Microsoft Corporation Common Criteria Evaluation

General Purpose Operating System Protection Profile

Operational and Administrative Guidance

Microsoft Windows 11 (version 22H2)
Microsoft Windows 10 (version 22H2)
Microsoft Windows Server 2022
Microsoft Windows Server 2022 Datacenter Azure Edition
Microsoft Azure Stack HCIv2 version 22H2
Microsoft Azure Stack Hub
Azure Stack Edge

Prepared By	Microsoft Corporation One Microsoft Way Redmond, WA 98052-6399
Version Number	8.0
Updated On	July 3, 2023

Copyright and disclaimer

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This document is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AS TO THE INFORMATION IN THIS DOCUMENT.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

The example companies, organizations, products, people and events depicted herein are fictitious. No association with any real company, organization, product, person or event is intended or should be inferred.

Contents

Change history

Version	Date	Description
1.0	March 20, 2018	Administrative Guide for Windows 10 and Windows Server Fall Creators Update (1709)
2.0	October 11, 2018	Administrative Guide for Windows 10 and Windows Server April 2018 Update (1803)
3.0	February 21, 2019	Administrative Guide for Windows 10 and Windows Server 2019, Version 1809 (October 2018 Update)
4.0	June 10, 2019	Administrative Guide for Windows 10 and Windows Server, Version 1903 (May 2019 Update)
5.0	January 16, 2020	Administrative Guide for Windows 10 and Windows Server, Version 1909 (November 2019 Update)
6.0	June 1, 2021	Administrative Guide for Windows 10 and Windows Server, Version 2004 (May 2020 Update)
7.0	September 13, 2022	Administrative Guide for Microsoft Windows 11, Windows 10 (versions 20H2, 21H1, and 21H2), Windows Server, Windows Server 2022, Azure Stack HCIv2 version 21H2, Azure Stack Hub, and Azure Stack Edge
7.1	October 21, 2022	Minor update to remove proprietary footer text.
8.0	July 3, 2023	Operational and Administrative Guidance for Microsoft Windows 11 (version 22H2), Microsoft Windows 10 (version 22H2), Microsoft Windows Server 2022, Microsoft Windows Server 2022 Datacenter Azure Edition, Microsoft Azure Stack HCIv2 version 22H2, Microsoft Azure Stack Hub, and Azure Stack Edge

1 Introduction

This administrative guide provides information for Microsoft Windows 11 (version 22H2), Microsoft Windows 10 (version 22H2), Microsoft Windows Server, Microsoft Windows Server 2022, Microsoft Azure Stack HCIv2 version 22H2, Microsoft Azure Stack Hub, and Azure Stack Edge, as required by the Common Criteria General Purpose Operating System (GP OS) protection profile. This document refers to all the product versions and editions collectively as "Windows" where appropriate. This guide enables an IT professional to configure Windows and its operational environment to match the configuration under which the product was evaluated, and to manage the Windows features in the scope of evaluation. The audience for this document is an IT Administrator familiar with current administrative practices for Windows. IT Administrators must follow the guidance in this document to ensure a device matches the evaluated configuration.

1.1 How this guide is organized

The sections in this administrative guide group information together categorically:

- Section 1, Introduction, provides an overview of the guide, explains conventions in the document, and includes general guidance that subsequent sections may refer to.
- Section 2, Evaluated editions, platforms, and roles, identifies the editions of Windows that were evaluated and the set of hardware platforms the evaluation was performed on. This section also provides a mapping of what administration solutions are available to each user role and on each Windows edition.
- Section 3, Evaluated configuration, covers deployment of the product and the set of operational prerequisites and configuration choices that must be followed to match the evaluated Windows configuration.
- Section 4, Managing evaluated features, covers management of the Windows features in scope for this evaluation. This includes guidance on relevant feature configuration choices and approaches to implementing them, organized by feature area.
- Section 5, Audit events, provides detailed information on the audit events relevant to the evaluated configuration that are available in Windows logs. This information enables administrators to perform security monitoring and forensics.

1.2 Links to public Microsoft documentation

This document provides links to public websites that host published Microsoft documentation. These topics provide additional information or detailed configuration instructions to satisfy evaluation requirements.

Note: Some linked topics may have originally been authored for earlier versions of Windows, e.g., Windows 8.x. In all cases, the information in the topics applies to the evaluated configuration.

1.3 Security Target reference

This Common Criteria evaluation requires a Security Target document that outlines the evaluation scope, which this guide may refer to. The Security Target for this evaluation is the Microsoft Windows, Windows Server, Azure Stack Security Target (public version 0.04, July 3, 2023) and is available on the following websites:

- Microsoft publishes all Common Criteria evaluation documentation, including the Security Target for each evaluation, at https://learn.microsoft.com/en-us/windows/security/threat-protection/windows-platform-common-criteria.
- The Common Criteria portal provides Security Targets for all certified products at https://www.commoncriteriaportal.org/products/.

2 Operating system variants, platforms, and roles

2.1 Windows variants and hardware platforms in scope

This administrative guide applies to the Windows operating system (OS) variants in scope for this Common Criteria evaluation, as listed in the Security Target and repeated below. Not all administrative solutions in this guide are available in all Windows variants. See the section, Guidance specific to Windows variants, for a listing of solutions available per variant.

The evaluation was performed on the following Windows variants. For each, the installation ISO file is listed along with its SHA256 hash. Customers may use the hash to validate that an ISO file is identical to one used in this evaluation, as .ISO file names may vary depending on the file download source. See the section, Validating the hash value of installation media for information on how to compare the hash value of installation media against the reference hash values listed below.

- Microsoft Windows 11 Enterprise, Education, Pro editions (x64)
  - Build: 10.0.22621.1
  - ISO: 22621.1.220506-1250.ni_release_CLIENT_BUSINESS_VOL_x64FRE_en-us.iso
  - ISO hash: DD606F093FCB9F785B3C41B27C6AE89CEEDDFF947824B8581F4BE75AF7286FBF

- Microsoft Windows 11 Pro edition (ARM)
  - Build: 10.0.22621.1
  - ISO: 22621.1.220506-1250.ni_release_CLIENT_BUSINESS_VOL_A64FRE_en-us.iso
  - ISO hash: 5A664CD759D6ADF3A9A8AD61C7A4D650ADD21EA81F12C2B1EBF919BC255E57BF

- Microsoft Windows 11 IoT Enterprise edition (x64)
  - Build: 10.0.22621.1
  - ISO: ENT:22621.1.220506-1250.ni_release_CLIENT_IOTENTERPRISES_vl_x64FRE_en-us.iso
  - ISO hash: E0F3B60E0474E29B027573BF8A5EDD406D0E669048425AD7716C1AFCE0A01EB2

- Microsoft Windows 10 version 22H2 Enterprise, Pro editions (x64)
  - Build: 10.0.19045.2006
  - ISO: 19045.2006.220908-0225.22h2_release_svc_refresh_CLIENT_BUSINESS_VOL_x64FRE_en-us.iso

  - ISO hash: 18A84E0DA1043D7C1D3CF46EE127E8F637D425AD57E115EE862AF203FA4932A8

- Microsoft Windows Server 2022 Standard, Datacenter editions
  - Build: 10.0.20348.587
  - ISO: 20348.587.220303-0313.fe_release_svc_refresh_SERVER_OEMRET_x64FRE_en-us.iso
  - ISO hash: 9225F530F09BDB566684C2F3F17373BC2B74E07B0F7205F6376D2B161FCA4B18

- Microsoft Windows Server Datacenter: Azure Edition
  - Build: 10.0.20348.1006
  - ISO: 20348.1006.220908-1954.fe_release_svc_refresh_SERVERDCAZURE_VOL_x64FRE_en-us.iso
  - ISO hash: F97D1EA3D5817CC31D74766EF30EE8D17F1FCDBCDE691D169F837ED29F101F88

- Microsoft Azure Stack HCIv2 version 22H2
  - Build: 10.0.20349.1129
  - ISO: 20349.1129.221007-2120.fe_release_hciv3_svc_refresh_SERVERAZURESTACKHCICOR_OEMRET_x64FRE_en-us.iso
  - ISO hash: DA90CAB6F6FFD07589F8E487BE3CE0BA170C23EDEA025D73A8094BCAB08A60AB

- Microsoft Azure Stack Hub and Edge
  - Build: 10.0.17784.1068
  - ISO: 17784.1068.200716-1400.rs5_release_svc_hci_SERVER_OEMRET_x64FRE_en-us.iso
  - ISO hash: 55A014606D1CA422A8AD851ABA83C896E094A1ED7587572E6243BCA4FEA15182

The evaluation was performed using the following real or virtualized hardware platforms:

- Microsoft Surface Laptop 5
- Microsoft Surface Pro 9
- Microsoft Surface Pro 9 5G (Qualcomm)
- Surface Studio 2+
- Microsoft Surface Laptop Go 2
- Microsoft Surface Go 3
- Microsoft Surface Laptop Studio
- Microsoft Surface Laptop 4 (AMD)

- Microsoft Surface Laptop 4 (Intel)
- Dell Latitude 7420
- Dell Latitude 9520
- HP 840 G10
- Lenovo ThinkPad Z13 (AMD)
- Panasonic CF-33
- Panasonic FZ-55 Toughbook
- Zebra L10ax / RTL 10C1
- Zebra ET80Z Tablet
- Microsoft Windows Server 2022 Hyper-V
- Microsoft Windows Server 2019 Hyper-V
- Dell PowerEdge R640
- Dell PowerEdge R6625
- Dell PowerEdge R760xp
- Dell PowerEdge R840
- HPE Edgeline EL8000 / ProLiant e910 Server Blade
- Voyager Klaas Telecom

2.2 Guidance specific to user roles

The evaluated configuration includes three user roles:

- **IT Administrator**: an administrator who manages a device remotely.
- **Local Administrator**: an administrator who manages a device locally. The administrator's account is part of the local Administrators group.
- **Standard User**: a non-administrator who manages a device locally. The standard user's account is not part of the local Administrators group.

The following table lists the solutions in scope for each user role. Any exception to this mapping is noted in the section that provides the guidance.

Solution	IT Administrator	Local Administrator	Standard User
PowerShell Remoting	In Scope	Out of Scope	Out of Scope
Local PowerShell	In Scope	In Scope	Out of Scope

Modern device management (MDM)	In Scope	Out of Scope	Out of Scope
Group Policy	In Scope	Out of Scope	Out of Scope
Windows Registry	In Scope	In Scope	Out of Scope
Windows Command Line utilities	In Scope	In Scope	Out of Scope
Windows graphical user interface (GUI) features	In Scope	In Scope	In Scope

Access to user-accessible functions is controlled by the rights and privileges assigned to these user roles. No additional configuration is needed to control access to the user-accessible functions in a secure processing environment. Attempts to access user-accessible functions that require local administrator rights or privileges are denied for the standard user role.

The following topics describe local accounts in Windows and provide details on how to manage them, including how to make a standard user account a member of the local Administrators group:

- Local accounts: https://learn.microsoft.com/en-us/windows/security/identity-protection/access-control/local-accounts
- PowerShell Add-LocalGroupMember cmdlet: https://learn.microsoft.com/en-us/powershell/module/microsoft.powershell.localaccounts/add-localgroupmember

2.3 Guidance specific to Windows variants

This administrative guide provides a variety of different solutions to achieve the evaluated configuration. Not all solutions are available in all Windows variants. For example, Windows Server Core Datacenter products do not provide a graphical user interface (GUI) by default, thus all GUI solutions in this guide are not available on devices running Windows Server Core Datacenter. The following table describes what solutions are available on which Windows variants. Any exception to this mapping is noted in the section that provides the guidance.

Solution	Windows 11 Enterprise edition	Windows 10 Pro and Enterprise editions	Windows Server and Windows Server 2022 Standard edition	Windows Server and Windows Server 2022 Datacenter edition	Azure Stack HCI, Hub, and Edge
PowerShell	Available	Available	Available	Available[1]	Available
Modern device management (MDM)	Available	Available	Not Available	Not Available	Not Available
Group Policy	Available	Available	Available	Available	Not Available
Windows Registry	Available	Available	Available	Available	Available
Windows Command Line utilities	Available	Available	Available	Available[1]	Not Available
Windows GUI features	Available	Available	Available	Not Available	Not Available

[1] PowerShell and command line utilities are available in Windows Server installations that include a shell (Desktop Experience). If the Server Core installation option has been chosen, no shell is present and neither PowerShell nor command line utilities may be accessed locally. Windows Server Core installations may still be managed remotely via PowerShell and command line utilities. For more information, see https://learn.microsoft.com/en-us/windows-server/administration/server-core/what-is-server-core.

2.4 Remote administration

This section outlines different approaches administrators may take to remotely administer windows. See the preceding section, Guidance specific to Windows variants, for the remote administration approaches that are supported in each Windows edition. Use the approach that best fits the Windows edition and operational environment.

2.4.1 Remote administration using modern device management (MDM)

The evaluation was performed both on devices enrolled in modern device management (MDM, also referred to as mobile device management), and on devices not enrolled in MDM.

Policies may be configured remotely by an IT Administrator using an MDM solution and Windows Configuration Service Providers (CSPs). This guide references specific CSPs and functions within them as solutions that may be used with different MDM systems. See the MDM solution documentation for detailed configuration actions to be taken with Windows CSPs.

📝 **Note**: MDM solutions may have prerequisites for enrollment, for example, trusting the MDM certificate. Guidance for MDM prerequisites are out of scope of this documentation. IT Administrators should consult the MDM system documentation to ensure prerequisites are met before enrollment is performed.

The following topics provide general information on using MDM to administer Windows, including solutions for enrolling and unenrolling (disconnecting) devices:

- Mobile device management overview: https://learn.microsoft.com/en-us/windows/client-management/mdm-overview
- MDM enrollment of Windows devices: https://learn.microsoft.com/en-us/windows/client-management/mdm/mdm-enrollment-of-windows-devices
- Disconnecting from the management infrastructure (unenrollment): https://learn.microsoft.com/en-us/windows/client-management/mdm/disconnecting-from-mdm-unenrollment

The following topics provide a reference to Windows CSPs and details on the Policy CSP, which contains many of the solutions detailed in this guide:

- Configuration service provider reference - https://learn.microsoft.com/en-us/windows/client-management/mdm/
- Policy CSP - https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-configuration-service-provider.

2.4.2 Remote administration of domain-joined devices using Group Policy Objects (GPO)

Group policy may be used to set Windows policies for domain-joined machines. For information on how to join a machine to a domain and add Active Directory users, see the following topics. Note that the directory server name and address to bind with must be supplied by your IT administrator.

- Join a Computer to a Domain: https://learn.microsoft.com/en-us/windows-server/identity/ad-fs/deployment/join-a-computer-to-a-domain
- PowerShell New-ADUser cmdlet: https://learn.microsoft.com/en-us/powershell/module/activedirectory/new-aduser

Once joined to a domain, policies are configured using the Group Policy Editor (gpedit.msc) or Local Security Policy Editor (secpol.msc). Group Policy Editor may also be used to remotely administrate policy on a machine by following these steps:

1. **Start** > **Run** > **mmc**
2. **File** > **Add/Remove Snap-in**
3. Under the **Standalone** tab, click **Add…**
4. Choose **Group Policy Object Editor**
5. In the following wizard, click the **Browse** button
6. Click the **Computers** tab, select the **Another Computer** radio button, and type the name of the computer or browse to it.
7. Click **OK**, then **Finish**, then **Close**, and finally **OK** again.

2.4.3 Remote administration using PowerShell and Group Policy Objects

Group policies may also be set with PowerShell scripts. The following topic provides an overview of the PowerShell cmdlets available to do this:

- Reference for PowerShell cmdlets under GroupPolicy: https://learn.microsoft.com/en-us/powershell/module/grouppolicy

For example, a PowerShell script may be used to enable the FIPS cryptography mode policy, which is one of the operational prerequisites for the evaluated configuration. For more information on this prerequisite, see the section in this document, FIPS 140 Approved cryptography mode. To enable this policy, cut and paste the below PowerShell command into an administrator PowerShell window on the target machine.

```
Set-ItemProperty -Path Registry::HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\fipsAlgorithmPolicy `
-Name Enabled `
-Value "1" `
-Type DWord
```

2.4.4 Remote administration using PowerShell remoting

Windows may also be remotely managed using PowerShell Remoting. PowerShell Remoting must be performed over a HTTPS connection. The following topic provides information about PowerShell Remoting:

- Running Remote Commands: https://learn.microsoft.com/en-US/powershell/scripting/learn/remoting/running-remote-commands

3 Evaluated configuration

This section provides guidance on deploying the operating system and meeting the prerequisites for operating Windows in the evaluated configuration. To operate the system in a secure state, administrators must utilize the guidance in this section and in subsequent sections, where applicable to the local environment, to administer devices.

3.1 Installing the operating system

The operating system is pre-installed on Azure Stack Hub, Edge, and HCI products, and may be pre-installed on other devices in the evaluated configuration. When the device is turned on for the first time, the Out of Box Experience (OOBE) runs to complete the initial configuration. The operating system may also be installed from installation media. The method for creating or obtaining installation media depends on the Windows version and edition, with some editions available through the Microsoft 365 Admin Center only. The guidance below provides information on how to obtain .ISO files for the Windows 11, Windows 10, and Windows Server products in this evaluation. See the section, Windows installation documentation, for links to installation documentation, and Validating the hash value of installation media for information on validating the hashes of downloaded .ISO files.

- Use the Microsoft public download site to obtain the Windows 11 version 22H2 multi-edition .ISO:
  - Go to the Download Windows 11 website at https://www.microsoft.com/en-us/software-download/windows11.
  - Choose **Create Windows 11 installation media** to download the Media Creation Tool. **Note:** the Media Creation Tool is a Windows application and requires Windows in order to run.
  - Run the tool, accept the license agreement, and accept the default language and edition: **English (United States)** and **Windows 11.**
  - Choose to create an **ISO file.**
  - Save the .ISO file locally.

- Use the Microsoft public download site to obtain the Windows 10 version 22H2 Pro edition .ISO:
    - Go to the Download Windows 10 website at https://www.microsoft.com/en-us/software-download/windows10.
    - Choose **Create Windows 10 installation media** to download the Media Creation Tool. **Note:** the Media Creation Tool is a Windows application and requires Windows in order to run.
    - Run the tool, accept the license agreement, and choose the option **Create installation media** to create an .ISO file.
    - Choose the **64-bit (x64)** architecture to match the evaluated configuration.
    - Choose to create an **ISO file**.
    - Save the .ISO file locally.

- Use the Microsoft 365 Admin Center (MAC) to obtain all evaluated editions of Windows 11 version 22H2, Windows 10 version 22H2, or Windows Server 2022 products:
    - The Microsoft 365 Admin Center (MAC) has been introduced to replace the Volume Licensing Service Center (VLSC) in April of 2023. The scope of products available for download from the MAC depends on the license(s) purchased. For more information and for help navigating the MAC, see the Overview of the Microsoft 365 Admin Center at https://learn.microsoft.com/en-us/microsoft-365/admin/admin-overview/admin-center-overview?view=o365-worldwide.
    - Go to the Microsoft 365 Admin Center (MAC) at https://admin.microsoft.com/Adminportal/Home?#/subscriptions/vlnew/downloadsandkeys.
    - Log in with a valid MAC account.
    - Use the **Filters** control to locate a product, e.g. Windows or Windows Server.
    - Select a product from the list to display its description, e.g. Windows 11 22H2 Enterprise.
    - Click the **Download** control in the header to show the download controls.
    - Choose the correct architecture from the **CPU & file type** control. All products in this evaluation are **64-bit** architecture.
    - A list of available .ISO installation media is displayed. Use the copy control in the **SHA256 Hash** column to copy the hash value for an available .ISO file to the clipboard. This value may then be compared with the hash values published in this Admin Guide to identify an exact match for an .ISO file used in this evaluation.
    - Click the download arrow control in the **Download** column to begin the download process.

3.1.1 Validating the hash value of installation media

To validate that an .ISO installation file is the same as one used for this evaluation, customers may compare the SHA256 hash value of the .ISO file against the reference hash values listed in Windows variants and hardware platforms in scope. The PowerShell utility, Get-FileHash, can be used to generate a hash value for this comparison. See the topic below for information on using Get-FileHash.

