Contents

Get started with Windows Server

12/17/2021 • 2 minutes to read • Edit Online

Windows Server is the platform for building an infrastructure of connected applications, networks, and web services, from the workgroup to the data center. It bridges on-premises environments with Azure, adding additional layers of security while helping you modernize your applications and infrastructure.

This collection of articles contains detailed information to help you understand and get the most from Windows Server, and help determine if you're ready to move to the latest version. Once you've checked the system requirements, upgrade options, and other information about Windows Server, you're ready to start down the path of installing the best edition and installation option for your needs.

> **TIP**
> To download Windows Server, see Windows Server evaluations in the Evaluation Center.

> **NOTE**
> If you're looking for information about earlier versions that are no longer supported, see the Windows previous versions documentation.

Support and feedback

For the latest news on Windows Server, visit the Windows Server blog to stay up to date on announcements, features, events, and other information from the Windows Server engineering teams. You can also visit the Windows Server Community to share best practices, get latest news, and learn from experts about Windows Server.

Learn

Browse several learning paths for Windows Server at Microsoft Learn to help learn new skills and accelerate your deployment with step-by-step guidance. You can learn how to deploy, configure and administer Windows Server, as well as network infrastructure, file servers and storage management, Hyper-V and virtualization, plus much more.

Windows Insider Program

The Windows Insider Program for Windows Server provides preview builds of Windows Server allowing you early access to learn, test, and help shape the future of Windows Server. To learn more, you can get started with the Windows Insider Program for Windows Server and participate in the Windows Server Insiders Community.

Next steps

To get started, find out more from these resources.

- What's new in Windows Server 2022 provides an overview of the latest features in Windows Server.
- Learn about the different servicing channels, which each is used for, and what it means for your workloads and support.
- Compare the differences in the editions in Windows Server 2022.

- Choose the right installation option based on whether you want the Desktop Experience or a minimal Core interface.
- Understand the hardware requirements to run Windows Server.
- Follow the Microsoft Learn learning path on Windows Server deployment, configuration, and administration.
- If you still need to use Windows Server 2008, Windows Server 2008 R2 (and in future Windows Server 2012, or Windows Server 2012 R2) Extended Security Updates are available to help keep you safe with security updates and bulletins rated critical and important.

What's new in Windows Server 2022

12/17/2021 • 11 minutes to read • Edit Online

Applies to: Windows Server 2022

This article describes some of the new features in Windows Server 2022. Windows Server 2022 is built on the strong foundation of Windows Server 2019 and brings many innovations on three key themes: security, Azure hybrid integration and management, and application platform. Also, Windows Server 2022 Datacenter: Azure Edition helps you use the benefits of cloud to keep your VMs up to date while minimizing downtime.

Security

The new security capabilities in Windows Server 2022 combine other security capabilities in Windows Server across multiple areas to provide defense-in-depth protection against advanced threats. Advanced multi-layer security in Windows Server 2022 provides the comprehensive protection that servers need today.

Secured-core server

Certified Secured-core server hardware from an OEM partner provides additional security protections that are useful against sophisticated attacks. This can provide increased assurance when handling mission critical data in some of the most data sensitive industries. A Secured-core server uses hardware, firmware, and driver capabilities to enable advanced Windows Server security features. Many of these features are available in Windows Secured-core PCs and are now also available with Secured-core server hardware and Windows Server 2022.

Hardware root-of-trust

Trusted Platform Module 2.0 (TPM 2.0) secure crypto-processor chips provide a secure, hardware-based store for sensitive cryptographic keys and data, including systems integrity measurements. TPM 2.0 can verify that the server has been started with legitimate code and can be trusted by subsequent code execution. This is known as a hardware root-of-trust and is used by features such as BitLocker drive encryption.

Firmware protection

Firmware executes with high privileges and is often invisible to traditional anti-virus solutions, which has lead to a rise in the number of firmware-based attacks. Secured-core server processors support measurement and verification of boot processes with Dynamic Root of Trust for Measurement (DRTM) technology and isolation of driver access to memory with Direct Memory Access (DMA) protection.

Virtualization-based security (VBS)

Secured-core servers support virtualization-based security (VBS) and hypervisor-based code integrity (HVCI). VBS uses hardware virtualization features to create and isolate a secure region of memory from the normal operating system, protecting against an entire class of vulnerabilities used in cryptocurrency mining attacks. VBS also allows for the use of Credential Guard, where user credentials and secrets are stored in a virtual container that the operating system cannot access directly.

HVCI uses VBS to significantly strengthen code integrity policy enforcement, including kernel mode integrity that checks all kernel mode drivers and binaries in a virtualized environment before they are started, preventing unsigned drivers or system files from being loaded into system memory.

Secure connectivity

Transport: HTTPS and TLS 1.3 enabled by default on Windows Server 2022

Secure connections are at the heart of today's interconnected systems. Transport Layer Security (TLS) 1.3 is the latest version of the internet's most deployed security protocol, which encrypts data to provide a secure

communication channel between two endpoints. HTTPS and TLS 1.3 is now enabled by default on Windows Server 2022, protecting the data of clients connecting to the server. It eliminates obsolete cryptographic algorithms, enhances security over older versions, and aims to encrypt as much of the handshake as possible. Learn more about supported TLS versions and about supported cipher suites.

Although TLS 1.3 in the protocol layer is now enabled by default, applications and services also need to actively support it. Please see documentation for those applications and services for more information. The Microsoft Security blog has more detail in the post Taking Transport Layer Security (TLS) to the next level with TLS 1.3.

Secure DNS: Encrypted DNS name resolution requests with DNS-over-HTTPS

DNS Client in Windows Server 2022 now supports DNS-over-HTTPS (DoH) which encrypts DNS queries using the HTTPS protocol. This helps keep your traffic as private as possible by preventing eavesdropping and your DNS data being manipulated. Learn more about configuring the DNS client to use DoH.

Server Message Block (SMB): SMB AES-256 encryption for the most security conscious

Windows Server now supports AES-256-GCM and AES-256-CCM cryptographic suites for SMB encryption. Windows will automatically negotiate this more advanced cipher method when connecting to another computer that also supports it, and it can also be mandated through Group Policy. Windows Server still supports AES-128 for down-level compatibility. AES-128-GMAC signing now also accelerates signing performance.

SMB: East-West SMB encryption controls for internal cluster communications

Windows Server failover clusters now support granular control of encrypting and signing intra-node storage communications for Cluster Shared Volumes (CSV) and the storage bus layer (SBL). This means that when using Storage Spaces Direct, you can decide to encrypt or sign east-west communications within the cluster itself for higher security.

SMB Direct and RDMA encryption

SMB Direct and RDMA supply high bandwidth, low latency networking fabric for workloads like Storage Spaces Direct, Storage Replica, Hyper-V, Scale-out File Server, and SQL Server. SMB Direct in Windows Server 2022 now supports encryption. Previously, enabling SMB encryption disabled direct data placement; this was intentional, but seriously impacted performance. Now data is encrypted data before placement, leading to far less performance degradation while adding AES-128 and AES-256 protected packet privacy.

More information on SMB encryption, signing acceleration, secure RDMA, and cluster support can be found at SMB security enhancements.

SMB over QUIC

SMB over QUIC updates the SMB 3.1.1 protocol in Windows Server 2022 Datacenter: Azure Edition and supported Windows clients to use the QUIC protocol instead of TCP. By using SMB over QUIC along with TLS 1.3, users and applications can securely and reliably access data from edge file servers running in Azure. Mobile and telecommuter users no longer need a VPN to access their file servers over SMB when on Windows. More information can be found at the SMB over QUIC documentation.

Azure hybrid capabilities

You can increase your efficiency and agility with built-in hybrid capabilities in Windows Server 2022 that allow you to extend your data centers to Azure more easily than ever before.

Azure Arc enabled Windows Servers

Azure Arc enabled servers with Windows Server 2022 brings on-premises and multi-cloud Windows Servers to Azure with Azure Arc. This management experience is designed to be consistent with how you manage native Azure virtual machines. When a hybrid machine is connected to Azure, it becomes a connected machine and is treated as a resource in Azure. More information can be found at the Azure Arc enables servers documentation.

Windows Admin Center

Improvements to Windows Admin Center to manage Windows Server 2022 include capabilities to both report

on the current state of the Secured-core features mentioned above, and where applicable, allow customers to enable the features. More information on these and many more improvements to Windows Admin Center can be found at the Windows Admin Center documentation.

Azure Automanage - Hotpatch

Hotpatch, part of Azure Automanage, is supported in Windows Server 2022 Datacenter: Azure Edition. Hotpatching is a new way to install updates on new Windows Server Azure Edition virtual machines (VMs) that doesn't require a reboot after installation. More information can be found at the Azure Automanage documentation.

Application platform

There are several platform improvements for Windows Containers, including application compatibility and the Windows Container experience with Kubernetes. A major improvement includes reducing the Windows Container image size by up to 40%, which leads to a 30% faster startup time and better performance.

You can now also run applications that depend on Azure Active Directory with group Managed Services Accounts (gMSA) without domain joining the container host, and Windows Containers now support Microsoft Distributed Transaction Control (MSDTC) and Microsoft Message Queuing (MSMQ).

There are several other enhancements that simplify the Windows Container experience with Kubernetes. These enhancements include support for host-process containers for node configuration, IPv6, and consistent network policy implementation with Calico.

In addition to platform improvements, Windows Admin Center has been updated to make it easy to containerize .NET applications. Once the application is in a container, you can host it on Azure Container Registry to then deploy it to other Azure services, including Azure Kubernetes Service.

With support for Intel Ice Lake processors, Windows Server 2022 supports business-critical and large-scale applications, such as SQL Server, that require up to 48 TB of memory and 2,048 logical cores running on 64 physical sockets. Confidential computing with Intel Secured Guard Extension (SGX) on Intel Ice Lake improves application security by isolating applications from each other with protected memory.

Other key features

Nested virtualization for AMD processors

Nested virtualization is a feature that allows you to run Hyper-V inside of a Hyper-V virtual machine (VM). Windows Server 2022 brings support for nested virtualization using AMD processors, giving more choices of hardware for your environments. More information can be found at the nested virtualization documentation.

Microsoft Edge browser

Microsoft Edge is included with Windows Server 2022, replacing Internet Explorer. It is built on Chromium open source and backed by Microsoft security and innovation. It can be used with the Server with Desktop Experience installation options. More information can be found at the Microsoft Edge Enterprise documentation. Note that Microsoft Edge, unlike the rest of Windows Server, follows the Modern Lifecycle for its support lifecycle. For details, see Microsoft Edge lifecycle documentation.

Networking performance

UDP performance improvements

UDP is becoming a very popular protocol carrying more and more network traffic due to the increasing popularity of RTP and custom (UDP) streaming and gaming protocols. The QUIC protocol, built on top of UDP, brings the performance of UDP to a level on par with TCP. Significantly, Windows Server 2022 includes UDP Segmentation Offload (USO). USO moves most of the work required to send UDP packets from the CPU to the network adapter's specialized hardware. Complimenting USO is UDP Receive Side Coalescing (UDP RSC), which coalesces packets and reduces CPU usage for UDP processing. In addition, we have also made hundreds of

improvements to the UDP data path both transmit and receive. Windows Server 2022 and Windows 11 both have this new capability.

TCP performance improvements

Windows Server 2022 uses TCP HyStart++ to reduce packet loss during connection start-up (especially in high-speed networks) and RACK to reduce Retransmit TimeOuts (RTO). These features are enabled in the transport stack by default and provide a smoother network data flow with better performance at high speeds. Windows Server 2022 and Windows 11 both have this new capability.

Hyper-V virtual switch improvements

Virtual switches in Hyper-V have been enhanced with updated Receive Segment Coalescing (RSC). This allows the hypervisor network to coalesce packets and process as one larger segment. CPU cycles are reduced and segments will remain coalesced across the entire data path until processed by the intended application. This means improved performance in both network traffic from an external host, received by a virtual NIC, as well as from a virtual NIC to another virtual NIC on the same host.

Storage

Storage Migration Service

Enhancements to Storage Migration Service in Windows Server 2022 makes it easier to migrate storage to Windows Server or to Azure from more source locations. Here are the features that are available when running the Storage Migration Server orchestrator on Windows Server 2022:

- Migrate local users and groups to the new server.
- Migrate storage from failover clusters, migrate to failover clusters, and migrate between standalone servers and failover clusters.
- Migrate storage from a Linux server that uses Samba.
- More easily synchronize migrated shares into Azure by using Azure File Sync.
- Migrate to new networks such as Azure.
- Migrate NetApp CIFS servers from NetApp FAS arrays to Windows servers and clusters.

Adjustable storage repair speed

User adjustable storage repair speed is a new feature in Storage Spaces Direct that offers more control over the data resync process by allocating resources to either repair data copies (resiliency) or run active workloads (performance). This helps improve availability and allows you to service your clusters more flexibly and efficiently.

Faster repair and resynchronization

Storage repair and resynchronization after events such as node reboots and disk failures are now twice as fast. Repairs have less variance in time taken so you can be more sure of how long the repairs will take, which has been achieved through adding more granularity to data tracking. This only moves the data that needs to be moved, and reduces the system resources used and the time taken.

Storage bus cache with Storage Spaces on standalone servers

Storage bus cache is now available for standalone servers. It can significantly improve read and write performance, while maintaining storage efficiency and keeping the operational costs low. Similar to its implementation for Storage Spaces Direct, this feature binds together faster media (for example, NVMe or SSD) with slower media (for example, HDD) to create tiers. A portion of the faster media tier is reserved for the cache. To learn more, see Enable storage bus cache with Storage Spaces on standalone servers.

ReFS file-level snapshots

Microsoft's Resilient File System (ReFS) now includes the ability to snapshot files using a quick metadata operation. Snapshots are different than ReFS block cloning in that clones are writable, whereas snapshots are read-only. This functionality is especially useful in virtual machine backup scenarios with VHD/VHDX files. ReFS snapshots are unique in that they take a constant time irrespective of file size. Support for snapshots is available in ReFSUtil or as an API.

SMB compression

Enhancement to SMB in Windows Server 2022 and Windows 11 allows a user or application to compress files as they transfer over the network. Users no longer have to manually zip files in order to transfer much faster on slower or more congested networks. For details, see SMB Compression.

What's new in Windows Server, version 2004 and 20H2

12/17/2021 • 2 minutes to read • Edit Online

To learn about the latest features in Windows, see What's New in Windows Server. This topic describes some of the new features in Windows Server, version 2004 and 20H2.

Windows Server, version 20H2 is the next Semi-Annual Channel release of Windows Server, version 2004. This version focuses on reliability, performance, and other general improvements, but has no new features. Like other Semi-Annual Channel releases, it's supported for 18 months after its release. To learn more about the support dates for Semi-Annual Channel releases, see Windows Server release information.

Server Core container improvements

We've reduced the overall size of Server Core container images for improved download speeds and performance. We've included the following improvements in Windows Server, version 2004:

- Removed most NGEN images from Server Core container image to make the image size smaller.
- .NET Framework runtime images built on Server Core container images are now optimized for ASP.NET apps and Windows PowerShell script performance.
- The .NET team has also ensured there's only one copy of each NGEN image, resulting a smaller size for .NET Framework images.

To give you a better idea of the size of these containers, the following table compares the current version of the container as of the May 2020 monthly security update (also known as the **5B** update) with previous versions.

CONTAINER VERSION	DOWNLOAD SIZE	SIZE ON DISK
Windows Server, version 1903	2.311 GB	5.1 GB
Windows Server, version 1909	2.257 GB	4.97 GB
Windows Server, version 2004	1.830 GB	3.98 GB

For more information about Windows Server, version 2004 see our blog post, or To learn more about Windows container updates in general, see Update Windows Server containers.

What's new in Windows Server, version 1903 and version 1909

12/17/2021 • 6 minutes to read • Edit Online

IMPORTANT

Windows Server, version 1903 has reached end of servicing as of December 8, 2020. Windows Server, version 1909 has reached end of servicing as of May 11, 2021.

This article describes some of the new features covering both Windows Server, version 1903, and Windows Server, version 1909, which are Semi-Annual Channel releases. These features include enhancements for running and managing containers, tools for working in Server Core installations, and the ability to migrate storage from Linux devices.

Windows Server, version 1909 is focused on reliability, performance, and other general improvements, but has no new features. Like other Semi-Annual Channel releases, it's supported for 18 months from its first availability. For more information on the support dates of Semi-Annual Channel releases, see Windows Server release information.

The system requirements for this release are the same as for Windows Server 2019, for more information see hardware requirements . To see what's been removed recently, see Features Removed or Planned for Replacement starting with Windows Server, version 1903 and version 1909

NOTE

Windows containers must use the same version of Windows as the host server, or an **earlier** version. For example, a host server running the released version of Windows Server, version 1903 (build 18342) can run Windows Server containers with the same or earlier version and build number (even if the container uses an Insider Preview version of Windows). For more information, see Windows container version compatibility.

Enhanced support for non-Microsoft container services

We enhanced platform capabilities to support Azure container services and non-Microsoft container services.

- We integrated CRI-containerd with Host Compute Service (HCS) to support pods of Windows containers and Linux containers on Windows (LCOW) on Azure.
- We worked with the Kubernetes community to enable Windows container support. With the release of Kubernetes v1.14, Windows Server node support officially graduated from beta to stable. For more information, see Windows containers now supported in Kubernetes.
- Tigera Calico for Windows is now generally available as part of Tigera Essentials subscription and offers both non-overlay networking and network policy interoperable across mixed Linux/Windows environments.
- We delivered scalability improvements enhancing overlay networking support for Windows containers, including integration with Kubernetes through the latest release of Flannel and Kubernetes v1.14. For more information, see Intro to Windows support in Kubernetes.

DirectX hardware acceleration in containers

We're enabling support for hardware acceleration of DirectX APIs in Windows containers to support scenarios

such as Machine Learning (ML) inference using local graphical processing unit (GPU) hardware. For more information, see the Bringing GPU acceleration to Windows containers blog post.

Updated container identity and group managed service account documentation

We added more examples and compatibility information to the Group Managed Service Accounts documentation, and made the Credential Spec PowerShell module available in the PowerShell Gallery. For more information, see the What's new for container identity blog post.

Add Task Scheduler and Hyper-V Manager to Server Core installations

As you might know, we recommend using the Server Core installation option when using Windows Server, Semi-Annual Channel in production. However, Server Core by default omits several useful management tools. You can add many of the most commonly used tools by installing the App Compatibility Feature on Demand feature package, but previously there were some key tools missing.

Based on customer feedback, we added two more tools to the App Compatibility Feature on Demand feature package in this version: Task Scheduler (taskschd.msc) and Hyper-V Manager (virtmgmt.msc). For more information, see Server Core App Compatibility Feature on Demand (FOD).

Storage Migration Service now migrates local accounts, clusters, and Linux servers

Storage Migration Service makes it easier to migrate servers to a newer version of Windows Server. It provides a graphical tool that inventories data on servers and then transfers the data and configuration to newer servers, all without changes to applications or users having to change anything.

When using this version of Windows Server to orchestrate migrations, we've added the following abilities:

- Migrate local users and groups to the new server
- Migrate storage from failover clusters
- Migrate storage from a Linux server that uses Samba
- More easily sync migrated shares into Azure by using Azure File Sync
- Migrate to new networks, such as Azure

For more information about Storage Migration Service, see Storage Migration Service overview.

System Insights disk anomaly detection

System Insights is a predictive analytics feature that locally analyzes Windows Server system data and provides insight into the functioning of the server. It comes with many built-in capabilities, but we've added the ability to install additional capabilities via Windows Admin Center, starting with disk anomaly detection.

Disk anomaly detection is a new capability that highlights when disks are behaving *differently* than usual. While different isn't necessarily a bad thing, seeing these anomalous moments can be helpful when troubleshooting issues on your systems.

This capability is also available for servers running Windows Server 2019.

Windows Admin Center enhancements

A new release of Windows Admin Center is available, adding new functionality for Windows Server. For information on the latest features, see Windows Admin Center.

Security baseline for Windows 10 and Windows Server

The draft release of the security configuration baseline settings for Windows 10 version 1903, and for Windows Server version 1903 is available.

SetupDiag

SetupDiag version 1.4.1 is available.

SetupDiag is a command-line tool that can help diagnose why a Windows Update failed to install. SetupDiag works by searching Windows Setup log files. When searching log files, SetupDiag uses a set of rules to match known issues. In the current version of SetupDiag there are 53 rules contained in the rules.xml file, which is extracted when SetupDiag is run. The rules.xml file will be updated as new versions of SetupDiag are made available.

Windows Update rollback improvements

Servers can now automatically recover from startup failures by removing updates if the startup failure was introduced after the installation of recent driver or quality Windows Updates. When a device is unable to start up properly after the recent installation of quality of driver updates, Windows will now automatically uninstall the updates to get the device back up and running normally.

This functionality requires the server to be using the Server Core installation option option with a Windows Recovery Environment partition.

Microsoft Defender Antivirus improvements

Windows Server includes Microsoft Defender Antivirus. This release includes the following improvements:

- Attack surface area reduction – IT admins can configure devices with advanced web protection that enables them to define allow and deny lists for specific URL's and IP addresses.
- Next generation protection – Controls have been extended to protection from ransomware, credential misuse, and attacks that are transmitted through removable storage.
  - Integrity enforcement capabilities – Enable remote runtime attestation.
  - Tamper-proofing capabilities – Uses virtualization-based security to isolate critical ATP security capabilities away from the OS and attackers.
- Microsoft Defender ATP next-gen protection technologies include:
  - **Advanced machine learning**: Improved with advanced machine learning and AI models that enable it to protect against apex attackers using innovative vulnerability exploit techniques, tools, and malware.
  - **Emergency outbreak protection**: Provides emergency outbreak protection, which will automatically update devices with new intelligence when a new outbreak has been detected.
  - **Certified ISO 27001 compliance**: Ensures that the cloud service has analyzed for threats, vulnerabilities and impacts, and that risk management and security controls are in place.
  - **Geolocation support**: Support geolocation and sovereignty of sample data as well as configurable retention policies.

What's new in Windows Server 2019

12/17/2021 • 9 minutes to read • Edit Online

This article describes some of the new features in Windows Server 2019. Windows Server 2019 is built on the strong foundation of Windows Server 2016 and brings numerous innovations on four key themes: Hybrid Cloud, Security, Application Platform, and Hyper-Converged Infrastructure (HCI).

General

Windows Admin Center

Windows Admin Center is a locally deployed, browser-based app for managing servers, clusters, hyper-converged infrastructure, and Windows 10 PCs. It comes at no additional cost beyond Windows and is ready to use in production.

You can install Windows Admin Center on Windows Server 2019 as well as Windows 10 and earlier versions of Windows and Windows Server, and use it to manage servers and clusters running Windows Server 2008 R2 and later.

For more info, see Windows Admin Center.

Desktop experience

Because Windows Server 2019 is a Long-Term Servicing Channel (LTSC) release, it includes the **Desktop Experience**. (Semi-Annual Channel (SAC) releases don't include the Desktop Experience by design; they are strictly Server Core and Nano Server container image releases.) As with Windows Server 2016, during setup of the operating system you can choose between Server Core installations or Server with Desktop Experience installations.

System Insights

System Insights is a new feature available in Windows Server 2019 that brings local predictive analytics capabilities natively to Windows Server. These predictive capabilities, each backed by a machine-learning model, locally analyze Windows Server system data, such as performance counters and events, providing insight into the functioning of your servers and helping you reduce the operational expenses associated with reactively managing issues in your Windows Server deployments.

Hybrid Cloud

Server Core App Compatibility Feature on Demand

The Server Core App Compatibility Feature on Demand (FOD) significantly improves the app compatibility of the Windows Server Core installation option by including a subset of binaries and components from Windows Server with the Desktop Experience, without adding the Windows Server Desktop Experience graphical environment itself. This is done to increase the functionality and compatibility of Server Core while keeping it as lean as possible.

This optional feature on demand is available on a separate ISO and can be added to Windows Server Core installations and images only, using DISM.

Security

Windows Defender Advanced Threat Protection (ATP)

ATP's deep platform sensors and response actions expose memory and kernel level attacks and respond by

suppressing malicious files and terminating malicious processes.

- For more information about Windows Defender ATP, see Overview of Windows Defender ATP capabilities.
- For more information on onboarding servers, see Onboard servers to Windows Defender ATP service.

Windows Defender ATP Exploit Guard is a new set of host-intrusion prevention capabilities. The four components of Windows Defender Exploit Guard are designed to lock down the device against a wide variety of attack vectors and block behaviors commonly used in malware attacks, while enabling you to balance security risk and productivity requirements.

- Attack Surface Reduction(ASR) is set of controls that enterprises can enable to prevent malware from getting on the machine by blocking suspicious malicious files (for example, Office files), scripts, lateral movement, ransomware behavior, and email-based threats.
- Network protection protects the endpoint against web-based threats by blocking any outbound process on the device to untrusted hosts/IP addresses through Windows Defender SmartScreen.
- Controlled folder access protects sensitive data from ransomware by blocking untrusted processes from accessing your protected folders.
- Exploit protection is a set of mitigations for vulnerability exploits (replacing EMET)that can be easily configured to protect your system and applications.
- Windows Defender Application Control (also known as Code Integrity (CI) policy) was released in Windows Server 2016. Customer feedback has suggested that it is a great concept, but hard to deploy. To address this, we have built default CI policies, which allows all Windows in-box files and Microsoft applications, such as SQL Server, and block known executables that can bypass CI.

Security with Software Defined Networking (SDN)

Security with SDN delivers many features to increase customer confidence in running workloads, either on-premises, or as a service provider in the cloud.

These security enhancements are integrated into the comprehensive SDN platform introduced in Windows Server 2016.

For a complete list of what's new in SDN see, What's New in SDN for Windows Server 2019.

Shielded Virtual Machines improvements

- Branch office improvements

  You can now run shielded virtual machines on machines with intermittent connectivity to the Host Guardian Service by using the new fallback HGS and offline mode features. Fallback HGS allows you to configure a second set of URLs for Hyper-V to try if it can't reach your primary HGS server.

  Offline mode allows you to continue to start up your shielded VMs, even if HGS can't be reached, as long as the VM has started successfully once, and the host's security configuration has not changed.

- Troubleshooting improvements

  We've also made it easier to troubleshoot your shielded virtual machines by enabling support for VMConnect Enhanced Session Mode and PowerShell Direct. These tools are particularly useful if you've lost network connectivity to your VM and need to update its configuration to restore access.

  These features do not need to be configured, and they become available automatically when a shielded VM is placed on a Hyper-V host running Windows Server version 1803 or later.

- Linux support

  If you run mixed-OS environments, Windows Server 2019 now supports running Ubuntu, Red Hat

Enterprise Linux, and SUSE Linux Enterprise Server inside shielded virtual machines.

HTTP/2 for a faster and safer Web

- Improved coalescing of connections to deliver an uninterrupted and properly encrypted browsing experience.
- Upgraded HTTP/2's server-side cipher suite negotiation for automatic mitigation of connection failures and ease of deployment.
- Changed our default TCP congestion provider to Cubic to give you more throughput!

Storage

Here are some of the changes we've made to storage in Windows Server 2019. For details, see What's new in Storage.

Storage Migration Service

Storage Migration Service is a new technology that makes it easier to migrate servers to a newer version of Windows Server. It provides a graphical tool that inventories data on servers, transfers the data and configuration to newer servers, and then optionally moves the identities of the old servers to the new servers so that apps and users don't have to change anything. For more info, see Storage Migration Service.

Storage Spaces Direct

Here's a list of what's new in Storage Spaces Direct. For details, see What's new in Storage Spaces Direct. Also see Azure Stack HCI for info on acquiring validated Storage Spaces Direct systems.

- Deduplication and compression for ReFS volumes
- Native support for persistent memory
- Nested resiliency for two-node hyper-converged infrastructure at the edge
- Two-server clusters using a USB flash drive as a witness
- Windows Admin Center support
- Performance history
- Scale up to 4 PB per cluster
- Mirror-accelerated parity is 2X faster
- Drive latency outlier detection
- Manually delimit the allocation of volumes to increase fault tolerance

Storage Replica

Here's what's new in Storage Replica. For details, see What's new in Storage Replica.

- Storage Replica is now available in Windows Server 2019 Standard Edition.
- Test failover is a new feature that allows mounting of destination storage to validate replication or backup data. For more information, see Frequently Asked Questions about Storage Replica.
- Storage Replica log performance improvements
- Windows Admin Center support

Failover Clustering

Here's a list of what's new in Failover Clustering. For details, see What's new in Failover Clustering.

- Cluster sets
- Azure-aware clusters
- Cross-domain cluster migration
- USB witness

- Cluster infrastructure improvements
- Cluster Aware Updating supports Storage Spaces Direct
- File share witness enhancements
- Cluster hardening
- Failover Cluster no longer uses NTLM authentication

Application Platform

Linux containers on Windows

It is now possible to run Windows and Linux-based containers on the same container host, using the same docker daemon. This enables you to have a heterogeneous container host environment while providing flexibility to application developers.

Built-in support for Kubernetes

Windows Server 2019 continues the improvements to compute, networking, and storage from the Semi-Annual Channel releases needed to support Kubernetes on Windows. More details are available in upcoming Kubernetes releases.

- Container Networking in Windows Server 2019 greatly improves usability of Kubernetes on Windows by enhancing platform networking resiliency and support of container networking plugins.
- Deployed workloads on Kubernetes are able to use network security to protect both Linux and Windows services using embedded tooling.

Container improvements

- **Improved integrated identity**

  We've made integrated Windows authentication in containers easier and more reliable, addressing several limitations from prior versions of Windows Server.

- **Better application compatibility**

  Containerizing Windows-based applications just got easier: The app compatibility for the existing **windowsservercore** image has been increased. For applications with additional API dependencies, there is now a third base image: **windows**.

- **Reduced size and higher performance**

  The base container image download sizes, size on disk and startup times have been improved. This speeds up container workflows

- **Management experience using Windows Admin Center (preview)**

  We've made it easier than ever to see which containers are running on your computer and manage individual containers with a new extension for Windows Admin Center. Look for the "Containers" extension in the Windows Admin Center public feed.

Encrypted Networks

Encrypted Networks - Virtual network encryption allows encryption of virtual network traffic between virtual machines that communicate with each other within subnets marked as **Encryption Enabled**. It also utilizes Datagram Transport Layer Security (DTLS) on the virtual subnet to encrypt packets. DTLS protects against eavesdropping, tampering, and forgery by anyone with access to the physical network.

Network performance improvements for virtual workloads

Network performance improvements for virtual workloads maximizes the network throughput to virtual machines without requiring you to constantly tune or over-provision your host. This lowers the operations and

maintenance cost while increasing the available density of your hosts. These new features are:

- Dynamic Virtual Machine Multi-Queue (d.VMMQ)
- Receive Segment Coalescing in the vSwitch

Low Extra Delay Background Transport

Low Extra Delay Background Transport (LEDBAT) is a latency optimized, network congestion control provider designed to automatically yield bandwidth to users and applications, while consuming the entire bandwidth available when the network is not in use. This technology is intended for use in deploying large, critical updates across an IT environment without impacting customer facing services and associated bandwidth.

Windows Time Service

The Windows Time Service includes true UTC-compliant leap second support, a new time protocol called Precision Time Protocol, and end-to-end traceability.

High performance SDN gateways

High performance SDN gateways in Windows Server 2019 greatly improves the performance for IPsec and GRE connections, providing ultra-high-performance throughput with much less CPU utilization.

New Deployment UI and Windows Admin Center extension for SDN

Now, with Windows Server 2019, it's easy to deploy and manage through a new deployment UI and Windows Admin Center extension that enable anyone to harness the power of SDN.

Persistent Memory support for Hyper-V VMs

To leverage the high throughput and low latency of persistent memory (also known as storage class memory) in virtual machines, it can now be projected directly into VMs. This can help to drastically reduce database transaction latency or reduce recovery times for low latency in-memory databases on failure.

What's new in Windows Server 2016

12/17/2021 • 11 minutes to read • Edit Online

This article describes some of the new features in Windows Server 2016 that are the ones most likely to have the greatest impact as you work with this release.

Compute

The Virtualization area includes virtualization products and features for the IT professional to design, deploy, and maintain Windows Server.

General

Physical and virtual machines benefit from greater time accuracy due to improvements in the Win32 Time and Hyper-V Time Synchronization Services. Windows Server can now host services that are compliant with upcoming regulations that require a 1ms accuracy with regard to UTC.

Hyper-V

- What's new in Hyper-V on Windows Server 2016. This topic explains the new and changed functionality of the Hyper-V role in Windows Server 2016, Client Hyper-V running on Windows 10, and Microsoft Hyper-V Server 2016.
- Windows Containers: Windows Server 2016 container support adds performance improvements, simplified network management, and support for Windows containers on Windows 10. For some additional information on containers, see Containers: Docker, Windows and Trends.

Nano Server

What's New in Nano Server. Nano Server now has an updated module for building Nano Server images, including more separation of physical host and guest virtual machine functionality as well as support for different Windows Server editions.

There are also improvements to the Recovery Console, including separation of inbound and outbound firewall rules as well as the ability to repair the configuration of WinRM.

Shielded Virtual Machines

Windows Server 2016 provides a new Hyper-V-based Shielded Virtual Machine to protect any Generation 2 virtual machine from a compromised fabric. Among the features introduced in Windows Server 2016 are the following:

- A new **Encryption Supported** mode that offers more protections than for an ordinary virtual machine, but less than **Shielded** mode, while still supporting vTPM, disk encryption, Live Migration traffic encryption, and other features, including direct fabric administration conveniences such as virtual machine console connections and PowerShell Direct.
- Full support for converting existing non-shielded Generation 2 virtual machines to shielded virtual machines, including automated disk encryption.
- Hyper-V Virtual Machine Manager can now view the fabrics upon which a shielded virtual is authorized to run, providing a way for the fabric administrator to open a shielded virtual machine's key protector (KP) and view the fabrics it is permitted to run on.
- You can switch Attestation modes on a running Host Guardian Service. Now you can switch on the fly between the less secure but simpler Active Directory-based attestation and TPM-based attestation.

- End-to-end diagnostics tooling based on Windows PowerShell that is able to detect misconfigurations or errors in both guarded Hyper-V hosts and the Host Guardian Service.
- A recovery environment that offers a means to securely troubleshoot and repair shielded virtual machines within the fabric in which they normally run while offering the same level of protection as the shielded virtual machine itself.
- Host Guardian Service support for existing safe Active Directory – you can direct the Host Guardian Service to use an existing Active Directory forest as its Active Directory instead of creating its own Active Directory instance

For more details and instructions for working with shielded virtual machines, see Guarded Fabric and Shielded VMs.

Identity and Access

New features in Identity improve the ability for organizations to secure Active Directory environments and help them migrate to cloud-only deployments and hybrid deployments, where some applications and services are hosted in the cloud and others are hosted on premises.

Active Directory Certificate Services

Active Directory Certificate Services (AD CS) in Windows Server 2016 increases support for TPM key attestation: You can now use Smart Card KSP for key attestation, and devices that are not joined to the domain can now use NDES enrollment to get certificates that can be attested for keys being in a TPM.

Active Directory Domain Services

Active Directory Domain Services includes improvements to help organizations secure Active Directory environments and provide better identity management experiences for both corporate and personal devices. For more information, see What's new in Active Directory Domain Services (AD DS) in Windows Server 2016.

Active Directory Federation Services

What's New in Active Directory Federation Services. Active Directory Federation Services (AD FS) in Windows Server 2016 includes new features that enable you to configure AD FS to authenticate users stored in Lightweight Directory Access Protocol (LDAP) directories. For more information, see What's New in AD FS for Windows Server 2016.

Web Application Proxy

The latest version of Web Application Proxy focuses on new features that enable publishing and pre-authentication for more applications and improved user experience. Check out the full list of new features that includes pre-authentication for rich client apps such as Exchange ActiveSync and wildcard domains for easier publishing of SharePoint apps. For more information, see Web Application Proxy in Windows Server 2016.

Administration

The Management and Automation area focuses on tool and reference information for IT pros who want to run and manage Windows Server 2016, including Windows PowerShell.

Windows PowerShell 5.1 includes significant new features, including support for developing with classes and new security features that extend its use, improve its usability, and allow you to control and manage Windows-based environments more easily and comprehensively. See New Scenarios and Features in WMF 5.1 for details.

New additions for Windows Server 2016 include: the ability to run PowerShell.exe locally on Nano Server (no longer remote only), new Local Users & Groups cmdlets to replace the GUI, added PowerShell debugging support, and added support in Nano Server for security logging & transcription and JEA.

Here are some other new administration features:

PowerShell Desired State Configuration (DSC) in Windows Management Framework (WMF) 5

Windows Management Framework 5 includes updates to Windows PowerShell Desired State Configuration (DSC), Windows Remote Management (WinRM), and Windows Management Instrumentation (WMI).

For more info about testing the DSC features of Windows Management Framework 5, see the series of blog posts discussed in Validate features of PowerShell DSC. To download, see Windows Management Framework 5.1.

PackageManagement unified package management for software discovery, installation, and inventory

Windows Server 2016 and Windows 10 includes a new PackageManagement feature (formerly called OneGet) that enables IT Professionals or DevOps to automate software discovery, installation, and inventory (SDII), locally or remotely, no matter what the installer technology is and where the software is located.

For more info, see https://github.com/OneGet/oneget/wiki.

PowerShell enhancements to assist digital forensics and help reduce security breaches

To help the team responsible for investigating compromised systems - sometimes known as the "blue team" - we've added additional PowerShell logging and other digital forensics functionality, and we've added functionality to help reduce vulnerabilities in scripts, such as constrained PowerShell, and secure CodeGeneration APIs.

For more info, see the PowerShell ♥ the Blue Team blog post.

Networking

The Networking area addresses networking products and features for the IT professional to design, deploy, and maintain Windows Server 2016.

Software-Defined Networking

You can now both mirror and route traffic to new or existing virtual appliances. Together with a distributed firewall and Network security groups, this enables you to dynamically segment and secure workloads in a manner similar to Azure. Second, you can deploy and manage the entire Software-defined networking (SDN) stack using System Center Virtual Machine Manager. Finally, you can use Docker to manage Windows Server container networking, and associate SDN policies not only with virtual machines but containers as well. For more information, see Plan a Software Defined Network Infrastructure.

TCP performance improvements

The default Initial Congestion Window (ICW) has been increased from 4 to 10 and TCP Fast Open (TFO) has been implemented. TFO reduces the amount of time required to establish a TCP connection and the increased ICW allows larger objects to be transferred in the initial burst. This combination can significantly reduce the time required to transfer an Internet object between the client and the cloud.

In order to improve TCP behavior when recovering from packet loss we have implemented TCP Tail Loss Probe (TLP) and Recent Acknowledgment (RACK). TLP helps convert Retransmit TimeOuts (RTOs) to Fast Recoveries and RACK reduces the time required for Fast Recovery to retransmit a lost packet.

Security and Assurance

The Security and Assurance area Includes security solutions and features for the IT professional to deploy in your data center and cloud environment. For information about security in Windows Server 2016 generally, see Security and Assurance.

Just Enough Administration

Just Enough Administration in Windows Server 2016 is security technology that enables delegated administration for anything that can be managed with Windows PowerShell. Capabilities include support for

running under a network identity, connecting over PowerShell Direct, securely copying files to or from JEA endpoints, and configuring the PowerShell console to launch in a JEA context by default. For more details, see JEA on GitHub.

Credential Guard

Credential Guard uses virtualization-based security to isolate secrets so that only privileged system software can access them. See Protect derived domain credentials with Credential Guard.

Remote Credential Guard

Credential Guard includes support for RDP sessions so that the user credentials remain on the client side and are not exposed on the server side. This also provides Single Sign On for Remote Desktop. See Protect derived domain credentials with Windows Defender Credential Guard.

Device Guard (Code Integrity)

Device Guard provides kernel mode code integrity (KMCI) and user mode code integrity (UMCI) by creating policies that specify what code can run on the server. See Introduction to Windows Defender Device Guard: virtualization-based security and code integrity policies.

Windows Defender

Windows Defender Overview for Windows Server 2016. Windows Server Antimalware is installed and enabled by default in Windows Server 2016, but the user interface for Windows Server Antimalware is not installed. However, Windows Server Antimalware will update antimalware definitions and protect the computer without the user interface. If you need the user interface for Windows Server Antimalware, you can install it after the operating system installation by using the Add Roles and Features Wizard.

Control Flow Guard

Control Flow Guard (CFG) is a platform security feature that was created to combat memory corruption vulnerabilities. See Control Flow Guard for more information.

Storage

Storage in Windows Server 2016 includes new features and enhancements for software-defined storage, as well as for traditional file servers. Below are a few of the new features, for more enhancements and further details, see What's New in Storage in Windows Server 2016.

Storage Spaces Direct

Storage Spaces Direct enables building highly available and scalable storage using servers with local storage. It simplifies the deployment and management of software-defined storage systems and unlocks use of new classes of disk devices, such as SATA SSD and NVMe disk devices, that were previously not possible with clustered Storage Spaces with shared disks.

For more info, see Storage Spaces Direct.

Storage Replica

Storage Replica enables storage-agnostic, block-level, synchronous replication between servers or clusters for disaster recovery, as well as stretching of a failover cluster between sites. Synchronous replication enables mirroring of data in physical sites with crash-consistent volumes to ensure zero data loss at the file-system level. Asynchronous replication allows site extension beyond metropolitan ranges with the possibility of data loss.

For more info, see Storage Replica.

Storage Quality of Service (QoS)

You can now use storage quality of service (QoS) to centrally monitor end-to-end storage performance and create management policies using Hyper-V and CSV clusters in Windows Server 2016.

For more info, see Storage Quality of Service.

Failover Clustering

Windows Server 2016 includes a number of new features and enhancements for multiple servers that are grouped together into a single fault-tolerant cluster using the Failover Clustering feature. Some of the additions are listed below; for a more complete listing, see What's New in Failover Clustering in Windows Server 2016.

Cluster Operating System Rolling Upgrade

Cluster Operating System Rolling Upgrade enables an administrator to upgrade the operating system of the cluster nodes from Windows Server 2012 R2 to Windows Server 2016 without stopping the Hyper-V or the Scale-Out File Server workloads. Using this feature, the downtime penalties against Service Level Agreements (SLA) can be avoided.

For more info, see Cluster Operating System Rolling Upgrade.

Cloud Witness

Cloud Witness is a new type of Failover Cluster quorum witness in Windows Server 2016 that leverages Microsoft Azure as the arbitration point. The Cloud Witness, like any other quorum witness, gets a vote and can participate in the quorum calculations. You can configure cloud witness as a quorum witness using the Configure a Cluster Quorum Wizard.

For more info, see Deploy Cloud Witness.

Health Service

The Health Service improves the day-to-day monitoring, operations, and maintenance experience of cluster resources on a Storage Spaces Direct cluster.

For more info, see Health Service.

Application development

Internet Information Services (IIS) 10.0

New features provided by the IIS 10.0 web server in Windows Server 2016 include:

- Support for the HTTP/2 protocol in the Networking stack and integrated with IIS 10.0, allowing IIS 10.0 websites to automatically serve HTTP/2 requests for supported configurations. This allows numerous enhancements over HTTP/1.1 such as more efficient reuse of connections and decreased latency, improving load times for web pages.
- Ability to run and manage IIS 10.0 in Nano Server. See IIS on Nano Server.
- Support for Wildcard Host Headers, enabling administrators to set up a web server for a domain and then have the web server serve requests for any subdomain.
- A new PowerShell module (IISAdministration) for managing IIS.

For more details see IIS.

Distributed Transaction Coordinator (MSDTC)

Three new features are added in Microsoft Windows 10 and Windows Server 2016:

- A new interface for Resource Manager Rejoin can be used by a resource manager to determine the outcome of an in-doubt transaction after a database restarts due to an error. See IResourceManagerRejoinable::Rejoin for details.
- The DSN name limit is enlarged from 256 bytes to 3072 bytes. See IDtcToXaHelperFactory::Create, IDtcToXaHelperSinglePipe::XARMCreate, or IDtcToXaMapper::RequestNewResourceManager for details.

- Improved tracing allowing you to set a registry key to include an image file path in the trace log file name so you can tell which trace log file to check. See How to enable diagnostic tracing for MS DTC on a Windows-based computer for details on configuring tracing for MSDTC.

Windows Server servicing channels

12/17/2021 • 9 minutes to read • Edit Online

Previously with Windows Server 2016 and Windows Server 2019 there have been two primary release channels available, the Long-Term Servicing Channel and the Semi-Annual Channel. The Long-Term Servicing Channel (LTSC) provides a longer term option focusing on stability, whereas the Semi-Annual Channel (SAC) provided more frequent releases enabling customers to take advantage of innovation more quickly.

Starting with Windows Server 2022, there is one primary release channel available, the Long-Term Servicing Channel. The Semi-Annual Channel in previous versions of Windows Server focused on containers and microservices, and that innovation will continue with Azure Stack HCI.

Long-Term Servicing Channel (LTSC)

With the Long-Term Servicing Channel, a new major version of Windows Server is released every 2-3 years. Users are entitled to 5 years of mainstream support and 5 years of extended support. This channel provides systems with a long servicing option and functional stability, and can be installed with Server Core or Server with Desktop Experience installation options. Deployments of the LTSC of Windows Server are not affected by Semi-Annual Channel releases. The Long-Term Servicing Channel will continue to receive security and non-security updates, but it will not receive the new features and functionality.

Semi-Annual Channel

The Semi-Annual Channel enabled customers who are innovating quickly to take advantage of new operating system capabilities at a faster pace, focused in on containers and microservices. Each release in this channel is supported for 18 months from the initial release.

> **NOTE**
>
> There will be no future Semi-Annual Channel releases of Windows Server. Customers using the SAC should move to Azure Stack HCI where the same release cadence and rapid innovation continues with features such as Azure Kubernetes Service on Azure Stack HCI. Alternatively, use the Long-Term Servicing Channel of Windows Server.

Most of the features introduced in the Semi-Annual Channel have been rolled up into the next Long-Term Servicing Channel release of Windows Server. The Semi-Annual Channel is available to volume-licensed customers with Software Assurance, as well as via the Azure Marketplace or other cloud/hosting service providers and loyalty programs such as Visual Studio Subscriptions.

> **NOTE**
>
> **The current Semi-Annual Channel release is Windows Server, version 20H2**. If you want to put servers in this channel, you should install Windows Server, version 20H2, which can be installed in Server Core mode or as Nano Server run in a container. In-place upgrades from a Long-Term Servicing Channel release aren't supported because they are in **different release channels**. This applies vice versa. You cannot upgrade or change from Semi-Annual Channel to Long-Term Servicing Channel without a clean installation.
>
> A Semi-Annual Channel release isn't an update – it's the next Windows Server release in the Semi-Annual Channel. In-place upgrades from one Semi-Annual Channel release to a later Semi-Annual Channel release are possible. This makes it easier to keep up with the relatively short release cadence.

In this model, Windows Server releases were identified by the year and month of release: for example, in 2017, a release in the 9th month (September) would be identified as **version 1709**. Fresh releases of Windows Server in the Semi-Annual Channel occurred twice each year. The support lifecycle for each release is 18 months. Starting with fall 2020 (20H2) releases, we changed the identification. Instead of a month, the release is named based on the release cycle. For example: **version 20H2**, for a release in the second half of the year 2020.

Key differences

The following table summarizes the key differences between the channels:

DESCRIPTION	LONG-TERM SERVICING CHANNEL (WINDOWS SERVER 2019)	SEMI-ANNUAL CHANNEL (WINDOWS SERVER)
Recommended scenarios	General purpose file servers, Microsoft and non-Microsoft workloads, traditional apps, infrastructure roles, software-defined Datacenter, and hyper-converged infrastructure	Containerized applications, container hosts, and application scenarios benefiting from faster innovation
New releases	Every 2–3 years	Every 6 months
Support	5 years of mainstream support, plus 5 years of extended support	18 months
Editions	All available Windows Server editions	Standard and Datacenter editions
Who can use it?	All customers through all channels	Software Assurance and cloud customers only
Installation options	Server Core and Server with Desktop Experience	Server Core for container host and image and Nano Server container image

IMPORTANT

Please understand that the set of roles and features in Windows Server SAC, only available as Server Core installation option, differs from Windows Server LTSC installed with the Server Core installation option. For example, you cannot use Windows Server SAC as a foundation for services like Storage Spaces Direct.

Device compatibility

Unless otherwise communicated, the minimum hardware requirements to run the Semi-Annual Channel releases are the same as the most recent Long-Term Servicing Channel release of Windows Server. Most hardware drivers will continue to function in these releases.

Servicing

Both the Long-Term Servicing Channel and the Semi-Annual Channel releases will be supported with security updates and non-security updates up to the dates listed in the Microsoft Lifecycle pages. The difference is the length of time that the release is supported, as described above.

Servicing tools

There are many tools with which IT pros can service Windows Server. Each option has its pros and cons, ranging from capabilities and control to simplicity and low administrative requirements. The following are examples of

the servicing tools available to manage servicing updates:

- **Windows Update (stand-alone)**: This option is only available for servers that are connected to the Internet and have Windows Update enabled.
- **Windows Server Update Services (WSUS)** provides extensive control over Windows Server and Windows client updates and is natively available in the Windows Server operating system. In addition to the ability to defer updates, organizations can add an approval layer for updates and choose to deploy them to specific computers or groups of computers whenever ready.
- **Microsoft Endpoint Configuration Manager** provides the greatest control over servicing. IT pros can defer updates, approve them, and have multiple options for targeting deployments and managing bandwidth usage and deployment times.

You've likely already chosen to use at least one of these options based on your resources, staff, and expertise. You can continue using the same process for Semi-Annual Channel Releases: for example, if you already use Configuration Manager to manage updates, you can continue to use it. Similarly, if you are using WSUS, you can continue to use that.

Where to obtain Semi-Annual Channel releases

Semi-Annual Channel releases should be installed as a clean installation. It is possible to use in-place upgrade via ISO from one SAC to a later version.

- Volume Licensing Service Center (VLSC): Volume-licensed customers with Software Assurance can obtain this release by going to the Volume Licensing Service Center and clicking **Sign In**. Then click **Downloads and Keys** and search for this release.
- Semi-Annual Channel releases are also available in Microsoft Azure.
- Visual Studio Subscriptions: Visual Studio Subscribers can obtain Semi-Annual Channel releases by downloading them from the Visual Studio Subscriber download page. If you are not already a subscriber, go to Visual Studio Subscriptions to sign up, and then visit the Visual Studio Subscriber download page as above. Releases obtained through Visual Studio Subscriptions are for development and testing only.

Activating Semi-Annual Channel releases

- If you're using Microsoft Azure, Semi-Annual Channel releases should be activated automatically.
- If you've obtained this release from the Volume Licensing Service Center or Visual Studio Subscriptions, you can activate it by using your Windows Server Customer Specific Volume License Key (CSVLK, also known as the KMS host key) with your Key Management System (KMS) environment. For more information, see KMS client setup keys.

> **NOTE**
> For easier maintenance and management of activation, you can use ADBA (Active Directory-based activation) for Windows Server 2012 or later, including Windows Server SAC. In addition, you can manage your licenses using VAMT 3.x (Volume Activation Management Tool), which is part of the latest ADK.

Semi-Annual Channel releases that were released with or after Windows Server 2019 use the Windows Server 2019 CSVLK. Semi-Annual Channel releases that were released before Windows Server 2019 use the Windows Server 2016 CSVLK.

Why do Semi-Annual Channel releases offer only the Server Core installation option?

One of the most important steps we take in planning each release of Windows Server is listening to customer feedback – how are you using Windows Server? What new features will have the greatest impact on your Windows Server deployments, and by extension, your day-to-day business? Your feedback tells us that delivering new innovation as quickly and efficiently as possible is a key priority. At the same time, for those customers innovating most quickly, you've told us that you're primarily using command line scripting with PowerShell to manage your data centers, and as such don't have a strong need for the desktop GUI available in the installation of Windows Server with Desktop Experience, especially now that Windows Admin Center is available to remotely manage your servers.

By focusing on the Server Core installation option, we're able to dedicate more resources toward those new innovations, while also maintaining traditional Windows Server platform functionality and application compatibility.

Starting with Windows Server, version 1809 and Windows Server 2019, Server Core App Compatibility Feature on Demand (FOD) is an optional feature package that significantly improves the app compatibility of the Windows Server Core installation option by including a subset of binaries and packages from Windows Server with Desktop Experience, without adding the Windows Server Desktop Experience graphical environment.

What about Nano Server?

Nano Server is available as a container operating system in the Semi-Annual Channel. See Changes to Nano Server in Windows Server Semi-Annual Channel for details.

How to tell whether a server is running an LTSC or SAC release

Previously, Long-Term Servicing Channel releases such as Windows Server 2019 were released at the same time as a new version of the Semi-Annual Channel, for example, Windows Server, version 1809 was released at the same time as Windows Server 2019. This can make it a little tricky to determine whether a server is running Semi-Annual Channel release. Instead of looking at the build number, you must look at the product name: Semi-Annual Channel releases use the Windows Server Standard or Windows Server Datacenter product name, without a version number, while Long-Term Servicing Channel releases include the version number, for example, Windows Server 2019 Datacenter.

> **NOTE**
> The below guidance is intended to help identify and differentiate between LTSC and SAC for lifecycle and general inventory purposes only. It is not intended for application compatibility or to represent a specific API surface. App developers should use guidance elsewhere to properly ensure compatibility as components, APIs, and functionality can be added over the life of a system, or not yet be added. Operating System Version is a better starting point for App Developers.

Open PowerShell and use the `Get-ItemProperty` cmdlet, or the `Get-ComputerInfo` cmdlet, to check these properties in the registry. Along with build number, this will indicate LTSC or SAC by the presence, or lack thereof, of the branded year, i.e. 2019 - LTSC has this, SAC does not. This will also return the timing of the release with ReleaseId or WindowsVersion, i.e. 1809, as well as whether the installation is Server Core or Server with Desktop Experience.

Windows Server 2019 Datacenter Edition (LTSC) with Desktop Experience example:

```
Get-ItemProperty -Path "HKLM:\Software\Microsoft\Windows NT\CurrentVersion" | Select ProductName, ReleaseId,
InstallationType, CurrentMajorVersionNumber,CurrentMinorVersionNumber,CurrentBuild
```

```
ProductName               : Windows Server 2019 Datacenter
ReleaseId                 : 1809
InstallationType          : Server
CurrentMajorVersionNumber : 10
CurrentMinorVersionNumber : 0
CurrentBuild              : 17763
```

Windows Server, version 1809 (SAC) Standard Edition Server Core example:

```
Get-ItemProperty -Path "HKLM:\Software\Microsoft\Windows NT\CurrentVersion" | Select ProductName, ReleaseId,
InstallationType, CurrentMajorVersionNumber,CurrentMinorVersionNumber,CurrentBuild
```

```
ProductName               : Windows Server Standard
ReleaseId                 : 1809
InstallationType          : Server Core
CurrentMajorVersionNumber : 10
CurrentMinorVersionNumber : 0
CurrentBuild              : 17763
```

Windows Server 2019 Standard Edition (LTSC) Server Core example:

```
Get-ComputerInfo | Select WindowsProductName, WindowsVersion, WindowsInstallationType, OsServerLevel,
OsVersion, OsHardwareAbstractionLayer
```

```
WindowsProductName            : Windows Server 2019 Standard
WindowsVersion                : 1809
WindowsInstallationType       : Server Core
OsServerLevel                 : ServerCore
OsVersion                     : 10.0.17763
OsHardwareAbstractionLayer    : 10.0.17763.107
```

To query if the new Server Core App Compatibility Feature on Demand (FOD) is present on a server, use the Get-WindowsCapability cmdlet and look for:

```
Name    :     ServerCore.AppCompatibility~~~~0.0.1.0
State   :     Installed
```

Comparison of Standard, Datacenter, and Datacenter: Azure Edition editions of Windows Server 2022

12/17/2021 • 5 minutes to read • Edit Online

Use this article to compare Standard, Datacenter, and Datacenter: Azure Edition editions of Windows Server 2022 to see which will be most appropriate.

