

Contents

Failover Clustering

What's New in Failover Clustering

Understand

- Scale-Out File Server for application data

- Cluster and pool quorum

- Fault domain awareness

- Simplified SMB Multichannel and multi-NIC cluster networks

- VM load balancing

- VM load balancing deep-dive

Plan

- Hardware requirements

- Use Cluster Shared Volumes (CSVs)

Deploy

- Create a failover cluster

- Prestage a cluster in AD DS

- Deploy a Cloud Witness for a Failover Cluster

- Cluster operating system rolling upgrades

Manage

- Cluster-Aware Updating

 - Requirements and best practices

 - Advanced options

 - FAQ

 - Plug-ins

Health Service

- Reports

- Faults

- Actions

- Settings

- Configure and manage quorum

[Troubleshooting using Windows Error Reporting](#)

[Insider Preview](#)

[Cluster sets](#)

[Change history for Failover Clustering topics](#)

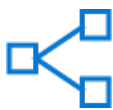
Failover Clustering in Windows Server

5/22/2018 • 2 minutes to read • [Edit Online](#)

Applies To: Windows Server (Semi-Annual Channel), Windows Server 2016

TIP

Looking for information about older versions of Windows Server? Check out our other [Windows Server libraries](#) on docs.microsoft.com. You can also [search this site](#) for specific information.



Failover clustering - a Windows Server feature that enables you to group multiple servers together into a fault-tolerant cluster - provides new and improved features for software-defined datacenter customers and many other workloads running clusters on physical hardware or in virtual machines.

A failover cluster is a group of independent computers that work together to increase the availability and scalability of clustered roles (formerly called clustered applications and services). The clustered servers (called nodes) are connected by physical cables and by software. If one or more of the cluster nodes fail, other nodes begin to provide service (a process known as failover). In addition, the clustered roles are proactively monitored to verify that they are working properly. If they are not working, they are restarted or moved to another node.

Failover clusters also provide Cluster Shared Volume (CSV) functionality that provides a consistent, distributed namespace that clustered roles can use to access shared storage from all nodes. With the Failover Clustering feature, users experience a minimum of disruptions in service.

Failover Clustering has many practical applications, including:

- Highly available or continuously available file share storage for applications such as Microsoft SQL Server and Hyper-V virtual machines
- Highly available clustered roles that run on physical servers or on virtual machines that are installed on servers running Hyper-V

What's new

Here are some of the new features in Windows Server 2016 - for more details, see [What's new in Failover Clustering](#):

Cluster operating system rolling upgrades

Enables an administrator to upgrade the operating system of the cluster nodes from without stopping the Hyper-V or the Scale-Out File Server workloads.

Cloud Witness for a Failover Cluster

A new type of quorum witness that leverages Microsoft Azure to help determine which cluster node should be considered authoritative if a node goes offline.

Health Service

Improves the day-to-day monitoring, operations, and maintenance experience of Storage Spaces Direct clusters.

Fault Domains

Enables you to define what fault domain to use with a Storage Spaces Direct cluster. A fault domain is a set of hardware that share a single point of failure, such as a server node, server chassis, or rack.

VM load balancing

Helps load be evenly distributed across nodes in a Failover Cluster by identifying busy nodes and live-migrating VMs on these nodes to less busy nodes.

Simplified SMB Multichannel and multi-NIC cluster networks

Enables easier configuration of multiple network adapters in a cluster.

Planning

- [Failover Clustering Hardware Requirements and Storage Options](#)
- [Use Cluster Shared Volumes \(CSVs\)](#)

Deployment

- [Prestage Cluster Computer Objects in Active Directory Domain Services](#)
- [Creating a Failover Cluster](#)
- [Deploy Hyper-V over SMB](#)
- [Deploy a Scale-Out File Server](#)
- [iSCSI Target Block Storage, How To](#)
- [Deploy an Active Directory Detached Cluster](#)
- [Using Guest Clustering for High Availability](#)
- [Deploy a Guest Cluster using a Shared Virtual Hard Disk](#)

Operations

- [Configure and Manage the Quorum in a Failover Cluster](#)
- [Use Cluster Shared Volumes in a Failover Cluster](#)
- [Cluster-Aware Updating Overview](#)
- [Windows IT Pro Support](#)

Tools and settings

- [Failover Clustering PowerShell Cmdlets](#)
- [Cluster Aware Updating PowerShell Cmdlets](#)

Community resources

- [High Availability \(Clustering\) Forum](#)
- [Failover Clustering and Network Load Balancing Team Blog](#)

What's new in Failover Clustering in Windows Server 2016

10/17/2017 • 8 minutes to read • [Edit Online](#)

Applies to: Windows Server (Semi-Annual Channel), Windows Server 2016

This topic explains the new and changed functionality in Failover Clustering in Windows Server 2016.

Cluster Operating System Rolling Upgrade

A new feature in Failover Clustering, 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.

What value does this change add?

Upgrading a Hyper-V or Scale-Out File Server cluster from Windows Server 2012 R2 to Windows Server 2016 no longer requires downtime. The cluster will continue to function at a Windows Server 2012 R2 level until all of the nodes in the cluster are running Windows Server 2016. The cluster functional level is upgraded to Windows Server 2016 by using the Windows PowerShell cmdlet `Update-ClusterFunctionalLevel`.

WARNING

- After you update the cluster functional level, you cannot go back to a Windows Server 2012 R2 cluster functional level.
- Until the `Update-ClusterFunctionalLevel` cmdlet is run, the process is reversible, and Windows Server 2012 R2 nodes can be added and Windows Server 2016 nodes can be removed.

What works differently?

A Hyper-V or Scale-Out File Server failover cluster can now easily be upgraded without any downtime or need to build a new cluster with nodes that are running the Windows Server 2016 operating system. Migrating clusters to Windows Server 2012 R2 involved taking the existing cluster offline and reinstalling the new operating system for each node, and then bringing the cluster back online. The old process was cumbersome and required downtime. However, in Windows Server 2016, the cluster does not need to go offline at any point.

The cluster operating systems for the upgrade in phases are as follows for each node in a cluster:

- The node is paused and drained of all virtual machines that are running on it.
- The virtual machines (or other cluster workload) are migrated to another node in the cluster. The virtual machines are migrated to another node in the cluster.
- The existing operating system is removed and a clean installation of the Windows Server 2016 operating system on the node is performed.
- The node running the Windows Server 2016 operating system is added back to the cluster.
- At this point, the cluster is said to be running in mixed mode, because the cluster nodes are running either Windows Server 2012 R2 or Windows Server 2016.
- The cluster functional level stays at Windows Server 2012 R2. At this functional level, new features in Windows Server 2016 that affect compatibility with previous versions of the operating system will be unavailable.
- Eventually, all nodes are upgraded to Windows Server 2016.

- Cluster functional level is then changed to Windows Server 2016 using the Windows PowerShell cmdlet `Update-ClusterFunctionalLevel`. At this point, you can take advantage of the Windows Server 2016 features.

For more information, see [Cluster Operating System Rolling Upgrade](#).

Storage Replica

Storage Replica is a new feature that 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.

What value does this change add?

Storage Replica enables you to do the following:

- Provide a single vendor disaster recovery solution for planned and unplanned outages of mission critical workloads.
- Use SMB3 transport with proven reliability, scalability, and performance.
- Stretch Windows failover clusters to metropolitan distances.
- Use Microsoft software end to end for storage and clustering, such as Hyper-V, Storage Replica, Storage Spaces, Cluster, Scale-Out File Server, SMB3, Data Deduplication, and ReFS/NTFS.
- Help reduce cost and complexity as follows:
 - Is hardware agnostic, with no requirement for a specific storage configuration like DAS or SAN.
 - Allows commodity storage and networking technologies.
 - Features ease of graphical management for individual nodes and clusters through Failover Cluster Manager.
 - Includes comprehensive, large-scale scripting options through Windows PowerShell.
- Help reduce downtime, and increase reliability and productivity intrinsic to Windows.
- Provide supportability, performance metrics, and diagnostic capabilities.

For more information, see the [Storage Replica in Windows Server 2016](#).

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.

What value does this change add?

Using Cloud Witness as a Failover Cluster quorum witness provides the following advantages:

- Leverages Microsoft Azure and eliminates the need for a third separate datacenter.
- Uses the standard publicly available Microsoft Azure Blob Storage which eliminates the extra maintenance overhead of VMs hosted in a public cloud.
- Same Microsoft Azure Storage Account can be used for multiple clusters (one blob file per cluster; cluster

unique id used as blob file name).

- Provides a very low on-going cost to the Storage Account (very small data written per blob file, blob file updated only once when cluster nodes' state changes).

For more information, see [Deploy a Cloud Witness For a Failover Cluster](#).

What works differently?

This capability is new in Windows Server 2016.

Virtual Machine Resiliency

Compute Resiliency Windows Server 2016 includes increased virtual machines compute resiliency to help reduce intra-cluster communication issues in your compute cluster as follows:

- **Resiliency options available for virtual machines:** You can now configure virtual machine resiliency options that define behavior of the virtual machines during transient failures:
 - **Resiliency Level:** Helps you define how the transient failures are handled.
 - **Resiliency Period:** Helps you define how long all the virtual machines are allowed to run isolated.
- **Quarantine of unhealthy nodes:** Unhealthy nodes are quarantined and are no longer allowed to join the cluster. This prevents flapping nodes from negatively affecting other nodes and the overall cluster.

For more information virtual machine compute resiliency workflow and node quarantine settings that control how your node is placed in isolation or quarantine, see [Virtual Machine Compute Resiliency in Windows Server 2016](#).

Storage Resiliency In Windows Server 2016, virtual machines are more resilient to transient storage failures. The improved virtual machine resiliency helps preserve tenant virtual machine session states in the event of a storage disruption. This is achieved by intelligent and quick virtual machine response to storage infrastructure issues.

When a virtual machine disconnects from its underlying storage, it pauses and waits for storage to recover. While paused, the virtual machine retains the context of applications that are running in it. When the virtual machine's connection to its storage is restored, the virtual machine returns to its running state. As a result, the tenant machine's session state is retained on recovery.

In Windows Server 2016, virtual machine storage resiliency is aware and optimized for guest clusters too.

Diagnostic Improvements in Failover Clustering

To help diagnose issues with failover clusters, Windows Server 2016 includes the following:

- Several enhancements to cluster log files (such as Time Zone Information and DiagnosticVerbose log) that makes it easier to troubleshoot failover clustering issues. For more information, see [Windows Server 2016 Failover Cluster Troubleshooting Enhancements - Cluster Log](#).
- A new dump type of **Active memory dump**, which filters out most memory pages allocated to virtual machines, and therefore makes the memory.dmp much smaller and easier to save or copy. For more information, see [Windows Server 2016 Failover Cluster Troubleshooting Enhancements - Active Dump](#).

Site-aware Failover Clusters

Windows Server 2016 includes site-aware failover clusters that enable group nodes in stretched clusters based on their physical location (site). Cluster site-awareness enhances key operations during the cluster lifecycle such as failover behavior, placement policies, heartbeat between the nodes, and quorum behavior. For more information, see [Site-aware Failover Clusters in Windows Server 2016](#).

Workgroup and Multi-domain clusters

In Windows Server 2012 R2 and previous versions, a cluster can only be created between member nodes joined to the same domain. Windows Server 2016 breaks down these barriers and introduces the ability to create a Failover Cluster without Active Directory dependencies. You can now create failover clusters in the following configurations:

- **Single-domain Clusters.** Clusters with all nodes joined to the same domain.
- **Multi-domain Clusters.** Clusters with nodes which are members of different domains.
- **Workgroup Clusters.** Clusters with nodes which are member servers / workgroup (not domain joined).

For more information, see [Workgroup and Multi-domain clusters in Windows Server 2016](#)

Virtual Machine Load Balancing

Virtual machine Load Balancing is a new feature in Failover Clustering that facilitates the seamless load balancing of virtual machines across the nodes in a cluster. Over-committed nodes are identified based on virtual machine Memory and CPU utilization on the node. Virtual machines are then moved (live migrated) from an over-committed node to nodes with available bandwidth (if applicable). The aggressiveness of the balancing can be tuned to ensure optimal cluster performance and utilization. Load Balancing is enabled by default in Windows Server 2016 Technical Preview. However, Load Balancing is disabled when SCVMM Dynamic Optimization is enabled.

Virtual Machine Start Order

Virtual machine Start Order is a new feature in Failover Clustering that introduces start order orchestration for Virtual machines (and all groups) in a cluster. Virtual machines can now be grouped into tiers, and start order dependencies can be created between different tiers. This ensures that the most important virtual machines (such as Domain Controllers or Utility virtual machines) are started first. Virtual machines are not started until the virtual machines that they have a dependency on are also started.

Simplified SMB Multichannel and Multi-NIC Cluster Networks

Failover Cluster networks are no longer limited to a single NIC per subnet / network. With Simplified SMB Multichannel and Multi-NIC Cluster Networks, network configuration is automatic and every NIC on the subnet can be used for cluster and workload traffic. This enhancement allows customers to maximize network throughput for Hyper-V, SQL Server Failover Cluster Instance, and other SMB workloads.

For more information, see [Simplified SMB Multichannel and Multi-NIC Cluster Networks](#).

See Also

- [Storage](#)
- [What's New in Storage in Windows Server 2016](#)

Scale-Out File Server for application data overview

6/20/2018 • 9 minutes to read • [Edit Online](#)

Applies to: Windows Server 2012 R2, Windows Server 2012, Windows Server 2016

Scale-Out File Server is a feature that is designed to provide scale-out file shares that are continuously available for file-based server application storage. Scale-out file shares provides the ability to share the same folder from multiple nodes of the same cluster. This scenario focuses on how to plan for and deploy Scale-Out File Server.

You can deploy and configure a clustered file server by using either of the following methods:

- **Scale-Out File Server for application data** This clustered file server feature was introduced in Windows Server 2012, and it lets you store server application data, such as Hyper-V virtual machine files, on file shares, and obtain a similar level of reliability, availability, manageability, and high performance that you would expect from a storage area network. All file shares are simultaneously online on all nodes. File shares associated with this type of clustered file server are called scale-out file shares. This is sometimes referred to as active-active. This is the recommended file server type when deploying either Hyper-V over Server Message Block (SMB) or Microsoft SQL Server over SMB.
- **File Server for general use** This is the continuation of the clustered file server that has been supported in Windows Server since the introduction of Failover Clustering. This type of clustered file server, and therefore all the shares associated with the clustered file server, is online on one node at a time. This is sometimes referred to as active-passive or dual-active. File shares associated with this type of clustered file server are called clustered file shares. This is the recommended file server type when deploying information worker scenarios.

Scenario description

With scale-out file shares, you can share the same folder from multiple nodes of a cluster. For instance, if you have a four-node file server cluster that is using Server Message Block (SMB) Scale-Out, a computer running Windows Server 2012 R2 or Windows Server 2012 can access file shares from any of the four nodes. This is achieved by leveraging new Windows Server Failover Clustering features and the capabilities of the Windows file server protocol, SMB 3.0. File server administrators can provide scale-out file shares and continuously available file services to server applications and respond to increased demands quickly by simply bringing more servers online. All of this can be done in a production environment, and it is completely transparent to the server application.

Key benefits provided by Scale-Out File Server include:

- **Active-Active file shares.** All cluster nodes can accept and serve SMB client requests. By making the file share content accessible through all cluster nodes simultaneously, SMB 3.0 clusters and clients cooperate to provide transparent failover to alternative cluster nodes during planned maintenance and unplanned failures with service interruption.
- **Increased bandwidth.** The maximum share bandwidth is the total bandwidth of all file server cluster nodes. Unlike previous versions of Windows Server, the total bandwidth is no longer constrained to the bandwidth of a single cluster node; but rather, the capability of the backing storage system defines the constraints. You can increase the total bandwidth by adding nodes.
- **CHKDSK with zero downtime.** CHKDSK in Windows Server 2012 is significantly enhanced to dramatically shorten the time a file system is offline for repair. Clustered shared volumes (CSVs) take this one step further by eliminating the offline phase. A CSV File System (CSVFS) can use CHKDSK without impacting applications with open handles on the file system.
- **Clustered Shared Volume cache.** CSVs in Windows Server 2012 introduces support for a Read cache, which can significantly improve performance in certain scenarios, such as in Virtual Desktop Infrastructure (VDI).

- **Simpler management.** With Scale-Out File Server, you create the scale-out file servers, and then add the necessary CSVs and file shares. It is no longer necessary to create multiple clustered file servers, each with separate cluster disks, and then develop placement policies to ensure activity on each cluster node.
- **Automatic rebalancing of Scale-Out File Server clients.** In Windows Server 2012 R2, automatic rebalancing improves scalability and manageability for scale-out file servers. SMB client connections are tracked per file share (instead of per server), and clients are then redirected to the cluster node with the best access to the volume used by the file share. This improves efficiency by reducing redirection traffic between file server nodes. Clients are redirected following an initial connection and when cluster storage is reconfigured.

In this scenario

The following topics are available to help you deploy a Scale-Out File Server:

- [Plan for Scale-Out File Server](#)
 - [Step 1: Plan for Storage in Scale-Out File Server](#)
 - [Step 2: Plan for Networking in Scale-Out File Server](#)
- [Deploy Scale-Out File Server](#)
 - [Step 1: Install Prerequisites for Scale-Out File Server](#)
 - [Step 2: Configure Scale-Out File Server](#)
 - [Step 3: Configure Hyper-V to Use Scale-Out File Server](#)
 - [Step 4: Configure Microsoft SQL Server to Use Scale-Out File Server](#)

When to use Scale-Out File Server

You should not use Scale-Out File Server if your workload generates a high number of metadata operations, such as opening files, closing files, creating new files, or renaming existing files. A typical information worker would generate a lot of metadata operations. You should use a Scale-Out File Server if you are interested in the scalability and simplicity that it offers and if you only require technologies that are supported with Scale-Out File Server.

The following table lists the capabilities in SMB 3.0, the common Windows file systems, file server data management technologies, and common workloads. You can see whether the technology is supported with Scale-Out File Server, or if it requires a traditional clustered file server (also known as a file server for general use).

TECHNOLOGY AREA	FEATURE	GENERAL USE FILE SERVER CLUSTER	SCALE-OUT FILE SERVER
SMB	SMB Continuous Availability	Yes	Yes
SMB	SMB Multichannel	Yes	Yes
SMB	SMB Direct	Yes	Yes
SMB	SMB Encryption	Yes	Yes
SMB	SMB Transparent failover	Yes (if continuous availability is enabled)	Yes
File System	NTFS	Yes	NA
File System	Resilient File System (ReFS)	Recommended with Storage Spaces Direct	Recommended with Storage Spaces Direct

TECHNOLOGY AREA	FEATURE	GENERAL USE FILE SERVER CLUSTER	SCALE-OUT FILE SERVER
File System	Cluster Shared Volume File System (CSV)	NA	Yes
File Management	BranchCache	Yes	No
File Management	Data Deduplication (Windows Server 2012)	Yes	No
File Management	Data Deduplication (Windows Server 2012 R2)	Yes	Yes (VDI only)
File Management	DFS Namespace (DFS) root server root	Yes	No
File Management	DFS Namespace (DFS) folder target server	Yes	Yes
File Management	DFS Replication (DFSR)	Yes	No
File Management	File Server Resource Manager (Screens and Quotas)	Yes	No
File Management	File Classification Infrastructure	Yes	No
File Management	Dynamic Access Control (claim-based access, CAP)	Yes	No
File Management	Folder Redirection	Yes	Not recommended
File Management	Offline Files (client side caching)	Yes	Not recommended
File Management	Roaming User Profiles	Yes	Not recommended
File Management	Home Directories	Yes	Not recommended
File Management	Work Folders	Yes	No
NFS	NFS Server	Yes	No
Applications	Hyper-V	Not recommended	Yes
Applications	Microsoft SQL Server	Not recommended	Yes

* Folder Redirection, Offline Files, Roaming User Profiles, or Home Directories generate a large number of writes that must be immediately written to disk (without buffering) when using continuously available file shares, reducing performance as compared to general purpose file shares. Continuously available file shares are also incompatible with File Server Resource Manager and PCs running Windows XP. Additionally, Offline Files might not transition to offline mode for 3-6 minutes after a user loses access to a share, which could frustrate users who aren't yet using the Always Offline mode of Offline Files.

Practical applications

Scale-Out File Servers are ideal for server application storage. Some examples of server applications that can store their data on a scale-out file share are listed below:

- The Internet Information Services (IIS) Web server can store configuration and data for Web sites on a scale-out file share. For more information, see [Shared Configuration](#).
- Hyper-V can store configuration and live virtual disks on a scale-out file share. For more information, see [Deploy Hyper-V over SMB](#).
- SQL Server can store live database files on a scale-out file share. For more information, see [Install SQL Server with SMB file share as a storage option](#).
- Virtual Machine Manager (VMM) can store a library share (which contains virtual machine templates and related files) on a scale-out file share. However, the library server itself can't be a Scale-Out File Server—it must be on a stand-alone server or a failover cluster that doesn't use the Scale-Out File Server cluster role.

If you use a scale-out file share as a library share, you can use only technologies that are compatible with Scale-Out File Server. For example, you can't use DFS Replication to replicate a library share hosted on a scale-out file share. It's also important that the scale-out file server have the latest software updates installed.

To use a scale-out file share as a library share, first add a library server (likely a virtual machine) with a local share or no shares at all. Then when you add a library share, choose a file share that's hosted on a scale-out file server. This share should be VMM-managed and created exclusively for use by the library server. Also make sure to install the latest updates on the scale-out file server. For more information about adding VMM library servers and library shares, see [Add profiles to the VMM library](#). For a list of currently available hotfixes for File and Storage Services, see [Microsoft Knowledge Base article 2899011](#).

NOTE

Some users, such as information workers, have workloads that have a greater impact on performance. For example, operations like opening and closing files, creating new files, and renaming existing files, when performed by multiple users, have an impact on performance. If a file share is enabled with continuous availability, it provides data integrity, but it also affects the overall performance. Continuous availability requires that data writes through to the disk to ensure integrity in the event of a failure of a cluster node in a Scale-Out File Server. Therefore, a user that copies several large files to a file server can expect significantly slower performance on continuously available file share.

Features included in this scenario

The following table lists the features that are part of this scenario and describes how they support it.

FEATURE	HOW IT SUPPORTS THIS SCENARIO
Failover Clustering	Failover clusters added the following features in Windows Server 2012 to support scale-Out file server: Distributed Network Name, the Scale-Out File Server resource type, Cluster Shared Volumes (CSV) 2, and the Scale-Out File Server High Availability role. For more information about these features, see What's New in Failover Clustering in Windows Server 2012 [redirected] .

FEATURE	HOW IT SUPPORTS THIS SCENARIO
Server Message Block	<p>SMB 3.0 added the following features in Windows Server 2012 to support scale-Out File Server: SMB Transparent Failover, SMB Multichannel, and SMB Direct.</p> <p>For more information on new and changed functionality for SMB in Windows Server 2012 R2, see What's New in SMB in Windows Server.</p>

More information

- [Software-Defined Storage Design Considerations Guide](#)
- [Increasing Server, Storage, and Network Availability](#)
- [Deploy Hyper-V over SMB](#)
- [Deploying Fast and Efficient File Servers for Server Applications](#)
- [To scale out or not to scale out, that is the question \(blog post\)](#)
- [Folder Redirection, Offline Files, and Roaming User Profiles](#)

Understanding cluster and pool quorum

5/14/2018 • 11 minutes to read • [Edit Online](#)

Applies To: Windows Server 2016

[Windows Server Failover Clustering](#) provides high availability for workloads. These resources are considered highly available if the nodes that host resources are up; however, the cluster generally requires more than half the nodes to be running, which is known as having *quorum*.

Quorum is designed to prevent *split-brain* scenarios which can happen when there is a partition in the network and subsets of nodes cannot communicate with each other. This can cause both subsets of nodes to try to own the workload and write to the same disk which can lead to numerous problems. However, this is prevented with Failover Clustering's concept of quorum which forces only one of these groups of nodes to continue running, so only one of these groups will stay online.

Quorum determines the number of failures that the cluster can sustain while still remaining online. Quorum is designed to handle the scenario when there is a problem with communication between subsets of cluster nodes, so that multiple servers don't try to simultaneously host a resource group and write to the same disk at the same time. By having this concept of quorum, the cluster will force the cluster service to stop in one of the subsets of nodes to ensure that there is only one true owner of a particular resource group. Once nodes which have been stopped can once again communicate with the main group of nodes, they will automatically rejoin the cluster and start their cluster service.

In Windows Server 2016, there are two components of the system that have their own quorum mechanisms:

- **Cluster Quorum:** This operates at the cluster level (i.e. you can lose nodes and have the cluster stay up)
- **Pool Quorum:** This operates on the pool level when Storage Spaces Direct is enabled (i.e. you can lose nodes and drives and have the pool stay up). Storage pools were designed to be used in both clustered and non-clustered scenarios, which is why they have a different quorum mechanism.

Cluster quorum overview

The table below gives an overview of the Cluster Quorum outcomes per scenario:

SERVER NODES	CAN SURVIVE ONE SERVER NODE FAILURE	CAN SURVIVE ONE SERVER NODE FAILURE, THEN ANOTHER	CAN SURVIVE TWO SIMULTANEOUS SERVER NODE FAILURES
2	50/50	No	No
2 + Witness	Yes	No	No
3	Yes	50/50	No
3 + Witness	Yes	Yes	No
4	Yes	Yes	50/50
4 + Witness	Yes	Yes	Yes
5 and above	Yes	Yes	Yes

Therefore, our guidance is:

- If you have two nodes, a witness is **required**.
- If you have three or four nodes, witness is **strongly recommended**.
- If you have internet access, use a **cloud witness**
- If you're in an IT environment with other machines and file shares, use a file share witness

How cluster quorum works

When nodes fail, or when some subset of nodes loses contact with another subset, surviving nodes need to verify that they constitute the *majority* of the cluster to remain online. If they can't verify that, they'll go offline.

But the concept of *majority* only works cleanly when the total number of nodes in the cluster is odd (for example, three nodes in a five node cluster). So, what about clusters with an even number of nodes (say, a four node cluster)?

There are two ways the cluster can make the *total number of votes* odd:

1. First, it can go *up* one by adding a *witness* with an extra vote. This requires user set-up.
2. Or, it can go *down* one by zeroing one unlucky node's vote (happens automatically as needed).

Whenever surviving nodes successfully verify they are the *majority*, the definition of *majority* is updated to be among just the survivors. This allows the cluster to lose one node, then another, then another, and so forth. This concept of the *total number of votes* adapting after successive failures is known as **Dynamic quorum**.

Dynamic witness

Dynamic witness toggles the vote of the witness to make sure that the *total number of votes* is odd. If there are an odd number of votes, the witness doesn't have a vote. If there is an even number of votes, the witness has a vote. Dynamic witness significantly reduces the risk that the cluster will go down because of witness failure. The cluster decides whether to use the witness vote based on the number of voting nodes that are available in the cluster.

Dynamic quorum works with Dynamic witness in the way described below.

Dynamic quorum behavior

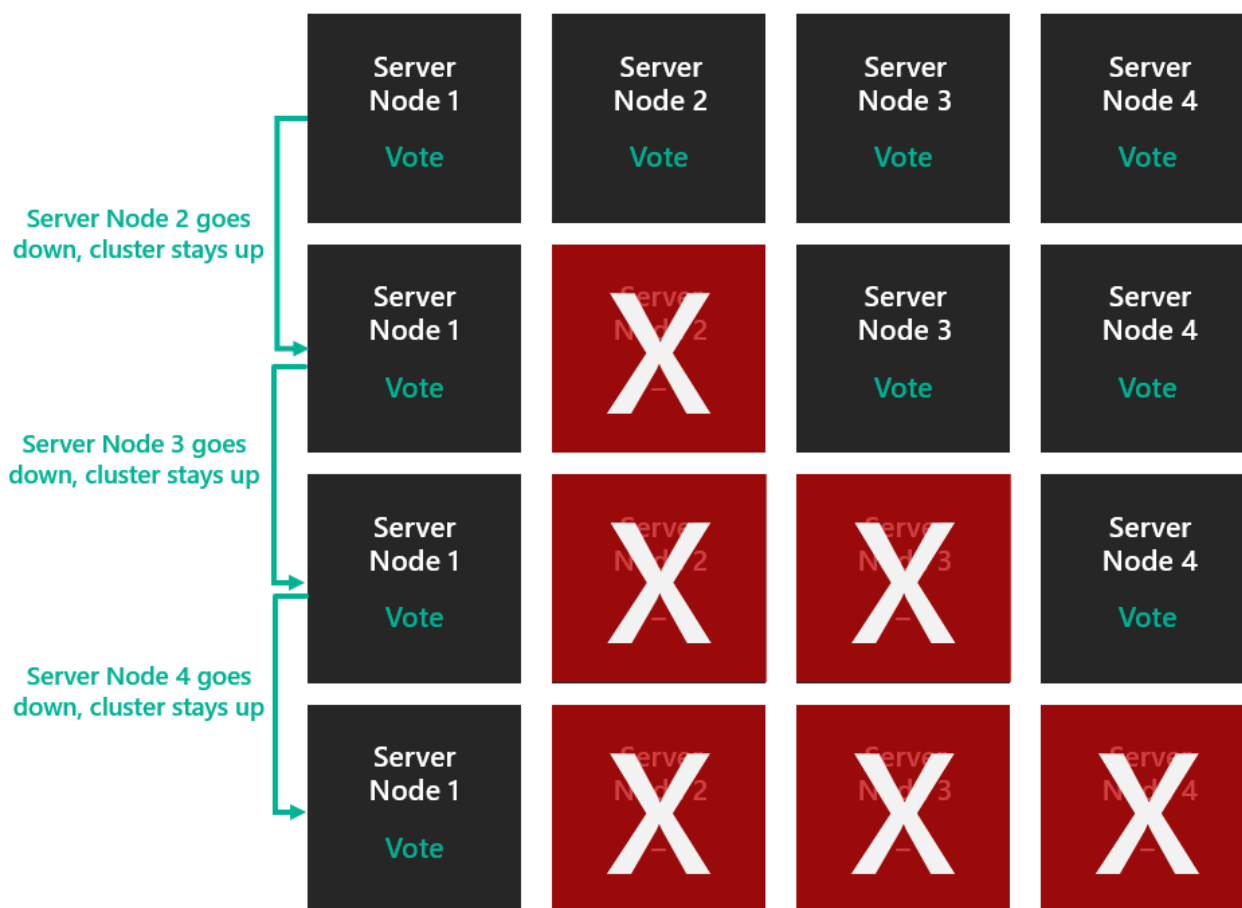
- If you have an **even** number of nodes and no witness, *one node gets its vote zeroed*. For example, only three of the four nodes get votes, so the *total number of votes* is three, and two survivors with votes are considered a majority.
- If you have an **odd** number of nodes and no witness, *they all get votes*.
- If you have an **even** number of nodes plus witness, *the witness votes*, so the total is odd.
- If you have an **odd** number of nodes plus witness, *the witness doesn't vote*.

Dynamic quorum enables the ability to assign a vote to a node dynamically to avoid losing the majority of votes and to allow the cluster to run with one node (known as last-man standing). Let's take a four-node cluster as an example. Assume that quorum requires 3 votes.

In this case, the cluster would have gone down if you lost two nodes.



However, dynamic quorum prevents this from happening. The *total number of votes* required for quorum is now determined based on the number of nodes available. So, with dynamic quorum, the cluster will stay up even if you lose three nodes.

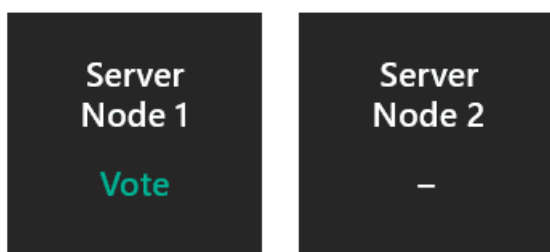


The above scenario applies to a general cluster that doesn't have Storage Spaces Direct enabled. However, when Storage Spaces Direct is enabled, the cluster can only support two node failures. This is explained more in the [pool quorum](#) section.

Examples

Two nodes without a witness.

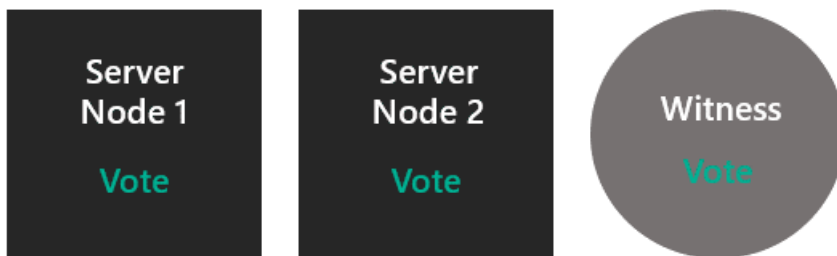
One node's vote is zeroed, so the *majority* vote is determined out of a total of **1 vote**. If the non-voting node goes down unexpectedly, the survivor has 1/1 and the cluster survives. If the voting node goes down unexpectedly, the survivor has 0/1 and the cluster goes down. If the voting node is gracefully powered down, the vote is transferred to the other node, and the cluster survives. ***This is why it's critical to configure a witness.***



- Can survive one server failure: **Fifty percent chance.**
- Can survive one server failure, then another: **No.**
- Can survive two server failures at once: **No.**

Two nodes with a witness.

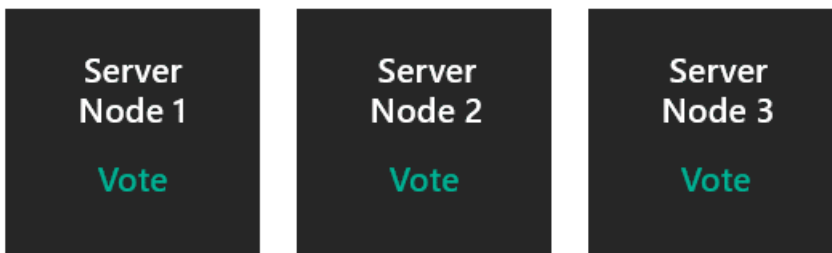
Both nodes vote, plus the witness votes, so the *majority* is determined out of a total of **3 votes**. If either node goes down, the survivor has 2/3 and the cluster survives.



- Can survive one server failure: **Yes**.
- Can survive one server failure, then another: **No**.
- Can survive two server failures at once: **No**.

Three nodes without a witness.

All nodes vote, so the *majority* is determined out of a total of **3 votes**. If any node goes down, the survivors are 2/3 and the cluster survives. The cluster becomes two nodes without a witness – at that point, you're in Scenario 1.



- Can survive one server failure: **Yes**.
- Can survive one server failure, then another: **Fifty percent chance**.
- Can survive two server failures at once: **No**.

Three nodes with a witness.

All nodes vote, so the witness doesn't initially vote. The *majority* is determined out of a total of **3 votes**. After one failure, the cluster has two nodes with a witness – which is back to Scenario 2. So, now the two nodes and the witness vote.



- Can survive one server failure: **Yes**.
- Can survive one server failure, then another: **Yes**.
- Can survive two server failures at once: **No**.

Four nodes without a witness

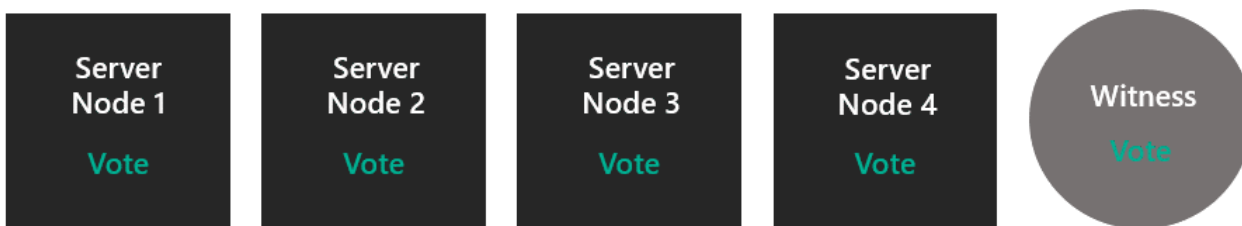
One node's vote is zeroed, so the *majority* is determined out of a total of **3 votes**. After one failure, the cluster becomes three nodes, and you're in Scenario 3.



- Can survive one server failure: **Yes**.
- Can survive one server failure, then another: **Yes**.
- Can survive two server failures at once: **Fifty percent chance**.

Four nodes with a witness.

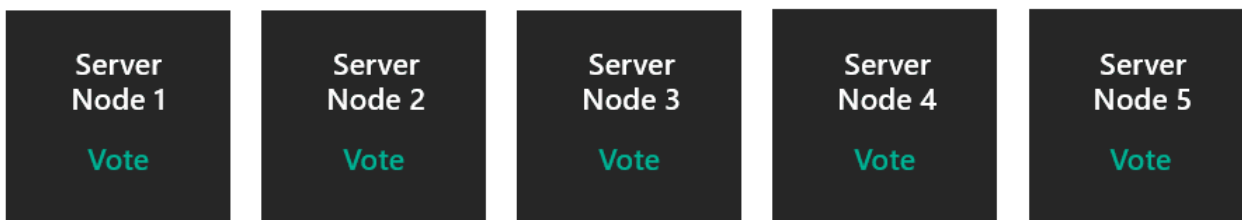
All nodes votes and the witness votes, so the *majority* is determined out of a total of **5 votes**. After one failure, you're in Scenario 4. After two simultaneous failures, you skip down to Scenario 2.