- Get-FileHash: https://learn.microsoft.com/en-us/powershell/module/microsoft.powershell.utility/get-filehash

When using Get-FileHash, specify the SHA256 algorithm using the -Algorithm parameter.

3.1.2 Validating the build number of installation media

To identify the build number of the Windows product packaged in an .ISO installation file, view the properties of the Setup executable following these steps:

- Open the .ISO file in Windows Explorer by double-clicking on it.
- Locate the **setup.exe** file, typically in the root folder of the .ISO.
- **Right click** on setup.exe and choose **Properties**. (On some Windows 11 systems, first choose **Show more options** and then **Properties**.)
- Click on the **Details** tab of the Properties dialog for setup.exe.
- The **File version** field displays the build number.

3.1.3 Validating the build number on an existing Windows installation

If Windows is pre-installed on a device, the SystemInfo command line and PowerShell utility may be used to identify the edition and build number. The following topic provides more information on SystemInfo:

- SystemInfo: https://learn.microsoft.com/en-us/windows-server/administration/windows-commands/systeminfo

The SystemInfo fields, "OS Name" and "OS Version," provide the product edition and build number, respectively.

3.1.4 Windows installation documentation

The following topics provide more information on installing the Windows 11, Windows 10, and Windows Server products included in this evaluation. (The operating system is pre-installed on Azure Stack Hub, Edge, and HCI products.)

- Help for Windows 11 installation and updates: https://support.microsoft.com/en-us/topic/installation-updates-2f9c1819-310d-48a7-ac12-25191269903c#WindowsVersion=Windows_11
- Help for Windows 10 installation and updates: https://support.microsoft.com/en-us/topic/installation-updates-2f9c1819-310d-48a7-ac12-25191269903c#WindowsVersion=Windows_10
- Install, upgrade, or migrate to Windows Server: https://learn.microsoft.com/en-us/windows-server/get-started/install-upgrade-migrate

3.2 Operational prerequisites

The following operational prerequisites are required to operate Windows in the evaluated configuration.

3.2.1 Trusted platforms

Windows must be installed on trusted hardware platforms to ensure a secure operating state. See section 2, Evaluated editions and platforms, for details on which hardware platforms the evaluation was performed on.

3.2.2 Device administration

See the sections, Guidance specific to user roles and Guidance specific to Windows editions, for context on which administration solutions are in scope for which evaluated editions of Windows. One of the solutions listed in these sections must be used to administer the device, e.g., using a local account that is part of the Administrators group or enrolling the device for remote IT administration. For Windows 10 and Windows 11, IT administration is joining the device to a Windows domain or enrolling the device for modern device management (MDM) to receive MDM policies. For Windows Server and Azure products, IT administration is joining the device to a Windows domain to receive domain group policies.

3.2.3 Security updates

See the Security Target for a list of the Windows builds under evaluation. At the time of evaluation, when Windows is obtained through the channels described in section 3.1, Installing the operating system, the build matching the Security Target (major, minor, build and revision number) will be provided. As new Windows security and functionality updates are released over time, the default downloaded build from these channels may contain additional updates. In that case, the update history websites listed below provide detailed information about each update package. Update packages are known as KBs and are identified by a unique numeric identifier, their release date, and any updates to the build number, e.g., “July 12, 2022 – KB 5015814 OS Build 22000.795.” The same websites provide links to download KB update packages individually, providing administrators control over the updates they choose to install. Administrators may start with the initial release of a Windows version and then manually install KBs progressively until the Security Target build number is matched exactly.

The Windows Release Health website provides current information on known and resolved issues, release notes, and additional information for all Windows and Windows Server versions and editions in scope for this evaluation:

- Windows Release Health: https://learn.microsoft.com/en-us/windows/release-health/

For Azure products, the following release notes or security update pages provide details the open and resolved issues for each product:

- Azure Stack HCI release information: https://learn.microsoft.com/en-us/azure-stack/hci/release-information
- Azure Stack Edge release notes: https://learn.microsoft.com/en-us/azure/databox-online/azure-stack-edge-gpu-2203-release-notes
- Azure Stack Hub release notes: https://learn.microsoft.com/en-us/azure-stack/operator/release-notes?view=azs-2108
- Azure Stack Hub security updates: https://learn.microsoft.com/en-us/azure-stack/operator/release-notes-security-updates?view=azs-2108

To view a current list of updates for each version of Windows and Windows Server, visit its update history page:

- Windows 11 22H2 update history: https://support.microsoft.com/en-us/help/5018680
- Windows 10 version 22H2 update history: https://support.microsoft.com/en-us/help/5018682
- Windows Server 2022 update history: https://support.microsoft.com/en-us/help/5005454

3.2.4 Mode of operation

Windows has four modes of operation, as listed below. The evaluated configuration for Windows is the Operational Mode.

- Operational Mode: The normal mode of operation when the system has booted. This is the only evaluated mode.
- Debug Mode: The mode where the Windows boot options are configured to enable kernel debugging of the operating system.
- Safe Mode: The mode where Windows boot options are configured to start the operating system in a limited state where only essential programs are loaded.
- Non-Operational Mode: The mode where the system has not booted normally. In this mode the system is not operational and must be reinstalled.

3.2.5 FIPS 140 Approved cryptography mode

To match the evaluated configuration, Windows must be placed into the FIPS 140 Approved cryptography mode. This leverages FIPS 140 compliant cryptographic algorithms, including encryption, hashing, and signing algorithms. See the following topic for more information on FIPS 140 mode:

- System cryptography: Use FIPS compliant algorithms for encryption, hashing, and signing: https://learn.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/system-cryptography-use-fips-compliant-algorithms-for-encryption-hashing-and-signing

3.2.5.1 Configuring with PowerShell

PowerShell may be used to enable the FIPS cryptography mode policy. To enable this policy, cut and paste the below PowerShell command into an administrator PowerShell window on the target machine.

```
Set-ItemProperty -Path Registry::HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\fipsAlgorithmPolicy `
-Name Enabled `
-Value "1" `
-Type Dword
```

3.2.5.2 Configuring with MDM

Setting FIPS 140 mode may be configured by an IT Administrator using MDM and the Cryptography function of the Policy CSP. See the MDM solution documentation for detailed management actions. The following topic provides information on the Cryptography function of the Policy CSP:

- Policy CSP – Cryptography https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-csp-cryptography

3.2.5.3 Configuring with Group Policy

Setting FIPS 140 mode may be configured using Group Policy. Specifically, enable the following security policy:

Security Policy	**Policy Setting**
Windows Settings\Security Settings\Local Policies\Security Options\System Cryptography: Use FIPS compliant algorithms for encryption, hashing, and signing	Enabled

3.2.5.4 Configuring with the Windows Registry

The Windows registry may also be used to set FIPS mode, using the following registry key.

Registry Key	Value
HKLM\System\CurrentControlSet\Control\Lsa\FIPSAlgorithmPolicy\Enabled	1

3.2.6 Additional cryptography configuration

In addition to enabling FIPS 140 mode, the following specific configuration guidance must be followed:

- Support for TLS 1.0 and 1.1 must be explicitly disabled according to section 4.3.1, Supported TLS versions and key establishment parameters.
- Cipher suite selection must be configured according to section 4.3, Managing Transport Layer Security (TLS).
- SHA1 algorithms should be prioritized at the bottom of the algorithm negotiation list. See section 4.3, Managing Transport Layer Security (TLS), for implementation guidance.
- When using RSA schemes for key generation, RSA machine certificates must be configured with templates to use a minimum 2048-bit key length. See section 4.3.7, Configuring authentication schemes, for implementation guidance.

3.2.7 Code integrity configuration

The evaluated configuration enforces an example code integrity policy using Windows Defender Application Control (WDAC) that protects the integrity of executable code relevant to different evaluation requirements. With this policy, any violations of code integrity for those executables are recorded in auditable log events (#3033 and #3077, as documented in Audit events). This policy, included with Windows, is intended as a starting point for administrators. Administrators must modify the policy as appropriate for the operational environment and to allow approved apps to run. Once modified, administrators may establish the policy using WDAC PowerShell commands and may distribute it locally or remotely via Group Policy. The steps below outline the procedure to establish and enable the unmodified example policy on a local machine.

- Start an elevated PowerShell command window.

- Navigate to the directory where the Windows code integrity example policies are located:

```
PS C:\> cd "Windows\schemas\CodeIntegrity\ExamplePolicies"
```

- Convert the DefaultWindows_ Enforced.xml policy to a binary form with the specific binary filename listed below:

```
PS C:\Windows\schemas\CodeIntegrity\ExamplePolicies> convertFrom-CIPolicy -XmlFilePath ".\DefaultWindows_Enforced.xml" -
BinaryFilePath ".\{A244370E-44C9-4C06-B551-F6016E563076}.cip"
```

- For Windows 11 22H2:

  - Use the inbox CiTool to apply the policy, per the instructions documented in the following topic: https://learn.microsoft.com/en-us/windows/security/threat-protection/windows-defender-application-control/deployment/deploy-wdac-policies-with-script#deploying-policies-for-windows-11-22h2-and-above.

- For all other Windows variants in scope:
  - Copy the binary policy to C:\Windows\System32\CodeIntegrity\CiPolicies\Active:

```
PS C:\Windows\schemas\CodeIntegrity\ExamplePolicies> cp ".\{A244370E-44C9-4C06-B551-F6016E563076}.cip"
C:\Windows\System32\CodeIntegrity\CiPolicies\Active
```

- Restart the machine to apply the policy.

The following topics provide additional reference information on enforcing and deploying WDAC policies:

- Enforce Windows Defender Application Control policies: https://learn.microsoft.com/en-us/windows/security/threat-protection/windows-defender-application-control/enforce-windows-defender-application-control-policies
- Deploy Windows Defender Application Control policies by using Group Policy: https://learn.microsoft.com/en-us/windows/security/threat-protection/windows-defender-application-control/deploy-windows-defender-application-control-policies-using-group-policy

3.2.8 Device access configuration

The following configuration guidance must be followed to ensure device access is secured.

- Complex passwords must be required. See section 4.8, Managing authentication methods, for implementation guidance.
- The password reveal button must be disabled. See section 4.8, Managing authentication methods, for implementation guidance.
- Session locking must be enabled. See section 4.9, Managing screen lock, session timeout, and TPM lockout, for implementation guidance.

4 Managing evaluated features

This section provides management information for the features in scope for this evaluation, including configuration details and options for implementing them. Each subsection contains information relevant for a single feature or a group of related features.

4.1 Managing cryptography

Cryptography functions in Windows are managed by the Cryptography API: Next Generation (CNG). The notes below call out a list of specific management functions relevant to this Common Criteria evaluation that are handled automatically by CNG when Windows is configured in FIPS 140 Approved cryptography mode as described in the FIPS 140 Approved cryptography mode section. The sections that follow in this Administrative Guide provide complementary information on managing specific cryptography functions within Windows.

📝 **Notes**:

- Key management, including AES key size, storage, and destruction is handled automatically by CNG and requires no additional configuration. Keys are destroyed during Device Wipe.
- Windows generates asymmetric RSA keys using methods that meet FIPS-PUB 186-4 Appendix B.3, no additional configuration is necessary.
- Windows generates asymmetric ECC keys using methods that meet FIPS-PUB 186-4 Appendix B.4, no additional configuration is necessary.
- Windows performs RSA-based key establishment that meet NIST SP 800-56B, no additional configuration is necessary.
- Windows performs DSA-based key establishment that meets NIST SP 800-56B, no additional configuration is necessary.
- Windows performs elliptic curve-based key schemes that meet NIST SP 800-56A, no additional configuration is necessary.
- Windows generates random numbers according to NIST SP 800-90A, no additional configuration is necessary.
- Unprotected keys are not stored in non-volatile memory.
- Key lengths of keys used with certificates are configured in the certificate templates on the Certificate Authority used during enrollment and are not configured by the user or local administrator.
- There is no global configuration for hashing algorithms. The use of required hash sizes is supported. No additional configuration is necessary.
- Cryptographic Algorithm Validation Program (CAVP) testing was performed on the system cryptographic engine. Other cryptographic engines may have been separately evaluated but were not part of this CC evaluation.

4.2 Managing X.509 certificates

4.2.1 Client certificates and Certificate Authorities

By default, the Trusted Root Certification Authorities certificate store is configured with a set of public CAs that has met the requirements of the Microsoft Root Certificate Program.. No additional configuration is necessary to use the public Certificate Authorities. Additional private Certificate Authorities may be managed on the device using the solutions detailed in the subsections below.

📝 **Notes**:

- There is no configuration necessary to use client authentication on the device once a device has client authentication certificates.
- To destroy all keys on a device, including any imported keys, wipe the device.

4.2.1.1 Configuring with the Certutil command line utility

The Certutil command-line utility is available to dump and display certification authority (CA) configuration information, configure Certificate Services, backup and restore CA components, and verify certificates, key pairs, and certificate chains. The following topic provides more information on Certutil:

- Certutil: https://learn.microsoft.com/en-us/windows-server/administration/windows-commands/certutil

4.2.1.2 Configuring with PowerShell

The PKI cmdlets enable a local administrator to complete a variety of tasks related to public key infrastructure (PKI), including importing a client certificate and creating a self-signed certificate for testing TLS mutual authentication. See the following topics for more information:

- PowerShell PKI cmdlets reference: https://learn.microsoft.com/en-us/powershell/module/pki
- Import-Certificate cmdlet: https://learn.microsoft.com/en-us/powershell/module/pki/import-certificate
- New-SelfSignedCertificate cmdlet: https://learn.microsoft.com/en-us/powershell/module/pki/new-selfsignedcertificate

4.2.1.3 Configuring with MDM

Client certificates may be managed by the IT Administrator using MDM. See the MDM solution documentation for detailed management actions. The following topic describes the MDM policy for client certificate management, including deleting certificates:

- ClientCertificateInstall CSP - https://learn.microsoft.com/en-us/windows/client-management/mdm/clientcertificateinstall-csp

4.2.1.4 Configuring with the Windows UI

The following topic describes how to manually import a certificate:

- Import a Certificate: http://technet.microsoft.com/en-us/library/cc754489.aspx

The user obtains a client certificate for authentication by following the procedures in the following topic:

- Obtain a Certificate: https://technet.microsoft.com/en-us/library/cc754246.aspx

To destroy all keys on a device, including any imported, wipe the device.

4.2.1.5 Configuring certificate request fields

Certificate requests with specific fields such as "Common Name", "Organization", "Organizational Unit", and/or "Country" may be generated by apps using the Certificates.CertificateEnrollmentManager.CreateRequestAsync API. The following link provides the documentation for the API:

- CertificateEnrollmentManager.CreateRequestAsync | createRequestAsync method: https://learn.microsoft.com/en-us/uwp/api/Windows.Security.Cryptography.Certificates.CertificateEnrollmentManager

Similarly, the Network Device Enrollment Service (NDES) PowerShell cmdlet may be used to configure the same specific fields for the registration authority. The following topic provides more information on installing and using NDES:

- Install-AdcsNetworkDeviceEnrollmentService: https://learn.microsoft.com/en-us/powershell/module/adcsdeployment/install-adcsnetworkdeviceenrollmentservice

4.2.2 Trusted root certificates

By default, the Windows Trusted Root Certification Authorities certificate store is configured with a set of public CAs that has met the requirements of the Microsoft Root Certificate Program. No additional configuration is necessary. The subsections below provide solutions for additional management of trusted root certificates.

4.2.2.1 Configuring with MDM

Certificate trust relationships may be managed by the IT Administrator using MDM. See the MDM solution documentation for detailed management actions. The following topic describes the CSP that enables MDM to affect the policy for trusted root certificates:

- RootCATrustedCertificates CSP: https://learn.microsoft.com/en-us/windows/client-management/mdm/rootcacertificates-csp

4.2.2.2 Configuring with group policy

The following topic describes how to distribute certificates using group policy:

- Distribute Certificates to Client Computers by Using Group Policy: https://learn.microsoft.com/en-us/windows-server/identity/ad-fs/deployment/distribute-certificates-to-client-computers-by-using-group-policy

4.2.2.3 Configuring with PowerShell

PowerShell provides multiple cmdlets to manage certificates, as described below.

The remove-item PowerShell cmdlet may be used to delete certificates and wipe the private keys associated with the certificate. The following topic describes how to use the cmdlet:

- Remove-Item cmdlet: https://learn.microsoft.com/en-us/powershell/module/microsoft.powershell.management/remove-item

The import-pfxcertificate PowerShell cmdlet may be used to import a certificate and private key from a PFX file. The following topic describes how to use the cmdlet:

- Import-PfxCertificate cmdlet: https://learn.microsoft.com/en-us/powershell/module/pki/import-pfxcertificate

The export-pfxcertificate may be used to export a certificate and private key to a PFX file. The following topic describes how to use the cmdlet:

- Export-PfxCertificate cmdlet: https://learn.microsoft.com/en-us/powershell/module/pki/export-pfxcertificate

4.2.3 Certificate validation and revocation check

Windows automatically compares the distinguished name (DN) in the certificate to the expected distinguished name and does not require additional configuration. The reference identifiers for TLS are the DNS name or IP address of the remote server (ID payload), which is compared against the DNS name as the presented identifier in either the Common Name or the Subject Alternative Name (SAN) of the certificate.

Note: When validating a certificate with modern Windows applications the connection to a configured revocation server must be available or the validation will fail. This configuration cannot be changed for modern applications. The content below provides information on configuring certificate validation for other scenarios.

4.2.3.1 Configuring revocation lists and certificate trust lists using CertMgr

The CertMgr utility provides a local administrator the ability to manage certificates, certificate trust lists, and certificate revocation lists. The following topic provides more information on CertMgr:

- CertMgr: https://learn.microsoft.com/en-us/windows/win32/seccrypto/certmgr

4.2.3.2 Configuring certificate validation for EAP-TLS

The administrator configures certificate validation for network connections based on EAP-TLS using the “Set Up a Connection or Network” wizard. The following topic provides more information:

- EAP configuration: https://learn.microsoft.com/en-us/windows/client-management/mdm/eap-configuration

4.2.3.3 Configuring certificate validation for HTTPS in web browsers

For Internet Explorer:

- Open the **Control Panel**
- Navigate to **Internet Options** > **Internet Properties** > **Advanced Tab**
- Configure certificate validation using the checkbox options. The **Warn about certificate address mismatch** setting configures whether the Web address must match the certificate subject field and warns the user of a mismatch

For Microsoft Edge: The administrator cannot configure certificate validation for HTTPS for Microsoft Edge. If the Web address does not match the certificate subject field, then the user is warned of a mismatch.

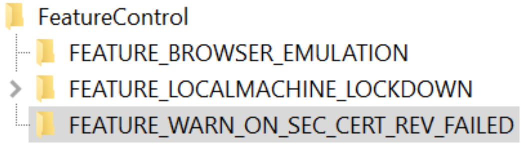
In all cases: When using HTTPS in a browsing scenario the user may choose to ignore a failed certificate validation and continue the connection.

4.2.3.4 Configuring warnings in Internet Explorer when the certificate revocation service is unavailable

If Internet Explorer is unable to check a certificate’s revocation status, for example, if no CRL or OCSP service is available, by default the browser will proceed to load the page without warning. Administrators may configure Internet Explorer to warn the user if the revocation check fails to complete by adding a key to the appropriate path in the Windows registry:

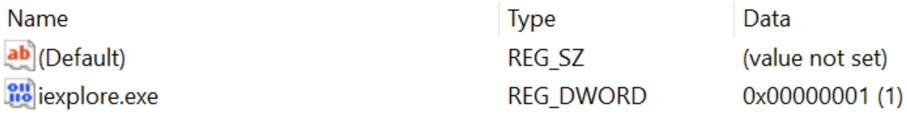
- Open the **Registry Editor** by typing **regedit** into the Windows search box or a command prompt
- Navigate to the registry path **Computer\HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Main\FeatureControl\**
- Right-click on **FeatureControl** and choose **New > Key** to create a new registry key

- Name the registry key **FEATURE_WARN_ON_SEC_CERT_REV_FAILED**



FeatureControl
FEATURE_BROWSER_EMULATION
FEATURE_LOCALMACHINE_LOCKDOWN
FEATURE_WARN_ON_SEC_CERT_REV_FAILED

- Right-click on the new key and choose **New > DWORD (32-bit) Value**
- Name the value **iexplore.exe**
- Double-click on the iexplore.exe value and set its **Value data** to **1**



Name	Type	Data
(Default)	REG_SZ	(value not set)
iexplore.exe	REG_DWORD	0x00000001 (1)

The following MSDN Blog topic provides more information on how Internet Explorer performs certificate revocation checks:

- Understanding Certificate Revocation Checks: https://blogs.msdn.microsoft.com/ieinternals/2011/04/07/understanding-certificate-revocation-checks/

4.2.3.5 Certificate validation and code signing

The administrator cannot configure certificate validation for code signing purposes.

