissions of the folder. Only permissions marked to be inherited will be inherited.

User rights

User rights grant specific privileges and sign-in rights to users and groups in your computing environment. Administrators can assign specific rights to group accounts or to individual user accounts. These rights authorize users to perform specific actions, such as signing in to a system interactively or backing up files and directories.

User rights are different from permissions because user rights apply to user accounts, and permissions are associated with objects. Although user rights can apply to individual user accounts, user rights are best administered on a group account basis. There is no support in the access control user interface to grant user rights. However, user rights assignment can be administered through **Local Security Settings**.

For more information about user rights, see [User Rights Assignment](#).

Object auditing

With administrator's rights, you can audit users' successful or failed access to objects. You can select which object access to audit by using the access control user interface, but first you must enable the audit policy by selecting **Audit object access** under **Local Policies** in **Local Security Settings**. You can then view these security-related events in the Security log in Event Viewer.

For more information about auditing, see [Security Auditing Overview](#).

See also

- For more information about access control and authorization, see [Access Control and Authorization Overview](#).

Dynamic Access Control Overview

7/1/2022 • 8 minutes to read • [Edit Online](#)

Applies to

- Windows Server 2016

This overview topic for the IT professional describes Dynamic Access Control and its associated elements, which were introduced in Windows Server 2012 and Windows 8.

Domain-based Dynamic Access Control enables administrators to apply access-control permissions and restrictions based on well-defined rules that can include the sensitivity of the resources, the job or role of the user, and the configuration of the device that is used to access these resources.

For example, a user might have different permissions when they access a resource from their office computer versus when they are using a portable computer over a virtual private network. Or access may be allowed only if a device meets the security requirements that are defined by the network administrators. When Dynamic Access Control is used, a user's permissions change dynamically without additional administrator intervention if the user's job or role changes (resulting in changes to the user's account attributes in AD DS). For more detailed examples of Dynamic Access Control in use, see the scenarios described in [Dynamic Access Control: Scenario Overview](#).

Dynamic Access Control is not supported in Windows operating systems prior to Windows Server 2012 and Windows 8. When Dynamic Access Control is configured in environments with supported and non-supported versions of Windows, only the supported versions will implement the changes.

Features and concepts associated with Dynamic Access Control include:

- [Central access rules](#)
- [Central access policies](#)
- [Claims](#)
- [Expressions](#)
- [Proposed permissions](#)

Central access rules

A central access rule is an expression of authorization rules that can include one or more conditions involving user groups, user claims, device claims, and resource properties. Multiple central access rules can be combined into a central access policy.

If one or more central access rules have been defined for a domain, file share administrators can match specific rules to specific resources and business requirements.

Central access policies

Central access policies are authorization policies that include conditional expressions. For example, let's say an organization has a business requirement to restrict access to personally identifiable information (PII) in files to only the file owner and members of the human resources (HR) department who are allowed to view PII information. This represents an organization-wide policy that applies to PII files wherever they are located on file servers across the organization. To implement this policy, an organization needs to be able to:

- Identify and mark the files that contain the PII.

- Identify the group of HR members who are allowed to view the PII information.
- Add the central access policy to a central access rule, and apply the central access rule to all files that contain the PII, wherever they are located amongst the file servers across the organization.

Central access policies act as security umbrellas that an organization applies across its servers. These policies are in addition to (but do not replace) the local access policies or discretionary access control lists (DACLS) that are applied to files and folders.

Claims

A claim is a unique piece of information about a user, device, or resource that has been published by a domain controller. The user's title, the department classification of a file, or the health state of a computer are valid examples of a claim. An entity can involve more than one claim, and any combination of claims can be used to authorize access to resources. The following types of claims are available in the supported versions of Windows:

- **User claims** Active Directory attributes that are associated with a specific user.
- **Device claims** Active Directory attributes that are associated with a specific computer object.
- **Resource attributes** Global resource properties that are marked for use in authorization decisions and published in Active Directory.

Claims make it possible for administrators to make precise organization- or enterprise-wide statements about users, devices, and resources that can be incorporated in expressions, rules, and policies.

Expressions

Conditional expressions are an enhancement to access control management that allow or deny access to resources only when certain conditions are met, for example, group membership, location, or the security state of the device. Expressions are managed through the Advanced Security Settings dialog box of the ACL Editor or the Central Access Rule Editor in the Active Directory Administrative Center (ADAC).

Expressions help administrators manage access to sensitive resources with flexible conditions in increasingly complex business environments.

Proposed permissions

Proposed permissions enable an administrator to more accurately model the impact of potential changes to access control settings without actually changing them.

Predicting the effective access to a resource helps you plan and configure permissions for those resources before implementing those changes.

Additional changes

Additional enhancements in the supported versions of Windows that support Dynamic Access Control include:

Support in the Kerberos authentication protocol to reliably provide user claims, device claims, and device groups.

By default, devices running any of the supported versions of Windows are able to process Dynamic Access Control-related Kerberos tickets, which include data needed for compound authentication. Domain controllers are able to issue and respond to Kerberos tickets with compound authentication-related information. When a domain is configured to recognize Dynamic Access Control, devices receive claims from domain controllers during initial authentication, and they receive compound authentication tickets when submitting service ticket requests. Compound authentication results in an access token that includes the identity of the user and the device on the resources that recognize Dynamic Access Control.

Support for using the Key Distribution Center (KDC) Group Policy setting to enable Dynamic Access Control for a domain.

Every domain controller needs to have the same Administrative Template policy setting, which is located at **Computer Configuration\Policies\Administrative Templates\System\KDC\Support Dynamic Access Control and Kerberos armoring**.

Support in Active Directory to store user and device claims, resource properties, and central access policy objects.

Support for using Group Policy to deploy central access policy objects.

The following Group Policy setting enables you to deploy central access policy objects to file servers in your organization: **Computer Configuration\Policies\ Windows Settings\Security Settings\File System\Central Access Policy**.

Support for claims-based file authorization and auditing for file systems by using Group Policy and Global Object Access Auditing

You must enable staged central access policy auditing to audit the effective access of central access policy by using proposed permissions. You configure this setting for the computer under **Advanced Audit Policy Configuration** in the **Security Settings** of a Group Policy Object (GPO). After you configure the security setting in the GPO, you can deploy the GPO to computers in your network.

Support for transforming or filtering claim policy objects that traverse Active Directory forest trusts

You can filter or transform incoming and outgoing claims that traverse a forest trust. There are three basic scenarios for filtering and transforming claims:

- **Value-based filtering** Filters can be based on the value of a claim. This allows the trusted forest to prevent claims with certain values from being sent to the trusting forest. Domain controllers in trusting forests can use value-based filtering to guard against an elevation-of-privilege attack by filtering the incoming claims with specific values from the trusted forest.
- **Claim type-based filtering** Filters are based on the type of claim, rather than the value of the claim. You identify the claim type by the name of the claim. You use claim type-based filtering in the trusted forest, and it prevents Windows from sending claims that disclose information to the trusting forest.
- **Claim type-based transformation** Manipulates a claim before sending it to the intended target. You use claim type-based transformation in the trusted forest to generalize a known claim that contains specific information. You can use transformations to generalize the claim-type, the claim value, or both.

Software requirements

Because claims and compound authentication for Dynamic Access Control require Kerberos authentication extensions, any domain that supports Dynamic Access Control must have enough domain controllers running the supported versions of Windows to support authentication from Dynamic Access Control-aware Kerberos clients. By default, devices must use domain controllers in other sites. If no such domain controllers are available, authentication will fail. Therefore, you must support one of the following conditions:

- Every domain that supports Dynamic Access Control must have enough domain controllers running the supported versions of Windows Server to support authentication from all devices running the supported versions of Windows or Windows Server.
- Devices running the supported versions of Windows or that do not protect resources by using claims or compound identity, should disable Kerberos protocol support for Dynamic Access Control.

For domains that support user claims, every domain controller running the supported versions of Windows server must be configured with the appropriate setting to support claims and compound authentication, and to provide Kerberos armoring. Configure settings in the KDC Administrative Template policy as follows:

- **Always provide claims** Use this setting if all domain controllers are running the supported versions of Windows Server. In addition, set the domain functional level to Windows Server 2012 or higher.

- **Supported** When you use this setting, monitor domain controllers to ensure that the number of domain controllers running the supported versions of Windows Server is sufficient for the number of client computers that need to access resources protected by Dynamic Access Control.

If the user domain and file server domain are in different forests, all domain controllers in the file server's forest root must be set at the Windows Server 2012 or higher functional level.

If clients do not recognize Dynamic Access Control, there must be a two-way trust relationship between the two forests.

If claims are transformed when they leave a forest, all domain controllers in the user's forest root must be set at the Windows Server 2012 or higher functional level.

A file server running a server operating system that supports Dynamic Access Control must have a Group Policy setting that specifies whether it needs to get user claims for user tokens that do not carry claims. This setting is set by default to **Automatic**, which results in this Group Policy setting to be turned **On** if there is a central policy that contains user or device claims for that file server. If the file server contains discretionary ACLs that include user claims, you need to set this Group Policy to **On** so that the server knows to request claims on behalf of users that do not provide claims when they access the server.

See also

- [Access control overview](#)

Security identifiers

7/1/2022 • 31 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016
- Windows Server 2019

This topic for the IT professional describes security identifiers and how they work in regards to accounts and groups in the Windows operating system.

What are security identifiers?

A security identifier (SID) is used to uniquely identify a security principal or security group. Security principals can represent any entity that can be authenticated by the operating system, such as a user account, a computer account, or a thread or process that runs in the security context of a user or computer account.

Each account or group, or process running in the security context of the account, has a unique SID that is issued by an authority, such as a Windows domain controller. It is stored in a security database. The system generates the SID that identifies a particular account or group at the time the account or group is created. When a SID has been used as the unique identifier for a user or group, it can never be used again to identify another user or group.

Each time a user signs in, the system creates an access token for that user. The access token contains the user's SID, user rights, and the SIDs for any groups the user belongs to. This token provides the security context for whatever actions the user performs on that computer.

In addition to the uniquely created, domain-specific SIDs that are assigned to specific users and groups, there are well-known SIDs that identify generic groups and generic users. For example, the Everyone and World SIDs identify a group that includes all users. Well-known SIDs have values that remain constant across all operating systems.

SIDs are a fundamental building block of the Windows security model. They work with specific components of the authorization and access control technologies in the security infrastructure of the Windows Server operating systems. This helps protect access to network resources and provides a more secure computing environment.

The content in this topic applies to computers that are running the supported versions of the Windows operating system as designated in the **Applies To** list at the beginning of this topic.

How security identifiers work

Users refer to accounts by using the account name, but the operating system internally refers to accounts and processes that run in the security context of the account by using their security identifiers (SIDs). For domain accounts, the SID of a security principal is created by concatenating the SID of the domain with a relative identifier (RID) for the account. SIDs are unique within their scope (domain or local), and they are never reused.

The operating system generates a SID that identifies a particular account or group at the time the account or group is created. The SID for a local account or group is generated by the Local Security Authority (LSA) on the computer, and it is stored with other account information in a secure area of the registry. The SID for a domain account or group is generated by the domain security authority, and it is stored as an attribute of the User or

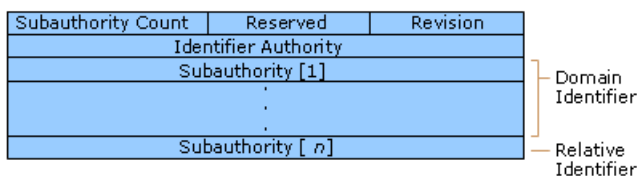
Group object in Active Directory Domain Services.

For every local account and group, the SID is unique for the computer where it was created. No two accounts or groups on the computer ever share the same SID. Likewise, for every domain account and group, the SID is unique within an enterprise. This means that the SID for an account or group that is created in one domain will never match the SID for an account or group created in any other domain in the enterprise.

SIDs always remain unique. Security authorities never issue the same SID twice, and they never reuse SIDs for deleted accounts. For example, if a user with a user account in a Windows domain leaves her job, an administrator deletes her Active Directory account, including the SID that identifies the account. If she later returns to a different job at the same company, an administrator creates a new account, and the Windows Server operating system generates a new SID. The new SID does not match the old one; so none of the user's access from her old account is transferred to the new account. Her two accounts represent two completely different security principals.

Security identifier architecture

A security identifier is a data structure in binary format that contains a variable number of values. The first values in the structure contain information about the SID structure. The remaining values are arranged in a hierarchy (similar to a telephone number), and they identify the SID-issuing authority (for example, "NT Authority"), the SID-issuing domain, and a particular security principal or group. The following image illustrates the structure of a SID.



The individual values of a SID are described in the following table.

COMMENT	DESCRIPTION
Revision	Indicates the version of the SID structure that is used in a particular SID.
Identifier authority	Identifies the highest level of authority that can issue SIDs for a particular type of security principal. For example, the identifier authority value in the SID for the Everyone group is 1 (World Authority). The identifier authority value in the SID for a specific Windows Server account or group is 5 (NT Authority).
Subauthorities	> Holds the most important information in a SID, which is contained in a series of one or more subauthority values. All values up to, but not including, the last value in the series collectively identify a domain in an enterprise. This part of the series is called the domain identifier. The last value in the series, which is called the relative identifier (RID), identifies a particular account or group relative to a domain.

The components of a SID are easier to visualize when SIDs are converted from a binary to a string format by using standard notation:

S-R-X-Y1-Y2-Yn-1-Yn

In this notation, the components of a SID are represented as shown in the following table.

COMMENT	DESCRIPTION
S	Indicates that the string is a SID
R	Indicates the revision level
X	Indicates the identifier authority value
Y	Represents a series of subauthority values, where <i>n</i> is the number of values

The SID's most important information is contained in the series of subauthority values. The first part of the series (-Y1-Y2-Y_{*n*}-1) is the domain identifier. This element of the SID becomes significant in an enterprise with several domains, because the domain identifier differentiates SIDs that are issued by one domain from SIDs that are issued by all other domains in the enterprise. No two domains in an enterprise share the same domain identifier.

The last item in the series of subauthority values (-Y_{*n*}) is the relative identifier. It distinguishes one account or group from all other accounts and groups in the domain. No two accounts or groups in any domain share the same relative identifier.

For example, the SID for the built-in Administrators group is represented in standardized SID notation as the following string:

```
S-1-5-32-544
```

This SID has four components:

- A revision level (1)
- An identifier authority value (5, NT Authority)
- A domain identifier (32, Builtin)
- A relative identifier (544, Administrators)

SIDs for built-in accounts and groups always have the same domain identifier value: 32. This value identifies the domain **Builtin**, which exists on every computer that is running a version of the Windows Server operating system. It is never necessary to distinguish one computer's built-in accounts and groups from another computer's built-in accounts and groups because they are local in scope. They are local to a single computer, or in the case of domain controllers for a network domain, they are local to several computers that are acting as one.

Built-in accounts and groups need to be distinguished from one another within the scope of the **Builtin** domain. Therefore, the SID for each account and group has a unique relative identifier. A relative identifier value of 544 is unique to the built-in Administrators group. No other account or group in the **Builtin** domain has a SID with a final value of 544.

In another example, consider the SID for the global group, Domain Admins. Every domain in an enterprise has a Domain Admins group, and the SID for each group is different. The following example represents the SID for the Domain Admins group in the Contoso, Ltd. domain (Contoso\Domain Admins):

```
S-1-5-21-1004336348-1177238915-682003330-512
```

The SID for Contoso\Domain Admins has:

- A revision level (1)
- An identifier authority (5, NT Authority)
- A domain identifier (21-1004336348-1177238915-682003330, Contoso)
- A relative identifier (512, Domain Admins)

The SID for Contoso\Domain Admins is distinguished from the SIDs for other Domain Admins groups in the same enterprise by its domain identifier: 21-1004336348-1177238915-682003330. No other domain in the enterprise uses this value as its domain identifier. The SID for Contoso\Domain Admins is distinguished from the SIDs for other accounts and groups that are created in the Contoso domain by its relative identifier, 512. No other account or group in the domain has a SID with a final value of 512.

Relative identifier allocation

When accounts and groups are stored in an account database that is managed by a local Security Accounts Manager (SAM), it is fairly easy for the system to generate a unique relative identifier for each account and in a group that it creates on a stand-alone computer. The SAM on a stand-alone computer can track the relative identifier values that it has used before and make sure that it never uses them again.

In a network domain, however, generating unique relative identifiers is a more complex process. Windows Server network domains can have several domain controllers. Each domain controller stores Active Directory account information. This means that, in a network domain, there are as many copies of the account database as there are domain controllers. In addition to this, every copy of the account database is a master copy. New accounts and groups can be created on any domain controller. Changes that are made to Active Directory on one domain controller are replicated to all other domain controllers in the domain. The process of replicating changes in one master copy of the account database to all other master copies is called a multimaster operation.

The process of generating unique relative identifiers is a single-master operation. One domain controller is assigned the role of relative identifier (RID) master, and it allocates a sequence of relative identifiers to each domain controller in the domain. When a new domain account or group is created in one domain controller's replica of Active Directory, it is assigned a SID. The relative identifier for the new SID is taken from the domain controller's allocation of relative identifiers. When its supply of relative identifiers begins to run low, the domain controller requests another block from the RID master.

Each domain controller uses each value in a block of relative identifiers only once. The RID master allocates each block of relative identifier values only once. This process assures that every account and group created in the domain has a unique relative identifier.

Security identifiers and globally unique identifiers

When a new domain user or group account is created, Active Directory stores the account's SID in the **ObjectSID** property of a User or Group object. It also assigns the new object a globally unique identifier (GUID), which is a 128-bit value that is unique not only in the enterprise, but also across the world. GUIDs are assigned to every object that is created by Active Directory, not only User and Group objects. Each object's GUID is stored in its **ObjectGUID** property.

Active Directory uses GUIDs internally to identify objects. For example, the GUID is one of an object's properties that is published in the global catalog. Searching the global catalog for a User object GUID produces results if the user has an account somewhere in the enterprise. In fact, searching for any object by **ObjectGUID** might be the most reliable way of finding the object you want to locate. The values of other object properties can change, but the **ObjectGUID** property never changes. When an object is assigned a GUID, it keeps that value for life.

If a user moves from one domain to another, the user gets a new SID. The SID for a group object does not change because groups stay in the domain where they were created. However, if people move, their accounts

can move with them. If an employee moves from North America to Europe, but stays in the same company, an administrator for the enterprise can move the employee's User object from, for example, Contoso\NoAm to Contoso\Europe. If the administrator does this, the User object for the account needs a new SID. The domain identifier portion of a SID that is issued in NoAm is unique to NoAm; so the SID for the user's account in Europe has a different domain identifier. The relative identifier portion of a SID is unique relative to the domain; so if the domain changes, the relative identifier also changes.

When a User object moves from one domain to another, a new SID must be generated for the user account and stored in the **ObjectSID** property. Before the new value is written to the property, the previous value is copied to another property of a User object, **SIDHistory**. This property can hold multiple values. Each time a User object moves to another domain, a new SID is generated and stored in the **ObjectSID** property, and another value is added to the list of old SIDs in **SIDHistory**. When a user signs in and is successfully authenticated, the domain authentication service queries Active Directory for all the SIDs that are associated with the user, including the user's current SID, the user's old SIDs, and the SIDs for the user's groups. All these SIDs are returned to the authentication client, and they are included in the user's access token. When the user tries to gain access to a resource, any one of the SIDs in the access token (including one of the SIDs in **SIDHistory**), can allow or deny the user access.

If you allow or deny users' access to a resource based on their jobs, you should allow or deny access to a group, not to an individual. That way, when users change jobs or move to other departments, you can easily adjust their access by removing them from certain groups and adding them to others.

However, if you allow or deny an individual user access to resources, you probably want that user's access to remain the same no matter how many times the user's account domain changes. The **SIDHistory** property makes this possible. When a user changes domains, there is no need to change the access control list (ACL) on any resource. If an ACL has the user's old SID, but not the new one, the old SID is still in the user's access token. It is listed among the SIDs for the user's groups, and the user is granted or denied access based on the old SID.

Well-known SIDs

The values of certain SIDs are constant across all systems. They are created when the operating system or domain is installed. They are called well-known SIDs because they identify generic users or generic groups.

There are universal well-known SIDs that are meaningful on all secure systems that use this security model, including operating systems other than Windows. In addition, there are well-known SIDs that are meaningful only on Windows operating systems.

The following table lists the universal well-known SIDs.

VALUE	UNIVERSAL WELL-KNOWN SID	IDENTIFIES
S-1-0-0	Null SID	A group with no members. This is often used when a SID value is not known.
S-1-1-0	World	A group that includes all users.
S-1-2-0	Local	Users who log on to terminals that are locally (physically) connected to the system.
S-1-2-1	Console Logon	A group that includes users who are logged on to the physical console.

VALUE	UNIVERSAL WELL-KNOWN SID	IDENTIFIES
S-1-3-0	Creator Owner ID	A security identifier to be replaced by the security identifier of the user who created a new object. This SID is used in inheritable ACEs.
S-1-3-1	Creator Group ID	A security identifier to be replaced by the primary-group SID of the user who created a new object. Use this SID in inheritable ACEs.
S-1-3-2	Creator Owner Server	
S-1-3-3	Creator Group Server	
S-1-3-4	Owner Rights	A group that represents the current owner of the object. When an ACE that carries this SID is applied to an object, the system ignores the implicit READ_CONTROL and WRITE_DAC permissions for the object owner.
S-1-4	Non-unique Authority	A SID that represents an identifier authority.
S-1-5	NT Authority	A SID that represents an identifier authority.
S-1-5-80-0	All Services	A group that includes all service processes configured on the system. Membership is controlled by the operating system.

The following table lists the predefined identifier authority constants. The first four values are used with universal well-known SIDs, and the rest of the values are used with well-known SIDs in Windows operating systems designated in the **Applies To** list.

IDENTIFIER AUTHORITY	VALUE	SID STRING PREFIX
SECURITY_NULL_SID_AUTHORITY	0	S-1-0
SECURITY_WORLD_SID_AUTHORITY	1	S-1-1
SECURITY_LOCAL_SID_AUTHORITY	2	S-1-2
SECURITY_CREATOR_SID_AUTHORITY	3	S-1-3
SECURITY_NT_AUTHORITY	5	S-1-5
SECURITY_AUTHENTICATION_AUTHORITY	18	S-1-18

The following RID values are used with universal well-known SIDs. The Identifier authority column shows the prefix of the identifier authority with which you can combine the RID to create a universal well-known SID.

RELATIVE IDENTIFIER AUTHORITY	VALUE	IDENTIFIER AUTHORITY
SECURITY_NULL_RID	0	S-1-0
SECURITY_WORLD_RID	0	S-1-1
SECURITY_LOCAL_RID	0	S-1-2
SECURITY_CREATOR_OWNER_RID	0	S-1-3
SECURITY_CREATOR_GROUP_RID	1	S-1-3

The SECURITY_NT_AUTHORITY (S-1-5) predefined identifier authority produces SIDs that are not universal and are meaningful only in installations of the Windows operating systems that are designated in the **Applies To** list at the beginning of this topic. The following table lists the well-known SIDs.

SID	DISPLAY NAME	DESCRIPTION
S-1-5-1	Dialup	A group that includes all users who are logged on to the system by means of a dial-up connection.
S-1-5-113	Local account	You can use this SID when restricting network logon to local accounts instead of "administrator" or equivalent. This SID can be effective in blocking network logon for local users and groups by account type regardless of what they are actually named.
S-1-5-114	Local account and member of Administrators group	You can use this SID when restricting network logon to local accounts instead of "administrator" or equivalent. This SID can be effective in blocking network logon for local users and groups by account type regardless of what they are actually named.
S-1-5-2	Network	A group that includes all users who are logged on by means of a network connection. Access tokens for interactive users do not contain the Network SID.
S-1-5-3	Batch	A group that includes all users who have logged on by means of a batch queue facility, such as task scheduler jobs.

SID	DISPLAY NAME	DESCRIPTION
S-1-5-4	Interactive	A group that includes all users who log on interactively. A user can start an interactive logon session by logging on directly at the keyboard, by opening a Remote Desktop Services connection from a remote computer, or by using a remote shell such as Telnet. In each case, the user's access token contains the Interactive SID. If the user signs in by using a Remote Desktop Services connection, the user's access token also contains the Remote Interactive Logon SID.
S-1-5-5- X-Y	Logon Session	The X and Y values for these SIDs uniquely identify a particular logon session.
S-1-5-6	Service	A group that includes all security principals that have signed in as a service.
S-1-5-7	Anonymous Logon	A user who has connected to the computer without supplying a user name and password. The Anonymous Logon identity is different from the identity that is used by Internet Information Services (IIS) for anonymous web access. IIS uses an actual account—by default, IUSR_ <i>ComputerName</i> , for anonymous access to resources on a website. Strictly speaking, such access is not anonymous because the security principal is known even though unidentified people are using the account. IUSR_ <i>ComputerName</i> (or whatever you name the account) has a password, and IIS logs on the account when the service starts. As a result, the IIS "anonymous" user is a member of Authenticated Users but Anonymous Logon is not.
S-1-5-8	Proxy	Does not currently apply: this SID is not used.
S-1-5-9	Enterprise Domain Controllers	A group that includes all domain controllers in a forest of domains.

SID	DISPLAY NAME	DESCRIPTION
S-1-5-10	Self	A placeholder in an ACE for a user, group, or computer object in Active Directory. When you grant permissions to Self, you grant them to the security principal that is represented by the object. During an access check, the operating system replaces the SID for Self with the SID for the security principal that is represented by the object.
S-1-5-11	Authenticated Users	A group that includes all users and computers with identities that have been authenticated. Authenticated Users does not include Guest even if the Guest account has a password. This group includes authenticated security principals from any trusted domain, not only the current domain.
S-1-5-12	Restricted Code	An identity that is used by a process that is running in a restricted security context. In Windows and Windows Server operating systems, a software restriction policy can assign one of three security levels to code: unrestricted, restricted, or disallowed. When code runs at the restricted security level, the Restricted SID is added to the user's access token.
S-1-5-13	Terminal Server User	A group that includes all users who sign in to a server with Remote Desktop Services enabled.
S-1-5-14	Remote Interactive Logon	A group that includes all users who log on to the computer by using a remote desktop connection. This group is a subset of the Interactive group. Access tokens that contain the Remote Interactive Logon SID also contain the Interactive SID.
S-1-5-15	This Organization	A group that includes all users from the same organization. Only included with Active Directory accounts and only added by a domain controller.
S-1-5-17	IUSR	An account that is used by the default Internet Information Services (IIS) user.

SID	DISPLAY NAME	DESCRIPTION
S-1-5-18	System (or LocalSystem)	<p>An identity that is used locally by the operating system and by services that are configured to sign in as LocalSystem.</p> <p>System is a hidden member of Administrators. That is, any process running as System has the SID for the built-in Administrators group in its access token.</p> <p>When a process that is running locally as System accesses network resources, it does so by using the computer's domain identity. Its access token on the remote computer includes the SID for the local computer's domain account plus SIDs for security groups that the computer is a member of, such as Domain Computers and Authenticated Users.</p>
S-1-5-19	NT Authority (LocalService)	<p>An identity that is used by services that are local to the computer, have no need for extensive local access, and do not need authenticated network access. Services that run as LocalService access local resources as ordinary users, and they access network resources as anonymous users. As a result, a service that runs as LocalService has significantly less authority than a service that runs as LocalSystem locally and on the network.</p>
S-1-5-20	Network Service	<p>An identity that is used by services that have no need for extensive local access but do need authenticated network access. Services running as NetworkService access local resources as ordinary users and access network resources by using the computer's identity. As a result, a service that runs as NetworkService has the same network access as a service that runs as LocalSystem, but it has significantly reduced local access.</p>

SID	DISPLAY NAME	DESCRIPTION
S-1-5- <i>domain</i> -500	Administrator	<p>A user account for the system administrator. Every computer has a local Administrator account and every domain has a domain Administrator account.</p> <p>The Administrator account is the first account created during operating system installation. The account cannot be deleted, disabled, or locked out, but it can be renamed.</p> <p>By default, the Administrator account is a member of the Administrators group, and it cannot be removed from that group.</p>
S-1-5- <i>domain</i> -501	Guest	<p>A user account for people who do not have individual accounts. Every computer has a local Guest account, and every domain has a domain Guest account.</p> <p>By default, Guest is a member of the Everyone and the Guests groups. The domain Guest account is also a member of the Domain Guests and Domain Users groups.</p> <p>Unlike Anonymous Logon, Guest is a real account, and it can be used to log on interactively. The Guest account does not require a password, but it can have one.</p>
S-1-5- <i>domain</i> -502	krbtgt	<p>A user account that is used by the Key Distribution Center (KDC) service. The account exists only on domain controllers.</p>
S-1-5- <i>domain</i> -512	Domain Admins	<p>A global group with members that are authorized to administer the domain. By default, the Domain Admins group is a member of the Administrators group on all computers that have joined the domain, including domain controllers.</p> <p>Domain Admins is the default owner of any object that is created in the domain's Active Directory by any member of the group. If members of the group create other objects, such as files, the default owner is the Administrators group.</p>
S-1-5- <i>domain</i> -513	Domain Users	<p>A global group that includes all users in a domain. When you create a new User object in Active Directory, the user is automatically added to this group.</p>

SID	DISPLAY NAME	DESCRIPTION
S-1-5- <i>domain</i> -514	Domain Guests	A global group, which by default, has only one member: the domain's built-in Guest account.
S-1-5- <i>domain</i> -515	Domain Computers	A global group that includes all computers that have joined the domain, excluding domain controllers.
S-1-5- <i>domain</i> -516	Domain Controllers	A global group that includes all domain controllers in the domain. New domain controllers are added to this group automatically.
S-1-5- <i>domain</i> -517	Cert Publishers	A global group that includes all computers that host an enterprise certification authority. Cert Publishers are authorized to publish certificates for User objects in Active Directory.
S-1-5- <i>root domain</i> -518	Schema Admins	A group that exists only in the forest root domain. It is a universal group if the domain is in native mode, and it is a global group if the domain is in mixed mode. The Schema Admins group is authorized to make schema changes in Active Directory. By default, the only member of the group is the Administrator account for the forest root domain.
S-1-5- <i>root domain</i> -519	Enterprise Admins	A group that exists only in the forest root domain. It is a universal group if the domain is in native mode, and it is a global group if the domain is in mixed mode. The Enterprise Admins group is authorized to make changes to the forest infrastructure, such as adding child domains, configuring sites, authorizing DHCP servers, and installing enterprise certification authorities. By default, the only member of Enterprise Admins is the Administrator account for the forest root domain. The group is a default member of every Domain Admins group in the forest.

SID	DISPLAY NAME	DESCRIPTION
S-1-5-domain-520	Group Policy Creator Owners	A global group that is authorized to create new Group Policy Objects in Active Directory. By default, the only member of the group is Administrator. Objects that are created by members of Group Policy Creator Owners are owned by the individual user who creates them. In this way, the Group Policy Creator Owners group is unlike other administrative groups (such as Administrators and Domain Admins). Objects that are created by members of these groups are owned by the group rather than by the individual.
S-1-5-domain-553	RAS and IAS Servers	A local domain group. By default, this group has no members. Computers that are running the Routing and Remote Access service are added to the group automatically. Members of this group have access to certain properties of User objects, such as Read Account Restrictions, Read Logon Information, and Read Remote Access Information.
S-1-5-32-544	Administrators	A built-in group. After the initial installation of the operating system, the only member of the group is the Administrator account. When a computer joins a domain, the Domain Admins group is added to the Administrators group. When a server becomes a domain controller, the Enterprise Admins group also is added to the Administrators group.
S-1-5-32-545	Users	A built-in group. After the initial installation of the operating system, the only member is the Authenticated Users group.
S-1-5-32-546	Guests	A built-in group. By default, the only member is the Guest account. The Guests group allows occasional or one-time users to log on with limited privileges to a computer's built-in Guest account.
S-1-5-32-547	Power Users	A built-in group. By default, the group has no members. Power users can create local users and groups; modify and delete accounts that they have created; and remove users from the Power Users, Users, and Guests groups. Power users also can install programs; create, manage, and delete local printers; and create and delete file shares.

SID	DISPLAY NAME	DESCRIPTION
S-1-5-32-548	Account Operators	A built-in group that exists only on domain controllers. By default, the group has no members. By default, Account Operators have permission to create, modify, and delete accounts for users, groups, and computers in all containers and organizational units of Active Directory except the Builtin container and the Domain Controllers OU. Account Operators do not have permission to modify the Administrators and Domain Admins groups, nor do they have permission to modify the accounts for members of those groups.
S-1-5-32-549	Server Operators	Description: A built-in group that exists only on domain controllers. By default, the group has no members. Server Operators can log on to a server interactively; create and delete network shares; start and stop services; back up and restore files; format the hard disk of the computer; and shut down the computer.
S-1-5-32-550	Print Operators	A built-in group that exists only on domain controllers. By default, the only member is the Domain Users group. Print Operators can manage printers and document queues.
S-1-5-32-551	Backup Operators	A built-in group. By default, the group has no members. Backup Operators can back up and restore all files on a computer, regardless of the permissions that protect those files. Backup Operators also can log on to the computer and shut it down.
S-1-5-32-552	Replicators	A built-in group that is used by the File Replication service on domain controllers. By default, the group has no members. Do not add users to this group.
S-1-5-32-554	Builtin\Pre-Windows 2000 Compatible Access	An alias added by Windows 2000. A backward compatibility group that allows read access on all users and groups in the domain.
S-1-5-32-555	Builtin\Remote Desktop Users	An alias. Members in this group are granted the right to log on remotely.
S-1-5-32-556	Builtin\Network Configuration Operators	An alias. Members in this group can have some administrative privileges to manage configuration of networking features.

SID	DISPLAY NAME	DESCRIPTION
S-1-5-32-557	Builtin\Incoming Forest Trust Builders	An alias. Members of this group can create incoming, one-way trusts to this forest.
S-1-5-32-558	Builtin\Performance Monitor Users	An alias. Members of this group have remote access to monitor this computer.
S-1-5-32-559	Builtin\Performance Log Users	An alias. Members of this group have remote access to schedule logging of performance counters on this computer.
S-1-5-32-560	Builtin\Windows Authorization Access Group	An alias. Members of this group have access to the computed tokenGroupsGlobalAndUniversal attribute on User objects.
S-1-5-32-561	Builtin\Terminal Server License Servers	An alias. A group for Terminal Server License Servers. When Windows Server 2003 Service Pack 1 is installed, a new local group is created.
S-1-5-32-562	Builtin\Distributed COM Users	An alias. A group for COM to provide computer-wide access controls that govern access to all call, activation, or launch requests on the computer.
S-1-5-32-568	Builtin\IIS_IUSRS	An alias. A built-in group account for IIS users.
S-1-5-32-569	Builtin\Cryptographic Operators	A built-in local group. Members are authorized to perform cryptographic operations.
S-1-5-32-573	Builtin\Event Log Readers	A built-in local group. Members of this group can read event logs from local computer.
S-1-5-32-574	Builtin\Certificate Service DCOM Access	A built-in local group. Members of this group are allowed to connect to Certification Authorities in the enterprise.
S-1-5-32-575	Builtin\RDS Remote Access Servers	A built-in local group. Servers in this group enable users of RemoteApp programs and personal virtual desktops access to these resources. In Internet-facing deployments, these servers are typically deployed in an edge network. This group needs to be populated on servers running RD Connection Broker. RD Gateway servers and RD Web Access servers used in the deployment need to be in this group.

SID	DISPLAY NAME	DESCRIPTION
S-1-5-32-576	Builtin\RDS Endpoint Servers	A built-in local group. Servers in this group run virtual machines and host sessions where users RemoteApp programs and personal virtual desktops run. This group needs to be populated on servers running RD Connection Broker, RD Session Host servers and RD Virtualization Host servers used in the deployment need to be in this group.
S-1-5-32-577	Builtin\RDS Management Servers	A builtin local group. Servers in this group can perform routine administrative actions on servers running Remote Desktop Services. This group needs to be populated on all servers in a Remote Desktop Services deployment. The servers running the RDS Central Management service must be included in this group.
S-1-5-32-578	Builtin\Hyper-V Administrators	A built-in local group. Members of this group have complete and unrestricted access to all features of Hyper-V.
S-1-5-32-579	Builtin\Access Control Assistance Operators	A built-in local group. Members of this group can remotely query authorization attributes and permissions for resources on this computer.
S-1-5-32-580	Builtin\Remote Management Users	A built-in local group. Members of this group can access WMI resources over management protocols (such as WS-Management via the Windows Remote Management service). This applies only to WMI namespaces that grant access to the user.
S-1-5-64-10	NTLM Authentication	A SID that is used when the NTLM authentication package authenticated the client
S-1-5-64-14	SChannel Authentication	A SID that is used when the SChannel authentication package authenticated the client.
S-1-5-64-21	Digest Authentication	A SID that is used when the Digest authentication package authenticated the client.
S-1-5-80	NT Service	A SID that is used as an NT Service account prefix.

SID	DISPLAY NAME	DESCRIPTION
S-1-5-80-0	All Services	A group that includes all service processes that are configured on the system. Membership is controlled by the operating system. SID S-1-5-80-0 equals NT SERVICES\ALL SERVICES. This SID was introduced in Windows Server 2008 R2.
S-1-5-83-0	NT VIRTUAL MACHINE\Virtual Machines	A built-in group. The group is created when the Hyper-V role is installed. Membership in the group is maintained by the Hyper-V Management Service (VMMS). This group requires the Create Symbolic Links right (SeCreateSymbolicLinkPrivilege), and also the Log on as a Service right (SeServiceLogonRight).

The following RIDs are relative to each domain.

RID	DECIMAL VALUE	IDENTIFIES
DOMAIN_USER_RID_ADMIN	500	The administrative user account in a domain.
DOMAIN_USER_RID_GUEST	501	The guest-user account in a domain. Users who do not have an account can automatically sign in to this account.
DOMAIN_GROUP_RID_USERS	513	A group that contains all user accounts in a domain. All users are automatically added to this group.
DOMAIN_GROUP_RID_GUESTS	514	The group Guest account in a domain.
DOMAIN_GROUP_RID_COMPUTERS	515	The Domain Computer group. All computers in the domain are members of this group.
DOMAIN_GROUP_RID_CONTROLLERS	516	The Domain Controller group. All domain controllers in the domain are members of this group.
DOMAIN_GROUP_RID_CERT_ADMINS	517	The certificate publishers' group. Computers running Active Directory Certificate Services are members of this group.
DOMAIN_GROUP_RID_SCHEMA_ADMINS	518	The schema administrators' group. Members of this group can modify the Active Directory schema.

RID	DECIMAL VALUE	IDENTIFIES
DOMAIN_GROUP_RID_ENTERPRISE_ADMINS	519	The enterprise administrators' group. Members of this group have full access to all domains in the Active Directory forest. Enterprise administrators are responsible for forest-level operations such as adding or removing new domains.
DOMAIN_GROUP_RID_POLICY_ADMINS	520	The policy administrators' group.

The following table provides examples of domain-relative RIDs that are used to form well-known SIDs for local groups.

RID	DECIMAL VALUE	IDENTIFIES
DOMAIN_ALIAS_RID_ADMINS	544	Administrators of the domain.
DOMAIN_ALIAS_RID_USERS	545	All users in the domain.
DOMAIN_ALIAS_RID_GUESTS	546	Guests of the domain.
DOMAIN_ALIAS_RID_POWER_USERS	547	A user or a set of users who expect to treat a system as if it were their personal computer rather than as a workstation for multiple users.
DOMAIN_ALIAS_RID_BACKUP_OPS	551	A local group that is used to control the assignment of file backup-and-restore user rights.
DOMAIN_ALIAS_RID_REPLICATOR	552	A local group that is responsible for copying security databases from the primary domain controller to the backup domain controllers. These accounts are used only by the system.
DOMAIN_ALIAS_RID_RAS_SERVERS	553	A local group that represents remote access and servers running Internet Authentication Service (IAS). This group permits access to various attributes of User objects.

Changes in security identifier's functionality

The following table describes changes in SID implementation in the Windows operating systems that are designated in the list.

CHANGE	OPERATING SYSTEM VERSION	DESCRIPTION AND RESOURCES
--------	--------------------------	---------------------------

CHANGE	OPERATING SYSTEM VERSION	DESCRIPTION AND RESOURCES
Most of the operating system files are owned by the TrustedInstaller security identifier (SID)	Windows Server 2008, Windows Vista	The purpose of this change is to prevent a process that is running as an administrator or under the LocalSystem account from automatically replacing the operating system files.
Restricted SID checks are implemented	Windows Server 2008, Windows Vista	When restricting SIDs are present, Windows performs two access checks. The first is the normal access check, and the second is the same access check against the restricting SIDs in the token. Both access checks must pass to allow the process to access the object.

Capability SIDs

Capability Security Identifiers (SIDs) are used to uniquely and immutably identify capabilities. Capabilities represent an unforgeable token of authority that grants access to resources (Examples: documents, camera, locations etc...) to Universal Windows Applications. An App that "has" a capability is granted access to the resource the capability is associated with, and one that "does not have" a capability is denied access to the resource.

All Capability SIDs that the operating system is aware of are stored in the Windows Registry in the path 'HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\SecurityManager\CapabilityClasses\AllCachedCapabilities'. Any Capability SID added to Windows by first or third-party applications will be added to this location.

Examples of registry keys taken from Windows 10, version 1909, 64-bit Enterprise edition

You may see the following registry keys under AllCachedCapabilities:

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\SecurityManager\CapabilityClasses\AllCachedCapabilities\capabilityClass_DevUnlock

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\SecurityManager\CapabilityClasses\AllCachedCapabilities\capabilityClass_DevUnlock_Internal

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\SecurityManager\CapabilityClasses\AllCachedCapabilities\capabilityClass_Enterprise

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\SecurityManager\CapabilityClasses\AllCachedCapabilities\capabilityClass_General

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\SecurityManager\CapabilityClasses\AllCachedCapabilities\capabilityClass_Restricted

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\SecurityManager\CapabilityClasses\AllCachedCapabilities\capabilityClass_Windows

All Capability SIDs are prefixed by S-1-15-3

Examples of registry keys taken from Windows 11, version 21H2, 64-bit Enterprise edition

You may see the following registry keys under AllCachedCapabilities:

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\SecurityManager\CapabilityClasses\AllCachedCapabilities\capabilityClass_DevUnlock

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\SecurityManager\CapabilityClasses\AllCachedCapabilities\capabilityClass_DevUnlock_Internal

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\SecurityManager\CapabilityClasses\AllCachedCapabilities\capabilityClass_Enterprise

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\SecurityManager\CapabilityClasses\AllCachedCapabilities\capabilityClass_General

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\SecurityManager\CapabilityClasses\AllCachedCapabilities\capabilityClass_Restricted

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\SecurityManager\CapabilityClasses\AllCachedCapabilities\capabilityClass_Windows

All Capability SIDs are prefixed by S-1-15-3

See also

- [Access Control Overview](#)

Security Principals

7/1/2022 • 10 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows Server 2016

This reference topic for the IT professional describes security principals in regards to Windows accounts and security groups, in addition to security technologies that are related to security principals.

What are security principals?

Security principals are any entity that can be authenticated by the operating system, such as a user account, a computer account, or a thread or process that runs in the security context of a user or computer account, or the security groups for these accounts. Security principals have long been a foundation for controlling access to securable resources on Windows computers. Each security principal is represented in the operating system by a unique security identifier (SID).

The following content applies to the versions of Windows that are designated in the **Applies To** list at the beginning of this topic.

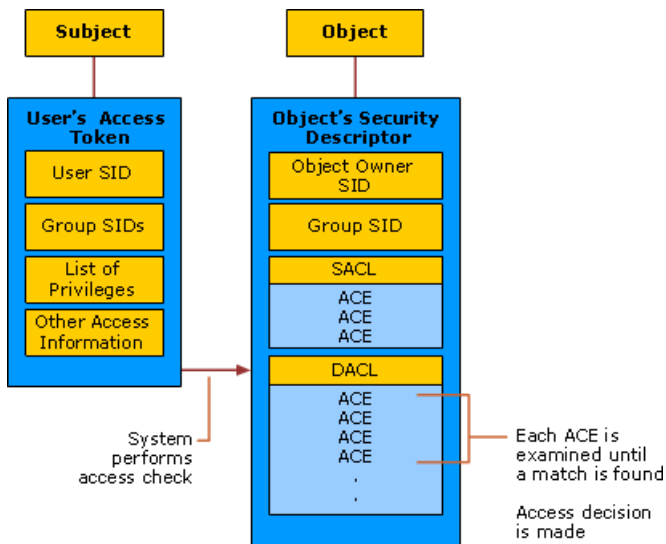
How security principals work

Security principals that are created in an Active Directory domain are Active Directory objects, which can be used to manage access to domain resources. Each security principal is assigned a unique identifier, which it retains for its entire lifetime. Local user accounts and security groups are created on a local computer, and they can be used to manage access to resources on that computer. Local user accounts and security groups are managed by the Security Accounts Manager (SAM) on the local computer.

Authorization and access control components

The following diagram illustrates the Windows authorization and access control process. In this diagram, the subject (a process that is initiated by a user) attempts to access an object, such as a shared folder. The information in the user's access token is compared to the access control entries (ACEs) in the object's security descriptor, and the access decision is made. The SIDs of security principals are used in the user's access token and in the ACEs in the object's security descriptor.

Authorization and access control process



Security principals are closely related to the following components and technologies:

- [Security identifiers](#)
- [Access tokens](#)
- [Security descriptors and access control lists](#)
- [Permissions](#)

Security identifiers

Security identifiers (SIDs) provide a fundamental building block of the Windows security model. They work with specific components of the authorization and access control technologies in the security infrastructure of the Windows Server operating systems. This helps protect access to network resources and provides a more secure computing environment.

A SID is a value of variable length that is used to uniquely identify a security principal that represents any entity that can be authenticated by the system. These entities include a user account, a computer account, or a thread or process that runs in the security context of a user or computer account. Each security principal is automatically assigned a SID when it is created. The SID is stored in a security database. When a SID is used as the unique identifier for a user or group, it can never be used to identify another user or group.

Each time a user signs in, the system creates an access token for that user. The access token contains the user's SID, user rights, and the SIDs for groups that the user belongs to. This token provides the security context for whatever actions the user performs on that computer.

In addition to the uniquely created, domain-specific SIDs that are assigned to specific users and groups, there are well-known SIDs that identify generic groups and generic users. For example, the Everyone and the World SIDs identify groups that includes all users. Well-known SIDs have values that remain constant across all operating systems.

Access tokens

An access token is a protected object that contains information about the identity and user rights that are associated with a user account.

When a user signs in interactively or tries to make a network connection to a computer running Windows, the sign-in process authenticates the user's credentials. If authentication is successful, the process returns a SID for the user and a list of SIDs for the user's security groups. The Local Security Authority (LSA) on the computer uses this information to create an access token (in this case, the primary access token). This includes the SIDs that are returned by the sign-in process and a list of user rights that are assigned by the local security policy to the user and to the user's security groups.

After the LSA creates the primary access token, a copy of the access token is attached to every thread and process that executes on the user's behalf. Whenever a thread or process interacts with a securable object or tries to perform a system task that requires user rights, the operating system checks the access token that is associated with the thread to determine the level of authorization.

There are two kinds of access tokens, primary and impersonation. Every process has a primary token that describes the security context of the user account that is associated with the process. A primary access token is typically assigned to a process to represent the default security information for that process. Impersonation tokens, on the other hand, are usually used for client and server scenarios. Impersonation tokens enable a thread to run in a security context that differs from the security context of the process that owns the thread.

Security descriptors and access control lists

A security descriptor is a data structure that is associated with each securable object. All objects in Active Directory and all securable objects on a local computer or on the network have security descriptors to help control access to the objects. Security descriptors include information about who owns an object, who can access it and in what way, and what types of access are audited. Security descriptors contain the access control list (ACL) of an object, which includes all of the security permissions that apply to that object. An object's security descriptor can contain two types of ACLs:

- A discretionary access control list (DACL), which identifies the users and groups who are allowed or denied access
- A system access control list (SACL), which controls how access is audited

You can use this access control model to individually secure objects and attributes such as files and folders, Active Directory objects, registry keys, printers, devices, ports, services, processes, and threads. Because of this individual control, you can adjust the security of objects to meet the needs of your organization, delegate authority over objects or attributes, and create custom objects or attributes that require unique security protections to be defined.

Permissions

Permissions enable the owner of each securable object, such as a file, Active Directory object, or registry key, to control who can perform an operation or a set of operations on the object or object property. Permissions are expressed in the security architecture as access control entries (ACEs). Because access to an object is at the discretion of the object's owner, the type of access control that is used in Windows is called discretionary access control.

Permissions are different from user rights in that permissions are attached to objects, and user rights apply to user accounts. Administrators can assign user rights to groups or users. These rights authorize users to perform specific actions, such as signing in to a system interactively or backing up files and directories.

On computers, user rights enable administrators to control who has the authority to perform operations that affect an entire computer, rather than a particular object. Administrators assign user rights to individual users or groups as part of the security settings for the computer. Although user rights can be managed centrally through Group Policy, they are applied locally. Users can (and usually do) have different user rights on different computers.

For information about which user rights are available and how they can be implemented, see [User Rights Assignment](#).

Security context in authentication

A user account enables a user to sign in to computers, networks, and domains with an identity that can be authenticated by the computer, network, or domain.

In Windows, any user, service, group, or computer that can initiate action is a security principal. Security principals have accounts, which can be local to a computer or domain-based. For example, domain-joined

Windows client computers can participate in a network domain by communicating with a domain controller, even when no user is signed in.

To initiate communications, the computer must have an active account in the domain. Before accepting communications from the computer, the Local Security Authority on the domain controller authenticates the computer's identity and then defines the computer's security context just as it would for a user's security principal.

This security context defines the identity and capabilities of a user or service on a particular computer, or of a user, service, group or computer on a network. For example, it defines the resources (such as a file share or printer) that can be accessed and the actions (such as Read, Write, or Modify) that can be performed by a user, service, or computer on that resource.

The security context of a user or computer can vary from one computer to another, such as when a user authenticates to a server or a workstation other than the user's primary workstation. It can also vary from one session to another, such as when an administrator modifies the user's rights and permissions. In addition, the security context is usually different when a user or computer is operating on a stand-alone basis, in a mixed network domain, or as part of an Active Directory domain.

Accounts and security groups

Accounts and security groups that are created in an Active Directory domain are stored in the Active Directory database and managed by using Active Directory tools. These security principals are directory objects, and they can be used to manage access to domain resources.

Local user accounts and security groups are created on a local computer, and they can be used to manage access to resources on that computer. Local user accounts and security groups are stored in and managed by the Security Accounts Manager (SAM) on the local computer.

User accounts

A user account uniquely identifies a person who is using a computer system. The account signals the system to enforce the appropriate authorization to allow or deny that user access to resources. User accounts can be created in Active Directory and on local computers, and administrators use them to:

- Represent, identify, and authenticate the identity of a user. A user account enables a user to sign in to computers, networks, and domains with a unique identifier that can be authenticated by the computer, network, or domain.
- Authorize (grant or deny) access to resources. After a user has been authenticated, the user is authorized access to resources based on the permissions that are assigned to that user for the resource.
- Audit the actions that are carried out on a user account.

Windows and the Windows Server operating systems have built-in user accounts, or you can create user accounts to meet the requirements of your organization.

Security groups

A security group is a collection of user accounts, computer accounts, and other groups of accounts that can be managed as a single unit from a security perspective. In Windows operating systems, there are several built-in security groups that are preconfigured with the appropriate rights and permissions for performing specific tasks. Additionally, you can (and, typically, will) create a security group for each unique combination of security requirements that applies to multiple users in your organization.

Groups can be Active Directory-based or local to a particular computer:

- Active Directory security groups are used to manage rights and permissions to domain resources.

- Local groups exist in the SAM database on local computers (on all Windows-based computers) except domain controllers. You use local groups to manage rights and permissions only to resources on the local computer.

By using security groups to manage access control, you can:

- Simplify administration. You can assign a common set of rights, a common set of permissions, or both to many accounts at one time, rather than assigning them to each account individually. Also, when users transfer jobs or leave the organization, permissions are not tied to their user accounts, making permission reassignment or removal easier.
- Implement a role-based access-control model. You can use this model to grant permissions by using groups with different scopes for appropriate purposes. Scopes that are available in Windows include local, global, domain local, and universal.
- Minimize the size of access control lists (ACLs) and speed security checking. A security group has its own SID; therefore, the group SID can be used to specify permissions for a resource. In an environment with more than a few thousand users, if the SIDs of individual user accounts are used to specify access to a resource, the ACL of that resource can become unmanageably large, and the time that is needed for the system to check permissions to the resource can become unacceptable.

For descriptions and settings information about the domain security groups that are defined in Active Directory, see [Active Directory Security Groups](#).

For descriptions and settings information about the Special Identities group, see [Special Identities](#).

See also

- [Access Control Overview](#)

Local Accounts

7/1/2022 • 20 minutes to read • [Edit Online](#)

Applies to

- Windows 11
- Windows 10
- Windows Server 2019
- Windows Server 2016

This reference topic for IT professionals describes the default local user accounts for servers, including how to manage these built-in accounts on a member or standalone server.

About local user accounts

Local user accounts are stored locally on the server. These accounts can be assigned rights and permissions on a particular server, but on that server only. Local user accounts are security principals that are used to secure and manage access to the resources on a standalone or member server for services or users.

This topic describes the following:

- [Default local user accounts](#)
 - [Administrator account](#)
 - [Guest Account](#)
 - [HelpAssistant account \(installed by using a Remote Assistance session\)](#)
 - [DefaultAccount](#)
- [Default local system accounts](#)
- [How to manage local accounts](#)
 - [Restrict and protect local accounts with administrative rights](#)
 - [Enforce local account restrictions for remote access](#)
 - [Deny network logon to all local Administrator accounts](#)
 - [Create unique passwords for local accounts with administrative rights](#)

For information about security principals, see [Security Principals](#).

Default local user accounts

The default local user accounts are built-in accounts that are created automatically when you install Windows.

After Windows is installed, the default local user accounts cannot be removed or deleted. In addition, default local user accounts do not provide access to network resources.

Default local user accounts are used to manage access to the local server's resources based on the rights and permissions that are assigned to the account. The default local user accounts, and the local user accounts that you create, are located in the Users folder. The Users folder is located in the Local Users and Groups folder in the local Computer Management Microsoft Management Console (MMC). Computer Management is a collection of

administrative tools that you can use to manage a single local or remote computer. For more information, see [How to manage local accounts](#) later in this topic.

Default local user accounts are described in the following sections.

Administrator account

The default local Administrator account is a user account for the system administrator. Every computer has an Administrator account (SID S-1-5-*domain*-500, display name Administrator). The Administrator account is the first account that is created during the Windows installation.

The Administrator account has full control of the files, directories, services, and other resources on the local computer. The Administrator account can create other local users, assign user rights, and assign permissions. The Administrator account can take control of local resources at any time simply by changing the user rights and permissions.

The default Administrator account cannot be deleted or locked out, but it can be renamed or disabled.

From Windows 10, Windows 11 and Windows Server 2016, Windows setup disables the built-in Administrator account and creates another local account that is a member of the Administrators group. Members of the Administrators groups can run apps with elevated permissions without using the **Run as Administrator** option. Fast User Switching is more secure than using Runas or different-user elevation.

Account group membership

By default, the Administrator account is installed as a member of the Administrators group on the server. It is a best practice to limit the number of users in the Administrators group because members of the Administrators group on a local server have Full Control permissions on that computer.

The Administrator account cannot be deleted or removed from the Administrators group, but it can be renamed.

Security considerations

Because the Administrator account is known to exist on many versions of the Windows operating system, it is a best practice to disable the Administrator account when possible to make it more difficult for malicious users to gain access to the server or client computer.

You can rename the Administrator account. However, a renamed Administrator account continues to use the same automatically assigned security identifier (SID), which can be discovered by malicious users. For more information about how to rename or disable a user account, see [Disable or activate a local user account](#) and [Rename a local user account](#).

As a security best practice, use your local (non-Administrator) account to sign in and then use **Run as administrator** to accomplish tasks that require a higher level of rights than a standard user account. Do not use the Administrator account to sign in to your computer unless it is entirely necessary. For more information, see [Run a program with administrative credentials](#).

In comparison, on the Windows client operating system, a user with a local user account that has Administrator rights is considered the system administrator of the client computer. The first local user account that is created during installation is placed in the local Administrators group. However, when multiple users run as local administrators, the IT staff has no control over these users or their client computers.

In this case, Group Policy can be used to enable secure settings that can control the use of the local Administrators group automatically on every server or client computer. For more information about Group Policy, see [Group Policy Overview](#).

IMPORTANT

- Blank passwords are not allowed in the versions designated in the **Applies To** list at the beginning of this topic.
- Even when the Administrator account has been disabled, it can still be used to gain access to a computer by using safe mode. In the Recovery Console or in safe mode, the Administrator account is automatically enabled. When normal operations are resumed, it is disabled.

Guest account

The Guest account is disabled by default on installation. The Guest account lets occasional or one-time users, who do not have an account on the computer, temporarily sign in to the local server or client computer with limited user rights. By default, the Guest account has a blank password. Because the Guest account can provide anonymous access, it is a security risk. For this reason, it is a best practice to leave the Guest account disabled, unless its use is entirely necessary.

Account group membership

By default, the Guest account is the only member of the default Guests group (SID S-1-5-32-546), which lets a user sign in to a server. On occasion, an administrator who is a member of the Administrators group can set up a user with a Guest account on one or more computers.

Security considerations

When enabling the Guest account, only grant limited rights and permissions. For security reasons, the Guest account should not be used over the network and made accessible to other computers.

In addition, the guest user in the Guest account should not be able to view the event logs. After the Guest account is enabled, it is a best practice to monitor the Guest account frequently to ensure that other users cannot use services and other resources, such as resources that were unintentionally left available by a previous user.

HelpAssistant account (installed with a Remote Assistance session)

The HelpAssistant account is a default local account that is enabled when a Remote Assistance session is run. This account is automatically disabled when no Remote Assistance requests are pending.

HelpAssistant is the primary account that is used to establish a Remote Assistance session. The Remote Assistance session is used to connect to another computer running the Windows operating system, and it is initiated by invitation. For solicited remote assistance, a user sends an invitation from their computer, through e-mail or as a file, to a person who can provide assistance. After the user's invitation for a Remote Assistance session is accepted, the default HelpAssistant account is automatically created to give the person who provides assistance limited access to the computer. The HelpAssistant account is managed by the Remote Desktop Help Session Manager service.

Security considerations

The SIDs that pertain to the default HelpAssistant account include:

- SID: S-1-5-<domain>-13, display name Terminal Server User. This group includes all users who sign in to a server with Remote Desktop Services enabled. Note that, in Windows Server 2008, Remote Desktop Services are called Terminal Services.
- SID: S-1-5-<domain>-14, display name Remote Interactive Logon. This group includes all users who connect to the computer by using a remote desktop connection. This group is a subset of the Interactive group. Access tokens that contain the Remote Interactive Logon SID also contain the Interactive SID.

For the Windows Server operating system, Remote Assistance is an optional component that is not installed by

default. You must install Remote Assistance before it can be used.

For details about the HelpAssistant account attributes, see the following table.

HelpAssistant account attributes

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-<domain>-13 (Terminal Server User), S-1-5-<domain>-14 (Remote Interactive Logon)
Type	User
Default container	CN=Users, DC=<domain>, DC=
Default members	None
Default member of	Domain Guests Guests
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Can be moved out, but we do not recommend it.
Safe to delegate management of this group to non-Service admins?	No

DefaultAccount

The DefaultAccount, also known as the Default System Managed Account (DSMA), is a built-in account introduced in Windows 10 version 1607 and Windows Server 2016. The DSMA is a well-known user account type. It is a user neutral account that can be used to run processes that are either multi-user aware or user-agnostic. The DSMA is disabled by default on the desktop SKUs (full windows SKUs) and WS 2016 with the Desktop.

The DSMA has a well-known RID of 503. The security identifier (SID) of the DSMA will thus have a well-known SID in the following format: S-1-5-21-<ComputerIdentifier>-503

The DSMA is a member of the well-known group **System Managed Accounts Group**, which has a well-known SID of S-1-5-32-581.

The DSMA alias can be granted access to resources during offline staging even before the account itself has been created. The account and the group are created during first boot of the machine within the Security Accounts Manager (SAM).

How Windows uses the DefaultAccount

From a permission perspective, the DefaultAccount is a standard user account. The DefaultAccount is needed to run multi-user-manifested-apps (MUMA apps). MUMA apps run all the time and react to users signing in and signing out of the devices. Unlike Windows Desktop where apps run in context of the user and get terminated when the user signs off, MUMA apps run by using the DSMA.

MUMA apps are functional in shared session SKUs such as Xbox. For example, Xbox shell is a MUMA app. Today, Xbox automatically signs in as Guest account and all apps run in this context. All the apps are multi-user-aware and respond to events fired by user manager. The apps run as the Guest account.

Similarly, Phone auto logs in as a "DefApps" account which is akin to the standard user account in Windows but with a few extra privileges. Brokers, some services and apps run as this account.

In the converged user model, the multi-user-aware apps and multi-user-aware brokers will need to run in a context different from that of the users. For this purpose, the system creates DSMA.

How the DefaultAccount gets created on domain controllers

If the domain was created with domain controllers that run Windows Server 2016, the DefaultAccount will exist on all domain controllers in the domain. If the domain was created with domain controllers that run an earlier version of Windows Server, the DefaultAccount will be created after the PDC Emulator role is transferred to a domain controller that runs Windows Server 2016. The DefaultAccount will then be replicated to all other domain controllers in the domain.

Recommendations for managing the Default Account (DSMA)

Microsoft does not recommend changing the default configuration, where the account is disabled. There is no security risk with having the account in the disabled state. Changing the default configuration could hinder future scenarios that rely on this account.

Default local system accounts

SYSTEM

The SYSTEM account is used by the operating system and by services that run under Windows. There are many services and processes in the Windows operating system that need the capability to sign in internally, such as during a Windows installation. The SYSTEM account was designed for that purpose, and Windows manages the SYSTEM account's user rights. It is an internal account that does not show up in User Manager, and it cannot be added to any groups.

On the other hand, the SYSTEM account does appear on an NTFS file system volume in File Manager in the **Permissions** portion of the **Security** menu. By default, the SYSTEM account is granted Full Control permissions to all files on an NTFS volume. Here the SYSTEM account has the same functional rights and permissions as the Administrator account.

NOTE

To grant the account Administrators group file permissions does not implicitly give permission to the SYSTEM account. The SYSTEM account's permissions can be removed from a file, but we do not recommend removing them.

NETWORK SERVICE

The NETWORK SERVICE account is a predefined local account used by the service control manager (SCM). A service that runs in the context of the NETWORK SERVICE account presents the computer's credentials to remote servers. For more information, see [NetworkService Account](#).

LOCAL SERVICE

The LOCAL SERVICE account is a predefined local account used by the service control manager. It has minimum privileges on the local computer and presents anonymous credentials on the network. For more information, see [LocalService Account](#).

How to manage local user accounts

The default local user accounts, and the local user accounts that you create, are located in the Users folder. The Users folder is located in Local Users and Groups. For more information about creating and managing local user accounts, see [Manage Local Users](#).

You can use Local Users and Groups to assign rights and permissions on the local server, and that server only, to limit the ability of local users and groups to perform certain actions. A right authorizes a user to perform certain actions on a server, such as backing up files and folders or shutting down a server. An access permission is a rule that is associated with an object, usually a file, folder, or printer. It regulates which users can have access to an

object on the server and in what manner.

You cannot use Local Users and Groups on a domain controller. However, you can use Local Users and Groups on a domain controller to target remote computers that are not domain controllers on the network.

NOTE

You use Active Directory Users and Computers to manage users and groups in Active Directory.

You can also manage local users by using NETEXE USER and manage local groups by using NETEXE LOCALGROUP, or by using a variety of PowerShell cmdlets and other scripting technologies.

Restrict and protect local accounts with administrative rights

An administrator can use a number of approaches to prevent malicious users from using stolen credentials, such as a stolen password or password hash, for a local account on one computer from being used to authenticate on another computer with administrative rights; this is also called "lateral movement".

The simplest approach is to sign in to your computer with a standard user account, instead of using the Administrator account for tasks, for example, to browse the Internet, send email, or use a word processor. When you want to perform an administrative task, for example, to install a new program or to change a setting that affects other users, you don't have to switch to an Administrator account. You can use User Account Control (UAC) to prompt you for permission or an administrator password before performing the task, as described in the next section.

The other approaches that can be used to restrict and protect user accounts with administrative rights include:

- Enforce local account restrictions for remote access.
- Deny network logon to all local Administrator accounts.
- Create unique passwords for local accounts with administrative rights.

Each of these approaches is described in the following sections.

NOTE

These approaches do not apply if all administrative local accounts are disabled.

Enforce local account restrictions for remote access

The User Account Control (UAC) is a security feature in Windows that has been in use in Windows Server 2008 and in Windows Vista, and the operating systems to which the **Applies To** list refers. UAC enables you to stay in control of your computer by informing you when a program makes a change that requires administrator-level permission. UAC works by adjusting the permission level of your user account. By default, UAC is set to notify you when applications try to make changes to your computer, but you can change how often UAC notifies you.

UAC makes it possible for an account with administrative rights to be treated as a standard user non-administrator account until full rights, also called elevation, is requested and approved. For example, UAC lets an administrator enter credentials during a non-administrator's user session to perform occasional administrative tasks without having to switch users, sign out, or use the **Run as** command.

In addition, UAC can require administrators to specifically approve applications that make system-wide changes before those applications are granted permission to run, even in the administrator's user session.

For example, a default feature of UAC is shown when a local account signs in from a remote computer by using Network logon (for example, by using NETEXE USE). In this instance, it is issued a standard user token with no administrative rights, but without the ability to request or receive elevation. Consequently, local accounts that

sign in by using Network logon cannot access administrative shares such as C\$, or ADMIN\$, or perform any remote administration.

For more information about UAC, see [User Account Control](#).

The following table shows the Group Policy and registry settings that are used to enforce local account restrictions for remote access.

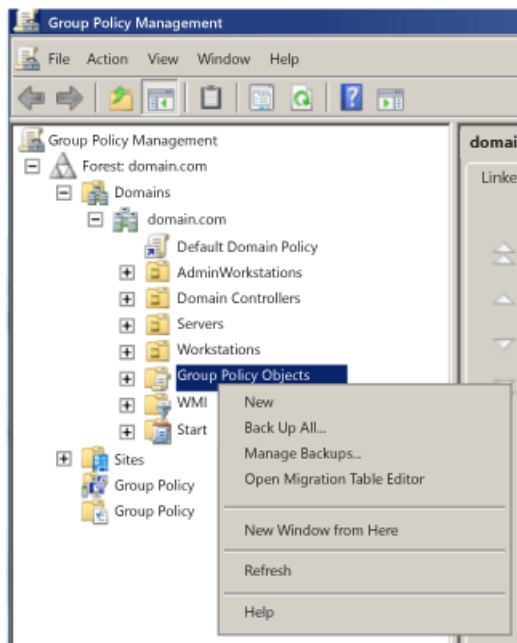
NO.	SETTING	DETAILED DESCRIPTION
	Policy location	Computer Configuration\Windows Settings\Security Settings\Local Policies\Security Options
1	Policy name	User Account Control: Run all administrators in Admin Approval Mode
	Policy setting	Enabled
2	Policy location	Computer Configuration\Windows Settings\Security Settings\Local Policies\Security Options
	Policy name	User Account Control: Run all administrators in Admin Approval Mode
	Policy setting	Enabled
3	Registry key	HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System
	Registry value name	LocalAccountTokenFilterPolicy
	Registry value type	DWORD
	Registry value data	0

NOTE

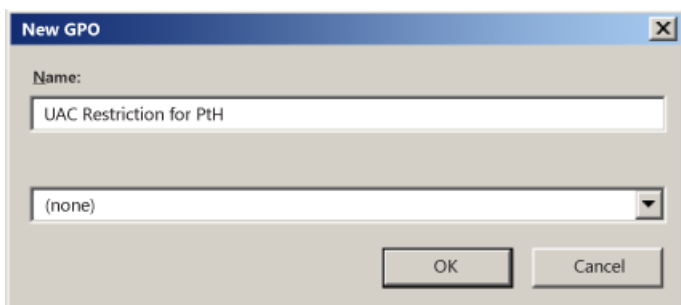
You can also enforce the default for LocalAccountTokenFilterPolicy by using the custom ADMX in Security Templates.

To enforce local account restrictions for remote access

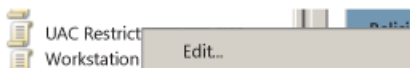
1. Start the **Group Policy Management** Console (GPMC).
2. In the console tree, expand *<Forest>\Domains\<Domain>*, and then **Group Policy Objects** where *forest* is the name of the forest, and *domain* is the name of the domain where you want to set the Group Policy Object (GPO).
3. In the console tree, right-click **Group Policy Objects**, and **> New**.



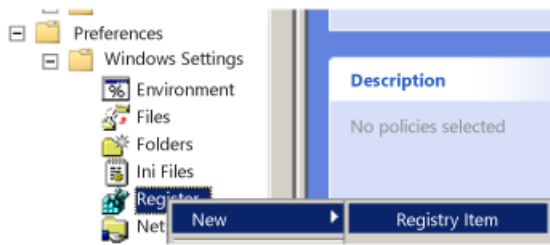
4. In the **New GPO** dialog box, type `<gpo_name>`, and **> OK** where *gpo_name* is the name of the new GPO. The GPO name indicates that the GPO is used to restrict local administrator rights from being carried over to another computer.



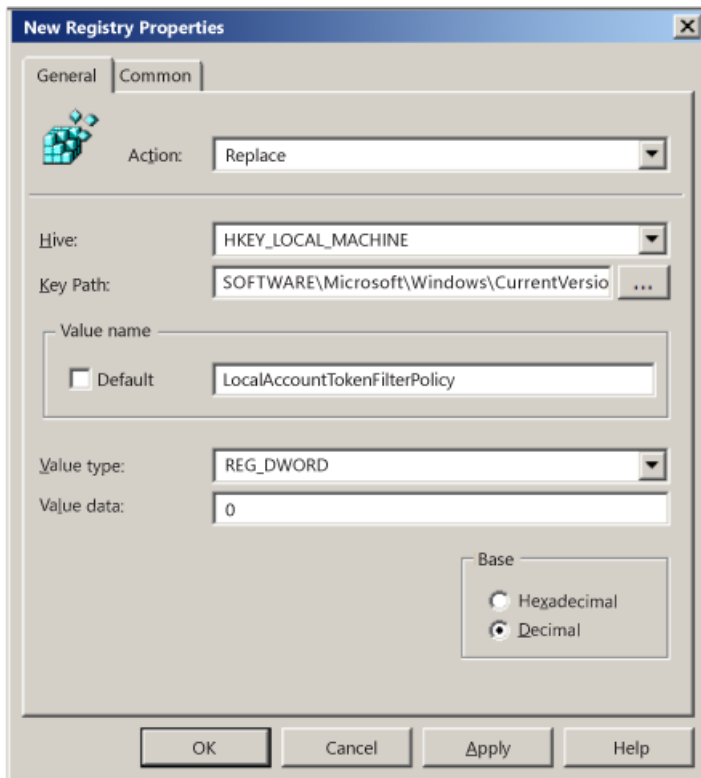
5. In the details pane, right-click `<gpo_name>`, and **> Edit**.



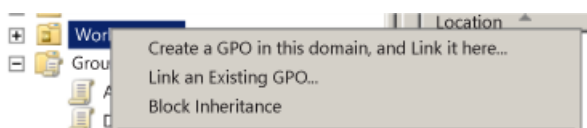
6. Ensure that UAC is enabled and that UAC restrictions apply to the default Administrator account by doing the following:
 - a. Navigate to the Computer Configuration\Windows Settings\Security Settings\Local Policies, and **> Security Options**.
 - b. Double-click **User Account Control: Run all administrators in Admin Approval Mode** **> Enabled** **> OK**.
 - c. Double-click **User Account Control: Admin Approval Mode for the Built-in Administrator account** **> Enabled** **> OK**.
7. Ensure that the local account restrictions are applied to network interfaces by doing the following:
 - a. Navigate to Computer Configuration\Preferences and Windows Settings, and **> Registry**.
 - b. Right-click **Registry**, and **> New** **> Registry Item**.



- c. In the **New Registry Properties** dialog box, on the **General** tab, change the setting in the **Action** box to **Replace**.
- d. Ensure that the **Hive** box is set to **HKEY_LOCAL_MACHINE**.
- e. Click (...), browse to the following location for **Key Path** > **Select for**:
SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System.
- f. In the **Value name** area, type **LocalAccountTokenFilterPolicy**.
- g. In the **Value type** box, from the drop-down list, select **REG_DWORD** to change the value.
- h. In the **Value data** box, ensure that the value is set to **0**.
- i. Verify this configuration, and > **OK**.



8. Link the GPO to the first **Workstations** organizational unit (OU) by doing the following:
 - a. Navigate to the <Forest>\Domains\<Domain>\OU path.
 - b. Right-click the **Workstations** OU, and > **Link an existing GPO**.



- c. Select the GPO that you just created, and > **OK**.
9. Test the functionality of enterprise applications on the workstations in that first OU and resolve any issues caused by the new policy.

10. Create links to all other OUs that contain workstations.

11. Create links to all other OUs that contain servers.

Deny network logon to all local Administrator accounts

Denying local accounts the ability to perform network logons can help prevent a local account password hash from being reused in a malicious attack. This procedure helps to prevent lateral movement by ensuring that the credentials for local accounts that are stolen from a compromised operating system cannot be used to compromise additional computers that use the same credentials.

NOTE

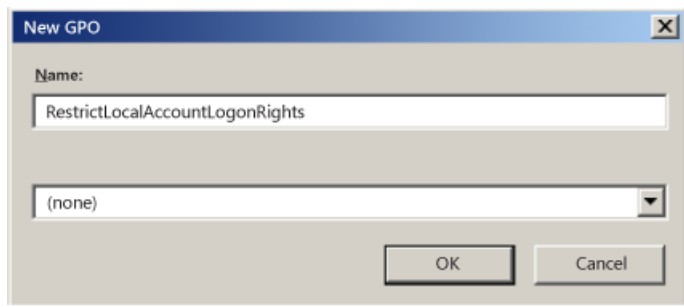
To perform this procedure, you must first identify the name of the local, default Administrator account, which might not be the default user name "Administrator", and any other accounts that are members of the local Administrators group.

The following table shows the Group Policy settings that are used to deny network logon for all local Administrator accounts.

NO.	SETTING	DETAILED DESCRIPTION
	Policy location	Computer Configuration\Windows Settings\Security Settings\Local Policies\User Rights Assignment
1	Policy name	Deny access to this computer from the network
	Policy setting	Local account and member of Administrators group
2	Policy location	Computer Configuration\Windows Settings\Security Settings\Local Policies\User Rights Assignment
	Policy name	Deny log on through Remote Desktop Services
	Policy setting	Local account and member of Administrators group

To deny network logon to all local administrator accounts

1. Start the **Group Policy Management** Console (GPMC).
2. In the console tree, expand <Forest>\Domains\<Domain>, and then **Group Policy Objects**, where *forest* is the name of the forest, and *domain* is the name of the domain where you want to set the Group Policy Object (GPO).
3. In the console tree, right-click **Group Policy Objects**, and > **New**.
4. In the **New GPO** dialog box, type <gpo_name>, and then > **OK** where *gpo_name* is the name of the new GPO indicates that it is being used to restrict the local administrative accounts from interactively signing in to the computer.



5. In the details pane, right-click <gpo_name>, and > **Edit**.



6. Configure the user rights to deny network logons for administrative local accounts as follows:

- a. Navigate to the Computer Configuration\Windows Settings\Security Settings\, and > **User Rights Assignment**.
- b. Double-click **Deny access to this computer from the network**.
- c. Click **Add User or Group**, type **Local account and member of Administrators group**, and > **OK**.

7. Configure the user rights to deny Remote Desktop (Remote Interactive) logons for administrative local accounts as follows:

- a. Navigate to Computer Configuration\Policies\Windows Settings and Local Policies, and then click **User Rights Assignment**.
- b. Double-click **Deny log on through Remote Desktop Services**.
- c. Click **Add User or Group**, type **Local account and member of Administrators group**, and > **OK**.

8. Link the GPO to the first **Workstations** OU as follows:

- a. Navigate to the <Forest>\Domains\<Domain>\OU path.
- b. Right-click the **Workstations** OU, and > **Link an existing GPO**.
- c. Select the GPO that you just created, and > **OK**.

9. Test the functionality of enterprise applications on the workstations in that first OU and resolve any issues caused by the new policy.

10. Create links to all other OUs that contain workstations.

11. Create links to all other OUs that contain servers.

NOTE

You might have to create a separate GPO if the user name of the default Administrator account is different on workstations and servers.

Create unique passwords for local accounts with administrative rights

Passwords should be unique per individual account. While this is generally true for individual user accounts, many enterprises have identical passwords for common local accounts, such as the default Administrator account. This also occurs when the same passwords are used for local accounts during operating system deployments.

Passwords that are left unchanged or changed synchronously to keep them identical add a significant risk for organizations. Randomizing the passwords mitigates "pass-the-hash" attacks by using different passwords for local accounts, which hampers the ability of malicious users to use password hashes of those accounts to compromise other computers.

Passwords can be randomized by:

- Purchasing and implementing an enterprise tool to accomplish this task. These tools are commonly referred to as "privileged password management" tools.
- Configuring [Local Administrator Password Solution \(LAPS\)](#) to accomplish this task.
- Creating and implementing a custom script or solution to randomize local account passwords.

See also

The following resources provide additional information about technologies that are related to local accounts.

- [Security Principals](#)
- [Security Identifiers](#)
- [Access Control Overview](#)

Active Directory Accounts

7/1/2022 • 35 minutes to read • [Edit Online](#)

Applies to

- Windows Server 2016

Windows Server operating systems are installed with default local accounts. In addition, you can create user accounts to meet the requirements of your organization. This reference topic for the IT professional describes the Windows Server default local accounts that are stored locally on the domain controller and are used in Active Directory.

This reference topic does not describe default local user accounts for a member or standalone server or for a Windows client. For more information, see [Local Accounts](#).

About this topic

This topic describes the following:

- [Default local accounts in Active Directory](#)
 - [Administrator account](#)
 - [Guest account](#)
 - [HelpAssistant account \(installed with a Remote Assistance session\)](#)
 - [KRBTGT account](#)
- [Settings for default local accounts in Active Directory](#)
- [Manage default local accounts in Active Directory](#)
- [Restrict and protect sensitive domain accounts](#)
 - [Separate administrator accounts from user accounts](#)
 - [Create dedicated workstation hosts without Internet and email access](#)
 - [Restrict administrator logon access to servers and workstations](#)
 - [Disable the account delegation right for administrator accounts](#)
- [Secure and manage domain controllers](#)

Default local accounts in Active Directory

Default local accounts are built-in accounts that are created automatically when a Windows Server domain controller is installed and the domain is created. These default local accounts have counterparts in Active Directory. These accounts also have domain-wide access and are completely separate from the default local user accounts for a member or standalone server.

You can assign rights and permissions to default local accounts on a particular domain controller, and only on that domain controller. These accounts are local to the domain. After the default local accounts are installed, they are stored in the Users container in Active Directory Users and Computers. It is a best practice to keep the default local accounts in the User container and not attempt to move these accounts, for example, to a different

organizational unit (OU).

The default local accounts in the Users container include: Administrator, Guest, and KRBTGT. The HelpAssistant account is installed when a Remote Assistance session is established. The following sections describe the default local accounts and their use in Active Directory.

Primarily, default local accounts do the following:

- Let the domain represent, identify, and authenticate the identity of the user that is assigned to the account by using unique credentials (user name and password). It is a best practice to assign each user to a single account to ensure maximum security. Multiple users are not allowed to share one account. A user account lets a user sign in to computers, networks, and domains with a unique identifier that can be authenticated by the computer, network, or domain.
- Authorize (grant or deny) access to resources. After a user's credentials have been authenticated, the user is authorized to access the network and domain resources based on the user's explicitly assigned rights on the resource.
- Audit the actions that are carried out on a user account.

In Active Directory, default local accounts are used by administrators to manage domain and member servers directly and from dedicated administrative workstations. Active Directory accounts provide access to network resources. Active Directory User accounts and Computer accounts can represent a physical entity, such as a computer or person, or act as dedicated service accounts for some applications.

Each default local account is automatically assigned to a security group that is preconfigured with the appropriate rights and permissions to perform specific tasks. Active Directory security groups collect user accounts, computer accounts, and other groups into manageable units. For more information, see [Active Directory Security Groups](#).

On an Active Directory domain controller, each default local account is referred to as a security principal. A security principal is a directory object that is used to secure and manage Active Directory services that provide access to domain controller resources. A security principal includes objects such as user accounts, computer accounts, security groups, or the threads or processes that run in the security context of a user or computer account. For more information, see [Security Principals](#).

A security principal is represented by a unique security identifier (SID). The SIDs that are related to each of the default local accounts in Active Directory are described in the sections below.

Some of the default local accounts are protected by a background process that periodically checks and applies a specific security descriptor. A security descriptor is a data structure that contains security information that is associated with a protected object. This process ensures that any successful unauthorized attempt to modify the security descriptor on one of the default local accounts or groups is overwritten with the protected settings.

This security descriptor is present on the AdminSDHolder object. If you want to modify the permissions on one of the service administrator groups or on any of its member accounts, you must modify the security descriptor on the AdminSDHolder object to ensure that it is applied consistently. Be careful when making these modifications, because you are also changing the default settings that are applied to all of your protected accounts.

Administrator account

The Administrator account is a default account that is used in all versions of the Windows operating system on every computer and device. The Administrator account is used by the system administrator for tasks that require administrative credentials. This account cannot be deleted or locked out, but the account can be renamed or disabled.

The Administrator account gives the user complete access (Full Control permissions) of the files, directories, services, and other resources that are on that local server. The Administrator account can be used to create local users, and assign user rights and access control permissions. Administrator can also be used to take control of local resources at any time simply by changing the user rights and permissions. Although files and directories can be protected from the Administrator account temporarily, the Administrator account can take control of these resources at any time by changing the access permissions.

Account group membership

The Administrator account has membership in the default security groups as described in the Administrator account attributes table later in this topic.

The security groups ensure that you can control administrator rights without having to change each Administrator account. In most instances, you do not have to change the basic settings for this account. However, you might have to change its advanced settings, such as membership in particular groups.

Security considerations

After installation of the server operating system, your first task is to set up the Administrator account properties securely. This includes setting up an especially long, strong password, and securing the Remote control and Remote Desktop Services profile settings.

The Administrator account can also be disabled when it is not required. Renaming or disabling the Administrator account makes it more difficult for malicious users to try to gain access to the account. However, even when the Administrator account is disabled, it can still be used to gain access to a domain controller by using safe mode.

On a domain controller, the Administrator account becomes the Domain Admin account. The Domain Admin account is used to sign in to the domain controller and this account requires a strong password. The Domain Admin account gives you access to domain resources.

NOTE

When the domain controller is initially installed, you can sign in and use Server Manager to set up a local Administrator account, with the rights and permissions you want to assign. For example, you can use a local Administrator account to manage the operating system when you first install it. By using this approach, you can set up the operating system without getting locked out. Generally, you do not need to use the account after installation. You can only create local user accounts on the domain controller, before Active Directory Domain Services is installed, and not afterwards.

When Active Directory is installed on the first domain controller in the domain, the Administrator account is created for Active Directory. The Administrator account is the most powerful account in the domain. It is given domain-wide access and administrative rights to administer the computer and the domain, and it has the most extensive rights and permissions over the domain. The person who installs Active Directory Domain Services on the computer creates the password for this account during the installation.

Administrator account attributes

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5- <input style="width: 50px;" type="text" value=" <domain> "/> -500
Type	User
Default container	CN=Users, DC= <input style="width: 50px;" type="text" value=" <domain> "/> , DC=
Default members	N/A

ATTRIBUTE	VALUE
Default member of	Administrators, Domain Admins, Enterprise Administrators, Domain Users. Note that the Primary Group ID of all user accounts is Domain Users. Group Policy Creator Owners, and Schema Admins in Active Directory Domain Users group
Protected by ADMINSDHOLDER?	Yes
Safe to move out of default container?	Yes
Safe to delegate management of this group to non-service administrators?	No

Guest account

The Guest account is a default local account that has limited access to the computer and is disabled by default. By default, the Guest account password is left blank. A blank password allows the Guest account to be accessed without requiring the user to enter a password.

The Guest account enables occasional or one-time users, who do not have an individual account on the computer, to sign in to the local server or domain with restricted rights and permissions. The Guest account can be enabled, and the password can be set up if needed, but only by a member of the Administrator group on the domain.

Account group membership

The Guest account has membership in the default security groups that are described in the following Guest account attributes table. By default, the Guest account is the only member of the default Guests group, which lets a user sign in to a server, and the Domain Guests global group, which lets a user sign in to a domain.

A member of the Administrators group or Domain Admins group can set up a user with a Guest account on one or more computers.

Security considerations

Because the Guest account can provide anonymous access, it is a security risk. It also has a well-known SID. For this reason, it is a best practice to leave the Guest account disabled, unless its use is required and then only with restricted rights and permissions for a very limited period of time.

When the Guest account is required, an Administrator on the domain controller is required to enable the Guest account. The Guest account can be enabled without requiring a password, or it can be enabled with a strong password. The Administrator also grants restricted rights and permissions for the Guest account. To help prevent unauthorized access:

- Do not grant the Guest account the [Shut down the system](#) user right. When a computer is shutting down or starting up, it is possible that a Guest user or anyone with local access, such as a malicious user, could gain unauthorized access to the computer.
- Do not provide the Guest account with the ability to view the event logs. After the Guest account is enabled, it is a best practice to monitor this account frequently to ensure that other users cannot use services and other resources, such as resources that were unintentionally left available by a previous user.
- Do not use the Guest account when the server has external network access or access to other computers.

If you decide to enable the Guest account, be sure to restrict its use and to change the password regularly. As with the Administrator account, you might want to rename the account as an added security precaution.

In addition, an administrator is responsible for managing the Guest account. The administrator monitors the Guest account, disables the Guest account when it is no longer in use, and changes or removes the password as needed.

For details about the Guest account attributes, see the following table.

Guest account attributes

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5- <input type="text" value="<domain>"/> -501
Type	User
Default container	CN=Users, DC= <input type="text" value="<domain>"/> , DC=
Default members	None
Default member of	Guests, Domain Guests
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Can be moved out, but we do not recommend it.
Safe to delegate management of this group to non-Service admins?	No

HelpAssistant account (installed with a Remote Assistance session)

The HelpAssistant account is a default local account that is enabled when a Remote Assistance session is run. This account is automatically disabled when no Remote Assistance requests are pending.

HelpAssistant is the primary account that is used to establish a Remote Assistance session. The Remote Assistance session is used to connect to another computer running the Windows operating system, and it is initiated by invitation. For solicited remote assistance, a user sends an invitation from their computer, through e-mail or as a file, to a person who can provide assistance. After the user's invitation for a Remote Assistance session is accepted, the default HelpAssistant account is automatically created to give the person who provides assistance limited access to the computer. The HelpAssistant account is managed by the Remote Desktop Help Session Manager service.

Security considerations

The SIDs that pertain to the default HelpAssistant account include:

- SID: S-1-5--13, display name Terminal Server User. This group includes all users who sign in to a server with Remote Desktop Services enabled. Note that, in Windows Server 2008, Remote Desktop Services are called Terminal Services.
- SID: S-1-5--14, display name Remote Interactive Logon. This group includes all users who connect to the computer by using a remote desktop connection. This group is a subset of the Interactive group. Access tokens that contain the Remote Interactive Logon SID also contain the Interactive SID.

For the Windows Server operating system, Remote Assistance is an optional component that is not installed by default. You must install Remote Assistance before it can be used.

For details about the HelpAssistant account attributes, see the following table.

HelpAssistant account attributes

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5- <domain> -13 (Terminal Server User), S-1-5- <domain> -14 (Remote Interactive Logon)
Type	User
Default container	CN=Users, DC= <domain> , DC=
Default members	None
Default member of	Domain Guests Guests
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Can be moved out, but we do not recommend it.
Safe to delegate management of this group to non-Service admins?	No

KRBTGT account

The KRBTGT account is a local default account that acts as a service account for the Key Distribution Center (KDC) service. This account cannot be deleted, and the account name cannot be changed. The KRBTGT account cannot be enabled in Active Directory.

KRBTGT is also the security principal name used by the KDC for a Windows Server domain, as specified by RFC 4120. The KRBTGT account is the entity for the KRBTGT security principal, and it is created automatically when a new domain is created.

Windows Server Kerberos authentication is achieved by the use of a special Kerberos ticket-granting ticket (TGT) enciphered with a symmetric key. This key is derived from the password of the server or service to which access is requested. The TGT password of the KRBTGT account is known only by the Kerberos service. In order to request a session ticket, the TGT must be presented to the KDC. The TGT is issued to the Kerberos client from the KDC.

KRBTGT account maintenance considerations

A strong password is assigned to the KRBTGT and trust accounts automatically. Like any privileged service accounts, organizations should change these passwords on a regular schedule. The password for the KDC account is used to derive a secret key for encrypting and decrypting the TGT requests that are issued. The password for a domain trust account is used to derive an inter-realm key for encrypting referral tickets.

Resetting the password requires you either to be a member of the Domain Admins group, or to have been delegated with the appropriate authority. In addition, you must be a member of the local Administrators group, or you must have been delegated the appropriate authority.

After you reset the KRBTGT password, ensure that event ID 9 in the (Kerberos) Key-Distribution-Center event

source is written to the System event log.

Security considerations

It is also a best practice to reset the KRBTGT account password to ensure that a newly restored domain controller does not replicate with a compromised domain controller. In this case, in a large forest recovery that is spread across multiple locations, you cannot guarantee that all domain controllers are shut down, and if they are shut down, they cannot be rebooted again before all of the appropriate recovery steps have been undertaken. After you reset the KRBTGT account, another domain controller cannot replicate this account password by using an old password.

An organization suspecting domain compromise of the KRBTGT account should consider the use of professional incident response services. The impact to restore the ownership of the account is domain-wide and labor intensive and should be undertaken as part of a larger recovery effort.

The KRBTGT password is the key from which all trust in Kerberos chains up to. Resetting the KRBTGT password is similar to renewing the root CA certificate with a new key and immediately not trusting the old key, resulting in almost all subsequent Kerberos operations will be affected.

For all account types (users, computers, and services)

- All the TGTs that are already issued and distributed will be invalid because the DCs will reject them. These tickets are encrypted with the KRBTGT so any DC can validate them. When the password changes, the tickets become invalid.
- All currently authenticated sessions that logged on users have established (based on their service tickets) to a resource (such as a file share, SharePoint site, or Exchange server) are good until the service ticket is required to re-authenticate.
- NTLM authenticated connections are not affected

Because it is impossible to predict the specific errors that will occur for any given user in a production operating environment, you must assume all computers and users will be affected.

IMPORTANT

Rebooting a computer is the only reliable way to recover functionality as this will cause both the computer account and user accounts to log back in again. Logging in again will request new TGTs that are valid with the new KRBTGT, correcting any KRBTGT related operational issues on that computer.

For information about how to help mitigate the risks associated with a potentially compromised KRBTGT account, see [KRBTGT Account Password Reset Scripts now available for customers](#).

Read-only domain controllers and the KRBTGT account

Windows Server 2008 introduced the read-only domain controller (RODC). The RODC is advertised as the Key Distribution Center (KDC) for the branch office. The RODC uses a different KRBTGT account and password than the KDC on a writable domain controller when it signs or encrypts ticket-granting ticket (TGT) requests. After an account is successfully authenticated, the RODC determines if a user's credentials or a computer's credentials can be replicated from the writable domain controller to the RODC by using the Password Replication Policy.

After the credentials are cached on the RODC, the RODC can accept that user's sign-in requests until the credentials change. When a TGT is signed with the KRBTGT account of the RODC, the RODC recognizes that it has a cached copy of the credentials. If another domain controller signs the TGT, the RODC forwards requests to a writable domain controller.

KRBTGT account attributes

For details about the KRBTGT account attributes, see the following table.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5- <domain> -502
Type	User
Default container	CN=Users, DC= <domain> , DC=
Default members	None
Default member of	Domain Users group. Note that the Primary Group ID of all user accounts is Domain Users.
Protected by ADMINSDHOLDER?	Yes
Safe to move out of default container?	Can be moved out, but we do not recommend it.
Safe to delegate management of this group to non-Service admins?	No

Settings for default local accounts in Active Directory

Each default local account in Active Directory has a number of account settings that you can use to configure password settings and security-specific information, as described in the following table.

Settings for default local accounts in Active Directory

ACCOUNT SETTINGS	DESCRIPTION
User must change password at next logon	Forces a password change the next time that the user logs signs in to the network. Use this option when you want to ensure that the user is the only person to know his or her password.
User cannot change password	Prevents the user from changing the password. Use this option when you want to maintain control over a user account, such as for a Guest or temporary account.
Password never expires	Prevents a user password from expiring. It is a best practice to enable this option with service accounts and to use strong passwords.
Store passwords using reversible encryption	Provides support for applications that use protocols requiring knowledge of the plaintext form of the user's password for authentication purposes. This option is required when using Challenge Handshake Authentication Protocol (CHAP) in Internet Authentication Services (IAS), and when using digest authentication in Internet Information Services (IIS).
Account is disabled	Prevents the user from signing in with the selected account. As an administrator, you can use disabled accounts as templates for common user accounts.