Features available generally

FEATURES AVAILABLE GENERALLY	WINDOWS SERVER 2022 STANDARD	WINDOWS SERVER 2022 DATACENTER	WINDOWS SERVER 2022 DATACENTER: AZURE EDITION
Azure Extended Network	No	No	**Yes**
Best Practices Analyzer	Yes	Yes	Yes
Containers	Yes	Yes	Yes
Direct Access	Yes	Yes	Yes
Dynamic Memory (in virtualization)	Yes	Yes	Yes
Hot Add/Replace RAM	Yes	Yes	Yes
Hotpatching	No	No	**Yes**
Microsoft Management Console	Yes	Yes	Yes
Minimal Server Interface	Yes	Yes	Yes
Network Load Balancing	Yes	Yes	Yes
Windows PowerShell	Yes	Yes	Yes
Server Core installation option	Yes	Yes	Yes
Server Manager	Yes	Yes	Yes
SMB Direct and SMB over RDMA	Yes	Yes	Yes (not supported in Azure)
SMB over QUIC	No	No	**Yes**

FEATURES AVAILABLE GENERALLY	WINDOWS SERVER 2022 STANDARD	WINDOWS SERVER 2022 DATACENTER	WINDOWS SERVER 2022 DATACENTER: AZURE EDITION
Software-defined Networking	No	**Yes**	**Yes**
Storage Migration Service	Yes	Yes	Yes
Storage Replica	Yes, (1 partnership and 1 resource group with a single 2TB volume)	Yes, **unlimited**	Yes, **unlimited**
Storage Spaces	Yes	Yes	Yes
Storage Spaces Direct	No	**Yes**	**Yes**
Volume Activation Services	Yes	Yes	Yes
VSS (Volume Shadow Copy Service) integration	Yes	Yes	Yes
Windows Server Update Services	Yes	Yes	Yes
Windows System Resource Manager	Yes	Yes	Yes
Server license logging	Yes	Yes	Yes
Inherited activation	As guest if hosted on Datacenter	**Can be a host or a guest**	**Can be a host or a guest**
Work Folders	Yes	Yes	Yes

Locks and Limits

LOCKS AND LIMITS	WINDOWS SERVER 2022 STANDARD	WINDOWS SERVER 2022 DATACENTER
Maximum number of users	Based on CALs	Based on CALs
Maximum SMB connections	16,777,216	16,777,216
Maximum RRAS connections	Unlimited	Unlimited
Maximum IAS connections	2,147,483,647	2,147,483,647
Maximum RDS connections	65,535	65,535
Maximum number of 64-bit sockets	64	64
Maximum number of cores	Unlimited	Unlimited

LOCKS AND LIMITS	WINDOWS SERVER 2022 STANDARD	WINDOWS SERVER 2022 DATACENTER
Maximum RAM	48 TB	48 TB
Can be used as virtualization guest	Yes; 2 virtual machines, plus one Hyper-V host per license	Yes; **unlimited virtual machines**, plus one Hyper-V host per license
Windows Server Containers	Unlimited	Unlimited
Virtual OSE/Hyper-V isolated Containers	2	Unlimited
Server can join a domain	Yes	Yes
Edge network protection/firewall	No	No
DirectAccess	Yes	Yes
DLNA codecs and web media streaming	Yes, if installed as Server with Desktop Experience	Yes, if installed as Server with Desktop Experience

Server roles

WINDOWS SERVER ROLES AVAILABLE	ROLE SERVICES	WINDOWS SERVER 2022 STANDARD	WINDOWS SERVER 2022 DATACENTER
Active Directory Certificate Services		Yes	Yes
Active Directory Domain Services		Yes	Yes
Active Directory Federation Services		Yes	Yes
AD Lightweight Directory Services		Yes	Yes
AD Rights Management Services		Yes	Yes
Device Health Attestation		Yes	Yes
DHCP Server		Yes	Yes
DNS Server		Yes	Yes
Fax Server		Yes	Yes
File and Storage Services	File Server	Yes	Yes
File and Storage Services	BranchCache for Network Files	Yes	Yes

WINDOWS SERVER ROLES AVAILABLE	ROLE SERVICES	WINDOWS SERVER 2022 STANDARD	WINDOWS SERVER 2022 DATACENTER
File and Storage Services	Data Deduplication	Yes	Yes
File and Storage Services	DFS Namespaces	Yes	Yes
File and Storage Services	DFS Replication	Yes	Yes
File and Storage Services	File Server Resource Manager	Yes	Yes
File and Storage Services	File Server VSS Agent Service	Yes	Yes
File and Storage Services	iSCSI Target Server	Yes	Yes
File and Storage Services	iSCSI Target Storage Provider	Yes	Yes
File and Storage Services	Server for NFS	Yes	Yes
File and Storage Services	Work Folders	Yes	Yes
File and Storage Services	Storage Services	Yes	Yes
Host Guardian Service		Yes	Yes
Hyper-V		Yes	Yes; including Shielded Virtual Machines
Network Controller		No	**Yes**
Network Policy and Access Services		Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Print and Document Services		Yes	Yes
Remote Access		Yes	Yes
Remote Desktop Services		Yes	Yes
Volume Activation Services		Yes	Yes
Web Services (IIS)		Yes	Yes
Windows Deployment Services		Yes	Yes
Windows Server Essentials Experience		No	No

WINDOWS SERVER ROLES AVAILABLE	ROLE SERVICES	WINDOWS SERVER 2022 STANDARD	WINDOWS SERVER 2022 DATACENTER
Windows Server Update Services		Yes	Yes

Features

WINDOWS SERVER FEATURES AVAILABLE	WINDOWS SERVER 2022 STANDARD	WINDOWS SERVER 2022 DATACENTER
.NET Framework 3.5	Yes	Yes
.NET Framework 4.8	Yes	Yes
Background Intelligent Transfer Service (BITS)	Yes	Yes
BitLocker Drive Encryption	Yes	Yes
BitLocker Network Unlock	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
BranchCache	Yes	Yes
Client for NFS	Yes	Yes
Containers	Yes	Yes
Data Center Bridging	Yes	Yes
Direct Play	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Enhanced Storage	Yes	Yes
Failover Clustering	Yes	Yes
Group Policy Management	Yes	Yes
Host Guardian Hyper-V Support	No	**Yes**
I/O Quality of Service	Yes	Yes
IIS Hostable Web Core	Yes	Yes
Internet Printing Client	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
IP Address Management (IPAM) Server	Yes	Yes

WINDOWS SERVER FEATURES AVAILABLE	WINDOWS SERVER 2022 STANDARD	WINDOWS SERVER 2022 DATACENTER
LPR Port Monitor	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Management OData IIS Extension	Yes	Yes
Media Foundation	Yes	Yes
Message Queueing	Yes	Yes
Microsoft Defender Antivirus	Installed	Installed
Multipath I/O	Yes	Yes
MultiPoint Connector	Yes	Yes
Network Load Balancing	Yes	Yes
Network Virtualization	Yes	Yes
Peer Name Resolution Protocol	Yes	Yes
Quality Windows Audio Video Experience	Yes	Yes
RAS Connection Manager Administration Kit (CMAK)	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Remote Assistance	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Remote Differential Compression	Yes	Yes
Remote Server Administration Tools (RSAT)	Yes	Yes
RPC over HTTP Proxy	Yes	Yes
Setup and Boot Event Collection	Yes	Yes
Simple TCP/IP Services	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
SMB 1.0/CIFS File Sharing Support	Installed	Installed
SMB Bandwidth Limit	Yes	Yes
SMTP Server	Yes	Yes
SNMP Service	Yes	Yes
Software Load Balancer	Yes	Yes

WINDOWS SERVER FEATURES AVAILABLE	WINDOWS SERVER 2022 STANDARD	WINDOWS SERVER 2022 DATACENTER
Storage Migration Service	Yes	Yes
Storage Migration Service Proxy	Yes	Yes
Storage Replica	Yes	Yes
System Data Archiver	Yes	Yes
System Insights	Yes	Yes
Telnet Client	Yes	Yes
TFTP Client	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
VM Shielding Tools for Fabric Management	Yes	Yes
WebDAV Redirector	Yes	Yes
Windows Biometric Framework	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Windows Identity Foundation 3.5	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Windows Internal Database	Yes	Yes
Windows PowerShell 5.1	Installed	Installed
Windows PowerShell 2.0 Engine	Yes	Yes
Windows PowerShell Desired State Configuration Service	Yes	Yes
Windows PowerShell Web Access	Yes	Yes
Windows Process Activation Service	Yes	Yes
Windows Search Service	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Windows Server Backup	Yes	Yes
Windows Server Migration Tools	Yes	Yes
Windows Standards-Based Storage Management	Yes	Yes
Windows Subsystem for Linux	Yes	Yes

WINDOWS SERVER FEATURES AVAILABLE	WINDOWS SERVER 2022 STANDARD	WINDOWS SERVER 2022 DATACENTER
Windows TIFF IFilter	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
WinRM IIS Extension	Yes	Yes
WINS Server	Yes	Yes
Wireless LAN Service	Yes	Yes
WoW64 support	Installed	Installed
XPS Viewer	Installed with Server with Desktop Experience	Installed with Server with Desktop Experience

Comparison of Standard and Datacenter editions of Windows Server 2019

12/17/2021 • 5 minutes to read • Edit Online

Use this article to compare Standard and Datacenter editions of Windows Server 2019 to see which will be most appropriate.

Features available generally

FEATURES AVAILABLE GENERALLY	WINDOWS SERVER 2019 STANDARD	WINDOWS SERVER 2019 DATACENTER
Best Practices Analyzer	Yes	Yes
Direct Access	Yes	Yes
Dynamic Memory (in virtualization)	Yes	Yes
Hot Add/Replace RAM	Yes	Yes
Microsoft Management Console	Yes	Yes
Minimal Server Interface	Yes	Yes
Network Load Balancing	Yes	Yes
Windows PowerShell	Yes	Yes
Server Core installation option	Yes	Yes
Server Manager	Yes	Yes
SMB Direct and SMB over RDMA	Yes	Yes
Software-defined Networking	No	**Yes**
Storage Migration Service	Yes	Yes
Storage Replica	Yes, (1 partnership and 1 resource group with a single 2TB volume)	Yes, **unlimited**
Storage Spaces	Yes	Yes
Storage Spaces Direct	No	**Yes**
Volume Activation Services	Yes	Yes
VSS (Volume Shadow Copy Service) integration	Yes	Yes

FEATURES AVAILABLE GENERALLY	WINDOWS SERVER 2019 STANDARD	WINDOWS SERVER 2019 DATACENTER
Windows Server Update Services	Yes	Yes
Windows System Resource Manager	Yes	Yes
Server license logging	Yes	Yes
Inherited activation	As guest if hosted on Datacenter	**Can be a host or a guest**
Work Folders	Yes	Yes

Locks and Limits

LOCKS AND LIMITS	WINDOWS SERVER 2019 STANDARD	WINDOWS SERVER 2019 DATACENTER
Maximum number of users	Based on CALs	Based on CALs
Maximum SMB connections	16,777,216	16,777,216
Maximum RRAS connections	unlimited	unlimited
Maximum IAS connections	2,147,483,647	2,147,483,647
Maximum RDS connections	65,535	65,535
Maximum number of 64-bit sockets	64	64
Maximum number of cores	unlimited	unlimited
Maximum RAM	24 TB	24 TB
Can be used as virtualization guest	Yes; 2 virtual machines, plus one Hyper-V host per license	Yes; **unlimited virtual machines**, plus one Hyper-V host per license
Server can join a domain	yes	yes
Edge network protection/firewall	no	no
DirectAccess	yes	yes
DLNA codecs and web media streaming	Yes, if installed as Server with Desktop Experience	Yes, if installed as Server with Desktop Experience

Server roles

WINDOWS SERVER ROLES AVAILABLE	ROLE SERVICES	WINDOWS SERVER 2019 STANDARD	WINDOWS SERVER 2019 DATACENTER
Active Directory Certificate Services		Yes	Yes

WINDOWS SERVER ROLES AVAILABLE	ROLE SERVICES	WINDOWS SERVER 2019 STANDARD	WINDOWS SERVER 2019 DATACENTER
Active Directory Domain Services		Yes	Yes
Active Directory Federation Services		Yes	Yes
AD Lightweight Directory Services		Yes	Yes
AD Rights Management Services		Yes	Yes
Device Health Attestation		Yes	Yes
DHCP Server		Yes	Yes
DNS Server		Yes	Yes
Fax Server		Yes	Yes
File and Storage Services	File Server	Yes	Yes
File and Storage Services	BranchCache for Network Files	Yes	Yes
File and Storage Services	Data Deduplication	Yes	Yes
File and Storage Services	DFS Namespaces	Yes	Yes
File and Storage Services	DFS Replication	Yes	Yes
File and Storage Services	File Server Resource Manager	Yes	Yes
File and Storage Services	File Server VSS Agent Service	Yes	Yes
File and Storage Services	iSCSI Target Server	Yes	Yes
File and Storage Services	iSCSI Target Storage Provider	Yes	Yes
File and Storage Services	Server for NFS	Yes	Yes
File and Storage Services	Work Folders	Yes	Yes
File and Storage Services	Storage Services	Yes	Yes
Host Guardian Service		Yes	Yes

WINDOWS SERVER ROLES AVAILABLE	ROLE SERVICES	WINDOWS SERVER 2019 STANDARD	WINDOWS SERVER 2019 DATACENTER
Hyper-V		Yes	Yes; including Shielded Virtual Machines
Network Controller		No	**Yes**
Network Policy and Access Services		Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Print and Document Services		Yes	Yes
Remote Access		Yes	Yes
Remote Desktop Services		Yes	Yes
Volume Activation Services		Yes	Yes
Web Services (IIS)		Yes	Yes
Windows Deployment Services		Yes*	Yes*
Windows Server Essentials Experience		No	No
Windows Server Update Services		Yes	Yes

*WDS Transport Server is new to Server Core installations in Windows Server 2019 (also in the Semi-Annual Channel starting with Windows Server, version 1803)

Features

WINDOWS SERVER FEATURES AVAILABLE	WINDOWS SERVER 2019 STANDARD	WINDOWS SERVER 2019 DATACENTER
.NET Framework 3.5	Yes	Yes
.NET Framework 4.7	Yes	Yes
Background Intelligent Transfer Service (BITS)	Yes	Yes
BitLocker Drive Encryption	Yes	Yes
BitLocker Network Unlock	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
BranchCache	Yes	Yes

WINDOWS SERVER FEATURES AVAILABLE	WINDOWS SERVER 2019 STANDARD	WINDOWS SERVER 2019 DATACENTER
Client for NFS	Yes	Yes
Containers	Yes (unlimited Windows containers; up to two Hyper-V containers)	Yes (**unlimited Windows and Hyper-V containers**)
Data Center Bridging	Yes	Yes
Direct Play	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Enhanced Storage	Yes	Yes
Failover Clustering	Yes	Yes
Group Policy Management	Yes	Yes
Host Guardian Hyper-V Support	No	**Yes**
I/O Quality of Service	Yes	Yes
IIS Hostable Web Core	Yes	Yes
Internet Printing Client	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
IPAM Server	Yes	Yes
iSNS Server service	Yes	Yes
LPR Port Monitor	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Management OData IIS Extension	Yes	Yes
Media Foundation	Yes	Yes
Message Queueing	Yes	Yes
Multipath I/O	Yes	Yes
MultiPoint Connector	Yes	Yes
Network Load Balancing	Yes	Yes
Peer Name Resolution Protocol	Yes	Yes
Quality Windows Audio Video Experience	Yes	Yes
RAS Connection Manager Administration Kit	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience

WINDOWS SERVER FEATURES AVAILABLE	WINDOWS SERVER 2019 STANDARD	WINDOWS SERVER 2019 DATACENTER
Remote Assistance	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Remote Differential Compression	Yes	Yes
RSAT	Yes	Yes
RPC over HTTP Proxy	Yes	Yes
Setup and Boot Event Collection	Yes	Yes
Simple TCP/IP Services	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
SMB 1.0/CIFS File Sharing Support	Installed	Installed
SMB Bandwidth Limit	Yes	Yes
SMTP Server	Yes	Yes
SNMP Service	Yes	Yes
Software Load Balancer	Yes	Yes
Storage Replica	Yes	Yes
Telnet Client	Yes	Yes
TFTP Client	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
VM Shielding Tools for Fabric Management	Yes	Yes
WebDAV Redirector	Yes	Yes
Windows Biometric Framework	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Windows Defender features	Installed	Installed
Windows Identity Foundation 3.5	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Windows Internal Database	Yes	Yes
Windows PowerShell	Installed	Installed
Windows Process Activation Service	Yes	Yes

WINDOWS SERVER FEATURES AVAILABLE	WINDOWS SERVER 2019 STANDARD	WINDOWS SERVER 2019 DATACENTER
Windows Search Service	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Windows Server Backup	Yes	Yes
Windows Server Migration Tools	Yes	Yes
Windows Standards-Based Storage Management	Yes	Yes
Windows TIFF IFilter	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
WinRM IIS Extension	Yes	Yes
WINS Server	Yes	Yes
Wireless LAN Service	Yes	Yes
WoW64 support	Installed	Installed
XPS Viewer	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience

Comparison of Standard and Datacenter editions of Windows Server 2016

12/17/2021 • 5 minutes to read • Edit Online

Use this article to compare Standard and Datacenter editions of Windows Server 2016 to see which will be most appropriate.

Features available generally

FEATURES AVAILABLE GENERALLY	WINDOWS SERVER 2016 STANDARD	WINDOWS SERVER 2016 DATACENTER
Best Practices Analyzer	Yes	Yes
Direct Access	Yes	Yes
Dynamic Memory (in virtualization)	Yes	Yes
Hot Add/Replace RAM	Yes	Yes
Microsoft Management Console	Yes	Yes
Minimal Server Interface	Yes	Yes
Network Load Balancing	Yes	Yes
Windows PowerShell	Yes	Yes
Server Core installation option	Yes	Yes
Nano Server installation option	Yes	Yes
Server Manager	Yes	Yes
SMB Direct and SMB over RDMA	Yes	Yes
Software-defined Networking	No	**Yes**
Storage Replica	No	**Yes**
Storage Spaces	Yes	Yes
Storage Spaces Direct	No	**Yes**
Volume Activation Services	Yes	Yes
VSS (Volume Shadow Copy Service) integration	Yes	Yes

FEATURES AVAILABLE GENERALLY	WINDOWS SERVER 2016 STANDARD	WINDOWS SERVER 2016 DATACENTER
Windows Server Update Services	Yes	Yes
Windows System Resource Manager	Yes	Yes
Server license logging	Yes	Yes
Inherited activation	As guest if hosted on Datacenter	**Can be host or guest**
Work Folders	Yes	Yes

Locks and Limits

LOCKS AND LIMITS	WINDOWS SERVER 2016 STANDARD	WINDOWS SERVER 2016 DATACENTER
Maximum number of users	Based on CALs	Based on CALs
Maximum SMB connections	16,777,216	16,777,216
Maximum RRAS connections	unlimited	unlimited
Maximum IAS connections	2,147,483,647	2,147,483,647
Maximum RDS connections	65535	65535
Maximum number of 64-bit sockets	64	64
Maximum number of cores	unlimited	unlimited
Maximum RAM	24 TB	24 TB
Can be used as virtualization guest	Yes; 2 virtual machines, plus one Hyper-V host per license	Yes; **unlimited virtual machines**, plus one Hyper-V host per license
Server can join a domain	yes	yes
Edge network protection/firewall	no	no
DirectAccess	yes	yes
DLNA codecs and web media streaming	Yes, if installed as Server with Desktop Experience	Yes, if installed as Server with Desktop Experience

Server roles

WINDOWS SERVER ROLES AVAILABLE	ROLE SERVICES	WINDOWS SERVER 2016 STANDARD	WINDOWS SERVER 2016 DATACENTER
Active Directory Certificate Services		Yes	Yes

WINDOWS SERVER ROLES AVAILABLE	ROLE SERVICES	WINDOWS SERVER 2016 STANDARD	WINDOWS SERVER 2016 DATACENTER
Active Directory Domain Services		Yes	Yes
Active Directory Federation Services		Yes	Yes
AD Lightweight Directory Services		Yes	Yes
AD Rights Management Services		Yes	Yes
Device Health Attestation		Yes	Yes
DHCP Server		Yes	Yes
DNS Server		Yes	Yes
Fax Server		Yes	Yes
File and Storage Services	File Server	Yes	Yes
File and Storage Services	BranchCache for Network Files	Yes	Yes
File and Storage Services	Data Deduplication	Yes	Yes
File and Storage Services	DFS Namespaces	Yes	Yes
File and Storage Services	DFS Replication	Yes	Yes
File and Storage Services	File Server Resource Manager	Yes	Yes
File and Storage Services	File Server VSS Agent Service	Yes	Yes
File and Storage Services	iSCSI Target Server	Yes	Yes
File and Storage Services	iSCSI Target Storage Provider	Yes	Yes
File and Storage Services	Server for NFS	Yes	Yes
File and Storage Services	Work Folders	Yes	Yes
File and Storage Services	Storage Services	Yes	Yes
Host Guardian Service		Yes	Yes

WINDOWS SERVER ROLES AVAILABLE	ROLE SERVICES	WINDOWS SERVER 2016 STANDARD	WINDOWS SERVER 2016 DATACENTER
Hyper-V		Yes	Yes; **including Shielded Virtual Machines**
MultiPoint Services		Yes	Yes
Network Controller		No	**Yes**
Network Policy and Access Services		Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Print and Document Services		Yes	Yes
Remote Access		Yes	Yes
Remote Desktop Services		Yes	Yes
Volume Activation Services		Yes	Yes
Web Services (IIS)		Yes	Yes
Windows Deployment Services		Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Windows Server Essentials Experience		Yes	Yes
Windows Server Update Services		Yes	Yes

Features

WINDOWS SERVER FEATURES AVAILABLE	WINDOWS SERVER 2016 STANDARD	WINDOWS SERVER 2016 DATACENTER
.NET Framework 3.5	Yes	Yes
.NET Framework 4.6	Yes	Yes
Background Intelligent Transfer Service (BITS)	Yes	Yes
BitLocker Drive Encryption	Yes	Yes
BitLocker Network Unlock	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
BranchCache	Yes	Yes

WINDOWS SERVER FEATURES AVAILABLE	WINDOWS SERVER 2016 STANDARD	WINDOWS SERVER 2016 DATACENTER
Client for NFS	Yes	Yes
Containers	Yes (Windows containers unlimited; Hyper-V containers up to 2)	Yes (all container types unlimited)
Data Center Bridging	Yes	Yes
Direct Play	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Enhanced Storage	Yes	Yes
Failover Clustering	Yes	Yes
Group Policy Management	Yes	Yes
Host Guardian Hyper-V Support	No	**Yes**
I/O Quality of Service	Yes	Yes
IIS Hostable Web Core	Yes	Yes
Internet Printing Client	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
IPAM Server	Yes	Yes
iSNS Server service	Yes	Yes
LPR Port Monitor	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Management OData IIS Extension	Yes	Yes
Media Foundation	Yes	Yes
Message Queueing	Yes	Yes
Multipath I/O	Yes	Yes
MultiPoint Connector	Yes	Yes
Network Load Balancing	Yes	Yes
Peer Name Resolution Protocol	Yes	Yes
Quality Windows Audio Video Experience	Yes	Yes
RAS Connection Manager Administration Kit	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience

WINDOWS SERVER FEATURES AVAILABLE	WINDOWS SERVER 2016 STANDARD	WINDOWS SERVER 2016 DATACENTER
Remote Assistance	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Remote Differential Compression	Yes	Yes
RSAT	Yes	Yes
RPC over HTTP Proxy	Yes	Yes
Setup and Boot Event Collection	Yes	Yes
Simple TCP/IP Services	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
SMB 1.0/CIFS File Sharing Support	Installed	Installed
SMB Bandwidth Limit	Yes	Yes
SMTP Server	Yes	Yes
SNMP Service	Yes	Yes
Software Load Balancer	No	**Yes**
Storage Replica	No	Yes
Telnet Client	Yes	Yes
TFTP Client	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
VM Shielding Tools for Fabric Management	Yes	Yes
WebDAV Redirector	Yes	Yes
Windows Biometric Framework	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Windows Defender features	Installed	Installed
Windows Identity Foundation 3.5	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Windows Internal Database	Yes	Yes
Windows PowerShell	Installed	Installed
Windows Process Activation Service	Yes	Yes

WINDOWS SERVER FEATURES AVAILABLE	WINDOWS SERVER 2016 STANDARD	WINDOWS SERVER 2016 DATACENTER
Windows Search Service	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
Windows Server Backup	Yes	Yes
Windows Server Migration Tools	Yes	Yes
Windows Standards-Based Storage Management	Yes	Yes
Windows TIFF IFilter	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience
WinRM IIS Extension	Yes	Yes
WINS Server	Yes	Yes
Wireless LAN Service	Yes	Yes
WoW64 support	Installed	Installed
XPS Viewer	Yes, when installed as Server with Desktop Experience	Yes, when installed as Server with Desktop Experience

Hardware requirements for Windows Server

12/17/2021 • 4 minutes to read • Edit Online

This article outlines the minimum hardware requirements to run Windows Server. If your computer has less than the minimum requirements, you will not be able to install this product correctly. Actual requirements will vary based on your system configuration and the applications and features you install.

Unless otherwise specified, these minimum hardware requirements apply to all installation options (Server Core and Server with Desktop Experience) and both Standard and Datacenter editions.

> **IMPORTANT**
> The highly diverse scope of potential deployments makes it unrealistic to state recommended hardware requirements that would be generally applicable. Consult documentation for each of the server roles you intend to deploy for more details about the resource needs of particular server roles. For the best results, conduct test deployments to determine appropriate hardware requirements for your particular deployment scenarios.

Processor

Processor performance depends not only on the clock frequency of the processor, but also on the number of processor cores and the size of the processor cache. The following are the processor requirements for this product:

Minimum:

- 1.4 GHz 64-bit processor
- Compatible with x64 instruction set
- Supports NX and DEP
- Supports CMPXCHG16b, LAHF/SAHF, and PrefetchW
- Supports Second Level Address Translation (EPT or NPT)

Coreinfo, part of Windows Sysinternals, is a tool you can use to confirm which of these capabilities your CPU has.

RAM

The following are the estimated RAM requirements for this product:

Minimum:

- 512 MB (2 GB for Server with Desktop Experience installation option)
- ECC (Error Correcting Code) type or similar technology, for physical host deployments

IMPORTANT

If you create a virtual machine with the minimum supported hardware parameters (1 processor core and 512 MB RAM) and then attempt to install this release on the virtual machine, Setup will fail.

To avoid this, do one of the following:

- Allocate more than 800 MB RAM to the virtual machine on which you intend to install this release. Once Setup has completed, you can change the allocation to as little as 512 MB RAM, depending on the actual server configuration. If you've modified the boot image for Setup with addition languages and updates, you may need to allocate more than 800 MB RAM in order to complete the installation
- Interrupt the boot process of this release on the virtual machine with the keyboard combination `SHIFT+F10`. In the command prompt that opens, use `diskpart.exe` to create and format an installation partition. Run `wpeutil createpagefile /path=C:\pf.sys` (assuming the installation partition you created was C:\). Then close the command prompt and proceed with Setup.

Storage controller and disk space requirements

Computers that run Windows Server must include a storage adapter that is compliant with the PCI Express architecture specification. Persistent storage devices on servers classified as hard disk drives must not be PATA. Windows Server does not allow ATA/PATA/IDE/EIDE for boot, page, or data drives.

The following are the estimated **minimum** disk space requirements for the system partition.

Minimum: 32 GB

NOTE

Be aware that 32 GB should be considered an *absolute minimum* value for successful installation. This minimum should allow you to install Windows Server 2022 using the Server Core installation option, with the Web Services (IIS) server role. A server in Server Core mode is about 4 GB smaller than the same server using the Server with Desktop Experience installation option.

The system partition will need extra space for any of the following circumstances:

- If you install the system over a network.
- Computers with more than 16 GB of RAM will require more disk space for paging, hibernation, and dump files.

Network adapter requirements

Network adapters used with this release should include these features:

Minimum:

- An ethernet adapter capable of at least 1 gigabit per second throughput
- Compliant with the PCI Express architecture specification.

A network adapter that supports network debugging (KDNet) is useful, but not a minimum requirement.

A network adapter that supports the Pre-boot Execution Environment (PXE) is useful, but not a minimum requirement.

Other requirements

Computers running this release also must have the following:

- DVD drive (if you intend to install the operating system from DVD media)

The following items are only required for certain features:

- UEFI 2.3.1c-based system and firmware that supports secure boot
- Trusted Platform Module
- Graphics device and monitor capable of Super VGA (1024 x 768) or higher-resolution
- Keyboard and Microsoft mouse (or other compatible pointing device)
- Internet access (fees may apply)

> **NOTE**
>
> A Trusted Platform Module (TPM) chip is required in order to use certain features such as BitLocker Drive Encryption. If your computer uses TPM, it must meet these requirements:
>
> - Hardware-based TPMs must implement version 2.0 of the TPM specification.
> - TPMs that implement version 2.0 must have an EK certificate that is either pre-provisioned to the TPM by the hardware vendor or be capable of being retrieved by the device during the first boot.
> - TPMs that implement version 2.0 must ship with SHA-256 PCR banks and implement PCRs 0 through 23 for SHA-256. It is acceptable to ship TPMs with a single switchable PCR bank that can be used for both SHA-1 and SHA-256 measurements.
>
> A UEFI option to turn off the TPM is not a requirement.

Features removed or no longer developed starting with Windows Server 2022

12/17/2021 • 3 minutes to read • Edit Online

Each release of Windows Server adds new features and functionality; we also occasionally remove features and functionality, usually because we've added a better option. Here are the details about the features and functionalities that we removed in Windows Server 2022.

TIP

- You can get early access to Windows Server builds by joining the Windows Insider Program for Business - this is a great way to test feature changes.

The list is subject to change and might not include every affected feature or functionality.

Semi-Annual Channel

As part of our customer-centric approach, we'll move to the Long-Term Servicing Channel (LTSC) as our primary release channel. Current Semi-Annual Channel (SAC) releases will continue through their mainstream support end dates, which are May 10, 2022 for Windows Server version 20H2 and December 14, 2021 for Windows Server version 2004

The focus on container and microservice innovation previously released in the Semi-Annual Channel will now continue with Azure Kubernetes Service (AKS), AKS on Azure Stack HCI, and other platform improvements made in collaboration with the Kubernetes community. And with the Long-Term Servicing Channel, a major new version of Windows Server will be released every 2-3 years, so customers can expect both container host and container images to align with that cadence.

Features we've removed in this release

We're removing the following features and functionalities from the installed product image in Windows Server 2022. Applications or code that depend on these features won't function in this release unless you use an alternate method.

FEATURE	EXPLANATION
Internet Storage Name Service (iSNS) Server service	The iSNS Server service has now been removed from Windows Server 2022 after it was considered for removal in Windows Server, version 1709. You can still connect to iSNS servers or add iSCSI targets individually.

Features we're no longer developing

We're no longer actively developing these features and may remove them from a future update. Some features have been replaced with other features or functionality, while others are now available from different sources.

FEATURE	EXPLANATION
Guarded Fabric and Shielded Virtual Machines (VMs)	Windows Server and Azure Stack HCI are aligning with Azure to take advantage of continuing enhancements to Azure Confidential Computing and Azure Security Center. Having this alignment translates to more cloud security offerings being extended to customer data centers (on-premises). Microsoft will continue to provide support for these features, but there will be no further development. On client versions of Windows the Remote Server Administration Tools (RSAT): Shielded VM Tools feature will be removed.
Launching SConfig from a command prompt (CMD) window by running `sconfig.cmd`	Starting with Windows Server 2022, SConfig is launched **by default** when you sign in to a server running Server Core installation option. Moreover, PowerShell is now the default shell on Server Core. If you exit SConfig, you get to a regular interactive PowerShell window. Similarly, you can opt out from SConfig autolaunch. In this case, you will get a PowerShell window at sign-in. In either scenario, you can launch SConfig from PowerShell by simply running `SConfig`. If needed, you can launch the legacy command prompt (CMD) from PowerShell as well. But to simplify different transition options, we're going to remove `sconfig.cmd` from the next version of the operating system. If you need to start SConfig from a CMD window, you will have to launch PowerShell first.
Windows Deployment Services (WDS) boot.wim image deployment	The operating system deployment functionality of WDS is being partially deprecated. Workflows that rely on boot.wim from Windows Server 2022 installation media will show a non-blocking deprecation notice, but the workflows will otherwise not be impacted. Windows 11 workflows and workflows for future versions of Windows Server that rely on boot.wim from installation media will be blocked. Alternatives to WDS, such as Microsoft Endpoint Configuration Manager or the Microsoft Deployment Toolkit (MDT), provide a better, more flexible, and feature-rich experience for deploying Windows images. You are advised to move to one of these solutions instead. WDS PXE boot is not affected. You can still use WDS to PXE boot devices to custom boot images. You can also still run setup from a network share. Workflows that use custom boot.wim images, such as with Configuration Manager or MDT, will also not be impacted by this change.

Features removed or no longer developed starting with Windows Server, versions 1903 and 1909

12/17/2021 • 2 minutes to read • Edit Online

Each release of Windows Server adds new features and functionality; we also occasionally remove features and functionality, usually because we've added a better option. Here are the details about the features and functionalities that we removed in Windows Server, versions 1903 and 1909.

This list is subject to change in subsequent releases and might not include every affected feature or functionality.

Features we've removed in this release

We haven't removed any features in this release.

Features we're no longer developing

We're no longer actively developing these features and may remove them from a future update. Some features have been replaced with other features or functionality, while others are now available from different sources.

FEATURE	EXPLANATION
Hyper-V vSwitch on LBFO	In a future release, the Hyper-V vSwitch will no longer have the capability to be bound to an LBFO team. Instead, it must be bound viaSwitch Embedded Teaming(SET).
XDDM-based remote display driver	Starting with this release the Remote Desktop Services uses a Windows Display Driver Model (WDDM) based Indirect Display Driver (IDD) for a single session remote desktop. The support for Windows 2000 Display Driver Model (XDDM) based remote display drivers will be removed in a future release. Independent Software Vendors that use XDDM-based remote display driver should plan a migration to the WDDM driver model. For more information on implementing remote display indirect display driver check out Updates for IddCx versions 1.4 and later.
UCS log collection tool	The UCS log collection tool, while not explicitly intended for use with Windows Server, is nonetheless being replaced by the Feedback hub on Windows 10.
Key Storage Drive in Hyper-V	We're no longer working on the Key Storage Drive feature in Hyper-V. If you're using generation 1 VMs, check out Generation 1 VM Virtualization Security for information about options going forward. If you're creating new VMs use Generation 2 virtual machines with TPM devices for a more secure solution.
Trusted Platform Module (TPM) management console	The information previously available in the TPM management console is now available on the Device security page in the Windows Defender Security Center.

FEATURE	EXPLANATION
Host Guardian Service Active Directory attestation mode	We're no longer developing Host Guardian Service Active Directory attestation mode. Instead we've added a new attestation mode, host key attestation, that's far simpler and equally as compatible as Active Directory based attestation. This new mode provides equivalent functionality with a setup experience, simpler management and fewer infrastructure dependencies than the Active Directory attestation. Host key attestation has no additional hardware requirements beyond what Active Directory attestation required, so all existing systems will remain compatible with the new mode. See Deploy guarded hosts for more information about your attestation options.
OneSync service	The OneSync service synchronizes data for the Mail, Calendar, and People apps. We've added a sync engine to the Outlook app that provides the same synchronization.
Remote Differential Compression API support	Remote Differential Compression API support enabled synchronizing data with a remote source using compression technologies, which minimized the amount of data sent across the network.
WFP lightweight filter switch extension	The WFP lightweight filter switch extension enables developers to build simple network packet filtering extensions for the Hyper-V virtual switch. You can achieve the same functionality by creating a full filtering extension. As such, we'll be removing this extension in the future.

Features removed or no longer developed starting with Windows Server 2019

12/17/2021 • 3 minutes to read • Edit Online

Each release of Windows Server adds new features and functionality; we also occasionally remove features and functionality, usually because we've added a better option. Here are the details about the features and functionalities that we removed in Windows Server 2019.

TIP

- You can get early access to Windows Server builds by joining the Windows Insider program - this is a great way to test feature changes.

The list is subject to change and might not include every affected feature or functionality.

Features we've removed in this release

We're removing the following features and functionalities from the installed product image in Windows Server 2019. Applications or code that depend on these features won't function in this release unless you use an alternate method.

FEATURE	EXPLANATION
Business Scanning, also called Distributed Scan Management (DSM)	We're removing this secure scanning and scanner management capability - there are no devices that support this feature.
Print components - now optional component for Server Core installations	In previous releases of Windows Server, the print components were **disabled** by default in the Server Core installation option. We changed that in Windows Server 2016, enabling them by default. In Windows Server 2019, those print components are once again disabled by default for Server Core. If you need to enable the print components, you can do so by running the `Install-WindowsFeature Print-Server` cmdlet.
Remote Desktop Connection Broker and Remote Desktop Virtualization Host in a Server Core installation	Most Remote Desktop Services deployments have these roles co-located with the Remote Desktop Session Host (RDSH), which requires Server with Desktop Experience; to be consistent with RDSH we're changing these roles to also require Server with Desktop Experience. These RDS roles are no longer available for use in a Server Core installation. If you need to deploy these roles as part of your Remote Desktop infrastructure, you can install them on Windows Server with Desktop Experience. These roles are also included in the Desktop Experience installation option of Windows Server 2019.
RemoteFX 3D Video Adapter (vGPU)	We're developing new graphics acceleration options for virtualized environments. You can also use Discrete Device Assignment (DDA) as an alternative.

Features we're no longer developing

We're no longer actively developing these features and may remove them from a future update. Some features have been replaced with other features or functionality, while others are now available from different sources.

FEATURE	EXPLANATION
Key Storage Drive in Hyper-V	We're no longer working on the Key Storage Drive feature in Hyper-V. If you're using generation 1 virtual machines (VMs), check out Generation 1 VM Virtualization Security for information about options going forward. If you're creating new VMs, use Generation 2 virtual machines with TPM devices for a more secure solution.
Trusted Platform Module (TPM) management console	The information previously available in the TPM management console is now available on the **Device security** page in the Windows Defender Security Center.
Host Guardian Service Active Directory attestation mode	We're no longer developing Host Guardian Service Active Directory attestation mode - instead we've added a new attestation mode, host key attestation, that's far simpler and equally as compatible as Active Directory based attestation. This new mode provides equivalent functionality with a setup experience, simpler management and fewer infrastructure dependencies than the Active Directory attestation. Host key attestation has no additional hardware requirements beyond what Active Directory attestation required, so all existing systems will remain compatible with the new mode. See Deploy guarded hosts for more information about your attestation options.
OneSync service	The OneSync service synchronizes data for the Mail, Calendar, and People apps. We've added a sync engine to the Outlook app that provides the same synchronization.
Remote Differential Compression API support	Remote Differential Compression API support enabled synchronizing data with a remote source using compression technologies, which minimized the amount of data sent across the network.
WFP lightweight filter switch extension	The WFP lightweight filter switch extension enables developers to build simple network packet filtering extensions for the Hyper-V virtual switch. You can achieve the same functionality by creating a full filtering extension. As such, we'll be removing this extension in the future.

Features Removed or Deprecated in Windows Server 2016

12/17/2021 • 2 minutes to read • Edit Online

Each release of Windows Server adds new features and functionality; we also occasionally remove features and functionality, usually because we've added a better option. Here are the details about the features and functionalities that we removed in Windows Server 2016.

TIP

- You can get early access to Windows Server builds by joining the Windows Insider Program for Business - this is a great way to test feature changes.

The list is subject to change and might not include every affected feature or functionality.

Features we've removed in this release

We're removing the following features and functionalities from the installed product image in Windows Server 2016. Applications or code that depend on these features won't function in this release unless you use an alternate method.

NOTE

If you are moving to Windows Server 2016 from a server release prior to Windows Server 2012 R2 or Windows Server 2012, you should also review Features Removed or Deprecated in Windows Server 2012 R2 and Features Removed or Deprecated in Windows Server 2012.

FEATURE	EXPLANATION
Share and Storage Management snap-in for Microsoft Management Console	If the computer you want to manage is running an operating system older than Windows Server 2016, connect to it with Remote Desktop and use the local version of the Share and Storage Management snap-in. On a computer running Windows 8.1 or earlier, use the Share and Storage Management snap-in from RSAT to view the computer you want to manage. Use Hyper-V on a client computer to run a virtual machine running Windows 7, Windows 8, or Windows 8.1 that has the Share and Storage Management snap-in in RSAT.
Journal.dll	The file `Journal.dll` is removed from Windows Server 2016. There is no replacement.
Security Configuration Wizard	The Security Configuration Wizard is removed. Instead, features are secured by default. If you need to control specific security settings, you can use either Group Policy or Microsoft Security Compliance Manager.

FEATURE	EXPLANATION
SQM	The opt-in components that manage participation in the Customer Experience Improvement Program have been removed.

Windows Update	The **wuauclt.exe /detectnow** command has been removed and is no longer supported. To trigger a scan for updates, run these PowerShell commands:
	`$AutoUpdates = New-Object -ComObject "Microsoft.Update.AutoUpdate"`
	`$AutoUpdates.DetectNow()`

Features we're no longer developing

We're no longer actively developing these features and may remove them from a future update. Some features have been replaced with other features or functionality, while others are now available from different sources.

FEATURE	EXPLANATION
Configuration tools	`scregedit.exe` is deprecated. If you have scripts that depend on scregedit.exe, adjust them to use reg.exe or PowerShell methods.
Sconfig.exe	Use Sconfig.cmd instead.
NetCfg custom APIs	Installation of PrintProvider, NetClient, and ISDN using NetCfg custom APIs is deprecated.
Remote management	WinRM.vbs is deprecated. Instead, use functionality in the WinRM provider of PowerShell.
SMB 2+ over NetBT	SMB 2+ over NetBT is deprecated. Instead, implement SMB over TCP or RDMA.

Windows Server release information

12/17/2021 • 2 minutes to read • Edit Online

Windows Server is moving to the Long-Term Servicing Channel (LTSC) as our primary release channel. Current Semi-Annual Channel (SAC) releases will continue through their mainstream support end dates, which are May 10, 2022 for Windows Server, version 20H2 and December 14, 2021 for Windows Server, version 2004.

The focus on container and microservice innovation previously released in the Semi-Annual Channel will now continue with Azure Kubernetes Service (AKS), AKS on Azure Stack HCI, and other platform improvements made in collaboration with the Kubernetes community. And with the Long-Term Servicing Channel, a major new version of Windows Server will be released every 2-3 years, so you can expect both container host and container images to align with that cadence.

Windows Server current versions by servicing option

(All dates are listed in ISO 8601 format: YYYY-MM-DD)

WINDOWS SERVER RELEASE	SERVICING OPTION	EDITIONS	AVAILABILITY	BUILD	MAINSTREAM SUPPORT END DATE	EXTENDED SUPPORT END DATE
Windows Server 2022	Long-Term Servicing Channel (LTSC)	Datacenter, Standard	2021-08-18	20348.169	2026-10-13	2031-10-14
Windows Server, version 20H2	Semi-Annual Channel	Datacenter Core, Standard Core	2020-10-20	19042.508	2022-05-10	Not applicable
Windows Server, version 2004	Semi-Annual Channel	Datacenter Core, Standard Core	2020-05-27	19041.264	End of servicing	Not applicable
Windows Server, version 1909	Semi-Annual Channel	Datacenter Core, Standard Core	2019-11-12	18363.418	End of servicing	Not applicable
Windows Server 2019 (version 1809)	Long-Term Servicing Channel (LTSC)	Datacenter, Essentials, Standard	2018-11-13	17763.107	2024-01-09	2029-01-09
Windows Server 2016 (version 1607)	Long-Term Servicing Channel (LTSC)	Datacenter, Essentials, Standard	2016-10-15	14393.0	2022-01-11	2027-01-11

NOTE

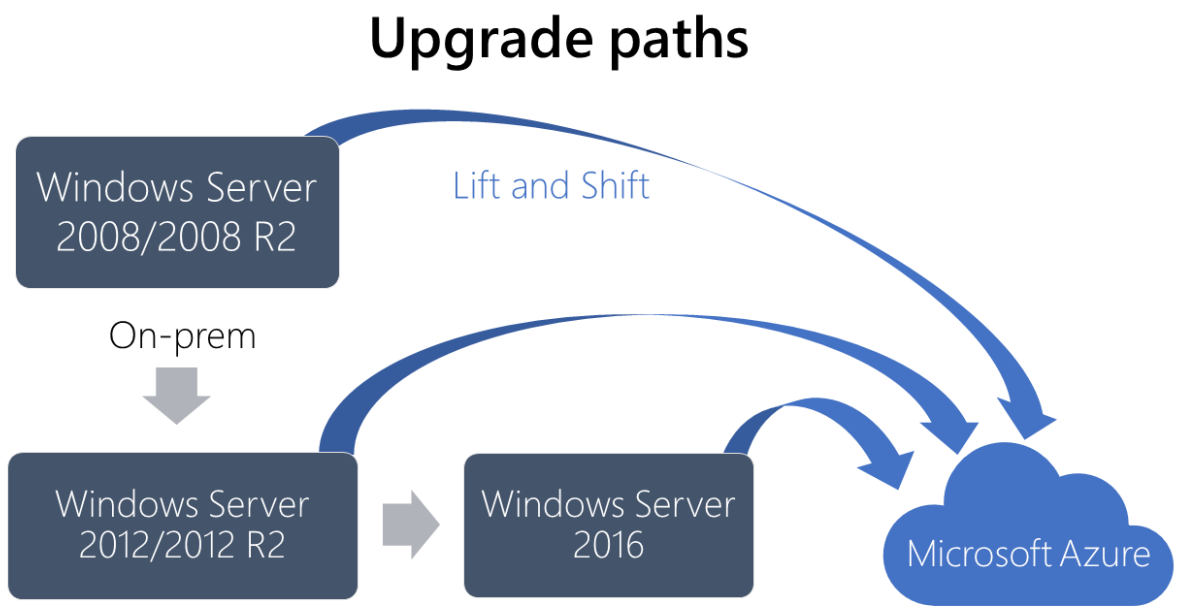
Windows Server, version 1803 and later are governed by the Modern Lifecycle Policy. See the Windows Lifecycle FAQ and Comparison of servicing channels for details regarding servicing requirements and other important information.

Extended Security Updates for Windows Server

12/17/2021 • 2 minutes to read • Edit Online

Extended support for Windows Server 2008 and Windows Server 2008 R2 ended on January 14, 2020, and extended support for Windows Server 2012 and Windows Server 2012 R2 will be ending on October 10, 2023. There are two modernization paths available: on-premises upgrade, or migration by rehosting in Azure. **If you rehost in Azure, you can migrate your existing Server images free of charge**.

After you understand these options, see how to use Windows Server Extended Security Updates.



On-premises upgrade

If you need to keep your servers on-premises, and you are running Windows Server 2008 or Windows Server 2008 R2, you will need to upgrade to Windows Server 2012/2012 R2 before you can upgrade to Windows Server 2016. There is no direct upgrade path from Windows Server 2008 R2 to Windows Server 2016 or later. Instead, upgrade first to Windows Server 2012 R2, and then upgrade to Windows Server 2016. As you upgrade, you still have the option to migrate to Azure by rehosting. See supported upgrade paths for Windows Server, for more information about your on-premises upgrade options.

If you are running Windows Server 2003, you will need to upgrade to Windows Server 2008. See upgrade paths for Windows Server 2008 for more information about your on-premises upgrade options.

Migrate to Azure

You can migrate your on-premises Windows Server 2008, Windows Server 2008 R2, Windows Server 2012, and Windows Server 2012 R2 servers to Azure, where you can continue to run them on virtual machines. In Azure you'll stay compliant, become more secure, and add cloud innovation to your work. The benefits of migrating to Azure include:

- Security updates in Azure.
- Get three more years of Windows Server critical and important security updates, included at no additional charge.

- No-charge upgrades in Azure.
- Adopt more cloud services as you are ready.
- By migrating SQL Server to Azure Managed Instances or VMs, you get three more years of Windows Server critical security updates, included at no additional charge.
- Benefit from Azure Hybrid Benefit, which means you can leverage existing SQL Server and Windows Server licenses for cloud savings unique to Azure.

To get started migrating, learn how to upload a generalized VHD and use it to create new VMs in Azure, or use Shared Image Galleries in Azure.

To help you understand how to analyze existing IT resources, assess what you have, and identify the benefits of moving specific services and applications to the cloud or keeping workloads on-premises and upgrading to the latest version of Windows Server, see Migration Guide for Windows Server.

Upgrade SQL Server 2008/2008 R2 in parallel with your Windows Servers

If you are running SQL Server 2008 or SQL Server 2008 R2, you can also upgrade to SQL Server 2016 or SQL Server 2017.

Install, upgrade, or migrate to Windows Server

12/17/2021 • 2 minutes to read • Edit Online

Is it time to move to a newer version of Windows Server? Depending on what you're running now, you have several options to get there.

> **IMPORTANT**
>
> Extended support for Windows Server 2008 R2 and Windows Server 2008 ended in January 2020. Extended Security Updates (ESU) are available, with one option to migrate your on-premises servers to Azure, where you can continue to run them on virtual machines. To find out more, see Extended Security Updates overview.

> **TIP**
>
> To download Windows Server 2022, see Windows Server Evaluations.

Clean install

Clean install is simplest way to install Windows Server, where you install on a blank server or overwrite an existing operating system, but you will need to back up your data first and plan to reinstall your applications. There are a few things to be aware of, such as hardware requirements, so be sure to check the details for Windows Server.

In-place upgrade

In-place upgrade enables you to keep the same hardware and all the server roles you have set up without wiping and reinstalling the operating system, by which you go from an older operating system to a newer one, keeping your settings, server roles and features, and data intact. For example, if your server is running Windows Server 2019, you can upgrade it to Windows Server 2022. However, not every older operating system has a pathway to every newer one and some roles or features don't support this or need you to take extra steps. In-place upgrade works best in virtual machines where specific OEM hardware drivers are not needed for a successful upgrade.

For step-by-step guidance and more information on upgrading, review the Windows Server upgrade content and Upgrade and migrate roles and features in Windows Server.

Cluster Operating System rolling upgrade

Cluster Operating System (OS) rolling upgrade gives an administrator the ability to upgrade the operating system of the cluster nodes without stopping the Hyper-V or the Scale-Out File Server workloads. For example, if nodes in your cluster are running Windows Server 2019 you can install Windows Server 2022 on them avoiding downtime to the cluster, which would otherwise impact Service Level Agreements. This feature is discussed in more detail at Cluster OS rolling upgrade.

Migration

Migration of Windows Server is when you move one role or feature at a time from a source computer that is running Windows Server to another destination computer that is running Windows Server, either the same or a newer version. For these purposes, migration is defined as moving one role or feature and its data to a different

computer, not upgrading the feature on the same computer.

License conversion

License conversion enables you to convert a particular edition of the release to another edition of the same release in a single step with a simple command and the appropriate license key for some Windows Server releases. For example, if your server is running Windows Server 2022 Standard, you can convert it to Windows Server 2022 Datacenter. Keep in mind that while you can move up from Windows Server 2022 Standard to Windows Server 2022 Datacenter, you are unable to reverse the process and go from Datacenter edition to Standard edition. In some releases of Windows Server, you can also freely convert between OEM, volume-licensed, and retail versions with the same command and the appropriate key.

Server Core vs Server with Desktop Experience install options

12/17/2021 • 2 minutes to read • Edit Online

When you install Windows Server using the setup wizard, you can choose between Server Core or Server with Desktop Experience install options. With Server Core, the standard graphical user interface (the Desktop Experience) is not installed; you manage the server from the command line using PowerShell, the Server Configuration tool (SConfig), or by remote methods. Server with Desktop Experience installs the standard graphical user interface and all tools, including client experience features.

We recommend that you choose the Server Core install option unless you have a particular need for the extra user interface elements and graphical management tools that are included in the Server with Desktop Experience install option.

The setup wizard lists the install options below. In this list, editions without **Desktop Experience** are the Server Core install options:

- Windows Server Standard
- Windows Server Standard with Desktop Experience
- Windows Server Datacenter
- Windows Server Datacenter with Desktop Experience

> **NOTE**
> Unlike some previous releases of Windows Server, you cannot convert between Server Core and Server with Desktop Experience after installation. You will need to do a clean installation if you install later decide to use a different option.

Differences

There are some key differences between Server Core and Server with Desktop Experience:

COMPONENT	SERVER CORE	SERVER WITH DESKTOP EXPERIENCE
User interface	Minimal, command line driven (PowerShell, SConfig, cmd)	Standard Windows graphical user interface
Disk space	Smaller requirement	Larger requirement
Install, configure, uninstall server roles locally	PowerShell	Server Manager or PowerShell

COMPONENT	SERVER CORE	SERVER WITH DESKTOP EXPERIENCE
Roles and Features	Some roles and features are not available. For more information, see Roles, Role Services, and Features not in Windows Server - Server Core. Some of the features from Server with Desktop Experience for application compatibility can be installed with the App Compatibility Feature on Demand (FOD).	All roles and features are available, including those for application compatibility.
Remote management	Yes, can be managed remotely using GUI tools, such as Windows Admin Center, Remote Server Administration Tools (RSAT), or Server Manager, or by PowerShell.	Yes, can be managed remotely using GUI tools, such as Windows Admin Center, Remote Server Administration Tools (RSAT), or Server Manager, or by PowerShell.
Potential attack surface	Greatly reduced attack surface	No reduction
Microsoft Management Console	Not installed - can be installed with the App Compatibility Feature on Demand (FOD).	Installed

NOTE

For RSAT, you must use the version included with Windows 10 or later.

Upgrade and migrate roles and features in Windows Server

12/17/2021 • 4 minutes to read • Edit Online

You can update roles and features to later versions of Windows Server by migrating to a new server, or many also support in-place upgrade where you install the new version of Windows Server over the top of the current one. This article contains links to migration guides as well a table with migration and in-place upgrade information to help you decide which method to use.

You can migrate many roles and features by using Windows Server Migration Tools, a feature built in to Windows Server for migrating roles and features, whereas file servers and storage can be migrated using Storage Migration Service.

The migration guides support migrations of specified roles and features from one server to another (not in-place upgrades). Unless otherwise noted in the guides, migrations are supported between physical and virtual computers, and between installation options of Windows Server with either Server with Desktop Experience or Server Core.

> **IMPORTANT**
>
> Before you begin migrating roles and features, verify that both source and destination servers are running the most current updates that are available for their operating systems.
>
> Whenever you migrate or upgrade to any version of Windows Server, you should review and understand the support lifecycle policy and time frame for that version and plan accordingly. You can search for the lifecycle information for the particular Windows Server release that you are interested in.

Windows Server Migration Tools

Windows Server Migration Tools enables you to migrate server roles, features, operating system settings, and other data and shares to servers, including later versions of Windows Server. It is a feature of Windows Server and so it is easily installed using the Add Roles and Features wizard, or PowerShell. Learn more about how to install, use, and remove Windows Server Migration Tools.

> **NOTE**
>
> Cross-subnet migrations using Windows Server Migration Tools is available with Windows Server 2012 and later releases. Previous versions of Windows Server Migration Tools only support migrations in the same subnet.

Migration guides

Below you can find links to migration guides for specific Windows Roles and Features.

Active Directory

- Active Directory Certificate Services Migration Guide for Windows Server 2012 R2
- Active Directory Certificate Services Migration Guide for Windows Server 2008 R2
- Migrate Active Directory Federation Services Role Service to Windows Server 2012 R2
- Migrate Active Directory Federation Services Role Services to Windows Server 2012
- Active Directory Rights Management Services Migration and Upgrade Guide

- Upgrade Domain Controllers to Windows Server 2012 R2 and Windows Server 2012
- Active Directory Domain Services and Domain Name System (DNS) Server Migration Guide for Windows Server 2008 R2

BranchCache

- BranchCache Migration Guide

DHCP

- Migrate DHCP Server to Windows Server 2012 R2
- Dynamic Host Configuration Protocol (DHCP) Server Migration Guide for Windows Server 2008 R2

Failover Clustering

- Migrate Cluster Roles to Windows Server 2012 R2
- Migrate Clustered Services and Applications to Windows Server 2012

File and Storage Services

- Storage Migration Service
- Migrate File and Storage Services to Windows Server 2012 R2

Hyper-V

- Migrate Hyper-V to Windows Server 2012 R2 from Windows Server 2012
- Migrate Hyper-V to Windows Server 2012 from Windows Server 2008 R2

Network Policy Server

- Migrate Network Policy Server to Windows Server 2012
- Migrate Health Registration Authority to Windows Server 2012

Print and Document Services

- Migrate Print and Document Services to Windows Server 2012

Remote Access

- Migrate Remote Access to Windows Server 2012

Remote Desktop Services

- Migrate Remote Desktop Services
- Migrate Remote Desktop Services to Windows Server 2012 R2
- Migrate MultiPoint Services

Routing and Remote Access

- RRAS Migration Guide

Web Server (IIS)

- Web Server (IIS)

Windows Server Update Services

- Migrate Windows Server Update Services to Windows Server 2012 R2

Other Windows migration guides

- Local User and Group Migration Guide
- IP Configuration Migration Guide

Upgrade and migration matrix

SERVER ROLE	UPGRADEABLE IN-PLACE?	MIGRATION SUPPORTED?	CAN MIGRATION BE COMPLETED WITHOUT DOWNTIME?
Active Directory Certificate Services	Yes	Yes	No
Active Directory Domain Services	Yes	Yes	Yes
Active Directory Federation Services	No	Yes	No (new nodes need to be added to the farm)
Active Directory Lightweight Directory Services	Yes	Yes	Yes
Active Directory Rights Management Services	Yes	Yes	No
DHCP Server	Yes	Yes	Yes
DNS Server	Yes	Yes	No
Failover Clustering	Yes with Cluster OS Rolling Upgrade process (Windows Server 2012 R2 and later) or when the server is removed by the cluster for upgrade and then added to a different cluster.	Yes	Yes for Failover Clusters with Hyper-V VMs or Failover Clusters running the Scale-out File Server role. See Cluster OS Rolling Upgrade (Windows Server 2012 R2 and later).
File and Storage Services	Yes	Varies by subfeature	No
Hyper-V	Yes with Cluster OS Rolling Upgrade process (Windows Server 2012 R2 and later)	Yes	Yes for Failover Clusters with Hyper-V VMs or Failover Clusters running the Scale-out File Server role. See Cluster OS Rolling Upgrade (Windows Server 2012 R2 and later).
Print and Fax Services	No	Yes (using Printbrm.exe)	No
Remote Desktop Services	Yes, for all subroles, but mixed mode farm is not supported	Yes	No
Web Server (IIS)	Yes	Yes	No
Windows Server Essentials Experience	Yes	Yes	No
Windows Server Update Services	Yes	Yes	No

SERVER ROLE	UPGRADEABLE IN-PLACE?	MIGRATION SUPPORTED?	CAN MIGRATION BE COMPLETED WITHOUT DOWNTIME?
Work Folders	Yes	Yes	Yes with Cluster OS Rolling Upgrade process (Windows Server 2012 R2 and later).