- Can survive one server failure: **Yes**.
- Can survive one server failure, then another: **Yes**.
- Can survive two server failures at once: **Yes**.

Five nodes and beyond.

All nodes vote, or all but one vote, whatever makes the total odd. Storage Spaces Direct cannot handle more than two nodes down anyway, so at this point, no witness is needed or useful.



- Can survive one server failure: **Yes**.
- Can survive one server failure, then another: **Yes**.
- Can survive two server failures at once: **Yes**.

Now that we understand how quorum works, let's look at the types of quorum witnesses.

Quorum witness types

Failover Clustering supports three types of Quorum Witnesses:

- **Cloud Witness** - Blob storage in Azure accessible by all nodes of the cluster. It maintains clustering information in a witness.log file, but doesn't store a copy of the cluster database.
- **File Share Witness** – A SMB file share that is configured on a file server running Windows Server. It maintains clustering information in a witness.log file, but doesn't store a copy of the cluster database.
- **Disk Witness** - A small clustered disk which is in the Cluster Available Storage group. This disk is highly-available and can failover between nodes. It contains a copy of the cluster database. **A Disk Witness isn't supported with Storage Spaces Direct.**

Pool quorum overview

We just talked about Cluster Quorum, which operates at the cluster level. Now, let's dive into Pool Quorum, which operates on the pool level (i.e. you can lose nodes and drives and have the pool stay up). Storage pools were designed to be used in both clustered and non-clustered scenarios, which is why they have a different quorum mechanism.

The table below gives an overview of the Pool Quorum outcomes per scenario:

SERVER NODES	CAN SURVIVE ONE SERVER NODE FAILURE	CAN SURVIVE ONE SERVER NODE FAILURE, THEN ANOTHER	CAN SURVIVE TWO SIMULTANEOUS SERVER NODE FAILURES
2	No	No	No
2 + Witness	Yes	No	No
3	Yes	No	No
3 + Witness	Yes	No	No
4	Yes	No	No
4 + Witness	Yes	Yes	Yes
5 and above	Yes	Yes	Yes

How pool quorum works

When drives fail, or when some subset of drives loses contact with another subset, surviving drives need to verify that they constitute the *majority* of the pool to remain online. If they can't verify that, they'll go offline. The pool is the entity that goes offline or stays online based on whether it has enough disks for quorum ($50\% + 1$). The pool resource owner (active cluster node) can be the +1.

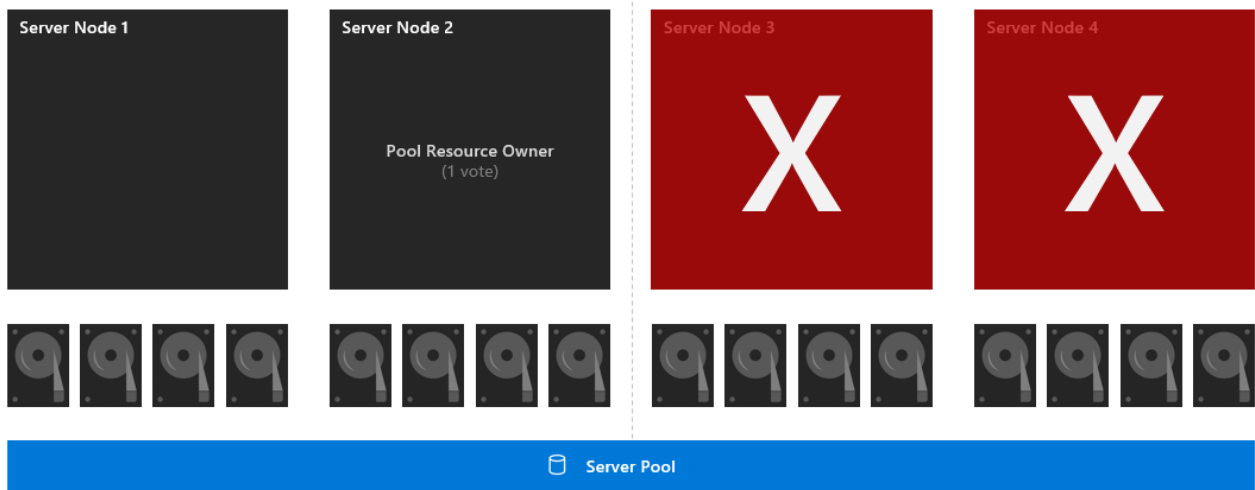
But pool quorum works differently from cluster quorum in the following ways:

- the pool uses one node in the cluster as a witness as a tie-breaker to survive half of drives gone (this node that is the pool resource owner)
- the pool does NOT have dynamic quorum
- the pool does NOT implement its own version of removing a vote

Examples

Four nodes with a symmetrical layout.

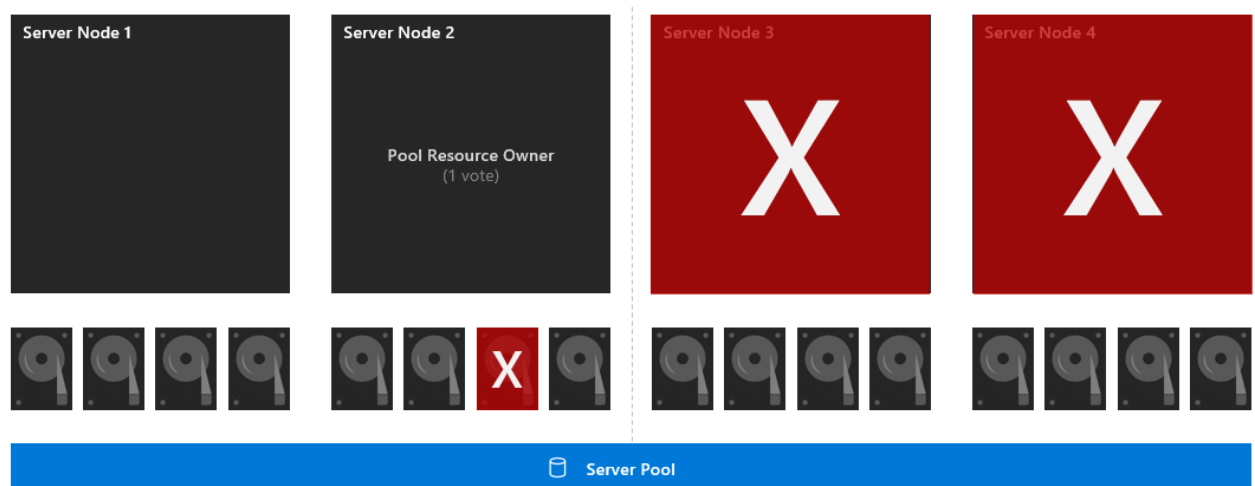
Each of the 16 drives has one vote and node two also has one vote (since it's the pool resource owner). The *majority* is determined out of a total of **16 votes**. If nodes three and four go down, the surviving subset has 8 drives and the pool resource owner, which is 9/16 votes. So, the pool survives.



- Can survive one server failure: **Yes**.
- Can survive one server failure, then another: **Yes**.
- Can survive two server failures at once: **Yes**.

Four nodes with a symmetrical layout and drive failure.

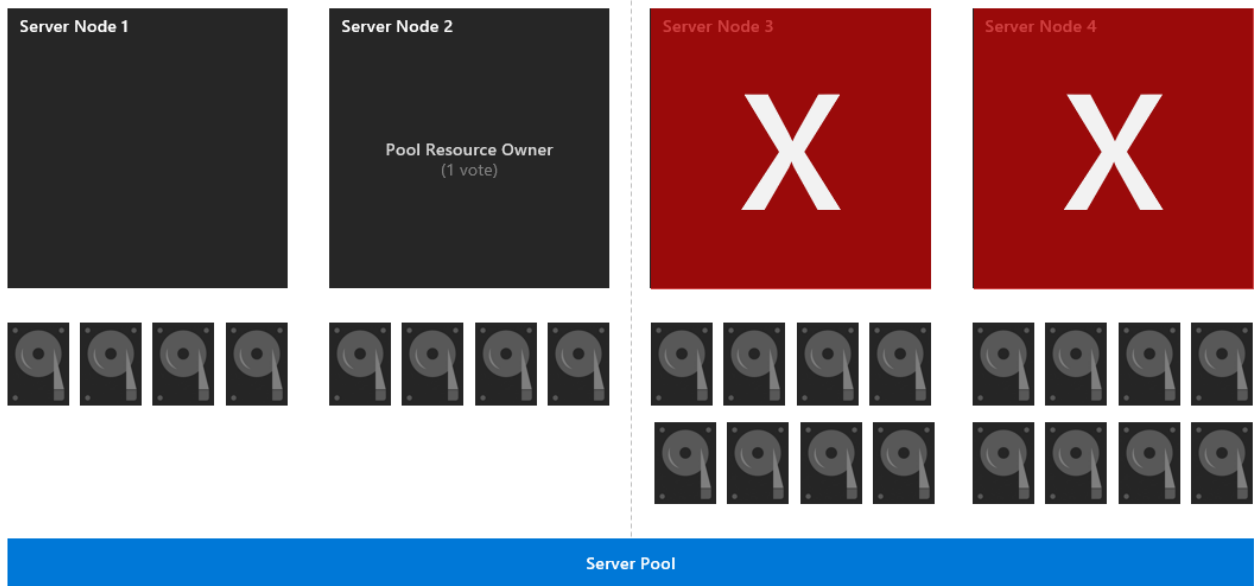
Each of the 16 drives has one vote and node 2 also has one vote (since it's the pool resource owner). The *majority* is determined out of a total of **16 votes**. First, drive 7 goes down. If nodes three and four go down, the surviving subset has 7 drives and the pool resource owner, which is 8/16 votes. So, the pool doesn't have majority and goes down.



- Can survive one server failure: **Yes**.
- Can survive one server failure, then another: **No**.
- Can survive two server failures at once: **No**.

Four nodes with a non-symmetrical layout.

Each of the 24 drives has one vote and node two also has one vote (since it's the pool resource owner). The *majority* is determined out of a total of **24 votes**. If nodes three and four go down, the surviving subset has 8 drives and the pool resource owner, which is 9/24 votes. So, the pool doesn't have majority and goes down.



- Can survive one server failure: **Yes**.
- Can survive one server failure, then another: **Depends** (cannot survive if both nodes three and four go down, but can survive all other scenarios).
- Can survive two server failures at once: **Depends** (cannot survive if both nodes three and four go down, but can survive all other scenarios).

Therefore, our guidance is:

- Ensure that each node in your cluster is symmetrical (each node has the same number of drives)
- Enable three-way mirror or dual parity so that you can tolerate a node failures and keep the virtual disks online. See our [volume guidance page](#) for more details.
- If more than two nodes are down, or two nodes and a disk on another node are down, volumes may not have access to all three copies of their data, and therefore be taken offline and be unavailable. It's recommended to bring the servers back or replace the disks quickly to ensure the most resiliency for all the data in the volume.

More information

For additional information on how to configure and manage quorum, see documentation on [configure and manage quorum](#).

Fault domain awareness in Windows Server 2016

10/17/2017 • 7 minutes to read • [Edit Online](#)

Applies to: Windows Server 2016

Failover Clustering enables multiple servers to work together to provide high availability – or put another way, to provide node fault tolerance. But today's businesses demand ever-greater availability from their infrastructure. To achieve cloud-like uptime, even highly unlikely occurrences such as chassis failures, rack outages, or natural disasters must be protected against. That's why Failover Clustering in Windows Server 2016 introduces chassis, rack, and site fault tolerance as well.

Fault domains and fault tolerance are closely related concepts. A fault domain is a set of hardware components that share a single point of failure. To be fault tolerant to a certain level, you need multiple fault domains at that level. For example, to be rack fault tolerant, your servers and your data must be distributed across multiple racks.

This short video presents an overview of fault domains in Windows Server 2016:

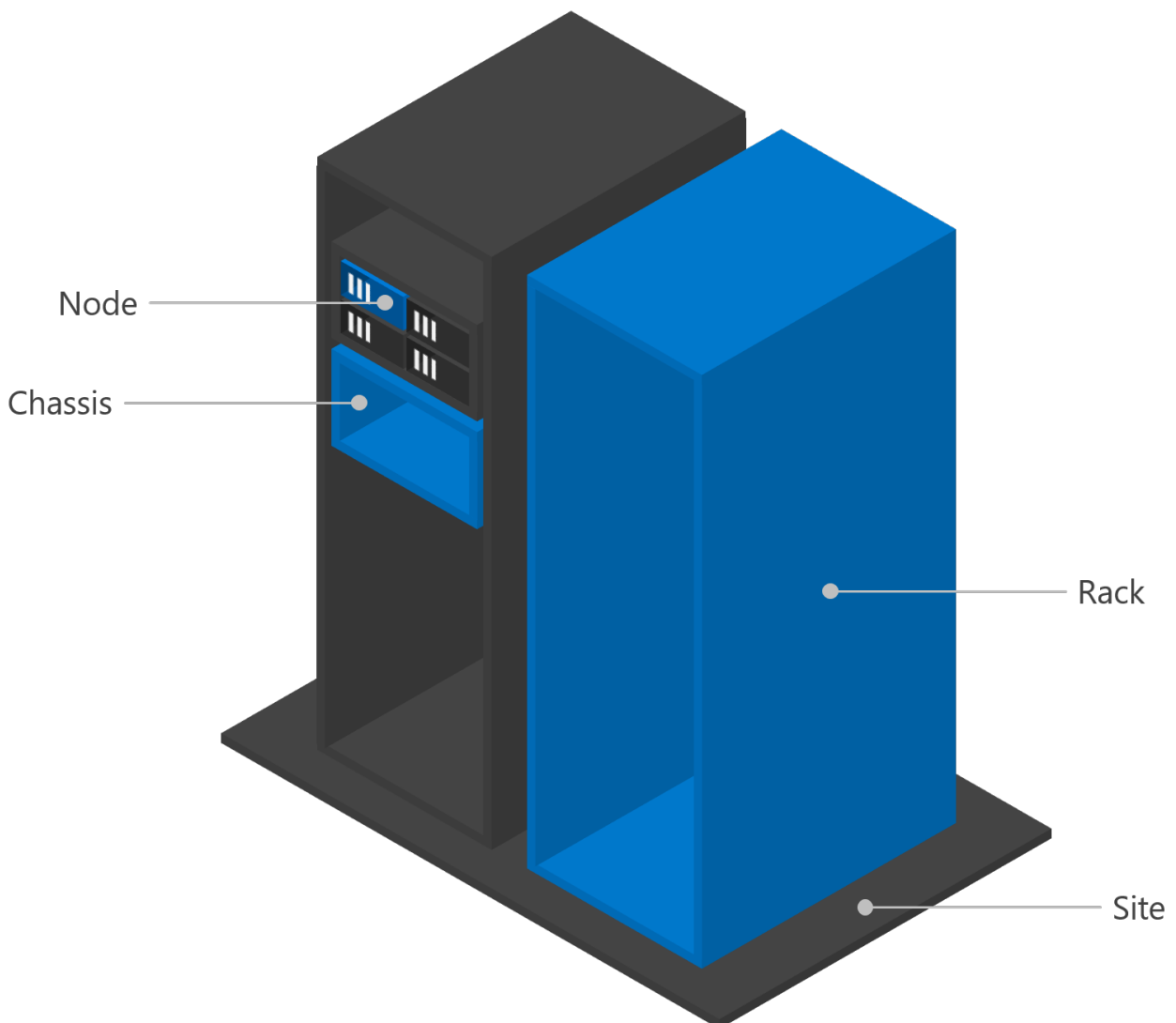


Benefits

- **Storage Spaces, including Storage Spaces Direct, uses fault domains to maximize data safety.**
Resiliency in Storage Spaces is conceptually like distributed, software-defined RAID. Multiple copies of all data are kept in sync, and if hardware fails and one copy is lost, others are recopied to restore resiliency. To achieve the best possible resiliency, copies should be kept in separate fault domains.
- **The [Health Service](#) uses fault domains to provide more helpful alerts.**
Each fault domain can be associated with location metadata, which will automatically be included in any subsequent alerts. These descriptors can assist operations or maintenance personnel and reduce errors by disambiguating hardware.
- **Stretch clustering uses fault domains for storage affinity.** Stretch clustering allows faraway servers to join a common cluster. For the best performance, applications or virtual machines should be run on servers that are nearby to those providing their storage. Fault domain awareness enables this storage affinity.

Levels of fault domains

There are four canonical levels of fault domains - site, rack, chassis, and node. Nodes are discovered automatically; each additional level is optional. For example, if your deployment does not use blade servers, the chassis level may not make sense for you.



Usage

You can use PowerShell or XML markup to specify fault domains. Both approaches are equivalent and provide full functionality.

IMPORTANT

Specify fault domains before enabling Storage Spaces Direct, if possible. This enables the automatic configuration to prepare the pool, tiers, and settings like resiliency and column count, for chassis or rack fault tolerance. Once the pool and volumes have been created, data will not retroactively move in response to changes to the fault domain topology. To move nodes between chassis or racks after enabling Storage Spaces Direct, you should first evict the node and its drives from the pool using `Remove-ClusterNode -CleanupDisks`.

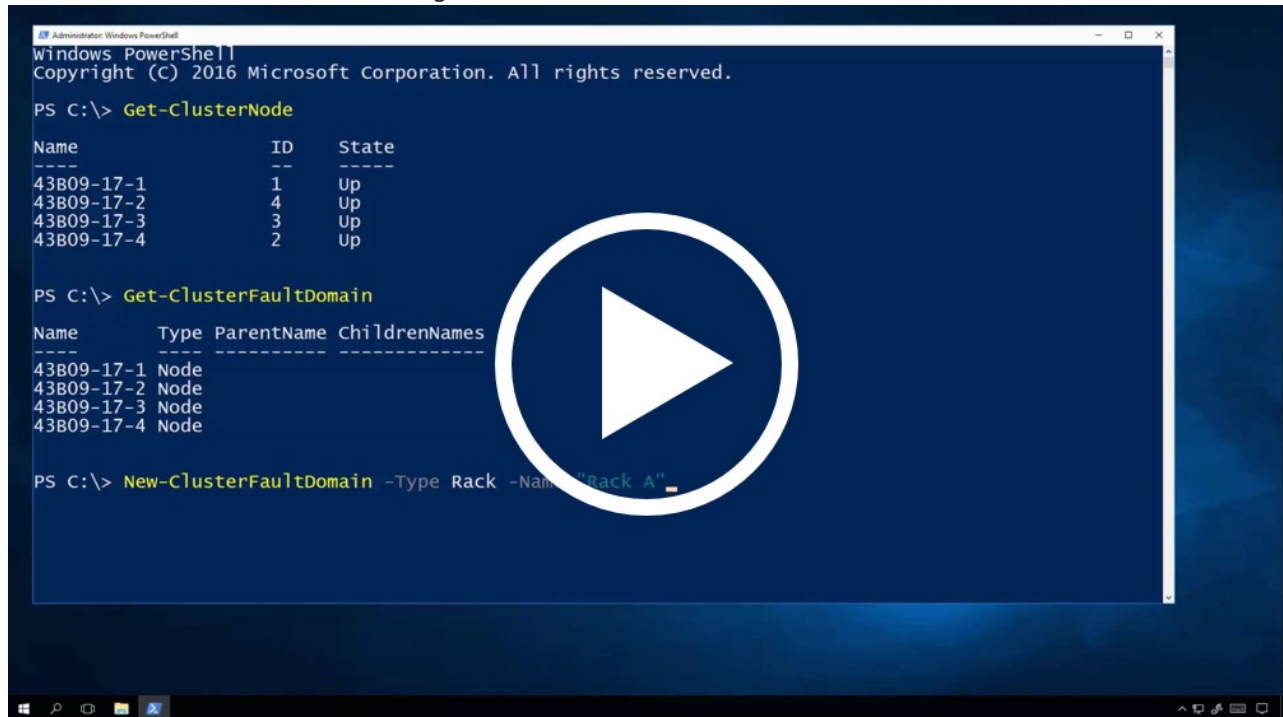
Defining fault domains with PowerShell

Windows Server 2016 introduces the following cmdlets to work with fault domains:

- `Get-ClusterFaultDomain`
- `Set-ClusterFaultDomain`

- `New-ClusterFaultDomain`
- `Remove-ClusterFaultDomain`

This short video demonstrates the usage of these cmdlets.



Use `Get-ClusterFaultDomain` to see the current fault domain topology. This will list all nodes in the cluster, plus any chassis, racks, or sites you have created. You can filter using parameters like **-Type** or **-Name**, but these are not required.

```

Get-ClusterFaultDomain
Get-ClusterFaultDomain -Type Rack
Get-ClusterFaultDomain -Name "server01.contoso.com"

```

Use `New-ClusterFaultDomain` to create new chassis, racks, or sites. The `-Type` and `-Name` parameters are required. The possible values for `-Type` are `Chassis`, `Rack`, and `Site`. The `-Name` can be any string. (For `Node` type fault domains, the name must be the actual node name, as set automatically).

```

New-ClusterFaultDomain -Type Chassis -Name "Chassis 007"
New-ClusterFaultDomain -Type Rack -Name "Rack A"
New-ClusterFaultDomain -Type Site -Name "Shanghai"

```

IMPORTANT

Windows Server cannot and does not verify that any fault domains you create correspond to anything in the real, physical world. (This may sound obvious, but it's important to understand.) If, in the physical world, your nodes are all in one rack, then creating two `-Type Rack` fault domains in software does not magically provide rack fault tolerance. You are responsible for ensuring the topology you create using these cmdlets matches the actual arrangement of your hardware.

Use `Set-ClusterFaultDomain` to move one fault domain into another. The terms "parent" and "child" are commonly used to describe this nesting relationship. The `-Name` and `-Parent` parameters are required. In `-Name`, provide the name of the fault domain that is moving; in `-Parent`, provide the name of the destination. To move multiple fault domains at once, list their names.


```
Set-ClusterFaultDomain -Name "server01.contoso.com" -Parent "Rack A"
Set-ClusterFaultDomain -Name "Rack A", "Rack B", "Rack C", "Rack D" -Parent "Shanghai"
```

IMPORTANT

When fault domains move, their children move with them. In the above example, if Rack A is the parent of server01.contoso.com, the latter does not separately need to be moved to the Shanghai site – it is already there by virtue of its parent being there, just like in the physical world.

You can see parent-child relationships in the output of `Get-ClusterFaultDomain`, in the `ParentName` and `ChildrenNames` columns.

You can also use `Set-ClusterFaultDomain` to modify certain other properties of fault domains. For example, you can provide optional `-Location` or `-Description` metadata for any fault domain. If provided, this information will be included in hardware alerting from the Health Service. You can also rename fault domains using the `-NewName` parameter. Do not rename `Node` type fault domains.

```
Set-ClusterFaultDomain -Name "Rack A" -Location "Building 34, Room 4010"
Set-ClusterFaultDomain -Type Node -Description "Contoso XYZ Server"
Set-ClusterFaultDomain -Name "Shanghai" -NewName "China Region"
```

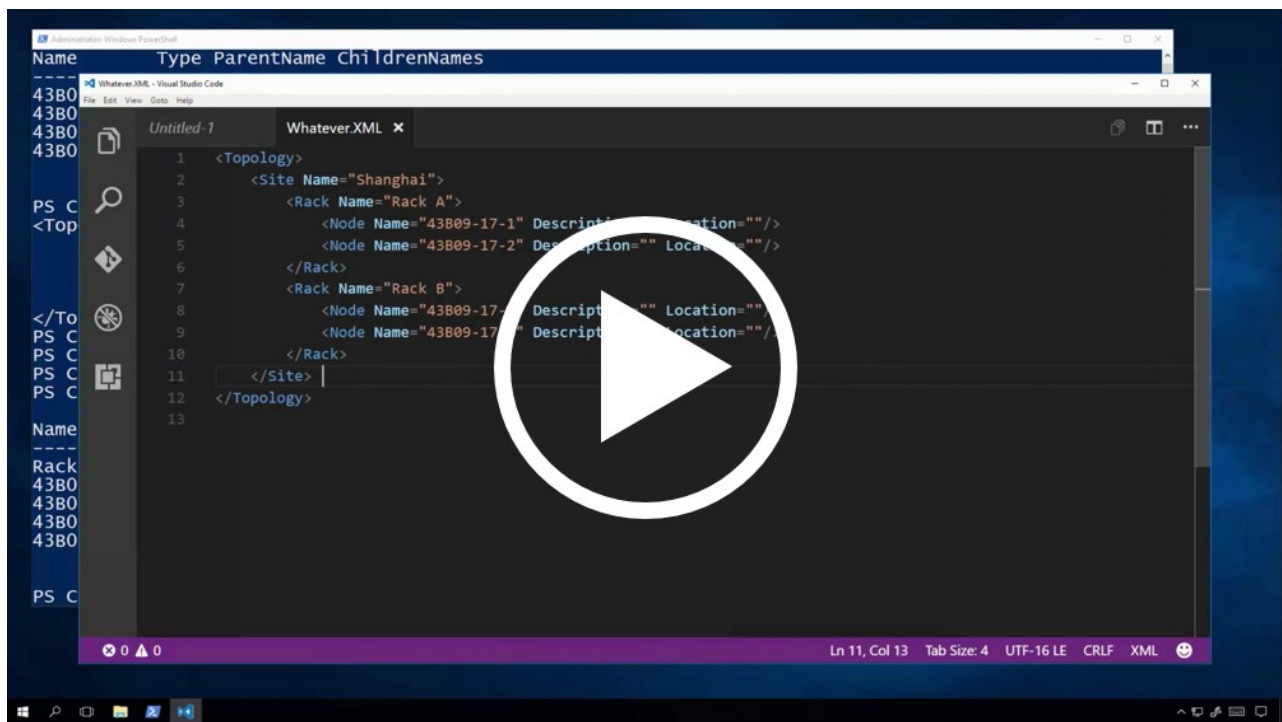
Use `Remove-ClusterFaultDomain` to remove chassis, racks, or sites you have created. The `-Name` parameter is required. You cannot remove a fault domain that contains children – first, either remove the children, or move them outside using `Set-ClusterFaultDomain`. To move a fault domain outside of all other fault domains, set its `-Parent` to the empty string (""). You cannot remove `Node` type fault domains. To remove multiple fault domains at once, list their names.

```
Set-ClusterFaultDomain -Name "server01.contoso.com" -Parent ""
Remove-ClusterFaultDomain -Name "Rack A"
```

Defining fault domains with XML markup

Fault domains can be specified using an XML-inspired syntax. We recommend using your favorite text editor, such as Visual Studio Code (available for free [here](#)) or Notepad, to create an XML document which you can save and reuse.

This short video demonstrates the usage of XML Markup to specify fault domains.



In PowerShell, run the following cmdlet: `Get-ClusterFaultDomainXML`. This returns the current fault domain specification for the cluster, as XML. This reflects every discovered `<Node>`, wrapped in opening and closing `<Topology>` tags.

Run the following to save this output to a file.

```
Get-ClusterFaultDomainXML | Out-File <Path>
```

Open the file, and add `<Site>`, `<Rack>`, and `<Chassis>` tags to specify how these nodes are distributed across sites, racks, and chassis. Every tag must be identified by a unique **Name**. For nodes, you must keep the node's name as populated by default.

IMPORTANT

While all additional tags are optional, they must adhere to the transitive Site > Rack > Chassis > Node hierarchy, and must be properly closed.

In addition to name, freeform `Location="..."` and `Description="..."` descriptors can be added to any tag.

Example: Two sites, one rack each

```
<Topology>
  <Site Name="SEA" Location="Contoso HQ, 123 Example St, Room 4010, Seattle">
    <Rack Name="A01" Location="Aisle A, Rack 01">
      <Node Name="Server01" Location="Rack Unit 33" />
      <Node Name="Server02" Location="Rack Unit 35" />
      <Node Name="Server03" Location="Rack Unit 37" />
    </Rack>
  </Site>
  <Site Name="NYC" Location="Regional Datacenter, 456 Example Ave, New York City">
    <Rack Name="B07" Location="Aisle B, Rack 07">
      <Node Name="Server04" Location="Rack Unit 20" />
      <Node Name="Server05" Location="Rack Unit 22" />
      <Node Name="Server06" Location="Rack Unit 24" />
    </Rack>
  </Site>
</Topology>
```

Example: two chassis, blade servers

```
<Topology>
  <Rack Name="A01" Location="Contoso HQ, Room 4010, Aisle A, Rack 01">
    <Chassis Name="Chassis01" Location="Rack Unit 2 (Upper)" >
      <Node Name="Server01" Location="Left" />
      <Node Name="Server02" Location="Right" />
    </Chassis>
    <Chassis Name="Chassis02" Location="Rack Unit 6 (Lower)" >
      <Node Name="Server03" Location="Left" />
      <Node Name="Server04" Location="Right" />
    </Chassis>
  </Rack>
</Topology>
```

To set the new fault domain specification, save your XML and run the following in PowerShell.

```
$xml = Get-Content <Path> | Out-String
Set-ClusterFaultDomainXML -XML $xml
```

This guide presents just two examples, but the `<Site>`, `<Rack>`, `<Chassis>`, and `<Node>` tags can be mixed and matched in many additional ways to reflect the physical topology of your deployment, whatever that may be. We hope these examples illustrate the flexibility of these tags and the value of freeform location descriptors to disambiguate them.

Optional: Location and description metadata

You can provide optional **Location** or **Description** metadata for any fault domain. If provided, this information will be included in hardware alerting from the Health Service. This short video demonstrates the value of adding such descriptors.



See Also

- [Windows Server 2016](#)
- [Storage Spaces Direct in Windows Server 2016](#)

Simplified SMB Multichannel and Multi-NIC Cluster Networks

10/17/2017 • 3 minutes to read • [Edit Online](#)

Applies to: Windows Server (Semi-Annual Channel), Windows Server 2016

Simplified SMB Multichannel and Multi-NIC Cluster Networks is a new feature in Windows Server 2016 that enables the use of multiple NICs on the same cluster network subnet, and automatically enables SMB Multichannel.

Simplified SMB Multichannel and Multi-NIC Cluster Networks provides the following benefits:

- Failover Clustering automatically recognizes all NICs on nodes that are using the same switch / same subnet - no additional configuration needed.
- SMB Multichannel is enabled automatically.
- Networks that only have IPv6 Link Local (fe80) IP Addresses resources are recognized on cluster-only (private) networks.
- A single IP Address resource is configured on each Cluster Access Point (CAP) Network Name (NN) by default.
- Cluster validation no longer issues warning messages when multiple NICs are found on the same subnet.

Requirements

- Multiple NICs per server, using the same switch / subnet.

How to take advantage of multi-NIC clusters networks and simplified SMB multichannel

This section describes how to take advantage of the new multi-NIC clusters networks and simplified SMB multichannel features in Windows Server 2016.

Use at least two networks for Failover Clustering

Although it is rare, network switches can fail - it is still best practice to use at least two networks for Failover Clustering. All networks that are found are used for cluster heartbeats. Avoid using a single network for your Failover Cluster in order to avoid a single point of failure. Ideally, there should be multiple physical communication paths between the nodes in the cluster, and no single point of failure.

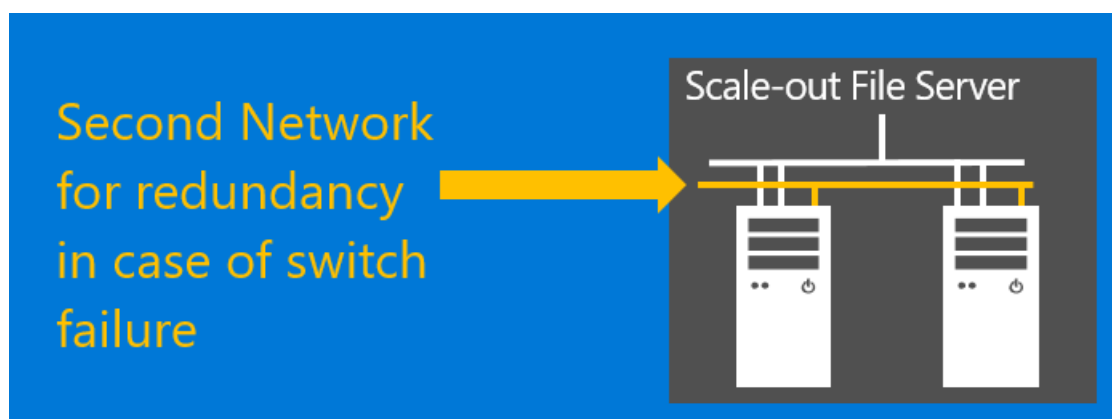


Figure 1: Use at least two networks for Failover Clustering

Use Multiple NICs across clusters

Maximum benefit of the simplified SMB multichannel is achieved when multiple NICs are used across clusters - in both storage and storage workload clusters. This allows the workload clusters (Hyper-V, SQL Server Failover Cluster Instance, Storage Replica, etc.) to use SMB multichannel and results in more efficient use of the network. In a converged (also known as disaggregated) cluster configuration where a Scale-out File Server cluster is used for storing workload data for a Hyper-V or SQL Server Failover Cluster Instance cluster, this network is often called "the North-South subnet" / network. Many customers maximize throughput of this network by investing in RDMA capable NIC cards and switches.

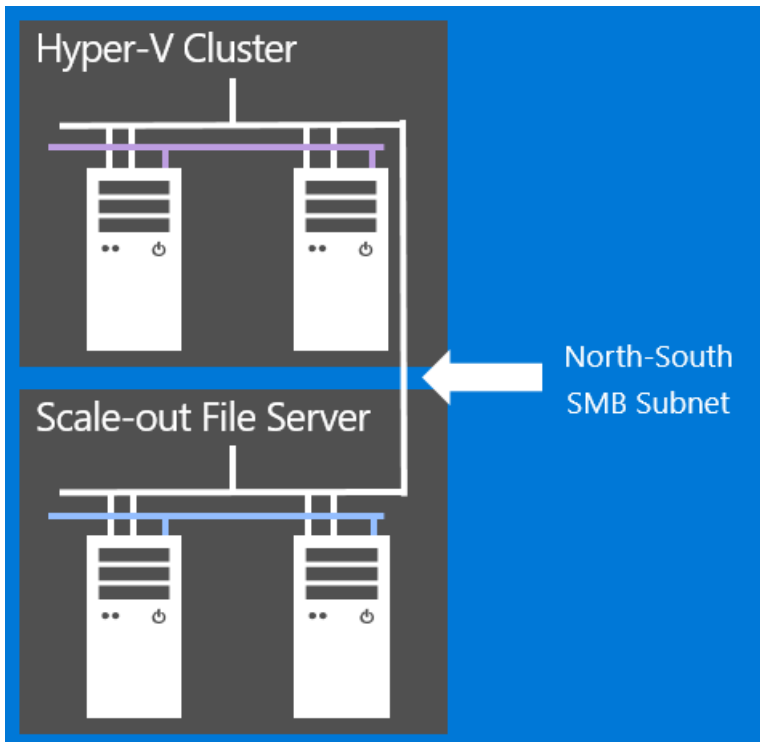


Figure 2: To achieve maximum network throughput, use multiple NICs on both the Scale-out File Server cluster and the Hyper-V or SQL Server Failover Cluster Instance cluster - which share the North-South subnet

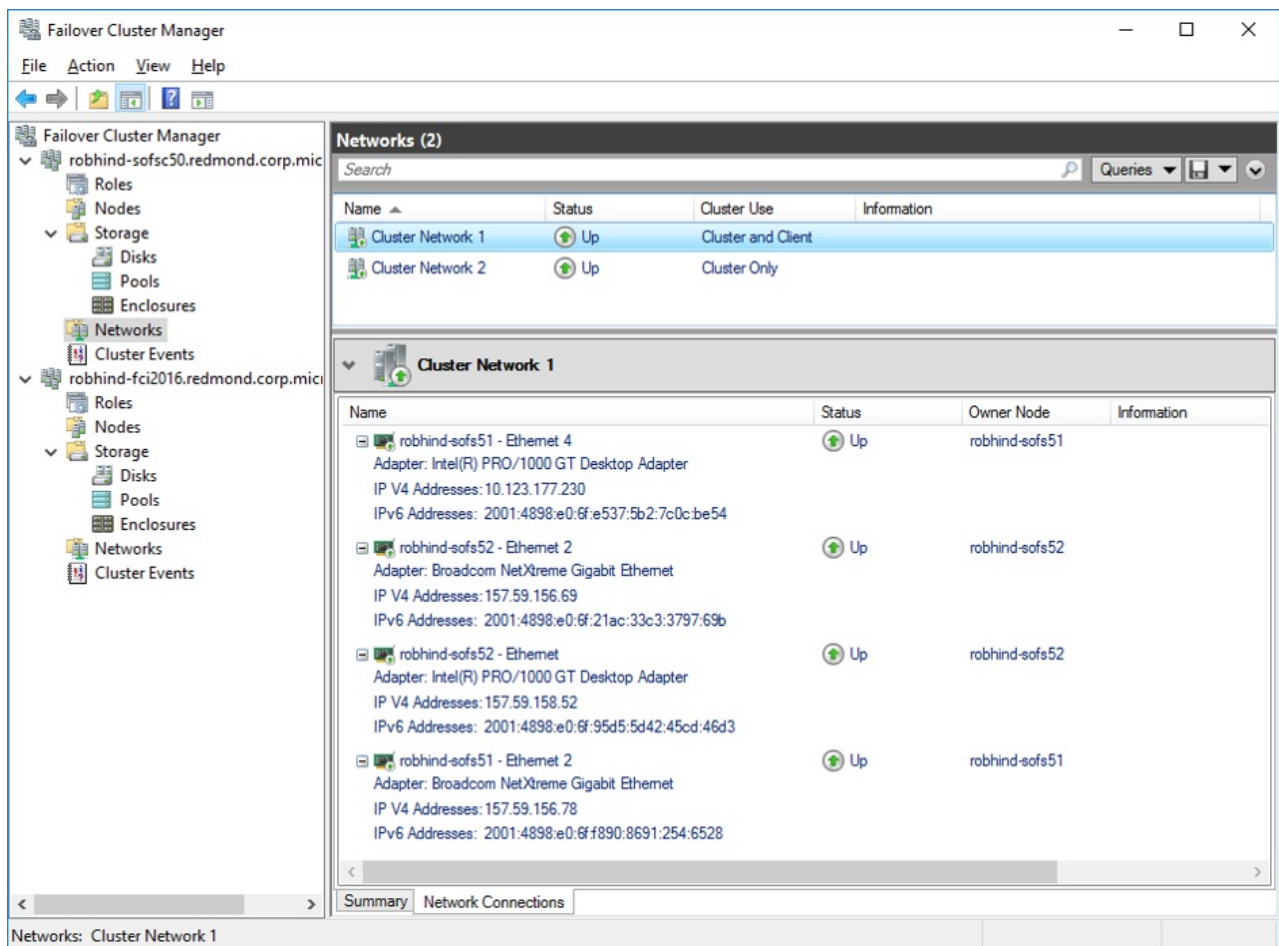


Figure 3: Two clusters (Scale-out File Server for storage, SQL Server FCI for workload) both use multiple NICs in the same subnet to leverage SMB Multichannel and achieve better network throughput.