ACCOUNT SETTINGS	DESCRIPTION
Smart card is required for interactive logon	<p>Requires that a user has a smart card to sign on to the network interactively. The user must also have a smart card reader attached to their computer and a valid personal identification number (PIN) for the smart card.</p> <p>When this attribute is applied on the account, the effect is as follows:</p> <ul style="list-style-type: none"> • The attribute only restricts initial authentication for interactive logon and Remote Desktop logon. When interactive or Remote Desktop logon requires a subsequent network logon, such as with a domain credential, an NT Hash provided by the domain controller is used to complete the smartcard authentication process • Each time the attribute is enabled on an account, the account's current password hash value is replaced with a 128-bit random number. This invalidates the use of any previously configured passwords for the account. The value does not change after that unless a new password is set or the attribute is disabled and re-enabled. • Accounts with this attribute cannot be used to start services or run scheduled tasks.
Account is trusted for delegation	<p>Lets a service running under this account perform operations on behalf of other user accounts on the network. A service running under a user account (also known as a service account) that is trusted for delegation can impersonate a client to gain access to resources, either on the computer where the service is running or on other computers. For example, in a forest that is set to the Windows Server 2003 functional level, this setting is found on the Delegation tab. It is available only for accounts that have been assigned service principal names (SPNs), which are set by using the setspn command from Windows Support Tools. This setting is security-sensitive and should be assigned cautiously.</p>
Account is sensitive and cannot be delegated	<p>Gives control over a user account, such as for a Guest account or a temporary account. This option can be used if this account cannot be assigned for delegation by another account.</p>
Use DES encryption types for this account	<p>Provides support for the Data Encryption Standard (DES). DES supports multiple levels of encryption, including Microsoft Point-to-Point Encryption (MPPE) Standard (40-bit and 56-bit), MPPE standard (56-bit), MPPE Strong (128-bit), Internet Protocol security (IPSec) DES (40-bit), IPSec 56-bit DES, and IPSec Triple DES (3DES).</p> <div data-bbox="820 1742 1417 2101" style="border: 1px solid black; padding: 5px;"> <p>Note: DES is not enabled by default in Windows Server operating systems starting with Windows Server 2008 R2, nor in Windows client operating systems starting with Windows 7. For these operating systems, computers will not use DES-CBC-MD5 or DES-CBC-CRC cipher suites by default. If your environment requires DES, then this setting might affect compatibility with client computers or services and applications in your environment. For more information, see Hunting down DES in order to securely deploy Kerberos</p> </div>

ACCOUNT SETTINGS	DESCRIPTION
Do not require Kerberos preauthentication	Provides support for alternate implementations of the Kerberos protocol. Because preauthentication provides additional security, use caution when enabling this option. Note that domain controllers running Windows 2000 or Windows Server 2003 can use other mechanisms to synchronize time.

Manage default local accounts in Active Directory

After the default local accounts are installed, these accounts reside in the Users container in Active Directory Users and Computers. Default local accounts can be created, disabled, reset, and deleted by using the Active Directory Users and Computers Microsoft Management Console (MMC) and by using command-line tools.

You can use Active Directory Users and Computers to assign rights and permissions on a given local domain controller, and that domain controller only, to limit the ability of local users and groups to perform certain actions. A right authorizes a user to perform certain actions on a computer, such as backing up files and folders or shutting down a computer. In contrast, an access permission is a rule that is associated with an object, usually a file, folder, or printer, that regulates which users can have access to the object and in what manner.

For more information about creating and managing local user accounts in Active Directory, see [Manage Local Users](#).

You can also use Active Directory Users and Computers on a domain controller to target remote computers that are not domain controllers on the network.

You can obtain recommendations from Microsoft for domain controller configurations that you can distribute by using the Security Compliance Manager (SCM) tool. For more information, see [Microsoft Security Compliance Manager](#).

Some of the default local user accounts are protected by a background process that periodically checks and applies a specific security descriptor, which is a data structure that contains security information that is associated with a protected object. This security descriptor is present on the AdminSDHolder object.

This means, when you want to modify the permissions on a service administrator group or on any of its member accounts, you are also required to modify the security descriptor on the AdminSDHolder object. This approach ensures that the permissions are applied consistently. Be careful when you make these modifications, because this action can also affect the default settings that are applied to all of your protected administrative accounts.

Restrict and protect sensitive domain accounts

Restricting and protecting domain accounts in your domain environment requires you to adopt and implement the following best practices approach:

- Strictly limit membership to the Administrators, Domain Admins, and Enterprise Admins groups.
- Stringently control where and how domain accounts are used.

Member accounts in the Administrators, Domain Admins, and Enterprise Admins groups in a domain or forest are high-value targets for malicious users. It is a best practice to strictly limit membership to these administrator groups to the smallest number of accounts in order to limit any exposure. Restricting membership in these groups reduces the possibility that an administrator might unintentionally misuse these credentials and create a vulnerability that malicious users can exploit.

Moreover, it is a best practice to stringently control where and how sensitive domain accounts are used. Restrict

the use of Domain Admins accounts and other administrator accounts to prevent them from being used to sign in to management systems and workstations that are secured at the same level as the managed systems. When administrator accounts are not restricted in this manner, each workstation from which a domain administrator signs in provides another location that malicious users can exploit.

Implementing these best practices is separated into the following tasks:

- [Separate administrator accounts from user accounts](#)
- [Create dedicated workstation hosts for administrators](#)
- [Restrict administrator logon access to servers and workstations](#)
- [Disable the account delegation right for administrator accounts](#)

Note that, to provide for instances where integration challenges with the domain environment are expected, each task is described according to the requirements for a minimum, better, and ideal implementation. As with all significant changes to a production environment, ensure that you test these changes thoroughly before you implement and deploy them. Then stage the deployment in a manner that allows for a rollback of the change in case technical issues occur.

Separate administrator accounts from user accounts

Restrict Domain Admins accounts and other sensitive accounts to prevent them from being used to sign in to lower trust servers and workstations. Restrict and protect administrator accounts by segregating administrator accounts from standard user accounts, by separating administrative duties from other tasks, and by limiting the use of these accounts. Create dedicated accounts for administrative personnel who require administrator credentials to perform specific administrative tasks, and then create separate accounts for other standard user tasks, according to the following guidelines:

- **Privileged account.** Allocate administrator accounts to perform the following administrative duties only:
 - **Minimum.** Create separate accounts for domain administrators, enterprise administrators, or the equivalent with appropriate administrator rights in the domain or forest. Use accounts that have been granted sensitive administrator rights only to administer domain data and domain controllers.
 - **Better.** Create separate accounts for administrators that have reduced administrative rights, such as accounts for workstation administrators, and accounts with user rights over designated Active Directory organizational units (OUs).
 - **Ideal.** Create multiple, separate accounts for an administrator who has a variety of job responsibilities that require different trust levels. Set up each administrator account with significantly different user rights, such as for workstation administration, server administration and domain administration, to let the administrator sign in to given workstations, servers and domain controllers based strictly on his or her job responsibilities.
- **Standard user account.** Grant standard user rights for standard user tasks, such as email, web browsing, and using line-of-business (LOB) applications. These accounts should not be granted administrator rights.

IMPORTANT

Ensure that sensitive administrator accounts cannot access email or browse the Internet as described in the following section.

Create dedicated workstation hosts without Internet and email access

Administrators need to manage job responsibilities that require sensitive administrator rights from a dedicated workstation because they do not have easy physical access to the servers. A workstation that is connected to the Internet and has email and web browsing access is regularly exposed to compromise through phishing, downloading, and other types of Internet attacks. Because of these threats, it is a best practice to set these administrators up by using workstations that are dedicated to administrative duties only, and not provide access to the Internet, including email and web browsing. For more information, see [Separate administrator accounts from user accounts](#).

NOTE

If the administrators in your environment can sign in locally to managed servers and perform all tasks without elevated rights or domain rights from their workstation, you can skip this task.

- **Minimum.** Build dedicated administrative workstations and block Internet access on those workstations including web browsing and email. Use the following ways to block Internet access:
 - Configure authenticating boundary proxy services, if they are deployed, to disallow administrator accounts from accessing the Internet.
 - Configure boundary firewall or proxy services to disallow Internet access for the IP addresses that are assigned to dedicated administrative workstations.
 - Block outbound access to the boundary proxy servers in the Windows Firewall.

The instructions for meeting this minimum requirement are described in the following procedure.

- **Better.** Do not grant administrators membership in the local Administrator group on the computer in order to restrict the administrator from bypassing these protections.
- **Ideal.** Restrict workstations from having any network connectivity, except for the domain controllers and servers that the administrator accounts are used to manage. Alternately, use AppLocker application control policies to restrict all applications from running, except for the operating system and approved administrative tools and applications. For more information about AppLocker, see [AppLocker](#).

The following procedure describes how to block Internet access by creating a Group Policy Object (GPO) that configures an invalid proxy address on administrative workstations. These instructions apply only to computers running Internet Explorer and other Windows components that use these proxy settings.

NOTE

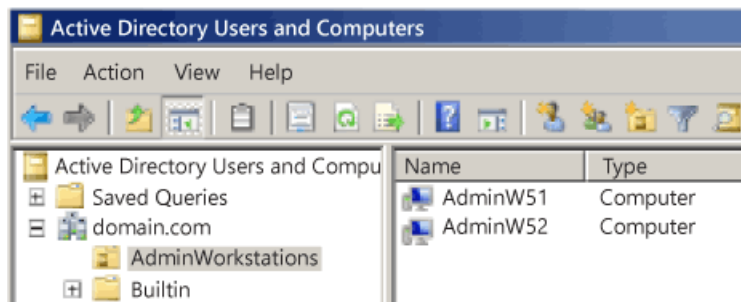
In this procedure, the workstations are dedicated to domain administrators. By simply modifying the administrator accounts to grant permission to administrators to sign in locally, you can create additional OUs to manage administrators that have fewer administrative rights to use the instructions described in the following procedure.

To install administrative workstations in a domain and block Internet and email access (minimum)

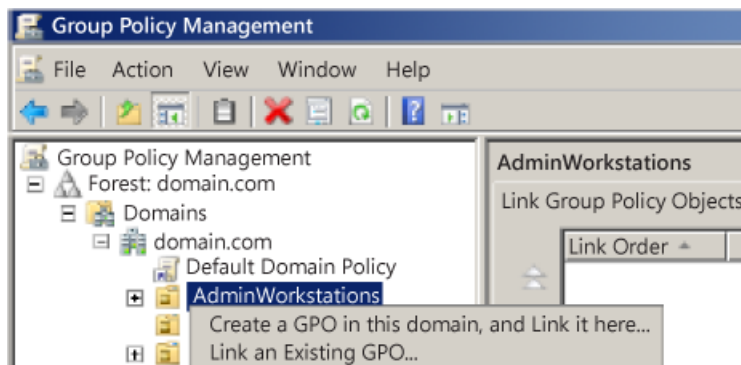
1. As a domain administrator on a domain controller, open Active Directory Users and Computers, and create a new OU for administrative workstations.
2. Create computer accounts for the new workstations.

NOTE

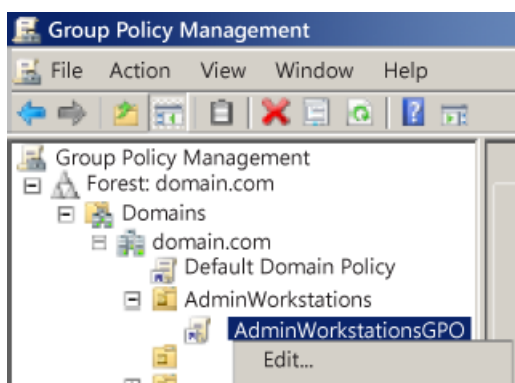
You might have to delegate permissions to join computers to the domain if the account that joins the workstations to the domain does not already have them. For more information, see [Delegation of Administration in Active Directory](#).



3. Close Active Directory Users and Computers.
4. Start the **Group Policy Management** Console (GPMC).
5. Right-click the new OU, and > **Create a GPO in this domain, and Link it here.**



6. Name the GPO, and > **OK.**
7. Expand the GPO, right-click the new GPO, and > **Edit.**

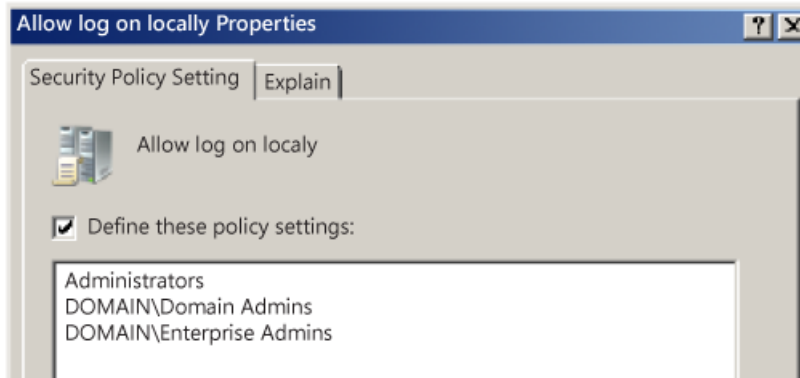


8. Configure which members of accounts can log on locally to these administrative workstations as follows:
 - a. Navigate to Computer Configuration\Policies\Windows Settings\Local Policies, and then click **User Rights Assignment**.
 - b. Double-click **Allow log on locally**, and then select the **Define these policy settings** check box.
 - c. Click **Add User or Group** > **Browse**, type **Enterprise Admins**, and > **OK**.
 - d. Click **Add User or Group** > **Browse**, type **Domain Admins**, and > **OK**.

IMPORTANT

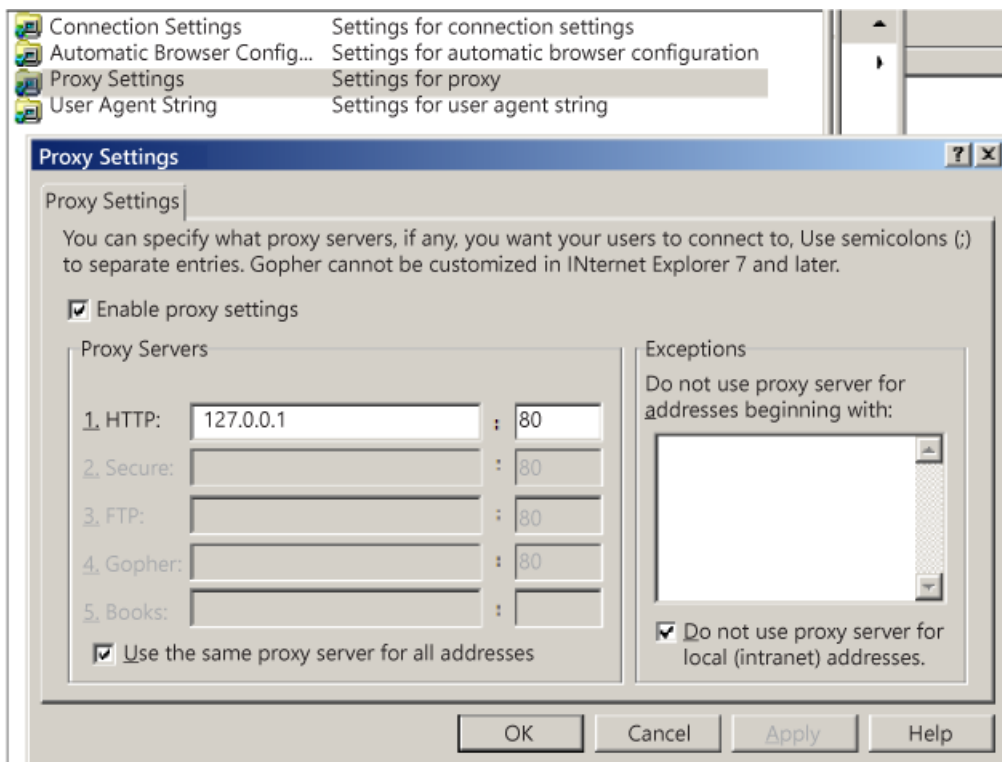
These instructions assume that the workstation is to be dedicated to domain administrators.

- e. Click **Add User or Group**, type **Administrators**, and > **OK**.



9. Configure the proxy configuration:

- a. Navigate to **User Configuration\Policies\Windows Settings\Internet Explorer**, and > **Connection**.
- b. Double-click **Proxy Settings**, select the **Enable proxy settings** check box, type **127.0.0.1** (the network Loopback IP address) as the proxy address, and > **OK**.



10. Configure the loopback processing mode to enable the user Group Policy proxy setting to apply to all users on the computer as follows:
 - a. Navigate to **Computer Configuration\Policies\Administrative Templates\System**, and > **Group Policy**.
 - b. Double-click **User Group Policy loopback policy processing mode**, and > **Enabled**.
 - c. Select **Merge Mode**, and > **OK**.
11. Configure software updates as follows:

- a. Navigate to Computer Configuration\Policies\Administrative Templates\Windows Components, and then click **Windows Update**.
- b. Configure Windows Update settings as described in the following table.

WINDOWS UPDATE SETTING	CONFIGURATION
Allow Automatic Updates immediate installation	Enabled
Configure Automatic Updates	Enabled4 - Auto download and schedule the installation0 - Every day 03:00
Enable Windows Update Power Management to automatically wake up the system to install scheduled updates	Enabled
Specify intranet Microsoft Update service location	Enabled <pre>http://<WSUSServername></pre> <pre>http://<WSUSServername></pre> Where <WSUSServername> is the DNS name or IP address of the Windows Server Update Services (WSUS) in the environment.
Automatic Updates detection frequency	6 hours
Re-prompt for restart with scheduled installations	1 minute
Delay restart for scheduled installations	5 minutes

NOTE

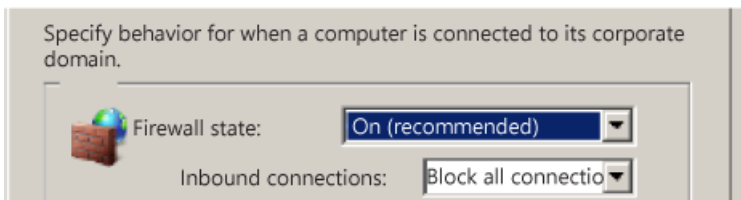
This step assumes that Windows Server Update Services (WSUS) is installed and configured in the environment. You can skip this step if you use another tool to deploy software updates. Also, if the public Microsoft Windows Update service only is used on the Internet, then these administrative workstations no longer receive updates.

12. Configure the inbound firewall to block all connections as follows:

- a. Right-click **Windows Firewall with Advanced Security LDAP://path**, and > **Properties**.



- b. On each profile, ensure that the firewall is enabled and that inbound connections are set to **Block all connections**.



c. Click **OK** to complete the configuration.

13. Close the Group Policy Management Console.

14. Install the Windows operating system on the workstations, give each workstation the same names as the computer accounts assigned to them, and then join them to the domain.

Restrict administrator logon access to servers and workstations

It is a best practice to restrict administrators from using sensitive administrator accounts to sign in to lower-trust servers and workstations. This restriction prevents administrators from inadvertently increasing the risk of credential theft by signing in to a lower-trust computer.

IMPORTANT

Ensure that you either have local access to the domain controller or that you have built at least one dedicated administrative workstation.

Restrict logon access to lower-trust servers and workstations by using the following guidelines:

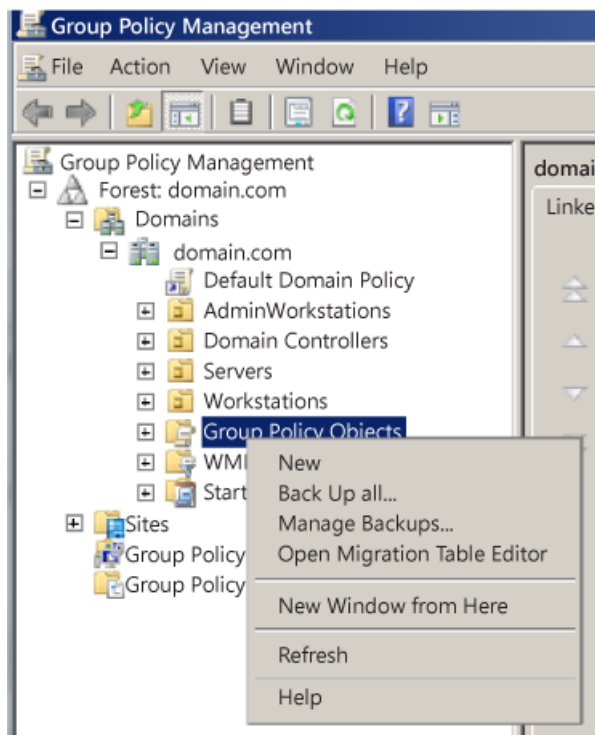
- **Minimum.** Restrict domain administrators from having logon access to servers and workstations. Before starting this procedure, identify all OUs in the domain that contain workstations and servers. Any computers in OUs that are not identified will not restrict administrators with sensitive accounts from signing-in to them.
- **Better.** Restrict domain administrators from non-domain controller servers and workstations.
- **Ideal.** Restrict server administrators from signing in to workstations, in addition to domain administrators.

NOTE

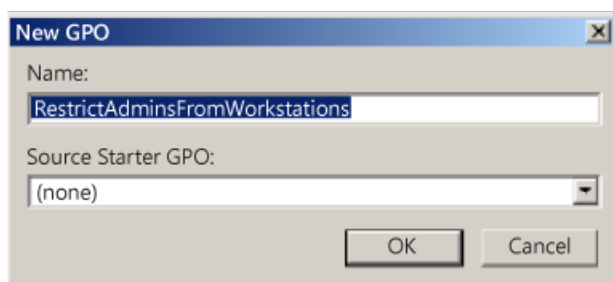
For this procedure, do not link accounts to the OU that contain workstations for administrators that perform administration duties only, and do not provide Internet or email access. For more information, see [Create dedicated workstation hosts for administrators](#)

To restrict domain administrators from workstations (minimum)

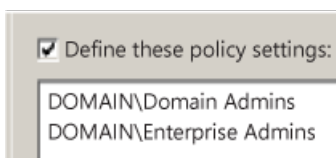
1. As a domain administrator, open the Group Policy Management Console (GPMC).
2. Open **Group Policy Management**, and expand `<forest>\Domains\<domain>`, and then expand to **Group Policy Objects**.
3. Right-click **Group Policy Objects**, and > **New**.



4. In the **New GPO** dialog box, name the GPO that restricts administrators from signing in to workstations, and > **OK**.



5. Right-click **New GPO**, and > **Edit**.
6. Configure user rights to deny logon locally for domain administrators.
7. Navigate to Computer Configuration\Policies\Windows Settings\Local Policies, and then click **User Rights Assignment**, and perform the following:
 - a. Double-click **Deny logon locally**, and > **Define these policy settings**.
 - b. Click **Add User or Group**, click **Browse**, type **Enterprise Admins**, and > **OK**.
 - c. Click **Add User or Group**, click **Browse**, type **Domain Admins**, and > **OK**.



NOTE

You can optionally add any groups that contain server administrators who you want to restrict from signing in to workstations.

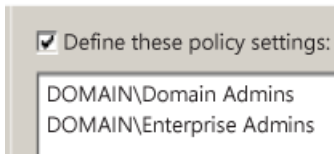
- d. Click **OK** to complete the configuration.

8. Configure the user rights to deny batch and service logon rights for domain administrators as follows:

NOTE

Completing this step might cause issues with administrator tasks that run as scheduled tasks or services with accounts in the Domain Admins group. The practice of using domain administrator accounts to run services and tasks on workstations creates a significant risk of credential theft attacks and therefore should be replaced with alternative means to run scheduled tasks or services.

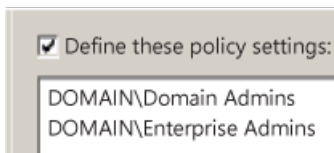
- a. Double-click **Deny logon as a batch job**, and > **Define these policy settings**.
- b. Click **Add User or Group** > **Browse**, type **Enterprise Admins**, and > **OK**.
- c. Click **Add User or Group** > **Browse**, type **Domain Admins**, and > **OK**.



NOTE

You can optionally add any groups that contain server administrators who you want to restrict from signing in to workstations.

- d. Double-click **Deny logon as a service**, and > **Define these policy settings**.
- e. Click **Add User or Group** > **Browse**, type **Enterprise Admins**, and > **OK**.
- f. Click **Add User or Group** > **Browse**, type **Domain Admins**, and > **OK**.



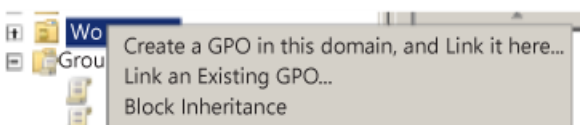
NOTE

You can optionally add any groups that contain server administrators who you want to restrict from signing in to workstations.

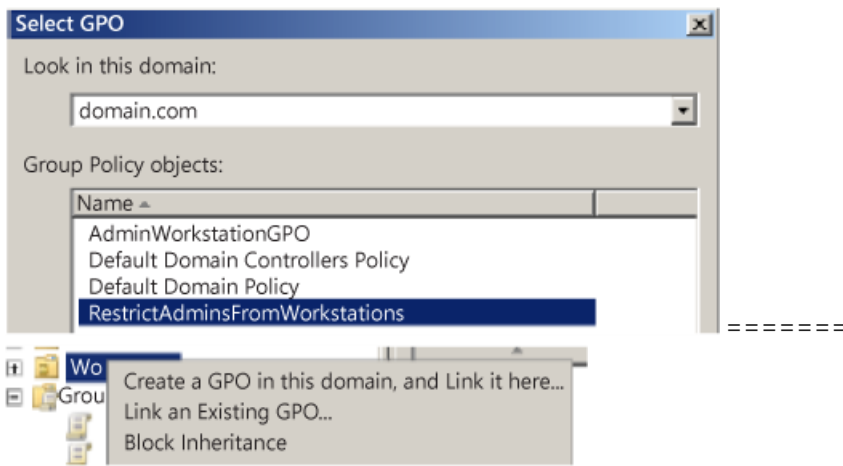
9. Link the GPO to the first Workstations OU.

Navigate to the <forest>\Domains\<domain>\OU Path, and then:

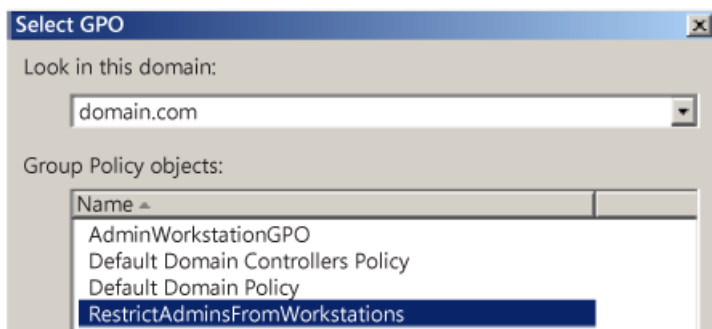
- a. Right-click the workstation OU, and then > **Link an Existing GPO**.



- b. Select the GPO that you just created, and > **OK**.



c. Select the GPO that you just created, and > OK.



10. Test the functionality of enterprise applications on workstations in the first OU and resolve any issues caused by the new policy.
11. Link all other OUs that contain workstations.

However, do not create a link to the Administrative Workstation OU if it is created for administrative workstations that are dedicated to administration duties only, and that are without Internet or email access. For more information, see [Create dedicated workstation hosts for administrators](#).

IMPORTANT

If you later extend this solution, do not deny logon rights for the **Domain Users** group. The **Domain Users** group includes all user accounts in the domain, including Users, Domain Administrators, and Enterprise Administrators.

Disable the account delegation right for sensitive administrator accounts

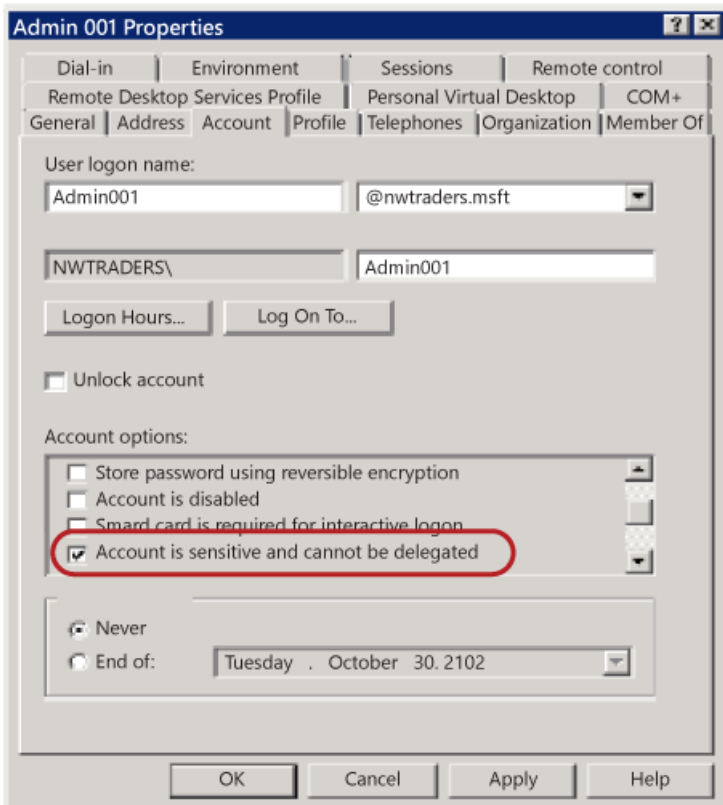
Although user accounts are not marked for delegation by default, accounts in an Active Directory domain can be trusted for delegation. This means that a service or a computer that is trusted for delegation can impersonate an account that authenticates to them to access other resources across the network.

For sensitive accounts, such as those belonging to members of the Administrators, Domain Admins, or Enterprise Admins groups in Active Directory, delegation can present a substantial risk of rights escalation. For example, if an account in the Domain Admins group is used to sign in to a compromised member server that is trusted for delegation, that server can request access to resources in the context of the Domain Admins account, and escalate the compromise of that member server to a domain compromise.

It is a best practice to configure the user objects for all sensitive accounts in Active Directory by selecting the **Account is sensitive and cannot be delegated** check box under **Account options** to prevent these accounts from being delegated. For more information, see [Setting for default local accounts in Active Directory](#).

As with any configuration change, test this enabled setting fully to ensure that it performs correctly before you

implement it.



Secure and manage domain controllers

It is a best practice to strictly enforce restrictions on the domain controllers in your environment. This ensures that the domain controllers:

1. Run only required software
2. Required software is regularly updated
3. Are configured with the appropriate security settings

One aspect of securing and managing domain controllers is to ensure that the default local user accounts are fully protected. It is of primary importance to restrict and secure all sensitive domain accounts, as described in the preceding sections.

Because domain controllers store credential password hashes of all accounts in the domain, they are high-value targets for malicious users. When domain controllers are not well managed and secured by using restrictions that are strictly enforced, they can be compromised by malicious users. For example, a malicious user could steal sensitive domain administrator credentials from one domain controller, and then use these credentials to attack the domain and forest.

In addition, installed applications and management agents on domain controllers might provide a path for escalating rights that malicious users can use to compromise the management service or administrators of that service. The management tools and services, which your organization uses to manage domain controllers and their administrators, are equally important to the security of the domain controllers and the domain administrator accounts. Ensure that these services and administrators are fully secured with equal effort.

See also

- [Security Principals](#)
- [Access Control Overview](#)

Microsoft Accounts

7/1/2022 • 9 minutes to read • [Edit Online](#)

Applies to

- Windows 10

This topic for the IT professional explains how a Microsoft account works to enhance security and privacy for users, and how you can manage this consumer account type in your organization.

Microsoft sites, services, and properties, as well as computers running Windows 10, can use a Microsoft account as a means of identifying a user. Microsoft account was previously called Windows Live ID. It has user-defined secrets, and consists of a unique email address and a password.

When a user signs in with a Microsoft account, the device is connected to cloud services. Many of the user's settings, preferences, and apps can be shared across devices.

How a Microsoft account works

The Microsoft account allows users to sign in to websites that support this service by using a single set of credentials. Users' credentials are validated by a Microsoft account authentication server that is associated with a website. The Microsoft Store is an example of this association. When new users sign in to websites that are enabled to use Microsoft accounts, they are redirected to the nearest authentication server, which asks for a user name and password. Windows uses the Schannel Security Support Provider to open a Transport Level Security/Secure Sockets Layer (TLS/SSL) connection for this function. Users then have the option to use Credential Manager to store their credentials.

When users sign in to websites that are enabled to use a Microsoft account, a time-limited cookie is installed on their computers, which includes a triple DES encrypted ID tag. This encrypted ID tag has been agreed upon between the authentication server and the website. This ID tag is sent to the website, and the website plants another time-limited encrypted HTTP cookie on the user's computer. When these cookies are valid, users are not required to supply a user name and password. If a user actively signs out of their Microsoft account, these cookies are removed.

Important Local Windows account functionality has not been removed, and it is still an option to use in managed environments.

How Microsoft accounts are created

To prevent fraud, the Microsoft system verifies the IP address when a user creates an account. A user who tries to create multiple Microsoft accounts with the same IP address is stopped.

Microsoft accounts are not designed to be created in batches, such as for a group of domain users within your enterprise.

There are two methods for creating a Microsoft account:

- **Use an existing email address.**

Users are able to use their valid email addresses to sign up for Microsoft accounts. The service turns the requesting user's email address into a Microsoft account. Users can also choose their personal passwords.

- **Sign up for a Microsoft email address.**

Users can sign up for an email account with Microsoft's webmail services. This account can be used to sign in to websites that are enabled to use Microsoft accounts.

How the Microsoft account information is safeguarded

Credential information is encrypted twice. The first encryption is based on the account's password. Credentials are encrypted again when they are sent across the Internet. The data that is stored is not available to other Microsoft or non-Microsoft services.

- **Strong password is required.**

Blank passwords are not allowed.

For more information, see [How to help keep your Microsoft account safe and secure](#).

- **Secondary proof of identity is required.**

Before user profile information and settings can be accessed on a second supported Windows computer for the first time, trust must be established for that device by providing secondary proof of identity. This can be accomplished by providing Windows with a code that is sent to a mobile phone number or by following the instructions that are sent to an alternate email address that a user specifies in the account settings.

- **All user profile data is encrypted on the client before it is transmitted to the cloud.**

User data does not roam over a wireless wide area network (WWAN) by default, thereby protecting profile data. All data and settings that leave a device are transmitted through the TLS/SSL protocol.

Microsoft account security information is added.

Users can add security information to their Microsoft accounts through the **Accounts** interface on computers running the supported versions of Windows. This feature allows the user to update the security information that they provided when they created their accounts. This security information includes an alternate email address or phone number so if their password is compromised or forgotten, a verification code can be sent to verify their identity. Users can potentially use their Microsoft accounts to store corporate data on a personal OneDrive or email app, so it is safe practice for the account owner to keep this security information up-to-date.

The Microsoft account in the enterprise

Although the Microsoft account was designed to serve consumers, you might find situations where your domain users can benefit by using their personal Microsoft account in your enterprise. The following list describes some advantages.

- **Download Microsoft Store apps:**

If your enterprise chooses to distribute software through the Microsoft Store, your users can use their Microsoft accounts to download and use them on up to five devices running any version of Windows 10, Windows 8.1, Windows 8, or Windows RT.

- **Single sign-on:**

Your users can use Microsoft account credentials to sign in to devices running Windows 10, Windows 8.1, Windows 8 or Windows RT. When they do this, Windows works with your Microsoft Store app to provide authenticated experiences for them. Users can associate a Microsoft account with their sign-in credentials for Microsoft Store apps or websites, so that these credentials roam across any devices running these supported versions.

- **Personalized settings synchronization:**

Users can associate their most commonly used operating-system settings with a Microsoft account.

These settings are available whenever a user signs in with that account on any device that is running a supported version of Windows and is connected to the cloud. After a user signs in, the device automatically attempts to get the user's settings from the cloud and apply them to the device.

- **App synchronization:**

Microsoft Store apps can store user-specific settings so that these settings are available to any device. As with operating system settings, these user-specific app settings are available whenever the user signs in with the same Microsoft account on any device that is running a supported version of Windows and is connected to the cloud. After the user signs in, that device automatically downloads the settings from the cloud and applies them when the app is installed.

- **Integrated social media services:**

Contact information and status for your users' friends and associates automatically stay up-to-date from sites such as Hotmail, Outlook, Facebook, Twitter, and LinkedIn. Users can also access and share photos, documents, and other files from sites such as OneDrive, Facebook, and Flickr.

Managing the Microsoft account in the domain

Depending on your IT and business models, introducing Microsoft accounts into your enterprise might add complexity or it might provide solutions. You should address the following considerations before you allow the use of these account types in your enterprise:

- [Restrict the use of the Microsoft account](#)
- [Configure connected accounts](#)
- [Provision Microsoft accounts in the enterprise](#)
- [Audit account activity](#)
- [Perform password resets](#)
- [Restrict app installation and usage](#)

Restrict the use of the Microsoft account

The following Group Policy settings help control the use of Microsoft accounts in the enterprise:

- [Block all consumer Microsoft account user authentication](#)
- [Accounts: Block Microsoft accounts](#)

Block all consumer Microsoft account user authentication

This setting controls whether users can provide Microsoft accounts for authentication for applications or services.

If this setting is enabled, all applications and services on the device are prevented from using Microsoft accounts for authentication. This applies both to existing users of a device and new users who may be added.

However, any application or service that has already authenticated a user will not be affected by enabling this setting until the authentication cache expires. It is recommended to enable this setting before any user signs in to a device to prevent cached tokens from being present.

If this setting is disabled or not configured, applications and services can use Microsoft accounts for authentication. By default, this setting is **Disabled**.

This setting does not affect whether users can sign in to devices by using Microsoft accounts, or the ability for users to provide Microsoft accounts via the browser for authentication with web-based applications.

The path to this setting is:

Computer Configuration\Administrative Templates\Windows Components\Microsoft account

Accounts: Block Microsoft accounts

This setting prevents using the **Settings** app to add a Microsoft account for single sign-on (SSO) authentication for Microsoft services and some background services, or using a Microsoft account for single sign-on to other applications or services.

There are two options if this setting is enabled:

- **Users can't add Microsoft accounts** means that existing connected accounts can still sign in to the device (and appear on the Sign in screen). However, users cannot use the **Settings** app to add new connected accounts (or connect local accounts to Microsoft accounts).
- **Users can't add or log on with Microsoft accounts** means that users cannot add new connected accounts (or connect local accounts to Microsoft accounts) or use existing connected accounts through **Settings**.

This setting does not affect adding a Microsoft account for application authentication. For example, if this setting is enabled, a user can still provide a Microsoft account for authentication with an application such as **Mail**, but the user cannot use the Microsoft account for single sign-on authentication for other applications or services (in other words, the user will be prompted to authenticate for other applications or services).

By default, this setting is **Not defined**.

The path to this setting is:

Computer Configuration\Windows Settings\Security Settings\Local Policies\Security Options

Configure connected accounts

Users can connect a Microsoft account to their domain account and synchronize the settings and preferences between them. This enables users to see the same desktop background, app settings, browser history and favorites, and other Microsoft account settings on their other devices.

Users can disconnect a Microsoft account from their domain account at any time as follows: In **PC settings**, tap or click **Users**, tap or click **Disconnect**, and then tap or click **Finish**.

Note Connecting Microsoft accounts with domain accounts can limit access to some high-privileged tasks in Windows. For example, Task Scheduler will evaluate the connected Microsoft account for access and fail. In these situations, the account owner should disconnect the account.

Provision Microsoft accounts in the enterprise

Microsoft accounts are private user accounts. There are no methods provided by Microsoft to provision Microsoft accounts for an enterprise. Enterprises should use domain accounts.

Audit account activity

Because Microsoft accounts are Internet-based, Windows does not have a mechanism to audit their use until the account is associated with a domain account. But this association does not restrict the user from disconnecting the account or disjoining from the domain. It is not possible to audit the activity of accounts that are not associated with your domain.

Perform password resets

Only the owner of the Microsoft account can change the password. Passwords can be changed in the [Microsoft account sign-in portal](#).

Restrict app installation and usage

Within your organization, you can set application control policies to regulate app installation and usage for Microsoft accounts. For more information, see [AppLocker](#) and [Packaged Apps and Packaged App Installer Rules in AppLocker](#).

See also

- [Managing Privacy: Using a Microsoft Account to Logon and Resulting Internet Communication](#)
- [Access Control Overview](#)

Service Accounts

7/1/2022 • 6 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows Server 2016

This topic for the IT professional explains group and standalone managed service accounts, and the computer-specific virtual computer account, and it points to resources about these service accounts.

Overview

A service account is a user account that is created explicitly to provide a security context for services running on Windows Server operating systems. The security context determines the service's ability to access local and network resources. The Windows operating systems rely on services to run various features. These services can be configured through the applications, the Services snap-in, or Task Manager, or by using Windows PowerShell.

This topic contains information about the following types of service accounts:

- [Standalone managed service accounts](#)
- [Group-managed service accounts](#)
- [Virtual accounts](#)

Standalone managed service accounts

A managed service account is designed to isolate domain accounts in crucial applications, such as Internet Information Services (IIS), and eliminate the need for an administrator to manually administer the service principal name (SPN) and credentials for the accounts.

To use managed service accounts, the server on which the application or service is installed must be running at least Windows Server 2008 R2. One managed service account can be used for services on a single computer. Managed service accounts cannot be shared between multiple computers, and they cannot be used in server clusters where a service is replicated on multiple cluster nodes. For this scenario, you must use a group-managed service account. For more information, see [Group-Managed Service Accounts Overview](#).

In addition to the enhanced security that is provided by having individual accounts for critical services, there are four important administrative benefits associated with managed service accounts:

- You can create a class of domain accounts that can be used to manage and maintain services on local computers.
- Unlike domain accounts in which administrators must manually reset passwords, the network passwords for these accounts are automatically reset.
- You do not have to complete complex SPN management tasks to use managed service accounts.
- You don't have to complete complex SPN management tasks to use managed service accounts.
- Administrative tasks for managed service accounts can be delegated to non-administrators.

Software requirements

Managed service accounts apply to the Windows operating systems that are designated in the **Applies To** list at the beginning of this topic.

Group-managed service accounts

Group-managed service accounts are an extension of the standalone-managed service accounts, which were introduced in Windows Server 2008 R2. These accounts are managed domain accounts that provide automatic password management and simplified service principal name (SPN) management, including delegation of management to other administrators.

The group-managed service account provides the same functionality as a standalone managed service account within the domain, but it extends that functionality over multiple servers. When connecting to a service that is hosted on a server farm, such as Network Load Balancing, the authentication protocols that support mutual authentication require all instances of the services to use the same principal. When group-managed service accounts are used as service principals, the Windows Server operating system manages the password for the account instead of relying on the administrator to manage the password.

The Microsoft Key Distribution Service (kdssvc.dll) provides the mechanism to securely obtain the latest key or a specific key with a key identifier for an Active Directory account. This service was introduced in Windows Server 2012, and it does not run on previous versions of the Windows Server operating system. The Key Distribution Service shares a secret, which is used to create keys for the account. These keys are periodically changed. For a group-managed service account, the domain controller computes the password on the key that is provided by the Key Distribution Services, in addition to other attributes of the group-managed service account.

Practical applications

Group-managed service accounts provide a single identity solution for services running on a server farm, or on systems that use Network Load Balancing. By providing a group-managed service account solution, services can be configured for the group-managed service account principal, and the password management is handled by the operating system.

By using a group-managed service account, service administrators do not need to manage password synchronization between service instances. The group-managed service account supports hosts that are kept offline for an extended time period and the management of member hosts for all instances of a service. This provision means that you can deploy a server farm that supports a single identity to which existing client computers can authenticate without knowing the instance of the service to which they are connecting.

Failover clusters do not support group-managed service accounts. However, services that run on top of the Cluster service can use a group-managed service account or a standalone managed service account if they are a Windows service, an App pool, a scheduled task, or if they natively support group-managed service account or standalone managed service accounts.

Software requirements

Group-managed service accounts can only be configured and administered on computers running at least Windows Server 2012, but they can be deployed as a single service identity solution in domains that still have domain controllers running operating systems earlier than Windows Server 2012. There are no domain or forest functional level requirements.

A 64-bit architecture is required to run the Windows PowerShell commands that are used to administer group-managed service accounts.

A managed service account is dependent on encryption types supported by Kerberos. When a client computer authenticates to a server by using Kerberos protocol, the domain controller creates a Kerberos service ticket that is protected with encryption that the domain controller and the server support. The domain controller uses the account's **msDS-SupportedEncryptionTypes** attribute to determine what encryption the server supports, and if there is no attribute, it assumes that the client computer does not support stronger encryption types. The Advanced Encryption Standard (AES) must always be configured for managed service accounts. If computers that host the managed service account are configured to not support RC4, authentication will always fail.

Note Introduced in Windows Server 2008 R2, the Data Encryption Standard (DES) is disabled by default. For

more information about supported encryption types, see [Changes in Kerberos Authentication](#).

Group-managed service accounts are not applicable in Windows operating systems prior to Windows Server 2012.

Virtual accounts

Virtual accounts were introduced in Windows Server 2008 R2 and Windows 7, and are managed local accounts that provide the following features to simplify service administration:

- The virtual account is automatically managed.
- The virtual account can access the network in a domain environment.
- No password management is required. For example, if the default value is used for the service accounts during SQL Server setup on Windows Server 2008 R2, a virtual account that uses the instance name as the service name is established in the format NT SERVICE\<<SERVICENAME>.

Services that run as virtual accounts access network resources by using the credentials of the computer account in the format <domain_name>\<computer_name>\$.

For information about how to configure and use virtual service accounts, see [Service Accounts Step-by-Step Guide](#).

Software requirements

Virtual accounts apply to the Windows operating systems that are designated in the **Applies To** list at the beginning of this topic.

See also

The following table provides links to other resources that are related to standalone managed service accounts, group-managed service accounts, and virtual accounts.

CONTENT TYPE	REFERENCES
Product evaluation	What's New for Managed Service Accounts Getting Started with Group Managed Service Accounts
Deployment	Windows Server 2012: Group Managed Service Accounts - Ask Premier Field Engineering (PFE) Platforms - Site Home - TechNet Blogs
Related technologies	Security Principals What's new in Active Directory Domain Services

Active Directory Security Groups

7/1/2022 • 57 minutes to read • [Edit Online](#)

Applies to

- Windows Server 2016 or later
- Windows 10 or later

This reference topic for the IT professional describes the default Active Directory security groups.

There are two forms of common security principals in Active Directory: user accounts and computer accounts. These accounts represent a physical entity (a person or a computer). User accounts can also be used as dedicated service accounts for some applications. Security groups are used to collect user accounts, computer accounts, and other groups into manageable units.

In the Windows Server operating system, there are several built-in accounts and security groups that are preconfigured with the appropriate rights and permissions to perform specific tasks. For Active Directory, there are two types of administrative responsibilities:

- **Service administrators** Responsible for maintaining and delivering Active Directory Domain Services (AD DS), including managing domain controllers and configuring the AD DS.
- **Data administrators** Responsible for maintaining the data that is stored in AD DS and on domain member servers and workstations.

About Active Directory groups

Groups are used to collect user accounts, computer accounts, and other groups into manageable units. Working with groups instead of with individual users helps simplify network maintenance and administration.

There are two types of groups in Active Directory:

- **Distribution groups** Used to create email distribution lists.
- **Security groups** Used to assign permissions to shared resources.

Distribution groups

Distribution groups can be used only with email applications (such as Exchange Server) to send email to collections of users. Distribution groups are not security enabled, which means that they cannot be listed in discretionary access control lists (DACLS).

Security groups

Security groups can provide an efficient way to assign access to resources on your network. By using security groups, you can:

- Assign user rights to security groups in Active Directory.

User rights are assigned to a security group to determine what members of that group can do within the scope of a domain or forest. User rights are automatically assigned to some security groups when Active Directory is installed to help administrators define a person's administrative role in the domain.

For example, a user who is added to the Backup Operators group in Active Directory has the ability to back up and restore files and directories that are located on each domain controller in the domain. This is

possible because, by default, the user rights **Backup files and directories** and **Restore files and directories** are automatically assigned to the Backup Operators group. Therefore, members of this group inherit the user rights that are assigned to that group.

You can use Group Policy to assign user rights to security groups to delegate specific tasks. For more information about using Group Policy, see [User Rights Assignment](#).

- Assign permissions to security groups for resources.

Permissions are different than user rights. Permissions are assigned to the security group for the shared resource. Permissions determine who can access the resource and the level of access, such as Full Control. Some permissions that are set on domain objects are automatically assigned to allow various levels of access to default security groups, such as the Account Operators group or the Domain Admins group.

Security groups are listed in DACLs that define permissions on resources and objects. When assigning permissions for resources (file shares, printers, and so on), administrators should assign those permissions to a security group rather than to individual users. The permissions are assigned once to the group, instead of several times to each individual user. Each account that is added to a group receives the rights that are assigned to that group in Active Directory, and the user receives the permissions that are defined for that group.

Like distribution groups, security groups can be used as an email entity. Sending an email message to the group sends the message to all the members of the group.

Group scope

Groups are characterized by a scope that identifies the extent to which the group is applied in the domain tree or forest. The scope of the group defines where the group can be granted permissions. The following three group scopes are defined by Active Directory:

- Universal
- Global
- Domain Local

NOTE

In addition to these three scopes, the default groups in the **Builtin** container have a group scope of Builtin Local. This group scope and group type cannot be changed.

The following table lists the three group scopes and more information about each scope for a security group.

Group scopes

SCOPE	POSSIBLE MEMBERS	SCOPE CONVERSION	CAN GRANT PERMISSIONS	POSSIBLE MEMBER OF
-------	------------------	------------------	-----------------------	--------------------

SCOPE	POSSIBLE MEMBERS	SCOPE CONVERSION	CAN GRANT PERMISSIONS	POSSIBLE MEMBER OF
Universal	<p>Accounts from any domain in the same forest</p> <p>Global groups from any domain in the same forest</p> <p>Other Universal groups from any domain in the same forest</p>	<p>Can be converted to Domain Local scope if the group is not a member of any other Universal groups</p> <p>Can be converted to Global scope if the group does not contain any other Universal groups</p>	On any domain in the same forest or trusting forests	<p>Other Universal groups in the same forest</p> <p>Domain</p> <p>Local groups in the same forest or trusting forests</p> <p>Local groups on computers in the same forest or trusting forests</p>
Global	<p>Accounts from the same domain</p> <p>Other Global groups from the same domain</p>	Can be converted to Universal scope if the group is not a member of any other global group	On any domain in the same forest, or trusting domains or forests	<p>Universal groups from any domain in the same forest</p> <p>Other Global groups from the same domain</p> <p>Domain Local groups from any domain in the same forest, or from any trusting domain</p>
Domain Local	<p>Accounts from any domain or any trusted domain</p> <p>Global groups from any domain or any trusted domain</p> <p>Universal groups from any domain in the same forest</p> <p>Other Domain Local groups from the same domain</p> <p>Accounts, Global groups, and Universal groups from other forests and from external domains</p>	Can be converted to Universal scope if the group does not contain any other Domain Local groups	Within the same domain	<p>Other Domain Local groups from the same domain</p> <p>Local groups on computers in the same domain, excluding built-in groups that have well-known SIDs</p>

Special identity groups

Special identities are generally referred to as groups. Special identity groups do not have specific memberships that can be modified, but they can represent different users at different times, depending on the circumstances.

Some of these groups include Creator Owner, Batch, and Authenticated User.

For information about all the special identity groups, see [Special Identities](#).

Default security groups

Default groups, such as the Domain Admins group, are security groups that are created automatically when you create an Active Directory domain. You can use these predefined groups to help control access to shared resources and to delegate specific domain-wide administrative roles.

Many default groups are automatically assigned a set of user rights that authorize members of the group to perform specific actions in a domain, such as logging on to a local system or backing up files and folders. For example, a member of the Backup Operators group has the right to perform backup operations for all domain controllers in the domain.

When you add a user to a group, the user receives all the user rights that are assigned to the group and all the permissions that are assigned to the group for any shared resources.

Default groups are located in the **Builtin** container and in the **Users** container in Active Directory Users and Computers. The **Builtin** container includes groups that are defined with the Domain Local scope. The **Users** container includes groups that are defined with Global scope and groups that are defined with Domain Local scope. You can move groups that are located in these containers to other groups or organizational units (OU) within the domain, but you cannot move them to other domains.

Some of the administrative groups that are listed in this topic and all members of these groups are protected by a background process that periodically checks for and applies a specific security descriptor. This descriptor is a data structure that contains security information associated with a protected object. This process ensures that any successful unauthorized attempt to modify the security descriptor on one of the administrative accounts or groups will be overwritten with the protected settings.

The security descriptor is present on the **AdminSDHolder** object. This means that if you want to modify the permissions on one of the service administrator groups or on any of its member accounts, you must modify the security descriptor on the **AdminSDHolder** object so that it will be applied consistently. Be careful when you make these modifications because you are also changing the default settings that will be applied to all of your protected administrative accounts.

Active Directory default security groups by operating system version

The following tables provide descriptions of the default groups that are located in the **Builtin** and **Users** containers in each operating system.

DEFAULT SECURITY GROUP	WINDOWS SERVER 2016	WINDOWS SERVER 2012 R2	WINDOWS SERVER 2012	WINDOWS SERVER 2008 R2
Access Control Assistance Operators	Yes	Yes	Yes	
Account Operators	Yes	Yes	Yes	Yes
Administrators	Yes	Yes	Yes	Yes
Allowed RODC Password Replication Group	Yes	Yes	Yes	Yes
Backup Operators	Yes	Yes	Yes	Yes

DEFAULT SECURITY GROUP	WINDOWS SERVER 2016	WINDOWS SERVER 2012 R2	WINDOWS SERVER 2012	WINDOWS SERVER 2008 R2
Certificate Service DCOM Access	Yes	Yes	Yes	Yes
Cert Publishers	Yes	Yes	Yes	Yes
Cloneable Domain Controllers	Yes	Yes	Yes	
Cryptographic Operators	Yes	Yes	Yes	Yes
Denied RODC Password Replication Group	Yes	Yes	Yes	Yes
Device Owners	Yes	Yes	Yes	Yes
Distributed COM Users	Yes	Yes	Yes	Yes
DnsUpdateProxy	Yes	Yes	Yes	Yes
DnsAdmins	Yes	Yes	Yes	Yes
Domain Admins	Yes	Yes	Yes	Yes
Domain Computers	Yes	Yes	Yes	Yes
Domain Controllers	Yes	Yes	Yes	Yes
Domain Guests	Yes	Yes	Yes	Yes
Domain Users	Yes	Yes	Yes	Yes
Enterprise Admins	Yes	Yes	Yes	Yes
Enterprise Key Admins	Yes			
Enterprise Read-only Domain Controllers	Yes	Yes	Yes	Yes
Event Log Readers	Yes	Yes	Yes	Yes
Group Policy Creator Owners	Yes	Yes	Yes	Yes
Guests	Yes	Yes	Yes	Yes
Hyper-V Administrators	Yes	Yes	Yes	

DEFAULT SECURITY GROUP	WINDOWS SERVER 2016	WINDOWS SERVER 2012 R2	WINDOWS SERVER 2012	WINDOWS SERVER 2008 R2
IIS_IUSRS	Yes	Yes	Yes	Yes
Incoming Forest Trust Builders	Yes	Yes	Yes	Yes
Key Admins	Yes			
Network Configuration Operators	Yes	Yes	Yes	Yes
Performance Log Users	Yes	Yes	Yes	Yes
Performance Monitor Users	Yes	Yes	Yes	Yes
Pre-Windows 2000 Compatible Access	Yes	Yes	Yes	Yes
Print Operators	Yes	Yes	Yes	Yes
Protected Users	Yes	Yes		
RAS and IAS Servers	Yes	Yes	Yes	Yes
RDS Endpoint Servers	Yes	Yes	Yes	
RDS Management Servers	Yes	Yes	Yes	
RDS Remote Access Servers	Yes	Yes	Yes	
Read-only Domain Controllers	Yes	Yes	Yes	Yes
Remote Desktop Users	Yes	Yes	Yes	Yes
Remote Management Users	Yes	Yes	Yes	
Replicator	Yes	Yes	Yes	Yes
Schema Admins	Yes	Yes	Yes	Yes
Server Operators	Yes	Yes	Yes	Yes
Storage Replica Administrators	Yes			

DEFAULT SECURITY GROUP	WINDOWS SERVER 2016	WINDOWS SERVER 2012 R2	WINDOWS SERVER 2012	WINDOWS SERVER 2008 R2
System Managed Accounts Group	Yes			
Terminal Server License Servers	Yes	Yes	Yes	Yes
Users	Yes	Yes	Yes	Yes
Windows Authorization Access Group	Yes	Yes	Yes	Yes
WinRMRemoteWMIUsers_		Yes	Yes	

Access Control Assistance Operators

Members of this group can remotely query authorization attributes and permissions for resources on the computer.

The Access Control Assistance Operators group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups](#) table.

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-579
Type	Builtin Local
Default container	CN=BuiltIn, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	
Default User Rights	None

Account Operators

The Account Operators group grants limited account creation privileges to a user. Members of this group can create and modify most types of accounts, including those of users, local groups, and global groups, and members can log in locally to domain controllers.

Members of the Account Operators group cannot manage the Administrator user account, the user accounts of administrators, or the [Administrators](#), [Server Operators](#), [Account Operators](#), [Backup Operators](#), or [Print](#)

[Operators](#) groups. Members of this group cannot modify user rights.

The Account Operators group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

NOTE

By default, this built-in group has no members, and it can create and manage users and groups in the domain, including its own membership and that of the Server Operators group. This group is considered a service administrator group because it can modify Server Operators, which in turn can modify domain controller settings. As a best practice, leave the membership of this group empty, and do not use it for any delegated administration. This group cannot be renamed, deleted, or moved.

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-548
Type	Builtin Local
Default container	CN=BuiltIn, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	Yes
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	No
Default User Rights	Allow log on locally : SeInteractiveLogonRight

Administrators

Members of the Administrators group have complete and unrestricted access to the computer, or if the computer is promoted to a domain controller, members have unrestricted access to the domain.

The Administrators group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

NOTE

The Administrators group has built-in capabilities that give its members full control over the system. This group cannot be renamed, deleted, or moved. This built-in group controls access to all the domain controllers in its domain, and it can change the membership of all administrative groups.

Membership can be modified by members of the following groups: the default service Administrators, Domain Admins in the domain, or Enterprise Admins. This group has the special privilege to take ownership of any object in the directory or any resource on a domain controller. This account is considered a service administrator group because its members have full access to the domain controllers in the domain.

This security group includes the following changes since Windows Server 2008:

- Default user rights changes: **Allow log on through Terminal Services** existed in Windows Server 2008, and it was replaced by [Allow log on through Remote Desktop Services](#).
- [Remove computer from docking station](#) was removed in Windows Server 2012 R2.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-544
Type	Builtin Local
Default container	CN=BuiltIn, DC= <domain>, DC=
Default members	Administrator, Domain Admins, Enterprise Admins
Default member of	None
Protected by ADMINSDHOLDER?	Yes
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	No

ATTRIBUTE	VALUE
Default User Rights	<p>Adjust memory quotas for a process: SeIncreaseQuotaPrivilege</p> <p>Access this computer from the network: SeNetworkLogonRight</p> <p>Allow log on locally: SeInteractiveLogonRight</p> <p>Allow log on through Remote Desktop Services: SeRemoteInteractiveLogonRight</p> <p>Back up files and directories: SeBackupPrivilege</p> <p>Bypass traverse checking: SeChangeNotifyPrivilege</p> <p>Change the system time: SeSystemTimePrivilege</p> <p>Change the time zone: SeTimeZonePrivilege</p> <p>Create a pagefile: SeCreatePagefilePrivilege</p> <p>Create global objects: SeCreateGlobalPrivilege</p> <p>Create symbolic links: SeCreateSymbolicLinkPrivilege</p> <p>Debug programs: SeDebugPrivilege</p> <p>Enable computer and user accounts to be trusted for delegation: SeEnableDelegationPrivilege</p> <p>Force shutdown from a remote system: SeRemoteShutdownPrivilege</p> <p>Impersonate a client after authentication: SeImpersonatePrivilege</p> <p>Increase scheduling priority: SeIncreaseBasePriorityPrivilege</p> <p>Load and unload device drivers: SeLoadDriverPrivilege</p> <p>Log on as a batch job: SeBatchLogonRight</p> <p>Manage auditing and security log: SeSecurityPrivilege</p> <p>Modify firmware environment values: SeSystemEnvironmentPrivilege</p> <p>Perform volume maintenance tasks: SeManageVolumePrivilege</p> <p>Profile system performance: SeSystemProfilePrivilege</p> <p>Profile single process: SeProfileSingleProcessPrivilege</p> <p>Remove computer from docking station: SeUndockPrivilege</p> <p>Restore files and directories: SeRestorePrivilege</p> <p>Shut down the system: SeShutdownPrivilege</p> <p>Take ownership of files or other objects: SeTakeOwnershipPrivilege</p>

Allowed RODC Password Replication Group

The purpose of this security group is to manage a RODC password replication policy. This group has no members by default, and it results in the condition that new Read-only domain controllers do not cache user credentials. The [Denied RODC Password Replication Group](#) group contains a variety of high-privilege accounts

and security groups. The Denied RODC Password Replication group supersedes the Allowed RODC Password Replication group.

The Allowed RODC Password Replication group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-21-<domain>-571
Type	Domain local
Default container	CN=Users DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	
Default User Rights	None

Backup Operators

Members of the Backup Operators group can back up and restore all files on a computer, regardless of the permissions that protect those files. Backup Operators also can log on to and shut down the computer. This group cannot be renamed, deleted, or moved. By default, this built-in group has no members, and it can perform backup and restore operations on domain controllers. Its membership can be modified by the following groups: default service Administrators, Domain Admins in the domain, or Enterprise Admins. It cannot modify the membership of any administrative groups. While members of this group cannot change server settings or modify the configuration of the directory, they do have the permissions needed to replace files (including operating system files) on domain controllers. Because of this, members of this group are considered service administrators.

The Backup Operators group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-551
Type	Builtin Local
Default container	CN=BuiltIn, DC= <domain>, DC=
Default members	None

ATTRIBUTE	VALUE
Default member of	None
Protected by ADMINSDHOLDER?	Yes
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	No
Default User Rights	Allow log on locally : SeInteractiveLogonRight Back up files and directories : SeBackupPrivilege Log on as a batch job : SeBatchLogonRight Restore files and directories : SeRestorePrivilege Shut down the system : SeShutdownPrivilege

Certificate Service DCOM Access

Members of this group are allowed to connect to certification authorities in the enterprise.

The Certificate Service DCOM Access group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-<domain>-574
Type	Domain Local
Default container	CN=Builtin, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	
Default User Rights	None

Cert Publishers

Members of the Cert Publishers group are authorized to publish certificates for User objects in Active Directory.

The Cert Publishers group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-21-<domain>-517
Type	Domain Local
Default container	CN=Users, DC= <domain>, DC=
Default members	None
Default member of	Denied RODC Password Replication Group
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	No
Default User Rights	None

Cloneable Domain Controllers

Members of the Cloneable Domain Controllers group that are domain controllers may be cloned. In Windows Server 2012 R2 and Windows Server 2012, you can deploy domain controllers by copying an existing virtual domain controller. In a virtual environment, you no longer have to repeatedly deploy a server image that is prepared by using sysprep.exe, promote the server to a domain controller, and then complete additional configuration requirements for deploying each domain controller (including adding the virtual domain controller to this security group).

For more information, see [Introduction to Active Directory Domain Services \(AD DS\) Virtualization \(Level 100\)](#).

This security group was introduced in Windows Server 2012, and it has not changed in subsequent versions.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-21-<domain>-522
Type	Global
Default container	CN=Users, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	
Default User Rights	None

Cryptographic Operators

Members of this group are authorized to perform cryptographic operations. This security group was added in Windows Vista Service Pack 1 (SP1) to configure Windows Firewall for IPsec in Common Criteria mode.

The Cryptographic Operators group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group was introduced in Windows Vista Service Pack 1, and it has not changed in subsequent versions.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-569
Type	Builtin Local
Default container	CN=Builtin, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	
Default User Rights	None

Denied RODC Password Replication Group

Members of the Denied RODC Password Replication group cannot have their passwords replicated to any Read-only domain controller.

The purpose of this security group is to manage a RODC password replication policy. This group contains a variety of high-privilege accounts and security groups. The Denied RODC Password Replication Group supersedes the [Allowed RODC Password Replication Group](#).

This security group includes the following changes since Windows Server 2008:

- Windows Server 2012 changed the default members to include [Cert Publishers](#).

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-21- <domain> -572
Type	Domain local
Default container	CN=Users, DC= <domain>, DC=

ATTRIBUTE	VALUE
Default members	Cert Publishers Domain Admins Domain Controllers Enterprise Admins Group Policy Creator Owners Read-only Domain Controllers Schema Admins
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	
Safe to delegate management of this group to non-Service admins?	
Default User Rights	None

Device Owners

This group is not currently used in Windows.

Microsoft does not recommend changing the default configuration where this security group has zero members. Changing the default configuration could hinder future scenarios that rely on this group.

The Device Owners group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-583
Type	Builtin Local
Default container	CN=BuiltIn, DC=<domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Can be moved out but it is not recommended
Safe to delegate management of this group to non-Service admins?	No

ATTRIBUTE	VALUE
Default User Rights	<p>Allow log on locally: SeInteractiveLogonRight</p> <p>Access this computer from the network: SeNetworkLogonRight</p> <p>Bypass traverse checking: SeChangeNotifyPrivilege</p> <p>Change the time zone: SeTimeZonePrivilege</p>

Distributed COM Users

Members of the Distributed COM Users group are allowed to launch, activate, and use Distributed COM objects on the computer. Microsoft Component Object Model (COM) is a platform-independent, distributed, object-oriented system for creating binary software components that can interact. Distributed Component Object Model (DCOM) allows applications to be distributed across locations that make the most sense to you and to the application. This group appears as a SID until the domain controller is made the primary domain controller and it holds the operations master role (also known as flexible single master operations or FSMO).

The Distributed COM Users group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-562
Type	Builtin Local
Default container	CN=Builtin, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	
Default User Rights	None

DnsUpdateProxy

Members of the DnsUpdateProxy group are DNS clients. They are permitted to perform dynamic updates on behalf of other clients (such as DHCP servers). A DNS server can develop stale resource records when a DHCP server is configured to dynamically register host (A) and pointer (PTR) resource records on behalf of DHCP clients by using dynamic update. Adding clients to this security group mitigates this scenario.

However, to protect against unsecured records or to permit members of the DnsUpdateProxy group to register records in zones that allow only secured dynamic updates, you must create a dedicated user account and configure DHCP servers to perform DNS dynamic updates by using the credentials of this account (user name, password, and domain). Multiple DHCP servers can use the credentials of one dedicated user account. This

group exists only if the DNS server role is or was once installed on a domain controller in the domain.

For information, see [DNS Record Ownership and the DnsUpdateProxy Group](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-21- <domain> - <variable RI>
Type	Global
Default container	CN=Users, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Yes
Safe to delegate management of this group to non-Service admins?	
Default User Rights	None

DnsAdmins

Members of DNSAdmins group have access to network DNS information. The default permissions are as follows: Allow: Read, Write, Create All Child objects, Delete Child objects, Special Permissions. This group exists only if the DNS server role is or was once installed on a domain controller in the domain.

For more information about security and DNS, see [DNSSEC in Windows Server 2012](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-21- <domain> - <variable RI>
Type	Builtin Local
Default container	CN=Users, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Yes
Safe to delegate management of this group to non-Service admins?	

ATTRIBUTE	VALUE
Default User Rights	None

Domain Admins

Members of the Domain Admins security group are authorized to administer the domain. By default, the Domain Admins group is a member of the Administrators group on all computers that have joined a domain, including the domain controllers. The Domain Admins group is the default owner of any object that is created in Active Directory for the domain by any member of the group. If members of the group create other objects, such as files, the default owner is the Administrators group.

The Domain Admins group controls access to all domain controllers in a domain, and it can modify the membership of all administrative accounts in the domain. Membership can be modified by members of the service administrator groups in its domain (Administrators and Domain Admins), and by members of the Enterprise Admins group. This is considered a service administrator account because its members have full access to the domain controllers in a domain.

The Domain Admins group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-21-<domain>-512
Type	Global
Default container	CN=Users, DC= <domain>, DC=
Default members	Administrator
Default member of	Administrators Denied RODC Password ReplicationGroup
Protected by ADMINSDHOLDER?	Yes
Safe to move out of default container?	Yes
Safe to delegate management of this group to non-Service admins?	No
Default User Rights	See Administrators See Denied RODC Password Replication Group

Domain Computers

This group can include all computers and servers that have joined the domain, excluding domain controllers. By default, any computer account that is created automatically becomes a member of this group.

The Domain Computers group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-21-<domain>-515
Type	Global
Default container	CN=Users, DC=<domain>, DC=
Default members	All computers joined to the domain, excluding domain controllers
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Yes (but not required)
Safe to delegate management of this group to non-Service admins?	Yes
Default User Rights	None

Domain Controllers

The Domain Controllers group can include all domain controllers in the domain. New domain controllers are automatically added to this group.

The Domain Controllers group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-21-<domain>-516
Type	Global
Default container	CN=Users, DC=<domain>, DC=
Default members	Computer accounts for all domain controllers of the domain
Default member of	Denied RODC Password Replication Group
Protected by ADMINSDHOLDER?	Yes
Safe to move out of default container?	No
Safe to delegate management of this group to non-Service admins?	No
Default User Rights	None

Domain Guests

The Domain Guests group includes the domain's built-in Guest account. When members of this group sign in as

local guests on a domain-joined computer, a domain profile is created on the local computer.

The Domain Guests group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-21-<domain>-514
Type	Global
Default container	CN=Users, DC= <domain>, DC=
Default members	Guest
Default member of	Guests
Protected by ADMINSDHOLDER?	Yes
Safe to move out of default container?	Can be moved out but it is not recommended
Safe to delegate management of this group to non-Service admins?	No
Default User Rights	See Guests

Domain Users

The Domain Users group includes all user accounts in a domain. When you create a user account in a domain, it is automatically added to this group.

By default, any user account that is created in the domain automatically becomes a member of this group. This group can be used to represent all users in the domain. For example, if you want all domain users to have access to a printer, you can assign permissions for the printer to this group (or add the Domain Users group to a local group on the print server that has permissions for the printer).

The Domain Users group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-21-<domain>-513
Type	Global
Default container	CN=Users, DC= <domain>, DC=
Default members	Administrator
krbtgt	

ATTRIBUTE	VALUE
Default member of	Users
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Yes
Safe to delegate management of this group to non-Service admins?	No
Default User Rights	See Users

Enterprise Admins

The Enterprise Admins group exists only in the root domain of an Active Directory forest of domains. It is a Universal group if the domain is in native mode; it is a Global group if the domain is in mixed mode. Members of this group are authorized to make forest-wide changes in Active Directory, such as adding child domains.

By default, the only member of the group is the Administrator account for the forest root domain. This group is automatically added to the Administrators group in every domain in the forest, and it provides complete access for configuring all domain controllers. Members in this group can modify the membership of all administrative groups. Membership can be modified only by the default service administrator groups in the root domain. This is considered a service administrator account.

The Enterprise Admins group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-21-<root domain>-519
Type	Universal (if Domain is in Native-Mode) else Global
Default container	CN=Users, DC= <domain>, DC=
Default members	Administrator
Default member of	Administrators
Denied RODC Password Replication Group	
Protected by ADMINSDHOLDER?	Yes
Safe to move out of default container?	Yes
Safe to delegate management of this group to non-Service admins?	No
Default User Rights	See Administrators See Denied RODC Password Replication Group

Enterprise Key Admins

Members of this group can perform administrative actions on key objects within the forest.

The Enterprise Key Admins group was introduced in Windows Server 2016.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-21-<domain>-527
Type	Global
Default container	CN=Users, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	Yes
Safe to move out of default container?	Yes
Safe to delegate management of this group to non-Service admins?	No
Default User Rights	None

Enterprise Read-Only Domain Controllers

Members of this group are Read-Only Domain Controllers in the enterprise. Except for account passwords, a Read-only domain controller holds all the Active Directory objects and attributes that a writable domain controller holds. However, changes cannot be made to the database that is stored on the Read-only domain controller. Changes must be made on a writable domain controller and then replicated to the Read-only domain controller.

Read-only domain controllers address some of the issues that are commonly found in branch offices. These locations might not have a domain controller. Or, they might have a writable domain controller, but not the physical security, network bandwidth, or local expertise to support it.

For more information, see [What Is an RODC?](#).

The Enterprise Read-Only Domain Controllers group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-21-<root domain>-498
Type	Universal
Default container	CN=Users, DC= <domain>, DC=
Default members	None

ATTRIBUTE	VALUE
Default member of	None
Protected by ADMINSDHOLDER?	Yes
Safe to move out of default container?	
Safe to delegate management of this group to non-Service admins?	
Default User Rights	None

Event Log Readers

Members of this group can read event logs from local computers. The group is created when the server is promoted to a domain controller.

The Event Log Readers group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-573
Type	Domain Local
Default container	CN=Users, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	
Default User Rights	None

Group Policy Creator Owners

This group is authorized to create, edit, or delete Group Policy Objects in the domain. By default, the only member of the group is Administrator.

For information about other features you can use with this security group, see [Group Policy Overview](#).

The Group Policy Creator Owners group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-21-<domain>-520
Type	Global
Default container	CN=Users, DC=<domain>, DC=
Default members	Administrator
Default member of	Denied RODC Password Replication Group
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	No
Safe to delegate management of this group to non-Service admins?	No
Default User Rights	See Denied RODC Password Replication Group

Guests

Members of the Guests group have the same access as members of the Users group by default, except that the Guest account has further restrictions. By default, the only member is the Guest account. The Guests group allows occasional or one-time users to sign in with limited privileges to a computer's built-in Guest account.

When a member of the Guests group signs out, the entire profile is deleted. This includes everything that is stored in the **%userprofile%** directory, including the user's registry hive information, custom desktop icons, and other user-specific settings. This implies that a guest must use a temporary profile to sign in to the system. This security group interacts with the Group Policy setting **Do not logon users with temporary profiles** when it is enabled. This setting is located under the following path:

Computer Configuration\Administrative Templates\System\User Profiles

NOTE

A Guest account is a default member of the Guests security group. People who do not have an actual account in the domain can use the Guest account. A user whose account is disabled (but not deleted) can also use the Guest account.

The Guest account does not require a password. You can set rights and permissions for the Guest account as in any user account. By default, the Guest account is a member of the built-in Guests group and the Domain Guests global group, which allows a user to sign in to a domain. The Guest account is disabled by default, and we recommend that it stay disabled.

The Guests group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-546

ATTRIBUTE	VALUE
Type	Builtin Local
Default container	CN=BuiltIn, DC= <domain>, DC=
Default members	Domain Guests
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	No
Default User Rights	None

Hyper-V Administrators

Members of the Hyper-V Administrators group have complete and unrestricted access to all the features in Hyper-V. Adding members to this group helps reduce the number of members required in the Administrators group, and further separates access.

NOTE

Prior to Windows Server 2012, access to features in Hyper-V was controlled in part by membership in the Administrators group.

This security group was introduced in Windows Server 2012, and it has not changed in subsequent versions.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-578
Type	Builtin Local
Default container	CN=BuiltIn, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	
Default User Rights	None

IIS_IUSRS

IIS_IUSRS is a built-in group that is used by Internet Information Services beginning with IIS 7.0. A built-in account and group are guaranteed by the operating system to always have a unique SID. IIS 7.0 replaces the IUSR_MachineName account and the IIS_WPG group with the IIS_IUSRS group to ensure that the actual names that are used by the new account and group will never be localized. For example, regardless of the language of the Windows operating system that you install, the IIS account name will always be IUSR, and the group name will be IIS_IUSRS.

For more information, see [Understanding Built-In User and Group Accounts in IIS 7](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-568
Type	Builtin Local
Default container	CN=BuiltIn, DC= <domain>, DC=
Default members	IUSR
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	
Safe to delegate management of this group to non-Service admins?	
Default User Rights	None

Incoming Forest Trust Builders

Members of the Incoming Forest Trust Builders group can create incoming, one-way trusts to this forest. Active Directory provides security across multiple domains or forests through domain and forest trust relationships. Before authentication can occur across trusts, Windows must determine whether the domain being requested by a user, computer, or service has a trust relationship with the logon domain of the requesting account.

To make this determination, the Windows security system computes a trust path between the domain controller for the server that receives the request and a domain controller in the domain of the requesting account. A secured channel extends to other Active Directory domains through interdomain trust relationships. This secured channel is used to obtain and verify security information, including security identifiers (SIDs) for users and groups.

NOTE

This group appears as a SID until the domain controller is made the primary domain controller and it holds the operations master role (also known as flexible single master operations or FSMO).

For more information, see [How Domain and Forest Trusts Work: Domain and Forest Trusts](#).

The Incoming Forest Trust Builders group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

NOTE

This group cannot be renamed, deleted, or moved.

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-557
Type	Builtin Local
Default container	CN=Builtin, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	No
Default User Rights	None

Key Admins

Members of this group can perform administrative actions on key objects within the domain.

The Key Admins group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-21- <domain> -526
Type	Global
Default container	CN=Users, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	Yes
Safe to move out of default container?	Yes
Safe to delegate management of this group to non-Service admins?	No

ATTRIBUTE	VALUE
Default User Rights	None

Network Configuration Operators

Members of the Network Configuration Operators group can have the following administrative privileges to manage configuration of networking features:

- Modify the Transmission Control Protocol/Internet Protocol (TCP/IP) properties for a local area network (LAN) connection, which includes the IP address, the subnet mask, the default gateway, and the name servers.
- Rename the LAN connections or remote access connections that are available to all the users.
- Enable or disable a LAN connection.
- Modify the properties of all of remote access connections of users.
- Delete all the remote access connections of users.
- Rename all the remote access connections of users.
- Issue `ipconfig`, `ipconfig /release`, or `ipconfig /renew` commands.
- Enter the PIN unblock key (PUK) for mobile broadband devices that support a SIM card.

NOTE

This group appears as a SID until the domain controller is made the primary domain controller and it holds the operations master role (also known as flexible single master operations or FSMO).

The Network Configuration Operators group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

NOTE

This group cannot be renamed, deleted, or moved.

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-556
Type	Builtin Local
Default container	CN=Builtin, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Cannot be moved

ATTRIBUTE	VALUE
Safe to delegate management of this group to non-Service admins?	Yes
Default User Rights	None

Performance Log Users

Members of the Performance Log Users group can manage performance counters, logs, and alerts locally on the server and from remote clients without being a member of the Administrators group. Specifically, members of this security group:

- Can use all the features that are available to the Performance Monitor Users group.
- Can create and modify Data Collector Sets after the group is assigned the [Log on as a batch job](#) user right.

WARNING

If you are a member of the Performance Log Users group, you must configure Data Collector Sets that you create to run under your credentials.

NOTE

In Windows Server 2016 or later, Data Collector Sets cannot be created by a member of the Performance Log Users group. If a member of the Performance Log Users group tries to create Data Collector Sets, they cannot complete creation because access will be denied.

- Cannot use the Windows Kernel Trace event provider in Data Collector Sets.

For members of the Performance Log Users group to initiate data logging or modify Data Collector Sets, the group must first be assigned the [Log on as a batch job](#) user right. To assign this user right, use the Local Security Policy snap-in in Microsoft Management Console.

NOTE

This group appears as a SID until the domain controller is made the primary domain controller and it holds the operations master role (also known as flexible single master operations or FSMO).

The Performance Log Users group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

NOTE

This account cannot be renamed, deleted, or moved.

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-559
Type	Builtin Local
Default container	CN=Builtin, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	Yes
Default User Rights	Log on as a batch job : SeBatchLogonRight

Performance Monitor Users

Members of this group can monitor performance counters on domain controllers in the domain, locally and from remote clients, without being a member of the Administrators or Performance Log Users groups. The Windows Performance Monitor is a Microsoft Management Console (MMC) snap-in that provides tools for analyzing system performance. From a single console, you can monitor application and hardware performance, customize what data you want to collect in logs, define thresholds for alerts and automatic actions, generate reports, and view past performance data in a variety of ways.

Specifically, members of this security group:

- Can use all the features that are available to the Users group.
- Can view real-time performance data in Performance Monitor.
 - Can change the Performance Monitor display properties while viewing data.
- Cannot create or modify Data Collector Sets.

WARNING

You cannot configure a Data Collector Set to run as a member of the Performance Monitor Users group.

NOTE

This group appears as a SID until the domain controller is made the primary domain controller and it holds the operations master role (also known as flexible single master operations or FSMO). This group cannot be renamed, deleted, or moved.

The Performance Monitor Users group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-558
Type	Builtin Local
Default container	CN=Builtin, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	Yes
Default User Rights	None

Pre-Windows 2000 Compatible Access

Members of the Pre-Windows 2000 Compatible Access group have Read access for all users and groups in the domain. This group is provided for backward compatibility for computers running Windows NT 4.0 and earlier. By default, the special identity group, Everyone, is a member of this group. Add users to this group only if they are running Windows NT 4.0 or earlier.

WARNING

This group appears as a SID until the domain controller is made the primary domain controller and it holds the operations master role (also known as flexible single master operations or FSMO).

The Pre-Windows 2000 Compatible Access group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-554
Type	Builtin Local
Default container	CN=Builtin, DC= <domain>, DC=
Default members	If you choose the Pre-Windows 2000 Compatible Permissions mode, Everyone and Anonymous are members, and if you choose the Windows 2000-only permissions mode, Authenticated Users are members.
Default member of	None
Protected by ADMINSDHOLDER?	No

ATTRIBUTE	VALUE
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	No
Default User Rights	Access this computer from the network: SeNetworkLogonRight Bypass traverse checking: SeChangeNotifyPrivilege

Print Operators

Members of this group can manage, create, share, and delete printers that are connected to domain controllers in the domain. They can also manage Active Directory printer objects in the domain. Members of this group can locally sign in to and shut down domain controllers in the domain.

This group has no default members. Because members of this group can load and unload device drivers on all domain controllers in the domain, add users with caution. This group cannot be renamed, deleted, or moved.

The Print Operators group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group has not changed since Windows Server 2008. However, in Windows Server 2008 R2, functionality was added to manage print administration. For more information, see [Assign Delegated Print Administrator and Printer Permission Settings in Windows Server 2012](#).

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-550
Type	Builtin Local
Default container	CN=Builtin, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	Yes
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	No
Default User Rights	Allow log on locally: SeInteractiveLogonRight Load and unload device drivers: SeLoadDriverPrivilege Shut down the system: SeShutdownPrivilege

Protected Users

Members of the Protected Users group are afforded additional protection against the compromise of credentials during authentication processes.

This security group is designed as part of a strategy to effectively protect and manage credentials within the enterprise. Members of this group automatically have non-configurable protection applied to their accounts. Membership in the Protected Users group is meant to be restrictive and proactively secure by default. The only method to modify the protection for an account is to remove the account from the security group.

This domain-related, global group triggers non-configurable protection on devices and host computers, starting with the Windows Server 2012 R2 and Windows 8.1 operating systems. It also triggers non-configurable protection on domain controllers in domains with a primary domain controller running Windows Server 2012 R2 or Windows Server 2016. This greatly reduces the memory footprint of credentials when users sign in to computers on the network from a non-compromised computer.

Depending on the account's domain functional level, members of the Protected Users group are further protected due to behavior changes in the authentication methods that are supported in Windows.

- Members of the Protected Users group cannot authenticate by using the following Security Support Providers (SSPs): NTLM, Digest Authentication, or CredSSP. Passwords are not cached on a device running Windows 8.1 or Windows 10, so the device fails to authenticate to a domain when the account is a member of the Protected User group.
- The Kerberos protocol will not use the weaker DES or RC4 encryption types in the preauthentication process. This means that the domain must be configured to support at least the AES cipher suite.
- The user's account cannot be delegated with Kerberos constrained or unconstrained delegation. This means that former connections to other systems may fail if the user is a member of the Protected Users group.
- The default Kerberos ticket-granting tickets (TGTs) lifetime setting of four hours is configurable by using Authentication Policies and Silos, which can be accessed through the Active Directory Administrative Center. This means that when four hours has passed, the user must authenticate again.

The Protected Users group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This group was introduced in Windows Server 2012 R2. For more information about how this group works, see [Protected Users Security Group](#).

The following table specifies the properties of the Protected Users group.

ATTRIBUTE	VALUE
Well-known SID/RID	S-1-5-21-<domain>-525
Type	Global
Default container	CN=Users, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Yes
Safe to delegate management of this group to non-service admins?	No

ATTRIBUTE	VALUE
Default user rights	None

RAS and IAS Servers

Computers that are members of the RAS and IAS Servers group, when properly configured, are allowed to use remote access services. By default, this group has no members. Computers that are running the Routing and Remote Access service are added to the group automatically, such as IAS servers and Network Policy Servers. Members of this group have access to certain properties of User objects, such as Read Account Restrictions, Read Logon Information, and Read Remote Access Information.

The RAS and IAS Servers group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-21-<domain>-553
Type	Builtin Local
Default container	CN=Users, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Yes
Safe to delegate management of this group to non-Service admins?	Yes
Default User Rights	None

RDS Endpoint Servers

Servers that are members in the RDS Endpoint Servers group can run virtual machines and host sessions where user RemoteApp programs and personal virtual desktops run. This group needs to be populated on servers running RD Connection Broker. Session Host servers and RD Virtualization Host servers used in the deployment need to be in this group.

For information about Remote Desktop Services, see [Host desktops and apps in Remote Desktop Services](#).

This security group was introduced in Windows Server 2012, and it has not changed in subsequent versions.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-576
Type	Builtin Local

ATTRIBUTE	VALUE
Default container	CN=Builtin, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	
Default User Rights	None

RDS Management Servers

Servers that are members in the RDS Management Servers group can be used to perform routine administrative actions on servers running Remote Desktop Services. This group needs to be populated on all servers in a Remote Desktop Services deployment. The servers running the RDS Central Management service must be included in this group.

This security group was introduced in Windows Server 2012, and it has not changed in subsequent versions.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-577
Type	Builtin Local
Default container	CN=Builtin, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	
Default User Rights	None

RDS Remote Access Servers

Servers in the RDS Remote Access Servers group provide users with access to RemoteApp programs and personal virtual desktops. In Internet facing deployments, these servers are typically deployed in an edge network. This group needs to be populated on servers running RD Connection Broker. RD Gateway servers and RD Web Access servers that are used in the deployment need to be in this group.

For more information, see [Host desktops and apps in Remote Desktop Services](#).

This security group was introduced in Windows Server 2012, and it has not changed in subsequent versions.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-575
Type	Builtin Local
Default container	CN=Builtin, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	
Default User Rights	None

Read-Only Domain Controllers

This group is comprised of the Read-only domain controllers in the domain. A Read-only domain controller makes it possible for organizations to easily deploy a domain controller in scenarios where physical security cannot be guaranteed, such as branch office locations, or in scenarios where local storage of all domain passwords is considered a primary threat, such as in an extranet or in an application-facing role.

Because administration of a Read-only domain controller can be delegated to a domain user or security group, an Read-only domain controller is well suited for a site that should not have a user who is a member of the Domain Admins group. A Read-only domain controller encompasses the following functionality:

- Read-only AD DS database
- Unidirectional replication
- Credential caching
- Administrator role separation
- Read-only Domain Name System (DNS)

For information about deploying a Read-only domain controller, see [Understanding Planning and Deployment for Read-Only Domain Controllers](#).

This security group was introduced in Windows Server 2008, and it has not changed in subsequent versions.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-21- <domain> -521
Type	Global
Default container	CN=Users, DC= <domain>, DC=

ATTRIBUTE	VALUE
Default members	None
Default member of	Denied RODC Password Replication Group
Protected by ADMINSDHOLDER?	Yes
Safe to move out of default container?	Yes
Safe to delegate management of this group to non-Service admins?	
Default User Rights	See Denied RODC Password Replication Group

Remote Desktop Users

The Remote Desktop Users group on an RD Session Host server is used to grant users and groups permissions to remotely connect to an RD Session Host server. This group cannot be renamed, deleted, or moved. It appears as a SID until the domain controller is made the primary domain controller and it holds the operations master role (also known as flexible single master operations or FSMO).

The Remote Desktop Users group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-555
Type	Builtin Local
Default container	CN=Builtin, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	Yes
Default User Rights	None

Remote Management Users

Members of the Remote Management Users group can access WMI resources over management protocols (such as WS-Management via the Windows Remote Management service). This applies only to WMI namespaces that grant access to the user.

The Remote Management Users group is generally used to allow users to manage servers through the Server Manager console, whereas the [WinRMRemoteWMIUsers_](#) group is allows remotely running Windows

PowerShell commands.

For more information, see [What's New in MI?](#) and [About WMI](#).

This security group was introduced in Windows Server 2012, and it has not changed in subsequent versions.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-580
Type	Builtin Local
Default container	CN=Builtin, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	
Default User Rights	None

Replicator

Computers that are members of the Replicator group support file replication in a domain. Windows Server operating systems use the File Replication service (FRS) to replicate system policies and logon scripts stored in the System Volume (SYSVOL). Each domain controller keeps a copy of SYSVOL for network clients to access. FRS can also replicate data for the Distributed File System (DFS), synchronizing the content of each member in a replica set as defined by DFS. FRS can copy and maintain shared files and folders on multiple servers simultaneously. When changes occur, content is synchronized immediately within sites and by a schedule between sites.

WARNING

In Windows Server 2008 R2, FRS cannot be used for replicating DFS folders or custom (non-SYSVOL) data. A Windows Server 2008 R2 domain controller can still use FRS to replicate the contents of a SYSVOL shared resource in a domain that uses FRS for replicating the SYSVOL shared resource between domain controllers.

However, Windows Server 2008 R2 servers cannot use FRS to replicate the contents of any replica set apart from the SYSVOL shared resource. The DFS Replication service is a replacement for FRS, and it can be used to replicate the contents of a SYSVOL shared resource, DFS folders, and other custom (non-SYSVOL) data. You should migrate all non-SYSVOL FRS replica sets to DFS Replication. For more information, see:

- [File Replication Service \(FRS\) Is Deprecated in Windows Server 2008 R2 \(Windows\)](#)
- [DFS Namespaces and DFS Replication Overview](#)

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-552
Type	Builtin Local
Default container	CN=Builtin, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	Yes
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	
Default User Rights	None

Schema Admins

Members of the Schema Admins group can modify the Active Directory schema. This group exists only in the root domain of an Active Directory forest of domains. It is a Universal group if the domain is in native mode; it is a Global group if the domain is in mixed mode.

The group is authorized to make schema changes in Active Directory. By default, the only member of the group is the Administrator account for the forest root domain. This group has full administrative access to the schema.

The membership of this group can be modified by any of the service administrator groups in the root domain. This is considered a service administrator account because its members can modify the schema, which governs the structure and content of the entire directory.

For more information, see [What Is the Active Directory Schema?: Active Directory](#).

The Schema Admins group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-21-<root domain>-518
Type	Universal (if Domain is in Native-Mode) else Global
Default container	CN=Users, DC= <domain>, DC=
Default members	Administrator
Default member of	Denied RODC Password Replication Group
Protected by ADMINSDHOLDER?	Yes

ATTRIBUTE	VALUE
Safe to move out of default container?	Yes
Safe to delegate management of this group to non-Service admins?	No
Default User Rights	See Denied RODC Password Replication Group

Server Operators

Members in the Server Operators group can administer domain controllers. This group exists only on domain controllers. By default, the group has no members. Members of the Server Operators group can sign in to a server interactively, create and delete network shared resources, start and stop services, back up and restore files, format the hard disk drive of the computer, and shut down the computer. This group cannot be renamed, deleted, or moved.

By default, this built-in group has no members, and it has access to server configuration options on domain controllers. Its membership is controlled by the service administrator groups Administrators and Domain Admins in the domain, and the Enterprise Admins group in the forest root domain. Members in this group cannot change any administrative group memberships. This is considered a service administrator account because its members have physical access to domain controllers, they can perform maintenance tasks (such as backup and restore), and they have the ability to change binaries that are installed on the domain controllers. Note the default user rights in the following table.

The Server Operators group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-549
Type	Builtin Local
Default container	CN=Builtin, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	Yes
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	No

ATTRIBUTE	VALUE
Default User Rights	<p>Allow log on locally: SeInteractiveLogonRight</p> <p>Back up files and directories: SeBackupPrivilege</p> <p>Change the system time: SeSystemTimePrivilege</p> <p>Change the time zone: SeTimeZonePrivilege</p> <p>Force shutdown from a remote system: SeRemoteShutdownPrivilege</p> <p>Restore files and directories: Restore files and directories SeRestorePrivilege</p> <p>Shut down the system: SeShutdownPrivilege</p>

Storage Replica Administrators

Members of this group have complete and unrestricted access to all features of Storage Replica.

The Storage Replica Administrators group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-582
Type	Builtin Local
Default container	CN=BuiltIn, DC= <domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Yes
Safe to delegate management of this group to non-Service admins?	No
Default User Rights	None

System Managed Accounts Group

Members of this group are managed by the system.

The System Managed Accounts group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-581
Type	Builtin Local

ATTRIBUTE	VALUE
Default container	CN=BuiltIn, DC= <domain>, DC=
Default members	Users
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Yes
Safe to delegate management of this group to non-Service admins?	No
Default User Rights	None

Terminal Server License Servers

Members of the Terminal Server License Servers group can update user accounts in Active Directory with information about license issuance. This is used to track and report TS Per User CAL usage. A TS Per User CAL gives one user the right to access a Terminal Server from an unlimited number of client computers or devices. This group appears as a SID until the domain controller is made the primary domain controller and it holds the operations master role (also known as flexible single master operations or FSMO).

For more information about this security group, see [Terminal Services License Server Security Group Configuration](#).

The Terminal Server License Servers group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

NOTE

This group cannot be renamed, deleted, or moved.

This security group only applies to Windows Server 2003 and Windows Server 2008 because Terminal Services was replaced by Remote Desktop Services in Windows Server 2008 R2.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-561
Type	Builtin Local
Default container	CN=Builtin, DC= <domain>, DC=
Default members	None
Default member of	None
Safe to move out of default container?	Cannot be moved
Protected by ADMINSDHOLDER?	No

ATTRIBUTE	VALUE
Safe to delegate management of this group to non-Service admins?	Yes
Default User Rights	None

Users

Members of the Users group are prevented from making accidental or intentional system-wide changes, and they can run most applications. After the initial installation of the operating system, the only member is the Authenticated Users group. When a computer joins a domain, the Domain Users group is added to the Users group on the computer.