Upgrade and conversion options for Windows Server

12/17/2021 • 5 minutes to read • Edit Online

You can upgrade or convert installations of Windows Server to newer versions, different editions, or switch between licensing options, such as evaluation, retail, and volume licensed. This article helps explain what the options are to help with your planning.

The process of upgrading or converting installations of Windows Server might vary greatly depending on which version and edition you have installed, how it is licensed, and the pathway you take. We use different terms to distinguish between actions, any of which could be involved in a deployment of Windows Server: clean install, in-place upgrade, cluster operating system (OS) rolling upgrade, migration, and license conversion. You can learn more about these terms at Install, upgrade, or migrate.

Upgrading licensed versions of Windows Server

Below are general guidelines for in-place upgrade paths where Windows Server is **already licensed** (that is, not evaluation):

- Upgrades from 32-bit to 64-bit architectures are not supported. All releases of Windows Server since Windows Server 2016 are 64-bit only.
- Upgrades from one language to another are not supported.
- If the server is an Active Directory domain controller, you cannot convert it to a retail version. See Upgrade Domain Controllers to Windows Server 2012 R2 and Windows Server 2012 for important information.
- Upgrades from pre-release versions (previews) of Windows Server are not supported. Perform a clean installation of Windows Server.
- Upgrades that switch from a Server Core installation to a Server with Desktop Experience installation (or vice versa) are not supported.
- Upgrades from a previous Windows Server installation to an evaluation copy of Windows Server are not supported. Evaluation versions should be installed as a clean installation.
- You can only change from Standard edition to Datacenter edition when upgrading. Changing from Datacenter edition to Standard edition is not supported.

> **IMPORTANT**
> If your server uses NIC Teaming, disable NIC Teaming prior to upgrade, and then re-enable it after upgrade is complete. See NIC Teaming Overview for details.

Converting an evaluation version to a retail version

You can convert the evaluation version of Windows Server to the retail version. If you have installed the evaluation of Standard edition, you can convert it to the retail version of either the Standard edition or Datacenter edition. Similarly, if you have installed the evaluation of the Datacenter edition, you can only convert it to the retail version of the Datacenter edition.

If you haven't already activated Windows, the bottom right-hand corner of the desktop shows the time remaining in the evaluation period.

> **IMPORTANT**
>
> For releases of Windows Server 2016 prior to 14393.0.161119-1705.RS1_REFRESH, you can only convert from evaluation to retail when Windows Server has been installed with the Server with Desktop Experience installation option (not Server Core). Starting with version 14393.0.161119-1705.RS1_REFRESH and later releases, you can convert evaluation editions to retail regardless of the installation option used.

> **NOTE**
>
> Before you attempt to convert from evaluation to retail, verify that your server is actually running an evaluation version. To do this, launch an elevated command prompt and run the command `slmgr.vbs /dlv`; evaluation versions will include **EVAL** in the output.

Windows Server Standard or Datacenter

If the server is running an evaluation version of Windows Server Standard edition or Windows Server Datacenter edition, you can convert it to a retail version as follows:

1. From an elevated command prompt or PowerShell session, run the following command to save the Microsoft Software License Terms for Windows Server, which you can then review:

   ```
   DISM /online /Set-Edition:ServerDatacenter /GetEula:C:\eula.rtf
   ```

2. Determine the current edition name by running the command below. The output is an abbreviated form of the edition name, for example Windows Server Datacenter edition is **ServerDatacenter**:

   ```
   DISM /online /Get-CurrentEdition
   ```

3. Verify which editions the current installation can be converted to by running the command below. The evaluation version of Windows Server Standard can be converted to the retail version of either the Standard or Datacenter editions of Windows Server, whereas the evaluation version of Windows Server Datacenter can only be converted to the retail version Windows Server Datacenter:

   ```
   DISM /online /Get-TargetEditions
   ```

4. Make note of the target edition name you want to convert to, and enter this and your retail product key in the command below. This process requires you to accept the Microsoft Software License Terms for Windows Server you saved previously.

   > **TIP**
   >
   > You can convert from the evaluation version of Windows Server Standard to the retail version of Windows Server Datacenter in one step by using the appropriate product key and edition ID.

   ```
   DISM /online /Set-Edition:<edition ID> /ProductKey:XXXXX-XXXXX-XXXXX-XXXXX-XXXXX /AcceptEula
   ```

   For example:

   ```
   DISM /online /Set-Edition:ServerDatacenter /ProductKey:ABCDE-12345-ABCDE-12345-ABCDE /AcceptEula
   ```

TIP

For more information about Dism.exe, see DISM Command-line options.

IMPORTANT

If the server is an Active Directory domain controller, you cannot convert it to a retail version. In this case, install an additional domain controller on a server that runs a retail version, migrate any FSMO roles held, and remove Active Directory Domain Services (AD DS) from the domain controller that runs on the evaluation version. For more information, see Upgrade Domain Controllers to Windows Server 2012 R2 and Windows Server 2012.

Windows Server Essentials

If the server is running Windows Server Essentials, you can convert it to the full retail version by entering a retail, volume license, or OEM key by launching an elevated command prompt and entering it as part of the following command:

```
slmgr.vbs /ipk XXXXX-XXXXX-XXXXX-XXXXX-XXXXX
```

Converting Windows Server Standard edition to Datacenter edition

At any time after installing Windows Server, you can convert Windows Server Standard edition to Datacenter edition. You can also run `setup.exe` from the installation media to upgrade or repair the installation (sometimes called in-place repair). If you run `setup.exe` to upgrade or repair in-place on any edition of Windows Server, the result will be the same edition you started with.

You can convert the Standard edition of Windows Server to the Datacenter edition as follows:

1. Determine that Windows Server Standard is the current edition name by running the command below. The output is an abbreviated form of the edition name, for example Windows Server Standard edition is **ServerStandard**:

   ```
   DISM /online /Get-CurrentEdition
   ```

2. Verify that Windows Server Datacenter is a valid option to convert to by running the following command:

   ```
   DISM /online /Get-TargetEditions
   ```

3. Enter **ServerDatacenter** and your retail product key in the command below:

   ```
   DISM /online /Set-Edition:ServerDatacenter /ProductKey:XXXXX-XXXXX-XXXXX-XXXXX-XXXXX /AcceptEula
   ```

Converting between retail, volume-licensed, and OEM licenses

At any time after installing Windows Server, you can freely convert between a retail license, a volume-licensed license, or an OEM license. The edition (Standard or Datacenter) remains the same during this conversion. If you are starting with an evaluation version, convert it to the retail version first, then you can convert between the versions.

To do this, run the following command from an elevated command prompt, including providing your volume-license, retail, or OEM product key:

```
slmgr.vbs /ipk XXXXX-XXXXX-XXXXX-XXXXX-XXXXX
```

Automatic Virtual Machine Activation in Windows Server

12/17/2021 • 4 minutes to read • Edit Online

Automatic Virtual Machine Activation (AVMA) acts as a proof-of-purchase mechanism, helping to ensure that Windows products are used in accordance with the Product Use Rights and Microsoft Software License Terms.

AVMA lets you activate Windows Server virtual machines (VMs) on Windows Server Hyper-V host that is properly activated, even in disconnected environments. AVMA binds the virtual machine activation to the licensed virtualization host and activates the virtual machine when it starts up. You can get real-time reporting on usage and historical data on the license state of the virtual machine when using AVMA. Reporting and tracking data is available on the virtualization host.

Practical applications

On virtualization hosts, AVMA offers several benefits.

Server data center managers can use AVMA to do the following:

- Activate virtual machines in remote locations
- Activate virtual machines with or without an internet connection
- Track virtual machine usage and licenses from the virtualization host, without requiring any access rights on the virtualized systems

Service Provider License Agreement (SPLA) partners and other hosting providers do not have to share product keys with tenants or access a tenant's virtual machine to activate it. Virtual machine activation is transparent to the tenant when AVMA is used. Hosting providers can use the server logs to verify license compliance and to track client usage history.

System requirements

The virtualization host that will run virtual machines needs to be activated. Keys can be obtained through the Volume Licensing Service Center or your OEM provider.

> **NOTE**
> In a failover cluster, each virtualization host in the cluster must be activated for VMs to stay activated regardless of which server they run on.

AVMA requires Windows Server Datacenter edition with the Hyper-V host role installed. The operating system version of the Hyper-V host determines which versions of operating system can be activated in a virtual machine. Here are the guests that the different version hosts can activate:

SERVER HOST VERSION	WINDOWS SERVER 2022 GUEST VM	WINDOWS SERVER 2019 GUEST VM	WINDOWS SERVER 2016 GUEST VM	WINDOWS SERVER 2012 R2 GUEST VM
Windows Server 2022	X	X	X	X

SERVER HOST VERSION	WINDOWS SERVER 2022 GUEST VM	WINDOWS SERVER 2019 GUEST VM	WINDOWS SERVER 2016 GUEST VM	WINDOWS SERVER 2012 R2 GUEST VM
Windows Server 2019		X	X	X
Windows Server 2016			X	X
Windows Server 2012 R2				X

> **NOTE**
>
> The table above applies all editions (Datacenter, Standard, or Essentials).
>
> AVMA does not work with other server virtualization technologies.

How to implement AVMA

To activate VMs with AVMA, you use a generic AVMA key (detailed in the AVMA keys section below) that corresponds to the version of Windows Server that you want to activate. To create a VM and activate it with an AVMA key, do the following:

1. On the server that will host virtual machines, install and configure the Microsoft Hyper-V Server role. For more information, see Install Hyper-V Server. Ensure that the server is successfully activated.
2. Create a virtual machine and install a supported Windows Server operating system on it.

   > **IMPORTANT**
   >
   > The Data Exchange integration service (also known as Key-Value Pair Exchange) must be enabled in the VM settings for AVMA to work. It is enabled by default for new VMs.

3. Once Windows Server is installed on the VM, you install the AVMA key in the VM. From PowerShell or an elevated Command Prompt, run the following command:

   ```
   slmgr /ipk <AVMA_key>
   ```

The virtual machine will automatically activate, providing the virtualization host itself is activated.

> **TIP**
>
> You can also add the AVMA keys in any Unattend setup file.

AVMA keys

The following AVMA keys can be used for Windows Server 2022:

EDITION	AVMA KEY
Datacenter	W3GNR-8DDXR-2TFRP-H8P33-DV9BG

EDITION	AVMA KEY
Standard	YDFWN-MJ9JR-3DYRK-FXXRW-78VHK

The following AVMA keys can be used for Windows Server 2019:

EDITION	AVMA KEY
Datacenter	H3RNG-8C32Q-Q8FRX-6TDXV-WMBMW
Standard	TNK62-RXVTB-4P47B-2D623-4GF74
Essentials	2CTP7-NHT64-BP62M-FV6GG-HFV28

The following AVMA keys can be used for Windows Server, versions 1909, 1903, and 1809:

EDITION	AVMA KEY
Datacenter	H3RNG-8C32Q-Q8FRX-6TDXV-WMBMW
Standard	TNK62-RXVTB-4P47B-2D623-4GF74

The following AVMA keys can be used for Windows Server, version 1803 and 1709:

EDITION	AVMA KEY
Datacenter	TMJ3Y-NTRTM-FJYXT-T22BY-CWG3J
Standard	C3RCX-M6NRP-6CXC9-TW2F2-4RHYD

The following AVMA keys can be used for Windows Server 2016:

EDITION	AVMA KEY
Datacenter	TMJ3Y-NTRTM-FJYXT-T22BY-CWG3J
Standard	C3RCX-M6NRP-6CXC9-TW2F2-4RHYD
Essentials	B4YNW-62DX9-W8V6M-82649-MHBKQ

The following AVMA keys can be used for Windows Server 2012 R2:

EDITION	AVMA KEY
Datacenter	Y4TGP-NPTV9-HTC2H-7MGQ3-DV4TW
Standard	DBGBW-NPF86-BJVTX-K3WKJ-MTB6V
Essentials	K2XGM-NMBT3-2R6Q8-WF2FK-P36R2

Reporting and tracking

The Key-Value Pair (KVP) exchange between the virtualization host and the VM provides real-time tracking data

for the guest operating systems, including activation information. This activation information is stored in the Windows registry of the virtual machine. Historical data about AVMA requests is logged in Event Viewer on the virtualization host.

See Data Exchange: Using key-value pairs to share information between the host and guest on Hyper-V for more information about KVP.

> **NOTE**
> KVP data is not secured. It can be modified and is not monitored for changes.

> **IMPORTANT**
> KVP data should be removed if the AVMA key is replaced with another product key (retail, OEM, or volume licensing key).

Since the AVMA activation process is transparent, error messages are not displayed. However, AVMA requests are also logged on the virtualization host in Event Viewer in the Application log with Event ID 12310, and on the virtual machine with Event ID 12309. The following events are captured on the virtual machines:

NOTIFICATION	DESCRIPTION
AVMA Success	The virtual machine was activated.
Invalid Host	The virtualization host is unresponsive. This can happen when the server is not running a supported version of Windows.
Invalid Data	This usually results from a failure in communication between the virtualization host and the virtual machine, often caused by corruption, encryption, or data mismatch.
Activation Denied	The virtualization host could not activate the guest operating system because the AVMA ID did not match.

Key Management Services (KMS) activation planning

12/17/2021 • 7 minutes to read • Edit Online

The following information outlines initial planning considerations that you need to review for Key Management Services (KMS) activation.

KMS uses a client-server model to active clients and is used for volume activation. KMS clients connect to a KMS server, called the KMS host, for activation. The KMS host must reside on your local network.

KMS hosts do not need to be dedicated servers, and KMS can be cohosted with other services. You can run a KMS host on any physical or virtual system that is running a supported Windows Server or Windows client operating system. A KMS host running on a Windows Server operating system can activate computers running both server and client operating systems, however a KMS host running on a Windows client operating system can only activate computers also running client operating systems.

To use KMS, a KMS host needs a key that activates, or authenticates, the KMS host with Microsoft. This key is sometimes referred to as the KMS host key, but it is formally known as a Microsoft Customer Specific Volume License Key (CSVLK). You can get this key from the Product Keys section of the Volume Licensing Service Center for the following agreements: Open, Open Value, Select, Enterprise, and Services Provider License. You can also get assistance by contacting your local Microsoft Activation Center.

Operational requirements

KMS can activate physical and virtual computers, but to qualify for KMS activation, a network must have a minimum number of computers (called the activation threshold). KMS clients activate only after this threshold is met. To ensure that the activation threshold is met, a KMS host counts the number of computers that are requesting activation on the network.

KMS hosts count the most recent connections. When a client or server contacts the KMS host, the host adds the machine ID to its count and then returns the current count value in its response. The client or server will activate if the count is high enough. Clients will activate if the count is 25 or higher. Servers and volume editions of Microsoft Office products will activate if the count is five or greater. The KMS only counts unique connections from the past 30 days, and only stores the 50 most recent contacts.

KMS activations are valid for 180 days, a period known as the activation validity interval. KMS clients must renew their activation by connecting to the KMS host at least once every 180 days to stay activated. By default, KMS client computers attempt to renew their activation every seven days. After a client's activation is renewed, the activation validity interval begins again.

A single KMS host can support an unlimited number of KMS clients. If you have more than 50 clients, we recommend that you have at least two KMS hosts in case one of your KMS hosts becomes unavailable. Most organizations can operate with as few as two KMS hosts for their entire infrastructure.

After the first KMS host is activated, the CSVLK that is used on the first host can be used to activate up to five more KMS hosts on your network for a total of six. After a KMS host is activated, administrators can reactivate the same host up to nine times with the same key.

If your organization needs more than six KMS hosts, you can request additional activations for your organization's CSVLK - for example, if you have 10 physical locations under one volume licensing agreement and you want each location to have a local KMS host. To request this exception, please contact your local

Microsoft Activation Center.

Computers that are running volume licensing editions of Windows Server and Windows client are, by default, KMS clients with no extra configuration needed.

If you are converting a computer from a KMS host, MAK, or retail edition of Windows to a KMS client, you will need to install the applicable KMS client setup key. For more information, see KMS client setup keys.

Network requirements

KMS activation requires TCP/IP connectivity. KMS hosts and clients are configured by default to use Domain Name System (DNS). KMS hosts use DNS dynamic updates to automatically publish the information that KMS clients need to find and connect to them. You can accept these default settings, or if you have special network and security configuration requirements, you can manually configure KMS hosts and clients.

By default, a KMS host is configured to use TCP on port 1688.

Activation versions

The following table summarizes KMS host and client versions for networks that include Windows Server and Windows client devices.

> **IMPORTANT**
>
> - Windows Updates might be required on the KMS server to support activation of newer clients. If you receive activation errors, check that you have the appropriate updates listed below this table.

CSVLK GROUP	CSVLK CAN BE HOSTED ON	WINDOWS EDITIONS ACTIVATED BY THIS KMS HOST

CSVLK GROUP	CSVLK CAN BE HOSTED ON	WINDOWS EDITIONS ACTIVATED BY THIS KMS HOST
Volume License for Windows Server 2022	• Windows Server 2022 • Windows Server 2019 • Windows Server 2016	• Windows Server 2022 (all editions) • Windows Server Semi-Annual Channel • Windows Server 2019 (all editions) • Windows Server 2016 (all editions) • Windows 11 Enterprise/Enterprise N • Windows 11 Professional/Professional N • Windows 11 Professional for Workstations/Professional N for Workstations • Windows 11 for Education/Education N • Windows 10 Enterprise LTSC/LTSC N/LTSB • Windows 10 Enterprise/Enterprise N • Windows 10 Professional/Professional N • Windows 10 Professional for Workstations/Professional N for Workstations • Windows 10 for Education/Education N • Windows Server 2012 R2 (all editions) • Windows 8.1 Professional • Windows 8.1 Enterprise • Windows Server 2012 (all editions) • Windows Server 2008 R2 (all editions) • Windows Server 2008 (all editions) • Windows 7 Professional • Windows 7 Enterprise

CSVLK GROUP	CSVLK CAN BE HOSTED ON	WINDOWS EDITIONS ACTIVATED BY THIS KMS HOST
Volume License for Windows Server 2019	• Windows Server 2019 • Windows Server 2016 • Windows Server 2012 R2	• Windows Server Semi-Annual Channel • Windows Server 2019 (all editions) • Windows Server 2016 (all editions) • Windows 10 Enterprise LTSC/LTSC N/LTSB • Windows 10 Enterprise/Enterprise N • Windows 10 Professional/Professional N • Windows 10 Professional for Workstations/Professional N for Workstations • Windows 10 for Education/Education N • Windows Server 2012 R2 (all editions) • Windows 8.1 Professional • Windows 8.1 Enterprise • Windows Server 2012 (all editions) • Windows Server 2008 R2 (all editions) • Windows Server 2008 (all editions) • Windows 7 Professional • Windows 7 Enterprise

CSVLK GROUP	CSVLK CAN BE HOSTED ON	WINDOWS EDITIONS ACTIVATED BY THIS KMS HOST
Volume License for Windows Server 2016	• Windows Server 2016 • Windows Server 2012 R2 • Windows Server 2012	• Windows Server Semi-Annual Channel • Windows Server 2016 (all editions) • Windows 10 LTSB (2015 and 2016) • Windows 10 Enterprise/Enterprise N • Windows 10 Professional/Professional N • Windows 10 Professional for Workstations/Professional N for Workstations • Windows 10 Education/Education N • Windows Server 2012 R2 (all editions) • Windows 8.1 Professional • Windows 8.1 Enterprise • Windows Server 2012 (all editions) • Windows Server 2008 R2 (all editions) • Windows Server 2008 (all editions) • Windows 7 Professional • Windows 7 Enterprise
Volume license for Windows 10	• Windows 10 • Windows 8.1 • Windows 7	• Windows 10 Professional • Windows 10 Professional N • Windows 10 Enterprise • Windows 10 Enterprise N • Windows 10 Education • Windows 10 Education N • Windows 10 Enterprise LTSB (2015) • Windows 10 Enterprise LTSB N (2015) • Windows 10 Pro for Workstations • Windows 8.1 Professional • Windows 8.1 Enterprise • Windows 7 Professional • Windows 7 Enterprise

KMS host required updates

Depending on which operating system your KMS host is running and which operating systems you want to activate, you might need to install one or more of the updates below. This is required when you want to activate a version of Windows that is newer than the version your KMS host is running.

NOTE

The updates listed below are the minimum required. Where later cumulative updates or monthly rollups are listed as an option, please install the latest available version for your operating system to benefit from additional security and other fixes.

KMS HOST OS VERSION	KMS CLIENT OS VERSION(S) TO ACTIVATE	REQUIRED UPDATE
Windows Server 2019	• Windows Server 2022	June 8, 2021—KB5003646 or later cumulative update
Windows Server 2016	• Windows Server 2022 • Windows Server 2019	June 8, 2021—KB5003638 or later cumulative update
Windows Server 2016	• Windows Server 2019	December 3, 2018—KB4478877 or later cumulative update
Windows Server 2012 R2	• Windows Server 2019 • Windows Server 2016 • Windows 10	November 27, 2018—KB4467695 (Preview of Monthly Rollup) or later monthly rollup
Windows Server 2012 R2	• Windows Server 2016 • Windows 10	July 2016 update rollup for Windows 8.1 and Windows Server 2012 R2 or later monthly rollup
Windows Server 2012	• Windows Server 2016 • Windows Server 2012 R2 • Windows 10	July 2016 update rollup for Windows Server 2012 or later monthly rollup
Windows Server 2008 R2	• Windows Server 2012 R2 • Windows Server 2012 • Windows 10	Update that enables Windows 7 and Windows Server 2008 R2 KMS hosts to activate Windows 10
Windows 8.1	• Windows 10	July 2016 update rollup for Windows 8.1 and Windows Server 2012 R2 or later monthly rollup
Windows 7	• Windows 10	Update that enables Windows 7 and Windows Server 2008 R2 KMS hosts to activate Windows 10

Server Core App Compatibility Feature on Demand (FOD)

12/17/2021 • 9 minutes to read • Edit Online

The Server Core App Compatibility Feature on Demand (FOD) is an optional feature package that can be added to Server Core installations of Windows Server (starting with Windows Server 2019) or Windows Server Semi-Annual Channel installations at any time.

For more information on other Features on Demand, see Features On Demand.

Why install the App Compatibility FOD?

App Compatibility, a Feature on Demand for Server Core, significantly improves the app compatibility of the Server Core installation option by including a subset of binaries and packages from the Server with Desktop Experience installation option, but without adding the graphical environment. This optional package is available on a separate ISO, or from Windows Update, but can only be added to Server Core installations and images.

The two primary values the App Compatibility FOD provides are:

- Increases the compatibility of Server Core for server applications that are already in market or have already been developed by organizations and deployed.
- Assists with providing OS components and increased app compatibility of software tools used in acute troubleshooting and debugging scenarios.

Operating system components that are available as part of the Server Core App Compatibility FOD include:

- Microsoft Management Console (mmc.exe)
- Event Viewer (Eventvwr.msc)
- Performance Monitor (PerfMon.exe)
- Resource Monitor (Resmon.exe)
- Device Manager (Devmgmt.msc)
- File Explorer (Explorer.exe)
- Windows PowerShell (Powershell_ISE.exe)
- Disk Management (Diskmgmt.msc)
- Failover Cluster Manager (CluAdmin.msc)

> **NOTE**
> Failover Cluster Manager requires adding the Failover Clustering Windows Server feature first, which can be done by running the following command from an an elevated PowerShell session:
>
> ```
> Install-WindowsFeature -Name Failover-Clustering -IncludeManagementTools
> ```

Starting with Windows Server, version 1903, the following components are also available (when using the same version of the App Compatibility FOD):

- Hyper-V Manager (virtmgmt.msc)
- Task Scheduler (taskschd.msc)

Installing the App Compatibility Feature on Demand

> **IMPORTANT**
> The App Compatibility FOD can only be installed on Server Core. Don't attempt to add the Server Core App Compatibility FOD to the Server with Desktop Experience installation option.

Caution

There is currently a known issue with App Compatibility FOD for Windows Server 2022. Once installed, if you try to connect to the server using Remote Desktop Protocol (RDP), you can be presented with a black screen and disconnected. This issue will be fixed in a future Cumulative Update. Windows Server 2019 and previous versions are not affected.

Connected to the internet

1. If the server can connect to Windows Update, all you have to do is run the following command from an elevated PowerShell session and then restart Windows Server after the command finishes running:

```
Add-WindowsCapability -Online -Name ServerCore.AppCompatibility~~~~0.0.1.0
```

Disconnected from the internet

1. If the server can't connect to Windows Update, instead download the Windows Server Languages and Optional Features ISO image file, and copy the ISO to a shared folder on your local network:

   - If you have a volume license, you can download the Windows Server Languages and Optional Features ISO image file from the same portal where the operating system ISO image file is obtained: Volume Licensing Service Center.
   - The Windows Server Languages and Optional Features ISO image file is also available on the Microsoft Evaluation Center or on the Visual Studio portal for subscribers.

   > **NOTE**
   > The Languages and Optional Features ISO image file is new for Windows Server 2022. Previous versions of Windows Server use the Features on Demand (FOD) ISO.

2. Sign in with an administrator account on the Server Core computer that is connected to your local network and that you want to add the App Compatibility FOD to.

Mount the FOD ISO

1. Use `New-PSDrive` from PowerShell, `net use` from Command Prompt, or some other method, to connect to the location of the FOD ISO. For example, in an elevated PowerShell session run the following command:

```
$credential = Get-Credential

New-PSDrive -Name FODShare -PSProvider FileSystem -Root "\\server\share" -Credential $credential
```

2. Copy the FOD ISO to a local folder of your choosing (this may take some time). Edit the variables below with your folder location and ISO filename, and run the following commands, for example:

```
$isoFolder = "C:\SetupFiles\WindowsServer\ISOs"
$fodIsoFilename = "FOD_ISO_filename.iso"

New-Item -ItemType Directory -Path $isoFolder
Copy-Item -Path "FODShare:\$fodIsoFilename" -Destination $isoFolder -Verbose
```

3. Mount the FOD ISO by using the following command:

```
$fodIso = Mount-DiskImage -ImagePath "$isoFolder\$fodIsoFilename"
```

4. Run the following command to get the drive letter that the FOD ISO has been mounted to:

```
$fodDriveLetter = ($fodIso | Get-Volume).DriveLetter
```

5. Run the following command (depending on the operating system version):

   For Windows Server 2022:

```
Add-WindowsCapability -Online -Name ServerCore.AppCompatibility~~~~0.0.1.0 -Source
${fodDriveLetter}:\LanguagesAndOptionalFeatures\ -LimitAccess
```

   For previous versions of Windows Server:

```
Add-WindowsCapability -Online -Name ServerCore.AppCompatibility~~~~0.0.1.0 -Source
${fodDriveLetter}:\ -LimitAccess
```

6. After the progress bar completes, restart the operating system.

Optionally add Internet Explorer 11 to Server Core

> **NOTE**
>
> The Server Core App Compatibility FOD is required for the addition of Internet Explorer 11, but Internet Explorer 11 is not required to add the Server Core App Compatibility FOD.

> **NOTE**
>
> Starting with Windows Server 2022, although Internet Explorer 11 can be added to Server Core installations of Windows Server, Microsoft Edge should be used instead. Microsoft Edge has Internet Explorer mode ("IE mode") built in, so you can access legacy Internet Explorer-based websites and applications straight from Microsoft Edge. Please see here for information on the lifecycle policy for Internet Explorer.

1. Sign in as Administrator on the Server Core computer that already has the App Compatibility FOD added and the FOD optional package ISO copied locally.

2. Mount the FOD ISO by using the command below. This step assumes that you have already copied the FOD ISO locally. If not, please complete steps 1 and 2 from Mount the FOD ISO above. The commands below follow on from these two steps. Edit the variables below with your folder location and ISO filename, and run the following commands, for example:

```
$isoFolder = "C:\SetupFiles\WindowsServer\ISOs"
$fodIsoFilename = "FOD_ISO_filename.iso"

$fodIso = Mount-DiskImage -ImagePath "$isoFolder\$fodIsoFilename"
```

3. Run the following command to get the drive letter that the FOD ISO has been mounted to:

```
$fodDriveLetter = ($fodIso | Get-Volume).DriveLetter
```

4. Run the following commands (depending on your operating system version), using the `$packagePath` variable as the path to the Internet Explorer .cab file:

   For Windows Server 2022:

```
$packagePath = "${fodDriveLetter}:\LanguagesAndOptionalFeatures\Microsoft-Windows-InternetExplorer-Optional-Package~31bf3856ad364e35~amd64~~.cab"

Add-WindowsPackage -Online -PackagePath $packagePath
```

   For previous versions of Windows Server:

```
$packagePath = "${fodDriveLetter}:\Microsoft-Windows-InternetExplorer-Optional-Package~31bf3856ad364e35~amd64~~.cab"

Add-WindowsPackage -Online -PackagePath $packagePath
```

5. After the progress bar completes, restart the operating system.

Release notes and suggestions

> **IMPORTANT**
> Features on Demand installed on Semi-Annual Channel versions of Windows Server won't remain in place after an in-place upgrade to a newer Semi-Annual Channel version, so you will have to install them again after the upgrade. Alternatively, you can add Features on Demand to the new Windows Server installation source prior to upgrading. This ensures that the new version of any Features on Demand are present after the upgrade completes. For more info, see the Adding capabilities and optional packages to an offline WIM Server Core image below.

- After installation of the App Compatibility FOD and reboot of the server, the command console window frame color will change to a different shade of blue.
- If you choose to also install the Internet Explorer 11 optional package, note that double-clicking to open locally saved .htm files is not supported. However, you can **right-click** and choose **Open with Internet Explorer**, or you can open it directly from Internet Explorer **File** -> **Open**.
- To further enhance the app compatibility of Server Core with the App Compatibility FOD, the IIS Management Console has been added to Server Core as an optional component. However, it is necessary to first add the App Compatibility FOD to use the IIS Management Console. IIS Management Console relies on the Microsoft Management Console (mmc.exe), which is only available on Server Core with the addition of the App Compatibility FOD. Use the PowerShell cmdlet **Install-WindowsFeature** to add IIS Management Console:

```
Install-WindowsFeature -Name Web-Mgmt-Console
```

- As a general point of guidance, when installing applications on Server Core (with or without these optional packages) it is sometimes necessary to use silent install options and instructions.

Adding to an offline WIM Server Core image

1. Download both the Languages and Optional Features ISO and the Windows Server ISO image files to a local folder on a Windows computer. This can be a desktop Windows PC and does not need to be running Windows Server with the Server Core installation option.

   - If you have a volume license, you can download the Windows Server Languages and Optional Features ISO image file from the same portal where the operating system ISO image file is obtained: Volume Licensing Service Center.
   - The Windows Server Languages and Optional Features ISO image file is also available on the Microsoft Evaluation Center or on the Visual Studio portal for subscribers.

   > **NOTE**
   > The Languages and Optional Features ISO image file is new for Windows Server 2022. Previous versions of Windows Server use the Features on Demand (FOD) ISO.

2. Mount both the Languages and Optional Features ISO and the Windows Server ISO by running the commands below in an elevated PowerShell session. Edit the variables below with your folder location and ISO filename, and run the following commands, for example::

```
$isoFolder = "C:\SetupFiles\WindowsServer\ISOs"
$fodIsoFilename = "FOD_ISO_filename.iso"
$wsIsoFilename = "Windows_Server_ISO_filename.iso"

$fodIso = Mount-DiskImage -ImagePath "$isoFolder\$fodIsoFilename"
$wsIso = Mount-DiskImage -ImagePath "$isoFolder\$wsIsoFilename"
```

3. Run the following command to get the drive letters that the FOD ISO and Windows Server ISO have been mounted to:

```
$fodDriveLetter = ($fodIso | Get-Volume).DriveLetter
$wsDriveLetter = ($wsIso | Get-Volume).DriveLetter
```

4. Copy the contents of the Windows Server ISO file to a local folder, for example, **C:\SetupFiles\WindowsServer\Files**. This may take some time:

```
$wsFiles = "C:\SetupFiles\WindowsServer\Files"
New-Item -ItemType Directory -Path $wsFiles

Copy-Item -Path ${wsDriveLetter}:\* -Destination $wsFiles -Recurse
```

5. Get the image name you want to modify within the install.wim file by using the following command. Add your path to the install.wim file to the `$installWimPath` variable, located inside the **sources** folder of the Windows Server ISO file. Note the names of the images available in this install.wim file from the output.

```
$installWimPath = "C:\SetupFiles\WindowsServer\Files\sources\install.wim"

Get-WindowsImage -ImagePath $installWimPath
```

6. Mount the install.wim file in a new folder by using the following command replacing the sample variable values with your own, and reusing the `$installWimPath` variable from the previous command.

   - `$wimImageName` - Enter the name of the image you want to mount from the output of the previous command. The example here uses **Windows Server 2022 Datacenter**.
   - `$wimMountFolder` - Specify an empty folder to use when accessing the contents of the install.wim file.

```
$wimImageName = "Windows Server 2022 Datacenter"
$wimMountFolder = "C:\SetupFiles\WindowsServer\WIM"

New-Item -ItemType Directory -Path $wimMountFolder
Set-ItemProperty -Path $installWimPath -Name IsReadOnly -Value $false
Mount-WindowsImage -ImagePath $installWimPath -Name $wimImageName -Path $wimMountFolder
```

7. Add the capabilities and packages you want to the mounted install.wim image by using the following commands (depending on the version), replacing the sample variable values with your own.

   - `$capabilityName` - Specify the name of the capability to install (in this case, the **AppCompatibility** capability).
   - `$packagePath` - Specify the path to the package to install (in this case, to the **Internet Explorer** cab file).

   For Windows Server 2022:

```
$capabilityName = "ServerCore.AppCompatibility~~~~0.0.1.0"
$packagePath = "${fodDriveLetter}:\LanguagesAndOptionalFeatures\Microsoft-Windows-InternetExplorer-
Optional-Package~31bf3856ad364e35~amd64~~.cab"

Add-WindowsCapability -Path $wimMountFolder -Name $capabilityName -Source
"${fodDriveLetter}:\LanguagesAndOptionalFeatures" -LimitAccess
Add-WindowsPackage -Path $wimMountFolder -PackagePath $packagePath
```

   For previous versions of Windows Server:

```
$capabilityName = "ServerCore.AppCompatibility~~~~0.0.1.0"
$packagePath = "${fodDriveLetter}:\Microsoft-Windows-InternetExplorer-Optional-
Package~31bf3856ad364e35~amd64~~.cab"

Add-WindowsCapability -Path $wimMountFolder -Name $capabilityName -Source "${fodDriveLetter}:\" -
LimitAccess
Add-WindowsPackage -Path $wimMountFolder -PackagePath $packagePath
```

8. Dismount and commit changes to the install.wim file by using the following command, which uses the `$wimMountFolder` variable from previous commands:

```
Dismount-WindowsImage -Path $wimMountFolder -Save
```

You can now upgrade your server by running setup.exe from the folder you created for the Windows Server installation files, in this example: **C:\SetupFiles\WindowsServer\Files**. This folder now contains the Windows Server installation files with the additional capabilities and optional packages included.

Windows Server 2022 and Microsoft server applications compatibility

12/17/2021 • 2 minutes to read • Edit Online

This table lists Microsoft server applications that support installation and functionality on Window Server 2022. This information is for quick reference and is not intended to replace the individual product specifications, requirements, announcements, or general communications of each individual server application. Refer to official documentation for each product to fully understand compatibility and options.

TIP

If you are a software vendor partner looking for more information on Windows Server compatibility with non-Microsoft applications, visit the Commercial App Certification portal.

PRODUCT	SUPPORTED ON SERVER CORE	SUPPORTED ON SERVER WITH DESKTOP EXPERIENCE	RELEASED	PRODUCT WEB LINK
Azure DevOps Server 2020.1	Yes*	Yes	Yes	Azure DevOps Server 2020.1 release notes
Configuration Manager (version 2107)	Yes as a managed client and distribution point. No as a site server.	Yes as a site server/site systems and a managed client.	Yes	Support for Windows Server 2022
Office Online Server	No	Yes	Yes	Plan Office Online Server
Project Server 2019	No	Yes	Yes	Software requirements for Project Server 2019 - Project Server
Project Server Subscription Edition	Yes	Yes	Yes	Software requirements for Project Server Subscription Edition
SharePoint Server 2019	No	Yes	Yes	Hardware and software requirements for SharePoint Server 2019
SharePoint Server Subscription Edition	Yes	Yes	Yes	System requirements for SharePoint Server Subscription edition

PRODUCT	SUPPORTED ON SERVER CORE	SUPPORTED ON SERVER WITH DESKTOP EXPERIENCE	RELEASED	PRODUCT WEB LINK
SQL Server 2017	Yes*	Yes	Yes	Hardware and Software Requirements for Installing SQL Server 2017
SQL Server 2019	Yes*	Yes	Yes	Hardware and Software Requirements for Installing SQL Server 2019
System Center Data Protection Manager 2019	Yes as a backup workload. No as a DPM server.	Yes as a backup workload. No as a DPM server.	Yes	Preparing your environment for System Center Data Protection Manager
System Center Data Protection Manager 2022	Yes*	Yes	No	
System Center Operations Manager 2019	Yes as an agent. No as a Management Server**	Yes as an agent. No as a Management Server**.	Yes	System requirements for System Center Operations Manager
System Center Operations Manager 2022	Yes*	Yes	No	
System Center Virtual Machine Manager 2022	Yes*	Yes	No	

*May have limitations or may require the Server Core App Compatibility Feature on Demand (FOD). Please refer to specific product or Feature on Demand documentation.

** Refer to Product Web Link

Windows Server 2019 and Microsoft server applications compatibility

12/17/2021 • 2 minutes to read • Edit Online

This table lists Microsoft server applications that support installation and functionality on Window Server 2019. This information is for quick reference and is not intended to replace the individual product specifications, requirements, announcements, or general communications of each individual server application. Refer to official documentation for each product to fully understand compatibility and options.

TIP

If you are a software vendor partner looking for more information on Windows Server compatibility with non-Microsoft applications, visit the Commercial App Certification portal.

PRODUCT	SUPPORTED ON SERVER CORE	SUPPORTED ON SERVER WITH DESKTOP EXPERIENCE	RELEASED	PRODUCT WEB LINK
Azure DevOps Server 2019	Yes*	Yes	Yes	Azure DevOps Server 2019
Azure DevOps Server 2020	Yes*	Yes	Yes	Azure DevOps Server 2020
Configuration Manager (version 1806)	Yes as managed client, No as site server	Yes as managed client, No as site server	Yes	What's new in version 1806 of Configuration Manager current branch
Exchange Server 2019	Yes	Yes	Yes	Exchange Server system requirements
Host Integration Server 2016, CU3	Yes	Yes	Yes	Host Integration Server system requirements
Office Online Server	No	Yes	Yes	Plan Office Online Server
Project Server 2016	No	Yes	Yes	Software requirements for Project Server 2016
Project Server 2019	No	Yes	Yes	Software requirements for Project Server 2019

PRODUCT	SUPPORTED ON SERVER CORE	SUPPORTED ON SERVER WITH DESKTOP EXPERIENCE	RELEASED	PRODUCT WEB LINK
Project Server Subscription Edition	Yes	Yes	Yes	Software requirements for Project Server Subscription Edition
SharePoint Server 2016	No	Yes	Yes	Hardware and software requirements for SharePoint Server 2016
SharePoint Server 2019	No	Yes	Yes	Hardware and software requirements for SharePoint Server 2019
SharePoint Server Subscription Edition	Yes	Yes	Yes	System requirements for SharePoint Server Subscription edition
Skype for Business 2019	No	Yes	Yes	Install prerequisites for Skype for Business Server
SQL Server 2014	Yes*	Yes	Yes	Hardware and Software Requirements for Installing SQL Server 2014
SQL Server 2016	Yes*	Yes	Yes	Hardware and Software Requirements for Installing SQL Server 2016
SQL Server 2017	Yes*	Yes	Yes	Hardware and Software Requirements for Installing SQL Server 2017
SQL Server 2019	Yes*	Yes	Yes	Hardware and Software Requirements for Installing SQL Server 2019
System Center Data Protection Manager 2019	No	Yes	Yes	Preparing your environment for System Center Data Protection Manager

PRODUCT	SUPPORTED ON SERVER CORE	SUPPORTED ON SERVER WITH DESKTOP EXPERIENCE	RELEASED	PRODUCT WEB LINK
System Center Operations Manager 2019	Yes*	Yes	Yes	System requirements for System Center Operations Manager
System Center Virtual Machine Manager 2019	Yes*	Yes	Yes	System requirements for System Center Virtual Machine Manager

*May have limitations or may require the Server Core App Compatibility Feature on Demand (FOD). Please refer to specific product or FOD documentation.

Windows Server 2016 and Microsoft server applications compatibility

12/17/2021 • 2 minutes to read • Edit Online

This table lists Microsoft server applications that support installation and functionality on Window Server 2016. This information is for quick reference and is not intended to replace the individual product specifications, requirements, announcements, or general communications of each individual server application. Refer to official documentation for each product to fully understand compatibility and options.

TIP

If you are a software vendor partner looking for more information on Windows Server compatibility with non-Microsoft applications, visit the Commercial App Certification portal.

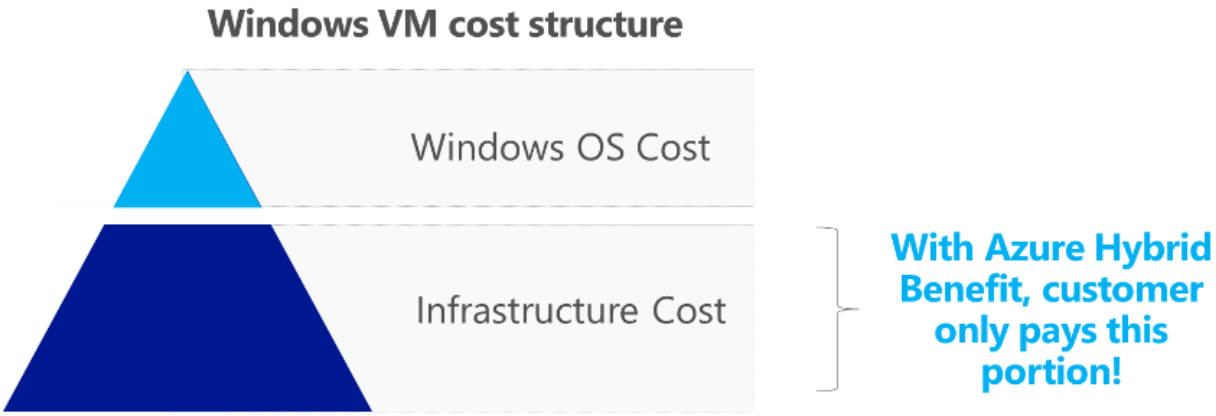
PRODUCT	RELEASED	PRODUCT WEB LINK
BizTalk Server 2016	Yes	Microsoft BizTalk Server
Configuration Manager (version 1606)	Yes	What's new in version 1606 of Configuration Manager
Exchange Server 2016	Yes	Updates for Exchange 2016
Host Integration Server 2016	Yes	What's New in HIS 2016
Office Online Server	Yes	Plan Office Online Server
Project Server 2016	Yes	Software requirements for Project Server 2016
Project Server 2019	Yes	Software requirements for Project Server 2019
SharePoint Server 2016	Yes	Hardware and software requirements for SharePoint Server 2016
SharePoint Server 2019	Yes	Hardware and software requirements for SharePoint Server 2019
Skype for Business Server 2015	Yes	How to install Skype for Business Server 2015 on Windows Server 2016
SQL Server 2012	Yes	Hardware and Software Requirements for Installing SQL Server 2012
SQL Server 2014	Yes	Hardware and Software Requirements for Installing SQL Server 2014
SQL Server 2016	Yes	SQL Server 2016

PRODUCT	RELEASED	PRODUCT WEB LINK
System Center Virtual Machine Manager 2016	Yes	What's New in System Center
System Center Operations Manager 2016	Yes	What's New in System Center
System Center Data Protection Manager 2016	Yes	What's New in System Center
Visual Studio Team Foundation Server 2017	Yes	Team Foundation Server 2017

Azure Hybrid Benefit for Windows Server

12/17/2021 • 5 minutes to read • Edit Online

Azure Hybrid Benefit for Windows Server allows you to save up to 40% on Windows Server virtual machines (VMs) in Azure by using your on-premises Windows Server licenses with Software Assurance or subscription license. With this benefit, you only need to pay for the infrastructure costs of the virtual machine because the license for Windows Server is covered by the Software Assurance benefit. The benefit is applicable to both Standard and Datacenter editions of Windows Server for Windows Server 2012 R2, Windows Server 2016, Windows Server 2019, and Windows Server 2022. This benefit is available across all regions and sovereign clouds.



Rules and use cases

All you need to qualify for the benefit is an active Software Assurance (SA) or a subscription license such as EAS, SCE subscription or Open Value Subscription on your Windows Server licenses.

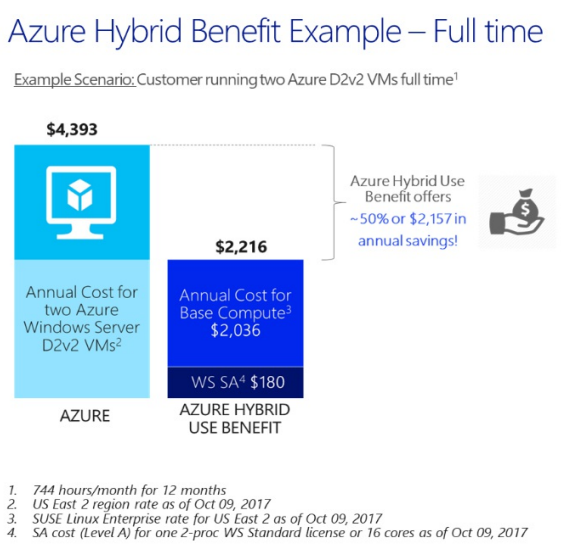
Each Windows Server two-processor license with active SA or subscription, and each set of 16 core licenses for Windows Server with active SA or subscription, entitles you to use Windows Server on Microsoft Azure with up to 16 virtual cores allocated across two or fewer Azure Base Instances (VMs). Each extra set of eight core licenses with active SA or subscription entitles use on up to eight virtual cores and one Azure Base Instance (VM). Differences how they can be used depends on which edition the licenses are for, Datacenter or Standard. The table below summarizes this:

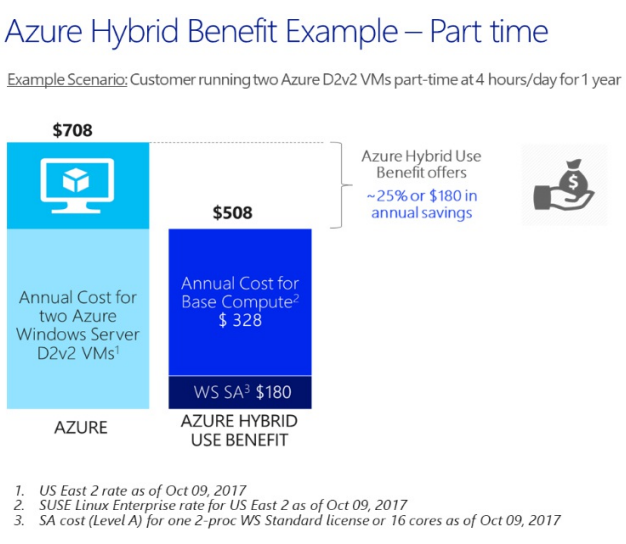
LICENSE WITH SA/SUBSCRIPTION	VMS AND CORES GRANTED	HOW THEY CAN BE USED
WS Datacenter (16 cores or two-processors)	Up to two VMs with up to 16 cores total	Run virtual machines both on-premises and in Azure
WS Standard (16 cores or two-processors)	Up to two VMs with up to 16 cores total	Run virtual machines either on-premises or in Azure

Windows Server VMs using Azure Hybrid Benefit can run in Azure only during the SA or subscription term. When the SA or subscription is nearing expiry, you need to either renew your SA or subscription, disable the hybrid benefit functionality, or de-provision those VMs using Azure Hybrid Benefit.

Example savings

The image below illustrates examples of Azure Hybrid Benefit for Windows Server being used full time and part time:





Below is a reference table to assist you with understanding the benefit rules with more granularity. The green column shows the quantity of same-type VMs, and the blue row shows the core density of each VM. The yellow cells show the number of two-processor licenses (or sets of 16 cores) you must have to deploy a certain number of VMs of a certain core density:

VM Core Density

Number of same-type VMs

	1	2	4	8	10	12	16	20	32	64	128
1	1	1	1	1	1	1	1	2	2	4	8
2	1	1	1	1	2	2	2	3	4	8	16
3	2	2	2	2	3	3	3	6	6	12	24
4	2	2	2	2	4	4	4	8	8	16	32
5	3	3	3	3	5	5	5	10	10	20	40
6	3	3	3	3	6	6	6	12	12	24	48
7	4	4	4	4	7	7	7	14	14	28	56
8	4	4	4	4	8	8	8	16	16	32	64
9	5	5	5	5	9	9	9	18	18	36	72
10	5	5	5	5	10	10	10	20	20	40	80

Examples:

1. To run 2 VMs of 4 cores each (such as A3) you need 1 WS license.

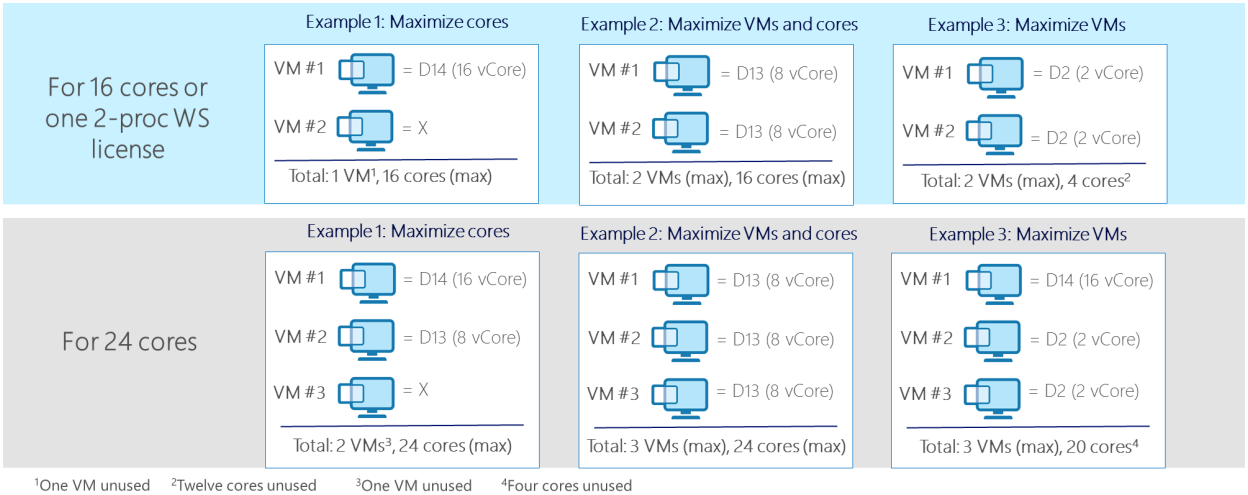
2. To run 5 VMs of 16 cores each (such as D5V2) you need 5 WS licenses

3. To run 10 VMs of 64 cores each (such as M64S) you need 40 WS licenses

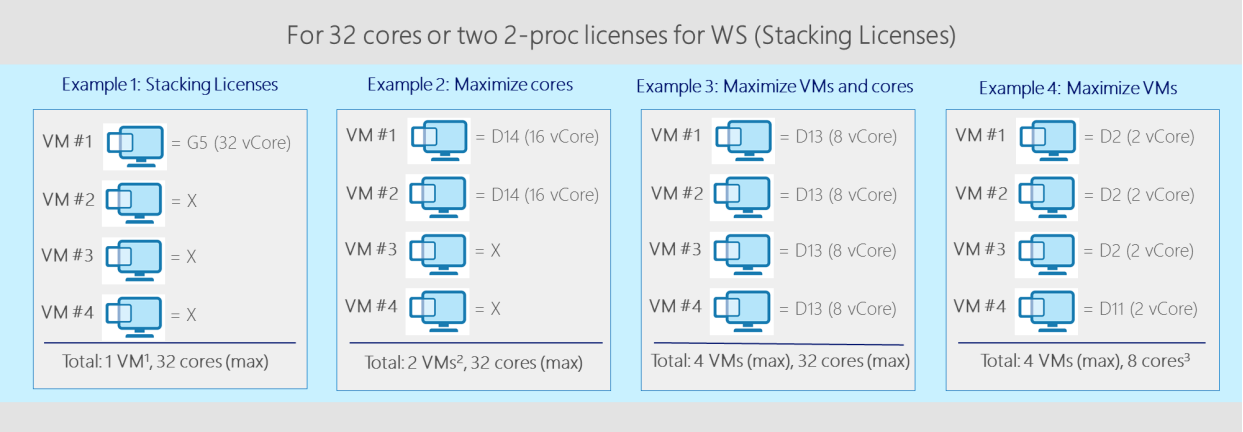
*Number of WS licenses covered with SA or WS subscriptions needed for each scenario**

Example configurations

Azure Hybrid Benefit for Windows Server also allows flexibility to run different VM configurations and combining VMs of different types. The examples below show configurations for some licensing scenarios:



[1]One VM unused [2]Twelve cores unused [3]One VM unused [4]Four cores unused



[1]Three VMs unused
[2]Two VMs unused
[3] Twenty-four cores unused

How to use Azure Hybrid Benefit for Windows Server

We have enabled pre-built gallery images that are available for all our customers who have eligible licenses, irrespective of where they bought them, and enabled partners to be able to perform the deployments on customers' behalf. You can learn how to deploy Windows Server VMs with Azure Hybrid Benefit in Azure documentation for Azure Hybrid Benefit for Windows Server.

How to maintain compliance

If you apply Azure Hybrid Benefit to your Windows Server VMs, you need to verify the number of eligible licenses and respective coverage period of your SA or subscription before any activation of this benefit, and apply the guidelines above to ensure you deploy the correct number of Windows Server VMs with Azure Hybrid Benefit.

If you already have Windows Server VMs running with Azure Hybrid Benefit, you need to perform an inventory of how many units you are running, and check this against the SA or subscription licenses you have. Please contact your Microsoft Enterprise Agreement licensing specialist to validate your SA or subscription licensing position.

To see and count all virtual machines deployed with Azure Hybrid Benefit for Windows Server in a subscription, please follow the steps in this article in Azure documentation for Azure Hybrid Benefit for Windows Server to list all VMs and Virtual Machine Scale Sets with Azure Hybrid Benefit for Windows Server in a subscription. You can also look at your Microsoft Azure bill to determine how many virtual machines with Azure Hybrid Benefit for Windows Server you are running. The information about the number of instances with the benefit shows under **Additional Info**:

```
"{"ImageType":"WindowsServerBYOL","ServiceType":"Standard_A1","VMName":"","UsageType":"ComputeHR"}"
```

Please note that billing does not apply in real time. There will be a delay of several hours from the time you've activated a Windows Server VM with Azure Hybrid Benefit before it shows on your bill. You can then populate the results in the Azure Hybrid Benefit for Windows Server SA Count Tool below to get to the number of Windows Server licenses covered by SA or subscriptions that are required.

Be sure to perform an inventory in each Azure subscription that you own to generate a comprehensive view of your licensing position. Once you have confirmed you are fully licensed for the number of Windows Server VMs you are running with Azure Hybrid Benefit, there is no need for any further action. You should perform an inventory regularly to ensure you are using any license benefits you are entitled to in order to reduce your costs, but also to ensure that you always have enough licenses to cover the number of Windows Server VMs you have deployed with Azure Hybrid Benefit.

If you do not have enough eligible Windows Server licenses for the number of VMs already deployed, you either need to purchase extra Windows Server on-premises licenses covered with SA or subscription through one of the channels listed in the table below, purchase Windows Server VMs at regular Azure hourly rates by disabling Azure Hybrid Benefit for some VMs, or deallocate some VMs. Please note that you may buy core licenses in increments of eight cores, to qualify for each additional Windows Server VM with Azure Hybrid Benefit.