Automatic recognition of IPv6 Link Local private networks

When private (cluster only) networks with multiple NICs are detected, the cluster will automatically recognize IPv6 Link Local (fe80) IP addresses for each NIC on each subnet. This saves administrators time since they no longer have to manually configure IPv6 Link Local (fe80) IP Address resources.

When using more than one private (cluster only) network, check the IPv6 routing configuration to ensure that routing is not configured to cross subnets, since this will reduce network performance.

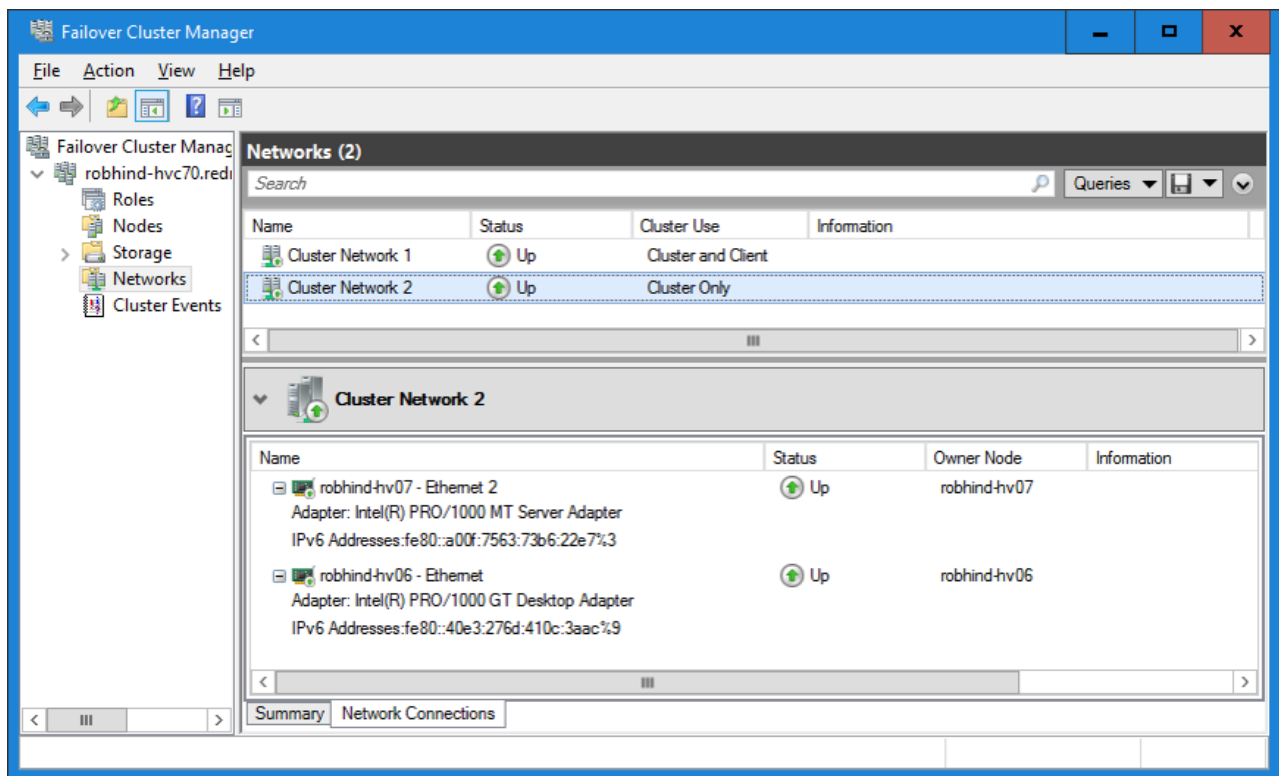


Figure 4: Automatic IPv6 Link Local (fe80) Address resource configuration

Throughput and Fault Tolerance

Windows Server 2016 automatically detects NIC capabilities and will attempt to use each NIC in the fastest possible configuration. NICs that are teamed, NICs using RSS, and NICs with RDMA capability can all be used. The table below summarizes the trade-offs when using these technologies. Maximum throughput is achieved when using multiple RDMA capable NICs. For more information, see [The basics of SMB Multichannel](#).

	Throughput	Fault Tolerance for SMB	Fault Tolerance for non-SMB	Lower CPU utilization
Single NIC (no RSS)	▲			
Multiple NICs (no RSS)	▲▲	▲		
Multiple NICs (no RSS) + NIC Teaming	▲▲	▲▲	▲	
Single NIC (with RSS)	▲▲			
Multiple NICs (with RSS)	▲▲▲	▲		
Multiple NICs (with RSS) + NIC Teaming	▲▲▲	▲▲	▲	
Single NIC (with RDMA)	▲▲			▲
Multiple NICs (with RDMA)	▲▲▲	▲		▲

Figure 5: Throughput and fault tolerance for various NIC configurations

Frequently asked questions

Are all NICs in a multi-NIC network used for cluster heart beating?

Yes.

Can a multi-NIC network be used for cluster communication only? Or can it only be used for client and cluster communication?

Either configuration will work - all cluster network roles will work on a multi-NIC network.

Is SMB Multichannel also used for CSV and cluster traffic?

Yes, by default all cluster and CSV traffic will use available multi-NIC networks. Administrators can use the Failover Clustering PowerShell cmdlets or Failover Cluster Manager UI to change the network role.

How can I see the SMB Multichannel settings?

Use the **Get-SMBServerConfiguration** cmdlet, look for the value of the EnableMultiChannel property.

Is the cluster common property PlumbAllCrossSubnetRoutes respected on a multi-NIC network?

Yes.

See also

- [What's New in Failover Clustering in Windows Server](#)

Virtual Machine Load Balancing overview

3/20/2018 • 2 minutes to read • [Edit Online](#)

Applies to: Windows Server (Semi-Annual Channel), Windows Server 2016

A key consideration for private cloud deployments is the capital expenditure (CapEx) required to go into production. It is very common to add redundancy to private cloud deployments to avoid under-capacity during peak traffic in production, but this increases CapEx. The need for redundancy is driven by unbalanced private clouds where some nodes are hosting more Virtual Machines (VMs) and others are underutilized (such as a freshly rebooted server).

Quick Video Overview

(6 minutes)

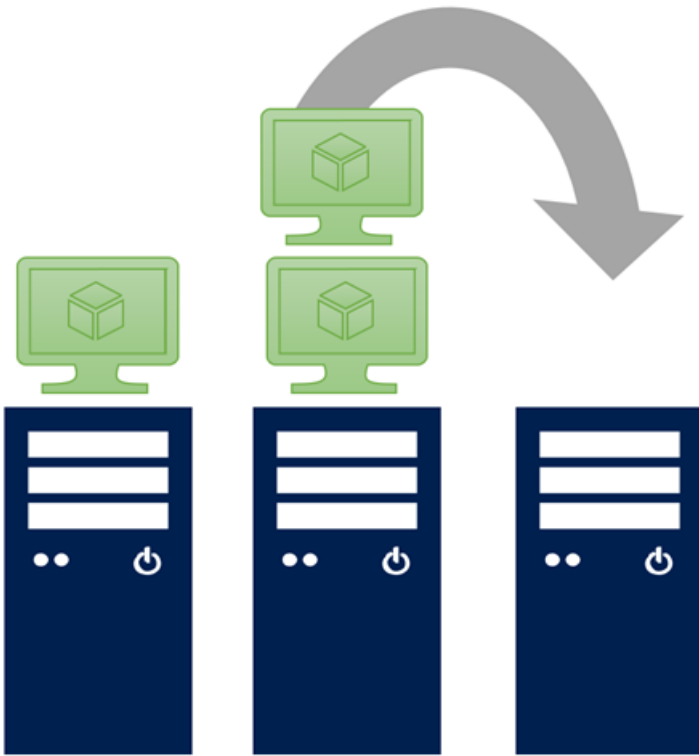
What is Virtual Machine Load Balancing?

VM Load Balancing is a new in-box feature in Windows Server 2016 that allows you to optimize the utilization of nodes in a Failover Cluster. It identifies over-committed nodes and re-distributes VMs from those nodes to under-committed nodes. Some of the salient aspects of this feature are as follows:

- *It is a zero-downtime solution:* VMs are live-migrated to idle nodes.
- *Seamless integration with your existing cluster environment:* Failure policies such as anti-affinity, fault domains and possible owners are honored.
- *Heuristics for balancing:* VM memory pressure and CPU utilization of the node.
- *Granular control:* Enabled by default. Can be activated on-demand or at a periodic interval.
- *Aggressiveness thresholds:* Three thresholds available based on the characteristics of your deployment.

The feature in action

A new node is added to your Failover Cluster



When you add new capacity to your Failover Cluster, the VM Load Balancing feature automatically balances capacity from the existing nodes, to the newly added node in the following order:

1. The pressure is evaluated on the existing nodes in the Failover Cluster.
2. All nodes exceeding threshold are identified.
3. The nodes with the highest pressure are identified to determine priority of balancing.
4. VMs are Live Migrated (with no down time) from a node exceeding threshold to a newly added node in the Failover Cluster.

Recurring load balancing



When configured for periodic balancing, the pressure on the cluster nodes is evaluated for balancing every 30 minutes. Alternately, the pressure can be evaluated on-demand. Here is the flow of the steps:

1. The pressure is evaluated on all nodes in the private cloud.
2. All nodes exceeding threshold and those below threshold are identified.
3. The nodes with the highest pressure are identified to determine priority of balancing.
4. VMs are Live Migrated (with no down time) from a node exceeding the threshold to node under minimum threshold.

See Also

- [Virtual Machine Load Balancing Deep-Dive](#)
- [Failover Clustering](#)
- [Hyper-V Overview](#)

Virtual Machine Load Balancing deep-dive

3/15/2018 • 2 minutes to read • [Edit Online](#)

Applies to: Windows Server (Semi-Annual Channel), Windows Server 2016

Windows Server 2016 introduces the [Virtual Machine Load Balancing feature](#) to optimize the utilization of nodes in a Failover Cluster. This document describes how to configure and control [VM](#) Load Balancing.

Heuristics for balancing

[VM](#) Virtual Machine Load Balancing evaluates a node's load based on the following heuristics:

1. Current **memory pressure**: Memory is the most common resource constraint on a Hyper-V host
2. **CPU utilization** of the Node averaged over a 5 minute window: Mitigates a node in the cluster becoming over-committed

Controlling the aggressiveness of balancing

The aggressiveness of balancing based on the Memory and CPU heuristics can be configured using the by the cluster common property 'AutoBalancerLevel'. To control the aggressiveness run the following in PowerShell:

```
(Get-Cluster).AutoBalancerLevel = <value>
```

AUTOBALANCERLEVEL	AGGRESSIVENESS	BEHAVIOR
1 (default)	Low	Move when host is more than 80% loaded
2	Medium	Move when host is more than 70% loaded
3	High	Average nodes and move when host is more than 5% above average

```
PS C:\Windows\system32> (Get-Cluster).AutoBalancerMode = 2
PS C:\Windows\system32> (Get-Cluster).AutoBalancerLevel = 2
PS C:\Windows\system32> Get-Cluster | fl AutoBalancer*

AutoBalancerMode : 2
AutoBalancerLevel : 2

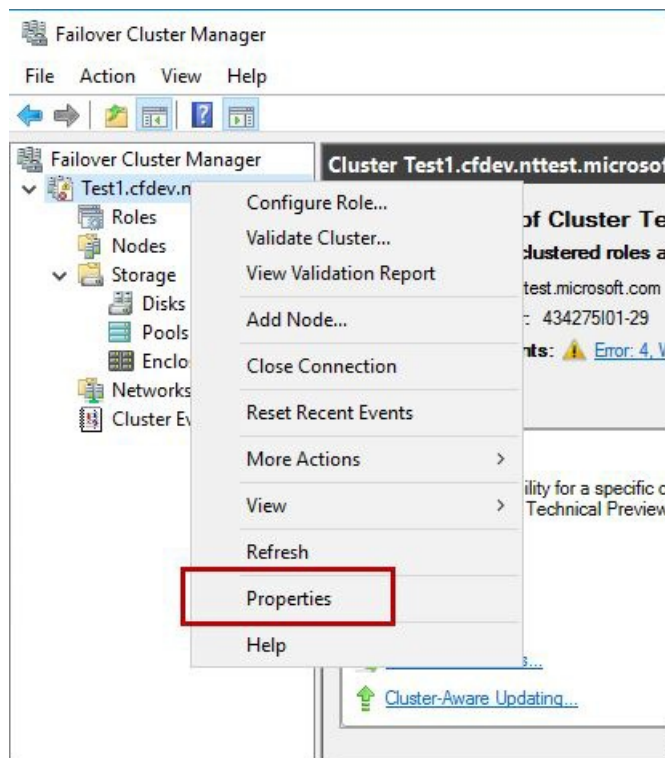
PS C:\Windows\system32> _
```

Controlling [VM](#) Load Balancing

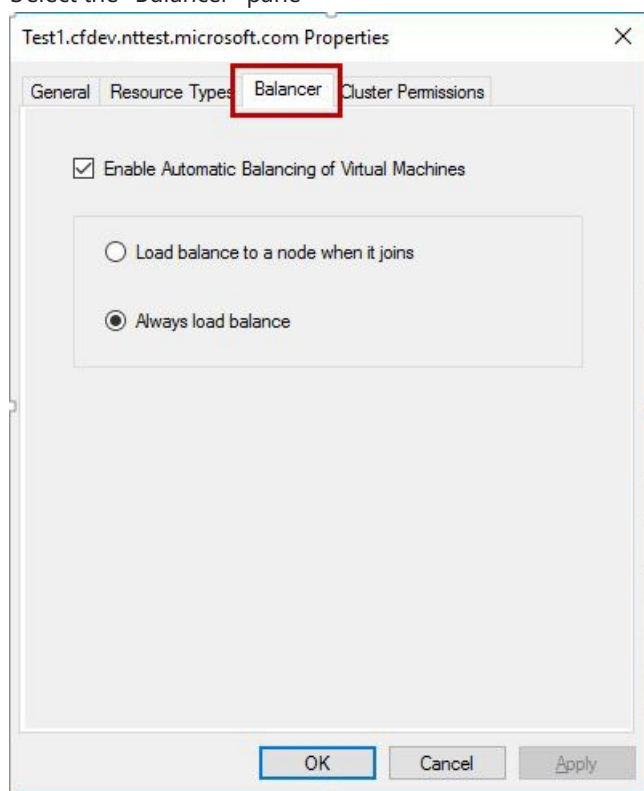
[VM](#) Load Balancing is enabled by default and when load balancing occurs can be configured by the cluster common property 'AutoBalancerMode'. To control when Node Fairness balances the cluster:

Using Failover Cluster Manager:

1. Right-click on your cluster name and select the "Properties" option



2. Select the "Balancer" pane



Using PowerShell:

Run the following:

```
(Get-Cluster).AutoBalancerMode = <value>
```

AUTOBALANCERMODE	BEHAVIOR
0	Disabled

AUTOBALANCERMODE	BEHAVIOR
1	Load balance on node join
2 (default)	Load balance on node join and every 30 minutes

VM Load Balancing vs. System Center Virtual Machine Manager Dynamic Optimization

The node fairness feature, provides in-box functionality, which is targeted towards deployments without System Center Virtual Machine Manager (SCVMM). SCVMM Dynamic Optimization is the recommended mechanism for balancing virtual machine load in your cluster for SCVMM deployments. SCVMM automatically disables the Windows Server VM Load Balancing when Dynamic Optimization is enabled.

See Also

- [Virtual Machine Load Balancing Overview](#)
- [Failover Clustering](#)
- [Hyper-V Overview](#)

Failover clustering hardware requirements and storage options

6/20/2018 • 5 minutes to read • [Edit Online](#)

Applies to: Windows Server 2012 R2, Windows Server 2012, Windows Server 2016

You need the following hardware to create a failover cluster. To be supported by Microsoft, all hardware must be certified for the version of Windows Server that you are running, and the complete failover cluster solution must pass all tests in the Validate a Configuration Wizard. For more information about validating a failover cluster, see [Validate Hardware for a Failover Cluster](#).

- **Servers:** We recommend that you use a set of matching computers that contain the same or similar components.
- **Network adapters and cable (for network communication):** If you use iSCSI, each network adapter should be dedicated to either network communication or iSCSI, not both.

In the network infrastructure that connects your cluster nodes, avoid having single points of failure. For example, you can connect your cluster nodes by multiple, distinct networks. Alternatively, you can connect your cluster nodes with one network that's constructed with teamed network adapters, redundant switches, redundant routers, or similar hardware that removes single points of failure.

NOTE

If you connect cluster nodes with a single network, the network will pass the redundancy requirement in the Validate a Configuration Wizard. However, the report from the wizard will include a warning that the network should not have single points of failure.

- **Device controllers or appropriate adapters for the storage:**
 - **Serial Attached SCSI or Fibre Channel:** If you are using Serial Attached SCSI or Fibre Channel, in all clustered servers, all elements of the storage stack should be identical. It's required that the multipath I/O (MPIO) software be identical and that the Device Specific Module (DSM) software be identical. It's recommended that the mass-storage device controllers—the host bus adapter (HBA), HBA drivers, and HBA firmware—that are attached to cluster storage be identical. If you use dissimilar HBAs, you should verify with the storage vendor that you are following their supported or recommended configurations.
 - **iSCSI:** If you are using iSCSI, each clustered server should have one or more network adapters or HBAs that are dedicated to the cluster storage. The network you use for iSCSI should not be used for network communication. In all clustered servers, the network adapters you use to connect to the iSCSI storage target should be identical, and we recommend that you use Gigabit Ethernet or higher.
- **Storage:** You must use [Storage Spaces Direct](#) or shared storage that's compatible with Windows Server 2012 R2 or Windows Server 2012. You can use shared storage that's attached, and you can also use SMB 3.0 file shares as shared storage for servers that are running Hyper-V that are configured in a failover cluster. For more information, see [Deploy Hyper-V over SMB](#).

In most cases, attached storage should contain multiple, separate disks (logical unit numbers, or LUNs) that are configured at the hardware level. For some clusters, one disk functions as the disk witness (described at the end of this subsection). Other disks contain the files required for the clustered roles (formerly called clustered services or applications). Storage requirements include the following:

- To use the native disk support included in Failover Clustering, use basic disks, not dynamic disks.

- We recommend that you format the partitions with NTFS. If you use Cluster Shared Volumes (CSV), the partition for each of those must be NTFS.

NOTE

If you have a disk witness for your quorum configuration, you can format the disk with either NTFS or Resilient File System (ReFS).

- For the partition style of the disk, you can use either master boot record (MBR) or GUID partition table (GPT).

A disk witness is a disk in the cluster storage that's designated to hold a copy of the cluster configuration database. A failover cluster has a disk witness only if this is specified as part of the quorum configuration. For more information, see [Understanding Quorum in Storage Spaces Direct](#).

Hardware requirements for Hyper-V

If you are creating a failover cluster that includes clustered virtual machines, the cluster servers must support the hardware requirements for the Hyper-V role. Hyper-V requires a 64-bit processor that includes the following:

- **Hardware-assisted virtualization.** This is available in processors that include a virtualization option—specifically processors with Intel Virtualization Technology (Intel VT) or AMD Virtualization (AMD-V) technology.
- **Hardware-enforced Data Execution Prevention (DEP)** must be available and enabled. Specifically, you must enable Intel XD bit (execute disable bit) or AMD NX bit (no execute bit).

For more information about the Hyper-V role, see [Hyper-V Overview](#).

Deploying storage area networks with failover clusters

When deploying a storage area network (SAN) with a failover cluster, follow these guidelines:

- **Confirm compatibility of the storage:** Confirm with manufacturers and vendors that the storage, including drivers, firmware, and software used for the storage, are compatible with failover clusters in the version of Windows Server that you are running.
- **Isolate storage devices, one cluster per device:** Servers from different clusters must not be able to access the same storage devices. In most cases, a LUN used for one set of cluster servers should be isolated from all other servers through LUN masking or zoning.
- **Consider using multipath I/O software or teamed network adapters:** In a highly available storage fabric, you can deploy failover clusters with multiple host bus adapters by using multipath I/O software or network adapter teaming (also called load balancing and failover, or LBFO). This provides the highest level of redundancy and availability. For Windows Server 2012 R2 or Windows Server 2012, your multipath solution must be based on Microsoft Multipath I/O (MPIO). Your hardware vendor will typically supply an MPIO device-specific module (DSM) for your hardware, although Windows Server includes one or more DSMs as part of the operating system.

For more information about LBFO, see [NIC Teaming Overview](#) in the Windows Server Technical Library.

IMPORTANT

Host bus adapters and multipath I/O software can be very version sensitive. If you are implementing a multipath solution for your cluster, work closely with your hardware vendor to choose the correct adapters, firmware, and software for the version of Windows Server that you are running.

- **Consider using Storage Spaces:** If you plan to deploy serial attached SCSI (SAS) clustered storage that's

configured using Storage Spaces, see [Deploy Clustered Storage Spaces](#) for the requirements.

More information

- [Failover Clustering](#)
- [Storage Spaces](#)
- [Using Guest Clustering for High Availability](#)

Use Cluster Shared Volumes in a failover cluster

6/20/2018 • 19 minutes to read • [Edit Online](#)

Applies to: Windows Server 2012 R2, Windows Server 2012, Windows Server 2016

Cluster Shared Volumes (CSV) enable multiple nodes in a failover cluster to simultaneously have read-write access to the same LUN (disk) that is provisioned as an NTFS volume. (In Windows Server 2012 R2, the disk can be provisioned as NTFS or Resilient File System (ReFS).) With CSV, clustered roles can fail over quickly from one node to another node without requiring a change in drive ownership, or dismounting and remounting a volume. CSV also help simplify the management of a potentially large number of LUNs in a failover cluster.

CSV provide a general-purpose, clustered file system, which is layered above NTFS (or ReFS in Windows Server 2012 R2). CSV applications include:

- Clustered virtual hard disk (VHD) files for clustered Hyper-V virtual machines
- Scale-out file shares to store application data for the Scale-Out File Server clustered role. Examples of the application data for this role include Hyper-V virtual machine files and Microsoft SQL Server data. (Be aware that ReFS is not supported for a Scale-Out File Server.) For more information about Scale-Out File Server, see [Scale-Out File Server for Application Data](#).

NOTE

CSV does not support the Microsoft SQL Server clustered workload in SQL Server 2012 and earlier versions of SQL Server.

In Windows Server 2012, CSV functionality was significantly enhanced. For example, dependencies on Active Directory Domain Services were removed. Support was added for the functional improvements in **chkdsk**, for interoperability with antivirus and backup applications, and for integration with general storage features such as BitLocker-encrypted volumes and Storage Spaces. For an overview of CSV functionality that was introduced in Windows Server 2012, see [What's New in Failover Clustering in Windows Server 2012 \[redirected\]](#).

Windows Server 2012 R2 introduces additional functionality, such as distributed CSV ownership, increased resiliency through availability of the Server service, greater flexibility in the amount of physical memory that you can allocate to CSV cache, better diagnosability, and enhanced interoperability that includes support for ReFS and deduplication. For more information, see [What's New in Failover Clustering](#).

NOTE

For information about using data deduplication on CSV for Virtual Desktop Infrastructure (VDI) scenarios, see the blog posts [Deploying Data Deduplication for VDI storage in Windows Server 2012 R2](#) and [Extending Data Deduplication to new workloads in Windows Server 2012 R2](#).

Review requirements and considerations for using CSV in a failover cluster

Before using CSV in a failover cluster, review the network, storage, and other requirements and considerations in this section.

Network configuration considerations

Consider the following when you configure the networks that support CSV.

- **Multiple networks and multiple network adapters.** To enable fault tolerance in the event of a network failure, we recommend that multiple cluster networks carry CSV traffic or that you configure teamed network adapters.

If the cluster nodes are connected to networks that should not be used by the cluster, you should disable them. For example, we recommend that you disable iSCSI networks for cluster use to prevent CSV traffic on those networks. To disable a network, in Failover Cluster Manager, select **Networks**, select the network, select the **Properties** action, and then select **Do not allow cluster network communication on this network**. Alternatively, you can configure the **Role** property of the network by using the [Get-ClusterNetwork](#) Windows PowerShell cmdlet.

- **Network adapter properties.** In the properties for all adapters that carry cluster communication, make sure that the following settings are enabled:
 - **Client for Microsoft Networks** and **File and Printer Sharing for Microsoft Networks.** These settings support Server Message Block (SMB) 3.0, which is used by default to carry CSV traffic between nodes. To enable SMB, also ensure that the Server service and the Workstation service are running and that they are configured to start automatically on each cluster node.

NOTE

In Windows Server 2012 R2, there are multiple Server service instances per failover cluster node. There is the default instance that handles incoming traffic from SMB clients that access regular file shares, and a second CSV instance that handles only inter-node CSV traffic. Also, if the Server service on a node becomes unhealthy, CSV ownership automatically transitions to another node.

SMB 3.0 includes the SMB Multichannel and SMB Direct features, which enable CSV traffic to stream across multiple networks in the cluster and to leverage network adapters that support Remote Direct Memory Access (RDMA). By default, SMB Multichannel is used for CSV traffic. For more information, see [Server Message Block overview](#).

- **Microsoft Failover Cluster Virtual Adapter Performance Filter.** This setting improves the ability of nodes to perform I/O redirection when it is required to reach CSV, for example, when a connectivity failure prevents a node from connecting directly to the CSV disk. For more information, see [About I/O synchronization and I/O redirection in CSV communication](#) later in this topic.
- **Cluster network prioritization.** We generally recommend that you do not change the cluster-configured preferences for the networks.
- **IP subnet configuration.** No specific subnet configuration is required for nodes in a network that use CSV. CSV can support multisubnet clusters.
- **Policy-based Quality of Service (QoS).** We recommend that you configure a QoS priority policy and a minimum bandwidth policy for network traffic to each node when you use CSV. For more information, see [Quality of Service \(QoS\)](#).
- **Storage network.** For storage network recommendations, review the guidelines that are provided by your storage vendor. For additional considerations about storage for CSV, see [Storage and disk configuration requirements](#) later in this topic.

For an overview of the hardware, network, and storage requirements for failover clusters, see [Failover Clustering Hardware Requirements and Storage Options](#).

About I/O synchronization and I/O redirection in CSV communication

- **I/O synchronization:** CSV enables multiple nodes to have simultaneous read-write access to the same shared storage. When a node performs disk input/output (I/O) on a CSV volume, the node communicates directly with the storage, for example, through a storage area network (SAN). However, at any time, a single node (called the coordinator node) "owns" the physical disk resource that is associated with the LUN. The

coordinator node for a CSV volume is displayed in Failover Cluster Manager as **Owner Node** under **Disks**. It also appears in the output of the [Get-ClusterSharedVolume](#) Windows PowerShell cmdlet.

NOTE

In Windows Server 2012 R2, CSV ownership is evenly distributed across the failover cluster nodes based on the number of CSV volumes that each node owns. Additionally, ownership is automatically rebalanced when there are conditions such as CSV failover, a node rejoins the cluster, you add a new node to the cluster, you restart a cluster node, or you start the failover cluster after it has been shut down.

When certain small changes occur in the file system on a CSV volume, this metadata must be synchronized on each of the physical nodes that access the LUN, not only on the single coordinator node. For example, when a virtual machine on a CSV volume is started, created, or deleted, or when a virtual machine is migrated, this information needs to be synchronized on each of the physical nodes that access the virtual machine. These metadata update operations occur in parallel across the cluster networks by using SMB 3.0. These operations do not require all the physical nodes to communicate with the shared storage.

- **I/O redirection:** Storage connectivity failures and certain storage operations can prevent a given node from communicating directly with the storage. To maintain function while the node is not communicating with the storage, the node redirects the disk I/O through a cluster network to the coordinator node where the disk is currently mounted. If the current coordinator node experiences a storage connectivity failure, all disk I/O operations are queued temporarily while a new node is established as a coordinator node.

The server uses one of the following I/O redirection modes, depending on the situation:

- **File system redirection** Redirection is per volume—for example, when CSV snapshots are taken by a backup application when a CSV volume is manually placed in redirected I/O mode.
- **Block redirection** Redirection is at the file-block level—for example, when storage connectivity is lost to a volume. Block redirection is significantly faster than file system redirection.

In Windows Server 2012 R2, you can view the state of a CSV volume on a per node basis. For example, you can see whether I/O is direct or redirected, or whether the CSV volume is unavailable. If a CSV volume is in I/O redirected mode, you can also view the reason. Use the Windows PowerShell cmdlet **Get-ClusterSharedVolumeState** to view this information.

NOTE

- In Windows Server 2012, because of improvements in CSV design, CSV perform more operations in direct I/O mode than occurred in Windows Server 2008 R2.
- Because of the integration of CSV with SMB 3.0 features such as SMB Multichannel and SMB Direct, redirected I/O traffic can stream across multiple cluster networks.
- You should plan your cluster networks to allow for the potential increase in network traffic to the coordinator node during I/O redirection.

Storage and disk configuration requirements

To use CSV, your storage and disks must meet the following requirements:

- **File system format.** In Windows Server 2012 R2, a disk or storage space for a CSV volume must be a basic disk that is partitioned with NTFS or ReFS. In Windows Server 2012, a disk or storage space for a CSV volume must be a basic disk that is partitioned with NTFS.

A CSV has the following additional requirements:

- In Windows Server 2012 R2, you cannot use a disk for a CSV that is formatted with FAT or FAT32.
- In Windows Server 2012, you cannot use a disk for a CSV that is formatted with FAT, FAT32, or ReFS.

- If you want to use a storage space for a CSV, you can configure a simple space or a mirror space. In Windows Server 2012 R2, you can also configure a parity space. (In Windows Server 2012, CSV does not support parity spaces.)
- A CSV cannot be used as a quorum witness disk. For more information about the cluster quorum, see [Understanding Quorum in Storage Spaces Direct](#).
- After you add a disk as a CSV, it is designated in the CSVFS format (for CSV File System). This allows the cluster and other software to differentiate the CSV storage from other NTFS or ReFS storage. Generally, CSVFS supports the same functionality as NTFS or ReFS. However, certain features are not supported. For example, in Windows Server 2012 R2, you cannot enable compression on CSV. In Windows Server 2012, you cannot enable data deduplication or compression on CSV.
- **Resource type in the cluster.** For a CSV volume, you must use the Physical Disk resource type. By default, a disk or storage space that is added to cluster storage is automatically configured in this way.
- **Choice of CSV disks or other disks in cluster storage.** When choosing one or more disks for a clustered virtual machine, consider how each disk will be used. If a disk will be used to store files that are created by Hyper-V, such as VHD files or configuration files, you can choose from the CSV disks or the other available disks in cluster storage. If a disk will be a physical disk that is directly attached to the virtual machine (also called a pass-through disk), you cannot choose a CSV disk, and you must choose from the other available disks in cluster storage.
- **Path name for identifying disks.** Disks in CSV are identified with a path name. Each path appears to be on the system drive of the node as a numbered volume under the **\ClusterStorage** folder. This path is the same when viewed from any node in the cluster. You can rename the volumes if needed.

For storage requirements for CSV, review the guidelines that are provided by your storage vendor. For additional storage planning considerations for CSV, see [Plan to use CSV in a failover cluster](#) later in this topic.

Node requirements

To use CSV, your nodes must meet the following requirements:

- **Drive letter of system disk.** On all nodes, the drive letter for the system disk must be the same.
- **Authentication protocol.** The NTLM protocol must be enabled on all nodes. This is enabled by default.

Plan to use CSV in a failover cluster

This section lists planning considerations and recommendations for using CSV in a failover cluster running Windows Server 2012 R2 or Windows Server 2012.

IMPORTANT

Ask your storage vendor for recommendations about how to configure your specific storage unit for CSV. If the recommendations from the storage vendor differ from information in this topic, use the recommendations from the storage vendor.

Arrangement of LUNs, volumes, and VHD files

To make the best use of CSV to provide storage for clustered virtual machines, it is helpful to review how you would arrange the LUNs (disks) when you configure physical servers. When you configure the corresponding virtual machines, try to arrange the VHD files in a similar way.

Consider a physical server for which you would organize the disks and files as follows:

- System files, including a page file, on one physical disk
- Data files on another physical disk

For an equivalent clustered virtual machine, you should organize the volumes and files in a similar way:

- System files, including a page file, in a VHD file on one CSV
- Data files in a VHD file on another CSV

If you add another virtual machine, where possible, you should keep the same arrangement for the VHDs on that virtual machine.

Number and size of LUNs and volumes

When you plan the storage configuration for a failover cluster that uses CSV, consider the following recommendations:

- To decide how many LUNs to configure, consult your storage vendor. For example, your storage vendor may recommend that you configure each LUN with one partition and place one CSV volume on it.
- There are no limitations for the number of virtual machines that can be supported on a single CSV volume. However, you should consider the number of virtual machines that you plan to have in the cluster and the workload (I/O operations per second) for each virtual machine. Consider the following examples:
 - One organization is deploying virtual machines that will support a virtual desktop infrastructure (VDI), which is a relatively light workload. The cluster uses high-performance storage. The cluster administrator, after consulting with the storage vendor, decides to place a relatively large number of virtual machines per CSV volume.
 - Another organization is deploying a large number of virtual machines that will support a heavily used database application, which is a heavier workload. The cluster uses lower-performing storage. The cluster administrator, after consulting with the storage vendor, decides to place a relatively small number of virtual machines per CSV volume.
- When you plan the storage configuration for a particular virtual machine, consider the disk requirements of the service, application, or role that the virtual machine will support. Understanding these requirements helps you avoid disk contention that can result in poor performance. The storage configuration for the virtual machine should closely resemble the storage configuration that you would use for a physical server that is running the same service, application, or role. For more information, see [Arrangement of LUNs, volumes, and VHD files](#) earlier in this topic.

You can also mitigate disk contention by having storage with a large number of independent physical hard disks. Choose your storage hardware accordingly, and consult with your vendor to optimize the performance of your storage.

- Depending on your cluster workloads and their need for I/O operations, you can consider configuring only a percentage of the virtual machines to access each LUN, while other virtual machines do not have connectivity and are instead dedicated to compute operations.

Add a disk to CSV on a failover cluster

The CSV feature is enabled by default in Failover Clustering. To add a disk to CSV, you must add a disk to the **Available Storage** group of the cluster (if it is not already added), and then add the disk to CSV on the cluster. You can use Failover Cluster Manager or the Failover Clusters Windows PowerShell cmdlets to perform these procedures.

Add a disk to Available Storage

1. In Failover Cluster Manager, in the console tree, expand the name of the cluster, and then expand **Storage**.
2. Right-click **Disks**, and then select **Add Disk**. A list appears showing the disks that can be added for use in a failover cluster.
3. Select the disk or disks you want to add, and then select **OK**.

The disks are now assigned to the **Available Storage** group.