Users can perform tasks such as running applications, using local and network printers, shutting down the computer, and locking the computer. Users can install applications that only they are allowed to use if the installation program of the application supports per-user installation. This group cannot be renamed, deleted, or moved.

The Users group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

This security group includes the following changes since Windows Server 2008:

- In Windows Server 2008 R2, INTERACTIVE was added to the default members list.
- In Windows Server 2012, the default **Member Of** list changed from Domain Users to none.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-545
Type	Builtin Local
Default container	CN=Builtin, DC= <domain>, DC=
Default members	Authenticated Users Domain Users INTERACTIVE
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	No
Default User Rights	None

Windows Authorization Access Group

Members of this group have access to the computed token GroupsGlobalAndUniversal attribute on User objects. Some applications have features that read the token-groups-global-and-universal (TGGAU) attribute on user account objects or on computer account objects in Active Directory Domain Services. Some Win32 functions

make it easier to read the TGGAU attribute. Applications that read this attribute or that call an API (referred to as a function) that reads this attribute do not succeed if the calling security context does not have access to the attribute. This group appears as a SID until the domain controller is made the primary domain controller and it holds the operations master role (also known as flexible single master operations or FSMO).

The Windows Authorization Access group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

NOTE

This group cannot be renamed, deleted, or moved.

This security group has not changed since Windows Server 2008.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-32-560
Type	Builtin Local
Default container	CN=Builtin, DC= <domain>, DC=
Default members	Enterprise Domain Controllers
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Cannot be moved
Safe to delegate management of this group to non-Service admins?	Yes
Default user rights	None

WinRMRemoteWMIUsers_

In Windows 8 and in Windows Server 2012, a **Share** tab was added to the Advanced Security Settings user interface. This tab displays the security properties of a remote file share. To view this information, you must have the following permissions and memberships, as appropriate for the version of Windows Server that the file server is running.

The WinRMRemoteWMIUsers_ group applies to versions of the Windows Server operating system listed in the [Active Directory Default Security Groups table](#).

- If the file share is hosted on a server that is running a supported version of the operating system:
 - You must be a member of the WinRMRemoteWMIUsers__ group or the BUILTIN\Administrators group.
 - You must have Read permissions to the file share.
- If the file share is hosted on a server that is running a version of Windows Server that is earlier than Windows Server 2012:
 - You must be a member of the BUILTIN\Administrators group.

- You must have Read permissions to the file share.

In Windows Server 2012, the Access Denied Assistance functionality adds the Authenticated Users group to the local WinRMRemoteWMIUsers__ group. Therefore, when the Access Denied Assistance functionality is enabled, all authenticated users who have Read permissions to the file share can view the file share permissions.

NOTE

The WinRMRemoteWMIUsers_ group allows running Windows PowerShell commands remotely whereas the [Remote Management Users](#) group is generally used to allow users to manage servers by using the Server Manager console.

This security group was introduced in Windows Server 2012, and it has not changed in subsequent versions.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-21-<domain>-<variable RI>
Type	Domain local
Default container	CN=Users, DC=<domain>, DC=
Default members	None
Default member of	None
Protected by ADMINSDHOLDER?	No
Safe to move out of default container?	Yes
Safe to delegate management of this group to non-Service admins?	
Default User Rights	None

See also

- [Security Principals](#)
- [Special Identities](#)
- [Access Control Overview](#)

Special Identities

7/1/2022 • 12 minutes to read • [Edit Online](#)

Applies to

- Windows Server 2016 or later

This reference topic for the IT professional describes the special identity groups (which are sometimes referred to as security groups) that are used in Windows access control.

Special identity groups are similar to Active Directory security groups as listed in the users and built-in containers. Special identity groups can provide an efficient way to assign access to resources in your network. By using special identity groups, you can:

- Assign user rights to security groups in Active Directory.
- Assign permissions to security groups for the purpose of accessing resources.

Servers that are running the supported Windows Server operating systems designated in the **Applies To** list at the beginning of this topic include several special identity groups. These special identity groups do not have specific memberships that can be modified, but they can represent different users at different times, depending on the circumstances.

Although the special identity groups can be assigned rights and permissions to resources, the memberships cannot be modified or viewed. Group scopes do not apply to special identity groups. Users are automatically assigned to these special identity groups whenever they sign in or access a particular resource.

For information about security groups and group scope, see [Active Directory Security Groups](#).

The special identity groups are described in the following tables:

- [Anonymous Logon](#)
- [Authenticated Users](#)
- [Batch](#)
- [Creator Group](#)
- [Creator Owner](#)
- [Dialup](#)
- [Digest Authentication](#)
- [Enterprise Domain Controllers](#)
- [Everyone](#)
- [Interactive](#)
- [Local Service](#)
- [LocalSystem](#)
- [Network](#)
- [Network Service](#)

- [NTLM Authentication](#)
- [Other Organization](#)
- [Principal Self](#)
- [Remote Interactive Logon](#)
- [Restricted](#)
- [SChannel Authentication](#)
- [Service](#)
- [Terminal Server User](#)
- [This Organization](#)
- [Window Manager\Window Manager Group](#)

Anonymous Logon

Any user who accesses the system through an anonymous logon has the Anonymous Logon identity. This identity allows anonymous access to resources, such as a web page that is published on corporate servers. The Anonymous Logon group is not a member of the Everyone group by default.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-7
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	None

Attested Key Property

A SID that means the key trust object had the attestation property.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-18-6
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	None

Authenticated Users

Any user who accesses the system through a sign-in process has the Authenticated Users identity. This identity allows access to shared resources within the domain, such as files in a shared folder that should be accessible to all the workers in the organization. Membership is controlled by the operating system.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-11
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	Access this computer from the network: SeNetworkLogonRight Add workstations to domain: SeMachineAccountPrivilege Bypass traverse checking: SeChangeNotifyPrivilege

Authentication Authority Asserted Identity

A SID that means the client's identity is asserted by an authentication authority based on proof of possession of client credentials.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-18-1
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	None

Batch

Any user or process that accesses the system as a batch job (or through the batch queue) has the Batch identity. This identity allows batch jobs to run scheduled tasks, such as a nightly cleanup job that deletes temporary files. Membership is controlled by the operating system.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-3
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	none

Console Logon

A group that includes users who are logged on to the physical console. This SID can be used to implement security policies that grant different rights based on whether a user has been granted physical access to the console.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-2-1
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	None

Creator Group

The person who created the file or the directory is a member of this special identity group. Windows Server operating systems use this identity to automatically grant access permissions to the creator of a file or directory.

A placeholder security identifier (SID) is created in an inheritable access control entry (ACE). When the ACE is inherited, the system replaces this SID with the SID for the primary group of the object's current owner. The primary group is used only by the Portable Operating System Interface for UNIX (POSIX) subsystem.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-3-1
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	none

Creator Owner

The person who created the file or the directory is a member of this special identity group. Windows Server operating systems use this identity to automatically grant access permissions to the creator of a file or directory. A placeholder SID is created in an inheritable ACE. When the ACE is inherited, the system replaces this SID with the SID for the object's current owner.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-3-0
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	none

Dialup

Any user who accesses the system through a dial-up connection has the Dial-Up identity. This identity distinguishes dial-up users from other types of authenticated users.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-1
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	none

Digest Authentication

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-64-21
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	none

Enterprise Domain Controllers

This group includes all domain controllers in an Active Directory forest. Domain controllers with enterprise-wide roles and responsibilities have the Enterprise Domain Controllers identity. This identity allows them to perform certain tasks in the enterprise by using transitive trusts. Membership is controlled by the operating system.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-9
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	Access this computer from the network: SeNetworkLogonRight Allow log on locally: SeInteractiveLogonRight

Everyone

All interactive, network, dial-up, and authenticated users are members of the Everyone group. This special identity group gives wide access to system resources. Whenever a user logs on to the network, the user is automatically added to the Everyone group.

On computers running Windows 2000 and earlier, the Everyone group included the Anonymous Logon group as a default member, but as of Windows Server 2003, the Everyone group contains only Authenticated Users and Guest; and it no longer includes Anonymous Logon by default (although this can be changed, using Registry Editor, by going to the `Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa` key

and setting the value of **everyoneincludesanonymous** (DWORD to 1).

Membership is controlled by the operating system.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-1-0
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	Access this computer from the network: SeNetworkLogonRight Act as part of the operating system: SeTcbPrivilege Bypass traverse checking: SeChangeNotifyPrivilege

Fresh Public Key Identity

A SID that means the client's identity is asserted by an authentication authority based on proof of current possession of client public key credentials.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-18-3
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	None

Interactive

Any user who is logged on to the local system has the Interactive identity. This identity allows only local users to access a resource. Whenever a user accesses a given resource on the computer to which they are currently logged on, the user is automatically added to the Interactive group. Membership is controlled by the operating system.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-4
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	None

IUSR

Internet Information Services (IIS) uses this account by default whenever anonymous authentication is enabled.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-17
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	None

Key Trust

A SID that means the client's identity is based on proof of possession of public key credentials using the key trust object.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-18-4
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	None

Local Service

The Local Service account is similar to an Authenticated User account. The Local Service account has the same level of access to resources and objects as members of the Users group. This limited access helps safeguard your system if individual services or processes are compromised. Services that run as the Local Service account access network resources as a null session with anonymous credentials. The name of the account is NT AUTHORITY\LocalService. This account does not have a password.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-19
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>

ATTRIBUTE	VALUE
Default User Rights	Adjust memory quotas for a process: SeIncreaseQuotaPrivilege Bypass traverse checking: SeChangeNotifyPrivilege Change the system time: SeSystemtimePrivilege Change the time zone: SeTimeZonePrivilege Create global objects: SeCreateGlobalPrivilege Generate security audits: SeAuditPrivilege Impersonate a client after authentication: SeImpersonatePrivilege Replace a process level token: SeAssignPrimaryTokenPrivilege

LocalSystem

This is a service account that is used by the operating system. The LocalSystem account is a powerful account that has full access to the system and acts as the computer on the network. If a service logs on to the LocalSystem account on a domain controller, that service has access to the entire domain. Some services are configured by default to log on to the LocalSystem account. Do not change the default service setting. The name of the account is LocalSystem. This account does not have a password.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-18
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	None

MFA Key Property

A SID that means the key trust object had the multifactor authentication (MFA) property.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-18-5
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	None

Network

This group implicitly includes all users who are logged on through a network connection. Any user who accesses the system through a network has the Network identity. This identity allows only remote users to access a resource. Whenever a user accesses a given resource over the network, the user is automatically added to the Network group. Membership is controlled by the operating system.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-2
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	None

Network Service

The Network Service account is similar to an Authenticated User account. The Network Service account has the same level of access to resources and objects as members of the Users group. This limited access helps safeguard your system if individual services or processes are compromised. Services that run as the Network Service account access network resources by using the credentials of the computer account. The name of the account is NT AUTHORITY\NetworkService. This account does not have a password.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-20
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	Adjust memory quotas for a process: SeIncreaseQuotaPrivilege Bypass traverse checking: SeChangeNotifyPrivilege Create global objects: SeCreateGlobalPrivilege Generate security audits: SeAuditPrivilege Impersonate a client after authentication: SeImpersonatePrivilege Replace a process level token: SeAssignPrimaryTokenPrivilege

NTLM Authentication

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-64-10
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	None

Other Organization

This group implicitly includes all users who are logged on to the system through a dial-up connection.

Membership is controlled by the operating system.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-1000
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	None

Owner Rights

A group that represents the current owner of the object. When an ACE that carries this SID is applied to an object, the system ignores the implicit READ_CONTROL and WRITE_DAC permissions for the object owner.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-3-4
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	None

Principal Self

This identity is a placeholder in an ACE on a user, group, or computer object in Active Directory. When you grant permissions to Principal Self, you grant them to the security principal that is represented by the object. During an access check, the operating system replaces the SID for Principal Self with the SID for the security principal that is represented by the object.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-10
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	None

Proxy

Identifies a SECURITY_NT_AUTHORITY Proxy.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-8
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	None

Remote Interactive Logon

This identity represents all users who are currently logged on to a computer by using a Remote Desktop connection. This group is a subset of the Interactive group. Access tokens that contain the Remote Interactive Logon SID also contain the Interactive SID.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-14
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	None

Restricted

Users and computers with restricted capabilities have the Restricted identity. This identity group is used by a process that is running in a restricted security context, such as running an application with the RunAs service. When code runs at the Restricted security level, the Restricted SID is added to the user's access token.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-12
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	None

SChannel Authentication

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-64-14

ATTRIBUTE	VALUE
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	None

Service

Any service that accesses the system has the Service identity. This identity group includes all security principals that are signed in as a service. This identity grants access to processes that are being run by Windows Server services. Membership is controlled by the operating system.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-6
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	Create global objects: SeCreateGlobalPrivilege Impersonate a client after authentication: SelImpersonatePrivilege

Service Asserted Identity

A SID that means the client's identity is asserted by a service.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-18-2
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	None

Terminal Server User

Any user accessing the system through Terminal Services has the Terminal Server User identity. This identity allows users to access Terminal Server applications and to perform other necessary tasks with Terminal Server services. Membership is controlled by the operating system.

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-13

ATTRIBUTE	VALUE
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	None

This Organization

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-15
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	None

Window Manager\Window Manager Group

ATTRIBUTE	VALUE
Well-Known SID/RID	S-1-5-90
Object Class	Foreign Security Principal
Default Location in Active Directory	cn=WellKnown Security Principals, cn=Configuration, dc=<forestRootDomain>
Default User Rights	Bypass traverse checking : SeChangeNotifyPrivilege Increase a process working set : SeIncreaseWorkingSetPrivilege

See also

- [Active Directory Security Groups](#)
- [Security Principals](#)
- [Access Control Overview](#)

User Account Control

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016 and above

User Account Control (UAC) helps prevent malware from damaging a PC and helps organizations deploy a better-managed desktop. With UAC, apps and tasks always run in the security context of a non-administrator account, unless an administrator specifically authorizes administrator-level access to the system. UAC can block the automatic installation of unauthorized apps and prevent inadvertent changes to system settings.

UAC allows all users to log on to their computers using a standard user account. Processes launched using a standard user token may perform tasks using access rights granted to a standard user. For instance, Windows Explorer automatically inherits standard user level permissions. Additionally, any apps that are started using Windows Explorer (for example, by double-clicking a shortcut) also run with the standard set of user permissions. Many apps, including those that are included with the operating system itself, are designed to work properly in this way.

Other apps, especially those that were not specifically designed with security settings in mind, often require additional permissions to run successfully. These types of apps are referred to as legacy apps. Additionally, actions such as installing new software and making configuration changes to the Windows Firewall, require more permissions than what is available to a standard user account.

When an app needs to run with more than standard user rights, UAC allows users to run apps with their administrator token (with administrative groups and privileges) instead of their default, standard user access token. Users continue to operate in the standard user security context, while enabling certain apps to run with elevated privileges, if needed.

Practical applications

Admin Approval Mode in UAC helps prevent malware from silently installing without an administrator's knowledge. It also helps protect from inadvertent system-wide changes. Lastly, it can be used to enforce a higher level of compliance where administrators must actively consent or provide credentials for each administrative process.

In this section

TOPIC	DESCRIPTION
How User Account Control works	User Account Control (UAC) is a fundamental component of Microsoft's overall security vision. UAC helps mitigate the impact of malware.
User Account Control security policy settings	You can use security policies to configure how User Account Control works in your organization. They can be configured locally by using the Local Security Policy snap-in (secpol.msc) or configured for the domain, OU, or specific groups by Group Policy.

TOPIC	DESCRIPTION
User Account Control Group Policy and registry key settings	Here's a list of UAC Group Policy and registry key settings that your organization can use to manage UAC.

How User Account Control works

7/1/2022 • 14 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016 and above

User Account Control (UAC) is a fundamental component of Microsoft's overall security vision. UAC helps mitigate the impact of malware.

UAC process and interactions

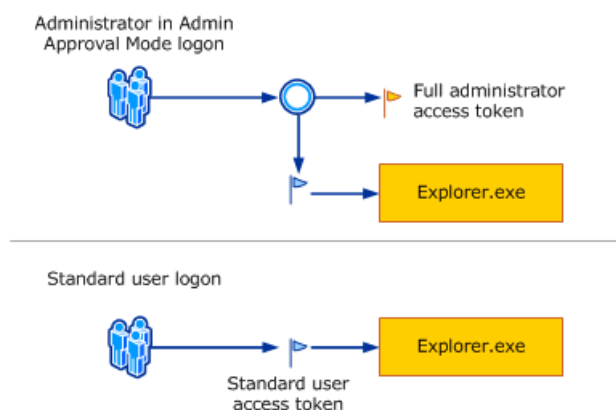
Each app that requires the administrator access token must prompt for consent. The one exception is the relationship that exists between parent and child processes. Child processes inherit the user's access token from the parent process. Both the parent and child processes, however, must have the same integrity level.

Windows protects processes by marking their integrity levels. Integrity levels are measurements of trust. A "high" integrity application is one that performs tasks that modify system data, such as a disk partitioning application, while a "low" integrity application is one that performs tasks that could potentially compromise the operating system, such as a Web browser. Apps with lower integrity levels cannot modify data in applications with higher integrity levels. When a standard user attempts to run an app that requires an administrator access token, UAC requires that the user provide valid administrator credentials.

To better understand how this process happens, let's look at the Windows logon process.

Logon process

The following shows how the logon process for an administrator differs from the logon process for a standard user.



By default, standard users and administrators access resources and run apps in the security context of standard users. When a user logs on to a computer, the system creates an access token for that user. The access token contains information about the level of access that the user is granted, including specific security identifiers (SIDs) and Windows privileges.

When an administrator logs on, two separate access tokens are created for the user: a standard user access token and an administrator access token. The standard user access token contains the same user-specific information as the administrator access token, but the administrative Windows privileges and SIDs are removed. The standard user access token is used to start apps that do not perform administrative tasks (standard user apps). The standard user access token is then used to display the desktop (explorer.exe). Explorer.exe is the

parent process from which all other user-initiated processes inherit their access token. As a result, all apps run as a standard user unless a user provides consent or credentials to approve an app to use a full administrative access token.

A user that is a member of the Administrators group can log on, browse the Web, and read e-mail while using a standard user access token. When the administrator needs to perform a task that requires the administrator access token, Windows 10 or Windows 11 automatically prompts the user for approval. This prompt is called an elevation prompt, and its behavior can be configured by using the Local Security Policy snap-in (Secpol.msc) or Group Policy. For more info, see [User Account Control security policy settings](#).

The UAC User Experience

When UAC is enabled, the user experience for standard users is different from that of administrators in Admin Approval Mode. The recommended and more secure method of running Windows 10 or Windows 11 is to make your primary user account a standard user account. Running as a standard user helps to maximize security for a managed environment. With the built-in UAC elevation component, standard users can easily perform an administrative task by entering valid credentials for a local administrator account. The default, built-in UAC elevation component for standard users is the credential prompt.

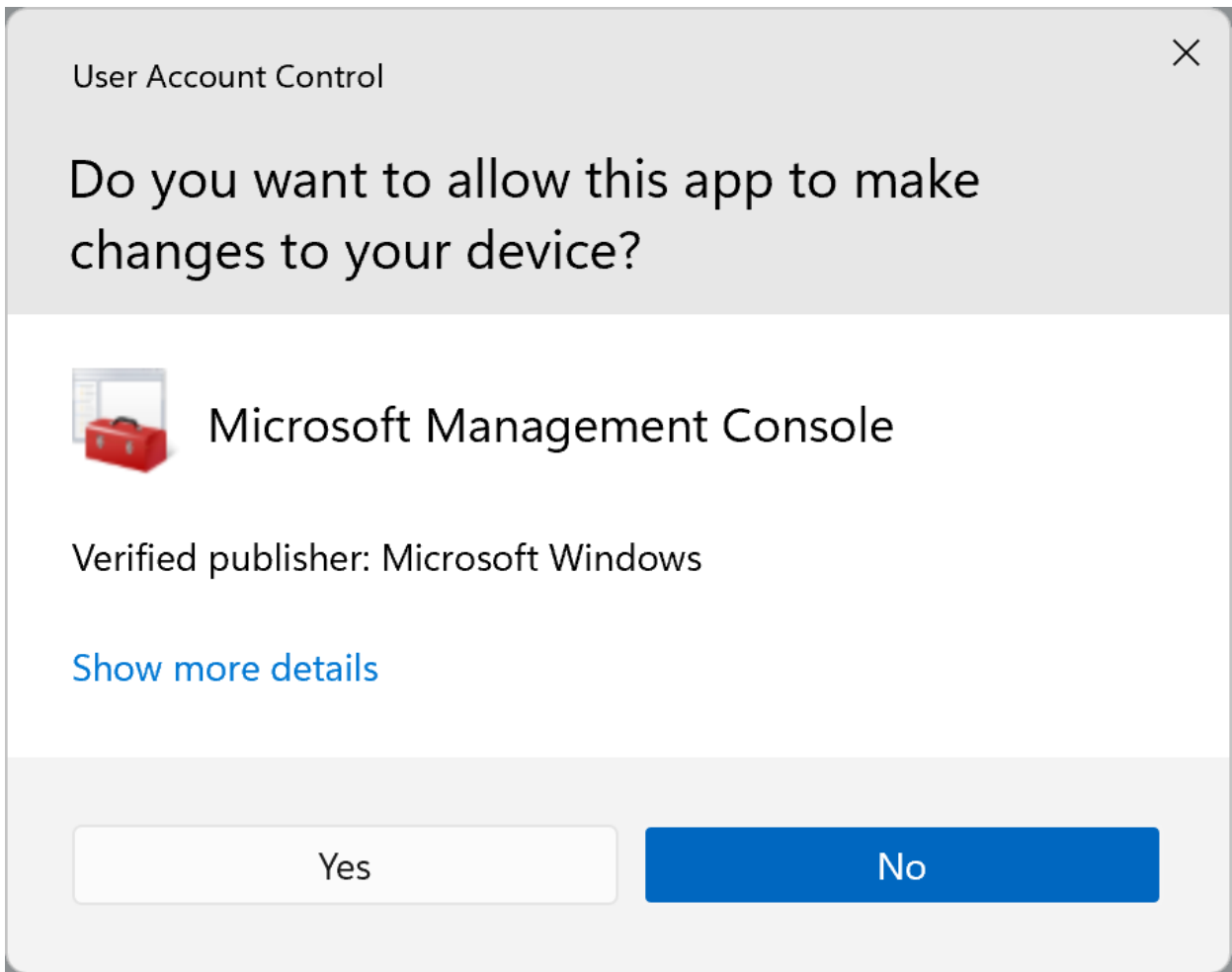
The alternative to running as a standard user is to run as an administrator in Admin Approval Mode. With the built-in UAC elevation component, members of the local Administrators group can easily perform an administrative task by providing approval. The default, built-in UAC elevation component for an administrator account in Admin Approval Mode is called the consent prompt.

The consent and credential prompts

With UAC enabled, Windows 10 or Windows 11 prompts for consent or prompts for credentials of a valid local administrator account before starting a program or task that requires a full administrator access token. This prompt ensures that no malicious software can be silently installed.

The consent prompt

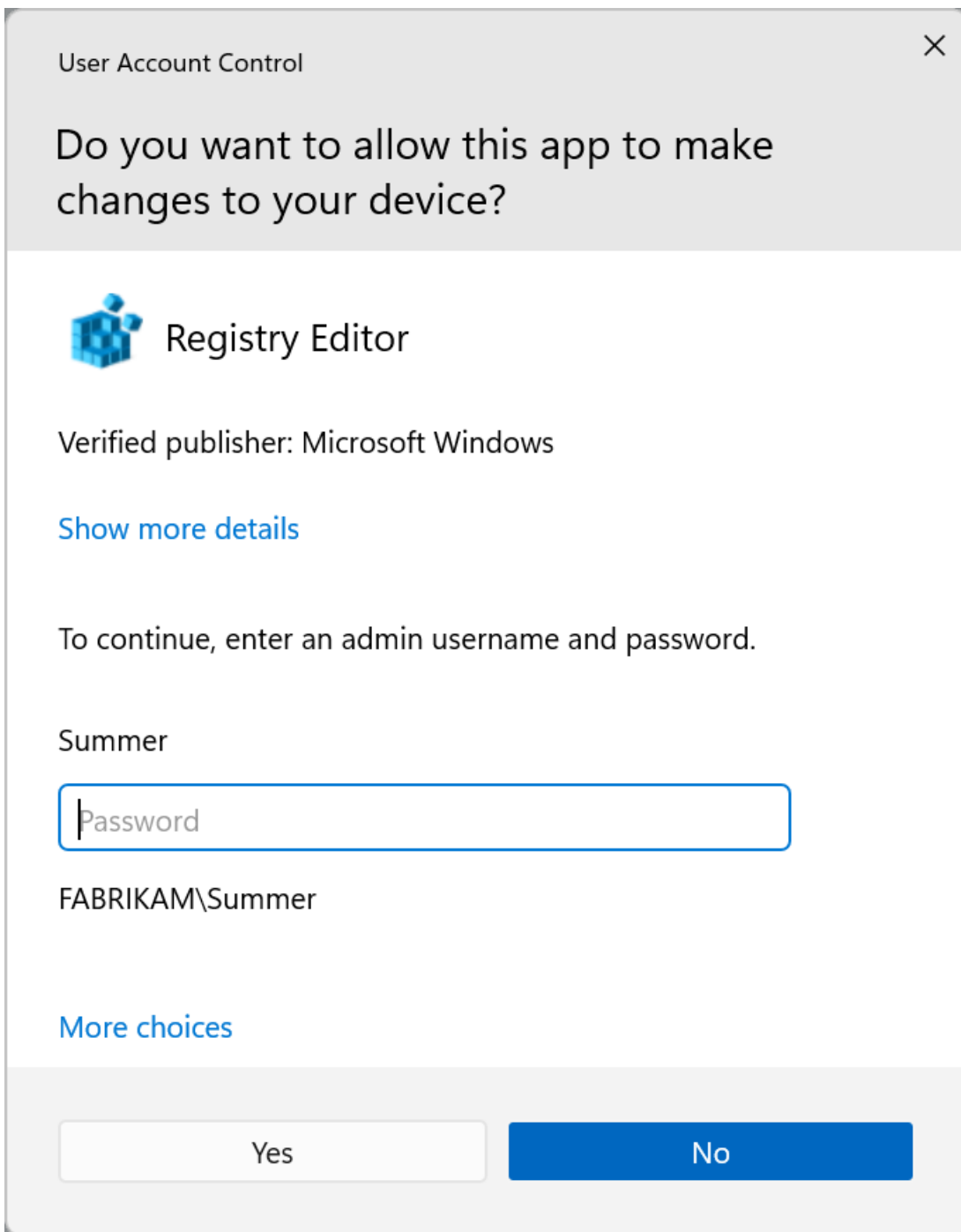
The consent prompt is presented when a user attempts to perform a task that requires a user's administrative access token. The following is an example of the UAC consent prompt.



The credential prompt

The credential prompt is presented when a standard user attempts to perform a task that requires a user's administrative access token. Administrators can also be required to provide their credentials by setting the **User Account Control: Behavior of the elevation prompt for administrators in Admin Approval Mode** policy setting value to **Prompt for credentials**.

The following is an example of the UAC credential prompt.



UAC elevation prompts

The UAC elevation prompts are color-coded to be app-specific, enabling for immediate identification of an application's potential security risk. When an app attempts to run with an administrator's full access token, Windows 10 or Windows 11 first analyzes the executable file to determine its publisher. Apps are first separated into three categories based on the file's publisher: Windows 10 or Windows 11, publisher verified (signed), and publisher not verified (unsigned). The following diagram illustrates how Windows determines which color elevation prompt to present to the user.

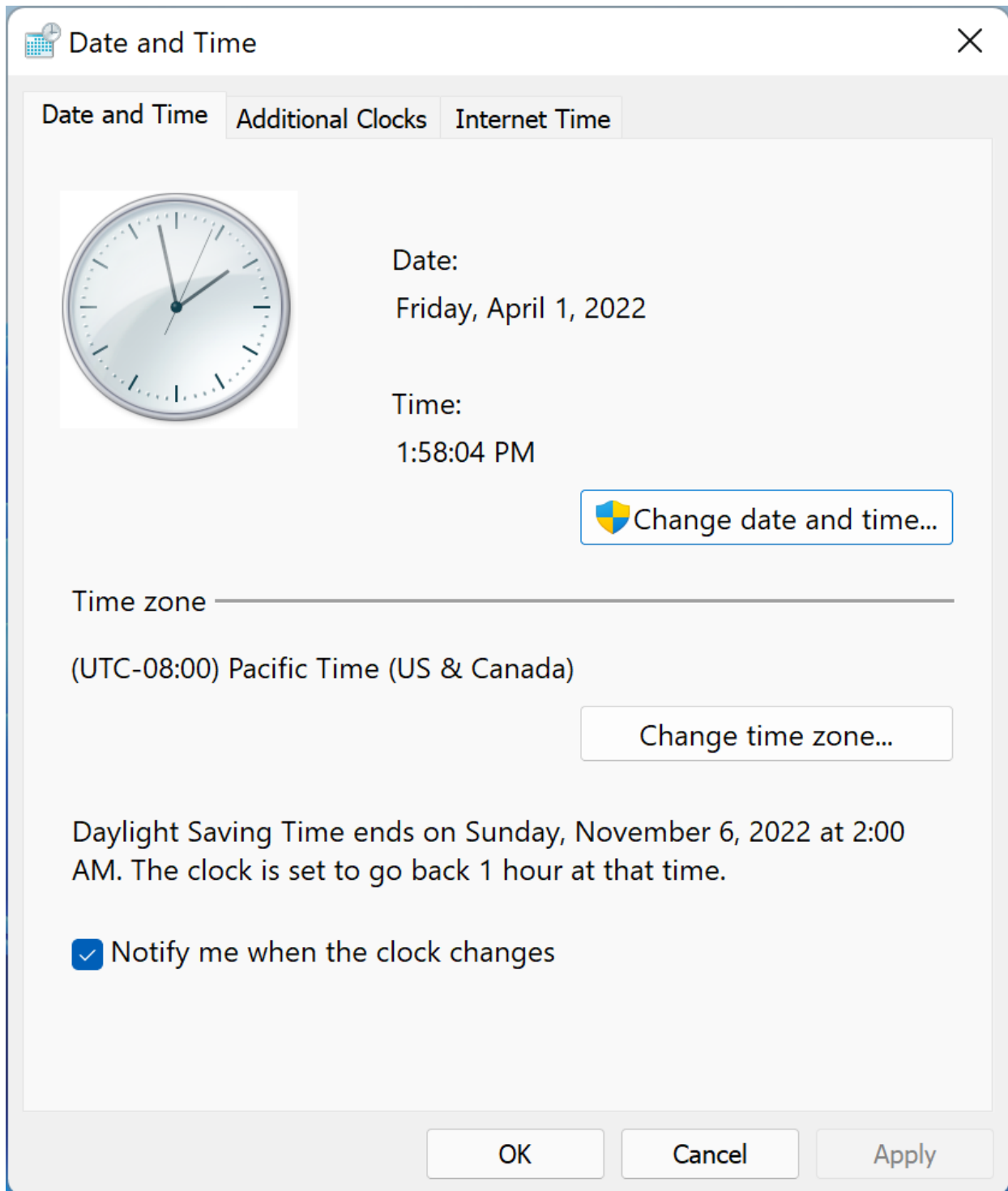
The elevation prompt color-coding is as follows:

- Red background with a red shield icon: The app is blocked by Group Policy or is from a publisher that is blocked.
- Blue background with a blue and gold shield icon: The application is a Windows 10 and Windows 11 administrative app, such as a Control Panel item.
- Blue background with a blue shield icon: The application is signed by using Authenticode and is trusted by the local computer.

- Yellow background with a yellow shield icon: The application is unsigned or signed but is not yet trusted by the local computer.

Shield icon

Some Control Panel items, such as **Date and Time Properties**, contain a combination of administrator and standard user operations. Standard users can view the clock and change the time zone, but a full administrator access token is required to change the local system time. The following is a screen shot of the **Date and Time Properties** Control Panel item.



The shield icon on the **Change date and time** button indicates that the process requires a full administrator access token and will display a UAC elevation prompt.

Securing the elevation prompt

The elevation process is further secured by directing the prompt to the secure desktop. The consent and

credential prompts are displayed on the secure desktop by default in Windows 10 and Windows 11. Only Windows processes can access the secure desktop. For higher levels of security, we recommend keeping the **User Account Control: Switch to the secure desktop when prompting for elevation** policy setting enabled.

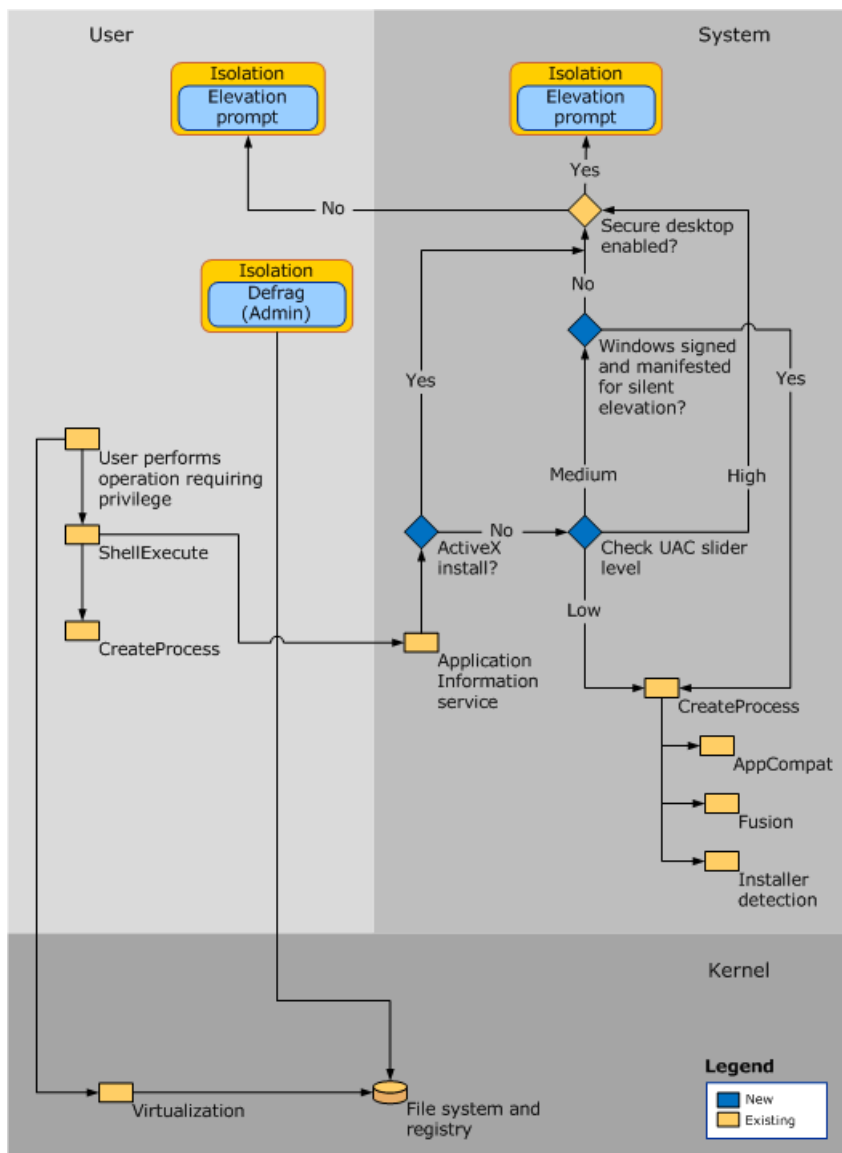
When an executable file requests elevation, the interactive desktop, also called the user desktop, is switched to the secure desktop. The secure desktop dims the user desktop and displays an elevation prompt that must be responded to before continuing. When the user clicks **Yes** or **No**, the desktop switches back to the user desktop.

Malware can present an imitation of the secure desktop, but when the **User Account Control: Behavior of the elevation prompt for administrators in Admin Approval Mode** policy setting is set to **Prompt for consent**, the malware does not gain elevation if the user clicks **Yes** on the imitation. If the policy setting is set to **Prompt for credentials**, malware imitating the credential prompt may be able to gather the credentials from the user. However, the malware does not gain elevated privilege and the system has other protections that mitigate malware from taking control of the user interface even with a harvested password.

While malware could present an imitation of the secure desktop, this issue cannot occur unless a user previously installed the malware on the PC. Because processes requiring an administrator access token cannot silently install when UAC is enabled, the user must explicitly provide consent by clicking **Yes** or by providing administrator credentials. The specific behavior of the UAC elevation prompt is dependent upon Group Policy.

UAC Architecture

The following diagram details the UAC architecture.



To better understand each component, review the table below:

User

COMPONENT	DESCRIPTION
User performs operation requiring privilege	If the operation changes the file system or registry, Virtualization is called. All other operations call ShellExecute.
ShellExecute	ShellExecute calls CreateProcess. ShellExecute looks for the ERROR_ELEVATION_REQUIRED error from CreateProcess. If it receives the error, ShellExecute calls the Application Information service to attempt to perform the requested task with the elevated prompt.
CreateProcess	If the application requires elevation, CreateProcess rejects the call with ERROR_ELEVATION_REQUIRED.

System

COMPONENT	DESCRIPTION
Application Information service	A system service that helps start apps that require one or more elevated privileges or user rights to run, such as local administrative tasks, and apps that require higher integrity levels. The Application Information service helps start such apps by creating a new process for the application with an administrative user's full access token when elevation is required and (depending on Group Policy) consent is given by the user to do so.
Elevating an ActiveX install	If ActiveX is not installed, the system checks the UAC slider level. If ActiveX is installed, the User Account Control: Switch to the secure desktop when prompting for elevation Group Policy setting is checked.

COMPONENT	DESCRIPTION
Check UAC slider level	<p>UAC has a slider to select from four levels of notification.</p> <ul style="list-style-type: none"> • Always notify will: <ul style="list-style-type: none"> ◦ Notify you when programs try to install software or make changes to your computer. ◦ Notify you when you make changes to Windows settings. ◦ Freeze other tasks until you respond. <p>Recommended if you often install new software or visit unfamiliar websites.</p> • Notify me only when programs try to make changes to my computer will: <ul style="list-style-type: none"> ◦ Notify you when programs try to install software or make changes to your computer. ◦ Not notify you when you make changes to Windows settings. ◦ Freeze other tasks until you respond. <p>Recommended if you do not often install apps or visit unfamiliar websites.</p> • Notify me only when programs try to make changes to my computer (do not dim my desktop) will: <ul style="list-style-type: none"> ◦ Notify you when programs try to install software or make changes to your computer. ◦ Not notify you when you make changes to Windows settings. ◦ Not freeze other tasks until you respond. <p>Not recommended. Choose this only if it takes a long time to dim the desktop on your computer.</p> • Never notify (Disable UAC prompts) will: <ul style="list-style-type: none"> ◦ Not notify you when programs try to install software or make changes to your computer. ◦ Not notify you when you make changes to Windows settings. ◦ Not freeze other tasks until you respond. <p>Not recommended due to security concerns.</p>
Secure desktop enabled	<p>The User Account Control: Switch to the secure desktop when prompting for elevation policy setting is checked:</p> <ul style="list-style-type: none"> • If the secure desktop is enabled, all elevation requests go to the secure desktop regardless of prompt behavior policy settings for administrators and standard users. • If the secure desktop is not enabled, all elevation requests go to the interactive user's desktop, and the per-user settings for administrators and standard users are used.

COMPONENT	DESCRIPTION
CreateProcess	CreateProcess calls AppCompat, Fusion, and Installer detection to assess if the app requires elevation. The file is then inspected to determine its requested execution level, which is stored in the application manifest for the file. CreateProcess fails if the requested execution level specified in the manifest does not match the access token and returns an error (ERROR_ELEVATION_REQUIRED) to ShellExecute.
AppCompat	The AppCompat database stores information in the application compatibility fix entries for an application.
Fusion	The Fusion database stores information from application manifests that describe the applications. The manifest schema is updated to add a new requested execution level field.
Installer detection	Installer detection detects setup files, which helps prevent installations from being run without the user's knowledge and consent.

Kernel

COMPONENT	DESCRIPTION
Virtualization	Virtualization technology ensures that non-compliant apps do not silently fail to run or fail in a way that the cause cannot be determined. UAC also provides file and registry virtualization and logging for applications that write to protected areas.
File system and registry	The per-user file and registry virtualization redirects per-computer registry and file write requests to equivalent per-user locations. Read requests are redirected to the virtualized per-user location first and to the per-computer location second.

The slider will never turn UAC completely off. If you set it to **Never notify**, it will:

- Keep the UAC service running.
- Cause all elevation request initiated by administrators to be auto-approved without showing a UAC prompt.
- Automatically deny all elevation requests for standard users.

IMPORTANT

In order to fully disable UAC you must disable the policy **User Account Control: Run all administrators in Admin Approval Mode**.

WARNING

Some Universal Windows Platform apps may not work when UAC is disabled.

Virtualization

Because system administrators in enterprise environments attempt to secure systems, many line-of-business (LOB) applications are designed to use only a standard user access token. As a result, you do not need to replace the majority of apps when UAC is turned on.

Windows 10 and Windows 11 include file and registry virtualization technology for apps that are not UAC-compliant and that require an administrator's access token to run correctly. When an administrative app that is not UAC-compliant attempts to write to a protected folder, such as Program Files, UAC gives the app its own virtualized view of the resource it is attempting to change. The virtualized copy is maintained in the user's profile. This strategy creates a separate copy of the virtualized file for each user that runs the non-compliant app.

Most app tasks operate properly by using virtualization features. Although virtualization allows a majority of applications to run, it is a short-term fix and not a long-term solution. App developers should modify their apps to be compliant as soon as possible, rather than relying on file, folder, and registry virtualization.

Virtualization is not an option in the following scenarios:

- Virtualization does not apply to apps that are elevated and run with a full administrative access token.
- Virtualization supports only 32-bit apps. Non-elevated 64-bit apps simply receive an access denied message when they attempt to acquire a handle (a unique identifier) to a Windows object. Native Windows 64-bit apps are required to be compatible with UAC and to write data into the correct locations.
- Virtualization is disabled if the app includes an app manifest with a requested execution level attribute.

Request execution levels

An app manifest is an XML file that describes and identifies the shared and private side-by-side assemblies that an app should bind to at run time. The app manifest includes entries for UAC app compatibility purposes. Administrative apps that include an entry in the app manifest prompt the user for permission to access the user's access token. Although they lack an entry in the app manifest, most administrative app can run without modification by using app compatibility fixes. App compatibility fixes are database entries that enable applications that are not UAC-compliant to work properly.

All UAC-compliant apps should have a requested execution level added to the application manifest. If the application requires administrative access to the system, then marking the app with a requested execution level of "require administrator" ensures that the system identifies this program as an administrative app and performs the necessary elevation steps. Requested execution levels specify the privileges required for an app.

Installer detection technology

Installation programs are apps designed to deploy software. Most installation programs write to system directories and registry keys. These protected system locations are typically writeable only by an administrator in installer detection technology, which means that standard users do not have sufficient access to install programs. Windows 10 and Windows 11 heuristically detect installation programs and requests administrator credentials or approval from the administrator user in order to run with access privileges. Windows 10 and Windows 11 also heuristically detect updates and programs that uninstall applications. One of the design goals of UAC is to prevent installations from being run without the user's knowledge and consent because installation programs write to protected areas of the file system and registry.

Installer detection only applies to:

- 32-bit executable files.

- Applications without a requested execution level attribute.
- Interactive processes running as a standard user with UAC enabled.

Before a 32-bit process is created, the following attributes are checked to determine whether it is an installer:

- The file name includes keywords such as "install," "setup," or "update."
- Versioning Resource fields contain the following keywords: Vendor, Company Name, Product Name, File Description, Original Filename, Internal Name, and Export Name.
- Keywords in the side-by-side manifest are embedded in the executable file.
- Keywords in specific StringTable entries are linked in the executable file.
- Key attributes in the resource script data are linked in the executable file.
- There are targeted sequences of bytes within the executable file.

NOTE

The keywords and sequences of bytes were derived from common characteristics observed from various installer technologies.

NOTE

The User Account Control: Detect application installations and prompt for elevation policy setting must be enabled for installer detection to detect installation programs. For more info, see [User Account Control security policy settings](#).

User Account Control security policy settings

7/1/2022 • 6 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016 and above

You can use security policies to configure how User Account Control works in your organization. They can be configured locally by using the Local Security Policy snap-in (secpol.msc) or configured for the domain, OU, or specific groups by Group Policy.

User Account Control: Admin Approval Mode for the Built-in Administrator account

This policy setting controls the behavior of Admin Approval Mode for the built-in Administrator account.

- **Enabled** The built-in Administrator account uses Admin Approval Mode. By default, any operation that requires elevation of privilege will prompt the user to approve the operation.
- **Disabled** (Default) The built-in Administrator account runs all applications with full administrative privilege.

User Account Control: Allow UIAccess application to prompt for elevation without using the secure desktop

This policy setting controls whether User Interface Accessibility (UIAccess or UIA) programs can automatically disable the secure desktop for elevation prompts used by a standard user.

- **Enabled** UIA programs, including Windows Remote Assistance, automatically disable the secure desktop for elevation prompts. If you do not disable the "User Account Control: Switch to the secure desktop when prompting for elevation" policy setting, the prompts appear on the interactive user's desktop instead of the secure desktop.
- **Disabled** (Default) The secure desktop can be disabled only by the user of the interactive desktop or by disabling the "User Account Control: Switch to the secure desktop when prompting for elevation" policy setting.

User Account Control: Behavior of the elevation prompt for administrators in Admin Approval Mode

This policy setting controls the behavior of the elevation prompt for administrators.

- **Elevate without prompting** Allows privileged accounts to perform an operation that requires elevation without requiring consent or credentials.

Note: Use this option only in the most constrained environments.

- **Prompt for credentials on the secure desktop** When an operation requires elevation of privilege, the user is prompted on the secure desktop to enter a privileged user name and password. If the user enters valid credentials, the operation continues with the user's highest available privilege.

- **Prompt for consent on the secure desktop** When an operation requires elevation of privilege, the user is prompted on the secure desktop to select either Permit or Deny. If the user selects Permit, the operation continues with the user's highest available privilege.
- **Prompt for credentials** When an operation requires elevation of privilege, the user is prompted to enter an administrative user name and password. If the user enters valid credentials, the operation continues with the applicable privilege.
- **Prompt for consent** When an operation requires elevation of privilege, the user is prompted to select either Permit or Deny. If the user selects Permit, the operation continues with the user's highest available privilege.
- **Prompt for consent for non-Windows binaries** (Default) When an operation for a non-Microsoft application requires elevation of privilege, the user is prompted on the secure desktop to select either Permit or Deny. If the user selects Permit, the operation continues with the user's highest available privilege.

User Account Control: Behavior of the elevation prompt for standard users

This policy setting controls the behavior of the elevation prompt for standard users.

- **Prompt for credentials** (Default) When an operation requires elevation of privilege, the user is prompted to enter an administrative user name and password. If the user enters valid credentials, the operation continues with the applicable privilege.
- **Automatically deny elevation requests** When an operation requires elevation of privilege, a configurable access denied error message is displayed. An enterprise that is running desktops as standard user may choose this setting to reduce help desk calls.
- **Prompt for credentials on the secure desktop** When an operation requires elevation of privilege, the user is prompted on the secure desktop to enter a different user name and password. If the user enters valid credentials, the operation continues with the applicable privilege.

User Account Control: Detect application installations and prompt for elevation

This policy setting controls the behavior of application installation detection for the computer.

- **Enabled** (Default) When an app installation package is detected that requires elevation of privilege, the user is prompted to enter an administrative user name and password. If the user enters valid credentials, the operation continues with the applicable privilege.
- **Disabled** App installation packages are not detected and prompted for elevation. Enterprises that are running standard user desktops and use delegated installation technologies, such as Group Policy or Microsoft Endpoint Manager should disable this policy setting. In this case, installer detection is unnecessary.

User Account Control: Only elevate executable files that are signed and validated

This policy setting enforces public key infrastructure (PKI) signature checks for any interactive applications that request elevation of privilege. Enterprise administrators can control which applications are allowed to run by adding certificates to the Trusted Publishers certificate store on local computers.

- **Enabled** Enforces the certificate certification path validation for a given executable file before it is permitted to run.
- **Disabled** (Default) Does not enforce the certificate certification path validation before a given executable file

is permitted to run.

User Account Control: Only elevate UIAccess applications that are installed in secure locations

This policy setting controls whether applications that request to run with a User Interface Accessibility (UIAccess) integrity level must reside in a secure location in the file system. Secure locations are limited to the following: - ...\\Program Files\\, including subfolders - ...\\Windows\\system32\\ - ...\\Program Files (x86)\\, including subfolders for 64-bit versions of Windows

Note: Windows enforces a digital signature check on any interactive app that requests to run with a UIAccess integrity level regardless of the state of this security setting.

- **Enabled** (Default) If an app resides in a secure location in the file system, it runs only with UIAccess integrity.
- **Disabled** An app runs with UIAccess integrity even if it does not reside in a secure location in the file system.

User Account Control: Turn on Admin Approval Mode

This policy setting controls the behavior of all User Account Control (UAC) policy settings for the computer. If you change this policy setting, you must restart your computer.

- **Enabled** (Default) Admin Approval Mode is enabled. This policy must be enabled and related UAC policy settings must also be set appropriately to allow the built-in Administrator account and all other users who are members of the Administrators group to run in Admin Approval Mode.
- **Disabled** Admin Approval Mode and all related UAC policy settings are disabled. Note: If this policy setting is disabled, the Windows Security app notifies you that the overall security of the operating system has been reduced.

User Account Control: Switch to the secure desktop when prompting for elevation

This policy setting controls whether the elevation request prompt is displayed on the interactive user's desktop or the secure desktop.

- **Enabled** (Default) All elevation requests go to the secure desktop regardless of prompt behavior policy settings for administrators and standard users.
- **Disabled** All elevation requests go to the interactive user's desktop. Prompt behavior policy settings for administrators and standard users are used.

User Account Control: Virtualize file and registry write failures to per-user locations

This policy setting controls whether application write failures are redirected to defined registry and file system locations. This policy setting mitigates applications that run as administrator and write run-time application data to %ProgramFiles%, %Windir%, %Windir%\\system32, or HKLM\\Software.

- **Enabled** (Default) App write failures are redirected at run time to defined user locations for both the file system and registry.
- **Disabled** Apps that write data to protected locations fail.

User Account Control Group Policy and registry key settings

7/1/2022 • 12 minutes to read • [Edit Online](#)

Applies to

- Windows 10
- Windows 11
- Windows Server 2016 and above

Group Policy settings

There are 10 Group Policy settings that can be configured for User Account Control (UAC). The table lists the default for each of the policy settings, and the following sections explain the different UAC policy settings and provide recommendations. These policy settings are located in **Security Settings\Local Policies\Security Options** in the Local Security Policy snap-in. For more information about each of the Group Policy settings, see the Group Policy description. For information about the registry key settings, see [Registry key settings](#).

GROUP POLICY SETTING	REGISTRY KEY	DEFAULT
User Account Control: Admin Approval Mode for the built-in Administrator account	FilterAdministratorToken	Disabled
User Account Control: Allow UIAccess applications to prompt for elevation without using the secure desktop	EnableUIADesktopToggle	Disabled
User Account Control: Behavior of the elevation prompt for administrators in Admin Approval Mode	ConsentPromptBehaviorAdmin	Prompt for consent for non-Windows binaries
User Account Control: Behavior of the elevation prompt for standard users	ConsentPromptBehaviorUser	Prompt for credentials
User Account Control: Detect application installations and prompt for elevation	EnableInstallerDetection	Enabled (default for home) Disabled (default for enterprise)
User Account Control: Only elevate executables that are signed and validated	ValidateAdminCodeSignatures	Disabled
User Account Control: Only elevate UIAccess applications that are installed in secure locations	EnableSecureUIAPaths	Enabled
User Account Control: Run all administrators in Admin Approval Mode	EnableLUA	Enabled

GROUP POLICY SETTING	REGISTRY KEY	DEFAULT
User Account Control: Switch to the secure desktop when prompting for elevation	PromptOnSecureDesktop	Enabled
User Account Control: Virtualize file and registry write failures to per-user locations	EnableVirtualization	Enabled

User Account Control: Admin Approval Mode for the built-in Administrator account

The **User Account Control: Admin Approval Mode for the built-in Administrator account** policy setting controls the behavior of Admin Approval Mode for the built-in Administrator account.

The options are:

- **Enabled.** The built-in Administrator account uses Admin Approval Mode. By default, any operation that requires elevation of privilege will prompt the user to approve the operation.
- **Disabled.** (Default) The built-in Administrator account runs all applications with full administrative privilege.

User Account Control: Allow UIAccess applications to prompt for elevation without using the secure desktop

The **User Account Control: Allow UIAccess applications to prompt for elevation without using the secure desktop** policy setting controls whether User Interface Accessibility (UIAccess or UIA) programs can automatically disable the secure desktop for elevation prompts used by a standard user.

The options are:

- **Enabled.** UIA programs, including Windows Remote Assistance, automatically disable the secure desktop for elevation prompts. If you do not disable the **User Account Control: Switch to the secure desktop when prompting for elevation** policy setting, the prompts appear on the interactive user's desktop instead of the secure desktop.
- **Disabled.** (Default) The secure desktop can be disabled only by the user of the interactive desktop or by disabling the **User Account Control: Switch to the secure desktop when prompting for elevation** policy setting.

UIA programs are designed to interact with Windows and application programs on behalf of a user. This policy setting allows UIA programs to bypass the secure desktop to increase usability in certain cases; however, allowing elevation requests to appear on the interactive desktop instead of the secure desktop can increase your security risk.

UIA programs must be digitally signed because they must be able to respond to prompts regarding security issues, such as the UAC elevation prompt. By default, UIA programs are run only from the following protected paths:

- ...\\Program Files, including subfolders
- ...\\Program Files (x86), including subfolders for 64-bit versions of Windows
- ...\\Windows\\System32

The **User Account Control: Only elevate UIAccess applications that are installed in secure locations** policy setting disables the requirement to be run from a protected path.

While this policy setting applies to any UIA program, it is primarily used in certain remote assistance scenarios, including the Windows Remote Assistance program in Windows 7.

If a user requests remote assistance from an administrator and the remote assistance session is established, any elevation prompts appear on the interactive user's secure desktop and the administrator's remote session is

paused. To avoid pausing the remote administrator's session during elevation requests, the user may select the **Allow IT Expert to respond to User Account Control prompts** check box when setting up the remote assistance session. However, selecting this check box requires that the interactive user respond to an elevation prompt on the secure desktop. If the interactive user is a standard user, the user does not have the required credentials to allow elevation.

If you enable this policy setting, requests for elevation are automatically sent to the interactive desktop (not the secure desktop) and also appear on the remote administrator's view of the desktop during a remote assistance session. This allows the remote administrator to provide the appropriate credentials for elevation.

This policy setting does not change the behavior of the UAC elevation prompt for administrators.

If you plan to enable this policy setting, you should also review the effect of the **User Account Control: Behavior of the elevation prompt for standard users** policy setting. If it is configured as **Automatically deny elevation requests**, elevation requests are not presented to the user.

User Account Control: Behavior of the elevation prompt for administrators in Admin Approval Mode

The **User Account Control: Behavior of the elevation prompt for administrators in Admin Approval Mode** policy setting controls the behavior of the elevation prompt for administrators.

The options are:

- **Elevate without prompting.** Allows privileged accounts to perform an operation that requires elevation without requiring consent or credentials.

Note Use this option only in the most constrained environments.
- **Prompt for credentials on the secure desktop.** When an operation requires elevation of privilege, the user is prompted on the secure desktop to enter a privileged user name and password. If the user enters valid credentials, the operation continues with the user's highest available privilege.
- **Prompt for consent on the secure desktop.** When an operation requires elevation of privilege, the user is prompted on the secure desktop to select either **Permit** or **Deny**. If the user selects **Permit**, the operation continues with the user's highest available privilege.
- **Prompt for credentials.** When an operation requires elevation of privilege, the user is prompted to enter an administrative user name and password. If the user enters valid credentials, the operation continues with the applicable privilege.
- **Prompt for consent.** When an operation requires elevation of privilege, the user is prompted to select either **Permit** or **Deny**. If the user selects **Permit**, the operation continues with the user's highest available privilege.
- **Prompt for consent for non-Windows binaries.** (Default) When an operation for a non-Microsoft application requires elevation of privilege, the user is prompted on the secure desktop to select either **Permit** or **Deny**. If the user selects **Permit**, the operation continues with the user's highest available privilege.

User Account Control: Behavior of the elevation prompt for standard users

The **User Account Control: Behavior of the elevation prompt for standard users** policy setting controls the behavior of the elevation prompt for standard users.

The options are:

- **Automatically deny elevation requests.** When an operation requires elevation of privilege, a configurable access denied error message is displayed. An enterprise that is running desktops as standard user may choose this setting to reduce help desk calls.
- **Prompt for credentials on the secure desktop.** When an operation requires elevation of privilege, the

user is prompted on the secure desktop to enter a different user name and password. If the user enters valid credentials, the operation continues with the applicable privilege.

- **Prompt for credentials.** (Default) When an operation requires elevation of privilege, the user is prompted to enter an administrative user name and password. If the user enters valid credentials, the operation continues with the applicable privilege.

User Account Control: Detect application installations and prompt for elevation

The **User Account Control: Detect application installations and prompt for elevation** policy setting controls the behavior of application installation detection for the computer.

The options are:

- **Enabled.** (Default for home) When an application installation package is detected that requires elevation of privilege, the user is prompted to enter an administrative user name and password. If the user enters valid credentials, the operation continues with the applicable privilege.
- **Disabled.** (Default for enterprise) Application installation packages are not detected and prompted for elevation. Enterprises that are running standard user desktops and use delegated installation technologies such as Group Policy Software Installation or Systems Management Server (SMS) should disable this policy setting. In this case, installer detection is unnecessary.

User Account Control: Only elevate executables that are signed and validated

The **User Account Control: Only elevate executables that are signed and validated** policy setting enforces public key infrastructure (PKI) signature checks for any interactive applications that request elevation of privilege. Enterprise administrators can control which applications are allowed to run by adding certificates to the Trusted Publishers certificate store on local computers.

The options are:

- **Enabled.** Enforces the PKI certification path validation for a given executable file before it is permitted to run.
- **Disabled.** (Default) Does not enforce PKI certification path validation before a given executable file is permitted to run.

User Account Control: Only elevate UIAccess applications that are installed in secure locations

The **User Account Control: Only elevate UIAccess applications that are installed in secure locations** policy setting controls whether applications that request to run with a User Interface Accessibility (UIAccess) integrity level must reside in a secure location in the file system. Secure locations are limited to the following:

- ...\\Program Files, including subfolders
- ...\\Windows\\system32
- ...\\Program Files (x86), including subfolders for 64-bit versions of Windows

Note Windows enforces a PKI signature check on any interactive application that requests to run with a UIAccess integrity level regardless of the state of this security setting.

The options are:

- **Enabled.** (Default) If an application resides in a secure location in the file system, it runs only with UIAccess integrity.
- **Disabled.** An application runs with UIAccess integrity even if it does not reside in a secure location in the file system.

User Account Control: Run all administrators in Admin Approval Mode

The **User Account Control: Run all administrators Admin Approval Mode** policy setting controls the behavior of all UAC policy settings for the computer. If you change this policy setting, you must restart your computer.

The options are:

- **Enabled.** (Default) Admin Approval Mode is enabled. This policy must be enabled and related UAC policy settings must also be set appropriately to allow the built-in Administrator account and all other users who are members of the **Administrators** group to run in Admin Approval Mode.
- **Disabled.** Admin Approval Mode and all related UAC policy settings are disabled.

Note If this policy setting is disabled, the Windows Security app notifies you that the overall security of the operating system has been reduced.

User Account Control: Switch to the secure desktop when prompting for elevation

The **User Account Control: Switch to the secure desktop when prompting for elevation** policy setting controls whether the elevation request prompt is displayed on the interactive user's desktop or the secure desktop.

The options are:

- **Enabled.** (Default) All elevation requests go to the secure desktop regardless of prompt behavior policy settings for administrators and standard users.
- **Disabled.** All elevation requests go to the interactive user's desktop. Prompt behavior policy settings for administrators and standard users are used.

When this policy setting is enabled, it overrides the **User Account Control: Behavior of the elevation prompt for administrators in Admin Approval Mode** policy setting. The following table describes the behavior of the elevation prompt for each of the administrator policy settings when the **User Account Control: Switch to the secure desktop when prompting for elevation** policy setting is enabled or disabled.

ADMINISTRATOR POLICY SETTING	ENABLED	DISABLED
Prompt for credentials on the secure desktop	The prompt appears on the secure desktop.	The prompt appears on the secure desktop.
Prompt for consent on the secure desktop	The prompt appears on the secure desktop.	The prompt appears on the secure desktop.
Prompt for credentials	The prompt appears on the secure desktop.	The prompt appears on the interactive user's desktop.
Prompt for consent	The prompt appears on the secure desktop.	The prompt appears on the interactive user's desktop.
Prompt for consent for non-Windows binaries	The prompt appears on the secure desktop.	The prompt appears on the interactive user's desktop.

When this policy setting is enabled, it overrides the **User Account Control: Behavior of the elevation prompt for standard users** policy setting. The following table describes the behavior of the elevation prompt for each of the standard user policy settings when the **User Account Control: Switch to the secure desktop when prompting for elevation** policy setting is enabled or disabled.

STANDARD POLICY SETTING	ENABLED	DISABLED
Automatically deny elevation requests	No prompt. The request is automatically denied.	No prompt. The request is automatically denied.

STANDARD POLICY SETTING	ENABLED	DISABLED
Prompt for credentials on the secure desktop	The prompt appears on the secure desktop.	The prompt appears on the secure desktop.
Prompt for credentials	The prompt appears on the secure desktop.	The prompt appears on the interactive user's desktop.

User Account Control: Virtualize file and registry write failures to per-user locations

The **User Account Control: Virtualize file and registry write failures to per-user locations** policy setting controls whether application write failures are redirected to defined registry and file system locations. This policy setting mitigates applications that run as administrator and write run-time application data to %ProgramFiles%, %Windir%, %Windir%\system32, or HKLM\Software.

The options are:

- **Enabled.** (Default) Application write failures are redirected at run time to defined user locations for both the file system and registry.
- **Disabled.** Applications that write data to protected locations fail.

Registry key settings

The registry keys are found in

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System. For information about each of the registry keys, see the associated Group Policy description.

REGISTRY KEY	GROUP POLICY SETTING	REGISTRY SETTING
FilterAdministratorToken	User Account Control: Admin Approval Mode for the built-in Administrator account	0 (Default) = Disabled 1 = Enabled
EnableUIADesktopToggle	User Account Control: Allow UIAccess applications to prompt for elevation without using the secure desktop	0 (Default) = Disabled 1 = Enabled
ConsentPromptBehaviorAdmin	User Account Control: Behavior of the elevation prompt for administrators in Admin Approval Mode	0 = Elevate without prompting 1 = Prompt for credentials on the secure desktop 2 = Prompt for consent on the secure desktop 3 = Prompt for credentials 4 = Prompt for consent 5 (Default) = Prompt for consent for non-Windows binaries
ConsentPromptBehaviorUser	User Account Control: Behavior of the elevation prompt for standard users	0 = Automatically deny elevation requests 1 = Prompt for credentials on the secure desktop 3 (Default) = Prompt for credentials
EnableInstallerDetection	User Account Control: Detect application installations and prompt for elevation	1 = Enabled (default for home) 0 = Disabled (default for enterprise)

REGISTRY KEY	GROUP POLICY SETTING	REGISTRY SETTING
ValidateAdminCodeSignatures	User Account Control: Only elevate executables that are signed and validated	0 (Default) = Disabled 1 = Enabled
EnableSecureUIAPaths	User Account Control: Only elevate UIAccess applications that are installed in secure locations	0 = Disabled 1 (Default) = Enabled
EnableLUA	User Account Control: Run all administrators in Admin Approval Mode	0 = Disabled 1 (Default) = Enabled
PromptOnSecureDesktop	User Account Control: Switch to the secure desktop when prompting for elevation	0 = Disabled 1 (Default) = Enabled
EnableVirtualization	User Account Control: Virtualize file and registry write failures to per-user locations	0 = Disabled 1 (Default) = Enabled

Smart Card Technical Reference

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies To: Windows 10, Windows 11, Windows Server 2016 and above

The Smart Card Technical Reference describes the Windows smart card infrastructure for physical smart cards and how smart card-related components work in Windows. This document also contains information about tools that information technology (IT) developers and administrators can use to troubleshoot, debug, and deploy smart card-based strong authentication in the enterprise.

Audience

This document explains how the Windows smart card infrastructure works. To understand this information, you should have basic knowledge of public key infrastructure (PKI) and smart card concepts. This document is intended for:

- Enterprise IT developers, managers, and staff who are planning to deploy or are using smart cards in their organization.
- Smart card vendors who write smart card minidrivers or credential providers.

What are smart cards?

Smart cards are tamper-resistant portable storage devices that can enhance the security of tasks such as authenticating clients, signing code, securing e-mail, and signing in with a Windows domain account.

Smart cards provide:

- Tamper-resistant storage for protecting private keys and other forms of personal information.
- Isolation of security-critical computations that involve authentication, digital signatures, and key exchange from other parts of the computer. These computations are performed on the smart card.
- Portability of credentials and other private information between computers at work, home, or on the road.

Smart cards can be used to sign in to domain accounts only, not local accounts. When you use a password to sign in interactively to a domain account, Windows uses the Kerberos version 5 (v5) protocol for authentication. If you use a smart card, the operating system uses Kerberos v5 authentication with X.509 v3 certificates.

Virtual smart cards were introduced in Windows Server 2012 and Windows 8 to alleviate the need for a physical smart card, the smart card reader, and the associated administration of that hardware. For information about virtual smart card technology, see [Virtual Smart Card Overview](#).

In this technical reference

This reference contains the following topics.

- [How Smart Card Sign-in Works in Windows](#)
 - [Smart Card Architecture](#)
 - [Certificate Requirements and Enumeration](#)
 - [Smart Card and Remote Desktop Services](#)

- [Smart Cards for Windows Service](#)
- [Certificate Propagation Service](#)
- [Smart Card Removal Policy Service](#)
- [Smart Card Tools and Settings](#)
 - [Smart Cards Debugging Information](#)
 - [Smart Card Group Policy and Registry Settings](#)
 - [Smart Card Events](#)

How Smart Card Sign-in Works in Windows

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies To: Windows 10, Windows 11, Windows Server 2016 and above

This topic for IT professional provides links to resources about the implementation of smart card technologies in the Windows operating system. It includes the following resources about the architecture, certificate management, and services that are related to smart card use:

- [Smart Card Architecture](#): Learn about enabling communications with smart cards and smart card readers, which can be different according to the vendor that supplies them.
- [Certificate Requirements and Enumeration](#): Learn about requirements for smart card certificates based on the operating system, and about the operations that are performed by the operating system when a smart card is inserted into the computer.
- [Smart Card and Remote Desktop Services](#): Learn about using smart cards for remote desktop connections.
- [Smart Cards for Windows Service](#): Learn about how the Smart Cards for Windows service is implemented.
- [Certificate Propagation Service](#): Learn about how the certificate propagation service works when a smart card is inserted into a computer.
- [Smart Card Removal Policy Service](#): Learn about using Group Policy to control what happens when a user removes a smart card.

Smart Card Architecture

7/1/2022 • 19 minutes to read • [Edit Online](#)

Applies To: Windows 10, Windows 11, Windows Server 2016 and above

This topic for the IT professional describes the system architecture that supports smart cards in the Windows operating system, including credential provider architecture and the smart card subsystem architecture.

Authentication is a process for verifying the identity of an object or person. When you authenticate an object, such as a smart card, the goal is to verify that the object is genuine. When you authenticate a person, the goal is to verify that you are not dealing with an imposter.

In a networking context, authentication is the act of proving identity to a network application or resource. Typically, identity is proven by a cryptographic operation that uses a key only the user knows (such as with public key cryptography), or a shared key. The server side of the authentication exchange compares the signed data with a known cryptographic key to validate the authentication attempt. Storing the cryptographic keys in a secure central location makes the authentication process scalable and maintainable.

For smart cards, Windows supports a provider architecture that meets the secure authentication requirements and is extensible so that you can include custom credential providers. This topic includes information about:

- [Credential provider architecture](#)
- [Smart card subsystem architecture](#)

Credential provider architecture

The following table lists the components that are included in the interactive sign-in architecture of the Windows Server and Windows operating systems.

COMPONENT	DESCRIPTION
Winlogon	Provides an interactive sign-in infrastructure.
Logon UI	Provides interactive UI rendering.
Credential providers (password and smart card)	Describes credential information and serializing credentials.
Local Security Authority (LSA)	Processes sign-in credentials.
Authentication packages	Includes NTLM and the Kerberos protocol. Communicates with server authentication packages to authenticate users.

Interactive sign-in in Windows begins when the user presses CTRL+ALT+DEL. The CTRL+ALT+DEL key combination is called a secure attention sequence (SAS). To keep other programs and processes from using it, Winlogon registers this sequence during the boot process.

After receiving the SAS, the UI then generates the sign-in tile from the information received from the registered credential providers. The following graphic shows the architecture for credential providers in the Windows operating system.

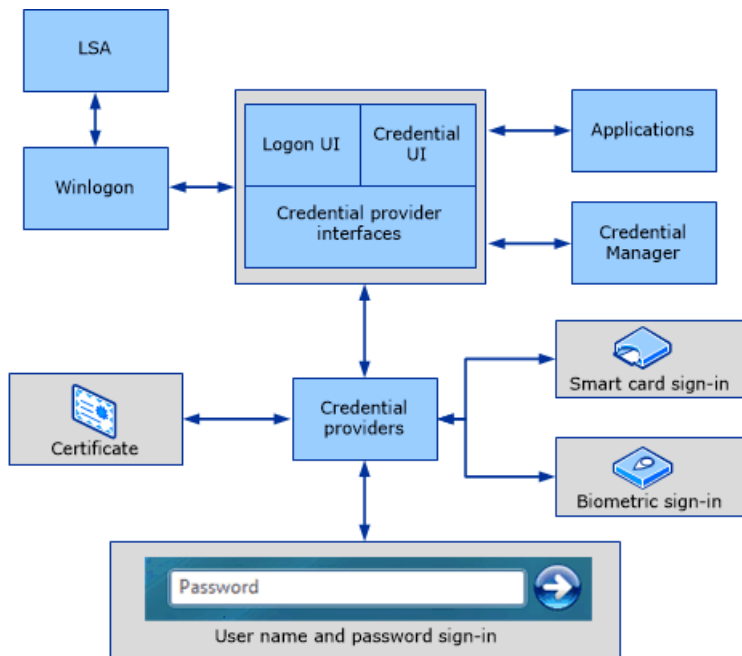


Figure 1 Credential provider architecture

Typically, a user who signs in to a computer by using a local account or a domain account must enter a user name and password. These credentials are used to verify the user's identity. For smart card sign-in, a user's credentials are contained on the smart card's security chip. A smart card reader lets the computer interact with the security chip on the smart card. When users sign in with a smart card, they enter a personal identification number (PIN) instead of a user name and password.

Credential providers are in-process COM objects that run on the local system and are used to collect credentials. The Logon UI provides interactive UI rendering, Winlogon provides interactive sign-in infrastructure, and credential providers work with both of these components to help gather and process credentials.

Winlogon instructs the Logon UI to display credential provider tiles after it receives an SAS event. The Logon UI queries each credential provider for the number of credentials it wants to enumerate. Credential providers have the option of specifying one of these tiles as the default. After all providers have enumerated their tiles, the Logon UI displays them to the user. The user interacts with a tile to supply the proper credentials. The Logon UI submits these credentials for authentication.

Combined with supporting hardware, credential providers can extend the Windows operating system to enable users to sign in by using biometrics (for example, fingerprint, retinal, or voice recognition), password, PIN, smart card certificate, or any custom authentication package. Enterprises and IT professionals can develop and deploy custom authentication mechanisms for all domain users, and they may explicitly require users to use this custom sign-in mechanism.

Note Credential providers are not enforcement mechanisms. They are used to gather and serialize credentials. The LSA and authentication packages enforce security.

Credential providers can be designed to support single sign-in (SSO). In this process, they authenticate users to a secure network access point (by using RADIUS and other technologies) for signing in to the computer. Credential providers are also designed to support application-specific credential gathering, and they can be used for authentication to network resources, joining computers to a domain, or to provide administrator consent for User Account Control (UAC).

Multiple credential providers can coexist on a computer.

Credential providers must be registered on a computer running Windows, and they are responsible for:

- Describing the credential information that is required for authentication.

- Handling communication and logic with external authentication authorities.
- Packaging credentials for interactive and network sign-in.

Note The Credential Provider API does not render the UI. It describes what needs to be rendered. Only the password credential provider is available in safe mode. The smart card credential provider is available in safe mode during networking.

Smart card subsystem architecture

Vendors provide smart cards and smart card readers, and in many cases the vendors are different for the smart card and the smart card reader. Drivers for smart card readers are written to the [Personal Computer/Smart Card \(PC/SC\) standard](#). Each smart card must have a Cryptographic Service Provider (CSP) that uses the CryptoAPI interfaces to enable cryptographic operations, and the WinSCard APIs to enable communications with smart card hardware.

Base CSP and smart card minidriver architecture

Figure 2 illustrates the relationship between the CryptoAPI, CSPs, the Smart Card Base Cryptographic Service Provider (Base CSP), and smart card minidrivers.

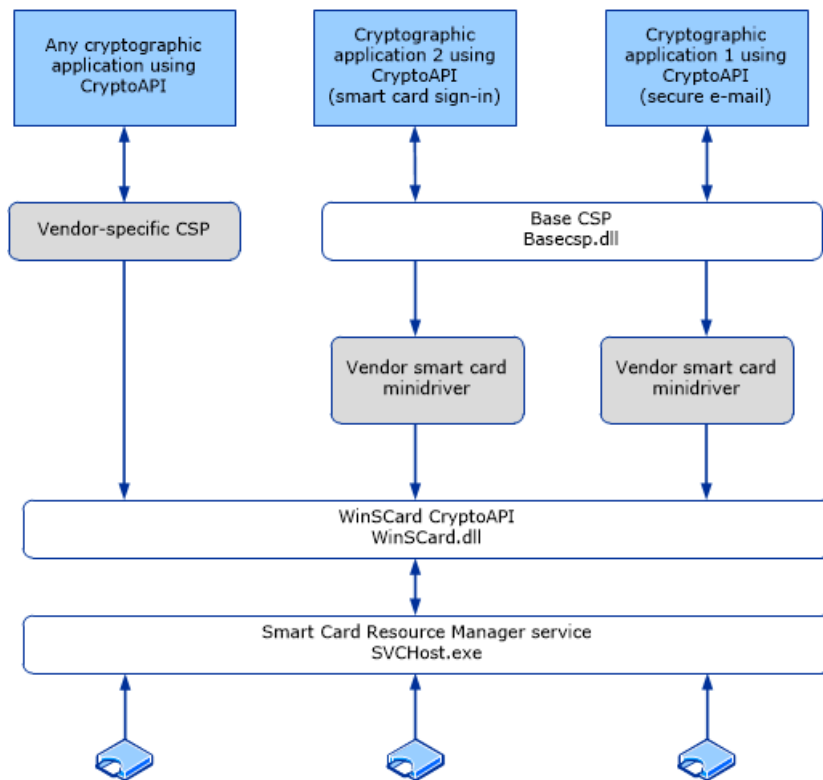


Figure 2 Base CSP and smart card minidriver architecture

Caching with Base CSP and smart card KSP

Smart card architecture uses caching mechanisms to assist in streamlining operations and to improve a user's access to a PIN.

- **Data caching:** The data cache provides for a single process to minimize smart card I/O operations.
- **PIN caching:** The PIN cache helps the user from having to reenter a PIN each time the smart card is unauthenticated.

Data caching

Each CSP implements the current smart card data cache separately. The Base CSP implements a robust caching mechanism that allows a single process to minimize smart card I/O operations.

The existing global cache works as follows:

1. The application requests a cryptographic operation. For example, a user certificate is to be read from the smart card.
2. The CSP checks its cache for the item.
3. If the item is not found in the cache, or if the item is cached but is not up-to-date, the item is read from the smart card.
4. After any item has been read from the smart card, it is added to the cache. Any existing out-of-date copy of that item is replaced.

Three types of objects or data are cached by the CSP: pins (for more information, see [PIN caching](#)), certificates, and files. If any of the cached data changes, the corresponding object is read from the smart card in successive operations. For example, if a file is written to the smart card, the CSP cache becomes out-of-date for the files, and other processes read the smart card at least once to refresh their CSP cache.

The global data cache is hosted in the Smart Cards for Windows service. Windows includes two public smart card API calls, `SCardWriteCache` and `SCardReadCache`. These API calls make global data caching functionality available to applications. Every smart card that conforms to the smart card minidriver specification has a 16-byte card identifier. This value is used to uniquely identify cached data that pertains to a given smart card. The standard Windows GUID type is used. These APIs allow an application to add data to and read data from the global cache.

PIN caching

The PIN cache protects the user from entering a PIN every time the smart card is unauthenticated. After a smart card is authenticated, it will not differentiate among host-side applications—any application can access private data on the smart card.

To mitigate this, the smart card enters an exclusive state when an application authenticates to the smart card. However, this means that other applications cannot communicate with the smart card and will be blocked. Therefore, such exclusive connections are minimized. The issue is that a protocol (such as the Kerberos protocol) requires multiple signing operations. Therefore, the protocol requires exclusive access to the smart card over an extended period, or it require multiple authentication operations. This is where the PIN cache is used to minimize exclusive use of the smart card without forcing the user to enter a PIN multiple times.

The following example illustrates how this works. In this scenario, there are two applications: Outlook and Internet Explorer. The applications use smart cards for different purposes.

1. The user starts Outlook and tries to send a signed e-mail. The private key is on the smart card.
2. Outlook prompts the user for the smart card PIN. The user enters the correct PIN.
3. E-mail data is sent to the smart card for the signature operation. The Outlook client formats the response and sends the e-mail.
4. The user opens Internet Explorer and tries to access a protected site that requires Transport Layer Security (TLS) authentication for the client.
5. Internet Explorer prompts the user for the smart card PIN. The user enters the correct PIN.
6. The TLS-related private key operation occurs on the smart card, and the user is authenticated and signed in.
7. The user returns to Outlook to send another signed e-mail. This time, the user is not prompted for a PIN because the PIN is cached from the previous operation. Similarly, if the user uses Internet Explorer again for another operation, Internet Explorer will not prompt the user for a PIN.

The Base CSP internally maintains a per-process cache of the PIN. The PIN is encrypted and stored in memory. The functions that are used to secure the PIN are RtlEncryptMemory, RtlDecryptMemory, and RtlSecureZeroMemory, which will empty buffers that contained the PIN.

Smart card selection

The following sections in this topic describe how Windows leverages the smart card architecture to select the correct smart card reader software, provider, and credentials for a successful smart card sign-in:

- [Container specification levels](#)
- [Container operations](#)
- [Context flags](#)
- [Create a new container in silent context](#)
- [Smart card selection behavior](#)
- [Make a smart card reader match](#)
- [Make a smart card match](#)
- [Open an existing default container \(no reader specified\)](#)
- [Open an existing GUID-named container \(no reader specified\)](#)
- [Create a new container \(no reader specified\)](#)
- [Delete a container](#)

Container specification levels

In response to a CryptAcquireContext call in CryptoAPI, the Base CSP tries to match the container that the caller specifies to a specific smart card and reader. The caller can provide a container name with varying levels of specificity, as shown in the following table, and sorted from most-specific to least-specific requests.

Similarly, in response to a NCryptOpenKey call in CNG, the smart card KSP tries to match the container the same way, and it takes the same container format, as shown in the following table.

Note Before opening a key by using the smart card KSP, a call to NCryptOpenStorageProvider (MS_SMART_CARD_KEY_STORAGE_PROVIDER) must be made.

TYPE	NAME	FORMAT
I	Reader Name and Container Name	\\.\<Reader Name>\<Container Name>
II	Reader Name and Container Name (NULL)	\\.\<Reader Name>
III	Container Name Only	<Container Name>
IV	Default Container (NULL) Only	NULL

The Base CSP and smart card KSP cache smart card handle information about the calling process and about the smart cards the process has accessed. When searching for a smart card container, the Base CSP or smart card KSP first checks its cache for the process. If the cached handle is invalid or no match is found, the SCardUIDlg API is called to get the card handle.

Container operations

The following three container operations can be requested by using CryptAcquireContext:

1. Create a new container. (The CNG equivalent of CryptAcquireContext with dwFlags set to CRYPT_NEWKEYSET is NCryptCreatePersistedKey.)
2. Open an existing container. (The CNG equivalent of CryptAcquireContext to open the container is NCryptOpenKey.)
3. Delete a container. (The CNG equivalent of CryptAcquireContext with dwFlags set to CRYPT_DELETEKEYSET is NCryptDeleteKey.)

The heuristics that are used to associate a cryptographic handle with a particular smart card and reader are based on the container operation requested and the level of container specification used.

The following table shows the restrictions for the container creation operation.

SPECIFICATION	RESTRICTION
No silent context	Key container creation must always be able to show UI, such as the PIN prompt.
No overwriting existing containers	If the specified container already exists on the chosen smart card, choose another smart card or cancel the operation.

Context flags

The following table shows the context flags used as restrictions for the container creation operation.

FLAG	DESCRIPTION
CRYPT_SILENT	No UI can be displayed during this operation.
CRYPT_MACHINE_KEYSET	No cached data should be used during this operation.
CRYPT_VERIFYCONTEXT	Only public data can be accessed on the smart card.

In addition to container operations and container specifications, you must consider other user options, such as the CryptAcquireContext flags, during smart card selection.

Important The CRYPT_SILENT flag cannot be used to create a new container.

Create a new container in silent context

Applications can call the Base CSP with CRYPT_DEFAULT_CONTAINER_OPTIONAL, set the PIN in silent context, and then create a new container in silent context. This operation occurs as follows:

1. Call CryptAcquireContext by passing the smart card reader name in as a type II container specification level, and specifying the CRYPT_DEFAULT_CONTAINER_OPTIONAL flag.
2. Call CryptSetProvParam by specifying PP_KEYEXCHANGE_PIN or PP_SIGNATURE_PIN and a null-terminated ASCII PIN.
3. Release the context acquired in Step 1.
4. Call CryptAcquireContext with CRYPT_NEWKEYSET, and specify the type I container specification level.
5. Call CryptGenKey to create the key.

Smart card selection behavior

In some of the following scenarios, the user can be prompted to insert a smart card. If the user context is silent,

this operation fails and no UI is displayed. Otherwise, in response to the UI, the user can insert a smart card or click **Cancel**. If the user cancels the operation, the operation fails. The flow chart in Figure 3 shows the selection steps performed by the Windows operating system.

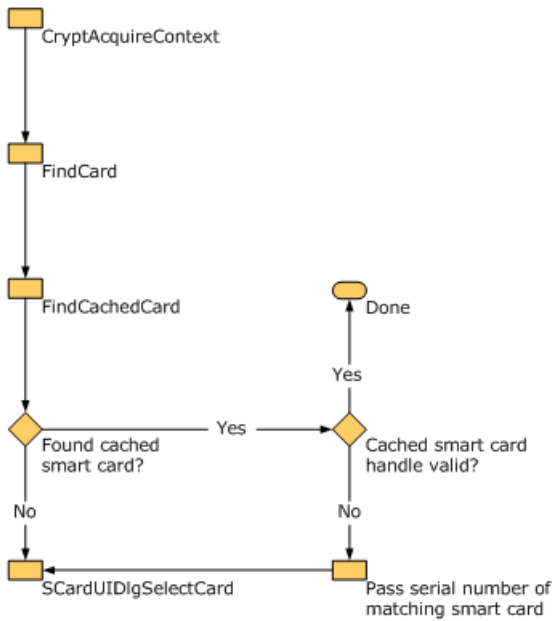


Figure 3 Smart card selection behavior

In general, smart card selection behavior is handled by the SCardUIDlgSelectCard API. The Base CSP interacts with this API by calling it directly. The Base CSP also sends callback functions that have the purpose of filtering and matching candidate smart cards. Callers of CryptAcquireContext provide smart card matching information. Internally, the Base CSP uses a combination of smart card serial numbers, reader names, and container names to find specific smart cards.

Each call to SCardUI * may result in additional information read from a candidate smart card. The Base CSP smart card selection callbacks cache this information.

Make a smart card reader match

For type I and type II container specification levels, the smart card selection process is less complex because only the smart card in the named reader can be considered a match. The process for matching a smart card with a smart card reader is:

1. Find the requested smart card reader. If it cannot be found, the process fails. (This requires a cache search by reader name.)
2. If no smart card is in the reader, the user is prompted to insert a smart card. (This is only in non-silent mode; if the call is made in silent mode, it will fail.)
3. For container specification level II only, the name of the default container on the chosen smart card is determined.
4. To open an existing container or delete an existing container, find the specified container. If the specified container cannot be found on this smart card, the user is prompted to insert a smart card.
5. If the system attempts to create a new container, if the specified container already exists on this smart card, the process fails.

Make a smart card match

For container specification levels III and IV, a broader method is used to match an appropriate smart card with a user context, because multiple cached smart cards might meet the criteria provided.

Open an existing default container (no reader specified)

Note This operation requires that you use the smart card with the Base CSP.

1. For each smart card that has been accessed by the Base CSP and the handle and container information are cached, the Base CSP looks for a valid default container. An operation is attempted on the cached SCARDHANDLE to verify its validity. If the smart card handle is not valid, the Base CSP continues to search for a new smart card.
2. If a matching smart card is not found in the Base CSP cache, the Base CSP calls to the smart card subsystem. SCardUIDlgSelectCard() is used with an appropriate callback filter to find a matching smart card with a valid default container.

Open an existing GUID-named container (no reader specified)

Note This operation requires that you use the smart card with the Base CSP.

1. For each smart card that is already registered with the Base CSP, search for the requested container. Attempt an operation on the cached SCARDHANDLE to verify its validity. If the smart card handle is not valid, the smart card's serial number is passed to the SCardUI * API to continue searching for this specific smart card (rather than only a general match for the container name).
2. If a matching smart card is not found in the Base CSP cache, a call is made to the smart card subsystem. SCardUIDlgSelectCard() is used with an appropriate callback filter to find a matching smart card with the requested container. Or, if a smart card serial number resulted from the search in Step 1, the callback filter attempts to match the serial number, not the container name.

Create a new container (no reader specified)

Note This operation requires that you use the smart card with the Base CSP.

If the PIN is not cached, no CRYPT_SILENT is allowed for the container creation because the user must be prompted for a PIN, at a minimum.

For other operations, the caller may be able to acquire a "verify" context against the default container (CRYPT_DEFAULT_CONTAINER_OPTIONAL) and then make a call with CryptSetProvParam to cache the user PIN for subsequent operations.

1. For each smart card already known by the CSP, refresh the stored SCARDHANDLE and make the following checks:
 - a. If the smart card has been removed, continue the search.
 - b. If the smart card is present, but it already has the named container, continue the search.
 - c. If the smart card is available, but a call to CardQueryFreeSpace indicates that the smart card has insufficient storage for an additional key container, continue the search.
 - d. Otherwise, use the first available smart card that meets the above criteria for the container creation.
2. If a matching smart card is not found in the CSP cache, make a call to the smart card subsystem. The callback that is used to filter enumerated smart cards verifies that a candidate smart card does not already have the named container, and that CardQueryFreeSpace indicates the smart card has sufficient space for an additional container. If no suitable smart card is found, the user is prompted to insert a smart card.

Delete a container

1. If the specified container name is NULL, the default container is deleted. Deleting the default container

causes a new default container to be selected arbitrarily. For this reason, this operation is not recommended.

2. For each smart card already known by the CSP, refresh the stored SCARDHANDLE and make the following checks:
 - a. If the smart card does not have the named container, continue the search.
 - b. If the smart card has the named container, but the smart card handle is no longer valid, store the serial number of the matching smart card and pass it to SCardUI *.
3. If a matching smart card is not found in the CSP cache, make a call to the smart card subsystem. The callback that is used to filter enumerated smart cards should verify that a candidate smart card has the named container. If a serial number was provided as a result of the previous cache search, the callback should filter enumerated smart cards on serial number rather than on container matches. If the context is non-silent and no suitable smart card is found, display UI that prompts the user to insert a smart card.

Base CSP and KSP-based architecture in Windows

Figure 4 shows the Cryptography architecture that is used by the Windows operating system.

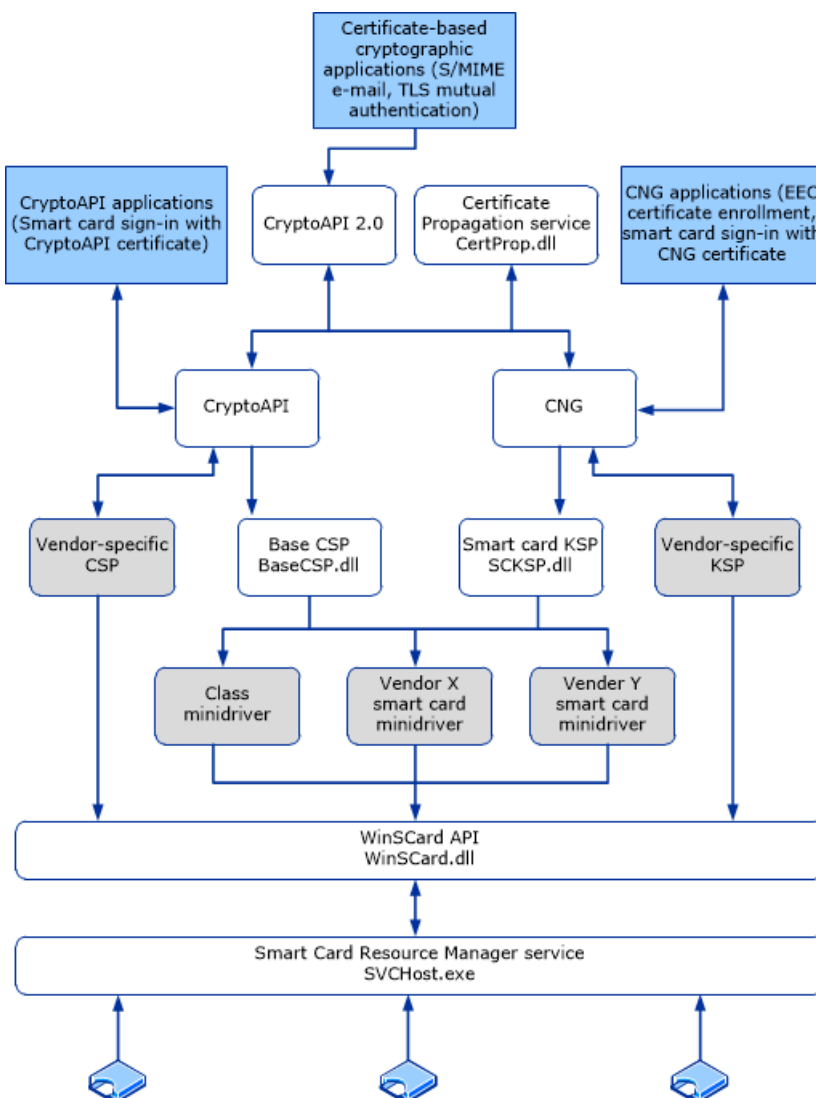


Figure 4 Cryptography architecture

Base CSP and smart card KSP properties in Windows

The following properties are supported in versions of Windows designated in the **Applies To** list at the beginning of this topic.

Note The API definitions are located in WinCrypt.h and WinSCard.h.

PROPERTY	DESCRIPTION
PP_USER_CERTSTORE	<ul style="list-style-type: none"> - Used to return an HCERTSTORE that contains all user certificates on the smart card - Read-only (used only by CryptGetProvParam) - Caller responsible for closing the certificate store - Certificate encoded using PKCS_7_ASN_ENCODING or X509_ASN_ENCODING - CSP should set KEY_PROV_INFO on certificates - Certificate store should be assumed to be an in-memory store - Certificates should have a valid CRYPT_KEY_PROV_INFO as a property
PP_ROOT_CERTSTORE	<ul style="list-style-type: none"> - Read and Write (used by CryptGetProvParam and CryptSetProvParam) - Used to write a collection of root certificates to the smart card or return HCERTSTORE, which contains root certificates from the smart card - Used primarily for joining a domain by using a smart card - Caller responsible for closing the certificate store
PP_SMARTCARD_READER	<ul style="list-style-type: none"> - Read-only (used only by CryptGetProvParam) - Returns the smart card reader name as an ANSI string that is used to construct a fully qualified container name (that is, a smart card reader plus a container)
PP_SMARTCARD_GUID	<ul style="list-style-type: none"> - Return smart card GUID (also known as a serial number), which should be unique for each smart card - Used by the certificate propagation service to track the source of a root certificate
PP_UI_PROMPT	<ul style="list-style-type: none"> - Used to set the search string for the SCardUIDlgSelectCard card insertion dialog box - Persistent for the entire process when it is set - Write-only (used only by CryptSetProvParam)

Implications for CSPs in Windows

Cryptographic Service Providers (CSPs), including custom smart card CSPs, continue to be supported but this approach is not recommended. Using the existing Base CSP and smart card KSP with the smart card minidriver model for smart cards provides significant benefits in terms of performance, and PIN and data caching. One minidriver can be configured to work under CryptoAPI and CNG layers. This provides benefits from enhanced cryptographic support, including elliptic curve cryptography and AES.

If a smart card is registered by a CSP and a smart card minidriver, the one that was installed most recently will be used to communicate with the smart card.