Windows Server Software Assurance and/or subscriptions are available for purchase through one of a combination of the following Microsoft licensing channels:

CHANNEL	OPEN	OVS	SELECT/ SELECT PLUS	MPSA	EA/EAS
Typical size (# of devices)	5-250	5-250	>250	>250	>500
SA / Subscription	Optional	Included	Optional	Optional	Included

> **NOTE**
> Microsoft reserves the right to audit customers at any time to verify eligibility for Azure Hybrid Benefit utilization.

If you want to learn more, including pricing, please see Azure Hybrid Benefit.

How to create a Key Management Services (KMS) activation host

12/17/2021 • 4 minutes to read • Edit Online

KMS uses a client-server model to active Windows clients and is used for volume activation on your local network. KMS clients connect to a KMS server, called the KMS host, for activation. The KMS clients that a KMS host can activate are dependent on the host key used to activate the KMS host. This article walks you through the steps you need to create a KMS host. To learn more about KMS and the initial planning considerations, see Key Management Services (KMS) activation planning.

Prerequisites

A single KMS host can support an unlimited number of KMS clients. If you have more than 50 clients, we recommend that you have at least two KMS hosts in case one of your KMS hosts becomes unavailable. Most organizations can operate with as few as two KMS hosts for their entire infrastructure.

KMS hosts do not need to be dedicated servers, and KMS can be co-hosted with other services. You can run a KMS host on any physical or virtual system that is running a supported Windows Server or Windows client operating system.

The version of Windows you use for your KMS host determines the version of Windows you can activate for your KMS clients. Please see the table of activation versions to help you decide which is right for your environment.

By default, KMS hosts automatically publish SRV resource records in DNS. This enables KMS clients to automatically discover the KMS host and activate without the need for any configuration on the KMS client. Automatic publishing can be disabled and the records can be created manually, which is also necessary for automatic activation if the DNS service does not support dynamic updates.

You will need:

- A computer running Windows Server or Windows. A KMS host running on a Windows Server operating system can activate computers running both server and client operating systems, however a KMS host running on a Windows client operating system can only activate computers also running client operating systems.
- The user account you use must be a member of the Administrators group on the KMS host.
- A KMS host key for your organization. You can get this key from the Product Keys section of the Volume Licensing Service Center.

Install and configure a KMS host

1. From an elevated PowerShell session, run the following command to install the Volume Activation Services role:

   ```
   Install-WindowsFeature -Name VolumeActivation -IncludeManagementTools
   ```

2. Configure the Windows Firewall to allow the Key Management Service to receive network traffic. You can allow this for any network profiles (default), or for any combination of Domain, Private, and Public network profiles. By default, a KMS host is configured to use TCP on port 1688. In the example below, the

firewall rule is configured to allow network traffic for the Domain and Private network profiles only:

```
Set-NetFirewallRule -Name SPPSVC-In-TCP -Profile Domain,Private -Enabled True
```

3. Launch the Volume Activation Tools wizard by running:

```
vmw.exe
```

4. Select **Next** on the introduction screen. Select **Key Management Service (KMS)** as the activation type and enter `localhost` to configure the local server or the hostname of the server you want to configure.
5. Select **Install your KMS host key** and enter the product key for your organization, then select **Commit**.
6. Once the product key has been installed, you need to activate the product. Click **Next**.
7. Select the product you want to activate from the dropdown menu, then select whether you want to activate online or by phone. In this example, select **Activate online** and then **Commit**.
8. Once activation is successful, the KMS host configuration will be shown. If this is the configuration you want, you can select **Close** to exit the wizard. DNS records will be created and you can start activating KMS clients. See the section below if you need to manually create DNS records. If you want to change the configuration settings, select **Next**.
9. **Optional**: Change the configuration values based on your requirements and select **Commit**.

> **NOTE**
> You can now start activating KMS clients, however a network must have a minimum number of computers (called the activation threshold). KMS hosts count the number of recent connections and so when a client or server contacts the KMS host, the host adds the machine ID to its count and then returns the current count value in its response. The client or server will activate if the count is high enough. Windows clients will activate if the count is 25 or higher. Windows Server and volume editions of Microsoft Office products will activate if the count is five or greater. The KMS only counts unique connections from the past 30 days, and only stores the 50 most recent contacts.

Manually create DNS records

If your DNS service does not support dynamic update, the resource records must be manually created to publish the KMS host. Create DNS resource records for KMS manually with your DNS service using the information below (altering the default port number if you changed this in the KMS host configuration):

PROPERTY	VALUE
Type	SRV
Service/Name	_vlmcs
Protocol	_tcp
Priority	0
Weight	0
Port number	1688

PROPERTY	VALUE
Hostname	*FQDN of the KMS host*

You should also disable publishing on all KMS hosts if your DNS service does not support dynamic update to prevent event logs from collecting failed DNS publishing events.

> **TIP**
> Manually created resource records can also coexist with resource records that KMS hosts automatically publish in other domains as long as all records are maintained to prevent conflicts.

Disable publishing of DNS records

To disable publishing of DNS records by the KMS host:

1. Launch the Volume Activation Tools wizard by running:

   ```
   vmw.exe
   ```

2. Select **Next** on the introduction screen. Select **Key Management Service (KMS)** as the activation type and enter `localhost` to configure the local server or the hostname of the server you want to configure.
3. Select **Skip to Configuration**, then select **Next**.
4. Uncheck the box for publish DNS records, then select **Commit**.

Key Management Services (KMS) client activation and product keys

12/17/2021 • 5 minutes to read • Edit Online

To use KMS, you need to have a KMS host available on your local network. Computers that activate with a KMS host need to have a specific product key. This key is sometimes referred to as the KMS client key, but it is formally known as a Microsoft Generic Volume License Key (GVLK). Computers that are running volume licensing editions of Windows Server and Windows client are, by default, KMS clients with no extra configuration needed as the relevant GVLK is already there.

There are some scenarios, however, where you will need to add the GVLK to the computer you wish to activate against a KMS host, such as:

- Converting a computer from using a Multiple Activation Key (MAK)
- Converting a retail license of Windows to a KMS client
- If the computer was previously a KMS host.

> **IMPORTANT**
>
> To use the keys listed here (which are GVLKs), you must first have a KMS host available on your local network. If you don't already have a KMS host, please see how to create a KMS host to learn more.
>
> If you want to activate Windows without a KMS host available and outside of a volume-activation scenario (for example, you're trying to activate a retail version of Windows client), **these keys will not work**. You will need to use another method of activating Windows, such as using a MAK, or purchasing a retail license. Get help to find your Windows product key and learn about genuine versions of Windows.

Install a product key

If you are converting a computer from a KMS host, MAK, or retail edition of Windows to a KMS client, install the applicable product key (GVLK) from the list below. To install a client product key, open an administrative command prompt on the client, and run the following command and then press `Enter`:

```
slmgr /ipk <product key>
```

For example, to install the product key for Windows Server 2022 Datacenter edition, run the following command and then press `Enter`:

```
slmgr /ipk WX4NM-KYWYW-QJJR4-XV3QB-6VM33
```

Generic Volume License Keys (GVLK)

In the tables that follow, you will find the GVLKs for each version and edition of Windows. LTSC is *Long-Term Servicing Channel*, while LTSB is *Long-Term Servicing Branch*.

Windows Server (LTSC versions)

Windows Server 2022

OPERATING SYSTEM EDITION	KMS CLIENT PRODUCT KEY
Windows Server 2022 Datacenter	WX4NM-KYWYW-QJJR4-XV3QB-6VM33
Windows Server 2022 Standard	VDYBN-27WPP-V4HQT-9VMD4-VMK7H

Windows Server 2019

OPERATING SYSTEM EDITION	KMS CLIENT PRODUCT KEY
Windows Server 2019 Datacenter	WMDGN-G9PQG-XVVXX-R3X43-63DFG
Windows Server 2019 Standard	N69G4-B89J2-4G8F4-WWYCC-J464C
Windows Server 2019 Essentials	WVDHN-86M7X-466P6-VHXV7-YY726

Windows Server 2016

OPERATING SYSTEM EDITION	KMS CLIENT PRODUCT KEY
Windows Server 2016 Datacenter	CB7KF-BWN84-R7R2Y-793K2-8XDDG
Windows Server 2016 Standard	WC2BQ-8NRM3-FDDYY-2BFGV-KHKQY
Windows Server 2016 Essentials	JCKRF-N37P4-C2D82-9YXRT-4M63B

Windows Server (Semi-Annual Channel versions)

Windows Server, versions 20H2, 2004, 1909, 1903, and 1809

OPERATING SYSTEM EDITION	KMS CLIENT PRODUCT KEY
Windows Server Datacenter	6NMRW-2C8FM-D24W7-TQWMY-CWH2D
Windows Server Standard	N2KJX-J94YW-TQVFB-DG9YT-724CC

Windows 11 and Windows 10 (Semi-Annual Channel versions)

See the Windows lifecycle fact sheet for information about supported versions and end of service dates.

OPERATING SYSTEM EDITION	KMS CLIENT PRODUCT KEY
Windows 11 Pro Windows 10 Pro	W269N-WFGWX-YVC9B-4J6C9-T83GX
Windows 11 Pro N Windows 10 Pro N	MH37W-N47XK-V7XM9-C7227-GCQG9
Windows 11 Pro for Workstations Windows 10 Pro for Workstations	NRG8B-VKK3Q-CXVCJ-9G2XF-6Q84J
Windows 11 Pro for Workstations N Windows 10 Pro for Workstations N	9FNHH-K3HBT-3W4TD-6383H-6XYWF
Windows 11 Pro Education Windows 10 Pro Education	6TP4R-GNPTD-KYYHQ-7B7DP-J447Y

OPERATING SYSTEM EDITION	KMS CLIENT PRODUCT KEY
Windows 11 Pro Education N Windows 10 Pro Education N	YVWGF-BXNMC-HTQYQ-CPQ99-66QFC
Windows 11 Education Windows 10 Education	NW6C2-QMPVW-D7KKK-3GKT6-VCFB2
Windows 11 Education N Windows 10 Education N	2WH4N-8QGBV-H22JP-CT43Q-MDWWJ
Windows 11 Enterprise Windows 10 Enterprise	NPPR9-FWDCX-D2C8J-H872K-2YT43
Windows 11 Enterprise N Windows 10 Enterprise N	DPH2V-TTNVB-4X9Q3-TJR4H-KHJW4
Windows 11 Enterprise G Windows 10 Enterprise G	YYVX9-NTFWV-6MDM3-9PT4T-4M68B
Windows 11 Enterprise G N Windows 10 Enterprise G N	44RPN-FTY23-9VTTB-MP9BX-T84FV

Windows 10 (LTSC/LTSB versions)

Windows 10 LTSC 2021 and 2019

OPERATING SYSTEM EDITION	KMS CLIENT PRODUCT KEY
Windows 10 Enterprise LTSC 2021 Windows 10 Enterprise LTSC 2019	M7XTQ-FN8P6-TTKYV-9D4CC-J462D
Windows 10 Enterprise N LTSC 2021 Windows 10 Enterprise N LTSC 2019	92NFX-8DJQP-P6BBQ-THF9C-7CG2H

Windows 10 LTSB 2016

OPERATING SYSTEM EDITION	KMS CLIENT PRODUCT KEY
Windows 10 Enterprise LTSB 2016	DCPHK-NFMTC-H88MJ-PFHPY-QJ4BJ
Windows 10 Enterprise N LTSB 2016	QFFDN-GRT3P-VKWWX-X7T3R-8B639

Windows 10 LTSB 2015

OPERATING SYSTEM EDITION	KMS CLIENT PRODUCT KEY
Windows 10 Enterprise 2015 LTSB	WNMTR-4C88C-JK8YV-HQ7T2-76DF9
Windows 10 Enterprise 2015 LTSB N	2F77B-TNFGY-69QQF-B8YKP-D69TJ

Earlier versions of Windows Server

Windows Server, version 1803

OPERATING SYSTEM EDITION	KMS CLIENT PRODUCT KEY
Windows Server Datacenter	2HXDN-KRXHB-GPYC7-YCKFJ-7FVDG
Windows Server Standard	PTXN8-JFHJM-4WC78-MPCBR-9W4KR

Windows Server, version 1709

OPERATING SYSTEM EDITION	KMS CLIENT PRODUCT KEY
Windows Server Datacenter	6Y6KB-N82V8-D8CQV-23MJW-BWTG6
Windows Server Standard	DPCNP-XQFKJ-BJF7R-FRC8D-GF6G4

Windows Server 2012 R2

OPERATING SYSTEM EDITION	KMS CLIENT PRODUCT KEY
Windows Server 2012 R2 Standard	D2N9P-3P6X9-2R39C-7RTCD-MDVJX
Windows Server 2012 R2 Datacenter	W3GGN-FT8W3-Y4M27-J84CP-Q3VJ9
Windows Server 2012 R2 Essentials	KNC87-3J2TX-XB4WP-VCPJV-M4FWM

Windows Server 2012

OPERATING SYSTEM EDITION	KMS CLIENT PRODUCT KEY
Windows Server 2012	BN3D2-R7TKB-3YPBD-8DRP2-27GG4
Windows Server 2012 N	8N2M2-HWPGY-7PGT9-HGDD8-GVGGY
Windows Server 2012 Single Language	2WN2H-YGCQR-KFX6K-CD6TF-84YXQ
Windows Server 2012 Country Specific	4K36P-JN4VD-GDC6V-KDT89-DYFKP
Windows Server 2012 Standard	XC9B7-NBPP2-83J2H-RHMBY-92BT4
Windows Server 2012 MultiPoint Standard	HM7DN-YVMH3-46JC3-XYTG7-CYQJJ
Windows Server 2012 MultiPoint Premium	XNH6W-2V9GX-RGJ4K-Y8X6F-QGJ2G
Windows Server 2012 Datacenter	48HP8-DN98B-MYWDG-T2DCC-8W83P

Windows Server 2008 R2

OPERATING SYSTEM EDITION	KMS CLIENT PRODUCT KEY
Windows Server 2008 R2 Web	6TPJF-RBVHG-WBW2R-86QPH-6RTM4
Windows Server 2008 R2 HPC edition	TT8MH-CG224-D3D7Q-498W2-9QCTX
Windows Server 2008 R2 Standard	YC6KT-GKW9T-YTKYR-T4X34-R7VHC

OPERATING SYSTEM EDITION	KMS CLIENT PRODUCT KEY
Windows Server 2008 R2 Enterprise	489J6-VHDMP-X63PK-3K798-CPX3Y
Windows Server 2008 R2 Datacenter	74YFP-3QFB3-KQT8W-PMXWJ-7M648
Windows Server 2008 R2 for Itanium-based Systems	GT63C-RJFQ3-4GMB6-BRFB9-CB83V

Windows Server 2008

OPERATING SYSTEM EDITION	KMS CLIENT PRODUCT KEY
Windows Web Server 2008	WYR28-R7TFJ-3X2YQ-YCY4H-M249D
Windows Server 2008 Standard	TM24T-X9RMF-VWXK6-X8JC9-BFGM2
Windows Server 2008 Standard without Hyper-V	W7VD6-7JFBR-RX26B-YKQ3Y-6FFFJ
Windows Server 2008 Enterprise	YQGMW-MPWTJ-34KDK-48M3W-X4Q6V
Windows Server 2008 Enterprise without Hyper-V	39BXF-X8Q23-P2WWT-38T2F-G3FPG
Windows Server 2008 HPC	RCTX3-KWVHP-BR6TB-RB6DM-6X7HP
Windows Server 2008 Datacenter	7M67G-PC374-GR742-YH8V4-TCBY3
Windows Server 2008 Datacenter without Hyper-V	22XQ2-VRXRG-P8D42-K34TD-G3QQC
Windows Server 2008 for Itanium-Based Systems	4DWFP-JF3DJ-B7DTH-78FJB-PDRHK

Earlier versions of Windows

Windows 8.1

OPERATING SYSTEM EDITION	KMS CLIENT PRODUCT KEY
Windows 8.1 Pro	GCRJD-8NW9H-F2CDX-CCM8D-9D6T9
Windows 8.1 Pro N	HMCNV-VVBFX-7HMBH-CTY9B-B4FXY
Windows 8.1 Enterprise	MHF9N-XY6XB-WVXMC-BTDCT-MKKG7
Windows 8.1 Enterprise N	TT4HM-HN7YT-62K67-RGRQJ-JFFXW

Windows 8

OPERATING SYSTEM EDITION	KMS CLIENT PRODUCT KEY
Windows 8 Pro	NG4HW-VH26C-733KW-K6F98-J8CK4
Windows 8 Pro N	XCVCF-2NXM9-723PB-MHCB7-2RYQQ
Windows 8 Enterprise	32JNW-9KQ84-P47T8-D8GGY-CWCK7
Windows 8 Enterprise N	JMNMF-RHW7P-DMY6X-RF3DR-X2BQT

OPERATING SYSTEM EDITION	KMS CLIENT PRODUCT KEY

Windows 7

OPERATING SYSTEM EDITION	KMS CLIENT PRODUCT KEY
Windows 7 Professional	FJ82H-XT6CR-J8D7P-XQJJ2-GPDD4
Windows 7 Professional N	MRPKT-YTG23-K7D7T-X2JMM-QY7MG
Windows 7 Professional E	W82YF-2Q76Y-63HXB-FGJG9-GF7QX
Windows 7 Enterprise	33PXH-7Y6KF-2VJC9-XBBR8-HVTHH
Windows 7 Enterprise N	YDRBP-3D83W-TY26F-D46B2-XCKRJ
Windows 7 Enterprise E	C29WB-22CC8-VJ326-GHFJW-H9DH4

How to use Extended Security Updates (ESU) for Windows Server

12/17/2021 • 7 minutes to read • Edit Online

Extended support for Windows Server 2008 and Windows Server 2008 R2 ended on January 14, 2020, and extended support for Windows Server 2012 and Windows Server 2012 R2 will be ending on October 10, 2023. Windows Server Long Term Servicing Channel (LTSC) has a minimum of ten years of support: five years for mainstream support and five years for extended support. This support includes regular security updates. You can find out information on support dates on Microsoft Lifecycle.

End of support also means the end of security updates. This scenario can cause security or compliance issues and put business applications at risk. Microsoft recommends that you upgrade to the current version of Windows Server for the most advanced security, performance, and innovation.

If you haven't already upgraded your servers, the following options will help protect your apps and data during the transition:

- Migrate existing Windows Server 2008 and 2008 R2 workloads as-is to Azure Virtual Machines (VMs).
  - This migration to Azure automatically provides an additional three years of Extended Security Updates. There's no additional charge for Extended Security Updates on top of an Azure VM's cost, and there's no additional configuration required.
- Purchase an extended security update subscription for your servers and remain protected until you're ready to upgrade to a newer Windows Server version.
  - These updates are provided for up to three years after the end of support lifecycle date.

After the three-year period of extended updates, we'll stop updating for Windows Server 2008 and 2008 R2. We recommend you update your version of Windows Server to a more recent version as soon as possible.

What are Extended Security Updates for Windows Server?

Extended Security Updates for Windows Server include security updates and bulletins rated **critical** and **important**, for a maximum of three years from the end of extended support dates (detailed on Microsoft Lifecycle). Extended Security Updates don't include the following:

- New features
- Customer-requested non-security hotfixes
- Design change requests

For more information, see Extended Security Updates frequently asked questions.

How to use Extended Security Updates

If you run Windows Server 2008 or 2008 R2 VMs in Azure, they're automatically enabled for Extended Security Updates. You don't need to configure anything, and there's no additional charge for using Extended Security Updates with Azure VMs. Extended Security Updates are automatically delivered to Azure VMs if they're configured to receive updates.

> **NOTE**
> Azure Classic VMs (Microsoft.ClassicCompute) require additional configuration to receive Extended Security Updates since they don't have access to the Azure Instance Metadata Service that determines Extended Security Updates eligibility. Please contact Microsoft support for more help.

For other environments, such as on-premises VMs or physical servers, you need to manually request and configure Extended Security Updates. You can purchase Extended Security Updates through Volume Licensing Programs such as Enterprise Agreement (EA), Enterprise Agreement Subscription (EAS), Enrollment for Education Solutions (EES), or Server and Cloud Enrollment (SCE).

Once you've purchased Extended Security Updates, you can use one of the following methods to get your keys:

- If you want to get Extended Security Update keys from the Azure portal, you can register for Extended Security Updates in the Azure portal.
- You can also sign in to the Microsoft Volume Licensing Service Center to get your keys without using the Azure portal.

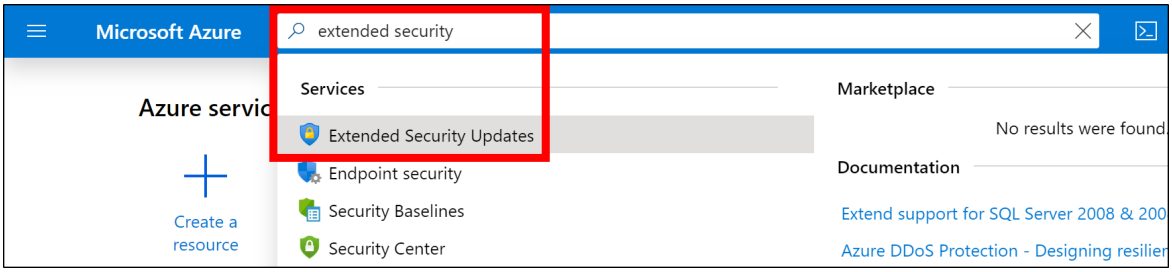
Register for Extended Security Updates in the Azure portal

To use Extended Security Updates on non-Azure VMs, create a multiple activation key (MAK) and apply it to Windows Server 2008 and 2008 R2 computers. The MAK key lets the Windows Update servers know that you can continue to receive security updates. You register for Extended Security Updates and manage these keys using the Azure portal, even if you only use on-premises computers.

> **NOTE**
> You don't need to register for Extended Security Updates if you're running Windows Server 2008 and 2008 R2 on Azure VMs. For other environments, such as on-premises VMs or physical servers, purchase Extended Security Updates before you try to register and use them.

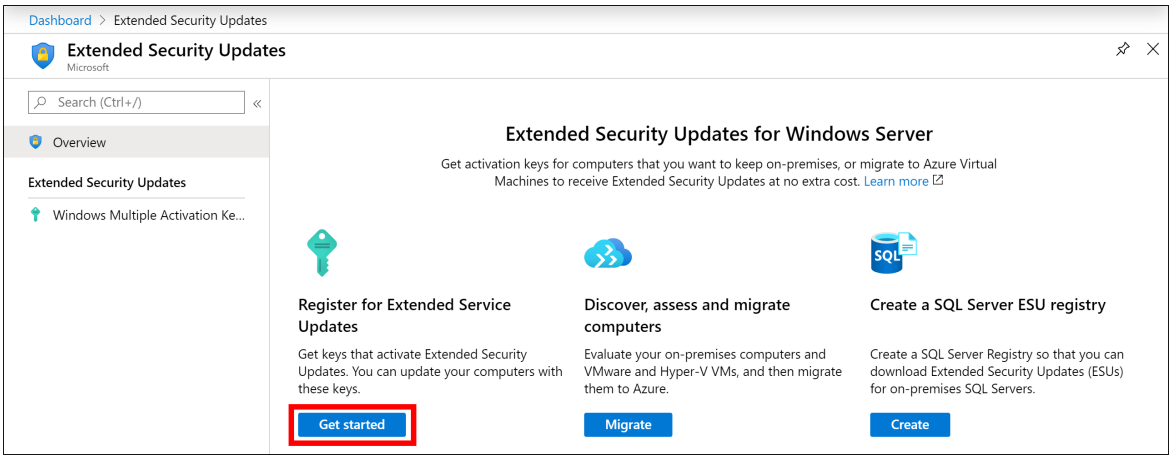
To register your VM for Extended Security Updates and create a key, open the Azure portal and follow these instructions:

1. Sign in to the Azure portal.
2. In the search box at the top of the Azure portal, search for and select **Extended Security Updates**.

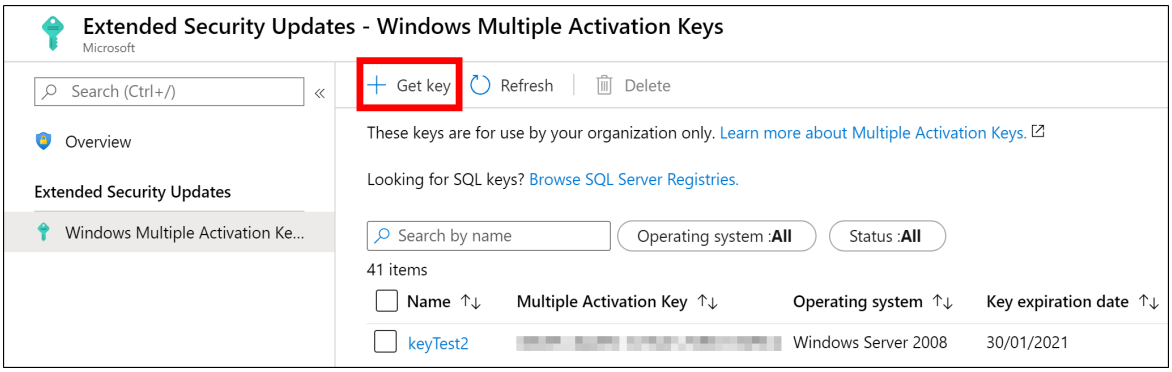


   If you haven't used Extended Security Updates before, select **+ Create** to create an Extended Security Updates resource first. Otherwise, select your resource from the list.

3. Under **Register for Extended Service Updates**, select **Get started**.



4. To create your first key, select **Get key**.



You need an Azure subscription associated with your account to create the Extended Security Update resource and key. If you don't have an Azure subscription associated with your account, sign in with a different user account or create an Azure subscription in the Azure portal.

Your Azure subscription must also be assigned the Contributor role for the security update to work. To check your role, enter "Subscriptions" into the search box. You'll see a table that will show you your role next to your subscription ID and name.

If you aren't a Contributor, you can ask the subscription owner to change your role. To find out who owns your subscription, go to the role table described in the previous paragraph and select your subscription's name. Next, go to the menu on the left side of the page and select **Access control (IAM)** > **Role assignments** and look for the **Owners** section in the table.

5. If you see a page titled **Register to get a Multiple Activation Key**, that means you need to request access to the preview before you can use Extended Security Updates. If you don't see this page, skip ahead to step 6.

   To request access, select **join the preview**. An email message window will open. This email is your access request to the Microsoft team.

   Include the following information in your request:

   - Customer name
   - Azure subscription ID
   - Agreement number (for ESU)
   - Number of ESU servers

   When you're done, send the email.

   The team will review the information you provide in your request email. If everything looks okay, they'll add you to the approved list.

If the team doesn't approve your request, you'll see the following error:

```
The resource type could not be found in the namespace 'Microsoft.WindowsESU'
```

6. Under **Azure details**, select your Azure subscription, a resource group, and location for your key.

   Under **Registration details**, enter the following information:

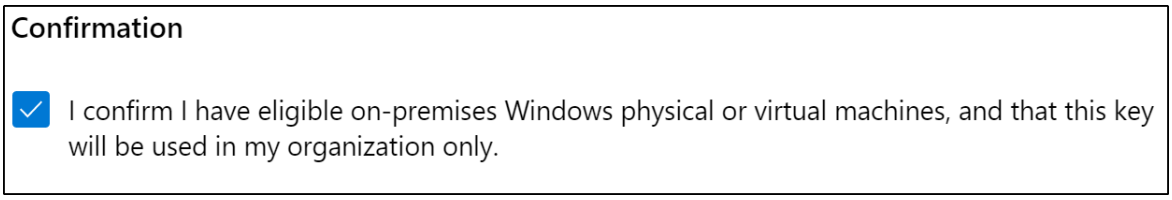
SETTING	VALUE
Key name	A display name for your key, such `Agreement01`.
Agreement number	Your agreement number generated by the volume licensing contract management system, or MSLicense for Enterprise Agreement programs.
Number of computers	Choose the number of computers on which you want to install Extended Security Updates with this key.
Operating system	Choose the operating system to use this key with, such as Windows Server 2008 or Windows Server 2008 R2.

   When ready, select **Review + register**.

> **NOTE**
> Make sure you've selected the Azure subscription that you joined the preview with in your global filter. Select the **Filter** button in the Azure Portal ribbon to check your global subscription filter.
>
> 

7. After successful validation, a summary of your choices for the new registry resource is shown. If needed, correct any validation errors or update your configuration choices. The Azure Terms of Use and Privacy Policy are available.

   Check the box to confirm that you have eligible computers and the key is only to be used within your organization:



   When ready, select **Create** to generate the Multiple Activation Key.

Extended Security Updates registration is now available for use with your servers. The key created should be applied to Windows Server 2008 and 2008 R2 servers that you wish to remain eligible for security updates.

Sign in to the Microsoft Volume Licensing Service Center

If you don't have access to the Azure portal, then you can use the Volume Licensing Service Center to view and download your activation keys.

To get your keys from the Volume Licensing Service Center:

1. Go to the Volume Licensing Service Center page and sign in with your Azure credentials.
2. Select **Licenses** > **Relationship Summary** > **Licensing ID** > **Product Keys**.

To learn more about how to get Extended Security Updates for eligible Windows devices, see our Tech Community post.

Download and apply Extended Security Updates

Delivery, download, and application of Extended Security Updates for Windows Server is no different than existing deployment processes. The updates provided through Extended Security Updates are only for *Security*.

You can install the updates using whatever tools and processes you already have in place. The only difference is that the system must be registered using the key generated in the previous section for the updates to download and install.

For Azure VMs, the process of enabling the computer for Extended Security Updates is automatically completed for you. Updates should download and install without additional configuration.

Install Nano Server

12/17/2021 • 4 minutes to read • Edit Online

Applies to: Windows Server 2016

IMPORTANT

Starting in Windows Server, version 1709, Nano Server will be available only as a container base OS image. Check out Changes to Nano Server to learn what this means.

Windows Server 2016 offers a new installation option: Nano Server. Nano Server is a remotely administered server operating system optimized for private clouds and datacenters. It is similar to Windows Server in Server Core mode, but significantly smaller, has no local logon capability, and only supports 64-bit applications, tools, and agents. It takes up far less disk space, sets up significantly faster, and requires far fewer updates and restarts than Windows Server. When it does restart, it restarts much faster. The Nano Server installation option is available for Standard and Datacenter editions of Windows Server 2016.

Nano Server is ideal for a number of scenarios:

- As a compute host for Hyper-V virtual machines, either in clusters or not
- As a storage host for Scale-Out File Server.
- As a DNS server
- As a web server running Internet Information Services (IIS)
- As a host for applications that are developed using cloud application patterns and run in a container or virtual machine guest operating system

Important differences in Nano Server

Because Nano Server is optimized as a lightweight operating system for running cloud-native applications based on containers and micro-services or as an agile and cost-effective datacenter host with a dramatically smaller footprint, there are important differences in Nano Server versus Server Core or Server with Desktop Experience installations:

- Nano Server is headless; there is no local logon capability or graphical user interface.
- Only 64-bit applications, tools, and agents are supported.
- Nano Server cannot serve as an Active Directory domain controller.
- Group Policy is not supported. However, you can use Desired State Configuration to apply settings at scale.
- Nano Server cannot be configured to use a proxy server to access the internet.
- NIC Teaming (specifically, load balancing and failover, or LBFO) is not supported. Switch-embedded teaming (SET) is supported instead.
- Microsoft Endpoint Configuration Manager and System Center Data Protection Manager are not supported.
- Best Practices Analyzer (BPA) cmdlets and BPA integration with Server Manager are not supported.
- Nano Server does not support virtual host bus adapters (HBAs).
- Nano Server does not need to be activated with a product key. When functioning as a Hyper-V host, Nano Server does not support Automatic Virtual Machine Activation (AVMA). Virtual machines running on a Nano Server host can be activated using Key Management Service (KMS) with a generic volume license key or

using Active Directory-based activation.

- The version of Windows PowerShell provided with Nano Server has important differences. For details, see PowerShell on Nano Server.
- Nano Server is supported only on the Current Branch for Business (CBB) model--there is no Long-Term Servicing Branch (LTSB) release for Nano Server at this time. See the following subsection for more information.

Current Branch for Business

Nano Server is serviced with a more active model, called Current Branch for Business (CBB), in order to support customers who are moving at a cloud cadence, using rapid development cycles. In this model, feature update releases of Nano Server are expected two to three times per year. This model requires Software Assurance for Nano Servers deployed and operated in production. To maintain support, administrators must stay no more than two CBB releases behind. However, these releases do not auto-update existing deployments; administrators perform manual installation of a new CBB release at their convenience. For some additional information, see Windows Server 2016 new Current Branch for Business servicing option.

The Server Core and Server with Desktop Experience installation options are still serviced on the Long-Term Servicing Branch (LTSB) model, comprising 5 years of mainstream support and 5 years of extended support.

Installation scenarios

Evaluation

You can obtain a 180-day-licensed evaluation copy of Windows Server from Windows Server Evaluations. To try out Nano Server, choose the **Nano Server | 64-bit EXE option**, and then come back to either Nano Server Quick Start or Deploy Nano Server to get started.

Clean installation

Because you install Nano Server by configuring a VHD, a clean installation is the quickest and simplest deployment method.

- To get started quickly with a basic deployment of Nano Server using DHCP to obtain an IP address, see the Nano Server Quick Start
- If you're already familiar with the basics of Nano Server, the more detailed topics starting with Deploy Nano Server offer a full set of instructions for customizing images, working with domains, installing packages for server roles and other features both online and offline, and much more.

> **IMPORTANT**
> Once Setup has completed and immediately after you have installed all of the server roles and features you need, check for and install updates available for Windows Server 2016. For Nano Server, see the Managing updates in Nano Server section of Manage Nano Server.

Upgrade

Since Nano Server is new for Windows Server 2016, there isn't an upgrade path from older operating system versions to Nano Server.

Migration

Since Nano Server is new for Windows Server 2016, there isn't migration path from older operating system versions to Nano Server.

If you need a different installation option, you can head back to the main Windows Server 2016 page

Changes to Nano Server in Windows Server Semi-Annual Channel

12/17/2021 • 2 minutes to read • Edit Online

> Applies to: Windows Server, Semi-Annual Channel

If you're already running Nano Server, the Window Server Semi-Annual Channel servicing model will be familiar, since it was formerly serviced by the Current Branch for Business (CBB) model. Windows Server Semi-Annual Channel is just a new name for the same model. In this model, feature update releases of Nano Server are expected two to three times per year.

However, starting with Windows Server, version 1803, Nano Server is available only as a **container base OS image**. You must run it as a container in a container host, such as a Server Core installation of Windows Server. Running a container based on Nano Server in this release differs from earlier releases in these ways:

- Nano Server has been optimized for .NET Core applications.
- Nano Server is even smaller than the Windows Server 2016 version.
- PowerShell Core, .NET Core, and WMI are no longer included by default, but you can include PowerShell Core and .NET Core container packages when building your container.
- There is no longer a servicing stack included in Nano Server. Microsoft publishes an updated Nano container to Docker Hub that you redeploy.
- You troubleshoot the new Nano Container by using Docker.
- You can now run Nano containers on IoT Core.

Nano Server Quick Start

12/17/2021 • 5 minutes to read • Edit Online

Applies to: Windows Server 2016

IMPORTANT

Starting in Windows Server, version 1709, Nano Server will be available only as a container base OS image. Check out Changes to Nano Server to learn what this means.

Follow the steps in this section to get started quickly with a basic deployment of Nano Server using DHCP to obtain an IP address. You can run a Nano Server VHD either in a virtual machine or boot to it on a physical computer; the steps are slightly different.

Once you've tried out the basics with these quick-start steps, you can find details of creating your own custom images, package management by several methods, domain operations, and much more in Deploy Nano Server.

Nano Server in a virtual machine

Follow these steps to create a Nano Server VHD that will run in a virtual machine.

To quickly deploy Nano Server in a virtual machine

1. Copy *NanoServerImageGenerator* folder from the \NanoServer folder in the Windows Server 2016 ISO to a folder on your hard drive.
2. Start Windows PowerShell as an administrator, change directory to the folder where you have placed the NanoServerImageGenerator folder and then import the module with
   `Import-Module .\NanoServerImageGenerator -Verbose`

   NOTE

   You might have to adjust the Windows PowerShell execution policy. `Set-ExecutionPolicy RemoteSigned` should work well.
3. Create a VHD for the Standard edition that sets a computer name and includes the Hyper-V **guest drivers** by running the following command which will prompt you for an administrator password for the new VHD:

   ```
   New-NanoServerImage -Edition Standard -DeploymentType Guest -MediaPath <path to root of media> -
   BasePath .\Base -TargetPath .\NanoServerVM\NanoServerVM.vhd -ComputerName <computer name>
   ```

   where

   - **-MediaPath <path to root of media>** specifies a path to the root of the contents of the Windows Server 2016 ISO. For example if you have copied the contents of the ISO to d:\TP5ISO you would use that path.
   - **-BasePath** (optional) specifies a folder that will be created to copy the Nano Server WIM and packages to.
   - **-TargetPath** specifies a path, including the filename and extension, where the resulting VHD or VHDX will be created.

- **Computer_name** specifies the computer name that the Nano Server virtual machine you are creating will have.

Example:

```
New-NanoServerImage -Edition Standard -DeploymentType Guest -MediaPath f:\ -BasePath .\Base -
TargetPath .\Nano1\Nano.vhd -ComputerName Nano1
```

This example creates a VHD from an ISO mounted as f:\. When creating the VHD it will use a folder called Base in the same directory where you ran New-NanoServerImage; it will place the VHD (called Nano.vhd) in a folder called Nano1 in the folder from where the command is run. The computer name will be Nano1. The resulting VHD will contain the Standard edition of Windows Server 2016 and will be suitable for Hyper-V virtual machine deployment. If you want a Generation 1 virtual machine, create a VHD image by specifying a **.vhd** extension for -TargetPath. For a Generation 2 virtual machine, create a VHDX image by specifying a **.vhdx** extension for -TargetPath. You can also directly generate a WIM file by specifying a **.wim** extension for -TargetPath.

> **NOTE**
> New-NanoServerImage is supported on Windows 8.1, Windows 10, Windows Server 2012 R2, and Windows Server 2016.

4. In Hyper-V Manager, create a new virtual machine and use the VHD created in Step 3.
5. Boot the virtual machine and in Hyper-V Manager connect to the virtual machine.
6. Log on to the Recovery Console (see the Nano Server Recovery Console section in this guide), using the administrator and password you supplied while running the script in Step 3.

> **NOTE**
> The Recovery Console only supports basic keyboard functions. Keyboard lights, 10-key sections, and keyboard layout switching such as caps lock and number lock are not supported.

7. Obtain the IP address of the Nano Server virtual machine and use Windows PowerShell remoting or other remote management tool to connect to and remotely manage the virtual machine.

Nano Server on a physical computer

You can also create a VHD that will run Nano Server on a physical computer, using the pre-installed device drivers. If your hardware requires a driver that is not already provided in order to boot or connect to a network, follow the steps in the Adding Additional Drivers section of this guide.

To quickly deploy Nano Server on a physical computer

1. Copy *NanoServerImageGenerator* folder from the \NanoServer folder in the Windows Server 2016 ISO to a folder on your hard drive.
2. Start Windows PowerShell as an administrator, change directory to the folder where you have placed the NanoServerImageGenerator folder and then import the module with
   `Import-Module .\NanoServerImageGenerator -Verbose`

> **NOTE**
> You might have to adjust the Windows PowerShell execution policy. `Set-ExecutionPolicy RemoteSigned` should work well.

3. Create a VHD that sets a computer name and includes the OEM drivers and Hyper-V by running the following command which will prompt you for an administrator password for the new VHD:

```
New-NanoServerImage -Edition Standard -DeploymentType Host -MediaPath <path to root of media> -
BasePath .\Base -TargetPath .\NanoServerPhysical\NanoServer.vhd -ComputerName <computer name> -
OEMDrivers -Compute -Clustering
```

   where

   - **-MediaPath <path to root of media>** specifies a path to the root of the contents of the Windows Server 2016 ISO. For example if you have copied the contents of the ISO to d:\TP5ISO you would use that path.
   - **BasePath** specifies a folder that will be created to copy the Nano Server WIM and packages to. (This parameter is optional.)
   - **TargetPath** specifies a path, including the filename and extension, where the resulting VHD or VHDX will be created.
   - **Computer_name** is the computer name for the Nano Server you are creating.

   Example:

```
New-NanoServerImage -Edition Standard -DeploymentType Host -MediaPath F:\ -BasePath .\Base -TargetPath
.\Nano1\NanoServer.vhd -ComputerName Nano-srv1 -OEMDrivers -Compute -Clustering
```

   This example creates a VHD from an ISO mounted as F:\. When creating the VHD it will use a folder called Base in the same directory where you ran New-NanoServerImage; it will place the VHD in a folder called Nano1 in the folder from where the command is run. The computer name will be Nano-srv1 and will have OEM drivers installed for most common hardware and has the Hyper-V role and clustering feature enabled. The Standard Nano edition is used.

4. Log in as an administrator on the physical server where you want to run the Nano Server VHD.
5. Copy the VHD that this script creates to the physical computer and configure it to boot from this new VHD. To do that, follow these steps:

   a. Mount the generated VHD. In this example, it's mounted under D:\.

   b. Run **bcdboot d:\windows**.

   c. Unmount the VHD.

6. Boot the physical computer into the Nano Server VHD.
7. Log on to the Recovery Console (see the Nano Server Recovery Console section in this guide), using the administrator and password you supplied while running the script in Step 3.

   > **NOTE**
   > The Recovery Console only supports basic keyboard functions. Keyboard lights, 10-key sections, and keyboard layout switching such as caps lock and number lock are not supported.

8. Obtain the IP address of the Nano Server computer and use Windows PowerShell remoting or other remote management tool to connect to and remotely manage the virtual machine.

Deploy Nano Server

12/17/2021 • 28 minutes to read • Edit Online

Applies to: Windows Server 2016

IMPORTANT

Starting in Windows Server, version 1709, Nano Server will be available only as a container base OS image. Check out Changes to Nano Server to learn what this means.

This topic covers information you need to deploy Nano Server images that are more customized to your needs compared to the simple examples in the Nano Server Quick Start topic. You'll find information about making a custom Nano Server image with exactly the features you want, installing Nano Server images from VHD or WIM, editing files, working with domains, dealing with packages by several methods, and working with server roles.

Nano Server Image Builder

The Nano Server Image Builder is a tool that helps you create a custom Nano Server image and bootable USB media with the aid of a graphical interface. Based on the inputs you provide, it generates reusable PowerShell scripts that allow you easily automate consistent installations of Nano Server running either Windows Server 2016 Datacenter or Standard editions.

Obtain the tool from the Download Center.

The tool also requires Windows Assessment and Deployment Kit (ADK).

Nano Server Image Builder creates customized Nano Server images in VHD, VHDX, or ISO formats and can create bootable USB media to deploy Nano server or detect the hardware configuration of a server. It also can do the following:

- Accept the license terms
- Create VHD, VHDX or ISO formats
- Add server roles
- Add device drivers
- Set machine name, administrator password, logfile path, and timezone
- Join a domain by using an existing Active Directory account or a harvested domain-join blob
- Enable WinRM for communication outside the local subnet
- Enable Virtual LAN IDs and configure static IP addresses
- Inject new servicing packages on the fly
- Add a setupcomplete.cmd or other customer scripts to run after the unattend.xml is processed
- Enable Emergency Management Services (EMS) for serial port console access
- Enable development services to enable test signed drivers and unsigned applications, PowerShell default shell
- Enable debugging over serial, USB, TCP/IP, or IEEE 1394 protocols
- Create USB media using WinPE that will partition the server and install the Nano image
- Create USB media using WinPE that will detect your existing Nano Server hardware configuration and report all the details in a log and on-screen. This includes network adapters, MAC addresses, and firmware Type

(BIOS or UEFI). The detection process will also list all of the volumes on the system and the devices that do not have a driver included in the Server Core drivers package.

If any of these are unfamiliar to you, review the remainder of this topic and the other Nano Server topics so that you'll be prepared to provide the tool with the information it will need.

Creating a custom Nano Server image

For Windows Server 2016, Nano Server is distributed on the physical media, where you will find a **NanoServer** folder; this contains a .wim image and a sub-folder called **Packages**. It is these package files that you use to add server roles and features to the VHD image, which you then boot to.

You can also find and install these packages with the NanoServerPackage provider of PackageManagement (OneGet) PowerShell module. See the Installing roles and features online section of this topic.

This table shows the roles and features that are available in this release of Nano Server, along with the Windows PowerShell options that will install the packages for them. Some packages are installed directly with their own Windows PowerShell switches (such as -Compute); others you install by passing package names to the -Package parameter, which you can combine in a comma-separated list. You can dynamically list available packages using the Get-NanoServerPackage cmdlet.

ROLE OR FEATURE	OPTION
Hyper-V role (including NetQoS)	-Compute
Failover Clustering and other components, detailed after this table	-Clustering
Basic drivers for a variety of network adapters and storage controllers. This is the same set of drivers included in a Server Core installation of Windows Server 2016.	-OEMDrivers
File Server role and other storage components, detailed after this table	-Storage
Windows Defender, including a default signature file	-Defender
Reverse forwarders for application compatibility, for example common application frameworks such as Ruby, Node.js, etc.	Now included by default
DNS Server role	-Package Microsoft-NanoServer-DNS-Package
PowerShell Desired State Configuration (DSC)	-Package Microsoft-NanoServer-DSC-Package **Note:** For full details, see Using DSC on Nano Server.
Internet Information Server (IIS)	-Package Microsoft-NanoServer-IIS-Package **Note:** See IIS on Nano Server for details about working with IIS.
Host support for Windows Containers	-Containers

ROLE OR FEATURE	OPTION
System Center Virtual Machine Manager agent	-Package Microsoft-NanoServer-SCVMM-Package -Package Microsoft-NanoServer-SCVMM-Compute-Package **Note:** Use the SCVMM Compute package only if you are monitoring Hyper-V. For hyper-converged deployments in VMM, you should also specify the -Storage parameter. For more details, see the VMM documentation.
System Center Operations Manager agent	Installed separately. See the System Center Operations Manager documentation for more details at https://technet.microsoft.com/system-center-docs/om/manage/install-agent-on-nano-server.
Data Center Bridging (including DCBQoS)	-Package Microsoft-NanoServer-DCB-Package
Deploying on a virtual machine	-Package Microsoft-NanoServer-Guest-Package
Deploying on a physical machine	- Package Microsoft-NanoServer-Host-Package
BitLocker, trusted platform module (TPM), volume encryption, platform identification, cryptography providers, and other functionality related to secure startup	-Package Microsoft-NanoServer-SecureStartup-Package
Hyper-V support for Shielded VMs	-Package Microsoft-NanoServer-ShieldedVM-Package **Note:** This package is only available for the Datacenter edition of Nano Server.
Simple Network Management Protocol (SNMP) agent	-Package Microsoft-NanoServer-SNMP-Agent-Package.cab **Note:** Not included with Windows Server 2016 installation media. Available online only. See Installing roles and features online for details.
IPHelper service which provides tunnel connectivity using IPv6 transition technologies (6to4, ISATAP, Port Proxy, and Teredo), and IP-HTTPS	-Package Microsoft-NanoServer-IPHelper-Service-Package.cab **Note:** Not included with Windows Server 2016 installation media. Available online only. See Installing roles and features online for details.

NOTE

When you install packages with these options, a corresponding language pack is also installed based on selected server media locale. You can find the available language packs and their locale abbreviations in the installation media in sub-folders named for the locale of the image.

NOTE

When you use the -Storage parameter to install File Services, File Services is not actually enabled. Enable this feature from a remote computer with Server Manager.

Failover Clustering items installed by the -Clustering parameter

- Failover Clustering role
- VM Failover Clustering
- Storage Spaces Direct (S2D)
- Storage Quality of Service
- Volume Replication Clustering
- SMB Witness Service

File and storage items installed by the -Storage parameter

- File Server role
- Data Deduplication
- Multipath I/O, including a driver for Microsoft Device-Specific Module (MSDSM)
- ReFS (v1 and v2)
- iSCSI Initiator (but not iSCSI Target)
- Storage Replica
- Storage Management Service with SMI-S support
- SMB Witness Service
- Dynamic Volumes
- Basic Windows storage providers (for Windows Storage Management)

Installing a Nano Server VHD

This example creates a GPT-based VHDX image with a given computer name and including Hyper-V guest drivers, starting with Nano Server installation media on a network share. In an elevated Windows PowerShell prompt, start with this cmdlet:

```
Import-Module <Server media location>\NanoServer\NanoServerImageGenerator; New-NanoServerImage -
DeploymentType Guest -Edition Standard -MediaPath \\Path\To\Media\server_en-us -BasePath .\Base -TargetPath
.\FirstStepsNano.vhdx -ComputerName FirstStepsNano
```

The cmdlet will accomplish all of these tasks:

1. Select Standard as a base edition
2. Prompt you for the Administrator password
3. Copy installation media from \\Path\To\Media\server_en-us into .\Base
4. Convert the WIM image to a VHD. (The file extension of the target path argument determines whether it creates an MBR-based VHD for Generation 1 virtual machines versus a GPT-based VHDX for Generation 2 virtual machines.)
5. Copy the resulting VHD into .\FirstStepsNano.vhdx
6. Set the Administrator password for the image as specified
7. Set the computer name of the image to FirstStepsNano
8. Install the Hyper-V guest drivers

All of this results in an image of .\FirstStepsNano.vhdx.

The cmdlet generates a log as it runs and will let you know where this log is located once it is finished. The WIM-to-VHD conversion accomplished by the companion script generates its own log in %TEMP%\Convert-WindowsImage\<GUID> (where <GUID> is a unique identifier per conversion session).

As long as you use the same base path, you can omit the media path parameter every time you run this cmdlet, since it will use cached files from the base path. If you don't specify a base path, the cmdlet will generate a default one in the TEMP folder. If you want to use different source media, but the same base path, you should specify the media path parameter, however.

> **NOTE**
> You now have the option to specify the Nano Server edition to build either the Standard or Datacenter edition. Use the -Edition parameter to specify *Standard* or *Datacenter* editions.

Once you have an existing image, you can modify it as needed using the *Edit-NanoServerImage* cmdlet.

If you do not specify a computer name, a random name will be generated.

Installing a Nano Server WIM

1. Copy the *NanoServerImageGenerator* folder from the \NanoServer folder in the Windows Server 2016 ISO a local folder on your computer.
2. Start Windows PowerShell as an administrator, change directory to the folder where you placed the NanoServerImageGenerator folder and then import the module with `Import-Module .\NanoServerImageGenerator -Verbose`.

   > **NOTE**
   > You might have to adjust the Windows PowerShell execution policy. `Set-ExecutionPolicy RemoteSigned` should work well.

To create a Nano Server image to serve as a Hyper-V host, run the following:

```
New-NanoServerImage -Edition Standard -DeploymentType Host -MediaPath <path to root of media> -BasePath
.\Base -TargetPath .\NanoServerPhysical\NanoServer.wim -ComputerName <computer name> -OEMDrivers -Compute -
Clustering`
```

Where

- MediaPath is the root of the DVD media or ISO image containing Windows Server 2016 .
- `-BasePath` will contain a copy of the Nano Server binaries, so you can use New-NanoServerImage -BasePath without having to specify -MediaPath in future runs.
- `-TargetPath` will contain the resulting .wim file containing the roles & features you selected. Make sure to specify the .wim extension.
- `-Compute` adds the Hyper-V role.
- `-OemDrivers` adds a number of common drivers.

You will be prompted to enter an administrator password.

For more information, run `Get-Help New-NanoServerImage -Full`.

Boot into WinPE and ensure that the .wim file just created is accessible from WinPE. (You could, for example, copy the .wim file to a bootable WinPE image on a USB flash drive.)

Once WinPE boots, use Diskpart.exe to prepare the target computer's hard drive. Run the following Diskpart commands (modify accordingly, if you're not using UEFI & GPT):

> **WARNING**
> These commands will delete all data on the hard drive:

```
Diskpart.exe
Select disk 0
Clean
Convert GPT
Create partition efi size=100
Format quick FS=FAT32 label=System
Assign letter=s
Create partition msr size=128
Create partition primary
Format quick FS=NTFS label=NanoServer
Assign letter=n
List volume
Exit
```

Apply the Nano Server image (adjust the path of the .wim file):

```
Dism.exe /apply-image /imagefile:.\NanoServer.wim /index:1 /applydir:n:\
Bcdboot.exe n:\Windows /s s:
```

Remove the DVD media or USB drive and reboot your system with **Wpeutil.exe Reboot**

Editing files on Nano Server locally and remotely

In either case, connect to Nano Server, such as with Windows PowerShell remoting.

Once you've connected to Nano Server, you can edit a file residing on your local computer by passing the file's relative or absolute path to the psEdit command, for example: `psEdit C:\Windows\Logs\DISM\dism.log` or `psEdit .\myScript.ps1`

Edit a file residing on the remote Nano Server by starting a remote session with `Enter-PSSession -ComputerName 192.168.0.100 -Credential ~\Administrator` and then passing the file's relative or absolute path to the psEdit command like this: `psEdit C:\Windows\Logs\DISM\dism.log`

Installing roles and features online

> **NOTE**
> If you install an optional Nano Server package from media or online repository, it won't have recent security fixes included. To avoid a version mismatch between the optional packages and base operating system, you should install the latest cumulative update immediately after installing any optional packages and **before** restarting the server.

Installing roles and features from a package repository

You can find and install Nano Server packages from the online package repository by using the NanoServerPackage provider of the PackageManagement PowerShell module. To install this provider, use these cmdlets:

```
Install-PackageProvider NanoServerPackage
Import-PackageProvider NanoServerPackage
```

> **NOTE**
> If you experience errors when running Install-PackageProvider, check that you have installed the latest cumulative update (KB3206632 or later), or use Save-Module as follows:

```
Save-Module -Path $Env:ProgramFiles\WindowsPowerShell\Modules\ -Name NanoServerPackage -MinimumVersion
1.0.1.0
Import-PackageProvider NanoServerPackage
```

Once this provider is installed and imported, you can search for, download, and install Nano Server packages using cmdlets designed specifically for working with Nano Server packages:

```
Find-NanoServerPackage
Save-NanoServerPackage
Install-NanoServerPackage
```

You can also use the generic PackageManagement cmdlets and specify the NanoServerPackage provider:

```
Find-Package -ProviderName NanoServerPackage
Save-Package -ProviderName NanoServerPackage
Install-Package -ProviderName NanoServerPackage
Get-Package -ProviderName NanoServerPackage
```

To use any of these cmdlets with Nano Server packages on Nano Server, add `-ProviderName NanoServerPackage`. If you don't add the -ProviderName parameter, PackageManagement will iterate all of the providers. For more details on these cmdlets, run `Get-Help <cmdlet>`. Here are some common usage examples:

Searching for Nano Server packages

You can use either `Find-NanoServerPackage` or `Find-Package -ProviderName NanoServerPackage` to search for and return a list of Nano Server packages that are available in the online repository. For example, you can get a list of all the latest packages:

```
Find-NanoServerPackage
```

Running `Find-Package -ProviderName NanoServerPackage -DisplayCulture` displays all available cultures.

If you need a specific locale version, such as US English, you could use `Find-NanoServerPackage -Culture en-us` or `Find-Package -ProviderName NanoServerPackage -Culture en-us` or `Find-Package -Culture en-us -DisplayCulture`.

To find a specific package by package name, use the -Name parameter. This parameter also accepts wildcards. For example, to find all packages with VMM in the name, use `Find-NanoServerPackage -Name *VMM*` or `Find-Package -ProviderName NanoServerPackage -Name *VMM*`.

You can find a particular version with the -RequiredVersion, -MinimumVersion, or -MaximumVersion parameters. To find all available versions, use -AllVersions. Otherwise, only the latest version is returned. For example: `Find-NanoServerPackage -Name *VMM* -RequiredVersion 10.0.14393.0`. Or, for all versions: `Find-Package -ProviderName NanoServerPackage -Name *VMM* -AllVersions`

Installing Nano Server packages

You can install a Nano Server package (including its dependency packages, if any) to Nano Server either locally or an offline image with either `Install-NanoServerPackage` or `Install-Package -ProviderName NanoServerPackage`.

Both of these accept input from the pipeline.

To install the latest version of a Nano Server package to an online Nano Server, use either `Install-NanoServerPackage -Name Microsoft-NanoServer-Containers-Package` or `Install-Package -Name Microsoft-NanoServer-Containers-Package`. PackageManagement will use the culture of the Nano Server.

You can install a Nano Server package to an offline image while specifying a particular version and culture, like this:

```
Install-NanoServerPackage -Name Microsoft-NanoServer-DCB-Package -Culture de-de -RequiredVersion 10.0.14393.0
-ToVhd C:\MyNanoVhd.vhd
```

or:

```
Install-Package -Name Microsoft-NanoServer-DCB-Package -Culture de-de -RequiredVersion 10.0.14393.0 -ToVhd
C:\MyNanoVhd.vhd
```

Here are some examples of pipelining package search results to the installation cmdlet:

`Find-NanoServerPackage *dcb* | Install-NanoServerPackage` finds any packages with dcb in the name and then installs them.