4.3 Managing Transport Layer Security (TLS)

Note: All TLS settings such as cipher suites also apply to DTLS.

4.3.1 Supported TLS versions and disabling legacy protocols

Windows supports multiple TLS protocol versions. In the evaluated configuration, TLS 1.0 and 1.1 are disabled, per Microsoft's current security best practice guidance, and TLS 1.2 is set as the preferred protocol. The following topic lists the TLS support by Windows version:

- Protocols in TLS / SSL (Schannel SSP): https://learn.microsoft.com/en-us/windows/win32/secauthn/protocols-in-tls-ssl--schannel-ssp-

The following announcement explains the background for the guidance to disable TLS 1.0 and 1.1:

- Transport Layer Security 1.0 and 1.1 disablement: https://learn.microsoft.com/en-us/lifecycle/announcements/transport-layer-security-1x-disablement

A registry key provides administrators a solution to explicitly disable any version of TLS, e.g., TLS 1.0 and 1.1. For more information on using the registry to control available TLS protocol versions, see the following topic:

- TLS Registry Settings: https://learn.microsoft.com/en-us/windows-server/security/tls/tls-registry-settings

4.3.2 Available TLS ciphersuites

The ciphersuites listed in the Security Target correlate with those available in the evaluated configuration as noted in the table below. All ciphersuites listed are enabled by default, unless otherwise noted in the table. To enable ciphersuites that are not enabled by default, see the solutions for ciphersuite management in the subsequent sections of this guide.

Ciphersuites listed in the Security Target	Setting name for the ciphersuite in Windows
TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 5246	TLS_RSA_WITH_AES_128_CBC_SHA
TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 5246	TLS_RSA_WITH_AES_256_CBC_SHA
TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246	TLS_RSA_WITH_AES_128_CBC_SHA256
TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246	TLS_RSA_WITH_AES_256_CBC_SHA256
TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288	TLS_RSA_WITH_AES_128_GCM_SHA256
TLS_RSA_WITH_AES_256_GCM_ SHA384 as defined in RFC 5288	TLS_RSA_WITH_AES_256_GCM_SHA384
TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288	TLS_DHE_RSA_WITH_AES_128_GCM_SHA256
TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288	TLS_DHE_RSA_WITH_AES_256_GCM_SHA384
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256_P256
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256_P256 TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256_P384 TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256_P521
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384_P384
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384_P384

Ciphersuites listed in the Security Target	Setting name for the ciphersuite in Windows
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384

See the following topics for additional information on TLS ciphersuites available in the evaluated configuration.

- TLS Cipher Suites in Windows 11 22H2: https://learn.microsoft.com/en-us/windows/win32/secauthn/tls-cipher-suites-in-windows-11-v22h2
- TLS Cipher Suites in Windows Server 2022: https://learn.microsoft.com/en-us/windows/win32/secauthn/tls-cipher-suites-in-windows-server-2022
- TLS Cipher Suites in Windows 10 22H2: https://learn.microsoft.com/en-us/windows/win32/secauthn/tls-cipher-suites-in-windows-10-v22h2

The following topic provides additional contextual information on ciphersuites in TLS:

- Cipher Suites in TLS/SSL (Schannel SSP): https://learn.microsoft.com/en-us/windows/desktop/SecAuthN/cipher-suites-in-schannel

4.3.3 Available EAP-TLS ciphersuites

The EAP-TLS ciphersuites listed in the Security Target correlate with those available in the evaluated configuration as follows:

Ciphersuites listed in the Security Target	Setting name for the ciphersuite in Windows
TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 5246	TLS_RSA_WITH_AES_128_CBC_SHA
TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246	TLS_RSA_WITH_AES_128_CBC_SHA256
TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246	TLS_RSA_WITH_AES_256_CBC_ SHA256
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384

Ciphersuites listed in the Security Target	Setting name for the ciphersuite in Windows
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5430	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256

The following topic provides additional contextual information on ciphersuites in TLS:

- Cipher Suites in TLS/SSL (Schannel SSP): https://learn.microsoft.com/en-us/windows/desktop/SecAuthN/cipher-suites-in-schannel

4.3.4 Configuring with MDM

TLS ciphersuite priority and restricting use of certain cryptographic algorithms may be configured by the IT Administrator using MDM. See the MDM solution documentation for detailed configuration actions. The following topic describes the functions within the Policy CSP that may be leveraged to configure cryptography policies:

- Policy CSP – Cryptography: https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-csp-cryptography

4.3.5 Configuring with PowerShell

Administrators may manage TLS ciphersuites and elliptic curves using PowerShell cmdlets in the TLS module. In addition to enabling and disabling ciphersuites and elliptic curves, the cmdlets may also be used to specify a position in the priority list. The topics below provide more information on the TLS module and relevant cmdlets.

Note: PowerShell is the recommended method to configure ciphersuites on domain-joined computers.

- PowerShell TLS module reference: https://learn.microsoft.com/en-us/powershell/module/tls/
- Enable-TlsCipherSuite cmdlet: https://learn.microsoft.com/en-us/powershell/module/tls/enable-tlsciphersuite
- Disable-TlsCipherSuite cmdlet: https://learn.microsoft.com/en-us/powershell/module/tls/disable-tlsciphersuite
- Enable-TlsEccCurve cmdlet: https://learn.microsoft.com/en-us/powershell/module/tls/enable-tlsecccurve
- Disable-TlsEccCurve cmdlet: https://learn.microsoft.com/en-us/powershell/module/tls/disable-tlsecccurve

4.3.6 Configuring with group policy

Note: PowerShell is recommended over group policy to configure ciphersuites on domain-joined computers. See the PowerShell guidance that precedes this section.

The following topic explains how an administrator modifies the set of TLS ciphersuites for priority and availability using Group Policy:

- Prioritizing Schannel Ciphersuites: https://learn.microsoft.com/en-us/windows/win32/secauthn/prioritizing-schannel-cipher-suites

Note: The configuration for elliptic curves uses an SSL ciphersuite order list and an ECC curve order list displayed in the Group Policy Editor and the Local Security Policy Editor. Enable/order the desired ciphersuites in the first list and enable/order the elliptic curves in the second. For example, to configure only TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 ciphersuite and secp256r1 curve, edit the first list to only include TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 and the curve order list to only include secp256r1 (or NistP256 as it is shown in the policy editor). Additional ciphersuites and curves in each list will generate additional options in the client. A reboot of the system is required after changing the ciphersuite or elliptic curves configuration.

4.3.7 Configuring TLS client authentication

Key lengths of keys used with certificates are configured in the certificate templates on the Certificate Authority used during enrollment and require no additional configuration by a user or administrator for TLS client authentication. The following topic provides more information on configuring the certificate template for TLS client authentication:

- Configure the server certificate template: https://learn.microsoft.com/en-us/windows-server/networking/core-network-guide/cncg/server-certs/configure-the-server-certificate-template

The following table maps the certificate template details to the ciphersuites listed in the Security Target.

Ciphersuites (per Security Target)	Selections in the certificate template
TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 5246 TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 5246 TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246 TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246 TLS_RSA_WITH_AES_128_GCM_SHA384 as defined in RFC 5288 TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288 TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288	Provider Category = Key Storage Provider Algorithm Name = RSA

Ciphersuites (per Security Target)	Selections in the certificate template
TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288 TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289 TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289 TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289	
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289 TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289	Provider Category = Key Storage Provider Algorithm Name = ECDSA_P256
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289	Provider Category = Key Storage Provider Algorithm Name = ECDSA_P384

4.3.8 Managing signature algorithms and key length with the Windows registry

The signature algorithm set that is acceptable to the client (offered in the signature_algorithm extension during client hello) is configurable by editing the following registry key:

- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Cryptography\Configuration\Local\SSL\00010003

To constrain the Diffie-Hellman key length, edit the following registry key:

- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\KeyExchangeAlgorithms\Diffie-Hellman

With both registry keys, remove the algorithm(s) or key length(s) that should not be used. No additional algorithms other than the default set may be specified. For more information, see the following topic:

- TLS registry settings: https://learn.microsoft.com/en-us/windows-server/security/tls/tls-registry-settings

4.3.9 Configuring TLS mutual authentication

TLS mutual authentication is not enabled by default in IIS. To enable it, begin by installing the IIS Client Certificate Mapping Authentication role on the server. The following topic provides information on installing the IIS Client Certificate Mapping Authentication role via Server Manager.

- IIS Client Certificate Mapping Authentication: https://learn.microsoft.com/en-us/iis/configuration/system.webserver/security/authentication/iisclientcertificatemappingauthentication/

Also using Server Manager, two options for the website must be set to enable TLS mutual authentication:

- Require SSL
- Require Client Certificate

Next, configure many-to-one client certificate mapping on the server. The following topic explains how to do this using IIS Configuration Editor:

- Configure Many-to-One Client Mappings: https://learn.microsoft.com/en-us/troubleshoot/iis/configure-many-to-one-client-mappings

Finally, appropriate client certificates must be distributed to clients. The recommendation is to configure user certificate auto-enrollment on the domain server where Active Directory Domain Services (AD DS) is installed, then join the client computer to the same domain. After joining the domain and connecting to the server, e.g., with a browser, the user will be prompted to confirm and select the certificate provided by AD DS. Note that:

- Key strength for key establishment follows the certificate strength provided by the server.
- No configuration is needed besides enabling auto-enrollment with the certificate templates desired.
- The configuration is the same for different types of certificates.

The following topic provides information on configuring user certificate auto-enrollment:

- Configure certificate auto-enrollment: https://learn.microsoft.com/en-us/windows-server/networking/core-network-guide/cncg/server-certs/configure-server-certificate-autoenrollment

For more information on managing client certificates and information on configuring certificate templates, see the section, Client certificates and Certificate Authorities.

4.3.10 Choosing TLS in a web browser

Users may choose using TLS with HTTPS by using https in the URL typed into the browser.

4.3.11 Securing LDAP with TLS (LDAP-S)

Administrators may secure Lightweight Directory Access Protocol (LDAP) connections in an Active Directory environment with TLS. Enabling LDAP signing (LDAP-S) is the preferred solution for this. The following topic provides an overview of the technology and solutions to implement it that leverage Group Policy, local computer policy, and the Windows registry:

- How to enable LDAP signing in Windows Server: https://learn.microsoft.com/en-us/troubleshoot/windows-server/identity/enable-ldap-signing-in-windows-server

4.4 Managing IPsec and VPN connections

This section provides information on managing IPsec functionality and VPN connections on a device. The content in each subsection that follows addresses a specific aspect of IPsec or VPN configuration, most of which are exposed through the features of Windows Defender Firewall with Advanced Security. For an overview of all Windows Defender Firewall with Advanced Security features and a step-by-step deployment guide, see the following topic:

- Windows Defender Firewall with Advanced Security Deployment Guide: https://learn.microsoft.com/en-us/windows/security/threat-protection/windows-firewall/windows-firewall-with-advanced-security-deployment-guide

Note:

- File integrity self-testing of the Windows IPsec and VPN features occurs automatically. No configuration is necessary or possible.

4.4.1 Configuring IPsec firewall rules using Windows Defender Firewall with Advanced Security

The Windows Filtering Platform (WFP) is the IPsec Security Policy Database (SPD) for Windows. WFP starts automatically with Windows and must always be running to support IPsec scenarios. WFP features, including IPsec configuration, are exposed through the Windows Defender Firewall with Advanced Security features (the Windows Firewall), which may be configured through the Windows Firewall user interface, by Group Policy, with PowerShell, or with MDM. The following topics introduce how WFP operates and provide an overview of the IPsec configuration it enables:

- WFP Operation: https://learn.microsoft.com/en-us/windows/win32/fwp/basic-operation
- IPsec Configuration: https://learn.microsoft.com/en-us/windows/win32/fwp/ipsec-configuration

By establishing filter rules for inbound or outbound traffic, Windows Defender Firewall can prevent traffic other than IPsec-protected traffic to and from a device. Common rules include Protect, Bypass, and Discard rules, which map to the Windows Firewall rule types listed below. The rules below are examples only, as the Windows Firewall offers many configuration options for each rule type. Administrators are expected to configure the rules as appropriate for the environment and scenario. The documentation topics linked to below provide additional information beyond the examples on how to configure rules using the Windows Firewall.

- **Protect**: create a custom Connection Security Rule, using the following parameters as a model. This example outlines a scenario that protects outbound and inbound TCP traffic over port 80 between a specific authenticated web client and server. This example presumes a Certificate Authority is present to authenticate the certificates that identify the web client and server.
  - On the web client:
    - Create a new outbound Connection Security Rule and choose the Server-to-server type.
    - Set the appropriate remote IP address(es) the rule applies to, or choose Any IP address to apply the rule to all IP addresses and network interfaces.
    - Choose Require authentication for inbound and outbound connections, and then select the appropriate Authentication Method. Use the Advanced option to specify machine and user certificates. For example, an administrator may add a specific computer certificate as a First Authentication Method and a user certificate as a Second Authentication Method.
    - Apply the rule to the desired network profiles. For example, an administrator may choose to apply the rule to all profiles: Domain, Private, and Public networks.
    - Give the rule a name and confirm the operation.
    - Once the rule is created, right-click on it to modify its Properties. For this example, the following properties are modified: from the Protocols and Ports tab, choose protocol type TCP and set the remote port to 80.
  - On the web (IIS) server:
    - Create an identical Connection Security Rule as above.

- **Bypass**: create a Connection Security Rule, similar to an Authentication Exemption List rule except with the following parameters.
  - Type: Server to Server.
  - Choose to require authentication for inbound connections.
  - Choose to request authentication for outbound connections.

- **Discard**: create a custom Inbound Rule with the action to block the connection.

Rules may be created using the Windows user interface, Group Policy, PowerShell, or MDM; see the subsections below for more information. The different solutions for creating rules listed in the following subsections provide control over the order in which they are executed.

The following topics provide additional information on configuring common rules:

- Create an Authentication Exemption List Rule: https://learn.microsoft.com/en-us/windows/security/threat-protection/windows-firewall/create-an-authentication-exemption-list-rule
- Create an Inbound Program or Service Rule: https://learn.microsoft.com/en-us/windows/security/threat-protection/windows-firewall/create-an-inbound-program-or-service-rule
- Create an Outbound Program or Service Rule: https://learn.microsoft.com/en-us/windows/security/threat-protection/windows-firewall/create-an-outbound-program-or-service-rule

4.4.1.1 Configuring with the user interface

To configure firewall rules described above using the user interface, follow these steps:

- Click **Start.**
- Type **Windows Defender Firewall with Advanced Security** to search for the app and open it.
- Click on the desired rule type from the left pane.
- From the **Actions** pane, choose **New Rule...**.
- Follow the prompts and enter the required details.

4.4.1.2 Configuring with Group Policy

An IT Administrator may define and deploy firewall rules like those described above via Group Policy. The policy objects are found under:

- **Computer Configuration** > **Policies** > **Windows Settings** > **Security Settings** > **Windows Firewall with Advanced Security**

For more information on configuring Windows Firewall with Group Policy, see the following topics:

- Open the Group Policy Management Console to Windows Firewall with Advanced Security: https://learn.microsoft.com/en-us/windows/security/threat-protection/windows-firewall/open-the-group-policy-management-console-to-windows-firewall-with-advanced-security
- Checklist: Creating Group Policy Objects: https://learn.microsoft.com/en-us/windows/security/threat-protection/windows-firewall/checklist-creating-group-policy-objects (part of the Windows Defender Firewall with Advanced Security Deployment Guide)

4.4.1.3 Configuring with PowerShell

Windows Defender Firewall with Advanced Security may be administered using PowerShell cmdlets. This includes creating firewall rules. The following topics provide an overview and information on a selection of the cmdlets, including a description of the functionality and the syntax required for each:

- Windows Defender Firewall with Advanced Security Administration with Windows PowerShell: https://learn.microsoft.com/en-us/windows/security/threat-protection/windows-firewall/windows-firewall-with-advanced-security-administration-with-windows-powershell
- Get-NetFirewallRule: https://learn.microsoft.com/en-us/powershell/module/netsecurity/get-netfirewallrule
- New-NetFirewallRule: https://learn.microsoft.com/en-us/powershell/module/netsecurity/new-netfirewallrule
- Remove-NetFirewallRule: https://learn.microsoft.com/en-us/powershell/module/netsecurity/remove-netfirewallrule

The NetSecurity PowerShell module provides additional cmdlets to manage firewall rules and related IPsec functionality. For a high-level overview of all PowerShell cmdlets related to network security, see the following topic:

- NetSecurity: https://learn.microsoft.com/en-us/powershell/module/netsecurity/

4.4.1.4 Configuring with MDM

Windows Defender Firewall with Advanced Security may be administered using MDM and the Firewall CSP. For detailed information, see the following topic:

- Firewall CSP: https://learn.microsoft.com/en-us/windows/client-management/mdm/firewall-csp

4.4.2 Configuring and using VPN connections and the VPN client

This section provides information on how to configure and use VPN connections and the RAS IPsec VPN client in Windows.

📝 **Notes on VPN client configuration limitations:**

- To prevent standard users from modifying or deleting IPsec VPN connections, administrators may use the -AllUserConnection parameter when creating or configuring the connection profile, as described in the section, Configuring VPN using PowerShell.
- Windows supports Network Address Translation (NAT) traversal automatically as part of the IKEv1 and IKEv2 protocols. No configuration is needed or possible.
- Security association lifetime settings for IKEv2 may only be configured on the VPN gateway. No client configuration is needed or possible in the VPN client.

- For IKEv1 connections, Windows supports only main mode. It is not possible to configure IKEv1 to use aggressive mode.
- For IKEv1 connections, XAUTH is not supported.
- When using a pre-shared key, the secret value input into the client must match the secret value configured on the VPN server. The key must be at least 22 characters in length, but less than 256 characters. The key may be composed of any combination of upper and lower case letters, numbers, and the following special characters: “!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, and “)”.

4.4.2.1 Configuring VPN using VPN profiles and MDM

An IT Administrator may use MDM to manage VPN profiles remotely on devices. VPN profiles support a variety of configuration options, including the following scenarios:

- Lockdown profiles, which force all network traffic to go through the VPN tunnel, other than the traffic necessary to establish the VPN connection (always-on VPN).
- Profiles that establish a VPN connection when a specified app is launched.
- Profiles that specify a connection type, e.g., IKEv1 (L2TP), IKEv1 (L2TP) with a pre-shared key, or IKEv2. Note: when using IKEv1 with a pre-shared key, the key is generated by the user or administrator.
- Profiles that specify an encryption algorithm, e.g. AES128 or AES256.