Write a smart card minidriver, CSP, or KSP

CSPs and KSPs are meant to be written only if specific functionality is not available in the current smart card minidriver architecture. For example, the smart card minidriver architecture supports hardware security modules, so a minidriver could be written for a hardware security module, and a CSP or KSP may not be required unless it is needed to support algorithms that are not implemented in the Base CSP or smart card KSP.

For more information about how to write a smart card minidriver, CSP, or KSP, see [Smart Card Minidrivers](#).

Certificate Requirements and Enumeration

7/1/2022 • 19 minutes to read • [Edit Online](#)

Applies To: Windows 10, Windows 11, Windows Server 2016 and above

This topic for the IT professional and smart card developers describes how certificates are managed and used for smart card sign-in.

When a smart card is inserted, the following steps are performed.

Note Unless otherwise mentioned, all operations are performed silently (CRYPT_SILENT is passed to CryptAcquireContext).

1. The smart card resource manager database searches for the smart card's cryptographic service provider (CSP).
2. A qualified container name is constructed by using the smart card reader name, and it is passed to the CSP. The format is `\\.\<Reader name>\`.
3. CryptAcquireContext is called to retrieve a context to the default container. If a failure occurs, the smart card will be unusable for smart card sign-in.
4. The name of the container is retrieved by using the PP_CONTAINER parameter with CryptGetProvParam.
5. Using the context acquired in Step 3, the CSP is queried for the PP_USER_CERTSTORE parameter (added in Windows Vista). For more information, see [Smart Card Architecture](#). If the operation is successful, the name of a certificate store is returned, and the program flow skips to Step 8.
6. If the operation in Step 5 fails, the default container context from Step 3 is queried for the AT_KEYEXCHANGE key.
7. The certificate is then queried from the key context by using KP_CERTIFICATE. The certificate is added to an in-memory certificate store.
8. For each certificate in the certificate store from Step 5 or Step 7, the following checks are performed:
 - a. The certificate must be valid, based on the computer system clock (not expired or valid with a future date).
 - b. The certificate must not be in the AT_SIGNATURE part of a container.
 - c. The certificate must have a valid user principal name (UPN).
 - d. The certificate must have the digital signature key usage.
 - e. The certificate must have the smart card logon EKU.

Any certificate that meets these requirements is displayed to the user with the certificate's UPN (or e-mail address or subject, depending on the presence of the certificate extensions).

Note These requirements are the same as those in Windows Server 2003, but they are performed before the user enters the PIN. You can override many of them by using Group Policy settings.

9. The process then chooses a certificate, and the PIN is entered.
10. LogonUI.exe packages the information and sends it to Lsass.exe to process the sign-in attempt.
11. If successful, LogonUI.exe closes. This causes the context acquired in Step 3 to be released.

About Certificate support for compatibility

Although versions of Windows earlier than Windows Vista include support for smart cards, the types of certificates that smart cards can contain are limited. The limitations are:

- Each certificate must have a user principal name (UPN) and the smart card sign-in object identifier (also known as OID) in the enhanced key usage (EKU) attribute field. There is a Group Policy setting, Allow ECC certificates to be used for logon and authentication, to make the EKU optional.
- Each certificate must be stored in the AT_KEYEXCHANGE portion of the default CryptoAPI container, and non-default CryptoAPI containers are not supported.

The following table lists the certificate support in older Windows operating system versions.

OPERATING SYSTEM	CERTIFICATE SUPPORT
Windows Server 2008 R2 and Windows 7	<p>Support for smart card sign-in with ECC-based certificates. ECC smart card sign-in is enabled through Group Policy.</p> <p>ECDH_P256 ECDH Curve P-256 from FIPS 186-2</p> <p>ECDSA_P256 ECDSA Curve P-256 from FIPS 186-2</p> <p>ECDH_P384 ECDH Curve P-384 from FIPS 186-2</p> <p>ECDH_P521 ECDH Curve P-521 from FIPS 186-2</p> <p>ECDSA_P256 ECDH Curve P-256 from FIPS 186-2</p> <p>ECDSA_P384 ECDSA Curve P-384 from FIPS 186-2</p> <p>ECDSA_P521 ECDSA Curve P-384 from FIPS 186-2</p>
Windows Server 2008 and Windows Vista	<p>Valid certificates are enumerated and displayed from all smart cards and presented to the user. Keys are no longer restricted to the default container, and certificates in different containers can be chosen. Elliptic curve cryptography (ECC)-based certificates are not supported for smart card sign-in</p>

Smart card sign-in flow in Windows

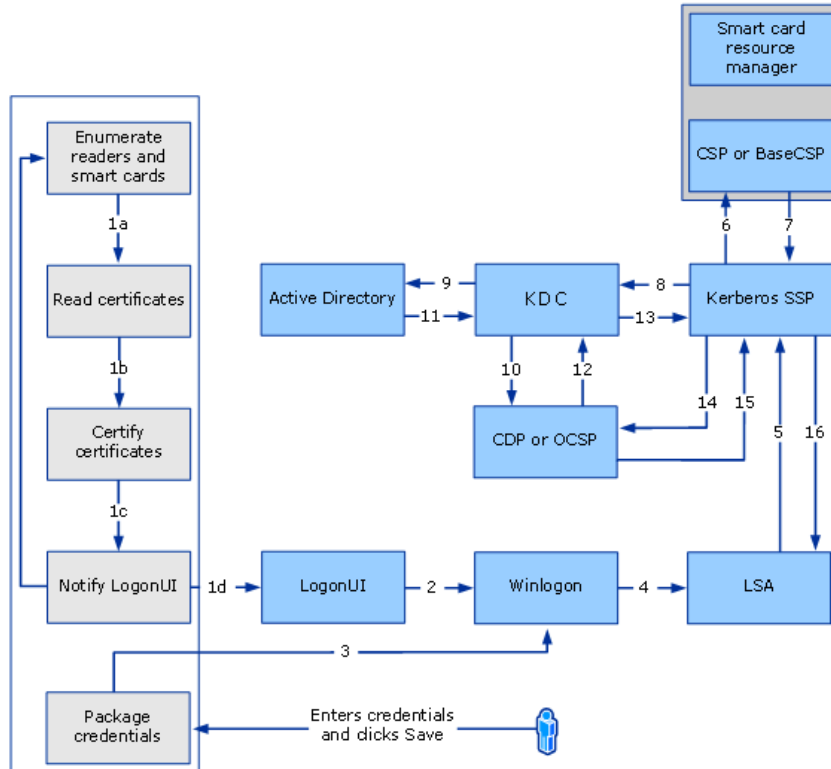
Most issues during authentication occur because of session behavior changes. When changes occur, the Local Security Authority (LSA) does not reacquire the session context; it relies instead on the Cryptographic Service Provider to handle the session change.

In the supported versions of Windows designated in the **Applies To** list at the beginning of this topic, client certificates that do not contain a UPN in the **subjectAltName** (SAN) field of the certificate can be enabled for sign-in, which supports a wider variety of certificates and supports multiple sign-in certificates on the same card.

Support for multiple certificates on the same card is enabled by default. New certificate types must be enabled through Group Policy.

If you enable the **Allow signature keys valid for Logon** credential provider policy, any certificates that are available on the smart card with a signature-only key are listed on the sign-in screen. This allows users to select their sign-in experience. If the policy is disabled or not configured, smart card signature-key-based certificates are not listed on the sign-in screen.

The following diagram illustrates how smart card sign-in works in the supported versions of Windows.



Smart card sign-in flow

Following are the steps that are performed during a smart card sign-in:

1. Winlogon requests the sign-in UI credential information.
2. Asynchronously, smart card resource manager starts, and the smart card credential provider does the following:
 - a. Gets credential information (a list of known credentials, or if no credentials exist, the smart card reader information that Windows detected).
 - b. Gets a list of smart card readers (by using the WinSCard API) and the list of smart cards inserted in each of them.
 - c. Enumerates each card to verify that a sign-in certificate that is controlled by Group Policy is present. If the certificate is present, the smart card credential provider copies it into a temporary, secure cache on the computer or terminal.

Note Smartcard cache entries are created for certificates with a subject name or with a subject key identifier. If the certificate has a subject name, it is stored with an index that is based on the subject name and certificate issuer. If another certificate with the same subject name and certificate issuer is used, it will replace the existing cached entry. A change in this behavior after Windows Vista, allows for the condition when the certificate does not have a subject name, the cache is created with an index that is based on the subject key identifier and certificate issuer. If another certificate has the same the subject key identifier and certificate issuer, the cache entry is replaced. When certificates have neither a subject name nor subject key identifier, a cached entry is not created.

- d. Notifies the sign-in UI that it has new credentials.
3. The sign-in UI requests the new credentials from the smart card credential provider. As a response, the smart card credential provider provides each sign-in certificate to the sign-in UI, and corresponding sign-in tiles are displayed. The user selects a smart card-based sign-in certificate tile, and Windows displays a PIN dialog box.
4. The user enters the PIN, and then presses ENTER. The smart card credential provider encrypts the PIN.
5. The credential provider that resides in the LogonUI system collects the PIN. As part of packaging credentials in the smart card credential provider, the data is packaged in a KERB_CERTIFICATE_LOGON structure. The main contents of the KERB_CERTIFICATE_LOGON structure are the smart card PIN, CSP data (such as reader name and container name), user name, and domain name. User name is required if the sign-in domain is not in the same forest because it enables a certificate to be mapped to multiple user accounts.
6. The credential provider wraps the data (such as the encrypted PIN, container name, reader name, and card key specification) and sends it back to LogonUI.
7. Winlogon presents the data from LogonUI to the LSA with the user information in LSALogonUser.
8. LSA calls the Kerberos authentication package (Kerberos SSP) to create a Kerberos authentication service request (KRB_AS_REQ), which containing a preauthenticator (as specified in RFC 4556: [Public Key Cryptography for Initial Authentication in Kerberos \(PKINIT\)](#)).

If the authentication is performed by using a certificate that uses a digital signature, the preauthentication data consists of the user's public certificate and the certificate that is digitally signed with the corresponding private key.

If the authentication is performed by using a certificate that uses key encipherment, the preauthentication data consists of the user's public certificate and the certificate that is encrypted with the corresponding private key.

9. To sign the request digitally (as per RFC 4556), a call is made to the corresponding CSP for a private key operation. Because the private key in this case is stored in a smart card, the smart card subsystem is called, and the necessary operation is completed. The result is sent back to the Kerberos security support provider (SSP).
10. The Kerberos SSP sends an authentication request for a ticket-granting-ticket (TGT) (per RFC 4556) to the Key Distribution Center (KDC) service that runs on a domain controller.
11. The KDC finds the user's account object in Active Directory Domain Services (AD DS), as detailed in [Client certificate requirements and mappings](#), and uses the user's certificate to verify the signature.
12. The KDC validates the user's certificate (time, path, and revocation status) to ensure that the certificate is from a trusted source. The KDC uses CryptoAPI to build a certification path from the user's certificate to a root certification authority (CA) certificate that resides in the root store on the domain controller. The KDC then uses CryptoAPI to verify the digital signature on the signed authenticator that was included in the preauthentication data fields. The domain controller verifies the signature and uses the public key from the user's certificate to prove that the request originated from the owner of the private key that corresponds to the public key. The KDC also verifies that the issuer is trusted and appears in the NTAUTH certificate store.
13. The KDC service retrieves user account information from AD DS. The KDC constructs a TGT, which is based on the user account information that it retrieves from AD DS. The TGT's authorization data fields include the user's security identifier (SID), the SIDs for universal and global domain groups to which the user belongs, and (in a multidomain environment) the SIDs for any universal groups of which the user is a member.
14. The domain controller returns the TGT to the client as part of the KRB_AS_REP response.

Note The KRB_AS_REP packet consists of:

- Privilege attribute certificate (PAC)

- User's SID
- SIDs of any groups of which the user is a member
- A request for ticket-granting service (TGS)
- Preauthentication data

TGT is encrypted with the master key of the KDC, and the session key is encrypted with a temporary key. This temporary key is derived based on RFC 4556. Using CryptoAPI, the temporary key is decrypted. As part of the decryption process, if the private key is on a smart card, a call is made to the smart card subsystem by using the specified CSP to extract the certificate corresponding to the user's public key. (Programmatic calls for the certificate include CryptAcquireContext, CryptSetProvParam with the PIN, CryptgetUserKey, and CryptGetKeyParam.) After the temporary key is obtained, the Kerberos SSP decrypts the session key.

15. The client validates the reply from the KDC (time, path, and revocation status). It first verifies the KDC's signature by the construction of a certification path from the KDC's certificate to a trusted root CA, and then it uses the KDC's public key to verify the reply signature.
16. Now that a TGT has been obtained, the client obtains a service ticket, which is used to sign in to the local computer.
17. With success, LSA stores the tickets and returns a success message to LSALogonUser. After this success message is issued, user profile for the device is selected and set, Group Policy refresh is instantiated, and other actions are performed.
18. After the user profile is loaded, the Certification Propagation Service (CertPropSvc) detects this event, reads the certificates from the smart card (including the root certificates), and then populates them into the user's certificate store (MYSTORE).
19. CSP to smart card resource manager communication happens on the LRPC Channel.
20. On successful authentication, certificates are propagated to the user's store asynchronously by the Certificate Propagation Service (CertPropSvc).
21. When the card is removed, certificates in the temporary secure cache store are removed. The Certificates are no longer available for sign-in, but they remain in the user's certificate store.

Note A SID is created for each user or group at the time a user account or a group account is created within the local security accounts database or within AD DS. The SID never changes, even if the user or group account is renamed.

For more information about the Kerberos protocol, see [Microsoft Kerberos](#).

By default, the KDC verifies that the client's certificate contains the smart card client authentication EKU szOID_KP_SMARTCARD_LOGON. However, if enabled, the **Allow certificates with no extended key usage certificate attribute** Group Policy setting allows the KDC to not require the SC-LOGON EKU. SC-LOGON EKU is not required for account mappings that are based on the public key.

KDC certificate

Active Directory Certificate Services provides three kinds of certificate templates:

- Domain controller
- Domain controller authentication
- Kerberos authentication

Depending on the configuration of the domain controller, one of these types of certificates is sent as a part of the AS_REP packet.

Client certificate requirements and mappings

Certificate requirements are listed by versions of the Windows operating system. Certificate mapping describes how information from the certificate is mapped to the user account.

Certificate requirements

The smart card certificate has specific format requirements when it is used with Windows XP and earlier operating systems. You can enable any certificate to be visible for the smart card credential provider.

COMPONENT	REQUIREMENTS FOR WINDOWS 8.1, WINDOWS 8, WINDOWS 7, WINDOWS VISTA, WINDOWS 10, AND WINDOWS 11	REQUIREMENTS FOR WINDOWS XP
CRL distribution point location	Not required	The location must be specified, online, and available, for example: [1]CRL Distribution Point Distribution Point Name: Full Name: URL= <https://server1.contoso.com/CertEnroll/caname.>
Key usage	Digital signature	Digital signature
Basic constraints	Not required	[Subject Type=End Entity, Path Length Constraint=None] (Optional)
Enhanced key usage (EKU)	The smart card sign-in object identifier is not required. Note If an EKU is present, it must contain the smart card sign-in EKU. Certificates with no EKU can be used for sign-in.	- Client Authentication (1.3.6.1.5.5.7.3.2) The client authentication object identifier is required only if a certificate is used for SSL authentication. - Smart Card Sign-in (1.3.6.1.4.1.311.20.2.2)
Subject alternative name	E-mail ID is not required for smart card sign-in.	Other Name: Principal Name=(UPN), for example: UPN=user1@contoso.com The UPN OtherName object identifier is 1.3.6.1.4.1.311.20.2.3. The UPN OtherName value must be an ASN1-encoded UTF8 string.
Subject	Not required	Distinguished name of user. This field is a mandatory extension, but the population of this field is optional.
Key exchange (AT_KEYEXCHANGE field)	Not required for smart card sign-in certificates if a Group Policy setting is enabled. (By default, Group Policy settings are not enabled.)	Not required
CRL	Not required	Not required
UPN	Not required	Not required

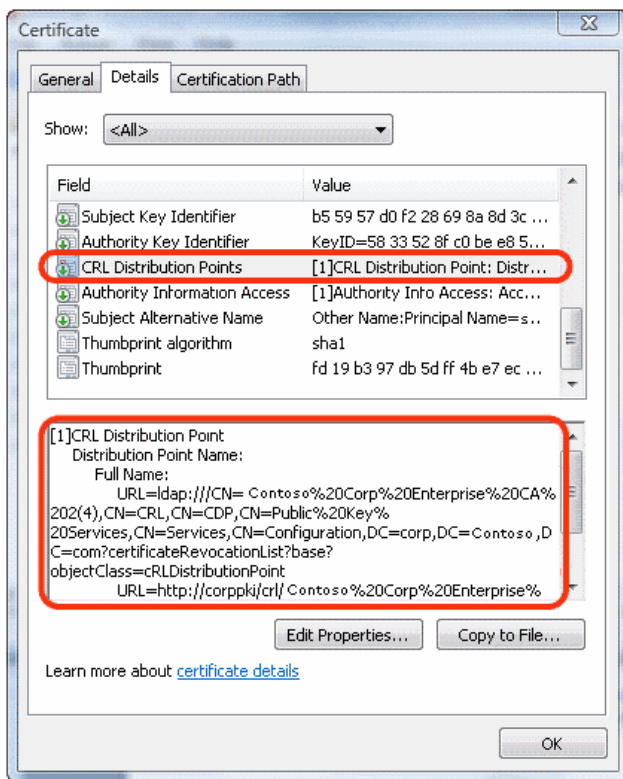
COMPONENT	REQUIREMENTS FOR WINDOWS 8.1, WINDOWS 8, WINDOWS 7, WINDOWS VISTA, WINDOWS 10, AND WINDOWS 11	REQUIREMENTS FOR WINDOWS XP
Notes	You can enable any certificate to be visible for the smart card credential provider.	There are two predefined types of private keys. These keys are Signature Only (AT_SIGNATURE) and Key Exchange (AT_KEYEXCHANGE). Smart card sign-in certificates must have a Key Exchange (AT_KEYEXCHANGE) private key type.

Client certificate mappings

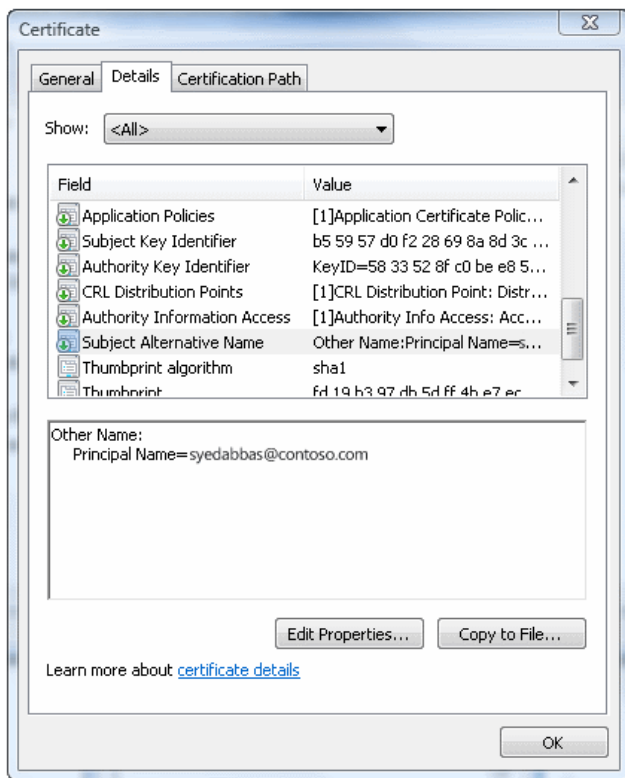
Certificate mapping is based on the UPN that is contained in the subjectAltName (SAN) field of the certificate. Client certificates that do not contain information in the SAN field are also supported.

SSL/TLS can map certificates that do not have SAN, and the mapping is done by using the AltSecID attributes on client accounts. The X509 AltSecID, which is used by SSL/TLS client authentication is of the form "X509:<I>"<Issuer Name>"<S>"<Subject Name>. The <Issuer Name> and <Subject Name> are taken from the client certificate, with '\r' and '\n' replaced with ','.

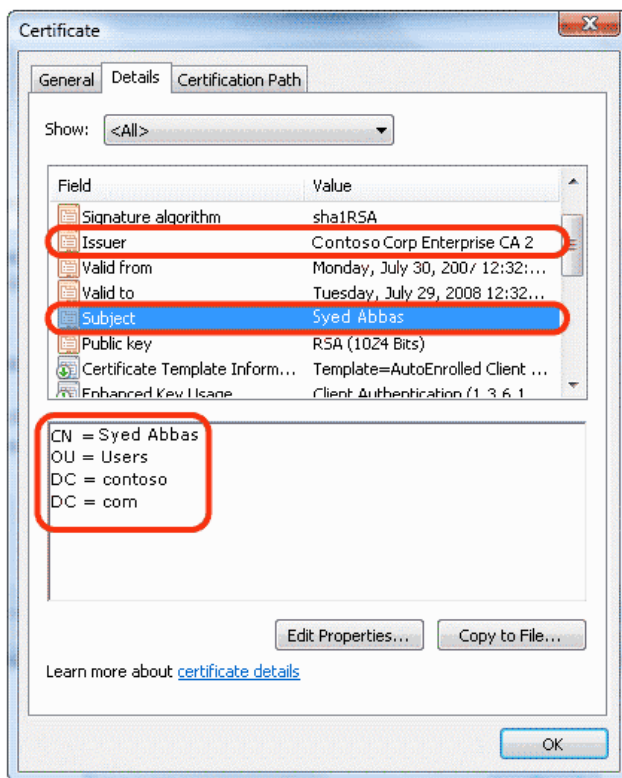
Certificate revocation list distribution points



UPN in Subject Alternative Name field

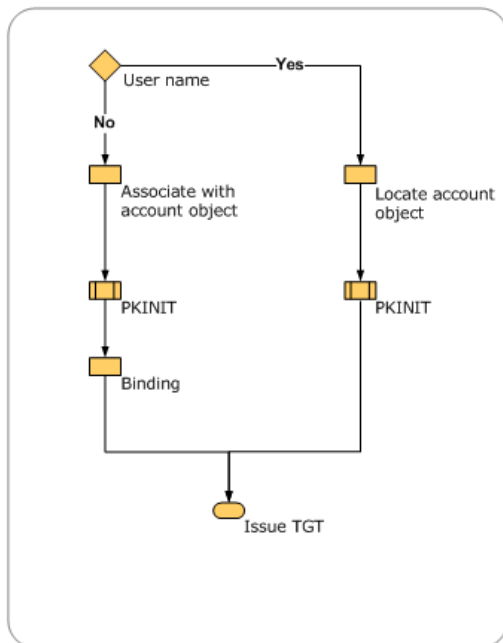


Subject and Issuer fields



This account mapping is supported by the KDC in addition to six other mapping methods. The following figure demonstrates a flow of user account mapping logic that is used by the KDC.

High-level flow of certificate processing for sign-in



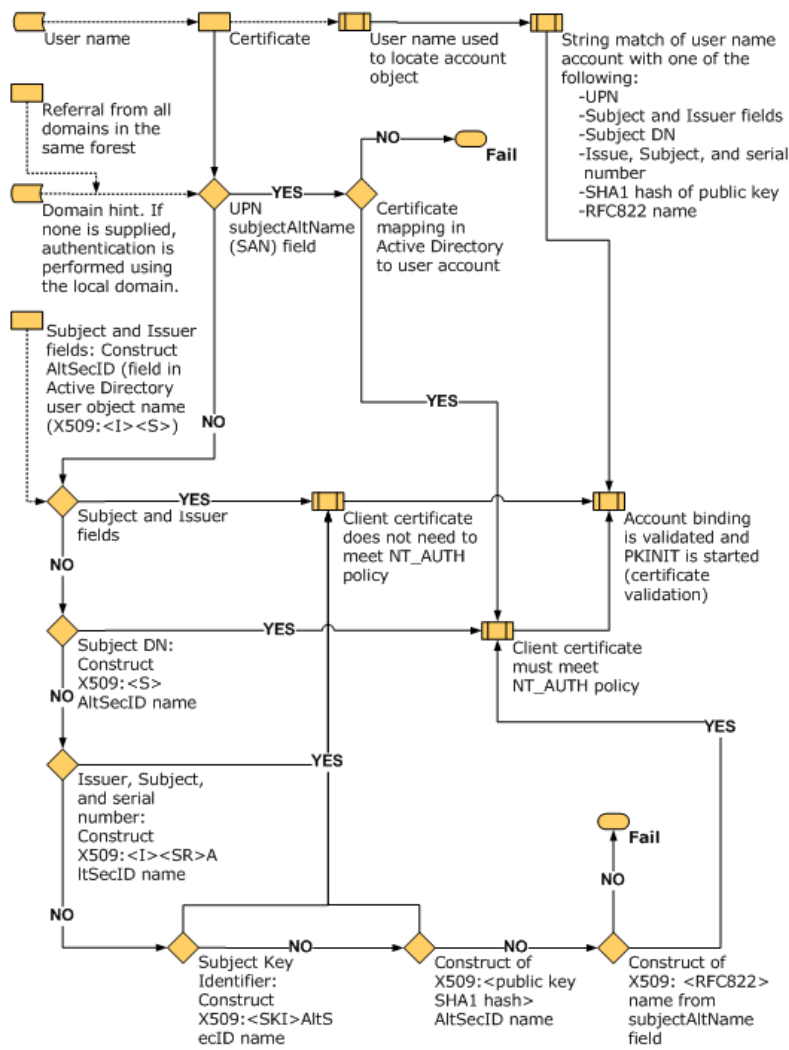
The certificate object is parsed to look for content to perform user account mapping.

- When a user name is provided with the certificate, the user name is used to locate the account object. This operation is the fastest, because string matching occurs.
- When only the certificate object is provided, a series of operations are performed to locate the user name to map the user name to an account object.
- When no domain information is available for authentication, the local domain is used by default. If any other domain is to be used for lookup, a domain name hint should be provided to perform the mapping and binding.

Mapping based on generic attributes is not possible because there is no generic API to retrieve attributes from a certificate. Currently, the first method that locates an account successfully stops the search. But a configuration error occurs if two methods map the same certificate to different user accounts when the client does not supply the client name through the mapping hints.

The following figure illustrates the process of mapping user accounts for sign-in in the directory by viewing various entries in the certificate.

Certificate processing logic



NT_AUTH policy is best described in the CERT_CHAIN_POLICY_NT_AUTH parameter section of the CertVerifyCertificateChainPolicy function. For more information, see [CertVerifyCertificateChainPolicy](#).

Smart card sign-in for a single user with one certificate into multiple accounts

A single user certificate can be mapped to multiple accounts. For example, a user might be able to sign in to a user account and also to sign in as a domain administrator. The mapping is done by using the constructed AltSecID based on attributes from client accounts. For information about how this mapping is evaluated, see [Client certificate requirements and mappings](#).

Note Because each account has a different user name, we recommend that you enable the **Allow user name hint** Group Policy setting (`X509HintsNeeded` registry key) to provide the optional fields that allow users to enter their user names and domain information to sign in.

Based on the information that is available in the certificate, the sign-in conditions are:

1. If no UPN is present in the certificate:
 - a. Sign-in can occur in the local forest or in another forest if a single user with one certificate needs to sign in to different accounts.
 - b. A hint must be supplied if mapping is not unique (for example, if multiple users are mapped to the same certificate).
2. If a UPN is present in the certificate:
 - a. The certificate cannot be mapped to multiple users in the same forest.

- b. The certificate can be mapped to multiple users in different forests. For a user to sign in to other forests, an X509 hint must be supplied to the user.

Smart card sign-in for multiple users into a single account

A group of users might sign in to a single account (for example, an administrator account). For that account, user certificates are mapped so that they are enabled for sign-in.

Several distinct certificates can be mapped to a single account. For this to work properly, the certificate cannot have UPNs.

For example, if Certificate1 has CN=CNName1, Certificate2 has CN=User1, and Certificate3 has CN=User2, the AltSecID of these certificates can be mapped to a single account by using the Active Directory Users and Computers name mapping.

Smart card sign-in across forests

For account mapping to work across forests, particularly in cases where there is not enough information available on the certificate, the user might enter a hint in the form of a user name, such as *domain\user*, or a fully qualified UPN such as *user@contoso.com*.

Note For the hint field to appear during smart card sign-in, the **Allow user name hint** Group Policy setting (**X509HintsNeeded** registry key) must be enabled on the client.

OCSP support for PKINIT

Online Certificate Status Protocol (OCSP), which is defined in RFC 2560, enables applications to obtain timely information about the revocation status of a certificate. Because OCSP responses are small and well bound, constrained clients might want to use OCSP to check the validity of the certificates for Kerberos on the KDC, to avoid transmission of large CRLs, and to save bandwidth on constrained networks. For information about CRL registry keys, see [Smart Card Group Policy and Registry Settings](#).

The KDCs in Windows attempt to get OCSP responses and use them when available. This behavior cannot be disabled. CryptoAPI for OCSP caches OCSP responses and the status of the responses. The KDC supports only OCSP responses for the signer certificate.

Windows client computers attempt to request the OCSP responses and use them in the reply when they are available. This behavior cannot be disabled.

Smart card root certificate requirements for use with domain sign-in

For sign-in to work in a smart card-based domain, the smart card certificate must meet the following conditions:

- The KDC root certificate on the smart card must have an HTTP CRL distribution point listed in its certificate.
- The smart card sign-in certificate must have the HTTP CRL distribution point listed in its certificate.
- The CRL distribution point must have a valid CRL published and a delta CRL, if applicable, even if the CRL distribution point is empty.
- The smart card certificate must contain one of the following:
 - A subject field that contains the DNS domain name in the distinguished name. If it does not, resolution to an appropriate domain fails, so Remote Desktop Services and the domain sign-in with the smart card fail.
 - A UPN where the domain name resolves to the actual domain. For example, if the domain name is Engineering.Corp.Contoso, the UPN is *username@engineering.corp.contoso.com*. If any part of the domain name is omitted, the Kerberos client cannot find the appropriate domain.

Although the HTTP CRL distribution points are on by default in Windows Server 2008, subsequent versions of the Windows Server operating system do not include HTTP CRL distribution points. To allow smart card sign-in to a domain in these versions, do the following:

1. Enable HTTP CRL distribution points on the CA.
2. Restart the CA.
3. Reissue the KDC certificate.
4. Issue or reissue the smart card sign-in certificate.
5. Propagate the updated root certificate to the smart card that you want to use for the domain sign-in.

The workaround is to enable the **Allow user name hint** Group Policy setting (**X509HintsNeeded** registry key), which allows the user to supply a hint in the credentials user interface for domain sign-in.

If the client computer is not joined to the domain or if it is joined to a different domain, the client computer can resolve the server domain only by looking at the distinguished name on the certificate, not the UPN. For this scenario to work, the certificate requires a full subject, including `DC= <DomainControllerName>`, for domain name resolution.

To deploy root certificates on a smart card for the currently joined domain, you can use the following command:

```
certutil -scroots update
```

For more information about this option for the command-line tool, see [-SCRoots](#).

See also

[How Smart Card Sign-in Works in Windows](#)

Smart Card and Remote Desktop Services

7/1/2022 • 5 minutes to read • [Edit Online](#)

Applies To: Windows 10, Windows 11, Windows Server 2016 and above

This topic for the IT professional describes the behavior of Remote Desktop Services when you implement smart card sign-in.

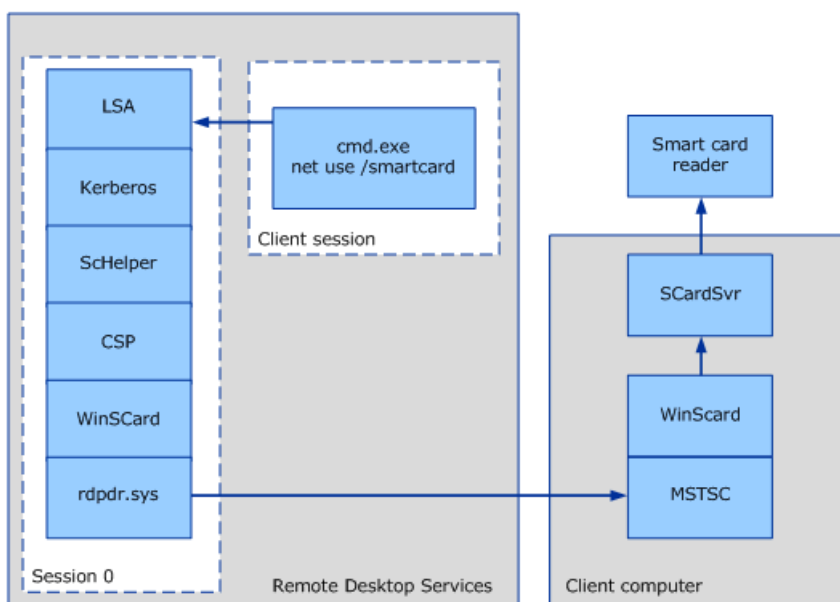
The content in this topic applies to the versions of Windows that are designated in the **Applies To** list at the beginning of this topic. In these versions, smart card redirection logic and **WinSCard** API are combined to support multiple redirected sessions into a single process.

Smart card support is required to enable many Remote Desktop Services scenarios. These include:

- Using Fast User Switching or Remote Desktop Services. A user is not able to establish a redirected smart card-based remote desktop connection. That is, the connect attempt is not successful in Fast User Switching or from a Remote Desktop Services session.
- Enabling Encrypting File System (EFS) to locate the user's smart card reader from the Local Security Authority (LSA) process in Fast User Switching or in a Remote Desktop Services session. If EFS is not able to locate the smart card reader or certificate, EFS cannot decrypt user files.

Remote Desktop Services redirection

In a Remote Desktop scenario, a user is using a remote server for running services, and the smart card is local to the computer that the user is using. In a smart card sign-in scenario, the smart card service on the remote server redirects to the smart card reader that is connected to the local computer where the user is trying to sign in.



Remote Desktop redirection

Notes about the redirection model:

1. This scenario is a remote sign-in session on a computer with Remote Desktop Services. In the remote session (labeled as "Client session"), the user runs **net use /smartcard**.

2. Arrows represent the flow of the PIN after the user types the PIN at the command prompt until it reaches the user's smart card in a smart card reader that is connected to the Remote Desktop Connection (RDC) client computer.
3. The authentication is performed by the LSA in session 0.
4. The CryptoAPI processing is performed in the LSA (Lsass.exe). This is possible because RDP redirector (rdpdr.sys) allows per-session, rather than per-process, context.
5. The WinSCard and SCRedir components, which were separate modules in operating systems earlier than Windows Vista, are now included in one module. The ScHelper library is a CryptoAPI wrapper that is specific to the Kerberos protocol.
6. The redirection decision is made on a per smart card context basis, based on the session of the thread that performs the SCardEstablishContext call.
7. Changes to WinSCard.dll implementation were made in Windows Vista to improve smart card redirection.

RD Session Host server single sign-in experience

As a part of the Common Criteria compliance, the RDC client must be configurable to use Credential Manager to acquire and save the user's password or smart card PIN. Common Criteria compliance requires that applications not have direct access to the user's password or PIN.

Common Criteria compliance requires specifically that the password or PIN never leave the LSA unencrypted. A distributed scenario should allow the password or PIN to travel between one trusted LSA and another, and it cannot be unencrypted during transit.

When smart card-enabled single sign-in (SSO) is used for Remote Desktop Services sessions, users still need to sign in for every new Remote Desktop Services session. However, the user is not prompted for a PIN more than once to establish a Remote Desktop Services session. For example, after the user double-clicks a Microsoft Word document icon that resides on a remote computer, the user is prompted to enter a PIN. This PIN is sent by using a secure channel that the credential SSP has established. The PIN is routed back to the RDC client over the secure channel and sent to Winlogon. The user does not receive any additional prompts for the PIN, unless the PIN is incorrect or there are smart card-related failures.

Remote Desktop Services and smart card sign-in

Remote Desktop Services enable users to sign in with a smart card by entering a PIN on the RDC client computer and sending it to the RD Session Host server in a manner similar to authentication that is based on user name and password.

In addition, Group Policy settings that are specific to Remote Desktop Services need to be enabled for smart card-based sign-in.

To enable smart card sign-in to a Remote Desktop Session Host (RD Session Host) server, the Key Distribution Center (KDC) certificate must be present on the RDC client computer. If the computer is not in the same domain or workgroup, the following command can be used to deploy the certificate:

```
certutil -dspublish NTAAuthCA "DSCDPContainer"
```

The *DSCDPContainer* Common Name (CN) is usually the name of the certification authority.

Example:

```
certutil -dspublish NTAAuthCA <CertFile> "CN=NTAuthCertificates,CN=Public Key Services,CN=Services,CN=Configuration,DC=engineering,DC=contoso,DC=com"
```

For information about this option for the command-line tool, see [-dsPublish](#).

Remote Desktop Services and smart card sign-in across domains

To enable remote access to resources in an enterprise, the root certificate for the domain must be provisioned on the smart card. From a computer that is joined to a domain, run the following command at the command line:

```
certutil -scroots update
```

For information about this option for the command-line tool, see [-SCRoots](#).

For Remote Desktop Services across domains, the KDC certificate of the RD Session Host server must also be present in the client computer's NTAUTH store. To add the store, run the following command at the command line:

```
certutil -addstore -enterprise NTAUTH <CertFile>
```

Where *<CertFile>* is the root certificate of the KDC certificate issuer.

For information about this option for the command-line tool, see [-addstore](#).

Note If you use the credential SSP on computers running the supported versions of the operating system that are designated in the **Applies To** list at the beginning of this topic: To sign in with a smart card from a computer that is not joined to a domain, the smart card must contain the root certification of the domain controller. A public key infrastructure (PKI) secure channel cannot be established without the root certification of the domain controller.

Sign-in to Remote Desktop Services across a domain works only if the UPN in the certificate uses the following form: *<ClientName>@<DomainDNSName>*

The UPN in the certificate must include a domain that can be resolved. Otherwise, the Kerberos protocol cannot determine which domain to contact. You can resolve this issue by enabling GPO X509 domain hints. For more information about this setting, see [Smart Card Group Policy and Registry Settings](#).

See also

[How Smart Card Sign-in Works in Windows](#)

Smart Cards for Windows Service

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies To: Windows 10, Windows 11, Windows Server 2016 and above

This topic for the IT professional and smart card developers describes how the Smart Cards for Windows service (formerly called Smart Card Resource Manager) manages readers and application interactions.

The Smart Cards for Windows service provides the basic infrastructure for all other smart card components as it manages smart card readers and application interactions on the computer. It is fully compliant with the specifications set by the PC/SC Workgroup. For information about these specifications, see the [PC/SC Workgroup Specifications website](#).

The Smart Cards for Windows service runs in the context of a local service, and it is implemented as a shared service of the services host (svchost) process. The Smart Cards for Windows service, Scardsvr, has the following service description:

```

<serviceData
  dependOnService="PlugPlay"
  description="@%SystemRoot%\System32\SCardSvr.dll,-5"
  displayName="@%SystemRoot%\System32\SCardSvr.dll,-1"
  errorControl="normal"
  group="SmartCardGroup"
  imagePath="%SystemRoot%\system32\svchost.exe -k LocalServiceAndNoImpersonation"
  name="SCardSvr"
  objectName="NT AUTHORITY\LocalService"
  requiredPrivileges="SeCreateGlobalPrivilege,SeChangeNotifyPrivilege"
  sidType="unrestricted"
  start="demand"
  type="win32ShareProcess"
>
<failureActions resetPeriod="900">
  <actions>
    <action
      delay="120000"
      type="restartService"
    />
    <action
      delay="300000"
      type="restartService"
    />
    <action
      delay="0"
      type="none"
    />
  </actions>
</failureActions>
<securityDescriptor name="ServiceXSecurity"/>
</serviceData>

<registryKeys buildFilter="">
  <registryKey keyName="HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SCardSvr\Parameters">
    <registryValue
      name="ServiceDll"
      value="%SystemRoot%\System32\SCardSvr.dll"
      valueType="REG_EXPAND_SZ"
    />
    <registryValue
      name="ServiceMain"
      value="CalaisMain"
      valueType="REG_SZ"
    />
    <registryValue
      name="ServiceDllUnloadOnStop"
      value="1"
      valueType="REG_DWORD"
    />
  </registryKey>
</registryKeys>

```

Note For winscard.dll to be invoked as the proper class installer, the INF file for a smart card reader must specify the following for **Class** and **ClassGUID**:

```
Class=SmartCardReader
```

```
ClassGuid={50DD5230-BA8A-11D1-BF5D-0000F805F530}
```

By default, the service is configured for manual mode. Creators of smart card reader drivers must configure their INFs so that they start the service automatically and winscard.dll files call a predefined entry point to start the service during installation. The entry point is defined as part of the **SmartCardReader** class, and it is not called directly. If a device advertises itself as part of this class, the entry point is automatically invoked to start the service when the device is inserted. Using this method ensures that the service is enabled when it is needed,

but it is also disabled for users who do not use smart cards.

When the service is started, it performs several functions:

1. It registers itself for service notifications.
2. It registers itself for Plug and Play (PnP) notifications related to device removal and additions.
3. It initializes its data cache and a global event that signals that the service has started.

Note For smart card implementations, consider sending all communications in Windows operating systems with smart card readers through the Smart Cards for Windows service. This provides an interface to track, select, and communicate with all drivers that declare themselves members of the smart card reader device group.

The Smart Cards for Windows service categorizes each smart card reader slot as a unique reader, and each slot is also managed separately, regardless of the device's physical characteristics. The Smart Cards for Windows service handles the following high-level actions:

- Device introduction
- Reader initialization
- Notifying clients of new readers
- Serializing access to readers
- Smart card access
- Tunneling of reader-specific commands

See also

[How Smart Card Sign-in Works in Windows](#)

Certificate Propagation Service

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies To: Windows 10, Windows 11, Windows Server 2016 and above

This topic for the IT professional describes the certificate propagation service (CertPropSvc), which is used in smart card implementation.

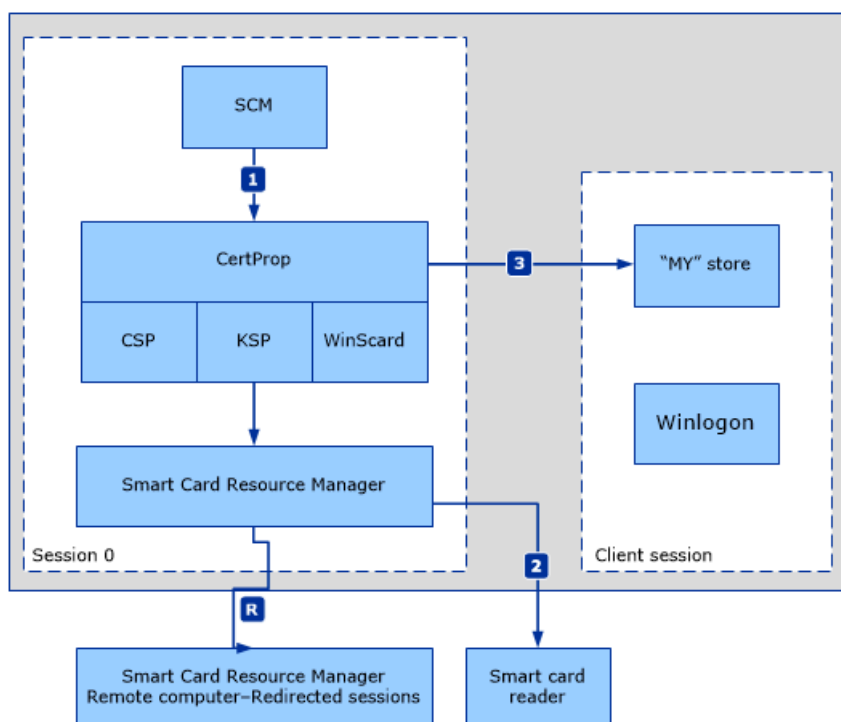
The certificate propagation service activates when a signed-in user inserts a smart card in a reader that is attached to the computer. This action causes the certificate to be read from the smart card. The certificates are then added to the user's Personal store. Certificate propagation service actions are controlled by using Group Policy. For more information, see [Smart Card Group Policy and Registry Settings](#).

Note The certificate propagation service must be running for smart card Plug and Play to work.

The following figure shows the flow of the certificate propagation service. The action begins when a signed-in user inserts a smart card.

1. The arrow labeled 1 indicates that the Service Control Manager (SCM) notifies the certificate propagation service (CertPropSvc) when a user signs in, and CertPropSvc begins to monitor the smart cards in the user session.
2. The arrow labeled R represents the possibility of a remote session and the use of smart card redirection.
3. The arrow labeled 2 indicates the certification to the reader.
4. The arrow labeled 3 indicates the access to the certificate store during the client session.

Certificate propagation service



1. A signed-in user inserts a smart card.
2. CertPropSvc is notified that a smart card was inserted.

- CertPropSvc reads all certificates from all inserted smart cards. The certificates are written to the user's personal certificate store.

Note The certificate propagation service is started as a Remote Desktop Services dependency.

Properties of the certificate propagation service include:

- CERT_STORE_ADD_REPLACE_EXISTING_INHERIT_PROPERTIES adds certificates to a user's Personal store.
- If the certificate has the CERT_ENROLLMENT_PROP_ID property (as defined by wincrypt.h), it filters empty requests and places them in the current user's request store, but it does not propagate them to the user's Personal store.
- The service does not propagate any computer certificates to a user's Personal store or propagate user certificates to a computer store.
- The service propagates certificates according to Group Policy options that are set, which may include:
 - **Turn on certificate propagation from the smart card** specifies whether a user's certificate should be propagated.
 - **Turn on root certificate propagation from smart card** specifies whether root certificates should be propagated.
 - **Configure root certificate cleanup** specifies how root certificates are removed.

Root certificate propagation service

Root certificate propagation is responsible for the following smart card deployment scenarios when public key infrastructure (PKI) trust has not yet been established:

- Joining the domain
- Accessing a network remotely

In both cases, the computer is not joined to a domain, and therefore, trust is not being managed by Group Policy. However, the objective is to authenticate to a remote server, such as the domain controller. Root certificate propagation provides the ability to use the smart card to include the missing trust chain.

When the smart card is inserted, the certificate propagation service propagates any root certificates on the card to the trusted smart card root computer certificate stores. This process establishes a trust relationship with the enterprise resources. You may also use a subsequent cleanup action when the user's smart card is removed from the reader, or when the user signs out. This is configurable with Group Policy. For more information, see [Smart Card Group Policy and Registry Settings](#).

For more information about root certificate requirements, see [Smart card root certificate requirements for use with domain sign-in](#).

See also

[How Smart Card Sign-in Works in Windows](#)

Smart Card Removal Policy Service

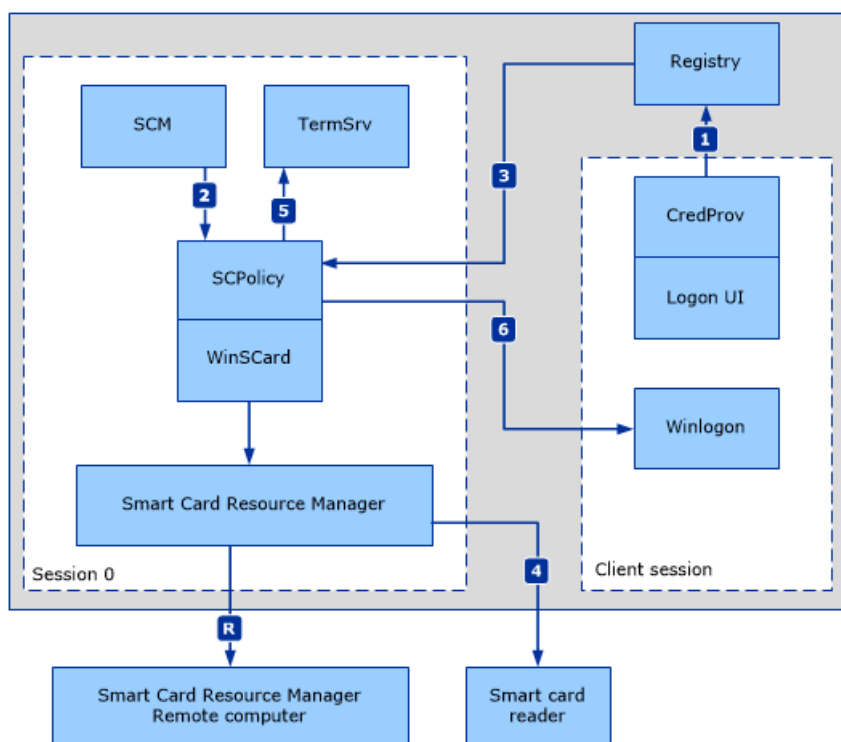
7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies To: Windows 10, Windows 11, Windows Server 2016

This topic for the IT professional describes the role of the removal policy service (ScPolicySvc) in smart card implementation.

The smart card removal policy service is applicable when a user has signed in with a smart card and then removes that smart card from the reader. The action that is performed when the smart card is removed is controlled by Group Policy settings. For more information, see [Smart Card Group Policy and Registry Settings](#).

Smart card removal policy service



The numbers in the previous figure represent the following actions:

1. Winlogon is not directly involved in monitoring for smart card removal events. The sequence of steps that are involved when a smart card is removed begins with the smart card credential provider in the sign-in UI process. When a user successfully signs in with a smart card, the smart card credential provider captures the reader name. This information is then stored in the registry with the session identifier where the sign in was initiated.
2. The smart card resource manager service notifies the smart card removal policy service that a sign-in has occurred.
3. ScPolicySvc retrieves the smart card information that the smart card credential provider stored in the registry. This call is redirected if the user is in a remote session. If the smart card is removed, ScPolicySvc is notified.
4. ScPolicySvc calls Remote Desktop Services to take the appropriate action if the request is to sign out the user or to disconnect the user's session, which might result in data loss. If the setting is configured to lock the computer when the smart card is removed, ScPolicySvc sends a message to Winlogon to lock the computer.

See also

[How Smart Card Sign-in Works in Windows](#)

Smart Card Tools and Settings

7/1/2022 • 2 minutes to read • [Edit Online](#)

Applies To: Windows 10, Windows 11, Windows Server 2016 and above

This topic for the IT professional and smart card developer links to information about smart card debugging, settings, and events.

This section of the Smart Card Technical Reference contains information about the following:

- [Smart Cards Debugging Information](#): Learn about tools and services in supported versions of Windows to help identify certificate issues.
- [Smart Card Group Policy and Registry Settings](#): Learn about smart card-related Group Policy settings and registry keys that can be set on a per-computer basis, including how to edit and apply Group Policy settings to local or domain computers.
- [Smart Card Events](#): Learn about events that can be used to manage smart cards in an organization, including how to monitor installation, use, and errors.

See also

[Smart Card Technical Reference](#)

Smart Card Troubleshooting

7/1/2022 • 5 minutes to read • [Edit Online](#)

Applies To: Windows 10, Windows 11, Windows Server 2016 and above

This article explains tools and services that smart card developers can use to help identify certificate issues with the smart card deployment.

Debugging and tracing smart card issues requires a variety of tools and approaches. The following sections provide guidance about tools and approaches you can use.

- [Certutil](#)
- [Debugging and tracing using Windows software trace preprocessor \(WPP\)](#)
- [Kerberos protocol, Key Distribution Center \(KDC\), and NTLM debugging and tracing](#)
- [Smart Card service](#)
- [Smart card readers](#)
- [CryptoAPI 2.0 Diagnostics](#)

Certutil

For a complete description of Certutil including examples that show how to use it, see [Certutil \[W2012\]](#).

List certificates available on the smart card

To list certificates that are available on the smart card, type `certutil -scinfo`.

NOTE

Entering a PIN is not required for this operation. You can press ESC if you are prompted for a PIN.

Delete certificates on the smart card

Each certificate is enclosed in a container. When you delete a certificate on the smart card, you're deleting the container for the certificate.

To find the container value, type `certutil -scinfo`.

To delete a container, type `certutil -delkey -csp "Microsoft Base Smart Card Crypto Provider" "<ContainerValue>"`.

Debugging and tracing using WPP

WPP simplifies tracing the operation of the trace provider. It provides a mechanism for the trace provider to log real-time binary messages. Logged messages can be converted to a human-readable trace of the operation. For more information, see [Diagnostics with WPP - The NDIS blog](#).

Enable the trace

Using WPP, use one of the following commands to enable tracing:

- `tracelog.exe -kd -rt -start <FriendlyName> -guid #<GUID> -f .\<LogFileName>.etl -flags <flags> -ft 1`

- **logman start** <FriendlyName> -ets -p {<GUID>} -<Flags> -ft 1 -rt -o .\<LogFileName>.etl -mode 0x00080000

You can use the parameters in the following table.

FRIENDLY NAME	GUID	FLAGS
scardsvr	13038e47-ffec-425d-bc69-5707708075fe	0xffff
winscard	3fce7c5f-fb3b-4bce-a9d8-55cc0ce1cf01	0xffff
basecsp	133a980d-035d-4e2d-b250-94577ad8fced	0x7
scksp	133a980d-035d-4e2d-b250-94577ad8fced	0x7
msclmd	fb36caf4-582b-4604-8841-9263574c4f2c	0x7
credprov	dba0e0e0-505a-4ab6-aa3f-22f6f743b480	0xffff
certprop	30eae751-411f-414c-988b-a8bfa8913f49	0xffff
scfilter	eed7f3c9-62ba-400e-a001-658869df9a91	0xffff
wudfusbccid	a3c09ba3-2f62-4be5-a50f-8278a646ac9d	0xffff

Examples

To enable tracing for the SCardSvr service:

- **tracelog.exe -kd -rt -start scardsvr -guid #13038e47-ffec-425d-bc69-5707708075fe -f .\scardsvr.etl -flags 0xffff -ft 1**
- **logman start scardsvr -ets -p {13038e47-ffec-425d-bc69-5707708075fe} 0xffff -ft 1 -rt -o .\scardsvr.etl -mode 0x00080000**

To enable tracing for scfilter.sys:

- **tracelog.exe -kd -rt -start scfilter -guid #eed7f3c9-62ba-400e-a001-658869df9a91 -f .\scfilter.etl -flags 0xffff -ft 1**

Stop the trace

Using WPP, use one of the following commands to stop the tracing:

- **tracelog.exe -stop <FriendlyName>**
- **logman -stop <FriendlyName> -ets**

Examples

To stop a trace:

- `tracelog.exe -stop scardsvr`
- `logman -stop scardsvr -ets`

Kerberos protocol, KDC, and NTLM debugging and tracing

You can use these resources to troubleshoot these protocols and the KDC:

- [Kerberos and LDAP Troubleshooting Tips](#).
- [Windows Driver Kit \(WDK\) and Debugging Tools for Windows \(WinDbg\)](#). You can use the trace log tool in this SDK to debug Kerberos authentication failures.

To begin tracing, you can use `TraceLog`. Different components use different control GUIDs as explained in these examples. For more information, see [TraceLog](#).

NTLM

To enable tracing for NTLM authentication, run the following command on the command line:

- `tracelog.exe -kd -rt -start ntlm -guid #5BBB6C18-AA45-49b1-A15F-085F7ED0AA90 -f .\ntlm.etl -flags 0x15003 -ft 1`

To stop tracing for NTLM authentication, run this command:

- `tracelog -stop ntlm`

Kerberos authentication

To enable tracing for Kerberos authentication, run this command:

- `tracelog.exe -kd -rt -start kerb -guid #6B510852-3583-4e2d-AFFE-A67F9F223438 -f .\kerb.etl -flags 0x43 -ft 1`

To stop tracing for Kerberos authentication, run this command:

- `tracelog.exe -stop kerb`

KDC

To enable tracing for the KDC, run the following command on the command line:

- `tracelog.exe -kd -rt -start kdc -guid #1BBA8B19-7F31-43c0-9643-6E911F79A06B -f .\kdc.etl -flags 0x803 -ft 1`

To stop tracing for the KDC, run the following command on the command line:

- `tracelog.exe -stop kdc`

To stop tracing from a remote computer, run this command: `logman.exe -s <ComputerName>`.

NOTE

The default location for `logman.exe` is `%systemroot%\system32\`. Use the `-s` option to supply a computer name.

Configure tracing with the registry

You can also configure tracing by editing the Kerberos registry values shown in the following table.

ELEMENT	REGISTRY KEY SETTING
---------	----------------------

ELEMENT	REGISTRY KEY SETTING
NTLM	HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\MSV1_0 Value name: NtLmInfoLevel Value type: DWORD Value data: c0015003
Kerberos	HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\Kerberos Value name: LogToFile Value type: DWORD Value data: 00000001 HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\Kerberos\Parameters Value name: KerbDebugLevel Value type: DWORD Value data: c0000043 HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\Kerberos\Parameters Value name: LogToFile Value type: DWORD Value data: 00000001
KDC	HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Service\s\Kdc Value name: KdcDebugLevel Value type: DWORD Value data: c0000803

If you used `TraceLog`, look for the following log file in your current directory: `kerb.etl/kdc.etl/ntlm.etl`.

If you used the registry key settings shown in the previous table, look for the trace log files in the following locations:

- NTLM: `%systemroot%\tracing\msv1_0`
- Kerberos: `%systemroot%\tracing\kerberos`
- KDC: `%systemroot%\tracing\kdcsvc`

To decode event trace files, you can use `Tracefmt` (`tracefmt.exe`). `Tracefmt` is a command-line tool that formats and displays trace messages from an event trace log file (.etl) or a real-time trace session. `Tracefmt` can display the messages in the Command Prompt window or save them in a text file. It is located in the `\tools\tracing` subdirectory of the Windows Driver Kit (WDK). For more information, see `Tracefmt`.

Smart Card service

The smart card resource manager service runs in the context of a local service. It's implemented as a shared service of the services host (svchost) process.

To check if Smart Card service is running

1. Press CTRL+ALT+DEL, and then select **Start Task Manager**.
2. In the **Windows Task Manager** dialog box, select the **Services** tab.
3. Select the **Name** column to sort the list alphabetically, and then type `s`.

4. In the **Name** column, look for **SCardSvr**, and then look under the **Status** column to see if the service is running or stopped.

To restart Smart Card service

1. Run as administrator at the command prompt.
2. If the **User Account Control** dialog box appears, confirm that the action it displays is what you want, and then select **Yes**.
3. At the command prompt, type `net stop SCardSvr`.
4. At the command prompt, type `net start SCardSvr`.

You can use the following command at the command prompt to check whether the service is running:

```
sc queryex scardsvr
```

The following code sample is an example output from this command:

```
SERVICE_NAME: scardsvr
    TYPE               : 20  WIN32_SHARE_PROCESS
    STATE               : 4  RUNNING
                      (STOPPABLE, NOT_PAUSABLE, ACCEPTS_SHUTDOWN)
    WIN32_EXIT_CODE     : 0  (0x0)
    SERVICE_EXIT_CODE  : 0  (0x0)
    CHECKPOINT         : 0x0
    WAIT_HINT          : 0x0
    PID                : 1320
    FLAGS              :
C:\>
```

Smart card readers

As with any device connected to a computer, Device Manager can be used to view properties and begin the debug process.

To check if smart card reader is working

1. Navigate to **Computer**.
2. Right-click **Computer**, and then select **Properties**.
3. Under **Tasks**, select **Device Manager**.
4. In Device Manager, expand **Smart card readers**, select the name of the smart card reader you want to check, and then select **Properties**.

NOTE

If the smart card reader is not listed in Device Manager, in the **Action** menu, select **Scan for hardware changes**.

CryptoAPI 2.0 Diagnostics

CryptoAPI 2.0 Diagnostics is available in Windows versions that support CryptoAPI 2.0 and can help you troubleshoot public key infrastructure (PKI) issues.

CryptoAPI 2.0 Diagnostics logs events in the Windows event log. The logs contain detailed information about certificate chain validation, certificate store operations, and signature verification. This information makes it easier to identify the causes of issues and reduces the time required for diagnosis.

For more information about CryptoAPI 2.0 Diagnostics, see [Troubleshooting an Enterprise PKI](#).

See also

[Smart Card Technical Reference](#)

Smart Card Group Policy and Registry Settings

7/1/2022 • 20 minutes to read • [Edit Online](#)

Applies to: Windows 10, Windows 11, Windows Server 2016 and above

This article for IT professionals and smart card developers describes the Group Policy settings, registry key settings, local security policy settings, and credential delegation policy settings that are available for configuring smart cards.

The following sections and tables list the smart card-related Group Policy settings and registry keys that can be set on a per-computer basis. If you use domain Group Policy Objects (GPOs), you can edit and apply Group Policy settings to local or domain computers.

- [Primary Group Policy settings for smart cards](#)
 - [Allow certificates with no extended key usage certificate attribute](#)
 - [Allow ECC certificates to be used for logon and authentication](#)
 - [Allow Integrated Unblock screen to be displayed at the time of logon](#)
 - [Allow signature keys valid for Logon](#)
 - [Allow time invalid certificates](#)
 - [Allow user name hint](#)
 - [Configure root certificate clean up](#)
 - [Display string when smart card is blocked](#)
 - [Filter duplicate logon certificates](#)
 - [Force the reading of all certificates from the smart card](#)
 - [Notify user of successful smart card driver installation](#)
 - [Prevent plaintext PINs from being returned by Credential Manager](#)
 - [Reverse the subject name stored in a certificate when displaying](#)
 - [Turn on certificate propagation from smart card](#)
 - [Turn on root certificate propagation from smart card](#)
 - [Turn on Smart Card Plug and Play service](#)
- [Base CSP and Smart Card KSP registry keys](#)
- [CRL checking registry keys](#)
- [Additional smart card Group Policy settings and registry keys](#)

Primary Group Policy settings for smart cards

The following smart card Group Policy settings are in Computer Configuration\Administrative Templates\Windows Components\Smart Card.

The registry keys are in the following locations:

- HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\ScPnP\EnableScPnP
- HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\SmartCardCredentialProvider
- HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\CertProp

NOTE

Smart card reader registry information is in
HKEY_LOCAL_MACHINE\Software\Microsoft\Cryptography\Calais\Readers.
Smart card registry information is in
HKEY_LOCAL_MACHINE\Software\Microsoft\Cryptography\Calais\SmartCards.

The following table lists the default values for these GPO settings. Variations are documented under the policy descriptions in this article.

SERVER TYPE OR GPO	DEFAULT VALUE
Default Domain Policy	Not configured
Default Domain Controller Policy	Not configured
Stand-Alone Server Default Settings	Not configured
Domain Controller Effective Default Settings	Disabled
Member Server Effective Default Settings	Disabled
Client Computer Effective Default Settings	Disabled

Allow certificates with no extended key usage certificate attribute

You can use this policy setting to allow certificates without an enhanced key usage (EKU) set to be used for sign in.

NOTE

Enhanced key usage certificate attribute is also known as extended key usage.
In versions of Windows before Windows Vista, smart card certificates that are used to sign in require an EKU extension with a smart card logon object identifier. This policy setting can be used to modify that restriction.

When this policy setting is turned on, certificates with the following attributes can also be used to sign in with a smart card:

- Certificates with no EKU
- Certificates with an All Purpose EKU
- Certificates with a Client Authentication EKU

When this policy setting isn't turned on, only certificates that contain the smart card logon object identifier can be used to sign in with a smart card.

ITEM	DESCRIPTION
Registry key	AllowCertificatesWithNoEKU
Default values	No changes per operating system versions Disabled and not configured are equivalent
Policy management	Restart requirement: None Sign off requirement: None Policy conflicts: None
Notes and resources	

Allow ECC certificates to be used for logon and authentication

You can use this policy setting to control whether elliptic curve cryptography (ECC) certificates on a smart card can be used to sign in to a domain.

When this setting is turned on, ECC certificates on a smart card can be used to sign in to a domain.

When this setting isn't turned on, ECC certificates on a smart card can't be used to sign in to a domain.

ITEM	DESCRIPTION
Registry key	EnumerateECCerts
Default values	No changes per operating system versions Disabled and not configured are equivalent
Policy management	Restart requirement: None Sign off requirement: None Policy conflicts: None
Notes and resources	This policy setting only affects a user's ability to sign in to a domain. ECC certificates on a smart card that are used for other applications, such as document signing, aren't affected by this policy setting. If you use an ECDSA key to sign in, you must also have an associated ECDH key to permit sign in when you're not connected to the network.

Allow Integrated Unblock screen to be displayed at the time of logon

You can use this policy setting to determine whether the integrated unblock feature is available in the sign-in user interface (UI). The feature was introduced as a standard feature in the Credential Security Support Provider in Windows Vista.

When this setting is turned on, the integrated unblock feature is available.

When this setting isn't turned on, the feature is not available.

ITEM	DESCRIPTION
Registry key	AllowIntegratedUnblock
Default values	No changes per operating system versions Disabled and not configured are equivalent

ITEM	DESCRIPTION
Policy management	Restart requirement: None Sign off requirement: None Policy conflicts: None
Notes and resources	To use the integrated unblock feature, the smart card must support it. Check with the hardware manufacturer to verify that the smart card supports this feature. You can create a custom message that the user sees when the smart card is blocked by configuring the policy setting Display string when smart card is blocked .

Allow signature keys valid for Logon

You can use this policy setting to allow signature key-based certificates to be enumerated and available for sign in.

When this setting is turned on, any certificates that are available on the smart card with a signature-only key are listed on the sign-in screen.

When this setting isn't turned on, certificates available on the smart card with a signature-only key aren't listed on the sign-in screen.

ITEM	DESCRIPTION
Registry key	AllowSignatureOnlyKeys
Default values	No changes per operating system versions Disabled and not configured are equivalent
Policy management	Restart requirement: None Sign off requirement: None Policy conflicts: None
Notes and resources	

Allow time invalid certificates

You can use this policy setting to permit certificates that are expired or not yet valid to be displayed for sign in.

NOTE

Before Windows Vista, certificates were required to contain a valid time and to not expire. For a certificate to be used, it must be accepted by the domain controller. This policy setting only controls which certificates are displayed on the client computer.

When this setting is turned on, certificates are listed on the sign-in screen whether they have an invalid time, or their time validity has expired.

When this policy setting isn't turned on, certificates that are expired or not yet valid aren't listed on the sign-in screen.

ITEM	DESCRIPTION
Registry key	AllowTimeInvalidCertificates

ITEM	DESCRIPTION
Default values	No changes per operating system versions Disabled and not configured are equivalent
Policy management	Restart requirement: None Sign off requirement: None Policy conflicts: None
Notes and resources	

Allow user name hint

You can use this policy setting to determine whether an optional field appears during sign in and provides a subsequent elevation process where users can enter their username or username and domain, which associates a certificate with the user.

When this policy setting is turned on, users see an optional field where they can enter their username or username and domain.

When this policy setting isn't turned on, users don't see this optional field.

ITEM	DESCRIPTION
Registry key	X509HintsNeeded
Default values	No changes per operating system versions Disabled and not configured are equivalent
Policy management	Restart requirement: None Sign off requirement: None Policy conflicts: None
Notes and resources	

Configure root certificate clean up

You can use this policy setting to manage the cleanup behavior of root certificates. Certificates are verified by using a trust chain, and the trust anchor for the digital certificate is the Root Certification Authority (CA). A CA can issue multiple certificates with the root certificate as the top certificate of the tree structure. A private key is used to sign other certificates. This creates an inherited trustworthiness for all certificates immediately under the root certificate.

When this policy setting is turned on, you can set the following cleanup options:

- **No cleanup.** When the user signs out or removes the smart card, the root certificates used during their session persist on the computer.
- **Clean up certificates on smart card removal.** When the smart card is removed, the root certificates are removed.
- **Clean up certificates on log off.** When the user signs out of Windows, the root certificates are removed.

When this policy setting isn't turned on, root certificates are automatically removed when the user signs out of Windows.

ITEM	DESCRIPTION
Registry key	RootCertificateCleanupOption
Default values	No changes per operating system versions Disabled and not configured are equivalent
Policy management	Restart requirement: None Sign off requirement: None Policy conflicts: None
Notes and resources	

Display string when smart card is blocked

You can use this policy setting to change the default message that a user sees if their smart card is blocked.

When this policy setting is turned on, you can create and manage the displayed message that the user sees when a smart card is blocked.

When this policy setting isn't turned on (and the integrated unblock feature is also enabled), the user sees the system's default message when the smart card is blocked.

ITEM	DESCRIPTION
Registry key	IntegratedUnblockPromptString
Default values	No changes per operating system versions Disabled and not configured are equivalent
Policy management	Restart requirement: None Sign off requirement: None Policy conflicts: This policy setting is only effective when the Allow Integrated Unblock screen to be displayed at the time of logon policy is enabled.
Notes and resources	

Filter duplicate logon certificates

You can use this policy setting to configure which valid sign-in certificates are displayed.

NOTE

During the certificate renewal period, a user's smart card can have multiple valid sign-in certificates issued from the same certificate template, which can cause confusion about which certificate to select. This behavior can occur when a certificate is renewed and the old certificate has not expired yet.

If two certificates are issued from the same template with the same major version and they are for the same user (this is determined by their UPN), they are determined to be the same.

When this policy setting is turned on, filtering occurs so that the user can select from only the most current valid certificates.

If this policy setting isn't turned on, all the certificates are displayed to the user.

This policy setting is applied to the computer after the [Allow time invalid certificates](#) policy setting is applied.

ITEM	DESCRIPTION
Registry key	FilterDuplicateCerts
Default values	No changes per operating system versions Disabled and not configured are equivalent
Policy management	Restart requirement: None Sign off requirement: None Policy conflicts: None
Notes and resources	If there are two or more of the same certificates on a smart card and this policy setting is enabled, the certificate that is used to sign in to computers running Windows 2000, Windows XP, or Windows Server 2003 will be displayed. Otherwise, the certificate with the most distant expiration time will be displayed.

Force the reading of all certificates from the smart card

You can use this policy setting to manage how Windows reads all certificates from the smart card for sign in. During sign in, Windows reads only the default certificate from the smart card unless it supports retrieval of all certificates in a single call. This policy setting forces Windows to read all the certificates from the smart card.

When this policy setting is turned on, Windows attempts to read all certificates from the smart card, regardless of the CSP feature set.

When this policy isn't turned on, Windows attempts to read only the default certificate from smart cards that don't support retrieval of all certificates in a single call. Certificates other than the default aren't available for sign in.

ITEM	DESCRIPTION
Registry key	ForceReadingAllCertificates
Default values	No changes per operating system versions Disabled and not configured are equivalent
Policy management	Restart requirement: None Sign off requirement: None Policy conflicts: None Important: Enabling this policy setting can adversely impact performance during the sign in process in certain situations.
Notes and resources	Contact the smart card vendor to determine if your smart card and associated CSP support the required behavior.

Notify user of successful smart card driver installation

You can use this policy setting to control whether the user sees a confirmation message when a smart card device driver is installed.

When this policy setting is turned on, the user sees a confirmation message when a smart card device driver is installed.

When this setting isn't turned on, the user doesn't see a smart card device driver installation message.

ITEM	DESCRIPTION
Registry key	ScPnPNotification
Default values	No changes per operating system versions Disabled and not configured are equivalent
Policy management	Restart requirement: None Sign off requirement: None Policy conflicts: None
Notes and resources	This policy setting applies only to smart card drivers that have passed the Windows Hardware Quality Labs (WHQL) testing process.

Prevent plaintext PINs from being returned by Credential Manager

You can use this policy setting to prevent Credential Manager from returning plaintext PINs.

NOTE

Credential Manager is controlled by the user on the local computer, and it stores credentials from supported browsers and Windows applications. Credentials are saved in special encrypted folders on the computer under the user's profile.

When this policy setting is turned on, Credential Manager doesn't return a plaintext PIN.

When this setting isn't turned on, Credential Manager can return plaintext PINs.

ITEM	DESCRIPTION
Registry key	DisallowPlaintextPin
Default values	No changes per operating system versions Disabled and not configured are equivalent
Policy management	Restart requirement: None Sign off requirement: None Policy conflicts: None
Notes and resources	If this policy setting is enabled, some smart cards might not work in computers running Windows. Consult the smart card manufacturer to determine whether this policy setting should be enabled.

Reverse the subject name stored in a certificate when displaying

You can use this policy setting to control the way the subject name appears during sign in.

NOTE

To help users distinguish one certificate from another, the user principal name (UPN) and the common name are displayed by default. For example, when this setting is enabled, if the certificate subject is CN=User1, OU=Users, DN=example, DN=com and the UPN is user1@example.com, "User1" is displayed with "user1@example.com." If the UPN is not present, the entire subject name is displayed. This setting controls the appearance of that subject name, and it might need to be adjusted for your organization.

When this policy setting is turned on, the subject name during sign in appears reversed from the way that it's stored in the certificate.

When this policy setting isn't turned on, the subject name appears the same as it's stored in the certificate.

ITEM	DESCRIPTION
Registry key	ReverseSubject
Default values	No changes per operating system versions Disabled and not configured are equivalent
Policy management	Restart requirement: None Sign off requirement: None Policy conflicts: None
Notes and resources	

Turn on certificate propagation from smart card

You can use this policy setting to manage the certificate propagation that occurs when a smart card is inserted.

NOTE

The certificate propagation service applies when a signed-in user inserts a smart card in a reader that is attached to the computer. This action causes the certificate to be read from the smart card. The certificates are then added to the user's Personal store.

When this policy setting is turned on, certificate propagation occurs when the user inserts the smart card.

When this policy setting is turned off, certificate propagation doesn't occur, and the certificates aren't available to applications, like Outlook.

ITEM	DESCRIPTION
Registry key	CertPropEnabled
Default values	No changes per operating system versions Enabled and not configured are equivalent
Policy management	Restart requirement: None Sign off requirement: None Policy conflicts: This policy setting must be enabled to allow the Turn on root certificate propagation from smart card setting to work when it is enabled.
Notes and resources	

Turn on root certificate propagation from smart card

You can use this policy setting to manage the root certificate propagation that occurs when a smart card is inserted.

NOTE

The certificate propagation service applies when a signed-in user inserts a smart card in a reader that is attached to the computer. This action causes the certificate to be read from the smart card. The certificates are then added to the user's Personal store.

When this policy setting is turned on, root certificate propagation occurs when the user inserts the smart card.

When this policy setting isn't turned on, root certificate propagation doesn't occur when the user inserts the smart card.

ITEM	DESCRIPTION
Registry key	EnableRootCertificate Propagation
Default values	No changes per operating system versions Enabled and not configured are equivalent
Policy management	Restart requirement: None Sign off requirement: None Policy conflicts: For this policy setting to work, the Turn on certificate propagation from smart card policy setting must also be enabled.
Notes and resources	

Turn on Smart Card Plug and Play service

You can use this policy setting to control whether Smart Card Plug and Play is enabled.

NOTE

Your users can use smart cards from vendors who have published their drivers through Windows Update without needing special middleware. These drivers will be downloaded in the same way as drivers for other devices in Windows. If an appropriate driver isn't available from Windows Update, a PIV-compliant mini driver that's included with any of the supported versions of Windows is used for these cards.

When this policy setting is turned on, the system attempts to install a smart card device driver the first time a smart card is inserted in a smart card reader.

When this policy setting isn't turned on, a device driver isn't installed when a smart card is inserted in a smart card reader.

ITEM	DESCRIPTION
Registry key	EnableScPnP
Default values	No changes per operating system versions Enabled and not configured are equivalent
Policy management	Restart requirement: None Sign off requirement: None Policy conflicts: None

ITEM	DESCRIPTION
Notes and resources	This policy setting applies only to smart card drivers that have passed the Windows Hardware Quality Labs (WHQL) testing process.

Base CSP and Smart Card KSP registry keys

The following registry keys can be configured for the base cryptography service provider (CSP) and the smart card key storage provider (KSP). The following tables list the keys. All keys use the DWORD type.

The registry keys for the Base CSP are in the registry in

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography\Defaults\Provider\Microsoft Base Smart Card Crypto Provider.

The registry keys for the smart card KSP are in

HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Control\Cryptography\Providers\Microsoft Smart Card Key Storage Provider.

Registry keys for the base CSP and smart card KSP

REGISTRY KEY	DESCRIPTION
AllowPrivateExchangeKeyImport	A non-zero value allows RSA exchange (for example, encryption) private keys to be imported for use in key archival scenarios. Default value: 00000000
AllowPrivateSignatureKeyImport	A non-zero value allows RSA signature private keys to be imported for use in key archival scenarios. Default value: 00000000
DefaultPrivateKeyLenBits	Defines the default length for private keys, if desired. Default value: 00000400 Default key generation parameter: 1024-bit keys
RequireOnCardPrivateKeyGen	This key sets the flag that requires on-card private key generation (default). If this value is set, a key generated on a host can be imported into the smart card. This is used for smart cards that don't support on-card key generation or where key escrow is required. Default value: 00000000
TransactionTimeoutMilliseconds	Default timeout values allow you to specify whether transactions that take an excessive amount of time will fail. Default value: 000005dc The default timeout for holding transactions to the smart card is 1.5 seconds.

Additional registry keys for the smart card KSP

REGISTRY KEY	DESCRIPTION
AllowPrivateECDHEKeyImport	This value allows Ephemeral Elliptic Curve Diffie-Hellman (ECDHE) private keys to be imported for use in key archival scenarios. Default value: 00000000

REGISTRY KEY	DESCRIPTION
AllowPrivateECDSAKeyImport	This value allows Elliptic Curve Digital Signature Algorithm (ECDSA) private keys to be imported for use in key archival scenarios. Default value: 00000000

CRL checking registry keys

The following table lists the keys and the corresponding values to turn off certificate revocation list (CRL) checking at the Key Distribution Center (KDC) or client. To manage CRL checking, you must configure settings for both the KDC and the client.

CRL checking registry keys

REGISTRY KEY	DETAILS
HKEY_LOCAL_MACHINE\SYSTEM\CCS\Services\Kdc\UseCachedCRLOnlyAndIgnoreRevocationUnknownErrors	Type = DWORD Value = 1
HKEY_LOCAL_MACHINE\SYSTEM\CCS\Control\LSA\Kerberos\Parameters\UseCachedCRLOnlyAndIgnoreRevocationUnknownErrors	Type = DWORD Value = 1

Additional smart card Group Policy settings and registry keys

In a smart card deployment, additional Group Policy settings can be used to enhance ease-of-use or security. Two of these policy settings that can complement a smart card deployment are:

- Turning off delegation for computers
- Interactive logon: Do not require CTRL+ALT+DEL (not recommended)

The following smart card-related Group Policy settings are in Computer Configuration\Windows Settings\Security Settings\Local Policies\Security Options.

Local security policy settings

GROUP POLICY SETTING AND REGISTRY KEY	DEFAULT	DESCRIPTION
Interactive logon: Require smart card scforceoption	Disabled	This security policy setting requires users to sign in to a computer by using a smart card. Enabled Users can sign in to the computer only by using a smart card. Disabled Users can sign in to the computer by using any method.

GROUP POLICY SETTING AND REGISTRY KEY	DEFAULT	DESCRIPTION
Interactive logon: Smart card removal behavior scremoveoption	This policy setting isn't defined, which means that the system treats it as No Action .	This setting determines what happens when the smart card for a signed-in user is removed from the smart card reader. The options are: No Action Lock Workstation: The workstation is locked when the smart card is removed, so users can leave the area, take their smart card with them, and still maintain a protected session. Force Logoff: The user is automatically signed out when the smart card is removed. Disconnect if a Remote Desktop Services session: Removal of the smart card disconnects the session without signing out the user. The user can reinsert the smart card and resume the session later, or at another computer that's equipped with a smart card reader, without having to sign in again. If the session is local, this policy setting functions identically to the Lock Workstation option. Note: In earlier versions of Windows Server, Remote Desktop Services was called Terminal Services.

From the Local Security Policy Editor (secpol.msc), you can edit and apply system policies to manage credential delegation for local or domain computers.

The following smart card-related Group Policy settings are in Computer Configuration\Administrative Templates\System\Credentials Delegation.

Registry keys are in

HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Control\Lsa\Credssp\PolicyDefaults.

NOTE

In the following table, fresh credentials are those that you are prompted for when running an application.

Credential delegation policy settings

GROUP POLICY SETTING AND REGISTRY KEY	DEFAULT	DESCRIPTION
---------------------------------------	---------	-------------

GROUP POLICY SETTING AND REGISTRY KEY	DEFAULT	DESCRIPTION
<p>Allow Delegating Fresh Credentials</p> <p>AllowFreshCredentials</p>	<p>Not configured</p>	<p>This policy setting applies: When server authentication was achieved through a trusted X509 certificate or Kerberos protocol. To applications that use the CredSSP component (for example, Remote Desktop Services).</p> <p>Enabled: You can specify the servers where the user's fresh credentials can be delegated.</p> <p>Not configured: After proper mutual authentication, delegation of fresh credentials is permitted to Remote Desktop Services running on any computer.</p> <p>Disabled: Delegation of fresh credentials to any computer isn't permitted.</p> <p>Note: This policy setting can be set to one or more service principal names (SPNs). The SPN represents the target server where the user credentials can be delegated. A single wildcard character is permitted when specifying the SPN, for example: Use *TERMSRV/** for Remote Desktop Session Host (RD Session Host) running on any computer. Use <i>TERMSRV/host.humanresources.fabrikam.com</i> for RD Session Host running on the host.humanresources.fabrikam.com computer. Use <i>TERMSRV/*.humanresources.fabrikam.com</i> for RD Session Host running on all computers in .humanresources.fabrikam.com</p>

GROUP POLICY SETTING AND REGISTRY KEY	DEFAULT	DESCRIPTION
<p>Allow Delegating Fresh Credentials with NTLM-only Server Authentication</p> <p>AllowFreshCredentialsWhenNTLM Only</p>	Not configured	<p>This policy setting applies: When server authentication was achieved by using NTLM. To applications that use the CredSSP component (for example, Remote Desktop).</p> <p>Enabled: You can specify the servers where the user's fresh credentials can be delegated. Not configured: After proper mutual authentication, delegation of fresh credentials is permitted to RD Session Host running on any computer (TERMSRV/*). Disabled: Delegation of fresh credentials isn't permitted to any computer.</p> <p>Note: This policy setting can be set to one or more SPNs. The SPN represents the target server where the user credentials can be delegated. A single wildcard character (*) is permitted when specifying the SPN. See the Allow Delegating Fresh Credentials policy setting description for examples.</p>
<p>Deny Delegating Fresh Credentials</p> <p>DenyFreshCredentials</p>	Not configured	<p>This policy setting applies to applications that use the CredSSP component (for example, Remote Desktop).</p> <p>Enabled: You can specify the servers where the user's fresh credentials can't be delegated. Disabled or Not configured: A server is not specified.</p> <p>Note: This policy setting can be set to one or more SPNs. The SPN represents the target server where the user credentials can't be delegated. A single wildcard character (*) is permitted when specifying the SPN. For examples, see the "Allow delegating fresh credentials" policy setting.</p>

If you're using Remote Desktop Services with smart card logon, you can't delegate default and saved credentials. The registry keys in the following table, which are at **HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Control\Lsa\Credssp\PolicyDefaults**, and the corresponding Group Policy settings are ignored.

REGISTRY KEY	CORRESPONDING GROUP POLICY SETTING
AllowDefaultCredentials	Allow Delegating Default Credentials

REGISTRY KEY	CORRESPONDING GROUP POLICY SETTING
AllowDefaultCredentialsWhenNTLMOnly	Allow Delegating Default Credentials with NTLM-only Server Authentication
AllowSavedCredentials	Allow Delegating Saved Credentials
AllowSavedCredentialsWhenNTLMOnly	Allow Delegating Saved Credentials with NTLM-only Server Authentication

See also

[Smart Card Technical Reference](#)

Smart Card Events

7/1/2022 • 13 minutes to read • [Edit Online](#)

Applies To: Windows 10, Windows 11, Windows Server 2016 and above

This topic for the IT professional and smart card developer describes events that are related to smart card deployment and development.

A number of events can be used to monitor smart card activities on a computer, including installation, use, and errors. The following sections describe the events and information that can be used to manage smart cards in an organization.

- [Smart card reader name](#)
- [Smart card warning events](#)
- [Smart card error events](#)
- [Smart card Plug and Play events](#)

Smart card reader name

The Smart Card resource manager does not use the device name from Device Manager to describe a smart card reader. Instead, the name is constructed from three device attributes that are queried directly from the smart card reader driver.

The following three attributes are used to construct the smart card reader name:

- Vendor name
- Interface device type
- Device unit

The smart card reader device name is constructed in the form *<VendorName> <Type> <DeviceUnit>*. For example 'Contoso Smart Card Reader 0' is constructed from the following information:

- Vendor name: Contoso
- Interface device type: Smart Card Reader
- Device unit: 0

Smart card warning events

Note IOCTL in the following table refers to input and output control.

EVENT ID	WARNING MESSAGE	DESCRIPTION
----------	-----------------	-------------

EVENT ID	WARNING MESSAGE	DESCRIPTION
620	Smart Card Resource Manager was unable to cancel IOCTL %3 for reader '%2': %1. The reader may no longer be responding. If this error persists, your smart card or reader may not be functioning correctly. %n%nCommand Header: %4	<p>This occurs if the resource manager attempts to cancel a command to the smart card reader when the smart card service is shutting down or after a smart card is removed from the smart card reader and the command could not to be canceled. This can leave the smart card reader in an unusable state until it is removed from the computer or the computer is restarted.</p> <p>%1 = Windows error code %2 = Smart card reader name %3 = IOCTL being canceled %4 = First 4 bytes of the command that was sent to the smart card</p>
619	Smart Card Reader '%2' has not responded to IOCTL %3 in %1 seconds. If this error persists, your smart card or reader may not be functioning correctly. %n%nCommand Header: %4	<p>This occurs when a reader has not responded to an IOCTL after an unusually long period of time. Currently, this error is sent after a reader does not respond for 150 seconds. This can leave the smart card reader in an unusable state until it is removed from the computer or the computer is restarted.</p> <p>%1 = Number of seconds the IOCTL has been waiting %2 = Smart card reader name %3 = IOCTL sent %4 = First 4 bytes of the command that was sent to the smart card</p>

Smart card error events

EVENT ID	ERROR MESSAGE	DESCRIPTION
202	Failed to initialize Server Application	An error occurred, and the service cannot initialize properly. Restarting the computer may resolve the issue.
203	Server Control has no memory for reader reference object.	This is an internal, unrecoverable error that indicates a failure in the smart card service. The most common cause is limited computer resources. Restarting the computer may resolve the issue.
204	Server Control failed to create shutdown event: %1	<p>This is an internal, unrecoverable error that indicates a failure in the smart card service. The most common cause is limited computer resources. Restarting the computer may resolve the issue.</p> <p>%1 = Windows error code</p>

EVENT ID	ERROR MESSAGE	DESCRIPTION
205	Reader object has duplicate name: %1	There are two smart card readers that have the same name. Remove the smart card reader that is causing this error message. %1 = Name of the smart card reader that is duplicated
206	Failed to create global reader change event.	This is an internal, unrecoverable error that indicates a failure in the smart card service. The most common cause is limited computer resources. Restarting the computer may resolve the issue.
401	Reader shutdown exception from eject smart card command	A smart card reader could not eject a smart card while the smart card reader was shutting down.
406	Reader object cannot Identify Device	A smart card reader did not properly respond to a request for information about the device, which is required for constructing the smart card reader name. The smart card reader will not be recognized by the service until it is removed from the computer and reinserted or until the computer is restarted.
502	Initialization of Service Status Critical Section failed	This is an internal, unrecoverable error that indicates a failure in the smart card service. The most common cause is limited computer resources. Restarting the computer may resolve the issue.
504	Resource Manager cannot create shutdown event flag: %1	This is an internal, unrecoverable error that indicates a failure in the smart card service. The most common cause is limited computer resources. Restarting the computer may resolve the issue. %1 = Windows error code
506	Smart Card Resource Manager failed to register service: %1	This is an internal, unrecoverable error that indicates a failure in the smart card service. The most common cause is limited computer resources. Restarting the computer may resolve the issue. %1 = Windows error code
506	Smart Card Resource Manager received unexpected exception from PnP event %1	An attempt to add a Plug and Play reader failed. The device may already be in use or may be defective. To resolve this error message, try to add the device again or restart the computer. %1 = The affected handle name

EVENT ID	ERROR MESSAGE	DESCRIPTION
507	No memory available for Service Status Critical Section	There is not enough system memory available. This prevents the service from managing the status. Restarting the computer may resolve the issue.
508	Smart Card Resource Manager received unexpected exception from PnP event %1	An attempt to add a Plug and Play reader failed. The device may already be in use or may be defective. To resolve this error message, try to add the device again or restart the computer. %1 = The affected handle name
509	Smart Card Resource Manager received unexpected exception from PnP event %1	An attempt to add a Plug and Play reader failed. The device may already be in use or may be defective. To resolve this error message, try to add the device again or restart the computer. %1 = The affected handle name
510	Smart Card Resource Manager received NULL handle from PnP event %1	An attempt to add a Plug and Play smart card reader failed. The device may already be in use or may be defective. To resolve this error message, try to add the device again or restart the computer. %1 = The affected handle name
511	Smart Card Resource Manager received unexpected exception from PnP event %1	An attempt to add a Plug and Play reader failed. The device may already be in use or may be defective. To resolve this error message, try to add the device again or restart the computer. %1 = The affected handle name
512	Smart Card Resource Manager received NULL handle from PnP event %1	An attempt to add a Plug and Play smart card reader failed. The device may already be in use or may be defective. To resolve this error message, try to add the device again or restart the computer. %1 = The affected handle name
513	Smart Card Resource Manager received unexpected exception from PnP event %1	An attempt to add a Plug and Play reader failed. The device may already be in use or may be defective. To resolve this error message, try to add the device again or restart the computer. %1 = The affected handle name

EVENT ID	ERROR MESSAGE	DESCRIPTION
514	Smart Card Resource Manager failed to add reader %2: %1	<p>This is an internal, unrecoverable error that indicates a failure in the smart card service. The most common cause is limited computer resources. Restarting the computer may resolve the issue.</p> <p>%1 = Windows error code %2 = Smart card reader name</p>
515	Smart Card Resource Manager failed to declare state: %1	<p>This is an internal unrecoverable error that indicates a failure in the smart card service. The smart card service may not operate properly. Restarting the service or computer may resolve this issue.</p> <p>%1 = Windows error code</p>
516	Smart Card Resource Manager Failed to declare shutdown: %1	<p>This is an internal, unrecoverable error that indicates a failure in the smart card service. The smart card service may not be able to stop. Restarting the computer may resolve this issue.</p> <p>%1 = Windows error code</p>
517	Smart Card Resource Manager received unexpected exception attempting to add reader %1	<p>This is an internal, unrecoverable error that indicates a failure in the smart card service. The most common cause is limited computer resources. Restarting the computer may resolve the issue.</p> <p>%1 = Smart card reader name</p>
521	Smart Card Resource Manager received NULL handle from PnP event %1	<p>An attempt to add a Plug and Play smart card reader failed. The device may already be in use or may be defective. To resolve this error message, try to add the device again or restart the computer.</p> <p>%1 = The affected handle name</p>
523	Smart Card Resource Manager received NULL handle from PnP event %1	<p>An attempt to add a Plug and Play smart card reader failed. The device may already be in use or may be defective. To resolve this error message, try to add the device again or restart the computer.</p> <p>%1 = The affected handle name</p>
602	WDM Reader driver initialization cannot open reader device: %1	<p>The service cannot open a communication channel with the smart card reader. You cannot use the smart card reader until the issue is resolved.</p> <p>%1 = Windows error code</p>

EVENT ID	ERROR MESSAGE	DESCRIPTION
603	WDM Reader driver initialization has no memory available to control device %1	There is not enough system memory available. This prevents the service from managing the smart card reader that was added. Restarting the computer may resolve the issue. %1 = Name of affected reader
604	Server control cannot set reader removal event: %1	This is an internal, unrecoverable error that indicates a failure in the smart card service. The most common cause is limited computer resources. Restarting the computer may resolve the issue. %1 = Windows error code
605	Reader object failed to create overlapped event: %1	This is an internal, unrecoverable error that indicates a failure in the smart card service. The most common cause is limited computer resources. Restarting the computer may resolve the issue. %1 = Windows error code
606	Reader object failed to create removal event: %1	This is an internal, unrecoverable error that indicates a failure in the smart card service. The most common cause is limited computer resources. Restarting the computer may resolve the issue. %1 = Windows error code
607	Reader object failed to start monitor thread: %1	This is an internal, unrecoverable error that indicates a failure in the smart card service. The most common cause is limited computer resources. Restarting the computer may resolve the issue. %1 = Windows error code
608	Reader monitor failed to create power down timer: %1	This is an internal, unrecoverable error that indicates a failure in the smart card service. The most common cause is limited computer resources. Restarting the computer may resolve the issue. %1 = Windows error code
609	Reader monitor failed to create overlapped event: %1	This is an internal, unrecoverable error that indicates a failure in the smart card service. The most common cause is limited computer resources. Restarting the computer may resolve the issue. %1 = Windows error code

EVENT ID	ERROR MESSAGE	DESCRIPTION
610	Smart Card Reader '%2' rejected IOCTL %3: %1 If this error persists, your smart card or reader may not be functioning correctly.%n%nCommand Header: %4	<p>The reader cannot successfully transmit the indicated IOCTL to the smart card. This can indicate hardware failure, but this error can also occur if a smart card or smart card reader is removed from the system while an operation is in progress.</p> <p>%1 = Windows error code %2 = Name of the smart card reader %3 = IOCTL that was sent %4 = First 4 bytes of the command sent to the smart card</p> <p>These events are caused by legacy functionality in the smart card stack. It can be ignored if there is no noticeable failure in the smart card usage scenarios. You might also see this error if your eSIM is recognized as a smartcard controller.</p>
611	Smart Card Reader initialization failed	<p>This is an internal, unrecoverable error that indicates a failure in the smart card service. The most common cause is limited computer resources. Restarting the computer may resolve this issue.</p>
612	Reader insertion monitor error retry threshold reached: %1	<p>This occurs when a smart card reader fails several times to respond properly to the IOCTL, which indicates whether a smart card is present in the reader. The smart card reader is marked as defective, and it is not recognized by the service until it is removed from the computer and reinserted or until the computer is restarted.</p> <p>%1 = Windows error code</p>
615	Reader removal monitor error retry threshold reached: %1	<p>This occurs when a smart card reader fails several times to respond properly to the IOCTL, which indicates whether a smart card is present in the reader. The smart card reader is marked as defective, and it is not recognized by the service until it is removed from the computer and reinserted or until the computer is restarted.</p> <p>%1 = Windows error code</p>

EVENT ID	ERROR MESSAGE	DESCRIPTION
616	Reader monitor '%2' received uncaught error code: %1	This occurs when a smart card reader fails several times to respond properly to the IOCTL, which indicates whether a smart card is present in the reader. The smart card reader is marked as defective, and it is not recognized by the service until it is removed from the computer and reinserted or until the computer is restarted. %1 = Windows error code %2 = Reader name
617	Reader monitor '%1' exception -- exiting thread	An unknown error occurred while monitoring a smart card reader for smart card insertions and removals. The smart card reader is marked as defective, and it is not recognized by the service until it is removed from the computer and reinserted or until the computer is restarted. %1 = Smart card reader name
618	Smart Card Resource Manager encountered an unrecoverable internal error.	This is an internal, unrecoverable error that indicates a failure in the smart card service. The most common cause is limited computer resources. Restarting the computer may resolve the issue.
621	Server Control failed to access start event: %1	This is an internal, unrecoverable error that indicates a failure in the smart card service. The most common cause is limited computer resources. Restarting the computer may resolve the issue. %1 = Windows error code These events are caused by legacy functionality in the smart card stack. It can be ignored if there is no noticeable failure in the smart card usage scenarios.
622	Server Control failed to access stop event: %1	This is an internal, unrecoverable error that indicates a failure in the smart card service. The most common cause is limited computer resources. Restarting the computer may resolve the issue. %1 = Windows error code

Smart card Plug and Play events

EVENT ID	EVENT TYPE	EVENT MESSAGE	DESCRIPTION
----------	------------	---------------	-------------

EVENT ID	EVENT TYPE	EVENT MESSAGE	DESCRIPTION
1000	Error	Could not get device ID for smart card in reader %1. The return code is %2.	Smart card Plug and Play could not obtain the device ID for the smart card. This information is required to determine the correct driver. The smart card may be defective. %1 = Smart card reader name %2 = Windows error code
1001	Information	Software successfully installed for smart card in reader %1. The smart card name is %2.	Smart card Plug and Play successfully installed a minidriver for the inserted card. %1 = Smart card reader name %2 = Name of new smart card device

See also

[Smart Card Technical Reference](#)

Virtual Smart Card Overview

7/1/2022 • 8 minutes to read • [Edit Online](#)

Applies To: Windows 10, Windows Server 2016

This topic for IT professional provides an overview of the virtual smart card technology that was developed by Microsoft and includes [links to additional topics](#) to help you evaluate, plan, provision, and administer virtual smart cards.