Windows PowerShell equivalent commands (add a disk to Available Storage)

The following Windows PowerShell cmdlet or cmdlets perform the same function as the preceding procedure. Enter each cmdlet on a single line, even though they may appear word-wrapped across several lines here because of formatting constraints.

The following example identifies the disks that are ready to be added to the cluster, and then adds them to the **Available Storage** group.

```
Get-ClusterAvailableDisk | Add-ClusterDisk
```

Add a disk in Available Storage to CSV

1. In Failover Cluster Manager, in the console tree, expand the name of the cluster, expand **Storage**, and then select **Disks**.
2. Select one or more disks that are assigned to **Available Storage**, right-click the selection, and then select **Add to Cluster Shared Volumes**.

The disks are now assigned to the **Cluster Shared Volume** group in the cluster. The disks are exposed to each cluster node as numbered volumes (mount points) under the %SystemDisk%ClusterStorage folder. The volumes appear in the CSVFS file system.

NOTE

You can rename CSV volumes in the %SystemDisk%ClusterStorage folder.

Windows PowerShell equivalent commands (add a disk to CSV)

The following Windows PowerShell cmdlet or cmdlets perform the same function as the preceding procedure. Enter each cmdlet on a single line, even though they may appear word-wrapped across several lines here because of formatting constraints.

The following example adds *Cluster Disk 1* in **Available Storage** to CSV on the local cluster.

```
Add-ClusterSharedVolume -Name "Cluster Disk 1"
```

Enable the CSV cache for read-intensive workloads (optional)

The CSV cache provides caching at the block level of read-only unbuffered I/O operations by allocating system memory (RAM) as a write-through cache. (Unbuffered I/O operations are not cached by the cache manager.) This can improve performance for applications such as Hyper-V, which conducts unbuffered I/O operations when accessing a VHD. The CSV cache can boost the performance of read requests without caching write requests. Enabling the CSV cache is also useful for Scale-Out File Server scenarios.

NOTE

We recommend that you enable the CSV cache for all clustered Hyper-V and Scale-Out File Server deployments.

By default in Windows Server 2012, the CSV cache is disabled. In Windows Server 2012 R2, the CSV cache is enabled by default. However, you must still allocate the size of the block cache to reserve.

The following table describes the two configuration settings that control the CSV cache.

PROPERTY NAME IN WINDOWS SERVER 2012 R2	PROPERTY NAME IN WINDOWS SERVER 2012	DESCRIPTION
BlockCacheSize	SharedVolumeBlockCacheSizeInMB	This is a cluster common property that allows you to define how much memory (in megabytes) to reserve for the CSV cache on each node in the cluster. For example, if a value of 512 is defined, then 512 MB of system memory is reserved on each node. (In many clusters, 512 MB is a recommended value.) The default setting is 0 (for disabled).
EnableBlockCache	CsvEnableBlockCache	This is a private property of the cluster Physical Disk resource. It allows you to enable CSV cache on an individual disk that is added to CSV. In Windows Server 2012, the default setting is 0 (for disabled). To enable CSV cache on a disk, configure a value of 1. By default, in Windows Server 2012 R2, this setting is enabled.

You can monitor the CSV cache in Performance Monitor by adding the counters under **Cluster CSV Volume Cache**.

Configure the CSV cache

1. Start Windows PowerShell as an administrator.
2. To define a cache of 512 MB to be reserved on each node, type the following:

- For Windows Server 2012 R2:

```
(Get-Cluster).BlockCacheSize = 512
```

- For Windows Server 2012:

```
(Get-Cluster).SharedVolumeBlockCacheSizeInMB = 512
```

3. In Windows Server 2012, to enable the CSV cache on a CSV named *Cluster Disk 1*, enter the following:

```
Get-ClusterSharedVolume "Cluster Disk 1" | Set-ClusterParameter CsvEnableBlockCache 1
```

NOTE

- In Windows Server 2012, you can allocate only 20% of the total physical RAM to the CSV cache. In Windows Server 2012 R2, you can allocate up to 80%. Because Scale-Out File Servers are not typically memory constrained, you can accomplish large performance gains by using the extra memory for the CSV cache.
- To avoid resource contention, you should restart each node in the cluster after you modify the memory that is allocated to the CSV cache. In Windows Server 2012 R2, a restart is no longer required.
- After you enable or disable CSV cache on an individual disk, for the setting to take effect, you must take the Physical Disk resource offline and bring it back online. (By default, in Windows Server 2012 R2, the CSV cache is enabled.)
- For more information about CSV cache that includes information about performance counters, see the blog post [How to Enable CSV Cache](#).

Back up CSV

There are multiple methods to back up information that is stored on CSV in a failover cluster. You can use a Microsoft backup application or a non-Microsoft application. In general, CSV do not impose special backup requirements beyond those for clustered storage formatted with NTFS or ReFS. CSV backups also do not disrupt other CSV storage operations.

You should consider the following factors when you select a backup application and backup schedule for CSV:

- Volume-level backup of a CSV volume can be run from any node that connects to the CSV volume.
- Your backup application can use software snapshots or hardware snapshots. Depending on the ability of your backup application to support them, backups can use application-consistent and crash-consistent Volume Shadow Copy Service (VSS) snapshots.
- If you are backing up CSV that have multiple running virtual machines, you should generally choose a management operating system-based backup method. If your backup application supports it, multiple virtual machines can be backed up simultaneously.
- CSV support backup requestors that are running Windows Server 2012 R2 Backup, Windows Server 2012 Backup or Windows Server 2008 R2 Backup. However, Windows Server Backup generally provides only a basic backup solution that may not be suited for organizations with larger clusters. Windows Server Backup does not support application-consistent virtual machine backup on CSV. It supports crash-consistent volume-level backup only. (If you restore a crash-consistent backup, the virtual machine will be in the same state it was if the virtual machine had crashed at the exact moment that the backup was taken.) A backup of a virtual machine on a CSV volume will succeed, but an error event will be logged indicating that this is not supported.
- You may require administrative credentials when backing up a failover cluster.

IMPORTANT

Be sure to carefully review what data your backup application backs up and restores, which CSV features it supports, and the resource requirements for the application on each cluster node.

WARNING

If you need to restore the backup data onto a CSV volume, be aware of the capabilities and limitations of the backup application to maintain and restore application-consistent data across the cluster nodes. For example, with some applications, if the CSV is restored on a node that is different from the node where the CSV volume was backed up, you might inadvertently overwrite important data about the application state on the node where the restore is taking place.

More information

- [Failover Clustering](#)
- [Deploy Clustered Storage Spaces](#)

Create a failover cluster

6/25/2018 • 12 minutes to read • [Edit Online](#)

Applies to: Windows Server 2012 R2, Windows Server 2012, Windows Server 2016

This topic shows how to create a failover cluster by using either the Failover Cluster Manager snap-in or Windows PowerShell. The topic covers a typical deployment, where computer objects for the cluster and its associated clustered roles are created in Active Directory Domain Services (AD DS).

NOTE

In Windows Server 2012 R2 you can also deploy an Active Directory-detached cluster. This deployment method enables you to create a failover cluster without permissions to create computer objects in AD DS or the need to request that computer objects are prestaged in AD DS. This option is only available through Windows PowerShell, and is only recommended for specific scenarios. For more information, see [Deploy an Active Directory-Detached Cluster](#).

Checklist: Create a failover cluster

STATUS	TASK	REFERENCE
<input type="checkbox"/>	Verify the prerequisites	Verify the prerequisites
<input type="checkbox"/>	Install the Failover Clustering feature on every server that you want to add as a cluster node	Install the Failover Clustering feature
<input type="checkbox"/>	Run the Cluster Validation Wizard to validate the configuration	Validate the configuration
<input type="checkbox"/>	Run the Create Cluster Wizard to create the failover cluster	Create the failover cluster
<input type="checkbox"/>	Create clustered roles to host cluster workloads	Create clustered roles

Verify the prerequisites

Before you begin, verify the following prerequisites:

- Make sure that all servers that you want to add as cluster nodes are running the same version of Windows Server.

NOTE

You can use the Failover Clustering feature on all editions of Windows Server 2012 R2 and Windows Server 2012. This includes Server Core installations.

- Review the hardware requirements to make sure that your configuration is supported. For more information, see [Failover Clustering Hardware Requirements and Storage Options](#).
- If you want to add clustered storage during cluster creation, make sure that all servers can access the storage.

(You can also add clustered storage after you create the cluster.)

- Make sure that all servers that you want to add as cluster nodes are joined to the same Active Directory domain.
- (Optional) Create an organizational unit (OU) and move the computer accounts for the servers that you want to add as cluster nodes into the OU. As a best practice, we recommend that you place failover clusters in their own OU in AD DS. This can help you better control which Group Policy settings or security template settings affect the cluster nodes. By isolating clusters in their own OU, it also helps prevent against accidental deletion of cluster computer objects.

Additionally, verify the following account requirements:

- Make sure that the account you want to use to create the cluster is a domain user who has administrator rights on all servers that you want to add as cluster nodes.
- Make sure that either of the following is true:
 - The user who creates the cluster has the **Create Computer objects** permission to the OU or the container where the servers that will form the cluster reside.
 - If the user does not have the **Create Computer objects** permission, ask a domain administrator to prestage a cluster computer object for the cluster. For more information, see [Prestage Cluster Computer Objects in Active Directory Domain Services](#).

NOTE

This requirement does not apply if you want to create an Active Directory-detached cluster in Windows Server 2012 R2. For more information, see [Deploy an Active Directory-Detached Cluster](#).

Install the Failover Clustering feature

You must install the Failover Clustering feature on every server that you want to add as a failover cluster node.

Install the Failover Clustering feature

1. Start Server Manager.
2. On the **Manage** menu, select **Add Roles and Features**.
3. On the **Before you begin** page, select **Next**.
4. On the **Select installation type** page, select **Role-based or feature-based installation**, and then select **Next**.
5. On the **Select destination server** page, select the server where you want to install the feature, and then select **Next**.
6. On the **Select server roles** page, select **Next**.
7. On the **Select features** page, select the **Failover Clustering** check box.
8. To install the failover cluster management tools, select **Add Features**, and then select **Next**.
9. On the **Confirm installation selections** page, select **Install**.

NOTE

A server restart is not required for the Failover Clustering feature.

10. When the installation is completed, select **Close**.
11. Repeat this procedure on every server that you want to add as a failover cluster node.

NOTE

After you install the Failover Clustering feature, we recommend that you apply the latest updates from Windows Update. Also, for a Windows Server 2012-based failover cluster, review the [Recommended hotfixes and updates for Windows Server 2012-based failover clusters](#) Microsoft Support article and install any updates that apply.

Validate the configuration

Before you create the failover cluster, we strongly recommend that you validate the configuration to make sure that the hardware and hardware settings are compatible with failover clustering. Microsoft supports a cluster solution only if the complete configuration passes all validation tests and if all hardware is certified for the version of Windows Server that the cluster nodes are running.

NOTE

You must have at least two nodes to run all tests. If you have only one node, many of the critical storage tests do not run.

Run cluster validation tests

1. On a computer that has the Failover Cluster Management Tools installed from the Remote Server Administration Tools, or on a server where you installed the Failover Clustering feature, start Failover Cluster Manager. To do this on a server, start Server Manager, and then on the **Tools** menu, select **Failover Cluster Manager**.
2. In the **Failover Cluster Manager** pane, under **Management**, select **Validate Configuration**.
3. On the **Before You Begin** page, select **Next**.
4. On the **Select Servers or a Cluster** page, in the **Enter name** box, enter the NetBIOS name or the fully qualified domain name of a server that you plan to add as a failover cluster node, and then select **Add**. Repeat this step for each server that you want to add. To add multiple servers at the same time, separate the names by a comma or by a semicolon. For example, enter the names in the format *server1.contoso.com*, *server2.contoso.com*. When you are finished, select **Next**.
5. On the **Testing Options** page, select **Run all tests (recommended)**, and then select **Next**.
6. On the **Confirmation** page, select **Next**.

The Validating page displays the status of the running tests.

7. On the **Summary** page, do either of the following:
 - If the results indicate that the tests completed successfully and the configuration is suited for clustering, and you want to create the cluster immediately, make sure that the **Create the cluster now using the validated nodes** check box is selected, and then select **Finish**. Then, continue to step 4 of the [Create the failover cluster](#) procedure.
 - If the results indicate that there were warnings or failures, select **View Report** to view the details and determine which issues must be corrected. Realize that a warning for a particular validation test indicates that this aspect of the failover cluster can be supported, but might not meet the recommended best practices.

NOTE

If you receive a warning for the Validate Storage Spaces Persistent Reservation test, see the blog post [Windows Failover Cluster validation warning indicates your disks don't support the persistent reservations for Storage Spaces](#) for more information.

For more information about hardware validation tests, see [Validate Hardware for a Failover Cluster](#).

Create the failover cluster

To complete this step, make sure that the user account that you log on as meets the requirements that are outlined in the [Verify the prerequisites](#) section of this topic.

1. Start Server Manager.
2. On the **Tools** menu, select **Failover Cluster Manager**.
3. In the **Failover Cluster Manager** pane, under **Management**, select **Create Cluster**.

The Create Cluster Wizard opens.

4. On the **Before You Begin** page, select **Next**.
5. If the **Select Servers** page appears, in the **Enter name** box, enter the NetBIOS name or the fully qualified domain name of a server that you plan to add as a failover cluster node, and then select **Add**. Repeat this step for each server that you want to add. To add multiple servers at the same time, separate the names by a comma or a semicolon. For example, enter the names in the format *server1.contoso.com*; *server2.contoso.com*. When you are finished, select **Next**.

NOTE

If you chose to create the cluster immediately after running validation in the [configuration validating procedure](#), you will not see the **Select Servers** page. The nodes that were validated are automatically added to the Create Cluster Wizard so that you do not have to enter them again.

6. If you skipped validation earlier, the **Validation Warning** page appears. We strongly recommend that you run cluster validation. Only clusters that pass all validation tests are supported by Microsoft. To run the validation tests, select **Yes**, and then select **Next**. Complete the Validate a Configuration Wizard as described in [Validate the configuration](#).
7. On the **Access Point for Administering the Cluster** page, do the following:
 - a. In the **Cluster Name** box, enter the name that you want to use to administer the cluster. Before you do, review the following information:
 - During cluster creation, this name is registered as the cluster computer object (also known as the *cluster name object* or *CNO*) in AD DS. If you specify a NetBIOS name for the cluster, the CNO is created in the same location where the computer objects for the cluster nodes reside. This can be either the default Computers container or an OU.
 - To specify a different location for the CNO, you can enter the distinguished name of an OU in the **Cluster Name** box. For example: *CN=ClusterName, OU=Clusters, DC=Contoso, DC=com*.
 - If a domain administrator has prestaged the CNO in a different OU than where the cluster nodes reside, specify the distinguished name that the domain administrator provides.
 - b. If the server does not have a network adapter that is configured to use DHCP, you must configure one or more static IP addresses for the failover cluster. Select the check box next to each network that you want to use for cluster management. Select the **Address** field next to a selected network, and then enter the IP address that you want to assign to the cluster. This IP address (or addresses) will be associated with the cluster name in Domain Name System (DNS).
 - c. When you are finished, select **Next**.
8. On the **Confirmation** page, review the settings. By default, the **Add all eligible storage to the cluster** check box is selected. Clear this check box if you want to do either of the following:
 - You want to configure storage later.
 - You plan to create clustered storage spaces through Failover Cluster Manager or through the Failover Clustering Windows PowerShell cmdlets, and have not yet created storage spaces in File and Storage

Services. For more information, see [Deploy Clustered Storage Spaces](#).

9. Select **Next** to create the failover cluster.
10. On the **Summary** page, confirm that the failover cluster was successfully created. If there were any warnings or errors, view the summary output or select **View Report** to view the full report. Select **Finish**.
11. To confirm that the cluster was created, verify that the cluster name is listed under **Failover Cluster Manager** in the navigation tree. You can expand the cluster name, and then select items under **Nodes**, **Storage** or **Networks** to view the associated resources.

Realize that it may take some time for the cluster name to successfully replicate in DNS. After successful DNS registration and replication, if you select **All Servers** in Server Manager, the cluster name should be listed as a server with a **Manageability** status of **Online**.

After the cluster is created, you can do things such as verify cluster quorum configuration, and optionally, create Cluster Shared Volumes (CSV). For more information, see [Understanding Quorum in Storage Spaces Direct](#) and [Use Cluster Shared Volumes in a failover cluster](#).

Create clustered roles

After you create the failover cluster, you can create clustered roles to host cluster workloads.

NOTE

For clustered roles that require a client access point, a virtual computer object (VCO) is created in AD DS. By default, all VCOs for the cluster are created in the same container or OU as the CNO. Realize that after you create a cluster, you can move the CNO to any OU.

Here's how to create a clustered role:

1. Use Server Manager or Windows PowerShell to install the role or feature that is required for a clustered role on each failover cluster node. For example, if you want to create a clustered file server, install the File Server role on all cluster nodes.

The following table shows the clustered roles that you can configure in the High Availability Wizard and the associated server role or feature that you must install as a prerequisite.

CLUSTERED ROLE	ROLE OR FEATURE PREREQUISITE
DFS Namespace Server	DFS Namespaces (part of File Server role)
DHCP Server	DHCP Server role
Distributed Transaction Coordinator (DTC)	None
File Server	File Server role
Generic Application	Not applicable
Generic Script	Not applicable
Generic Service	Not applicable
Hyper-V Replica Broker	Hyper-V role

CLUSTERED ROLE	ROLE OR FEATURE PREREQUISITE
iSCSI Target Server	iSCSI Target Server (part of File Server role)
iSNS Server	iSNS Server Service feature
Message Queuing	Message Queuing Services feature
Other Server	None
Virtual Machine	Hyper-V role
WINS Server	WINS Server feature

2. In Failover Cluster Manager, expand the cluster name, right-click **Roles**, and then select **Configure Role**.
3. Follow the steps in the High Availability Wizard to create the clustered role.
4. To verify that the clustered role was created, in the **Roles** pane, make sure that the role has a status of **Running**. The Roles pane also indicates the owner node. To test failover, right-click the role, point to **Move**, and then select **Select Node**. In the **Move Clustered Role** dialog box, select the desired cluster node, and then select **OK**. In the **Owner Node** column, verify that the owner node changed.

Create a failover cluster by using Windows PowerShell

The following Windows PowerShell cmdlets perform the same functions as the preceding procedures in this topic. Enter each cmdlet on a single line, even though they may appear word-wrapped across several lines because of formatting constraints.

NOTE

You must use Windows PowerShell to create an Active Directory-detached cluster in Windows Server 2012 R2. For information about the syntax, see [Deploy an Active Directory-Detached Cluster](#).

The following example installs the Failover Clustering feature.

```
Install-WindowsFeature -Name Failover-Clustering -IncludeManagementTools
```

The following example runs all cluster validation tests on computers that are named *Server1* and *Server2*.

```
Test-Cluster -Node Server1, Server2
```

NOTE

The **Test-Cluster** cmdlet outputs the results to a log file in the current working directory. For example:
C:\Users<username>\AppData\Local\Temp.

The following example creates a failover cluster that is named *MyCluster* with nodes *Server1* and *Server2*, assigns the static IP address *192.168.1.12*, and adds all eligible storage to the failover cluster.

```
New-Cluster -Name MyCluster -Node Server1, Server2 -StaticAddress 192.168.1.12
```

The following example creates the same failover cluster as in the previous example, but it does not add eligible storage to the failover cluster.

```
New-Cluster -Name MyCluster -Node Server1, Server2 -StaticAddress 192.168.1.12 -NoStorage
```

The following example creates a cluster that is named *MyCluster* in the *Cluster* OU of the domain *Contoso.com*.

```
New-Cluster -Name CN=MyCluster,OU=Cluster,DC=Contoso,DC=com -Node Server1, Server2
```

For examples of how to add clustered roles, see topics such as [Add-ClusterFileServerRole](#) and [Add-ClusterGenericApplicationRole](#).

More information

- [Failover Clustering](#)
- [Deploy a Hyper-V Cluster](#)
- [Scale-Out File Server for Application Data](#)
- [Deploy an Active Directory-Detached Cluster](#)
- [Using Guest Clustering for High Availability](#)
- [Cluster-Aware Updating](#)
- [New-Cluster](#)
- [Test-Cluster](#)

Prestage cluster computer objects in Active Directory Domain Services

6/20/2018 • 8 minutes to read • [Edit Online](#)

Applies to: Windows Server 2012 R2, Windows Server 2012, Windows Server 2016

This topic shows how to prestage cluster computer objects in Active Directory Domain Services (AD DS). You can use this procedure to enable a user or group to create a failover cluster when they do not have permissions to create computer objects in AD DS.

When you create a failover cluster by using the Create Cluster Wizard or by using Windows PowerShell, you must specify a name for the cluster. If you have sufficient permissions when you create the cluster, the cluster creation process automatically creates a computer object in AD DS that matches the cluster name. This object is called the *cluster name object* or CNO. Through the CNO, virtual computer objects (VCOs) are automatically created when you configure clustered roles that use client access points. For example, if you create a highly available file server with a client access point that is named *FileServer1*, the CNO will create a corresponding VCO in AD DS.

NOTE

In Windows Server 2012 R2, there is the option to create an Active Directory-detached cluster, where no CNO or VCOs are created in AD DS. This is targeted for specific types of cluster deployments. For more information, see [Deploy an Active Directory-Detached Cluster](#).

To create the CNO automatically, the user who creates the failover cluster must have the **Create Computer objects** permission to the organizational unit (OU) or the container where the servers that will form the cluster reside. To enable a user or group to create a cluster without having this permission, a user with appropriate permissions in AD DS (typically a domain administrator) can prestage the CNO in AD DS. This also provides the domain administrator more control over the naming convention that is used for the cluster, and control over which OU the cluster objects are created in.

Step 1: Prestage the CNO in AD DS

Before you begin, make sure that you know the following:

- The name that you want to assign the cluster
- The name of the user account or group to which you want to grant rights to create the cluster

As a best practice, we recommend that you create an OU for the cluster objects. If an OU already exists that you want to use, membership in the **Account Operators** group is the minimum required to complete this step. If you need to create an OU for the cluster objects, membership in the **Domain Admins** group, or equivalent, is the minimum required to complete this step.

NOTE

If you create the CNO in the default Computers container instead of an OU, you do not have to complete Step 3 of this topic. In this scenario, a cluster administrator can create up to 10 VCOs without any additional configuration.

Prestage the CNO in AD DS

1. On a computer that has the AD DS Tools installed from the Remote Server Administration Tools, or on a domain controller, open **Active Directory Users and Computers**. To do this on a server, start Server Manager, and then on the **Tools** menu, select **Active Directory Users and Computers**.
2. To create an OU for the cluster computer objects, right-click the domain name or an existing OU, point to **New**, and then select **Organizational Unit**. In the **Name** box, enter the name of the OU, and then select **OK**.
3. In the console tree, right-click the OU where you want to create the CNO, point to **New**, and then select **Computer**.
4. In the **Computer name** box, enter the name that will be used for the failover cluster, and then select **OK**.

NOTE

This is the cluster name that the user who creates the cluster will specify on the **Access Point for Administering the Cluster** page in the Create Cluster wizard or as the value of the `-Name` parameter for the **New-Cluster** Windows PowerShell cmdlet.

5. As a best practice, right-click the computer account that you just created, select **Properties**, and then select the **Object** tab. On the **Object** tab, select the **Protect object from accidental deletion** check box, and then select **OK**.
6. Right-click the computer account that you just created, and then select **Disable Account**. Select **Yes** to confirm, and then select **OK**.

NOTE

You must disable the account so that during cluster creation, the cluster creation process can confirm that the account is not currently in use by an existing computer or cluster in the domain.

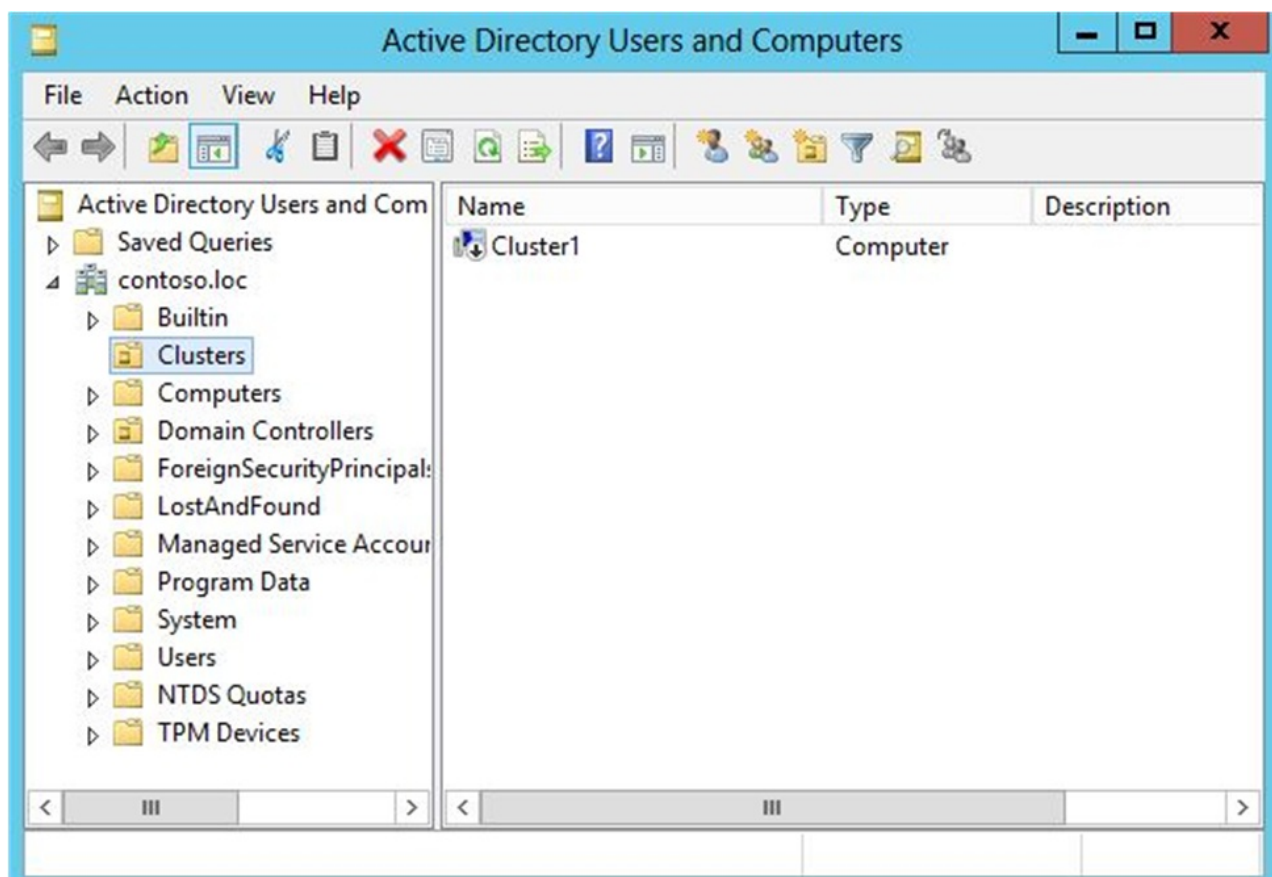


Figure 1. Disabled CNO in the example Clusters OU

Step 2: Grant the user permissions to create the cluster

You must configure permissions so that the user account that will be used to create the failover cluster has Full Control permissions to the CNO.

Membership in the **Account Operators** group is the minimum required to complete this step.

Here's how to grant the user permissions to create the cluster:

1. In Active Directory Users and Computers, on the **View** menu, make sure that **Advanced Features** is selected.
2. Locate and then right-click the CNO, and then select **Properties**.
3. On the **Security** tab, select **Add**.
4. In the **Select Users, Computers, or Groups** dialog box, specify the user account or group that you want to grant permissions to, and then select **OK**.
5. Select the user account or group that you just added, and then next to **Full control**, select the **Allow** check box.

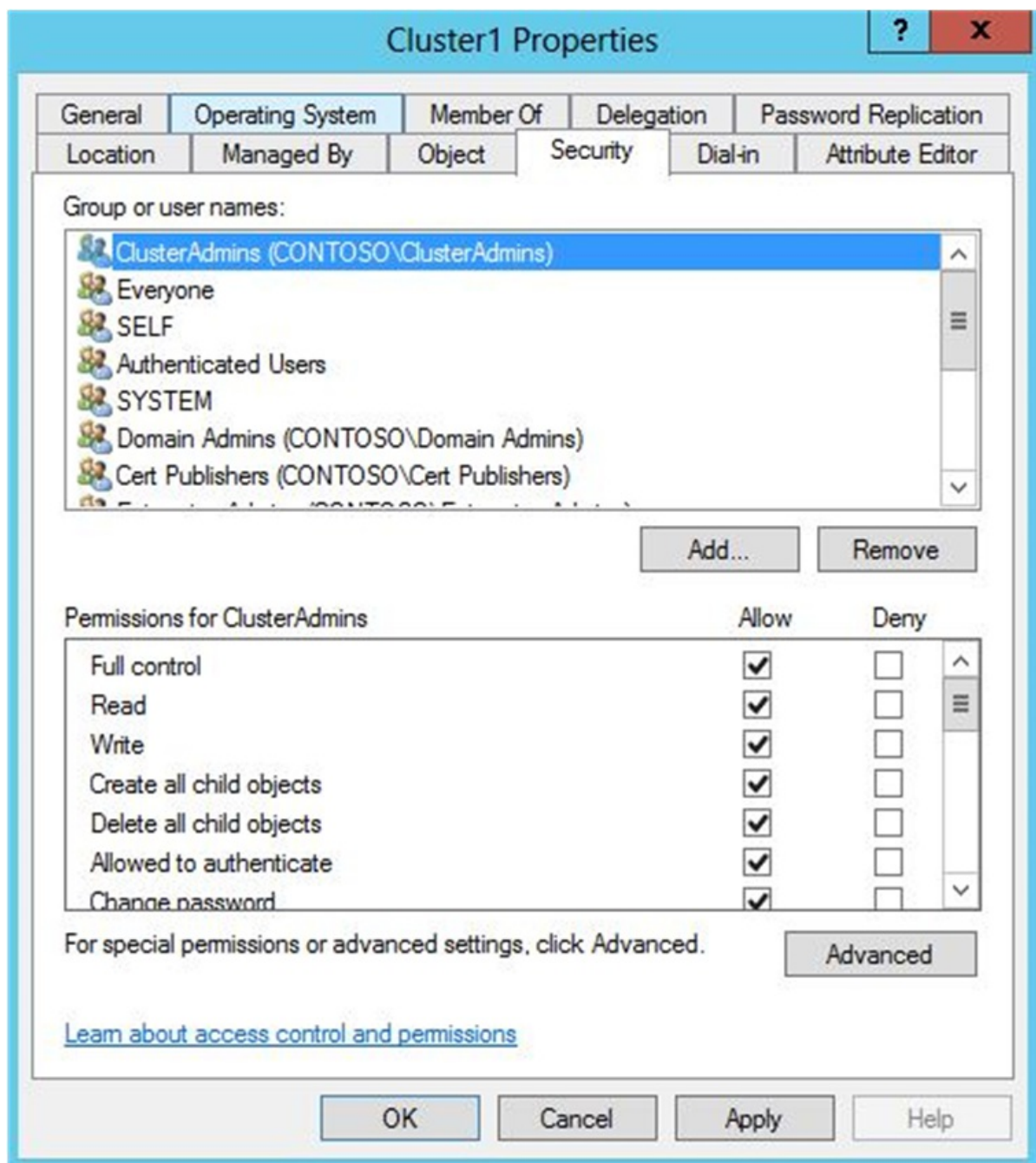


Figure 2. Granting Full Control to the user or group that will create the cluster

6. Select **OK**.

After you complete this step, the user who you granted permissions to can create the failover cluster. However, if the CNO is located in an OU, the user cannot create clustered roles that require a client access point until you complete Step 3.

NOTE

If the CNO is in the default Computers container, a cluster administrator can create up to 10 VCOs without any additional configuration. To add more than 10 VCOs, you must explicitly grant the **Create Computer objects** permission to the CNO for the Computers container.

Step 3: Grant the CNO permissions to the OU or prestage VCOs for clustered roles

When you create a clustered role with a client access point, the cluster creates a VCO in the same OU as the CNO. For this to occur automatically, the CNO must have permissions to create computer objects in the OU.

If you prestaged the CNO in AD DS, you can do either of the following to create VCOs:

- Option 1: [Grant the CNO permissions to the OU](#). If you use this option, the cluster can automatically create VCOs in AD DS. Therefore, an administrator for the failover cluster can create clustered roles without having to request that you prestage VCOs in AD DS.

NOTE

Membership in the **Domain Admins** group, or equivalent, is the minimum required to complete the steps for this option.

- Option 2: [Prestage a VCO for a clustered role](#). Use this option if it is necessary to prestage accounts for clustered roles because of requirements in your organization. For example, you may want to control the naming convention, or control which clustered roles are created.

NOTE

Membership in the **Account Operators** group is the minimum required to complete the steps for this option.

Grant the CNO permissions to the OU

1. In Active Directory Users and Computers, on the **View** menu, make sure that **Advanced Features** is selected.
2. Right-click the OU where you created the CNO in [Step 1: Prestage the CNO in AD DS](#), and then select **Properties**.
3. On the **Security** tab, select **Advanced**.
4. In the **Advanced Security Settings** dialog box, select **Add**.
5. Next to **Principal**, select **Select a principal**.
6. In the **Select User, Computer, Service Account, or Groups** dialog box, select **Object Types**, select the **Computers** check box, and then select **OK**.
7. Under **Enter the object names to select**, enter the name of the CNO, select **Check Names**, and then select **OK**. In response to the warning message that says that you are about to add a disabled object, select **OK**.
8. In the **Permission Entry** dialog box, make sure that the **Type** list is set to **Allow**, and the **Applies to** list is set to **This object and all descendant objects**.
9. Under **Permissions**, select the **Create Computer objects** check box.

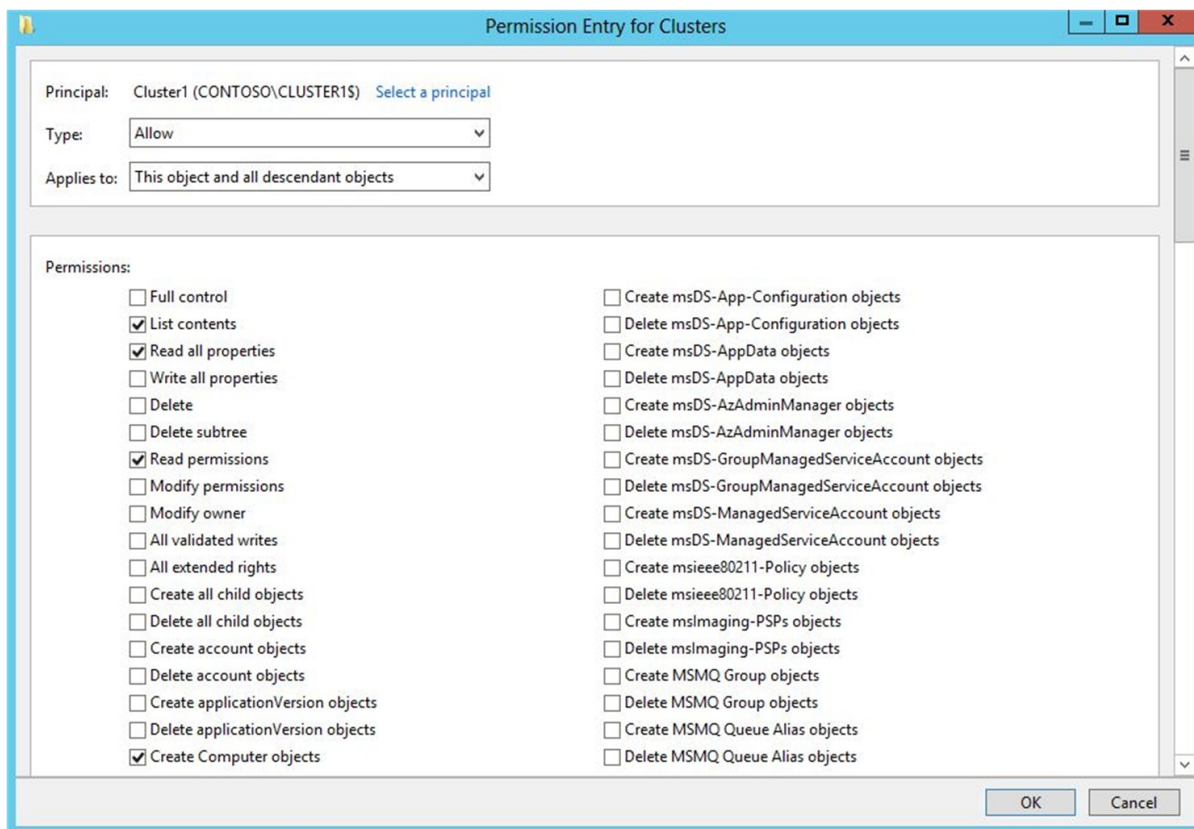


Figure 3. Granting the Create Computer objects permission to the CNO

10. Select **OK** until you return to the Active Directory Users and Computers snap-in.

An administrator on the failover cluster can now create clustered roles with client access points, and bring the resources online.