The following topic describes the VPNv2 CSP, which may be used to implement any of the options listed above:

- VPNv2 CSP: https://learn.microsoft.com/en-us/windows/client-management/mdm/vpnv2-csp

The following topics provide additional context by describing the different VPN profile options, including lockdown profiles, the VPN connection types supported (e.g., IKEv1, IKEv1 with a pre-shared key, or IKEv2), choosing an authentication method (e.g., PAP, CHAP, MSCHAPv2, or EAP), options for auto-triggering profiles, and additional parameters:

- VPN profile options: https://learn.microsoft.com/en-us/windows/security/identity-protection/vpn/vpn-profile-options
- VPN connection types: https://learn.microsoft.com/en-us/windows/security/identity-protection/vpn/vpn-connection-type
- VPN auto-triggered profile options: https://learn.microsoft.com/en-us/windows/security/identity-protection/vpn/vpn-auto-trigger-profile

Additionally, the following topic provides an example of how to create an Extensible Authentication Protocol (EAP) configuration XML for the VPN profile:

- EAP configuration: https://learn.microsoft.com/en-us/windows/client-management/mdm/eap-configuration

4.4.2.2 Configuring VPN using PowerShell

The VpnClient PowerShell module provides a variety of cmdlets to create and manage the VPN client and VPN connections. The following topic provides an overview of all the cmdlets related to the VPN client:

- PowerShell VpnClient module: https://learn.microsoft.com/en-us/powershell/module/vpnclient/

The Add-VpnConnection and Set-VpnConnection cmdlets may be used to add new IPsec VPN connections and to specify their connection type (e.g., IKEv1 / L2TP, IKEv1 / L2TP with a pre-shared key, or IKEv2), the tunnel type (PPTP or L2TP in the evaluated configuration), the authentication method (e.g., PAP, CHAP, MSCHAPv2, EAP, or machine certificates), and many more parameters. The following topics provide more information on Add-VpnConnection and Set-VpnConnection:

- Add-VpnConnection: https://learn.microsoft.com/en-us/powershell/module/vpnclient/add-vpnconnection
- Set-VpnConnection: https://learn.microsoft.com/en-us/powershell/module/vpnclient/set-vpnconnection

📝 **Notes on using PowerShell cmdlets to add IPsec VPN connections:**

- When using IKEv1 with a pre-shared key, the key is generated by the user or administrator.
- The **-AllUserConnection** parameter may be used to store the IPsec VPN connection in the global phone book, making it available to all users and preventing standard users from deleting or modifying it.

The Set-VpnConnectionIPsecConfiguration cmdlet may be used to specify additional IPsec parameters, including the ESP encryption algorithm, (AES-GCM-128, AES-GCM-256, AES-CBC-128, and AES-CBC-256 in the evaluated configuration) and Diffie-Hellman (DH) group (DH groups 14, 19, 20, or 24 in the evaluated configuration). The following topic provides more information on Set-VpnConnectionIPsecConfiguration:

- Set-VpnConnectionIPsecConfiguration: https://learn.microsoft.com/en-us/powershell/module/vpnclient/set-vpnconnectionipsecconfiguration?view=win10-ps

4.4.2.3 Configuring a new VPN connection with the Windows UI

The following steps outline how to create and configure a new connection in the Windows RAS IPSec VPN client, including choosing options for IKEv1, IKEv1 with a pre-shared key, and IKEv2. The UI in the RAS IPsec VPN client supports the limited configuration options listed below; to configure more aspects of a VPN connection, use the PowerShell cmdlet solutions.

- Open the **Settings** app
- Navigate to **Network & Internet** and choose **VPN**
- Choose **Add a VPN connection** / **Add VPN**

- From **VPN provider,** choose the option for **Windows (built in)**
- Enter the **Connection name** as a text string
- Enter the **Server name** or address as a DNS name or an IP address. Note that the Subject name of the server's certificate must match the DNS name or IP address entered.
- (Optional) to specify the connection type, choose one of the following:
  - For IKEv1, from **VPN type** choose **L2TP/IPsec with certificate**
  - For IKEv1 with a pre-shared key, from **VPN type** choose **L2TP/IPsec with pre-shared key** and enter the text of the key
  - For IKEv2, from **VPN type** choose the **IKEv2** option and choose **Certificate** as the type of sign-in info (see the additional note below if using machine certificates)
- Choose the authentication method from **Type of sign-in info**
- Configure the user credentials as appropriate
- **Save** the connection
- **Note if using machine certificates:** To configure the VPN connection to use machine certificates, you must adjust the properties via the Adapter Properties feature.
  - Open **Network Connections** from the Control Panel (or, choose **Change adapter options** from the VPN panel in Settings).
  - Right-click on the VPN connection and select **Properties**.
  - Select the **Security** tab.
  - Select the **Use machine certificates** option.

4.4.2.4 Connecting to a VPN gateway with the Windows UI

The following steps outline how to connect to a VPN gateway once a VPN connection has been configured.

- Open the **Settings** app
- Navigate to **Network & Internet** and choose **VPN**
- Select the desired VPN connection and choose **Connect**
- If the device should always attempt to connect to this VPN connection, choose the **Connect automatically** checkbox

4.4.2.5 VPN client security association lifetime

SA lifetime settings for tunnel mode using the RAS IPsec VPN interface for IKEv1 and IKEv2 are configured on the VPN gateway. The following are the default values used for lifetimes by the RAS IPsec VPN Client:

- Main Mode

    - Lifetime in Seconds: 28800
- Quick Mode
    - Lifetime in Seconds: 3600
    - Lifetime in Packets: 2147483647
    - Lifetime in Kilobytes: 250000
    - Idle Duration in Seconds: 300

If a connection is broken due to network interruption, then the established SA remains in use until the SA lifetime limits are reached.

4.4.3 Configuring security association (SA) parameters for IPsec connections

Windows supports a variety of parameters to configure security associations (SAs) between devices when connecting via IPsec.

Notes on supported algorithms in the evaluated configuration:

- The following encryption algorithms are supported in the evaluated configuration: AES-GCM-128, AES-GCM-256, AES-CBC-128, and AES-CBC-256. Of those, only AES-CBC-128 and AES-CBC-256 may be used for IKEv1 and IKEv2 connections.
- For IKEv1 connections, the strength of the AES algorithm used for Phase 1 must be equal to or greater than the strength of the algorithm used for Phase 2. For IKEv2 connections, the strength of the AES algorithm used for the IKE_SA must be equal to or greater than the strength of the algorithm used for IKE_SA2. These attributes may be configured using Set-VpnConnectionIPsecConfiguration, as described in the section, Configuring VPN using PowerShell.
- The following authentication algorithms are supported in the evaluated configuration: HMAC-SHA1, HMAC-SHA-256, and HMAC-SHA-384; Diffie-Hellman Groups 14, 19, 20, and 24.
- The following signature algorithms are supported in the evaluated configuration: RSA, ECDSA P256, and ECDSA P384

4.4.3.1 Configuring IPsec in transport or tunnel mode

Windows supports tunnel mode IPsec security associations (SAs) via the IKEv2 protocol. When configuring an IPsec connection using one of the solutions in the section, Configuring and using VPN connections and the VPN client, choose IKEv2 for a tunnel mode connection. The following topic provides additional detail on securing IPsec connections using IKEv2:

- Securing end-to-end IPsec connections by using IKEv2: https://learn.microsoft.com/en-us/windows/security/threat-protection/windows-firewall/securing-end-to-end-ipsec-connections-by-using-ikev2

Windows supports transport mode IPsec SAs via the Layer 2 Tunneling Protocol (L2TP). When configuring an IPsec connection using one of the solutions in the section, Configuring and using VPN connections and the VPN client, choose L2TP for a transport mode connection.

For additional information about IPsec transport and tunnel mode scenarios in Windows, see the following topics:

- IPsec Configuration – Transport Mode: https://learn.microsoft.com/en-us/windows/win32/fwp/regular-transport-mode
- IPsec Configuration - Tunnel Mode: https://learn.microsoft.com/en-us/windows/win32/fwp/tunnel-mode

4.4.3.2 Configuring IPsec security association lifetime using PowerShell

SA lifetime may be configured on the client for IPsec connections. The PowerShell script below provides an example that an administrator may customize to define SA lifetime for an IPsec connection. The script defines an authorization proposal, a MainMode crypto set, a MainMode rule, a QuickMode crypto set, and a connection security rule. The SA lifetime is defined in both crypto sets using the -MaxMinutes parameter.

```
# Replace "DC=com, DC=EC, CN= EC-DC-CA " with the details from your Certificate Authority.  The details in this example use a
CA with the name “EC-DC-CA”, in the “EC.com” domain.  In this script, the elements of the CA name must be split and referred
to as below.  This example leverages machine certificates.

#Create the AuthProposal
$certprop = New-NetIPsecAuthProposal -machine -cert -Authority "DC=com, DC=EC, CN=EC-DC-CA"
$myauth = New-NetIPsecPhase1AuthSet -DisplayName "IKEv1TestEncSet" -proposal $certprop

#Create a MainModeCryptoSet
$algprop = New-NetIPsecMainModeCryptoProposal -KeyExchange DH19 -Encryption AES256 -Hash SHA256
$MainModeCS = New-NetIPsecMainModeCryptoSet -DisplayName "Main Mode DH CryptoSet" -Proposal $algprop -MaxMinutes 13

#Create the MainMode rule
New-NetIPsecMainModeRule -DisplayName "IKEv1 MM Rule" -RemoteAddress any -MainModeCryptoSet $MainModeCS.Name -Phase1AuthSet
$myauth.InstanceID

#Create a QuickModeCryptoSet with Encryption
$Enc = New-NetIpsecQuickModeCryptoProposal -Encapsulation ESP -ESPHash SHA256 -Encryption AES256 -MaxMinutes 11
$AES256CS = New-NetIpsecQuickModeCryptoSet -DisplayName "Quick Mode CryptoSet" -Proposal $Enc

#Create the Connection Security rule
New-NetIPsecRule  -DisplayName "IKEv1 QM Rule" -RemoteAddress any -Phase1AuthSet $myauth.InstanceID -InboundSecurity Require -
OutboundSecurity Request -KeyModule IKEv1 -QuickModeCryptoSet $AES256CS.Name
```

The following topics provide more information on the PowerShell cmdlets used in this script:

- New-NetIPsecAuthProposal: https://learn.microsoft.com/en-us/powershell/module/netsecurity/new-netipsecauthproposal
- New-NetIPsecPhase1AuthSet: https://learn.microsoft.com/en-us/powershell/module/netsecurity/new-netipsecphase1authset
- New-NetIPsecMainModeCryptoProposal: https://learn.microsoft.com/en-us/powershell/module/netsecurity/new-netipsecmainmodecryptoproposal

- New-NetIPsecMainModeCryptoSet: https://learn.microsoft.com/en-us/powershell/module/netsecurity/new-netipsecmainmodecryptoset
- New-NetIPsecMainModeRule: https://learn.microsoft.com/en-us/powershell/module/netsecurity/new-netipsecmainmoderule
- New-NetIpsecQuickModeCryptoProposal: https://learn.microsoft.com/en-us/powershell/module/netsecurity/new-netipsecquickmodecryptoproposal
- New-NetIpsecQuickModeCryptoSet: https://learn.microsoft.com/en-us/powershell/module/netsecurity/new-netipsecquickmodecryptoset
- New-NetIPsecRule: https://learn.microsoft.com/en-us/powershell/module/netsecurity/new-netipsecrule

All of the preceding cmdlets are contained within the PowerShell NetSecurity module, documented in the following reference topic:

- NetSecurity: https://learn.microsoft.com/en-us/powershell/module/netsecurity/

4.4.3.3 Configuring authentication signature algorithms

Windows supports the following signature algorithms for IPsec authentication with certificates:

- RSA
- ECDSA P256
- ECDSA P384

The New-NetIpsecAuthProposal PowerShell cmdlet is used to configure authentication techniques to be used and the signature algorithms to use with certificate authentication. The following topic provides more information:

- New-NetIpsecAuthProposal: https://learn.microsoft.com/en-us/powershell/module/netsecurity/new-netipsecauthproposal

The SubjectName and SubjectNameType options combined with the ValidationCriteria option for the New-NetIpsecAuthProposal cmdlet are used to configure how the name of the remote certificate will be verified. Use the DomainName value for the SubjectNameType parameter to configure either a domain name (DN) or fully qualified domain name (FQDN). Use the CommonName value for the SubjectNameType to configure an IP address. In addition, the RemoteAddress and SubjectName options for the New-NetIpsecAuthProposal cmdlet must be set to the IP address in the certificate.

4.4.3.4 Configuring certificate validation and revocation checks

Windows performs certificate validation by default when using IPsec with certificates. No configuration is necessary to enable certificate validation. To configure Windows to require certificate revocation checking, set the -CertValidationLevel parameter to RequireCrlCheck using the Set-NetFirewallSetting PowerShell cmdlet. The same PowerShell cmdlet may be used to configure a Group Policy object, which may then be distributed via Group Policy. The following topic provides more information:

- Set-NetFirewallSetting (see the CertValidationLevel parameter): https://learn.microsoft.com/en-us/powershell/module/netsecurity/set-netfirewallsetting

Note that extensions in a certificate specify the mechanism(s) to perform revocation checking for the particular certificate, either CRL or OCSP. The RequireCrlCheck setting applies to whichever revocation mechanism(s) are specified in certificates. Windows will automatically use a protected communication path with the entity providing the revocation information when such a communication path is configured in the certificate being validated. For example, if the CRL distribution point is a HTTPS URL in the extension in the certificate or if the OCSP server uses a HTTPS URL in the extension in the certificate then Windows will use HTTPS for the communication path with the CRL distribution point or the OCSP server.

4.4.3.5 Using pre-shared keys

Windows supports the use of pre-shared keys for IKEv1 / L2TP connections. The secret value for the pre-shared key must be a text-based value manually entered in the input field for a pre-shared key.   The secret value input into the client must match the secret value configured on the VPN server. The key must be at least 22 characters in length, but less than 256 characters. The key may be composed of any combination of upper- and lower-case letters, numbers, and the following special characters: "!", "@", "#", "$", "%", "^", "&", "*", "(", and ")".

4.5 Managing network connections

This section collects configuration information for networking, including both wired Local Area Network (LAN) connections and Wireless Local Area Network (WLAN or Wi-Fi) connections.

4.5.1 Enabling or disabling network connections with the Windows user interface

A user or administrator may enable or disable wired or wireless network connections by enabling or disabling the network devices that provide the connection using Device Manager. The steps to do so are:

- Open **Device Manager**
- Locate the **Network adapters** node and expand it
- Right-click on the appropriate network adapter and choose **Properties**
- Select the **Driver** tab
- Choose **Disable Device** to disable it or **Enable Device** to enable it

Wi-Fi connections may also be turned off using the Settings app without disabling the wireless network device. The steps to do so are:

- Open the **Settings** app
- Locate the **Network & internet** category and select it
- Turn the **Wi-Fi** setting on or off using the toggle control

4.5.2 Enabling or disabling network connections with PowerShell

Wired and wireless network connections may be enabled and disabled using PowerShell. The following topics provide information on how to enable and disable network adapters with PowerShell:

- Disable-NetAdapter: https://learn.microsoft.com/en-us/powershell/module/netadapter/disable-netadapter
- Enable-NetAdapter: https://learn.microsoft.com/en-us/powershell/module/netadapter/enable-netadapter

4.5.3 Configuring Wi-Fi access with MDM

The availability of Wi-Fi and several Wi-Fi settings may be configured by the IT Administrator using MDM. See the MDM solution documentation for detailed configuration actions. The following topics provide information on the two relevant CSPs for managing Wi-Fi with MDM:

- Policy CSP – Wifi: https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-csp-wifi
- Wifi CSP: https://learn.microsoft.com/en-us/windows/client-management/mdm/wifi-csp

4.5.4 Configuring allowed Wi-Fi networks with MDM

MDM may be used to specify the wireless networks (SSIDs) that a device may connect to. See the MDM solution documentation for detailed configuration actions. The following topic provides information on the relevant CSP for configuring allowed SSIDs.

- WiFi CSP: https://learn.microsoft.com/en-us/windows/client-management/mdm/wifi-csp

4.5.5 Configuring allowed Wi-Fi networks with Group Policy

Group policy may be used to configure wireless network policies. The following topic provides more information:

- Wireless access deployment (see Configure Wireless Network Policies): https://learn.microsoft.com/en-us/windows-server/networking/core-network-guide/cncg/wireless/e-wireless-access-deployment#bkmk_policies

4.5.6 Selecting a secure Wi-Fi connection with the Windows user interface

The following steps outline how to select and connect to an available Wi-Fi network using a higher level of security:

- Open the **Settings** app
- Locate the **Network & internet** category and select it
- Select **Wi-Fi**
- Choose an available secured network (identified by a lock icon)
- Follow the prompts to enter credentials (or equivalent) and connect to the network

If the Wi-Fi connection is unintentionally broken, Windows will automatically attempt to reconnect to the same connection when it becomes available again. No action is required by the user.

4.5.7 Configuring a Wi-Fi connection profile with the Windows user interface

The following steps provide information on how to manually configure a WLAN connection profile (e.g. EAP-TLS using WPA2-Enterprise) using the Windows user interface.

> 📝 **Note**: Configuration options may be different depending on the specific selections for your environment.

- Open the **Control Panel**
- Navigate to **Network and Sharing Center**
- Select **Set up a new connection or network**
- Select **Manually connect to a wireless network** to create a new WLAN profile
- In the **Network name** box, enter the name of the SSID to connect to
- From the **Security type** list, choose the security type (e.g. WPA2 Enterprise)
- Select **Next** and then **Change connection settings** to open the **<SSID name> Wireless Network Properties** window
- Select the **Security** tab
- Choose the authentication method from the **Choose a network authentication method** list (e.g. for EAP-TLS certificate-based authentication choose “Microsoft: Smart card or other certificate”)
- Select **Advanced Settings,** which will bring up a window with the **802.1X settings** tab
- Check the **Specify authentication mode** checkbox and then select the type of authentication certificate that has been configured (e.g. “User authentication” for a client authentication certificate)
- In the same window, configure the PMK caching if desired

- In the same window, configure pre-authentication for the WLAN network if desired
- Select **OK** to return to the **<SSID name> Wireless Network Properties** window
- On the **Security** tab click **Settings** to open the **Smart Card or other Certificate Properties** window
- Check **Use a certificate on this computer** and click the **Advanced** button to open the **Configure Certificate Selection** window
- Check the **Certificate Issuer** checkbox and then in the **Select one or multiple certificate issuers to be used for the certificate** list, check the Certificate Authority that issued the authentication certificate(s) configured on the client
- Click **OK** to return to the **Smart Card or other Certificate Properties** window
- Check the **Verify the server's identity by validating the certificate** if desired
- Check the **Connect to these servers…** checkbox if desired and enter the FQDN of acceptable WLAN server authentication server certificates in the textbox
- Check the Certificate Authority corresponding to the certificate issuer for the server certificate configured on the WLAN authentication server and then click **OK**
- Click **Close** to complete configuration for the WLAN connection profile

4.6 Managing personal hotspots

This section provides information on allowing or disallowing personal hotspots (wireless network bridging capability) on a device.

Note: The guidance in this section applies to all forms of authentication to the personal hotspot, including pre-shared keys.

4.6.1 Configuring with MDM

Sharing a personal hotspot may be enabled/disabled may be managed by the IT Administrator using MDM. See the MDM solution documentation for detailed management actions. The following topic describes the CSP that enables MDM to affect the policy for personal hotspots:

- Wi-Fi CSP: https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-csp-wifi#wifi-allowinternetsharing

4.6.2 Configuring with group policy

Administrators may use group policy to enable or disable the use of hotspot sharing. The policy objects are found under:

- **Computer configuration** > **Administrative templates** > **Network** > **Network Connections**

The two group policy objects are:

- Prohibit use of Internet Connection Sharing on your DNS Domain network
- Prohibit installation and configuration of Network Bridge on your DNS Domain network

4.6.3 Configuring with the Windows user interface

Standard users may enable or disable hotspot sharing via the Windows Settings app:

- Open the **Settings** app
- Locate the **Network & internet** category and select it
- Select **Mobile hotspot**
- Turn **Mobile hotspot** to **On** via the toggle
- Select options as appropriate, e.g., network name and password

4.7 Managing Bluetooth

This section provides configuration instructions for managing Bluetooth.

📝 **Notes**:

- No additional configuration is necessary to ensure the Bluetooth services provided before login are limited.
- No additional configuration is necessary to ensure Bluetooth pairing uses a protected communication channel.
- When a Bluetooth policy is changed by either MDM, Group Policy, or the Windows Registry, Bluetooth must be reset in order for the policy to apply. This can be achieved either by restarting the computer or turning the Bluetooth radio off and on as described in Enabling or disabling the Bluetooth device or all Bluetooth services with the Windows user interface.