Did you mean...

- [Smart Cards](#)

NOTE

[Windows Hello for Business](#) is the modern, two-factor authentication for Windows 10. Microsoft will be deprecating virtual smart cards in the future, but no date has been set at this time. Customers using Windows 10 and virtual smart cards should move to Windows Hello for Business. Microsoft will publish the date early to ensure customers have adequate lead time to move to Windows Hello for Business. We recommend that new Windows 10 deployments use Windows Hello for Business. Virtual smart cards remain supported for Windows 7 and Windows 8.

Feature description

Virtual smart card technology from Microsoft offers comparable security benefits to physical smart cards by using two-factor authentication. Virtual smart cards emulate the functionality of physical smart cards, but they use the Trusted Platform Module (TPM) chip that is available on computers in many organizations, rather than requiring the use of a separate physical smart card and reader. Virtual smart cards are created in the TPM, where the keys that are used for authentication are stored in cryptographically secured hardware.

By utilizing TPM devices that provide the same cryptographic capabilities as physical smart cards, virtual smart cards accomplish the three key properties that are desired for smart cards: non-exportability, isolated cryptography, and anti-hammering.

Practical applications

Virtual smart cards are functionally similar to physical smart cards and appear in Windows as smart cards that are always-inserted. Virtual smart cards can be used for authentication to external resources, protection of data by secure encryption, and integrity through reliable signing. They are easily deployed by using in-house methods or a purchased solution, and they can become a full replacement for other methods of strong authentication in a corporate setting of any scale.

Authentication use cases

Two-factor authentication–based remote access

After a user has a fully functional TPM virtual smart card, provisioned with a sign-in certificate, the certificate is used to gain strongly authenticated access to corporate resources. When the proper certificate is provisioned to the virtual card, the user need only provide the PIN for the virtual smart card, as if it was a physical smart card, to sign in to the domain.

In practice, this is as easy as entering a password to access the system. Technically, it is far more secure. Using the virtual smart card to access the system proves to the domain that the user who is requesting authentication

has possession of the personal computer upon which the card has been provisioned and knows the virtual smart card PIN. Because this request could not have possibly originated from a system other than the system certified by the domain for this user's access, and the user could not have initiated the request without knowing the PIN, a strong two-factor authentication is established.

Client authentication

Virtual smart cards can also be used for client authentication by using Secure Socket Layer (SSL) or a similar technology. Similar to domain access with a virtual smart card, an authentication certificate can be provisioned for the virtual smart card, provided to a remote service, as requested in the client authentication process. This adheres to the principles of two-factor authentication because the certificate is only accessible from the computer that hosts the virtual smart card, and the user is required to enter the PIN for initial access to the card.

Virtual smart card redirection for remote desktop connections

The concept of two-factor authentication associated with virtual smart cards relies on the proximity of users to the computers that they access domain resources through. Therefore, when a user remotely connects to a computer that is hosting virtual smart cards, the virtual smart cards that are located on the remote computer cannot be used during the remote session. However, the virtual smart cards that are stored on the connecting computer (which is under physical control of the user) are loaded onto the remote computer, and they can be used as if they were installed by using the remote computer's TPM. This extends a user's privileges to the remote computer, while maintaining the principles of two-factor authentication.

Windows To Go and virtual smart cards

Virtual smart cards work well with Windows To Go, where a user can boot into a supported version of Windows from a compatible removable storage device. A virtual smart card can be created for the user, and it is tied to the TPM on the physical host computer to which the removable storage device is connected. When the user boots the operating system from a different physical computer, the virtual smart card will not be available. This can be used for scenarios when a single physical computer is shared by many users. Each user can be given a removable storage device for Windows To Go, which has a virtual smart card provisioned for the user. This way, users are only able to access their personal virtual smart card.

Confidentiality use cases

S/MIME email encryption

Physical smart cards are designed to hold private keys that can be used for email encryption and decryption. This functionality also exists in virtual smart cards. By using S/MIME with a user's public key to encrypt email, the sender of an email can be assured that only the person with the corresponding private key will be able to decrypt the email. This assurance is a result of the non-exportability of the private key. It never exists within reach of malicious software, and it remains protected by the TPM—even during decryption.

BitLocker for data volumes

sBitLocker Drive Encryption technology makes use of symmetric-key encryption to protect the content of a user's hard drive. This ensures that if the physical ownership of a hard drive is compromised, an adversary will not be able to read data off the drive. The key used to encrypt the drive can be stored in a virtual smart card, which necessitates knowledge of the virtual smart card PIN to access the drive and possession of the computer that is hosting the TPM virtual smart card. If the drive is obtained without access to the TPM that hosts the virtual smart card, any brute force attack will be very difficult.

BitLocker can also be used to encrypt portable drives, which involves storing keys in virtual smart cards. In this scenario (unlike using BitLocker with a physical smart card), the encrypted drive can be used only when it is connected to the host for the virtual smart card that is used to encrypt the drive, because the BitLocker key is only accessible from this computer. However, this method can be useful to ensure the security of backup drives and personal storage uses outside the main hard drive.

Data integrity use case

Signing data

To verify authorship of data, a user can sign it by using a private key that is stored in the virtual smart card. Digital signatures confirm the integrity and origin of the data. If the key is stored in an operating system that is accessible, a malicious user could access it and use it to modify already signed data or to spoof the key owner's identity. However, if this key is stored in a virtual smart card, it can be used only to sign data on the host computer. It cannot be exported to other systems (intentionally or unintentionally, such as with malware theft). This makes digital signatures far more secure than other methods for private key storage.

New and changed functionality as of Windows 8.1

Enhancements in Windows 8.1 enabled developers to build Microsoft Store apps to create and manage virtual smart cards.

The DCOM Interfaces for Trusted Platform Module (TPM) Virtual Smart Card device management protocol provides a Distributed Component Object Model (DCOM) Remote Protocol interface used for creating and destroying virtual smart cards. A virtual smart card is a device that presents a device interface complying with the PC/SC specification for PC-connected interface devices to its host operating system (OS) platform. This protocol does not assume anything about the underlying implementation of virtual smart card devices. In particular, while it is primarily intended for the management of virtual smart cards based on TPMs, it can also be used to manage other types of virtual smart cards.

What value does this change add?

Starting with Windows 8.1, application developers can build into their apps the following virtual smart card maintenance capabilities to relieve some of your administrative burdens.

- Create a new virtual smart card or select a virtual smart card from the list of available virtual smart cards on the system. Identify the one that the application is supposed to work with.
- Personalize the virtual smart card.
- Change the admin key.
- Diversify the admin key which allows the user to unblock the PIN in a PIN-blocked scenario.
- Change the PIN.
- Reset or Unblock the PIN.
- Destroy the virtual smart card.

What works differently?

Starting with Windows 8.1, Microsoft Store app developers are able to build apps that have the capability to prompt the user to reset or unblock and change a virtual smart card PIN. This places more responsibility on the user to maintain their virtual smart card but it can also provide a more consistent user experience and administration experience in your organization.

For more information about developing Microsoft Store apps with these capabilities, see [Trusted Platform Module Virtual Smart Card Management Protocol](#).

For more information about managing these capabilities in virtual smart cards, see [Understanding and Evaluating Virtual Smart Cards](#).

Hardware requirements

To use the virtual smart card technology, TPM 1.2 is the minimum required for computers running Windows 10

or Windows Server 2016.

Software requirements

To use the virtual smart card technology, computers must be running one of the following operating systems:

- Windows Server 2016
- Windows Server 2012 R2
- Windows Server 2012
- Windows 10
- Windows 8.1
- Windows 8

See also

- [Understanding and Evaluating Virtual Smart Cards](#)
- [Get Started with Virtual Smart Cards: Walkthrough Guide](#)
- [Use Virtual Smart Cards](#)
- [Deploy Virtual Smart Cards](#)
- [Evaluate Virtual Smart Card Security](#)
- [Tpmvscmgr](#)

Understanding and Evaluating Virtual Smart Cards

7/1/2022 • 13 minutes to read • [Edit Online](#)

Applies To: Windows 10, Windows Server 2016

This topic for the IT professional describes the virtual smart card technology that was developed by Microsoft; suggests how it can fit into your authentication design; and provides links to additional resources that you can use to design, deploy, and troubleshoot virtual smart cards.

Virtual smart card technology uses cryptographic keys that are stored on computers that have the Trusted Platform Module (TPM) installed. Virtual smart cards offer comparable security benefits to conventional smart cards by using two-factor authentication. The technology also offers more convenience for users and has a lower cost to deploy. By utilizing TPM devices that provide the same cryptographic capabilities as conventional smart cards, virtual smart cards accomplish the three key properties that are desired for smart cards: non-exportability, isolated cryptography, and anti-hammering.

Virtual smart cards are functionally similar to physical smart cards. They appear as always-inserted smart cards, and they can be used for authentication to external resources, protection of data by secure encryption, and integrity through reliable signing. Because TPM-enabled hardware is readily available and virtual smart cards can be easily deployed by using existing certificate enrollment methods, virtual smart cards can become a full replacement for other methods of strong authentication in a corporate setting of any scale.

This topic contains the following sections:

- [Comparing virtual smart cards with physical smart cards](#): Compares properties, functional aspects, security, and cost.
- [Authentication design options](#): Describes how passwords, smart cards, and virtual smart cards can be used to reach authentication goals in your organization.
- [See also](#): Links to other topics that can help you design, deploy, and troubleshoot virtual smart cards.

Comparing virtual smart cards with physical smart cards

Virtual smart cards function much like physical smart cards, but they differ in that they protect private keys by using the TPM of the computer instead of smart card media.

A virtual smart card appears to applications as a conventional smart card. Private keys in the virtual smart card are protected, not by isolation of physical memory, but by the cryptographic capabilities of the TPM. All sensitive information is encrypted by using the TPM and then stored on the hard drive in its encrypted form.

All cryptographic operations occur in the secure, isolated environment of the TPM, and the unencrypted private keys are never used outside this environment. So like physical smart cards, virtual smart cards remain secure from any malware on the host. Additionally, if the hard drive is compromised in some way, a malicious user will not be able to access keys that are stored in the virtual smart card because they are securely encrypted by using the TPM. Keys can also be protected by BitLocker Drive Encryption.

Virtual smart cards maintain the three key properties of physical smart cards:

- **Non-exportability**: Because all private information on the virtual smart card is encrypted by using the TPM on the host computer, it cannot be used on a different computer with a different TPM. Additionally, TPMs are designed to be tamper-resistant and non-exportable, so a malicious user cannot reverse engineer an identical TPM or install the same TPM on a different computer. For more information, see

Evaluate Virtual Smart Card Security.

- **Isolated cryptography:** TPMs provide the same properties of isolated cryptography that are offered by physical smart cards, and this is utilized by virtual smart cards. Unencrypted copies of private keys are loaded only within the TPM and never into memory that is accessible by the operating system. All cryptographic operations with these private keys occur inside the TPM.
- **Anti-hammering:** If a user enters a PIN incorrectly, the virtual smart card responds by using the anti-hammering logic of the TPM, which rejects further attempts for a period of time instead of blocking the card. This is also known as lockout. For more information, see [Evaluate Virtual Smart Card Security](#).

The following subsections compare the functionality, security, and cost of virtual smart cards and physical smart cards.

Functionality

The virtual smart card system that was designed by Microsoft closely mimics the functionality of conventional smart cards. The most striking difference to the end user is that the virtual smart card is essentially a smart card that is always inserted into the computer. There is no method to export the user's virtual smart card for use on other computers, which adds to the security of virtual smart cards. If a user requires access to network resources on multiple computers, multiple virtual smart cards can be issued for that user. Additionally, a computer that is shared among multiple users can host multiple virtual smart cards for different users.

The basic user experience for a virtual smart card is as simple as using a password to access a network. Because the smart card is loaded by default, the user must simply enter the PIN that is tied to the card to gain access. Users are no longer required to carry cards and readers or to take physical action to use the card.

Additionally, although the anti-hammering functionality of the virtual smart card is equally secure to that of a physical smart card, virtual smart card users are never required to contact an administrator to unblock the card. Instead, they simply wait a period of time (depending on the TPM specifications) before they reattempt to enter the PIN. Alternatively, the administrator can reset the lockout by providing owner authentication data to the TPM on the host computer.

Security

Physical smart cards and virtual smart cards offer comparable levels of security. They both implement two-factor authentication for using network resources. However, they differ in certain aspects, including physical security and the practicality of an attack. Due to their compact and portable design, conventional smart cards are most frequently kept close to their intended user. They offer little opportunity for acquisition by a potential adversary, so any sort of interaction with the card is difficult without committing some variety of theft.

TPM virtual smart cards, however, reside on a user's computer that may frequently be left unattended, which provides an opportunity for a malicious user to hammer the TPM. Although virtual smart cards are fully protected from hammering (as are physical smart cards), this accessibility makes the logistics of an attack somewhat simpler. Additionally, the anti-hammering behavior of a TPM smart card differs in that it only presents a time delay in response to repeated PIN failures, as opposed to fully blocking the user.

However, there are several advantages provided by virtual smart cards to mitigate these slight security deficits. Most importantly, a virtual smart card is much less likely to be lost. Virtual smart cards are integrated into computers and devices that the user already owns for other purposes and has incentive to keep safe. If the computer or device that hosts the virtual smart card is lost or stolen, a user will more immediately notice its loss than the loss of a physical smart card. When a computer or device is identified as lost, the user can notify the administrator of the system, who can revoke the certificate that is associated with the virtual smart card on that device. This precludes any future unauthorized access on that computer or device if the PIN for the virtual smart card is compromised.

Cost

If a company wants to deploy physical smart cards, they need to purchase smart cards and smart card readers for all employees. Although relatively inexpensive options can be found, options that ensure the three key properties of smart card security (most notably, non-exportability) are more expensive. If employees have computers with a built-in TPM, virtual smart cards can be deployed with no additional material costs. These computers and devices are relatively common in the market.

Additionally, the maintenance cost of virtual smart cards is less than that for physical smart cards, which are easily lost, stolen, or broken from normal wear. TPM virtual smart cards are only lost or broken if the host computer or device is lost or broken, which in most cases is much less frequently.

Comparison summary

PHYSICAL SMART CARDS	TPM VIRTUAL SMART CARDS
Protects private keys by using the built-in cryptographic functionality of the card.	Protects private keys by using the cryptographic functionality of the TPM.
Stores private keys in isolated non-volatile memory on the card, which means that access to private keys is only from the card, and access is never allowed to the operating system.	Stores encrypted private keys on the hard drive. The encryption ensures that these keys can only be decrypted and used in the TPM, not in the accessible memory of the operating system.
Guarantees non-exportability through the card manufacturer, which includes isolating private information from operating system access.	Guarantees non-exportability through the TPM manufacturer, which includes the inability of an adversary to replicate or remove the TPM.
Performs and isolates cryptographic operations within the built-in capabilities of the card.	Performs and isolates cryptographic operations in the TPM of the user's computer or device.
Provides anti-hammering through the card. After a certain number of failed PIN entry attempts, the card blocks further access until administrative action is taken.	Provides anti-hammering through the TPM. Successive failed attempts increase the device lockout time (the time the user has to wait before trying again). This can be reset by an administrator.
Requires that users carry their smart card and smart card reader with them to access network resources.	Allows users to access their TPM-enabled computers or devices, and potentially access the network, without additional equipment.
Enables credential portability by inserting the smart card into smart card readers that are attached to other computers.	Prevents exporting credentials from a given computer or device. However, virtual smart cards can be issued for the same user on multiple computers or devices by using additional certificates.
Enables multiple users to access network resources through the same computer by inserting their personal smart cards.	Enables multiple users to access network resources through the same computer or device by issuing a virtual smart card for each user on that computer or device.
Requires the user to carry the card, making it more difficult for an attacker to access the device and launch a hammering attempt.	Stores virtual smart card on the user's computer, which may be left unattended and allow a greater risk window for hammering attempts.
Provides a generally single-purpose device that is carried explicitly for the purpose of authentication. The smart card can be easily misplaced or forgotten.	Installs the virtual smart card on a device that has other purposes for the user, so the user has greater incentive to be responsible for the computer or device.

PHYSICAL SMART CARDS	TPM VIRTUAL SMART CARDS
Alerts users that their card is lost or stolen only when they need to sign in and notice it is missing.	Installs the virtual smart card on a device that the user likely needs for other purposes, so users will notice its loss much more quickly. This reduces the associated risk window.
Requires companies to invest in smart cards and smart card readers for all employees.	Requires that companies ensure all employees have TPM-enabled computers, which are relatively common.
Enables using a smart card removal policy to affect system behavior when the smart card is removed. For example, the policy can dictate if the user's sign-in session is locked or terminated when the user removes the card.	Eliminates the necessity for a smart card removal policy because a TPM virtual smart card is always present and cannot be removed from the computer.

Authentication design options

The following section presents several commonly used options and their respective strengths and weaknesses, which organizations can consider for authentication.

Passwords

A password is a secret string of characters that is tied to the identification credentials for a user's account. This establishes the user's identity. Although passwords are the most commonly used form of authentication, they are also the weakest. In a system where passwords are used as the sole method of user authentication, only individuals who know their passwords are considered valid users.

Password authentication places a great deal of responsibility on the user. Passwords must be sufficiently complex so they cannot be easily guessed, but they must be simple enough to be committed to memory and not stored in a physical location. Even if this balance is successfully achieved, a wide variety of attacks exist (such as brute force attacks, eavesdropping, and social engineering tactics) where a malicious user can acquire a user's password and impersonate that person's identity. A user often will not realize that the password is compromised, which makes it is easy for a malicious user to maintain access to a system if a valid password has been obtained.

One-time passwords

A one-time password (OTP) is similar to a traditional password, but it is more secure in that it can be used only once to authenticate a user. The method for determining each new password varies by implementation. However, assuming a secure deployment of each new password, OTPs have several advantages over the classic password model of authentication. Most importantly, if a given OTP token is intercepted in transmission between the user and the system, the interceptor cannot use it for any future transactions. Similarly, if a malicious user obtains a valid user's OTP, the interceptor will have limited access to the system (only one session).

Smart cards

Smart cards are physical authentication devices, which improve on the concept of a password by requiring that users actually have their smart card device with them to access the system, in addition to knowing the PIN that provides access to the smart card. Smart cards have three key properties that help maintain their security:

- **Non-exportability:** Information stored on the card, such as the user's private keys, cannot be extracted from one device and used in another medium.
- **Isolated cryptography:** Any cryptographic operations that are related to the card (such as secure encryption and decryption of data) occur in a cryptographic processor on the card, so malicious software on the host computer cannot observe the transactions.

- **Anti-hammering:** To prevent access to the card by a brute-force attack, a set number of consecutive unsuccessful PIN entry attempts blocks the card until administrative action is taken.

Smart cards provide greatly enhanced security over passwords alone, because it is much more difficult for a malicious user to gain and maintain access to a system. Most importantly, access to a smart card system requires that users have a valid card and that they know the PIN that provides access to that card. It is extremely difficult for a thief to acquire the card and the PIN.

Additional security is achieved by the singular nature of the card because only one copy of the card exists, only one individual can use the sign-in credentials, and users will quickly notice if the card has been lost or stolen. This greatly reduces the risk window of credential theft when compared to using a password alone.

Unfortunately, this additional security comes with added material and support costs. Traditional smart cards are expensive to purchase (cards and card readers must be supplied to employees), and they also can be easily misplaced or stolen.

Virtual smart cards

To address these issues, virtual smart cards emulate the functionality of traditional smart cards, but instead of requiring the purchase of additional hardware, they utilize technology that users already own and are more likely to have with them at all times. Theoretically, any device that can provide the three key properties of smart cards (non-exportability, isolated cryptography, and anti-hammering) can be commissioned as a virtual smart card. However, the virtual smart card platform developed by Microsoft is currently limited to the use of the Trusted Platform Module (TPM) chip, which is installed on most modern computers.

Virtual smart cards that utilize a TPM provide the three main security principles of traditional smart cards (non-exportability, isolated cryptography, and anti-hammering). They are also less expensive to implement and more convenient for users. Because many corporate computers already have a built-in TPM, there is no cost associated with purchasing new hardware. The user's possession of a computer or device is equivalent to the possession of a smart card, and a user's identity cannot be assumed from any other computer or device without administrative provisioning of further credentials. Thus, two-factor authentication is achieved because the user must have a computer that is set up with a virtual smart card and know the PIN to use the virtual smart card.

See also

- [Get Started with Virtual Smart Cards: Walkthrough Guide](#)
- [Use Virtual Smart Cards](#)
- [Deploy Virtual Smart Cards](#)
- [Evaluate Virtual Smart Card Security](#)

Get Started with Virtual Smart Cards: Walkthrough Guide

7/1/2022 • 5 minutes to read • [Edit Online](#)

Applies To: Windows 10, Windows Server 2016

This topic for the IT professional describes how to set up a basic test environment for using TPM virtual smart cards.

Virtual smart cards are a technology from Microsoft, which offer comparable security benefits in two-factor authentication to physical smart cards. They also offer more convenience for users and lower cost for organizations to deploy. By utilizing Trusted Platform Module (TPM) devices that provide the same cryptographic capabilities as physical smart cards, virtual smart cards accomplish the three key properties that are desired by smart cards: non-exportability, isolated cryptography, and anti-hammering.

This step-by-step walkthrough shows you how to set up a basic test environment for using TPM virtual smart cards. After you complete this walkthrough, you will have a functional virtual smart card installed on the Windows computer.

Time requirements

You should be able to complete this walkthrough in less than one hour, excluding installing software and setting up the test domain.

Walkthrough steps

- [Prerequisites](#)
- [Step 1: Create the certificate template](#)
- [Step 2: Create the TPM virtual smart card](#)
- [Step 3: Enroll for the certificate on the TPM Virtual Smart Card](#)

Important This basic configuration is for test purposes only. It is not intended for use in a production environment.

Prerequisites

You will need:

- A computer running Windows 10 with an installed and fully functional TPM (version 1.2 or version 2.0).
- A test domain to which the computer listed above can be joined.
- Access to a server in that domain with a fully installed and running certification authority (CA).

Step 1: Create the certificate template

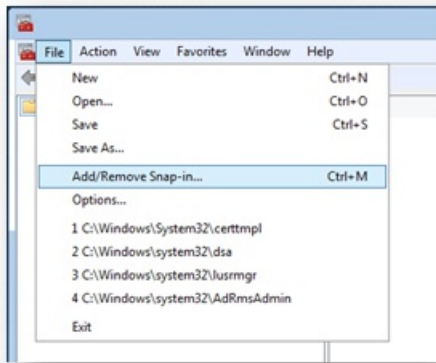
On your domain server, you need to create a template for the certificate that you will request for the virtual smart card.

To create the certificate template

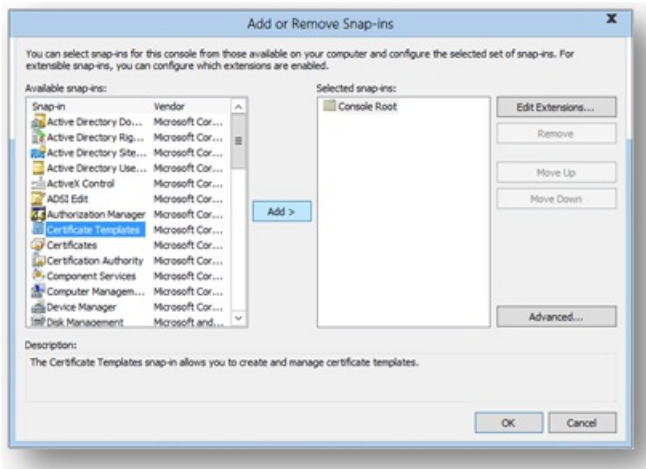
1. On your server, open the Microsoft Management Console (MMC). One way to do this is to type **mmc.exe**

from the **Start** menu, right-click **mmc.exe**, and click **Run as administrator**.

2. Click **File**, and then click **Add/Remove Snap-in**.

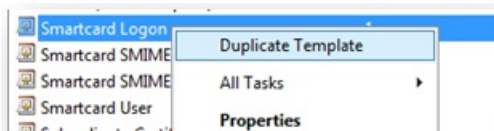


3. In the available snap-ins list, click **Certificate Templates**, and then click **Add**.

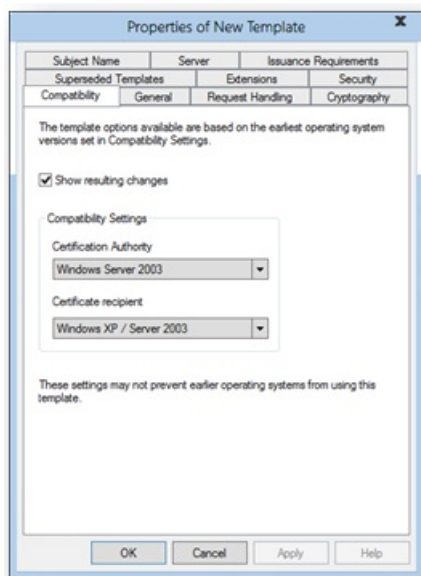


4. Certificate Templates is now located under **Console Root** in the MMC. Double-click it to view all the available certificate templates.

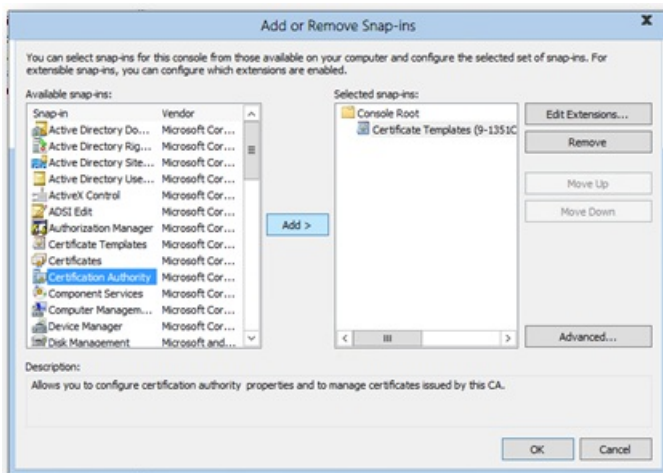
5. Right-click the **Smartcard Logon** template, and click **Duplicate Template**.



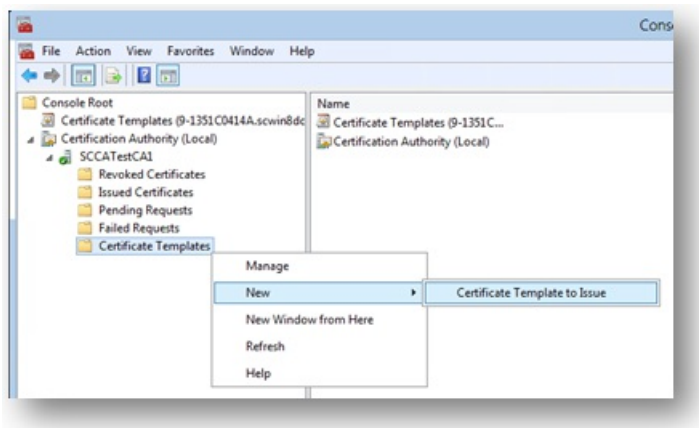
6. On the **Compatibility** tab, under **Certification Authority**, review the selection, and change it if needed.



7. On the **General** tab:
 - a. Specify a name, such as **TPM Virtual Smart Card Logon**.
 - b. Set the validity period to the desired value.
8. On the **Request Handling** tab:
 - a. Set the **Purpose** to **Signature and smartcard logon**.
 - b. Click **Prompt the user during enrollment**.
9. On the **Cryptography** tab:
 - a. Set the minimum key size to 2048.
 - b. Click **Requests must use one of the following providers**, and then select **Microsoft Base Smart Card Crypto Provider**.
10. On the **Security** tab, add the security group that you want to give **Enroll** access to. For example, if you want to give access to all users, select the **Authenticated users** group, and then select **Enroll** permissions for them.
11. Click **OK** to finalize your changes and create the new template. Your new template should now appear in the list of Certificate Templates.
12. Select **File**, then click **Add/Remove Snap-in** to add the Certification Authority snap-in to your MMC console. When asked which computer you want to manage, select the computer on which the CA is located, probably **Local Computer**.

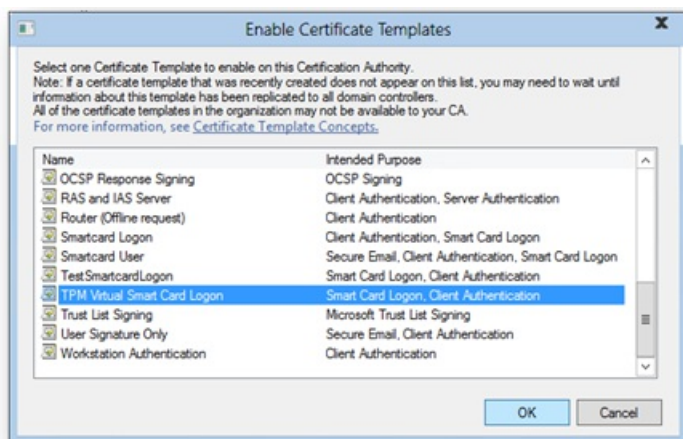


13. In the left pane of the MMC, expand **Certification Authority (Local)**, and then expand your CA within the Certification Authority list.
14. Right-click **Certificate Templates**, click **New**, and then click **Certificate Template to Issue**.

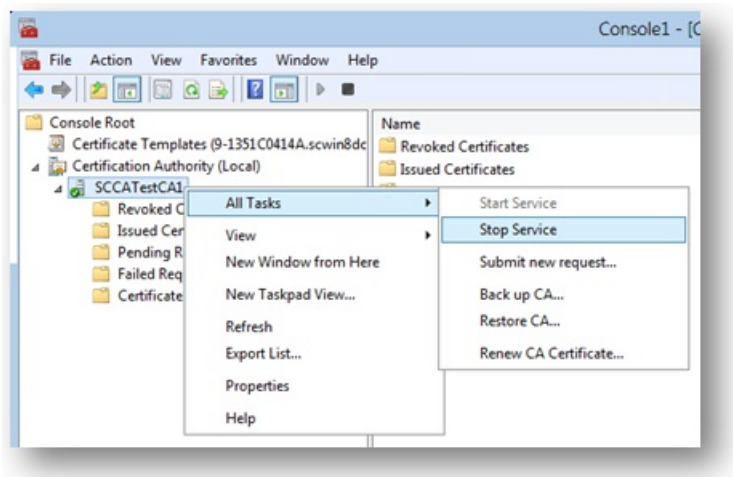


15. From the list, select the new template that you just created (**TPM Virtual Smart Card Logon**), and then click **OK**.

Note It can take some time for your template to replicate to all servers and become available in this list.



16. After the template replicates, in the MMC, right-click in the Certification Authority list, click **All Tasks**, and then click **Stop Service**. Then, right-click the name of the CA again, click **All Tasks**, and then click **Start Service**.

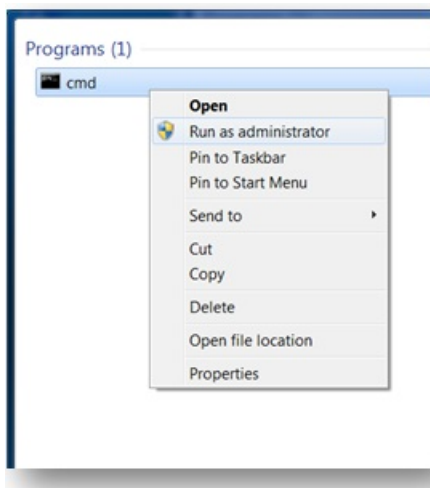


Step 2: Create the TPM virtual smart card

In this step, you will create the virtual smart card on the client computer by using the command-line tool, [Tpmvscmgr.exe](#).

To create the TPM virtual smart card

1. On a domain-joined computer, open a Command Prompt window with Administrative credentials.



2. At the command prompt, type the following, and then press ENTER:

```
tpmvscmgr.exe create /name TestVSC /pin default /adminkey random /generate
```

This will create a virtual smart card with the name **TestVSC**, omit the unlock key, and generate the file system on the card. The PIN will be set to the default, 12345678. To be prompted for a PIN, instead of `/pin default` you can type `/pin prompt`.

For more information about the `Tpmvscmgr` command-line tool, see [Use Virtual Smart Cards](#) and [Tpmvscmgr](#).

3. Wait several seconds for the process to finish. Upon completion, `Tpmvscmgr.exe` will provide you with the device instance ID for the TPM Virtual Smart Card. Store this ID for later reference because you will need it to manage or remove the virtual smart card.

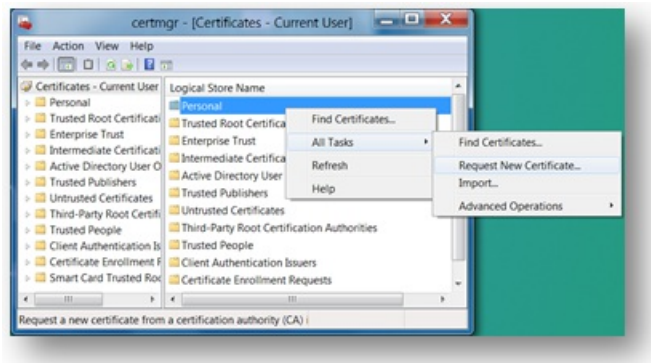
Step 3: Enroll for the certificate on the TPM Virtual Smart Card

The virtual smart card must be provisioned with a sign-in certificate for it to be fully functional.

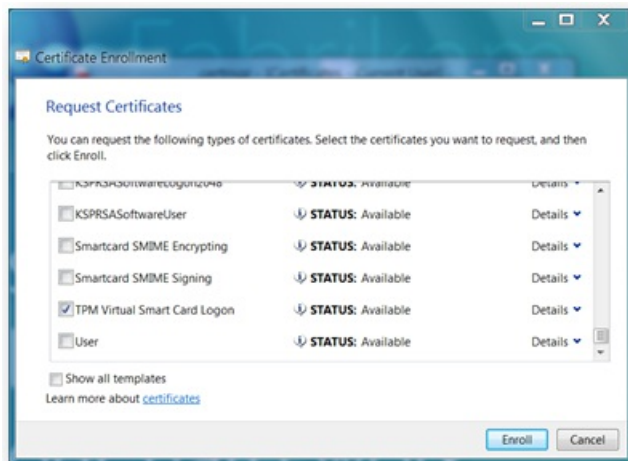
To enroll the certificate

1. Open the Certificates console by typing `certsmgr.msc` on the Start menu.

2. Right-click **Personal**, click **All Tasks**, and then click **Request New Certificate**.



3. Follow the prompts and when offered a list of templates, select the **TPM Virtual Smart Card Logon** check box (or whatever you named the template in Step 1).



4. If prompted for a device, select the Microsoft virtual smart card that corresponds to the one you created in the previous section. It displays as **Identity Device (Microsoft Profile)**.
5. Enter the PIN that was established when you created the TPM virtual smart card, and then click **OK**.
6. Wait for the enrollment to finish, and then click **Finish**.

The virtual smart card can now be used as an alternative credential to sign in to your domain. To verify that your virtual smart card configuration and certificate enrollment were successful, sign out of your current session, and then sign in. When you sign in, you will see the icon for the new TPM virtual smart card on the Secure Desktop (sign in) screen or you will be automatically directed to the TPM smart card sign-in dialog box. Click the icon, enter your PIN (if necessary), and then click **OK**. You should be signed in to your domain account.

See also

- [Understanding and Evaluating Virtual Smart Cards](#)
- [Use Virtual Smart Cards](#)
- [Deploy Virtual Smart Cards](#)

Use Virtual Smart Cards

7/1/2022 • 5 minutes to read • [Edit Online](#)

Applies To: Windows 10, Windows Server 2016

This topic for the IT professional describes requirements for virtual smart cards, how to use virtual smart cards, and tools that are available to help you create and manage them.

Requirements, restrictions, and limitations

AREA	REQUIREMENTS AND DETAILS
Supported operating systems	Windows Server 2016 Windows Server 2012 R2 Windows Server 2012 Windows 10 Windows 8.1 Windows 8
Supported Trusted Platform Module (TPM)	Any TPM that adheres to the TPM main specifications for version 1.2 or version 2.0 (as set by the Trusted Computing Group) is supported for use as a virtual smart card. For more information, see the TPM Main Specification .
Supported virtual smart cards per computer	Ten smart cards can be connected to a computer or device at one time. This includes physical and virtual smart cards combined. Note You can create more than one virtual smart card; however, after creating more than four virtual smart cards, you may start to notice performance degradation. Because all smart cards appear as if they are always inserted, if more than one person shares a computer or device, each person can see all the virtual smart cards that are created on that computer or device. If the user knows the PIN values for all the virtual smart cards, the user will also be able to use them.
Supported number of certificates on a virtual smart card	A single TPM virtual smart card can contain 30 distinct certificates with the corresponding private keys. Users can continue to renew certificates on the card until the total number of certificates on a card exceeds 90. The reason that the total number of certificates is different from the total number of private keys is that sometimes the renewal can be done with the same private key—in which case a new private key is not generated.
PIN, PIN Unlock Key (PUK), and Administrative key requirements	The PIN and the PUK must be a minimum of eight characters that can include numerals, alphabetic characters, and special characters. The Administrative key must be entered as 48 hexadecimal characters. It is a 3-key triple DES with ISO/IEC 9797 padding method 2 in CBC chaining mode.

Using Tpmvscmgr.exe

To create and delete TPM virtual smart cards for end users, the Tpmvscmgr command-line tool is included as a command-line tool with the operating system. You can use the **Create** and **Delete** parameters to manage virtual smart cards on local or remote computers. For information about using this tool, see [Tpmvscmgr](#).

Create and delete virtual smart cards programmatically

Virtual smart cards can also be created and deleted by using APIs. For more information, see the following classes and interfaces:

- [TpmVirtualSmartCardManager](#)
- [RemoteTpmVirtualSmartCardManager](#)
- [ITpmVirtualSmartCardManager](#)
- [ITPMVirtualSmartCardManagerStatusCallBack](#)

You can use APIs that were introduced in the Windows.Device.SmartCards namespace in Windows Server 2012 R2 and Windows 8.1 to build Microsoft Store apps to manage the full lifecycle of virtual smart cards. For information about how to build an app to do this, see [Strong Authentication: Building Apps That Leverage Virtual Smart Cards in Enterprise, BYOD, and Consumer Environments | Build 2013 | Channel 9](#).

The following table describes the features that can be developed in a Microsoft Store app:

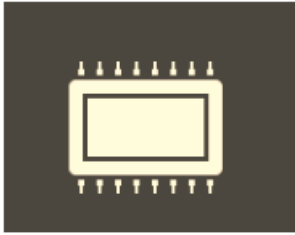
FEATURE	PHYSICAL SMART CARD	VIRTUAL SMART CARD
Query and monitor smart card readers	Yes	Yes
List available smart cards in a reader, and retrieve the card name and card ID	Yes	Yes
Verify if the administrative key of a card is correct	Yes	Yes
Provision (or reformat) a card with a given card ID	Yes	Yes
Change the PIN by entering the old PIN and specifying a new PIN	Yes	Yes
Change the administrative key, reset the PIN, or unblock the smart card by using a challenge/response method	Yes	Yes
Create a virtual smart card	Not applicable	Yes
Delete a virtual smart card	Not applicable	Yes
Set PIN policies	No	Yes

For more information about these Windows APIs, see:

- [Windows.Devices.SmartCards namespace \(Windows\)](#)
- [Windows.Security.Cryptography.Certificates namespace \(Windows\)](#)

Distinguishing TPM-based virtual smart cards from physical smart cards

To help users visually distinguish a Trusted Platform Module (TPM)-based virtual smart card from physical smart cards, the virtual smart card has a different icon. The following icon is displayed during sign in, and on other screens that require the user to enter the PIN for a virtual smart card.



A TPM-based virtual smart card is labeled **Security Device** in the user interface.

Changing the PIN

The PIN for a virtual smart card can be changed by following these steps:

- Sign in with the old PIN or password.
- Press Ctrl+Alt+Del and choose **Change a password**.
- Select **Sign-in Options**.
- Select the virtual smart card icon.
- Enter and confirm the new PIN.

Resolving issues

TPM not provisioned

For a TPM-based virtual smart card to function properly, a provisioned TPM must be available on the computer. If the TPM is disabled in the BIOS, or it is not provisioned with full ownership and the storage root key, the TPM virtual smart card creation will fail.

If the TPM is initialized after creating a virtual smart card, the card will no longer function, and it will need to be re-created.

If the TPM ownership was established on a Windows Vista installation, the TPM will not be ready to use virtual smart cards. The system administrator needs to clear and initialize the TPM for it to be suitable for creating TPM virtual smart cards.

If the operating system is reinstalled, prior TPM virtual smart cards are no longer available and need to be re-created. If the operating system is upgraded, prior TPM virtual smart cards will be available to use in the upgraded operating system.

TPM in lockout state

Sometimes, due to frequent incorrect PIN attempts from a user, the TPM may enter the lockout state. To resume using the TPM virtual smart card, it is necessary to reset the lockout on the TPM by using the owner's password or to wait for the lockout to expire. Unblocking the user PIN does not reset the lockout in the TPM. When the TPM is in lockout, the TPM virtual smart card appears as if it is blocked. When the TPM enters the lockout state because the user entered an incorrect PIN too many times, it may be necessary to reset the user PIN by using the virtual smart card management tools, such as Tpmvscmgr command-line tool.

See also

For information about authentication, confidentiality, and data integrity use cases, see [Virtual Smart Card Overview](#).

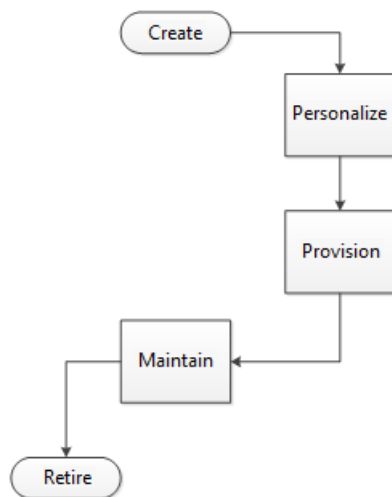
Deploy Virtual Smart Cards

7/1/2022 • 24 minutes to read • [Edit Online](#)

Applies To: Windows 10, Windows Server 2016

This topic for the IT professional discusses the factors to consider when you deploy a virtual smart card authentication solution.

Traditional identity devices, such as physical smart cards, follow a predictable lifecycle in any deployment, as shown in the following diagram.



Physical devices are created by a dedicated manufacturer and then purchased by the corporation that will ultimately deploy it. The device passes through the personalization stage, where its unique properties are set. In smart cards, these properties are the administrator key, Personal Identification Number (PIN), PIN Unlock Key (PUK), and its physical appearance. To provision the device, it is loaded with the required certificates, such as a sign-in certificate. After you provision the device, it is ready for use. The device must simply be maintained. For example, you must replace cards when they are lost or stolen and reset PINs when users forget them. Finally, you'll retire devices when they exceed their intended lifetime or when employees leave the company.

This topic contains information about the following phases in a virtual smart card lifecycle:

- [Create and personalize virtual smart cards](#)
- [Provision virtual smart cards](#)
- [Maintain virtual smart cards](#)

Create and personalize virtual smart cards

A corporation purchases the devices to deploy then. The device passes through the personalization stage, where its unique properties are set. In smart cards, these properties are the administrator key, Personal Identification Number (PIN), PIN Unlock Key (PUK), and its physical appearance. The security that is provided for a TPM virtual smart card is fully provisioned in the host TPM.

Trusted Platform Module readiness

The TPM Provisioning Wizard, which is launched from the **TPM Management Console**, takes the user through all the steps to prepare the TPM for use.

When you create virtual smart cards, consider the following actions in the TPM:

- **Enable and Activate:** TPMs are built in to many industry ready computers, but they often are not enabled and activated by default. In some cases, the TPM must be enabled and activated through the BIOS. For more information, see [Initialize and Configure Ownership of the TPM](#).
- **Take ownership:** When you provision the TPM, you set an owner password for managing the TPM in the future, and you establish the storage root key. To provide anti-hammering protection for virtual smart cards, the user or a domain administrator must be able to reset the TPM owner password. For corporate use of TPM virtual smart cards, we recommend that the corporate domain administrator restrict access to the TPM owner password by storing it in Active Directory, not in the local registry. When TPM ownership is set in Windows Vista, the TPM needs to be cleared and reinitialized. For more information, see [Trusted Platform Module Technology Overview](#).
- **Manage:** You can manage ownership of a virtual smart card by changing the owner password, and you can manage anti-hammering logic by resetting the lockout time. For more information, see [Manage TPM Lockout](#).

A TPM might operate in reduced functionality mode. This could occur, for example, if the operating system cannot determine if the owner password is available to the user. In those cases, the TPM can be used to create a virtual smart card, but it is strongly recommended to bring the TPM to a fully ready state so that any unexpected circumstances will not leave the user blocked from using the computer.

Those smart card deployment management tools that require a status check of a TPM before attempting to create a TPM virtual smart card can do so using the TPM WMI interface.

Depending on the setup of the computer that is designated for installing TPM virtual smart cards, it might be necessary to provision the TPM before continuing with the virtual smart card deployment. For more information about provisioning, see [Use Virtual Smart Cards](#).

For more information about managing TPMs by using built-in tools, see [Trusted Platform Module Services Group Policy Settings](#).

Creation

A TPM virtual smart card simulates a physical smart card, and it uses the TPM to provide the same functionality as physical smart card hardware. A virtual smart card appears within the operating system as a physical smart card that is always inserted. Supported versions of the Windows operating system present a virtual smart card reader and virtual smart card to applications with the same interface as physical smart cards, but messages to and from the virtual smart card are translated to TPM commands. This process ensures the integrity of the virtual smart card through the three properties of smart card security:

- **Non-exportability:** Because all private information on the virtual smart card is encrypted by using the TPM on the host computer, it cannot be used on a different computer with a different TPM. Additionally, TPMs are designed to be tamper-resistant and non-exportable, so a malicious user cannot reverse engineer an identical TPM or install the same TPM on a different computer. For more information, see [Evaluate Virtual Smart Card Security](#).
- **Isolated cryptography:** TPMs provide the same properties of isolated cryptography that is offered by physical smart cards, and this is utilized by virtual smart cards. Unencrypted copies of private keys are loaded only within the TPM and never into memory that is accessible by the operating system. All cryptographic operations with these private keys occur inside the TPM.
- **Anti-hammering:** If a user enters a PIN incorrectly, the virtual smart card responds by using the anti-hammering logic of the TPM, which rejects further attempts for a period of time instead of blocking the card. This is also known as lockout. For more information, see [Blocked virtual smart card](#) and [Evaluate Virtual Smart Card Security](#).

There are several options for creating virtual smart cards, depending on the size of the deployment and budget

of the organization. The lowest cost option is using `Tpmvscmgr.exe` to create cards individually on users' computers. Alternatively, a virtual smart card management solution can be purchased to more easily accomplish virtual smart card creation on a larger scale and aid in further phases of deployment. Virtual smart cards can be created on computers that are to be provisioned for an employee or on those that are already in an employee's possession. In either approach, there should be some central control over personalization and provisioning. If a computer is intended for use by multiple employees, multiple virtual smart cards can be created on a computer.

For information about the TPM Virtual Smart Card command-line tool, see [Tpmvscmgr](#).

Personalization

During virtual smart card personalization, the values for the administrator key, PIN, and PUK are assigned. As with a physical card, knowing the administrator key is important for resetting the PIN or for deleting the card in the future. (If a PUK is set, the administrator key can no longer be used to reset the PIN.)

Because the administrator key is critical to the security of the card, it is important to consider the deployment environment and decide on the proper administrator key setting strategy. Options for these strategies include:

- **Uniform:** Administrator keys for all the virtual smart cards that are deployed in the organization are the same. Although this makes the maintenance infrastructure easy (only one key needs to be stored), it is highly insecure. This strategy might be sufficient for very small organizations, but if the administrator key is compromised, all virtual smart cards that use this key must be reissued.
- **Random, not stored:** Administrator keys are assigned randomly for all virtual smart cards, and they are not recorded. This is a valid option if the deployment administrators do not require the ability to reset PINs, and instead prefer to delete and reissue virtual smart cards. This could also be a viable strategy if the administrator prefers to set PUK values for the virtual smart cards and then use this value to reset PINs, if necessary.
- **Random, stored:** Administrator keys are assigned randomly and stored in a central location. Each card's security is independent of the others. This is secure on a large scale unless the administrator key database is compromised.
- **Deterministic:** Administrator keys are the result of some function or known information. For example, the user ID could be used to randomly generate data that can be further processed through a symmetric encryption algorithm by using a secret. This administrator key can be similarly regenerated when needed, and it does not need to be stored. The security of this method relies on the security of the secret used.

Although the PUK and the administrator key methodologies provide unlocking and resetting functionality, they do so in different ways. The PUK is a PIN that is simply entered on the computer to enable a user PIN reset.

The administrator key methodology takes a challenge-response approach. The card provides a set of random data after users verify their identity to the deployment administrator. The administrator then encrypts the data with the administrator key and gives the encrypted data back to the user. If the encrypted data matches that produced by the card during verification, the card will allow PIN reset. Because the administrator key is never accessible by anyone other than the deployment administrator, it cannot be intercepted or recorded by any other party (including employees). This provides significant security benefits beyond using a PUK, an important consideration during the personalization process.

TPM virtual smart cards can be personalized on an individual basis when they are created with the `Tpmvscmgr` command-line tool. Or organizations can purchase a management solution that can incorporate personalization into an automated routine. An additional advantage of such a solution is the automated creation of administrator keys. `Tpmvscmgr.exe` allows users to create their own administrator keys, which can be detrimental to the security of the virtual smart cards.

Provision virtual smart cards

Provisioning is the process of loading specific credentials onto a TPM virtual smart card. These credentials consist of certificates that are created to give users access to a specific service, such as domain sign in. A maximum of 30 certificates is allowed on each virtual smart card. As with physical smart cards, several decisions must be made regarding the provisioning strategy, based on the environment of the deployment and the desired level of security.

A high-assurance level of secure provisioning requires absolute certainty about the identity of the individual who is receiving the certificate. Therefore, one method of high-assurance provisioning is utilizing previously provisioned strong credentials, such as a physical smart card, to validate identity during provisioning. In-person proofing at enrollment stations is another option, because an individual can easily and securely prove his or her identity with a passport or driver's license, although this can become infeasible on a larger scale. To achieve a similar level of assurance, a large organization can implement an "enroll-on-behalf-of" strategy, in which employees are enrolled with their credentials by a superior who can personally verify their identities. This creates a chain of trust that ensures individuals are checked in person against their proposed identities, but without the administrative strain of provisioning all virtual smart cards from a single central enrollment station.

For deployments in which a high-assurance level is not a primary concern, you can use self-service solutions. These can include using an online portal to obtain credentials or simply enrolling for certificates by using Certificate Manager, depending on the deployment. Consider that virtual smart card authentication is only as strong as the method of provisioning. For example, if weak domain credentials (such as a password alone) are used to request the authentication certificate, virtual smart card authentication will be equivalent to using only the password, and the benefits of two-factor authentication are lost.

For information about using Certificate Manager to configure virtual smart cards, see [Get Started with Virtual Smart Cards: Walkthrough Guide](#).

High-assurance and self-service solutions approach virtual smart card provisioning by assuming that the user's computer has been issued prior to the virtual smart card deployment, but this is not always the case. If virtual smart cards are being deployed with new computers, they can be created, personalized, and provisioned on the computer before the user has contact with that computer.

In this situation, provisioning becomes relatively simple, but identity checks must be put in place to ensure that the recipient of the computer is the individual who was expected during provisioning. This can be accomplished by requiring the employee to set the initial PIN under the supervision of the deployment administrator or manager.

When you are provisioning your computers, you should also consider the longevity of credentials that are supplied for virtual smart cards. This choice must be based on the risk threshold of the organization. Although longer lived credentials are more convenient, they are also more likely to become compromised during their lifetime. To decide on the appropriate lifetime for credentials, the deployment strategy must take into account the vulnerability of their cryptography (how long it could take to crack the credentials), and the likelihood of attack.

If a virtual smart card is compromised, administrators should be able to revoke the associated credentials, like they would with a lost or stolen laptop. This requires a record of which credentials match which user and computer, which is functionality that does not exist natively in Windows. Deployment administrators might want to consider add-on solutions to maintain such a record.

Virtual smart cards on consumer devices used for corporate access

There are techniques that allow employees to provision virtual smart cards and enroll for certificates that can be used to authenticate the users. This is useful when employees attempt to access corporate resources from devices that are not joined to the corporate domain. Those devices can be further defined to not allow users to download and run applications from sources other than the Windows Store (for example, devices running Windows RT).

You can use APIs that were introduced in Windows Server 2012 R2 and Windows 8.1 to build Windows Store

apps that you can use to manage the full lifecycle of virtual smart cards. For more information, see [Create and delete virtual smart cards programmatically](#).

TPM ownerAuth in the registry

When a device or computer is not joined to a domain, the TPM ownerAuth is stored in the registry under HKEY_LOCAL_MACHINE. This exposes some threats. Most of the threat vectors are protected by BitLocker, but threats that are not protected include:

- A malicious user possesses a device that has an active local sign-in session before the device locks. The malicious user could attempt a brute-force attack on the virtual smart card PIN, and then access the corporate secrets.
- A malicious user possesses a device that has an active virtual private network (VPN) session. The device is then compromised.

The proposed mitigation for the previous scenarios is to use Exchange ActiveSync (EAS) policies to reduce the automatic lockout time from five minutes to 30 seconds of inactivity. Policies for automatic lockout can be set while provisioning virtual smart cards. If an organization wants more security, they can also configure a setting to remove the ownerAuth from the local device.

For configuration information about the TPM ownerAuth registry key, see the Group Policy setting [Configure the level of TPM owner authorization information available to the operating system](#).

For information about EAS policies, see [Exchange ActiveSync Policy Engine Overview](#).

Managed and unmanaged cards

The following table describes the important differences between managed and unmanaged virtual smart cards that exist on consumer devices:

OPERATION	MANAGED AND UNMANAGED CARDS	UNMANAGED CARDS
Reset PIN when the user forgets the PIN	Yes	No, the card has to be deleted and created again.
Allow user to change the PIN	Yes	No, the card has to be deleted and created again.

Managed cards

A managed virtual smart card can be serviced by the IT administrator or another person in that designated role. It allows the IT administrator to have influence or complete control over specific aspects of the virtual smart card from its creation to deletion. To manage these cards, a virtual smart card deployment management tool is often required.

Managed card creation

A user can create blank virtual smart card by using the `Tpmvscmgr` command-line tool, which is a built-in tool that is run with administrative credentials through an elevated command prompt. This virtual smart card needs to be created with well-known parameters (such as default values), and it should be left unformatted (specifically, the `/generate` option should not be specified).

The following command creates a virtual smart card that can later be managed by a smart card management tool launched from another computer (as explained in the next section):

```
tpmvscmgr.exe create /name "VirtualSmartCardForCorpAccess" /AdminKey DEFAULT /PIN PROMPT
```

Alternatively, instead of using a default administrator key, a user can enter an administrator key at the command line:

```
tpmvscmgr.exe create /name "VirtualSmartCardForCorpAccess" /AdminKey PROMPT /PIN PROMPT
```

In either case, the card management system needs to be aware of the initial administrator key that is used so that it can take ownership of the virtual smart card and change the administrator key to a value that is only accessible through the card management tool operated by the IT administrator. For example, when the default value is used, the administrator key is set to:

```
10203040506070801020304050607080102030405060708
```

For information about using this command-line tool, see [Tpmvscmgr](#).

Managed card management

After the virtual smart card is created, the user needs to open a remote desktop connection to an enrollment station, for example, in a computer that is joined to the domain. Virtual smart cards that are associated with a client computer are available for use in the remote desktop connection. The user can open a card management tool inside the remote session that can take ownership of the card and provision it for use by the user. This requires that a user is allowed to establish a remote desktop connection from a non-domain-joined computer to a domain-joined computer. This might require a specific network configuration, such as through IPsec policies.

When users need to reset or change a PIN, they need to use the remote desktop connection to complete these operations. They can use the built-in tools for PIN unlock and PIN change or the smart card management tool.

Certificate management for managed cards

Similar to physical smart cards, virtual smart cards require certificate enrollment.

Certificate issuance

Users can enroll for certificates from within a remote desktop session that is established to provision the card. This process can also be managed by the smart card management tool that the user runs through the remote desktop connection. This model works for deployments that require the user to sign a request for enrollment by using a physical smart card. The driver for the physical smart card does not need to be installed on the client computer if it is installed on the remote computer. This is made possible by smart card redirection functionality that was introduced in Windows Server 2003, which ensures that smart cards that are connected to the client computer are available for use during a remote session.

Alternatively, without establishing a remote desktop connection, users can enroll for certificates from the Certificate Management console (certmgr.msc) on a client computer. Users can also create a request and submit it to a server from within a custom certificate enrollment application (for example, a registration authority) that has controlled access to the certification authority (CA). This requires specific enterprise configuration and deployments for Certificate Enrollment Policies (CEP) and Certificate Enrollment Services (CES).

Certificate lifecycle management

You can renew certificates through remote desktop connections, certificate enrollment policies, or certificate enrollment services. Renewal requirements could be different from the initial issuance requirements, based on the renewal policy.

Certificate revocation requires careful planning. When information about the certificate to be revoked is reliably available, the specific certificate can be easily revoked. When information about the certificate to be revoked is not easy to determine, all certificates that are issued to the user under the policy that was used to issue the certificate might need to be revoked. For example, this could occur if an employee reports a lost or compromised device, and information that associates the device with a certificate is not available.

Unmanaged cards

Unmanaged virtual smart cards are not serviceable by an IT administrator. Unmanaged cards might be suitable if an organization does not have an elaborate smart card deployment management tool and using remote desktop connections to manage the card is not desirable. Because unmanaged cards are not serviceable by the

IT administrator, when a user needs help with a virtual smart card (for example, resetting or unlocking a PIN), the only option available to the user is to delete the card and create it again. This results in loss of the user's credentials and he or she must re-enroll.

Unmanaged card creation

A user can create a virtual smart card by using the `Tpmvscmgr` command-line tool, which is run with administrative credentials through an elevated command prompt. The following command creates an unmanaged card that can be used to enroll certificates:

```
tpmvscmgr.exe create /name "VirtualSmartCardForCorpAccess" /AdminKey RANDOM /PIN PROMPT /generate
```

This command creates a card with a randomized administrator key. The key is automatically discarded after the creation of the card. If users forget or want to change their PIN, they need to delete the card and create it again. To delete the card, a user can run the following command:

```
tpmvscmgr.exe destroy /instance <instance ID>
```

where <instance ID> is the value that is printed on the screen when the user creates the card. Specifically, for the first card created, the instance ID is `ROOT\SMARTCARDREADER\0000`.

Certificate management for unmanaged cards

Depending on the security requirements that are unique to an organization, users can initially enroll for certificates from the certificate management console (`certmgr.msc`) or from within custom certificate enrollment applications. The latter method can create a request and submit it to a server that has access to the Certification Authority. This requires specific organizational configurations and deployments for certificate enrollment policies and certificate enrollment services. Windows has built-in tools, specifically `Certreq.exe` and `Certutil.exe`, which can be used by scripts to perform the enrollment from the command line.

Requesting the certificate by providing domain credentials only

The simplest way for users to request certificates is to provide their domain credentials through a script that can perform the enrollment through built-in components you have in place for certificate requests.

Alternatively, an application (such as a line-of-business app) can be installed on the computer to perform enrollment by generating a request on the client. The request is submitted to an HTTP server, which can forward it to a registration authority.

Another option is to have the user access an enrollment portal that is available through Internet Explorer. The webpage can use the scripting APIs to perform certificate enrollment.

Signing the request with another certificate

You can provide users with a short-term certificate through a Personal Information Exchange (`.pfx`) file. You can generate the `.pfx` file by initiating a request from a domain-joined computer. Additional policy constraints can be enforced on the `.pfx` file to assert the identity of the user.

The user can import the certificate into the **MY** store (which is the user's certificate store). And your organization can present the user with a script that can be used to sign the request for the short-term certificate and to request a virtual smart card.

For deployments that require users to use a physical smart card to sign the certificate request, you can use the procedure:

1. Users initiate a request on a domain-joined computer.
2. Users complete the request by using a physical smart card to sign the request.
3. Users download the request to the virtual smart card on their client computer.

Using one-time password for enrollment

Another option to ensure that users are strongly authenticated before virtual smart card certificates are issued

is to send a user a one-time password through SMS, email, or phone. The user then types the one-time password during the certificate enrollment from an application or a script on a desktop that invokes built-in command-line tools.

Certificate lifecycle management

Certificate renewal can be done from the same tools that are used for the initial certificate enrollment. Certificate enrollment policies and certificate enrollment services can also be used to perform automatic renewal.

Certificate revocation requires careful planning. When information about the certificate to be revoked is reliably available, the specific certificate can be easily revoked. When information about the certificate to be revoked is not easy to determine, all certificates that are issued to the user under the policy that was used to issue the certificate might need to be revoked. For example, this could occur if an employee reports a lost or compromised device, and information that associates the device with a certificate is not available.

Maintain virtual smart cards

Maintenance is a significant portion of the virtual smart card lifecycle and one of the most important considerations from a management perspective. After virtual smart cards are created, personalized, and provisioned, they can be used for convenient two-factor authentication. Deployment administrators must be aware of several common administrative scenarios, which can be approached by using a purchased virtual smart card solution or on a case-by-case basis with in-house methods.

Renewal: Renewing virtual smart card credentials is a regular task that is necessary to preserve the security of a virtual smart card deployment. Renewal is the result of a signed request from a user who specifies the key pair desired for the new credentials. Depending on user's choice or deployment specification, the user can request credentials with the same key pair as previously used, or choose a newly generated key pair.

When renewing with a previously used key, no extra steps are required because a strong certificate with this key was issued during the initial provisioning. However, when the user requests a new key pair, you must take the same steps that were used during provisioning to assure the strength of the credentials. Renewal with new keys should occur periodically to counter sophisticated long-term attempts by malicious users to infiltrate the system. When new keys are assigned, you must ensure that the new keys are being used by the expected individuals on the same virtual smart cards.

Resetting PINs: Resetting virtual smart card PINs is also a frequent necessity, because employees forget their PINs. There are two ways to accomplish this, depending on choices made earlier in the deployment: Use a PUK (if the PUK is set), or use a challenge-response approach with the administration key. Before resetting the PIN, the user's identity must be verified by using some means other than the card—most likely the verification method that you used during initial provisioning (for example, in-person proofing). This is necessary in user-error scenarios when users forget their PINs. However, you should never reset a PIN if it has been compromised because the level of vulnerability after the PIN is exposed is difficult to identify. The entire card should be reissued.

Lockout reset: A frequent precursor to resetting a PIN is the necessity of resetting the TPM lockout time because the TPM anti-hammering logic will be engaged with multiple PIN entry failures for a virtual smart card. This is currently device specific.

Retiring cards: The final aspect of virtual smart card management is retiring cards when they are no longer needed. When an employee leaves the company, it is desirable to revoke domain access. Revoking sign-in credentials from the certification authority (CA) accomplishes this goal.

The card should be reissued if the same computer is used by other employees without reinstalling the operating system. Reusing the former card can allow the former employee to change the PIN after leaving the organization, and then hijack certificates that belong to the new user to obtain unauthorized domain access. However, if the employee takes the virtual smart card-enabled computer, it is only necessary to revoke the

certificates that are stored on the virtual smart card.

Emergency preparedness

Card reissuance

The most common scenario in an organization is reissuing virtual smart cards, which can be necessary if the operating system is reinstalled or if the virtual smart card is compromised in some manner. Reissuance is essentially the recreation of the card, which involves establishing a new PIN and administrator key and provisioning a new set of associated certificates. This is an immediate necessity when a card is compromised, for example, if the virtual smart card-protected computer is exposed to an adversary who might have access to the correct PIN. Reissuance is the most secure response to an unknown exposure of a card's privacy. Additionally, reissuance is necessary after an operating system is reinstalled because the virtual smart card device profile is removed with all other user data when the operating system is reinstalled.

Blocked virtual smart card

The anti-hammering behavior of a TPM virtual smart card is different from that of a physical smart card. A physical smart card blocks itself after the user enters the wrong PIN a few times. A TPM virtual smart card enters a timed delay after the user enters the wrong PIN a few times. If the TPM is in the timed-delay mode, when the user attempts to use the TPM virtual smart card, the user is notified that the card is blocked. Furthermore, if you enable the integrated unlock functionality, the user can see the user interface to unlock the virtual smart card and change the PIN. Unlocking the virtual smart card does not reset the TPM lockout. The user needs to perform an extra step to reset the TPM lockout or wait for the timed delay to expire.

For more information about setting the Allow Integrated Unblock policy, see [Allow Integrated Unblock screen to be displayed at the time of logon](#).

See also

[Understanding and Evaluating Virtual Smart Cards](#)

[Get Started with Virtual Smart Cards: Walkthrough Guide](#)

[Use Virtual Smart Cards](#)

[Evaluate Virtual Smart Card Security](#)

[Tpmvscmgr](#)

Evaluate Virtual Smart Card Security

7/1/2022 • 3 minutes to read • [Edit Online](#)

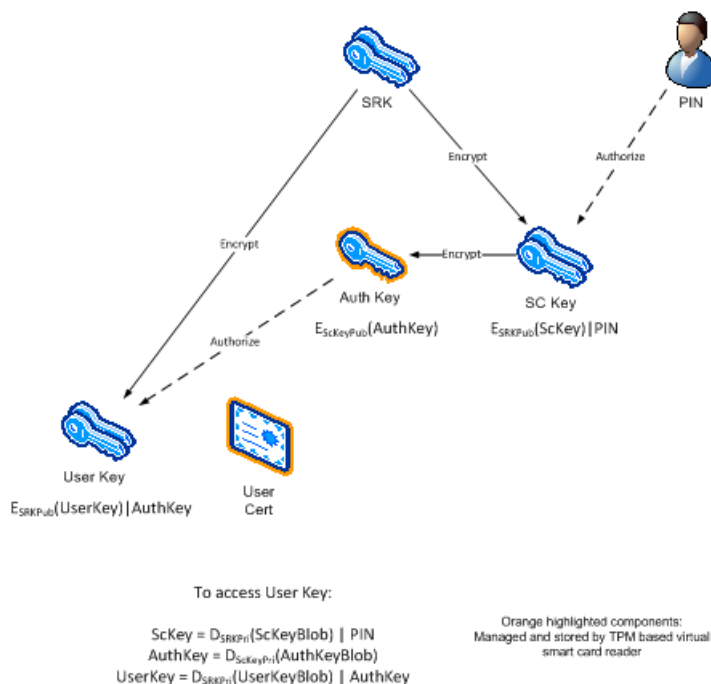
Applies To: Windows 10, Windows Server 2016

This topic for the IT professional describes security characteristics and considerations when deploying TPM virtual smart cards.

Virtual smart card non-exportability details

A crucial aspect of TPM virtual smart cards is their ability to securely store and use secret data, specifically that the secured data is non-exportable. Data can be accessed and used within the virtual smart card system, but it is meaningless outside of its intended environment. In TPM virtual smart cards, security is ensured with a secure key hierarchy, which includes several chains of encryption. This originates with the TPM storage root key, which is generated and stored within the TPM and never exposed outside the chip. The TPM key hierarchy is designed to allow encryption of user data with the storage root key, but it authorizes decryption with the user PIN in such a way that changing the PIN doesn't require re-encryption of the data.

The following diagram illustrates the secure key hierarchy and the process of accessing the user key.



The following keys are stored on the hard disk:

- User key
- Smart card key, which is encrypted by the storage root key
- Authorization key for the user key decryption, which is encrypted by the public portion of the smart card key

When the user enters a PIN, the use of the decrypted smart card key is authorized with this PIN. If this authorization succeeds, the decrypted smart card key is used to decrypt the auth key. The auth key is then provided to the TPM to authorize the decryption and use of the user's key that is stored on the virtual smart card.

The auth key is the only sensitive data that is used as plaintext outside the TPM, but its presence in memory is protected by Microsoft Data Protection API (DPAPI), such that before being stored in any way, it is encrypted. All data other than the auth key is processed only as plaintext within the TPM, which is completely isolated from external access.

Virtual smart card anti-hammering details

The anti-hammering functionality of virtual smart cards relies on the anti-hammering functionality of the TPM that is enabling the virtual smart card. However, the TPM version 1.2 and subsequent specifications (as designed by the Trusted Computing Group) provide very flexible guidelines for responding to hammering. The spec requires only that the TPM implement protection against trial-and-error attacks on the user PIN, PUK, and challenge/response mechanism.

The Trusted Computing Group also specifies that if the response to attacks involves suspending proper function of the TPM for some period of time or until administrative action is taken, the TPM must prevent running the authorized TPM commands. The TPM can prevent running any TPM commands until the termination of the attack response. Beyond using a time delay or requiring administrative action, a TPM could also force a reboot when an attack is detected. The Trusted Computing Group allows manufacturers a level of creativity in their choice of implementation. Whatever methodology is chosen by TPM manufacturers determines the anti-hammering response of TPM virtual smart cards. Some typical aspects of protection from attacks include:

1. Allow only a limited number of wrong PIN attempts before enabling a lockout that enforces a time delay before any further commands are accepted by the TPM.

Note Introduced in Windows Server 2012 R2 and Windows 8.1, if the user enters the wrong PIN five consecutive times for a virtual smart card (which works in conjunction with the TPM), the card is blocked. When the card is blocked, it has to be unblocked by using the administrative key or the PUK.

2. Increase the time delay exponentially as the user enters the wrong PIN so that an excessive number of wrong PIN attempts quickly trigger long delays in accepting commands.
3. Have a failure leakage mechanism to allow the TPM to reset the timed delays over a period of time. This is useful in cases where a valid user has entered the wrong PIN occasionally, for example, due to complexity of the PIN.

As an example, it will take 14 years to guess an 8-character PIN for a TPM that implements the following protection:

1. Number of wrong PINs allowed before entering lockout (threshold): 9
2. Time the TPM is in lockout after the threshold is reached: 10 seconds
3. Timed delay doubles for each wrong PIN after the threshold is reached

See also

[Understanding and Evaluating Virtual Smart Cards](#)

Tpmvscmgr

7/1/2022 • 5 minutes to read • [Edit Online](#)

Applies To: Windows 10, Windows Server 2016

The Tpmvscmgr command-line tool allows users with Administrative credentials to create and delete TPM virtual smart cards on a computer. For examples of how this command can be used, see [Examples](#).

Syntax

```
Tpmvscmgr create [/quiet] /name <name> /AdminKey {DEFAULT | PROMPT | RANDOM} [/PIN {DEFAULT | PROMPT}] [/PUK {DEFAULT | PROMPT}] [/generate] [/machine <machine name>] [/pinpolicy [policy options]] [/attestation {AIK_AND_CERT | AIK_ONLY}] [/?]
```

```
Tpmvscmgr destroy [/quiet] [/instance <device instance ID>] [/machine <machine name>] [/?]
```

Parameters for Create command

The Create command sets up new virtual smart cards on the user's system. It returns the instance ID of the newly created card for later reference if deletion is required. The instance ID is in the format ROOT\SMARTCARDREADER\000n where n starts from 0 and is increased by 1 each time you create a new virtual smart card.

PARAMETER	DESCRIPTION
/name	Required. Indicates the name of the new virtual smart card.
/AdminKey	Indicates the desired administrator key that can be used to reset the PIN of the card if the user forgets the PIN. DEFAULT Specifies the default value of 010203040506070801020304050607080102030405060708. PROMPT Prompts the user to enter a value for the administrator key. RANDOM Results in a random setting for the administrator key for a card that is not returned to the user. This creates a card that might not be manageable by using smart card management tools. When generated with RANDOM, the administrator key is set as 48 hexadecimal characters.
/PIN	Indicates desired user PIN value. DEFAULT Specifies the default PIN of 12345678. PROMPT Prompts the user to enter a PIN at the command line. The PIN must be a minimum of eight characters, and it can contain numerals, characters, and special characters.
/PUK	Indicates the desired PIN Unlock Key (PUK) value. The PUK value must be a minimum of eight characters, and it can contain numerals, characters, and special characters. If the parameter is omitted, the card is created without a PUK. DEFAULT Specifies the default PUK of 12345678. PROMPT Prompts the user to enter a PUK at the command line.

PARAMETER	DESCRIPTION
/generate	Generates the files in storage that are necessary for the virtual smart card to function. If the /generate parameter is omitted, it is equivalent to creating a card without this file system. A card without a file system can be managed only by a smart card management system such as Microsoft Endpoint Configuration Manager.
/machine	Allows you to specify the name of a remote computer on which the virtual smart card can be created. This can be used in a domain environment only, and it relies on DCOM. For the command to succeed in creating a virtual smart card on a different computer, the user running this command must be a member in the local administrators group on the remote computer.
/pinpolicy	<p>If /pin prompt is used, /pinpolicy allows you to specify the following PIN policy options:</p> <p>minlen <minimum PIN length> If not specified, defaults to 8. The lower bound is 4.</p> <p>maxlen <maximum PIN length> If not specified, defaults to 127. The upper bound is 127.</p> <p>uppercase Can be ALLOWED, DISALLOWED, or REQUIRED. Default is ALLOWED.</p> <p>lowercase Can be ALLOWED, DISALLOWED, or REQUIRED. Default is ALLOWED.</p> <p>digits Can be ALLOWED, DISALLOWED, or REQUIRED. Default is ALLOWED.</p> <p>specialchars Can be ALLOWED, DISALLOWED, or REQUIRED. Default is ALLOWED.</p> <p>When using /pinpolicy, PIN characters must be printable ASCII characters.</p>
/attestation	<p>Configures attestation (subject only). This attestation uses an Attestation Identity Key (AIK) certificate as a trust anchor to vouch that the virtual smart card keys and certificates are truly hardware bound. The attestation methods are:</p> <p>AIK_AND_CERT Creates an AIK and obtains an AIK certificate from the Microsoft cloud certification authority (CA). This requires the device to have a TPM with an EK certificate. If this option is specified and there is no network connectivity, it is possible that creation of the virtual smart card will fail.</p> <p>AIK_ONLY Creates an AIK but does not obtain an AIK certificate.</p>
/?	Displays Help for this command.

Parameters for Destroy command

The Destroy command securely deletes a virtual smart card from a computer.

WARNING

When a virtual smart card is deleted, it cannot be recovered.

PARAMETER	DESCRIPTION
/instance	Specifies the instance ID of the virtual smart card to be removed. The instanceID was generated as output by Tpmvscmgr.exe when the card was created. The /instance parameter is a required field for the Destroy command.
/machine	Allows you to specify the name of a remote computer on which the virtual smart card will be deleted. This can be used in a domain environment only, and it relies on DCOM. For the command to succeed in deleting a virtual smart card on a different computer, the user running this command must be a member in the local administrators group on the remote computer.
/?	Displays Help for this command.

Remarks

Membership in the Administrators group (or equivalent) on the target computer is the minimum required to run all the parameters of this command.

For alphanumeric inputs, the full 127 character ASCII set is allowed.

Examples

The following command shows how to create a virtual smart card that can be later managed by a smart card management tool launched from another computer.

```
tpmvscmgr.exe create /name "VirtualSmartCardForCorpAccess" /AdminKey DEFAULT /PIN PROMPT
```

Alternatively, instead of using a default administrator key, you can create an administrator key at the command line. The following command shows how to create an administrator key.

```
tpmvscmgr.exe create /name "VirtualSmartCardForCorpAccess" /AdminKey PROMPT /PIN PROMPT
```

The following command will create the unmanaged virtual smart card that can be used to enroll certificates.

```
tpmvscmgr.exe create /name "VirtualSmartCardForCorpAccess" /AdminKey RANDOM /PIN PROMPT /generate
```

The preceding command will create a virtual smart card with a randomized administrator key. The key is automatically discarded after the card is created. This means that if the user forgets the PIN or wants to change the PIN, the user needs to delete the card and create it again. To delete the card, the user can run the following command.

```
tpmvscmgr.exe destroy /instance <instance ID>
```

where <instance ID> is the value printed on the screen when the user created the card. Specifically, for the first card created, the instance ID is ROOT\SMARTCARDREADER\0000.

The following command will create a TPM virtual smart card with the default value for the administrator key and a specified PIN policy and attestation method:

```
tpmvmgr.exe create /name "VirtualSmartCardForCorpAccess" /PIN PROMPT /pinpolicy minlen 4 maxlen 8  
/AdminKey DEFAULT /attestation AIK_AND_CERT /generate
```

Additional references

- [Virtual Smart Card Overview](#)

Windows and cloud security

7/1/2022 • 2 minutes to read • [Edit Online](#)

Today's workforce has more freedom and mobility than ever before. With the growth of enterprise cloud adoption, increased personal app usage, and increased use of third-party apps, the risk of data exposure is at its highest. Enabling Zero-Trust protection, Windows 11 works with Microsoft cloud services. Windows and cloud services together help organizations strengthen their multi-cloud security infrastructure, protect hybrid cloud workloads, and safeguard sensitive information while controlling access and mitigating threats.

Windows 11 includes the cloud services that are listed in the following table:

SERVICE TYPE	DESCRIPTION
Mobile device management (MDM) and Microsoft Endpoint Manager	<p>Windows 11 supports MDM, an enterprise management solution to help you manage your organization's security policies and business applications. MDM enables your security team to manage devices without compromising people's privacy on their personal devices.</p> <p>Non-Microsoft servers can be used to manage Windows 11 by using industry standard protocols.</p> <p>To learn more, see Mobile device management.</p>
Microsoft account	<p>When users add their Microsoft account to Windows 11, they can bring their Windows, Microsoft Edge, Xbox settings, web page favorites, files, photos, and more across their devices.</p> <p>The Microsoft account enables people to manage everything in one place. They can keep tabs on their subscriptions and order history, organize their family's digital life, update their privacy and security settings, track the health and safety of their devices, and even get rewards.</p> <p>To learn more, see Microsoft Accounts.</p>

SERVICE TYPE	DESCRIPTION
OneDrive	<p>OneDrive is your online storage for your files, photos, and data. OneDrive provides extra security, backup, and restore options for important files and photos. With options for both personal and business, people can use OneDrive to store and protect files in the cloud, allowing users to them on their laptops, desktops, and mobile devices. If a device is lost or stolen, people can quickly recover all their important files, photos, and data.</p> <p>The OneDrive Personal Vault also provides protection for your most sensitive files without losing the convenience of anywhere access. Files are secured by identity verification, yet easily accessible to users across their devices. Learn how to set up your Personal Vault.</p> <p>In the event of a ransomware attack, OneDrive can enable recovery. And if you've configured backups in OneDrive, you have more options to mitigate and recover from a ransomware attack. Learn more about how to recover from a ransomware attack using Office 365.</p>
Access to Azure Active Directory	<p>Microsoft Azure Active Directory (Azure AD) is a complete cloud identity and access management solution for managing identities and directories, enabling access to applications, and protecting identities from security threats.</p> <p>With Azure AD, you can manage and secure identities for your employees, partners, and customers to access the applications and services they need. Windows 11 works seamlessly with Azure Active Directory to provide secure access, identity management, and single sign-on to apps and services from anywhere.</p> <p>To learn more, see What is Azure AD?</p>

Next steps

- [Learn more about MDM and Windows 11](#)
- [Learn more about Windows security](#)

Windows security foundations

7/1/2022 • 2 minutes to read • [Edit Online](#)

Microsoft is committed to continuously invest in improving our software development process, building highly secure-by-design software, and addressing security compliance requirements. At Microsoft, we embed security and privacy considerations from the earliest life-cycle phases of all our software development processes. We build in security from the ground for powerful defense in today's threat environment.

Our strong security foundation uses Microsoft Security Development Lifecycle (SDL) Bug Bounty, support for product security standards and certifications, and Azure Code signing. As a result, we improve security by producing software with fewer defects and vulnerabilities instead of relying on applying updates after vulnerabilities have been identified.

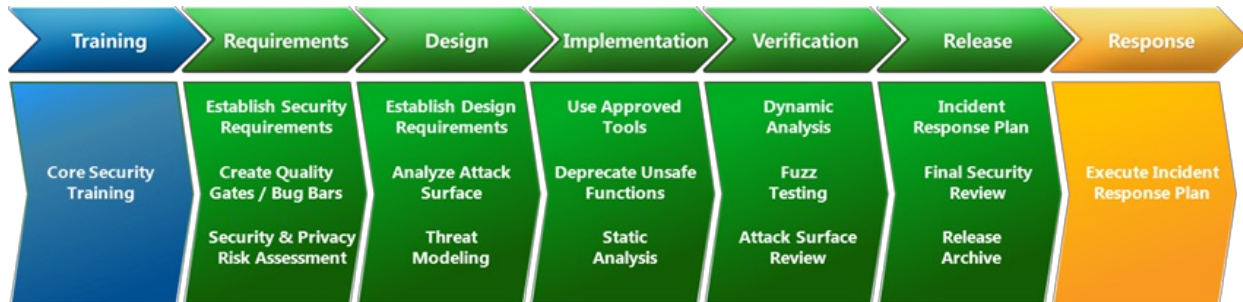
Use the links in the following table to learn more about the security foundations:

CONCEPT	DESCRIPTION
FIPS 140-2 Validation	<p>The Federal Information Processing Standard (FIPS) Publication 140-2 is a U.S. government standard. FIPS is based on Section 5131 of the Information Technology Management Reform Act of 1996. It defines the minimum security requirements for cryptographic modules in IT products. Microsoft maintains an active commitment to meeting the requirements of the FIPS 140-2 standard, having validated cryptographic modules against it since it was first established in 2001.</p> <p>Learn more about FIPS 140-2 Validation.</p>
Common Criteria Certifications	<p>Microsoft supports the Common Criteria certification program, ensures that products incorporate the features and functions required by relevant Common Criteria Protection Profiles, and completes Common Criteria certifications of Microsoft Windows products.</p> <p>Learn more about Common Criteria Certifications.</p>
Microsoft Security Development Lifecycle	<p>The Security Development Lifecycle (SDL) is a security assurance process that is focused on software development. The SDL has played a critical role in embedding security and privacy in software and culture at Microsoft.</p> <p>Learn more about Microsoft SDL.</p>
Microsoft Bug Bounty Program	<p>If you find a vulnerability in a Microsoft product, service, or device, we want to hear from you! If your vulnerability report affects a product or service that is within scope of one of our bounty programs below, you could receive a bounty award according to the program descriptions.</p> <p>Learn more about the Microsoft Bug Bounty Program.</p>

Microsoft Security Development Lifecycle

7/1/2022 • 2 minutes to read • [Edit Online](#)

The Security Development Lifecycle (SDL) is a security assurance process that is focused on software development. As a Microsoft-wide initiative and a mandatory policy since 2004, the SDL has played a critical role in embedding security and privacy in software and culture at Microsoft.



Combining a holistic and practical approach, the SDL aims to reduce the number and severity of vulnerabilities in software. The SDL introduces security and privacy throughout all phases of the development process.

The Microsoft SDL is based on three core concepts:

- Education
- Continuous process improvement
- Accountability

To learn more about the SDL, visit the [Security Engineering site](#).

And, download the [Simplified Implementation of the Microsoft SDL whitepaper](#).

FIPS 140-2 Validation

7/1/2022 • 166 minutes to read • [Edit Online](#)

FIPS 140-2 standard overview

The Federal Information Processing Standard (FIPS) Publication 140-2 is a U.S. government standard. FIPS is based on Section 5131 of the Information Technology Management Reform Act of 1996. It defines the minimum security requirements for cryptographic modules in IT products.

The [Cryptographic Module Validation Program \(CMVP\)](#) is a joint effort of the U.S. National Institute of Standards and Technology (NIST) and the Canadian Centre for Cyber Security (CCCS). It validates cryptographic modules against the Security Requirements for Cryptographic Modules (part of FIPS 140-2) and related FIPS cryptography standards. The FIPS 140-2 security requirements cover 11 areas related to the design and implementation of a cryptographic module. The NIST Information Technology Laboratory operates a related program that validates the FIPS approved cryptographic algorithms in the module.

Microsoft's approach to FIPS 140-2 validation

Microsoft maintains an active commitment to meeting the requirements of the FIPS 140-2 standard, having validated cryptographic modules against it since it was first established in 2001. Microsoft validates its cryptographic modules under the NIST CMVP, as described above. Multiple Microsoft products, including Windows 10, Windows Server, and many cloud services, use these cryptographic modules.

Using Windows in a FIPS 140-2 approved mode of operation

Windows 10 and Windows Server may be configured to run in a FIPS 140-2 approved mode of operation, commonly referred to as "FIPS mode." If you turn on FIPS mode, the Cryptographic Primitives Library (bcryptprimitives.dll) and Kernel Mode Cryptographic Primitives Library (CNG.sys) modules will run self-tests before Windows runs cryptographic operations. These self-tests are run according to FIPS 140-2 Section 4.9. They ensure that the modules are functioning properly.

The Cryptographic Primitives Library and the Kernel Mode Cryptographic Primitives Library are the only modules affected by FIPS mode. FIPS mode won't prevent Windows and its subsystems from using non-FIPS validated cryptographic algorithms. FIPS mode is merely advisory for applications or components other than the Cryptographic Primitives Library and the Kernel Mode Cryptographic Primitives Library.

US government regulations continue to mandate FIPS mode for government devices running Windows. Other customers should decide for themselves if FIPS mode is right for them. There are many applications and protocols that use FIPS mode policy to determine which cryptographic functionality to run. Customers seeking to follow the FIPS 140-2 standard should research the configuration settings of their applications and protocols. This research will help ensure that they can be configured to use FIPS 140-2 validated cryptography.

Achieving this FIPS 140-2 approved mode of operation of Windows requires administrators to complete all four steps outlined below.

Step 1: Ensure FIPS 140-2 validated cryptographic modules are installed

Administrators must ensure that all cryptographic modules installed are FIPS 140-2 validated. Tables listing validated modules, organized by operating system release, are available later in this article.

Step 2: Ensure all security policies for all cryptographic modules are followed

Each of the cryptographic modules has a defined security policy that must be met for the module to operate in

its FIPS 140-2 approved mode. The security policy may be found in each module's published Security Policy Document (SPD). The SPDs for each module may be found in the table of validated modules at the end of this article. Select the module version number to view the published SPD for the module.

Step 3: Enable the FIPS security policy

Windows provides the security policy setting, *System cryptography: Use FIPS-compliant algorithms for encryption, hashing, and signing*. This setting is used by some Microsoft products to determine whether to run in FIPS mode. When this policy is turned on, the validated cryptographic modules in Windows will also operate in FIPS mode. This policy may be set using Local Security Policy, as part of Group Policy, or through a Modern Device Management (MDM) solution. For more information on the policy, see [System cryptography: Use FIPS-compliant algorithms for encryption, hashing, and signing](#).

Step 4: Ensure that only FIPS validated cryptographic algorithms are used

FIPS mode is enforced at the level of the application or service. It is not enforced by the operating system or by individual cryptographic modules. Applications or services running in FIPS mode must follow the security policies of validated modules. They must not use a cryptographic algorithm that isn't FIPS-compliant.

In short, an application or service is running in FIPS mode if it:

- Checks for the policy flag
- Enforces security policies of validated modules

Frequently asked questions

How long does it take to certify a cryptographic module?

Microsoft begins certification of cryptographic modules after each major feature release of Windows 10 and Windows Server. The duration of each evaluation varies, depending on many factors.

When does Microsoft undertake a FIPS 140 validation?

The cadence for starting module validation aligns with the feature updates of Windows 10 and Windows Server. As the software industry evolves, operating systems release more frequently. Microsoft completes validation work on major releases but, in between releases, seeks to minimize the changes to the cryptographic modules.

What is the difference between *FIPS 140 validated* and *FIPS 140 compliant*?

FIPS 140 validated means that the cryptographic module, or a product that embeds the module, has been validated ("certified") by the CMVP as meeting the FIPS 140-2 requirements. *FIPS 140 compliant* is an industry term for IT products that rely on FIPS 140 validated products for cryptographic functionality.

How do I know if a Windows service or application is FIPS 140-2 validated?

The cryptographic modules used in Windows are validated through the CMVP. They aren't validated by individual services, applications, hardware peripherals, or other solutions. Any compliant solution must call a FIPS 140-2 validated cryptographic module in the underlying OS, and the OS must be configured to run in FIPS mode. Contact the vendor of the service, application, or product for information on whether it calls a validated cryptographic module.

What does *When operated in FIPS mode* mean on a certificate?

This label means that certain configuration and security rules must be followed to use the cryptographic module in compliance with its FIPS 140-2 security policy. Each module has its own security policy—a precise specification of the security rules under which it will operate—and employs approved cryptographic algorithms, cryptographic key management, and authentication techniques. The security rules are defined in the Security Policy Document (SPD) for each module.

What is the relationship between FIPS 140-2 and Common Criteria?

FIPS 140-2 and Common Criteria are two separate security standards with different, but complementary,

purposes. FIPS 140-2 is designed specifically for validating software and hardware cryptographic modules. Common Criteria are designed to evaluate security functions in IT software and hardware products. Common Criteria evaluations often rely on FIPS 140-2 validations to provide assurance that basic cryptographic functionality is implemented properly.

How does FIPS 140 relate to Suite B?

Suite B is a set of cryptographic algorithms defined by the U.S. National Security Agency (NSA) as part of its Cryptographic Modernization Program. The set of Suite B cryptographic algorithms are to be used for both unclassified information and most classified information. The Suite B cryptographic algorithms are a subset of the FIPS approved cryptographic algorithms allowed by the FIPS 140-2 standard.

Is SMB3 (Server Message Block) FIPS 140 compliant in Windows?

SMB3 can be FIPS 140 compliant, if Windows is configured to operate in FIPS 140 mode on both client and server. In FIPS mode, SMB3 relies on the underlying Windows FIPS 140 validated cryptographic modules for cryptographic operations.

Microsoft FIPS 140-2 validated cryptographic modules

The following tables identify the cryptographic modules used in an operating system, organized by release.