`Find-Package *nanoserver-compute-* | Install-Package` finds packages with nanoserver-compute- in the name and installs them.

`Find-NanoServerPackage -Name *nanoserver-compute* | Install-NanoServerPackage -ToVhd C:\MyNanoVhd.vhd` finds packages with compute in the name and installs them to an offline image.

`Find-Package -ProviderName NanoserverPackage *nanoserver-compute-* | Install-Package -ToVhd C:\MyNanoVhd.vhd` does the same thing with any package that has nanoserver-compute- in the name.

Downloading Nano Server packages

`Save-NanoServerPackage` or `Save-Package` allow you to download packages and save them without installing them. Both cmdlets accept input from the pipeline.

For example, to download and save a Nano Server package to a directory that matches the wildcard path, use `Save-NanoServerPackage -Name Microsoft-NanoServer-DNS-Package -Path C:\` In this example, -Culture wasn't specified, so the culture of the local machine will be used. No version was specified, so the latest version will be saved.

```
Save-Package -ProviderName NanoServerPackage -Name Microsoft-NanoServer-IIS-Package -Path C:\ -Culture it-IT
-MinimumVersion 10.0.14393.0
```

saves a particular version and for the Italian language and locale.

You can send search results through the pipeline as in these examples:

```
Find-NanoServerPackage -Name *containers* -MaximumVersion 10.2 -MinimumVersion 1.0 -Culture es-ES | Save-
NanoServerPackage -Path C:\
```

or

```
Find-Package -ProviderName NanoServerPackage -Name *shield* -Culture es-ES | Save-Package -Path
```

Inventory installed packages

You can discover which Nano Server packages are installed with `Get-Package`. For example, see which packages are on Nano Server with `Get-Package -ProviderName NanoserverPackage`.

To check the Nano Server packages that are installed in an offline image, run `Get-Package -ProviderName NanoserverPackage -FromVhd C:\MyNanoVhd.vhd`.

Installing roles and features from local source

Though offline installation of server roles and other packages is recommended, you might need to install them online (with the Nano Server running) in container scenarios. To do this, follow these steps:

1. Copy the Packages folder from the installation media locally to the running Nano Server (for example, to C:\packages).

2. Create a new Unattend.xml file on another computer and then copy it to Nano Server. You can copy and paste this XML content into the XML file you created (this example shows installing the IIS package):

```
<?xml version=1.0 encoding=utf-8?>
    <unattend xmlns=urn:schemas-microsoft-com:unattend>
    <servicing>
        <package action=install>
            <assemblyIdentity name=Microsoft-NanoServer-IIS-Feature-Package version=10.0.14393.0
processorArchitecture=amd64 publicKeyToken=31bf3856ad364e35 language=neutral />
            <source location=c:\packages\Microsoft-NanoServer-IIS-Package.cab />
        </package>
        <package action=install>
            <assemblyIdentity name=Microsoft-NanoServer-IIS-Feature-Package version=10.0.14393.0
processorArchitecture=amd64 publicKeyToken=31bf3856ad364e35 language=en-US />
            <source location=c:\packages\en-us\Microsoft-NanoServer-IIS-Package_en-us.cab />
        </package>
    </servicing>
    <cpi:offlineImage cpi:source= xmlns:cpi=urn:schemas-microsoft-com:cpi />
</unattend>
```

3. In the new XML file you created (or copied), edit C:\packages to the directory you copied the content of Packages to.

4. Switch to the directory with the newly created XML file and run:
   `dism /online /apply-unattend:.\unattend.xml`

5. Confirm that the package and its associated language pack is installed correctly by running:
   `dism /online /get-packages`

   You should see Package Identity : Microsoft-NanoServer-IIS-Package~31bf3856ad364e35~amd64~en-US~10.0.10586.0 listed twice, once for Release Type : Language Pack and once for Release Type : Feature Pack.

Customizing an existing Nano Server VHD

You can change the details of an existing VHD by using the Edit-NanoServerImage cmdlet, as in this example:
`Edit-NanoServerImage -BasePath .\Base -TargetPath .\BYOVHD.vhd`

This cmdlet does the same things as New-NanoServerImage, but changes the existing image instead of converting a WIM to a VHD. It supports the same parameters as New-NanoServerImage with the exception of -MediaPath and -MaxSize, so the initial VHD must have been created with those parameters before you can make changes with Edit-NanoServerImage.

Additional tasks you can accomplish with New-NanoServerImage and Edit-NanoServerImage

Joining domains

New-NanoServerImage offers two methods of joining a domain; both rely on offline domain provisioning, but one harvests a blob to accomplish the join. In this example, the cmdlet harvests a domain blob for the Contoso domain from the local computer (which of course must be part of the Contoso domain), then it performs offline provisioning of the image using the blob:

```
New-NanoServerImage -Edition Standard -DeploymentType Host -MediaPath \\Path\To\Media\en_us -BasePath .\Base
-TargetPath .\JoinDomHarvest.vhdx -ComputerName JoinDomHarvest -DomainName Contoso
```

When this cmdlet completes, you should find a computer named JoinDomHarvest in the Active Directory computer list.

You can also use this cmdlet on a computer that is not joined to a domain. To do this, harvest a blob from any computer that is joined to the domain, and then provide the blob to the cmdlet yourself. Note that when you harvest such a blob from another computer, the blob already includes that computer's name--so if you try to add the *-ComputerName* parameter, an error will result.

You can harvest the blob with this command:

```
djoin
/Provision
/Domain Contoso
/Machine JoiningDomainsNoHarvest
/SaveFile JoiningDomainsNoHarvest.djoin
```

Run New-NanoServerImage using the harvested blob:

```
New-NanoServerImage -DeploymentType Host -Edition Standard -MediaPath \\Path\To\Media\en_us -BasePath .\Base
-TargetPath .\JoinDomNoHrvest.vhd -DomainBlobPath .\Path\To\Domain\Blob\JoinDomNoHrvestContoso.djoin
```

In the event that you already have a node in the domain with the same computer name as your future Nano Server, you could reuse the computer name by adding the `-ReuseDomainNode` parameter.

Adding additional drivers

Nano Server offers a package that includes a set of basic drivers for a variety of network adapters and storage controllers; it's possible that drivers for your network adapters might not be included. You can use these steps to find drivers in a working system, extract them, and then add them to the Nano Server image.

1. Install Windows Server 2016 on the physical computer where you will run Nano Server.
2. Open Device Manager and identify devices in the following categories:
3. Network adapters
4. Storage controllers
5. Disk drives
6. For each device in these categories, right-click the device name, and click **Properties**. In the dialog that opens, click the **Driver** tab, and then click **Driver Details**.
7. Note the filename and path of the driver file that appears. For example, let's say the driver file is e1i63x64.sys, which is in C:\Windows\System32\Drivers.
8. In a command prompt, search for the driver file and search for all instances with dir e1i*.sys /s /b. In this example, the driver file is also present in the path C:\Windows\System32\DriverStore\FileRepository\net1ic64.inf_amd64_fafa7441408bbecd\e1i63x64.sys.
9. In an elevated command prompt, navigate to the directory where the Nano Server VHD is and run the following commands:

```
md mountdir
dism\dism /Mount-Image /ImageFile:.\NanoServer.vhd /Index:1 /MountDir:.\mountdir
dism\dism /Add-Driver /image:.\mountdir /driver:
C:\Windows\System32\DriverStore\FileRepository\net1ic64.inf_amd64_fafa7441408bbecd
dism\dism /Unmount-Image /MountDir:.\MountDir /Commit
```

10. Repeat these steps for each driver file you need.

> **NOTE**
> In the folder where you keep your drivers, both the SYS files and corresponding INF files must be present. Also, Nano Server only supports signed, 64-bit drivers.

Injecting drivers

Nano Server offers a package that includes a set of basic drivers for a variety of network adapters and storage controllers; it's possible that drivers for your network adapters might not be included. You can use this syntax to have New-NanoServerImage search the directory for available drivers and inject them into the Nano Server image:

```
New-NanoServerImage -DeploymentType Host -Edition Standard -MediaPath \\Path\To\Media\en_us -BasePath .\Base
-TargetPath .\InjectingDrivers.vhdx -DriverPath .\Extra\Drivers
```

> **NOTE**
> In the folder where you keep your drivers, both the SYS files and corresponding INF files must be present. Also, Nano Server only supports signed, 64-bit drivers.

Using the -DriverPath parameter, you can also pass a array of paths to driver .inf files:

```
New-NanoServerImage -DeploymentType Host -Edition Standard -MediaPath \\Path\To\Media\en_us -BasePath .\Base
-TargetPath .\InjectingDrivers.vhdx -DriverPath .\Extra\Drivers\netcard64.inf
```

Connecting with WinRM

To be able to connect to a Nano Server computer using Windows Remote Management (WinRM) (from another computer that is not on the same subnet), open port 5985 for inbound TCP traffic on the Nano Server image. Use this cmdlet:

```
New-NanoServerImage -DeploymentType Host -Edition Standard -MediaPath \\Path\To\Media\en_us -BasePath .\Base
-TargetPath .\ConnectingOverWinRM.vhd -EnableRemoteManagementPort
```

Setting static IP addresses

To configure a Nano Server image to use static IP addresses, first find the name or index of the interface you want to modify by using Get-NetAdapter, netsh, or the Nano Server Recovery Console. Use the -Ipv6Address, -Ipv6Dns, -Ipv4Address, -Ipv4SubnetMask, -Ipv4Gateway and -Ipv4Dns parameters to specify the configuration, as in this example:

```
New-NanoServerImage -DeploymentType Host -Edition Standard -MediaPath \\Path\To\Media\en_us -BasePath .\Base
-TargetPath .\StaticIpv4.vhd -InterfaceNameOrIndex Ethernet -Ipv4Address 192.168.1.2 -Ipv4SubnetMask
255.255.255.0 -Ipv4Gateway 192.168.1.1 -Ipv4Dns 192.168.1.1
```

Custom image size

You can configure the Nano Server image to be a dynamically expanding VHD or VHDX with the -MaxSize parameter, as in this example:

```
New-NanoServerImage -DeploymentType Host -Edition Standard -MediaPath \\Path\To\Media\en_us -BasePath .\Base
-TargetPath .\BigBoss.vhd -MaxSize 100GB
```

Embedding custom data

To embed your own script or binaries in the Nano Server image, use the -CopyPath parameter to pass an array of files and directories to be copied. The -CopyPath parameter can also accept a hashtable to specify the destination path for files and directories.

```
New-NanoServerImage -DeploymentType Host -Edition Standard -MediaPath \\Path\To\Media\en_us -BasePath .\Base
-TargetPath .\BigBoss.vhd -CopyPath .\tools
```

Running custom commands after the first boot

To run custom commands as part of setupcomplete.cmd, use the -SetupCompleteCommand parameter to pass an array of commands:

```
New-NanoServerImage -DeploymentType Host -Edition Standard -MediaPath \\Path\To\Media\en_us -BasePath .\Base
-TargetPath .\NanoServer.wim -SetupCompleteCommand @(echo foo, echo bar)
```

Running custom PowerShell scripts as part of image creation

To run custom PowerShell scripts as part of the image creation process, use the -OfflineScriptPath parameter to pass an array of paths to .ps1 scripts. If those scripts take arguments, use the -OfflineScriptArgument to pass a hashtable of additional arguments to the scripts.

```
New-NanoServerImage -DeploymentType Host -Edition Standard -MediaPath \\Path\To\Media\en_us -BasePath .\Base
-TargetPath .\NanoServer.wim -OfflineScriptPath C:\MyScripts\custom.ps1 -OfflineScriptArgument
@{Param1=Value1; Param2=Value2}
```

Support for development scenarios

If you want to develop and test on Nano Server, you can use the -Development parameter. This will enable PowerShell as the default local shell, enable installation of unsigned drivers, copy debugger binaries, open a port for debugging, enable test signing, and enable installation of AppX packages without a developer license:

```
New-NanoServerImage -DeploymentType Guest -Edition Standard -MediaPath \\Path\To\Media\en_us -BasePath .\Base
-TargetPath .\NanoServer.wim -Development
```

Custom unattend file

If you want to use your own unattend file, use the -UnattendPath parameter:

```
New-NanoServerImage -DeploymentType Guest -Edition Standard -MediaPath \\Path\To\Media\en_us -BasePath .\Base
-TargetPath .\NanoServer.wim -UnattendPath \\path\to\unattend.xml
```

Specifying an administrator password or computer name in this unattend file will override the values set by -AdministratorPassword and -ComputerName.

> **NOTE**
>
> Nano Server does not support setting TCP/IP settings via unattend files. You can use Setupcomplete.cmd to configure TCP/IP settings.

Collecting log files

If you want to collect the log files during image creation, use the -LogPath parameter to specify a directory where all the log files are copied.

```
New-NanoServerImage -DeploymentType Guest -Edition Standard -MediaPath \\Path\To\Media\en_us -BasePath .\Base
-TargetPath .\NanoServer.wim -LogPath C:\Logs
```

> **NOTE**
>
> Some parameters on New-NanoServerImage and Edit-NanoServerImage are for internal use only and can be safely ignored. These include the -SetupUI and -Internal parameters.

Windows Server App installer

Windows Server App (WSA) installer provides a reliable installation option for Nano Server. Since Windows Installer (MSI) is not supported on Nano Server, WSA is also the only installation technology available for non-Microsoft products. WSA leverages Windows app package technology designed to install and service applications safely and reliably, using a declarative manifest. It extends the Windows app package installer to support Windows Server-specific extensions, with the limitation that WSA does not support installing drivers.

Creating and installing a WSA package on Nano Server involves steps for both the publisher and the consumer of the package.

The package publisher should do the following:

1. Install Windows 10 SDK, which includes the tools needed to create a WSA package: MakeAppx, MakeCert, Pvk2Pfx, SignTool.
2. Declare a manifest: Follow the WSA manifest extension schema to create the manifest file, AppxManifest.xml.
3. Use the **MakeAppx** tool to create a WSA package.
4. Use **MakeCert** and **Pvk2Pfx** tools to create the certificate, and then use **Signtool** to sign the package.

Next, the package consumer should follow these steps:

1. Run the *Import-Certificate* PowerShell cmdlet to import the publisher's certificate from Step 4 above to Nano Server with the certStoreLocation at Cert:\LocalMachine\TrustedPeople. For example:
   `Import-Certificate -FilePath .\xyz.cer -CertStoreLocation Cert:\LocalMachine\TrustedPeople`
2. Install the app on Nano Server by running the **Add-AppxPackage** PowerShell cmdlet to install a WSA package on Nano Server. For example: `Add-AppxPackage wsaSample.appx`

Additional resources for creating apps

WSA is server extension of Windows app package technology (though it is not hosted in Microsoft Store). If you want to publish apps with WSA,these topics will help you familiarize yourself with the app package pipeline:

- How to create a basic package manifest
- App Packager (MakeAppx.exe)
- How to create an app package signing certificate
- SignTool

Installing drivers on Nano Server

You can install non-Microsoft drivers on Nano Server by using INF driver packages. These include both Plug-and-Play (PnP) driver packages and File System Filter driver packages. Network Filter drivers are not currently supported on Nano Server.

Both PnP and File System Filter driver packages must follow the Universal driver requirements and installation process, as well as general driver package guidelines such as signing. They are documented at these locations:

- Driver Signing
- Using a Universal INF File

Installing driver packages offline

Supported driver packages can be installed on Nano Server offline via DISM.exe or DISM PowerShell cmdlets.

Installing driver packages online

PnP driver packages can be installed to Nano Server online by using PnpUtil. Online driver installation for non-PnP driver packages is not currently supported on Nano Server.

Joining Nano Server to a domain

To add Nano Server to a domain online

1. Harvest a data blob from a computer in the domain that is already running Windows Threshold Server using this command:

   ```
   djoin.exe /provision /domain <domain-name> /machine <machine-name> /savefile .\odjblob
   ```

   This saves the data blob in a file called odjblob.

2. Copy the odjblob file to the Nano Server computer with these commands:

   net use z: \\<ip address of Nano Server>\c$

NOTE

If the net use command fails, you probably need to adjust Windows Firewall rules. To do this, first open an elevated command prompt, start Windows PowerShell and then connect to the Nano Server computer with Windows PowerShell Remoting with these commands:

```
Set-Item WSMan:\localhost\Client\TrustedHosts <IP address of Nano Server>
```

```
$ip = <ip address of Nano Server>
```

```
Enter-PSSession -ComputerName $ip -Credential $ip\Administrator
```

When prompted, provide the Administrator password, then run this command to set the firewall rule:

netsh advfirewall firewall set rule group=File and Printer Sharing new enable=yes

Exit Windows PowerShell with `Exit-PSSession`, and then retry the net use command. If successful, continue copying the odjblob file contents to the Nano Server.

md z:\Temp

copy odjblob z:\Temp

3. Check the domain you want to join Nano Server to and ensure that DNS is configured. Also, verify that name resolution of the domain or a domain controller works as expected. To do this, open an elevated command prompt, start Windows PowerShell and then connect to the Nano Server computer with Windows PowerShell remoting with these commands:

   ```
   Set-Item WSMan:\localhost\Client\TrustedHosts <IP address of Nano Server>
   ```

   ```
   $ip = <ip address of Nano Server>
   ```

   ```
   Enter-PSSession -ComputerName $ip -Credential $ip\Administrator
   ```

   When prompted, provide the Administrator password. Nslookup is not available on Nano Server, so you can verify name resolution with Resolve-DNSName.

4. If name resolution succeeds, then in the same Windows PowerShell session, run this command to join the domain:

   ```
   djoin /requestodj /loadfile c:\Temp\odjblob /windowspath c:\windows /localos
   ```

5. Restart the Nano Server computer, and then exit the Windows PowerShell session:

   ```
   shutdown /r /t 5
   ```

   ```
   Exit-PSSession
   ```

6. After you have joined Nano Server to a domain, add the domain user account to the Administrators group on the Nano Server.

7. For security, remove the Nano Server from the trusted hosts list with this command:
   `Set-Item WSMan:\localhost\client\TrustedHosts`

Alternate method to join a domain in one step

First, harvest the data blob from another computer running Windows Threshold Server that is already in your domain using this command:

```
djoin.exe /provision /domain <domain-name> /machine <machine-name> /savefile .\odjblob
```

Open the file odjblob (perhaps in Notepad), copy its contents, and then paste the contents into the

<AccountData> section of the Unattend.xml file below.

Put this Unattend.xml file into the C:\NanoServer folder, and then use the following commands to mount the VHD and apply the settings in the `offlineServicing` section:

```
dism\dism /Mount-ImagemediaFile:.\NanoServer.vhd /Index:1 /MountDir:.\mountdir
dism\dismmedia:.\mountdir /Apply-Unattend:.\unattend.xml
```

Create a Panther folder (used by Windows systems for storing files during setup; see Windows 7, Windows Server 2008 R2, and Windows Vista setup log file locations if you're curious), copy the Unattend.xml file to it, and then unmount the VHD with these commands:

```
md .\mountdir\windows\panther
copy .\unattend.xml .\mountdir\windows\panther
dism\dism /Unmount-Image /MountDir:.\mountdir /Commit
```

The first time you boot Nano Server from this VHD, the other settings will be applied.

After you have joined Nano Server to a domain, add the domain user account to the Administrators group on the Nano Server.

Working with server roles on Nano Server

Using Hyper-V on Nano Server

Hyper-V works the same on Nano Server as it does on Windows Server in Server Core mode, with two exceptions:

- You must perform all management remotely and the management computer must be running the same build of Windows Server as the Nano Server. Older versions of Hyper-V Manager or Hyper-V Windows PowerShell cmdlets will not work.
- RemoteFX is not available.

In this release, these features of Hyper-V have been verified:

- Enabling Hyper-V
- Creation of Generation 1 and Generation 2 virtual machines
- Creation of virtual switches
- Starting virtual machines and running Windows guest operating systems
- Hyper-V Replica

If you want to perform a live migration of virtual machines, create a virtual machine on an SMB share, or connect resources on an existing SMB share to an existing virtual machine, it is vital that you configure authentication correctly. You have two options for doing this:

Constrained delegation

Constrained delegation works exactly the same as in previous releases. Refer to these articles for more information:

- Enabling Hyper-V Remote Management - Configuring Constrained Delegation For SMB and Highly Available SMB
- Enabling Hyper-V Remote Management - Configuring Constrained Delegation For Non-Clustered Live

Migration

CredSSP

First, refer to the Using Windows PowerShell remoting section of this topic to enable and test CredSSP. Then, on the management computer, you can use Hyper-V Manager and select the option to connect as another user. Hyper-V Manager will use CredSSP. You should do this even if you are using your current account.

Windows PowerShell cmdlets for Hyper-V can use CimSession or Credential parameters, either of which work with CredSSP.

Using Failover Clustering on Nano Server

Failover clustering works the same on Nano Server as it does on Windows Server in Server Core mode, but keep these caveats in mind:

- Clusters must be managed remotely with Failover Cluster Manager or Windows PowerShell.
- All Nano Server cluster nodes must be joined to the same domain, similar to cluster nodes in Windows Server.
- The domain account must have Administrator privileges on all Nano Server nodes, as with cluster nodes in Windows Server.
- All commands must be run in an elevated command prompt.

> **NOTE**
> Additionally, certain features are not supported in this release:
>
> - You cannot run failover clustering cmdlets on a local Nano Server through Windows PowerShell.
> - Clustering roles other than Hyper-V and File Server.

You'll find these Windows PowerShell cmdlets useful in managing Failover clusters:

You can create a new cluster with `New-Cluster -Name <clustername> -Node <comma-separated cluster node list>`

Once you've established a new cluster, you should run `Set-StorageSetting -NewDiskPolicy OfflineShared` on all nodes.

Add an additional node to the cluster with
`Add-ClusterNode -Name <comma-separated cluster node list> -Cluster <clustername>`

Remove a node from the cluster with
`Remove-ClusterNode -Name <comma-separated cluster node list> -Cluster <clustername>`

Create a Scale-Out File Server with `Add-ClusterScaleoutFileServerRole -name <sofsname> -cluster <clustername>`

You can find additional cmdlets for failover clustering at Microsoft.FailoverClusters.PowerShell.

Using DNS Server on Nano Server

To provide Nano Server with the DNS Server role, add the Microsoft-NanoServer-DNS-Package to the image (see the Creating a custom Nano Server image section of this topic. Once the Nano Server is running, connect to it and run this command from and elevated Windows PowerShell console to enable the feature:

```
Enable-WindowsOptionalFeature -Online -FeatureName DNS-Server-Full-Role
```

Using IIS on Nano Server

For steps to use the Internet Information Services (IIS) role, see IIS on Nano Server.

Using MPIO on Nano Server

For steps to use MPIO, see MPIO on Nano Server

Using SSH on Nano Server

For instructions on how to install and use SSH on Nano Server with the OpenSSH project, see the Win32-OpenSSH wiki.

Appendix: Sample Unattend.xml file that joins Nano Server to a domain

> **NOTE**
> Be sure to delete the trailing space in the contents of odjblob once you paste it into the Unattend file.

```
<?xml version='1.0' encoding='utf-8'?>
<unattend xmlns=urn:schemas-microsoft-com:unattend
xmlns:wcm=https://schemas.microsoft.com/WMIConfig/2002/State xmlns:xsi=http://www.w3.org/2001/XMLSchema-
instance>

  <settings pass=offlineServicing>
    <component name=Microsoft-Windows-UnattendedJoin processorArchitecture=amd64
publicKeyToken=31bf3856ad364e35 language=neutral versionScope=nonSxS>
        <OfflineIdentification>
           <Provisioning>
             <AccountData>
AAAAAAARUABLEABLEABAoAAAAAAAMABSUABLEABLEABAwAAAAAAAAABbMAAdYABc8ABYkABLAABbMAAEAAAAMAAA0ABY4ABZ8ABbIABa0AAc
IABY4ABb8ABZUABAsAAAAAAQAAZoABNUABOYABZYAANQABMoAAOEAAMIAAOkAANoAAMAAAXwAAJAAAAYAAA0ABY4ABZ8ABbIABa0AAcIABY
4ABb8ABZUABLEAALMABLQABU0AATMABXAAAAAAAKdf/mhfXoAAUAAAQAAAAb8ABLQABbMABcMABb4ABc8ABAIAAAAAAb8ABLQABbMABcMABb
4ABc8ABLQABb0ABZIAAGAAAAsAAR4ABTQABUAAAAAAACAAAQwABZMAAZcAAUgABVcAAegAARcABKkABVIAASwAAY4ABbcABW8ABQoAAT0ABN
8AAO8ABekAAJMAAVkAAZUABckABXEABJUAAQ8AAJ4AAIsABZMABdoAAOsABIsABKkABQEABUEABIwABKoAAaABXgABNwAAegAAAkAAAAABA
MABLIABdIABc8ABY4AADAAAA4AAZ4ABbQABcAAAAAAACAAkKBW0ID8nJDWYAHnBAXE77j7BAEWEkl+lKB98XC2G0/9+Wd1DJQW4IYAkKBAAD
hAnKBWEwhiDAAAM2zzDCEAM6IAAAgAAAAAAQAAAAAAAAAAABwzzAAA
              </AccountData>
           </Provisioning>
         </OfflineIdentification>
    </component>
  </settings>

  <settings pass=oobeSystem>
    <component name=Microsoft-Windows-Shell-Setup processorArchitecture=amd64
publicKeyToken=31bf3856ad364e35 language=neutral versionScope=nonSxS>
      <UserAccounts>
        <AdministratorPassword>
           <Value>Tuva</Value>
           <PlainText>true</PlainText>
        </AdministratorPassword>
      </UserAccounts>
      <TimeZone>Pacific Standard Time</TimeZone>
    </component>
  </settings>

  <settings pass=specialize>
    <component name=Microsoft-Windows-Shell-Setup processorArchitecture=amd64
publicKeyToken=31bf3856ad364e35 language=neutral versionScope=nonSxS>
      <RegisteredOwner>My Team</RegisteredOwner>
      <RegisteredOrganization>My Corporation</RegisteredOrganization>
    </component>
  </settings>
</unattend>
```

IIS on Nano Server

12/17/2021 • 10 minutes to read • Edit Online

Applies to: Windows Server 2016

IMPORTANT

Starting in Windows Server, version 1709, Nano Server will be available only as a container base OS image. Check out Changes to Nano Server to learn what this means.

You can install the Internet Information Services (IIS) server role on Nano Server by using the -Package parameter with Microsoft-NanoServer-IIS-Package. For information about configuring Nano Server, including installing packages, see Install Nano Server.

In this release of Nano Server, the following IIS features are available:

FEATURE	ENABLED BY DEFAULT
Common HTTP Features	
Default document	x
Directory browsing	x
HTTP Errors	x
Static content	x
HTTP redirection	
Health and Diagnostics	
HTTP logging	x
Custom logging	
Request monitor	
Tracing	
Performance	
Static content compression	x
Dynamic content compression	
Security	

FEATURE	ENABLED BY DEFAULT
Request filtering	x
Basic authentication	
Client certificate mapping authentication	
Digest authentication	
IIS client certificate mapping authentication	
IP and domain restrictions	
URL authorization	
Windows authentication	
Application Development	
Application initialization	
CGI	
ISAPI extensions	
ISAPI filters	
Server-side includes	
WebSocket protocol	
Management Tools	
IISAdministration module for Windows PowerShell	x

A series of articles on other configurations of IIS (such as using ASP.NET, PHP, and Java) and other related content is published at https://iis.net/learn.

Installing IIS on Nano Server

You can install this server role either offline (with the Nano Server off) or online (with the Nano Server running); offline installation is the recommended option.

For offline installation, add the package with the -Packages parameter of New-NanoServerImage, as in this example:

```
New-NanoServerImage -Edition Standard -DeploymentType Guest -MediaPath f:\ -BasePath .\Base -TargetPath
.\Nano1.vhd -ComputerName Nano1 -Package Microsoft-NanoServer-IIS-Package
```

If you have an existing VHD file, you can install IIS offline with DISM.exe by mounting the VHD, and then using the **Add-Package** option. The following example steps assume that you are running from the directory specified by BasePath option, which was created after running New-NanoServerImage.

1. mkdir mountdir
2. .\Tools\dism.exe /Mount-Image /ImageFile:.\NanoServer.vhd /Index:1 /MountDir:.\mountdir
3. .\Tools\dism.exe /Add-Package /PackagePath:.\packages\Microsoft-NanoServer-IIS-Package.cab /Image:.\mountdir
4. .\Tools\dism.exe /Add-Package /PackagePath:.\packages\en-us\Microsoft-NanoServer-IIS-Package_en-us.cab /Image:.\mountdir
5. .\Tools\dism.exe /Unmount-Image /MountDir:.\MountDir /Commit

> **NOTE**
> Note that Step 4 adds the language pack--this example installs EN-US.

At this point, you can start Nano Server with IIS.

Installing IIS on Nano Server online

Though offline installation of the server role is recommended, you might need to install it online (with the Nano Server running) in container scenarios. To do this, follow these steps:

1. Copy the Packages folder from the installation media locally to the running Nano Server (for example, to C:\packages).

2. Create a new Unattend.xml file on another computer and then copy it to the Nano Server. You can copy and paste this XML content into the XML file you created:

```
    <unattend xmlns=urn:schemas-microsoft-com:unattend>
    <servicing>
        <package action=install>
            <assemblyIdentity name=Microsoft-NanoServer-IIS-Package version=10.0.14393.0
processorArchitecture=amd64 publicKeyToken=31bf3856ad364e35 language=neutral />
            <source location=c:\packages\Microsoft-NanoServer-IIS-Package.cab />
        </package>
        <package action=install>
            <assemblyIdentity name=Microsoft-NanoServer-IIS-Package version=10.0.14393.0
processorArchitecture=amd64 publicKeyToken=31bf3856ad364e35 language=en-US />
            <source location=c:\packages\en-us\Microsoft-NanoServer-IIS-Package_en-us.cab />
        </package>
    </servicing>
    <cpi:offlineImage cpi:source= xmlns:cpi=urn:schemas-microsoft-com:cpi />
</unattend>
```

3. In the new XML file you created (or copied), edit C:\packages to the directory you copied the content of Packages to.

4. Switch to the directory with the newly created XML file and run

   dism /online /apply-unattend:.\unattend.xml

5. Confirm that the IIS package and its associated language pack are installed correctly by running:

   dism /online /get-packages

   You should see Package Identity : Microsoft-NanoServer-IIS-Package~31bf3856ad364e35~amd64~~10.0.14393.1000 listed twice, once for Release Type : Language Pack and once for Release Type : Feature Pack.

6. Start the W3SVC service either with **net start w3svc** or by restarting the Nano Server.

Starting IIS

Once IIS is installed and running, it is ready to serve web requests. Verify that IIS is running by browsing the default IIS web page at http://<IP address of Nano Server>. On a physical computer, you can determine the IP address by using the Recovery Console. On a virtual machine, you can get the IP address by using a Windows PowerShell prompt and running:

```
Get-VM -name <VM name> | Select -ExpandProperty networkadapters | select IPAddresses
```

If you are not able to access the default IIS web page, double-check the IIS installation by looking for the **c:\inetpub** directory on the Nano Server.

Enabling and disabling IIS features

A number of IIS features are enabled by default when you install the IIS role (see the table in the Overview of IIS on Nano Server section of this topic). You can enable (or disable) additional features using DISM.exe

Each feature of IIS exists as a set of configuration elements. For example, the Windows authentication feature comprises these elements:

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=WindowsAuthenticationModule image=%windir%\System32\inetsrv\authsspi.dll`
`<modules>`	`<add name=WindowsAuthenticationModule lockItem=true \/>`
`<windowsAuthentication>`	`<windowsAuthentication enabled=false authPersistNonNTLM\=true><providers><add value=Negotiate /><add value=NTLM />  </providers> </windowsAuthentication>`

The full set of IIS sub-features is included in Appendix 1 of this topic and their corresponding configuration elements is included in Appendix 2 of this topic.

Example: installing Windows authentication

1. Open a Windows PowerShell remote session console on the Nano Server.
2. Use `DISM.exe` to install the Windows authentication module:

   ```
   dism /Enable-Feature /online /featurename:IIS-WindowsAuthentication /all
   ```

   The `/all` switch will install any feature that the chosen feature depends on.

Example: uninstalling Windows authentication

1. Open a Windows PowerShell remote session console on the Nano Server.
2. Use `DISM.exe` to uninstall the Windows authentication module:

   ```
   dism /Disable-Feature /online /featurename:IIS-WindowsAuthentication
   ```

Other common IIS configuration tasks

Creating websites

Use this cmdlet:

```
PS D:\> New-IISSite -Name TestSite -BindingInformation *:80:TestSite -PhysicalPath c:\test
```

You can then run `Get-IISSite` to verify the state of the site (returns the web site name, ID, state, physical path, and bindings).

Deleting web sites

Run `Remove-IISSite -Name TestSite -Confirm:$false`.

Creating virtual directories

You can create virtual directories by using the IISServerManager object returned by Get-IISServerManager, which exposes the .NET Microsoft.Web.Administration.ServerManager API. In this example, these commands access the Default Web Site element of the Sites collection and the root application element (/) of the Applications section. They then call the Add() method of the VirtualDirectories collection for that application element to create the new directory:

```
PS C:\> $sm = Get-IISServerManager
PS C:\> $sm.Sites["Default Web Site"].Applications[/].VirtualDirectories.Add(/DemoVirtualDir1,
c:\test\virtualDirectory1)
PS C:\> $sm.Sites["Default Web Site"].Applications[/].VirtualDirectories.Add(/DemoVirtualDir2,
c:\test\virtualDirectory2)
PS C:\> $sm.CommitChanges()
```

Creating application pools

Similarly you can use Get-IISServerManager to create application pools:

```
PS C:\> $sm = Get-IISServerManager
PS C:\> $sm.ApplicationPools.Add(DemoAppPool)
```

Configuring HTTPS and certificates

Use the Certoc.exe utility to import certificates, as in this example, which shows configuring HTTPS for a website on a Nano Server:

1. On another computer that is not running Nano Server, create a certificate (using your own certificate name and password), and then export it to c:\temp\test.pfx.

   `$newCert = New-SelfSignedCertificate -DnsName www.foo.bar.com -CertStoreLocation cert:\LocalMachine\my`

   `$mypwd = ConvertTo-SecureString -String YOUR_PFX_PASSWD -Force -AsPlainText`

   `Export-PfxCertificate -FilePath c:\temp\test.pfx -Cert $newCert -Password $mypwd`

2. Copy the test.pfx file to the Nano Server computer.

3. On the Nano Server, import the certificate to the My store with this command:

   certoc.exe -ImportPFX -p YOUR_PFX_PASSWD My c:\temp\test.pfx

4. Retrieve the thumbprint of this new certificate (in this example, 61E71251294B2A7BB8259C2AC5CF7BA622777E73) with `Get-ChildItem Cert:\LocalMachine\my`.

5. Add the HTTPS binding to the Default Web Site (or whatever website you want to add the binding to) by using these Windows PowerShell commands:

```
$certificate = get-item Cert:\LocalMachine\my\61E71251294B2A7BB8259C2AC5CF7BA622777E73
# Use your actual thumbprint instead of this example
$hash = $certificate.GetCertHash()

Import-Module IISAdministration
$sm = Get-IISServerManager
$sm.Sites["Default Web Site"].Bindings.Add("*:443:", $hash, "My", "0")    # My is the certificate
store name
$sm.CommitChanges()
```

You could also use Server Name Indication (SNI) with a specific host name with this syntax:

```
$sm.Sites["Default Web Site"].Bindings.Add("*:443:www.foo.bar.com", $hash, "My", "SNI")
```

Appendix 1: List of IIS sub-features

- IIS-WebServer
- IIS-CommonHttpFeatures
- IIS-StaticContent
- IIS-DefaultDocument
- IIS-DirectoryBrowsing
- IIS-HttpErrors
- IIS-HttpRedirect
- IIS-ApplicationDevelopment
- IIS-CGI
- IIS-ISAPIExtensions
- IIS-ISAPIFilter
- IIS-ServerSideIncludes
- IIS-WebSockets
- IIS-ApplicationInit
- IIS-Security
- IIS-BasicAuthentication
- IIS-WindowsAuthentication
- IIS-DigestAuthentication
- IIS-ClientCertificateMappingAuthentication
- IIS-IISCertificateMappingAuthentication
- IIS-URLAuthorization
- IIS-RequestFiltering
- IIS-IPSecurity
- IIS-CertProvider
- IIS-Performance
- IIS-HttpCompressionStatic
- IIS-HttpCompressionDynamic
- IIS-HealthAndDiagnostics
- IIS-HttpLogging
- IIS-LoggingLibraries
- IIS-RequestMonitor
- IIS-HttpTracing
- IIS-CustomLogging

Appendix 2: Elements of HTTP features

Each feature of IIS exists as a set of configuration elements. This appendix lists the configuration elements for all of the features in this release of Nano Server

Common HTTP features

Default document

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=DefaultDocumentModule image=%windir%\System32\inetsrv\defdoc.dll />`
`<modules>`	`<add name=DefaultDocumentModule lockItem=true />`
`<handlers>`	`<add name=StaticFile path=* verb=* modules=DefaultDocumentModule resourceType=EiSecther requireAccess=Read />`
`<defaultDocument>`	`<defaultDocument enabled=true> <files>  <add value=Default.htm />  <add value=Default.asp />  <add value=index.htm />   <add value=index.html />  <add value=iisstart.htm />  </files>  </defaultDocument>`

The `StaticFile <handlers>` entry might already be present; if so, just add DefaultDocumentModule to the <modules> attribute, separated by a comma.

Directory browsing

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=DirectoryListingModule image=%windir%\System32\inetsrv\dirlist.dll />`
`<modules>`	`<add name=DirectoryListingModule lockItem=true />`
`<handlers>`	`<add name=StaticFile path=* verb=* modules=DirectoryListingModule resourceType=Either requireAccess=Read />`

The `StaticFile <handlers>` entry might already be present; if so, just add DirectoryListingModule to the <modules> attribute, separated by a comma.

HTTP errors

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=CustomErrorModule image=%windir%\System32\inetsrv\custerr.dll />`
`<modules>`	`<add name=CustomErrorModule lockItem=true />`

SECTION	CONFIGURATION ELEMENTS
`<httpErrors>`	`<httpErrors lockAttributes=allowAbsolutePathsWhenDelegated,defaultPath> <error statusCode=401 prefixLanguageFilePath=%SystemDrive%\inetpub\custerr path=401.htm > <error statusCode=403 prefixLanguageFilePath=%SystemDrive%\inetpub\custerr path=403.htm /> <error statusCode=404 prefixLanguageFilePath=%SystemDrive%\inetpub\custerr path=404.htm /> <error statusCode=405 prefixLanguageFilePath=%SystemDrive%\inetpub\custerr path=405.htm /> <error statusCode=406 prefixLanguageFilePath=%SystemDrive%\inetpub\custerr path=406.htm /> <error statusCode=412 prefixLanguageFilePath=%SystemDrive%\inetpub\custerr path=412.htm /> <error statusCode=500 prefixLanguageFilePath=%SystemDrive%\inetpub\custerr path=500.htm /> <error statusCode=501 prefixLanguageFilePath=%SystemDrive%\inetpub\custerr path=501.htm /> <error statusCode=502 prefixLanguageFilePath=%SystemDrive%\inetpub\custerr path=502.htm /> </httpErrors>`

Static content

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=StaticFileModule image=%windir%\System32\inetsrv\static.dll />`
`<modules>`	`<add name=StaticFileModule lockItem=true />`
`<handlers>`	`<add name=StaticFile path=* verb=* modules=StaticFileModule resourceType=Either requireAccess=Read />`

The `StaticFile \<handlers>` entry might already be present; if so, just add StaticFileModule to the <modules> attribute, separated by a comma.

HTTP redirection

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=HttpRedirectionModule image=%windir%\System32\inetsrv\redirect.dll />`
`<modules>`	`<add name=HttpRedirectionModule lockItem=true />`
`<httpRedirect>`	`<httpRedirect enabled=false />`

Health and diagnostics

HTTP logging

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=HttpLoggingModule image=%windir%\System32\inetsrv\loghttp.dll />`
`<modules>`	`<add name=HttpLoggingModule lockItem=true />`

SECTION	CONFIGURATION ELEMENTS
`<httpLogging>`	`<httpLogging dontLog=false />`

Custom logging

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=CustomLoggingModule image=%windir%\System32\inetsrv\logcust.dll />`
`<modules>`	`<add name=CustomLoggingModule lockItem=true />`

Request monitor

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=RequestMonitorModule image=%windir%\System32\inetsrv\iisreqs.dll />`

Tracing

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=TracingModule image=%windir%\System32\inetsrv\iisetw.dll \/> <add name=FailedRequestsTracingModule image=%windir%\System32\inetsrv\iisfreb.dll />`
`<modules>`	`<add name=FailedRequestsTracingModule lockItem=true />`
`<traceProviderDefinitions>`	`<traceProviderDefinitions> <add name=WWW Server guid\={3a2a4e84-4c21-4981-ae10-3fda0d9b0f83}> <areas> <clear /> <add name=Authentication value=2 /> <add name=Security value=4 /> <add name=Filter value=8 /> <add name=StaticFile value=16 /> <add name=CGI value=32 /> <add name=Compression value=64 /> <add name=Cache value=128 /> <add name=RequestNotifications value=256 /> <add name=Module value=512 /> <add name=FastCGI value=4096 /> <add name=WebSocket value=16384 /> </areas> </add> <add name=ISAPI Extension guid={a1c2040e-8840-4c31-ba11-9871031a19ea}> <areas> <clear /> </areas> </add> </traceProviderDefinitions>`

Performance

Static content compression

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=StaticCompressionModule image=%windir%\System32\inetsrv\compstat.dll />`
`<modules>`	`<add name=StaticCompressionModule lockItem=true />`

SECTION	CONFIGURATION ELEMENTS
`<httpCompression>`	`<httpCompression directory=%SystemDrive%\inetpub\temp\IIS Temporary Compressed Files> <scheme name=gzip dll=%Windir%\system32\inetsrv\gzip.dll /> <staticTypes> <add mimeType=text/* enabled=true /> <add mimeType=message/* enabled=true /> <add mimeType=application/javascript enabled=true \/> <add mimeType=application/atom+xml enabled=true /> <add mimeType=application/xaml+xml enabled=true /> <add mimeType=** enabled=false /> </staticTypes> </httpCompression>`

Dynamic content compression

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=DynamicCompressionModule image=%windir%\System32\inetsrv\compdyn.dll />`
`<modules>`	`<add name=DynamicCompressionModule lockItem=true />`
`<httpCompression>`	`<httpCompression directory\=%SystemDrive%\inetpub\temp\IIS Temporary Compressed Files> <scheme name=gzip dll=%Windir%\system32\inetsrv\gzip.dll \/> \ <dynamicTypes> <add mimeType=text/* enabled=true \/> <add mimeType=message/* enabled=true /> <add mimeType=application/x-javascript enabled=true /> <add mimeType=application/javascript enabled=true /> <add mimeType=*/* enabled=false /> <\/dynamicTypes> </httpCompression>`

Security

Request filtering

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=RequestFilteringModule image=%windir%\System32\inetsrv\modrqflt.dll />`
`<modules>`	`<add name=RequestFilteringModule lockItem=true />`
`<requestFiltering>`	`<requestFiltering>  <fileExtensions allowUnlisted=true applyToWebDAV=true />  <verbs allowUnlisted=true applyToWebDAV=true />  <hiddenSegments applyToWebDAV=true>  <add segment=web.config />  </hiddenSegments>  </requestFiltering>`

Basic authentication

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=BasicAuthenticationModule image=%windir%\System32\inetsrv\authbas.dll />`
`<modules>`	`<add name=WindowsAuthenticationModule lockItem=true />`

SECTION	CONFIGURATION ELEMENTS
`<basicAuthentication>`	`<basicAuthentication enabled=false />`

Client certificate mapping authentication

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=CertificateMappingAuthentication image=%windir%\System32\inetsrv\authcert.dll />`
`<modules>`	`<add name=CertificateMappingAuthenticationModule lockItem=true />`
`<clientCertificateMappingAuthentication>`	`<clientCertificateMappingAuthentication enabled=false />`

Digest authentication

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=DigestAuthenticationModule image=%windir%\System32\inetsrv\authmd5.dll />`
`<modules>`	`<add name=DigestAuthenticationModule lockItem=true />`
`<other>`	`<digestAuthentication enabled=false />`

IIS client certificate mapping authentication

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=CertificateMappingAuthenticationModule image=%windir%\System32\inetsrv\authcert.dll />`
`<modules>`	`<add name=CertificateMappingAuthenticationModule lockItem=true` />`
`<clientCertificateMappingAuthentication>`	`<clientCertificateMappingAuthentication enabled=false />`

IP and domain restrictions

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=IpRestrictionModule image=%windir%\System32\inetsrv\iprestr.dll /> <add name=DynamicIpRestrictionModule image=%windir%\System32\inetsrv\diprestr.dll />`
`<modules>`	`<add name=IpRestrictionModule lockItem=true \/> <add name=DynamicIpRestrictionModule lockItem=true \/>`
`<ipSecurity>`	`<ipSecurity allowUnlisted=true />`

URL authorization

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=UrlAuthorizationModule image=%windir%\System32\inetsrv\urlauthz.dll />`
`<modules>`	`<add name=UrlAuthorizationModule lockItem=true />`
`<authorization>`	`<authorization>  <add accessType=Allow users=* /> </authorization>`

Windows authentication

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=WindowsAuthenticationModule image=%windir%\System32\inetsrv\authsspi.dll />`
`<modules>`	`<add name=WindowsAuthenticationModule lockItem=true />`
`<windowsAuthentication>`	`<windowsAuthentication enabled=false authPersistNonNTLM\=true> <providers> <add value=Negotiate /> <add value=NTLM /> <\providers> <\windowsAuthentication> <windowsAuthentication enabled=false authPersistNonNTLM\=true> <providers> <add value=Negotiate /> <add value=NTLM /> <\/providers> <\/windowsAuthentication>`

Application development

Application initialization

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=ApplicationInitializationModule image=%windir%\System32\inetsrv\warmup.dll />`
`<modules>`	`<add name=ApplicationInitializationModule lockItem=true />`

CGI

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=CgiModule image=%windir%\System32\inetsrv\cgi.dll />  <add name=FastCgiModule image=%windir%\System32\inetsrv\iisfcgi.dll />`
`<modules>`	`<add name=CgiModule lockItem=true /> <add name=FastCgiModule lockItem=true />`
`<handlers>`	`<add name=CGI-exe path=*.exe verb=\* modules=CgiModule resourceType=File requireAccess=Execute allowPathInfo=true />`

ISAPI extensions

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=IsapiModule image=%windir%\System32\inetsrv\isapi.dll />`
`<modules>`	`<add name=IsapiModule lockItem=true />`
`<handlers>`	`<add name=ISAPI-dll path=*.dll verb=* modules=IsapiModule resourceType=File requireAccess=Execute allowPathInfo=true />`

ISAPI filters

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=IsapiFilterModule image=%windir%\System32\inetsrv\filter.dll />`
`<modules>`	`<add name=IsapiFilterModule lockItem=true />`

Server-side includes

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=ServerSideIncludeModule image=%windir%\System32\inetsrv\iis_ssi.dll />`
`<modules>`	`<add name=ServerSideIncludeModule lockItem=true />`
`<handlers>`	`<add name=SSINC-stm path=*.stm verb=GET,HEAD,POST modules=ServerSideIncludeModule resourceType=File \/> <add name=SSINC-shtm path=*.shtm verb=GET,HEAD,POST modules=ServerSideIncludeModule resourceType=File /> <add name=SSINC-shtml path=*.shtml verb=GET,HEAD,POST modules=ServerSideIncludeModule resourceType=File />`
`<serverSideInclude>`	`<serverSideInclude ssiExecDisable=false />`

WebSocket protocol

SECTION	CONFIGURATION ELEMENTS
`<globalModules>`	`<add name=WebSocketModule image=%windir%\System32\inetsrv\iiswsock.dll />`
`<modules>`	`<add name=WebSocketModule lockItem=true />`

MPIO on Nano Server

12/17/2021 • 5 minutes to read • Edit Online

Applies to: Windows Server 2016

IMPORTANT

Starting in Windows Server, version 1709, Nano Server will be available only as a container base OS image. Check out Changes to Nano Server to learn what this means.

This topic introduces the use of MPIO in Nano Server installations of Windows Server 2016. For general information about MPIO in Windows Server, see Multipath I/O Overview.

Using MPIO on Nano Server

You can use MPIO on Nano Server, but with these differences:

- Only MSDSM is supported.
- The Load Balancing Policy is chosen dynamically and cannot be modified. The policy has these characteristics:
  - Default -- RoundRobin (active/active)
  - SAS HDD -- LeastBlocks
  - ALUA -- RoundRobin with Subset
- Path states (active/passive) for ALUA arrays are picked up from the target array.
- Storage devices are claimed by bus type (for example, FC, iSCSI, or SAS). When MPIO is installed on Nano Server, disks are still exposed as duplicates (one available per path) until MPIO is configured to claim and manage particular disks. The sample script in this topic will claim or unclaim disks for MPIO.
- iSCSI boot is not supported.

Enable MPIO with this Windows PowerShell cmdlet:

```
Enable-WindowsOptionalFeature -Online -FeatureName MultiPathIO
```

This sample script will allow the caller to claim or unclaim disks for MPIO by changing certain registry keys. Though you can claim other storage devices by adding them to these keys, manipulating the keys directly is not recommended.

```
#
#  Copyright (c) 2015 Microsoft Corporation.  All rights reserved.
#
#  THIS CODE AND INFORMATION IS PROVIDED AS IS WITHOUT WARRANTY
#  OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED
#  TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND/OR FITNESS FOR A
#  PARTICULAR PURPOSE
#

<#
.Synopsis
```

```
    This powershell script allows you to enable Multipath-IO support using Microsoft's
    in-box DSM (MSDSM) for storage devices attached by certain bus types.

    After running this script you will have to either:
    1. Disable and then re-enable the relevant Host Bus Adapters (HBAs); or
    2. Reboot the system.

.Description

.Parameter BusType
    Specifies the bus type for which the claim/unclaim should be done.

    If omitted, this parameter defaults to All.

    All - Will claim/unclaim storage devices attached through Fibre Channel, iSCSI, or SAS.

    FC - Will claim/unclaim storage devices attached through Fibre Channel.

    iSCSI - Will claim/unclaim storage devices attached through iSCSI.

    SAS - Will claim/unclaim storage devices attached through SAS.

.Parameter Server
    Allows you to specify a remote system, either via computer name or IP address.

    If omitted, this parameter defaults to the local system.

.Parameter Unclaim
    If specified, the script will unclaim storage devices of the bus type specified by the
    BusType parameter.

    If omitted, the script will default to claiming storage devices instead.

.Example
MultipathIoClaim.ps1

Claims all storage devices attached through Fibre Channel, iSCSI, or SAS.

.Example
MultipathIoClaim.ps1 FC

Claims all storage devices attached through Fibre Channel.

.Example
MultipathIoClaim.ps1 SAS -Unclaim

Unclaims all storage devices attached through SAS.

.Example
MultipathIoClaim.ps1 iSCSI 12.34.56.78

Claims all storage devices attached through iSCSI on the remote system with IP address 12.34.56.78.

#>
[CmdletBinding()]
param
(
    [ValidateSet('all','fc','iscsi','sas')]
    [string]$BusType='all',

    [string]$Server=127.0.0.1,

    [switch]$Unclaim
)

#
# Constants
#
$type = [Microsoft.Win32.RegistryHive]::LocalMachine
```

```
[string]$mpioKeyName = SYSTEM\CurrentControlSet\Control\MPDEV
[string]$mpioValueName = MpioSupportedDeviceList
[string]$msdsmKeyName = SYSTEM\CurrentControlSet\Services\msdsm\Parameters
[string]$msdsmValueName = DsmSupportedDeviceList

[string]$fcHwid = MSFT2015FCBusType_0x6
[string]$sasHwid = MSFT2011SASBusType_0xA
[string]$iscsiHwid = MSFT2005iSCSIBusType_0x9

#
# Functions
#

function AddHardwareId
{
    param
    (
        [Parameter(Mandatory=$True)]
        [string]$Hwid,

        [string]$Srv=127.0.0.1,

        [string]$KeyName=SYSTEM\CurrentControlSet\Control\MultipathIoClaimTest,

        [string]$ValueName=DeviceList
    )

    $regKey = [Microsoft.Win32.RegistryKey]::OpenRemoteBaseKey($type, $Srv)
    $key = $regKey.OpenSubKey($KeyName, 'true')
    $val = $key.GetValue($ValueName)
    $val += $Hwid
    $key.SetValue($ValueName, [string[]]$val, 'MultiString')
}

function RemoveHardwareId
{
    param
    (
        [Parameter(Mandatory=$True)]
        [string]$Hwid,

        [string]$Srv=127.0.0.1,

        [string]$KeyName=SYSTEM\CurrentControlSet\Control\MultipathIoClaimTest,

        [string]$ValueName=DeviceList
    )

    [string[]]$newValues = @()
    $regKey = [Microsoft.Win32.RegistryKey]::OpenRemoteBaseKey($type, $Srv)
    $key = $regKey.OpenSubKey($KeyName, 'true')
    $values = $key.GetValue($ValueName)
    foreach($val in $values)
    {
        # Only copy values that don't match the given hardware ID.
        if ($val -ne $Hwid)
        {
            $newValues += $val
            Write-Debug $($val) will remain in the key.
        }
        else
        {
            Write-Debug $($val) will be removed from the key.
        }
    }
    $key.SetValue($ValueName, [string[]]$newValues, 'MultiString')
}

function HardwareIdClaimed
```

```
{
    param
    (
        [Parameter(Mandatory=$True)]
        [string]$Hwid,

        [string]$Srv=127.0.0.1,

        [string]$KeyName=SYSTEM\CurrentControlSet\Control\MultipathIoClaimTest,

        [string]$ValueName=DeviceList
    )

    $regKey = [Microsoft.Win32.RegistryKey]::OpenRemoteBaseKey($type, $Srv)
    $key = $regKey.OpenSubKey($KeyName)
    $values = $key.GetValue($ValueName)
    foreach($val in $values)
    {
        if ($val -eq $Hwid)
        {
            return 'true'
        }
    }

    return 'false'
}

function GetBusTypeName
{
    param
    (
        [Parameter(Mandatory=$True)]
        [string]$Hwid
    )

    if ($Hwid -eq $fcHwid)
    {
        return Fibre Channel
    }
    elseif ($Hwid -eq $sasHwid)
    {
        return SAS
    }
    elseif ($Hwid -eq $iscsiHwid)
    {
        return iSCSI
    }

    return Unknown
}

#
# Execution starts here.
#

#
# Create the list of hardware IDs to claim or unclaim.
#
[string[]]$hwids = @()

if ($BusType -eq 'fc')
{
    $hwids += $fcHwid
}
elseif ($BusType -eq 'iscsi')
{
    $hwids += $iscsiHwid
}
elseif ($BusType -eq 'sas')
```

```
{
    $hwids += $sasHwid
}
elseif ($BusType -eq 'all')
{
    $hwids += $fcHwid
    $hwids += $sasHwid
    $hwids += $iscsiHwid
}
else
{
    Write-Host Please provide a bus type (FC, iSCSI, SAS, or All).
}

$changed = 'false'

#
# Attempt to claim or unclaim each of the hardware IDs.
#
foreach($hwid in $hwids)
{
    $busTypeName = GetBusTypeName $hwid

    #
    # The device is only considered claimed if it's in both the MPIO and MSDSM lists.
    #
    $mpioClaimed = HardwareIdClaimed $hwid $Server $mpioKeyName $mpioValueName
    $msdsmClaimed = HardwareIdClaimed $hwid $Server $msdsmKeyName $msdsmValueName
    if ($mpioClaimed -eq 'true' -and $msdsmClaimed -eq 'true')
    {
        $claimed = 'true'
    }
    else
    {
        $claimed = 'false'
    }

    if ($mpioClaimed -eq 'true')
    {
        Write-Debug $($hwid) is in the MPIO list.
    }
    else
    {
        Write-Debug $($hwid) is NOT in the MPIO list.
    }

    if ($msdsmClaimed -eq 'true')
    {
        Write-Debug $($hwid) is in the MSDSM list.
    }
    else
    {
        Write-Debug $($hwid) is NOT in the MSDSM list.
    }

    if ($Unclaim)
    {
        #
        # Unclaim this hardware ID.
        #
        if ($claimed -eq 'true')
        {
            RemoveHardwareId $hwid $Server $mpioKeyName $mpioValueName
            RemoveHardwareId $hwid $Server $msdsmKeyName $msdsmValueName
            $changed = 'true'
            Write-Host $($busTypeName) devices will not be claimed.
        }
        else
        {
```

```
        {
            Write-Host $($busTypeName) devices are not currently claimed.
        }

    }
    else
    {
        #
        # Claim this hardware ID.
        #
        if ($claimed -eq 'true')
        {
            Write-Host $($busTypeName) devices are already claimed.
        }
        else
        {
            AddHardwareId $hwid $Server $mpioKeyName $mpioValueName
            AddHardwareId $hwid $Server $msdsmKeyName $msdsmValueName
            $changed = 'true'
            Write-Host $($busTypeName) devices will be claimed.
        }
    }
}

#
# Finally, if we changed any of the registry keys remind the user to restart.
#
if ($changed -eq 'true')
{
    Write-Host The system must be restarted for the changes to take effect.
}
```

Manage Nano Server

12/17/2021 • 12 minutes to read • Edit Online

Applies to: Windows Server 2016

IMPORTANT

Starting in Windows Server, version 1709, Nano Server will be available only as a container base OS image. Check out Changes to Nano Server to learn what this means.