Prestage a VCO for a clustered role

1. Before you begin, make sure that you know the name of the cluster and the name that the clustered role will have.
2. In Active Directory Users and Computers, on the **View** menu, make sure that **Advanced Features** is selected.
3. In Active Directory Users and Computers, right-click the OU where the CNO for the cluster resides, point to **New**, and then select **Computer**.
4. In the **Computer name** box, enter the name that you will use for the clustered role, and then select **OK**.
5. As a best practice, right-click the computer account that you just created, select **Properties**, and then select the **Object** tab. On the **Object** tab, select the **Protect object from accidental deletion** check box, and then select **OK**.
6. Right-click the computer account that you just created, and then select **Properties**.
7. On the **Security** tab, select **Add**.
8. In the **Select User, Computer, Service Account, or Groups** dialog box, select **Object Types**, select the **Computers** check box, and then select **OK**.
9. Under **Enter the object names to select**, enter the name of the CNO, select **Check Names**, and then select **OK**. If you receive a warning message that says that you are about to add a disabled object, select **OK**.
10. Make sure that the CNO is selected, and then next to **Full control**, select the **Allow** check box.
11. Select **OK**.

An administrator on the failover cluster can now create the clustered role with a client access point that matches the prestaged VCO name, and bring the resource online.

More information

- [Failover Clustering](#)

Deploy a Cloud Witness for a Failover Cluster

4/30/2018 • 8 minutes to read • [Edit Online](#)

Applies to: Applies to: Windows Server (Semi-Annual Channel), Windows Server 2016

Cloud Witness is a new type of Failover Cluster quorum witness being introduced in Windows Server 2016. This topic provides an overview of the Cloud Witness feature, the scenarios that it supports, and instructions about how to configure a cloud witness for a Failover Cluster that is running Windows Server 2016.

Cloud Witness overview

Figure 1 illustrates a multi-site stretched Failover Cluster quorum configuration with Windows Server 2016. In this example configuration (figure 1), there are 2 nodes in 2 datacenters (referred to as Sites). Note, it is possible for a cluster to span more than 2 datacenters. Also, each datacenter can have more than 2 nodes. A typical cluster quorum configuration in this setup (automatic failover SLA) gives each node a vote. One extra vote is given to the quorum witness to allow cluster to keep running even if either one of the datacenter experiences a power outage. The math is simple - there are 5 total votes and you need 3 votes for the cluster to keep it running.

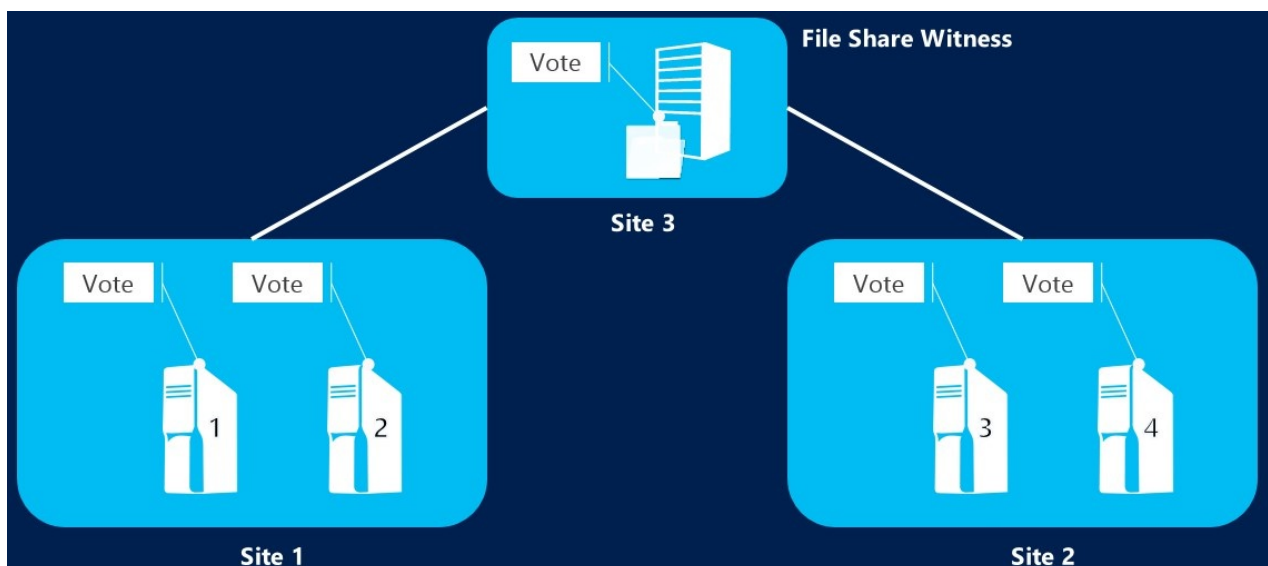


Figure 1: Using a File Share Witness as a quorum witness

In case of power outage in one datacenter, to give equal opportunity for the cluster in other datacenter to keep it running, it is recommended to host the quorum witness in a location other than the two datacenters. This typically means requiring a third separate datacenter (site) to host a File Server that is backing the File Share which is used as the quorum witness (File Share Witness).

Most organizations do not have a third separate datacenter that will host File Server backing the File Share Witness. This means organizations primarily host the File Server in one of the two datacenters, which by extension, makes that datacenter the primary datacenter. In a scenario where there is power outage in the primary datacenter, the cluster would go down as the other datacenter would only have 2 votes which is below the quorum majority of 3 votes needed. For the customers that have third separate datacenter to host the File Server, it is an overhead to maintain the highly available File Server backing the File Share Witness. Hosting virtual machines in the public cloud that have the File Server for File Share Witness running in Guest OS is a significant overhead in terms of both setup & maintenance.

Cloud Witness is a new type of Failover Cluster quorum witness that leverages Microsoft Azure as the arbitration point (figure 2). It uses Azure Blob Storage to read/write a blob file which is then used as an arbitration point in

case of split-brain resolution.

There are significant benefits which this approach:

1. Leverages Microsoft Azure (no need for third separate datacenter).
2. Uses standard available Azure Blob Storage (no extra maintenance overhead of virtual machines hosted in public cloud).
3. Same Azure Storage Account can be used for multiple clusters (one blob file per cluster; cluster unique id used as blob file name).
4. Very low on-going \$cost to the Storage Account (very small data written per blob file, blob file updated only once when cluster nodes' state changes).
5. Built-in Cloud Witness resource type.

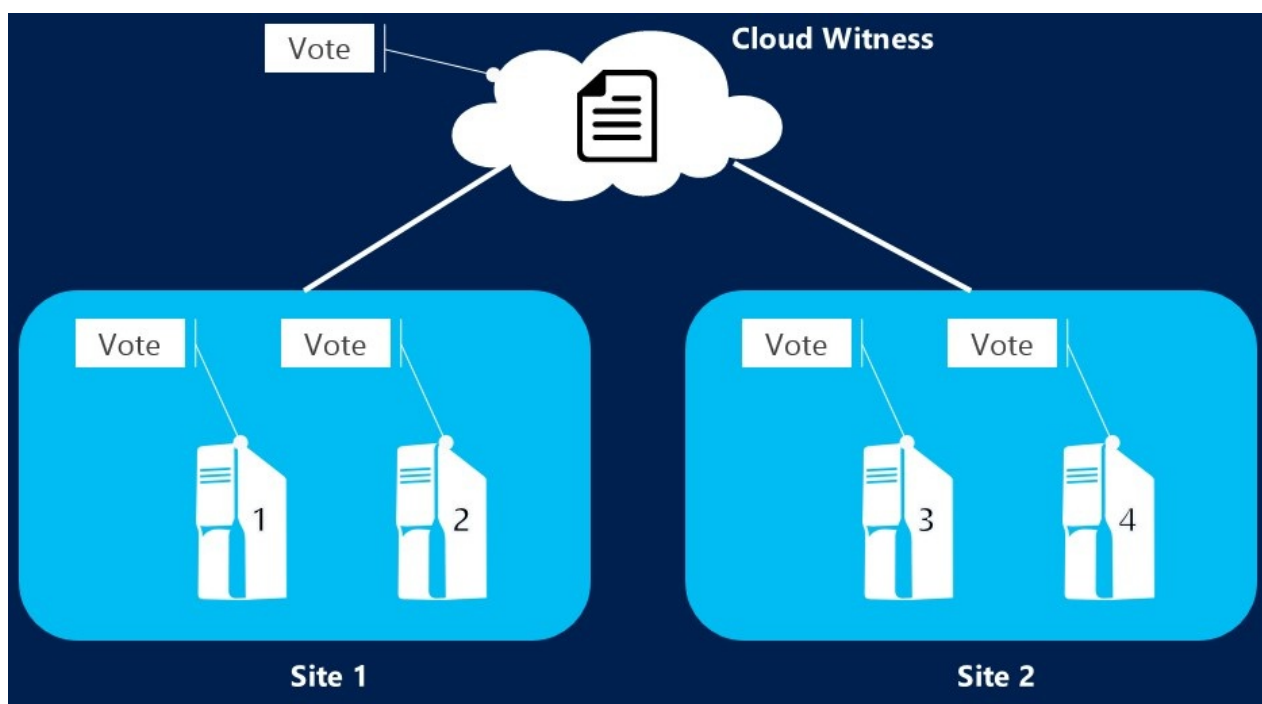


Figure 2: Multi-site stretched clusters with Cloud Witness as a quorum witness

As shown in figure 2, there is no third separate site that is required. Cloud Witness, like any other quorum witness, gets a vote and can participate in quorum calculations.

Cloud Witness: Supported scenarios for single witness type

If you have a Failover Cluster deployment, where all nodes can reach the internet (by extension of Azure), it is recommended that you configure a Cloud Witness as your quorum witness resource.

Some of the scenarios that are supported use of Cloud Witness as a quorum witness are as follows:

- Disaster recovery stretched multi-site clusters (see figure 2).
- Failover Clusters without shared storage (SQL Always On etc.).
- Failover Clusters running inside Guest OS hosted in Microsoft Azure Virtual Machine Role (or any other public cloud).
- Failover Clusters running inside Guest OS of Virtual Machines hosted in private clouds.
- Storage clusters with or without shared storage, such as Scale-out File Server clusters.
- Small branch-office clusters (even 2-node clusters)

Starting with Windows Server 2012 R2, it is recommended to always configure a witness as the cluster automatically manages the witness vote and the nodes vote with Dynamic Quorum.

Set up a Cloud Witness for a cluster

To set up a Cloud Witness as a quorum witness for your cluster, complete the following steps:

1. Create an Azure Storage Account to use as a Cloud Witness
2. Configure the Cloud Witness as a quorum witness for your cluster.

Create an Azure Storage Account to use as a Cloud Witness

This section describes how to create a storage account and view and copy endpoint URLs and access keys for that account.

To configure Cloud Witness, you must have a valid Azure Storage Account which can be used to store the blob file (used for arbitration). Cloud Witness creates a well-known Container **msft-cloud-witness** under the Microsoft Storage Account. Cloud Witness writes a single blob file with corresponding cluster's unique ID used as the file name of the blob file under this **msft-cloud-witness** container. This means that you can use the same Microsoft Azure Storage Account to configure a Cloud Witness for multiple different clusters.

When you use the same Azure Storage Account for configuring Cloud Witness for multiple different clusters, a single **msft-cloud-witness** container gets created automatically. This container will contain one-blob file per cluster.

To create an Azure storage account

1. Sign in to the [Azure Portal](#).
2. On the Hub menu, select New -> Data + Storage -> Storage account.
3. In the Create a storage account page, do the following:
 - a. Enter a name for your storage account.
Storage account names must be between 3 and 24 characters in length and may contain numbers and lowercase letters only. The storage account name must also be unique within Azure.
 - b. For **Account kind**, select **General purpose**.
You can't use a Blob storage account for a Cloud Witness.
 - c. For **Performance**, select **Standard**.
You can't use Azure Premium Storage for a Cloud Witness.
 - d. For **Replication**, select **Locally-redundant storage (LRS)**.
Failover Clustering uses the blob file as the arbitration point, which requires some consistency guarantees when reading the data. Therefore you must select **Locally-redundant storage** for **Replication** type.

View and copy storage access keys for your Azure Storage Account

When you create a Microsoft Azure Storage Account, it is associated with two Access Keys that are automatically generated - Primary Access key and Secondary Access key. For a first-time creation of Cloud Witness, use the **Primary Access Key**. There is no restriction regarding which key to use for Cloud Witness.

To view and copy storage access keys

In the Azure Portal, navigate to your storage account, click **All settings** and then click **Access Keys** to view, copy, and regenerate your account access keys. The Access Keys blade also includes pre-configured connection strings using your primary and secondary keys that you can copy to use in your applications (see figure 4).

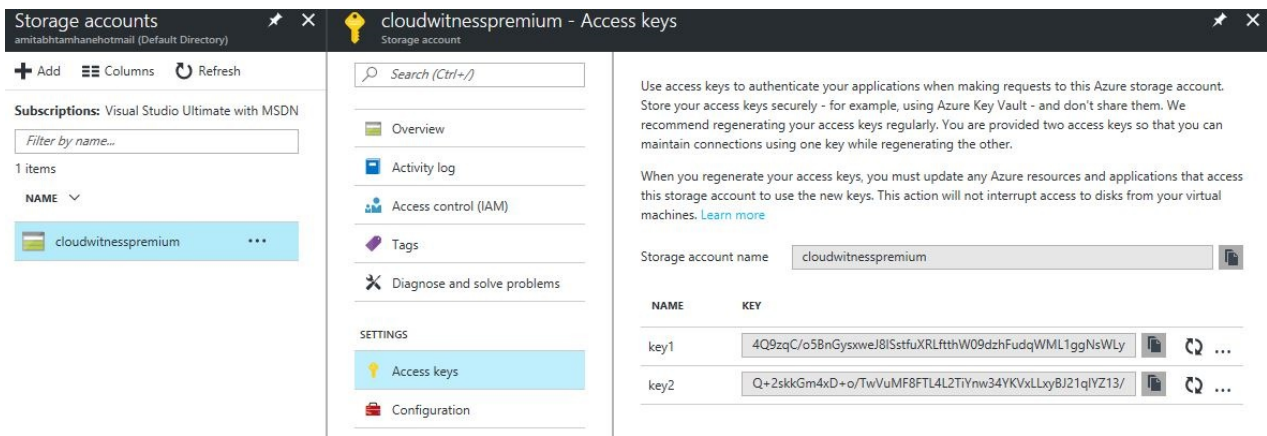


Figure 4: Storage Access Keys

View and copy endpoint URL Links

When you create a Storage Account, the following URLs are generated using the format:

`https://<Storage Account Name>.<Storage Type>.<Endpoint>`

Cloud Witness always uses **Blob** as the storage type. Azure uses **.core.windows.net** as the Endpoint. When configuring Cloud Witness, it is possible that you configure it with a different endpoint as per your scenario (for example the Microsoft Azure datacenter in China has a different endpoint).

NOTE

The endpoint URL is generated automatically by Cloud Witness resource and there is no extra step of configuration necessary for the URL.

To view and copy endpoint URL links

In the Azure Portal, navigate to your storage account, click **All settings** and then click **Properties** to view and copy your endpoint URLs (see figure 5).

services	
SERVICE	ENDPOINT
Blobs	https://mycloudwitness.blob.core.windows.net/
Tables	https://mycloudwitness.table.core.windows.net/
Queues	https://mycloudwitness.queue.core.windows.net/

Figure 5: Cloud Witness endpoint URL links

For more information about creating and managing Azure Storage Accounts, see [About Azure Storage Accounts](#)

Configure Cloud Witness as a quorum witness for your cluster

Cloud Witness configuration is well-integrated within the existing Quorum Configuration Wizard built into the Failover Cluster Manager.

To configure Cloud Witness as a Quorum Witness

1. Launch Failover Cluster Manager.
2. Right-click the cluster -> **More Actions** -> **Configure Cluster Quorum Settings** (see figure 6). This launches the Configure Cluster Quorum wizard.

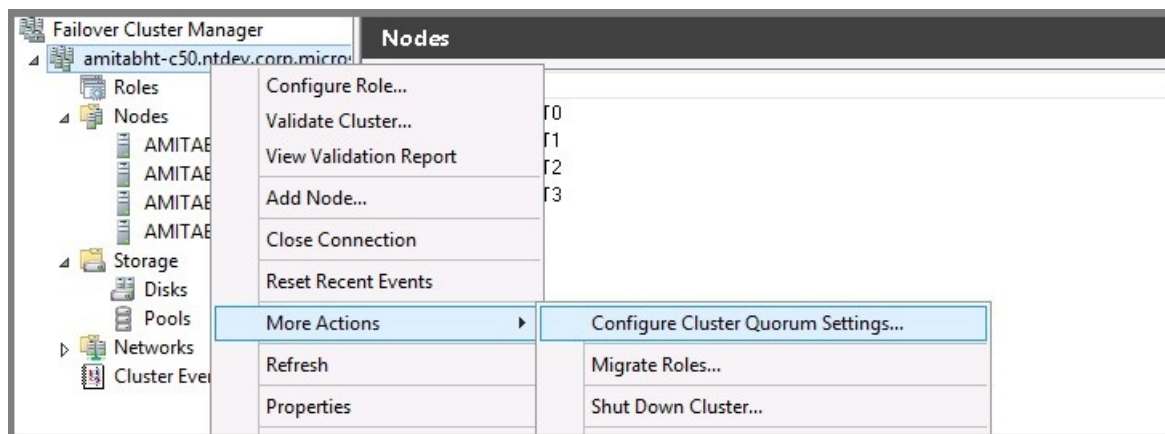


Figure 6. Cluster Quorum Settings

3. On the **Select Quorum Configurations** page, select **Select the quorum witness** (see figure 7).

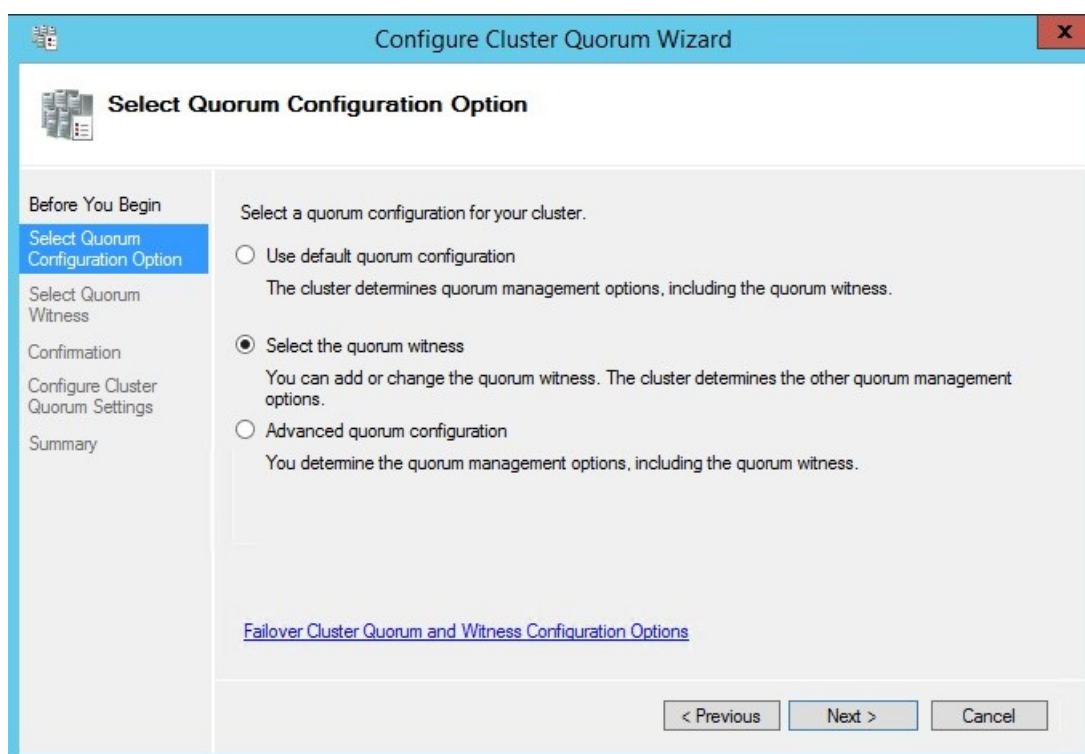


Figure 7. Select the Quorum Configuration

4. On the **Select Quorum Witness** page, select **Configure a cloud witness** (see figure 8).

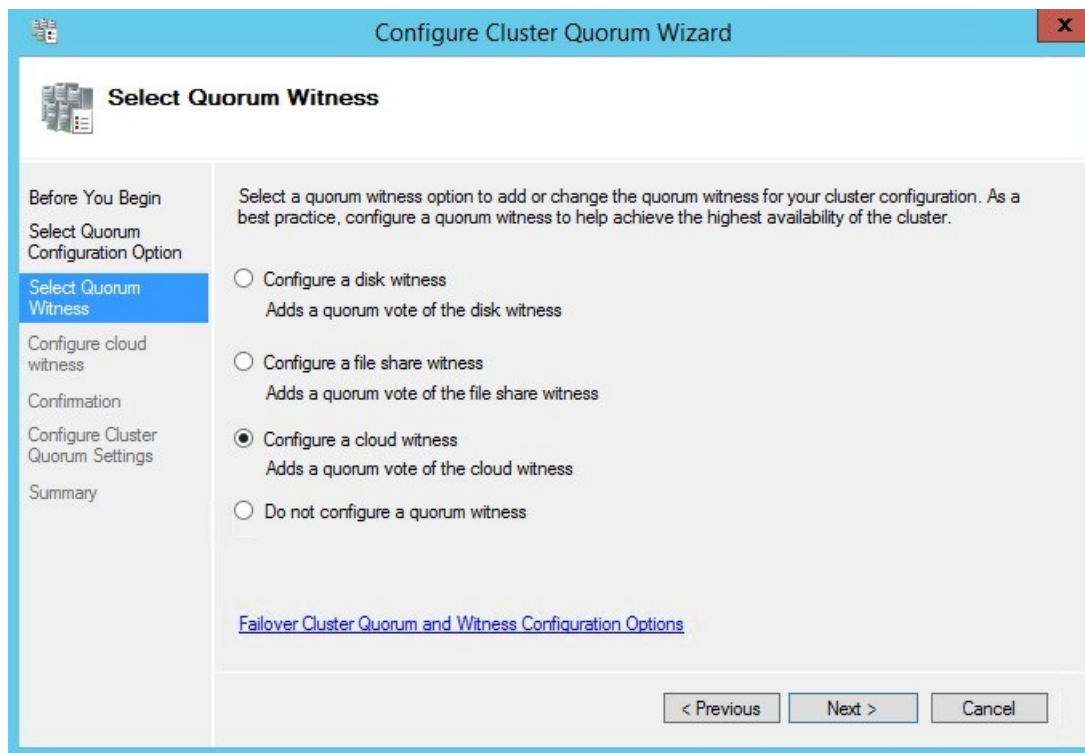


Figure 8. Select the Quorum Witness

5. On the **Configure Cloud Witness** page, enter the following information:
 - a. (Required parameter) Azure Storage Account Name.
 - b. (Required parameter) Access Key corresponding to the Storage Account.
 - a. When creating for the first time, use Primary Access Key (see figure 5)
 - b. When rotating the Primary Access Key, use Secondary Access Key (see figure 5)
 - c. (Optional parameter) If you intend to use a different Azure service endpoint (for example the Microsoft Azure service in China), then update the endpoint server name.

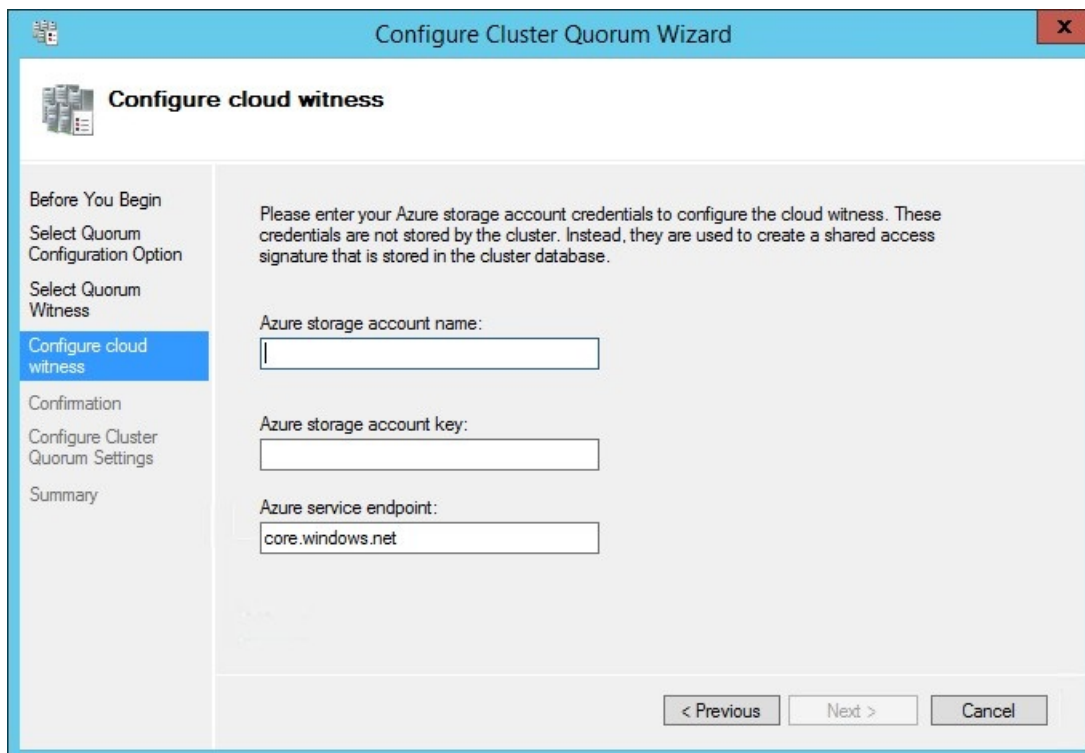
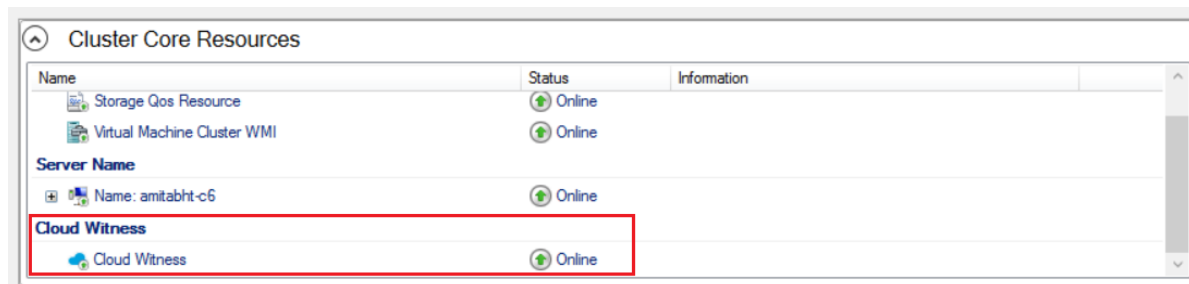


Figure 9: Configure your Cloud Witness

6. Upon successful configuration of Cloud Witness, you can view the newly created witness resource in the Failover Cluster Manager snap-in (see figure 10).



Name	Status	Information
Storage Qos Resource	Online	
Virtual Machine Cluster WMI	Online	
Server Name		
Name: amitabht-c6	Online	
Cloud Witness		
Cloud Witness	Online	

Figure 10: Successful configuration of Cloud Witness

Configuring Cloud Witness using PowerShell

The existing Set-ClusterQuorum PowerShell command has new additional parameters corresponding to Cloud Witness.

You can configure Cloud Witness using the [Set-ClusterQuorum](#) following PowerShell command:

```
Set-ClusterQuorum -CloudWitness -AccountName <StorageAccountName> -AccessKey <StorageAccountAccessKey>
```

In case you need to use a different endpoint (rare):

```
Set-ClusterQuorum -CloudWitness -AccountName <StorageAccountName> -AccessKey <StorageAccountAccessKey> -  
Endpoint <servername>
```

Azure Storage Account considerations with Cloud Witness

When configuring a Cloud Witness as a quorum witness for your Failover Cluster, consider the following:

- Instead of storing the Access Key, your Failover Cluster will generate and securely store a Shared Access Security (SAS) token.
- The generated SAS token is valid as long as the Access Key remains valid. When rotating the Primary Access Key, it is important to first update the Cloud Witness (on all your clusters that are using that Storage Account) with the Secondary Access Key before regenerating the Primary Access Key.
- Cloud Witness uses HTTPS REST interface of the Azure Storage Account service. This means it requires the HTTPS port to be open on all cluster nodes.

Proxy considerations with Cloud Witness

Cloud Witness uses HTTPS (default port 443) to establish communication with Azure blob service. Ensure that HTTPS port is accessible via network Proxy.

See Also

- [What's New in Failover Clustering in Windows Server](#)

Cluster operating system rolling upgrade

4/10/2018 • 15 minutes to read • [Edit Online](#)

Applies to: Windows Server (Semi-Annual Channel), Windows Server 2016

Cluster OS Rolling Upgrade enables an administrator to upgrade the operating system of the cluster nodes 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.

Cluster OS Rolling Upgrade provides the following benefits:

- Failover clusters running Hyper-V virtual machine and Scale-out File Server (SOFS) workloads can be upgraded from Windows Server 2012 R2 (running on all nodes in the cluster) to Windows Server 2016 (running on all cluster nodes of the cluster) without downtime. Other cluster workloads, such as SQL Server, will be unavailable during the time (typically less than five minutes) it takes to failover to Windows Server 2016.
- It doesn't require any additional hardware. Although, you can add additional cluster nodes temporarily to small clusters to improve availability of the cluster during the Cluster OS Rolling Upgrade process.
- The cluster doesn't need to be stopped or restarted.
- A new cluster is not required. The existing cluster is upgraded. In addition, existing cluster objects stored in Active Directory are used.
- The upgrade process is reversible until the customer chooses the "point-of-no-return", when all cluster nodes are running Windows Server 2016, and when the Update-ClusterFunctionalLevel PowerShell cmdlet is run.
- The cluster can support patching and maintenance operations while running in the mixed-OS mode.
- It supports automation via PowerShell and WMI.
- The cluster public property **ClusterFunctionalLevel** property indicates the state of the cluster on Windows Server 2016 cluster nodes. This property can be queried using the PowerShell cmdlet from a Windows Server 2016 cluster node that belongs to a failover cluster:

```
Get-Cluster | Select ClusterFunctionalLevel
```

A value of **8** indicates that the cluster is running at the Windows Server 2012 R2 functional level. A value of **9** indicates that the cluster is running at the Windows Server 2016 functional level.

This guide describes the various stages of the Cluster OS Rolling Upgrade process, installation steps, feature limitations, and frequently asked questions (FAQs), and is applicable to the following Cluster OS Rolling Upgrade scenarios in Windows Server 2016:

- Hyper-V clusters
- Scale-Out File Server clusters

The following scenario is not supported in Windows Server 2016:

- Cluster OS Rolling Upgrade of guest clusters using virtual hard disk (.vhdx file) as shared storage

Cluster OS Rolling Upgrade is fully supported by System Center Virtual Machine Manager (SCVMM) 2016. If you are using SCVMM 2016, see [Upgrading Windows Server 2012 R2 clusters to Windows Server 2016 in VMM](#) for guidance on upgrading the clusters and automating the steps that are described in this document.

Requirements

Complete the following requirements before you begin the Cluster OS Rolling Upgrade process:

- Start with a Failover Cluster running Windows Server (Semi-Annual Channel), Windows Server 2016, or Windows Server 2012 R2.
- Upgrading a Storage Spaces Direct cluster to Windows Server, version 1709 isn't supported.
- If the cluster workload is Hyper-V VMs, or Scale-Out File Server, you can expect zero-downtime upgrade.
- Verify that the Hyper-V nodes have CPUs that support Second-Level Addressing Table (SLAT) using one of the following methods;
 - Review the [Are you SLAT Compatible? WP8 SDK Tip 01](#) article that describes two methods to check if a CPU supports SLATs
 - Download the [Coreinfo v3.31](#) tool to determine if a CPU supports SLAT.

Cluster transition states during Cluster OS Rolling Upgrade

This section describes the various transition states of the Windows Server 2012 R2 cluster that is being upgraded to Windows Server 2016 using Cluster OS Rolling Upgrade.

In order to keep the cluster workloads running during the Cluster OS Rolling Upgrade process, moving a cluster workload from a Windows Server 2012 R2 node to Windows Server 2016 node works as if both nodes were running the Windows Server 2012 R2 operating system. When Windows Server 2016 nodes are added to the cluster, they operate in a Windows Server 2012 R2 compatibility mode. A new conceptual cluster mode, called "mixed-OS mode", allows nodes of different versions to exist in the same cluster (see Figure 1).

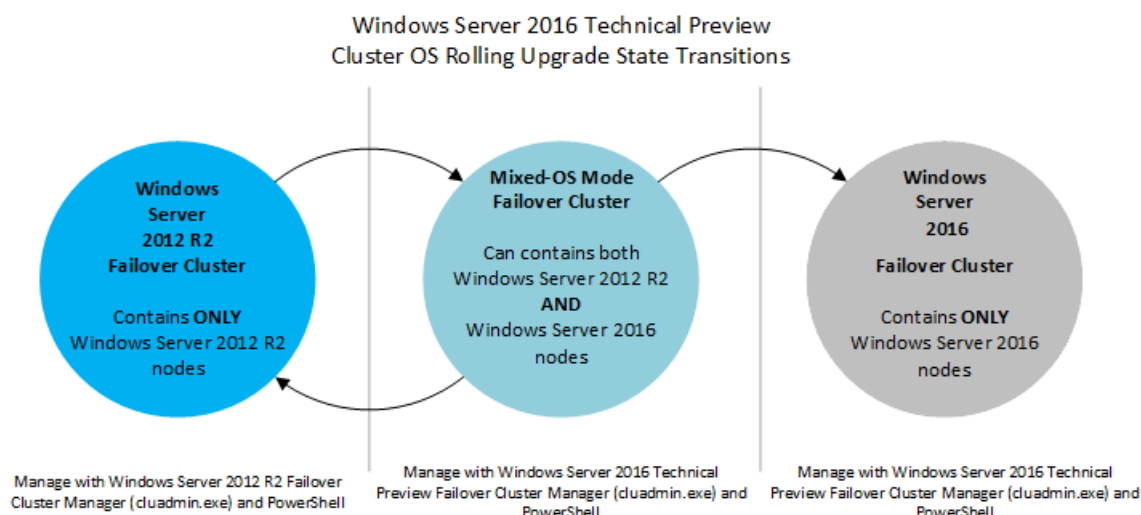


Figure 1: Cluster operating system state transitions

A Windows Server 2012 R2 cluster enters mixed-OS mode when a Windows Server 2016 node is added to the cluster. The process is fully reversible - Windows Server 2016 nodes can be removed from the cluster and Windows Server 2012 R2 nodes can be added to the cluster in this mode. The "point of no return" occurs when the Update-ClusterFunctionalLevel PowerShell cmdlet is run on the cluster. In order for this cmdlet to succeed, all nodes must be Windows Server 2016, and all nodes must be online.

Transition states of a four-node cluster while performing Rolling OS Upgrade

This section illustrates and describes the four different stages of a cluster with shared storage whose nodes are upgraded from Windows Server 2012 R2 to Windows Server 2016.

"Stage 1" is the initial state - we start with a Windows Server 2012 R2 cluster.

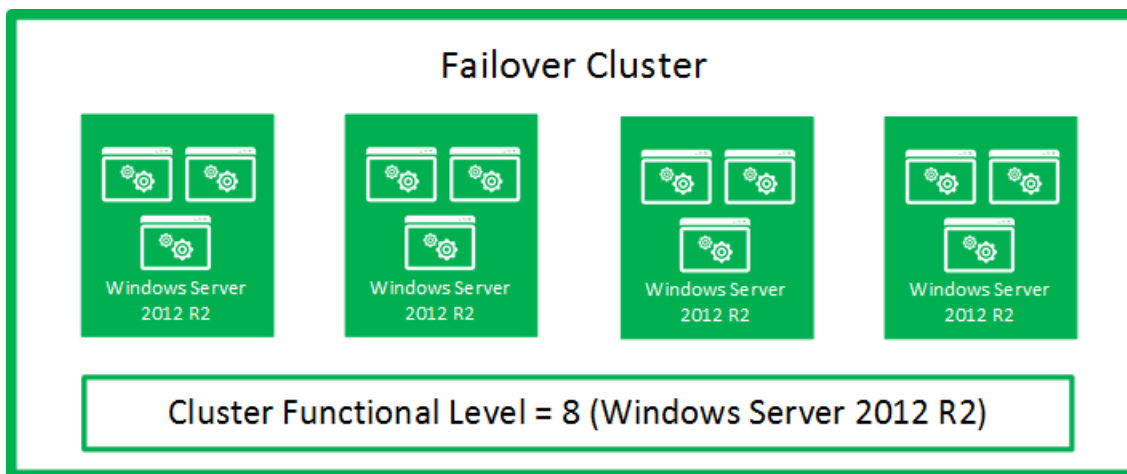


Figure 2: Initial State: Windows Server 2012 R2 Failover Cluster (Stage 1)

In "Stage 2", two nodes have been paused, drained, evicted, reformatted, and installed with Windows Server 2016.