Modules used by Windows

Windows 10 Fall 2018 Update (Version 1809)

Validated Editions: Home, Pro, Enterprise, Education

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Cryptographic Primitives Library	10.0.17763	#3197	See Security Policy and Certificate page for algorithm information
Kernel Mode Cryptographic Primitives Library	10.0.17763	#3196	See Security Policy and Certificate page for algorithm information
Code Integrity	10.0.17763	#3644	See Security Policy and Certificate page for algorithm information
Windows OS Loader	10.0.17763	#3615	See Security Policy and Certificate page for algorithm information
Secure Kernel Code Integrity	10.0.17763	#3651	See Security Policy and Certificate page for algorithm information
BitLocker Dump Filter	10.0.17763	#3092	See Security Policy and Certificate page for algorithm information
Boot Manager	10.0.17763	#3089	See Security Policy and Certificate page for algorithm information

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Virtual TPM	10.0.17763	#3690	See Security Policy and Certificate page for algorithm information

Windows 10 Spring 2018 Update (Version 1803)

Validated Editions: Home, Pro, Enterprise, Education

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Cryptographic Primitives Library	10.0.17134	#3197	See Security Policy and Certificate page for algorithm information
Kernel Mode Cryptographic Primitives Library	10.0.17134	#3196	See Security Policy and Certificate page for algorithm information
Code Integrity	10.0.17134	#3195	See Security Policy and Certificate page for algorithm information
Windows OS Loader	10.0.17134	#3480	See Security Policy and Certificate page for algorithm information
Secure Kernel Code Integrity	10.0.17134	#3096	See Security Policy and Certificate page for algorithm information
BitLocker Dump Filter	10.0.17134	#3092	See Security Policy and Certificate page for algorithm information
Boot Manager	10.0.17134	#3089	See Security Policy and Certificate page for algorithm information

Windows 10 Fall Creators Update (Version 1709)

Validated Editions: Home, Pro, Enterprise, Education, S, Surface Hub, Mobile

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Cryptographic Primitives Library	10.0.16299	#3197	See Security Policy and Certificate page for algorithm information
Kernel Mode Cryptographic Primitives Library	10.0.16299	#3196	See Security Policy and Certificate page for algorithm information
Code Integrity	10.0.16299	#3195	See Security Policy and Certificate page for algorithm information

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Windows OS Loader	10.0.16299	#3194	See Security Policy and Certificate page for algorithm information
Secure Kernel Code Integrity	10.0.16299	#3096	See Security Policy and Certificate page for algorithm information
BitLocker Dump Filter	10.0.16299	#3092	See Security Policy and Certificate page for algorithm information
Windows Resume	10.0.16299	#3091	See Security Policy and Certificate page for algorithm information
Boot Manager	10.0.16299	#3089	See Security Policy and Certificate page for algorithm information

Windows 10 Creators Update (Version 1703)

Validated Editions: Home, Pro, Enterprise, Education, S, Surface Hub, Mobile

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
----------------------	-----------------------------------	--------------------	------------

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Cryptographic Primitives Library (bcryptprimitives.dll and ncryptssp.dll)	10.0.15063	#3095	<p>FIPS approved algorithms: AES (Cert. #4624); CKG (vendor affirmed); CVL (Certs #1278 and #1281); DRBG (Cert. #1555); DSA (Cert. #1223); ECDSA (Cert. #1133); HMAC (Cert. #3061); KAS (Cert. #127); KBKDF (Cert. #140); KTS (AES Cert. #4626; key establishment methodology provides between 128 bits and 256 bits of encryption strength); PBKDF (vendor affirmed); RSA (Certs. #2521 and #2522); SHS (Cert. #3790); Triple-DES (Cert. #2459)</p> <p>Other algorithms: HMAC-MD5; MD5; DES; Legacy CAPI KDF; MD2; MD4; RC2; RC4; RSA (encrypt/decrypt)</p> <p>Validated Component Implementations: FIPS186-4 ECDSA - Signature Generation of hash sized messages (Cert. #1133); FIPS186-4 RSA; PKCS#1 v2.1 - RSASP1 Signature Primitive (Cert. #2521); FIPS186-4 RSA; RSADP - RSADP Primitive (Cert. #1281); SP800-135 - Section 4.1.1, IKEv1 Section 4.1.2, IKEv2 Section 4.2, TLS (Cert. #1278)</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Kernel Mode Cryptographic Primitives Library (cng.sys)	10.0.15063	#3094	<p>#3094</p> <p>FIPS approved algorithms: AES (Certs. #4624 and #4626); CKG (vendor affirmed); CVL (Certs. #1278 and #1281); DRBG (Cert. #1555); DSA (Cert. #1223); ECDSA (Cert. #1133); HMAC (Cert. #3061); KAS (Cert. #127); KBKDF (Cert. #140); KTS (AES Cert. #4626; key establishment methodology provides between 128 bits and 256 bits of encryption strength); PBKDF (vendor affirmed); RSA (Certs. #2521 and #2523); SHS (Cert. #3790); Triple-DES (Cert. #2459)</p> <p>Other algorithms: HMAC-MD5; MD5; NDRNG; DES; Legacy CAPI KDF; MD2; MD4; RC2; RC4; RSA (encrypt/decrypt)</p> <p>Validated Component Implementations: FIPS186-4 ECDSA - Signature Generation of hash sized messages (Cert.; FIPS186-4 RSA; PKCS#1 v2.1 - RSASP1 Signature Primitive (Cert.) (https://csrc.nist.gov/projects/cryptographic-module-validation-program/certificate/3094)#2521); FIPS186-4 RSA; RSADP - RSADP Primitive (Cert.) (https://csrc.nist.gov/projects/cryptographic-module-validation-program/certificate/3094)</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Boot Manager	10.0.15063	#3089	FIPS approved algorithms: AES (Certs. #4624 and #4625); CKG (vendor affirmed); HMAC (Cert. #3061); PBKDF (vendor affirmed); RSA (Cert. #2523); SHS (Cert. #3790) Other algorithms: PBKDF (vendor affirmed); VMK KDF (vendor affirmed)
Windows OS Loader	10.0.15063	#3090	FIPS approved algorithms: AES (Certs. #4624 and #4625); RSA (Cert. #2523); SHS (Cert. #3790) Other algorithms: NDRNG
Windows Resume ^[1]	10.0.15063	#3091	FIPS approved algorithms: AES (Certs. #4624 and #4625); RSA (Cert. #2523); SHS (Cert. #3790)
BitLocker® Dump Filter ^[2]	10.0.15063	#3092	FIPS approved algorithms: AES (Certs. #4624 and #4625); RSA (Cert. #2522); SHS (Cert. #3790)
Code Integrity (ci.dll)	10.0.15063	#3093	FIPS approved algorithms: AES (Cert. #4624); RSA (Certs. #2522 and #2523); SHS (Cert. #3790) Validated Component Implementations: FIPS186-4 RSA; PKCS#1 v1.5 - RSASP1 Signature Primitive (Cert. #1282)
Secure Kernel Code Integrity (skci.dll) ^[3]	10.0.15063	#3096	FIPS approved algorithms: AES (Cert. #4624); RSA (Certs. #2522 and #2523); SHS (Cert. #3790) Validated Component Implementations: FIPS186-4 RSA; PKCS#1 v1.5 - RSASP1 Signature Primitive (Cert. #1282)

^[1] Applies only to Home, Pro, Enterprise, Education, and S.

^[2] Applies only to Pro, Enterprise, Education, S, Mobile, and Surface Hub

[3] Applies only to Pro, Enterprise, Education, and S

Windows 10 Anniversary Update (Version 1607)

Validated Editions: Home, Pro, Enterprise, Enterprise LTSB, Mobile

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Cryptographic Primitives Library (bcryptprimitives.dll and ncryptsslp.dll)	10.0.14393	#2937	<p>FIPS approved algorithms: AES (Cert. #4064); DRBG (Cert. #1217); DSA (Cert. #1098); ECDSA (Cert. #911); HMAC (Cert. #2651); KAS (Cert. #92); KBKDF (Cert. #101); KTS (AES Cert. #4062; key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); PBKDF (vendor affirmed); RSA (Certs. #2192, #2193, and #2195); SHS (Cert. #3347); Triple-DES (Cert. #2227)</p> <p>Other algorithms: HMAC-MD5; MD5; DES; Legacy CAPI KDF; MD2; MD4; RC2; RC4; RSA (encrypt/decrypt)</p> <p>Validated Component Implementations: FIPS186-4 ECDSA - Signature Generation of hash sized messages (Cert. #922); FIPS186-4 RSA; PKCS#1 v2.1 - RSASP1 Signature Primitive (Cert. #888); FIPS186-4 RSA; RSADP - RSADP Primitive (Cert. #887); SP800-135 - Section 4.1.1, IKEv1 Section 4.1.2, IKEv2 Section 4.2, TLS (Cert. #886)</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Kernel Mode Cryptographic Primitives Library (cng.sys)	10.0.14393	#2936	<p>FIPS approved algorithms: AES (Cert. #4064); DRBG (Cert. #1217); DSA (Cert. #1098); ECDSA (Cert. #911); HMAC (Cert. #2651); KAS (Cert. #92); KBKDF (Cert. #101); KTS (AES Cert. #4062; key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); PBKDF (vendor affirmed); RSA (Certs. #2192, #2193, and #2195); SHS (Cert. #3347); Triple-DES (Cert. #2227)</p> <p>Other algorithms: HMAC-MD5; MD5; NDRNG; DES; Legacy CAPI KDF; MD2; MD4; RC2; RC4; RSA (encrypt/decrypt)</p> <p>Validated Component Implementations: FIPS186-4 ECDSA - Signature Generation of hash sized messages (Cert. #922); FIPS186-4 RSA; PKCS#1 v2.1 - RSASP1 Signature Primitive (Cert. #888); FIPS186-4 RSA; RSADP - RSADP Primitive (Cert. #887)</p>
Boot Manager	10.0.14393	#2931	<p>FIPS approved algorithms: AES (Certs. #4061 and #4064); HMAC (Cert. #2651); PBKDF (vendor affirmed); RSA (Cert. #2193); SHS (Cert. #3347)</p> <p>Other algorithms: MD5; PBKDF (non-compliant); VMK KDF</p>
BitLocker® Windows OS Loader (winload)	10.0.14393	#2932	<p>FIPS approved algorithms: AES (Certs. #4061 and #4064); RSA (Cert. #2193); SHS (Cert. #3347)</p> <p>Other algorithms: NDRNG; MD5</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
BitLocker® Windows Resume (winresume) ^[1]	10.0.14393	#2933	FIPS approved algorithms: AES (Certs. #4061 and #4064); RSA (Cert. #2193); SHS (Cert. #3347) Other algorithms: MD5
BitLocker® Dump Filter (dumpfve.sys) ^[2]	10.0.14393	#2934	FIPS approved algorithms: AES (Certs. #4061 and #4064)
Code Integrity (ci.dll)	10.0.14393	#2935	FIPS approved algorithms: RSA (Cert. #2193); SHS (Cert. #3347) Other algorithms: AES (non-compliant); MD5 Validated Component Implementations: FIPS186-4 RSA; PKCS#1 v2.1 - RSASP1 Signature Primitive (Cert. #888)
Secure Kernel Code Integrity (skci.dll) ^[3]	10.0.14393	#2938	FIPS approved algorithms: RSA (Certs. #2193); SHS (Certs. #3347) Other algorithms: MD5 Validated Component Implementations: FIPS186-4 RSA; PKCS#1 v2.1 - RSASP1 Signature Primitive (Cert. #888)

^[1] Applies only to Home, Pro, Enterprise, and Enterprise LTSB

^[2] Applies only to Pro, Enterprise, Enterprise LTSB, and Mobile

^[3] Applies only to Pro, Enterprise, and Enterprise LTSB

Windows 10 November 2015 Update (Version 1511)

Validated Editions: Home, Pro, Enterprise, Enterprise LTSB, Mobile, Surface Hub

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
----------------------	-----------------------------------	--------------------	------------

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Cryptographic Primitives Library (bcryptprimitives.dll and ncryptssp.dll)	10.0.10586	#2606	<p>FIPS approved algorithms: AES (Certs. #3629); DRBG (Certs. #955); DSA (Certs. #1024); ECDSA (Certs. #760); HMAC (Certs. #2381); KAS (Certs. #72; key agreement; key establishment methodology provides between 112 bits and 256 bits of encryption strength); KBKDF (Certs. #72); KTS (AES Certs. #3653; key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); PBKDF (vendor affirmed); RSA (Certs. #1887, #1888, and #1889); SHS (Certs. #3047); Triple-DES (Certs. #2024)</p> <p>Other algorithms: DES; HMAC-MD5; Legacy CAPI KDF; MD2; MD4; MD5; RC2; RC4; RSA (encrypt/decrypt)</p> <p>Validated Component Implementations: FIPS186-4 ECDSA - Signature Generation of hash sized messages (Cert. #666); FIPS186-4 RSA; PKCS#1 v2.1 - RSASP1 Signature Primitive (Cert. #665); FIPS186-4 RSA; RSADP - RSADP Primitive (Cert. #663); SP800-135 - Section 4.1.1, IKEv1 Section 4.1.2, IKEv2 Section 4.2, TLS (Cert. #664)</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Kernel Mode Cryptographic Primitives Library (cng.sys)	10.0.10586	#2605	<p>FIPS approved algorithms: AES (Certs. #3629); DRBG (Certs. #955); DSA (Certs. #1024); ECDSA (Certs. #760); HMAC (Certs. #2381); KAS (Certs. #72; key agreement; key establishment methodology provides between 112 bits and 256 bits of encryption strength); KBKDF (Certs. #72); KTS (AES Certs. #3653; key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); PBKDF (vendor affirmed); RSA (Certs. #1887, #1888, and #1889); SHS (Certs. #3047); Triple-DES (Certs. #2024)</p> <p>Other algorithms: DES; HMAC-MD5; Legacy CAPI KDF; MD2; MD4; MD5; RC2; RC4; RSA (encrypt/decrypt)</p> <p>Validated Component Implementations: FIPS186-4 ECDSA - Signature Generation of hash sized messages (Cert. #666); FIPS186-4 RSA; PKCS#1 v2.1 - RSASP1 Signature Primitive (Cert. #665); FIPS186-4 RSA; RSADP - RSADP Primitive (Cert. #663)</p>
Boot Manager ^[4]	10.0.10586	#2700	<p>FIPS approved algorithms: AES (Certs. #3653); HMAC (Cert. #2381); PBKDF (vendor affirmed); RSA (Cert. #1871); SHS (Certs. #3047 and #3048)</p> <p>Other algorithms: MD5; KDF (non-compliant); PBKDF (non-compliant)</p>
BitLocker® Windows OS Loader (winload) ^[5]	10.0.10586	#2701	<p>FIPS approved algorithms: AES (Certs. #3629 and #3653); RSA (Cert. #1871); SHS (Cert. #3048)</p> <p>Other algorithms: MD5; NDRNG</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
BitLocker® Windows Resume (winresume) ^[6]	10.0.10586	#2702	FIPS approved algorithms: AES (Certs. #3653); RSA (Cert. #1871); SHS (Cert. #3048) Other algorithms: MD5
BitLocker® Dump Filter (dumpfve.sys) ^[7]	10.0.10586	#2703	FIPS approved algorithms: AES (Certs. #3653)
Code Integrity (ci.dll)	10.0.10586	#2604	FIPS approved algorithms: RSA (Certs. #1871); SHS (Certs. #3048) Other algorithms: AES (non-compliant); MD5 Validated Component Implementations: FIPS186-4 RSA; PKCS#1 v2.1 - RSASP1 Signature Primitive (Cert. #665)
Secure Kernel Code Integrity (skci.dll) ^[8]	10.0.10586	#2607	FIPS approved algorithms: RSA (Certs. #1871); SHS (Certs. #3048) Other algorithms: MD5 Validated Component Implementations: FIPS186-4 RSA; PKCS#1 v2.1 - RSASP1 Signature Primitive (Cert. #665)

^[4] Applies only to Home, Pro, Enterprise, Mobile, and Surface Hub

^[5] Applies only to Home, Pro, Enterprise, Mobile, and Surface Hub

^[6] Applies only to Home, Pro, and Enterprise

^[7] Applies only to Pro, Enterprise, Mobile, and Surface Hub

^[8] Applies only to Enterprise and Enterprise LTSB

Windows 10 (Version 1507)

Validated Editions: Home, Pro, Enterprise, Enterprise LTSB, Mobile, and Surface Hub

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
----------------------	-----------------------------------	--------------------	------------

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Cryptographic Primitives Library (bcryptprimitives.dll and ncryptssp.dll)	10.0.10240	#2606	<p>FIPS approved algorithms: AES (Certs. #3497); DRBG (Certs. #868); DSA (Certs. #983); ECDSA (Certs. #706); HMAC (Certs. #2233); KAS (Certs. #64; key agreement; key establishment methodology provides between 112 bits and 256 bits of encryption strength); KBKDF (Certs. #66); KTS (AES Certs. #3507; key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); PBKDF (vendor affirmed); RSA (Certs. #1783, #1798, and #1802); SHS (Certs. #2886); Triple-DES (Certs. #1969)</p> <p>Other algorithms: DES; HMAC-MD5; Legacy CAPI KDF; MD2; MD4; MD5; RC2; RC4; RSA (encrypt/decrypt)</p> <p>Validated Component Implementations: FIPS186-4 RSA; PKCS#1 v2.1 - RSASP1 Signature Primitive (Cert. #572); FIPS186-4 RSA; RSADP - RSADP Primitive (Cert. #576); SP800-135 - Section 4.1.1, IKEv1 Section 4.1.2, IKEv2 Section 4.2, TLS (Cert. #575)</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Kernel Mode Cryptographic Primitives Library (cng.sys)	10.0.10240	#2605	<p>FIPS approved algorithms: AES (Certs. #3497); DRBG (Certs. #868); DSA (Certs. #983); ECDSA (Certs. #706); HMAC (Certs. #2233); KAS (Certs. #64; key agreement; key establishment methodology provides between 112 bits and 256 bits of encryption strength); KBKDF (Certs. #66); KTS (AES Certs. #3507; key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); PBKDF (vendor affirmed); RSA (Certs. #1783, #1798, and #1802); SHS (Certs. #2886); Triple-DES (Certs. #1969)</p> <p>Other algorithms: DES; HMAC-MD5; Legacy CAPI KDF; MD2; MD4; MD5; RC2; RC4; RSA (encrypt/decrypt)</p> <p>Validated Component Implementations: FIPS186-4 RSA; PKCS#1 v2.1 - RSASP1 Signature Primitive (Cert. #572); FIPS186-4 RSA; RSADP - RSADP Primitive (Cert. #576)</p>
Boot Manager ^[9]	10.0.10240	#2600	<p>FIPS approved algorithms: AES (Cert. #3497); HMAC (Cert. #2233); KTS (AES Cert. #3498); PBKDF (vendor affirmed); RSA (Cert. #1784); SHS (Certs. #2871 and #2886)</p> <p>Other algorithms: MD5; KDF (non-compliant); PBKDF (non-compliant)</p>
BitLocker® Windows OS Loader (winload) ^[10]	10.0.10240	#2601	<p>FIPS approved algorithms: AES (Certs. #3497 and #3498); RSA (Cert. #1784); SHS (Cert. #2871)</p> <p>Other algorithms: MD5; NDRNG</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
BitLocker® Windows Resume (winresume) ^[11]	10.0.10240	#2602	FIPS approved algorithms: AES (Certs. #3497 and #3498); RSA (Cert. #1784); SHS (Cert. #2871) Other algorithms: MD5
BitLocker® Dump Filter (dumpfve.sys) ^[12]	10.0.10240	#2603	FIPS approved algorithms: AES (Certs. #3497 and #3498)
Code Integrity (ci.dll)	10.0.10240	#2604	FIPS approved algorithms: RSA (Certs. #1784); SHS (Certs. #2871) Other algorithms: AES (non-compliant); MD5 Validated Component Implementations: FIPS186-4 RSA; PKCS#1 v2.1 - RSASP1 Signature Primitive (Cert. #572)
Secure Kernel Code Integrity (skci.dll) ^[13]	10.0.10240	#2607	FIPS approved algorithms: RSA (Certs. #1784); SHS (Certs. #2871) Other algorithms: MD5 Validated Component Implementations: FIPS186-4 RSA; PKCS#1 v2.1 - RSASP1 Signature Primitive (Cert. #572)

^[9] Applies only to Home, Pro, Enterprise, and Enterprise LTSC

^[10] Applies only to Home, Pro, Enterprise, and Enterprise LTSC

^[11] Applies only to Home, Pro, Enterprise, and Enterprise LTSC

^[12] Applies only to Pro, Enterprise, and Enterprise LTSC

^[13] Applies only to Enterprise and Enterprise LTSC

Windows 8.1

Validated Editions: RT, Pro, Enterprise, Phone, Embedded

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
----------------------	-----------------------------------	--------------------	------------

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Cryptographic Primitives Library (bcryptprimitives.dll and ncryptssp.dll)	6.3.9600 6.3.9600.17031	#2357	<p>FIPS approved algorithms: AES (Cert. #2832); DRBG (Certs. #489); DSA (Cert. #855); ECDSA (Cert. #505); HMAC (Cert. #1773); KAS (Cert. #47); KBKDF (Cert. #30); PBKDF (vendor affirmed); RSA (Certs. #1487, #1493, and #1519); SHS (Cert. #2373); Triple-DES (Cert. #1692)</p> <p>Other algorithms: AES (Cert. #2832, key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); AES-GCM encryption (non-compliant); DES; HMAC MD5; Legacy CAPI KDF; MD2; MD4; MD5; NDRNG; RC2; RC4; RSA (encrypt/decrypt)#2832, key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); AES-GCM encryption (non-compliant); DES; HMAC MD5; Legacy CAPI KDF; MD2; MD4; MD5; NDRNG; RC2; RC4; RSA (encrypt/decrypt)</p> <p>Validated Component Implementations: FIPS186-4 ECDSA - Signature Generation of hash sized messages (Cert. #288); FIPS186-4 RSA; PKCS#1 v2.1 - RSASP1 Signature Primitive (Cert. #289); SP800-135 - Section 4.1.1, IKEv1 Section 4.1.2, IKEv2 Section 4.2, TLS (Cert. #323)</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Kernel Mode Cryptographic Primitives Library (cng.sys)	6.3.9600 6.3.9600.17042	#2356	<p>FIPS approved algorithms: AES (Cert. #2832); DRBG (Certs. #489); ECDSA (Cert. #505); HMAC (Cert. #1773); KAS (Cert. #47); KBKDF (Cert. #30); PBKDF (vendor affirmed); RSA (Certs. #1487, #1493, and #1519); SHS (Cert. # 2373); Triple-DES (Cert. #1692)</p> <p>Other algorithms: AES (Cert. #2832, key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); AES-GCM encryption (non-compliant); DES; HMAC MD5; Legacy CAPI KDF; MD2; MD4; MD5; NDRNG; RC2; RC4; RSA (encrypt/decrypt)</p> <p>Validated Component Implementations: FIPS186-4 ECDSA - Signature Generation of hash sized messages (Cert. #288); FIPS186-4 RSA; PKCS#1 v2.1 - RSASP1 Signature Primitive (Cert. #289)</p>
Boot Manager	6.3.9600 6.3.9600.17031	#2351	<p>FIPS approved algorithms: AES (Cert. #2832); HMAC (Cert. #1773); PBKDF (vendor affirmed); RSA (Cert. #1494); SHS (Certs. # 2373 and #2396)</p> <p>Other algorithms: MD5; KDF (non-compliant); PBKDF (non-compliant)</p>
BitLocker® Windows OS Loader (winload)	6.3.9600 6.3.9600.17031	#2352	<p>FIPS approved algorithms: AES (Cert. #2832); RSA (Cert. #1494); SHS (Cert. #2396)</p> <p>Other algorithms: MD5; NDRNG</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
BitLocker® Windows Resume (winresume) ^[14]	6.3.9600.6.3.9600.17031	#2353	FIPS approved algorithms: AES (Cert. #2832); RSA (Cert. #1494); SHS (Certs. #2373 and #2396) Other algorithms: MD5
BitLocker® Dump Filter (dumpfve.sys)	6.3.9600.6.3.9600.17031	#2354	FIPS approved algorithms: AES (Cert. #2832) Other algorithms: N/A
Code Integrity (ci.dll)	6.3.9600.6.3.9600.17031	#2355	FIPS approved algorithms: RSA (Cert. #1494); SHS (Cert. #2373) Other algorithms: MD5 Validated Component Implementations: PKCS#1 v2.1 - RSASP1 Signature Primitive (Cert. #289)

[14] Applies only to Pro, Enterprise, and Embedded 8.

Windows 8

Validated Editions: RT, Home, Pro, Enterprise, Phone

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
----------------------	-----------------------------------	--------------------	------------

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Cryptographic Primitives Library (BCRYPTPRIMITIVES.DLL)	6.2.9200	#1892	<p>FIPS approved algorithms: AES (Certs. #2197 and #2216); DRBG (Certs. #258); DSA (Cert. #687); ECDSA (Cert. #341); HMAC (Cert. #1345); KAS (Cert. #36); KBKDF (Cert. #3); PBKDF (vendor affirmed); RSA (Certs. #1133 and #1134); SHS (Cert. #1903); Triple-DES (Cert. #1387)</p> <p>Other algorithms: AES (Cert. #2197, key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); DES; Legacy CAPI KDF; MD2; MD4; MD5; HMAC MD5; RC2; RC4; RSA (encrypt/decrypt)#258); DSA (Cert.); ECDSA (Cert.); HMAC (Cert.); KAS (Cert); KBKDF (Cert.); PBKDF (vendor affirmed); RSA (Certs. and); SHS (Cert.); Triple-DES (Cert.)</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Kernel Mode Cryptographic Primitives Library (cng.sys)	6.2.9200	#1891	<p>FIPS approved algorithms: AES (Certs. #2197 and #2216); DRBG (Certs. #258 and #259); ECDSA (Cert. #341); HMAC (Cert. #1345); KAS (Cert. #36); KBKDF (Cert. #3); PBKDF (vendor affirmed); RNG (Cert. #1110); RSA (Certs. #1133 and #1134); SHS (Cert. #1903); Triple-DES (Cert. #1387)</p> <p>Other algorithms: AES (Cert. #2197, key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); DES; Legacy CAPI KDF; MD2; MD4; MD5; HMAC MD5; RC2; RC4; RSA (encrypt/decrypt)#258 and); ECDSA (Cert.); HMAC (Cert.); KAS (Cert.); KBKDF (Cert.); PBKDF (vendor affirmed); RNG (Cert.); RSA (Certs. and); SHS (Cert.); Triple-DES (Cert.)</p> <p>Other algorithms: AES (Certificate, key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); DES; Legacy CAPI KDF; MD2; MD4; MD5; HMAC MD5; RC2; RC4; RSA (encrypt/decrypt)</p>
Boot Manager	6.2.9200	#1895	<p>FIPS approved algorithms: AES (Certs. #2196 and #2198); HMAC (Cert. #1347); RSA (Cert. #1132); SHS (Cert. #1903)</p> <p>Other algorithms: MD5</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
BitLocker® Windows OS Loader (WINLOAD)	6.2.9200	#1896	<p>FIPS approved algorithms: AES (Certs. #2196 and #2198); RSA (Cert. #1132); SHS (Cert. #1903)</p> <p>Other algorithms: AES (Cert. #2197; non-compliant); MD5; Non-Approved RNG</p>
BitLocker® Windows Resume (WINRESUME) ^[15]	6.2.9200	#1898	<p>FIPS approved algorithms: AES (Certs. #2196 and #2198); RSA (Cert. #1132); SHS (Cert. #1903)</p> <p>Other algorithms: MD5</p>
BitLocker® Dump Filter (DUMPFVE.SYS)	6.2.9200	#1899	<p>FIPS approved algorithms: AES (Certs. #2196 and #2198)</p> <p>Other algorithms: N/A</p>
Code Integrity (CI.DLL)	6.2.9200	#1897	<p>FIPS approved algorithms: RSA (Cert. #1132); SHS (Cert. #1903)</p> <p>Other algorithms: MD5</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Enhanced DSS and Diffie-Hellman Cryptographic Provider (DSSENH.DLL)	6.2.9200	#1893	<p>FIPS approved algorithms: DSA (Cert. #686); SHS (Cert. #1902); Triple-DES (Cert. #1386); Triple-DES MAC (Triple-DES Cert. #1386, vendor affirmed)</p> <p>Other algorithms: DES; DES MAC; DES40; DES40 MAC; Diffie-Hellman; MD5; RC2; RC2 MAC; RC4; Triple-DES (Cert. #1386, key wrapping; key establishment methodology provides 112 bits of encryption strength; non-compliant less than 112 bits of encryption strength)#1902); Triple-DES (Cert.); Triple-DES MAC (Triple-DES Certificate, vendor affirmed)</p> <p>Other algorithms: DES; DES MAC; DES40; DES40 MAC; Diffie-Hellman; MD5; RC2; RC2 MAC; RC4; Triple-DES (Certificate, key wrapping; key establishment methodology provides 112 bits of encryption strength; non-compliant less than 112 bits of encryption strength)</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Enhanced Cryptographic Provider (RSAENH.DLL)	6.2.9200	#1894	<p>FIPS approved algorithms: AES (Cert. #2196); HMAC (Cert. #1346); RSA (Cert. #1132); SHS (Cert. #1902); Triple-DES (Cert. #1386)</p> <p>Other algorithms: AES (Cert. #2196, key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); DES; MD2; MD4; MD5; RC2; RC4; RSA (key wrapping; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength); Triple-DES (Cert. #1386, key wrapping; key establishment methodology provides 112 bits of encryption strength; non-compliant less than 112 bits of encryption strength)</p>

[15] Applies only to Home and Pro

Windows 7

Validated Editions: Windows 7, Windows 7 SP1

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
----------------------	-----------------------------------	--------------------	------------

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Cryptographic Primitives Library (BCRYPTPRIMITIVES.DLL)	6.1.7600.16385 6.1.7601.17514	1329	<p>FIPS approved algorithms: AES (Certs. #1168 and #1178); AES GCM (Cert. #1168, vendor-affirmed); AES GMAC (Cert. #1168, vendor-affirmed); DRBG (Certs. #23 and #24); DSA (Cert. #386); ECDSA (Cert. #141); HMAC (Cert. #677); KAS (SP 800-56A, vendor affirmed, key agreement; key establishment methodology provides 80 bits to 256 bits of encryption strength); RNG (Cert. #649); RSA (Certs. #559 and #560); SHS (Cert. #1081); Triple-DES (Cert. #846)</p> <p>Other algorithms: AES (Cert. #1168, key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); DES; Diffie-Hellman (key agreement; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength); MD2; MD4; MD5; HMAC MD5; RC2; RC4#559 and); SHS (Cert.); Triple-DES (Cert.)</p> <p>Other algorithms: AES (Certificate, key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); DES; Diffie-Hellman (key agreement; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength); MD2; MD4; MD5; HMAC MD5;</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Kernel Mode Cryptographic Primitives Library (cng.sys)	6.1.7600.16385 6.1.7600.16915 6.1.7600.21092 6.1.7601.17514 6.1.7601.17725 6.1.7601.17919 6.1.7601.21861 6.1.7601.22076	1328	RC2; RC4 FIPS approved algorithms: AES (Certs. #1168 and #1178); AES GCM (Cert. #1168 , vendor-affirmed); AES GMAC (Cert. #1168 , vendor-affirmed); DRBG (Certs. #23 and #24); ECDSA (Cert. #141); HMAC (Cert. #677); KAS (SP 800-56A, vendor affirmed, key agreement; key establishment methodology provides 80 bits to 256 bits of encryption strength); RNG (Cert. #649); RSA (Certs. #559 and #560); SHS (Cert. #1081); Triple-DES (Cert. #846) Other algorithms: AES (Cert. #1168 , key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); DES; Diffie-Hellman (key agreement; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength); MD2; MD4; MD5; HMAC MD5; RC2; RC4
Boot Manager	6.1.7600.16385 6.1.7601.17514	1319	FIPS approved algorithms: AES (Certs. #1168 and #1177); HMAC (Cert. #675); RSA (Cert. #557); SHS (Cert. #1081) Other algorithms: MD5#1168 and); HMAC (Cert.); RSA (Cert.); SHS (Cert.) Other algorithms: MD5

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Winload OS Loader (winload.exe)	6.1.7600.16385 6.1.7600.16757 6.1.7600.20897 6.1.7600.20916 6.1.7601.17514 6.1.7601.17556 6.1.7601.21655 6.1.7601.21675	1326	FIPS approved algorithms: AES (Certs. #1168 and #1177); RSA (Cert. #557); SHS (Cert. #1081) Other algorithms: MD5
BitLocker™ Drive Encryption	6.1.7600.16385 6.1.7600.16429 6.1.7600.16757 6.1.7600.20536 6.1.7600.20873 6.1.7600.20897 6.1.7600.20916 6.1.7601.17514 6.1.7601.17556 6.1.7601.21634 6.1.7601.21655 6.1.7601.21675	1332	FIPS approved algorithms: AES (Certs. #1168 and #1177); HMAC (Cert. #675); SHS (Cert. #1081) Other algorithms: Elephant Diffuser
Code Integrity (CI.DLL)	6.1.7600.16385 6.1.7600.17122v6.1.7600.21320 6.1.7601.17514 6.1.7601.17950v6.1.7601.22108	1327	FIPS approved algorithms: RSA (Cert. #557); SHS (Cert. #1081) Other algorithms: MD5
Enhanced DSS and Diffie-Hellman Cryptographic Provider (DSSENH.DLL)	6.1.7600.16385 (no change in SP1)	1331	FIPS approved algorithms: DSA (Cert. #385); RNG (Cert. #649); SHS (Cert. #1081); Triple-DES (Cert. #846); Triple-DES MAC (Triple-DES Cert. #846 , vendor affirmed) Other algorithms: DES; DES MAC; DES40; DES40 MAC; Diffie-Hellman; MD5; RC2; RC2 MAC; RC4

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Enhanced Cryptographic Provider (RSAENH.DLL)	6.1.7600.16385 (no change in SP1)	1330	FIPS approved algorithms: AES (Cert. #1168); DRBG (Cert. #23); HMAC (Cert. #673); SHS (Cert. #1081); RSA (Certs. #557 and #559); Triple-DES (Cert. #846) Other algorithms: DES; MD2; MD4; MD5; RC2; RC4; RSA (key wrapping; key establishment methodology provides between 112 bits and 256 bits of encryption strength; non-compliant less than 112 bits of encryption strength)

Windows Vista SP1

Validated Editions: Ultimate Edition

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Boot Manager (bootmgr)	6.0.6001.18000 and 6.0.6002.18005	978	FIPS approved algorithms: AES (Certs. #739 and #760); HMAC (Cert. #415); RSA (Cert. #354); SHS (Cert. #753)
Winload OS Loader (winload.exe)	6.0.6001.18000 , 6.0.6001.18027 , 6.0.6001.18606 , 6.0.6001.22125 , 6.0.6001.22861 , 6.0.6002.18005 , 6.0.6002.18411 and 6.0.6002.22596	979	FIPS approved algorithms: AES (Certs. #739 and #760); RSA (Cert. #354); SHS (Cert. #753) Other algorithms: MD5
Code Integrity (ci.dll)	6.0.6001.18000 , 6.0.6001.18023 , 6.0.6001.22120 , and 6.0.6002.18005	980	FIPS approved algorithms: RSA (Cert. #354); SHS (Cert. #753) Other algorithms: MD5

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Kernel Mode Security Support Provider Interface (ksecdd.sys)	6.0.6001.18709 , 6.0.6001.18272 , 6.0.6001.18796 , 6.0.6001.22202 , 6.0.6001.22450 , 6.0.6001.22987 , 6.0.6001.23069 , 6.0.6002.18005 , 6.0.6002.18051 , 6.0.6002.18541 , 6.0.6002.18643 , 6.0.6002.22152 , 6.0.6002.22742 , and 6.0.6002.22869	1000	<p>FIPS approved algorithms: AES (Certs. #739 and #756); ECDSA (Cert. #82); HMAC (Cert. #412); RNG (Cert. #435 and SP 800-90 AES-CTR, vendor-affirmed); RSA (Certs. #353 and #357); SHS (Cert. #753); Triple-DES (Cert. #656)#739 and); ECDSA (Cert.); HMAC (Cert.); RNG (Cert. and SP 800-90 AES-CTR, vendor-affirmed); RSA (Certs. and); SHS (Cert.); Triple-DES (Cert.)</p> <p>Other algorithms: AES (GCM and GMAC; non-compliant); DES; Diffie-Hellman (key agreement; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength); EC Diffie-Hellman (key agreement; key establishment methodology provides between 128 bits and 256 bits of encryption strength); MD2; MD4; MD5; HMAC MD5; RC2; RC4; RNG (SP 800-90 Dual-EC; non-compliant); RSA (key wrapping; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength)</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Cryptographic Primitives Library (bcrypt.dll)	6.0.6001.22202 , 6.0.6002.18005 , and 6.0.6002.22872	1001	<p>FIPS approved algorithms: AES (Certs. #739 and #756); DSA (Cert. #283); ECDSA (Cert. #82); HMAC (Cert. #412); RNG (Cert. #435 and SP 800-90, vendor affirmed); RSA (Certs. #353 and #357); SHS (Cert. #753); Triple-DES (Cert. #656)</p> <p>Other algorithms: AES (GCM and GMAC; non-compliant); DES; Diffie-Hellman (key agreement; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength); EC Diffie-Hellman (key agreement; key establishment methodology provides between 128 bits and 256 bits of encryption strength); MD2; MD4; MD5; RC2; RC4; RNG (SP 800-90 Dual-EC; non-compliant); RSA (key wrapping; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant provides less than 112 bits of encryption strength)</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Enhanced Cryptographic Provider (RSAENH)	6.0.6001.22202 and 6.0.6002.18005	1002	<p>FIPS approved algorithms: AES (Cert. #739); HMAC (Cert. #407); RNG (SP 800-90, vendor affirmed); RSA (Certs. #353 and #354); SHS (Cert. #753); Triple-DES (Cert. #656)</p> <p>Other algorithms: DES; MD2; MD4; MD5; RC2; RC4; RSA (key wrapping; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength)</p>
Enhanced DSS and Diffie-Hellman Cryptographic Provider (DSSENH)	6.0.6001.18000 and 6.0.6002.18005	1003	<p>FIPS approved algorithms: DSA (Cert. #281); RNG (Cert. #435); SHS (Cert. #753); Triple-DES (Cert. #656); Triple-DES MAC (Triple-DES Cert. #656, vendor affirmed)</p> <p>Other algorithms: DES; DES MAC; DES40; DES40 MAC; Diffie-Hellman (key agreement; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength); MD5; RC2; RC2 MAC; RC4</p>

Windows Vista

Validated Editions: Ultimate Edition

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
----------------------	-----------------------------------	--------------------	------------

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Enhanced Cryptographic Provider (RSAENH)	6.0.6000.16386	893	<p>FIPS approved algorithms: AES (Cert. #553); HMAC (Cert. #297); RNG (Cert. #321); RSA (Certs. #255 and #258); SHS (Cert. #618); Triple-DES (Cert. #549)</p> <p>Other algorithms: DES; MD2; MD4; MD5; RC2; RC4; RSA (key wrapping; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength)</p>
Enhanced DSS and Diffie-Hellman Cryptographic Provider (DSSENH)	6.0.6000.16386	894	<p>FIPS approved algorithms: DSA (Cert. #226); RNG (Cert. #321); SHS (Cert. #618); Triple-DES (Cert. #549); Triple-DES MAC (Triple-DES Cert. #549, vendor affirmed)</p> <p>Other algorithms: DES; DES MAC; DES40; DES40 MAC; Diffie-Hellman (key agreement; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength); MD5; RC2; RC2 MAC; RC4</p>
BitLocker™ Drive Encryption	6.0.6000.16386	947	<p>FIPS approved algorithms: AES (Cert. #715); HMAC (Cert. #386); SHS (Cert. #737)</p> <p>Other algorithms: Elephant Diffuser</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Kernel Mode Security Support Provider Interface (ksecdd.sys)	6.0.6000.16386 , 6.0.6000.16870 and 6.0.6000.21067	891	FIPS approved algorithms: AES (Cert. #553); ECDSA (Cert. #60); HMAC (Cert. #298); RNG (Cert. #321); RSA (Certs. #257 and #258); SHS (Cert. #618); Triple-DES (Cert. #549) Other algorithms: DES; Diffie-Hellman (key agreement; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength); EC Diffie-Hellman (key agreement; key establishment methodology provides 128 bits to 256 bits of encryption strength); MD2; MD4; MD5; RC2; RC4; HMAC MD5

Windows XP SP3

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Kernel Mode Cryptographic Module (FIPS.SYS)	5.1.2600.5512	997	FIPS approved algorithms: HMAC (Cert. #429); RNG (Cert. #449); SHS (Cert. #785); Triple-DES (Cert. #677); Triple-DES MAC (Triple-DES Cert. #677, vendor affirmed) Other algorithms: DES; MD5; HMAC MD5
Enhanced DSS and Diffie-Hellman Cryptographic Provider (DSSSENH)	5.1.2600.5507	990	FIPS approved algorithms: DSA (Cert. #292); RNG (Cert. #448); SHS (Cert. #784); Triple-DES (Cert. #676); Triple-DES MAC (Triple-DES Cert. #676, vendor affirmed) Other algorithms: DES; DES40; Diffie-Hellman (key agreement; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits); MD5; RC2; RC4

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Enhanced Cryptographic Provider (RSAENH)	5.1.2600.5507	989	<p>FIPS approved algorithms: AES (Cert. #781); HMAC (Cert. #428); RNG (Cert. #447); RSA (Cert. #371); SHS (Cert. #783); Triple-DES (Cert. #675); Triple-DES MAC (Triple-DES Cert. #675, vendor affirmed)</p> <p>Other algorithms: DES; MD2; MD4; MD5; HMAC MD5; RC2; RC4; RSA (key wrapping; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits)</p>

Windows XP SP2

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
DSS/Diffie-Hellman Enhanced Cryptographic Provider	5.1.2600.2133	240	<p>FIPS approved algorithms: Triple-DES (Cert. #16); DSA/SHA-1 (Cert. #29)</p> <p>Other algorithms: DES (Cert. #66); RC2; RC4; MD5; DES40; Diffie-Hellman (key agreement)</p>
Microsoft Enhanced Cryptographic Provider	5.1.2600.2161	238	<p>FIPS approved algorithms: Triple-DES (Cert. #81); AES (Cert. #33); SHA-1 (Cert. #83); RSA (PKCS#1, vendor affirmed); HMAC-SHA-1 (Cert. #83, vendor affirmed)</p> <p>Other algorithms: DES (Cert. #156); RC2; RC4; MD5</p>

Windows XP SP1

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
----------------------	-----------------------------------	--------------------	------------

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Microsoft Enhanced Cryptographic Provider	5.1.2600.1029	238	FIPS approved algorithms: Triple-DES (Cert. #81); AES (Cert. #33); SHA-1 (Cert. #83); RSA (PKCS#1, vendor affirmed); HMAC-SHA-1 (Cert. #83 , vendor affirmed) Other algorithms: DES (Cert. #156); RC2; RC4; MD5

Windows XP

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Kernel Mode Cryptographic Module	5.1.2600.0	241	FIPS approved algorithms: Triple-DES (Cert. #16); DSA/SHA-1 (Cert. #35); HMAC-SHA-1 (Cert. #35 , vendor affirmed) Other algorithms: DES (Cert. #89)

Windows 2000 SP3

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Kernel Mode Cryptographic Module (FIPS.SYS)	5.0.2195.1569	106	FIPS approved algorithms: Triple-DES (Cert. #16); SHA-1 (Certs. #35) Other algorithms: DES (Certs. #89)
Base DSS Cryptographic Provider, Base Cryptographic Provider, DSS/Diffie-Hellman Enhanced Cryptographic Provider, and Enhanced Cryptographic Provider	(Base DSS: 5.0.2195.3665 [SP3]) (Base: 5.0.2195.3839 [SP3]) (DSS/DH Enh: 5.0.2195.3665 [SP3]) (Enh: 5.0.2195.3839 [SP3])	103	FIPS approved algorithms: Triple-DES (Cert. #16); DSA/SHA-1 (Certs. #28 and #29); RSA (vendor affirmed) Other algorithms: DES (Certs. #65 , #66 , #67 and #68); Diffie-Hellman (key agreement); RC2; RC4; MD2; MD4; MD5

Windows 2000 SP2

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Kernel Mode Cryptographic Module (FIPS.SYS)	5.0.2195.1569	106	FIPS approved algorithms: Triple-DES (Cert. #16); SHA-1 (Certs. #35) Other algorithms: DES (Certs. #89)

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Base DSS Cryptographic Provider, Base Cryptographic Provider, DSS/Diffie-Hellman Enhanced Cryptographic Provider, and Enhanced Cryptographic Provider	(Base DSS: 5.0.2195.2228 [SP2]) (Base: 5.0.2195.2228 [SP2]) (DSS/DH Enh: 5.0.2195.2228 [SP2]) (Enh: 5.0.2195.2228 [SP2])	103	FIPS approved algorithms: Triple-DES (Cert. #16); DSA/SHA-1 (Certs. #28 and #29); RSA (vendor affirmed) Other algorithms: DES (Certs. #65, 66, 67 and 68); Diffie-Hellman (key agreement); RC2; RC4; MD2; MD4; MD5

Windows 2000 SP1

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Base DSS Cryptographic Provider, Base Cryptographic Provider, DSS/Diffie-Hellman Enhanced Cryptographic Provider, and Enhanced Cryptographic Provider	(Base DSS: 5.0.2150.1391 [SP1]) (Base: 5.0.2150.1391 [SP1]) (DSS/DH Enh: 5.0.2150.1391 [SP1]) (Enh: 5.0.2150.1391 [SP1])	103	FIPS approved algorithms: Triple-DES (Cert. #16); DSA/SHA-1 (Certs. #28 and #29); RSA (vendor affirmed) Other algorithms: DES (Certs. #65, 66, 67 and 68); Diffie-Hellman (key agreement); RC2; RC4; MD2; MD4; MD5

Windows 2000

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Base DSS Cryptographic Provider, Base Cryptographic Provider, DSS/Diffie-Hellman Enhanced Cryptographic Provider, and Enhanced Cryptographic Provider	5.0.2150.1	76	FIPS approved algorithms: Triple-DES (vendor affirmed); DSA/SHA-1 (Certs. #28 and 29); RSA (vendor affirmed) Other algorithms: DES (Certs. #65, 66, 67 and 68); RC2; RC4; MD2; MD4; MD5; Diffie-Hellman (key agreement)

Windows 95 and Windows 98

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Base DSS Cryptographic Provider, Base Cryptographic Provider, DSS/Diffie-Hellman Enhanced Cryptographic Provider, and Enhanced Cryptographic Provider	5.0.1877.6 and 5.0.1877.7	75	FIPS approved algorithms: Triple-DES (vendor affirmed); SHA-1 (Certs. #20 and 21); DSA/SHA-1 (Certs. #25 and 26); RSA (vendor- affirmed) Other algorithms: DES (Certs. #61 , 62 , 63 and 64); RC2; RC4; MD2; MD4; MD5; Diffie-Hellman (key agreement)

Windows NT 4.0

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Base Cryptographic Provider	5.0.1877.6 and 5.0.1877.7	68	FIPS approved algorithms: SHA-1 (Certs. #20 and 21); DSA/SHA- 1 (Certs. #25 and 26); RSA (vendor affirmed) Other algorithms: DES (Certs. #61 , 62 , 63 and 64); Triple-DES (allowed for US and Canadian Government use); RC2; RC4; MD2; MD4; MD5; Diffie-Hellman (key agreement)

Modules used by Windows Server

Windows Server 2019 (Version 1809)

Validated Editions: Standard, Datacenter

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Cryptographic Primitives Library	10.0.17763	#3197	See Security Policy and Certificate page for algorithm information
Kernel Mode Cryptographic Primitives Library	10.0.17763	#3196	See Security Policy and Certificate page for algorithm information
Code Integrity	10.0.17763	#3644	See Security Policy and Certificate page for algorithm information
Windows OS Loader	10.0.17763	#3615	See Security Policy and Certificate page for algorithm information

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Secure Kernel Code Integrity	10.0.17763	#3651	See Security Policy and Certificate page for algorithm information
BitLocker Dump Filter	10.0.17763	#3092	See Security Policy and Certificate page for algorithm information
Boot Manager	10.0.17763	#3089	See Security Policy and Certificate page for algorithm information
Virtual TPM	10.0.17763	#3690	See Security Policy and Certificate page for algorithm information

Windows Server (Version 1803)

Validated Editions: Standard, Datacenter

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Cryptographic Primitives Library	10.0.17134	#3197	See Security Policy and Certificate page for algorithm information
Kernel Mode Cryptographic Primitives Library	10.0.17134	#3196	See Security Policy and Certificate page for algorithm information
Code Integrity	10.0.17134	#3195	See Security Policy and Certificate page for algorithm information
Windows OS Loader	10.0.17134	#3480	See Security Policy and Certificate page for algorithm information
Secure Kernel Code Integrity	10.0.17134	#3096	See Security Policy and Certificate page for algorithm information
BitLocker Dump Filter	10.0.17134	#3092	See Security Policy and Certificate page for algorithm information
Boot Manager	10.0.17134	#3089	See Security Policy and Certificate page for algorithm information

Windows Server (Version 1709)

Validated Editions: Standard, Datacenter

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Cryptographic Primitives Library	10.0.16299	#3197	See Security Policy and Certificate page for algorithm information
Kernel Mode Cryptographic Primitives Library	10.0.16299	#3196	See Security Policy and Certificate page for algorithm information
Code Integrity	10.0.16299	#3195	See Security Policy and Certificate page for algorithm information
Windows OS Loader	10.0.16299	#3194	See Security Policy and Certificate page for algorithm information
Secure Kernel Code Integrity	10.0.16299	#3096	See Security Policy and Certificate page for algorithm information
BitLocker Dump Filter	10.0.16299	#3092	See Security Policy and Certificate page for algorithm information
Windows Resume	10.0.16299	#3091	See Security Policy and Certificate page for algorithm information
Boot Manager	10.0.16299	#3089	See Security Policy and Certificate page for algorithm information

Windows Server 2016

Validated Editions: Standard, Datacenter, Storage Server

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
----------------------	-----------------------------------	--------------------	------------

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Cryptographic Primitives Library (bcryptprimitives.dll and ncryptssp.dll)	10.0.14393	2937	<p>FIPS approved algorithms: AES (Cert. #4064); DRBG (Cert. #1217); DSA (Cert. #1098); ECDSA (Cert. #911); HMAC (Cert. #2651); KAS (Cert. #92); KBKDF (Cert. #101); KTS (AES Cert. #4062; key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); PBKDF (vendor affirmed); RSA (Certs. #2192, #2193, and #2195); SHS (Cert. #3347); Triple-DES (Cert. #2227)</p> <p>Other algorithms: HMAC-MD5; MD5; DES; Legacy CAPI KDF; MD2; MD4; RC2; RC4; RSA (encrypt/decrypt)</p>
Kernel Mode Cryptographic Primitives Library (cng.sys)	10.0.14393	2936	<p>FIPS approved algorithms: AES (Cert. #4064); DRBG (Cert. #1217); DSA (Cert. #1098); ECDSA (Cert. #911); HMAC (Cert. #2651); KAS (Cert. #92); KBKDF (Cert. #101); KTS (AES Cert. #4062; key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); PBKDF (vendor affirmed); RSA (Certs. #2192, #2193, and #2195); SHS (Cert. #3347); Triple-DES (Cert. #2227)</p> <p>Other algorithms: HMAC-MD5; MD5; NDRNG; DES; Legacy CAPI KDF; MD2; MD4; RC2; RC4; RSA (encrypt/decrypt)</p>
Boot Manager	10.0.14393	2931	<p>FIPS approved algorithms: AES (Certs. #4061 and #4064); HMAC (Cert. #2651); PBKDF (vendor affirmed); RSA (Cert. #2193); SHS (Cert. #3347)</p> <p>Other algorithms: MD5; PBKDF (non-compliant); VMK KDF</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
BitLocker® Windows OS Loader (winload)	10.0.14393	2932	FIPS approved algorithms: AES (Certs. #4061 and #4064); RSA (Cert. #2193); SHS (Cert. #3347) Other algorithms: NDRNG; MD5
BitLocker® Windows Resume (winresume)	10.0.14393	2933	FIPS approved algorithms: AES (Certs. #4061 and #4064); RSA (Cert. #2193); SHS (Cert. #3347) Other algorithms: MD5
BitLocker® Dump Filter (dumpfve.sys)	10.0.14393	2934	FIPS approved algorithms: AES (Certs. #4061 and #4064)
Code Integrity (ci.dll)	10.0.14393	2935	FIPS approved algorithms: RSA (Cert. #2193); SHS (Cert. #3347) Other algorithms: AES (non-compliant); MD5
Secure Kernel Code Integrity (skci.dll)	10.0.14393	2938	FIPS approved algorithms: RSA (Certs. #2193); SHS (Certs. #3347) Other algorithms: MD5

Windows Server 2012 R2

Validated Editions: Server, Storage Server,

StorSimple 8000 Series, Azure StorSimple Virtual Array Windows Server 2012 R2

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
----------------------	-----------------------------------	--------------------	------------

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Cryptographic Primitives Library (bcryptprimitives.dll and ncryptssp.dll)	6.3.9600 6.3.9600.17031	2357	<p>FIPS approved algorithms: AES (Cert. #2832); DRBG (Certs. #489); DSA (Cert. #855); ECDSA (Cert. #505); HMAC (Cert. #1773); KAS (Cert. #47); KBKDF (Cert. #30); PBKDF (vendor affirmed); RSA (Certs. #1487, #1493, and #1519); SHS (Cert. #2373); Triple-DES (Cert. #1692)</p> <p>Other algorithms: AES (Cert. #2832, key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); AES-GCM encryption (non-compliant); DES; HMAC MD5; Legacy CAPI KDF; MD2; MD4; MD5; NDRNG; RC2; RC4; RSA (encrypt/decrypt)</p>
Kernel Mode Cryptographic Primitives Library (cng.sys)	6.3.9600 6.3.9600.17042	2356	<p>FIPS approved algorithms: AES (Cert. #2832); DRBG (Certs. #489); ECDSA (Cert. #505); HMAC (Cert. #1773); KAS (Cert. #47); KBKDF (Cert. #30); PBKDF (vendor affirmed); RSA (Certs. #1487, #1493, and #1519); SHS (Cert. #2373); Triple-DES (Cert. #1692)</p> <p>Other algorithms: AES (Cert. #2832, key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); AES-GCM encryption (non-compliant); DES; HMAC MD5; Legacy CAPI KDF; MD2; MD4; MD5; NDRNG; RC2; RC4; RSA (encrypt/decrypt)</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Boot Manager	6.3.9600 6.3.9600.17031	2351	FIPS approved algorithms: AES (Cert. #2832); HMAC (Cert. #1773); PBKDF (vendor affirmed); RSA (Cert. #1494); SHS (Certs. #2373 and #2396) Other algorithms: MD5; KDF (non-compliant); PBKDF (non-compliant)
BitLocker® Windows OS Loader (winload)	6.3.9600 6.3.9600.17031	2352	FIPS approved algorithms: AES (Cert. #2832); RSA (Cert. #1494); SHS (Cert. #2396) Other algorithms: MD5; NDRNG
BitLocker® Windows Resume (winresume) ^[16]	6.3.9600 6.3.9600.17031	2353	FIPS approved algorithms: AES (Cert. #2832); RSA (Cert. #1494); SHS (Certs. #2373 and #2396) Other algorithms: MD5
BitLocker® Dump Filter (dumpfve.sys) ^[17]	6.3.9600 6.3.9600.17031	2354	FIPS approved algorithms: AES (Cert. #2832) Other algorithms: N/A
Code Integrity (ci.dll)	6.3.9600 6.3.9600.17031	2355	FIPS approved algorithms: RSA (Cert. #1494); SHS (Cert. #2373) Other algorithms: MD5

^[16] Doesn't apply to Azure StorSimple Virtual Array Windows Server 2012 R2

^[17] Doesn't apply to Azure StorSimple Virtual Array Windows Server 2012 R2

Windows Server 2012

Validated Editions: Server, Storage Server

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
----------------------	-----------------------------------	--------------------	------------

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Cryptographic Primitives Library (BCRYPTPRIMITIVES.DLL)	6.2.9200	1892	<p>FIPS approved algorithms: AES (Certs. #2197 and #2216); DRBG (Certs. #258); DSA (Cert. #687); ECDSA (Cert. #341); HMAC (Cert. #1345); KAS (Cert. #36); KBKDF (Cert. #3); PBKDF (vendor affirmed); RSA (Certs. #1133 and #1134); SHS (Cert. #1903); Triple-DES (Cert. #1387)</p> <p>Other algorithms: AES (Cert. #2197, key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); DES; Legacy CAPI KDF; MD2; MD4; MD5; HMAC MD5; RC2; RC4; RSA (encrypt/decrypt)#687); ECDSA (Cert.); HMAC (Cert. #); KAS (Cert.); KBKDF (Cert.); PBKDF (vendor affirmed); RSA (Certs. and); SHS (Cert.); Triple-DES (Cert.)</p> <p>Other algorithms: AES (Certificate, key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); DES; Legacy CAPI KDF; MD2; MD4; MD5; HMAC MD5; RC2; RC4; RSA (encrypt/decrypt)</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Kernel Mode Cryptographic Primitives Library (cng.sys)	6.2.9200	1891	<p>FIPS approved algorithms: AES (Certs. #2197 and #2216); DRBG (Certs. #258 and #259); ECDSA (Cert. #341); HMAC (Cert. #1345); KAS (Cert. #36); KBKDF (Cert. #3); PBKDF (vendor affirmed); RNG (Cert. #1110); RSA (Certs. #1133 and #1134); SHS (Cert. #1903); Triple-DES (Cert. #1387)</p> <p>Other algorithms: AES (Cert. #2197, key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); DES; Legacy CAPI KDF; MD2; MD4; MD5; HMAC MD5; RC2; RC4; RSA (encrypt/decrypt)#1110); RSA (Certs. and); SHS (Cert.); Triple-DES (Cert.)</p> <p>Other algorithms: AES (Certificate, key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); DES; Legacy CAPI KDF; MD2; MD4; MD5; HMAC MD5; RC2; RC4; RSA (encrypt/decrypt)</p>
Boot Manager	6.2.9200	1895	<p>FIPS approved algorithms: AES (Certs. #2196 and #2198); HMAC (Cert. #1347); RSA (Cert. #1132); SHS (Cert. #1903)</p> <p>Other algorithms: MD5</p>
BitLocker® Windows OS Loader (WINLOAD)	6.2.9200	1896	<p>FIPS approved algorithms: AES (Certs. #2196 and #2198); RSA (Cert. #1132); SHS (Cert. #1903)</p> <p>Other algorithms: AES (Cert. #2197; non-compliant); MD5; Non-Approved RNG</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
BitLocker® Windows Resume (WINRESUME)	6.2.9200	1898	FIPS approved algorithms: AES (Certs. #2196 and #2198); RSA (Cert. #1132); SHS (Cert. #1903) Other algorithms: MD5
BitLocker® Dump Filter (DUMPFVE.SYS)	6.2.9200	1899	FIPS approved algorithms: AES (Certs. #2196 and #2198) Other algorithms: N/A
Code Integrity (CI.DLL)	6.2.9200	1897	FIPS approved algorithms: RSA (Cert. #1132); SHS (Cert. #1903) Other algorithms: MD5
Enhanced DSS and Diffie-Hellman Cryptographic Provider (DSSENH.DLL)	6.2.9200	1893	FIPS approved algorithms: DSA (Cert. #686); SHS (Cert. #1902); Triple-DES (Cert. #1386); Triple-DES MAC (Triple-DES Cert. #1386 , vendor affirmed) Other algorithms: DES; DES MAC; DES40; DES40 MAC; Diffie-Hellman; MD5; RC2; RC2 MAC; RC4; Triple-DES (Cert. #1386 , key wrapping; key establishment methodology provides 112 bits of encryption strength; non-compliant less than 112 bits of encryption strength)

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Enhanced Cryptographic Provider (RSAENH.DLL)	6.2.9200	1894	<p>FIPS approved algorithms: AES (Cert. #2196); HMAC (Cert. #1346); RSA (Cert. #1132); SHS (Cert. #1902); Triple-DES (Cert. #1386)</p> <p>Other algorithms: AES (Cert. #2196, key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); DES; MD2; MD4; MD5; RC2; RC4; RSA (key wrapping; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength); Triple-DES (Cert. #1386, key wrapping; key establishment methodology provides 112 bits of encryption strength; non-compliant less than 112 bits of encryption strength)</p>

Windows Server 2008 R2

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Boot Manager (bootmgr)	6.1.7600.16385 or 6.1.7601.17514	1321	<p>FIPS approved algorithms: AES (Certs. #1168 and #1177); HMAC (Cert. #675); RSA (Cert. #568); SHS (Cert. #1081)</p> <p>Other algorithms: MD5</p>
Winload OS Loader (winload.exe)	6.1.7600.16385 , 6.1.7600.16757 , 6.1.7600.20897 , 6.1.7600.20916 , 6.1.7601.17514 , 6.1.7601.17556 , 6.1.7601.21655 and 6.1.7601.21675	1333	<p>FIPS approved algorithms: AES (Certs. #1168 and #1177); RSA (Cert. #568); SHS (Cert. #1081)</p> <p>Other algorithms: MD5</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Code Integrity (ci.dll)	6.1.7600.16385 , 6.1.7600.17122 , 6.1.7600.21320 , 6.1.7601.17514 , 6.1.7601.17950 and 6.1.7601.22108	1334	FIPS approved algorithms: RSA (Cert. #568); SHS (Cert. #1081) Other algorithms: MD5
Kernel Mode Cryptographic Primitives Library (cng.sys)	6.1.7600.16385 , 6.1.7600.16915 , 6.1.7600.21092 , 6.1.7601.17514 , 6.1.7601.17919 , 6.1.7601.17725 , 6.1.7601.21861 and 6.1.7601.22076	1335	FIPS approved algorithms: AES (Certs. #1168 and #1177); AES GCM (Cert. #1168 , vendor-affirmed); AES GMAC (Cert. #1168 , vendor-affirmed); DRBG (Certs. #23 and #27); ECDSA (Cert. #142); HMAC (Cert. #686); KAS (SP 800-56A, vendor affirmed, key agreement; key establishment methodology provides between 80 bits and 256 bits of encryption strength); RNG (Cert. #649); RSA (Certs. #559 and #567); SHS (Cert. #1081); Triple-DES (Cert. #846) Other algorithms: AES (Cert. #1168 , key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); DES; Diffie-Hellman (key agreement; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength); MD2; MD4; MD5; HMAC MD5; RC2; RC4

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Cryptographic Primitives Library (bcryptprimitives.dll)	66.1.7600.16385 or 6.1.7601.17514	1336	<p>FIPS approved algorithms: AES (Certs. #1168 and #1177); AES GCM (Cert. #1168, vendor-affirmed); AES GMAC (Cert. #1168, vendor-affirmed); DRBG (Certs. #23 and #27); DSA (Cert. #391); ECDSA (Cert. #142); HMAC (Cert. #686); KAS (SP 800-56A, vendor affirmed, key agreement; key establishment methodology provides between 80 bits and 256 bits of encryption strength); RNG (Cert. #649); RSA (Certs. #559 and #567); SHS (Cert. #1081); Triple-DES (Cert. #846)</p> <p>Other algorithms: AES (Cert. #1168, key wrapping; key establishment methodology provides between 128 bits and 256 bits of encryption strength); DES; HMAC MD5; MD2; MD4; MD5; RC2; RC4</p>
Enhanced Cryptographic Provider (RSAENH)	6.1.7600.16385	1337	<p>FIPS approved algorithms: AES (Cert. #1168); DRBG (Cert. #23); HMAC (Cert. #687); SHS (Cert. #1081); RSA (Certs. #559 and #568); Triple-DES (Cert. #846)</p> <p>Other algorithms: DES; MD2; MD4; MD5; RC2; RC4; RSA (key wrapping; key establishment methodology provides between 112 bits and 256 bits of encryption strength; non-compliant less than 112 bits of encryption strength)</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Enhanced DSS and Diffie-Hellman Cryptographic Provider (DSSSENH)	6.1.7600.16385	1338	FIPS approved algorithms: DSA (Cert. #390); RNG (Cert. #649); SHS (Cert. #1081); Triple-DES (Cert. #846); Triple-DES MAC (Triple-DES Cert. #846 , vendor affirmed) Other algorithms: DES; DES MAC; DES40; DES40 MAC; Diffie-Hellman; MD5; RC2; RC2 MAC; RC4
BitLocker™ Drive Encryption	6.1.7600.16385 , 6.1.7600.16429 , 6.1.7600.16757 , 6.1.7600.20536 , 6.1.7600.20873 , 6.1.7600.20897 , 6.1.7600.20916 , 6.1.7601.17514 , 6.1.7601.17556 , 6.1.7601.21634 , 6.1.7601.21655 or 6.1.7601.21675	1339	FIPS approved algorithms: AES (Certs. #1168 and #1177); HMAC (Cert. #675); SHS (Cert. #1081) Other algorithms: Elephant Diffuser

Windows Server 2008

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Boot Manager (bootmgr)	6.0.6001.18000 , 6.0.6002.18005 and 6.0.6002.22497	1004	FIPS approved algorithms: AES (Certs. #739 and #760); HMAC (Cert. #415); RSA (Cert. #355); SHS (Cert. #753) Other algorithms: N/A
Winload OS Loader (winload.exe)	6.0.6001.18000 , 6.0.6001.18606 , 6.0.6001.22861 , 6.0.6002.18005 , 6.0.6002.18411 , 6.0.6002.22497 and 6.0.6002.22596	1005	FIPS approved algorithms: AES (Certs. #739 and #760); RSA (Cert. #355); SHS (Cert. #753) Other algorithms: MD5
Code Integrity (ci.dll)	6.0.6001.18000 and 6.0.6002.18005	1006	FIPS approved algorithms: RSA (Cert. #355); SHS (Cert. #753) Other algorithms: MD5
Kernel Mode Security Support Provider Interface (ksecdd.sys)	6.0.6001.18709 , 6.0.6001.18272 , 6.0.6001.18796 , 6.0.6001.22202 , 6.0.6001.22450 , 6.0.6001.22987 ,	1007	FIPS approved algorithms: AES (Certs. #739 and #757); ECDSA (Cert. #83); HMAC (Cert. #413); RNG (Cert. #435 and SP800-90 AES-CTR, vendor affirmed);

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
	<p>6.0.6001.23069, 6.0.6002.18005, 6.0.6002.18051, 6.0.6002.18541, 6.0.6002.18643, 6.0.6002.22152, 6.0.6002.22742 and 6.0.6002.22869</p>		<p>RSA (Certs. #353 and #358); SHS (Cert. #753); Triple-DES (Cert. #656)</p> <p>Other algorithms: AES (GCM and GMAC; non-compliant); DES; Diffie-Hellman (key agreement; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength); EC Diffie-Hellman (key agreement; key establishment methodology provides between 128 bits and 256 bits of encryption strength); MD2; MD4; MD5; HMAC MD5; RC2; RC4; RNG (SP 800-90 Dual-EC; non-compliant); RSA (key wrapping: key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength)#83); HMAC (Cert.); RNG (Cert. and SP800-90 AES-CTR, vendor affirmed); RSA (Certs. and); SHS (Cert.); Triple-DES (Cert.)</p> <p>Other algorithms: AES (GCM and GMAC; non-compliant); DES; Diffie-Hellman (key agreement; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength); EC Diffie-Hellman (key agreement; key establishment methodology provides between 128 bits and 256 bits of encryption strength); MD2; MD4; MD5; HMAC MD5; RC2; RC4; RNG (SP</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
			<p>800-90 Dual-EC; non-compliant); RSA (key wrapping; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength)</p>
Cryptographic Primitives Library (bcrypt.dll)	6.0.6001.22202 , 6.0.6002.18005 and 6.0.6002.22872	1008	<p>FIPS approved algorithms: AES (Certs. #739 and #757); DSA (Cert. #284); ECDSA (Cert. #83); HMAC (Cert. #413); RNG (Cert. #435 and SP800-90, vendor affirmed); RSA (Certs. #353 and #358); SHS (Cert. #753); Triple-DES (Cert. #656)</p> <p>Other algorithms: AES (GCM and GMAC; non-compliant); DES; Diffie-Hellman (key agreement; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength); EC Diffie-Hellman (key agreement; key establishment methodology provides between 128 bits and 256 bits of encryption strength); MD2; MD4; MD5; RC2; RC4; RNG (SP 800-90 Dual-EC; non-compliant); RSA (key wrapping; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant provides less than 112 bits of encryption strength)</p>

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Enhanced DSS and Diffie-Hellman Cryptographic Provider (DSSENH)	6.0.6001.18000 and 6.0.6002.18005	1009	<p>FIPS approved algorithms: DSA (Cert. #282); RNG (Cert. #435); SHS (Cert. #753); Triple-DES (Cert. #656); Triple-DES MAC (Triple-DES Cert. #656, vendor affirmed)</p> <p>Other algorithms: DES; DES MAC; DES40; DES40 MAC; Diffie-Hellman (key agreement; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength); MD5; RC2; RC2 MAC; RC4</p>
Enhanced Cryptographic Provider (RSAENH)	6.0.6001.22202 and 6.0.6002.18005	1010	<p>FIPS approved algorithms: AES (Cert. #739); HMAC (Cert. #408); RNG (SP 800-90, vendor affirmed); RSA (Certs. #353 and #355); SHS (Cert. #753); Triple-DES (Cert. #656)</p> <p>Other algorithms: DES; MD2; MD4; MD5; RC2; RC4; RSA (key wrapping; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength)</p>

Windows Server 2003 SP2

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
----------------------	-----------------------------------	--------------------	------------

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Enhanced DSS and Diffie-Hellman Cryptographic Provider (DSSENH)	5.2.3790.3959	875	FIPS approved algorithms: DSA (Cert. #221); RNG (Cert. #314); RSA (Cert. #245); SHS (Cert. #611); Triple-DES (Cert. #543) Other algorithms: DES; DES40; Diffie-Hellman (key agreement; key establishment methodology provides between 112 bits and 150 bits of encryption strength; non-compliant less than 112 bits of encryption strength); MD5; RC2; RC4
Kernel Mode Cryptographic Module (FIPS.SYS)	5.2.3790.3959	869	FIPS approved algorithms: HMAC (Cert. #287); RNG (Cert. #313); SHS (Cert. #610); Triple-DES (Cert. #542) Other algorithms: DES; HMAC-MD5
Enhanced Cryptographic Provider (RSAENH)	5.2.3790.3959	868	FIPS approved algorithms: AES (Cert. #548); HMAC (Cert. #289); RNG (Cert. #316); RSA (Cert. #245); SHS (Cert. #613); Triple-DES (Cert. #544) Other algorithms: DES; RC2; RC4; MD2; MD4; MD5; RSA (key wrapping; key establishment methodology provides between 112 bits and 256 bits of encryption strength; non-compliant less than 112 bits of encryption strength)

Windows Server 2003 SP1

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
----------------------	-----------------------------------	--------------------	------------

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Kernel Mode Cryptographic Module (FIPS.SYS)	5.2.3790.1830 [SP1]	405	FIPS approved algorithms: Triple-DES (Certs. #201 [1] and #370 [1]); SHS (Certs. #177 [1] and #371 [2]) Other algorithms: DES (Cert. #230 [1]); HMAC-MD5; HMAC-SHA-1 (non-compliant) [1] x86 [2] SP1 x86, x64, IA64
Enhanced Cryptographic Provider (RSAENH)	5.2.3790.1830 [Service Pack 1]	382	FIPS approved algorithms: Triple-DES (Cert. #192 [1] and #365 [2]); AES (Certs. #80 [1] and #290 [2]); SHS (Cert. #176 [1] and #364 [2]); HMAC (Cert. #176 , vendor affirmed[1] and #99 [2]); RSA (PKCS#1, vendor affirmed[1] and #81 [2]) Other algorithms: DES (Cert. #226 [1]); SHA-256[1]; SHA-384[1]; SHA-512[1]; RC2; RC4; MD2; MD4; MD5 [1] x86 [2] SP1 x86, x64, IA64
Enhanced DSS and Diffie-Hellman Cryptographic Provider (DSSENH)	5.2.3790.1830 [Service Pack 1]	381	FIPS approved algorithms: Triple-DES (Certs. #199 [1] and #381 [2]); SHA-1 (Certs. #181 [1] and #385 [2]); DSA (Certs. #95 [1] and #146 [2]); RSA (Cert. #81) Other algorithms: DES (Cert. #229 [1]); Diffie-Hellman (key agreement); RC2; RC4; MD5; DES 40 [1] x86 [2] SP1 x86, x64, IA64

Windows Server 2003

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
----------------------	-----------------------------------	--------------------	------------

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Kernel Mode Cryptographic Module (FIPS.SYS)	5.2.3790.0	405	<p>FIPS approved algorithms: Triple-DES (Certs. #201[1] and #370[1]); SHS (Certs. #177[1] and #371[2])</p> <p>Other algorithms: DES (Cert. #230[1]); HMAC-MD5; HMAC-SHA-1 (non-compliant)</p> <p>[1] x86</p> <p>[2] SP1 x86, x64, IA64</p>
Enhanced Cryptographic Provider (RSAENH)	5.2.3790.0	382	<p>FIPS approved algorithms: Triple-DES (Cert. #192[1] and #365[2]); AES (Certs. #80[1] and #290[2]); SHS (Cert. #176[1] and #364[2]); HMAC (Cert. #176, vendor affirmed[1] and #99[2]); RSA (PKCS#1, vendor affirmed[1] and #81[2])</p> <p>Other algorithms: DES (Cert. #226[1]); SHA-256[1]; SHA-384[1]; SHA-512[1]; RC2; RC4; MD2; MD4; MD5</p> <p>[1] x86</p> <p>[2] SP1 x86, x64, IA64</p>
Enhanced DSS and Diffie-Hellman Cryptographic Provider (DSENH)	5.2.3790.0	381	<p>FIPS approved algorithms: Triple-DES (Certs. #199[1] and #381[2]); SHA-1 (Certs. #181[1] and #385[2]); DSA (Certs. #95[1] and #146[2]); RSA (Cert. #81)</p> <p>Other algorithms: DES (Cert. #229[1]); Diffie-Hellman (key agreement); RC2; RC4; MD5; DES 40</p> <p>[1] x86</p> <p>[2] SP1 x86, x64, IA64</p>

Other Products

Windows Embedded Compact 7 and Windows Embedded Compact 8

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
----------------------	-----------------------------------	--------------------	------------

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Enhanced Cryptographic Provider	7.00.2872 [1] and 8.00.6246 [2]	2957	<p>FIPS approved algorithms:</p> <p>AES (Certs.#4433and#4434);</p> <p>CKG (vendor affirmed);</p> <p>DRBG (Certs.#1432and#1433);</p> <p>HMAC (Certs.#2946and#2945);</p> <p>RSA (Certs.#2414and#2415);</p> <p>SHS (Certs.#3651and#3652);</p> <p>Triple-DES (Certs.#2383and#2384)</p> <p>Allowed algorithms: HMAC-MD5, MD5, NDRNG</p>
Cryptographic Primitives Library (bcrypt.dll)	7.00.2872 [1] and 8.00.6246 [2]	2956	<p>FIPS approved algorithms:</p> <p>AES (Certs.#4430and#4431);</p> <p>CKG (vendor affirmed); CVL (Certs.#1139and#1140);</p> <p>DRBG (Certs.#1429and#1430);</p> <p>DSA (Certs.#1187and#1188);</p> <p>ECDSA (Certs.#1072and#1073);</p> <p>HMAC (Certs.#2942and#2943);</p> <p>KAS (Certs.#114and#115);</p> <p>RSA (Certs.#2411and#2412);</p> <p>SHS (Certs.#3648and#3649);</p> <p>Triple-DES (Certs.#2381and#2382)</p> <p>Allowed algorithms: MD5, NDRNG, RSA (key wrapping; key establishment methodology provides between 112 bits and 150 bits of encryption strength</p>

Windows CE 6.0 and Windows Embedded Compact 7

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
----------------------	-----------------------------------	--------------------	------------

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Enhanced Cryptographic Provider	6.00.1937 [1] and 7.00.1687 [2]	825	FIPS approved algorithms: AES (Certs. #516 [1] and #2024 [2]); HMAC (Certs. #267 [1] and #1227 [2]); RNG (Certs. #292 [1] and #1060 [2]); RSA (Cert. #230 [1] and #1052 [2]); SHS (Certs. #589 [1] and #1774 [2]); Triple-DES (Certs. #526 [1] and #1308 [2]) Other algorithms: MD5; HMAC-MD5; RC2; RC4; DES

Outlook Cryptographic Provider

CRYPTOGRAPHIC MODULE	VERSION (LINK TO SECURITY POLICY)	FIPS CERTIFICATE #	ALGORITHMS
Outlook Cryptographic Provider (EXCHCSP)	SR-1A (3821)	110	FIPS approved algorithms: Triple-DES (Cert. #18); SHA-1 (Certs. #32); RSA (vendor affirmed) Other algorithms: DES (Certs. #91); DES MAC; RC2; MD2; MD5

Cryptographic Algorithms

The following tables are organized by cryptographic algorithms with their modes, states, and key sizes. For each algorithm implementation (operating system / platform), there is a link to the Cryptographic Algorithm Validation Program (CAVP) issued certificate.

Advanced Encryption Standard (AES)

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>AES-CBC:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Key Lengths: 128, 192, 256 (bits) <p>AES-CFB128:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Key Lengths: 128, 192, 256 (bits) <p>AES-CTR:</p> <p>Counter Source: Internal</p> <ul style="list-style-type: none"> • Key Lengths: 128, 192, 256 (bits) <p>AES-OFB:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Key Lengths: 128, 192, 256 (bits) 	<p>Microsoft Surface Hub Virtual TPM Implementations #4904 Version 10.0.15063.674</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>AES-CBC:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Key Lengths: 128, 192, 256 (bits) <p>AES-CFB128:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Key Lengths: 128, 192, 256 (bits) <p>AES-CTR:</p> <p>Counter Source: Internal</p> <ul style="list-style-type: none"> • Key Lengths: 128, 192, 256 (bits) <p>AES-OFB:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Key Lengths: 128, 192, 256 (bits) 	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); Virtual TPM Implementations #4903</p> <p>Version 10.0.16299</p>
<p>AES-CBC:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Key Lengths: 128, 192, 256 (bits) <p>AES-CCM:</p> <ul style="list-style-type: none"> • Key Lengths: 128, 192, 256 (bits) • Tag Lengths: 32, 48, 64, 80, 96, 112, 128 (bits) • IV Lengths: 56, 64, 72, 80, 88, 96, 104 (bits) • Plain Text Length: 0-32 • Additional authenticated data length: 0-65536 <p>AES-CFB128:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Key Lengths: 128, 192, 256 (bits) <p>AES-CFB8:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Key Lengths: 128, 192, 256 (bits) <p>AES-CMAC:</p> <ul style="list-style-type: none"> • Generation: <p>AES-128:</p> <ul style="list-style-type: none"> • Block Sizes: Full, Partial • Message Length: 0-65536 • Tag Length: 16-16 <p>AES-192:</p> <ul style="list-style-type: none"> • Block Sizes: Full, Partial • Message Length: 0-65536 • Tag Length: 16-16 <p>AES-256:</p> <ul style="list-style-type: none"> • Block Sizes: Full, Partial • Message Length: 0-65536 • Tag Length: 16-16 <p>Verification:</p> <p>AES-128:</p> <ul style="list-style-type: none"> • Block Sizes: Full, Partial • Message Length: 0-65536 • Tag Length: 16-16 <p>AES-192:</p>	<p>Microsoft Surface Hub SymCrypt Cryptographic Implementations #4902</p> <p>Version 10.0.15063.674</p>

<ul style="list-style-type: none"> Block Sizes: Full, Partial MODES / STATES / KEY SIZES Message Length: 0-65536 	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<ul style="list-style-type: none"> Tag Length: 16-16 <p>AES-256:</p> <ul style="list-style-type: none"> Block Sizes: Full, Partial Message Length: 0-65536 Tag Length: 16-16 <p>AES-CTR:</p> <p>Counter Source: Internal</p> <ul style="list-style-type: none"> Key Lengths: 128, 192, 256 (bits) <p>AES-ECB:</p> <ul style="list-style-type: none"> Modes: Decrypt, Encrypt Key Lengths: 128, 192, 256 (bits) <p>AES-GCM:</p> <ul style="list-style-type: none"> Modes: Decrypt, Encrypt Key Lengths: 128, 192, 256 (bits) Tag Lengths: 96, 104, 112, 120, 128 (bits) Plain Text Lengths: 0, 8, 1016, 1024 (bits) Additional authenticated data lengths: 0, 8, 1016, 1024 (bits) 96 bit IV supported <p>AES-XTS:</p> <ul style="list-style-type: none"> Key Size: 128: Modes: Decrypt, Encrypt Block Sizes: Full Key Size: 256: Modes: Decrypt, Encrypt Block Sizes: Full 	
<p>AES-CBC:</p> <ul style="list-style-type: none"> Modes: Decrypt, Encrypt Key Lengths: 128, 192, 256 (bits) <p>AES-CCM:</p> <ul style="list-style-type: none"> Key Lengths: 128, 192, 256 (bits) Tag Lengths: 32, 48, 64, 80, 96, 112, 128 (bits) IV Lengths: 56, 64, 72, 80, 88, 96, 104 (bits) Plain Text Length: 0-32 Additional authenticated data length: 0-65536 <p>AES-CFB128:</p> <ul style="list-style-type: none"> Modes: Decrypt, Encrypt Key Lengths: 128, 192, 256 (bits) <p>AES-CFB8:</p> <ul style="list-style-type: none"> Modes: Decrypt, Encrypt Key Lengths: 128, 192, 256 (bits) <p>AES-CMAC:</p> <ul style="list-style-type: none"> Generation: <p>AES-128:</p> <ul style="list-style-type: none"> Block Sizes: Full, Partial Message Length: 0-65536 Tag Length: 16-16 <p>AES-192:</p> <ul style="list-style-type: none"> Block Sizes: Full, Partial Message Length: 0-65536 Tag Length: 16-16 	<p>Windows 10 Mobile (version 1709) SymCrypt Cryptographic Implementations #4901 Version 10.0.15254</p>

AES-256: MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<ul style="list-style-type: none"> • Block Sizes: Full, Partial • Message Length: 0-65536 • Tag Length: 16-16 • Verification: <p>AES-128:</p> <ul style="list-style-type: none"> • Block Sizes: Full, Partial • Message Length: 0-65536 • Tag Length: 16-16 <p>AES-192:</p> <ul style="list-style-type: none"> • Block Sizes: Full, Partial • Message Length: 0-65536 • Tag Length: 16-16 <p>AES-256:</p> <ul style="list-style-type: none"> • Block Sizes: Full, Partial • Message Length: 0-65536 • Tag Length: 16-16 <p>AES-CTR:</p> <p>Counter Source: Internal</p> <ul style="list-style-type: none"> • Key Lengths: 128, 192, 256 (bits) <p>AES-ECB:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Key Lengths: 128, 192, 256 (bits) <p>AES-GCM:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Key Lengths: 128, 192, 256 (bits) • Tag Lengths: 96, 104, 112, 120, 128 (bits) • Plain Text Lengths: 0, 8, 1016, 1024 (bits) • Additional authenticated data lengths: 0, 8, 1016, 1024 (bits),96 bit IV supported <p>AES-XTS:</p> <ul style="list-style-type: none"> • Key Size: 128: • Modes: Decrypt, Encrypt • Block Sizes: Full • Key Size: 256: • Modes: Decrypt, Encrypt • Block Sizes: Full 	
<p>AES-CBC:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Key Lengths: 128, 192, 256 (bits) <p>AES-CCM:</p> <ul style="list-style-type: none"> • Key Lengths: 128, 192, 256 (bits) • Tag Lengths: 32, 48, 64, 80, 96, 112, 128 (bits) • IV Lengths: 56, 64, 72, 80, 88, 96, 104 (bits) • Plain Text Length: 0-32 • Additional authenticated data length: 0-65536 <p>AES-CFB128:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Key Lengths: 128, 192, 256 (bits) <p>AES-CFB8:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Key Lengths: 128, 192, 256 (bits) <p>AES-CMAC:</p> <ul style="list-style-type: none"> • Generation: 	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); SymCrypt Cryptographic Implementations #4897</p> <p>Version 10.0.16299</p>

AES-128: MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<ul style="list-style-type: none"> • Block Sizes: Full, Partial • Message Length: 0-65536 • Tag Length: 16-16 <p>AES-192:</p> <ul style="list-style-type: none"> • Block Sizes: Full, Partial • Message Length: 0-65536 • Tag Length: 16-16 <p>AES-256:</p> <ul style="list-style-type: none"> • Block Sizes: Full, Partial • Message Length: 0-65536 • Tag Length: 16-16 <p>Verification:</p> <p>AES-128:</p> <ul style="list-style-type: none"> • Block Sizes: Full, Partial • Message Length: 0-65536 • Tag Length: 16-16 <p>AES-192:</p> <ul style="list-style-type: none"> • Block Sizes: Full, Partial • Message Length: 0-65536 • Tag Length: 16-16 <p>AES-256:</p> <ul style="list-style-type: none"> • Block Sizes: Full, Partial • Message Length: 0-65536 • Tag Length: 16-16 <p>AES-CTR:</p> <p>Counter Source: Internal</p> <ul style="list-style-type: none"> • Key Lengths: 128, 192, 256 (bits) <p>AES-ECB:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Key Lengths: 128, 192, 256 (bits) <p>AES-GCM:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • IV Generation: External • Key Lengths: 128, 192, 256 (bits) • Tag Lengths: 96, 104, 112, 120, 128 (bits) • Plain Text Lengths: 0, 8, 1016, 1024 (bits) • Additional authenticated data lengths: 0, 8, 1016, 1024 (bits) • 96 bit IV supported <p>AES-XTS:</p> <ul style="list-style-type: none"> • Key Size: 128: • Modes: Decrypt, Encrypt • Block Sizes: Full • Key Size: 256: • Modes: Decrypt, Encrypt • Block Sizes: Full 	

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>AES-KW:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • CIPHER transformation direction: Forward • Key Lengths: 128, 192, 256 (bits) • Plain Text Lengths: 128, 192, 256, 320, 2048 (bits) <p>AES validation number 4902</p>	<p>Microsoft Surface Hub Cryptography Next Generation (CNG) Implementations #4900</p> <p>Version 10.0.15063.674</p>
<p>AES-KW:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • CIPHER transformation direction: Forward • Key Lengths: 128, 192, 256 (bits) • Plain Text Lengths: 128, 192, 256, 320, 2048 (bits) <p>AES validation number 4901</p>	<p>Windows 10 Mobile (version 1709) Cryptography Next Generation (CNG) Implementations #4899</p> <p>Version 10.0.15254</p>
<p>AES-KW:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • CIPHER transformation direction: Forward • Key Lengths: 128, 192, 256 (bits) • Plain Text Lengths: 128, 192, 256, 320, 2048 (bits) <p>AES validation number 4897</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); Cryptography Next Generation (CNG) Implementations #4898</p> <p>Version 10.0.16299</p>
<p>AES-CCM:</p> <ul style="list-style-type: none"> • Key Lengths: 256 (bits) • Tag Lengths: 128 (bits) • IV Lengths: 96 (bits) • Plain • Text Length: 0-32 • Additional authenticated data length: 0-65536 <p>AES validation number 4902</p>	<p>Microsoft Surface Hub BitLocker(R) Cryptographic Implementations #4896</p> <p>Version 10.0.15063.674</p>
<p>AES-CCM:</p> <ul style="list-style-type: none"> • Key Lengths: 256 (bits) • Tag Lengths: 128 (bits) • IV Lengths: 96 (bits) • Plain Text Length: 0-32 • Additional authenticated data length: 0-65536 <p>AES validation number 4901</p>	<p>Windows 10 Mobile (version 1709) BitLocker(R) Cryptographic Implementations #4895</p> <p>Version 10.0.15254</p>
<p>AES-CCM:</p> <ul style="list-style-type: none"> • Key Lengths: 256 (bits) • Tag Lengths: 128 (bits) • IV Lengths: 96 (bits) • Plain Text Length: 0-32 • Additional authenticated data length: 0-65536 <p>AES validation number 4897</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); BitLocker(R) Cryptographic Implementations #4894</p> <p>Version 10.0.16299</p>
<p>CBC (e/d; 128, 192, 256);</p> <p>CFB128 (e/d; 128, 192, 256);</p> <p>OFB (e/d; 128, 192, 256);</p> <p>CTR (int only; 128, 192, 256)</p>	<p>Windows 10 Creators Update (version 1703) Pro, Enterprise, Education Virtual TPM Implementations #4627</p> <p>Version 10.0.15063</p>


MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>KW (AE, AD, AES-128, AES-192, AES-256, FWD, 128, 256, 192, 320, 2048) AES validation number 4624</p>	<p>Windows 10 Creators Update (version 1703) Home, Pro, Enterprise, Education, Windows 10 S, Windows 10 Mobile Cryptography Next Generation (CNG) Implementations #4626 Version 10.0.15063</p>
<p>CCM (KS: 256) (Assoc. Data Len Range: 0-0, 2¹⁶) (Payload Length Range: 0 - 32 (Nonce Length(s): 12 (Tag Length(s): 16) AES validation number 4624</p>	<p>Windows 10 Creators Update (version 1703) Home, Pro, Enterprise, Education, Windows 10 S, Windows 10 Mobile BitLocker(R) Cryptographic Implementations #4625 Version 10.0.15063</p>
<p>ECB (e/d; 128, 192, 256); CBC (e/d; 128, 192, 256); CFB8 (e/d; 128, 192, 256); CFB128 (e/d; 128, 192, 256); CTR (int only; 128, 192, 256) CCM (KS: 128, 192, 256) (Assoc. Data Len Range: 0-0, 2¹⁶) (Payload Length Range: 0 - 32 (Nonce Length(s): 7 8 9 10 11 12 13 (Tag Length(s): 4 6 8 10 12 14 16) CMAC (Generation/Verification) (KS: 128; Block Size(s): Full/Partial; Msg Len(s) Min: 0 Max: 2¹⁶; Tag Len(s) Min: 16 Max: 16) (KS: 192; Block Size(s): Full/Partial; Msg Len(s) Min: 0 Max: 2¹⁶; Tag Len(s) Min: 16 Max: 16) (KS: 256; Block Size(s): Full/Partial; Msg Len(s) Min: 0 Max: 2¹⁶; Tag Len(s) Min: 16 Max: 16) GCM (KS: AES_128(e/d) Tag Length(s): 128 120 112 104 96) (KS: AES_192(e/d) Tag Length(s): 128 120 112 104 96) (KS: AES_256(e/d) Tag Length(s): 128 120 112 104 96) IV Generated: (External); PT Lengths Tested: (0, 1024, 8, 1016); Additional authenticated data lengths tested: (0, 1024, 8, 1016); 96 bit IV supported GMAC supported XTS((KS: XTS_128((e/d)(f)) KS: XTS_256((e/d)(f))</p>	<p>Windows 10 Creators Update (version 1703) Home, Pro, Enterprise, Education, Windows 10 S, Windows 10 Mobile SymCrypt Cryptographic Implementations #4624 Version 10.0.15063</p>
<p>ECB (e/d; 128, 192, 256); CBC (e/d; 128, 192, 256);</p>	<p>Windows Embedded Compact Enhanced Cryptographic Provider (RSAENH) #4434 Version 7.00.2872</p>
<p>ECB (e/d; 128, 192, 256); CBC (e/d; 128, 192, 256);</p>	<p>Windows Embedded Compact Enhanced Cryptographic Provider (RSAENH) #4433 Version 8.00.6246</p>
<p>ECB (e/d; 128, 192, 256); CBC (e/d; 128, 192, 256); CTR (int only; 128, 192, 256)</p>	<p>Windows Embedded Compact Cryptographic Primitives Library (bcrypt.dll) #4431 Version 7.00.2872</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>ECB (e/d; 128, 192, 256); CBC (e/d; 128, 192, 256); CTR (int only; 128, 192, 256)</p>	<p>Windows Embedded Compact Cryptographic Primitives Library (bcrypt.dll) #4430 Version 8.00.6246</p>
<p>CBC (e/d; 128, 192, 256); CFB128 (e/d; 128, 192, 256); OFB (e/d; 128, 192, 256); CTR (int only; 128, 192, 256)</p>	<p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, and Surface Pro 3 w/ Windows 10 Anniversary Update Virtual TPM Implementations #4074 Version 10.0.14393</p>
<p>ECB (e/d; 128, 192, 256); CBC (e/d; 128, 192, 256); CFB8 (e/d; 128, 192, 256); CFB128 (e/d; 128, 192, 256); CTR (int only; 128, 192, 256)</p> <p>CCM (KS: 128, 192, 256) (Assoc. Data Len Range: 0-0, 2^16) (Payload Length Range: 0 - 32 (Nonce Length(s): 7 8 9 10 11 12 13 (Tag Length(s): 4 6 8 10 12 14 16)</p> <p>CMAC (Generation/Verification) (KS: 128; Block Size(s): Full/Partial; Msg Len(s) Min: 0 Max: 2^16; Tag Len(s) Min: 0 Max: 16) (KS: 192; Block Size(s): Full/Partial; Msg Len(s) Min: 0 Max: 2^16; Tag Len(s) Min: 0 Max: 16) (KS: 256; Block Size(s): Full/Partial; Msg Len(s) Min: 0 Max: 2^16; Tag Len(s) Min: 0 Max: 16)</p> <p>GCM (KS: AES_128(e/d) Tag Length(s): 128 120 112 104 96) (KS: AES_192(e/d) Tag Length(s): 128 120 112 104 96)</p> <p>(KS: AES_256(e/d) Tag Length(s): 128 120 112 104 96)</p> <p>IV Generated: (Externally); PT Lengths Tested: (0, 1024, 8, 1016); Additional authenticated data lengths tested: (0, 1024, 8, 1016); IV Lengths Tested: (0, 0); 96 bit IV supported</p> <p>GMAC supported</p> <p>XTS((KS: XTS_128((e/d)(f)) KS: XTS_256((e/d)(f)))</p>	<p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, Surface Pro 3 and Surface 3 w/ Windows 10 Anniversary Update; Microsoft Lumia 950 and Lumia 650 w/ Windows 10 Mobile Anniversary Update SymCrypt Cryptographic Implementations #4064 Version 10.0.14393</p>
<p>ECB (e/d; 128, 192, 256); CBC (e/d; 128, 192, 256); CFB8 (e/d; 128, 192, 256);</p>	<p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, Surface Pro 3 and Surface 3 w/ Windows 10 Anniversary Update; Microsoft Lumia 950 and Lumia 650 w/ Windows 10 Mobile Anniversary Update RSA32 Algorithm Implementations #4063 Version 10.0.14393</p>
<p>KW (AE, AD, AES-128, AES-192, AES-256, FWD, 128, 192, 256, 320, 2048) AES validation number 4064</p>	<p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, Surface Pro 3 and Surface 3 w/ Windows 10 Anniversary Update; Microsoft Lumia 950 and Lumia 650 w/ Windows 10 Mobile Anniversary Update Cryptography Next Generation (CNG) Implementations #4062 Version 10.0.14393</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>CCM (KS: 256) (Assoc. Data Len Range: 0-0, 2¹⁶) (Payload Length Range: 0 - 32 (Nonce Length(s): 12 (Tag Length(s): 16)</p> <p>AES validation number 4064</p>	<p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, Surface Pro 3 and Surface 3 w/ Windows 10 Anniversary Update; Microsoft Lumia 950 and Lumia 650 w/ Windows 10 Mobile Anniversary Update BitLocker® Cryptographic Implementations #4061 Version 10.0.14393</p>
<p>KW (AE, AD, AES-128, AES-192, AES-256, FWD, 128, 256, 192, 320, 2048)</p> <p>AES validation number 3629</p>	<p>Microsoft Windows 10 November 2015 Update; Microsoft Surface Book, Surface Pro 4, Surface Pro 3, Surface 3, Surface Pro 2, and Surface Pro w/ Windows 10 November 2015 Update; Windows 10 Mobile for Microsoft Lumia 950 and Microsoft Lumia 635; Windows 10 for Microsoft Surface Hub 84" and Surface Hub 55" Cryptography Next Generation (CNG) Implementations #3652 Version 10.0.10586</p>
<p>CCM (KS: 256) (Assoc. Data Len Range: 0-0, 2¹⁶) (Payload Length Range: 0 - 32 (Nonce Length(s): 12 (Tag Length(s): 16)</p> <p>AES validation number 3629</p>	<p>Microsoft Windows 10 November 2015 Update; Microsoft Surface Book, Surface Pro 4, Surface Pro 3, Surface 3, Surface Pro 2, and Surface Pro w/ Windows 10 November 2015 Update; Windows 10 Mobile for Microsoft Lumia 950 and Microsoft Lumia 635; Windows 10 for Microsoft Surface Hub 84" and Surface Hub 55" BitLocker® Cryptographic Implementations #3653 Version 10.0.10586</p>
<p>ECB (e/d; 128, 192, 256); CBC (e/d; 128, 192, 256); CFB8 (e/d; 128, 192, 256);</p>	<p>Microsoft Windows 10 November 2015 Update; Microsoft Surface Book, Surface Pro 4, Surface Pro 3, Surface 3, Surface Pro 2, and Surface Pro w/ Windows 10 November 2015 Update; Windows 10 Mobile for Microsoft Lumia 950 and Microsoft Lumia 635; Windows 10 for Microsoft Surface Hub 84" and Surface Hub 55" RSA32 Algorithm Implementations #3630 Version 10.0.10586</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>ECB (e/d; 128, 192, 256); CBC (e/d; 128, 192, 256); CFB8 (e/d; 128, 192, 256);</p> <p>CFB128 (e/d; 128, 192, 256); CTR (int only; 128, 192, 256)</p> <p>CCM (KS: 128, 192, 256) (Assoc. Data Len Range: 0-0, 2¹⁶) (Payload Length Range: 0 - 32 (Nonce Length(s): 7 8 9 10 11 12 13 (Tag Length(s): 4 6 8 10 12 14 16)</p> <p>CMAC (Generation/Verification) (KS: 128; Block Size(s): Full/Partial; Msg Len(s) Min: 0 Max: 2¹⁶; Tag Len(s) Min: 0 Max: 16) (KS: 192; Block Size(s): Full/Partial; Msg Len(s) Min: 0 Max: 2¹⁶; Tag Len(s) Min: 0 Max: 16) (KS: 256; Block Size(s): Full/Partial; Msg Len(s) Min: 0 Max: 2¹⁶; Tag Len(s) Min: 0 Max: 16)</p> <p>GCM (KS: AES_128(e/d) Tag Length(s): 128 120 112 104 96) (KS: AES_192(e/d) Tag Length(s): 128 120 112 104 96)</p> <p>(KS: AES_256(e/d) Tag Length(s): 128 120 112 104 96)IV Generated: (Externally); PT Lengths Tested: (0, 1024, 8, 1016); Additional authenticated data lengths tested: (0, 1024, 8, 1016); IV Lengths Tested: (0, 0); 96 bit IV supported</p> <p>GMAC supported</p> <p>XTS((KS: XTS_128((e/d) (f)) KS: XTS_256((e/d) (f))</p>	<p>Microsoft Windows 10 November 2015 Update; Microsoft Surface Book, Surface Pro 4, Surface Pro 3, Surface 3, Surface Pro 2, and Surface Pro w/ Windows 10 November 2015 Update; Windows 10 Mobile for Microsoft Lumia 950 and Microsoft Lumia 635; Windows 10 for Microsoft Surface Hub 84" and Surface Hub 55" SymCrypt Cryptographic Implementations #3629</p> <p>Version 10.0.10586</p>
<p>KW (AE, AD, AES-128, AES-192, AES-256, FWD, 128, 256, 192, 320, 2048)</p> <p>AES validation number 3497</p>	<p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, Surface Pro 3 and Surface 3 w/ Windows 10 Anniversary Update; Microsoft Lumia 950 and Lumia 650 w/ Windows 10 Mobile Anniversary Update Cryptography Next Generation (CNG) Implementations #3507</p> <p>Version 10.0.10240</p>
<p>CCM (KS: 256) (Assoc. Data Len Range: 0-0, 2¹⁶) (Payload Length Range: 0 - 32 (Nonce Length(s): 12 (Tag Length(s): 16)</p> <p>AES validation number 3497</p>	<p>Microsoft Windows 10, Microsoft Surface Pro 3 with Windows 10, Microsoft Surface 3 with Windows 10, Microsoft Surface Pro 2 with Windows 10, Microsoft Surface Pro with Windows 10 BitLocker[®] Cryptographic Implementations #3498</p> <p>Version 10.0.10240</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>ECB (e/d; 128, 192, 256); CBC (e/d; 128, 192, 256); CFB8 (e/d; 128, 192, 256);</p> <p>CFB128 (e/d; 128, 192, 256); CTR (int only; 128, 192, 256)</p> <p>CCM (KS: 128, 192, 256) (Assoc. Data Len Range: 0-0, 2¹⁶) (Payload Length Range: 0 - 32 (Nonce Length(s): 7 8 9 10 11 12 13 (Tag Length(s): 4 6 8 10 12 14 16)</p> <p>CMAC(Generation/Verification) (KS: 128; Block Size(s): Full/Partial; Msg Len(s) Min: 0 Max: 2¹⁶; Tag Len(s) Min: 0 Max: 16) (KS: 192; Block Size(s): Full/Partial; Msg Len(s) Min: 0 Max: 2¹⁶; Tag Len(s) Min: 0 Max: 16) (KS: 256; Block Size(s): Full/Partial; Msg Len(s) Min: 0 Max: 2¹⁶; Tag Len(s) Min: 0 Max: 16)</p> <p>GCM (KS: AES_128(e/d) Tag Length(s): 128 120 112 104 96) (KS: AES_192(e/d) Tag Length(s): 128 120 112 104 96)</p> <p>(KS: AES_256(e/d) Tag Length(s): 128 120 112 104 96)</p> <p>IV Generated: (Externally); PT Lengths Tested: (0, 1024, 8, 1016); Additional authenticated data lengths tested: (0, 1024, 8, 1016); IV Lengths Tested: (0, 0); 96 bit IV supported</p> <p>GMAC supported</p> <p>XTS((KS: XTS_128((e/d)(f)) KS: XTS_256((e/d)(f)))</p>	<p>Microsoft Windows 10, Microsoft Surface Pro 3 with Windows 10, Microsoft Surface 3 with Windows 10, Microsoft Surface Pro 2 with Windows 10, Microsoft Surface Pro with Windows 10 SymCrypt Cryptographic Implementations #3497</p> <p>Version 10.0.10240</p>
<p>ECB (e/d; 128, 192, 256);</p> <p>CBC (e/d; 128, 192, 256);</p> <p>CFB8 (e/d; 128, 192, 256);</p>	<p>Microsoft Windows 10, Microsoft Surface Pro 3 with Windows 10, Microsoft Surface 3 with Windows 10, Microsoft Surface Pro 2 with Windows 10, Microsoft Surface Pro with Windows 10 RSA32 Algorithm Implementations #3476</p> <p>Version 10.0.10240</p>
<p>ECB (e/d; 128, 192, 256);</p> <p>CBC (e/d; 128, 192, 256);</p> <p>CFB8 (e/d; 128, 192, 256);</p>	<p>Microsoft Windows 8.1, Microsoft Windows Server 2012 R2, Microsoft Windows Storage Server 2012 R2, Microsoft Windows RT 8.1, Microsoft Surface with Windows RT 8.1, Microsoft Surface Pro with Windows 8.1, Microsoft Surface 2, Microsoft Surface Pro 2, Microsoft Surface Pro 3, Microsoft Windows Phone 8.1, Microsoft Windows Embedded 8.1 Industry RSA32 Algorithm Implementations #2853</p> <p>Version 6.3.9600</p>
<p>CCM (KS: 256) (Assoc. Data Len Range: 0-0, 2¹⁶) (Payload Length Range: 0 - 32 (Nonce Length(s): 12 (Tag Length(s): 16)</p> <p>AES validation number 2832</p>	<p>Microsoft Windows 8.1, Microsoft Windows Server 2012 R2, Microsoft Windows Storage Server 2012 R2, Microsoft Windows RT 8.1, Microsoft Surface with Windows RT 8.1, Microsoft Surface Pro with Windows 8.1, Microsoft Surface 2, Microsoft Surface Pro 2, Microsoft Surface Pro 3, Microsoft Windows Phone 8.1, Microsoft Windows Embedded 8.1 Industry, and Microsoft StorSimple 8100 BitLocker Cryptographic Implementations #2848</p> <p>Version 6.3.9600</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>CCM (KS: 128, 192, 256) (Assoc. Data Len Range: 0-0, 2¹⁶) (Payload Length Range: 0 - 0 (Nonce Length(s): 7 8 9 10 11 12 13 (Tag Length(s): 4 6 8 10 12 14 16)</p> <p>CMAC (Generation/Verification) (KS: 128; Block Size(s): Full/Partial; Msg Len(s) Min: 0 Max: 2¹⁶; Tag Len(s) Min: 0 Max: 16) (KS: 192; Block Size(s): Full/Partial; Msg Len(s) Min: 0 Max: 2¹⁶; Tag Len(s) Min: 0 Max: 16) (KS: 256; Block Size(s): Full/Partial; Msg Len(s) Min: 0 Max: 2¹⁶; Tag Len(s) Min: 0 Max: 16)</p> <p>GCM (KS: AES_128(e/d) Tag Length(s): 128 120 112 104 96) (KS: AES_192(e/d) Tag Length(s): 128 120 112 104 96)</p> <p>(KS: AES_256(e/d) Tag Length(s): 128 120 112 104 96)</p> <p>IV Generated: (Externally); PT Lengths Tested: (0, 128, 1024, 8, 1016); Additional authenticated data lengths tested: (0, 128, 1024, 8, 1016); IV Lengths Tested: (8, 1024); 96 bit IV supported;</p> <p>OtherIVLen_Supported</p> <p>GMAC supported</p>	<p>Windows Storage Server 2012 R2, Microsoft Windows RT 8.1, Microsoft Surface with Windows RT 8.1, Microsoft Surface Pro with Windows 8.1, Microsoft Surface 2, Microsoft Surface Pro 2, Microsoft Surface Pro 3, Microsoft Windows Phone 8.1, Microsoft Windows Embedded 8.1 Industry, and Microsoft StorSimple 8100 SymCrypt Cryptographic Implementations #2832</p> <p>Version 6.3.9600</p>
<p>CCM (KS: 128, 192, 256) (Assoc. Data Len Range: 0-0, 2¹⁶) (Payload Length Range: 0 - 32 (Nonce Length(s): 7 8 9 10 11 12 13 (Tag Length(s): 4 6 8 10 12 14 16)</p> <p>AES validation number 2197</p> <p>CMAC (Generation/Verification) (KS: 128; Block Size(s); Msg Len(s) Min: 0 Max: 2¹⁶; Tag Len(s) Min: 16 Max: 16) (KS: 192; Block Size(s); Msg Len(s) Min: 0 Max: 2¹⁶; Tag Len(s) Min: 16 Max: 16) (KS: 256; Block Size(s); Msg Len(s) Min: 0 Max: 2¹⁶; Tag Len(s) Min: 16 Max: 16)</p> <p>AES validation number 2197</p> <p>GCM(KS: AES_128(e/d) Tag Length(s): 128 120 112 104 96) (KS: AES_192(e/d) Tag Length(s): 128 120 112 104 96)</p> <p>(KS: AES_256(e/d) Tag Length(s): 128 120 112 104 96)</p> <p>IV Generated: (Externally); PT Lengths Tested: (0, 128, 1024, 8, 1016); Additional authenticated data lengths tested: (0, 128, 1024, 8, 1016); IV Lengths Tested: (8, 1024); 96 bit IV supported</p> <p>GMAC supported</p>	<p>Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 Cryptography Next Generation (CNG) Implementations #2216</p>
<p>CCM (KS: 256) (Assoc. Data Len Range: 0 - 0, 2¹⁶) (Payload Length Range: 0 - 32 (Nonce Length(s): 12 (Tag Length(s): 16)</p> <p>AES validation number 2196</p>	<p>Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 BitLocker  Cryptographic Implementations #2198</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
ECB (e/d; 128, 192, 256); CBC (e/d; 128, 192, 256); CFB8 (e/d; 128, 192, 256); CFB128 (e/d; 128, 192, 256); CTR (int only; 128, 192, 256)	Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 Next Generation Symmetric Cryptographic Algorithms Implementations (SYMCRYPT) #2197
ECB (e/d; 128, 192, 256); CBC (e/d; 128, 192, 256); CFB8 (e/d; 128, 192, 256);	Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 Symmetric Algorithm Implementations (RSA32) #2196
CCM (KS: 128, 192, 256) (Assoc. Data Len Range: 0 – 0, 2[^]16) (Payload Length Range: 0 - 32 (Nonce Length(s): 7 8 9 10 11 12 13 (Tag Length(s): 4 6 8 10 12 14 16) AES validation number 1168	Windows Server 2008 R2 and SP1 CNG algorithms #1187 Windows 7 Ultimate and SP1 CNG algorithms #1178
CCM (KS: 128, 256) (Assoc. Data Len Range: 0 - 8) (Payload Length Range: 4 - 32 (Nonce Length(s): 7 8 12 13 (Tag Length(s): 4 6 8 14 16) AES validation number 1168	Windows 7 Ultimate and SP1 and Windows Server 2008 R2 and SP1 BitLocker Algorithm Implementations #1177
ECB (e/d; 128, 192, 256); CBC (e/d; 128, 192, 256); CFB8 (e/d; 128, 192, 256);	Windows 7 and SP1 and Windows Server 2008 R2 and SP1 Symmetric Algorithm Implementation #1168
GCM GMAC	Windows 7 and SP1 and Windows Server 2008 R2 and SP1 Symmetric Algorithm Implementation #1168 , vendor-affirmed
CCM (KS: 128, 256) (Assoc. Data Len Range: 0 - 8) (Payload Length Range: 4 - 32 (Nonce Length(s): 7 8 12 13 (Tag Length(s): 4 6 8 14 16)	Windows Vista Ultimate SP1 and Windows Server 2008 BitLocker Algorithm Implementations #760
CCM (KS: 128, 192, 256) (Assoc. Data Len Range: 0 - 0, 2[^]16) (Payload Length Range: 1 - 32 (Nonce Length(s): 7 8 9 10 11 12 13 (Tag Length(s): 4 6 8 10 12 14 16**)**	Windows Server 2008 CNG algorithms #757 Windows Vista Ultimate SP1 CNG algorithms #756
CBC (e/d; 128, 256); CCM (KS: 128, 256) (Assoc. Data Len Range: 0 - 8) (Payload Length Range: 4 - 32 (Nonce Length(s): 7 8 12 13 (Tag Length(s): 4 6 8 14 16)	Windows Vista Ultimate BitLocker Drive Encryption #715 Windows Vista Ultimate BitLocker Drive Encryption #424
ECB (e/d; 128, 192, 256); CBC (e/d; 128, 192, 256); CFB8 (e/d; 128, 192, 256);	Windows Vista Ultimate SP1 and Windows Server 2008 Symmetric Algorithm Implementation #739 Windows Vista Symmetric Algorithm Implementation #553

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>ECB (e/d; 128, 192, 256); CBC (e/d; 128, 192, 256); CTR (int only; 128, 192, 256)</p>	<p>Windows Embedded Compact 7 Cryptographic Primitives Library (bcrypt.dll) #2023</p>
<p>ECB (e/d; 128, 192, 256); CBC (e/d; 128, 192, 256);</p>	<p>Windows Embedded Compact 7 Enhanced Cryptographic Provider (RSAENH) #2024</p> <p>Windows Server 2003 SP2 Enhanced Cryptographic Provider (RSAENH) #818</p> <p>Windows XP Professional SP3 Enhanced Cryptographic Provider (RSAENH) #781</p> <p>Windows 2003 SP2 Enhanced Cryptographic Provider (RSAENH) #548</p> <p>Windows CE 6.0 and Windows CE 6.0 R2 and Windows Mobile Enhanced Cryptographic Provider (RSAENH) #516</p> <p>Windows CE and Windows Mobile 6, 6.1, and 6.5 Enhanced Cryptographic Provider (RSAENH) #507</p> <p>Windows Server 2003 SP1 Enhanced Cryptographic Provider (RSAENH) #290</p> <p>Windows CE 5.0 and 5.1 Enhanced Cryptographic Provider (RSAENH) #224</p> <p>Windows Server 2003 Enhanced Cryptographic Provider (RSAENH) #80</p> <p>Windows XP, SP1, and SP2 Enhanced Cryptographic Provider (RSAENH) #33</p>

Deterministic Random Bit Generator (DRBG)

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>Counter:</p> <ul style="list-style-type: none"> • Modes: AES-256 • Derivation Function States: Derivation Function not used • Prediction Resistance Modes: Not Enabled <p>Prerequisite: AES #4904</p>	<p>Microsoft Surface Hub Virtual TPM Implementations #1734 Version 10.0.15063.674</p>
<p>Counter:</p> <ul style="list-style-type: none"> • Modes: AES-256 • Derivation Function States: Derivation Function not used • Prediction Resistance Modes: Not Enabled <p>Prerequisite: AES #4903</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); Virtual TPM Implementations #1733 Version 10.0.16299</p>
<p>Counter:</p> <ul style="list-style-type: none"> • Modes: AES-256 • Derivation Function States: Derivation Function used • Prediction Resistance Modes: Not Enabled <p>Prerequisite: AES #4902</p>	<p>Microsoft Surface Hub SymCrypt Cryptographic Implementations #1732 Version 10.0.15063.674</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>Counter:</p> <ul style="list-style-type: none"> • Modes: AES-256 • Derivation Function States: Derivation Function used • Prediction Resistance Modes: Not Enabled <p>Prerequisite: AES #4901</p>	<p>Windows 10 Mobile (version 1709) SymCrypt Cryptographic Implementations #1731 Version 10.0.15254</p>
<p>Counter:</p> <ul style="list-style-type: none"> • Modes: AES-256 • Derivation Function States: Derivation Function used • Prediction Resistance Modes: Not Enabled <p>Prerequisite: AES #4897</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); SymCrypt Cryptographic Implementations #1730 Version 10.0.16299</p>
<p>CTR_DRBG: [Prediction Resistance Tested: Not Enabled; BlockCipher_No_df: (AES-256) (AES validation number 4627)]</p>	<p>Windows 10 Creators Update (version 1703) Pro, Enterprise, Education Virtual TPM Implementations #1556 Version 10.0.15063</p>
<p>CTR_DRBG: [Prediction Resistance Tested: Not Enabled; BlockCipher_Use_df: (AES-256 (AES validation number 4624))]</p>	<p>Windows 10 Creators Update (version 1703) Home, Pro, Enterprise, Education, Windows 10 S, Windows 10 Mobile SymCrypt Cryptographic Implementations #1555 Version 10.0.15063</p>
<p>CTR_DRBG: [Prediction Resistance Tested: Not Enabled; BlockCipher_No_df: (AES-256) (AES validation number 4434)]</p>	<p>Windows Embedded Compact Enhanced Cryptographic Provider (RSAENH) #1433 Version 7.00.2872</p>
<p>CTR_DRBG: [Prediction Resistance Tested: Not Enabled; BlockCipher_No_df: (AES-256) (AES validation number 4433)]</p>	<p>Windows Embedded Compact Enhanced Cryptographic Provider (RSAENH) #1432 Version 8.00.6246</p>
<p>CTR_DRBG: [Prediction Resistance Tested: Not Enabled; BlockCipher_No_df: (AES-256) (AES validation number 4431)]</p>	<p>Windows Embedded Compact Cryptographic Primitives Library (bcrypt.dll) #1430 Version 7.00.2872</p>
<p>CTR_DRBG: [Prediction Resistance Tested: Not Enabled; BlockCipher_No_df: (AES-256) (AES validation number 4430)]</p>	<p>Windows Embedded Compact Cryptographic Primitives Library (bcrypt.dll) #1429 Version 8.00.6246</p>
<p>CTR_DRBG: [Prediction Resistance Tested: Not Enabled; BlockCipher_No_df: (AES-256) (AES validation number 4074)]</p>	<p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, and Surface Pro 3 w/ Windows 10 Anniversary Update Virtual TPM Implementations #1222 Version 10.0.14393</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
CTR_DRBG:[Prediction Resistance Tested: Not Enabled; BlockCipher_Use_df: (AES-256) (AES validation number 4064)]	Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, Surface Pro 3 and Surface 3 w/ Windows 10 Anniversary Update; Microsoft Lumia 950 and Lumia 650 w/ Windows 10 Mobile Anniversary Update SymCrypt Cryptographic Implementations #1217 Version 10.0.14393
CTR_DRBG:[Prediction Resistance Tested: Not Enabled; BlockCipher_Use_df: (AES-256) (AES validation number 3629)]	Microsoft Windows 10 November 2015 Update; Microsoft Surface Book, Surface Pro 4, Surface Pro 3, Surface 3, Surface Pro 2, and Surface Pro w/ Windows 10 November 2015 Update; Windows 10 Mobile for Microsoft Lumia 950 and Microsoft Lumia 635; Windows 10 for Microsoft Surface Hub and Surface Hub SymCrypt Cryptographic Implementations #955 Version 10.0.10586
CTR_DRBG:[Prediction Resistance Tested: Not Enabled; BlockCipher_Use_df: (AES-256) (AES validation number 3497)]	Microsoft Windows 10, Microsoft Surface Pro 3 with Windows 10, Microsoft Surface 3 with Windows 10, Microsoft Surface Pro 2 with Windows 10, Microsoft Surface Pro with Windows 10 SymCrypt Cryptographic Implementations #868 Version 10.0.10240
CTR_DRBG:[Prediction Resistance Tested: Not Enabled; BlockCipher_Use_df: (AES-256) (AES validation number 2832)]	Windows Storage Server 2012 R2, Microsoft Windows RT 8.1, Microsoft Surface with Windows RT 8.1, Microsoft Surface Pro with Windows 8.1, Microsoft Surface 2, Microsoft Surface Pro 2, Microsoft Surface Pro 3, Microsoft Windows Phone 8.1, Microsoft Windows Embedded 8.1 Industry, and Microsoft StorSimple 8100 SymCrypt Cryptographic Implementations #489 Version 6.3.9600
CTR_DRBG:[Prediction Resistance Tested: Not Enabled; BlockCipher_Use_df: (AES-256) (AES validation number 2197)]	Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 Next Generation Symmetric Cryptographic Algorithms Implementations (SYMCRYPT) #258
CTR_DRBG:[Prediction Resistance Tested: Not Enabled; BlockCipher_No_df: (AES-256) (AES validation number 2023)]	Windows Embedded Compact 7 Cryptographic Primitives Library (bcrypt.dll) #193
CTR_DRBG:[Prediction Resistance Tested: Not Enabled; BlockCipher_No_df: (AES-256) (AES validation number 1168)]	Windows 7 Ultimate and SP1 and Windows Server 2008 R2 and SP1 RNG Library #23
DRBG (SP 800-90)	Windows Vista Ultimate SP1, vendor-affirmed

Digital Signature Algorithm (DSA)

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>DSA:</p> <ul style="list-style-type: none"> • 186-4: <p>PQGen:</p> <ul style="list-style-type: none"> • L = 2048, N = 256 SHA: SHA-256 • L = 3072, N = 256 SHA: SHA-256 <p>PQVer:</p> <ul style="list-style-type: none"> • L = 2048, N = 256 SHA: SHA-256 • L = 3072, N = 256 SHA: SHA-256 <p>SigGen:</p> <ul style="list-style-type: none"> • L = 2048, N = 256 SHA: SHA-256 • L = 3072, N = 256 SHA: SHA-256 <p>SigVer:</p> <ul style="list-style-type: none"> • L = 2048, N = 256 SHA: SHA-256 • L = 3072, N = 256 SHA: SHA-256 <p>KeyPair:</p> <ul style="list-style-type: none"> • L = 2048, N = 256 • L = 3072, N = 256 <p>Prerequisite: SHS #4011, DRBG #1732</p>	<p>Microsoft Surface Hub SymCrypt Cryptographic Implementations #1303 Version 10.0.15063.674</p>
<p>DSA:</p> <ul style="list-style-type: none"> • 186-4: <p>PQGen:</p> <ul style="list-style-type: none"> • L = 2048, N = 256 SHA: SHA-256 • L = 3072, N = 256 SHA: SHA-256 <p>PQVer:</p> <ul style="list-style-type: none"> • L = 2048, N = 256 SHA: SHA-256 • L = 3072, N = 256 SHA: SHA-256 <p>SigGen:</p> <ul style="list-style-type: none"> • L = 2048, N = 256 SHA: SHA-256 • L = 3072, N = 256 SHA: SHA-256 <p>SigVer:</p> <ul style="list-style-type: none"> • L = 2048, N = 256 SHA: SHA-256 • L = 3072, N = 256 SHA: SHA-256 <p>KeyPair:</p> <ul style="list-style-type: none"> • L = 2048, N = 256 • L = 3072, N = 256 <p>Prerequisite: SHS #4010, DRBG #1731</p>	<p>Windows 10 Mobile (version 1709) SymCrypt Cryptographic Implementations #1302 Version 10.0.15254</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>DSA:</p> <ul style="list-style-type: none"> • 186-4: <p>PQGGen:</p> <ul style="list-style-type: none"> • L = 2048, N = 256 SHA: SHA-256 • L = 3072, N = 256 SHA: SHA-256 <p>PQGVer:</p> <ul style="list-style-type: none"> • L = 2048, N = 256 SHA: SHA-256 • L = 3072, N = 256 SHA: SHA-256 <p>SigGen:</p> <ul style="list-style-type: none"> • L = 2048, N = 256 SHA: SHA-256 • L = 3072, N = 256 SHA: SHA-256 <p>SigVer:</p> <ul style="list-style-type: none"> • L = 2048, N = 256 SHA: SHA-256 • L = 3072, N = 256 SHA: SHA-256 <p>KeyPair:</p> <ul style="list-style-type: none"> • L = 2048, N = 256 • L = 3072, N = 256 <p>Prerequisite: SHS #4009, DRBG #1730</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); SymCrypt Cryptographic Implementations #1301</p> <p>Version 10.0.16299</p>
<p>FIPS186-4:</p> <p>PQG(gen) PARMS TESTED: [(2048,256)SHA(256); (3072,256) SHA(256)]</p> <p>**PQG(ver)**PARMS TESTED: [(2048,256) SHA(256); (3072,256) SHA(256)]</p> <p>KeyPairGen: [(2048,256); (3072,256)]</p> <p>**SIG(gen)**PARMS TESTED: [(2048,256) SHA(256); (3072,256) SHA(256)]</p> <p>SIG(ver) PARMS TESTED: [(2048,256) SHA(256); (3072,256) SHA(256)]</p> <p>SHS: validation number 3790</p> <p>DRBG: validation number 1555</p>	<p>Windows 10 Creators Update (version 1703) Home, Pro, Enterprise, Education, Windows 10 S, Windows 10 Mobile SymCrypt Cryptographic Implementations #1223</p> <p>Version 10.0.15063</p>
<p>FIPS186-4:</p> <p>PQG(ver)PARMS TESTED: [(1024,160) SHA(1)]</p> <p>SIG(ver)PARMS TESTED: [(1024,160) SHA(1)]</p> <p>SHS: validation number 3649</p>	<p>Windows Embedded Compact Cryptographic Primitives Library (bcrypt.dll) #1188</p> <p>Version 7.00.2872</p>
<p>FIPS186-4:</p> <p>PQG(ver)PARMS TESTED: [(1024,160) SHA(1)]</p> <p>SIG(ver)PARMS TESTED: [(1024,160) SHA(1)]</p> <p>SHS: validation number 3648</p>	<p>Windows Embedded Compact Cryptographic Primitives Library (bcrypt.dll) #1187</p> <p>Version 8.00.6246</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-4: PQG(gen) PARMS TESTED: [(2048,256)SHA(256); (3072,256) SHA(256)] **PQG(ver)**PARMS TESTED: [(2048,256) SHA(256); (3072,256) SHA(256)] KeyPairGen: [(2048,256); (3072,256)]</p> <p>**SIG(gen)**PARMS TESTED: [(2048,256) SHA(256); (3072,256) SHA(256)]</p> <p>**SIG(ver)**PARMS TESTED: [(2048,256) SHA(256); (3072,256) SHA(256)]</p> <p>SHS: validation number 3347</p> <p>DRBG: validation number 1217</p>	<p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, Surface Pro 3 and Surface 3 w/ Windows 10 Anniversary Update; Microsoft Lumia 950 and Lumia 650 w/ Windows 10 Mobile Anniversary Update MsBignum Cryptographic Implementations #1098 Version 10.0.14393</p>
<p>FIPS186-4: PQG(gen) PARMS TESTED: [(2048,256)SHA(256); (3072,256) SHA(256)] **PQG(ver)**PARMS TESTED: [(2048,256) SHA(256); (3072,256) SHA(256)] KeyPairGen: [(2048,256); (3072,256)] **SIG(gen)**PARMS TESTED: [(2048,256) SHA(256); (3072,256) SHA(256)]</p> <p>**SIG(ver)**PARMS TESTED: [(2048,256) SHA(256); (3072,256) SHA(256)]</p> <p>SHS: validation number 3047</p> <p>DRBG: validation number 955</p>	<p>Microsoft Windows 10 November 2015 Update; Microsoft Surface Book, Surface Pro 4, Surface Pro 3, Surface 3, Surface Pro 2, and Surface Pro w/ Windows 10 November 2015 Update; Windows 10 Mobile for Microsoft Lumia 950 and Microsoft Lumia 635; Windows 10 for Microsoft Surface Hub 84" and Surface Hub 55" MsBignum Cryptographic Implementations #1024 Version 10.0.10586</p>
<p>FIPS186-4: PQG(gen) PARMS TESTED: [(2048,256)SHA(256); (3072,256) SHA(256)] **PQG(ver)**PARMS TESTED: [(2048,256) SHA(256); (3072,256) SHA(256)] KeyPairGen: [(2048,256); (3072,256)]</p> <p>**SIG(gen)**PARMS TESTED: [(2048,256) SHA(256); (3072,256) SHA(256)] **SIG(ver)**PARMS TESTED: [(2048,256) SHA(256); (3072,256) SHA(256)]</p> <p>SHS: validation number 2886</p> <p>DRBG: validation number 868</p>	<p>Microsoft Windows 10, Microsoft Surface Pro 3 with Windows 10, Microsoft Surface 3 with Windows 10, Microsoft Surface Pro 2 with Windows 10, Microsoft Surface Pro with Windows 10 MsBignum Cryptographic Implementations #983 Version 10.0.10240</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-4: PQG(gen) PARMS TESTED: [(2048,256)SHA(256); (3072,256) SHA(256)] PQG(ver)PARMS TESTED: [(2048,256), SHA(256); (3072,256) SHA(256)] KeyPairGen: [(2048,256); (3072,256)]</p> <p>**SIG(gen)**PARMS TESTED: [(2048,256) SHA(256); (3072,256) SHA(256)]</p> <p>**SIG(ver)**PARMS TESTED: [(2048,256) SHA(256); (3072,256) SHA(256)]</p> <p>SHS: validation number 2373</p> <p>DRBG: validation number 489</p>	<p>Microsoft Windows 8.1, Microsoft Windows Server 2012 R2, Microsoft Windows Storage Server 2012 R2, Microsoft Windows RT 8.1, Microsoft Surface with Windows RT 8.1, Microsoft Surface Pro with Windows 8.1, Microsoft Surface 2, Microsoft Surface Pro 2, Microsoft Surface Pro 3, Microsoft Windows Phone 8.1, Microsoft Windows Embedded 8.1 Industry, and Microsoft StorSimple 8100 MsBignum Cryptographic Implementations #855</p> <p>Version 6.3.9600</p>
<p>FIPS186-2: PQG(ver) MOD(1024); SIG(ver) MOD(1024);</p> <p>SHS: #1903</p> <p>DRBG: #258</p> <p>FIPS186-4: PQG(gen)PARMS TESTED: [(2048,256)SHA(256); (3072,256) SHA(256)]</p> <p>PQG(ver)PARMS TESTED: [(2048,256) SHA(256); (3072,256) SHA(256)]</p> <p>SIG(gen)PARMS TESTED: [(2048,256) SHA(256); (3072,256) SHA(256)]</p> <p>SIG(ver)PARMS TESTED: [(2048,256) SHA(256); (3072,256) SHA(256)]</p> <p>SHS: #1903</p> <p>DRBG: #258</p> <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical DSA List validation number 687.</p>	<p>Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 Cryptography Next Generation (CNG) Implementations #687</p>
<p>FIPS186-2: PQG(ver) MOD(1024); SIG(ver) MOD(1024);</p> <p>SHS: #1902</p> <p>DRBG: #258</p> <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical DSA List validation number 686.</p>	<p>Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 DSS and Diffie-Hellman Enhanced Cryptographic Provider (DSENH) #686</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-2: SIG(ver) MOD(1024); SHS: validation number 1773 DRBG: validation number 193</p> <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical DSA List validation number 645.</p>	<p>Windows Embedded Compact 7 Cryptographic Primitives Library (bcrypt.dll) #645</p>
<p>FIPS186-2: SIG(ver) MOD(1024); SHS: validation number 1081 DRBG: validation number 23</p> <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical DSA List validation number 391. See Historical DSA List validation number 386.</p>	<p>Windows Server 2008 R2 and SP1 CNG algorithms #391 Windows 7 Ultimate and SP1 CNG algorithms #386</p>
<p>FIPS186-2: SIG(ver) MOD(1024); SHS: validation number 1081 RNG: validation number 649</p> <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical DSA List validation number 390. See Historical DSA List validation number 385.</p>	<p>Windows Server 2008 R2 and SP1 Enhanced DSS (DSENH) #390 Windows 7 Ultimate and SP1 Enhanced DSS (DSENH) #385</p>
<p>FIPS186-2: SIG(ver) MOD(1024); SHS: validation number 753</p> <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical DSA List validation number 284. See Historical DSA List validation number 283.</p>	<p>Windows Server 2008 CNG algorithms #284 Windows Vista Ultimate SP1 CNG algorithms #283</p>
<p>FIPS186-2: SIG(ver) MOD(1024); SHS: validation number 753 RNG: validation number 435</p> <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical DSA List validation number 282. See Historical DSA List validation number 281.</p>	<p>Windows Server 2008 Enhanced DSS (DSENH) #282 Windows Vista Ultimate SP1 Enhanced DSS (DSENH) #281</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-2: SIG(ver) MOD(1024); SHS: validation number 618 RNG: validation number 321</p> <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical DSA List validation number 227. See Historical DSA List validation number 226.</p>	<p>Windows Vista CNG algorithms #227 Windows Vista Enhanced DSS (DSSENH) #226</p>
<p>FIPS186-2: SIG(ver) MOD(1024); SHS: validation number 784 RNG: validation number 448</p> <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical DSA List validation number 292.</p>	<p>Windows XP Professional SP3 Enhanced DSS and Diffie-Hellman Cryptographic Provider (DSSENH) #292</p>
<p>FIPS186-2: SIG(ver) MOD(1024); SHS: validation number 783 RNG: validation number 447</p> <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical DSA List validation number 291.</p>	<p>Windows XP Professional SP3 Enhanced Cryptographic Provider (RSAENH) #291</p>
<p>FIPS186-2: PQG(gen) MOD(1024); PQG(ver) MOD(1024); KEYGEN(Y) MOD(1024); SIG(gen) MOD(1024); SIG(ver) MOD(1024); SHS: validation number 611 RNG: validation number 314</p>	<p>Windows 2003 SP2 Enhanced DSS and Diffie-Hellman Cryptographic Provider #221</p>
<p>FIPS186-2: PQG(gen) MOD(1024); PQG(ver) MOD(1024); KEYGEN(Y) MOD(1024); SIG(gen) MOD(1024); SIG(ver) MOD(1024); SHS: validation number 385</p>	<p>Windows Server 2003 SP1 Enhanced DSS and Diffie-Hellman Cryptographic Provider (DSSENH) #146</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-2: PQG(ver) MOD(1024); KEYGEN(Y) MOD(1024);vSIG(gen) MOD(1024); SIG(ver) MOD(1024); SHS: validation number 181</p>	<p>Windows Server 2003 Enhanced DSS and Diffie-Hellman Cryptographic Provider (DSSSENH) #95</p>
<p>FIPS186-2: PQG(gen) MOD(1024); PQG(ver) MOD(1024); KEYGEN(Y) MOD(1024); SIG(gen) MOD(1024); SHS: SHA-1 (BYTE) SIG(ver) MOD(1024); SHS: SHA-1 (BYTE)</p>	<p>Windows 2000 DSSSENH.DLL #29 Windows 2000 DSSBASE.DLL #28 Windows NT 4 SP6 DSSSENH.DLL #26 Windows NT 4 SP6 DSSBASE.DLL #25</p>
<p>FIPS186-2: PRIME; FIPS186-2: **KEYGEN(Y):**SHS: SHA-1 (BYTE) SIG(gen):SIG(ver) MOD(1024); SHS: SHA-1 (BYTE)</p>	<p>Windows NT 4.0 SP4 Microsoft Enhanced DSS and Diffie-Hellman Cryptographic Provider #17</p>

Elliptic Curve Digital Signature Algorithm (ECDSA)

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>ECDSA:186-4: Key Pair Generation:</p> <ul style="list-style-type: none"> • Curves: P-256, P-384, P-521 • Generation Methods: Extra Random Bits <p>Public Key Validation:</p> <ul style="list-style-type: none"> • Curves: P-256, P-384, P-521 <p>Signature Generation:</p> <ul style="list-style-type: none"> • P-256 SHA: SHA-256 • P-384 SHA: SHA-384 • P-521 SHA: SHA-512 <p>Signature Verification:</p> <ul style="list-style-type: none"> • P-256 SHA: SHA-256 • P-384 SHA: SHA-384 • P-521 SHA: SHA-512 <p>Prerequisite: SHS #2373, DRBG #489</p>	<p>Microsoft Windows 8.1, Microsoft Windows Server 2012 R2, Microsoft Windows Storage Server 2012 R2, Microsoft Windows RT 8.1, Microsoft Surface with Windows RT 8.1, Microsoft Surface Pro with Windows 8.1, Microsoft Surface 2, Microsoft Surface Pro 2, Microsoft Surface Pro 3, Microsoft Windows Phone 8.1, Microsoft Windows Embedded 8.1 Industry, and Microsoft StorSimple 8100 MsBignum Cryptographic Implementations #1263 Version 6.3.9600</p>
<p>ECDSA:186-4: Key Pair Generation:</p> <ul style="list-style-type: none"> • Curves: P-256, P-384 • Generation Methods: Testing Candidates <p>Prerequisite: SHS #4011, DRBG #1734</p>	<p>Microsoft Surface Hub Virtual TPM Implementations #1253 Version 10.0.15063.674</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>ECDSA:186-4: Key Pair Generation:</p> <ul style="list-style-type: none"> • Curves: P-256, P-384 • Generation Methods: Testing Candidates <p>Prerequisite: SHS #4009, DRBG #1733</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); Virtual TPM Implementations #1252</p> <p>Version 10.0.16299</p>
<p>ECDSA:186-4: Key Pair Generation:</p> <ul style="list-style-type: none"> • Curves: P-256, P-384, P-521 • Generation Methods: Extra Random Bits <p>Public Key Validation:</p> <ul style="list-style-type: none"> • Curves: P-256, P-384, P-521 <p>Signature Generation:</p> <ul style="list-style-type: none"> • P-256 SHA: SHA-256 • P-384 SHA: SHA-384 • P-521 SHA: SHA-512 <p>Signature Verification:</p> <ul style="list-style-type: none"> • P-256 SHA: SHA-256 • P-384 SHA: SHA-384 • P-521 SHA: SHA-512 <p>Prerequisite: SHS #4011, DRBG #1732</p>	<p>Microsoft Surface Hub MsBignum Cryptographic Implementations #1251</p> <p>Version 10.0.15063.674</p>
<p>ECDSA:186-4: Key Pair Generation:</p> <ul style="list-style-type: none"> • Curves: P-256, P-384, P-521 • Generation Methods: Extra Random Bits <p>Public Key Validation:</p> <ul style="list-style-type: none"> • Curves: P-256, P-384, P-521 <p>Signature Generation:</p> <ul style="list-style-type: none"> • P-256 SHA: SHA-256 • P-384 SHA: SHA-384 • P-521 SHA: SHA-512 <p>Signature Verification:</p> <ul style="list-style-type: none"> • P-256 SHA: SHA-256 • P-384 SHA: SHA-384 • P-521 SHA: SHA-512 <p>Prerequisite: SHS #4011, DRBG #1732</p>	<p>Microsoft Surface Hub SymCrypt Cryptographic Implementations #1250</p> <p>Version 10.0.15063.674</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>ECDSA:186-4:</p> <p>Key Pair Generation:</p> <ul style="list-style-type: none"> • Curves: P-256, P-384, P-521 • Generation Methods: Extra Random Bits <p>Public Key Validation:</p> <ul style="list-style-type: none"> • Curves: P-256, P-384, P-521 <p>Signature Generation:</p> <ul style="list-style-type: none"> • P-256 SHA: SHA-256 • P-384 SHA: SHA-384 • P-521 SHA: SHA-512 <p>Signature Verification:</p> <ul style="list-style-type: none"> • P-256 SHA: SHA-256 • P-384 SHA: SHA-384 • P-521 SHA: SHA-512 <p>Prerequisite: SHS #4010, DRBG #1731</p>	<p>Windows 10 Mobile (version 1709) SymCrypt Cryptographic Implementations #1249</p> <p>Version 10.0.15254</p>
<p>ECDSA:186-4:</p> <p>Key Pair Generation:</p> <ul style="list-style-type: none"> • Curves: P-256, P-384, P-521 • Generation Methods: Extra Random Bits <p>Public Key Validation:</p> <ul style="list-style-type: none"> • Curves: P-256, P-384, P-521 <p>Signature Generation:</p> <ul style="list-style-type: none"> • P-256 SHA: SHA-256 • P-384 SHA: SHA-384 • P-521 SHA: SHA-512 <p>Signature Verification:</p> <ul style="list-style-type: none"> • P-256 SHA: SHA-256 • P-384 SHA: SHA-384 • P-521 SHA: SHA-512 <p>Prerequisite: SHS #4010, DRBG #1731</p>	<p>Windows 10 Mobile (version 1709) MsBignum Cryptographic Implementations #1248</p> <p>Version 10.0.15254</p>
<p>ECDSA:186-4:</p> <p>Key Pair Generation:</p> <ul style="list-style-type: none"> • Curves: P-256, P-384, P-521 • Generation Methods: Extra Random Bits <p>Public Key Validation:</p> <ul style="list-style-type: none"> • Curves: P-256, P-384, P-521 <p>Signature Generation:</p> <ul style="list-style-type: none"> • P-256 SHA: SHA-256 • P-384 SHA: SHA-384 • P-521 SHA: SHA-512 <p>Signature Verification:</p> <ul style="list-style-type: none"> • P-256 SHA: SHA-256 • P-384 SHA: SHA-384 • P-521 SHA: SHA-512 <p>Prerequisite: SHS #4009, DRBG #1730</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); MsBignum Cryptographic Implementations #1247</p> <p>Version 10.0.16299</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>ECDSA:186-4: Key Pair Generation:</p> <ul style="list-style-type: none"> • Curves: P-256, P-384, P-521 • Generation Methods: Extra Random Bits <p>Public Key Validation:</p> <ul style="list-style-type: none"> • Curves: P-256, P-384, P-521 <p>Signature Generation:</p> <ul style="list-style-type: none"> • P-256 SHA: SHA-256 • P-384 SHA: SHA-384 • P-521 SHA: SHA-512 <p>Signature Verification:</p> <ul style="list-style-type: none"> • P-256 SHA: SHA-256 • P-384 SHA: SHA-384 • P-521 SHA: SHA-512 <p>Prerequisite: SHS #4009, DRBG #1730</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); SymCrypt Cryptographic Implementations #1246 Version 10.0.16299</p>
<p>FIPS186-4: PKG: CURVES(P-256 P-384 TestingCandidates) SHS: validation number 3790 DRBG: validation number 1555</p>	<p>Windows 10 Creators Update (version 1703) Pro, Enterprise, Education Virtual TPM Implementations #1136 Version 10.0.15063</p>
<p>FIPS186-4: PKG: CURVES(P-256 P-384 P-521 ExtraRandomBits) PKV: CURVES(P-256 P-384 P-521) SigGen: CURVES(P-256: (SHA-256) P-384: (SHA-384) P-521: (SHA-512)) SigVer: CURVES(P-256: (SHA-256) P-384: (SHA-384) P-521: (SHA-512)) SHS: validation number 3790 DRBG: validation number 1555</p>	<p>Windows 10 Creators Update (version 1703) Home, Pro, Enterprise, Education, Windows 10 S, Windows 10 Mobile MsBignum Cryptographic Implementations #1135 Version 10.0.15063</p>
<p>FIPS186-4: PKG: CURVES(P-256 P-384 P-521 ExtraRandomBits) PKV: CURVES(P-256 P-384 P-521) SigGen: CURVES(P-256: (SHA-256) P-384: (SHA-384) P-521: (SHA-512)) SigVer: CURVES(P-256: (SHA-256) P-384: (SHA-384) P-521: (SHA-512)) SHS: validation number 3790 DRBG: validation number 1555</p>	<p>Windows 10 Creators Update (version 1703) Home, Pro, Enterprise, Education, Windows 10 S, Windows 10 Mobile SymCrypt Cryptographic Implementations #1133 Version 10.0.15063</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-4: PKG: CURVES(P-256 P-384 P-521 ExtraRandomBits) PKV: CURVES(P-256 P-384 P-521)</p> <p>SigGen: CURVES(P-256: (SHA-1, 256) P-384: (SHA-1, 384) P-521: (SHA-1, 512) SIG(gen) with SHA-1 affirmed for use with protocols only.</p> <p>SigVer: CURVES(P-256: (SHA-1, 256) P-384: (SHA-1, 384) P-521: (SHA-1, 512))</p> <p>SHS:validation number 3649</p> <p>DRBG:validation number 1430</p>	<p>Windows Embedded Compact Cryptographic Primitives Library (bcrypt.dll) #1073 Version 7.00.2872</p>
<p>FIPS186-4: PKG: CURVES(P-256 P-384 P-521 ExtraRandomBits) PKV: CURVES(P-256 P-384 P-521)</p> <p>SigGen: CURVES(P-256: (SHA-1, 256) P-384: (SHA-1, 384) P-521: (SHA-1, 512) SIG(gen) with SHA-1 affirmed for use with protocols only.</p> <p>SigVer: CURVES(P-256: (SHA-1, 256) P-384: (SHA-1, 384) P-521: (SHA-1, 512))</p> <p>SHS:validation number 3648</p> <p>DRBG:validation number 1429</p>	<p>Windows Embedded Compact Cryptographic Primitives Library (bcrypt.dll) #1072 Version 8.00.6246</p>
<p>FIPS186-4: PKG: CURVES(P-256 P-384 TestingCandidates)vPKV: CURVES(P-256 P-384) SigGen: CURVES(P-256: (SHA-1, 256) P-384: (SHA-1, 256, 384) SIG(gen) with SHA-1 affirmed for use with protocols only.vSigVer: CURVES(P-256: (SHA-1, 256) P-384: (SHA-1, 256, 384))</p> <p>SHS: validation number 3347</p> <p>DRBG: validation number 1222</p>	<p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, and Surface Pro 3 w/ Windows 10 Anniversary Update Virtual TPM Implementations #920 Version 10.0.14393</p>
<p>FIPS186-4: PKG: CURVES(P-256 P-384 P-521 ExtraRandomBits) PKV: CURVES(P-256 P-384 P-521)</p> <p>SigGen: CURVES(P-256: (SHA-256) P-384: (SHA-384) P-521: (SHA-512))</p> <p>SigVer: CURVES(P-256: (SHA-256) P-384: (SHA-384) P-521: (SHA-512))vSHS: validation number 3347</p> <p>DRBG: validation number 1217</p>	<p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, Surface Pro 3 and Surface 3 w/ Windows 10 Anniversary Update; Microsoft Lumia 950 and Lumia 650 w/ Windows 10 Mobile Anniversary Update MsBignum Cryptographic Implementations #911 Version 10.0.14393</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-4: PKG: CURVES(P-256 P-384 P-521 ExtraRandomBits) SigGen: CURVES(P-256: (SHA-256) P-384: (SHA-384) P-521: (SHA-512)) SigVer: CURVES(P-256: (SHA-256) P-384: (SHA-384) P-521: (SHA-512)) SHS: validation number 3047 DRBG: validation number 955</p>	<p>Microsoft Windows 10 November 2015 Update; Microsoft Surface Book, Surface Pro 4, Surface Pro 3, Surface 3, Surface Pro 2, and Surface Pro w/ Windows 10 November 2015 Update; Windows 10 Mobile for Microsoft Lumia 950 and Microsoft Lumia 635; Windows 10 for Microsoft Surface Hub 84" and Surface Hub 55" MsBignum Cryptographic Implementations #760 Version 10.0.10586</p>
<p>FIPS186-4: PKG: CURVES(P-256 P-384 P-521 ExtraRandomBits) SigGen: CURVES(P-256: (SHA-256) P-384: (SHA-384) P-521: (SHA-512)) SigVer: CURVES(P-256: (SHA-256) P-384: (SHA-384) P-521: (SHA-512)) SHS: validation number 2886 DRBG: validation number 868</p>	<p>Microsoft Windows 10, Microsoft Surface Pro 3 with Windows 10, Microsoft Surface 3 with Windows 10, Microsoft Surface Pro 2 with Windows 10, Microsoft Surface Pro with Windows 10 MsBignum Cryptographic Implementations #706 Version 10.0.10240</p>
<p>FIPS186-4: PKG: CURVES(P-256 P-384 P-521 ExtraRandomBits) SigGen: CURVES(P-256: (SHA-256) P-384: (SHA-384) P-521: (SHA-512)) SigVer: CURVES(P-256: (SHA-256) P-384: (SHA-384) P-521: (SHA-512)) SHS: validation number 2373 DRBG: validation number 489</p>	<p>Microsoft Windows 8.1, Microsoft Windows Server 2012 R2, Microsoft Windows Storage Server 2012 R2, Microsoft Windows RT 8.1, Microsoft Surface with Windows RT 8.1, Microsoft Surface Pro with Windows 8.1, Microsoft Surface 2, Microsoft Surface Pro 2, Microsoft Surface Pro 3, Microsoft Windows Phone 8.1, Microsoft Windows Embedded 8.1 Industry, and Microsoft StorSimple 8100 MsBignum Cryptographic Implementations #505 Version 6.3.9600</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-2: PKG: CURVES(P-256 P-384 P-521) SHS: #1903 DRBG: #258 SIG(ver): CURVES(P-256 P-384 P-521) SHS: #1903 DRBG: #258</p> <p>FIPS186-4: PKG: CURVES(P-256 P-384 P-521 ExtraRandomBits) SigGen: CURVES(P-256: (SHA-256) P-384: (SHA-384) P-521: (SHA-512)) SigVer: CURVES(P-256: (SHA-256) P-384: (SHA-384) P-521: (SHA-512)) SHS: #1903 DRBG: #258</p> <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical ECDSA List validation number 341.</p>	<p>Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 Cryptography Next Generation (CNG) Implementations #341</p>
<p>FIPS186-2: PKG: CURVES(P-256 P-384 P-521) SHS: validation number 1773 DRBG: validation number 193 SIG(ver): CURVES(P-256 P-384 P-521) SHS: validation number 1773 DRBG: validation number 193</p> <p>FIPS186-4: PKG: CURVES(P-256 P-384 P-521 ExtraRandomBits) SigGen: CURVES(P-256: (SHA-256) P-384: (SHA-384) P-521: (SHA-512)) SigVer: CURVES(P-256: (SHA-256) P-384: (SHA-384) P-521: (SHA-512)) SHS: validation number 1773 DRBG: validation number 193</p> <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical ECDSA List validation number 295.</p>	<p>Windows Embedded Compact 7 Cryptographic Primitives Library (bcrypt.dll) #295</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-2: PKG: CURVES(P-256 P-384 P-521) SHS: validation number 1081</p> <p>DRBG: validation number 23</p> <p>SIG(ver): CURVES(P-256 P-384 P-521) SHS: validation number 1081 DRBG: validation number 23</p> <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical ECDSA List validation number 142. See Historical ECDSA List validation number 141.</p>	<p>Windows Server 2008 R2 and SP1 CNG algorithms #142 Windows 7 Ultimate and SP1 CNG algorithms #141</p>
<p>FIPS186-2: PKG: CURVES(P-256 P-384 P-521) SHS: validation number 753</p> <p>SIG(ver): CURVES(P-256 P-384 P-521) SHS: validation number 753</p> <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical ECDSA List validation number 83. See Historical ECDSA List validation number 82.</p>	<p>Windows Server 2008 CNG algorithms #83 Windows Vista Ultimate SP1 CNG algorithms #82</p>
<p>FIPS186-2: PKG: CURVES(P-256 P-384 P-521) SHS: validation number 618</p> <p>RNG: validation number 321</p> <p>SIG(ver): CURVES(P-256 P-384 P-521) SHS: validation number 618 RNG: validation number 321</p> <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical ECDSA List validation number 60.</p>	<p>Windows Vista CNG algorithms #60</p>


Keyed-Hash Message Authentication Code (HMAC)

MODES / STATES / • KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>HMAC-SHA-1:</p> <ul style="list-style-type: none"> • • Key Sizes < Block Size • • Key Sizes > Block Size • • Key Sizes = Block Size <p>HMAC-SHA2-256:</p> <ul style="list-style-type: none"> • • Key Sizes < Block Size • • Key Sizes > Block Size • • Key Sizes = Block Size <p>HMAC-SHA2-384:</p> <ul style="list-style-type: none"> • Key Sizes < Block Size • Key Sizes > Block Size • Key Sizes = Block Size <p>Prerequisite: SHS #4011</p>	<p>Microsoft Surface Hub Virtual TPM Implementations #3271 Version 10.0.15063.674</p>
<p>HMAC-SHA-1:</p> <ul style="list-style-type: none"> • Key Sizes < Block Size • Key Sizes > Block Size • Key Sizes = Block Size <p>HMAC-SHA2-256:</p> <ul style="list-style-type: none"> • Key Sizes < Block Size • Key Sizes > Block Size • Key Sizes = Block Size <p>HMAC-SHA2-384:</p> <ul style="list-style-type: none"> • Key Sizes < Block Size • Key Sizes > Block Size • Key Sizes = Block Size <p>Prerequisite: SHS #4009</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); Virtual TPM Implementations #3270 Version 10.0.16299</p>

<p>MODES / STATES /</p> <ul style="list-style-type: none"> KEY SIZES 	<p>ALGORITHM IMPLEMENTATION AND CERTIFICATE #</p>
<p>HMAC-SHA-1:</p> <ul style="list-style-type: none"> Key Sizes < Block Size Key Sizes > Block Size Key Sizes = Block Size <p>HMAC-SHA2-256:</p> <ul style="list-style-type: none"> Key Sizes < Block Size Key Sizes > Block Size Key Sizes = Block Size <p>HMAC-SHA2-384:</p> <ul style="list-style-type: none"> Key Sizes < Block Size Key Sizes > Block Size Key Sizes = Block Size <p>HMAC-SHA2-512:</p> <ul style="list-style-type: none"> Key Sizes < Block Size Key Sizes > Block Size Key Sizes = Block Size <p>Prerequisite: SHS #4011</p>	<p>Microsoft Surface Hub SymCrypt Cryptographic Implementations #3269 Version 10.0.15063.674</p>
<p>HMAC-SHA-1:</p> <ul style="list-style-type: none"> Key Sizes < Block Size Key Sizes > Block Size Key Sizes = Block Size <p>HMAC-SHA2-256:</p> <ul style="list-style-type: none"> Key Sizes < Block Size Key Sizes > Block Size Key Sizes = Block Size <p>HMAC-SHA2-384:</p> <ul style="list-style-type: none"> Key Sizes < Block Size Key Sizes > Block Size Key Sizes = Block Size <p>HMAC-SHA2-512:</p> <ul style="list-style-type: none"> Key Sizes < Block Size Key Sizes > Block Size Key Sizes = Block Size <p>Prerequisite: SHS #4010</p>	<p>Windows 10 Mobile (version 1709) SymCrypt Cryptographic Implementations #3268 Version 10.0.15254</p>

MODES / STATES / • KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>HMAC-SHA-1:</p> <ul style="list-style-type: none"> • Key Sizes < Block Size • Key Sizes > Block Size • Key Sizes = Block Size <p>HMAC-SHA2-256:</p> <ul style="list-style-type: none"> • Key Sizes < Block Size • Key Sizes > Block Size • Key Sizes = Block Size <p>HMAC-SHA2-384:</p> <ul style="list-style-type: none"> • Key Sizes < Block Size • Key Sizes > Block Size • Key Sizes = Block Size <p>HMAC-SHA2-512:</p> <ul style="list-style-type: none"> • Key Sizes < Block Size • Key Sizes > Block Size • Key Sizes = Block Size <p>Prerequisite: SHS #4009</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); SymCrypt Cryptographic Implementations #3267</p> <p>Version 10.0.16299</p>
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 3790</p> <p>HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 3790</p> <p>HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 3790</p>	<p>Windows 10 Creators Update (version 1703) Pro, Enterprise, Education Virtual TPM Implementations #3062</p> <p>Version 10.0.15063</p>
<p>HMAC-SHA1(Key Sizes Ranges Tested: KSBS) SHS validation number 3790</p> <p>HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 3790</p> <p>HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 3790</p> <p>HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS validation number 3790</p>	<p>Windows 10 Creators Update (version 1703) Home, Pro, Enterprise, Education, Windows 10 S, Windows 10 Mobile SymCrypt Cryptographic Implementations #3061</p> <p>Version 10.0.15063</p>
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 3652</p> <p>HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 3652</p> <p>HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 3652</p> <p>HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHSvalidation number 3652</p>	<p>Windows Embedded Compact Enhanced Cryptographic Provider (RSAENH) #2946</p> <p>Version 7.00.2872</p>

MODES / STATES / • KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p> HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 3651 </p> <p> HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 3651 </p> <p> HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 3651 </p> <p> HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS validation number 3651 </p>	<p> Windows Embedded Compact Enhanced Cryptographic Provider (RSAENH) #2945 Version 8.00.6246 </p>
<p> HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 3649 </p> <p> HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 3649 </p> <p> HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 3649 </p> <p> HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS validation number 3649 </p>	<p> Windows Embedded Compact Cryptographic Primitives Library (bcrypt.dll) #2943 Version 7.00.2872 </p>
<p> HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 3648 </p> <p> HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 3648 </p> <p> HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 3648 </p> <p> HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS validation number 3648 </p>	<p> Windows Embedded Compact Cryptographic Primitives Library (bcrypt.dll) #2942 Version 8.00.6246 </p>
<p> HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 3347 </p> <p> HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 3347 </p> <p> HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 3347 </p>	<p> Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, and Surface Pro 3 w/ Windows 10 Anniversary Update Virtual TPM Implementations #2661 Version 10.0.14393 </p>
<p> HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 3347 </p> <p> HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 3347 </p> <p> HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 3347 </p> <p> HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS validation number 3347 </p>	<p> Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, Surface Pro 3 and Surface 3 w/ Windows 10 Anniversary Update; Microsoft Lumia 950 and Lumia 650 w/ Windows 10 Mobile Anniversary Update SymCrypt Cryptographic Implementations #2651 Version 10.0.14393 </p>

<p>MODES / STATES /</p> <ul style="list-style-type: none"> KEY SIZES 	<p>ALGORITHM IMPLEMENTATION AND CERTIFICATE #</p>
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 3047</p> <p>HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 3047</p> <p>HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 3047</p> <p>HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS validation number 3047</p>	<p>Microsoft Windows 10 November 2015 Update; Microsoft Surface Book, Surface Pro 4, Surface Pro 3, Surface 3, Surface Pro 2, and Surface Pro w/ Windows 10 November 2015 Update; Windows 10 Mobile for Microsoft Lumia 950 and Microsoft Lumia 635; Windows 10 for Microsoft Surface Hub 84" and Surface Hub 55" SymCrypt Cryptographic Implementations #2381</p> <p>Version 10.0.10586</p>
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 2886</p> <p>HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 2886</p> <p>HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 2886</p> <p>HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS validation number 2886</p>	<p>Microsoft Windows 10, Microsoft Surface Pro 3 with Windows 10, Microsoft Surface 3 with Windows 10, Microsoft Surface Pro 2 with Windows 10, Microsoft Surface Pro with Windows 10 SymCrypt Cryptographic Implementations #2233</p> <p>Version 10.0.10240</p>
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 2373</p> <p>HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 2373</p> <p>HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 2373</p> <p>HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS validation number 2373</p>	<p>Windows Storage Server 2012 R2, Microsoft Windows RT 8.1, Microsoft Surface with Windows RT 8.1, Microsoft Surface Pro with Windows 8.1, Microsoft Surface 2, Microsoft Surface Pro 2, Microsoft Surface Pro 3, Microsoft Windows Phone 8.1, Microsoft Windows Embedded 8.1 Industry, and Microsoft StorSimple 8100 SymCrypt Cryptographic Implementations #1773</p> <p>Version 6.3.9600</p>
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 2764</p> <p>HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 2764</p> <p>HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 2764</p> <p>HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS validation number 2764</p>	<p>Windows CE and Windows Mobile, and Windows Embedded Handheld Enhanced Cryptographic Provider (RSAENH) #2122</p> <p>Version 5.2.29344</p>
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KS) #1902</p> <p>HMAC-SHA256 (Key Size Ranges Tested: KS) #1902</p>	<p>Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 BitLocker  Cryptographic Implementations #1347</p>

MODES / STATES / • KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p> HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS#1902 </p> <p> HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS#1902 </p> <p> HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS#1902 </p> <p> HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS#1902 </p>	<p> Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 Enhanced Cryptographic Provider (RSAENH) #1346 </p>
<p> HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS#1903 </p> <p> HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS#1903 </p> <p> HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS#1903 </p> <p> HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS#1903 </p>	<p> Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 Next Generation Symmetric Cryptographic Algorithms Implementations (SYMCRYPT) #1345 </p>
<p> HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 1773 </p> <p> HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 1773 </p> <p> Tinker HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 1773 </p> <p> HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS validation number 1773 </p>	<p> Windows Embedded Compact 7 Cryptographic Primitives Library (bcrypt.dll), #1364 </p>
<p> HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 1774 </p> <p> HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 1774 </p> <p> HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 1774 </p> <p> HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS validation number 1774 </p>	<p> Windows Embedded Compact 7 Enhanced Cryptographic Provider (RSAENH) #1227 </p>
<p> HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 1081 </p> <p> HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 1081 </p> <p> HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 1081 </p> <p> HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS validation number 1081 </p>	<p> Windows Server 2008 R2 and SP1 CNG algorithms #686 Windows 7 and SP1 CNG algorithms #677 </p> <p> Windows Server 2008 R2 Enhanced Cryptographic Provider (RSAENH) #687 </p> <p> Windows 7 Enhanced Cryptographic Provider (RSAENH) #673 </p>

MODES / STATES / • KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>HMAC-SHA1(Key Sizes Ranges Tested: KSvalidation number 1081)</p> <p>HMAC-SHA256 (Key Size Ranges Tested: KSvalidation number 1081)</p>	<p>Windows 7 and SP1 and Windows Server 2008 R2 and SP1 BitLocker Algorithm Implementations #675</p>
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 816</p> <p>HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 816</p> <p>HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 816</p> <p>HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS validation number 816</p>	<p>Windows Server 2003 SP2 Enhanced Cryptographic Provider (RSAENH) #452</p>
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KSvalidation number 753)</p> <p>HMAC-SHA256 (Key Size Ranges Tested: KSvalidation number 753)</p>	<p>Windows Vista Ultimate SP1 and Windows Server 2008 BitLocker Algorithm Implementations #415</p>
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 753</p> <p>HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 753</p> <p>HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 753</p> <p>HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS validation number 753</p>	<p>Windows Server 2008 Enhanced Cryptographic Provider (RSAENH) #408</p> <p>Windows Vista Enhanced Cryptographic Provider (RSAENH) #407</p>
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KSBS)SHS validation number 618</p> <p>HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 618</p> <p>HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 618</p> <p>HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS validation number 618</p>	<p>Windows Vista Enhanced Cryptographic Provider (RSAENH) #297</p>
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 785</p>	<p>Windows XP Professional SP3 Kernel Mode Cryptographic Module (fips.sys) #429</p> <p>Windows XP, vendor-affirmed</p>

MODES / STATES / • KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 783</p> <p>HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 783</p> <p>HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 783</p> <p>HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS validation number 783</p>	<p>Windows XP Professional SP3 Enhanced Cryptographic Provider (RSAENH) #428</p>
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 613</p> <p>HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 613</p> <p>HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 613</p> <p>HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS validation number 613</p>	<p>Windows Server 2003 SP2 Enhanced Cryptographic Provider (RSAENH) #289</p>
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 610</p>	<p>Windows Server 2003 SP2 Kernel Mode Cryptographic Module (fips.sys) #287</p>
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 753</p> <p>HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 753</p> <p>HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 753</p> <p>HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS validation number 753</p>	<p>Windows Server 2008 CNG algorithms #413</p> <p>Windows Vista Ultimate SP1 CNG algorithms #412</p>
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KS) validation number 737</p> <p>HMAC-SHA256 (Key Size Ranges Tested: KS) validation number 737</p>	<p>Windows Vista Ultimate BitLocker Drive Encryption #386</p>
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 618</p> <p>HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 618</p> <p>HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 618</p> <p>HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS validation number 618</p>	<p>Windows Vista CNG algorithms #298</p>

MODES / STATES / • KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 589</p> <p>HMAC-SHA256 (Key Size Ranges Tested: KSBS)SHS validation number 589</p> <p>HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 589</p> <p>HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS validation number 589</p>	<p>Windows CE 6.0 and Windows CE 6.0 R2 and Windows Mobile Enhanced Cryptographic Provider (RSAENH) #267</p>
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 578</p> <p>HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 578</p> <p>HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 578</p> <p>HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS validation number 578</p>	<p>Windows CE and Windows Mobile 6.0 and Windows Mobil 6.5 Enhanced Cryptographic Provider (RSAENH) #260</p>
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KS) validation number 495</p> <p>HMAC-SHA256 (Key Size Ranges Tested: KS) validation number 495</p>	<p>Windows Vista BitLocker Drive Encryption #199</p>
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 364</p>	<p>Windows Server 2003 SP1 Enhanced Cryptographic Provider (RSAENH) #99</p> <p>Windows XP, vendor-affirmed</p>
<p>HMAC-SHA1 (Key Sizes Ranges Tested: KSBS) SHS validation number 305</p> <p>HMAC-SHA256 (Key Size Ranges Tested: KSBS) SHS validation number 305</p> <p>HMAC-SHA384 (Key Size Ranges Tested: KSBS) SHS validation number 305</p> <p>HMAC-SHA512 (Key Size Ranges Tested: KSBS) SHS validation number 305</p>	<p>Windows CE 5.00 and Windows CE 5.01 Enhanced Cryptographic Provider (RSAENH) #31</p>

Key Agreement Scheme (KAS)

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>KAS ECC: Functions: Domain Parameter Generation, Domain Parameter Validation, Full Public Key Validation, Key Pair Generation, Public Key Regeneration Schemes: Full Unified:</p> <ul style="list-style-type: none"> • Key Agreement Roles: Initiator, Responder • KDFs: Concatenation • Parameter Sets: <p>EC:</p> <ul style="list-style-type: none"> • Curve: P-256 • SHA: SHA-256 • MAC: HMAC <p>ED:</p> <ul style="list-style-type: none"> • Curve: P-384 • SHA: SHA-384 • MAC: HMAC <p>Prerequisite: SHS #4011, ECDSA #1253, DRBG #1734</p>	<p>Microsoft Surface Hub Virtual TPM Implementations #150 Version 10.0.15063.674</p>
<p>KAS ECC: Functions: Domain Parameter Generation, Domain Parameter Validation, Full Public Key Validation, Key Pair Generation, Public Key Regeneration Schemes: Full Unified:</p> <ul style="list-style-type: none"> • Key Agreement Roles: Initiator, Responder • KDFs: Concatenation • Parameter Sets: <p>EC:</p> <ul style="list-style-type: none"> • Curve: P-256 • SHA: SHA-256 • MAC: HMAC <p>ED:</p> <ul style="list-style-type: none"> • Curve: P-384 • SHA: SHA-384 • MAC: HMAC <p>Prerequisite: SHS #4009, ECDSA #1252, DRBG #1733</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); Virtual TPM Implementations #149 Version 10.0.16299</p>
<p>KAS ECC: Functions: Domain Parameter Generation, Domain Parameter Validation, Key Pair Generation, Partial Public Key Validation, Public Key Regeneration Schemes: Ephemeral Unified:</p> <ul style="list-style-type: none"> • Key Agreement Roles: Initiator, Responder • KDFs: Concatenation • Parameter Sets: <p>EC:</p> <ul style="list-style-type: none"> • Curve: P-256 • SHA: SHA-256 • MAC: HMAC 	<p>Microsoft Surface Hub SymCrypt Cryptographic Implementations #148 Version 10.0.15063.674</p>

ED:
M0DES / STATES / KEY SIZES

- Curve: P-384
- SHA: SHA-384
- MAC: HMAC

EE:

- Curve: P-521
- SHA: SHA-512
- MAC: HMAC

One-Pass DH:

- Key Agreement Roles: Initiator, Responder
- Parameter Sets:

EC:

- Curve: P-256
- SHA: SHA-256
- MAC: HMAC

ED:

- Curve: P-384
- SHA: SHA-384
- MAC: HMAC

EE:

- Curve: P-521
- SHA: SHA-512
- MAC: HMAC

Static Unified:

- Key Agreement Roles: Initiator, Responder
- Parameter Sets:

EC:

- Curve: P-256
- SHA: SHA-256
- MAC: HMAC

ED:

- Curve: P-384
- SHA: SHA-384
- MAC: HMAC

EE:

- Curve: P-521
- SHA: SHA-512
- MAC: HMAC

Prerequisite: SHS [#4011](#), ECDSA [#1250](#), DRBG [#1732](#)

KAS FFC:

Functions: Domain Parameter Generation, Domain Parameter Validation, Key Pair Generation, Partial Public Key Validation

Schemes:

dhEphem:

- Key Agreement Roles: Initiator, Responder
- Parameter Sets:

FB:

- SHA: SHA-256
- MAC: HMAC

FC:

- SHA: SHA-256
- MAC: HMAC

ALGORITHM IMPLEMENTATION AND CERTIFICATE #

dhOneFlow: MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<ul style="list-style-type: none"> • Key Agreement Roles: Initiator, Responder • Parameter Sets: <p>FB:</p> <ul style="list-style-type: none"> • SHA: SHA-256 • MAC: HMAC <p>FC</p> <ul style="list-style-type: none"> • SHA: SHA-256 • MAC: HMAC <p>dhStatic:</p> <ul style="list-style-type: none"> • Key Agreement Roles: Initiator, Responder • Parameter Sets: <p>FB:</p> <ul style="list-style-type: none"> • SHA: SHA-256 • MAC: HMAC <p>FC:</p> <ul style="list-style-type: none"> • SHA: SHA-256 • MAC: HMAC <p>Prerequisite: SHS #4011, DSA #1303, DRBG #1732</p>	
<p>KAS ECC: Functions: Domain Parameter Generation, Domain Parameter Validation, Key Pair Generation, Partial Public Key Validation, Public Key Regeneration</p> <p>Schemes:</p> <p>Ephemeral Unified:</p> <ul style="list-style-type: none"> • Key Agreement Roles: Initiator, Responder • KDFs: Concatenation • Parameter Sets: <p>EC:</p> <ul style="list-style-type: none"> • Curve: P-256 • SHA: SHA-256 • MAC: HMA <p>ED:</p> <ul style="list-style-type: none"> • Curve: P-384 • SHA: SHA-384 • MAC: HMAC <p>EE:</p> <ul style="list-style-type: none"> • Curve: P-521 • SHA: SHA-512 • MAC: HMAC <p>One-Pass DH:</p> <ul style="list-style-type: none"> • Key Agreement Roles: Initiator, Responder • Parameter Sets: <p>EC:</p> <ul style="list-style-type: none"> • Curve: P-256 • SHA: SHA-256 • MAC: HMAC <p>ED:</p> <ul style="list-style-type: none"> • Curve: P-384 • SHA: SHA-384 • MAC: HMAC <p>EE:</p>	<p>Windows 10 Mobile (version 1709) SymCrypt Cryptographic Implementations #147 Version 10.0.15254</p>

- Curve: P-521
- MODES / STATES / KEY SIZES
- SHA: SHA-512

ALGORITHM IMPLEMENTATION AND CERTIFICATE #

- MAC: HMAC

Static Unified:

- Key Agreement Roles: Initiator, Responder
- Parameter Sets:

EC:

- Curve: P-256
- SHA: SHA-256
- MAC: HMAC

ED:

- Curve: P-384
- SHA: SHA-384
- MAC: HMAC

EE:

- Curve: P-521
- SHA: SHA-512
- MAC: HMAC

Prerequisite: SHS [#4010](#), ECDSA [#1249](#), DRBG [#1731](#)

KAS FFC:

Functions: Domain Parameter Generation, Domain Parameter Validation, Key Pair Generation, Partial Public Key Validation

Schemes:

dhEphem:

- Key Agreement Roles: Initiator, Responder
- Parameter Sets:

FB:

- SHA: SHA-256
- MAC: HMAC

FC:

- SHA: SHA-256
- MAC: HMAC

dhOneFlow:

- Key Agreement Roles: Initiator, Responder
- Parameter Sets:

FB:

- SHA: SHA-256
- MAC: HMAC

FC

- SHA: SHA-256
- MAC: HMAC

dhStatic:

- Key Agreement Roles: Initiator, Responder
- Parameter Sets:

FB:

- SHA: SHA-256
- MAC: HMAC

FC:

- SHA: SHA-256
- MAC: HMAC

Prerequisite: SHS [#4010](#), DSA [#1302](#), DRBG [#1731](#)

KAS ECC:
MODES / STATES / KEY SIZES

Windows 10 Home, Pro, Enterprise, Education, Windows 10
ALGORITHM IMPLEMENTATION AND CERTIFICATE #
S Fall Creators Update, Windows Server, Windows Server

Functions: Domain Parameter Generation, Domain
Parameter Validation, Key Pair Generation, Partial Public
Key Validation, Public Key Regeneration

Datacenter (version 1709); SymCrypt Cryptographic
Implementations #146

Version 10.0.16299

Schemes:

Ephemeral Unified:

- Key Agreement Roles: Initiator, Responder
- KDFs: Concatenation
- Parameter Sets:

EC:

- Curve: P-256
- SHA: SHA-256
- MAC: HMAC

ED:

- Curve: P-384
- SHA: SHA-384
- MAC: HMAC

EE:

- Curve: P-521
- SHA: SHA-512
- MAC: HMAC

One-Pass DH:

- Key Agreement Roles: Initiator, Responder
- Parameter Sets: EC:
- Curve: P-256
- SHA: SHA-256
- MAC: HMAC

ED

- Curve: P-384
- SHA: SHA-384
- MAC: HMAC

EE:

- Curve: P-521
- SHA: SHA-512
- MAC: HMAC

Static Unified:

- Key Agreement Roles: Initiator, Responder
- Parameter Sets:

EC:

- Curve: P-256
- SHA: SHA-256
- MAC: HMAC

ED:

- Curve: P-384
- SHA: SHA-384
- MAC: HMAC

EE:

- Curve: P-521
- SHA: SHA-512
- MAC: HMAC

Prerequisite: SHS #4009, ECDSA #1246, DRBG #1730

KAS FFC:

Functions: Domain Parameter Generation, Domain
Parameter Validation, Key Pair Generation, Partial Public

Key Validation MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>Schemes:</p> <p>dhEphem:</p> <ul style="list-style-type: none"> • Key Agreement Roles: Initiator, Responder • Parameter Sets: <p>FB:</p> <ul style="list-style-type: none"> • SHA: SHA-256 • MAC: HMAC <p>FC:</p> <ul style="list-style-type: none"> • SHA: SHA-256 • MAC: HMAC <p>dhOneFlow:</p> <ul style="list-style-type: none"> • Key Agreement Roles: Initiator, Responder • Parameter Sets: <p>FB:</p> <ul style="list-style-type: none"> • SHA: SHA-256 • MAC: HMAC <p>FC:</p> <ul style="list-style-type: none"> • SHA: SHA-256 • MAC: HMAC <p>dhStatic:</p> <ul style="list-style-type: none"> • Key Agreement Roles: Initiator, Responder • Parameter Sets: <p>FB:</p> <ul style="list-style-type: none"> • SHA: SHA-256 • MAC: HMAC <p>FC:</p> <ul style="list-style-type: none"> • SHA: SHA-256 • MAC: HMAC <p>Prerequisite: SHS #4009, DSA #1301, DRBG #1730</p>	
<p>ECC: (FUNCTIONS INCLUDED IN IMPLEMENTATION: DPG DPV KPG Full Validation Key Regeneration) SCHEMES [FullUnified (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC)]</p> <p>SHS validation number 3790</p> <p>DSA validation number 1135</p> <p>DRBG validation number 1556</p>	<p>Windows 10 Creators Update (version 1703) Pro, Enterprise, Education Virtual TPM Implementations #128</p> <p>Version 10.0.15063</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FFC: (FUNCTIONS INCLUDED IN IMPLEMENTATION: DPG DPV KPG Partial Validation)</p> <p>SCHEMES [dhEphem (KARole(s): Initiator / Responder) (FB: SHA256) (FC: SHA256)]</p> <p>[dhOneFlow (FB: SHA256) (FC: SHA256)]</p> <p>[dhStatic (No_KC < KARole(s): Initiator / Responder>) (FB: SHA256 HMAC) (FC: SHA256 HMAC)]</p> <p>SHS validation number 3790</p> <p>DSA validation number 1223</p> <p>DRBG validation number 1555ECC: (FUNCTIONS INCLUDED IN IMPLEMENTATION: DPG DPV KPG Partial Validation) SCHEMES [EphemeralUnified (No_KC < KARole(s): Initiator / Responder>) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512)))]</p> <p>[OnePassDH (No_KC < KARole(s): Initiator / Responder>) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512))]</p> <p>[StaticUnified (No_KC < KARole(s): Initiator / Responder>) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512))]</p> <p>SHS validation number 3790</p> <p>ECDSA validation number 1133DRBG validation number 1555</p>	<p>Windows 10 Creators Update (version 1703) Home, Pro, Enterprise, Education, Windows 10 S, Windows 10 Mobile SymCrypt Cryptographic Implementations #127</p> <p>Version 10.0.15063</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FFC: (FUNCTIONS INCLUDED IN IMPLEMENTATION: DPG DPV KPG Partial Validation)</p> <p>SCHEMES [dhEphem (KARole(s): Initiator / Responder) (FB: SHA256) (FC: SHA256)]</p> <p>[dhOneFlow (KARole(s): Initiator / Responder) (FB: SHA256) (FC: SHA256)] [dhStatic (No_KC < KARole(s): Initiator / Responder>) (FB: SHA256 HMAC) (FC: SHA256 HMAC)]</p> <p>SHS validation number 3649</p> <p>DSA validation number 1188</p> <p>DRBG validation number 1430</p> <p>ECC: (FUNCTIONS INCLUDED IN IMPLEMENTATION: DPG DPV KPG Partial Validation Key Regeneration)</p> <p>SCHEMES [EphemeralUnified (No_KC < KARole(s): Initiator / Responder>) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512)))]</p> <p>[OnePassDH (No_KC < KARole(s): Initiator / Responder>) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512))]</p> <p>[StaticUnified (No_KC < KARole(s): Initiator / Responder>) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512))]</p>	<p>Windows Embedded Compact Cryptographic Primitives Library (bcrypt.dll) #115</p> <p>Version 7.00.2872</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FFC: (FUNCTIONS INCLUDED IN IMPLEMENTATION: DPG DPV KPG Partial Validation)</p> <p>SCHEMES [dhEphem (KARole(s): Initiator / Responder) (FB: SHA256) (FC: SHA256)]</p> <p>[dhHybridOneFlow (No_KC < KARole(s): Initiator / Responder>) (**FB:**SHA256 HMAC) (FC: SHA256 HMAC)]</p> <p>[dhStatic (No_KC < KARole(s): Initiator / Responder>) (**FB:**SHA256 HMAC) (FC: SHA256 HMAC)]</p> <p>SHS validation number 3648</p> <p>DSA validation number 1187</p> <p>DRBG validation number 1429</p> <p>ECC: (FUNCTIONS INCLUDED IN IMPLEMENTATION: DPG DPV KPG Partial Validation Key Regeneration)</p> <p>SCHEMES [EphemeralUnified (No_KC) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512)))]</p> <p>[OnePassDH (No_KC < KARole(s): Initiator / Responder>) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512))]</p> <p>[StaticUnified (No_KC < KARole(s): Initiator / Responder>) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512))]</p> <p>SHS validation number 3648</p> <p>ECDSA validation number 1072</p> <p>DRBG validation number 1429</p>	<p>Windows Embedded Compact Cryptographic Primitives Library (bcrypt.dll) #114</p> <p>Version 8.00.6246</p>
<p>ECC: (FUNCTIONS INCLUDED IN IMPLEMENTATION: DPG DPV KPG Full Validation Key Regeneration)</p> <p>SCHEMES [FullUnified (No_KC < KARole(s): Initiator / Responder > < KDF: CONCAT >) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC)]</p> <p>SHS validation number 3347 ECDSA validation number 920 DRBG validation number 1222</p>	<p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, and Surface Pro 3 w/ Windows 10 Anniversary Update Virtual TPM Implementations #93</p> <p>Version 10.0.14393</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FFC: (FUNCTIONS INCLUDED IN IMPLEMENTATION: DPG DPV KPG Partial Validation)</p> <p>SCHEMES [dhEphem (KARole(s): Initiator / Responder) (FB: SHA256) (FC: SHA256)]</p> <p>[dhOneFlow (KARole(s): Initiator / Responder) (FB: SHA256) (FC: SHA256)] [dhStatic (No_KC < KARole(s): Initiator / Responder >) (FB: SHA256 HMAC) (FC: SHA256 HMAC)]</p> <p>SHS validation number 3347 DSA validation number 1098 DRBG validation number 1217</p> <p>ECC: (FUNCTIONS INCLUDED IN IMPLEMENTATION: DPG DPV KPG Partial Validation Key Regeneration)</p> <p>SCHEMES [EphemeralUnified (No_KC < KARole(s): Initiator / Responder >) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512)))]</p> <p>[OnePassDH (No_KC < KARole(s): Initiator / Responder >) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512)))]</p> <p>[StaticUnified (No_KC < KARole(s): Initiator / Responder >) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512)))]</p> <p>SHS validation number 3347 DSA validation number 1098 ECDSA validation number 911 DRBG validation number 1217 HMAC validation number 2651</p>	<p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, Surface Pro 3 and Surface 3 w/ Windows 10 Anniversary Update; Microsoft Lumia 950 and Lumia 650 w/ Windows 10 Mobile Anniversary Update Cryptography Next Generation (CNG) Implementations #92 Version 10.0.14393</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FFC: (FUNCTIONS INCLUDED IN IMPLEMENTATION: DPG DPV KPG Partial Validation) SCHEMES [dhEphem (KARole(s): Initiator / Responder)(FB: SHA256) (FC: SHA256)] [dhOneFlow (KARole(s): Initiator / Responder) (FB: SHA256) (FC: SHA256)] [dhStatic (No_KC < KARole(s): Initiator / Responder >) (FB: SHA256 HMAC) (FC: SHA256 HMAC)]</p> <p>SHS validation number 3047 DSA validation number 1024 DRBG validation number 955</p> <p>ECC: (FUNCTIONS INCLUDED IN IMPLEMENTATION: DPG DPV KPG Partial Validation Key Regeneration) SCHEMES [EphemeralUnified (No_KC < KARole(s): Initiator / Responder >) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512)))]</p> <p>[OnePassDH (No_KC < KARole(s): Initiator / Responder >) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512))]</p> <p>[StaticUnified (No_KC < KARole(s): Initiator / Responder >) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512))]</p> <p>SHS validation number 3047 ECDSA validation number 760 DRBG validation number 955</p>	<p>Microsoft Windows 10 November 2015 Update; Microsoft Surface Book, Surface Pro 4, Surface Pro 3, Surface 3, Surface Pro 2, and Surface Pro w/ Windows 10 November 2015 Update; Windows 10 Mobile for Microsoft Lumia 950 and Microsoft Lumia 635; Windows 10 for Microsoft Surface Hub and Surface Hub Cryptography Next Generation (CNG) Implementations #72</p> <p>Version 10.0.10586</p>
<p>FFC: (FUNCTIONS INCLUDED IN IMPLEMENTATION: DPG DPV KPG Partial Validation) SCHEMES [dhEphem (KARole(s): Initiator / Responder)(FB: SHA256) (FC: SHA256)] [dhOneFlow (KARole(s): Initiator / Responder) (FB: SHA256) (FC: SHA256)] [dhStatic (No_KC < KARole(s): Initiator / Responder >) (FB: SHA256 HMAC) (FC: SHA256 HMAC)]</p> <p>SHS validation number 2886 DSA validation number 983 DRBG validation number 868</p> <p>ECC: (FUNCTIONS INCLUDED IN IMPLEMENTATION: DPG DPV KPG Partial Validation Key Regeneration) SCHEMES [EphemeralUnified (No_KC < KARole(s): Initiator / Responder >) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512)))]</p> <p>[OnePassDH (No_KC < KARole(s): Initiator / Responder >) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512))]</p> <p>[StaticUnified (No_KC < KARole(s): Initiator / Responder >) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512))]</p> <p>SHS validation number 2886 ECDSA validation number 706 DRBG validation number 868</p>	<p>Microsoft Windows 10, Microsoft Surface Pro 3 with Windows 10, Microsoft Surface 3 with Windows 10, Microsoft Surface Pro 2 with Windows 10, Microsoft Surface Pro with Windows 10 Cryptography Next Generation (CNG) Implementations #64</p> <p>Version 10.0.10240</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FFC: (FUNCTIONS INCLUDED IN IMPLEMENTATION: DPG DPV KPG Partial Validation) SCHEMES [dhEphem (KARole(s): Initiator / Responder)(FB: SHA256) (FC: SHA256)] [dhOneFlow (KARole(s): Initiator / Responder) (FB: SHA256) (FC: SHA256)] [dhStatic (No_KC < KARole(s): Initiator / Responder >) (FB: SHA256 HMAC) (FC: SHA256 HMAC)]</p> <p>SHS validation number 2373 DSA validation number 855 DRBG validation number 489</p> <p>ECC: (FUNCTIONS INCLUDED IN IMPLEMENTATION: DPG DPV KPG Partial Validation Key Regeneration) SCHEMES [EphemeralUnified (No_KC < KARole(s): Initiator / Responder >) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512)))]</p> <p>[OnePassDH (No_KC < KARole(s): Initiator / Responder >) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512))]</p> <p>[StaticUnified (No_KC < KARole(s): Initiator / Responder >) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512))]</p> <p>SHS validation number 2373 ECDSA validation number 505 DRBG validation number 489</p>	<p>Windows Storage Server 2012 R2, Microsoft Windows RT 8.1, Microsoft Surface with Windows RT 8.1, Microsoft Surface Pro with Windows 8.1, Microsoft Surface 2, Microsoft Surface Pro 2, Microsoft Surface Pro 3, Microsoft Windows Phone 8.1, Microsoft Windows Embedded 8.1 Industry, and Microsoft StorSimple 8100 Cryptography Next Generation Cryptographic Implementations #47</p> <p>Version 6.3.9600</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FFC: (FUNCTIONS INCLUDED IN IMPLEMENTATION: DPG DPV KPG Partial Validation) SCHEMES [dhEphem (KARole(s): Initiator / Responder) (FA: SHA256) (FB: SHA256) (FC: SHA256)]</p> <p>[dhOneFlow (KARole(s): Initiator / Responder) (FA: SHA256) (FB: SHA256) (FC: SHA256)]</p> <p>[dhStatic (No_KC < KARole(s): Initiator / Responder>) (FA: SHA256 HMAC) (FB: SHA256 HMAC) (FC: SHA256 HMAC)]</p> <p>SHS #1903 DSA validation number 687 DRBG #258</p> <p>ECC: (FUNCTIONS INCLUDED IN IMPLEMENTATION: DPG DPV KPG Partial Validation Key Regeneration) SCHEMES</p> <p>[EphemeralUnified (No_KC < KARole(s): Initiator / Responder>) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512)))]</p> <p>[OnePassDH(No_KC < KARole(s): Initiator / Responder>) (EC: P-256 SHA256) (ED: P-384 SHA384) (EE: P-521 (SHA512, HMAC_SHA512)))]</p> <p>[StaticUnified (No_KC < KARole(s): Initiator / Responder>) (EC: P-256 SHA256 HMAC) (ED: P-384 SHA384 HMAC) (EE: P-521 HMAC (SHA512, HMAC_SHA512)))]</p> <p>SHS #1903</p> <p>ECDSA validation number 341 DRBG #258</p>	<p>Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 Cryptography Next Generation (CNG) Implementations #36</p>
<p>KAS (SP 800–56A)</p> <ul style="list-style-type: none"> Key Agreement: Key establishment methodology provides 80 bits to 256 bits of encryption strength 	<p>Windows 7 and SP1, vendor-affirmed</p> <p>Windows Server 2008 R2 and SP1, vendor-affirmed</p>

SP 800-108 Key-Based Key Derivation Functions (KBKDF)