Nano Server is managed remotely. There is no local logon capability at all, nor does it support Terminal Services. However, you have a wide variety of options for managing Nano Server remotely, including Windows PowerShell, Windows Management Instrumentation (WMI), Windows Remote Management, and Emergency Management Services (EMS).

To use any remote management tool, you will probably need to know the IP address of the Nano Server. Some ways to find out the IP address include:

- Use the Nano Recovery Console (see the Using the Nano Server Recovery Console section of this topic for details).
- Connect a serial cable to the computer and use EMS.
- Using the computer name you assigned to the Nano Server while configuring it, you can get the IP address with ping. For example, `ping NanoServer-PC /4`.

Using Windows PowerShell remoting

To manage Nano Server with Windows PowerShell remoting, you need to add the IP address of the Nano Server to your management computer's list of trusted hosts, add the account you are using to the Nano Server's administrators, and enable CredSSP if you plan to use that feature.

NOTE

If the target Nano Server and your management computer are in the same AD DS forest (or in forests with a trust relationship), you should not add the Nano Server to the trusted hosts list--you can connect to the Nano Server by using its fully qualified domain name, for example: PS C:> Enter-PSSession -ComputerName nanoserver.contoso.com -Credential (Get-Credential)

To add the Nano Server to the list of trusted hosts, run this command at an elevated Windows PowerShell prompt:

```
Set-Item WSMan:\localhost\Client\TrustedHosts <IP address of Nano Server>
```

To start the remote Windows PowerShell session, start an elevated local Windows PowerShell session, and then run these commands:

```
$ip = <IP address of Nano Server>
$user = $ip\Administrator
Enter-PSSession -ComputerName $ip -Credential $user
```

You can now run Windows PowerShell commands on the Nano Server as normal.

> **NOTE**
>
> Not all Windows PowerShell commands are available in this release of Nano Server. To see which are available, run `Get-Command -CommandType Cmdlet`

Stop the remote session with the command `Exit-PSSession`

Using Windows PowerShell CIM sessions over WinRM

You can use CIM sessions and instances in Windows PowerShell to run WMI commands over Windows Remote Management (WinRM).

Start the CIM session by running these commands in a Windows PowerShell prompt:

```
$ip = <IP address of the Nano Server\>
$user = $ip\Administrator
$cim = New-CimSession -Credential $user -ComputerName $ip
```

With the session established, you can run various WMI commands, for example:

```
Get-CimInstance -CimSession $cim -ClassName Win32_ComputerSystem | Format-List *
Get-CimInstance -CimSession $Cim -Query SELECT * from Win32_Process WHERE name LIKE 'p%'
```

Windows Remote Management

You can run programs remotely on the Nano Server with Windows Remote Management (WinRM). To use WinRM, first configure the service and set the code page with these commands at an elevated command prompt:

```
winrm quickconfig
winrm set winrm/config/client @{TrustedHosts=<ip address of Nano Server>}
chcp 65001
```

Now you can run commands remotely on the Nano Server. For example:

```
winrs -r:<IP address of Nano Server> -u:Administrator -p:<Nano Server administrator password> ipconfig
```

For more information about Windows Remote Management, see Windows Remote Management (WinRM) Overview.

Running a network trace on Nano Server

Netsh trace, Tracelog.exe, and Logman.exe are not available in Nano Server. To capture network packets, you can use these Windows PowerShell cmdlets:

```
New-NetEventSession [-Name]
Add-NetEventPacketCaptureProvider -SessionName
Start-NetEventSession [-Name]
Stop-NetEventSession [-Name]
```

These cmdlets are documented in detail at Network Event Packet Capture Cmdlets in Windows PowerShell

Installing servicing packages

If you want install a servicing packages, use the -ServicingPackagePath parameter (you can pass an array of paths to .cab files):

```
New-NanoServerImage -DeploymentType Guest -Edition Standard -MediaPath \\Path\To\Media\en_us -BasePath .\Base
-TargetPath .\NanoServer.wim -ServicingPackagePath \\path\to\kb123456.cab
```

Often, a servicing package or hotfix is downloaded as a KB item which contains a .cab file. Follow these steps to extract the .cab file, which you can then install with the -ServicingPackagePath parameter:

1. Download the servicing package (from the associated Knowledge Base article or from Microsoft Update Catalog. Save it to a local directory or network share, for example: C:\ServicingPackages
2. Create a folder in which you will save the extracted servicing package. Example: c:\KB3157663_expanded
3. Open a Windows PowerShell console and use the `Expand` command specifying the path to the .msu file of the servicing package, including the `-f:*` parameter and the path where you want servicing package to be extracted to. For example:
   ```
   Expand C:\ServicingPackages\Windows10.0-KB3157663-x64.msu -f:* C:\KB3157663_expanded
   ```
   The expanded files should look similar to this: C:>dir C:\KB3157663_expanded Volume in drive C is OS Volume Serial Number is B05B-CC3D

   Directory of C:\KB3157663_expanded

   04/19/2016 01:17 PM <DIR> . 04/19/2016 01:17 PM <DIR> .. 04/17/2016 12:31 AM 517 Windows10.0-KB3157663-x64-pkgProperties.txt 04/17/2016 12:30 AM 93,886,347 Windows10.0-KB3157663-x64.cab 04/17/2016 12:31 AM 454 Windows10.0-KB3157663-x64.xml 04/17/2016 12:36 AM 185,818 WSUSSCAN.cab 4 File(s) 94,073,136 bytes 2 Dir(s) 328,559,427,584 bytes free
4. Run `New-NanoServerImage` with the -ServicingPackagePath parameter pointing to the .cab file in this directory, for example:
   ```
   New-NanoServerImage -DeploymentType Guest -Edition Standard -MediaPath \\Path\To\Media\en_us -BasePath
   .\Base -TargetPath .\NanoServer.wim -ServicingPackagePath C:\KB3157663_expanded\Windows10.0-KB3157663-
   x64.cab
   ```

Managing updates in Nano Server

Currently you can use the Windows Update provider for Windows Management Instrumentation (WMI) to find the list of applicable updates, and then install all or a subset of them. If you use Windows Server Update Services (WSUS), you can also configure Nano Server to contact the WSUS server to obtain updates.

In all cases, first establish a remote Windows PowerShell session to the Nano Server computer. These examples use *$sess* for the session; if you are using something else, replace that element as needed.

View all available updates

Obtain the full list of applicable updates with these commands:

```
$sess = New-CimInstance -Namespace root/Microsoft/Windows/WindowsUpdate -ClassName MSFT_WUOperationsSession

$scanResults = Invoke-CimMethod -InputObject $sess -MethodName ScanForUpdates -Arguments
@{SearchCriteria=IsInstalled=0;OnlineScan=$true}
```

Note: If no updates are available, this command will return the following error:

```
Invoke-CimMethod : A general error occurred that is not covered by a more specific error code.

At line:1 char:16

+ ... anResults = Invoke-CimMethod -InputObject $sess -MethodName ScanForUp ...

+                 ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

    + CategoryInfo          : NotSpecified: (MSFT_WUOperatio...-5b842a3dd45d)

   :CimInstance) [Invoke-CimMethod], CimException

    + FullyQualifiedErrorId : MI RESULT 1,Microsoft.Management.Infrastructure.

   CimCmdlets.InvokeCimMethodCommand
```

Install all available updates

You can detect, download, and install **all** available updates at one time by using these commands:

```
$sess = New-CimInstance -Namespace root/Microsoft/Windows/WindowsUpdate -ClassName MSFT_WUOperationsSession

$scanResults = Invoke-CimMethod -InputObject $sess -MethodName ApplyApplicableUpdates

Restart-Computer
```

Note: Windows Defender will prevent updates from installing. To work around this, uninstall Windows Defender, install the updates, and then reinstall Windows Defender. Alternately, you can download the updates on another computer, copy them to the Nano Server, and then apply them with DISM.exe.

Verify installation of updates

Use these commands to get a list of the updates currently installed:

```
$sess = New-CimInstance -Namespace root/Microsoft/Windows/WindowsUpdate -ClassName MSFT_WUOperationsSession

$scanResults = Invoke-CimMethod -InputObject $sess -MethodName ScanForUpdates -Arguments
@{SearchCriteria=IsInstalled=1;OnlineScan=$true}
```

Note: These commands list what is installed, but do not specifically quote installed in the output. If you need output including that, such as for a report, you can run

```
Get-WindowsPackage -Online
```

Using WSUS

The commands listed above will query the Windows Update and Microsoft Update serviceon the Internet to find and download updates. If you use WSUS, you can set registry keys on the Nano Server to use your WSUS server instead.

See the Windows Update Agent Environment Options Registry Keys table in Configure Automatic Updates in a Non-Active-Directory Environment

You should set at least the **WUServer** and **WUStatusServer** registry keys, but depending on how you have implemented WSUS, other values might be needed. You can always confirm these settings by examining another Windows Server in the same environment.

Once these values are set for your WSUS, the commands in the section above will query that server for updates

and use it the download source.

Automatic Updates

Currently, the way to automate update installation is to convert the steps above into a local Windows PowerShell script and then create a scheduled task to run it and restart the system on your schedule.

Performance and event monitoring on Nano Server

Nano Server fully supports the Event Tracing for Windows (ETW) framework, but some familiar tools used to manage tracing and performance counters are not currently available on Nano Server. However, Nano Server has tools and cmdlets to accomplish most common performance analysis scenarios.

The high-level workflow remains the same as on any Window Server installation -- low-overhead tracing is performed on the target (Nano Server) computer, and the resulting trace files and/or logs are post-processed offline on a separate computer using tools such as Windows Performance Analyzer, Message Analyzer, or others.

> **NOTE**
> Refer to How to copy files to and from Nano Server for a refresher on how to transfer files using PowerShell remoting.

The following sections list the most common performance data collection activities along with a supported way to accomplish them on Nano Server.

Query available event providers

Windows Performance Recorder is tool to query available event providers as follows:

```
wpr.exe -providers
```

You can filter the output on the type of events that are of interest. For example:

```
PS C:\> wpr.exe -providers | select-string Storage

       595f33ea-d4af-4f4d-b4dd-9dacdd17fc6e                              : Microsoft-Windows-
StorageManagement-WSP-Host
       595f7f52-c90a-4026-a125-8eb5e083f15e                              : Microsoft-Windows-StorageSpaces-
Driver
       69c8ca7e-1adf-472b-ba4c-a0485986b9f6                              : Microsoft-Windows-StorageSpaces-
SpaceManager
       7e58e69a-e361-4f06-b880-ad2f4b64c944                              : Microsoft-Windows-
StorageManagement
       88c09888-118d-48fc-8863-e1c6d39ca4df                              : Microsoft-Windows-
StorageManagement-WSP-Spaces
```

Record traces from a single ETW provider

You can use new Event Tracing Management cmdlets for this. Here is an example workflow:

Create and start the trace, specifying a file name for storing the events.

```
PS C:\> New-EtwTraceSession -Name ExampleTrace -LocalFilePath c:\etrace.etl
```

Add a provider GUID to the trace. Use `wpr.exe -providers` for Provider Name to GUID translation.

```
PS C:\> wpr.exe -providers | select-string Kernel-Memory

       d1d93ef7-e1f2-4f45-9943-03d245fe6c00                              : Microsoft-Windows-Kernel-Memory

PS C:\> Add-EtwTraceProvider -Guid {d1d93ef7-e1f2-4f45-9943-03d245fe6c00} -SessionName ExampleTrace
```

Remove the trace -- this stops the trace session, flushing events to the associated log file.

```
PS C:\> Remove-EtwTraceSession -Name ExampleTrace

PS C:\> dir .\etrace.etl

    Directory: C:\

Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        9/14/2016  11:17 AM       16515072 etrace.etl
```

> **NOTE**
>
> This example shows adding a single trace provider to the session, but you can also use the `Add-EtwTraceProvider` cmdlet multiple times on a trace session with different provider GUIDs to enable tracing from multiple sources. Another alternative is to use `wpr.exe` profiles described below.

Record traces from multiple ETW providers

The `-profiles` option of Windows Performance Recorder enables tracing from multiple providers at the same time. There are a number of built-in profiles like CPU, Network, and DiskIO to choose from:

```
PS C:\Users\Administrator\Documents> wpr.exe -profiles

Microsoft Windows Performance Recorder Version 10.0.14393 (CoreSystem)
Copyright (c) 2015 Microsoft Corporation. All rights reserved.

        GeneralProfile              First level triage
        CPU                         CPU usage
        DiskIO                      Disk I/O activity
        FileIO                      File I/O activity
        Registry                    Registry I/O activity
        Network                     Networking I/O activity
        Heap                        Heap usage
        Pool                        Pool usage
        VirtualAllocation           VirtualAlloc usage
        Audio                       Audio glitches
        Video                       Video glitches
        Power                       Power usage
        InternetExplorer            Internet Explorer
        EdgeBrowser                 Edge Browser
        Minifilter                  Minifilter I/O activity
        GPU                         GPU activity
        Handle                      Handle usage
        XAMLActivity                XAML activity
        HTMLActivity                HTML activity
        DesktopComposition          Desktop composition activity
        XAMLAppResponsiveness       XAML App Responsiveness analysis
        HTMLResponsiveness          HTML Responsiveness analysis
        ReferenceSet                Reference Set analysis
        ResidentSet                 Resident Set analysis
        XAMLHTMLAppMemoryAnalysis   XAML/HTML application memory analysis
        UTC                         UTC Scenarios
        DotNET                      .NET Activity
        WdfTraceLoggingProvider     WDF Driver Activity
```

For detailed guidance on creating custom profiles, see the WPR.exe documentation.

Record ETW traces during operating system boot time

Use the `New-AutologgerConfig` cmdlet to collect events during system boot. Usage is very similar to the `New-EtwTraceSession` cmdlet, but providers added to the Autologger's configuration will only be enabled early at next boot. The overall workflow looks like this:

First, create a new Autologger config.

```
PS C:\> New-AutologgerConfig -Name BootPnpLog -LocalFilePath c:\bootpnp.etl
```

Add a ETW provider to it. This example uses the Kernel PnP provider. Invoke `Add-EtwTraceProvider` again, specifying the same Autologger name but a different GUID to enable boot trace collection from multiple sources.

```
Add-EtwTraceProvider -Guid {9c205a39-1250-487d-abd7-e831c6290539} -AutologgerName BootPnpLog
```

This does not start an ETW session immediately, but rather configures one to start at next boot. After rebooting, a new ETW session with the Autologger configuration name is automatically started with the added trace providers enabled. After Nano Server boots, the following command will stop the trace session after flushing the logged events to the associated trace file:

```
PS C:\> Remove-EtwTraceSession -Name BootPnpLog
```

To prevent another trace session from being auto-created at next boot, remove the Autologger configuration as follows:

```
PS C:\> Remove-AutologgerConfig -Name BootPnpLog
```

To collect boot and setup traces across a number of systems or on a diskless system, consider using Setup and Boot Event Collection.

Capture performance counter data

Usually, you monitor performance counter data with Perfmon.exe GUI. On Nano Server, use the `Typeperf.exe` command-line equivalent. For example:

Query available counters--you can filter the output to easily find the ones of interest.

```
PS C:\> typeperf.exe -q | Select-String UDPv6

\UDPv6\Datagrams/sec
\UDPv6\Datagrams Received/sec
\UDPv6\Datagrams No Port/sec
\UDPv6\Datagrams Received Errors
\UDPv6\Datagrams Sent/sec
```

Options allow specifying the number of times and the interval at which counter values are collected. In the example below, Processor Idle Time is collected 5 times every 3 seconds.

```
PS C:\> typeperf.exe \Processor Information(0,0)\% Idle Time -si 3 -sc 5

(PDH-CSV 4.0),\\ns-g2\Processor Information(0,0)\% Idle Time
09/15/2016 09:20:56.002,99.982990
09/15/2016 09:20:59.002,99.469634
09/15/2016 09:21:02.003,99.990081
09/15/2016 09:21:05.003,99.990454
09/15/2016 09:21:08.003,99.998577
Exiting, please wait...
The command completed successfully.
```

Other command-line options allow you to specify performance counter names of interest in a configuration file, redirecting output to a log file, among other things. See the typeperf.exe documentation for details.

You can also use Perfmon.exe's graphical interface remotely with Nano Server targets. When adding performance counters to the view, specify the Nano Server target in the computer name instead of the default *<Local computer>*.

Interact with the Windows Event Log

Nano Server supports the `Get-WinEvent` cmdlet, which provides Windows Event Log filtering and querying capabilities, both locally as well as on a remote computer. Detailed options and examples are available at the Get-WinEvent documentation page. This simple example retrieves the *Errors* noted in the *System* log during the past two days.

```
PS C:\> $StartTime = (Get-Date) - (New-TimeSpan -Day 2)
PS C:\> Get-WinEvent -FilterHashTable @{LogName='System'; Level=2; StartTime=$StartTime} | select
TimeCreated, Message

TimeCreated           Message
-----------           -------
9/15/2016 11:31:19 AM Task Scheduler service failed to start Task Compatibility module. Tasks may not be
able to reg...
9/15/2016 11:31:16 AM The Virtualization Based Security enablement policy check at phase 6 failed with
status: {File...
9/15/2016 11:31:16 AM The Virtualization Based Security enablement policy check at phase 0 failed with
status: {File...
```

Nano Server also supports `wevtutil.exe` which allows retrieving information about event logs and publishers. See wevtutil.exe documentation for more details.

Graphical interface tools

Web-based server management tools can be used to remotely manage Nano Server targets and present a Nano Server Event Log by using a web browser. Finally, the MMC snap-in Event Viewer (eventvwr.msc) can also be used to view logs -- just open it on a computer with a desktop and point it to a remote Nano Server.

Using Windows PowerShell Desired State Configuration with Nano Server

You can manage Nano Server as target nodes with Windows PowerShell Desired State Configuration (DSC). Currently, you can manage nodes running Nano Server with DSC in push mode only. Not all DSC features function with Nano Server.

For full details, see Using DSC on Nano Server.

Updating Nano Server

12/17/2021 • 6 minutes to read • Edit Online

IMPORTANT

Starting in Windows Server, version 1709, Nano Server will be available only as a container base OS image. Check out Changes to Nano Server to learn what this means.

Nano Server offers a variety of methods for staying up to date. Compared to other installation options of Windows Server, Nano Server follows a more active servicing model similar to that of Windows 10. These periodic releases are known as **Current Branch for Business (CBB)** releases. This approach supports customers who want to innovate more quickly and move at a cloud cadence of rapid development lifecycles. More information about CBB is available on the Windows Server Blog.

Between these CBB releases, Nano Server stays current with a series of *cumulative updates*. For example, the first cumulative update for Nano Server was released on September 26, 2016 with KB4093120. With this and subsequent cumulative updates, we provide various options for installing these updates on Nano Server. In this article, we'll use the KB3192366 update as an example to illustrate how to obtain and apply cumulative updates to Nano Server. For more information on the cumulative update model, see the Microsoft Update blog.

NOTE

If you install an optional Nano Server package from media or online repository, it won't have recent security fixes included. To avoid a version mismatch between the optional packages and base operating system, you should install the latest cumulative update immediately after installing any optional packages and **before** restarting the server.

In the case of the Cumulative Update for Windows Server 2016: September 26, 2016 (KB3192366), you should first install the latest Servicing Stack Update for Windows 10 Version 1607: August 23, 2016 as a prerequisite (KB3176936). For most of the options below, you need the .msu files containing the .cab update packages. Visit the Microsoft Update Catalog to download each of these update packages:

- https://catalog.update.microsoft.com/v7/site/Search.aspx?q=KB3192366
- https://catalog.update.microsoft.com/v7/site/Search.aspx?q=KB3176936

After downloading the .msu files from the Microsoft Update Catalog, save them to a network share or local directory such as C:\ServicingPackages. You can rename the .msu files based on their KB number as we've done below to make them easier to identify. Then use the EXPAND utility to extract the .cab files from the .msu files into separate directories and copy the .cabs into a single folder.

```
mkdir C:\ServicingPackages_expanded
mkdir C:\ServicingPackages_expanded\KB3176936
mkdir C:\ServicingPackages_expanded\KB3192366
Expand C:\ServicingPackages\KB3176936.msu -F:* C:\ServicingPackages_expanded\KB3176936
Expand C:\ServicingPackages\KB3192366.msu -F:* C:\ServicingPackages_expanded\KB3192366
mkdir C:\ServicingPackages_cabs
copy C:\ServicingPackages_expanded\KB3176936\Windows10.0-KB3176936-x64.cab C:\ServicingPackages_cabs
copy C:\ServicingPackages_expanded\KB3192366\Windows10.0-KB3192366-x64.cab C:\ServicingPackages_cabs
```

Now you can use the extracted .cab files to apply the updates to a Nano Server image in a few different ways, depending on your needs. The following options are presented in no particular order of preference - use the

option that makes the most sense for your environment.

> **NOTE**
> When using the DISM tools to service Nano Server, you must use a version of DISM that is the same as or newer than the version of Nano Server you're servicing. You can achieve this by running DISM from a matching version of Windows, installing a matching version of the Windows Asssessment and Deployment Kit (ADK), or running DISM on Nano Server itself.

Option 1: Integrate a cumulative update into a new image

If you are building a new Nano Server image, you can integrate the latest cumulative update directly into the image so that it's fully patched on first boot.

```
New-NanoServerImage -ServicingPackagePath 'C:\ServicingPackages_cabs\Windows10.0-KB3176936-x64.cab',
'C:\ServicingPackages_cabs\Windows10.0-KB3192366-x64.cab' -<other parameters>
```

Option 2: Integrate a cumulative update into an existing image

If you have an existing Nano Server image that you use as a baseline for creating specific instances of Nano Server, you can integrate the latest cumulative update directly into your existing baseline image so that machines created using the image are fully patched on first boot.

```
Edit-NanoServerImage -ServicingPackagePath 'C:\ServicingPackages_cabs\Windows10.0-KB3176936-x64.cab',
'C:\ServicingPackages_cabs\Windows10.0-KB3192366-x64.cab' -TargetPath .\NanoServer.wim
```

Option 3: Apply the cumulative update to an existing offline VHD or VHDX

If you have an existing virtual hard disk (VHD or VHDX), you can use the DISM tools to apply the update to the virtual hard disk. You need to make sure the disk is not in use either by shutting down any VMs using the disk or unmounting the virtual hard disk file.

- Using PowerShell

```
Mount-WindowsImage -ImagePath .\NanoServer.vhdx -Path .\MountDir -Index 1
Add-WindowsPackage -Path .\MountDir -PackagePath  C:\ServicingPackages_cabs
Dismount-WindowsImage -Path .\MountDir -Save
```

- Using dism.exe

```
dism.exe /Mount-Image /ImageFile:C:\NanoServer.vhdx /Index:1 /MountDir:C:\MountDir
dism.exe /Image:C:\MountDir /Add-Package /PackagePath:C:\ServicingPackages_cabs
dism.exe /Unmount-Image /MountDir:C:\MountDir /Commit
```

Option 4: Apply the cumulative update to a running Nano Server

If you have a running Nano Server VM or physical host and you've downloaded the .cab file for the update, you can use the DISM tools to apply the update while the operating system is online. You will need to copy the .cab file locally on the Nano Server or to an accessible network location. If you're applying a servicing stack update, make sure to restart the server after applying the servicing stack update before applying additional updates.

NOTE

If you've created the Nano Server VHD or VHDX image using the New-NanoServerImage cmdlet and didn't specify a MaxSize for the virtual hard disk file, the default size of 4GB is too small to apply the cumulative update. Prior to installing the update, use Hyper-V Manager, Disk Management, PowerShell, or other tool to expand the size of the virtual hard disk and system volume to at least 10GB, or use the ScratchDir parameter on the DISM tools to set the scratch directory to a volume with at least 10GB of free space.

```
$s = New-PSSession -ComputerName (Read-Host "Enter Nano Server IP address") -Credential (Get-Credential)
Copy-Item -ToSession $s -Path C:\ServicingPackages_cabs -Destination C:\ServicingPackages_cabs -Recurse
Enter-PSSession $s
```

- Using PowerShell

```
# Apply the servicing stack update first and then restart
Add-WindowsPackage -Online -PackagePath C:\ServicingPackages_cabs\Windows10.0-KB3176936-x64.cab
Restart-Computer; exit

# After restarting, apply the cumulative update and then restart
Enter-PSSession -ComputerName (Read-Host "Enter Nano Server IP address") -Credential (Get-Credential)
Add-WindowsPackage -Online -PackagePath C:\ServicingPackages_cabs\Windows10.0-KB3192366-x64.cab
Restart-Computer; exit
```

- Using dism.exe

```
# Apply the servicing stack update first and then restart
dism.exe /Online /Add-Package /PackagePath:C:\ServicingPackages_cabs\Windows10.0-KB3176936-x64.cab

# After the operation completes successfully and you are prompted to restart, it's safe to
# press Ctrl+C to cancel the pipeline and return to the prompt
Restart-Computer; exit

# After restarting, apply the cumulative update and then restart
Enter-PSSession -ComputerName (Read-Host "Enter Nano Server IP address") -Credential (Get-Credential)
dism.exe /Online /Add-Package /PackagePath:C:\ServicingPackages_cabs\Windows10.0-KB3192366-x64.cab
Restart-Computer; exit
```

Option 5: Download and install the cumulative update to a running Nano Server

If you have a running Nano Server VM or physical host, you can use the Windows Update WMI provider to download and install the update while the operating system is online. With this method, you don't need to download the .msu file separately from the Microsoft Update Catalog. The WMI provider will detect, download, and install all available updates at once.

```
Enter-PSSession -ComputerName (Read-Host "Enter Nano Server IP address") -Credential (Get-Credential)
```

- Scan for available updates

```
$ci = New-CimInstance -Namespace root/Microsoft/Windows/WindowsUpdate -ClassName
MSFT_WUOperationsSession
$result = $ci | Invoke-CimMethod -MethodName ScanForUpdates -Arguments
@{SearchCriteria="IsInstalled=0";OnlineScan=$true}
$result.Updates
```

- Install all available updates

```
$ci = New-CimInstance -Namespace root/Microsoft/Windows/WindowsUpdate -ClassName
MSFT_WUOperationsSession
Invoke-CimMethod -InputObject $ci -MethodName ApplyApplicableUpdates
Restart-Computer; exit
```

- Get a list of installed updates

```
$ci = New-CimInstance -Namespace root/Microsoft/Windows/WindowsUpdate -ClassName
MSFT_WUOperationsSession
$result = $ci | Invoke-CimMethod -MethodName ScanForUpdates -Arguments
@{SearchCriteria="IsInstalled=1";OnlineScan=$true}
$result.Updates
```

Additional Options

Other methods for updating Nano Server might overlap or complement the options above. Such options include using Windows Server Update Services (WSUS), System Center Virtual Machine Manager (VMM), Task Scheduler, or a non-Microsoft solution.

- Configuring Windows Update for WSUS by setting the following registry keys:
  - WUServer
  - WUStatusServer (generally uses the same value as WUServer)
  - UseWUServer
  - AUOptions
- Managing Fabric Updates in VMM
- Registering a Scheduled Task

Developing for Nano Server

12/17/2021 • 2 minutes to read • Edit Online

Applies to: Windows Server 2016

IMPORTANT

Starting in Windows Server, version 1709, Nano Server will be available only as a container base OS image. Check out Changes to Nano Server to learn what this means.

These topics explain important differences in PowerShell on Nano Server and also provide guidance for developing your own PowerShell cmdlets for use on Nano Server.

- PowerShell on Nano Server
- Developing PowerShell Cmdlets for Nano Server

Using Windows PowerShell remoting

To manage Nano Server with Windows PowerShell remoting, you need to add the IP address of the Nano Server to your management computer's list of trusted hosts, add the account you are using to the Nano Server's administrators, and enable CredSSP if you plan to use that feature.

NOTE

If the target Nano Server and your management computer are in the same AD DS forest (or in forests with a trust relationship), you should not add the Nano Server to the trusted hosts list--you can connect to the Nano Server by using its fully qualified domain name, for example: PS C:> Enter-PSSession -ComputerName nanoserver.contoso.com -Credential (Get-Credential)

To add the Nano Server to the list of trusted hosts, run this command at an elevated Windows PowerShell prompt:

```
Set-Item WSMan:\localhost\Client\TrustedHosts <IP address of Nano Server>
```

To start the remote Windows PowerShell session, start an elevated local Windows PowerShell session, and then run these commands:

```
$ip = \<IP address of Nano Server>
$user = $ip\Administrator
Enter-PSSession -ComputerName $ip -Credential $user
```

You can now run Windows PowerShell commands on the Nano Server as normal.

NOTE

Not all Windows PowerShell commands are available in this release of Nano Server. To see which are available, run `Get-Command -CommandType Cmdlet`

Stop the remote session with the command `Exit-PSSession`

Using Windows PowerShell CIM sessions over WinRM

You can use CIM sessions and instances in Windows PowerShell to run WMI commands over Windows Remote Management (WinRM).

Start the CIM session by running these commands in a Windows PowerShell prompt:

```
$ip = <IP address of the Nano Server\>
$ip\Administrator
$cim = New-CimSession -Credential $user -ComputerName $ip
```

With the session established, you can run various WMI commands, for example:

```
Get-CimInstance -CimSession $cim -ClassName Win32_ComputerSystem | Format-List *
Get-CimInstance -CimSession $Cim -Query SELECT * from Win32_Process WHERE name LIKE 'p%'
```

PowerShell on Nano Server

12/17/2021 • 3 minutes to read • Edit Online

Applies to: Windows Server 2016

IMPORTANT

Starting in Windows Server, version 1709, Nano Server will be available only as a container base OS image. Check out Changes to Nano Server to learn what this means.

PowerShell Editions

Starting with version 5.1, PowerShell is available in different editions which denote varying feature sets and platform compatibility.

- **Desktop Edition:** Built on .NET Framework and provides compatibility with scripts and modules targeting versions of PowerShell running on full footprint editions of Windows such as Server Core and Windows Desktop.
- **Core Edition:** Built on .NET Core and provides compatibility with scripts and modules targeting versions of PowerShell running on reduced footprint editions of Windows such as Nano Server and Windows IoT.

The running edition of PowerShell is shown in the PSEdition property of $PSVersionTable.

```
$PSVersionTable

Name                           Value
----                           -----
PSVersion                      5.1.14300.1000
PSEdition                      Desktop
PSCompatibleVersions           {1.0, 2.0, 3.0, 4.0...}
CLRVersion                     4.0.30319.42000
BuildVersion                   10.0.14300.1000
WSManStackVersion              3.0
PSRemotingProtocolVersion      2.3
SerializationVersion           1.1.0.1
```

Module authors can declare their modules to be compatible with one or more PowerShell editions using the CompatiblePSEditions module manifest key. This key is only supported on PowerShell 5.1 or later.

```
New-ModuleManifest -Path .\TestModuleWithEdition.psd1 -CompatiblePSEditions Desktop,Core -PowerShellVersion
5.1
$moduleInfo = Test-ModuleManifest -Path \TestModuleWithEdition.psd1
$moduleInfo.CompatiblePSEditions
Desktop
Core

$moduleInfo | Get-Member CompatiblePSEditions

   TypeName: System.Management.Automation.PSModuleInfo

Name                 MemberType Definition
----                 ---------- ----------
CompatiblePSEditions Property   System.Collections.Generic.IEnumerable[string] CompatiblePSEditions {get;}
```

When getting a list of available modules, you can filter the list by PowerShell edition.

```
Get-Module -ListAvailable | ? CompatiblePSEditions -Contains Desktop

    Directory: C:\Program Files\WindowsPowerShell\Modules


ModuleType Version    Name                                ExportedCommands
---------- -------    ----                                ----------------
Manifest   1.0        ModuleWithPSEditions

Get-Module -ListAvailable | ? CompatiblePSEditions -Contains Core | % CompatiblePSEditions
Desktop
Core
```

Script authors can prevent a script from executing unless it is run on a compatible edition of PowerShell using the PSEdition parameter on a #requires statement.

```
Set-Content C:\script.ps1 -Value #requires -PSEdition Core
Get-Process -Name PowerShell
Get-Content C:\script.ps1
#requires -PSEdition Core
Get-Process -Name PowerShell

C:\script.ps1
C:\script.ps1 : The script 'script.ps1' cannot be run because it contained a #requires statement for
PowerShell editions 'Core'. The edition of PowerShell that is required by the script does not match the
currently running PowerShell Desktop edition.
At line:1 char:1
+ C:\script.ps1
+ ~~~~~~~~~~~~~
    + CategoryInfo          : NotSpecified: (script.ps1:String) [], RuntimeException
    + FullyQualifiedErrorId : ScriptRequiresUnmatchedPSEdition
```

Differences in PowerShell on Nano Server

Nano Server includes PowerShell Core by default in all Nano Server installations. PowerShell Core is a reduced footprint edition of PowerShell that is built on .NET Core and runs on reduced footprint editions of Windows, such as Nano Server and Windows IoT Core. PowerShell Core functions in the same way as other editions of PowerShell, such as Windows PowerShell running on Windows Server 2016. However, the reduced footprint of Nano Server means that not all PowerShell features from Windows Server 2016 are available in PowerShell Core on Nano Server.

Windows PowerShell features not available in Nano Server

- ADSI, ADO, and WMI type adapters
- Enable-PSRemoting, Disable-PSRemoting (PowerShell remoting is enabled by default; see the Using Windows PowerShell Remoting section of Install Nano Server).
- Scheduled jobs and PSScheduledJob module
- Computer cmdlets for joining a domain { Add | Remove } (for different methods to join Nano Server to a domain, see the Joining Nano Server to a domain section of Install Nano Server).
- Reset-ComputerMachinePassword, Test-ComputerSecureChannel
- Profiles (you can add a startup script for incoming remote connections with `Set-PSSessionConfiguration`)
- Clipboard cmdlets
- EventLog cmdlets { Clear | Get | Limit | New | Remove | Show | Write } (use the New-WinEvent and Get-WinEvent cmdlets instead).
- Get-PfxCertificate cmdlet
- TraceSource cmdlets { Get | Set }
- Counter cmdlets { Get | Export | Import }
- Some web-related cmdlets { New-WebServiceProxy, Send-MailMessage, ConvertTo-Html }
- Logging and tracing using PSDiagnostics module
- Get-HotFix (to obtain and manage updates on Nano Server, see Manage Nano Server).
- Implicit remoting cmdlets { Export-PSSession | Import-PSSession }
- New-PSTransportOption
- PowerShell transactions and Transaction cmdlets { Complete | Get | Start | Undo | Use }
- PowerShell Workflow infrastructure, modules, and cmdlets
- Out-Printer
- Update-List
- WMI v1 cmdlets: Get-WmiObject, Invoke-WmiMethod, Register-WmiEvent, Remove-WmiObject, Set-WmiInstance (use CimCmdlets module instead.)

Using Windows PowerShell Desired State Configuration with Nano Server

You can manage Nano Server as target nodes with Windows PowerShell Desired State Configuration (DSC). Currently, you can manage nodes running Nano Server with DSC in push mode only. Not all DSC features function with Nano Server.

For full details, see Using DSC on Nano Server.

Developing PowerShell Cmdlets for Nano Server

12/17/2021 • 9 minutes to read • Edit Online

Applies to: Windows Server 2016

IMPORTANT

Starting in Windows Server, version 1709, Nano Server will be available only as a container base OS image. Check out Changes to Nano Server to learn what this means.

Overview

Nano Server includes PowerShell Core by default in all Nano Server installations. PowerShell Core is a reduced-footprint edition of PowerShell that is built on .NET Core and runs on reduced-footprint editions of Windows, such as Nano Server and Windows IoT Core. PowerShell Core functions in the same way as other editions of PowerShell, such as Windows PowerShell running on Windows Server 2016. However, the reduced footprint of Nano Server means that not all PowerShell features from Windows Server 2016 are available in PowerShell Core on Nano Server.

If you have existing PowerShell cmdlets that you'd like to run on Nano Server, or are developing new ones for that purpose, this topic includes tips and suggestions that should help make that easier.

PowerShell editions

Starting with version 5.1, PowerShell is available in different editions which denote varying feature sets and platform compatibility.

- **Desktop Edition:** Built on .NET Framework and provides compatibility with scripts and modules targeting versions of PowerShell running on full footprint editions of Windows such as Server Core and Windows Desktop.
- **Core Edition:** Built on .NET Core and provides compatibility with scripts and modules targeting versions of PowerShell running on reduced footprint editions of Windows such as Nano Server and Windows IoT.

The running edition of PowerShell is shown in the PSEdition property of $PSVersionTable.

```
$PSVersionTable

Name                           Value
----                           -----
PSVersion                      5.1.14300.1000
PSEdition                      Desktop
PSCompatibleVersions           {1.0, 2.0, 3.0, 4.0...}
CLRVersion                     4.0.30319.42000
BuildVersion                   10.0.14300.1000
WSManStackVersion              3.0
PSRemotingProtocolVersion      2.3
SerializationVersion           1.1.0.1
```

Module authors can declare their modules to be compatible with one or more PowerShell editions using the CompatiblePSEditions module manifest key. This key is only supported on PowerShell 5.1 or later.

```
New-ModuleManifest -Path .\TestModuleWithEdition.psd1 -CompatiblePSEditions Desktop,Core -PowerShellVersion
5.1
$moduleInfo = Test-ModuleManifest -Path \TestModuleWithEdition.psd1
$moduleInfo.CompatiblePSEditions
Desktop
Core

$moduleInfo | Get-Member CompatiblePSEditions

   TypeName: System.Management.Automation.PSModuleInfo

Name                 MemberType Definition
----                 ---------- ----------
CompatiblePSEditions Property   System.Collections.Generic.IEnumerable[string] CompatiblePSEditions {get;}
```

When getting a list of available modules, you can filter the list by PowerShell edition.

```
Get-Module -ListAvailable | ? CompatiblePSEditions -Contains Desktop

    Directory: C:\Program Files\WindowsPowerShell\Modules


ModuleType Version    Name                                ExportedCommands
---------- -------    ----                                ----------------
Manifest   1.0        ModuleWithPSEditions

Get-Module -ListAvailable | ? CompatiblePSEditions -Contains Core | % CompatiblePSEditions
Desktop
Core
```

Script authors can prevent a script from executing unless it is run on a compatible edition of PowerShell using the PSEdition parameter on a #requires statement.

```
Set-Content C:\script.ps1 -Value #requires -PSEdition Core
Get-Process -Name PowerShell
Get-Content C:\script.ps1
#requires -PSEdition Core
Get-Process -Name PowerShell

C:\script.ps1
C:\script.ps1 : The script 'script.ps1' cannot be run because it contained a #requires statement for
PowerShell editions 'Core'. The edition of PowerShell that is required by the script does not match the
currently running PowerShell Desktop edition.
At line:1 char:1
+ C:\script.ps1
+ ~~~~~~~~~~~~~
    + CategoryInfo          : NotSpecified: (script.ps1:String) [], RuntimeException
    + FullyQualifiedErrorId : ScriptRequiresUnmatchedPSEdition
```

Installing Nano Server

Quick-start and detailed steps for installing Nano Server on virtual or physical machines are provided in Install Nano Server, which is the parent topic for this one.

NOTE

For development work on Nano Server, you might find it useful to install Nano Server by using the -Development parameter of New-NanoServerImage. This will enable installation of unsigned drivers, copy debugger binaries, open a port for debugging, enable test signing and enable installation of AppX packages without a developer license. For example:

```
New-NanoServerImage -DeploymentType Guest -Edition Standard -MediaPath \\Path\To\Media\en_us -BasePath
.\Base -TargetPath .\NanoServer.wim -Development
```

Determining the type of cmdlet implementation

PowerShell supports a number of implementation types for cmdlets, and the one you've used determines the process and tools involved in creating or porting it to work on Nano Server. Supported implementation types are:

- CIM - consists of CDXML files layered over CIM (WMIv2) providers
- .NET - consists of .NET assemblies implementing managed cmdlet interfaces, typically written in C#
- PowerShell Script - consists of script modules (.psm1) or scripts (.ps1) written in the PowerShell language

If you're not sure which implementation you've used for existing cmdlets you want to port, install your product or feature and then look for the PowerShell module folder in one of the following locations:

- %windir%\system32\WindowsPowerShell\v1.0\Modules
- %ProgramFiles%\WindowsPowerShell\Modules
- %UserProfile%\Documents\WindowsPowerShell\Modules
- <your product installation location>

  Check in these locations for these details:

  - CIM cmdlets have .cdxml file extensions.
  - .NET cmdlets have .dll file extensions, or have assemblies installed to the GAC listed in the .psd1 file under the RootModule, ModuleToProcess, or NestedModules fields.
- PowerShell script cmdlets have .psm1 or .ps1 file extensions.

Porting CIM cmdlets

Generally, these cmdlets should work in Nano Server without any conversion necessary. However, you must port the underlying WMI v2 provider to run on Nano Server if that has not already been done.

Building C++ for Nano Server

To get C++ DLLs working on Nano Server, compile them for Nano Server rather than for a specific edition.

For prerequisites and a walkthrough of developing C++ on Nano Server, see Developing Native Apps on Nano Server.

Porting .NET cmdlets

Most C# code is supported on Nano Server. You can use ApiPort to scan for incompatible APIs.

Powershell Core SDK

The module Microsoft.PowerShell.NanoServer.SDK is available in the PowerShell Gallery to facilitate developing .NET cmdlets using Visual Studio 2015 Update 2 that target the versions of CoreCLR and PowerShell Core available in Nano Server. You can install the module using PowerShellGet with this command:

```
Find-Module Microsoft.PowerShell.NanoServer.SDK -Repository PSGallery | Install-Module -Scope <scope>
```

The PowerShell Core SDK module exposes cmdlets to set up the correct CoreCLR and PowerShell Core reference assemblies, create a C# project in Visual Studio 2015 targeting those reference assemblies, and set up the remote debugger on a Nano Server machine so that developers can debug their .NET cmdlets running on Nano Server remotely in Visual Studio 2015.

The PowerShell Core SDK module requires Visual Studio 2015 Update 2. If you do not have Visual Studio 2015 installed, you can install Visual Studio Community 2015.

The SDK module also depends on the following feature to be installed in Visual Studio 2015:

- Windows and Web Development -> Universal Windows App Development Tools -> Tools (1.3.1) and Windows 10 SDK

Review your Visual Studio installation before using the SDK module to ensure these prerequisites are satisfied. Make sure you select to install the above feature during the Visual Studio installation, or modify your existing Visual Studio 2015 installation to install it.

The PowerShell Core SDK module includes the following cmdlets:

- New-NanoCSharpProject: Creates a new Visual Studio C# project targeting CoreCLR and PowerShell Core included in the Windows Server 2016 release of Nano Server.
- Show-SdkSetupReadMe: Opens the SDK root folder in File Explorer and opens the README.txt file for manual setup.
- Install-RemoteDebugger: Installs and configures the Visual Studio remote debugger on a Nano Server machine.
- Start-RemoteDebugger: Starts the remote debugger on a remote machine running Nano Server.
- Stop-RemoteDebugger: Stops the remote debugger on a remote machine running Nano Server.

For detailed information about how to use those cmdlets, run Get-Help on each cmdlet after installing and importing the module as follows:

```
Get-Command -Module Microsoft.PowerShell.NanoServer.SDK | Get-Help -Full
```

Searching for compatible APIs

You can search in the API catalog for .NET Core or disassemble Core CLR reference assemblies. For more information about platform portability of .NET APIs, see Platform Portability

PInvoke

In the Core CLR that Nano Server uses, some fundamental DLLs such as kernel32.dll and advapi32.dll were split into numerous API sets, so you'll need to ensure that your PInvokes reference the correct API. Any incompatibility will manifest as a runtime error.

For a list of native APIs supported on Nano Server, see Nano Server APIs.

Building C# for Nano Server

Once a C# project is created in Visual Studio 2015 by using `New-NanoCSharpProject`, you can simply build it in Visual Studio by clicking the **Build** menu and selecting **Build Project** or **Build Solution**. The generated assemblies will be targeting the correct CoreCLR and PowerShell Core shipped in Nano Server, and you can just copy the assemblies to a computer running Nano Server and use them.

Building managed C++ (CPP/CLI) for Nano Server

Managed C++ is not supported for CoreCLR. When porting to CoreCLR, rewrite managed C++ code in C# and make all native calls through PInvoke.

Porting PowerShell script cmdlets

PowerShell Core has full PowerShell language parity with other editions of PowerShell, including the edition running on Windows Server 2016 and Windows 10. However, when porting PowerShell script cmdlets to Nano Server, keep these factors in mind:

- Are there dependencies on other cmdlets? If so, are those cmdlets available on Nano Server. See PowerShell on Nano Server for information about what is not available.
- If you have dependencies on assemblies that are loaded at runtime, will they still work?
- How can you debug the script remotely?
- How can you migrate from WMI .Net to MI .Net?

Dependency on built-in cmdlets

Not all cmdlets in Windows Server 2016 are available on Nano Server (see PowerShell on Nano Server). The best approach is to set up a Nano Server virtual machine and discover whether the cmdlets you need are available. To do this, run `Enter-PSSession` to connect to the target Nano Server and then run `Get-Command -CommandType Cmdlet, Function` to get the list of available cmdlets.

Consider using PowerShell classes

Add-Type is supported on Nano Server for compiling inline C# code. If you're writing new code or porting existing code, you might also consider using PowerShell classes to define custom types. You can use PowerShell classes for property bag scenarios as well as for Enums. If you need to do a PInvoke, do this via C# using Add-Type or in a pre-compiled assembly. Here's a sample showing the use of Add-Type:

```
Add-Type -ReferencedAssemblies ([Microsoft.Management.Infrastructure.Ciminstance].Assembly.Location) -
TypeDefinition @'
public class TestNetConnectionResult
{
   // The compute name
   public string ComputerName = null;
   // The Remote IP address used for connectivity
   public System.Net.IPAddress RemoteAddress = null;
}
'@
# Create object and set properties
$result = New-Object TestNetConnectionResult
$result.ComputerName = Foo
$result.RemoteAddress = 1.1.1.1
```

This sample shows using PowerShell classes on Nano Server:

```
class TestNetConnectionResult
{
   # The compute name
  [string] $ComputerName

  #The Remote IP address used for connectivity
  [System.Net.IPAddress] $RemoteAddress
}
# Create object and set properties
$result = [TestNetConnectionResult]::new()
$result.ComputerName = Foo
$result.RemoteAddress = 1.1.1.1
```

Remotely debugging scripts

To remotely debug a script, connect to the remote computer using `Enter-PSsession` from the PowerShell ISE.

Once inside the session, you can run `psedit <file_path>` and a copy of the file will be open in your local PowerShell ISE. Then, you can debug the script as if it were running locally by setting breakpoints. Also, any changes you make to this file will be saved in the remote version.

Migrating from WMI .NET to MI .NET

WMI .NET is not supported, so all cmdlets using the old API must migrate to the supported WMI API: MI. NET. You can access MI .NET directly through C# or through the cmdlets in the CimCmdlets module.

CimCmdlets module

The WMI v1 cmdlets (e.g., Get-WmiObject) are not supported on Nano Server. However, the CIM cmdlets (e.g., Get-CimInstance) in the CimCmdlets module are supported. The CIM cmdlets map pretty closely to the WMI v1 cmdlets. For example, Get-WmiObject correlates with Get-CimInstance using very similar parameters. Method invocation syntax is slightly different, but is well documented via Invoke-CimMethod. Be careful regarding parameter typing. MI .NET has stricter requirements regarding method parameter types.

C# API

WMI .NET wraps the WMIv1 interface, while MI .NET wraps the WMIv2 (CIM) interface. The classes exposed might be different, but the underlying operations are very similar. You enumerate or get instances of objects and invoke operations on them to accomplish tasks.

Troubleshooting Windows volume activation

12/17/2021 • 2 minutes to read • Edit Online

Product activation is the process of validating software after it's installed on a specific computer. Activation confirms that the product is genuine (not a fraudulent copy) and that the product key or serial number is valid and has not been compromised or revoked. Activation also establishes a link or relationship between the product key and the installation.

Volume activation is the process of activating volume-licensed products. To become a volume licensing customer, an organization must set up a volume licensing agreement with Microsoft. Microsoft offers customized volume licensing programs that accommodate the organization's size and purchasing preference. For more information, see the Microsoft Volume Licensing Service Center.

The Windows Server 2016 Activation Guide focuses on the Key Management Service (KMS) activation technology. This section addresses common issues and provides troubleshooting guidelines for KMS and several other volume activation technologies.

Best practices for volume activation

The following articles provide technical information and best practices for Microsoft's volume activation technologies.

Key Management Service (KMS)

- Plan for volume activation
- Understanding KMS
- Deploying KMS Activation
- Configuring KMS Hosts
- Configuring DNS
- Activate using Key Management Service

Active Directory-based activation (ADBA)

- Deploy Active-Directory-based Activation
- Activate using Active Directory-based activation
- Active Directory-Based Activation overview

Multiple Activation Key (MAK) activation

- Using MAK Activation
- Understanding MAK Activation
- Activating MAK Clients

Subscription activation

- Windows 10 Subscription Activation
- Deploy Windows 10 Enterprise licenses
- Windows 10 Enterprise E3 in CSP

Resources for troubleshooting activation issues

The following articles provide guidelines and information about tools for troubleshooting volume activation issues:

- Guidelines for troubleshooting the Key Management Service (KMS)
- Slmgr.vbs options for obtaining volume activation information
- Example: Troubleshooting ADBA clients that do not activate

The following articles provide guidance for addressing more specific activation issues:

- Resolving common activation error codes
- KMS activation: known issues
- MAK activation: known issues
- Guidelines for troubleshooting DNS-related activation issues
- How to rebuild the Tokens.dat file

Guidelines for troubleshooting the Key Management Service (KMS)

12/17/2021 • 13 minutes to read • Edit Online

As part of their deployment process, many enterprise customers set up the Key Management Service (KMS) to enable activation of Windows in their environment. It is a simple process to set up the KMS host, after which the KMS clients discover the host and try to activate on their own. But what happens if that process doesn't work? What do you do next? This article walks you through the resources that you require in order to troubleshoot the issue. For more information about event log entries and the Slmgr.vbs script, see Volume Activation Technical Reference.

KMS overview

Let's start with a quick refresher on KMS activation. KMS is a client-server model. Conceptually, it resembles DHCP. Instead of handing out IP addresses to clients on their request, KMS enables product activation. KMS is also a renewal model, in which the clients try to reactivate on a regular interval. There are two roles: the *KMS host* and the *KMS client*.

- The **KMS host** runs the activation service and enables activation in the environment. To configure a KMS host, you have to install a KMS key from the Volume License Service Center (VLSC) and then activate the service.
- The **KMS client** is the Windows operating system that is deployed in the environment and has to activate. KMS clients can be running any edition of Windows that uses volume activation. The KMS clients are supplied with a pre-installed key, called the Generic Volume License Key (GVLK) or KMS Client Setup Key. The presence of the GVLK is what makes a system a KMS client. The KMS clients use DNS SRV records (_vlmcs._tcp) to identify the KMS host. Then the clients automatically try to discover and use this service to activate themselves. During the 30-day out-of-the-box grace period, they will try to activate every two hours. After activating, the KMS clients try to renew their activation every seven days.

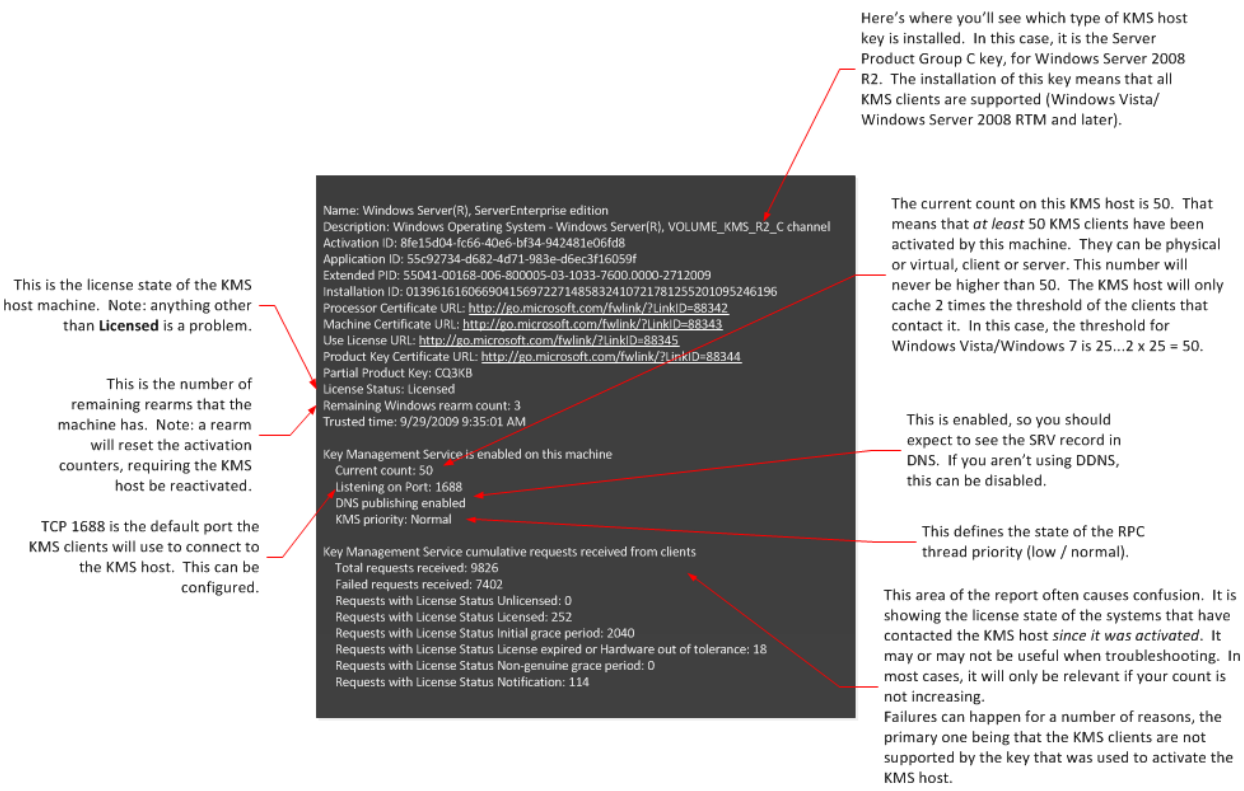
From a troubleshooting perspective, you may have to look at both sides (host and client) to determine what is going on.

KMS host

There are two areas to examine on the KMS host. First, check the status of the host software license service. Second, check the Event Viewer for events that are related to licensing or activation.

Slmgr.vbs and the Software Licensing service

To see verbose output from the Software Licensing service, open an elevated Command Prompt window and enter **slmgr.vbs /dlv** at the command prompt. The following screenshot shows the results of this command on one of our KMS hosts within Microsoft.



The most important fields for troubleshooting are the following. What you are looking for may differ, depending on the issue to be solved.

- **Version Information**. At the top of the **slmgr.vbs /dlv** output is the Software Licensing Service Version. This may be useful to determine whether the current version of the service is installed. For example, updates to the KMS service on Windows Server 2003 support different KMS host keys. This data can be used to evaluate whether or not the version is current and supports the KMS host key that you are trying to install. For more information about these updates, see An update is available for Windows Vista and for Windows Server 2008 to extend KMS activation support for Windows 7 and for Windows Server 2008 R2.

- **Name**. This indicates the edition of Windows that is installed on the KMS host system. This can be important for troubleshooting if you are having trouble adding or changing the KMS host key (for example, to verify that the key is supported on that OS edition).

- **Description**. This is where you see the key that is installed. Use this field to verify which key was used to activate the service and whether or not it is the correct one for the KMS clients that you have deployed.

- **License Status**. This is the status of the KMS host system. The value should be **Licensed**. Any other value means that something is wrong and you may have to reactivate the host.

- **Current Count**. The count displayed will be between **0** and **50**. The count is cumulative (between operating systems) and indicates the number of valid systems that have tried to activate within a 30-day period.

  If the count is **0**, either the service was recently activated or no valid clients have connected to the KMS host.