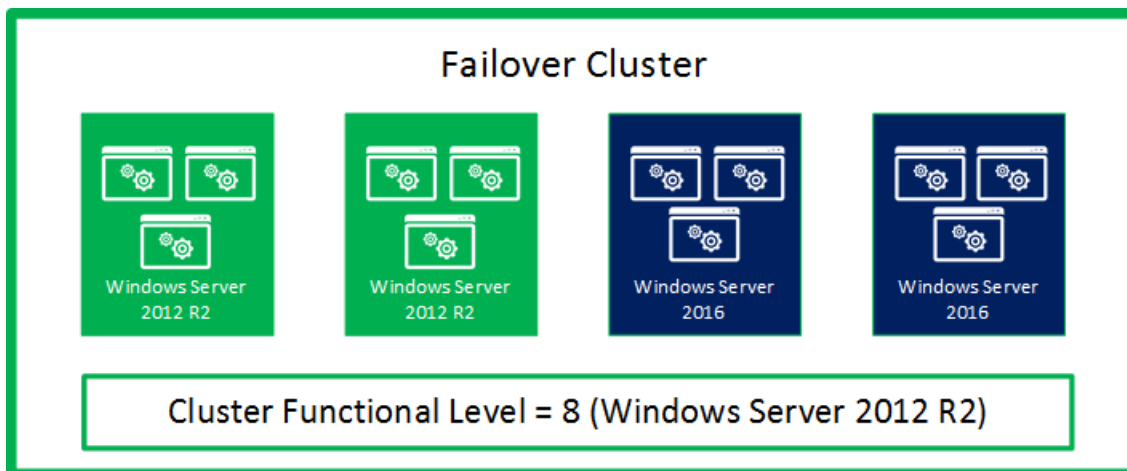


Figure 3: Intermediate State: Mixed-OS mode: Windows Server 2012 R2 and Windows Server 2016 Failover cluster (Stage 2)

At "Stage 3", all of the nodes in the cluster have been upgraded to Windows Server 2016, and the cluster is ready to be upgraded with Update-ClusterFunctionalLevel PowerShell cmdlet.

NOTE

At this stage, the process can be fully reversed, and Windows Server 2012 R2 nodes can be added to this cluster.

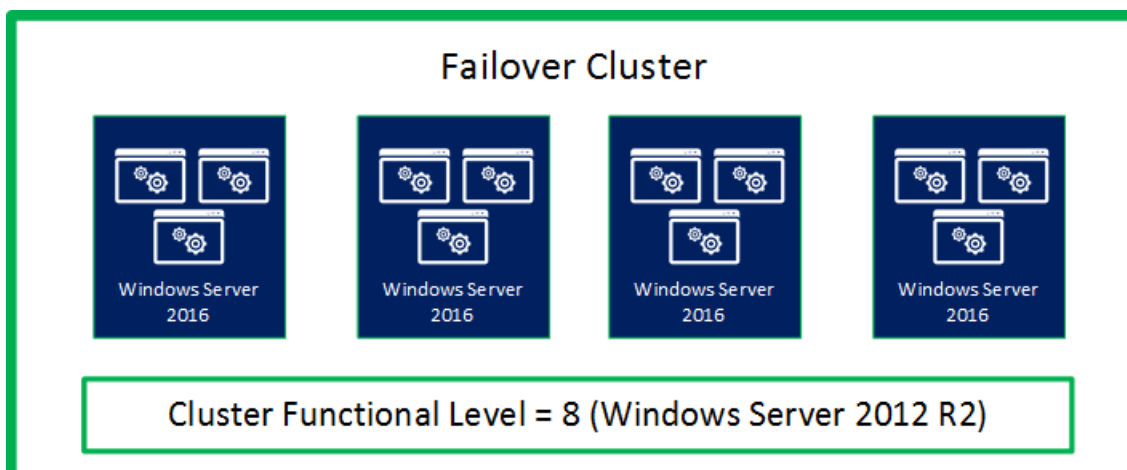


Figure 4: Intermediate State: All nodes upgraded to Windows Server 2016, ready for Update-ClusterFunctionalLevel (Stage 3)

After the Update-ClusterFunctionalLevelcmdlet is run, the cluster enters "Stage 4", where new Windows Server

2016 cluster features can be used.

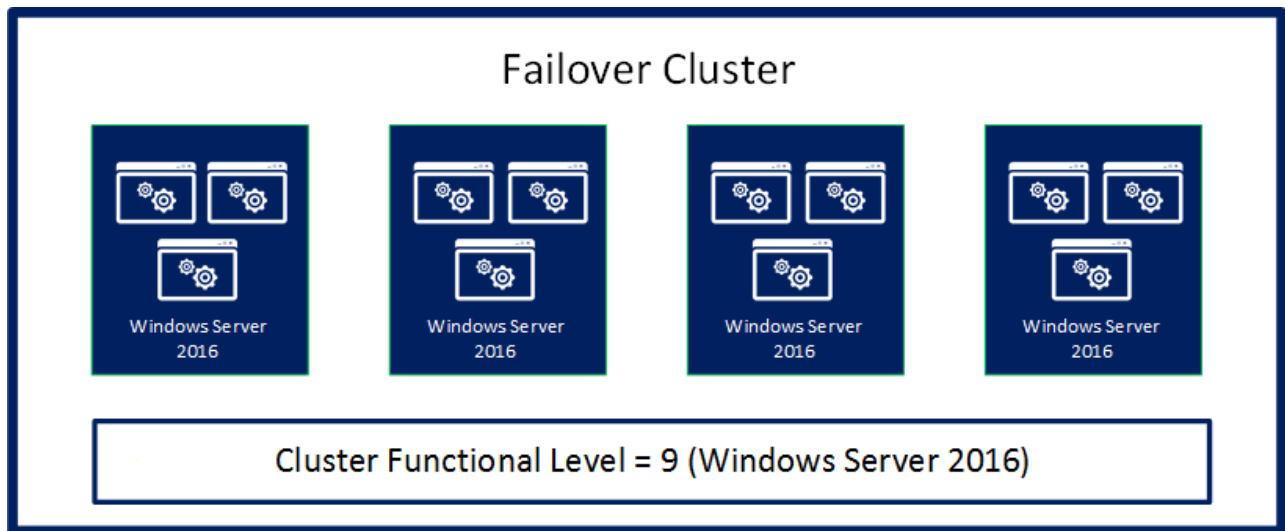


Figure 5: Final State: Windows Server 2016 Failover Cluster (Stage 4)

Cluster OS Rolling Upgrade Process

This section describes the workflow for performing Cluster OS Rolling Upgrade.

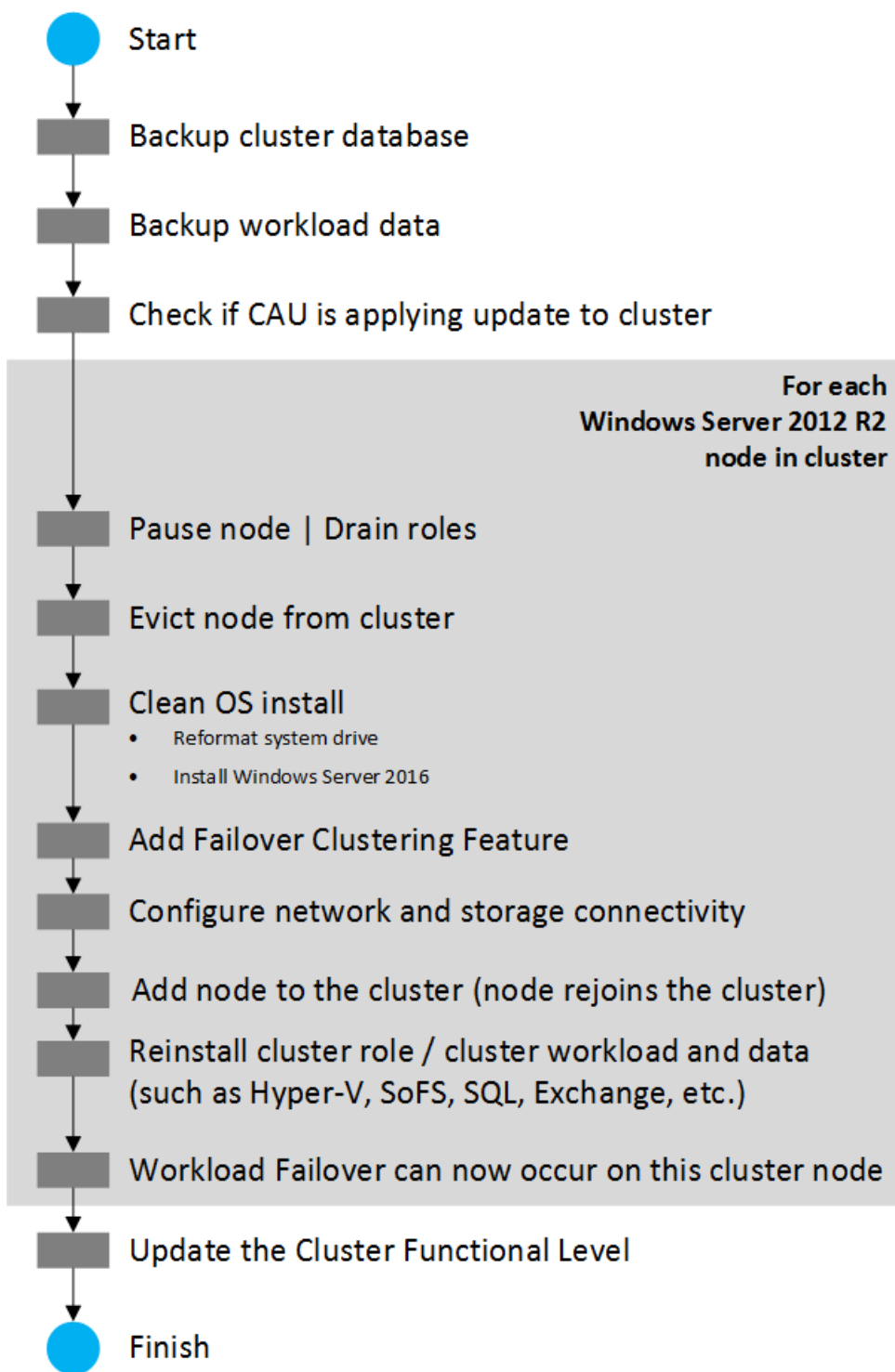


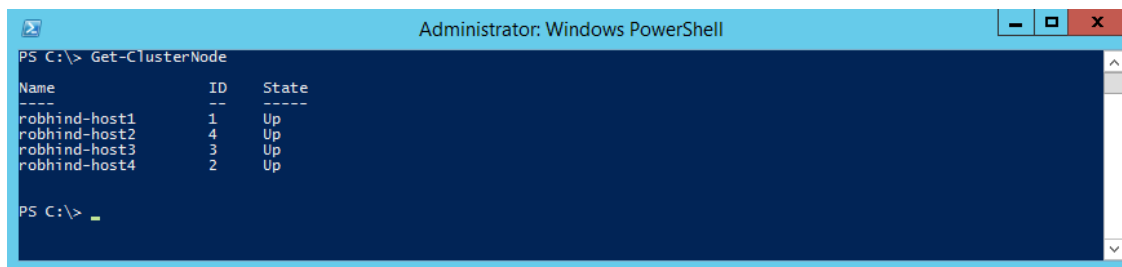
Figure 6: Cluster OS Rolling Upgrade Process Workflow

Cluster OS Rolling upgrade includes the following steps:

1. Prepare the cluster for the operating system upgrade as follows:
 - a. Cluster OS Rolling Upgrade requires removing one node at a time from the cluster. Check if you have sufficient capacity on the cluster to maintain HA SLAs when one of the cluster nodes is removed from the cluster for an operating system upgrade. In other words, do you require the capability to failover workloads to another node when one node is removed from the cluster during the process of Cluster OS Rolling Upgrade? Does the cluster have the capacity to run the required workloads when one node is removed from the cluster for Cluster OS Rolling Upgrade?
 - b. For Hyper-V workloads, check that all Windows Server 2016 Hyper-V hosts have CPU support Second-Level Address Table (SLAT). Only SLAT-capable machines can use the Hyper-V role in Windows Server 2016.
 - c. Check that any workload backups have completed, and consider backing-up the cluster. Stop backup

operations while adding nodes to the cluster.

- d. Check that all cluster nodes are online /running/up using the `Get-ClusterNode` cmdlet (see Figure 7).



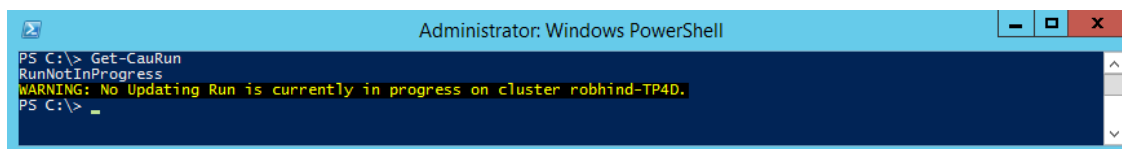
```
PS C:\> Get-ClusterNode

Name                ID  State
----                -  -
robbind-host1       1   Up
robbind-host2       4   Up
robbind-host3       3   Up
robbind-host4       2   Up

PS C:\> _
```

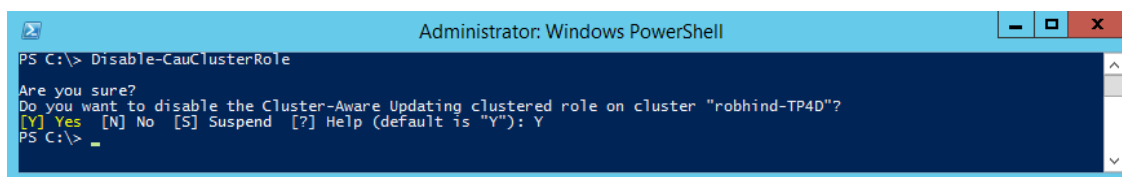
Figure 7: Determining node status using `Get-ClusterNode` cmdlet

- e. If you are running Cluster Aware Updates (CAU), verify if CAU is currently running by using the **Cluster-Aware Updating** UI, or the `Get-CauRun` cmdlet (see Figure 8). Stop CAU using the `Disable-CauClusterRole` cmdlet (see Figure 9) to prevent any nodes from being paused and drained by CAU during the Cluster OS Rolling Upgrade process.



```
PS C:\> Get-CauRun
RunNotInProgress
WARNING: No Updating Run is currently in progress on cluster robhind-TP4D.
PS C:\> _
```

Figure 8: Using the `Get-CauRun` cmdlet to determine if Cluster Aware Updates is running on the cluster



```
PS C:\> Disable-CauClusterRole

Are you sure?
Do you want to disable the Cluster-Aware Updating clustered role on cluster "robbind-TP4D"?
[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"): Y
PS C:\> _
```

Figure 9: Disabling the Cluster Aware Updates role using the `Disable-CauClusterRole` cmdlet

2. For each node in the cluster, complete the following:
- a. Using Cluster Manager UI, select a node and use the **Pause | Drain** menu option to drain the node (see Figure 10) or use the `Suspend-ClusterNode` cmdlet (see Figure 11).

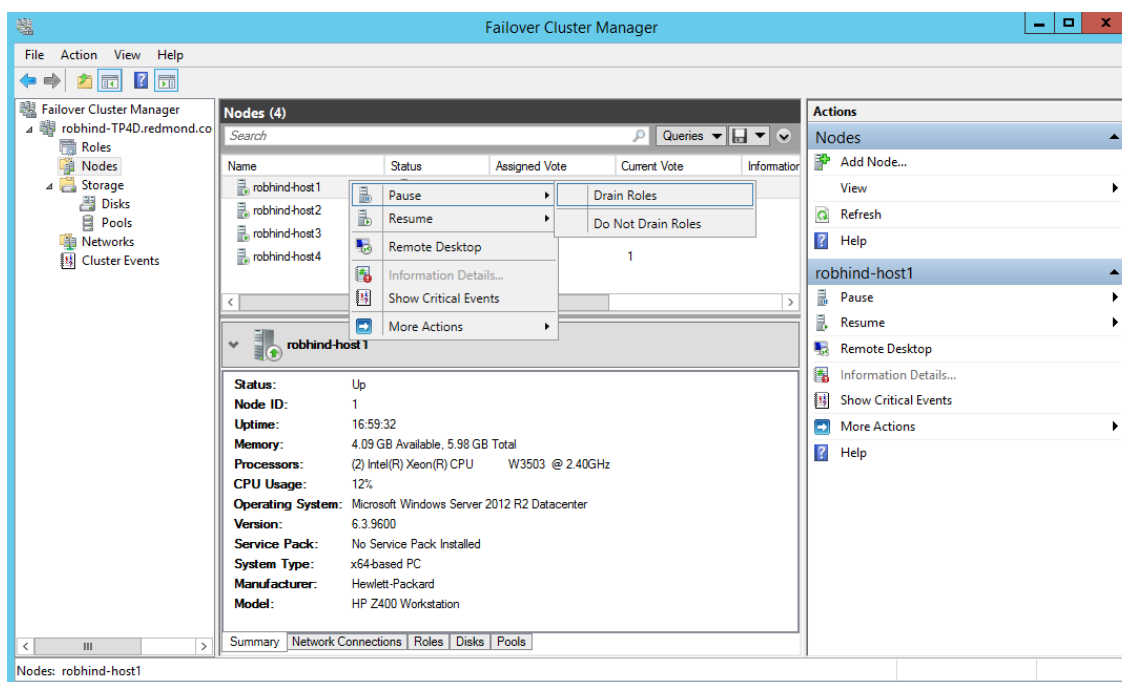


Figure 10: Draining roles from a node using Failover Cluster Manager

```
Administrator: Windows PowerShell
PS C:\> Suspend-ClusterNode -Name robhind-host1

Name      ID      State
----      -
robhind-host1 1      Paused
```

Figure 11: Draining roles from a node using the `Suspend-ClusterNode` cmdlet

- b. Using Cluster Manager UI, **Evict** the paused node from cluster, or use the `Remove-ClusterNode` cmdlet.

```
Administrator: Windows PowerShell
PS C:\> Remove-ClusterNode -Name robhind-host1

Remove-ClusterNode
Are you sure you want to evict node robhind-host1?
[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"): Y
PS C:\>
```

Figure 12: Remove a node from the cluster using `Remove-ClusterNode` cmdlet

- c. Reformat the system drive and perform a "clean operating system install" of Windows Server 2016 on the node using the **Custom: Install Windows only (advanced)** installation (See Figure 13) option in setup.exe. Avoid selecting the **Upgrade: Install Windows and keep files, settings, and applications** option since Cluster OS Rolling Upgrade doesn't encourage in-place upgrade.

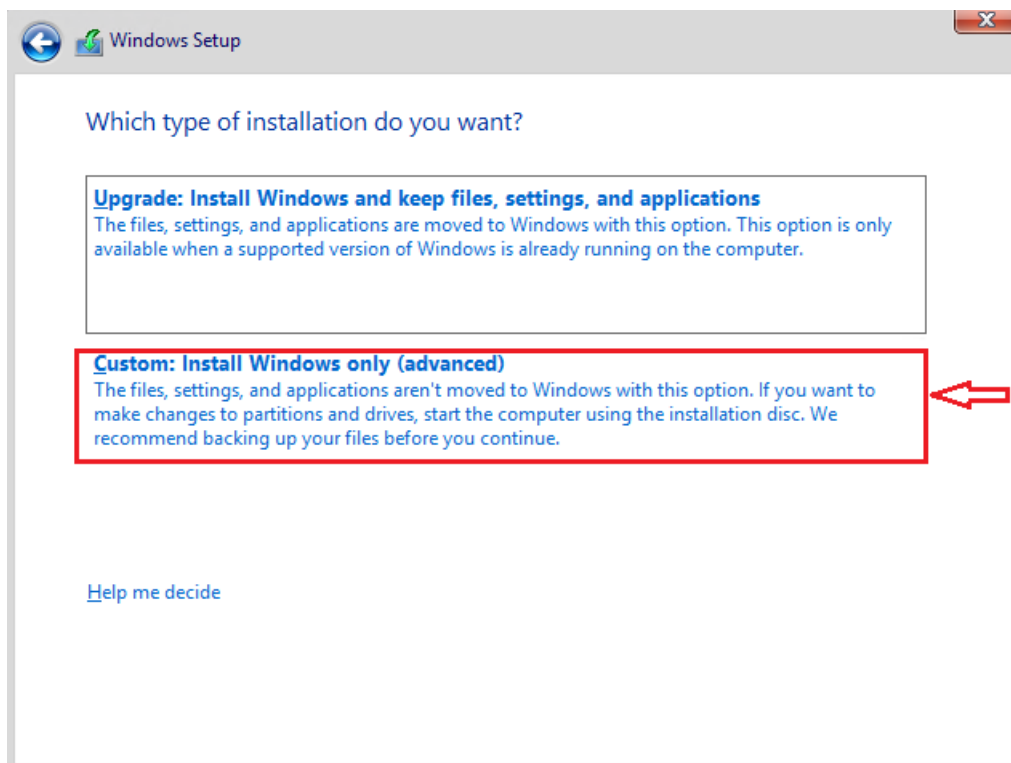


Figure 13: Available installation options for Windows Server 2016

- d. Add the node to the appropriate Active Directory domain.
- e. Add the appropriate users to the Administrators group.
- f. Using the Server Manager UI or `Install-WindowsFeature` PowerShell cmdlet, install any server roles that you need, such as Hyper-V.

```
Install-WindowsFeature -Name Hyper-V
```

- g. Using the Server Manager UI or `Install-WindowsFeature` PowerShell cmdlet, install the Failover Clustering feature.

```
Install-WindowsFeature -Name Failover-Clustering
```

- h. Install any additional features needed by your cluster workloads.
- i. Check network and storage connectivity settings using the Failover Cluster Manager UI.
- j. If Windows Firewall is used, check that the Firewall settings are correct for the cluster. For example, Cluster Aware Updating (CAU) enabled clusters may require Firewall configuration.
- k. For Hyper-V workloads, use the Hyper-V Manager UI to launch the Virtual Switch Manager dialog (see Figure 14).

Check that the name of the Virtual Switch(s) used are identical for all Hyper-V host nodes in the cluster.

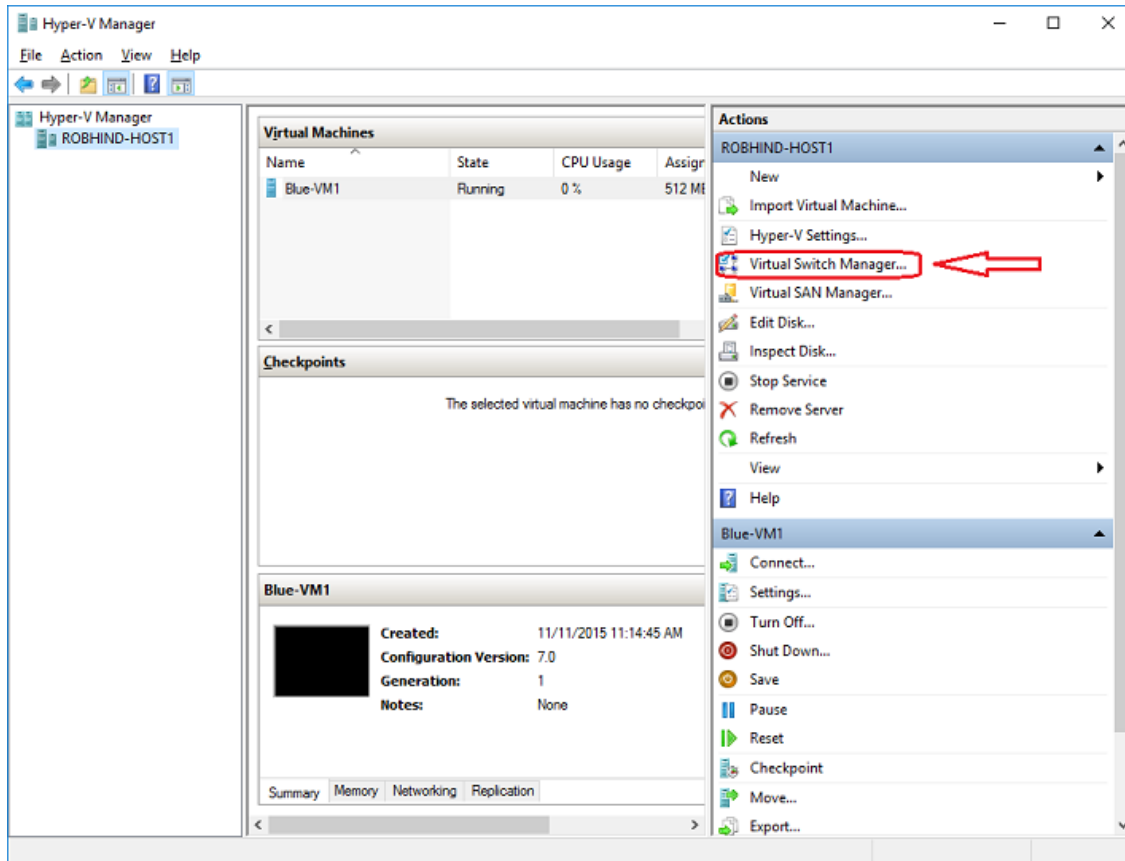


Figure 14: Virtual Switch Manager

- l. On a Windows Server 2016 node (do not use a Windows Server 2012 R2 node), use the Failover Cluster Manager (see Figure 15) to connect to the cluster.

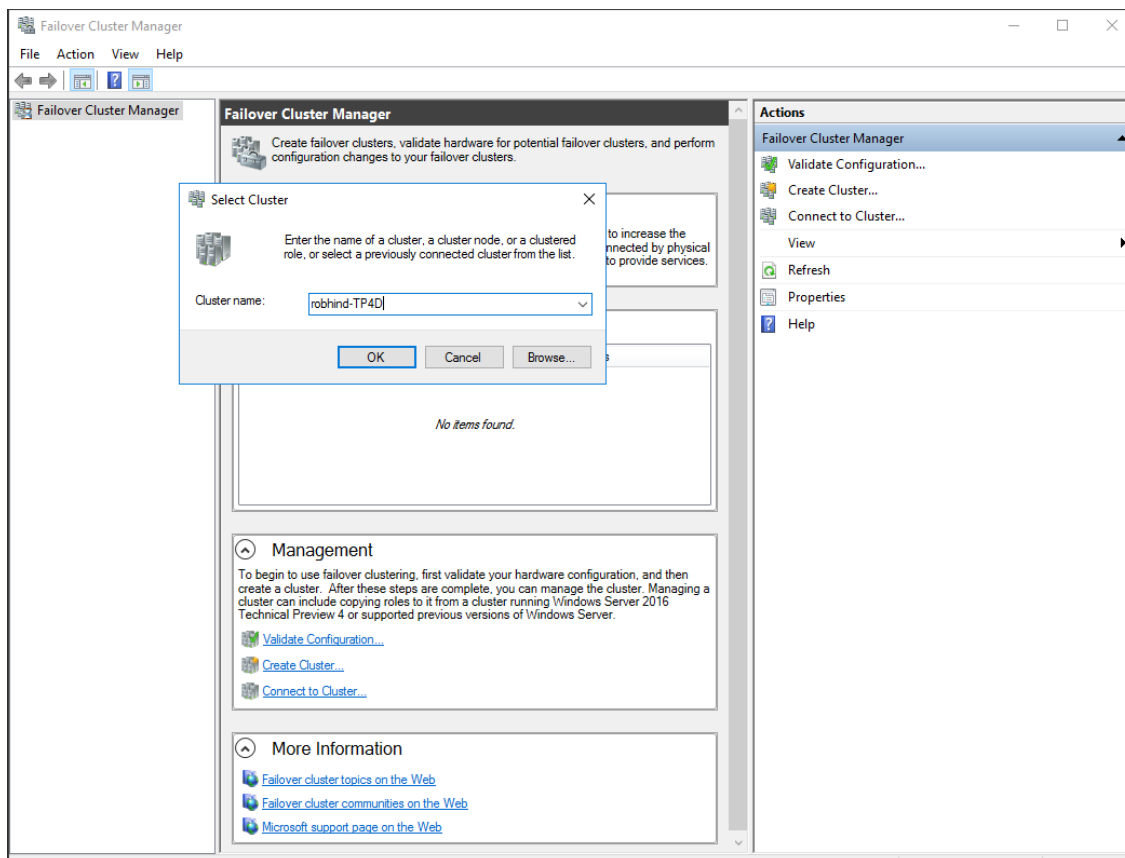


Figure 15: Adding a node to the cluster using Failover Cluster Manager

- m. Use either the Failover Cluster Manager UI or the `Add-ClusterNode` cmdlet (see Figure 16) to add the node to the cluster.

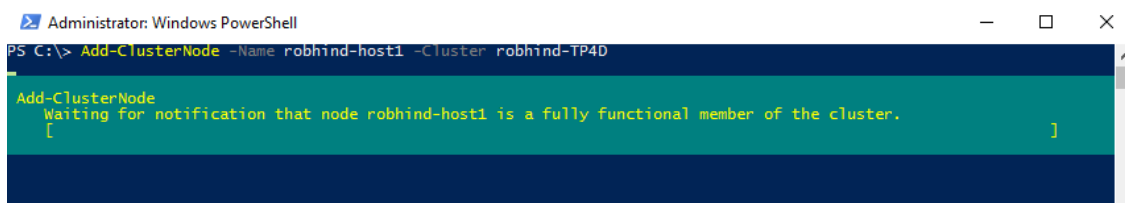


Figure 16: Adding a node to the cluster using `Add-ClusterNode` cmdlet

NOTE

When the first Windows Server 2016 node joins the cluster, the cluster enters "Mixed-OS" mode, and the cluster core resources are moved to the Windows Server 2016 node. A "Mixed-OS" mode cluster is a fully functional cluster where the new nodes run in a compatibility mode with the old nodes. "Mixed-OS" mode is a transitory mode for the cluster. It is not intended to be permanent and customers are expected to update all nodes of their cluster within four weeks.

- n. After the Windows Server 2016 node is successfully added to the cluster, you can (optionally) move some of the cluster workload to the newly added node in order to rebalance the workload across the cluster as follows:

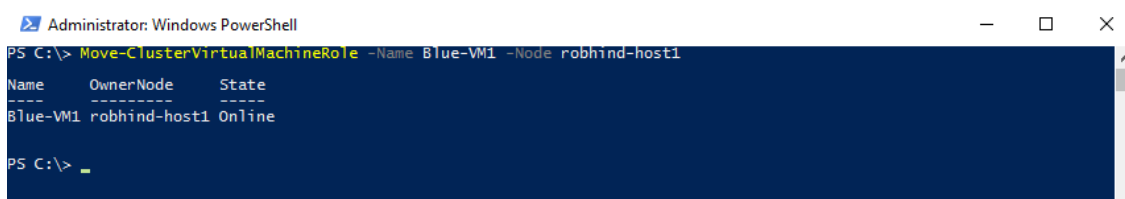


Figure 17: Moving a cluster workload (cluster VM role) using `Move-ClusterVirtualMachineRole` cmdlet

- a. Use **Live Migration** from the Failover Cluster Manager for virtual machines or the `Move-ClusterVirtualMachineRole` cmdlet (see Figure 17) to perform a live migration of the virtual machines.

```
Move-ClusterVirtualMachineRole -Name VM1 -Node robhind-host3
```

- b. Use **Move** from the Failover Cluster Manager or the `Move-ClusterGroup` cmdlet for other cluster workloads.

3. When every node has been upgraded to Windows Server 2016 and added back to the cluster, or when any remaining Windows Server 2012 R2 nodes have been evicted, do the following:

IMPORTANT

- After you update the cluster functional level, you cannot go back to Windows Server 2012 R2 functional level and Windows Server 2012 R2 nodes cannot be added to the cluster.
- Until the `Update-ClusterFunctionalLevel` cmdlet is run, the process is fully reversible and Windows Server 2012 R2 nodes can be added to this cluster and Windows Server 2016 nodes can be removed.
- After the `Update-ClusterFunctionalLevel` cmdlet is run, new features will be available.

- a. Using the Failover Cluster Manager UI or the `Get-ClusterGroup` cmdlet, check that all cluster roles are running on the cluster as expected. In the following example, Available Storage is not being used, instead CSV is used, hence, Available Storage displays an **Offline** status (see Figure 18).



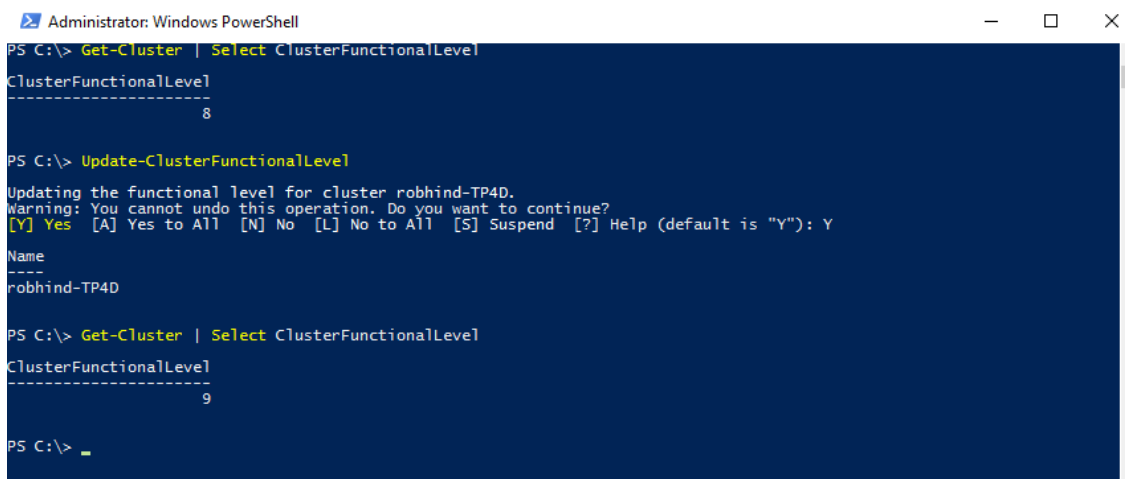
```
Administrator: Windows PowerShell
PS C:\> Get-ClusterGroup

Name                OwnerNode    State
----                -
Available Storage   robhind-host2 Offline
Blue-VM1             robhind-host1 Online
Blue-VM2             robhind-host2 Online
Blue-VM3             robhind-host3 Online
Blue-VM4             robhind-host4 Online
Cluster Group       robhind-host1 Online

PS C:\>
```

Figure 18: Verifying that all cluster groups (cluster roles) are running using the `Get-ClusterGroup` cmdlet

- b. Check that all cluster nodes are online and running using the `Get-ClusterNode` cmdlet.
- c. Run the `Update-ClusterFunctionalLevel` cmdlet - no errors should be returned (see Figure 19).



```
Administrator: Windows PowerShell
PS C:\> Get-Cluster | Select ClusterFunctionalLevel

ClusterFunctionalLevel
-----
8

PS C:\> Update-ClusterFunctionalLevel

Updating the functional level for cluster robhind-TP4D.
Warning: You cannot undo this operation. Do you want to continue?
[Y] Yes [A] Yes to All [N] No [L] No to All [S] Suspend [?] Help (default is "Y"): Y

Name
----
robhind-TP4D

PS C:\> Get-Cluster | Select ClusterFunctionalLevel

ClusterFunctionalLevel
-----
9

PS C:\>
```

Figure 19: Updating the functional level of a cluster using PowerShell

d. After the `Update-ClusterFunctionalLevel` cmdlet is run, new features are available.

4. Windows Server 2016 - resume normal cluster updates and backups:

- a. If you were previously running CAU, restart it using the CAU UI or use the `Enable-CauClusterRole` cmdlet (see Figure 20).

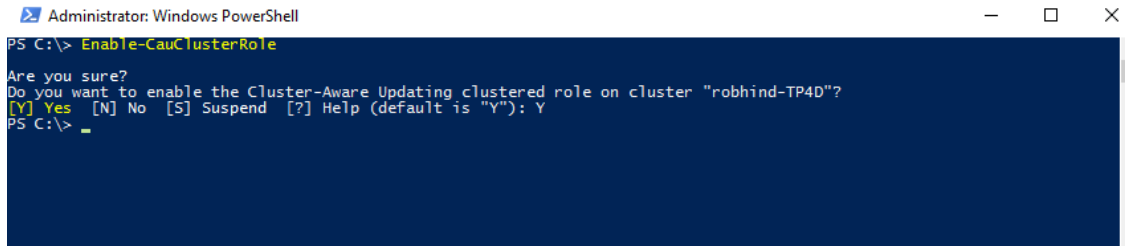


Figure 20: Enable Cluster Aware Updates role using the `Enable-CauClusterRole` cmdlet

- b. Resume backup operations.