4.7.1 Enabling or disabling the Bluetooth device or all Bluetooth services with MDM

IT Administrators may enable or disable individual Bluetooth device or all Bluetooth services via MDM using the Connectivity area of the Policy CSP:

- Policy CSP, Connectivity area: https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-csp-connectivity#connectivity-allowbluetooth.

4.7.2 Enabling or disabling the Bluetooth device or all Bluetooth services with the Windows user interface

An administrator may enable or disable the Bluetooth device radio or all Bluetooth services with the Windows Device Manager. The steps to do so are:

- Open **Device Manager**
- Locate the **Bluetooth** node and expand it
- Right-click on the appropriate Bluetooth adapter and choose **Properties**
- Select the **Driver** tab
- Choose **Disable Device** to disable it or **Enable Device** to enable it

4.7.3 Enabling or disabling the Bluetooth device or all Bluetooth services with PowerShell

An administrator may also use PowerShell cmdlets that leverage Windows Device Manager extensibility to enable or disable each Bluetooth adapter or all Bluetooth services. Use the Get-PnpDevice cmdlet to first gather the required information on the adapter (e.g., the unique identifier, InstanceID), then Enable- and Disable-PnpDevice to control the adapter. The following topics provide more information:

- Get-PnpDevice: https://learn.microsoft.com/en-us/powershell/module/pnpdevice/get-pnpdevice
- Enable-PnpDevice: https://learn.microsoft.com/en-us/powershell/module/pnpdevice/enable-pnpdevice
- Disable-PnpDevice: https://learn.microsoft.com/en-us/powershell/module/pnpdevice/disable-pnpdevice

Note: if PowerShell truncates the InstanceID in the output, use the Format-Table (ft) PowerShell utility along with the -width parameter to increase the column width and avoid truncation, for example:

```
Get-PnpDevice -Class 'Bluetooth' | Format-Table Status, Class, FriendlyName, InstanceId | Out-String -Width 800
```

4.7.4 Configuring Bluetooth properties with MDM

IT Administrators may configure multiple Bluetooth properties via MDM and the Bluetooth Policy CSP. These properties include:

- Enabling and disabling BR/EDR discoverable mode (via the AllowDiscoverableMode policy)
- Enabling or disabling LE advertising mode (via the AllowAdvertising policy)
- Allowing or disallowing out-of-band connections of pre-paired devices (via the AllowPrepairing policy)
- Changing the Bluetooth device name (via the LocalDeviceName policy)
- Listing and modifying the available Bluetooth profiles and services (via the ServicesAllowedList policy, when used according to its Usage Guide)

- Setting the encryption strength required when pairing Bluetooth devices, which may be used to prevent cryptographically weaker devices from being paired (via the SetMinimumEncryptionKeySize policy)

The following topic provides information on the Bluetooth Policy CSP, which provides the policies noted above to manage Bluetooth properties:

- Policy CSP – Bluetooth: https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-csp-bluetooth

4.7.5 Pairing Bluetooth devices with the Windows user interface

The Windows user interface provides a straightforward way for users to pair Bluetooth devices. For complete information, including the specific authorization screens, see the following topic:

- Connect a Bluetooth device: https://support.microsoft.com/en-au/help/15290/windows-connect-bluetooth-device

4.7.6 Managing Bluetooth Encryption Key Size with MDM

For devices that are enrolled in MDM, the Bluetooth Policy CSP is available to set the minimum Bluetooth encryption key size requirement. When this policy is enforced via MDM, Windows will read the encryption key size and reject the encrypted Bluetooth data connection if the minimum is not met but will not disconnect the remote device. To learn more about configuring minimum key size requirement for Bluetooth encryption via MDM, see this topic:

- Bluetooth Policy CSP - Windows Client Management | Microsoft Learn: https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-csp-bluetooth#setminimumencryptionkeysize

Some Bluetooth controllers (chips/radios) may enforce their own minimum Bluetooth encryption key size requirement. For example, all Intel Bluetooth radios require a minimum of 7 bytes for the encryption key length. Therefore, it is necessary to check whether the Bluetooth controller in the device does this.

4.7.7 Managing Bluetooth Encryption Key Size with the Windows Registry

The Windows registry may also be used to set the minimum Bluetooth encryption key size requirement. Specifically, to enforce the minimum encryption key size:

- Navigate to **HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Policies\Hardware\Bluetooth** and set the key **EnableMinimumEncryptionKeySize** value to 1 in a hexadecimal representation to enforce the minimum key size requirement. Once enabled, the system requires at least 7 bytes for the encryption key length.

- To change the minimum key size value, navigate to **HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\BTHPORT\Parameters** and set the **MinimumEncryptionKeySize** value to the desired value, which is from 1 byte to 16 bytes in a hexadecimal representation. Note that EnableMinimumEncryptionKeySize key must also be set to 1 for the MinimumEncryptionKeySize key to function.
- Navigate to **HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\BTHPORT\Parameters\Policy** and set the **MinimumEncryptionKeySize** value to the desired minimum encryption key size in hexadecimal representation. Note that EnableMinimumEncryptionKeySize key must also be set to 1 for the MinimumEncryptionKeySize key to function.
- Restart the Windows device. If you do not want to restart your Windows device, you can reset your Bluetooth device instead.

Note: if the keys mentioned above are not found, add them.

For more information on configuring minimum key size requirement for Bluetooth encryption via the Windows registry, see this topic:

- Windows guidance for Bluetooth key length enforcement: https://support.microsoft.com/en-au/topic/windows-guidance-for-bluetooth-key-length-enforcement-1b80c5b9-ddc1-31c7-1c3e-78e07c4fe877

4.7.8 Managing Bluetooth discovery mode with the Windows user interface

A user or administrator may enable or disable the Bluetooth discovery mode using Windows user interface. The steps to do so are:

On Windows 11:

- Open **Settings**
- Navigate to **Bluetooth & devices**
- Open the **Devices** panel
- Scroll down to find **More Bluetooth settings**
- Select the **Options** tab
- Check or uncheck the **Discovery** checkbox to enable or disable Bluetooth discovery mode
- Click **Apply** and **OK**
- Close all of the Bluetooth dialog boxes to apply the new setting. (If any Bluetooth dialog boxes remain open, the device will remain discoverable.)

On Windows 10:

- Open **Settings**

- Navigate to **Bluetooth & other devices**
- From the **Related settings** panel on the right, choose **More Bluetooth settings**
- Select the **Options** tab
- Check or uncheck the **Discovery** checkbox to enable or disable Bluetooth discovery mode
- Click **Apply** and **OK**
- Close all of the Bluetooth dialog boxes to apply the new setting. (If any Bluetooth dialog boxes remain open, the device will remain discoverable.)

4.7.9 Managing Bluetooth discovery mode with MDM

Bluetooth discovery mode may also be enabled or disabled using MDM described in the section 4.7.4 Configuring Bluetooth properties with MDM.

4.8 Managing authentication methods

The following sections provide multiple options for managing user authentication in Windows.

📝 **Notes**

- Services provided before logon are automatically limited, no configuration is necessary.
- For general best practices on setting password complexity, see the topic, Password must meet complexity requirements: https://learn.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/password-must-meet-complexity-requirements

4.8.1 Configuring password policies with MDM

Password policy may be configured using MDM. The DeviceLock and CredentialsUI policies, part of the larger Policy CSP, provide a variety of management functions for password policy. Note that some DeviceLock functions may not be available on Windows Home. The documentation for each function notes which editions the function may be used with. The following topics provide an overview of DeviceLock and a listing of all functions available:

- Policy CSP – overview, including a list of all DeviceLock policies: https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-configuration-service-provider
- Policy CSP – DeviceLock policy functions: https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-csp-devicelock
- Policy CSP – CredentialsUI: https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-csp-credentialsui

The following sample of DeviceLock policy functions address common needs of IT administrators. Each topic provides detailed information on the policy function, including which Windows editions it may be used with, and how to implement it.

- Requiring a password: Use the policy **DeviceLock/DevicePasswordEnabled**. For more information, see: https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-csp-devicelock#devicelock-devicepasswordenabled.
- Disabling the Guest account: Use the policy **LocalPoliciesSecurityOptions/Accounts_EnableGuestAccountStatus**. For more information, see: https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-csp-localpoliciessecurityoptions#localpoliciessecurityoptions-accounts-enableguestaccountstatus.
- Specifying password length: Use the policy **DeviceLock/MinDevicePasswordLength.** For more information, see: https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-csp-devicelock#devicelock-mindevicepasswordlength.
- Specifying password complexity: Use the policy **DeviceLock/MinDevicePasswordComplexCharacters**. For more information, see: https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-csp-devicelock#devicelock-mindevicepasswordcomplexcharacters.
- Specifying password expiration: Use the policy **DeviceLock/DevicePasswordExpiration**. For more information, see: https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-csp-devicelock#devicelock-devicepasswordexpiration
- Disabling the password reveal button: Use the policy **CredentialsUI/DisablePasswordReveal**. For more information, see: https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-csp-credentialsui#credentialsui-disablepasswordreveal

4.8.2 Configuring authentication, password, and PIN policies with group policy

Authentication policies, including password and Windows Hello PIN policies, may be configured using group policy. The following topics provide a sample of Account Policy functions that address common needs of IT administrators. The linked topics provide detailed information on the policy function and how to implement it.

- Configuring Windows to require a smart card or Windows Hello factor for interactive logon: Use the setting **Interactive logon: Require smart card** under **Computer Configuration\Windows Settings\Security Settings\Local Policies\Security Options**. For more information, see https://learn.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/interactive-logon-require-smart-card.
- Disabling the Guest account: Use the setting **Accounts: Guest account status** under **Computer Configuration\Windows Settings\Security Settings\Local Policies\Security Options**. For more information, see: https://learn.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/accounts-guest-account-status.
- Specifying the maximum number of authentication failures: Use the setting **Account lockout threshold** under **Computer Configuration\Windows Settings\Security Settings\Account Policies\Account Lockout Policy** to set the maximum number of failures. For more information, see: https://learn.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/account-lockout-threshold.

- Specifying the duration within which to enforce the maximum number of authentication failures: Use the setting **Account lockout duration** under **Computer Configuration\Windows Settings\Security Settings\Account Policies\Password Policy**. For more information, see: https://learn.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/account-lockout-duration.

The Password Policy set, part of the larger Account Policies, provide a variety of group policy management functions for password policy. The following sample of Password Policy functions addresses the common needs of IT administrators. The linked topics provide detailed information on the policy function and how to implement it.

- Overview of group policies available in the Password Policy set: https://learn.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/password-policy
- Specifying password length: Use the setting **Minimum password length** under **Computer Configuration\Windows Settings\Security Settings\Account Policies\Password Policy**. Note that the largest minimum length supported with this GPO is 14 characters. To configure a minimum length above 14, use the net.exe solution. For more information, see: https://learn.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/minimum-password-length.
- Specifying password complexity: Use the setting **Passwords must meet complexity requirements** under **Computer Configuration\Windows Settings\Security Settings\Account Policies\Password Policy**. For more information, see: https://learn.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/password-must-meet-complexity-requirements
- Specifying password expiration: Use the setting **Maximum password age** under **Computer Configuration\Windows Settings\Security Settings\Account Policies\Password Policy.** For more information, see: https://learn.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/maximum-password-age.
- Disabling the password reveal button: Use the setting **DisablePasswordreveal** under **Computer Configuration\Administrative Templates\Windows Components\Credential User Interface**.

Windows Hello PIN, i.e., a combination of username and PIN, may be configured via group policy. The relevant policy is:

- **Turn on Windows Hello PIN sign-in** under **Computer Configuration\Administrative Templates\System\Logon**

4.8.3 Configuring account and password policies with the net.exe accounts utility

The net.exe accounts utility may be used to manage some aspects of password and account lockout policy. The management functions available via net accounts include:

- Forcing user logoff after a time interval
- Minimum and maximum password age (days)

- Minimum password length
- Length of password history maintained
- Lockout threshold
- Lockout duration (minutes)
- Lockout observation window (minutes)

The following topic provides an overview of net accounts and how to use it:

- Net Commands on Operating Systems: https://learn.microsoft.com/en-us/troubleshoot/windows-server/networking/net-commands-on-operating-systems
- Net Accounts: https://learn.microsoft.com/en-us/previous-versions/windows/it-pro/windows-xp/bb490698(v=technet.10)

In addition to the parameters given in the referenced topic the following are also valid options:

- **/lockoutthreshold:*number*:** Sets the number of times a bad password may be entered until the account is locked out. If set to 0 then the account is never locked out.
- **/lockoutwindow:*minutes*:** Sets the number of minutes of the lockout window.
- **/lockoutduration:*minutes*:** Sets the number of minutes the account will be locked out for.

4.8.4 Configuring a Windows Hello PIN with the Windows user interface

To enable a Windows Hello PIN in place of passwords, follow the steps below. Note that a Windows Hello PIN may only be used for local interactive logon.

- Login to the user account
- Navigate to **Settings** > **Accounts** > **Sign-in options**
- Under the **PIN** heading tap the **Add** button
- Choose a new PIN value in the Set a PIN window. This requires entering a username and password to confirm the operation
- Sign out

4.8.5 Configuring smart card logon

Smartcard logon is supported on Windows domain-joined devices. IT administrators must enable an account for smartcard logon and issue a smartcard to a user. For more information about how smart card authentication works in Windows and how to enable it, see the following topic and its sub-topics:

- How Smart Card Sign-in Works in Windows: https://learn.microsoft.com/en-us/windows/security/identity-protection/smart-cards/smart-card-how-smart-card-sign-in-works-in-windows
- Get started with Virtual Smart Cards: Walkthrough Guide: https://learn.microsoft.com/en-us/windows/security/identity-protection/virtual-smart-cards/virtual-smart-card-get-started

For more information on how an IT administrator may configure Windows to require a smart card for interactive logon, see the following topic:

- https://learn.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/interactive-logon-require-smart-card

4.8.6 Logging on as an administrator

The Windows welcome screen provides administrators access to interactive logon for all authentication methods.

- From the welcome screen, press **any key** to log on. Windows defaults to the last user account and authentication method used.
- To choose a different authentication method, choose **Sign-in options** from the welcome screen and select from the authentication methods enabled on the device.
- To choose a different user account, select an account from the list of users in the corner of the welcome screen. On a domain-joined machine, choose **Other user** to enter the credentials of a different domain user.

The following topic provides additional detail on local Windows accounts, including the Administrator account:

- Local accounts: https://learn.microsoft.com/en-us/windows/security/identity-protection/access-control/local-accounts

4.9 Managing screen lock, session timeout, and TPM lockout

4.9.1 Configuring screen lock and session timeout with MDM

The Policy CSP provides IT Administrators with multiple options for configuring screen lock and session timeout. The documentation for each policy notes which Windows editions the policy may be applied to. The following topic provide an overview of the Policy CSP:

- Policy CSP: https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-configuration-service-provider

The DeviceLock node of the Policy CSP provides many policies to control device locking scenarios, including enabling screen lock, setting the lock timeout period, and more. See the following topic provides more information on each DeviceLock policy:

- Policy CSP – DeviceLock: Policy CSP – overview, including a list of all DeviceLock policies: https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-csp-devicelock

The AboveLock node of the Policy CSP provides policies to control the scope of notifications displayed above the lock screen when a device is locked. The following topic provides more information on each AboveLock policy:

- Policy CSP – AboveLock: https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-csp-abovelock

4.9.2 Configuring screen lock and session timeout with group policy

Screen lock and session timeout may both be configured by group policy. The relevant policy is:

- **Interactive logon: Machine inactivity limit** under **Computer Configuration\Windows Settings\Security Settings\Local Policies\Security Options**.

The following topic provides more information.

- Interactive logon: Machine inactivity limit: https://learn.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/interactive-logon-machine-inactivity-limit.

4.9.3 Configuring screen lock and session timeout with the Windows registry

The following topics provide information on registry settings which may be used to configure screen lock:

- ScreenSaveActive: https://learn.microsoft.com/en-us/previous-versions/windows/it-pro/windows-2000-server/cc978620(v=technet.10)
- ScreenSaverIsSecure: https://learn.microsoft.com/en-us/previous-versions/windows/it-pro/windows-2000-server/cc959646(v=technet.10)
- ScreenSaveTimeout: https://learn.microsoft.com/en-us/previous-versions/windows/it-pro/windows-2000-server/cc978621(v=technet.10)

4.9.4 Transitioning to the locked state with the Windows user interface

The user has two options to initiate a screen lock manually:

- Click on the **Start / Windows** button
- Click on the user profile picture and choose **Lock**

  - or –

- Type the Windows logo key + L

4.9.5 Configuring TPM lockout

TPM lockout, e.g., for smart card authentication scenarios, may be managed by the TPM MMC, Group Policy, or PowerShell. See the following topic for more information on the solutions available for managing TPM lockout:

- Manage TPM lockout: https://learn.microsoft.com/en-us/windows/security/information-protection/tpm/manage-tpm-lockout

4.10 Managing the logon banner

4.10.1 Configuring with MDM

The logon banner (title and text) displayed to users during interactive logon may be configured by the IT administrator using MDM. See the MDM solution documentation for detailed configuration actions. Use the LocalPoliciesSecurityOptions node of the Policy CSP to configure the logon banner. The following topic provides more information:

- Policy CSP – LocalPoliciesSecurityOptions: https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-csp-localpoliciessecurityoptions#localpoliciessecurityoptions-interactivelogon-displayuserinformationwhenthesessionislocked.

4.10.2 Configuring with group policy

The logon banner (title and text) displayed to users during interactive logon may be configured by the IT administrator using group policy objects. The following topics provide more information on how to configure each (title and text):

- Interactive logon: Message title for users attempting to log on: https://learn.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/interactive-logon-message-title-for-users-attempting-to-log-on
- Interactive logon: Message text for users attempting to log on: https://learn.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/interactive-logon-message-text-for-users-attempting-to-log-on

4.10.3 Configuring with the Windows registry

The logon banner (title and text) may also be configured by modifying the following Windows registry key values. Note that a reboot of the machine is required after modifying the keys to see the updated logon banner. The two registry keys are:

- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System\legalnoticecaption – affects the string that displays as the caption (title) of the legal notice dialog box
- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System\legalnoticetext – affects the string that displays as the text of the legal notice dialog box

4.11 Managing USB

4.11.1 Configuring via Device Manager

An administrator or user may enable or disable USB ports using the Windows Device Manager. To do so, follow these steps:

- Open the **Device Manager**
- Find the **Universal Serial Bus controllers** node and expand it
- Right-click on the **USB Root Hub** child node and select the **Properties** menu item to open the **USB Root Hub Properties** window
- Select the **Driver** tab and click the **Enable** or **Disable** button

4.11.2 Configuring with PowerShell

USB controllers may be enabled or disabled with PowerShell. The following topics describe the PowerShell cmdlets that may be used to disable USB controllers:

- Get-PnpDevice: https://learn.microsoft.com/en-us/powershell/module/pnpdevice/get-pnpdevice
- Disable-PnpDevice: https://learn.microsoft.com/en-us/powershell/module/pnpdevice/disable-pnpdevice
- Enable-PnpDevice: https://learn.microsoft.com/en-us/powershell/module/pnpdevice/enable-pnpdevice

4.11.3 Configuring with the Windows registry

The Windows registry may also be used to manage USB. Specifically, to disable the use of USB storage devices:

- Find the registry key, HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\UsbStor
- Change the Start REG_DWORD value to 4. (The default is 3.)
- Restart the machine.

For more information on the CurrentControlSet\Services registry tree, see this topic:

- HKLM\SYSTEM\CurrentControlSet\Services Registry Tree: https://learn.microsoft.com/en-us/windows-hardware/drivers/install/hklm-system-currentcontrolset-services-registry-tree

4.12 Managing updates

The following topic provides an overview of Windows Update and a matching set of FAQs:

- Windows Update FAQ: https://support.microsoft.com/en-us/help/12373/windows-update-faq

📝 **Note**: Windows Update may be configured to use enterprise Windows Server Update Services (WSUS) rather the default Microsoft Update. Configuring WSUS is outside the scope of this document.