  The count will not increase above **50**, no matter how many valid systems exist in the environment. This is because they count is set to cache only twice the maximum license policy that is returned by a KMS client. The maximum policy today is set by the Windows client OS, which requires a count of **25** or higher from the KMS host to activate itself. Therefore, the highest count on the KMS host is 2 x 25, or 50. Note that in environments that contain only Windows Server KMS clients, the maximum count on the KMS host will be **10**. This is because the threshold for Windows Server editions is **5** (2 x 5, or 10).

A common issue that is related to the count is if the environment has an activated KMS host and enough clients, but the count does not increase beyond one. The core problem is that the deployed client image was not configured correctly (**sysprep /generalize**) and the systems do not have unique Client Machine IDs (CMIDs). For more information, see KMS client and The KMS current count does not increase when you add new Windows Vista or Windows 7-based client computers to the network. One of our Support Escalation Engineers has also blogged about this issue, in KMS Host Client Count not Increasing Due to Duplicate CMID'S.

Another reason why the count may not be increasing is that there are too many KMS hosts in the environment and the count is distributed over all of them.

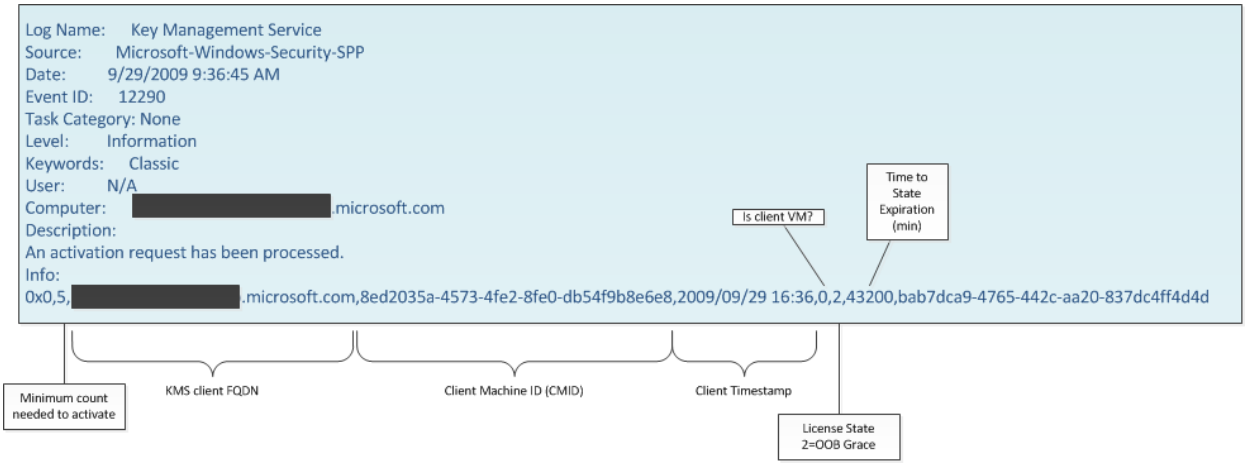
- **Listening on Port**. Communication with KMS uses anonymous RPC. By default, the clients use the 1688 TCP port to connect to the KMS host. Make sure that this port is open between your KMS clients and the KMS host. You can change or configure the port on the KMS host. During their communication, the KMS host sends the port designation to the KMS clients. If you change the port on a KMS client, the port designation is overwritten when that client contacts the host.

We often get asked about the "cumulative requests" section of the **slmgr.vbs /dlv** output. Generally this data is not helpful for troubleshooting. The KMS host keeps an ongoing record of the state of each KMS client that tries to activate or reactivate. Failed requests indicate KMS clients that the KMS host does not support. For example, if a Windows 7 KMS client tries to activate against a KMS host that was activated by using a Windows Vista KMS key, the activation fails. The "Requests with License Status" lines describe all the possible license states, past and present. From a troubleshooting perspective, this data is relevant only if the count is not increasing as expected. In that case, you should see the number of failed requests increasing. This indicates that you should check the product key that was used to activate the KMS host system. Also, notice that the cumulative request values reset only if you reinstall the KMS host system.

Useful KMS host events

Event ID 12290

The KMS host logs Event ID 12290 when a KMS client contacts the host in order to activate. Event ID 12290 provides a significant amount of information that you can use to figure out what kind of client contacted the host and why a failure occurred. The following segment of an event ID 12290 entry comes from the Key Management Service event log of our KMS host.



The event details include the following information:

- **Minimum count needed to activate**. The KMS client is reporting that the count from the KMS host must be **5** in order to activate. That means that this is a Windows Server OS, although it does not indicate a specific edition. If your clients are not activating, make sure that the count is sufficient on the host.
- **Client Machine ID (CMID)**. This is a unique value on each system. If this value is not unique, it is because an image was not prepared correctly for distribution (**sysprep /generalize**). This issue manifests on the KMS host as a count that will not increase, even though there are enough clients in the environment. For

more information, see The KMS current count does not increase when you add new Windows Vista or Windows 7-based client computers to the network.

- **License State and Time to State Expiration**. This is the current license state of the client. It can help you differentiate a client that is trying to activate for the first time from one that is trying to reactivate. The time entry tells you how much longer the client will remain in that state, if nothing changes.

If you are troubleshooting a client and cannot find a corresponding event ID 12290 on the KMS host, that client is not connecting to the KMS host. Some reasons why an event ID 12290 entry may not exist are as follows:

- A network outage has occurred.
- The host is not resolving or is not registered in DNS.
- The firewall is blocking TCP 1688. The port could be blocked in many places within the environment, including on the KMS host system itself. By default, the KMS host has a firewall exception for KMS, but it is not automatically enabled. You have to turn on the exception.
- The event log is full.

KMS clients log two corresponding events, event ID 12288 and event ID 12289. For information about these events, see the KMS client section.

Event ID 12293

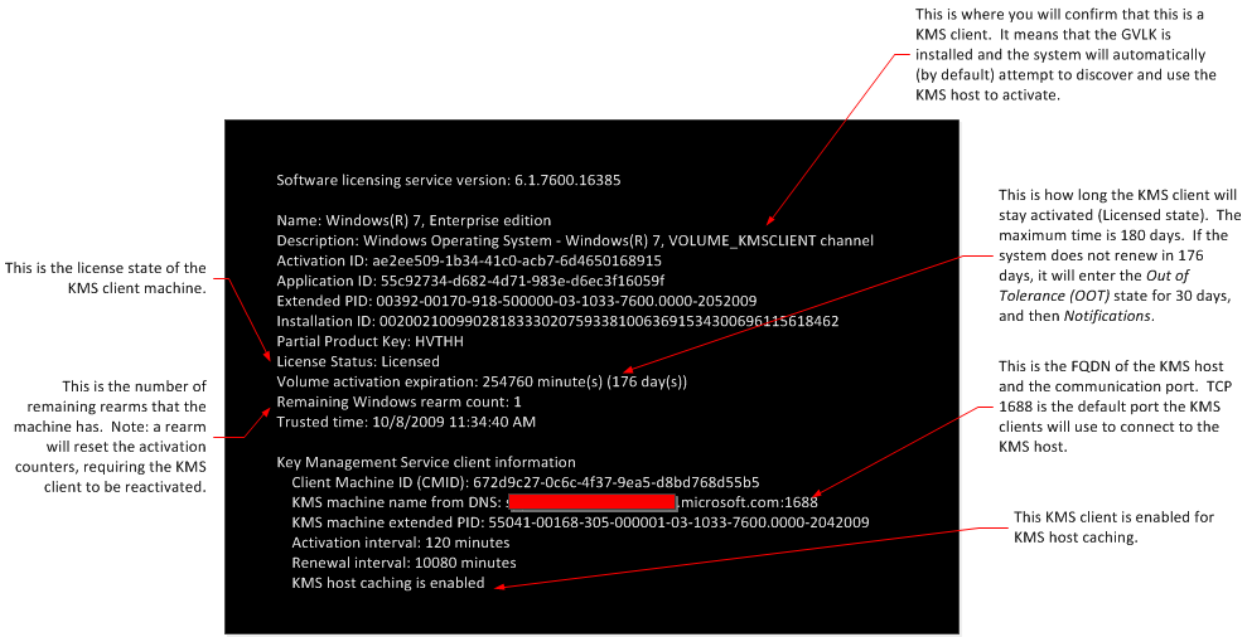
Another relevant event to look for on your KMS host is event ID 12293. This event indicates that the host did not publish the required records in DNS. This situation is known to cause failures, and it is something that you should verify *after* you set up your host and *before* you deploy clients. For more information about DNS issues, see Common troubleshooting procedures for KMS and DNS issues.

KMS client

On the clients you use the same tools (Slmgr and Event Viewer) to troubleshoot activation.

Slmgr.vbs and the Software Licensing service

To see verbose output from the Software Licensing service, open an elevated Command Prompt window and enter **slmgr.vbs /dlv** at the command prompt. The following screenshot shows the results of this command on one of our KMS hosts within Microsoft.



The following list includes the most important fields for troubleshooting. What you are looking for may differ, depending on the issue to be solved.

- **Name**. This value is the edition of Windows that is installed on the KMS client system. Use this to verify that

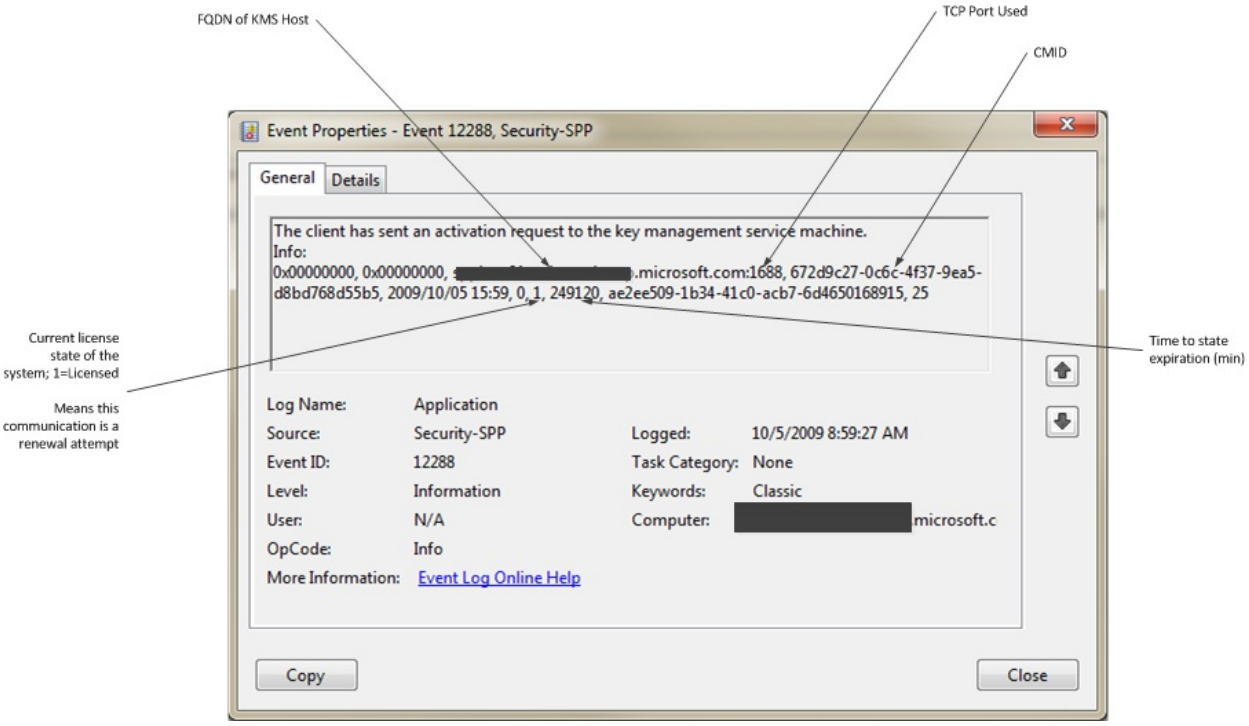
the version of Windows you are trying to activate can use KMS. For example, our Help desk has seen incidents in which customers try to install the KMS Client Setup Key on an edition of Windows that does not use volume activation, such as Windows Vista Ultimate.

- **Description**. This value shows the key that is installed. VOLUME_KMSCLIENT indicates that the KMS Client Setup Key (or GVLK) is installed (the default configuration for volume license media) and that this system automatically tries to activate by using a KMS host. If you see something else here, such as MAK, you'll have to reinstall the GVLK to configure this system as a KMS client. You can manually install the key by using **slmgr.vbs /ipk** *<GVLK>* (as described in KMS client setup keys) or use the Volume Activation Management Tool (VAMT). For information about obtaining and using VAMT, see Volume Activation Management Tool (VAMT) Technical Reference.
- **Partial Product Key**. As the **Name** field, you can use this information to determine whether the correct KMS Client Setup Key is installed on this computer (in other words, the key matches the operating system that is installed on the KMS client). By default, the correct key is present on systems that are built by using media from the Volume License Service Center (VLSC) portal. In some cases, customers may use Multiple Activation Key (MAK) activation until there are enough systems in the environment to support KMS activation. The KMS Client Setup key has to be installed on these systems to transition them from MAK to KMS. Use VAMT to install this key and make sure that the correct key is applied.
- **License Status**. This value shows the status of the KMS client system. For a system that was activated by using KMS, this value should be **Licensed**. Any other value may indicate that there is a problem. For example, if the KMS host is functioning correctly and the KMS client does not activate (for example, it remains in a **Grace** state), something may be preventing the client from reaching the host system (such as a firewall issue, network outage, or something similar).
- **Client Machine ID (CMID)**. Each KMS client should have a unique CMID. As mentioned in the KMS host section, a common issue related to count is if the environment has an activated KMS host and enough clients, but the count does not increase beyond **1**. For more information, see The KMS current count does not increase when you add new Windows Vista or Windows 7-based client computers to the network.
- **KMS Machine Name from DNS**. This value shows the FQDN of the KMS host that the client successfully used for activation, and the TCP port used for the communication.
- **KMS Host Caching**. The final value shows whether or not caching is enabled. By default, it is enabled. What this means is that the KMS client caches the name KMS host that it used for activation, and it communicates directly with this host (instead of querying DNS) when it is time to reactivate. If the client cannot contact the cached KMS host, it queries DNS to discover a new KMS host.

Useful KMS client events

Event ID 12288 and Event ID 12289

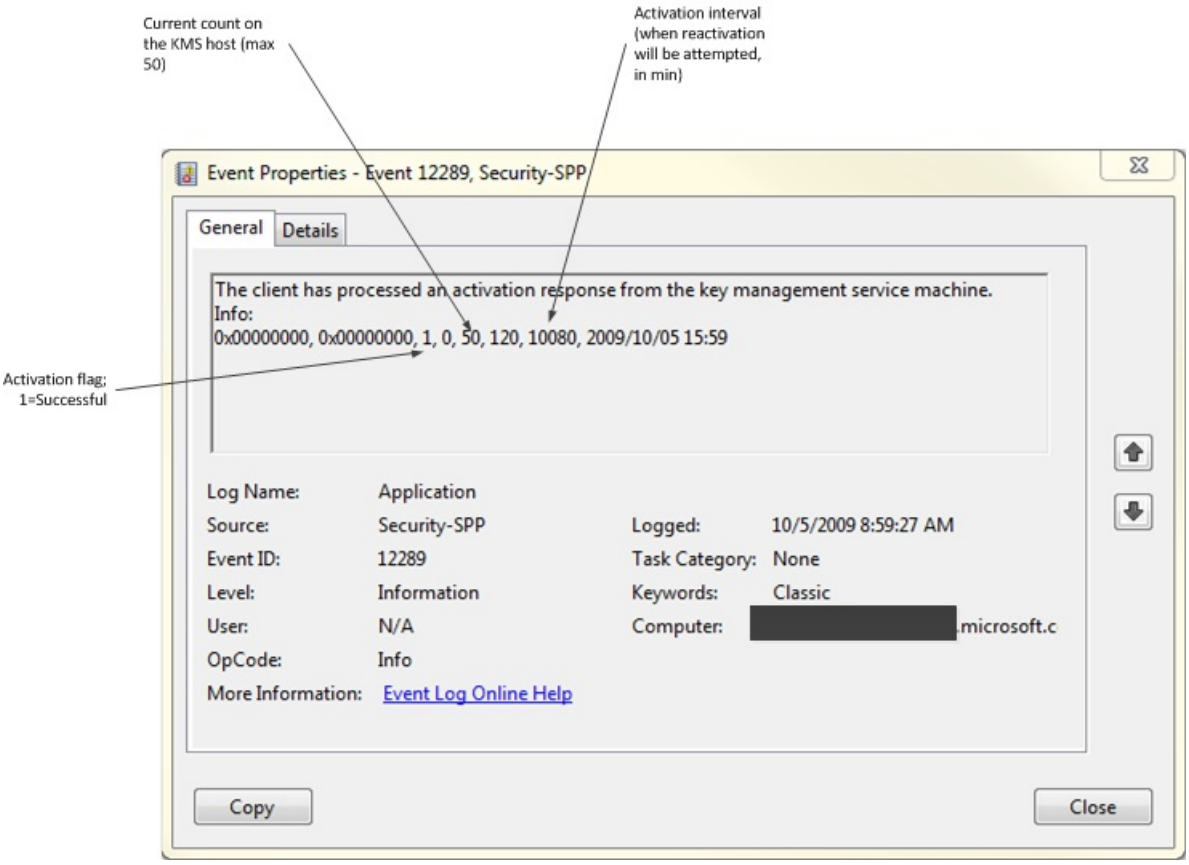
When a KMS client successfully activates or reactivates, the client logs two events: event ID 12288 and event ID 12289. The following segment of an event ID 12288 entry comes from the Key Management Service event log of our KMS client.



If you see only event ID 12288 (without a corresponding event ID 12289), this means that the KMS client was not able to reach the KMS host, the KMS host did not respond, or the client did not receive the response. In this case, verify that the KMS host is discoverable and that the KMS clients can contact it.

The most relevant information in event ID 12288 is the data in the Info section. For example, this section shows the current state of the client plus the FQDN and TCP port that the client used when it tried to activate. You can use the FQDN to troubleshoot cases in which the count on a KMS host is not increasing. For example, if there are too many KMS hosts available to the clients (either legitimate or rogue systems) then the count may be distributed over all of them.

An unsuccessful activation does not always mean that the client has 12288 and not 12289. A failed activation or reactivation may also have both events. In this case, you have to examine the second event to verify the reason for the failure.



The Info section of event ID 12289 provides the following information:

- **Activation Flag**. This value indicates whether the activation succeeded(**1**) or failed (**0**).
- **Current Count on the KMS Host**. This value reflects the count value on the KMS host when the client tries to activate. If activation fails, it may be because the count is insufficient for this client OS or that there are not enough systems in the environment to build the count.

What does support ask for?

If you have to call Support to troubleshoot activation, the Support Engineer typically asks for the following information:

- **Slmgr.vbs /dlv** output from the KMS host and KMS client systems. Whether you use wscript or cscript to run the command, you can use Ctrl+C to copy the output, and then paste it into Notepad to send it to the support contact.
- Event logs from both the KMS host (Key Management Service log) and KMS client systems (Application log)

Additional References

- Ask the Core Team: #Activation

Slmgr.vbs options for obtaining volume activation information

12/17/2021 • 10 minutes to read • Edit Online

The following describes the syntax of the Slmgr.vbs script, and the tables in this article describe each command-line option.

```
slmgr.vbs [<ComputerName> [<User> <Password>]] [<Options>]
```

NOTE

In this article, square brackets [] enclose optional arguments, and angle brackets <> enclose placeholders. When you type these statements, omit the brackets and replace the placeholders by using corresponding values.

NOTE

For information about other software products that use volume activation, see the documents specifically written for those applications.

Using Slmgr on remote computers

To manage remote clients, use the Volume Activation Management Tool (VAMT) version 1.2 or later, or create custom WMI scripts that are aware of the differences between platforms. For more information about WMI properties and methods for Volume Activation, see WMI Properties and Methods for Volume Activation.

IMPORTANT

Because of WMI changes in Windows 7 and Windows Server 2008 R2, the Slmgr.vbs script is not intended to work across platforms. Using Slmgr.vbs to manage a Windows 7 or Windows Server 2008 R2 system from the Windows Vista® operating system is not supported. Trying to manage an older system from Windows 7 or Windows Server 2008 R2 will generate a specific version mismatch error. For example, running **cscript slmgr.vbs <vista_machine_name> /dlv** produces the following output:

```
Microsoft (R) Windows Script Host Version 5.8 Copyright (C) Microsoft Corporation. All rights reserved.

The remote machine does not support this version of SLMgr.vbs
```

General Slmgr.vbs options

OPTION	DESCRIPTION
[<ComputerName>]	Name of a remote computer (default is local computer)
[<User>]	Account that has the required privilege on the remote computer

OPTION	DESCRIPTION
[<Password>]	Password for the account that has the required privileges on the remote computer

Global options

OPTION	DESCRIPTION
/ipk <ProductKey>	Tries to install a 5×5 product key. The product key provided by the parameter is confirmed valid and applicable to the installed operating system. If not, an error is returned. If the key is valid and applicable, the key is installed. If a key is already installed, it is silently replaced. To prevent instability in the license service, the system should be restarted or the Software Protection Service should be restarted. This operation must be run from an elevated Command Prompt window, or the Standard User Operations registry value must be set to allow unprivileged users extra access to the Software Protection Service.
/ato [<Activation ID>]	For retail editions and volume systems that have a KMS host key or a Multiple Activation Key (MAK) installed, **/ato** prompts Windows to try online activation. For systems that have a Generic Volume License Key (GVLK) installed, this prompts a KMS activation attempt. Systems that have been set to suspend automatic KMS activation attempts (**/stao**) still try KMS activation when **/ato** is run. **Note:** Starting in Windows 8 (and Windows Server 2012), the **/stao** option is deprecated. Use the **/act-type** option instead. The parameter <**Activation ID**> expands **/ato** support to identify a Windows edition installed on the computer. Specifying the <**Activation ID**> parameter isolates the effects of the option to the edition associated with that Activation ID. Run **slmgr.vbs /dlv all** to get the Activation IDs for the installed version of Windows. If you have to support other applications, see the guidance provided by that application for further instruction. KMS activation does not require elevated privileges. However, online activation does require elevation, or the Standard User Operations registry value must be set to allow unprivileged users extra access to the Software Protection Service.
/dli [<Activation ID> | All]	Display license information. By default, **/dli** displays the license information for the installed active Windows edition. Specifying the <**Activation ID**> parameter displays the license information for the specified edition that is associated with that Activation ID. Specifying **All** as the parameter displays license information for all applicable installed products. This operation does not require elevated privileges.

OPTION	DESCRIPTION
/dlv [<Activation ID> | All]	Display detailed license information. By default, **/dlv** displays the license information for the installed operating system. Specifying the <**Activation ID**> parameter displays the license information for the specified edition associated with that Activation ID. Specifying the **All** parameter displays license information for all applicable installed products. This operation does not require elevated privileges.
/xpr [<Activation ID>]	Display the activation expiration date for the product. By default, this refers to the current Windows edition and is primarily useful for KMS clients, because MAK and retail activation is perpetual. Specifying the <**Activation ID**> parameter displays the activation expiration date of the specified edition that is associated with that Activation ID.This operation does not require elevated privileges.

Advanced options

OPTION	DESCRIPTION
/cpky	Some servicing operations require the product key to be available in the registry during Out-of-Box Experience (OOBE) operations. The **/cpky** option removes the product key from the registry to prevent this key from being stolen by malicious code. For retail installations that deploy keys, best practices recommend running this option. This option is not required for MAK and KMS host keys, because this is the default behavior for those keys. This option is required only for other types of keys whose default behavior is not to clear the key from the registry. This operation must be run in an elevated Command Prompt window.
/ilc <license_file>	This option installs the license file specified by the required parameter. These licenses may be installed as a troubleshooting measure, to support token-based activation, or as part of a manual installation of an on-boarded application. Licenses are not validated during this process: License validation is out of scope for Slmgr.vbs. Instead, validation is handled by the Software Protection Service at runtime. This operation must be run from an elevated Command Prompt window, or the **Standard User Operations** registry value must be set to allow unprivileged users extra access to the Software Protection Service.

OPTION	DESCRIPTION
/rilc	This option reinstalls all licenses stored in %SystemRoot%\system32\oem and %SystemRoot%\System32\spp\tokens. These are "known-good" copies that were stored during installation. Any matching licenses in the Trusted Store are replaced. Any additional licenses—for example, Trusted Authority (TA) Issuance Licenses (ILs), licenses for applications—are not affected. This operation must be run in an elevated Command Prompt window, or the **Standard User Operations** registry value must be set to allow unprivileged users extra access to the Software Protection Service.
/rearm	This option resets the activation timers. The **/rearm** process is also called by **sysprep /generalize**. This operation does nothing if the **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SoftwareProtectionPlatform\SkipRearm** registry entry is set to **1**. See Registry Settings for Volume Activation for details about this registry entry. This operation must be run in an elevated Command Prompt window, or the **Standard User Operations** registry value must be set to allow unprivileged users extra access to the Software Protection Service.
/rearm-app <Application ID>	Resets the licensing status of the specified app.
/rearm-sku <Application ID>	Resets the licensing status of the specified SKU.
/upk [<Application ID>]	This option uninstalls the product key of the current Windows edition. After a restart, the system will be in an Unlicensed state unless a new product key is installed. Optionally, you can use the <**Activation ID**> parameter to specify a different installed product. This operation must be run from an elevated Command Prompt window.
/dti [<Activation ID>]	Displays installation ID for offline activation.
/atp <Confirmation ID>	Activate product by using user-provided confirmation ID.

KMS client options

OPTION	DESCRIPTION
/skms <Name[:Port] | : port> [<Activation ID>]	This option specifies the name and, optionally, the port of the KMS host computer to contact. Setting this value disables auto-detection of the KMS host. If the KMS host uses Internet Protocol version 6 (IPv6) only, the address must be specified in the format <hostname>:<port>. IPv6 addresses contain colons (:), which the Slmgr.vbs script does not parse correctly. This operation must be run in an elevated Command Prompt window.

OPTION	DESCRIPTION
/skms-domain <FQDN> [<Activation ID>]	Sets the specific DNS domain in which all KMS SRV records can be found. This setting has no effect if the specific single KMS host is set by using the **/skms** option. Use this option, especially in disjoint namespace environments, to force KMS to ignore the DNS suffix search list and look for KMS host records in the specified DNS domain instead.
/ckms [<Activation ID>]	This option removes the specified KMS host name, address, and port information from the registry and restores KMS auto-discovery behavior. This operation must be run in an elevated Command Prompt window.
/skhc	This option enables KMS host caching (default). After the client discovers a working KMS host, this setting prevents the Domain Name System (DNS) priority and weight from affecting further communication with the host. If the system can no longer contact the working KMS host, the client tries to discover a new host. This operation must be run in an elevated Command Prompt window.
/ckhc	This option disables KMS host caching. This setting instructs the client to use DNS auto-discovery each time it tries KMS activation (recommended when using priority and weight). This operation must be run in an elevated Command Prompt window.

KMS host configuration options

OPTION	DESCRIPTION
/sai <Interval>	This option sets the interval in minutes for unactivated clients to try to connect to KMS. The activation interval must be between 15 minutes and 30 days, although the default value (two hours) is recommended. The KMS client initially picks up this interval from registry but switches to the KMS setting after it receives the first KMS response. This operation must be run in an elevated Command Prompt window.
/sri <Interval>	This option sets the renewal interval in minutes for activated clients to try to connect to KMS. The renewal interval must be between 15 minutes and 30 days. This option is set initially on both the KMS server and client sides. The default value is 10,080 minutes (7 days). The KMS client initially picks up this interval from the registry but switches to the KMS setting after it receives the first KMS response. This operation must be run in an elevated Command Prompt window.
/sprt <Port>	This option sets the port on which the KMS host listens for client activation requests. The default TCP port is 1688. This operation must be run from an elevated Command Prompt window.

OPTION	DESCRIPTION
/sdns	Enable DNS publishing by the KMS host (default). This operation must be run in an elevated Command Prompt window.
/cdns	Disable DNS publishing by the KMS host. This operation must be run in an elevated Command Prompt window.
/spri	Set the KMS priority to normal (default). This operation must be run in an elevated Command Prompt window.
/cpri	Set the KMS priority to low. Use this option to minimize contention from KMS in a co-hosted environment. Note that this could cause KMS starvation, depending on what other applications or server roles are active. Use with care. This operation must be run in an elevated Command Prompt window.
/act-type [<Activation-Type>] [<Activation ID>]	This option sets a value in the registry that limits volume activation to a single type. Activation Type **1** limits activation to Active Directory only; **2** limits it to KMS activation; **3** to token-based activation. The **0** option allows any activation type and is the default value.

Token-based activation configuration options

OPTION	DESCRIPTION
/lil	List the installed token-based activation issuance licenses.
/ril <ILID> <ILvID>	Remove an installed token-based activation issuance license. This operation must be run from an elevated Command Prompt window.
/stao	Set the **Token-based Activation Only** flag, disabling automatic KMS activation. This operation must be run in an elevated Command Prompt window. This option was removed in Windows Server 2012 R2 and Windows 8.1. Use the **/act–type** option instead.
/ctao	Clear the **Token-based Activation Only** flag (default), enabling automatic KMS activation. This operation must be run in an elevated Command Prompt window. This option was removed in Windows Server 2012 R2 and Windows 8.1. Use the **/act–type** option instead.
/ltc	List valid token-based activation certificates that can activate installed software.

OPTION	DESCRIPTION
/fta <Certificate Thumbprint> [<PIN>]	Force token-based activation by using the identified certificate. The optional personal identification number (PIN) is provided to unlock the private key without a PIN prompt if you use certificates that are protected by hardware (for example, smart cards).

Active Directory-based activation configuration options

OPTION	DESCRIPTION
/ad-activation-online <Product Key> [<Activation Object name>]	Collects Active Directory data and starts Active Directory forest activation using the credentials that the command prompt is running. Local administrator access is not required. However, Read/Write access to the activation object container in the root domain of the forest is required.
/ad-activation-get-IID <Product Key>	This option starts Active Directory forest activation in phone mode. The output is the installation ID (IID) that can be used to activate the forest over the telephone if internet connectivity is not available. Upon providing the IID in the activation phone call, a CID is returned that is used to complete activation.
/ad-activation-apply-cid <Product Key> <Confirmation ID> [<Activation Object name>]	When you use this option, enter the CID that was provided in the activation telephone call to complete activation
[/name: <AO_Name>]	Optionally, you can append the **/name** option to any of these commands to specify a name for the activation object stored in Active Directory. The name must not exceed 40 Unicode characters. Use double quotation marks to explicitly define the name string. In Windows Server 2012 R2 and Windows 8.1, you can append the name directly after **/ad-activation-online <Product Key>** and **/ad-activation-apply-cid** without having to use the **/name** option.
/ao-list	Displays all of the activation objects that are available to the local computer.
/del-ao <AO_DN> /del-ao <AO_RDN>	Deletes the specified activation object from the forest.

Additional References

- Volume Activation Technical Reference
- Volume Activation Overview

Resolve Windows activation error codes

12/17/2021 • 14 minutes to read • Edit Online

> **NOTE**
> This article is intended for technical support agents and IT professionals. If you're looking for more information about Windows activation error messages, see Get help with Windows activation errors.

This article provides troubleshooting information to help you respond to error messages that you may receive when you try to use a Multiple Activation Key (MAK) or the Key Management Service (KMS) to perform Volume Activation on one or more Windows-based computers. Look for the error code in the following table, and then select the link to see more information about that error code and how to resolve it.

For more information about volume activation, see Plan for volume activation.

For more information about volume activation for current and recent versions of Windows, see Volume Activation [client].

For more information about volume activation for older versions of Windows, see KB 929712, Volume Activation information for Windows Vista, Windows Server 2008, Windows Server 2008 R2 and Windows 7.

Diagnostic tool

> **NOTE**
> This tool is intended to help fix Windows activation problems on computers that run Enterprise, Professional, or Server edition of Windows.

Microsoft Support and Recovery Assistant (SaRA) simplifies Windows KMS Activation troubleshooting. Download the diagnostic tool from here.

This tool will try to activate Windows. If it returns an activation error code, the tool will display targeted solutions for known error codes.

The following error codes are supported: 0xC004F038, 0xC004F039, 0xC004F041, 0xC004F074, 0xC004C008, 0x8007007b, 0xC004C003, 0x8007232B.

Summary of error codes

ERROR CODE	ERROR MESSAGE	ACTIVATION TYPE
0x8004FE21	This computer is not running genuine Windows.	MAK KMS client
0x80070005	Access denied. The requested action requires elevated privileges.	MAK KMS client KMS host
0x8007007b	0x8007007b DNS name does not exist.	KMS client

ERROR CODE	ERROR MESSAGE	ACTIVATION TYPE
0x80070490	The product key you entered didn't work. Check the product key and try again, or enter a different one.	MAK
0x800706BA	The RPC server is unavailable.	KMS client
0x8007232A	DNS server failure.	KMS host
0x8007232B	DNS name does not exist.	KMS client
0x8007251D	No records found for DNS query.	KMS client
0x80092328	DNS name does not exist.	KMS client
0xC004B100	The activation server determined that the computer could not be activated.	MAK
0xC004C001	The activation server determined the specified product key is invalid	MAK
0xC004C003	The activation server determined the specified product key is blocked	MAK
0xC004C008	The activation server determined that the specified product key could not be used.	KMS
0xC004C020	The activation server reported that the Multiple Activation Key has exceeded its limit.	MAK
0xC004C021	The activation server reported that the Multiple Activation Key extension limit has been exceeded.	MAK
0xC004F009	The Software Protection Service reported that the grace period expired.	MAK
0xC004F00F	The Software Licensing Server reported that the hardware ID binding is beyond level of tolerance.	MAK KMS client KMS host
0xC004F014	The Software Protection Service reported that the product key is not available	MAK KMS client
0xC004F02C	The Software Protection Service reported that the format for the offline activation data is incorrect.	MAK KMS client
0xC004F035	The Software Protection Service reported that the computer could not be activated with a Volume license product key.	KMS client KMS host

ERROR CODE	ERROR MESSAGE	ACTIVATION TYPE
0xC004F038	The Software Protection Service reported that the computer could not be activated. The count reported by your Key Management Service (KMS) is insufficient. Please contact your system administrator.	KMS client
0xC004F039	The Software Protection Service reported that the computer could not be activated. The Key Management Service (KMS) is not enabled.	KMS client
0xC004F041	The Software Protection Service determined that the Key Management Server (KMS) is not activated. KMS needs to be activated.	KMS client
0xC004F042	The Software Protection Service determined that the specified Key Management Service (KMS) cannot be used.	KMS client
0xC004F050	The Software Protection Service reported that the product key is invalid.	MAK KMS KMS client
0xC004F051	The Software Protection Service reported that the product key is blocked.	MAK KMS
0xC004F064	The Software Protection Service reported that the non-genuine grace period expired.	MAK
0xC004F065	The Software Protection Service reported that the application is running within the valid non-genuine period.	MAK KMS client
0xC004F06C	The Software Protection Service reported that the computer could not be activated. The Key Management Service (KMS) determined that the request timestamp is invalid.	KMS client
0xC004F074	The Software Protection Service reported that the computer could not be activated. No Key Management Service (KMS) could be contacted. Please see the Application Event Log for additional information.	KMS client

Causes and resolutions

0x8004FE21 This computer is not running genuine Windows

Possible cause

This issue can occur for several reasons. The most likely reason is that language packs (MUI) have been installed on computers that are running Windows editions that are not licensed for additional language packs.

> **NOTE**
> This issue is not necessarily an indication of tampering. Some applications can install multi-lingual support even when that edition of Windows is not licensed for those language packs.)

This issue may also occur if Windows has been modified by malware to allow additional features to be installed. This issue may also occur if certain system files are corrupted.

Resolution

To resolve this issue, you must reinstall the operating system.

0x80070005 Access denied

The full text of this error message resembles the following:

> Access denied. The requested action requires elevated privileges.

Possible cause

User Account Control (UAC) prohibits activation processes from running in a non-elevated Command Prompt window.

Resolution

Run **slmgr.vbs** from an elevated command prompt. To do this, on the **Start menu**, right-click **cmd.exe**, and then select **Run as administrator**.

0x8007007b DNS name does not exist

Possible cause

This issue may occur if the KMS client cannot find the KMS SRV resource records in DNS.

Resolution

For more information about troubleshooting such DNS-related issues, see Common troubleshooting procedures for KMS and DNS issues.

0x80070490 The product key you entered didn't work

The full text of this error resembles the following:

> The product key that you entered didn't work. Check the product key and try again, or enter a different one.

Possible cause

This issue occurs because the MAK that was entered was not valid, or because of a known issue in Windows Server 2019.

Resolution

To work around this issue and activate the computer, run **slmgr -ipk <5x5 key>** at an elevated command prompt.

0x800706BA The RPC server is unavailable

Possible cause

Firewall settings are not configured on the KMS host, or DNS SRV records are stale.

Resolution

On the KMS host, make sure that a firewall exception is enabled for the Key Management Service (TCP port 1688).

Make sure that the DNS SRV records point to a valid KMS host.

Troubleshoot network connections.

For more information about troubleshooting such DNS-related issues, see Common troubleshooting procedures for KMS and DNS issues.

0x8007232A DNS server failure

Possible cause

The system has network or DNS issues.

Resolution

Troubleshoot network and DNS.

0x8007232B DNS name does not exist

Possible cause

The KMS client cannot find KMS server resource records (SRV RRs) in DNS.

Resolution

Verify that a KMS host has been installed and DNS publishing is enabled (default). If DNS is unavailable, point the KMS client to the KMS host by using **slmgr.vbs /skms <*kms_host_name*>**.

If you do not have a KMS host, obtain and install an MAK. Then, activate the system.

For more information about troubleshooting such DNS-related issues, see Common troubleshooting procedures for KMS and DNS issues.

0x8007251D No records found for DNS query

Possible cause

The KMS client cannot find KMS SRV records in DNS.

Resolution

Troubleshoot network connections and DNS. For more information about how to troubleshoot such DNS-related issues, see Common troubleshooting procedures for KMS and DNS issues.

0x80092328 DNS name does not exist

Possible cause

This issue may occur if the KMS client cannot find the KMS SRV resource records in DNS.

Resolution

For more information about troubleshooting such DNS-related issues, see Common troubleshooting procedures for KMS and DNS issues.

0xC004B100 The activation server determined that the computer could not be activated

Possible cause

The MAK is not supported.

Resolution

To troubleshoot this issue, verify that the MAK that you are using is the MAK that was provided by Microsoft. To verify that the MAK is valid, contact the Microsoft Licensing Activation Centers.

0xC004C001 The activation server determined the specified product key is invalid

Possible cause

The MAK that you entered is not valid.

Resolution

Verify that the key is the MAK that was provided by Microsoft. For additional assistance, contact the Microsoft Licensing Activation Centers.

0xC004C003 The activation server determined the specified product key is blocked

Possible cause

The MAK is blocked on the activation server.

Resolution

To obtain a new MAK, contact the Microsoft Licensing Activation Centers. After you obtain the new MAK, try installing and activating Windows again.

0xC004C008 The activation server determined that the specified product key could not be used

Possible cause

The KMS key has exceeded its activation limit. A KMS host key can be activated up to 10 times on up to six different computers.

Resolution

If you require additional activations, contact the Microsoft Licensing Activation Centers.

0xC004C020 The activation server reported that the Multiple Activation Key has exceeded its limit

Possible cause

The MAK has exceeded its activation limit. By design, MAKs can be activated a limited number of times.

Resolution

If you require additional activations, contact the Microsoft Licensing Activation Centers.

0xC004C021 The activation server reported that the Multiple Activation Key extension limit has been exceeded

Possible cause

The MAK has exceeded its activation limit. By design, MAKs activate a limited number of times.

Resolution

If you need additional activations, contact the Microsoft Licensing Activation Centers.

0xC004F009 The Software Protection Service reported that the grace period expired

Possible cause

The grace period expired before the system was activated. Now, the system is in the Notifications state.

Resolution

For assistance, contact the Microsoft Licensing Activation Centers.

0xC004F00F The Software Licensing Server reported that the hardware ID binding is beyond level of tolerance

Possible cause

The hardware has changed or the drivers were updated on the system.

Resolution

If you are using MAK activation, use either online or phone activation to reactivate the system during the OOT grace period.

If you are using KMS activation, restart Windows or run `slmgr.vbs /ato`.

0xC004F014 The Software Protection Service reported that the product key is not available

Possible cause

No product keys are installed on the system.

Resolution

If you are using MAK activation, install a MAK product key.

If you are using KMS activation, check the Pid.txt file (located on the installation media in the \sources folder) for a KMS Setup key. Install the key.

0xC004F02C The Software Protection Service reported that the format for the offline activation data is incorrect

Possible cause

The system has detected that the data entered during phone activation is not valid.

Resolution

Verify that the CID is entered correctly.

0xC004F035 Invalid Volume License Key

The full text of this error message resembles the following:

> Error: Invalid Volume License Key. In order to activate, you need to change your product key to a valid Multiple Activation Key (MAK) or Retail key. You must have a qualifying operating system license AND a Volume license Windows 7 upgrade license, or a full license for Windows 7 from a retail source. ANY OTHER INSTALLATION OF THIS SOFTWARE IS IN VIOLATION OF YOUR AGREEMENT AND APPLICABLE COPYRIGHT LAW.

The error text is correct, but is ambiguous. This error indicates that the computer is missing a Windows marker in its BIOS that identifies it as an OEM system that is running a qualifying edition of Windows. This information is required for KMS client activation. The more specific meaning of this code is "Error: Invalid Volume License Key"

Possible cause

Windows 7 Volume editions are licensed only for upgrade. Microsoft does not support installing a Volume operating system on a computer that does not have a qualifying operating system installed.

Resolution

In order to activate, you need to do one of the following:

- Change your product key to a valid Multiple Activation Key (MAK) or Retail key. You must have a qualifying operating system license AND a Volume license Windows 7 upgrade license, or a full license for Windows 7 from a retail source.

  > **NOTE**
  > If you receive error 0x80072ee2 when you attempt to activate, use the phone activation method that follows instead.

- Activate by phone by following these steps:
  1. Run **slmgr /dti** and then record the value of the Installation ID.
- Contact the Microsoft Licensing Activation Centers and provide the Installation ID in order to receive a Confirmation ID.
- To activate by using the Confirmation ID, run **slmgr /atp <Confirmation ID>**.

0xC004F038 The count reported by your Key Management Service (KMS) is insufficient

The full text of this error message resembles the following:

> The Software Protection Service reported that the computer could not be activated. The count reported by your Key Management Service (KMS) is insufficient. Please contact your system administrator.

Possible cause

The count on the KMS host is not high enough. For Windows Server, the KMS count must be greater than or equal to 5. For Windows (client), the KMS count must be greater than or equal to 25.

Resolution

Before you can use KMS to activate Windows, you must have more computers in the KMS pool. To obtain the current count on the KMS host, run **Slmgr.vbs /dli**.

0xC004F039 The Key Management Service (KMS) is not enabled

The full text of this error message resembles the following:

> The Software Protection Service reported that the computer could not be activated. The Key Management Service (KMS) is not enabled.

Possible cause

KMS did not respond to the KMS request.

Resolution

Troubleshoot the network connection between the KMS host and the client. Make sure that TCP port 1688 (default) is not blocked by a firewall or is otherwise filtered.

0xC004F041 The Software Protection Service determined that the Key Management Server (KMS) is not activated

The full text of this error message resembles the following:

> The Software Protection Service determined that the Key Management Server (KMS) is not activated. KMS needs to be activated.

Possible cause

The KMS host is not activated.

Resolution

Activate the KMS host by using either online or telephone activation.

0xC004F042 The Software Protection Service determined that the specified Key Management Service (KMS) cannot be used

Possible cause

This error occurs if the KMS client contacted a KMS host that could not activate the client software. This can be common in mixed environments that contain application-specific and operating system-specific KMS hosts, for example.

Resolution

Make sure that if you use specific KMS hosts to activate specific applications or operating systems, the KMS clients connect to the correct hosts.

0xC004F050 The Software Protection Service reported that the product key is invalid

Possible cause

This can be caused by a typo in the KMS key or by typing in a Beta key on a Released version of the operating system.

Resolution

Install the appropriate KMS key on the corresponding version of Windows. Check the spelling. If the key is being copied and pasted, make sure that em-dashes were not substituted for the hyphens in the key.

0xC004F051 The Software Protection Service reported that the product key is blocked

Possible cause

The activation server determined that Microsoft has blocked the product key.

Resolution

Obtain a new MAK or KMS key, install it on the system, and activate.

0xC004F064 The Software Protection Service reported that the non-genuine grace period expired

Possible cause

Windows Activation Tools (WAT) has determined that the system is not genuine.

Resolution

For assistance, contact the Microsoft Licensing Activation Centers.

0xC004F065 The Software Protection Service reported that the application is running within the valid non-genuine period

Possible cause

Windows Activation Tools has determined that the system is not genuine. The system will continue to run during the Non-Genuine grace period.

Resolution

Obtain and install a genuine product key, and activate the system during the grace period. Otherwise, the system will go into the Notifications state at the end of the grace period.

0xC004F06C The request timestamp is invalid

The full text of this error message resembles the following:

> The Software Protection Service reported that the computer could not be activated. The Key Management Service (KMS) determined that the request timestamp is invalid.

Possible cause

The system time on the client computer is too different from the time on the KMS host. Time sync is important to system and network security for a variety of reasons.

Resolution

Fix this issue by changing the system time on the client to sync with the KMS host. We recommend that you use a Network Time Protocol (NTP) time source or Active Directory Domain Services for time synchronization. This issue uses UTP time and does not depend on Time Zone selection.

0xC004F074 No Key Management Service (KMS) could be contacted

The full text of this error message resembles the following:

> The Software Protection Service reported that the computer could not be activated. No Key Management Service (KMS) could be contacted. Please see the Application Event Log for additional information.

Possible cause

All of the KMS host systems returned an error.

Resolution

In the Application Event Log, identify each event that has Event ID 12288 and is associated with the activation attempt. Troubleshoot the errors from these events.

For more information about troubleshooting DNS-related issues, see Common troubleshooting procedures for KMS and DNS issues.

KMS activation: known issues

12/17/2021 • 4 minutes to read • Edit Online

This article describes common questions and issues that can arisse during Key Management Service (KMS) activations, and provides guidance for addressing the issues.

> **NOTE**
> If you suspect that your issue is related to DNS, see Common troubleshooting procedures for KMS and DNS issues.

Should I back up KMS host information?

Backup is not required for KMS hosts. However, if you use a tool to routinely clean up event logs, the activation history stored in the logs can be lost. If you use the event log to track or document KMS activations, periodically export the Key Management Service event log from the Applications and Services Logs folder of Event Viewer.

If you use System Center Operations Manager, the System Center Data Warehouse database stores event log data for reporting, therefore you do not have to back up the event logs separately.

Is the KMS client computer activated?

On the KMS client computer, open the **System** control panel, and look for the **Windows is activated** message. Alternatively, run Slmgr.vbs and use the the **/dli** command-line option.

The KMS client computer does not activate

Verify that the KMS activation threshold is met. On the KMS host computer, run Slmgr.vbs and use the **/dli** command-line option to determine the host's current count. Until the KMS host has a count of 25, Windows 7 client computers cannot be activated. Windows Server 2008 R2 KMS clients require a KMS count of 5 for activation. For more information about KMS requirements, see the Volume Activation Planning Guide.

On the KMS client computer, look in the Application event log for event ID 12289. Check this event for the following information:

- Is the result code **0**? Anything else is an error.
- Is the KMS host name in the event correct?
- Is the KMS port correct?
- Is the KMS host accessible?
- If the client is running a non-Microsoft firewall, does the outbound port have to be configured?

On the KMS host computer, look in the KMS event log for event ID 12290. Check this event for the following information:

- Did the KMS host log a request from the client computer? Verify that the name of the KMS client computer is listed. Verify that the client and KMS host can communicate. Did the client receive the response?
- If no event is logged from the KMS client, the request did not reach the KMS host or the KMS host was unable to process it. Make sure that routers do not block traffic using TCP port 1688 (if the default port is used) and that stateful traffic to the KMS client is allowed.

What does this error code mean?

Except for KMS events that have event ID 12290, Windows logs all activation events to the Application event log under the event provider name Microsoft-Windows-Security-SPP. Windows logs KMS events to the Key Management Service log in the Applications and Services folder. IT pros can run Slui.exe to display a description of most activation-related error codes. The general syntax for this command is as follows:

```
slui.exe 0x2a ErrorCode
```

For example, if event ID 12293 contains error code 0x8007267C, you can display a description of that error by running the following command:

```
slui.exe 0x2a 0x8007267C
```

For more information about specific error codes and how to address them, see Resolving common activation error codes.

Clients are not adding to the KMS count

To reset the client computer ID (CMID) and other product-activation information, run **sysprep /generalize** or **slmgr /rearm**. Otherwise, each client computer looks identical, and the KMS host does not count them as separate KMS clients.

KMS hosts are unable to create SRV records

Domain Name System (DNS) may restrict Write access or may not support dynamic DNS (DDNS). In this case, give the KMS host Write access to the DNS database, or create the service (SRV) resource record (RR) manually. For more information about KMS and DNS issues, see Common troubleshooting procedures for KMS and DNS issues.

Only the first KMS host is able to create SRV records

If the organization has more than one KMS host, the other hosts might not able to update the SRV RR unless the SRV default permissions are changed. For more information about KMS and DNS issues, see Common troubleshooting procedures for KMS and DNS issues.

I installed a KMS key on the KMS client

KMS keys should be installed only on KMS hosts, not on KMS clients. Run **slmgr.vbs -ipk <SetupKey>**. For tables of keys that you can use to configure the computer as a KMS client, see KMS client setup keys. These keys are publicly known and are edition-specific. Remember to delete any unnecessary SRV RRs from DNS, and then restart the computers.

A KMS host failed

If a KMS host fails, you must install a KMS host key on a new host and then activate the host. Make sure that the new KMS host has an SRV RR in the DNS database. If you install the new KMS host using the same computer name and IP address as the failed KMS host, the new KMS host can use the DNS SRV record of the failed host. If the new host has a different computer name, you can manually remove the DNS SRV RR of the failed host or (if scavenging is enabled in DNS) let DNS automatically remove it. If the network is using DDNS, the new KMS host automatically creates a new SRV RR on the DNS server. The new KMS host then starts collecting client renewal requests and begins activating clients as soon as the KMS activation threshold is met.

If your KMS clients use auto-discovery, they automatically select another KMS host if the original KMS host does not respond to renewal requests. If the clients do not use auto-discovery, you must manually update the KMS

client computers that were assigned to the failed KMS host by running **slmgr.vbs /skms**. To avoid this scenario, configure the KMS clients to use auto-discovery. For more information, see the Volume Activation Deployment Guide.

MAK activation: known issues

12/17/2021 • 2 minutes to read • Edit Online

This article describes common issues that can occur during Multiple Activation Key (MAK) activations, and provides guidance for addressing those issues.

How can I tell whether my computer is activated?

On the computer, open the **System** control panel and look for **Windows is activated**. Alternatively, run Slmgr.vbs and use the **/dli** command-line option.

The computer does not activate over the internet

Make sure that the required ports are open in the firewall. For a list of ports, see the Volume Activation Deployment Guide.

Internet and telephone activation fail

Contact a local Microsoft Activation Center. For the telephone numbers of Microsoft Activation Centers worldwide, go to Microsoft Licensing Activation Centers worldwide telephone numbers. Make sure to provide the Volume License agreement information and proof of purchase when you call.

Slmgr.vbs /ato returns an error code

If Slmgr.vbs returns a hexadecimal error code, determine the corresponding error message by running the following script:

```
slui.exe 0x2a 0x <ErrorCode>
```

For more information about specific error codes and how to address them, see Resolving common activation error codes.

Guidelines for troubleshooting DNS-related activation issues

12/17/2021 • 11 minutes to read • Edit Online

You may have to use some of these methods if one or more of the following conditions are true:

- You use volume-licensed media and a Volume License generic product key to install one of the following operating systems:
  - Windows Server 2019
  - Windows Server 2016
  - Windows Server 2012 R2
  - Windows Server 2012
  - Windows Server 2008 R2
  - Windows Server 2008
  - Windows 10
  - Windows 8.1
  - Windows 8
- The activation wizard cannot connect to a KMS host computer.

When you try to activate a client system, the activation wizard uses DNS to locate a corresponding computer that's running the KMS software. If the wizard queries DNS and does not find the DNS entry for the KMS host computer, the wizard reports an error.

Review the following list to find an approach that fits your circumstances:

- If you cannot install a KMS host or if you cannot use KMS activation, try the Change the product key to an MAK procedure.
- If you have to install and configure a KMS host, use the Configure a KMS host for the clients to activate against procedure.
- If the client cannot locate your existing KMS host, use the following procedures to troubleshoot your routing configurations. These procedures are arranged from the simplest to the most complex.
  - Verify basic IP connectivity to the DNS server
  - Verify the KMS host configuration
  - Determine the type of routing issue
  - Verify the DNS configuration
  - Manually create a KMS SRV record
  - Manually assign a KMS host to a KMS client
  - Configure the KMS host to publish in multiple DNS domains

Change the product key to an MAK

If you cannot install a KMS host or, for some other reason, you cannot use KMS activation, change the product key to an MAK. If you downloaded Windows images from the Microsoft Developer Network (MSDN), or from TechNet, the stock-keeping units (SKUs) that are listed below the media are generally volume licensed-media, and the product key that's provided is an MAK key.

To change the product key to an MAK, follow these steps:

1. Open an elevated Command Prompt window. To do this, press the Windows logo key+X, right-click **Command Prompt**, and then select **Run as administrator**. If you are prompted for an administrator password or for confirmation, type the password or provide confirmation.
2. At the command prompt, run the following command:

```
slmgr -ipk xxxxx-xxxxx-xxxxx-xxxxx-xxxxx
```

> **NOTE**
> The **xxxxx-xxxxx-xxxxx-xxxxx-xxxxx** placeholder represents your MAK product key.

Return to the procedure list.

Configure a KMS host for the clients to activate against

KMS activation requires that a KMS host be configured for the clients to activate against. If there are no KMS hosts configured in your environment, install and activate one by using an appropriate KMS host key. After you configure a computer on the network to host the KMS software, publish the Domain Name System (DNS) settings.

For information about the KMS host configuration process, see Activate using Key Management Service and Install and Configure VAMT.

Return to the procedure list.

Verify basic IP connectivity to the DNS server

Verify basic IP connectivity to the DNS server by using the ping command. To do this, follow these steps on both the KMS client that is experiencing the error and the KMS host computer:

1. Open an elevated Command Prompt window.
2. At the command prompt, run the following command:

```
ping <DNS_Server_IP_address>
```

> **NOTE**
> If the output from this command does not include the phrase "Reply from," there is a network problem or DNS issue that you must resolve before you can use the other procedures in this article. For more information about how to troubleshoot TCP/IP issues if you cannot ping the DNS server, see Advanced troubleshooting for TCP/IP issues.

Return to the procedure list.

Verify the configuration of the KMS host

Check the registry of the KMS host server to determine whether it is registering with DNS. By default, a KMS host server dynamically registers a DNS SRV record one time every 24 hours.

IMPORTANT

Follow the steps in this section carefully. Serious problems might occur if you modify the registry incorrectly. Before you modify it, back up the registry for restoration in case problems occur.

To check this setting, follow these steps:

1. Start Registry Editor. To do this, right-click **Start**, select **Run**, type **regedit**, and then press Enter.
2. Locate the **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SoftwareProtectionPlatform** subkey (previously **SL** instead of **SoftwareProtectionPlatform** in Windows Server 2008 and Windows Vista), and check the value of the **DisableDnsPublishing** entry. This entry has the following possible values:
   - **0** or undefined (default): The KMS host server registers a SRV record once every 24 hours.
   - **1**: The KMS host server does not automatically register SRV records. If your implementation does not support dynamic updates, see Manually create a KMS SRV record.
3. If the **DisableDnsPublishing** entry is missing, create it (the type is DWORD). If dynamic registration is acceptable, leave the value undefined or set it to **0**.

Return to the procedure list.

Determine the type of routing issue

You can use the following commands to determine whether this is a name resolution issue or an SRV record issue.

1. On a KMS client, open an elevated Command Prompt window.
2. At the command prompt, run the following commands:

```
cscript \windows\system32\slmgr.vbs -skms <KMS_FQDN>:<port>
cscript \windows\system32\slmgr.vbs -ato
```

NOTE

In this command, <KMS_FQDN> represents the fully qualified domain name (FQDN) of the KMS host computer and <port> represents the TCP port that KMS uses.

If these commands resolve the problem, this is an SRV record issue. You can you can troubleshoot it by using one of the commands that are documented in the Manually assign a KMS host to a KMS client procedure.

3. If the problem persists, run the following commands:

```
cscript \windows\system32\slmgr.vbs -skms <IP Address>:<port>
cscript \windows\system32\slmgr.vbs -ato
```

NOTE

In this command, <IP Address> represents the IP address of the KMS host computer and <port> represents the TCP port that KMS uses.

If these commands resolve the problem, this is most likely a name resolution issue. For additional

troubleshooting information, see the Verify the DNS configuration procedure.