5. Enable and use the Windows Server 2016 features on Hyper-V Virtual Machines.

- a. After the cluster has been upgraded to Windows Server 2016 functional level, many workloads like Hyper-V VMs will have new capabilities. For a list of new Hyper-V capabilities, see [Migrate and upgrade virtual machines](#)
- b. On each Hyper-V host node in the cluster, use the `Get-VMHostSupportedVersion` cmdlet to view the Hyper-V VM configuration versions that are supported by the host.

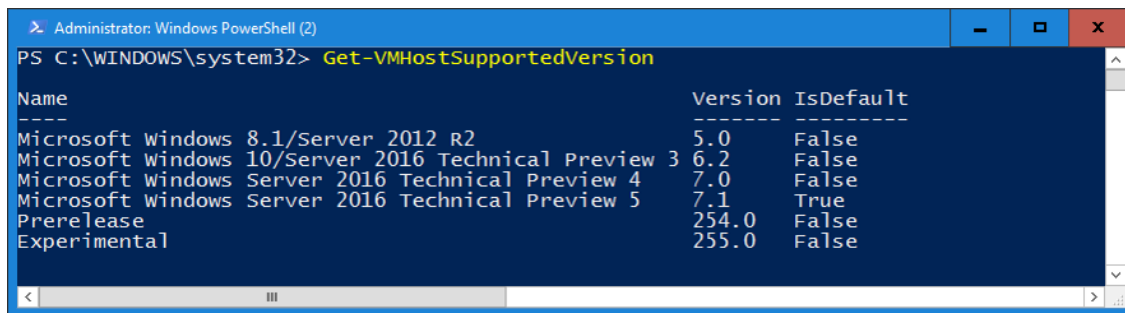
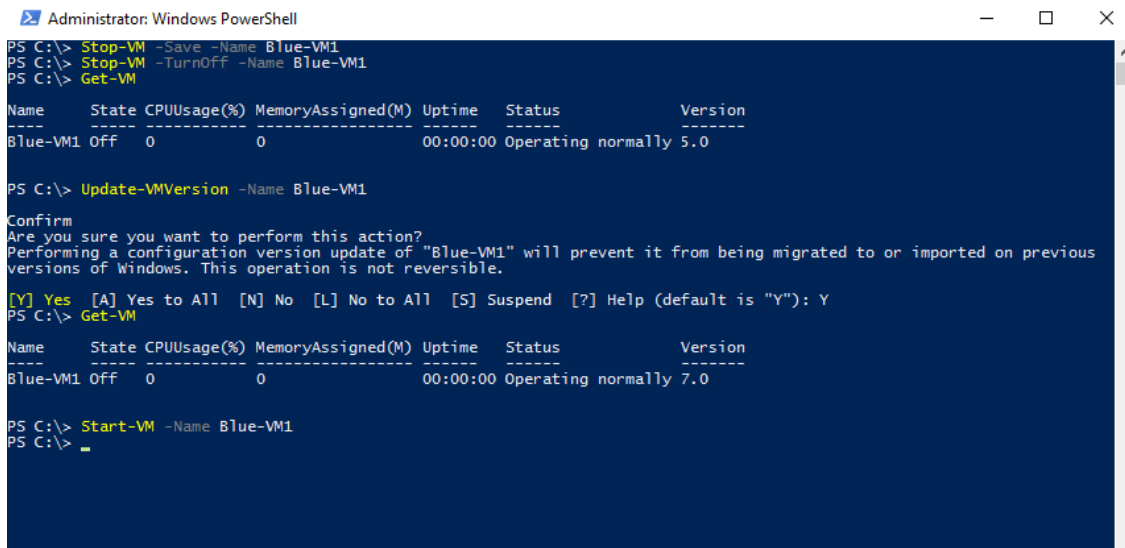


Figure 21: Viewing the Hyper-V VM configuration versions supported by the host

- a. On each Hyper-V host node in the cluster, Hyper-V VM configuration versions can be upgraded by scheduling a brief maintenance window with users, backing up, turning off virtual machines, and running the `Update-VMVersion` cmdlet (see Figure 22). This will update the virtual machine version, and enable new Hyper-V features, eliminating the need for future Hyper-V Integration Component (IC) updates. This cmdlet can be run from the Hyper-V node that is hosting the VM, or the `-ComputerName` parameter can be used to update the VM Version remotely. In this example, here we upgrade the configuration version of VM1 from 5.0 to 7.0 to take advantage of many new Hyper-V features associated with this VM configuration version such as Production Checkpoints (Application Consistent backups), and binary VM configuration file.



```
Administrator: Windows PowerShell
PS C:\> Stop-VM -Save -Name Blue-VM1
PS C:\> Stop-VM -TurnOff -Name Blue-VM1
PS C:\> Get-VM

Name      State CPUUsage(%) MemoryAssigned(M) Uptime    Status           Version
-----
Blue-VM1 Off      0           0              00:00:00 Operating normally 5.0

PS C:\> Update-VMVersion -Name Blue-VM1

Confirm
Are you sure you want to perform this action?
Performing a configuration version update of "Blue-VM1" will prevent it from being migrated to or imported on previous
versions of Windows. This operation is not reversible.

[Y] Yes [A] Yes to All [N] No [L] No to All [S] Suspend [?] Help (default is "Y"): Y
PS C:\> Get-VM

Name      State CPUUsage(%) MemoryAssigned(M) Uptime    Status           Version
-----
Blue-VM1 Off      0           0              00:00:00 Operating normally 7.0

PS C:\> Start-VM -Name Blue-VM1
PS C:\>
```

Figure 22: Upgrading a VM version using the Update-VMVersion PowerShell cmdlet

6. Storage pools can be upgraded using the [Update-StoragePool](#) PowerShell cmdlet - this is an online operation.

Although we are targeting Private Cloud scenarios, specifically Hyper-V and Scale-out File Server clusters, which can be upgraded without downtime, the Cluster OS Rolling Upgrade process can be used for any cluster role.

Restrictions / Limitations

- This feature works only for Windows Server 2012 R2 to Windows Server 2016 versions only. This feature cannot upgrade earlier versions of Windows Server such as Windows Server 2008, Windows Server 2008 R2, or Windows Server 2012 to Windows Server 2016.
- Each Windows Server 2016 node should be reformatted/new installation only. "In-place" or "upgrade" installation type is discouraged.
- A Windows Server 2016 node must be used to add Windows Server 2016 nodes to the cluster.
- When managing a mixed-OS mode cluster, always perform the management tasks from an uplevel node that is running Windows Server 2016. Downlevel Windows Server 2012 R2 nodes cannot use UI or management tools against Windows Server 2016.
- We encourage customers to move through the cluster upgrade process quickly because some cluster features are not optimized for mixed-OS mode.
- Avoid creating or resizing storage on Windows Server 2016 nodes while the cluster is running in mixed-OS mode because of possible incompatibilities on failover from a Windows Server 2016 node to down-level Windows Server 2012 R2 nodes.

Frequently asked questions

How long can the failover cluster run in mixed-OS mode?

We encourage customers to complete the upgrade within four weeks. There are many optimizations in Windows Server 2016. We have successfully upgraded Hyper-V and Scale-out File Server clusters with zero downtime in less than four hours total.

Will you port this feature back to Windows Server 2012, Windows Server 2008 R2, or Windows Server 2008?

We do not have any plans to port this feature back to previous versions. Cluster OS Rolling Upgrade is our vision for upgrading Windows Server 2012 R2 clusters to Windows Server 2016 and beyond.

Does the Windows Server 2012 R2 cluster need to have all the software updates installed before starting the Cluster OS Rolling Upgrade process?

Yes, before starting the Cluster OS Rolling Upgrade process, verify that all cluster nodes are updated with the latest software updates.

Can I run the `Update-ClusterFunctionalLevel` cmdlet while nodes are Off or Paused?

No. All cluster nodes must be on and in active membership for the `Update-ClusterFunctionalLevel` cmdlet to work.

Does Cluster OS Rolling Upgrade work for any cluster workload? Does it work for SQL Server?

Yes, Cluster OS Rolling Upgrade works for any cluster workload. However, it is only zero-downtime for Hyper-V and Scale-out File Server clusters. Most other workloads incur some downtime (typically a couple of minutes) when they failover, and failover is required at least once during the Cluster OS Rolling Upgrade process.

Can I automate this process using PowerShell?

Yes, we have designed Cluster OS Rolling Upgrade to be automated using PowerShell.

For a large cluster that has extra workload and failover capacity, can I upgrade multiple nodes simultaneously?

Yes. When one node is removed from the cluster to upgrade the OS, the cluster will have one less node for failover, hence will have a reduced failover capacity. For large clusters with enough workload and failover capacity, multiple nodes can be upgraded simultaneously. You can temporarily add cluster nodes to the cluster to provide improved workload and failover capacity during the Cluster OS Rolling Upgrade process.

What if I discover an issue in my cluster after `Update-ClusterFunctionalLevel` has been run successfully?

If you have backed-up the cluster database with a System State backup before running `Update-ClusterFunctionalLevel`, you should be able to perform an Authoritative restore on a Windows Server 2012 R2 cluster node and restore the original cluster database and configuration.

Can I use in-place upgrade for each node instead of using clean-OS install by reformatting the system drive?

We do not encourage the use of in-place upgrade of Windows Server, but we are aware that it works in some cases where default drivers are used. Please carefully read all warning messages displayed during in-place upgrade of a cluster node.

If I am using Hyper-V replication for a Hyper-V VM on my Hyper-V cluster, will replication remain intact during and after the Cluster OS Rolling Upgrade process?

Yes, Hyper-V replica remains intact during and after the Cluster OS Rolling Upgrade process.

Can I use System Center 2016 Virtual Machine Manager (SCVMM) to automate the Cluster OS Rolling Upgrade process?

Yes, you can automate the Cluster OS Rolling Upgrade process using VMM in System Center 2016.

See also

- [Release Notes: Important Issues in Windows Server 2016](#)
- [What's New in Windows Server 2016](#)
- [What's New in Failover Clustering in Windows Server](#)

Cluster-Aware Updating overview

4/30/2018 • 7 minutes to read • [Edit Online](#)

Applies to: Windows Server (Semi-Annual Channel), Windows Server 2016, Windows Server 2012 R2, Windows Server 2012

This topic provides an overview of Cluster-Aware Updating (CAU), a feature that automates the software updating process on clustered servers while maintaining availability.

NOTE

When updating [Storage Spaces Direct](#) clusters, we recommend using Cluster-Aware Updating.

Feature description

Cluster-Aware Updating is an automated feature that enables you to update servers in a [failover cluster](#) with little or no loss in availability during the update process. During an Updating Run, Cluster-Aware Updating transparently performs the following tasks:

1. Puts each node of the cluster into node maintenance mode.
2. Moves the clustered roles off the node.
3. Installs the updates and any dependent updates.
4. Performs a restart if necessary.
5. Brings the node out of maintenance mode.
6. Restores the clustered roles on the node.
7. Moves to update the next node.

For many clustered roles in the cluster, the automatic update process triggers a planned failover. This can cause a transient service interruption for connected clients. However, in the case of continuously available workloads, such as Hyper-V with live migration or file server with SMB Transparent Failover, Cluster-Aware Updating can coordinate cluster updates with no impact to the service availability.

Practical applications

- CAU reduces service outages in clustered services, reduces the need for manual updating workarounds, and makes the end-to-end cluster updating process more reliable for the administrator. When the CAU feature is used in conjunction with continuously available cluster workloads, such as continuously available file servers (file server workload with SMB Transparent Failover) or Hyper-V, the cluster updates can be performed with zero impact to service availability for clients.
- CAU facilitates the adoption of consistent IT processes across the enterprise. Updating Run Profiles can be created for different classes of failover clusters and then managed centrally on a file share to ensure that CAU deployments throughout the IT organization apply updates consistently, even if the clusters are managed by different lines-of-business or administrators.
- CAU can schedule Updating Runs on regular daily, weekly, or monthly intervals to help coordinate cluster updates with other IT management processes.
- CAU provides an extensible architecture to update the cluster software inventory in a cluster-aware

fashion. This can be used by publishers to coordinate the installation of software updates that are not published to Windows Update or Microsoft Update or that are not available from Microsoft, for example, updates for non-Microsoft device drivers.

- CAU self-updating mode enables a "cluster in a box" appliance (a set of clustered physical machines, typically packaged in one chassis) to update itself. Typically, such appliances are deployed in branch offices with minimal local IT support to manage the clusters. Self-updating mode offers great value in these deployment scenarios.

Important functionality

The following is a description of important Cluster-Aware Updating functionality:

- A user interface (UI) - the Cluster Aware Updating window - and a set of cmdlets that you can use to preview, apply, monitor, and report on the updates
- An end-to-end automation of the cluster-updating operation (an Updating Run), orchestrated by one or more Update Coordinator computers
- A default plug-in that integrates with the existing Windows Update Agent (WUA) and Windows Server Update Services (WSUS) infrastructure in Windows Server to apply important Microsoft updates
- A second plug-in that can be used to apply Microsoft hotfixes, and that can be customized to apply non-Microsoft updates
- Updating Run Profiles that you configure with settings for Updating Run options, such as the maximum number of times that the update will be retried per node. Updating Run Profiles enable you to rapidly reuse the same settings across Updating Runs and easily share the update settings with other failover clusters.
- An extensible architecture that supports new plug-in development to coordinate other node-updating tools across the cluster, such as custom software installers, BIOS updating tools, and network adapter or host bus adapter (HBA) updating tools.

Cluster-Aware Updating can coordinate the complete cluster updating operation in two modes:

- **Self-updating mode** For this mode, the CAU clustered role is configured as a workload on the failover cluster that is to be updated, and an associated update schedule is defined. The cluster updates itself at scheduled times by using a default or custom Updating Run profile. During the Updating Run, the CAU Update Coordinator process starts on the node that currently owns the CAU clustered role, and the process sequentially performs updates on each cluster node. To update the current cluster node, the CAU clustered role fails over to another cluster node, and a new Update Coordinator process on that node assumes control of the Updating Run. In self-updating mode, CAU can update the failover cluster by using a fully automated, end-to-end updating process. An administrator can also trigger updates on-demand in this mode, or simply use the remote-updating approach if desired. In self-updating mode, an administrator can get summary information about an Updating Run in progress by connecting to the cluster and running the **Get-CauRun** Windows PowerShell cmdlet.
- **Remote-updating mode** For this mode, a remote computer, which is called an Update Coordinator, is configured with the CAU tools. The Update Coordinator is not a member of the cluster that is updated during the Updating Run. From the remote computer, the administrator triggers an on-demand Updating Run by using a default or custom Updating Run profile. Remote-updating mode is useful for monitoring real-time progress during the Updating Run, and for clusters that are running on Server Core installations.

Hardware and software requirements

CAU can be used on all editions of Windows Server, including Server Core installations. For detailed

requirements information, see [Cluster-Aware Updating requirements and best practices](#).

Installing Cluster-Aware Updating

To use CAU, install the Failover Clustering feature in Windows Server and create a failover cluster. The components that support CAU functionality are automatically installed on each cluster node.

To install the Failover Clustering feature, you can use the following tools:

- Add Roles and Features Wizard in Server Manager
- [Install-WindowsFeature](#) Windows PowerShell cmdlet
- Deployment Image Servicing and Management (DISM) command-line tool

For more information, see [Install or Uninstall Roles, Role Services, or Features](#).

You must also install the Failover Clustering Tools, which are part of the Remote Server Administration Tools and are installed by default when you install the Failover Clustering feature in Server Manager. The Failover Clustering tools include the Cluster-Aware Updating user interface and PowerShell cmdlets.

You must install the Failover Clustering Tools as follows to support the different CAU updating modes:

- To use CAU in self-updating mode, install the Failover Clustering Tools on each cluster node.
- To enable remote-updating mode, install the Failover Clustering Tools on a computer that has network connectivity to the failover cluster.

NOTE

- You can't use the Failover Clustering Tools on Windows Server 2012 to manage Cluster-Aware Updating on a newer version of Windows Server.
- To use CAU only in remote-updating mode, installation of the Failover Clustering Tools on the cluster nodes is not required. However, certain CAU features will not be available. For more information, see [Requirements and Best Practices for Cluster-Aware Updating](#).
- Unless you are using CAU only in self-updating mode, the computer on which the CAU tools are installed and that coordinates the updates cannot be a member of the failover cluster.

Enabling self-updating mode

To enable the self-updating mode, you must add the Cluster-Aware Updating clustered role to the failover cluster. To do so, use one of the following methods:

- In Server Manager, select **Tools > Cluster-Aware Updating**, then in the Cluster-Aware Updating window, select **Configure cluster self-updating options**.
- In a PowerShell session, run the [Add-CauClusterRole](#) cmdlet.

To uninstall CAU, uninstall the Failover Clustering feature or Failover Clustering Tools by using Server Manager, the [Uninstall-WindowsFeature](#) cmdlet, or the DISM command-line tools.

Additional requirements and best practices

To ensure that CAU can update the cluster nodes successfully, and for additional guidance for configuring your failover cluster environment to use CAU, you can run the CAU Best Practices Analyzer.

For detailed requirements and best practices for using CAU, and information about running the CAU Best Practices Analyzer, see [Requirements and Best Practices for Cluster-Aware Updating](#).

Starting Cluster-Aware Updating

To start Cluster-Aware Updating from Server Manager

1. Start Server Manager.

2. Do one of the following:

- On the **Tools** menu, click **Cluster-Aware Updating**.
- If one or more cluster nodes, or the cluster, is added to Server Manager, on the **All Servers** page, right-click the name of a node (or the name of the cluster), and then click **Update Cluster**.

See also

The following links provide more information about using Cluster-Aware Updating.

- [Requirements and Best Practices for Cluster-Aware Updating](#)
- [Cluster-Aware Updating: Frequently Asked Questions](#)
- [Advanced Options and Updating Run Profiles for CAU](#)
- [How CAU Plug-ins Work](#)
- [Cluster-Aware Updating Cmdlets in Windows PowerShell](#)
- [Cluster-Aware Updating Plug-in Reference](#)

Cluster-Aware Updating requirements and best practices

4/30/2018 • 20 minutes to read • [Edit Online](#)

Applies to: Windows Server (Semi-Annual Channel), Windows Server 2016, Windows Server 2012 R2, Windows Server 2012

This section describes the requirements and dependencies that are needed to use [Cluster-Aware Updating](#) (CAU) to apply updates to a failover cluster running Windows Server.

NOTE

You may need to independently validate that your cluster environment is ready to apply updates if you use a plug-in other than **Microsoft.WindowsUpdatePlugin**. If you are using a non-Microsoft plug-in, contact the publisher for more information. For more information about plug-ins, see [How Plug-ins Work](#).

Install the Failover Clustering feature and the Failover Clustering Tools

CAU requires an installation of the Failover Clustering feature and the Failover Clustering Tools. The Failover Clustering Tools include the CAU tools (clusterawareupdating.dll), the Failover Clustering cmdlets, and other components needed for CAU operations. For steps to install the Failover Clustering feature, see [Installing the Failover Clustering Feature and Tools](#).

The exact installation requirements for the Failover Clustering Tools depend on whether CAU coordinates updates as a clustered role on the failover cluster (by using self-updating mode) or from a remote computer. The self-updating mode of CAU additionally requires the installation of the CAU clustered role on the failover cluster by using the CAU tools.

The following table summarizes the CAU feature installation requirements for the two CAU updating modes.

INSTALLED COMPONENT	SELF-UPDATING MODE	REMOTE-UPDATING MODE
Failover Clustering feature	Required on all cluster nodes	Required on all cluster nodes
Failover Clustering Tools	Required on all cluster nodes	<ul style="list-style-type: none">- Required on remote-updating computer- Required on all cluster nodes to run the Save-CauDebugTrace cmdlet
CAU clustered role	Required	Not required

Obtain an administrator account

The following administrator requirements are necessary to use CAU features.

- To preview or apply update actions by using the CAU user interface (UI) or the Cluster-Aware Updating cmdlets, you must use a domain account that has local administrator rights and permissions on all the cluster nodes. If the account doesn't have sufficient privileges on every node, you are prompted in the Cluster-Aware Updating window to supply the necessary credentials when you perform these actions. To

use the [Cluster-Aware Updating](#) cmdlets, you can supply the necessary credentials as a cmdlet parameter.

- If you use CAU in remote-updating mode when you are signed in with an account that doesn't have local administrator rights and permissions on the cluster nodes, you must run the CAU tools as an administrator by using a local administrator account on the Update Coordinator computer, or by using an account that has the **Impersonate a client after authentication** user right.
- To run the CAU Best Practices Analyzer, you must use an account that has administrative privileges on the cluster nodes and local administrative privileges on the computer that is used to run the [Test-CauSetup](#) cmdlet or to analyze cluster updating readiness using the Cluster-Aware Updating window. For more information, see [Test cluster updating readiness](#).

Verify the cluster configuration

The following are general requirements for a failover cluster to support updates by using CAU. Additional configuration requirements for remote management on the nodes are listed in [Configure the nodes for remote management](#) later in this topic.

- Sufficient cluster nodes must be online so that the cluster has quorum.
- All cluster nodes must be in the same Active Directory domain.
- The cluster name must be resolved on the network using DNS.
- If CAU is used in remote-updating mode, the Update Coordinator computer must have network connectivity to the failover cluster nodes, and it must be in the same Active Directory domain as the failover cluster.
- The Cluster service should be running on all cluster nodes. By default this service is installed on all cluster nodes and is configured to start automatically.
- To use PowerShell pre-update or post-update scripts during a CAU Updating Run, ensure that the scripts are installed on all cluster nodes or that they are accessible to all nodes, for example, on a highly available network file share. If scripts are saved to a network file share, configure the folder for Read permission for the Everyone group.

Configure the nodes for remote management

To use Cluster-Aware Updating, all nodes of the cluster must be configured for remote management. By default, the only task you must perform to configure the nodes for remote management is to [Enable a firewall rule to allow automatic restarts](#).

The following table lists the complete remote management requirements, in case your environment diverges from the defaults.

These requirements are in addition to the installation requirements for the [Install the Failover Clustering feature and the Failover Clustering Tools](#) and the general clustering requirements that are described in previous sections in this topic.

REQUIREMENT	DEFAULT STATE	SELF-UPDATING MODE	REMOTE-UPDATING MODE
Enable a firewall rule to allow automatic restarts	Disabled	Required on all cluster nodes if a firewall is in use	Required on all cluster nodes if a firewall is in use
Enable Windows Management Instrumentation	Enabled	Required on all cluster nodes	Required on all cluster nodes

REQUIREMENT	DEFAULT STATE	SELF-UPDATING MODE	REMOTE-UPDATING MODE
Enable Windows PowerShell 3.0 or 4.0 and Windows PowerShell remoting	Enabled	Required on all cluster nodes	Required on all cluster nodes to run the following: <ul style="list-style-type: none"> - The Save-CauDebugTrace cmdlet - PowerShell pre-update and post-update scripts during an Updating Run - Tests of cluster updating readiness using the Cluster-Aware Updating window or the Test-CauSetup Windows PowerShell cmdlet
Install .NET Framework 4.6 or 4.5	Enabled	Required on all cluster nodes	Required on all cluster nodes to run the following: <ul style="list-style-type: none"> - The Save-CauDebugTrace cmdlet - PowerShell pre-update and post-update scripts during an Updating Run - Tests of cluster updating readiness using the Cluster-Aware Updating window or the Test-CauSetup Windows PowerShell cmdlet

Enable a firewall rule to allow automatic restarts

To allow automatic restarts after updates are applied (if the installation of an update requires a restart), if Windows Firewall or a non-Microsoft firewall is in use on the cluster nodes, a firewall rule must be enabled on each node that allows the following traffic:

- Protocol: TCP
- Direction: inbound
- Program: wininit.exe
- Ports: RPC Dynamic Ports
- Profile: Domain

If Windows Firewall is used on the cluster nodes, you can do this by enabling the **Remote Shutdown** Windows Firewall rule group on each cluster node. When you use the Cluster-Aware Updating window to apply updates and to configure self-updating options, the **Remote Shutdown** Windows Firewall rule group is automatically enabled on each cluster node.

NOTE

The **Remote Shutdown** Windows Firewall rule group cannot be enabled when it will conflict with Group Policy settings that are configured for Windows Firewall.

The **Remote Shutdown** firewall rule group is also enabled by specifying the **-EnableFirewallRules** parameter when running the following CAU cmdlets: [Add-CauClusterRole](#), [Invoke-CauRun](#), and [SetCauClusterRole](#).

The following PowerShell example shows an additional method to enable automatic restarts on a cluster node.

```
Set-NetFirewallRule -Group "@firewallapi.dll,-36751" -Profile Domain -Enabled true
```

Enable Windows Management Instrumentation (WMI)

All cluster nodes must be configured for remote management using Windows Management Instrumentation (WMI). This is enabled by default.

To manually enable remote management, do the following:

1. In the Services console, start the **Windows Remote Management** service and set the startup type to **Automatic**.
2. Run the [Set-WSManQuickConfig](#) cmdlet, or run the following command from an elevated command prompt:

```
winrm quickconfig -q
```

To support WMI remoting, if Windows Firewall is in use on the cluster nodes, the inbound firewall rule for **Windows Remote Management (HTTP-In)** must be enabled on each node. By default, this rule is enabled.

Enable Windows PowerShell and Windows PowerShell remoting

To enable self-updating mode and certain CAU features in remote-updating mode, PowerShell must be installed and enabled to run remote commands on all cluster nodes. By default, PowerShell is installed and enabled for remoting.

To enable PowerShell remoting, use one of the following methods:

- Run the [Enable-PSRemoting](#) cmdlet.
- Configure a domain-level Group Policy setting for Windows Remote Management (WinRM).

For more information about enabling PowerShell remoting, see [about_Remote_Requirements](#).

Install .NET Framework 4.6 or 4.5

To enable self-updating mode and certain CAU features in remote-updating mode,.NET Framework 4.6, or .NET Framework 4.5 (on Windows Server 2012 R2) must be installed on all cluster nodes. By default, NET Framework is installed.

To install .NET Framework 4.6 (or 4.5) using PowerShell if it's not already installed, use the following command:

```
Install-WindowsFeature -Name NET-Framework-45-Core
```

Best practices recommendations for using Cluster-Aware Updating

Recommendations for applying Microsoft updates

We recommend that when you begin to use CAU to apply updates with the default

Microsoft.WindowsUpdatePlugin plug-in on a cluster, you stop using other methods to install software updates from Microsoft on the cluster nodes.

Caution

Combining CAU with methods that update individual nodes automatically (on a fixed time schedule) can cause unpredictable results, including interruptions in service and unplanned downtime.

We recommend that you follow these guidelines:

- For optimal results, we recommend that you disable settings on the cluster nodes for automatic updating,

for example, through the Automatic Updates settings in Control Panel, or in settings that are configured using Group Policy.

Caution

Automatic installation of updates on the cluster nodes can interfere with installation of updates by CAU and can cause CAU failures.

If they are needed, the following Automatic Updates settings are compatible with CAU, because the administrator can control the timing of update installation:

- Settings to notify before downloading updates and to notify before installation
- Settings to automatically download updates and to notify before installation

However, if Automatic Updates is downloading updates at the same time as a CAU Updating Run, the Updating Run might take longer to complete.

- Do not configure an update system such as Windows Server Update Services (WSUS) to apply updates automatically (on a fixed time schedule) to cluster nodes.
- All cluster nodes should be uniformly configured to use the same update source, for example, a WSUS server, Windows Update, or Microsoft Update.
- If you use a configuration management system to apply software updates to computers on the network, exclude cluster nodes from all required or automatic updates. Examples of configuration management systems include Microsoft System Center Configuration Manager 2007 and Microsoft System Center Virtual Machine Manager 2008.
- If internal software distribution servers (for example, WSUS servers) are used to contain and deploy the updates, ensure that those servers correctly identify the approved updates for the cluster nodes.

Apply Microsoft updates in branch office scenarios

To download Microsoft updates from Microsoft Update or Windows Update to cluster nodes in certain branch office scenarios, you may need to configure proxy settings for the Local System account on each node. For example, you might need to do this if your branch office clusters access Microsoft Update or Windows Update to download updates by using a local proxy server.

If necessary, configure WinHTTP proxy settings on each node to specify a local proxy server and configure local address exceptions (that is, a bypass list for local addresses). To do this, you can run the following command on each cluster node from an elevated command prompt:

```
netsh winhttp set proxy <ProxyServerFQDN>:<port> "<local>"
```

where *<ProxyServerFQDN>* is the fully qualified domain name for the proxy server and *<port>* is the port over which to communicate (usually port 443).

For example, to configure WinHTTP proxy settings for the Local System account specifying the proxy server *MyProxy.CONTOSO.com*, with port 443 and local address exceptions, type the following command:

```
netsh winhttp set proxy MyProxy.CONTOSO.com:443 "<local>"
```

Recommendations for using the Microsoft.HotfixPlugin

- We recommend that you configure permissions in the hotfix root folder and hotfix configuration file to restrict Write access to only local administrators on the computers that are used to store these files. This helps prevent tampering with these files by unauthorized users that could compromise the functionality of the failover cluster when hotfixes are applied.

- To help ensure data integrity for the server message block (SMB) connections that are used to access the hotfix root folder, you should configure SMB Encryption in the SMB shared folder, if it is possible to configure it. The **Microsoft.HotfixPlugin** requires that SMB signing or SMB Encryption is configured to help ensure data integrity for the SMB connections.

For more information, see [Restrict access to the hotfix root folder and hotfix configuration file](#).

Additional recommendations

- To avoid interfering with a CAU Updating Run that may be scheduled at the same time, do not schedule password changes for cluster name objects and virtual computer objects during scheduled maintenance windows.
- You should set appropriate permissions on pre-update and post-update scripts that are saved on network shared folders to prevent potential tampering with these files by unauthorized users.
- To configure CAU in self-updating mode, a virtual computer object (VCO) for the CAU clustered role must be created in Active Directory. CAU can create this object automatically at the time that the CAU clustered role is added, if the failover cluster has sufficient permissions. However, because of the security policies in certain organizations, it may be necessary to prestage the object in Active Directory. For a procedure to do this, see [Steps for prestaging an account for a clustered role](#).
- To save and reuse Updating Run settings across failover clusters with similar updating needs in the IT organization, you can create Updating Run Profiles. Additionally, depending on the updating mode, you can save and manage the Updating Run Profiles on a file share that is accessible to all remote Update Coordinator computers or failover clusters. For more information, see [Advanced Options and Updating Run Profiles for CAU](#).

Test cluster updating readiness

You can run the CAU Best Practices Analyzer (BPA) model to test whether a failover cluster and the network environment meet many of the requirements to have software updates applied by CAU. Many of the tests check the environment for readiness to apply Microsoft updates by using the default plug-in,

Microsoft.WindowsUpdatePlugin.

NOTE

You might need to independently validate that your cluster environment is ready to apply software updates by using a plug-in other than **Microsoft.WindowsUpdatePlugin**. If you are using a non-Microsoft plug-in, such as one provided by your hardware manufacturer, contact the publisher for more information.

You can run the BPA in the following two ways:

1. Select **Analyze cluster updating readiness** in the CAU console. After the BPA completes the readiness tests, a test report appears. If issues are detected on cluster nodes, the specific issues and the nodes where the issues appear are identified so that you can take corrective action. The tests can take several minutes to complete.
2. Run the [Test-CauSetup](#) cmdlet. You can run the cmdlet on a local or remote computer on which the Failover Clustering Module for Windows PowerShell (part of the Failover Clustering Tools) is installed. You can also run the cmdlet on a node of the failover cluster.

NOTE

- You must use an account that has administrative privileges on the cluster nodes and local administrative privileges on the computer that is used to run the **Test-CauSetup** cmdlet or to analyze cluster updating readiness using the Cluster-Aware Updating window. To run the tests using the Cluster-Aware Updating window, you must be logged on to the computer with the necessary credentials.
- The tests assume that the CAU tools that are used to preview and apply software updates run from the same computer and with the same user credentials as are used to test cluster updating readiness.

IMPORTANT

We highly recommend that you test the cluster for updating readiness in the following situations:

- Before you use CAU for the first time to apply software updates.
- After you add a node to the cluster or perform other hardware changes in the cluster that require running the Validate a Cluster Wizard.
- After you change an update source, or change update settings or configurations (other than CAU) that can affect the application of updates on the nodes.

Tests for cluster updating readiness

The following table lists the cluster updating readiness tests, some common issues, and resolution steps.

TEST	POSSIBLE ISSUES AND IMPACTS	RESOLUTION STEPS
The failover cluster must be available	Cannot resolve the failover cluster name, or one or more cluster nodes cannot be accessed. The BPA cannot run the cluster readiness tests.	<ul style="list-style-type: none">- Check the spelling of the name of the cluster specified during the BPA run.- Ensure that all nodes of the cluster are online and running.- Check that the Validate a Configuration Wizard can successfully run on the failover cluster.
The failover cluster nodes must be enabled for remote management via WMI	One or more failover cluster nodes are not enabled for remote management by using Windows Management Instrumentation (WMI). CAU cannot update the cluster nodes if the nodes are not configured for remote management.	Ensure that all failover cluster nodes are enabled for remote management through WMI. For more information, see Configure the nodes for remote management in this topic.
PowerShell remoting should be enabled on each failover cluster node	PowerShell isn't installed or isn't enabled for remoting on one or more failover cluster nodes. CAU cannot be configured for self-updating mode or use certain features in remote-updating mode.	<p>Ensure that PowerShell is installed on all cluster nodes and is enabled for remoting.</p> <p>For more information, see Configure the nodes for remote management in this topic.</p>
Failover cluster version	One or more nodes in the failover cluster don't run Windows Server 2016, Windows Server 2012 R2, or Windows Server 2012. CAU cannot update the failover cluster.	<p>Verify that the failover cluster that is specified during the BPA run is running Windows Server 2016, Windows Server 2012 R2, or Windows Server 2012.</p> <p>For more information, see Verify the cluster configuration in this topic.</p>

TEST	POSSIBLE ISSUES AND IMPACTS	RESOLUTION STEPS
The required versions of .NET Framework and Windows PowerShell must be installed on all failover cluster nodes	.NET Framework 4.6, 4.5 or Windows PowerShell isn't installed on one or more cluster nodes. Some CAU features might not work.	<p>Ensure that .NET Framework 4.6 or 4.5 and Windows PowerShell are installed on all cluster nodes, if they are required.</p> <p>For more information, see Configure the nodes for remote management in this topic.</p>
The Cluster service should be running on all cluster nodes	The Cluster service is not running on one or more nodes. CAU cannot update the failover cluster.	<ul style="list-style-type: none"> - Ensure that the Cluster service (clussvc) is started on all nodes in the cluster, and it is configured to start automatically. - Check that the Validate a Configuration Wizard can successfully run on the failover cluster. <p>For more information, see Verify the cluster configuration in this topic.</p>
Automatic Updates must not be configured to automatically install updates on any failover cluster node	On at least one failover cluster node, Automatic Updates is configured to automatically install Microsoft updates on that node. Combining CAU with other update methods can result in unplanned downtime or unpredictable results.	<p>If Windows Update functionality is configured for Automatic Updates on one or more cluster nodes, ensure that Automatic Updates is not configured to automatically install updates.</p> <p>For more information, see Recommendations for applying Microsoft updates.</p>
The failover cluster nodes should use the same update source	One or more failover cluster nodes are configured to use an update source for Microsoft updates that is different from the rest of the nodes. Updates might not be applied uniformly on the cluster nodes by CAU.	<p>Ensure that every cluster node is configured to use the same update source, for example, a WSUS server, Windows Update, or Microsoft Update.</p> <p>For more information, see Recommendations for applying Microsoft updates.</p>
A firewall rule that allows remote shutdown should be enabled on each node in the failover cluster	One or more failover cluster nodes do not have a firewall rule enabled that allows remote shutdown, or a Group Policy setting prevents this rule from being enabled. An Updating Run that applies updates that require restarting the nodes automatically might not complete properly.	<p>If Windows Firewall or a non-Microsoft firewall is in use on the cluster nodes, configure a firewall rule that allows remote shutdown.</p> <p>For more information, see Enable a firewall rule to allow automatic restarts in this topic.</p>
The proxy server setting on each failover cluster node should be set to a local proxy server	<p>One or more failover cluster nodes have an incorrect proxy server configuration.</p> <p>If a local proxy server is in use, the proxy server setting on each node must be configured properly for the cluster to access Microsoft Update or Windows Update.</p>	<p>Ensure that the WinHTTP proxy settings on each cluster node are set to a local proxy server if it is needed. If a proxy server is not in use in your environment, this warning can be ignored.</p> <p>For more information, see Apply updates in branch office scenarios in this topic.</p>

TEST	POSSIBLE ISSUES AND IMPACTS	RESOLUTION STEPS
The CAU clustered role should be installed on the failover cluster to enable self-updating mode	The CAU clustered role is not installed on this failover cluster. This role is required for cluster self-updating.	<p>To use CAU in self-updating mode, add the CAU clustered role on the failover cluster in one of the following ways:</p> <ul style="list-style-type: none"> - Run the Add-CauClusterRole PowerShell cmdlet. - Select the Configure cluster self-updating options action in the Cluster-Aware Updating window.
The CAU clustered role should be enabled on the failover cluster to enable self-updating mode	The CAU clustered role is disabled. For example, the CAU clustered role is not installed, or it has been disabled by using the Disable-CauClusterRole PowerShell cmdlet. This role is required for cluster self-updating.	<p>To use CAU in self-updating mode, enable the CAU clustered role on this failover cluster in one of the following ways:</p> <ul style="list-style-type: none"> - Run the Enable-CauClusterRole PowerShell cmdlet. - Select the Configure cluster self-updating options action in the Cluster-Aware Updating window.
The configured CAU plug-in for self-updating mode must be registered on all failover cluster nodes	The CAU clustered role on one or more nodes of this failover cluster cannot access the CAU plug-in module that is configured in the self-updating options. A self-updating run might fail.	<ul style="list-style-type: none"> - Ensure that the configured CAU plug-in is installed on all cluster nodes by following the installation procedure for the product that supplies the CAU plug-in. - Run the Register-CauPlugin PowerShell cmdlet to register the plug-in on the required cluster nodes.
All failover cluster nodes should have the same set of registered CAU plug-ins	A self-updating run might fail if the plug-in that is configured to be used in an Updating Run is changed to one that is not available on all cluster nodes.	<ul style="list-style-type: none"> - Ensure that the configured CAU plug-in is installed on all cluster nodes by following the installation procedure for the product that supplies the CAU plug-in. - Run the Register-CauPlugin PowerShell cmdlet to register the plug-in on the required cluster nodes.
The configured Updating Run options must be valid	The self-updating schedule and Updating Run options that are configured for this failover cluster are incomplete or are not valid. A self-updating run might fail.	Configure a valid self-updating schedule and set of Updating Run options. For example, you can use the Set-CauClusterRole PowerShell cmdlet to configure the CAU clustered role.
At least two failover cluster nodes must be owners of the CAU clustered role	An Updating Run launched in self-updating mode will fail because the CAU clustered role does not have a possible owner node to move to.	Use the Failover Clustering Tools to ensure that all cluster nodes are configured as possible owners of the CAU clustered role. This is the default configuration.
All failover cluster nodes must be able to access Windows PowerShell scripts	Not all possible owner nodes of the CAU clustered role can access the configured Windows PowerShell pre-update and post-update scripts. A self-updating run will fail.	Ensure that all possible owner nodes of the CAU clustered role have permissions to access the configured PowerShell pre-update and post-update scripts.