4.12.1 Configuring using MDM

The IT administrator may configure Automatic Updates or Windows Server Update Services (WSUS) using MDM and the Policy CSP. See the MDM solution documentation for detailed configuration actions. The following topic describes Update node of the Policy CSP, which is used to configure update policies:

- Policy CSP – Update: https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-csp-update#update-policies

4.12.2 Configuring using group policy

The following topic provides details on configuring updates using domain group policy:

- Configure Group Policy Settings for Automatic Updates: https://learn.microsoft.com/en-us/windows-server/administration/windows-server-update-services/deploy/4-configure-group-policy-settings-for-automatic-updates

4.12.3 Configuring using the Server Configuration tool

The Server Configuration tool (sconfig.cmd) is available to configure Windows Update and other features on Windows Server installations. The following topic describes how to use sconfig to configure Windows Server, including the Windows Update settings:

- Configure a Server Core installation of Windows with Sconfig.cmd: https://learn.microsoft.com/en-us/windows-server/get-started/sconfig-on-ws2016#windows-update-settings

4.12.4 Checking for OS updates using the Windows user interface

To manually check for available Windows updates, follow these steps:

- Open Settings
- Navigate to Update & Security
- Choose **Windows Update** from the categories in the left navigation
- Click the **Check for updates** button

To check for installed updates, including any failed updates, follow these steps to view the device's update history:

- Open Settings
- Navigate to Update & Security
- Choose **Windows Update** from the categories in the left navigation
- Choose View update history

4.12.5 Querying for Windows version and hardware information

The Windows user interface and PowerShell may be used to query for Windows version information. The following topic provides details on how to do this via System Properties, System Info, the Command Prompt, and PowerShell:

- What version of Windows am I running? https://learn.microsoft.com/en-us/windows/client-management/windows-version-search

To query for hardware information and detailed Windows version information, leverage the systeminfo command. The following topic provides details on systeminfo:

- Systeminfo: https://learn.microsoft.com/en-us/windows-server/administration/windows-commands/systeminfo

IT administrators may also query for Windows version and hardware information with MDM. The DevDetail CSP returns many device-specific parameters to the device management server. The following topic provides detailed information on the DevDetail CSP:

- DevDetail CSP: https://learn.microsoft.com/en-us/windows/client-management/mdm/devdetail-csp

4.12.6 Installing Windows updates via the command line

Windows update packages may be installed manually via the command line interface on Windows and Windows Server editions. The Windows Update Standalone Installer (Wusa.exe) provides features that enable manual installation. For details on how to use Wusa.exe to, see the following topics:

- Patch a Server Core installation: https://learn.microsoft.com/en-us/windows-server/administration/server-core/server-core-servicing (for Server Core)
- Windows Update Standalone Installer in Windows: https://support.microsoft.com/en-us/help/934307/description-of-the-windows-update-standalone-installer-in-windows (for all editions)

4.12.7 Checking for Windows Store application updates

The following topic describes how to check for updates to applications installed from the Windows Store:

- Check for updates for apps and games from Windows Store: https://support.microsoft.com/en-us/help/4026259/microsoft-store-check-updates-for-apps-and-games

4.12.8 Querying for installed application version information

The PowerShell Get-AppxPackage cmdlet may be used to gather detailed information, including version, on all app packages installed for one or more users on a device. The following topic provides detailed information on the Get-AppxPackage cmdlet:

- Get-AppxPackage: https://learn.microsoft.com/en-us/powershell/module/appx/get-appxpackage?view=win10-ps

IT administrators may also leverage MDM to query for installed application version information. The following topic provides more information:

- EnterpriseModernAppManagement CSP: https://learn.microsoft.com/en-us/windows/client-management/mdm/enterprisemodernappmanagement-csp

4.13 Managing diagnostic data

Windows offers granular control over the diagnostic data that is collected and shared with Microsoft. This section provides multiple solutions to control diagnostic data. See the following topic for an overview of what diagnostic data is, how it benefits the ongoing development of Windows, and how customers may control it:

- Configure Windows diagnostic data in your organization: https://learn.microsoft.com/en-us/windows/privacy/configure-windows-diagnostic-data-in-your-organization

4.13.1 Configuring with group policy

The diagnostic data level and diagnostic components may be managed using group policy. The following topic outlines the different data level settings available and how to implement them via group policy:

- Configure Windows diagnostic data in your organization, Manage diagnostic data using Group Policy and MDM section: https://learn.microsoft.com/en-us/windows/privacy/configure-windows-diagnostic-data-in-your-organization#manage-diagnostic-data-using-group-policy-and-mdm

4.13.2 Configuring with MDM

The diagnostic data level and diagnostic components may be managed using MDM. The following topic outlines the different data level settings available and a high-level guide to implementing them via MDM:

- Configure Windows diagnostic data in your organization, Manage diagnostic data using Group Policy and MDM section: https://learn.microsoft.com/en-us/windows/privacy/configure-windows-diagnostic-data-in-your-organization#manage-diagnostic-data-using-group-policy-and-mdm

The System/AllowTelemetry policy within the Policy CSP is the solution for managing diagnostic data with MDM. The following topic provides additional information on the System/AllowTelemetry policy within the Policy CSP:

- Policy CSP – System, System/AllowTelemetry section: https://learn.microsoft.com/en-us/windows/client-management/mdm/policy-csp-system#system-allowtelemetry

4.13.3 Configuring with the Windows user interface

The diagnostic data level may be managed via the Settings app in the Windows UI.

- Open the **Settings** app
- Navigate to the **Privacy** / **Privacy & security** category
- Choose **Diagnostics & feedback**
- Choose the desired level under **Diagnostic data**

4.14 Managing the firewall

4.14.1 Configuring with PowerShell

The following topic describes how the Windows Firewall is managed using PowerShell cmdlets:

- Network Security Cmdlets in Windows PowerShell: https://learn.microsoft.com/en-us/powershell/module/netsecurity/

4.15 Managing domains

The following topic provides an overview of how to join a client computer to an Active Directory domain:

- How to Join Your Computer to a Domain: https://learn.microsoft.com/en-us/windows-server/identity/ad-fs/deployment/join-a-computer-to-a-domain

The name of the domain that is indicated for the Domain entry in step (2) should be provided by your IT administrator.

📝 **Note**: Choosing a domain is equivalent to enrolling with a MDM.

4.15.1 Configuring with PowerShell

The following topic describes how to join a computer to a domain using PowerShell:

- Add-Computer: https://learn.microsoft.com/en-us/powershell/module/microsoft.powershell.management/add-computer

4.16 Configuring the time server

A dedicated set of tools are available to administrators to manage the Windows Time Service and related settings, including configuring the name and address of the time server. The following topic describes the W32tm command, used to synchronize with a time server:

- Windows Time Service Tools and Settings: https://learn.microsoft.com/en-us/windows-server/networking/windows-time-service/windows-time-service-tools-and-settings

4.17 Managing apps and software restriction policies

📝 **Notes**:

- Administrators must exercise discretion when installing apps based upon examining app metadata describing claimed capabilities. For example: Installing apps that declare the shareduserCertificates app capability allows the app to approve exceptions for shared use or destruction of keys/secrets that were imported by another app. For more information on app capabilities, see the topic App capability declarations: https://learn.microsoft.com/en-us/windows/uwp/packaging/app-capability-declarations.

4.17.1 Configuring with Windows Defender Application Control

Windows Defender Application Control (WDAC) is the recommended approach for setting software restriction policies in Windows. WDAC offers a variety of policies that may be implemented via Group Policy or MDM. See the following topics for an overview, design guidance, and implementation guidance:

- Windows Defender Application Control: https://learn.microsoft.com/en-us/windows/security/threat-protection/windows-defender-application-control/windows-defender-application-control
- Deploy Windows Defender Application Control policies by using Group Policy: https://learn.microsoft.com/en-us/windows/security/threat-protection/windows-defender-application-control/deploy-windows-defender-application-control-policies-using-group-policy
- Deploy Windows Defender Application Control policies by using Microsoft Intune: https://learn.microsoft.com/en-us/windows/security/threat-protection/windows-defender-application-control/deploy-windows-defender-application-control-policies-using-intune

4.17.2 Configuring with AppLocker

In addition to Windows Defender Application Control (WDAC), AppLocker may also be used to manage software restriction policies. As a best practice, organizations should use WDAC to enforce software restriction policies at the most restrictive level possible, and then use AppLocker to fine-tune the restrictions to an even lower level if needed. The following topic provides more information on the relationship between WDAC and AppLocker:

- Windows Defender Device Guard with Applocker, including its relationship to WDAC: https://learn.microsoft.com/en-us/windows/security/threat-protection/windows-defender-application-control/windows-defender-device-guard-and-applocker

AppLocker policies may be configured locally via PowerShell or SecPol, or configured and deployed remotely via Group Policy. The scope of policies includes enabling or disabling access to the Microsoft Store. The following topics provide information on both local and remote options for AppLocker management:

- AppLocker Overview: https://learn.microsoft.com/en-us/windows/security/threat-protection/windows-defender-application-control/applocker/applocker-overview
- Administer AppLocker: https://learn.microsoft.com/en-us/windows/security/threat-protection/windows-defender-application-control/applocker/administer-applocker
- Use the AppLocker Windows PowerShell cmdlets: https://learn.microsoft.com/en-us/windows/security/threat-protection/windows-defender-application-control/applocker/use-the-applocker-windows-powershell-cmdlets
- Configure access to Microsoft Store: https://learn.microsoft.com/en-us/windows/configuration/stop-employees-from-using-microsoft-store

4.17.3 Configuring with enterprise app management and MDM

IT Administrators may also configure some aspects of app management via MDM. The following topic provides an overview of enterprise app management functionality in Windows:

- Enterprise app management: https://learn.microsoft.com/en-us/windows/client-management/mdm/enterprise-app-management

The EnterpriseModernAppManagement CSP is used to implement enterprise app management policies via MDM. The following topic provides more information:

- EnterpriseModernAppManagementCSP: https://learn.microsoft.com/en-us/windows/client-management/mdm/enterprisemodernappmanagement-csp

4.17.4 Installing and uninstalling apps with the Windows user interface

Windows supports a wide variety of applications. Applications may be installed from the Microsoft Store, from installation media provided by app developers, or from installation files downloaded from the internet. The following topic provides a summary of the options for installing applications in Windows:

- How to install programs on Windows: https://support.microsoft.com/en-us/help/4026235/windows-10-how-to-install-programs

The following topic provides a summary of the options for removing applications from Windows:

- Repair or remove programs in Windows: https://support.microsoft.com/en-us/help/4028054/windows-10-repair-or-remove-programs

The following topic provides information on installing programs from online sources:

- How to install programs from online sources on Windows: https://support.microsoft.com/en-us/windows/how-to-install-programs-from-online-sources-on-windows-10-a503e8b6-e45b-fd5a-f4c5-5a08c8bd9821

4.18 Developing applications

This section of the operational guidance is provided for application developers and is not related to the management functions that may be performed by the administrator or user roles described in the other sections of this document.

Developers may use Microsoft Visual Studio 2022 for development of applications. The following is a link to documentation for Microsoft Visual Studio 2022:

- Visual Studio : https://learn.microsoft.com/en-us/visualstudio/ide/visual-studio-ide

Applications developed in Microsoft Visual Studio 2022 will by default have the /GS flag set. The following is a link to documentation about the /GS flag in Microsoft Visual Studio:

- /GS (Buffer Security Check) : https://learn.microsoft.com/en-us/cpp/build/reference/gs-buffer-security-check

4.19 Managing self-test and health attestation

Note: Windows automatically checks file integrity of files related to security functions. No configuration is necessary or possible.

4.19.1 Configuring with MDM

Health attestation policies may be managed to determine the health of enrolled Windows and Windows Server devices using MDM. The following topic provides details on the correct CSP to use to manage health attestation policies with MDM and to assess if an enrolled device has booted to a trusted and compliant state:

- Device HealthAttestation CSP: https://learn.microsoft.com/en-us/windows/client-management/mdm/healthattestation-csp

4.19.2 Accessing health attestation logs

Administrators may audit the events listed for FPT_GVI_EXT.1 and FPT_ML_EXT.1 as measurements of the health subsystem. The device will create a health attestation log every time the system boots. The logs are found in the following directory:

- %windir%\Logs\MeasuredBoot

The logs are in a binary format. To decode the logs, use the TPM Platform Crypto Provider and Toolkit utility, available for download from Microsoft here:

- TPM Platform Crypto Provider and Toolkit: https://www.microsoft.com/en-us/download/details.aspx?id=52487&from=http%3A%2F%2Fresearch.microsoft.com%2Fen-us%2Fdownloads%2F74c45746-24ad-4cb7-ba4b-0c6df2f92d5d%2F

4.20 Managing audit policy and event logs

This section provides more information for IT administrators on event auditing functionality in Windows, including solutions available to adjust logging scope and settings. This information is provided to enable IT Administrators to implement security monitoring and forensics required by their organization.

The following log locations are always enabled:

- Windows Logs -> System
- Windows Logs -> Setup
- Windows Logs -> Security (for startup and shutdown of the audit functions and of the OS and kernel, and clearing the audit log)

For additional background on event logging and configuring audit policies in Windows, see these topics:

- Event types in Windows: https://learn.microsoft.com/en-us/windows/win32/eventlog/event-types
- Basic security audit policies: https://learn.microsoft.com/en-us/windows/security/threat-protection/auditing/basic-security-audit-policies
- Advanced security audit policies, including categories of audits in the Windows Security log: https://learn.microsoft.com/en-us/windows/security/threat-protection/auditing/advanced-security-auditing

4.20.1 Managing audit policy with the Auditpol command

The Auditpol command displays information about and performs functions to manipulate audit policies, including selecting events by attribute to audit. The following topic provides an overview of the Auditpol command, including a list of all its commands and their syntax:

- Auditpol: https://learn.microsoft.com/en-us/windows-server/administration/windows-commands/auditpol

The Auditpol set command sets the per-user audit policy, system audit policy, or auditing options. The following topic provides information on how to use Auditpol set:

- Auditpol set: https://learn.microsoft.com/en-us/windows-server/administration/windows-commands/auditpol-set

For example, to enable all audits in the given subcategories of the Windows Logs -> Security log run the following commands at an elevated command prompt:

- Logon operations:

```
auditpol /set /subcategory:”Logon” /success:enable /failure:enable
```

- Audit policy changes:

```
auditpol /set /subcategory:"Audit Policy Change" /success:enable /failure:enable
```

- IPsec operations:

```
auditpol /set /subcategory:”IPsec Main Mode” /success:enable /failure:enable
auditpol /set /subcategory: “IPsec Quick Mode” /success:enable /failure:enable
```

- Configuring IKEv1 and IKEv2 connection properties:

```
auditpol /set /subcategory:"Filtering Platform Policy Change" /success:enable /failure:enable
auditpol /set /subcategory:"Other Policy Change Events" /success:enable /failure:enable
```

- Registry changes (modifying TLS ciphersuite priority):

```
auditpol /set /subcategory:"Registry" /success:enable /failure:enable
```

4.20.2 Managing audit policy with the Secpol snap-in

The local security policy snap-in utility (secpol.msc) is used as an alternative to the auditpol utility for managing security policy settings, including audits. The following topic provides information on administering security policy settings, including how to use the security policy sanp-in:

- Administer security policy settings: https://learn.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/administer-security-policy-settings

4.20.3 Managing audit policy with the Wevtutil utility

Wevtutil is a system utility that performs many of the management functions related to system and audit logons including the following:

- Configuring local audit storage capacity, including enabling or disabling automatic log overwriting
- Configuring audit rules, including enabling and disabling optional event logging by feature area
- Configuring log retention policy between automatic overwrite or retain
- Enabling analytic and debug logs (e.g. Microsoft-Windows-CodeIntegrity/Verbose)
- Enumerating log names
- Clearing logs

See the following topic for more info on Wevtutil:

- Wevtutil: https://learn.microsoft.com/en-us/windows-server/administration/windows-commands/wevtutil

4.20.4 Retrieving and viewing audit logs using the Windows Event Viewer

The Windows Event Viewer may be used to retrieve and view audit logs on a local or remote computer. To launch Event Viewer, follow these steps:

- From the **Start** menu, navigate to **Windows Administrative Tools**.
- Choose **Event Viewer**.
- Use the tree in the left pane to navigate between different Windows, application, and services logs.
- Event Viewer defaults to the Local Computer. To connect to a remote computer that you have administrative rights to, right-click on **Event Viewer (Local)** in the tree and choose **Connect to Another Computer…** .

- To view any Analytic or Debug logs, select the option **Show Analytic and Debug Logs** from the **View** menu.

4.20.5 Retrieving and viewing audit logs using PowerShell

The PowerShell Get-WinEvent cmdlet may be used to retrieve and view audit logs on a local or remote computer. For information on how to use Get-WinEvent, see the following topic:

- Get-WinEvent https://learn.microsoft.com/en-us/powershell/module/microsoft.powershell.diagnostics/get-winevent

4.20.6 Configuring System Access Control Lists to audit registry changes

In addition to enabling audit policy as noted in the preceding sections, administrators may configure auditing for changes to the Windows registry by changing the audit permissions of the System Access Control List (SACL) for the appropriate registry key. For general information on System Access Control Lists, see the following topic:

- Access Control Lists: https://learn.microsoft.com/en-us/windows/desktop/secauthz/access-control-lists

In the evaluated configuration, the following registry key have been configured for auditing:

- \REGISTRY\MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System

4.20.6.1 Configuring registry key SACLs using the registry editor

The following process describes how to set the SACL for a registry object via the registry editor:

1. Start the registry editor tool by executing the command **regedit.exe** as an administrator
2. Navigate to the registry path for the key that should be audited, right-click the key's node and select **Permissions…** on the key's context menu to open the **Permissions** dialog
3. Click the **Advanced** button to open the **Advanced Security Settings** dialog, click on the **Auditing** tab and click the **Add** button to open the **Auditing Entry** dialog
4. Click the **Select a principal** to open the **Select User or Group** dialog to select a user (e.g. everyone or a specific administrator) and click the OK button.
5. Choose the desired audits using the **Type**, **Applies to** and **Basic Permissions** attributes and click **OK**
6. Click **OK** on the **Advanced Security Settings** dialog
7. Click **OK** on the **Permissions** dialog

4.20.6.2 Configuring registry key SACLs using PowerShell

PowerShell may also be used to set the SACL on a registry key. See the following topics for more information:

- Get-Acl cmdlet: https://learn.microsoft.com/en-us/powershell/module/microsoft.powershell.security/get-acl
- Set-Acl cmdlet: https://learn.microsoft.com/en-us/powershell/module/microsoft.powershell.security/set-acl

4.20.6.3 Configuring a global SACL for registry changes with Auditpol

The Auditpol command may be used to set a global SACL, for example, to generate auditable events for all registry changes. See the following topic for more information:

- Auditpol resourceSACL: https://learn.microsoft.com/en-us/windows-server/administration/windows-commands/auditpol-resourcesacl

5 Audit events

This section provides a reference for Windows log data that may be used for security auditing and forensic investigation, as required for this Common Criteria evaluation. Audit events by scenario groups events by security function, indexed by the corresponding Common Criteria requirement. Audit event field details provides additional details on each event, indexed by event ID.

5.1 Audit events by scenario

The following table lists the set of auditable events in scope for this Common Criteria evaluation, ordered per the selections in the Security Target document. Prerequisite steps are noted for each scenario, for example, setting specific audit policy or enabling specific event log configuration options. For more information on the utilities used to configure audit policy or event logs, see the section Managing audit policy. Reference the subsequent section, Audit event field details, for the message and field details for each event ID listed in this table.

Requirement(s)	**Scenario(s)**	**Additional Audit Contents**	**Log Name: Event ID (Details)** ***Prerequisite Steps***
Events required by FAU_GEN, including management functions.			