4. If none of these commands resolves the problem, check the computer's firewall configuration. Any activation communications that occur between KMS clients and the KMS host use the 1688 TCP port. The firewalls on both the KMS client and the KMS host must allow communication over port 1688.

Return to the procedure list.

Verify the DNS configuration

> **NOTE**
> Unless otherwise stated, follow these steps on a KMS client that has experienced the applicable error.

1. Open an elevated Command Prompt window
2. At the command prompt, run the following command:

```
IPCONFIG /all
```

3. From the command results, note the following information:
   - The assigned IP address of the KMS client computer
   - The IP address of the Primary DNS server that the KMS client computer uses
   - The IP address of the default gateway that the KMS client computer uses
   - The DNS suffix search list that the KMS client computer uses
4. Verify that the KMS host SRV records are registered in DNS. To do this, follow these steps:
   a. Open an elevated Command Prompt window.
   b. At the command prompt, run the following command:

```
nslookup -type=all _vlmcs._tcp>kms.txt
```

   c. Open the KMS.txt file that the command generates. This file should contain one or more entries that resemble the following entry:

```
_vlmcs._tcp.contoso.com SRV service location:
priority = 0
weight = 0
port = 1688 svr hostname = kms-server.contoso.com
```

> **NOTE**
> In this entry, contoso.com represents the domain of the KMS host.

   a. Verify the IP address, host name, port, and domain of the KMS host.
   b. If these **_vlmcs** entries exist, and if they contain the expected KMS host names, go to Manually assign a KMS host to a KMS client.

> **NOTE**
>
> If the **nslookup** command finds the KMS host, it does not mean that the DNS client can find the KMS host. If the **nslookup** command finds the KMS host, but you still cannot activate by using the KMS host, check the other DNS settings, such as the primary DNS suffix and the search list of the DNS suffix.

5. Verify that the search list of the primary DNS suffix contains the DNS domain suffix that is associated with the KMS host. If the search list does not include this information, go to the Configure the KMS host to publish in multiple DNS domains procedure.

Return to the procedure list.

Manually create a KMS SRV record

To manually create an SRV record for a KMS host that uses a Microsoft DNS server, follow these steps:

1. On the DNS server, open DNS Manager. To open DNS Manager, select **Start**, select **Administrative Tools**, and then select **DNS**.
2. Select the DNS server on which you have to create the SRV resource record.
3. In the console tree, expand **Forward Lookup Zones**, right-click the domain, and then select **Other New Records**.
4. Scroll down the list, select **Service Location (SRV)**, and then select **Create Record**.
5. Type the following information:
   - Service: **_VLMCS**
   - Protocol: **_TCP**
   - Port number: **1688**
   - Host offering the service: <*FQDN of the KMS host*>
6. When you are finished, select **OK**, and then select **Done**.

To manually create an SRV record for a KMS host that uses a BIND 9.x-compliant DNS server, follow the instructions for that DNS server, and provide the following information for the SRV record:

- Name: **_vlmcs._TCP**
- Type: **SRV**
- Priority: **0**
- Weight: **0**
- Port: **1688**
- Hostname: <*FQDN or A-Name of the KMS host*>

> **NOTE**
>
> KMS does not use the **Priority** or **Weight** values. However, the record must include them.

To configure a BIND 9.x-compatible DNS server to support KMS auto-publishing, configure the DNS server to enable resource record updates from KMS hosts. For example, add the following line to the zone definition in Named.conf or in Named.conf.local:

```
allow-update { any; };
```

Manually assign a KMS host to a KMS client

By default, the KMS clients use the automatic discovery process. According to this process, a KMS client queries DNS for a list of servers that have published `_vlmcs` SRV records within the membership zone of the client. DNS returns the list of KMS hosts in a random order. The client picks a KMS host and tries to establish a session on it. If this attempt works, the client caches the name of the KMS host and tries to use it for the next renewal attempt. If the session setup fails, the client randomly picks another KMS host. We highly recommend that you use the automatic discovery process.

However, you can manually assign a KMS host to a particular KMS client. To do this, follow these steps.

1. On a KMS client, open an elevated Command Prompt window.
2. Depending on your implementation, follow one of these steps:
   - To assign a KMS host by using the FQDN of the host, run the following command:

     ```
     cscript \windows\system32\slmgr.vbs -skms <KMS_FQDN>:<port>
     ```

   - To assign a KMS host by using the version 4 IP address of the host, run the following command:

     ```
     cscript \windows\system32\slmgr.vbs -skms <IPv4Address>:<port>
     ```

   - To assign a KMS host by using the version 6 IP address of the host, run the following command:

     ```
     cscript \windows\system32\slmgr.vbs -skms <IPv6Address>:<port>
     ```

   - To assign a KMS host by using the NETBIOS name of the host, run the following command:

     ```
     cscript \windows\system32\slmgr.vbs -skms <NETBIOSName>:<port>
     ```

   - To revert to automatic discovery on a KMS client, run the following command:

     ```
     cscript \windows\system32\slmgr.vbs -ckms
     ```

     NOTE

     These commands use the following placeholders:

     - **<KMS_FQDN>** represents the fully qualified domain name (FQDN) of the KMS host computer
     - **<IPv4Address>** represents the IP version 4 address of the KMS host computer
     - **<IPv6Address>** represents the IP version 6 address of the KMS host computer
     - **<NETBIOSName>** represents the NETBIOS name of the KMS host computer
     - **<port>** represents the TCP port that KMS uses.

Configure the KMS host to publish in multiple DNS domains

IMPORTANT

Follow the steps in this section carefully. Serious problems might occur if you modify the registry incorrectly. Before you modify it, back up the registry for restoration in case problems occur.

As described in Manually assign a KMS host to a KMS client, KMS clients typically use the automatic discovery process to identify KMS hosts. This process requires that the `_vlmcs` SRV records must be available in the DNS

zone of the KMS client computer. The DNS zone corresponds to either the primary DNS suffix of the computer or to one of the following:

- For domain-joined computers, the computer's domain as assigned by the DNS system (such as Active Directory Domain Services (AD DS) DNS).
- For workgroup computers, the computer's domain as assigned by the Dynamic Host Configuration Protocol (DHCP). This domain name is defined by the option that has the code value of 15 as defined in Request for Comments (RFC) 2132.

By default, a KMS host registers its SRV records in the DNS zone that corresponds to the domain of the KMS host computer. For example, assume that a KMS host joins the contoso.com domain. In this scenario, the KMS host registers its `_vlmcs` SRV record under the contoso.com DNS zone. Therefore, the record identifies the service as `_VLMCS._TCP.CONTOSO.COM`.

If the KMS host and KMS clients use different DNS zones, you must configure the KMS host to automatically publish its SRV records in multiple DNS domains. To do this, follow these steps:

1. On the KMS host, start Registry Editor.
2. Locate and then select the **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SoftwareProtectionPlatform** subkey (previously **SL** instead of **SoftwareProtectionPlatform** in Windows Server 2008 and Windows Vista).
3. In the **Details** pane, right-click a blank area, select **New**, and then select **Multi-String Value**.
4. For the name of the new entry, enter **DnsDomainPublishList**.
5. Right-click the new **DnsDomainPublishList** entry, and then select **Modify**.
6. In the **Edit Multi-String** dialog box, type each DNS domain suffix that KMS publishes on a separate line, and then select **OK**.

   > **NOTE**
   >
   > For Windows Server 2008 R2, the format for **DnsDomainPublishList** differs. For more information, see the Volume Activation Technical Reference Guide.

7. Use the Services administrative tool to restart the Software Protection service (previously the Software Licensing service in Windows Server 2008 and Windows Vista). This operation creates the SRV records.
8. Verify that by using a typical method, the KMS client can contact the KMS host that you configured. Verify that the KMS client correctly identifies the KMS host both by name and by IP address. If either of these verifications fails, investigate this DNS client resolver issue.
9. To clear any previously cached KMS host names on the KMS client, open an elevated Command Prompt window on the KMS client, and then run the following command:

   ```
   cscript C:\Windows\System32\slmgr.vbs -ckms
   ```

Rebuild the Tokens.dat file

12/17/2021 • 2 minutes to read • Edit Online

When you troubleshoot Windows activation issues, you may have to rebuild the Tokens.dat file. This article describes in detail how to do this.

Resolution

To rebuild the Tokens.dat file, follow these steps:

1. Open an elevated Command Prompt window: **For Windows 10**

   a. Open the **Start** menu, and enter **cmd**.

   b. In the search results, right-click **Command Prompt**, and the select **Run as administrator**.

   For Windows 8.1

   a. Swipe in from the right edge of the screen, and then tap **Search**. Or, if you are using a mouse, point to the lower-right corner of the screen, and then select **Search**.

   b. In the search box, enter **cmd**.

   c. Swipe across or right-click the displayed **Command Prompt** icon.

   d. Tap or click **Run as administrator**.

   For Windows 7

   a. Open the **Start** menu, and enter **cmd**.

   b. In the search results, right-click **cmd.exe**, and the select **Run as administrator**.

2. Enter the list of commands that is appropriate for your operating system.

   For Windows 10, Windows Server 2016 and later versions of Windows, enter the following commands in sequence:

   ```
   net stop sppsvc
   cd %Systemdrive%\Windows\System32\spp\store\2.0\
   ren tokens.dat tokens.bar
   net start sppsvc
   cscript.exe %windir%\system32\slmgr.vbs /rilc
   ```

   For Windows 8.1, Windows Server 2012 and Windows Server 2012 R2, enter the following commands in sequence:

   ```
   net stop sppsvc
   cd %Systemdrive%\Windows\System32\spp\store\
   ren tokens.dat tokens.bar
   net start sppsvc
   cscript.exe %windir%\system32\slmgr.vbs /rilc
   ```

   For Windows 7, Windows Server 2008 and Windows Server 2008 R2, enter the following commands in sequence:

```
net stop sppsvc
cd
%Systemdrive%\Windows\ServiceProfiles\NetworkService\AppData\Roaming\Microsoft\SoftwareProtectionPlat
form
ren tokens.dat tokens.bar
net start sppsvc
cscript.exe %windir%\system32\slmgr.vbs /rilc
```

3. Restart the computer.

More information

After you rebuild the Tokens.dat file, you must reinstall your product key by using one of the following methods:

- At the same elevated prompt command, type the following command, and then press Enter:

  ```
  cscript.exe %windir%\system32\slmgr.vbs /ipk <Product key>
  ```

  > **IMPORTANT**
  > Do not use the **/upk** switch to uninstall a product key. To install a product key over an existing product key, use the **/ipk** switch.

- Right-click **My Computer**, select **Properties**, and then select **Change product key**.

For more information about KMS client setup keys, see KMS client setup keys.

Example: Troubleshooting Active Directory Based Activation (ADBA) clients that do not activate

12/17/2021 • 7 minutes to read • Edit Online

NOTE

This article was originally published as a TechNet blog on March 26, 2018.

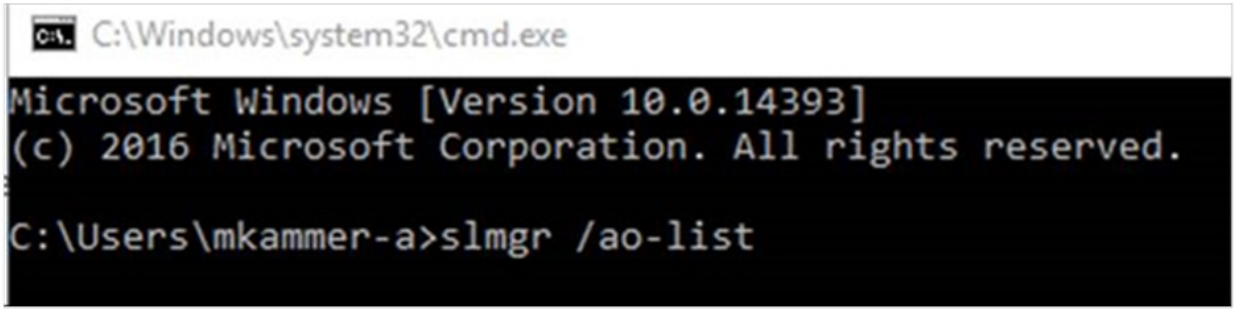
Hello everyone! My name is Mike Kammer, and I have been a Platforms PFE with Microsoft for just over two years now. I recently helped a customer with deploying Windows Server 2016 in their environment. We took this opportunity to also migrate their activation methodology from a KMS Server to Active Directory Based Activation.

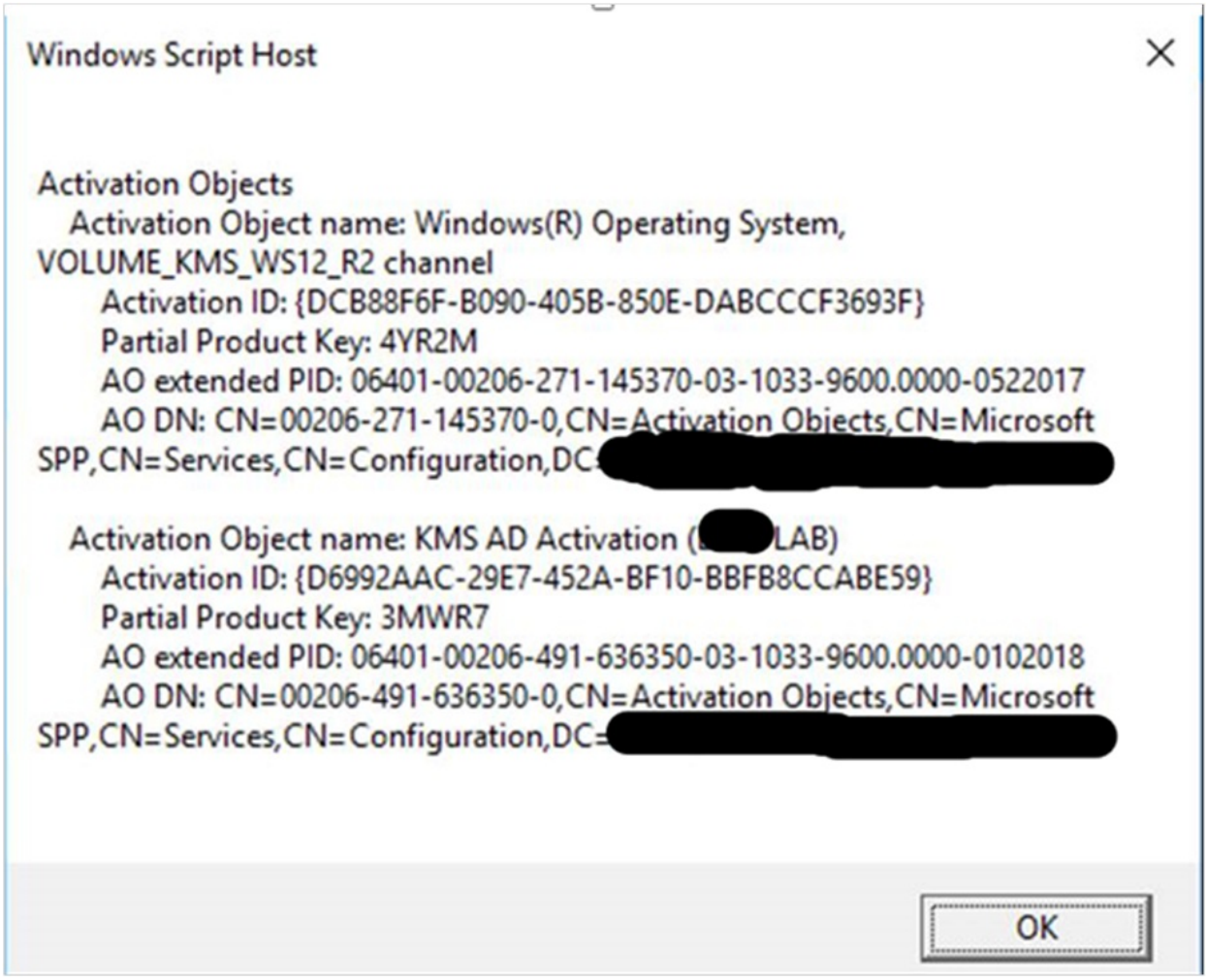
As proper procedure for making all changes, we started our migration in the customer's test environment. We began our deployment by following the instructions in this excellent blog post by Charity Shelbourne, Active Directory-Based Activation vs. Key Management Services. The domain controllers in our test environment were all running Windows Server 2012 R2, so we did not need to prep our forest. We installed the role on a Windows Server 2012 R2 Domain Controller and chose Active Directory Based Activation as our volume activation method. We installed our KMS key and gave it a name of "KMS AD Activation (** LAB)". We pretty much followed the blog post step by step.

We started by building four virtual machines, two Windows 2016 Standard and two Windows 2016 Datacenter. At this point everything was great, and everyone was happy. We built a physical server running Windows 2016 Standard, and the machine activated properly. And that's where our story ends.

Haha! Just kidding! Nothing is ever that easy. Truthfully, the set up and configuration were super easy, so that part was simple and straight forward. I came back into the office on Monday, and all the virtual machines I had built the week prior showed that they weren't activated. Hey! That's not right! I went back to the physical machine and it was fine. I went to the customer to discuss what had happened. Of course, the first question was "What changed over the weekend?" And as usual the answer was "nothing." This time, nothing really had been changed, and we had to figure out what was going on.

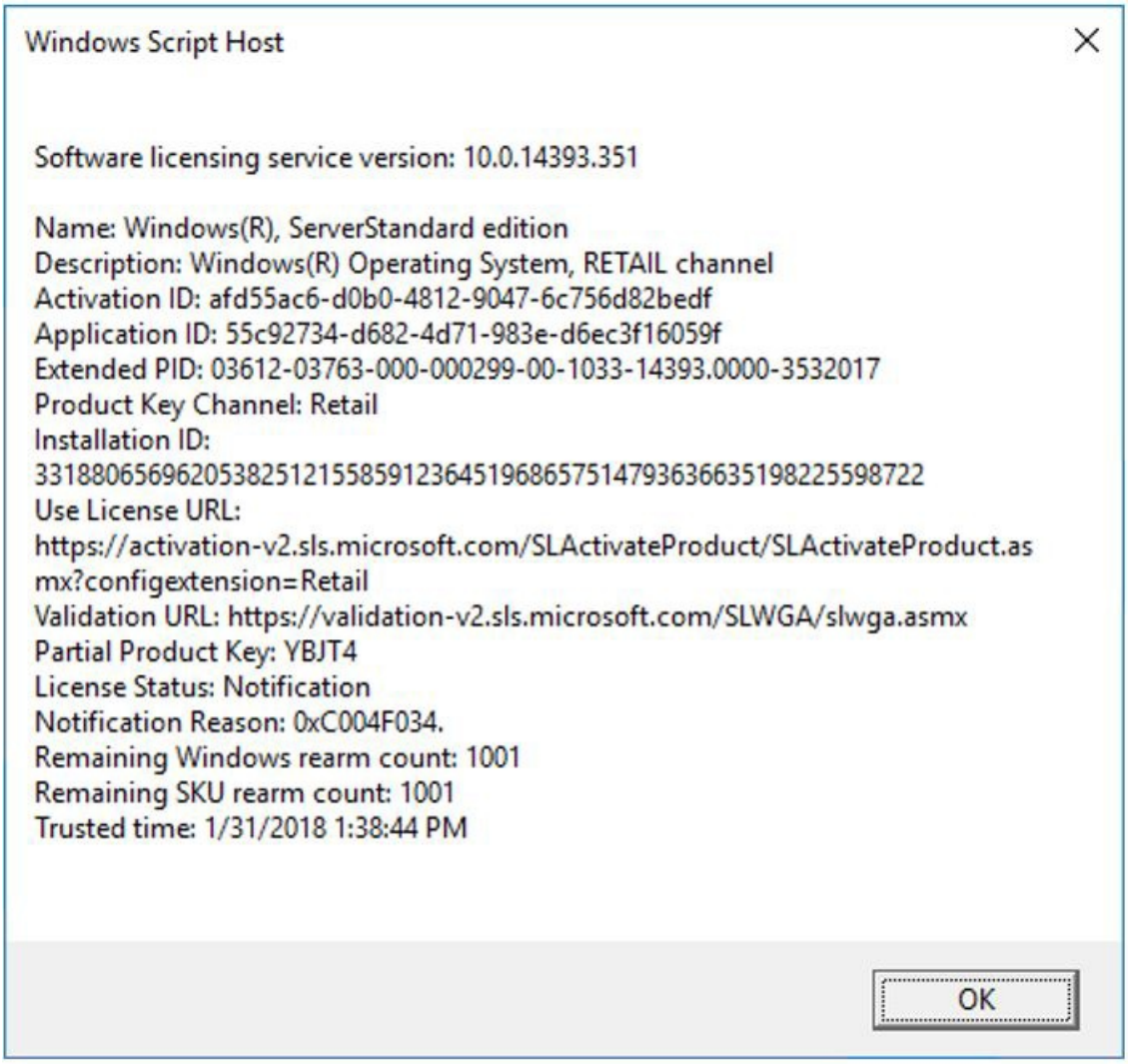
I went to one of my problem servers, opened a command prompt, and checked my output from the **slmgr /ao-list** command. The **/ao-list** switch displays all activation objects in Active Directory.





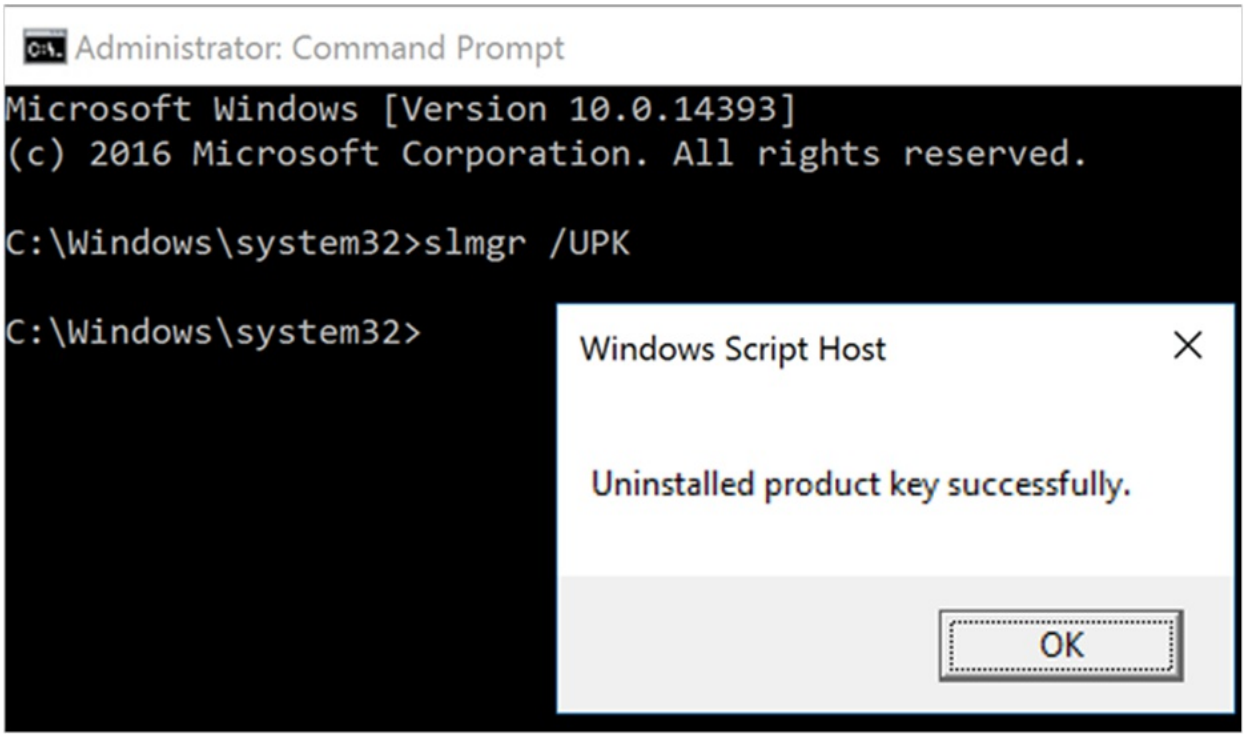
The results show that we have two Activation Objects: one for Server 2012 R2, and our newly created KMS AD Activation (** LAB) which is our Windows Server 2016 license. This confirms our Active Directory is correctly configured to activate Windows KMS Clients

Knowing that the **slmgr** command is my friend for license activation, I continued with different options. I tried the **/dlv** switch, which will display detailed license information. This looked fine to me, I was running the Standard version of Windows Server 2016, there's an Activation ID, an Installation ID, a validation URL, even a partial Product Key.

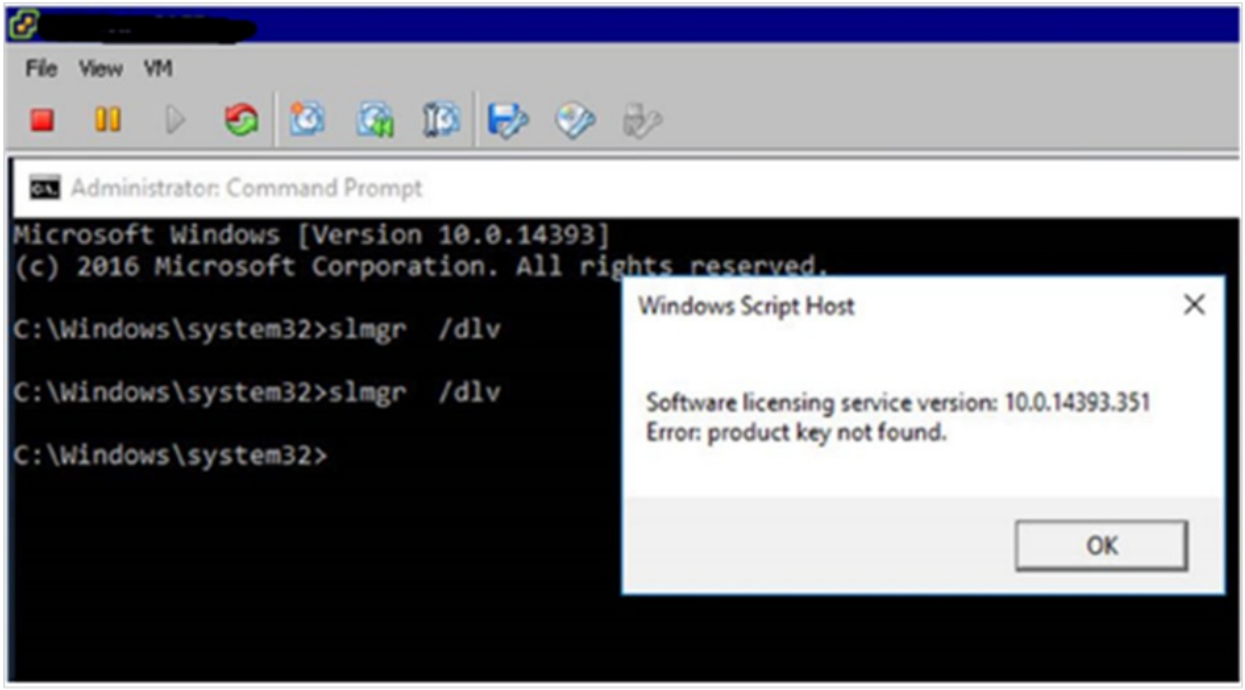


Does anyone see what I missed at this point? We'll come back to it after my other troubleshooting steps but suffice it to say the answer is in this screenshot.

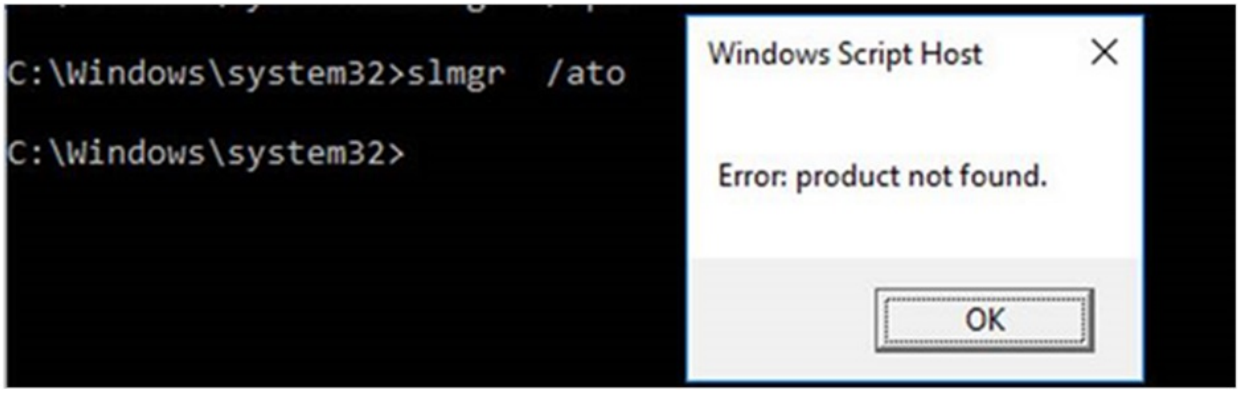
My thinking now is that for some reason the key is broken, so I use the **/upk** switch, which uninstalls the current key. While this was effective in removing the key, it is generally not the best way to do it. Should the server get rebooted before getting a new key it may leave the server in a bad state. I found that using the **/ipk** switch (which I do later in my troubleshooting) overwrites the existing key and is a much safer route to take. Learn from my missteps!



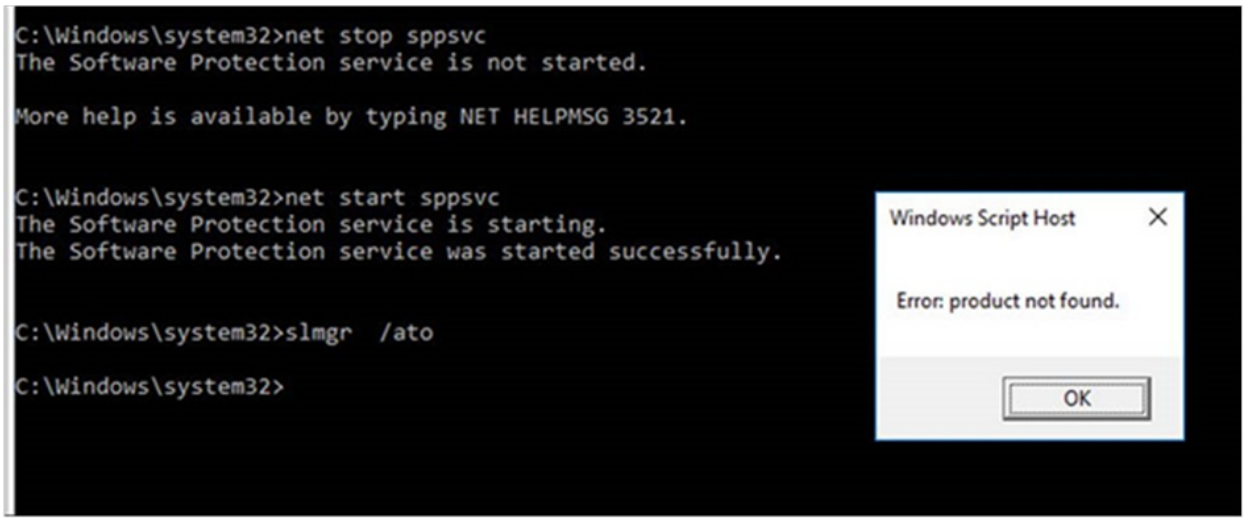
I ran the **/dlv** switch again, to see the detailed license information. Unfortunately for me that didn't give me any helpful information, just a product key not found error. Because, of course, there's no key since I just uninstalled it!



I figured it was a long shot, but I tried the **/ato** switch, which should activate Windows against the known KMS servers (or Active Directory as the case may be). Again, just a product not found error.

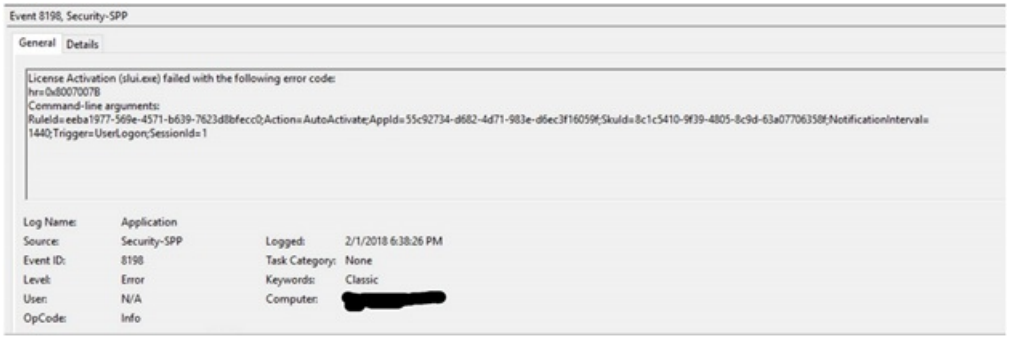


My next thought was that sometimes stopping and starting a service does the trick, so I tried that next. I need to stop and start the Microsoft Software Protection Platform Service (SPPSvc service). From an administrative command prompt, I use the trusty **net stop** and **net start** commands. I notice at first that the service isn't running, so I think this must be it!



But no. After starting the service and attempting to activate Windows again, I still get the product not found error.

I then looked at the Application Event Log on one of the trouble servers. I find an error related to License Activation, Event ID 8198, that has a code of 0x8007007B.



While looking up this code, I found an article that says my error code means that the file name, directory name, or volume label syntax is incorrect. Reading through the methods described in the article, it didn't seem that any of them fit my situation. When I ran the **nslookup -type=all _vlmcs._tcp** command, I found the existing KMS server (still lots of Windows 7 and Server 2008 machines in the environment, so it was necessary to keep it around), but also the five domain controllers as well. This indicated that it was not a DNS problem and my issues were elsewhere.

```
nslookup -type=all _vlmcs._tcp>kms.txt

Server:  labdns1.CONTOSO.COM
Address:  10.10.14.11

_vlmcs._tcp.CONTOSO.COM    SRV service location:
          priority       = 0
          weight         = 0
          port           = 1688
          svr hostname   = labKMS.CONTOSO.COM

_tcp.CONTOSO.COM   nameserver = labDC2.CONTOSO.COM
_tcp.CONTOSO.COM   nameserver = remDC1.CONTOSO.COM
_tcp.CONTOSO.COM   nameserver = labDC4.CONTOSO.COM
_tcp.CONTOSO.COM   nameserver = labDC1.CONTOSO.COM
_tcp.CONTOSO.COM   nameserver = labDC3.CONTOSO.COM
labKMS.CONTOSO.COM  internet address = 10.10.14.100
labDC1.CONTOSO.COM   internet address = 10.10.14.26
remDC1.CONTOSO.COM   internet address = 10.10.20.88
labDC4.CONTOSO.COM   internet address = 10.10.14.27
labDC3.CONTOSO.COM   internet address = 10.10.14.34
labDC2.CONTOSO.COM   internet address = 10.10.14.44
```

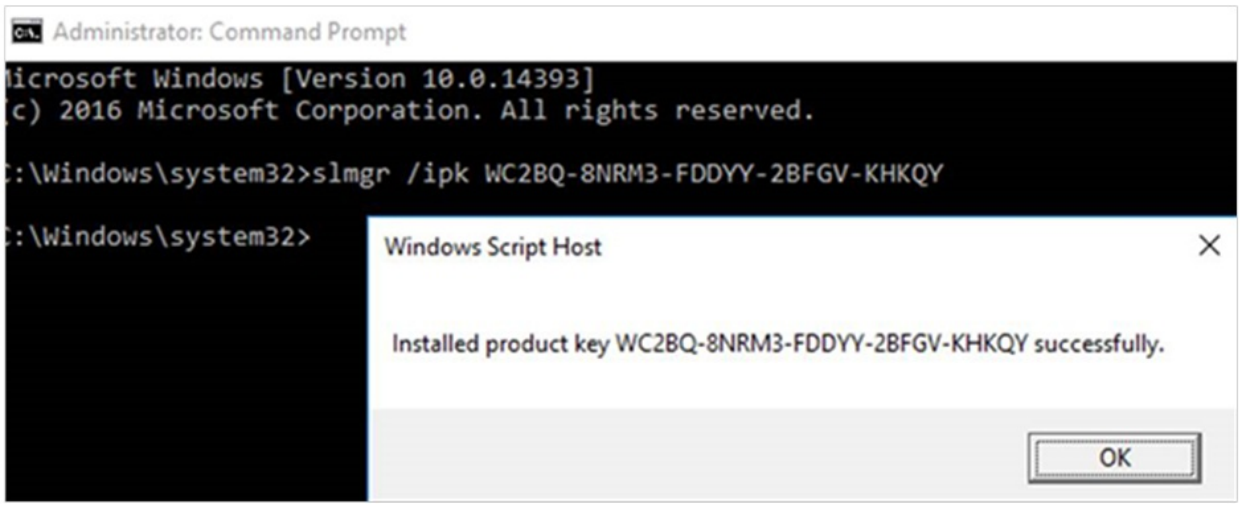
So I know DNS is fine. Active Directory is properly configured as a KMS activation source. My physical server has been activated properly. Could this be an issue with just VMs? As an interesting side note at this point, my customer informs me that someone in a different department has decided to build more than a dozen virtual Windows Server 2016 machines as well. So now I assume I've got another dozen servers to deal with that won't be activating. But no! Those servers activated just fine.

Well, I headed back to my **slmgr** command to figure out how to get these monsters activated. This time I'm going to use the **/ipk** switch, which will allow me to install a product key. I went to this site to get the appropriate keys for my Standard version of Windows Server 2016. Some of my servers are Datacenter, but I need to fix this one first.

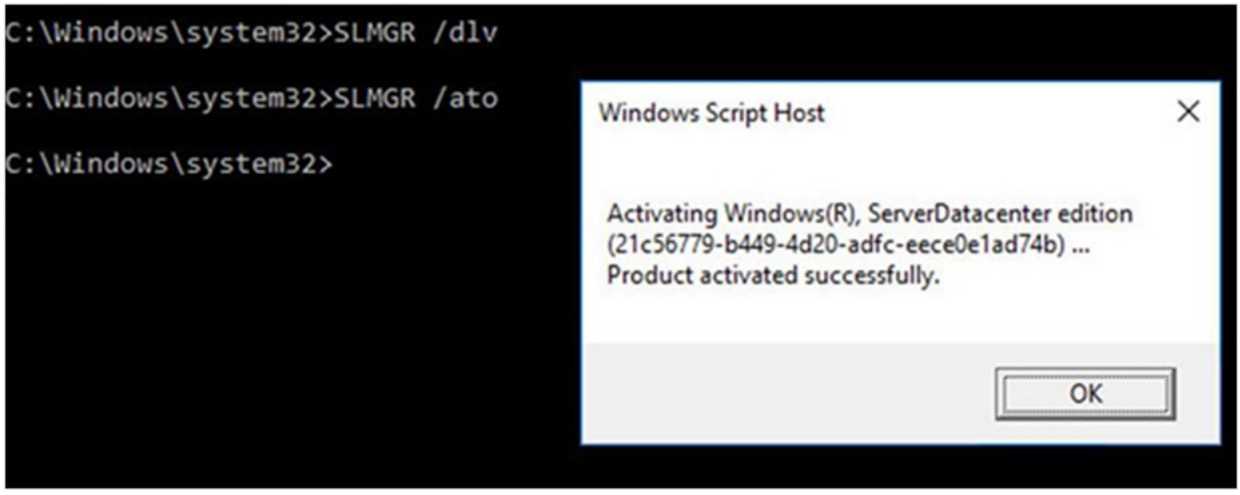
Windows Server 2016

Operating system edition	KMS Client Setup Key
Windows Server 2016 Datacenter	CB7KF-BWN84-R7R2Y-793K2-8XDDG
Windows Server 2016 Standard	WC2BQ-8NRM3-FDDYY-2BFGV-KHKQY
Windows Server 2016 Essentials	JCKRF-N37P4-C2D82-9YXRT-4M63B

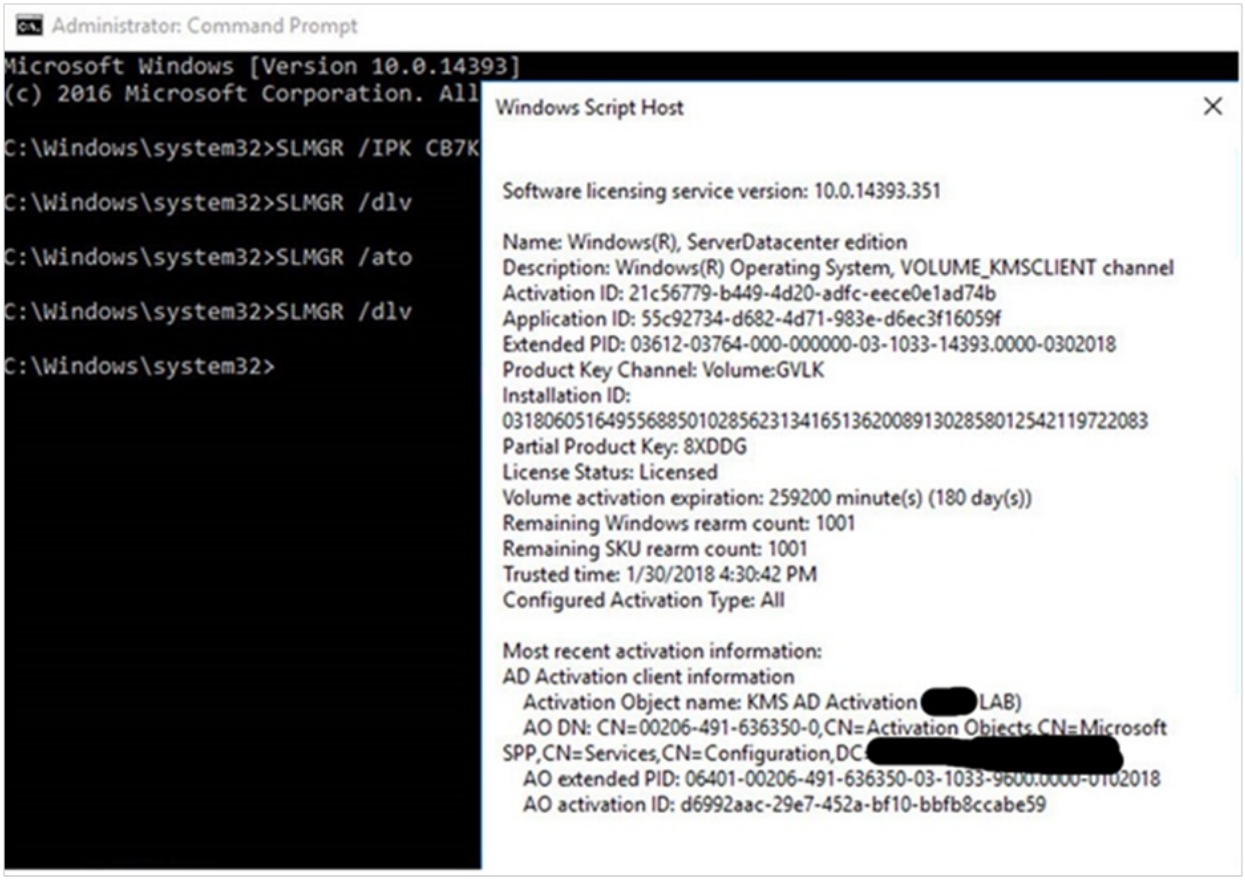
I used the **/ipk** switch to install a product key, choosing the Windows Server 2016 Standard key.



From here on out I only captured results from my Datacenter experiences, but they were the same. I used the **/ato** switch to force the activation. We get the awesome message that the product has been activated successfully!

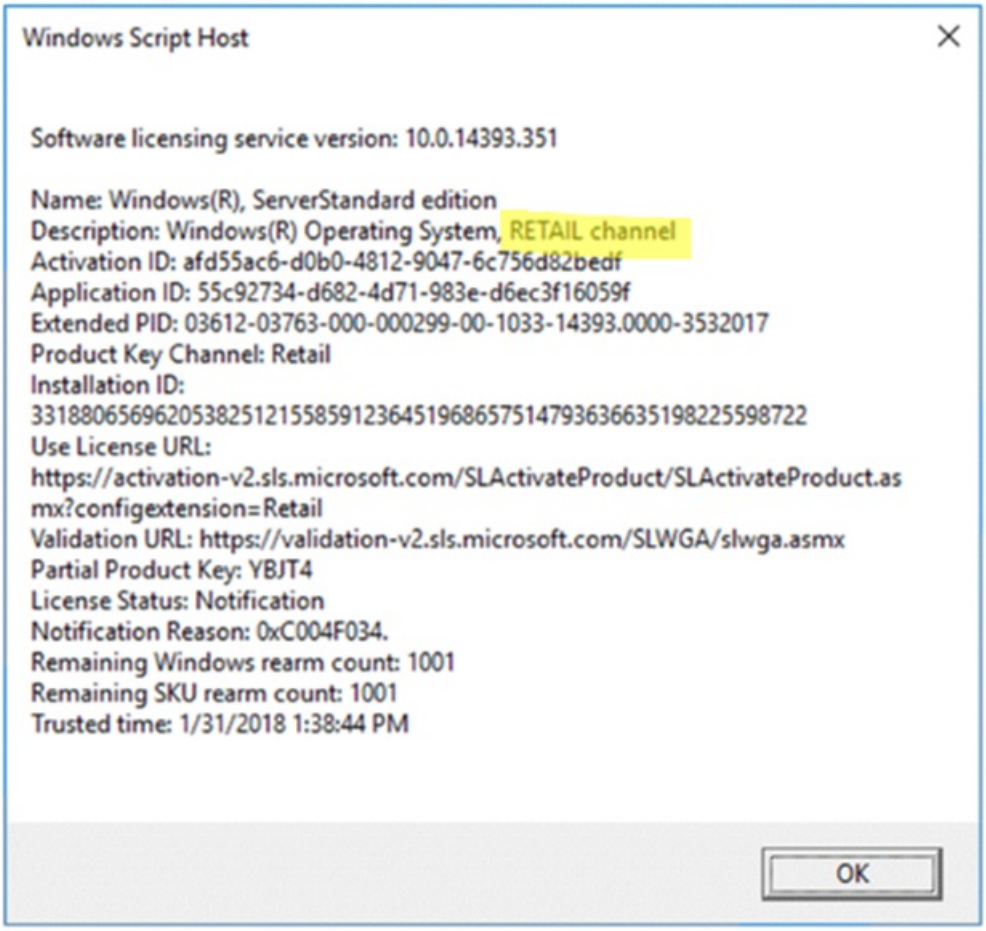


Using the **/dlv** switch again, we can see that now we have been activated by Active Directory.

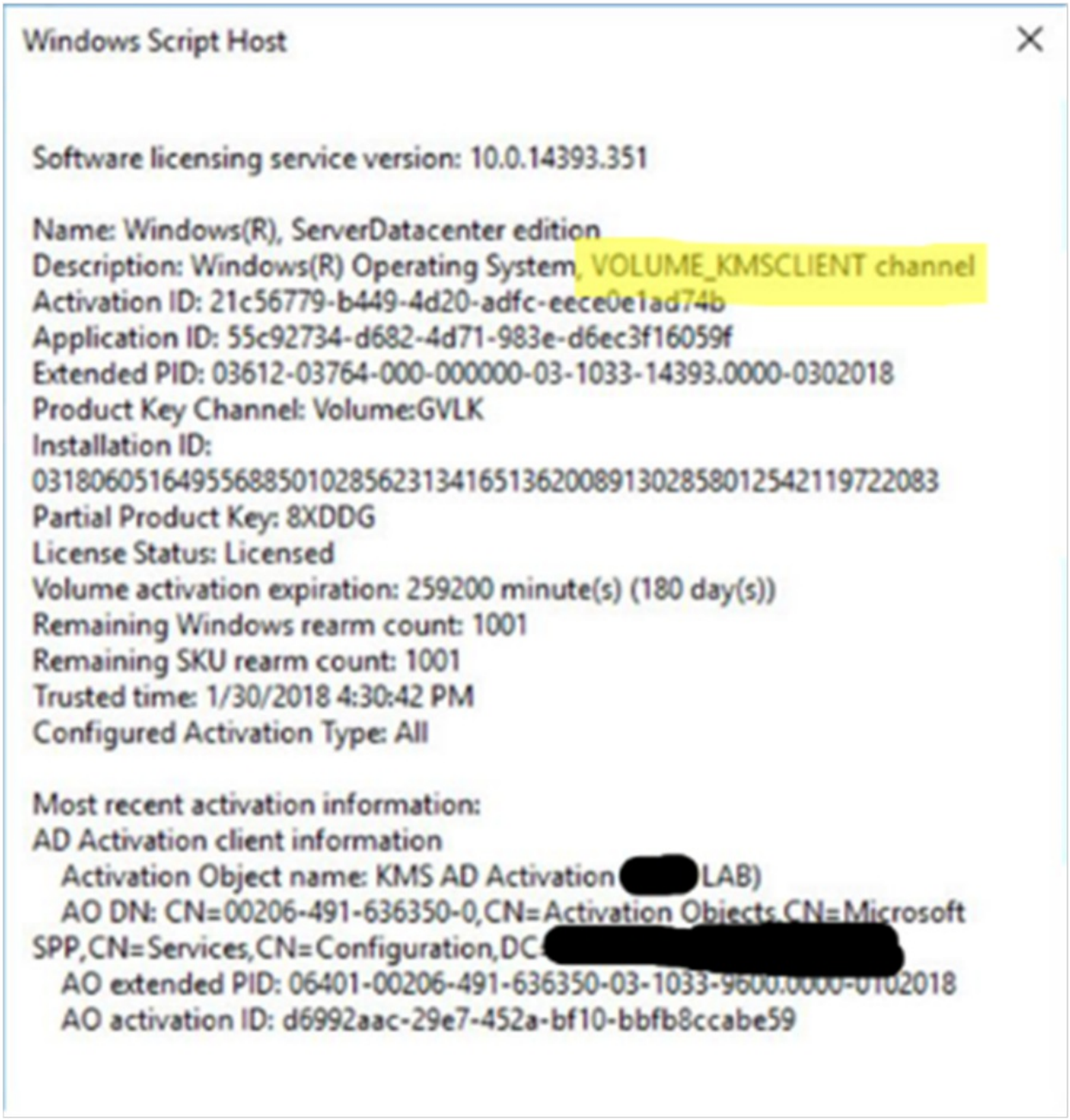


Now, what had gone wrong? Why did I have to remove the installed key and add those generic keys to get these machines to activate properly? Why did the other dozen or so machines activate with no issues? As I said earlier, I missed something key in the initial stages of looking at the issue. I was thoroughly confused, so reached out to Charity from the initial blog post to see if she could help me. She saw the problem right away and helped me understand what I had missed early on.

When I ran the first **/dlv** switch, in the description was the key. The description was Windows® Operating System, RETAIL Channel. I had looked at that and thought that RETAIL Channel meant that it had been purchased and was a valid key.



When we look at the output of the **/dlv** switch from a properly activated server, notice the description now states VOLUME_KMSCLIENT channel. This lets us know that it is indeed a volume license.



So what does that RETAIL channel mean then? Well, it means the media that was used to install the operating system was an MSDN ISO. I went back to my customer and asked if, by some chance, there was a second Windows Server 2016 ISO floating around the network. Turns out that yes, there was another ISO on the network, and it had been used to create the other dozen machines. They compared the two ISOs and sure enough the one that was given to me to build the virtual servers was, in fact, an MSDN ISO. They removed that MSDN ISO from their network and now we have all our existing servers activated and no more worries about the activation failing on future builds.

I hope this has been helpful and may save you some time going forward!

Mike

Troubleshooting Nano Server

12/17/2021 • 3 minutes to read • Edit Online

Applies to: Windows Server 2016

IMPORTANT

Starting in Windows Server, version 1709, Nano Server will be available only as a container base OS image. Check out Changes to Nano Server to learn what this means.

This topic includes information about tools you can use to connect to, diagnose, and repair Nano Server installations.

Using the Nano Server Recovery Console

Nano Server includes a Recovery Console that ensures you can access your Nano Server even if a network mis-configuration interferes with connecting to the Nano Server. You can use the Recovery Console to fix the network and then use your usual remote management tools.

When you boot Nano Server in either a virtual machine or on a physical computer that has a monitor and keyboard attached, you'll see a full-screen, text-mode logon prompt. Log into this prompt with an administrator account to see the computer name and IP address of the Nano Server. You can use these commands to navigate in this console:

- Use arrow keys to scroll
- Use TAB to move to any text that starts with >; then press ENTER to select.
- To go back one screen or page, press ESC. If you're on the home page, pressing ESC will log you off.
- Some screens have additional capabilities displayed on the last line of the screen. For example, if you explore a network adapter, F4 will disable the network adapter.

The Recovery Console allows you to view and configure network adapters and TCP/IP settings, as well as firewall rules.

NOTE

The Recovery Console only supports basic keyboard functions. Keyboard lights, 10-key sections, and keyboard layout switching such as caps lock and number lock are not supported. Only English keyboards and character set are supported.

Accessing Nano Server over a serial port with Emergency Management Services

Emergency Management Services (EMS) lets you perform basic troubleshooting, get network status, and open console sessions (including CMD/PowerShell) by using a terminal emulator over a serial port. This replaces the need for a keyboard and monitor to troubleshoot a server. For more information about EMS, see Emergency Management Services Technical Reference.

To enable EMS on a Nano Server image so that it's ready should you need it later, run this cmdlet:

```
New-NanoServerImage -MediaPath \\Path\To\Media\en_us -BasePath .\Base -TargetPath .\EnablingEMS.vhdx -
EnableEMS -EMSPort 3 -EMSBaudRate 9600
```

This example cmdlet enables EMS on serial port 3 with a baud rate of 9600 bps. If you don't include those parameters, the defaults are port 1 and 115200 bps. To use this cmdlet for VHDX media, be sure to include the Hyper-V feature and the corresponding Windows PowerShell modules.

Kernel debugging

You can configure the Nano Server image to support kernel debugging by a variety of methods. To use kernel debugging with a VHDX image, be sure to include the Hyper-V feature and the corresponding Windows PowerShell modules. For more information about remote kernel debugging generally see Setting Up Kernel-Mode Debugging over a Network Cable Manually and Remote Debugging Using WinDbg.

Debugging using a serial port

Use this example cmdlet to enable the image to be debugged using a serial port:

```
New-NanoServerImage -MediaPath \\Path\To\Media\en_us -BasePath .\Base -TargetPath .\KernelDebuggingSerial -
DebugMethod Serial -DebugCOMPort 1 -DebugBaudRate 9600
```

This example enables serial debugging over port 2 with a baud rate of 9600 bps. If you don't specify these parameters, the defaults are prot 2 and 115200 bps. If you intend to use both EMS and kernel debugging, you'll have to configure them to use two separate serial ports.

Debugging over a TCP/IP network

Use this example cmdlet to enable the image to be debugged over a TCP/IP network:

```
New-NanoServerImage -MediaPath \\Path\To\Media\en_us -BasePath .\Base -TargetPath .\KernelDebuggingNetwork -
DebugMethod Net -DebugRemoteIP 192.168.1.100 -DebugPort 64000
```

This cmdlet enables kernel debugging such that only the computer with the IP address of 192.168.1.100 is allowed to connect, with all communications over port 64000. The -DebugRemoteIP and -DebugPort parameters are mandatory, with a port number greater than 49152. This cmdlet generates an encryption key in a file alongside the resulting VHD which is required for communication over the port. Alternately, you can specify your own key with the -DebugKey parameter, as in this example:

```
New-NanoServerImage -MediaPath \\Path\To\Media\en_us -BasePath .\Base -TargetPath .\KernelDebuggingNetwork -
DebugMethod Net -DebugRemoteIP 192.168.1.100 -DebugPort 64000 -DebugKey 1.2.3.4
```

Debugging using the IEEE1394 protocol (Firewire)

To enable debugging over IEEE1394 use this example cmdlet:

```
New-NanoServerImage -MediaPath \\Path\To\Media\en_us -BasePath .\Base -TargetPath .\KernelDebuggingFireWire -
DebugMethod 1394 -DebugChannel 3
```

The -DebugChannel parameter is mandatory.

Debugging using USB

You can enable debugging over USB with this cmdlet:

```
New-NanoServerImage -MediaPath \\Path\To\Media\en_us -BasePath .\Base -TargetPath .\KernelDebuggingUSB -
DebugMethod USB -DebugTargetName KernelDebuggingUSBNano
```

When you connect the remote debugger to the resulting Nano Server, specify the target name as set by the -DebugTargetName parameter.

Windows Server - License Terms

12/17/2021 • 2 minutes to read • Edit Online

Review our Windows Server-related license terms.

- Additional software for Windows Server 2016
- Windows Server Technical Preview Expiration
- Windows Server 2016 Technical Preview License Terms
- Microsoft Software License Terms - MICROSOFT.WINDOWSSERVER.SYSTEMINSIGHTS
- Microsoft Software License Terms - MICROSOFT.WINDOWSSERVER.SYSTEMINSIGHTS.CAPABILITIES
- Windows Admin Center - License Terms