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>Counter:</p> <p>MACs: HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384</p> <p>MAC prerequisite: HMAC #3271</p> <ul style="list-style-type: none"> Counter Location: Before Fixed Data R Length: 32 (bits) SPs used to generate K: SP 800-56A, SP 800-90A <p>K prerequisite: DRBG #1734, KAS #150</p>	<p>Microsoft Surface Hub Virtual TPM Implementations #161</p> <p>Version 10.0.15063.674</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>Counter:</p> <p>MACs: HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384</p> <p>MAC prerequisite: HMAC #3270</p> <ul style="list-style-type: none"> Counter Location: Before Fixed Data R Length: 32 (bits) SPs used to generate K: SP 800-56A, SP 800-90A <p>K prerequisite: DRBG #1733, KAS #149</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); Virtual TPM Implementations #160</p> <p>Version 10.0.16299</p>
<p>Counter:</p> <p>MACs: CMAC-AES-128, CMAC-AES-192, CMAC-AES-256, HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512</p> <p>MAC prerequisite: AES #4902, HMAC #3269</p> <ul style="list-style-type: none"> Counter Location: Before Fixed Data R Length: 32 (bits) SPs used to generate K: SP 800-56A, SP 800-90A <p>K prerequisite: KAS #148</p>	<p>Microsoft Surface Hub Cryptography Next Generation (CNG) Implementations #159</p> <p>Version 10.0.15063.674</p>
<p>Counter:</p> <p>MACs: CMAC-AES-128, CMAC-AES-192, CMAC-AES-256, HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512</p> <p>MAC prerequisite: AES #4901, HMAC #3268</p> <ul style="list-style-type: none"> Counter Location: Before Fixed Data R Length: 32 (bits) SPs used to generate K: SP 800-56A, SP 800-90A <p>K prerequisite: KAS #147</p>	<p>Windows 10 Mobile (version 1709) Cryptography Next Generation (CNG) Implementations #158</p> <p>Version 10.0.15254</p>
<p>Counter:</p> <p>MACs: CMAC-AES-128, CMAC-AES-192, CMAC-AES-256, HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512</p> <p>MAC prerequisite: AES #4897, HMAC #3267</p> <ul style="list-style-type: none"> Counter Location: Before Fixed Data R Length: 32 (bits) SPs used to generate K: SP 800-56A, SP 800-90A <p>K prerequisite: KAS #146</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); Cryptography Next Generation (CNG) Implementations #157</p> <p>Version 10.0.16299</p>
<p>CTR_Mode: (Llength(Min0 Max0) MACSupported([HMACSHA1] [HMACSHA256] [HMACSHA384]) LocationCounter([BeforeFixedData]) rlength([32]))</p> <p>KAS validation number 128</p> <p>DRBG validation number 1556</p> <p>MAC validation number 3062</p>	<p>Windows 10 Creators Update (version 1703) Pro, Enterprise, Education Virtual TPM Implementations #141</p> <p>Version 10.0.15063</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>CTR_Mode: (Length(Min20 Max64) MACSupported([CMACAES128] [CMACAES192] [CMACAES256] [HMACSHA1] [HMACSHA256] [HMACSHA384] [HMACSHA512]) LocationCounter([BeforeFixedData]) rlength([32])) KAS validation number 127 AES validation number 4624 DRBG validation number 1555 MAC validation number 3061</p>	<p>Windows 10 Creators Update (version 1703) Home, Pro, Enterprise, Education, Windows 10 S, Windows 10 Mobile Cryptography Next Generation (CNG) Implementations #140 Version 10.0.15063</p>
<p>CTR_Mode: (Length(Min20 Max64) MACSupported([HMACSHA1] [HMACSHA256] [HMACSHA384]) LocationCounter([BeforeFixedData]) rlength([32])) KAS validation number 93 DRBG validation number 1222 MAC validation number 2661</p>	<p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, and Surface Pro 3 w/ Windows 10 Anniversary Update Virtual TPM Implementations #102 Version 10.0.14393</p>
<p>CTR_Mode: (Length(Min20 Max64) MACSupported([CMACAES128] [CMACAES192] [CMACAES256] [HMACSHA1] [HMACSHA256] [HMACSHA384] [HMACSHA512]) LocationCounter([BeforeFixedData]) rlength([32])) KAS validation number 92 AES validation number 4064 DRBG validation number 1217 MAC validation number 2651</p>	<p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, Surface Pro 3 and Surface 3 w/ Windows 10 Anniversary Update; Microsoft Lumia 950 and Lumia 650 w/ Windows 10 Mobile Anniversary Update Cryptography Next Generation (CNG) Implementations #101 Version 10.0.14393</p>
<p>CTR_Mode: (Length(Min20 Max64) MACSupported([CMACAES128] [CMACAES192] [CMACAES256] [HMACSHA1] [HMACSHA256] [HMACSHA384] [HMACSHA512]) LocationCounter([BeforeFixedData]) rlength([32])) KAS validation number 72 AES validation number 3629 DRBG validation number 955 MAC validation number 2381</p>	<p>Microsoft Windows 10 November 2015 Update; Microsoft Surface Book, Surface Pro 4, Surface Pro 3, Surface 3, Surface Pro 2, and Surface Pro w/ Windows 10 November 2015 Update; Windows 10 Mobile for Microsoft Lumia 950 and Microsoft Lumia 635; Windows 10 for Microsoft Surface Hub 84" and Surface Hub 55" Cryptography Next Generation (CNG) Implementations #72 Version 10.0.10586</p>
<p>CTR_Mode: (Length(Min20 Max64) MACSupported([CMACAES128] [CMACAES192] [CMACAES256] [HMACSHA1] [HMACSHA256] [HMACSHA384] [HMACSHA512]) LocationCounter([BeforeFixedData]) rlength([32])) KAS validation number 64 AES validation number 3497 RBG validation number 868 MAC validation number 2233</p>	<p>Microsoft Windows 10, Microsoft Surface Pro 3 with Windows 10, Microsoft Surface 3 with Windows 10, Microsoft Surface Pro 2 with Windows 10, Microsoft Surface Pro with Windows 10 Cryptography Next Generation (CNG) Implementations #66 Version 10.0.10240</p>
<p>CTR_Mode: (Length(Min0 Max0) MACSupported([HMACSHA1] [HMACSHA256] [HMACSHA512]) LocationCounter([BeforeFixedData]) rlength([32])) DRBG validation number 489 MAC validation number 1773</p>	<p>Windows Storage Server 2012 R2, Microsoft Windows RT 8.1, Microsoft Surface with Windows RT 8.1, Microsoft Surface Pro with Windows 8.1, Microsoft Surface 2, Microsoft Surface Pro 2, Microsoft Surface Pro 3, Microsoft Windows Phone 8.1, Microsoft Windows Embedded 8.1 Industry, and Microsoft StorSimple 8100 Cryptography Next Generation Cryptographic Implementations #30 Version 6.3.9600</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>CTR_Mode: (Llength(Min0 Max4) MACSupported([HMACSHA1] [HMACSHA256] [HMACSHA512]) LocationCounter([BeforeFixedData]) rlength([32])) DRBG #258 HMAC validation number 1345</p>	<p>Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 Cryptography Next Generation (CNG) Implementations #3</p>

Random Number Generator (RNG)

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS 186-2 General Purpose [(x-Original); (SHA-1)]</p>	<p>Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 Cryptography Next Generation (CNG) Implementations #1110</p>
<p>FIPS 186-2 [(x-Original); (SHA-1)]</p>	<p>Windows Embedded Compact 7 Enhanced Cryptographic Provider (RSAENH) #1060</p> <p>Windows CE 6.0 and Windows CE 6.0 R2 and Windows Mobile Enhanced Cryptographic Provider (RSAENH) #292</p> <p>Windows CE and Windows Mobile 6.0 and Windows Mobile 6.5 Enhanced Cryptographic Provider (RSAENH) #286</p> <p>Windows CE 5.00 and Windows CE 5.01 Enhanced Cryptographic Provider (RSAENH) #66</p>
<p>FIPS 186-2 [(x-Change Notice); (SHA-1)]; FIPS 186-2 General Purpose [(x-Change Notice); (SHA-1)]</p>	<p>Windows 7 and SP1 and Windows Server 2008 R2 and SP1 RNG Library #649</p> <p>Windows Vista Ultimate SP1 and Windows Server 2008 RNG Implementation #435</p> <p>Windows Vista RNG implementation #321</p>
<p>FIPS 186-2 General Purpose [(x-Change Notice); (SHA-1)]</p>	<p>Windows Server 2003 SP2 Enhanced Cryptographic Provider (RSAENH) #470</p> <p>Windows XP Professional SP3 Kernel Mode Cryptographic Module (fips.sys) #449</p> <p>Windows XP Professional SP3 Enhanced Cryptographic Provider (RSAENH) #447</p> <p>Windows Server 2003 SP2 Enhanced Cryptographic Provider (RSAENH) #316</p> <p>Windows Server 2003 SP2 Kernel Mode Cryptographic Module (fips.sys) #313</p>
<p>FIPS 186-2 [(x-Change Notice); (SHA-1)]</p>	<p>Windows XP Professional SP3 Enhanced DSS and Diffie- Hellman Cryptographic Provider (DSSENH) #448</p> <p>Windows Server 2003 SP2 Enhanced DSS and Diffie- Hellman Cryptographic Provider #314</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>RSA: 186-4:</p> <p>Signature Generation PKCS1.5:</p> <p>Mod 2048 SHA: SHA-1,</p> <ul style="list-style-type: none">• SHA-256,• SHA-384 <p>Signature Generation PSS:</p> <p>Mod 2048:</p> <ul style="list-style-type: none">• SHA-1: Salt Length: 160 (bits)• SHA-256: Salt Length: 256 (bits)• SHA-384: Salt Length: 384 (bits) <p>Signature Verification PKCS1.5:</p> <p>Mod 1024 SHA: SHA-1,</p> <ul style="list-style-type: none">• SHA-256,• SHA-384 <p>Mod 2048 SHA: SHA-1,</p> <ul style="list-style-type: none">• SHA-256,• SHA-384 <p>Signature Verification PSS:</p> <p>Mod 2048:</p> <ul style="list-style-type: none">• SHA-1: Salt Length: 160 (bits)• SHA-256: Salt Length: 256 (bits)• SHA-384: Salt Length: 384 (bits) <p>Mod 3072:</p> <ul style="list-style-type: none">• SHA-1: Salt Length: 160 (bits)• SHA-256: Salt Length: 256 (bits)• SHA-384: Salt Length: 384 (bits) <p>Prerequisite: SHS #4011, DRBG #1734</p>	<p>Microsoft Surface Hub Virtual TPM Implementations #2677 Version 10.0.15063.674</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>RSA: 186-4:</p> <p>Signature Generation PKCS1.5:</p> <p>Mod 2048 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384 <p>Signature Generation PSS:</p> <p>Mod 2048:</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 240 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) <p>Signature Verification PKCS1.5:</p> <p>Mod 1024 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384 <p>Mod 2048 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384 <p>Signature Verification PSS:</p> <p>Mod 1024</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) <p>Mod 2048:</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) <p>Prerequisite: SHS #4009, DRBG #1733</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (Version 1709); Virtual TPM Implementations #2676</p> <p>Version 10.0.16299</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>RSA: 186-4: Key Generation: Signature Verification PKCS1.5: Mod 1024 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Mod 2048 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256 • SHA-384, • SHA-512 <p>Mod 3072 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Prerequisite: SHS #4011, DRBG #1732</p>	<p>Microsoft Surface Hub RSA32 Algorithm Implementations #2675 Version 10.0.15063.674</p>
<p>RSA: 186-4: Signature Verification PKCS1.5: Mod 1024 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Mod 2048 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Mod 3072 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Prerequisite: SHS #4009, DRBG #1730</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); RSA32 Algorithm Implementations #2674 Version 10.0.16299</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>RSA: 186-4: Signature Verification PKCS1.5: Mod 1024 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Mod 2048 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Mod 3072 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Prerequisite: SHS #4010, DRBG #1731</p>	<p>Windows 10 Mobile (version 1709) RSA32 Algorithm Implementations #2673 Version 10.0.15254</p>
<p>RSA: 186-4: Key Generation:</p> <ul style="list-style-type: none"> • Public Key Exponent: Fixed (10001) • Provable Primes with Conditions: <p>Mod lengths: 2048, 3072 (bits) Primality Tests: C.3 Signature Generation PKCS1.5: Mod 2048 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Mod 3072 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Signature Generation PSS: Mod 2048:</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 512 (bits) <p>Mod 3072</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 512 (bits) <p>Signature Verification PKCS1.5</p>	<p>Microsoft Surface Hub MsBignum Cryptographic Implementations #2672 Version 10.0.15063.674</p>

Mod 1024 SHA: MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 Mod 2048 SHA: <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 Mod 3072 SHA: <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 Signature Verification PSS Mod 1024 <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 496 (bits) Mod 2048: <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 512 (bits) Mod 3072: <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 512 (bits) Prerequisite: SHS #4011 , DRBG #1732	
RSA: 186-4: Key Generation: Probable Random Primes: Mod lengths: 2048, 3072 (bits) Primality Tests: C 2 Signature Generation PKCS1.5: Mod 2048 SHA: <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 Mod 3072 SHA: <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 Signature Generation PSS: Mod 2048:	Microsoft Surface Hub SymCrypt Cryptographic Implementations #2671 Version 10.0.15063.674

<p>• SHA-1: Salt Length: 160 (bits)</p> <p>MODES / STATES / KEY SIZES</p> <p>• SHA-256: Salt Length: 256 (bits)</p>	<p>ALGORITHM IMPLEMENTATION AND CERTIFICATE #</p>
<p>• SHA-384: Salt Length: 384 (bits)</p> <p>• SHA-512: Salt Length: 512 (bits)</p> <p>Mod 3072:</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 512 (bits) <p>Signature Verification PKCS1.5:</p> <p>Mod 1024 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Mod 2048 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Mod 3072 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Signature Verification PSS:</p> <p>Mod 1024:</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 496 (bits) <p>Mod 2048:</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 512 (bits) <p>Mod 3072:</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 512 (bits) <p>Prerequisite: SHS #4011, DRBG #1732</p>	
<p>RSA:</p> <p>186-4:</p> <p>Key Generation:</p> <p>Probable Random Primes:</p> <p>Mod lengths: 2048, 3072 (bits)</p> <p>Primality Tests: C.2</p> <p>Signature Generation PKCS1.5:</p> <p>Mod 2048 SHA:</p>	<p>Windows 10 Mobile (version 1709) SymCrypt Cryptographic Implementations #2670</p> <p>Version 10.0.15254</p>

<ul style="list-style-type: none"> • SHA-1, MODES / STATES / KEY SIZES • SHA-256, 	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<ul style="list-style-type: none"> • SHA-384, • SHA-512 <p>Mod 3072 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Signature Generation PSS:</p> <p>Mod 2048:</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 512 (bits) <p>Mod 3072:</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 512 (bits) <p>Signature Verification PKCS1.5:</p> <p>Mod 1024 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Mod 2048 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Mod 3072 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Signature Verification PSS:</p> <p>Mod 1024:</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 496 (bits) <p>Mod 2048</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 512 (bits) <p>Mod 3072:</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 512 (bits) <p>Prerequisite: SHS #4010, DRBG #1731</p>	
RSA:	Windows 10 Mobile (version 1709) MsBignum

Key Generation:

Public Key Exponent: Fixed (10001)

Provable Primes with Conditions:

Mod lengths: 2048, 3072 (bits)

Primality Tests: C.3

Signature Generation PKCS1.5:

Mod 2048 SHA:

- SHA-1,
- SHA-256,
- SHA-384,
- SHA-512

Mod 3072 SHA:

- SHA-1,
- SHA-256,
- SHA-384,
- SHA-512

Signature Generation PSS:

Mod 2048:

- SHA-1: Salt Length: 160 (bits)
- SHA-256: Salt Length: 256 (bits)
- SHA-384: Salt Length: 384 (bits)
- SHA-512: Salt Length: 512 (bits)

Mod 3072

- SHA-1: Salt Length: 160 (bits)
- SHA-256: Salt Length: 256 (bits)
- SHA-384: Salt Length: 384 (bits)
- SHA-512: Salt Length: 512 (bits)

Signature Verification PKCS1.5

Mod 1024 SHA:

- SHA-1,
- SHA-256,
- SHA-384,
- SHA-512

Mod 2048 SHA:

- SHA-1,
- SHA-256,
- SHA-384,
- SHA-512

Mod 3072 SHA:

- SHA-1,
- SHA-256,
- SHA-384,
- SHA-512

Signature Verification PSS:

Mod 1024

- SHA-1: Salt Length: 160 (bits)
- SHA-256: Salt Length: 256 (bits)
- SHA-384: Salt Length: 384 (bits)
- SHA-512: Salt Length: 496 (bits)

Mod 2048:

- SHA-1: Salt Length: 160 (bits)

<ul style="list-style-type: none"> • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) 	<p>ALGORITHM IMPLEMENTATION AND CERTIFICATE #</p>
<ul style="list-style-type: none"> • SHA-512: Salt Length: 512 (bits) <p>Mod 3072:</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 512 (bits) <p>Prerequisite: SHS #4010, DRBG #1731</p>	
<p>186-4:</p> <p>Key Generation:</p> <p>Public Key Exponent: Fixed (10001)</p> <p>Provable Primes with Conditions:</p> <p>Mod lengths: 2048, 3072 (bits)</p> <p>Primality Tests: C.3</p> <p>Signature Generation PKCS1.5:</p> <p>Mod 2048 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Mod 3072 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Signature Generation PSS:</p> <p>Mod 2048:</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 512 (bits) <p>Mod 3072</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 512 (bits) <p>Signature Verification PKCS1.5</p> <p>Mod 1024 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Mod 2048 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Mod 3072 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, 	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); MsBignum Cryptographic Implementations #2668</p> <p>Version 10.0.16299</p>

<ul style="list-style-type: none"> • SHA-384, MODES / STATES / KEY SIZES • SHA-512 	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>Signature Verification PSS:</p> <p>Mod 1024</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 496 (bits) <p>Mod 2048:</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 512 (bits) <p>Mod 3072:</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 512 (bits) <p>Prerequisite: SHS #4009, DRBG #1730</p>	
<p>186-4:</p> <p>Key Generation</p> <p>Probable Random Primes:</p> <p>Mod lengths: 2048, 3072 (bits)</p> <p>Primality Tests: C.2</p> <p>Signature Generation PKCS1.5:</p> <p>Mod 2048 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-51 <p>Mod 3072 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Signature Generation PSS:</p> <p>Mod 2048:</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 512 (bits) <p>Mod 3072:</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 512 (bits) <p>Signature Verification PKCS1.5:</p> <p>Mod 1024 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); SymCrypt Cryptographic Implementations #2667</p> <p>Version 10.0.16299</p>

Mod 2048 SHA: MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Mod 3072 SHA:</p> <ul style="list-style-type: none"> • SHA-1, • SHA-256, • SHA-384, • SHA-512 <p>Signature Verification PSS:</p> <p>Mod 1024:</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 496 (bits) <p>Mod 2048:</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 512 (bits) <p>Mod 3072:</p> <ul style="list-style-type: none"> • SHA-1: Salt Length: 160 (bits) • SHA-256: Salt Length: 256 (bits) • SHA-384: Salt Length: 384 (bits) • SHA-512: Salt Length: 512 (bits) <p>Prerequisite: SHS #4009, DRBG #1730</p>	
<p>FIPS186-4: ALG[RSASSA-PKCS1_V1_5] SIG(gen) (2048 SHA(1, 256, 384)) SIG(gen) with SHA-1 affirmed for use with protocols only.</p> <p>SIG(ver) (1024 SHA(1, 256, 384)) (2048 SHA(1, 256, 384))</p> <p>[RSASSA-PSS]:</p> <p>Sig(Gen): (2048 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48))) **SIG(gen) with SHA-1 affirmed for use with protocols only.</p> <p>**SIG(ver): (1024 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48))) (2048 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48)))</p> <p>SHA validation number 3790</p>	<p>Windows 10 Creators Update (version 1703) Pro, Enterprise, Education Virtual TPM Implementations #2524</p> <p>Version 10.0.15063</p>
<p>FIPS186-4: ALG[RSASSA-PKCS1_V1_5] SIG(Ver) (1024 SHA(1, 256, 384, 512)) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512))</p> <p>SHA validation number 3790</p>	<p>Windows 10 Creators Update (version 1703) Home, Pro, Enterprise, Education, Windows 10 S, Windows 10 Mobile RSA32 Algorithm Implementations #2523</p> <p>Version 10.0.15063</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-4:</p> <p>186-4KEY(gen):</p> <p>FIPS186-4_Fixed_e (10001);</p> <p>PGM(ProbPrimeCondition): 2048, 3072 PPTT:(C.3)</p> <p>ALG[RSASSA-PKCS1_V1_5] SIG(gen) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512))SIG(gen) with SHA-1 affirmed for use with protocols only.</p> <p>SIG(ver) (1024 SHA(1, 256, 384, 512)) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512))</p> <p>[RSASSA-PSS]:</p> <p>Sig(Gen): (2048 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))) (3072 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))) **SIG(gen) with SHA-1 affirmed for use with protocols only.</p> <p>**SIG(ver): (1024 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(62))) (2048 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))) (3072 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64)))</p> <p>SHA validation number 3790</p> <p>DRBG: validation number 1555</p>	<p>Windows 10 Creators Update (version 1703) Home, Pro, Enterprise, Education, Windows 10 S, Windows 10 Mobile MsBignum Cryptographic Implementations #2522 Version 10.0.15063</p>
<p>FIPS186-4:</p> <p>186-4KEY(gen):PGM(ProbRandom: (2048, 3072) PPTT:(C.2)</p> <p>ALG[RSASSA-PKCS1_V1_5]** SIG(gen) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512)) SIG(gen) with SHA-1 affirmed for use with protocols only.</p> <p>SIG(ver) (1024 SHA(1, 256, 384, 512)) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512))</p> <p>[RSASSA-PSS]:</p> <p>Sig(Gen): (2048 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))) (3072 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))) **SIG(gen) with SHA-1 affirmed for use with protocols only.</p> <p>**SIG(ver): (1024 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(62))) (2048 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))) (3072 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64)))</p> <p>SHA validation number 3790</p>	<p>Windows 10 Creators Update (version 1703) Home, Pro, Enterprise, Education, Windows 10 S, Windows 10 Mobile SymCrypt Cryptographic Implementations #2521 Version 10.0.15063</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-2: ALG[ANSIX9.31]: SIG(ver); 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 3652 ALG[RSASSA-PKCS1_V1_5]: SIG(gen) 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256validation number 3652, • SHA-384validation number 3652, • SHA-512validation number 3652, SIG(ver): 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 3652, • SHA-256validation number 3652, • SHA-384validation number 3652, • SHA-512validation number 3652 <p>FIPS186-4: ALG[ANSIX9.31] Sig(Gen): (2048 SHA(1)) (3072 SHA(1))SIG(gen) with SHA-1 affirmed for use with protocols only.SIG(ver): (1024 SHA(1)) (2048 SHA(1)) (3072 SHA(1)) ALG[RSASSA-PKCS1_V1_5] SIG(gen) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512)) **SIG(gen) with SHA-1 affirmed for use with protocols only **SIG(ver) (1024 SHA(1, 256, 384, 512)) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512)) SHA validation number 3652</p>	<p>Windows Embedded Compact Enhanced Cryptographic Provider (RSAENH) #2415 Version 7.00.2872</p>
<p>FIPS186-2: ALG[ANSIX9.31]: SIG(ver); 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 3651 ALG[RSASSA-PKCS1_V1_5]: SIG(gen) 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256validation number 3651, • SHA-384validation number 3651, • SHA-512validation number 3651SIG(ver): 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 3651, • SHA-256validation number 3651, • SHA-384validation number 3651, • SHA-512validation number 3651 <p>FIPS186-4: ALG[ANSIX9.31] Sig(Gen): (2048 SHA(1)) (3072 SHA(1))SIG(gen) with SHA-1 affirmed for use with protocols only. SIG(ver): (1024 SHA(1)) (2048 SHA(1)) (3072 SHA(1)) ALG[RSASSA-PKCS1_V1_5] SIG(gen) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512)) **SIG(gen) with SHA-1 affirmed for use with protocols only. **SIG(ver) (1024 SHA(1, 256, 384, 512)) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512)) SHA validation number 3651</p>	<p>Windows Embedded Compact Enhanced Cryptographic Provider (RSAENH) #2414 Version 8.00.6246</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-2: ALG[RSASSA-PKCS1_V1_5]: SIG(gen) 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256validation number 3649, • SHA-384validation number 3649, • SHA-512validation number 3649SIG(ver): 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 3649, • SHA-256validation number 3649, • SHA-384validation number 3649, • SHA-512validation number 3649 <p>FIPS186-4: 186-4KEY(gen):</p> <p>FIPS186-4_Fixed_e (10001); PGM(ProbRandom: (2048, 3072) PPTT:(C.2) ALG[RSASSA-PKCS1_V1_5] SIG(gen) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512)) **SIG(gen) with SHA-1 affirmed for use with protocols only.</p> <p>**SIG(ver) (1024 SHA(1, 256, 384, 512)) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512))</p> <p>SHA validation number 3649</p> <p>DRBG: validation number 1430</p>	<p>Windows Embedded Compact Cryptographic Primitives Library (bcrypt.dll) #2412 Version 7.00.2872</p>
<p>FIPS186-2: ALG[RSASSA-PKCS1_V1_5]: SIG(gen) 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256validation number 3648, • SHA-384validation number 3648, • SHA-512validation number 3648, SIG(ver): 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 3648, • SHA-256validation number 3648, • SHA-384validation number 3648, • SHA-512validation number 3648 <p>FIPS186-4: 186-4KEY(gen):</p> <p>FIPS186-4_Fixed_e (10001); PGM(ProbRandom: (2048, 3072) PPTT:(C.2) ALG[RSASSA-PKCS1_V1_5] SIG(gen) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512)) **SIG(gen) with SHA-1 affirmed for use with protocols only.</p> <p>**SIG(ver) (1024 SHA(1, 256, 384, 512)) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512))</p> <p>SHA validation number 3648</p> <p>DRBG: validation number 1429</p>	<p>Windows Embedded Compact Cryptographic Primitives Library (bcrypt.dll) #2411 Version 8.00.6246</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-4: ALG[RSASSA-PKCS1_V1_5] SIG(gen) (2048 SHA(1, 256, 384)) SIG(gen) with SHA-1 affirmed for use with protocols only.SIG(Ver) (1024 SHA(1, 256, 384)) (2048 SHA(1, 256, 384))</p> <p>[RSASSA-PSS]: Sig(Gen): (2048 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48))) SIG(gen) with SHA-1 affirmed for use with protocols only.Sig(Ver): (1024 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48))) (2048 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48)))</p> <p>SHA validation number 3347</p>	<p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, and Surface Pro 3 w/ Windows 10 Anniversary Update Virtual TPM Implementations #2206 Version 10.0.14393</p>
<p>FIPS186-4: 186-4KEY(gen): FIPS186-4_Fixed_e (10001) PGM(ProbPrimeCondition): 2048, 3072 PPTT:(C.3)</p> <p>SHA validation number 3347 DRBG: validation number 1217</p>	<p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, Surface Pro 3 and Surface 3 w/ Windows 10 Anniversary Update; Microsoft Lumia 950 and Lumia 650 w/ Windows 10 Mobile Anniversary Update RSA Key Generation Implementation #2195 Version 10.0.14393</p>
<p>FIPS186-4: ALG[RSASSA-PKCS1_V1_5] SIG(Ver) (1024 SHA(1, 256, 384, 512)) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512))</p> <p>SHA validation number 3346</p>	<p>soft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, Surface Pro 3 and Surface 3 w/ Windows 10 Anniversary Update; Microsoft Lumia 950 and Lumia 650 w/ Windows 10 Mobile Anniversary Update RSA32 Algorithm Implementations #2194 Version 10.0.14393</p>
<p>FIPS186-4: ALG[RSASSA-PKCS1_V1_5] SIG(gen) (2048 SHA(256, 384, 512)) (3072 SHA(256, 384, 512))</p> <p>SIG(Ver) (1024 SHA(1, 256, 384, 512)) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512))</p> <p>SHA validation number 3347 DRBG: validation number 1217</p>	<p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, Surface Pro 3 and Surface 3 w/ Windows 10 Anniversary Update; Microsoft Lumia 950 and Lumia 650 w/ Windows 10 Mobile Anniversary Update MsBignum Cryptographic Implementations #2193 Version 10.0.14393</p>
<p>FIPS186-4: [RSASSA-PSS]: Sig(Gen): (2048 SHA(256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))) (3072 SHA(256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64)))</p> <p>Sig(Ver): (1024 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(62))) (2048 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))) (3072 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64)))</p> <p>SHA validation number 3347 DRBG: validation number 1217</p>	<p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, Surface Pro 3 and Surface 3 w/ Windows 10 Anniversary Update; Microsoft Lumia 950 and Lumia 650 w/ Windows 10 Mobile Anniversary Update Cryptography Next Generation (CNG) Implementations #2192 Version 10.0.14393</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-4: 186-4KEY(gen) : FIPS186-4_Fixed_e (10001); PGM(ProbPrimeCondition): 2048, 3072 PPTT:(C.3) SHA validation number 3047 DRBG: validation number 955</p>	<p>Microsoft Windows 10 November 2015 Update; Microsoft Surface Book, Surface Pro 4, Surface Pro 3, Surface 3, Surface Pro 2, and Surface Pro w/ Windows 10 November 2015 Update; Windows 10 Mobile for Microsoft Lumia 950 and Microsoft Lumia 635; Windows 10 for Microsoft Surface Hub 84" and Surface Hub 55" RSA Key Generation Implementation #1889 Version 10.0.10586</p>
<p>FIPS186-4: ALG[RSASSA-PKCS1_V1_5] SIG(Ver) (1024 SHA(1, 256, 384, 512)) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512)) SHA validation number 3048</p>	<p>Microsoft Windows 10 November 2015 Update; Microsoft Surface Book, Surface Pro 4, Surface Pro 3, Surface 3, Surface Pro 2, and Surface Pro w/ Windows 10 November 2015 Update; Windows 10 Mobile for Microsoft Lumia 950 and Microsoft Lumia 635; Windows 10 for Microsoft Surface Hub and Surface Hub RSA32 Algorithm Implementations #1871 Version 10.0.10586</p>
<p>FIPS186-4: ALG[RSASSA-PKCS1_V1_5] SIG(gen) (2048 SHA(256, 384, 512)) (3072 SHA(256, 384, 512)) SIG(Ver) (1024 SHA(1, 256, 384, 512)) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512)) SHA validation number 3047</p>	<p>Microsoft Windows 10 November 2015 Update; Microsoft Surface Book, Surface Pro 4, Surface Pro 3, Surface 3, Surface Pro 2, and Surface Pro w/ Windows 10 November 2015 Update; Windows 10 Mobile for Microsoft Lumia 950 and Microsoft Lumia 635; Windows 10 for Microsoft Surface Hub and Surface Hub MsBignum Cryptographic Implementations #1888 Version 10.0.10586</p>
<p>FIPS186-4: [RSASSA-PSS]: Sig(Gen): (2048 SHA(256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))) (3072 SHA(256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))) Sig(Ver): (1024 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(62))) (2048 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))) (3072 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))) SHA validation number 3047</p>	<p>Microsoft Windows 10 November 2015 Update; Microsoft Surface Book, Surface Pro 4, Surface Pro 3, Surface 3, Surface Pro 2, and Surface Pro w/ Windows 10 November 2015 Update; Windows 10 Mobile for Microsoft Lumia 950 and Microsoft Lumia 635; Windows 10 for Microsoft Surface Hub and Surface Hub Cryptography Next Generation (CNG) Implementations #1887 Version 10.0.10586</p>
<p>FIPS186-4: 186-4KEY(gen): FIPS186-4_Fixed_e (10001);PGM(ProbPrimeCondition): 2048, 3072 PPTT:(C.3) SHA validation number 2886 DRBG: validation number 868</p>	<p>Microsoft Windows 10, Microsoft Surface Pro 3 with Windows 10, Microsoft Surface 3 with Windows 10, Microsoft Surface Pro 2 with Windows 10, Microsoft Surface Pro with Windows 10 RSA Key Generation Implementation #1798 Version 10.0.10240</p>
<p>FIPS186-4: ALG[RSASSA-PKCS1_V1_5] SIG(Ver) (1024 SHA(1, 256, 384, 512)) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512)) SHA validation number 2871</p>	<p>Microsoft Windows 10, Microsoft Surface Pro 3 with Windows 10, Microsoft Surface 3 with Windows 10, Microsoft Surface Pro 2 with Windows 10, Microsoft Surface Pro with Windows 10 RSA32 Algorithm Implementations #1784 Version 10.0.10240</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-4: ALG[RSASSA-PKCS1_V1_5] SIG(Ver) (1024 SHA(1, 256, 384, 512)) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512))</p> <p>SHA validation number 2871</p>	<p>Microsoft Windows 10, Microsoft Surface Pro 3 with Windows 10, Microsoft Surface 3 with Windows 10, Microsoft Surface Pro 2 with Windows 10, Microsoft Surface Pro with Windows 10 MsBignum Cryptographic Implementations #1783</p> <p>Version 10.0.10240</p>
<p>FIPS186-4: [RSASSA-PSS]: Sig(Gen): (2048 SHA(256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))) (3072 SHA(256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))), Sig(Ver): (2048 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))) (3072 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64)))</p> <p>SHA validation number 2886</p>	<p>Microsoft Windows 10, Microsoft Surface Pro 3 with Windows 10, Microsoft Surface 3 with Windows 10, Microsoft Surface Pro 2 with Windows 10, Microsoft Surface Pro with Windows 10 Cryptography Next Generation (CNG) Implementations #1802</p> <p>Version 10.0.10240</p>
<p>FIPS186-4: 186-4KEY(gen): FIPS186-4_Fixed_e; PGM(ProbPrimeCondition): 2048, 3072 PPTT:(C.3)</p> <p>SHA validation number 2373 DRBG: validation number 489</p>	<p>Microsoft Windows 8.1, Microsoft Windows Server 2012 R2, Microsoft Windows Storage Server 2012 R2, Microsoft Windows RT 8.1, Microsoft Surface with Windows RT 8.1, Microsoft Surface Pro with Windows 8.1, Microsoft Surface 2, Microsoft Surface Pro 2, Microsoft Surface Pro 3, Microsoft Windows Phone 8.1, Microsoft Windows Embedded 8.1 Industry, and Microsoft StorSimple 8100 RSA Key Generation Implementation #1487</p> <p>Version 6.3.9600</p>
<p>FIPS186-4: ALG[RSASSA-PKCS1_V1_5] SIG(Ver) (1024 SHA(1, 256, 384, 512)) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512))</p> <p>SHA validation number 2373</p>	<p>Microsoft Windows 8.1, Microsoft Windows Server 2012 R2, Microsoft Windows Storage Server 2012 R2, Microsoft Windows RT 8.1, Microsoft Surface with Windows RT 8.1, Microsoft Surface Pro with Windows 8.1, Microsoft Surface 2, Microsoft Surface Pro 2, Microsoft Surface Pro 3, Microsoft Windows Phone 8.1, Microsoft Windows Embedded 8.1 Industry RSA32 Algorithm Implementations #1494</p> <p>Version 6.3.9600</p>
<p>FIPS186-4: ALG[RSASSA-PKCS1_V1_5] SIG(gen) (2048 SHA(256, 384, 512)) (3072 SHA(256, 384, 512)), SIG(Ver) (1024 SHA(1, 256, 384, 512)) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512))</p> <p>SHA validation number 2373</p>	<p>Microsoft Windows 8.1, Microsoft Windows Server 2012 R2, Microsoft Windows Storage Server 2012 R2, Microsoft Windows RT 8.1, Microsoft Surface with Windows RT 8.1, Microsoft Surface Pro with Windows 8.1, Microsoft Surface 2, Microsoft Surface Pro 2, Microsoft Surface Pro 3, Microsoft Windows Phone 8.1, Microsoft Windows Embedded 8.1 Industry, and Microsoft StorSimple 8100 MsBignum Cryptographic Implementations #1493</p> <p>Version 6.3.9600</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-4: [RSASSA-PSS]: Sig(Gen): (2048 SHA(256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))) (3072 SHA(256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))), Sig(Ver): (1024 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(62))) (2048 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64))) (3072 SHA(1 SaltLen(20), 256 SaltLen(32), 384 SaltLen(48), 512 SaltLen(64)))</p> <p>SHA validation number 2373</p>	<p>Windows Storage Server 2012 R2, Microsoft Windows RT 8.1, Microsoft Surface with Windows RT 8.1, Microsoft Surface Pro with Windows 8.1, Microsoft Surface 2, Microsoft Surface Pro 2, Microsoft Surface Pro 3, Microsoft Windows Phone 8.1, Microsoft Windows Embedded 8.1 Industry, and Microsoft StorSimple 8100 Cryptography Next Generation Cryptographic Implementations #1519</p> <p>Version 6.3.9600</p>
<p>FIPS186-4: ALG[RSASSA-PKCS1_V1_5] SIG(gen) (2048 SHA(256, 384, 512-256)) (3072 SHA(256, 384, 512-256)), SIG(Ver) (1024 SHA(1, 256, 384, 512-256)) (2048 SHA(1, 256, 384, 512-256)) (3072 SHA(1, 256, 384, 512-256))</p> <p>[RSASSA-PSS]: Sig(Gen): (2048 SHA(256, 384, 512)) (3072 SHA(256, 384, 512)), Sig(Ver): (1024 SHA(1, 256, 384, 512)) (2048 SHA(1, 256, 384, 512)) (3072 SHA(1, 256, 384, 512, 512)), SHA #1903</p> <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical RSA List validation number 1134.</p>	<p>Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 Cryptography Next Generation (CNG) Implementations #1134</p>
<p>FIPS186-4: 186-4KEY(gen): FIPS186-4_Fixed_e, FIPS186-4_Fixed_e_Value PGM(ProbPrimeCondition): 2048, 3072 PPTT:(C.3) SHA #1903 DRBG: #258</p>	<p>Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 RSA Key Generation Implementation #1133</p>
<p>FIPS186-2: ALG[ANSIX9.31]: Key(gen)(MOD: 2048, 3072, 4096 PubKey Values: 65537 DRBG: #258 ALG[RSASSA-PKCS1_V1_5]: SIG(gen) 2048, 3072, 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256#1902, • SHA-384#1902, • SHA-512#1902,, SIG(ver): 1024, 1536, 2048, 3072, 4096, SHS: SHA-1#1902, • SHA-256#1902, SHA-#1902, • SHA-512#1902, <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical RSA List validation number 1132.</p>	<p>Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 Enhanced Cryptographic Provider (RSAENH) #1132</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-2:ALG[ANSIX9.31]: SIG(ver): 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 1774</p> <p>ALG[RSASSA-PKCS1_V1_5]: SIG(gen) 2048, 3072, 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256validation number 1774, • SHA-384validation number 1774, • SHA-512validation number 1774,SIG(ver): 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 1774, • SHA-256validation number 1774, • SHA-384validation number 1774, • SHA-512validation number 1774, <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical RSA List validation number 1052.</p>	<p>Windows Embedded Compact 7 Enhanced Cryptographic Provider (RSAENH) #1052</p>
<p>FIPS186-2:</p> <p>ALG[ANSIX9.31]: Key(gen)(MOD: 2048, 3072, 4096 PubKey Values: 65537 DRBG: validation number 193</p> <p>ALG[RSASSA-PKCS1_V1_5]: SIG(gen) 2048, 3072, 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256validation number 1773, • SHA-384validation number 1773, • SHA-512validation number 1773,SIG(ver): 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 1773, • SHA-256validation number 1773, • SHA-384validation number 1773, • SHA-512validation number 1773, <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical RSA List validation number 1051.</p>	<p>Windows Embedded Compact Cryptographic Primitives Library (bcrypt.dll) #1051</p>
<p>FIPS186-2:</p> <p>ALG[RSASSA-PKCS1_V1_5]: SIG(gen) 2048, 3072, 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256validation number 1081, • SHA-384validation number 1081, • SHA-512validation number 1081,SIG(ver): 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 1081, • SHA-256validation number 1081, • SHA-384validation number 1081, • SHA-512validation number 1081, <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical RSA List validation number 568.</p>	<p>Windows Server 2008 R2 and SP1 Enhanced Cryptographic Provider (RSAENH) #568</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-2: ALG[RSASSA-PKCS1_V1_5]: SIG(gen) 2048, 3072, 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256validation number 1081, • SHA-384validation number 1081, • SHA-512validation number 1081, SIG(ver): 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 1081, • SHA-256validation number 1081, • SHA-384validation number 1081, • SHA-512validation number 1081, <p>ALG[RSASSA-PSS]: SIG(gen); 2048, 3072, 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256validation number 1081, • SHA-384validation number 1081, • SHA-512validation number 1081, SIG(ver); 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 1081, • SHA-256validation number 1081, • SHA-384validation number 1081, • SHA-512validation number 1081 <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical RSA List validation number 567. See Historical RSA List validation number 560.</p>	<p>Windows Server 2008 R2 and SP1 CNG algorithms #567 Windows 7 and SP1 CNG algorithms #560</p>
<p>FIPS186-2: ALG[ANSIX9.31]: Key(gen)(MOD: 2048, 3072, 4096 PubKey Values: 65537 DRBG: validation number 23</p> <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical RSA List validation number 559.</p>	<p>Windows 7 and SP1 and Server 2008 R2 and SP1 RSA Key Generation Implementation #559</p>
<p>FIPS186-2: ALG[RSASSA-PKCS1_V1_5]: SIG(gen) 2048, 3072, 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256validation number 1081, • SHA-384validation number 1081, • SHA-512validation number 1081, SIG(ver): 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 1081, • SHA-256validation number 1081, • SHA-384validation number 1081, • SHA-512validation number 1081, <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical RSA List validation number 557.</p>	<p>Windows 7 and SP1 Enhanced Cryptographic Provider (RSAENH) #557</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-2: ALG[ANSIX9.31]: ALG[RSASSA-PKCS1_V1_5]: SIG(gen) 2048, 3072, 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256validation number 816, • SHA-384validation number 816, • SHA-512validation number 816,SIG(ver): 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 816, • SHA-256validation number 816, • SHA-384validation number 816, • SHA-512validation number 816, <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical RSA List validation number 395.</p>	<p>Windows Server 2003 SP2 Enhanced Cryptographic Provider (RSAENH) #395</p>
<p>FIPS186-2: ALG[ANSIX9.31]: SIG(ver); 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 783 ALG[RSASSA-PKCS1_V1_5]: SIG(gen) 2048, 3072, 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256validation number 783, • SHA-384validation number 783, • SHA-512validation number 783, <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical RSA List validation number 371.</p>	<p>Windows XP Professional SP3 Enhanced Cryptographic Provider (RSAENH) #371</p>
<p>FIPS186-2: ALG[RSASSA-PKCS1_V1_5]: SIG(gen) 2048, 3072, 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256validation number 753, • SHA-384validation number 753, • SHA-512validation number 753, SIG(ver): 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 753, • SHA-256validation number 753, • SHA-384validation number 753, • SHA-512validation number 753, <p>ALG[RSASSA-PSS]: SIG(gen); 2048, 3072, 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256validation number 753, • SHA-384validation number 753, • SHA-512validation number 753, SIG(ver); 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 753, • SHA-256validation number 753, • SHA-384validation number 753, • SHA-512validation number 753 <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical RSA List validation number 358. See Historical RSA List validation number 357.</p>	<p>Windows Server 2008 CNG algorithms #358 Windows Vista SP1 CNG algorithms #357</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-2: ALG[ANSIX9.31]: SIG(ver); 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 753 ALG[RSASSA-PKCS1_V1_5]: SIG(gen) 2048, 3072, 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256validation number 753, • SHA-384validation number 753, • SHA-512validation number 753, SIG(ver): 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 753, • SHA-256validation number 753, • SHA-384validation number 753, • SHA-512validation number 753, <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical RSA List validation number 355. See Historical RSA List validation number 354.</p>	<p>Windows Server 2008 Enhanced Cryptographic Provider (RSAENH) #355 Windows Vista SP1 Enhanced Cryptographic Provider (RSAENH) #354</p>
<p>FIPS186-2: ALG[ANSIX9.31]: Key(gen)(MOD: 2048, 3072, 4096 PubKey Values: 65537</p> <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical RSA List validation number 353.</p>	<p>Windows Vista SP1 and Windows Server 2008 RSA Key Generation Implementation #353</p>
<p>FIPS186-2: ALG[ANSIX9.31]: Key(gen)(MOD: 2048, 3072, 4096 PubKey Values: 65537 RNG: validation number 321</p> <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical RSA List validation number 258.</p>	<p>Windows Vista RSA key generation implementation #258</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-2: ALG[RSASSA-PKCS1_V1_5]: SIG(gen) 2048, 3072, 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256validation number 618, • SHA-384validation number 618, • SHA-512validation number 618,SIG(ver): 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 618, • SHA-256validation number 618, • SHA-384validation number 618, • SHA-512validation number 618, <p>ALG[RSASSA-PSS]: SIG(gen); 2048, 3072, 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256validation number 618, • SHA-384validation number 618, • SHA-512validation number 618, SIG(ver); 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 618, • SHA-256validation number 618, • SHA-384validation number 618, • SHA-512validation number 618 <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical RSA List validation number 257.</p>	<p>Windows Vista CNG algorithms #257</p>
<p>FIPS186-2: ALG[RSASSA-PKCS1_V1_5]: SIG(gen) 2048, 3072, 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256validation number 618, • SHA-384validation number 618, • SHA-512validation number 618,, SIG(ver): 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 618, • SHA-256validation number 618, • SHA-384validation number 618, • SHA-512validation number 618, <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical RSA List validation number 255.</p>	<p>Windows Vista Enhanced Cryptographic Provider (RSAENH) #255</p>
<p>FIPS186-2: ALG[ANSIX9.31]: SIG(ver); 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 613 ALG[RSASSA-PKCS1_V1_5]: SIG(gen) 2048, 3072, 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256validation number 613, • SHA-384validation number 613, • SHA-512validation number 613, SIG(ver): 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 613, • SHA-256validation number 613, • SHA-384validation number 613, • SHA-512validation number 613, <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical RSA List validation number 245.</p>	<p>Windows Server 2003 SP2 Enhanced Cryptographic Provider (RSAENH) #245</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-2: ALG[ANSIX9.31]: SIG(ver); 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 589 ALG[RSASSA-PKCS1_V1_5]: SIG(gen) 2048, 3072, 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256validation number 589, • SHA-384validation number 589, • SHA-512validation number 589,, SIG(ver): 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 589, • SHA-256validation number 589, • SHA-384validation number 589, • SHA-512validation number 589, <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical RSA List validation number 230.</p>	<p>Windows CE 6.0 and Windows CE 6.0 R2 and Windows Mobile Enhanced Cryptographic Provider (RSAENH) #230</p>
<p>FIPS186-2: ALG[ANSIX9.31]: SIG(ver); 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 578 ALG[RSASSA-PKCS1_V1_5]: SIG(gen) 2048, 3072, 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256validation number 578, • SHA-384validation number 578, • SHA-512validation number 578,, SIG(ver): 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 578, • SHA-256validation number 578, • SHA-384validation number 578, • SHA-512validation number 578, <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical RSA List validation number 222.</p>	<p>Windows CE and Windows Mobile 6 and Windows Mobile 6.1 Enhanced Cryptographic Provider (RSAENH) #222</p>
<p>FIPS186-2: ALG[RSASSA-PKCS1_V1_5]:</p> <p>SIG(ver): 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 364</p> <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical RSA List validation number 81.</p>	<p>Windows Server 2003 SP1 Enhanced Cryptographic Provider (RSAENH) #81</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-2: ALG[ANSIX9.31]: SIG(ver); 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 305 ALG[RSASSA-PKCS1_V1_5]: SIG(gen) 2048, 3072, 4096, SHS:</p> <ul style="list-style-type: none"> • SHA-256validation number 305, • SHA-384validation number 305, • SHA-512validation number 305,, SIG(ver): 1024, 1536, 2048, 3072, 4096, SHS: SHA-1validation number 305, • SHA-256validation number 305, • SHA-384validation number 305, • SHA-512validation number 305, <p>Some of the previously validated components for this validation have been removed because they're now non-compliant per the SP800-131A transition. See Historical RSA List validation number 52.</p>	<p>Windows CE 5.00 and Windows CE 5.01 Enhanced Cryptographic Provider (RSAENH) #52</p>
<p>FIPS186-2::</p> <ul style="list-style-type: none"> • PKCS#1 v1.5, Signature generation, and verification • Mod sizes: 1024, 1536, 2048, 3072, 4096 • SHS: SHA-1/256/384/512 	<p>Windows XP, vendor-affirmed Windows 2000, vendor-affirmed</p>

Secure Hash Standard (SHS)

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>SHA-1: Supports Empty Message</p> <p>SHA-256: Supports Empty Message</p> <p>SHA-384: Supports Empty Message</p> <p>SHA-512: Supports Empty Message</p>	<p>Microsoft Surface Hub SymCrypt Cryptographic Implementations #4011 Version 10.0.15063.674</p>
<p>SHA-1: Supports Empty Message</p> <p>SHA-256: Supports Empty Message</p> <p>SHA-384: Supports Empty Message</p> <p>SHA-512: Supports Empty Message</p>	<p>Windows 10 Mobile (version 1709) SymCrypt Cryptographic Implementations #4010 Version 10.0.15254</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
SHA-1: Supports Empty Message SHA-256: Supports Empty Message SHA-384: Supports Empty Message SHA-512: Supports Empty Message	Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); SymCrypt Cryptographic Implementations #4009 Version 10.0.16299
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) • SHA-256 (BYTE-only) • SHA-384 (BYTE-only) • SHA-512 (BYTE-only) 	Windows 10 Creators Update (version 1703) Home, Pro, Enterprise, Education, Windows 10 S, Windows 10 Mobile SymCrypt Cryptographic Implementations #3790 Version 10.0.15063
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) • SHA-256 (BYTE-only) • SHA-384 (BYTE-only) • SHA-512 (BYTE-only) 	Windows Embedded Compact Enhanced Cryptographic Provider (RSAENH) #3652 Version 7.00.2872
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) • SHA-256 (BYTE-only) • SHA-384 (BYTE-only) • SHA-512 (BYTE-only) 	Windows Embedded Compact Enhanced Cryptographic Provider (RSAENH) #3651 Version 8.00.6246
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) • SHA-256 (BYTE-only) • SHA-384 (BYTE-only) • SHA-512 (BYTE-only) 	Windows Embedded Compact Cryptographic Primitives Library (bcrypt.dll) #3649 Version 7.00.2872
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) • SHA-256 (BYTE-only) • SHA-384 (BYTE-only) • SHA-512 (BYTE-only) 	Windows Embedded Compact Cryptographic Primitives Library (bcrypt.dll) #3648 Version 8.00.6246
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) • SHA-256 (BYTE-only) • SHA-384 (BYTE-only) • SHA-512 (BYTE-only) 	Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, Surface Pro 3 and Surface 3 w/ Windows 10 Anniversary Update; Microsoft Lumia 950 and Lumia 650 w/ Windows 10 Mobile Anniversary Update SymCrypt Cryptographic Implementations #3347 Version 10.0.14393
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) • SHA-256 (BYTE-only) • SHA-384 (BYTE-only) • SHA-512 (BYTE-only) 	Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, Surface Pro 3 and Surface 3 w/ Windows 10 Anniversary Update; Microsoft Lumia 950 and Lumia 650 w/ Windows 10 Mobile Anniversary Update RSA32 Algorithm Implementations #3346 Version 10.0.14393

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) • SHA-256 (BYTE-only) • SHA-384 (BYTE-only) • SHA-512 (BYTE-only) 	<p>Microsoft Windows 10 November 2015 Update; Microsoft Surface Book, Surface Pro 4, Surface Pro 3, Surface 3, Surface Pro 2, and Surface Pro w/ Windows 10 November 2015 Update; Windows 10 Mobile for Microsoft Lumia 950 and Microsoft Lumia 635; Windows 10 for Microsoft Surface Hub and Surface Hub RSA32 Algorithm Implementations #3048 Version 10.0.10586</p>
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) • SHA-256 (BYTE-only) • SHA-384 (BYTE-only) • SHA-512 (BYTE-only) 	<p>Microsoft Windows 10 November 2015 Update; Microsoft Surface Book, Surface Pro 4, Surface Pro 3, Surface 3, Surface Pro 2, and Surface Pro w/ Windows 10 November 2015 Update; Windows 10 Mobile for Microsoft Lumia 950 and Microsoft Lumia 635; Windows 10 for Microsoft Surface Hub and Surface Hub SymCrypt Cryptographic Implementations #3047 Version 10.0.10586</p>
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) • SHA-256 (BYTE-only) • SHA-384 (BYTE-only) • SHA-512 (BYTE-only) 	<p>Microsoft Windows 10, Microsoft Surface Pro 3 with Windows 10, Microsoft Surface 3 with Windows 10, Microsoft Surface Pro 2 with Windows 10, Microsoft Surface Pro with Windows 10 SymCrypt Cryptographic Implementations #2886 Version 10.0.10240</p>
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) • SHA-256 (BYTE-only) • SHA-384 (BYTE-only) • SHA-512 (BYTE-only) 	<p>Microsoft Windows 10, Microsoft Surface Pro 3 with Windows 10, Microsoft Surface 3 with Windows 10, Microsoft Surface Pro 2 with Windows 10, Microsoft Surface Pro with Windows 10 RSA32 Algorithm Implementations #2871 Version 10.0.10240</p>
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) • SHA-256 (BYTE-only) • SHA-384 (BYTE-only) • SHA-512 (BYTE-only) 	<p>Microsoft Windows 8.1, Microsoft Windows Server 2012 R2, Microsoft Windows Storage Server 2012 R2, Microsoft Windows RT 8.1, Microsoft Surface with Windows RT 8.1, Microsoft Surface Pro with Windows 8.1, Microsoft Surface 2, Microsoft Surface Pro 2, Microsoft Surface Pro 3, Microsoft Windows Phone 8.1, Microsoft Windows Embedded 8.1 Industry RSA32 Algorithm Implementations #2396 Version 6.3.9600</p>
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) • SHA-256 (BYTE-only) • SHA-384 (BYTE-only) • SHA-512 (BYTE-only) 	<p>Windows Storage Server 2012 R2, Microsoft Windows RT 8.1, Microsoft Surface with Windows RT 8.1, Microsoft Surface Pro with Windows 8.1, Microsoft Surface 2, Microsoft Surface Pro 2, Microsoft Surface Pro 3, Microsoft Windows Phone 8.1, Microsoft Windows Embedded 8.1 Industry, and Microsoft StorSimple 8100 SymCrypt Cryptographic Implementations #2373 Version 6.3.9600</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) • SHA-256 (BYTE-only) • SHA-384 (BYTE-only) • SHA-512 (BYTE-only) <p>Implementation does not support zero-length (null) messages.</p>	<p>Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 Next Generation Symmetric Cryptographic Algorithms Implementations (SYMCRYPT) #1903</p> <p>Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 Symmetric Algorithm Implementations (RSA32) #1902</p>
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) • SHA-256 (BYTE-only) • SHA-384 (BYTE-only) • SHA-512 (BYTE-only) 	<p>Windows Embedded Compact 7 Enhanced Cryptographic Provider (RSAENH) #1774</p> <p>Windows Embedded Compact 7 Cryptographic Primitives Library (bcrypt.dll) #1773</p>
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) • SHA-256 (BYTE-only) • SHA-384 (BYTE-only) • SHA-512 (BYTE-only) 	<p>Windows 7 and SP1 and Windows Server 2008 R2 and SP1 Symmetric Algorithm Implementation #1081</p> <p>Windows Server 2003 SP2 Enhanced Cryptographic Provider (RSAENH) #816</p>
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) 	<p>Windows XP Professional SP3 Kernel Mode Cryptographic Module (fips.sys) #785</p> <p>Windows XP Professional SP3 Enhanced DSS and Diffie-Hellman Cryptographic Provider (DSSENH) #784</p>
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) • SHA-256 (BYTE-only) • SHA-384 (BYTE-only) • SHA-512 (BYTE-only) 	<p>Windows XP Professional SP3 Enhanced Cryptographic Provider (RSAENH) #783</p>
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) • SHA-256 (BYTE-only) • SHA-384 (BYTE-only) • SHA-512 (BYTE-only) 	<p>Windows Vista SP1 and Windows Server 2008 Symmetric Algorithm Implementation #753</p> <p>Windows Vista Symmetric Algorithm Implementation #618</p>
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) • SHA-256 (BYTE-only) 	<p>Windows Vista BitLocker Drive Encryption #737</p> <p>Windows Vista Beta 2 BitLocker Drive Encryption #495</p>
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) • SHA-256 (BYTE-only) • SHA-384 (BYTE-only) • SHA-512 (BYTE-only) 	<p>Windows Server 2003 SP2 Enhanced Cryptographic Provider (RSAENH) #613</p> <p>Windows Server 2003 SP1 Enhanced Cryptographic Provider (RSAENH) #364</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) 	<p>Windows Server 2003 SP2 Enhanced DSS and Diffie-Hellman Cryptographic Provider #611</p> <p>Windows Server 2003 SP2 Kernel Mode Cryptographic Module (fips.sys) #610</p> <p>Windows Server 2003 SP1 Enhanced DSS and Diffie-Hellman Cryptographic Provider (DSSENH) #385</p> <p>Windows Server 2003 SP1 Kernel Mode Cryptographic Module (fips.sys) #371</p> <p>Windows Server 2003 Enhanced DSS and Diffie-Hellman Cryptographic Provider (DSSENH) #181</p> <p>Windows Server 2003 Kernel Mode Cryptographic Module (fips.sys) #177</p> <p>Windows Server 2003 Enhanced Cryptographic Provider (RSAENH) #176</p>
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) • SHA-256 (BYTE-only) • SHA-384 (BYTE-only) • SHA-512 (BYTE-only) 	<p>Windows CE 6.0 and Windows CE 6.0 R2 and Windows Mobile Enhanced Cryptographic Provider (RSAENH) #589</p> <p>Windows CE and Windows Mobile 6 and Windows Mobile 6.5 Enhanced Cryptographic Provider (RSAENH) #578</p> <p>Windows CE 5.00 and Windows CE 5.01 Enhanced Cryptographic Provider (RSAENH) #305</p>
<ul style="list-style-type: none"> • SHA-1 (BYTE-only) 	<p>Windows XP Microsoft Enhanced Cryptographic Provider #83</p> <p>Crypto Driver for Windows 2000 (fips.sys) #35</p> <p>Windows 2000 Microsoft Outlook Cryptographic Provider (EXCHCSPDLL) SR-1A (3821) #32</p> <p>Windows 2000 RSAENH.DLL #24</p> <p>Windows 2000 RSABASE.DLL #23</p> <p>Windows NT 4 SP6 RSAENH.DLL #21</p> <p>Windows NT 4 SP6 RSABASE.DLL #20</p>

Triple DES

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>TDES-CBC:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Keying Option: 1 <p>TDES-CFB64:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Keying Option: 1 <p>TDES-CFB8:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Keying Option: 1 <p>TDES-ECB:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Keying Option: 1 	<p>Microsoft Surface Hub SymCrypt Cryptographic Implementations #2558 Version 10.0.15063.674</p>
<p>TDES-CBC:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Keying Option: 1 <p>TDES-CFB64:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Keying Option: 1 <p>TDES-CFB8:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Keying Option: 1 <p>TDES-ECB:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Keying Option: 1 	<p>Windows 10 Mobile (version 1709) SymCrypt Cryptographic Implementations #2557 Version 10.0.15254</p>
<p>TDES-CBC:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Keying Option: 1 <p>TDES-CFB64:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Keying Option: 1 <p>TDES-CFB8:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Keying Option: 1 <p>TDES-ECB:</p> <ul style="list-style-type: none"> • Modes: Decrypt, Encrypt • Keying Option: 1 	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); SymCrypt Cryptographic Implementations #2556 Version 10.0.16299</p>
<p>TECB(KO 1 e/d); TCBC(KO 1 e/d); TCFB8(KO 1 e/d); TCFB64(KO 1 e/d)</p>	<p>Windows 10 Creators Update (version 1703) Home, Pro, Enterprise, Education, Windows 10 S, Windows 10 Mobile SymCrypt Cryptographic Implementations #2459 Version 10.0.15063</p>

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
TECB(KO 1 e/d);TCBC(KO 1 e/d)	Windows Embedded Compact Enhanced Cryptographic Provider (RSAENH) #2384 Version 8.00.6246
TECB(KO 1 e/d);TCBC(KO 1 e/d)	Windows Embedded Compact Enhanced Cryptographic Provider (RSAENH) #2383 Version 8.00.6246
TECB(KO 1 e/d);TCBC(KO 1 e/d);CTR (int only)	Windows Embedded Compact Cryptographic Primitives Library (bcrypt.dll) #2382 Version 7.00.2872
TECB(KO 1 e/d);TCBC(KO 1 e/d)	Windows Embedded Compact Cryptographic Primitives Library (bcrypt.dll) #2381 Version 8.00.6246
TECB(KO 1 e/d);TCBC(KO 1 e/d);TCFB8(KO 1 e/d);TCFB64(KO 1 e/d)	Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, Surface Pro 3 and Surface 3 w/ Windows 10 Anniversary Update; Microsoft Lumia 950 and Lumia 650 w/ Windows 10 Mobile Anniversary Update SymCrypt Cryptographic Implementations #2227 Version 10.0.14393
TECB(KO 1 e/d);TCBC(KO 1 e/d);TCFB8(KO 1 e/d);TCFB64(KO 1 e/d)	Microsoft Windows 10 November 2015 Update; Microsoft Surface Book, Surface Pro 4, Surface Pro 3, Surface 3, Surface Pro 2, and Surface Pro w/ Windows 10 November 2015 Update; Windows 10 Mobile for Microsoft Lumia 950 and Microsoft Lumia 635; Windows 10 for Microsoft Surface Hub and Surface Hub SymCrypt Cryptographic Implementations #2024 Version 10.0.10586
TECB(KO 1 e/d);TCBC(KO 1 e/d);TCFB8(KO 1 e/d);TCFB64(KO 1 e/d)	Microsoft Windows 10, Microsoft Surface Pro 3 with Windows 10, Microsoft Surface 3 with Windows 10, Microsoft Surface Pro 2 with Windows 10, Microsoft Surface Pro with Windows 10 SymCrypt Cryptographic Implementations #1969 Version 10.0.10240
TECB(KO 1 e/d);TCBC(KO 1 e/d);TCFB8(KO 1 e/d);TCFB64(KO 1 e/d)	Windows Storage Server 2012 R2, Microsoft Windows RT 8.1, Microsoft Surface with Windows RT 8.1, Microsoft Surface Pro with Windows 8.1, Microsoft Surface 2, Microsoft Surface Pro 2, Microsoft Surface Pro 3, Microsoft Windows Phone 8.1, Microsoft Windows Embedded 8.1 Industry, and Microsoft StorSimple 8100 SymCrypt Cryptographic Implementations #1692 Version 6.3.9600

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
TECB(e/d; KO 1, 2);TCBC(e/d; KO 1, 2);TCFB8(e/d; KO 1, 2);TCFB64(e/d; KO 1, 2)	Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 Next Generation Symmetric Cryptographic Algorithms Implementations (SYMCRYPT) #1387
TECB(e/d; KO 1, 2);TCBC(e/d; KO 1, 2);TCFB8(e/d; KO 1, 2)	Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 Symmetric Algorithm Implementations (RSA32) #1386
TECB(e/d; KO 1, 2);TCBC(e/d; KO 1, 2);TCFB8(e/d; KO 1, 2)	Windows 7 and SP1 and Windows Server 2008 R2 and SP1 Symmetric Algorithm Implementation #846
TECB(e/d; KO 1, 2);TCBC(e/d; KO 1, 2);TCFB8(e/d; KO 1, 2)	Windows Vista SP1 and Windows Server 2008 Symmetric Algorithm Implementation #656
TECB(e/d; KO 1, 2);TCBC(e/d; KO 1, 2);TCFB8(e/d; KO 1, 2)	Windows Vista Symmetric Algorithm Implementation #549
Triple DES MAC	Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 #1386 , vendor-affirmed Windows 7 and SP1 and Windows Server 2008 R2 and SP1 #846 , vendor-affirmed

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
<p>TECB(e/d; KO 1, 2);TCBC(e/d; KO 1, 2)</p>	<p>Windows Embedded Compact 7 Enhanced Cryptographic Provider (RSAENH) #1308Windows Embedded Compact 7 Cryptographic Primitives Library (bcrypt.dll) #1307</p> <p>Windows Server 2003 SP2 Enhanced Cryptographic Provider (RSAENH) #691</p> <p>Windows XP Professional SP3 Kernel Mode Cryptographic Module (fips.sys) #677</p> <p>Windows XP Professional SP3 Enhanced DSS and Diffie-Hellman Cryptographic Provider (DSSENH) #676</p> <p>Windows XP Professional SP3 Enhanced Cryptographic Provider (RSAENH) #675</p> <p>Windows Server 2003 SP2 Enhanced Cryptographic Provider (RSAENH) #544</p> <p>Windows Server 2003 SP2 Enhanced DSS and Diffie-Hellman Cryptographic Provider #543</p> <p>Windows Server 2003 SP2 Kernel Mode Cryptographic Module (fips.sys) #542Windows CE 6.0 and Windows CE 6.0 R2 and Windows Mobile Enhanced Cryptographic Provider (RSAENH) #526</p> <p>Windows CE and Windows Mobile 6 and Windows Mobile 6.1 and Windows Mobile 6.5 Enhanced Cryptographic Provider (RSAENH) #517</p> <p>Windows Server 2003 SP1 Enhanced DSS and Diffie-Hellman Cryptographic Provider (DSSENH) #381</p> <p>Windows Server 2003 SP1 Kernel Mode Cryptographic Module (fips.sys) #370</p> <p>Windows Server 2003 SP1 Enhanced Cryptographic Provider (RSAENH) #365Windows CE 5.00 and Windows CE 5.01 Enhanced Cryptographic Provider (RSAENH) #315</p> <p>Windows Server 2003 Kernel Mode Cryptographic Module (fips.sys) #201</p> <p>Windows Server 2003 Enhanced DSS and Diffie-Hellman Cryptographic Provider (DSSENH) #199</p> <p>Windows Server 2003 Enhanced Cryptographic Provider (RSAENH) #192Windows XP Microsoft Enhanced Cryptographic Provider #81</p> <p>Windows 2000 Microsoft Outlook Cryptographic Provider (EXCHCSPDLL) SR-1A (3821) #18Crypto Driver for Windows 2000 (fips.sys) #16</p>

SP 800-132 Password-Based Key Derivation Function (PBKDF)

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
----------------------------	--

MODES / STATES / KEY SIZES	ALGORITHM IMPLEMENTATION AND CERTIFICATE #
PBKDF (vendor affirmed)	<p>Kernel Mode Cryptographic Primitives Library (cng.sys) Cryptographic Primitives Library (bcryptprimitives.dll and ncryptsslp.dll) in Microsoft Windows 10, Windows 10 Pro, Windows 10 Enterprise, Windows 10 Enterprise LTSC, Windows 10 Mobile, Windows Server 2016 Standard, Windows Server 2016 Datacenter, Windows Storage Server 2016 #2937 (Software Version: 10.0.14393)</p> <p>Microsoft Windows 10, Windows 10 Pro, Windows 10 Enterprise, Windows 10 Enterprise LTSC, Windows 10 Mobile, Windows Server 2016 Standard, Windows Server 2016 Datacenter, Windows Storage Server 2016 #2936 (Software Version: 10.0.14393)</p> <p>Code Integrity (ci.dll) in Microsoft Windows 10, Windows 10 Pro, Windows 10 Enterprise, Windows 10 Enterprise LTSC, Windows 10 Mobile, Windows Server 2016 Standard, Windows Server 2016 Datacenter, Windows Storage Server 2016 #2935 (Software Version: 10.0.14393)</p>
PBKDF (vendor affirmed)	<p>Kernel Mode Cryptographic Primitives Library (cng.sys) in Microsoft Windows 10, Windows 10 Pro, Windows 10 Enterprise, Windows 10 Enterprise LTSC, Windows 10 Mobile, Windows Server 2016 Standard, Windows Server 2016 Datacenter, Windows Storage Server 2016 #2936 (Software Version: 10.0.14393)</p> <p>Windows 8, Windows RT, Windows Server 2012, Surface Windows RT, Surface Windows 8 Pro, and Windows Phone 8 Cryptography Next Generation (CNG), vendor-affirmed</p>

Component Validation List

PUBLICATION / COMPONENT VALIDATED / DESCRIPTION	IMPLEMENTATION AND CERTIFICATE #
<p>ECDSA SigGen:</p> <ul style="list-style-type: none"> • P-256 SHA: SHA-256 • P-384 SHA: SHA-384 • P-521 SHA: SHA-512 <p>Prerequisite: DRBG #489</p>	<p>Microsoft Windows 8.1, Microsoft Windows Server 2012 R2, Microsoft Windows Storage Server 2012 R2, Microsoft Windows RT 8.1, Microsoft Surface with Windows RT 8.1, Microsoft Surface Pro with Windows 8.1, Microsoft Surface 2, Microsoft Surface Pro 2, Microsoft Surface Pro 3, Microsoft Windows Phone 8.1, Microsoft Windows Embedded 8.1 Industry, and Microsoft StorSimple 8100 MsBignum Cryptographic Implementations #1540 Version 6.3.9600</p>
<p>RSASP1:</p> <p>Modulus Size: 2048 (bits) Padding Algorithms: PKCS 1.5</p>	<p>Microsoft Surface Hub Virtual TPM Implementations #1519 Version 10.0.15063.674</p>
<p>RSASP1:</p> <p>Modulus Size: 2048 (bits) Padding Algorithms: PKCS 1.5</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); Virtual TPM Implementations #1518 Version 10.0.16299</p>

PUBLICATION / COMPONENT VALIDATED / DESCRIPTION	IMPLEMENTATION AND CERTIFICATE #
RSADP: Modulus Size: 2048 (bits)	Microsoft Surface Hub MsBignum Cryptographic Implementations #1517 Version 10.0.15063.674
RSASP1: Modulus Size: 2048 (bits) Padding Algorithms: PKCS 1.5	Microsoft Surface Hub MsBignum Cryptographic Implementations #1516 Version 10.0.15063.674
ECDSA SigGen: <ul style="list-style-type: none"> • P-256 SHA: SHA-256 • P-384 SHA: SHA-384 • P-521 SHA: SHA-512 Prerequisite: DRBG #1732	Microsoft Surface Hub MsBignum Cryptographic Implementations #1515 Version 10.0.15063.674
ECDSA SigGen: <ul style="list-style-type: none"> • P-256 SHA: SHA-256 • P-384 SHA: SHA-384 • P-521 SHA: SHA-512 Prerequisite: DRBG #1732	Microsoft Surface Hub SymCrypt Cryptographic Implementations #1514 Version 10.0.15063.674
RSADP: Modulus Size: 2048 (bits)	Microsoft Surface Hub SymCrypt Cryptographic Implementations #1513 Version 10.0.15063.674
RSASP1: Modulus Size: 2048 (bits) Padding Algorithms: PKCS 1.5	Microsoft Surface Hub SymCrypt Cryptographic Implementations #1512 Version 10.0.15063.674

PUBLICATION / COMPONENT VALIDATED / DESCRIPTION	IMPLEMENTATION AND CERTIFICATE #
<p>IKEv1:</p> <ul style="list-style-type: none"> • Methods: Digital Signature, Pre-shared Key, Public Key Encryption • Pre-shared Key Length: 64-2048 <p>Diffie-Hellman shared secrets:</p> <ul style="list-style-type: none"> • Length: 2048 (bits) • SHA Functions: SHA-256 <p>Diffie-Hellman shared secret:</p> <ul style="list-style-type: none"> • Length: 256 (bits) • SHA Functions: SHA-256 <p>Diffie-Hellman shared secret:</p> <ul style="list-style-type: none"> • Length: 384 (bits) • SHA Functions: SHA-384 <p>Prerequisite: SHS #4011, HMAC #3269</p> <p>IKEv2:</p> <ul style="list-style-type: none"> • Derived Keying Material length: 192-1792 <p>Diffie-Hellman shared secret:</p> <ul style="list-style-type: none"> • Length: 2048 (bits) • SHA Functions: SHA-256 <p>Diffie-Hellman shared secret:</p> <ul style="list-style-type: none"> • Length: 256 (bits) • SHA Functions: SHA-256 <p>Diffie-Hellman shared secret:</p> <ul style="list-style-type: none"> • Length: 384 (bits) • SHA Functions: SHA-384 <p>Prerequisite: SHS #4011, HMAC #3269</p> <p>TLS:</p> <ul style="list-style-type: none"> • Supports TLS 1.0/1.1 • Supports TLS 1.2: <p>SHA Functions: SHA-256, SHA-384</p> <p>Prerequisite: SHS #4011, HMAC #3269</p>	<p>Microsoft Surface Hub SymCrypt Cryptographic Implementations #1511 Version 10.0.15063.674</p>
<p>ECDSA SigGen:</p> <ul style="list-style-type: none"> • P-256 SHA: SHA-256 • P-384 SHA: SHA-384 • P-521 SHA: SHA-512 <p>Prerequisite: DRBG #1731</p>	<p>Windows 10 Mobile (version 1709) SymCrypt Cryptographic Implementations #1510 Version 10.0.15254</p>
<p>RSADP: Modulus Size: 2048 (bits)</p>	<p>Windows 10 Mobile (version 1709) SymCrypt Cryptographic Implementations #1509 Version 10.0.15254</p>
<p>RSASP1: Modulus Size: 2048 (bits) Padding Algorithms: PKCS 1.5</p>	<p>Windows 10 Mobile (version 1709) SymCrypt Cryptographic Implementations #1508 Version 10.0.15254</p>

PUBLICATION / COMPONENT VALIDATED / DESCRIPTION	IMPLEMENTATION AND CERTIFICATE #
<p>IKEv1:</p> <ul style="list-style-type: none"> • Methods: Digital Signature, Pre-shared Key, Public Key Encryption • Pre-shared Key Length: 64-2048 <p>Diffie-Hellman shared secret:</p> <ul style="list-style-type: none"> • Length: 2048 (bits) • SHA Functions: SHA-256 <p>Diffie-Hellman shared secret:</p> <ul style="list-style-type: none"> • Length: 256 (bits) • SHA Functions: SHA-256 <p>Diffie-Hellman shared secret:</p> <ul style="list-style-type: none"> • Length: 384 (bits) • SHA Functions: SHA-384 <p>Prerequisite: SHS #4010, HMAC #3268</p> <p>IKEv2:</p> <ul style="list-style-type: none"> • Derived Keying Material length: 192-1792 <p>Diffie-Hellman shared secret:</p> <ul style="list-style-type: none"> • Length: 2048 (bits) • SHA Functions: SHA-256 <p>Diffie-Hellman shared secret:</p> <ul style="list-style-type: none"> • Length: 256 (bits) • SHA Functions: SHA-256 <p>Diffie-Hellman shared secret:</p> <ul style="list-style-type: none"> • Length: 384 (bits) • SHA Functions: SHA-384 <p>Prerequisite: SHS #4010, HMAC #3268</p> <p>TLS:</p> <ul style="list-style-type: none"> • Supports TLS 1.0/1.1 • Supports TLS 1.2: <p>SHA Functions: SHA-256, SHA-384</p> <p>Prerequisite: SHS #4010, HMAC #3268</p>	<p>Windows 10 Mobile (version 1709) SymCrypt Cryptographic Implementations #1507</p> <p>Version 10.0.15254</p>
<p>ECDSA SigGen:</p> <ul style="list-style-type: none"> • P-256 SHA: SHA-256 • P-384 SHA: SHA-384 • P-521 SHA: SHA-512 <p>Prerequisite: DRBG #1731</p>	<p>Windows 10 Mobile (version 1709) MsBignum Cryptographic Implementations #1506</p> <p>Version 10.0.15254</p>
<p>RSADP:</p> <p>Modulus Size: 2048 (bits)</p>	<p>Windows 10 Mobile (version 1709) MsBignum Cryptographic Implementations #1505</p> <p>Version 10.0.15254</p>
<p>RSASP1:</p> <p>Modulus Size: 2048 (bits)</p> <p>Padding Algorithms: PKCS 1.5</p>	<p>Windows 10 Mobile (version 1709) MsBignum Cryptographic Implementations #1504</p> <p>Version 10.0.15254</p>

PUBLICATION / COMPONENT VALIDATED / DESCRIPTION	IMPLEMENTATION AND CERTIFICATE #
<p>ECDSA SigGen:</p> <ul style="list-style-type: none"> • P-256 SHA: SHA-256 • P-384 SHA: SHA-384 • P-521 SHA: SHA-512 <p>Prerequisite: DRBG #1730</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); MsBignum Cryptographic Implementations #1503</p> <p>Version 10.0.16299</p>
<p>RSADP:</p> <p>Modulus Size: 2048 (bits)</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); MsBignum Cryptographic Implementations #1502</p> <p>Version 10.0.16299</p>
<p>RSASP1:</p> <p>Modulus Size: 2048 (bits)</p> <p>Padding Algorithms: PKCS 1.5</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); MsBignum Cryptographic Implementations #1501</p> <p>Version 10.0.16299</p>
<p>ECDSA SigGen:</p> <ul style="list-style-type: none"> • P-256 SHA: SHA-256 • P-384 SHA: SHA-384 • P-521 SHA: SHA-512 <p>Prerequisite: DRBG #1730</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); SymCrypt Cryptographic Implementations #1499</p> <p>Version 10.0.16299</p>
<p>RSADP:</p> <p>Modulus Size: 2048 (bits)</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); SymCrypt Cryptographic Implementations #1498</p> <p>Version 10.0.16299</p>
<p>RSASP1:</p> <p>Modulus Size: 2048 (bits)</p> <p>Padding Algorithms: PKCS 1.5</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); SymCrypt Cryptographic Implementations #1497</p> <p>Version 10.0.16299</p>

PUBLICATION / COMPONENT VALIDATED / DESCRIPTION	IMPLEMENTATION AND CERTIFICATE #
<p>IKEv1:</p> <ul style="list-style-type: none"> • Methods: Digital Signature, Pre-shared Key, Public Key Encryption • Pre-shared Key Length: 64-2048 <p>Diffie-Hellman shared secret:</p> <ul style="list-style-type: none"> • Length: 2048 (bits) • SHA Functions: SHA-256 <p>Diffie-Hellman shared secret:</p> <ul style="list-style-type: none"> • Length: 256 (bits) • SHA Functions: SHA-256 <p>Diffie-Hellman shared secret:</p> <ul style="list-style-type: none"> • Length: 384 (bits) • SHA Functions: SHA-384 <p>Prerequisite: SHS #4009, HMAC #3267</p> <p>IKEv2:</p> <ul style="list-style-type: none"> • Derived Keying Material length: 192-1792 <p>Diffie-Hellman shared secret:</p> <ul style="list-style-type: none"> • Length: 2048 (bits) • SHA Functions: SHA-256 <p>Diffie-Hellman shared secret:</p> <ul style="list-style-type: none"> • Length: 256 (bits) • SHA Functions: SHA-256 <p>Diffie-Hellman shared secret:</p> <ul style="list-style-type: none"> • Length: 384 (bits) • SHA Functions: SHA-384 <p>Prerequisite: SHS #4009, HMAC #3267</p> <p>TLS:</p> <ul style="list-style-type: none"> • Supports TLS 1.0/1.1 • Supports TLS 1.2: <p>SHA Functions: SHA-256, SHA-384</p> <p>Prerequisite: SHS #4009, HMAC #3267</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); SymCrypt Cryptographic Implementations #1496</p> <p>Version 10.0.16299</p>

PUBLICATION / COMPONENT VALIDATED / DESCRIPTION	IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-4 ECDSA</p> <p>Signature Generation of hash sized messages</p> <p>ECDSA SigGen Component: CURVES(P-256 P-384 P-521)</p>	<p>Windows 10 Creators Update (version 1703) Home, Pro, Enterprise, Education, Windows 10 S, Windows 10 Mobile MsBignum Cryptographic Implementations #1284</p> <p>Version 10.0. 15063</p> <p>Windows 10 Creators Update (version 1703) Home, Pro, Enterprise, Education, Windows 10 S, Windows 10 Mobile SymCrypt Cryptographic Implementations #1279</p> <p>Version 10.0. 15063</p> <p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, Surface Pro 3 and Surface 3 w/ Windows 10 Anniversary Update; Microsoft Lumia 950 and Lumia 650 w/ Windows 10 Mobile Anniversary Update MsBignum Cryptographic Implementations #922</p> <p>Version 10.0.14393</p> <p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, and Surface Pro 3 w/ Windows 10 Anniversary Update Virtual TPM Implementations #894</p> <p>Version 10.0.14393 Microsoft Windows 10 November 2015 Update; Microsoft Surface Book, Surface Pro 4, Surface Pro 3, Surface 3, Surface Pro 2, and Surface Pro w/ Windows 10 November 2015 Update; Windows 10 Mobile for Microsoft Lumia 950 and Microsoft Lumia 635; Windows 10 for Microsoft Surface Hub 84" and Surface Hub 55" MsBignum Cryptographic Implementations #666</p> <p>Version 10.0.10586</p> <p>Microsoft Windows 8.1, Microsoft Windows Server 2012 R2, Microsoft Windows Storage Server 2012 R2, Microsoft Windows RT 8.1, Microsoft Surface with Windows RT 8.1, Microsoft Surface Pro with Windows 8.1, Microsoft Surface 2, Microsoft Surface Pro 2, Microsoft Surface Pro 3, Microsoft Windows Phone 8.1, Microsoft Windows Embedded 8.1 Industry, and Microsoft StorSimple 8100 MsBignum Cryptographic Implementations #288</p> <p>Version 6.3.9600</p>

PUBLICATION / COMPONENT VALIDATED / DESCRIPTION	IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-4 RSA; PKCS#1 v2.1</p> <p>RSASP1 Signature Primitive</p> <p>RSASP1: (Mod2048: PKCS1.5 PKCSPSS)</p>	<p>Windows 10 Creators Update (version 1703) Pro, Enterprise, Education Virtual TPM Implementations #1285</p> <p>Version 10.0.15063</p> <p>Windows 10 Creators Update (version 1703) Home, Pro, Enterprise, Education, Windows 10 S, Windows 10 Mobile MsBignum Cryptographic Implementations #1282</p> <p>Version 10.0.15063</p> <p>Windows 10 Creators Update (version 1703) Home, Pro, Enterprise, Education, Windows 10 S, Windows 10 Mobile SymCrypt Cryptographic Implementations #1280</p> <p>Version 10.0.15063</p> <p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, and Surface Pro 3 w/ Windows 10 Anniversary Update Virtual TPM Implementations #893</p> <p>Version 10.0.14393</p> <p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, Surface Pro 3 and Surface 3 w/ Windows 10 Anniversary Update; Microsoft Lumia 950 and Lumia 650 w/ Windows 10 Mobile Anniversary Update MsBignum Cryptographic Implementations #888</p> <p>Version 10.0.14393</p> <p>Microsoft Windows 10 November 2015 Update; Microsoft Surface Book, Surface Pro 4, Surface Pro 3, Surface 3, Surface Pro 2, and Surface Pro w/ Windows 10 November 2015 Update; Windows 10 Mobile for Microsoft Lumia 950 and Microsoft Lumia 635; Windows 10 for Microsoft Surface Hub 84" and Surface Hub 55" MsBignum Cryptographic Implementations #665</p> <p>Version 10.0.10586</p> <p>Microsoft Windows 10, Microsoft Surface Pro 3 with Windows 10, Microsoft Surface 3 with Windows 10, Microsoft Surface Pro 2 with Windows 10, Microsoft Surface Pro with Windows 10 MsBignum Cryptographic Implementations #572</p> <p>Version 10.0.10240</p> <p>Microsoft Windows 8.1, Microsoft Windows Server 2012 R2, Microsoft Windows Storage Server 2012 R2, Microsoft Windows RT 8.1, Microsoft Surface with Windows RT 8.1, Microsoft Surface Pro with Windows 8.1, Microsoft Surface 2, Microsoft Surface Pro 2, Microsoft Surface Pro 3, Microsoft Windows Phone 8.1, Microsoft Windows Embedded 8.1 Industry MsBignum Cryptographic Implementations #289</p> <p>Version 6.3.9600</p>

PUBLICATION / COMPONENT VALIDATED / DESCRIPTION	IMPLEMENTATION AND CERTIFICATE #
<p>FIPS186-4 RSA; RSADP RSADP Primitive RSADP: (Mod2048)</p>	<p>Windows 10 Creators Update (version 1703) Home, Pro, Enterprise, Education, Windows 10 S, Windows 10 Mobile MsBignum Cryptographic Implementations #1283 Version 10.0.15063</p> <p>Windows 10 Creators Update (version 1703) Home, Pro, Enterprise, Education, Windows 10 S, Windows 10 Mobile SymCrypt Cryptographic Implementations #1281 Version 10.0.15063</p> <p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, and Surface Pro 3 w/ Windows 10 Anniversary Update Virtual TPM Implementations #895 Version 10.0.14393</p> <p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, Surface Pro 3 and Surface 3 w/ Windows 10 Anniversary Update; Microsoft Lumia 950 and Lumia 650 w/ Windows 10 Mobile Anniversary Update Cryptography Next Generation (CNG) Implementations #887 Version 10.0.14393</p> <p>Microsoft Windows 10 November 2015 Update; Microsoft Surface Book, Surface Pro 4, Surface Pro 3, Surface 3, Surface Pro 2, and Surface Pro w/ Windows 10 November 2015 Update; Windows 10 Mobile for Microsoft Lumia 950 and Microsoft Lumia 635; Windows 10 for Microsoft Surface Hub 84" and Surface Hub 55" Cryptography Next Generation (CNG) Implementations #663 Version 10.0.10586</p> <p>Microsoft Windows 10, Microsoft Surface Pro 3 with Windows 10, Microsoft Surface 3 with Windows 10, Microsoft Surface Pro 2 with Windows 10, Microsoft Surface Pro with Windows 10 Cryptography Next Generation (CNG) Implementations #576 Version 10.0.10240</p>

PUBLICATION / COMPONENT VALIDATED / DESCRIPTION	IMPLEMENTATION AND CERTIFICATE #
<p>SP800-135 Section 4.1.1, IKEv1 Section 4.1.2, IKEv2 Section 4.2, TLS</p>	<p>Windows 10 Home, Pro, Enterprise, Education, Windows 10 S Fall Creators Update; Windows Server, Windows Server Datacenter (version 1709); SymCrypt Cryptographic Implementations #1496 Version 10.0.16299</p> <p>Windows 10 Creators Update (version 1703) Home, Pro, Enterprise, Education, Windows 10 S, Windows 10 Mobile SymCrypt Cryptographic Implementations #1278 Version 10.0.15063</p> <p>Windows Embedded Compact Cryptographic Primitives Library (bcrypt.dll) #1140 Version 7.00.2872</p> <p>Windows Embedded Compact Cryptographic Primitives Library (bcrypt.dll) #1139 Version 8.00.6246</p> <p>Microsoft Windows 10 Anniversary Update, Windows Server 2016, Windows Storage Server 2016; Microsoft Surface Book, Surface Pro 4, Surface Pro 3 and Surface 3 w/ Windows 10 Anniversary Update; Microsoft Lumia 950 and Lumia 650 w/ Windows 10 Mobile Anniversary Update BcryptPrimitives and NCryptSSLp #886 Version 10.0.14393</p> <p>Microsoft Windows 10 November 2015 Update; Microsoft Surface Book, Surface Pro 4, Surface Pro 3, Surface 3, Surface Pro 2, and Surface Pro w/ Windows 10 November 2015 Update; Windows 10 Mobile for Microsoft Lumia 950 and Microsoft Lumia 635; Windows 10 for Microsoft Surface Hub 84" and Surface Hub 55" BCryptPrimitives and NCryptSSLp #664 Version 10.0.10586</p> <p>Microsoft Windows 10, Microsoft Surface Pro 3 with Windows 10, Microsoft Surface 3 with Windows 10, Microsoft Surface Pro 2 with Windows 10, Microsoft Surface Pro with Windows 10 BCryptPrimitives and NCryptSSLp #575 Version 10.0.10240</p> <p>Microsoft Windows 8.1, Microsoft Windows Server 2012 R2, Microsoft Windows Storage Server 2012 R2, Microsoft Windows RT 8.1, Microsoft Surface with Windows RT 8.1, Microsoft Surface Pro with Windows 8.1, Microsoft Surface 2, Microsoft Surface Pro 2, Microsoft Surface Pro 3, Microsoft Windows Phone 8.1, Microsoft Windows Embedded 8.1 Industry, and Microsoft StorSimple 8100 BCryptPrimitives and NCryptSSLp #323 Version 6.3.9600</p>

Contact

fips@microsoft.com

References

- [FIPS 140-2, Security Requirements for Cryptographic Modules](#))
- [Cryptographic Module Validation Program \(CMVP\) FAQ](#)
- [SP 800-57 - Recommendation for Key Management – Part 1: General \(Revised\)](#)
- [SP 800-131A - Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths](#)

Common Criteria Certifications

7/1/2022 • 5 minutes to read • [Edit Online](#)

Microsoft is committed to optimizing the security of its products and services. As part of that commitment, Microsoft supports the Common Criteria certification program, ensures that products incorporate the features and functions required by relevant Common Criteria Protection Profiles, and completes Common Criteria certifications of Microsoft Windows products. This topic lists the current and archived certified Windows products, together with relevant documentation from each certification.

Certified Products

The product releases below are currently certified against the cited Protection Profile, as listed on the [Common Criteria Portal](#). The Security Target describes the product edition(s) in scope, the security functionality in the product, and the assurance measures from the Protection Profile used as part of the evaluation. The Administrative Guide provides guidance on configuring the product to match the evaluated configuration. The Certification Report or Validation Report documents the results of the evaluation by the validation team, with the Assurance Activity Report providing details on the evaluator's actions.

Microsoft Windows 10, Windows Server version 2004 (May 2020 Update); Microsoft Windows Server Core Datacenter (Azure Fabric Controller); Microsoft Windows Server Core Datacenter (Azure Stack)

Certified against the Protection Profile for General Purpose Operating Systems, including the Extended Package for Wireless Local Area Network Clients and the Module for Virtual Private Network Clients.

- [Security Target](#)
- [Administrative Guide](#)
- [Validation Report](#)
- [Assurance Activity Report](#)

Microsoft Windows Server, Windows 10 version 1909 (November 2019 Update), Microsoft Windows Server 2019 (version 1809) Hyper-V

Certified against the Protection Profile for Virtualization, including the Extended Package for Server Virtualization.

- [Security Target](#)
- [Administrative Guide](#)
- [Validation Report](#)
- [Assurance Activities Report](#)

Microsoft Windows 10 and Windows Server (November 2019 Update, version 1909)

Certified against the Protection Profile for General Purpose Operating Systems, including the Extended Package for Wireless Local Area Network Clients and the Module for Virtual Private Network Clients.

- [Security Target](#)
- [Administrative Guide](#)
- [Certification Report](#)
- [Assurance Activity Report](#)

Microsoft Windows 10 and Windows Server (May 2019 Update, version 1903)

Certified against the Protection Profile for General Purpose Operating Systems, including the Extended Package for Wireless Local Area Network Clients.

- [Security Target](#)
- [Administrative Guide](#)
- [Certification Report](#)
- [Assurance Activity Report](#)

Microsoft Windows 10 and Windows Server (October 2018 Update, version 1809)

Certified against the Protection Profile for General Purpose Operating Systems, including the Extended Package for Wireless Local Area Network Clients.

- [Security Target](#)
- [Administrative Guide](#)
- [Certification Report](#)
- [Assurance Activity Report](#)

Microsoft Windows 10 and Windows Server (April 2018 Update, version 1803)

Certified against the Protection Profile for General Purpose Operating Systems, including the Extended Package for Wireless Local Area Network Clients.

- [Security Target](#)
- [Administrative Guide](#)
- [Certification Report](#)
- [Assurance Activity Report](#)

Microsoft Windows 10 and Windows Server (Fall Creators Update, version 1709)

Certified against the Protection Profile for General Purpose Operating Systems.

- [Security Target](#)
- [Administrative Guide](#)
- [Certification Report](#)
- [Assurance Activity Report](#)

Microsoft Windows 10 (Creators Update, version 1703)

Certified against the Protection Profile for General Purpose Operating Systems.

- [Security Target](#)
- [Administrative Guide](#)
- [Certification Report](#)
- [Assurance Activity Report](#)

Microsoft Windows 10 (Anniversary Update, version 1607) and Windows Server 2016

Certified against the Protection Profile for General Purpose Operating Systems.

- [Security Target](#)
- [Administrative Guide](#)
- [Validation Report](#)
- [Assurance Activity Report](#)

Microsoft Windows 10 (version 1507) and Windows Server 2012 R2

Certified against the Protection Profile for General Purpose Operating Systems.

- [Security Target](#)
- [Administrative Guide](#)
- [Certification Report](#)

- [Assurance Activity Report](#)

Archived Certified Products

The product releases below were certified against the cited Protection Profile and are now archived, as listed on the [Common Criteria Portal](#). The Security Target describes the product edition(s) in scope, the security functionality in the product, and the assurance measures from the Protection Profile used as part of the evaluation. The Administrative Guide provides guidance on configuring the product to match the evaluated configuration. The Validation Report documents the results of the evaluation by the validation team, with the Assurance Activity Report, where available, providing details on the evaluator's actions.

Microsoft Windows Server 2016, Windows Server 2012 R2, and Windows 10

Certified against the Protection Profile for Server Virtualization.

- [Security Target](#)
- [Administrative Guide](#)
- [Validation Report](#)
- [Assurance Activity Report](#)

Microsoft Windows 10 and Windows 10 Mobile (Anniversary Update, version 1607)

Certified against the Protection Profile for Mobile Device Fundamentals.

- [Security Target](#)
- [Administrative Guide](#)
- [Validation Report](#)
- [Assurance Activity Report](#)

Microsoft Windows 10 (Anniversary Update, version 1607) and Windows Server 2016

Certified against the Protection Profile for IPsec Virtual Private Network (VPN) Clients.

- [Security Target](#)
- [Administrative Guide](#)
- [Validation Report](#)
- [Assurance Activity Report](#)

Microsoft Windows 10 (November 2015 Update, version 1511)

Certified against the Protection Profile for Mobile Device Fundamentals.

- [Security Target](#)
- [Administrative Guide](#)
- [Validation Report](#)
- [Assurance Activity Report](#)

Microsoft Windows 10 and Windows 10 Mobile (version 1507)

Certified against the Protection Profile for Mobile Device Fundamentals.

- [Security Target](#)
- [Administrative Guide](#)
- [Validation Report](#)
- [Assurance Activity Report](#)

Microsoft Windows 10 (version 1507)

Certified against the Protection Profile for IPsec Virtual Private Network (VPN) Clients.

- [Security Target](#)
- [Administrative Guide](#)
- [Validation Report](#)
- [Assurance Activity Report](#)

Windows 8.1 with Surface 3 and Windows Phone 8.1 with Lumia 635 and Lumia 830

Certified against the Protection Profile for Mobile Device Fundamentals.

- [Security Target](#)
- [Administrative Guide](#)
- [Validation Report](#)

Microsoft Surface Pro 3 and Windows 8.1

Certified against the Protection Profile for Mobile Device Fundamentals.

- [Security Target](#)
- [Administrative Guide](#)
- [Validation Report](#)

Windows 8.1 and Windows Phone 8.1

Certified against the Protection Profile for Mobile Device Fundamentals.

- [Security Target](#)
- [Administrative Guide](#)
- [Validation Report](#)

Windows 8 and Windows Server 2012

Certified against the Protection Profile for General Purpose Operating Systems.

- [Security Target](#)
- [Administrative Guide](#)
- [Validation Report](#)

Windows 8 and Windows RT

Certified against the Protection Profile for General Purpose Operating Systems.

- [Security Target](#)
- [Administrative Guide](#)
- [Validation Report](#)

Windows 8 and Windows Server 2012 BitLocker

Certified against the Protection Profile for Full Disk Encryption.

- [Security Target](#)
- [Administrative Guide](#)
- [Validation Report](#)

Windows 8, Windows RT, and Windows Server 2012 IPsec VPN Client

Certified against the Protection Profile for IPsec Virtual Private Network (VPN) Clients.

- [Security Target](#)
- [Administrative Guide](#)
- [Validation Report](#)

Windows 7 and Windows Server 2008 R2

Certified against the Protection Profile for General Purpose Operating Systems.

- [Security Target](#)
- [Administrative Guide](#)
- [Validation Report](#)

Microsoft Windows Server 2008 R2 Hyper-V Role

- [Security Target](#)
- [Administrative Guide](#)
- [Validation Report](#)

Windows Vista and Windows Server 2008 at EAL4+

- [Security Target](#)
- [Administrative Guide](#)
- [Validation Report](#)

Windows Vista and Windows Server 2008 at EAL1

- [Security Target](#)
- [Administrative Guide](#)
- [Certification Report](#)

Microsoft Windows Server 2008 Hyper-V Role

- [Security Target](#)
- [Administrative Guide](#)
- [Certification Report](#)

Windows Server 2003 Certificate Server

- [Security Target](#)
- [Administrator's Guide](#)
- [Configuration Guide](#)
- [User's Guide](#)
- [Evaluation Technical Report](#)
- [Validation Report](#)

Windows Rights Management Services

- [Security Target](#)
- [Validation Report](#)