Requirement(s)	**Scenario(s)**	**Additional Audit Contents**	**Log Name: Event ID (Details)** ***Prerequisite Steps***
FAU_GEN.1.1 FAU_GEN.1.1 (WLAN) FAU_GEN.1.1 (VPN)	Start-up and shut-down of the audit functions		Security: **4608** (Startup) Security: **1100** (Shut down) *Enable logging of startup and shutdown events with the following command:* ***auditpol /set /subcategory:"Security State Change" /success:enable /failure:enable***
FAU_GEN.1.1	Authentication events (Success/Failure)		Security: **4624** (Authentication attempt, successful) Security: **4625** (Authentication attempt, failed)
FAU_GEN.1.1	Use of privileged/special rights events (Successful and unsuccessful security, audit, and configuration changes)		Security: **4670** (WRITE_DAC) Security: **4656** (All other object access writes)
FAU_GEN.1.1	Privilege or role escalation events (Success/Failure)		Security: **4673** (Success) Security: **4674** (Failure)
FAU_GEN.1.1	File and object events (Successful and unsuccessful attempts to create, access, delete, modify, modify permissions)		Security: **4656**
FAU_GEN.1.1	User and Group management events (Successful and unsuccessful add, delete, modify, disable)		Security: **4720** (add user) Security: **4732** (add user to group) Security: **4726** (delete user) Security: **4733** (delete user from group) Security: **4731** (add group) Security: **4734** (delete group) Security: **4735** (modify group)

Requirement(s)	**Scenario(s)**	**Additional Audit Contents**	**Log Name: Event ID (Details)** ***Prerequisite Steps***
			Security: **4738** (modify user account) Security: **4725** (disable user)
FAU_GEN.1.1	Audit and log data access events (Success/Failure),		Security: **4673** (Success and failure)
FAU_GEN.1.1	Cryptographic verification of software (Success/Failure)		Security: **2** (Success) Security: **3** (Failure)
FAU_GEN.1.1	Attempted application invocation with arguments (Success/Failure)		Security: **3038** (WDAC/Device Guard, Success) Security: **8020** (AppLocker, Success) Security: **3077** (WDAC/Device Guard, Failure) Security: **8022** (AppLocker, Failure)
FAU_GEN.1.1	System reboot, restart, and shutdown events (Success/Failure)		Security: **4608** (Startup) Security: **1100** (Shut down)
FAU_GEN.1.1	Kernel module loading and unloading events (Success/Failure)		**Windows Boot Configuration Log** (Boot kernel module success) Security: **3038** (Other kernel modules, Success) **Recovery Screen** (Failure, Boot kernel module) Security: **3004** (Failure, other kernel modules),
FAU_GEN.1.1	Administrator or root-level access events (Success/Failure)		Security: **4624** (Success) Security: **4625** (Failure)
FAU_GEN.1.1	Lock and unlock a user account		Security: **4740** (Lock / disable) Security: **4767** (Unlock / re-enable)

Requirement(s)	**Scenario(s)**	**Additional Audit Contents**	**Log Name: Event ID (Details)** ***Prerequisite Steps***
FAU_GEN.1.1 FMT_SMF_EXT.1 (Function #1)	Enable/disable screen lock		Security: **4663**
FAU_GEN.1.1 FMT_SMF_EXT.1 (Function #2)	Configure screen lock inactivity timeout		Security: **4663**
FAU_GEN.1.1 FMT_SMF_EXT.1 (Function #3)	Configure local audit storage capacity		Security: **4657** (ObjectValueName: **MaxSize**) *Enable logging by configuring the SACL for the following registry key for auditing. See Configuring System Access Control Lists to audit registry keys.* ***\REGISTRY\MACHINE\SYSTEM\ControlSet001\Services\EventLog\Security***
FAU_GEN.1.1 FMT_SMF_EXT.1 (Function #4)	Configure minimum password Length		Security: **4739** *Enable logging for authentication policy change events with the following command:* ***auditpol /set /subcategory:"Authentication Policy Change" /success:enable /failure:enable***
FAU_GEN.1.1 FMT_SMF_EXT.1 (Function #9)	Configure lockout policy for unsuccessful authentication attempts through [timeouts between attempts, limiting number of attempts during a time period]		Security: **4739** *Enable logging for authentication policy change events with the following command:* ***auditpol /set /subcategory:"Authentication Policy Change" /success:enable /failure:enable***
FAU_GEN.1.1 FMT_SMF_EXT.1 (Function #10)	Configure host-based firewall		Security: **4950**

Requirement(s)	**Scenario(s)**	**Additional Audit Contents**	**Log Name: Event ID (Details)** ***Prerequisite Steps***
FAU_GEN.1.1 FMT_SMF_EXT.1 (Function #11)	Configure name/address of directory server to bind with		System: **3260** (non-virtual device) System: **4096** (virtual device)
FAU_GEN.1.1 FMT_SMF_EXT.1 (Function #12)	Configure name/address of remote management server from which to receive management settings		System: **3260** (non-virtual device) System: **4096** (virtual device)
FAU_GEN.1.1 FMT_SMF_EXT.1 (Function #14)	Configure audit rules		Security: **4719** *Enable events for audit policy changes with the following command:* ***auditpol /set /subcategory:"Audit Policy Change" /success:enable /failure:enable***
FAU_GEN.1.1 FMT_SMF_EXT.1 (Function #15)	Configure name/address of network time server		System: **37**
FAU_GEN.1.1 FMT_SMF_EXT.1 (Function #16)	Enable/disable automatic software update		Security: **4657** (ObjectValueName: **NoAutoUpdate**) *Enable logging by configuring the SACL for the following registry key for auditing. See Configuring System Access Control Lists to audit registry keys*. ***\REGISTRY\MACHINE\SOFTWARE\Policies\Microsoft\Windows\WindowsUpdate\AU***
FAU_GEN.1.1 FMT_SMF_EXT.1 (Function #17)	Configure Wi-Fi interface		Security: **6420** (enable) Security: **6422** (disable)

Requirement(s)	**Scenario(s)**	**Additional Audit Contents**	**Log Name: Event ID (Details)** ***Prerequisite Steps***
FAU_GEN.1.1 FMT_SMF_EXT.1 (Function #18)	Enable/disable Bluetooth interface		Security: **6420** (enable) Security: **6422** (disable)
FAU_GEN.1.1 FMT_SMF_EXT.1 (Function #19)	Enable/disable local area network interface		Security: **6420** (enable) Security: **6422** (disable)
FAU_GEN.1.1 FMT_SMF_EXT.1 (Function #19)	Configure USB interfaces		Security: **6420** (enable) Security: **6422** (disable)
FAU_GEN.1.1 FMT_SMF_EXT.1 (Function #20)	Manage Windows Diagnostics settings		
FAU_GEN.1.1 FMT_SMF_EXT.1 (Function #20)	Configure remote connection inactivity timeout		Security: **4657** (ObjectValueName: **MaxIdleTime**) *Enable logging by configuring the SACL for the following registry key for auditing. See Configuring System Access Control Lists to audit registry keys.* ***\REGISTRY\MACHINE\SOFTWARE\Microsoft\Windows NT\Terminal Services***

Events required by the WLAN extended package.			
FCS_TLSC_EXT.1 (WLAN)	Failure to establish an EAP-TLS session.	Reason for failure.	System: **36871** Microsoft-Windows-CAPI2/Operational: **11, 30**
FCS_TLSC_EXT.1 (WLAN)	Establishment/termination of an EAP-TLS session.	Non-TOE endpoint of connection.	System: **36880** (Establishment) Microsoft-Windows-SChannel-Events/Perf: **1793** (Termination)
FIA_X509_EXT.1 (WLAN)	Failure to validate X.509v3 certificate	Reason for failure of validation.	Applications and Services Logs > Microsoft > Windows >CAPI2 > Operational: **11**
FIA_X509_EXT.6(WLAN)	Attempts to load certificates.		Applications and Services Logs > Microsoft > Windows >CAPI2 > Operational: **11**
FIA_X509_EXT.6 (WLAN)	Attempts to revoke certificates.		Applications and Services Logs > Microsoft > Windows >CAPI2 > Operational: **11**
FPT_TST_EXT.1 (WLAN)	Execution of the set of TSF self-tests.		System: **20**
FPT_TST_EXT.3 (WLAN)	Detected integrity violation of TSF self-tests.	The TSF binary file that caused the integrity violation.	System: **20**
FTA_WSE_EXT.1 (WLAN)	All attempts to connect to access points.	Identity of access point being connected to as well as success and failures (including reason for failure.	Microsoft-Windows-WLAN-AutoConfig/Operational log event: **8001** (successful WLAN connection) **8002** (WLAN connection failure) **8003** (successful WLAN disconnection) **8004** (wireless network blocked) **11005** (wireless security succeeded) **11006** (wireless security failed) **12013** (failure due to user account)

FTP_ITC_EXT.1 (WLAN)	All attempts to establish a trusted channel.	Identification of the non-TOE endpoint of the channel.	System: **36880** (Establishment) Microsoft-Windows-SChannel-Events/Perf: **1793** (Termination)

Events required by the VPN extended package.			
FAU_GEN.1 (VPN) FMT_SMF.1 (VPN) (Function #1)	Specify VPN gateways to use for connections		Microsoft-Windows-VPN-Client/Operational: **10001** (Success) Microsoft-Windows-VPN-Client/Operational: **10002** (Failure)
FAU_GEN.1 (VPN) FMT_SMF.1 (VPN) (Function #2)	Specify IPsec VPN Clients to use for connections		Microsoft-Windows-VPN-Client/Operational: **10001** (Success) Microsoft-Windows-VPN-Client/Operational: **10002** (Failure)
FAU_GEN.1 (VPN) FMT_SMF.1 (VPN) (Function #3)	Specify IPsec-capable network devices to use for connections]		Microsoft-Windows-VPN-Client/Operational: **10001** (Success) Microsoft-Windows-VPN-Client/Operational: **10002** (Failure)
FAU_GEN.1 (VPN) FMT_SMF.1 (VPN) (Function #4)	Specify client credentials to be used for connections		Microsoft-Windows-VPN-Client/Operational: **10001** (Success) Microsoft-Windows-VPN-Client/Operational: **10002** (Failure)
FAU_GEN.1 (VPN) FMT_SMF.1 (VPN) (Function #5)	Configure the reference identifier of the peer		Microsoft-Windows-VPN-Client/Operational: **10001** (Success) Microsoft-Windows-VPN-Client/Operational: **10002** (Failure)

FAU_SEL.1	All modifications to the audit configuration that occur while the audit collection functions are operating.		Security: **4719**
FCS_IPSEC_EXT.1	Decisions to DISCARD or BYPASS network packets processed by the TOE.	Presumed identity of source subject. Identity of destination subject. Transport layer protocol, if applicable. Source subject service identifier, if applicable. The entry in the SPD that applied to the decision.	Security: **5152** (Discard), **5156** (Bypass), **5157** (Protect)
FCS_IPSEC_EXT.1	Failure to establish an IPsec SA.	Reason for failure. Non-TOE endpoint of connection (IP address) for both successes and failures.	Security: **4653, 4654**
FCS_IPSEC_EXT.1	Establishment/Termination of an IPsec SA.	Non-TOE endpoint of connection (IP address) for both successes and failures.	Security: **4650, 4655, 5451, 5452**
FPT_TUD_EXT.1	Initiation of the update. Any failure to verify the integrity of the update.		Setup: **1** (Initiation) Setup: **3** (Failure)
Events required by the Bluetooth Protection Profile.			
FAU_GEN.1(BT) FIA_BLT_EXT.1	Failed user authorization of a Bluetooth device.	User authorization decision (e.g., user rejected connection, incorrect PIN entry).	System: **16**
FAU_GEN.1(BT) FIA_BLT_EXT.1	Failed user authorization for local Bluetooth service.	Bluetooth address.	System: **16**

FAU_GEN.1(BT) FIA_BLT_EXT.2	Initiation of Bluetooth connection.	Bluetooth address and name of device.	System: **8** *Event 8 contains the remote device ID only. To log the remote device name as well, configure a SACL for the following registry key for auditing, which will generate Event 6416. See Configuring System Access Control Lists to audit registry keys.* ***\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\DeviceClasses\{00f40965-e89d-4487-9890-87c3abb211f4}***
FAU_GEN.1(BT) FIA_BLT_EXT.2	Failure of Bluetooth connection.	Reason for failure.	System: **16** System: **49**

5.2 Audit event field details

The following table maps the event IDs referenced in the preceding tables to specific Windows logs, including details on where to find the information in the log, the specific log message, and the fields included. The fields in the table refer to the hierarchical field names used in Event Viewer event data, on the Details tab, when the Friendly View radio button is selected. The field names also correspond to the node names in XML files provided as evidence. The Message values correspond to the message displayed in the General tab.

Event ID	Log Location	Message	Fields
1	**Windows Logs->Setup**	Initiating changes for package	**System**->**TimeCreated**[**SystemTime**]: <Date and time of event> **System->Provider[Name]**: <Type of event> **System->Security[UserID]**: <Subject identifier > **System->Level**: <Outcome as Success or Failure>
2	**Windows Logs -> System**	Package was successfully changed to the Installed state	**System**->**TimeCreated**[**SystemTime**]: <Date and time of event> **System->Provider[Name]**: <Type of event> **System->Security[UserID]**: <Subject identifier > **System->Level**: <Outcome as Success or Failure>

Event ID	Log Location	Message	Fields
3	**Windows Logs->Setup**	Windows update could not be installed because … "The data is invalid"	**System**->**TimeCreated**[**SystemTime**]: <Date and time of event> **System->Provider[Name]**: <Type of event> **System->Security[UserID]**: <Subject identifier > **System->Level**: <Outcome as Success or Failure>
8	**Windows Logs -> System**	The remote adapter (*[Remote device Bluetooth address]*) successfully paired with the local adapter.	**System**->**TimeCreated**[**SystemTime**]: <Date and time of event> **System->Provider[Name]**: <Type of event> **System->Security[UserID]**: <Subject identifier > **System->Level**: <Outcome as Success or Failure>
11	**Microsoft-Windows-CAPI2/Operational**	Build Chain	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Provider[Name]**: <Type of event> **System->Level**: <Outcome as Success or Failure> **System->Security[UserID]**: <Subject identity> **UserData->Result**: <Reason for failure of validation>
16	**Windows Logs -> System**	The mutual authentication between the local Bluetooth adapter and a device with Bluetooth adapter address (*[Remote device Bluetooth address]*) failed.	**System**->**TimeCreated**[**SystemTime**]: <Date and time of event> **System->Provider[Name]**: <Type of event> **System->Security[UserID]**: <Subject identifier > **System->Level**: <Outcome as Success or Failure>
20	**Windows Logs -> System**	The last boot's success was <LastBootGood event data>.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Provider[Name]**: <type of event> **System-> Security[UserID]**: <subject identifier > **EventData->LastBootGood**: <Outcome as true or false indicating if the kernel-mode cryptographic self-tests and RNG initialization succeeded or failed>

Event ID	Log Location	Message	Fields
30	**Microsoft-Windows-CAPI2/Operational**	Verify Chain Policy	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Provider[Name]**: <Type of event> **System->Level**: <Outcome as Success or Failure> **System->Security[UserID]**: <Subject identity> **UserData->CertVerifyCertificateChainPolicy->Certificate**: <Issuer Name and Subject Name of certificate>
49	Windows Logs -> System	Windows rejected a connection from your Bluetooth device (%2) because the resulting encryption key size was smaller than the system required minimum.	System->TimeCreated[SystemTime]: <Date and time of event> System->Provider[Name]: <Type of event> System->Level: <Outcome as Success or Failure> System->Task: <Type of event> System->Keywords: <Outcome as Success or Failure> System->Computer: <Subject identifier>
1006	**Applications and Services Logs-> Microsoft->Windows-> CertificateServicesClient-Lifecycle-User -> Operational** **Applications and Services Logs -> Microsoft -> Windows -> CertificateServicesClient-Lifecycle-System -> Operational**	A new certificate has been installed.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **System->Security[UserID]**: <Subject identifier>
1100	**Windows Logs->Setup**	The event logging service has shut down	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **N/A**: <Subject identifier>

Event ID	Log Location	Message	Fields
1793	**Microsoft-Windows-SChannel-Events/Perf**	A TLS Security Context handle is being deleted	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Provider[Name]**: <type of event> **System-> Security[UserID]**: <subject identifier > **System->Level**: <Outcome as Success or Failure> **EventData->ContextHandle:** <non-TOE endpoint>
3004	**Application and Services Logs->Microsoft->Windows->CodeIntegrity**	Windows is unable to verify the image integrity of the file <pathname> because the file hash could not be found on the system.	Subcategory: Security State Change
3033	**Application and Services Logs->Microsoft->Windows->CodeIntegrity->Operational**	Code Integrity determined that a process <process name> attempted to load <executable file name> that did not meet the Enterprise signing level requirements	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Provider[Name]**: <Type of event> **System->Level**: <Outcome as Success or Failure> **System->Security[UserID]**: <Subject identifier>
3038	**Application and Services Logs->Microsoft->Windows->CodeIntegrity->Verbose**	Code Integrity started validaging image header of <kernel module pathname> file	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Provider[Name]**: <Type of event> **System->Level**: <Outcome as Success or Failure> **System->Security[UserID]**: <Subject identifier>

Event ID	Log Location	Message	Fields
3077	**Application and Services Logs->Microsoft->Windows->CodeIntegrity->Operational**	Code Integrity determined that a process <process name> attempted to load <target process name> that did not meet the Enterprise signing level requirements or violated code integrity policy. <policy ID>.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Provider[Name]**: <Type of event> **System->Level**: <Outcome as Success or Failure> **System->Security[UserID]**: <Subject identifier>
3260	**Windows Logs -> System**	This computer has been successfully joined to domain '<Domain Name>'.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Provider[Name]**: <Type of event> **System->Level**: <Outcome as Success or Failure> **System->Security[UserID]**: <Subject identifier>
4096	**Windows Logs -> System**	The machine <Machine Name> successfully joined the domain <Domain Name>.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Provider[Name]**: <Type of event> **System->Level**: <Outcome as Success or Failure> **System->Security[UserID]**: <Subject identifier>
4608	**Windows Logs->Security** Subcategory: Security State Change	Startup of audit functions	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **N/A**: <Subject identifier>
4624	**Windows Logs->Security** Subcategory: Logon	An account was successfully logged on.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->TargetUserSid**: <Subject identifier>

Event ID	Log Location	Message	Fields
4625	**Windows Logs->Security** Subcategory: Logon	An account failed to log on.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->TargetUserSid**: <Subject identifier>
4651	**Windows Logs->Security** Subcategory: IPsec Main Mode	IPsec main mode security association was established. A certificate was used for authentication.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **System->Computer**: <Subject identifier> **EventData->RemoteMMPrincipalName:** <Presumed identity of source subject> **EventData->RemoteAddress**<Non-TOE endpoint of connection> **EventData->LocalMMPrincipalName:** <Identity of destination subject> **N/A:** <Transport layer protocol> **EventData->MMFilterID: <**The entry in the SPD that applied to the decision> **N/A:**<Reason for failure>
4652	**Windows Logs -> Security** IPsec Main Mode	IPsec main mode negotiation failed	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **System->Computer**: <Subject identifier> **EventData->RemoteMMPrincipalName:** <Presumed identity of source subject> **EventData->RemoteAddress**<Non-TOE endpoint of connection> **EventData->LocalMMPrincipalName:** <Identity of destination subject> **N/A:** <Transport layer protocol> **EventData->MMFilterID: <**The entry in the SPD that applied to the decision> **EventData->FailureReason:**<Reason for failure>

Event ID	Log Location	Message	Fields
4653	**Windows Logs -> Security** IPsec Main Mode	IPsec main mode negotiation failed	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **System->Computer**: <Subject identifier> **EventData->RemoteMMPrincipalName:** <Presumed identity of source subject> **EventData->RemoteAddress**<Non-TOE endpoint of connection> **EventData->LocalMMPrincipalName:** <Identity of destination subject> **N/A:** <Transport layer protocol> **EventData->MMFilterID: <**The entry in the SPD that applied to the decision> **EventData->FailureReason:**<Reason for failure>
4654	**Windows Logs -> Security** IPsec Quick Mode	IPsec quick mode negotiation failed	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **System->Computer**: <Subject identifier> **EventData->RemoteAddress, RemotePort:** <Presumed identity of source subject> <Non-TOE endpoint of connection> **EventData->LocalAddress,LocalPort:** <Identity of destination subject> **EventData->Protocol:** <Transport layer protocol> **EventData->QMFilterID,MMSAID,TunnelId: <**The entry in the SPD that applied to the decision> **EventData->FailureReason:**<Reason for failure>

Event ID	Log Location	Message	Fields
4655	**Windows Logs->Security** Subcategory: IPsec Main Mode	IPsec main mode security association ended	**System->TimeCreated[SystemTime]**: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **System->Computer**: <Subject identifier> **EventData->RemoteAddress**<Non-TOE endpoint of connection> **EventData-MMSAID**:<Presumed identity of source subject><Identity of destination subject>**<**The entry in the SPD that applied to the decision> **N/A:** <Transport layer protocol> **N/A:**<Reason for failure>
4656	**Windows Logs->Security** Subcategory: Handle Manipulation	A handle to an object was requested.	**System->TimeCreated[SystemTime]**: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->SubjectUserSid**: <Subject identifier>
4670	**Windows Logs->Security** Subcategory: Policy Change	Permissions on an object were changed.	**System->TimeCreated[SystemTime]**: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->SubjectUserSid**: <Subject identifier>
4673	**Windows Logs->Security** Subcategory: Sensitive Privilege Use	A privileged service was called.	**System->TimeCreated[SystemTime]**: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->SubjectUserSid**: <Subject identifier>
4674	**Windows Logs->Security** Subcategory: Sensitive Privilege Use	An operation was attempted on a privileged object.	**System->TimeCreated[SystemTime]**: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->SubjectUserSid**: <Subject identifier>
4720	**Windows Logs->Security** Subcategory: User Account Management	A user account was created.	**System->TimeCreated[SystemTime]**: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->SubjectUserSid**: <Subject identifier>

Event ID	Log Location	Message	Fields
4725	**Windows Logs->Security** Subcategory: User Account Management	A user account was disabled.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->SubjectUserSid**: <Subject identifier>
4726	**Windows Logs->Security** Subcategory: User Account Management	A user account was deleted.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->SubjectUserSid**: <Subject identifier>
4719	**Windows Logs -> Security** Subcategory: Audit Policy Change	System audit policy was changed	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData ->SubjectUserSid**: <Subject identifier >
4731	**Windows Logs->Security** Subcategory: User Account Management	A security-enabled local group was created.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->SubjectUserSid**: <Subject identifier>
4732	**Windows Logs->Security** Subcategory: User Account Management	A member was added to a security-enabled group.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->SubjectUserSid**: <Subject identifier>
4733	**Windows Logs->Security** Subcategory: User Account Management	A member was removed from a security-enabled group.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->SubjectUserSid**: <Subject identifier>
4734	**Windows Logs->Security** Subcategory: User Account Management	A security-enabled local group was deleted.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->SubjectUserSid**: <Subject identifier>

Event ID	Log Location	Message	Fields
4735	**Windows Logs->Security** Subcategory: User Account Management	A security-enabled local group was changed.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->SubjectUserSid**: <Subject identifier>
4738	**Windows Logs->Security** Subcategory: User Account Management	A user account was changed	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->SubjectUserSid**: <Subject identifier>
4740	**Windows Logs->Security** Subcategory: Account Lockout	A user account was locked out.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->SubjectUserSid**: <Subject identifier>
4767	**Windows Logs->Security** Subcategory: Account Lockout	A user account was unlocked.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->SubjectUserSid**: <Subject identifier>
4950	**Windows Logs->Security** Subcategory: MPSSVC Rule-Level Policy Change	A Windows Firewall setting has changed.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **System->Computer**: <Subject identifier>
5040	**Windows Logs->Security** Subcategory: Filtering Platform Policy Change	A change was made to IPsec settings. An authentication set was added.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **System->Computer**: <Subject identifier>
5043	**Windows Logs->Security** Subcategory: Filtering Platform Policy Change	A change was made to IPsec settings. A connection security rule was added.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **System->Computer**: <Subject identifier>

Event ID	Log Location	Message	Fields
5046	**Windows Logs->Security** Subcategory: Filtering Platform Policy Change	A change was made to IPsec settings. A crypto set was added.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **System->Computer**: <Subject identifier>
5152	**Windows Logs->Security** Filtering Platform Packet Blocked	The Windows Filtering Platform has blocked a packet.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->ProcessId, Application**: <Subject identifier> **EventData->SourceAddress,SourcePort:** <Presumed identity of source subject><Non-TOE endpoint of connection> **EventData->DestAddress,DestPort:** <Identity of destination subject> **EventData->Protocol:** <Transport layer protocol> **EventData->FilterRTID,LayerName,LayerRTID: <**The entry in the SPD that applied to the decision><Reason for failure>
5156	**Windows Logs->Security** Filtering Platform Connection	The Windows Filtering Platform has permitted a connection	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->ProcessId, Application**: <Subject identifier> **EventData->SourceAddress,SourcePort:** <Presumed identity of source subject><Non-TOE endpoint of connection> **EventData->DestAddress,DestPort:** <Identity of destination subject> **EventData->Protocol:** <Transport layer protocol> **EventData->FilterRTID,LayerName,LayerRTID: <**The entry in the SPD that applied to the decision><Reason for failure>

Event ID	Log Location	Message	Fields
5157	**Windows Logs->Security** Filtering Platform Connection Blocked	The Windows Filtering Platform has blocked a connection.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->ProcessId, Application**: <Subject identifier> **EventData->SourceAddress,SourcePort:** <Presumed identity of source subject><Non-TOE endpoint of connection> **EventData->DestAddress,DestPort:** <Identity of destination subject> **EventData->Protocol:** <Transport layer protocol> **EventData->FilterRTID,LayerName,LayerRTID: <**The entry in the SPD that applied to the decision><Reason for failure>
5451	**Windows Logs->Security** Subcategory: IPsec Quick Mode	IPsec quick mode security association was established	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **System->Computer**: <Subject identifier> **EventData->MainModeSaId**<Presumed identity of source subject> **EventData->RemoteAddress, RemotePort:** <Non-TOE endpoint of connection> **EventData-> MainModeSaId,LocalAddress,LocalPort:** <Identity of destination subject> **EventData->IpProtocol:** <Transport layer protocol> **EventData->TunnelFilerId,TunnelId: <**The entry in the SPD that applied to the decision> **N/A:**<Reason for failure>

Event ID	Log Location	Message	Fields
5452	**Windows Logs->Security** Subcategory: IPsec Quick Mode	IPsec quick mode security association ended	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **System->Computer**: <Subject identifier> **EventData->QuickModeSaId**<Presumed identity of source subject> **EventData->RemoteAddress, RemotePort:** <Non-TOE endpoint of connection> **EventData-> QuickModeSaId,LocalAddress,LocalPort:** <Identity of destination subject> **EventData->IpProtocol:** <Transport layer protocol> **EventData->TunnelId,TrafficSelectorId: <**The entry in the SPD that applied to the decision> **N/A:**<Reason for failure>
6416	**Windows Logs->Security** Task Category: Plug and Play Events	A new external device was recognized by the system. Subject: <Windows device details> Device ID: <Remote Bluetooth Device ID> Device Name: <Remote Bluetooth device name>	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->Computer**: <Computer Name> **DeviceId:** <Remote Bluetooth Device ID> **DeviceDescription:** <Remote Bluetooth device name>
6420	**Windows Logs->Security** Task Category: Plug and Play Events	A device was disabled. <Device and scenario details>	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->Computer**: <Computer Name>
6422	**Windows Logs->Security** Task Category: Plug and Play Events	A device was enabled. <Device and scenario details>	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Task**: <Type of event> **System->Keywords**: <Outcome as Success or Failure> **EventData->Computer**: <Computer Name>

Event ID	Log Location	Message	Fields
8001	**Microsoft-Windows-WLAN-AutoConfig/Operational**	WLAN AutoConfig service has successfully connected to a wireless network	**System->TimeCreated[SystemTime]**: <Date and time of event> **System->Provider[Name]**: <Type of event> **System->Level**: <Outcome as Success or Failure> **System->Security[UserID]**: <Subject identity> **EventData->PHYType, AuthenticationAlgorithm**: <Trusted channel protocol> **EventData->SSID**: <Non-TOE endpoint of connection>
8002	**Microsoft-Windows-WLAN-AutoConfig/Operational**	WLAN AutoConfig service failed to connect to a wireless network	**System->TimeCreated[SystemTime]**: <Date and time of event> **System->Provider[Name]**: <Type of event> **System->Level**: <Outcome as Success or Failure> **System->Security[UserID]**: <Subject identity> **EventData->PHYType, AuthenticationAlgorithm**: <Trusted channel protocol> **EventData->SSID**: <Non-TOE endpoint of connection>
8003	**Microsoft-Windows-WLAN-AutoConfig/Operational**	WLAN AutoConfig service has successfully disconnected from a wireless network	**System->TimeCreated[SystemTime]**: <Date and time of event> **System->Provider[Name]**: <Type of event> **System->Level**: <Outcome as Success or Failure> **System->Security[UserID]**: <Subject identity> **EventData->ConnectionId**: <Trusted channel protocol> **EventData->SSID**: <Non-TOE endpoint of connection>
8004	Microsoft-Windows-WLAN-AutoConfig/Operational	Wireless security stopped.	System->TimeCreated[SystemTime]: <Date and time of event> System->Provider[Name]: <Type of event> System->Level: <Outcome as Success or Failure> System->Security[UserID]: <Subject identity> EventData->SSID: <Non-TOE endpoint of connection>
8020	**Application and Services Logs->Microsoft->Windows->AppLocker->Packaged app-Execution**	<Packaged app name> was allowed to run.	**System->TimeCreated[SystemTime]**: <Date and time of event> **System->Provider[Name]**: <Type of event> **System->Level**: <Outcome as Success or Failure> **System->Security[UserID]**: <Subject identifier>

Event ID	Log Location	Message	Fields
8022	**Application and Services Logs->Microsoft->Windows->AppLocker->Packaged app-Execution**	<Packaged app name> was prevented from running.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Provider[Name]**: <Type of event> **System->Level**: <Outcome as Success or Failure> **System->Security[UserID]**: <Subject identifier>
10001	**Microsoft-Windows-VPN-Client/Operational**	VPN Profile <Name> has been created with the following properties: <Detailed Connection Properties>	**TimeCreated**[**SystemTime**]: <Date and time of event> **Provider[Name]**: <Provider name> **OpcodeDisplayName:** <Success or failure> **Properties:** <Properties>
10002	**Microsoft-Windows-VPN-Client/Operational**	VPN Profile <Name>} could not be created. <Error ID> <Detailed Connection Properties>	**TimeCreated**[**SystemTime**]: <Date and time of event> **Provider[Name]**: <Provider name> **OpcodeDisplayName:** <Success or failure> **Properties:** <Properties>
11005	**Microsoft-Windows-WLAN-AutoConfig/Operational**	Wireless security succeeded.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Provider[Name]**: <Type of event> **System->Level**: <Outcome as Success or Failure> **System->Security[UserID]**: <Subject identity> **EventData->SSID**: <Non-TOE endpoint of connection>
11006	**Microsoft-Windows-WLAN-AutoConfig/Operational**	Wireless security failed.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Provider[Name]**: <Type of event> **System->Level**: <Outcome as Success or Failure> **System->Security[UserID]**: <Subject identity> **EventData->SSID**: <Non-TOE endpoint of connection> **EventData->ReasonText**: <Failure condition> **EventData->ReasonCode**: <Failure condition error code>

Event ID	Log Location	Message	Fields
12013	**Microsoft-Windows-WLAN-AutoConfig/Operational**	Wireless 802.1x authentication failed.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Provider[Name]**: <Type of event> **System->Level**: <Outcome as Success or Failure> **System->Security[UserID]**: <Subject identity> **EventData->SSID**: <Non-TOE endpoint of connection>
36871	Windows Logs -> System	A fatal error occurred while creating an SSL (client or server) credential	System->TimeCreated[SystemTime]: <Date and time of event> System->Provider[Name]: <Type of event> System->Level: <Outcome as Success or Failure> System->Task: <Type of event> System->Opcode: <Success or failure> System->Keywords: <Outcome as Success or Failure> System->Computer: <Subject identifier> System->Security[UserID]: <Subject identity> EventData->Type: <Non-TOE endpoint of connection> EventData->ErrorState: <Non-TOE endpoint of connection>
36880	**Windows Logs -> System**	An TLS server handshake completed successfully. The negotiated cryptographic parameters are as follows:	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Provider[Name]**: <type of event> **System->Security[UserID]**: <subject identifier > **UserData->EventXML->TargetName**: <Non-TOE endpoint>

Event ID	Log Location	Message	Fields
36888	**Windows Logs -> System**	A fatal alert was generated and sent to the remote endpoint. This may result in termination of the connection. The TLS protocol defined fatal error code is %1.	**System->TimeCreated**[**SystemTime**]: <Date and time of event> **System->Provider[Name]**: <type of event> **System->Security[UserID]**: <subject identifier > **UserData->EventXML->TargetName**: <Non-TOE endpoint > **UserData->EventXML->AlertDesc:** < Reason for failure> **UserData->EventXML->ErrorState:** < Reason for failure > The following are the possible error codes: 10 Unexpected message 20 Bad record MAC 22 Record overflow 30 Decompression fail 40 Handshake failure 47 Illegal parameter 48 Unknown CA 49 Access denied 50 Decode error 51 Decrypt error 70 Protocol version 71 Insufficient security 80 Internal error 110 Unsupported extension

6 Configuration Annex

This section provides guidance for IT administrators who manage systems that must meet the requirements of the Configuration Annex to the Protection Profile (PP) for General Purpose Operating Systems (GPOS). The guidance is aligned with the following version of the Configuration Annex:

- Configuration Annex Release 1 for PP GPOS version 4.2.1, https://www.niap-ccevs.org/MMO/PP/PP_OS-v4.2.1_configannex.pdf

6.1 Supported Configuration Actions

The following table lists the Configuration Actions supported by Windows. For each action, the table lists the methods available in Windows to configure it and a reference to the section of this Operational and Administrative Guide with implementation instructions.

Configuration Action	NIST Control Reference	GP OS SFR	Specific Value for DoD or CNSSI 1253 (If Published)	Windows Management Method(s)	Admin Guide Reference
Configure Minimum Password Length to 12 Characters	IA-5 (1)(a)	FMT_MOF_EXT.1	12 characters	MDM Group Policy Command Line	4.8 Managing
Require at Least 1 Special Character in Password	IA-5 (1)(a)	FMT_MOF_EXT.1	At least one	MDM Group Policy	4.8 Managing
Require at Least 1 Numeric Character in Password	IA-5 (1)(a)	FMT_MOF_EXT.1	At least one	MDM Group Policy	4.8 Managing
Require at Least 1 Uppercase Character in Password	IA-5 (1)(a)	FMT_MOF_EXT.1	At least one	MDM Group Policy	4.8 Managing
Require at Least 1 Lowercase Character in Password	IA-5 (1)(a)	FMT_MOF_EXT.1	At least one	MDM Group Policy	4.8 Managing

Configuration Action	NIST Control Reference	GP OS SFR	Specific Value for DoD or CNSSI 1253 (If Published)	Windows Management Method(s)	Admin Guide Reference
Enable Screen Lock	AC-11a	FMT_MOF_EXT.1		MDM Group Policy	4.9 Managing screen lock, session timeout
Set Screen Lock Timeout Period to 30 Minutes or Less	AC-11a	FMT_MOF_EXT.1	30 minutes	MDM Group Policy	4.9 Managing screen lock, session timeout
Disable Unauthenticated Login (such as Guest Accounts)	AC-11a	FIA_AFL.1		MDM Group Policy	4.8 Managing
Set Maximum Number of Authentication Failures to 3 Within 15 Minutes	AC-7a	FMT_MOF_EXT.1	3 failures 15 minutes	MDM Group Policy Command Line	4.8 Managing
Enable Host-Based Firewall	SC-7(12)	FMT_MOF_EXT.1		PowerShell	4.14 Managing the firewall
Configure Name/Address of Remote Management Server from Which to Receive Config Settings	CM-3(3)	FMT_MOF_EXT.1		MDM Group Policy	**Error! Reference source not found. Error! Reference source not found.** **Error! Reference source not found. Error! Reference source not found.**
Configure the System to Offload Audit Records to a Log Server	AU-4(1)	FAU_GEN.1.1.c		*Not in scope for this GPOS evaluation per the Security Target.*	
Set Logon Warning Banner	AC-8a	FMT_MOF_EXT.1	See text below	MDM Group Policy Registry	4.10 Managing the logon banner

Configuration Action	**NIST Control Reference**	**GP OS SFR**	**Specific Value for DoD or CNSSI 1253 (If Published)**	**Windows Management Method(s)**	**Admin Guide Reference**
Audit All Logons (Success/Failure) and Logoffs (Successful)	AU-2a	FAU_GEN.1.1.c	Authentication events: (1) Logons (Success/Failure) (2) Logoffs (Success)	Group Policy Command Line	4.20 Managing audit policy
Audit File and Object Events (Unsuccessful)	AU-2a	FAU_GEN.1.1.c	File and object events: (1) Create (Success/Failure) (2) Access (Success/Failure) (3) Delete (Success/Failure) (4) Modify (Success/Failure) (5) Permission Modification (Success/Failure) (6) Ownership Modification (Success/Failure)	Group Policy Command Line	4.20 Managing audit policy
Audit User and Group Management Events (Success/Failure)	AU-2a	FAU_GEN.1.1.c	User and group management events: (1) User add, delete, modify, disable, enable (Success/Failure) (2) Group/Role add, delete, modify (Success/Failure)	Group Policy Command Line	4.20 Managing audit policy

Configuration Action	NIST Control Reference	GP OS SFR	Specific Value for DoD or CNSSI 1253 (If Published)	Windows Management Method(s)	Admin Guide Reference
Audit Privilege or Role Escalation Events (Success/Failure)	AU-2a	FAU_GEN.1.1.c	Privilege/Role escalation (Success/Failure)	Group Policy Command Line	4.20 Managing audit policy
Audit All Audit and Log Data Accesses (Success/Failure)	AU-2a	FAU_GEN.1.1.c	Audit and log data access (Success/Failure)	Group Policy Command Line	4.20 Managing audit policy
Audit Cryptographic Verification of Software (Success/Failure)	AU-2a	FAU_GEN.1.1.c		Group Policy Command Line	4.20 Managing audit policy
Audit Program Initiations (Success/Failure)	AU-2a	FAU_GEN.1.1.c	Application (e.g., Firefox, Internet Explorer, MS Office Suite, etc.) initialization (Success/Failure)	Group Policy Command Line	4.20 Managing audit policy
Audit System Reboot, Restart, and Shutdown Events (Success/Failure)	AU-2a	FAU_GEN.1.1.c	System reboot, restart and shutdown (Success/Failure)	Group Policy Command Line	4.20 Managing audit policy
Audit Kernel Module Loading and Unloading Events (Success/Failure)	AU-2a	FAU_GEN.1.1.c		Group Policy Command Line	4.20 Managing audit policy
Enable Automatic Software Update	SI-2	FMT_MOF_EXT.1		MDM Group Policy Command Line Windows GUI	4.12 Managing updates

6.1.1 Logon banner text

The following text is the value specified for the logon banner, copied here from the published Configuration Annex for convenience.

6.1.1.1 For DoD Systems:

```
You are accessing a U.S. Government (USG) Information System (IS) that is provided for USG-authorized use only. By using this
IS (which includes any device attached to this IS), you consent to the following conditions:
-The USG routinely intercepts and monitors communications on this IS for purposes including, but not limited to,
penetration testing, COMSEC monitoring, network operations and defense, personnel misconduct (PM), law enforcement (LE), and
counterintelligence (CI) investigations.
-At any time, the USG may inspect and seize data stored on this IS.
-Communications using, or data stored on, this IS are not private, are subject to routine monitoring, interception, and
search, and may be disclosed or used for any USG-authorized purpose.
-This IS includes security measures (e.g., authentication and access controls) to protect USG interests--not for your personal
benefit or privacy.
-Notwithstanding the above, using this IS does not constitute consent to PM, LE or CI investigative searching or
monitoring of the content of privileged communications, or work product, related to personal representation or services by
attorneys, psychotherapists, or clergy, and their assistants. Such communications and work product are private and
confidential. See User Agreement for details.
```

6.1.1.2 For non-DoD NSS:

```
Organization-defined system use notification message or banner.
```