TEST	POSSIBLE ISSUES AND IMPACTS	RESOLUTION STEPS
All failover cluster nodes should use identical Windows PowerShell scripts	Not all possible owner nodes of the CAU clustered role use the same copy of the specified Windows PowerShell pre-update and post-update scripts. A self-updating run might fail or show unexpected behavior.	Ensure that all possible owner nodes of the CAU clustered role use the same PowerShell pre-update and post-update scripts.
The WarnAfter setting specified for the Updating Run should be less than the StopAfter setting	The specified CAU Updating Run timeout values make the warning timeout ineffective. An Updating Run might be canceled before a warning event log can be generated.	In the Updating Run options, configure a WarnAfter option value that is less than the StopAfter option value.

See also

- [Cluster-Aware Updating overview](#)

Cluster-Aware Updating advanced options and updating run profiles

4/30/2018 • 8 minutes to read • [Edit Online](#)

Applies to: Windows Server (Semi-Annual Channel), Windows Server 2016, Windows Server 2012 R2, Windows Server 2012

This topic describes Updating Run options that can be configured for a [Cluster-Aware Updating](#) (CAU) Updating Run. These advanced options can be configured when you use either the CAU UI or the CAU Windows PowerShell cmdlets to apply updates or to configure self-updating options.

Most configuration settings can be saved as an XML file called an Updating Run Profile and reused for later Updating Runs. The default values for the Updating Run options that are provided by CAU can also be used in many cluster environments.

For information about additional options that you can specify for each Updating Run and about Updating Run Profiles, see the following sections later in this topic:

Options that you specify when you request an Updating Run Use Updating Run Profiles Options that can be set in an Updating Run Profile

The following table lists options that you can set in a CAU Updating Run Profile.

NOTE

To set the PreUpdateScript or PostUpdateScript option, ensure that Windows PowerShell and .NET Framework 4.6 or 4.5 are installed and that PowerShell remoting is enabled on each node in the cluster. For more information, see [Configure the nodes for remote management in Requirements and Best Practices for Cluster-Aware Updating](#).

OPTION	DEFAULT VALUE	DETAILS
StopAfter	Unlimited time	Time in minutes after which the Updating Run will be stopped if it has not completed. Note: If you specify a pre-update or a post-update PowerShell script, the entire process of running scripts and performing updates must be complete within the StopAfter time limit.
WarnAfter	By default, no warning appears	Time in minutes after which a warning will appear if the Updating Run (including a pre-update script and a post-update script, if they are configured) has not completed.
MaxRetriesPerNode	3	Maximum number of times that the update process (including a pre-update script and a post-update script, if they are configured) will be retried per node. The maximum is 64.

OPTION	DEFAULT VALUE	DETAILS
MaxFailedNodes	For most clusters, an integer that is approximately one-third of the number of cluster nodes	<p>Maximum number of nodes on which updating can fail, either because the nodes fail or the Cluster service stops running. If one more node fails, the Updating Run is stopped.</p> <p>The valid range of values is 0 to 1 less than the number of cluster nodes.</p>
RequireAllNodesOnline	None	Specifies that all nodes must be online and reachable before updating begins.
RebootTimeoutMinutes	15	Time in minutes that CAU will allow for restarting a node (if a restart is necessary) and starting all auto-start services. If the restart process doesn't complete within this time, the Updating Run on that node is marked as failed.
PreUpdateScript	None	<p>The path and file name for a PowerShell script to run on each node before updating begins, and before the node is put into maintenance mode. The file name extension must be .ps1, and the total length of the path plus file name must not exceed 260 characters. As a best practice, the script should be located on a disk in cluster storage, or at a highly available network file share, to ensure that it is always accessible to all of the cluster nodes. If the script is located on a network file share, ensure that you configure the file share for Read permission for the Everyone group, and restrict write access to prevent tampering with the files by unauthorized users.</p> <p>If you specify a pre-update script, be sure that settings such as the time limits (for example, StopAfter) are configured to allow the script to run successfully. These limits span the entire process of running scripts and installing updates, not just the process of installing updates.</p>

OPTION	DEFAULT VALUE	DETAILS
PostUpdateScript	None	<p>The path and file name for a PowerShell script to run after updating completes (after the node leaves maintenance mode). The file name extension must be .ps1 and the total length of the path plus file name must not exceed 260 characters. As a best practice, the script should be located on a disk in cluster storage, or at a highly available network file share, to ensure that it is always accessible to all of the cluster nodes. If the script is located on a network file share, ensure that you configure the file share for Read permission for the Everyone group, and restrict write access to prevent tampering with the files by unauthorized users.</p> <p>If you specify a post-update script, be sure that settings such as the time limits (for example, StopAfter) are configured to allow the script to run successfully. These limits span the entire process of running scripts and installing updates, not just the process of installing updates.</p>
ConfigurationName	<p>This setting only has an effect if you run scripts.</p> <p>If you specify a pre-update script or a post-update script, but you do not specify a ConfigurationName, the default session configuration for PowerShell (Microsoft.PowerShell) is used.</p>	<p>Specifies the PowerShell session configuration that defines the session in which scripts (specified by PreUpdateScript and PostUpdateScript) are run, and can limit the commands that can be run.</p>
CauPluginName	Microsoft.WindowsUpdatePlugin	<p>Plug-in that you configure Cluster-Aware Updating to use to preview updates or perform an Updating Run. For more information, see How Cluster-Aware Updating plug-ins work.</p>

OPTION	DEFAULT VALUE	DETAILS
CauPluginArguments	None	<p>A set of <i>name=value</i> pairs (arguments) for the updating plug-in to use, for example:</p> <p>Domain=Domain.local</p> <p>These <i>name=value</i> pairs must be meaningful to the plug-in that you specify in CauPluginName.</p> <p>To specify an argument using the CAU UI, type the <i>name</i>, press the Tab key, and then type the corresponding <i>value</i>. Press the Tab key again to provide the next argument. Each <i>name</i> and <i>value</i> are automatically separated with an equal (=) sign. Multiple pairs are automatically separated with semicolons.</p> <p>For the default Microsoft.WindowsUpdatePlugin plug-in, no arguments are needed. However, you can specify an optional argument, for example to specify a standard Windows Update Agent query string to filter the set of updates that are applied by the plug-in. For a <i>name</i>, use QueryString, and for a <i>value</i>, enclose the full query in quotation marks.</p> <p>For more information, see How Cluster-Aware Updating plug-ins work.</p>

Options that you specify when you request an Updating Run

The following table lists options (other than those in an Updating Run Profile) that you can specify when you request an Updating Run. For information about options that you can set in an Updating Run Profile, see the preceding table.

OPTION	DEFAULT VALUE	DETAILS
ClusterName	<p>None</p> <p>Note: This option must be set only when the CAU UI is not run on a failover cluster node, or you want to reference a failover cluster different from where the CAU UI is run.</p>	NetBIOS name of the cluster on which to perform the Updating Run.

OPTION	DEFAULT VALUE	DETAILS
Credential	Current account credentials	Administrative credentials for the target cluster on which the Updating Run will be performed. You may already have the necessary credentials if you start the CAU UI (or open a PowerShell session, if you're using the CAU PowerShell cmdlets) from an account that has administrator rights and permissions on the cluster.
NodeOrder	By default, CAU starts with the node that owns the smallest number of clustered roles, then progresses to the node that has the second smallest number, and so on.	Names of the cluster nodes in the order that they should be updated (if possible).

Use Updating Run Profiles

Each Updating Run can be associated with a specific Updating Run Profile. The default Updating Run Profile is stored in the `%windir%\cluster` folder. If you're using the CAU UI in remote-updating mode, you can specify an Updating Run Profile at the time that you apply updates, or you can use the default Updating Run profile. If you're using CAU in self-updating mode, you can import the settings from a specified Updating Run Profile when you configure the self-updating options. In both cases, you can override the displayed values for the Updating Run options according to your needs. If you want, you can save the Updating Run options as an Updating Run Profile with the same file name or a different file name. The next time that you apply updates or configure self-updating options, CAU automatically selects the Updating Run Profile that was previously selected.

You can modify an existing Updating Run Profile or create a new one by selecting **Create or modify Updating Run Profile** in the CAU UI.

Here are some important notes about using Updating Run Profiles:

- An Updating Run Profile doesn't store cluster-specific information such as administrative credentials. If you're using CAU in self-updating mode, the Updating Run Profile also doesn't store the self-updating schedule information. This makes it possible to share an Updating Run Profile across all failover clusters in a specified class.
- If you configure self-updating options using an Updating Run Profile and later modify the profile with different values for the Updating Run options, the self-updating configuration doesn't change automatically. To apply the new Updating Run settings, you must configure the self-updating options again.
- The Run Profile Editor unfortunately doesn't support file paths that include spaces, such as `C:\Program Files`. As a workaround, store your pre and post update scripts in a path that doesn't include spaces, or use PowerShell exclusively to manage Run Profiles, putting quotes around the path when running **Invoke-CauRun**.

Windows PowerShell equivalent commands

You can import the settings from an Updating Run Profile when you run the **Invoke-CauRun**, **Add-CauClusterRole**, or **Set-CauClusterRole** cmdlet.

The following example performs a scan and a full Updating Run on the cluster named *CONTOSO-FC1*, using the Updating Run options that are specified in `C:\Windows\Cluster\DefaultParameters.xml`. Default values are used for the remaining cmdlet parameters.

```
$MyRunProfile = Import-Clixml C:\Windows\Cluster\DefaultParameters.xml  
Invoke-CauRun -ClusterName CONTOSO-FC1 @MyRunProfile
```

By using an Updating Run Profile, you can update a failover cluster in a repeatable fashion with consistent settings for exception management, time bounds, and other operational parameters. Because these settings are typically specific to a class of failover clusters—such as “All Microsoft SQL Server clusters”, or “My business-critical clusters”—you might want to name each Updating Run Profile according to the class of Failover Clusters it will be used with. In addition, you might want to manage the Updating Run Profile on a file share that is accessible to all of the failover clusters of a specific class in your IT organization.

See also

- [Cluster-Aware Updating](#)
- [Cluster-Aware Updating Cmdlets in Windows PowerShell](#)

Cluster-Aware Updating: Frequently Asked Questions

4/30/2018 • 10 minutes to read • [Edit Online](#)

Applies to: Windows Server (Semi-Annual Channel), Windows Server 2016, Windows Server 2012 R2, Windows Server 2012

Cluster-Aware Updating (CAU) is a feature that coordinates software updates on all servers in a failover cluster in a way that doesn't impact the service availability any more than a planned failover of a cluster node. For some applications with continuous availability features (such as Hyper-V with live migration, or an SMB 3.x file server with SMB Transparent Failover), CAU can coordinate automated cluster updating with no impact on service availability.

Does CAU support updating Storage Spaces Direct clusters?

Yes. CAU supports updating **Storage Spaces Direct** clusters regardless of the deployment type: hyper-converged or converged. Specifically, CAU orchestration ensures that suspending each cluster node waits for the underlying clustered storage space to be healthy.

Does CAU work with Windows Server 2008 R2 or Windows 7?

No. CAU coordinates the cluster updating operation only from computers running Windows Server 2016, Windows Server 2012 R2, Windows Server 2012, Windows 10, Windows 8.1, or Windows 8. The failover cluster being updated must run Windows Server 2016, Windows Server 2012 R2, or Windows Server 2012.

Is CAU limited to specific clustered applications?

No. CAU is agnostic to the type of the clustered application. CAU is an external cluster-updating solution that is layered on top of clustering APIs and PowerShell cmdlets. As such, CAU can coordinate updating for any clustered application that is configured in a Windows Server failover cluster.

NOTE

Currently, the following clustered workloads are tested and certified for CAU: SMB, Hyper-V, DFS Replication, DFS Namespaces, iSCSI, and NFS.

Does CAU support updates from Microsoft Update and Windows Update?

Yes. By default, CAU is configured with a plug-in that uses the Windows Update Agent (WUA) utility APIs on the cluster nodes. The WUA infrastructure can be configured to point to Microsoft Update and Windows Update or to Windows Server Update Services (WSUS) as its source of updates.

Does CAU support WSUS updates?

Yes. By default, CAU is configured with a plug-in that uses the Windows Update Agent (WUA) utility APIs on the cluster nodes. The WUA infrastructure can be configured to point to Microsoft Update and Windows Update or to a local Windows Server Update Services (WSUS) server as its source of updates.

Can CAU apply limited distribution release updates?

Yes. Limited distribution release (LDR) updates, also called hotfixes, are not published through Microsoft Update or Windows Update, so they cannot be downloaded by the Windows Update Agent (WUA) plug-in that CAU uses by default.

However, CAU includes a second plug-in that you can select to apply hotfix updates. This hotfix plug-in can also be customized to apply non-Microsoft driver, firmware, and BIOS updates.

Can I use CAU to apply cumulative updates?

Yes. If the cumulative updates are general distribution release updates or LDR updates, CAU can apply them.

Can I schedule updates?

Yes. CAU supports the following updating modes, both of which allow updates to be scheduled:

Self-updating Enables the cluster to update itself according to a defined profile and a regular schedule, such as during a monthly maintenance window. You can also start a Self-Updating Run on demand at any time. To enable self-updating mode, you must add the CAU clustered role to the cluster. The CAU self-updating feature performs like any other clustered workload, and it can work seamlessly with the planned and unplanned failovers of an update coordinator computer.

Remote-updating Enables you to start an Updating Run at any time from a computer running Windows or Windows Server. You can start an Updating run through the Cluster-Aware Updating window or by using the **Invoke-CauRun** PowerShell cmdlet. Remote-updating is the default updating mode for CAU. You can use Task Scheduler to run the [Invoke-CauRun](#) cmdlet on a desired schedule from a remote computer that is not one of the cluster nodes.

Can I schedule updates to apply during a backup?

Yes. CAU doesn't impose any constraints in this regard. However, performing software updates on a server (with the associated potential restarts) while a server backup is in progress is not an IT best practice. Be aware that CAU relies only on clustering APIs to determine resource failovers and failbacks; thus, CAU is unaware of the server backup status.

Can CAU work with System Center Configuration Manager?

CAU is a tool that coordinates software updates on a cluster node, and Configuration Manager also performs server software updates. It's important to configure these tools so that they don't have overlapping coverage of the same servers in any datacenter deployment, including using different Windows Server Update Services servers. This ensures that the objective behind using CAU is not inadvertently defeated, because Configuration Manager-driven updating doesn't incorporate cluster awareness.

Do I need administrative credentials to run CAU?

Yes. For running the CAU tools, CAU needs administrative credentials on the local server, or it needs the **Impersonate a Client after Authentication** user right on the local server or the client computer on which it is running. However, to coordinate software updates on the cluster nodes, CAU requires cluster administrative credentials on every node. Although the CAU UI can start without the credentials, it prompts for the cluster administrative credentials when it connects to a cluster instance to preview or apply updates.

Can I script CAU?

Yes. CAU comes with PowerShell cmdlets that offer a rich set of scripting options. These are the same cmdlets that

the CAU UI calls to perform CAU actions.

What happens to active clustered roles?

Clustered roles (formerly called applications and services) that are active on a node, fail over to other nodes before software updating can commence. CAU orchestrates these failovers by using the maintenance mode, which pauses and drains the node of all active clustered roles. When the software updates are complete, CAU resumes the node and the clustered roles fail back to the updated node. This ensures that the distribution of clustered roles relative to nodes stays the same across the CAU Updating Runs of a cluster.

How does CAU select target nodes for clustered roles?

CAU relies on clustering APIs to coordinate the failovers. The clustering API implementation selects the target nodes by relying on internal metrics and intelligent placement heuristics (such as workload levels) across the target nodes.

Does CAU load balance the clustered roles?

CAU doesn't load balance the clustered nodes, but it attempts to preserve the distribution of clustered roles. When CAU finishes updating a cluster node, it attempts to fail back previously hosted clustered roles to that node. CAU relies on clustering APIs to fail back the resources to the beginning of the pause process. Thus in the absence of unplanned failovers and preferred owner settings, the distribution of clustered roles should remain unchanged.

How does CAU select the order of nodes to update?

By default, CAU selects the order of nodes to update based on the level of activity. The nodes that are hosting the fewest clustered roles are updated first. However, an administrator can specify a particular order for updating the nodes by specifying a parameter for the Updating Run in the CAU UI or by using the PowerShell cmdlets.

What happens if a cluster node is offline?

The administrator who initiates an Updating Run can specify the acceptable threshold for the number of nodes that can be offline. Therefore, an Updating Run can proceed on a cluster even if all the cluster nodes are not online.

Can I use CAU to update only a single node?

No. CAU is a cluster-scoped updating tool, so it only allows you to select clusters to update. If you want to update a single node, you can use existing server updating tools independently of CAU.

Can CAU report updates that are initiated from outside CAU?

No. CAU can only report Updating Runs that are initiated from within CAU. However, when a subsequent CAU Updating Run is launched, updates that were installed through non-CAU methods are appropriately considered to determine the additional updates that might be applicable to each cluster node.

Can CAU support my unique IT process needs?

Yes. CAU offers the following dimensions of flexibility to suit enterprise customers' unique IT process needs:

Scripts An Updating Run can specify a pre-update PowerShell script and a post-update PowerShell script. The pre-update script runs on each cluster node before the node is paused. The post-update script runs on each cluster node after the node updates are installed.

NOTE

.NET Framework 4.6 or 4.5 and PowerShell must be installed on each cluster node on which you want to run the pre-update and post-update scripts. You must also enable PowerShell remoting on the cluster nodes. For detailed system requirements, see [Requirements and Best Practices for Cluster-Aware Updating](#).

Advanced Updating Run options The administrator can additionally specify from a large set of advanced Updating Run options such as the maximum number of times that the update process is retried on each node. These options can be specified using either the CAU UI or the CAU PowerShell cmdlets. These custom settings can be saved in an Updating Run Profile and reused for later Updating Runs.

Public plug-in architecture CAU includes features to Register, Unregister, and Select plug-ins. CAU ships with two default plug-ins: one coordinates the Windows Update Agent (WUA) APIs on each cluster node; the second applies hotfixes that are manually copied to a file share that is accessible to the cluster nodes. If an enterprise has unique needs that cannot be met with these two plug-ins, the enterprise can build a new CAU plug-in according to the public API specification. For more information, see [Cluster-Aware Updating Plug-in Reference](#).

For information about configuring and customizing CAU plug-ins to support different updating scenarios, see [How Plug-ins Work](#).

How can I export the CAU preview and update results?

CAU offers export options through the command-line interface and through the UI.

Command-line interface options:

- Preview results by using the PowerShell cmdlet **Invoke-CauScan | ConvertTo-Xml**. Output: XML
- Report results by using the PowerShell cmdlet **Invoke-CauRun | ConvertTo-Xml**. Output: XML
- Report results by using the PowerShell cmdlet **Get-CauReport | Export-CauReport**. Output: HTML, CSV

UI options:

- Copy the report results from the **Preview updates** screen. Output: CSV
- Copy the report results from the **Generate report** screen. Output: CSV
- Export the report results from the **Generate report** screen. Output: HTML

How do I install CAU?

A CAU installation is seamlessly integrated into the Failover Clustering feature. CAU is installed as follows:

- When Failover Clustering is installed on a cluster node, the CAU Windows Management Instrumentation (WMI) provider is automatically installed.
- When the Failover Clustering Tools feature is installed on a server or client computer, the Cluster-Aware Updating UI and PowerShell cmdlets are automatically installed.

Does CAU need components running on the cluster nodes that are being updated?

CAU doesn't need a service running on the cluster nodes. However, CAU needs a software component (the WMI provider) installed on the cluster nodes. This component is installed with the Failover Clustering feature.

To enable self-updating mode, the CAU clustered role must also be added to the cluster.

What is the difference between using CAU and VMM?

- System Center Virtual Machine Manager (VMM) is focused on updating only Hyper-V clusters, whereas CAU can update any type of supported failover cluster, including Hyper-V clusters.
- VMM requires additional licensing, whereas CAU is licensed for all Windows Server. The CAU features, tools, and UI are installed with Failover Clustering components.
- If you already own a System Center license, you can continue to use VMM to update Hyper-V clusters because it offers an integrated management and software updating experience.
- CAU is supported only on clusters that are running Windows Server 2016, Windows Server 2012 R2, and Windows Server 2012. VMM also supports Hyper-V clusters on computers running Windows Server 2008 R2 and Windows Server 2008.

Can I use remote-updating on a cluster that is configured for self-updating?

Yes. A failover cluster in a self-updating configuration can be updated through remote-updating on-demand, just as you can force a Windows Update scan at any time on your computer, even if Windows Update is configured to install updates automatically. However, you need to make sure that an Updating Run is not already in progress.

Can I reuse my cluster update settings across clusters?

Yes. CAU supports a number of Updating Run options that determine how the Updating Run behaves when it updates the cluster. These options can be saved as an Updating Run Profile, and they can be reused across any cluster. We recommend that you save and reuse your settings across failover clusters that have similar updating needs. For example, you might create a "Business-Critical SQL Server Cluster Updating Run Profile" for all Microsoft SQL Server clusters that support business-critical services.

Where is the CAU plug-in specification?

- [Cluster-Aware Updating Plug-in Reference](#)
- [Cluster Aware Updating plug-in sample](#)

See also

- [Cluster-Aware Updating Overview](#)

How Cluster-Aware Updating plug-ins work

4/30/2018 • 21 minutes to read • [Edit Online](#)

Applies to: Windows Server (Semi-Annual Channel), Windows Server 2016, Windows Server 2012 R2, Windows Server 2012

Cluster-Aware Updating (CAU) uses plug-ins to coordinate the installation of updates across nodes in a failover cluster. This topic provides information about using the built-in CAU plug-ins or other plug-ins that you install for CAU.

Install a plug-in

A plug-in other than the default plug-ins that are installed with CAU (**Microsoft.WindowsUpdatePlugin** and **Microsoft.HotfixPlugin**) must be installed separately. If CAU is used in self-updating mode, the plug-in must be installed on all cluster nodes. If CAU is used in remote-updating mode, the plug-in must be installed on the remote Update Coordinator computer. A plug-in that you install may have additional installation requirements on each node.

To install a plug-in, follow the instructions from the plug-in publisher. To manually register a plug-in with CAU, run the [Register-CauPlugin](#) cmdlet on each computer where the plug-in is installed.

Specify a plug-in and plug-in arguments

Specify a CAU plug-in

In the CAU UI, you select a plug-in from a drop-down list of available plug-ins when you use CAU to perform the following actions:

- Apply updates to the cluster
- Preview updates for the cluster
- Configure cluster self-updating options

By default, CAU selects the plug-in **Microsoft.WindowsUpdatePlugin**. However, you can specify any plug-in that is installed and registered with CAU.

TIP

In the CAU UI, you can only specify a single plug-in for CAU to use to preview or to apply updates during an Updating Run. By using the CAU PowerShell cmdlets, you can specify one or more plug-ins. If you need to install multiple types of updates on the cluster, it is usually more efficient to specify multiple plug-ins in one Updating Run, rather than using a separate Updating Run for each plug-in. For example, fewer node restarts will typically occur.

By using the CAU PowerShell cmdlets that are listed in the following table, you can specify one or more plug-ins for an Updating Run or scan by passing the **-CauPluginName** parameter. You can specify multiple plug-in names by separating them with commas. If you specify multiple plug-ins, you can also control how the plug-ins influence each other during an Updating Run by specifying the **-RunPluginsSerially**, **-StopOnPluginFailure**, and **-SeparateReboots** parameters. For more information about using multiple plug-ins, use the links provided to the cmdlet documentation in the following table.

CMDLET	DESCRIPTION
Add-CauClusterRole	Adds the CAU clustered role that provides the self-updating functionality to the specified cluster.
Invoke-CauRun	Performs a scan of cluster nodes for applicable updates and installs those updates through an Updating Run on the specified cluster.
Invoke-CauScan	Performs a scan of cluster nodes for applicable updates and returns a list of the initial set of updates that would be applied to each node in the specified cluster.
Set-CauClusterRole	Sets configuration properties for the CAU clustered role on the specified cluster.

If you do not specify a CAU plug-in parameter by using these cmdlets, the default is the plug-in

Microsoft.WindowsUpdatePlugin.

Specify CAU plug-in arguments

When you configure the Updating Run options, you can specify one or more *name=value* pairs (arguments) for the selected plug-in to use. For example, in the CAU UI, you can specify multiple arguments as follows:

Name1=Value1;Name2=Value2;Name3=Value3

These *name=value* pairs must be meaningful to the plug-in that you specify. For some plug-ins the arguments are optional.

The syntax of the CAU plug-in arguments follows these general rules:

- Multiple *name=value* pairs are separated by semicolons.
- A value that contains spaces is surrounded by quotation marks, for example: **Name1="Value with Spaces"**.
- The exact syntax of *value* depends on the plug-in.

To specify plug-in arguments by using the CAU PowerShell cmdlets that support the **-CauPluginParameters** parameter, pass a parameter of the form:

-CauPluginArguments @{Name1=Value1;Name2=Value2;Name3=Value3}

You can also use a predefined PowerShell hash table. To specify plug-in arguments for more than one plug-in, pass multiple hash tables of arguments, separated with commas. Pass the plug-in arguments in the plug-in order that is specified in **CauPluginName**.

Specify optional plug-in arguments

The plug-ins that CAU installs (**Microsoft.WindowsUpdatePlugin** and **Microsoft.HotfixPlugin**) provide additional options that you can select. In the CAU UI, these appear on an **Additional Options** page after you configure Updating Run options for the plug-in. If you are using the CAU PowerShell cmdlets, these options are configured as optional plug-in arguments. For more information, see [Use the Microsoft.WindowsUpdatePlugin](#) and [Use the Microsoft.HotfixPlugin](#) later in this topic.

Manage plug-ins using Windows PowerShell cmdlets

CMDLET	DESCRIPTION
Get-CauPlugin	Retrieves information about one or more software updating plug-ins that are registered on the local computer.
Register-CauPlugin	Registers a CAU software updating plug-in on the local computer.
Unregister-CauPlugin	Removes a software updating plug-in from the list of plug-ins that can be used by CAU. Note: The plug-ins that are installed with CAU (Microsoft.WindowsUpdatePlugin and the Microsoft.HotfixPlugin) cannot be unregistered.

Using the Microsoft.WindowsUpdatePlugin

The default plug-in for CAU, **Microsoft.WindowsUpdatePlugin**, performs the following actions:

- Communicates with the Windows Update Agent on each failover cluster node to apply updates that are needed for the Microsoft products that are running on each node.
- Installs cluster updates directly from Windows Update or Microsoft Update, or from an on-premises Windows Server Update Services (WSUS) server.
- Installs only selected, general distribution release (GDR) updates. By default, the plug-in applies only important software updates. No configuration is required. The default configuration downloads and installs important GDR updates on each node.

NOTE

To apply updates other than the important software updates that are selected by default (for example, driver updates), you can configure an optional plug-in parameter. For more information, see [Configure the Windows Update Agent query string](#).

Requirements

- The failover cluster and remote Update Coordinator computer (if used) must meet the requirements for CAU and the configuration that is required for remote management listed in [Requirements and Best Practices for CAU](#).
- Review [Recommendations for applying Microsoft updates](#), and then make any necessary changes to your Microsoft Update configuration for the failover cluster nodes.
- For best results, we recommend that you run the CAU Best Practices Analyzer (BPA) to ensure that the cluster and update environment are configured properly to apply updates by using CAU. For more information, see [Test CAU updating readiness](#).

NOTE

Updates that require the acceptance of Microsoft license terms or require user interaction are excluded, and they must be installed manually.

Additional options

Optionally, you can specify the following plug-in arguments to augment or restrict the set of updates that are applied by the plug-in:

- To configure the plug-in to apply recommended updates in addition to important updates on each node, in the CAU UI, on the **Additional Options** page, select the **Give me recommended updates the same way that I receive important updates** check box.

Alternatively, configure the '**IncludeRecommendedUpdates**'='True' plug-in argument.

- To configure the plug-in to filter the types of GDR updates that are applied to each cluster node, specify a Windows Update Agent query string using a **QueryString** plug-in argument. For more information, see [Configure the Windows Update Agent query string](#).

Configure the Windows Update Agent query string

You can configure a plug-in argument for the default plug-in, **Microsoft.WindowsUpdatePlugin**, that consists of a Windows Update Agent (WUA) query string. This instruction uses the WUA API to identify one or more groups of Microsoft updates to apply to each node, based on specific selection criteria. You can combine multiple criteria by using a logical AND or a logical OR. The WUA query string is specified in a plug-in argument as follows:

QueryString="Criterion1=Value1 and/or Criterion2=Value2 and/or..."

For example, **Microsoft.WindowsUpdatePlugin** automatically selects important updates by using a default **QueryString** argument that is constructed using the **IsInstalled**, **Type**, **IsHidden**, and **IsAssigned** criteria:

QueryString="IsInstalled=0 and Type='Software' and IsHidden=0 and IsAssigned=1"

If you specify a **QueryString** argument, it is used in place of the default **QueryString** that is configured for the plug-in.

Example 1

To configure a **QueryString** argument that installs a specific update as identified by ID *f6ce46c1-971c-43f9-a2aa-783df125f003*:

QueryString="UpdateID='f6ce46c1-971c-43f9-a2aa-783df125f003' and IsInstalled=0"

NOTE

The preceding example is valid for applying updates by using the Cluster-Aware Updating Wizard. If you want to install a specific update by configuring self-updating options with the CAU UI or by using the **Add-CauClusterRole** or **Set-CauClusterRole** PowerShell cmdlet, you must format the UpdateID value with two single-quote characters:

QueryString="UpdateID='f6ce46c1-971c-43f9-a2aa-783df125f003' and IsInstalled=0"

Example 2

To configure a **QueryString** argument that installs only drivers:

QueryString="IsInstalled=0 and Type='Driver' and IsHidden=0"

For more information about query strings for the default plug-in, **Microsoft.WindowsUpdatePlugin**, the search criteria (such as **IsInstalled**), and the syntax that you can include in the query strings, see the section about search criteria in the [Windows Update Agent \(WUA\) API Reference](#).

Use the Microsoft.HotfixPlugin

The plug-in **Microsoft.HotfixPlugin** can be used to apply Microsoft limited distribution release (LDR) updates (also called hotfixes, and formerly called QFEs) that you download independently to address specific Microsoft software issues. The plug-in installs updates from a root folder on an SMB file share and can also be customized to apply non-Microsoft driver, firmware, and BIOS updates.

NOTE

Hotfixes are sometimes available for download from Microsoft in Knowledge Base articles, but they are also provided to customers on an as-needed basis.

Requirements

- The failover cluster and remote Update Coordinator computer (if used) must meet the requirements for CAU and the configuration that is required for remote management listed in [Requirements and Best Practices for CAU](#).
- Review [Recommendations for using the Microsoft.HotfixPlugin](#).
- For best results, we recommend that you run the CAU Best Practices Analyzer (BPA) model to ensure that the cluster and update environment are configured properly to apply updates by using CAU. For more information, see [Test CAU updating readiness](#).
- Obtain the updates from the publisher, and copy them or extract them to a Server Message Block (SMB) file share (hotfix root folder) that supports at least SMB 2.0 and that is accessible by all of the cluster nodes and the remote Update Coordinator computer (if CAU is used in remote-updating mode). For more information, see [Configure a hotfix root folder structure](#) later in this topic.

NOTE

By default, this plug-in only installs hotfixes with the following file name extensions: .msu, .msi, and .msp.

- Copy the DefaultHotfixConfig.xml file (which is provided in the **%systemroot%\System32\WindowsPowerShell\v1.0\Modules\ClusterAwareUpdating** folder on a computer where the CAU tools are installed) to the hotfix root folder that you created and under which you extracted the hotfixes. For example, copy the configuration file to `\\MyFileServer\Hotfixes\Root\`.

NOTE

To install most hotfixes provided by Microsoft and other updates, the default hotfix configuration file can be used without modification. If your scenario requires it, you can customize the configuration file as an advanced task. The configuration file can include custom rules, for example, to handle hotfix files that have specific extensions, or to define behaviors for specific exit conditions. For more information, see [Customize the hotfix configuration file](#) later in this topic.

Configuration

Configure the following settings. For more information, see the links to sections later in this topic.

- The path to the shared hotfix root folder that contains the updates to apply and that contains the hotfix configuration file. You can type this path in the CAU UI or configure the **HotfixRootFolderPath= <Path>** PowerShell plug-in argument.

NOTE

You can specify the hotfix root folder as a local folder path or as a UNC path of the form `\\ServerName\Share\RootFolderName`. A domain-based or standalone DFS Namespace path can be used. However, the plug-in features that check access permissions in the hotfix configuration file are incompatible with a DFS Namespace path, so if you configure one, you must disable the check for access permissions by using the CAU UI or by configuring the **DisableAclChecks='True'** plug-in argument.

- Settings on the server that hosts the hotfix root folder to check for appropriate permissions to access the folder and ensure the integrity of the data accessed from the SMB shared folder (SMB signing or SMB Encryption). For more information, see [Restrict access to the hotfix root folder](#).

Additional options

- Optionally, configure the plug-in so that SMB Encryption is enforced when accessing data from the hotfix file share. In the CAU UI, on the **Additional Options** page, select the **Require SMB Encryption in accessing**

the hotfix root folder option, or configure the **RequireSMBEncryption='True'** PowerShell plug-in argument. > [!IMPORTANT] > You must perform additional configuration steps on the SMB server to enable SMB data integrity with SMB signing or SMB Encryption. For more information, see Step 4 in [Restrict access to the hotfix root folder](#). If you select the option to enforce the use of SMB Encryption, and the hotfix root folder is not configured for access by using SMB Encryption, the Updating Run will fail.

- Optionally, disable the default checks for sufficient permissions for the hotfix root folder and the hotfix configuration file. In the CAU UI, select **Disable check for administrator access to the hotfix root folder and configuration file**, or configure the **DisableAclChecks='True'** plug-in argument.
- Optionally, configure the **HotfixInstallerTimeoutMinutes=** argument to specify how long the hotfix plug-in waits for the hotfix installer process to return. (The default is 30 minutes.) For example, to specify a timeout period of two hours, set **HotfixInstallerTimeoutMinutes=120**.
- Optionally, configure the **HotfixConfigFileName =** plug-in argument to specify a name for the hotfix configuration file that is located in the hotfix root folder. If not specified, the default name DefaultHotfixConfig.xml is used.

Configure a hotfix root folder structure

For the hotfix plug-in to work, hotfixes must be stored in a well-defined structure in an SMB file share (hotfix root folder), and you must configure the hotfix plug-in with the path to the hotfix root folder by using the CAU UI or the CAU PowerShell cmdlets. This path is passed to the plug-in as the **HotfixRootFolderPath** argument. You can choose one of several structures for the hotfix root folder, according to your updating needs, as shown in the following examples. Files or folders that do not adhere to the structure are ignored.

Example 1 - Folder structure used to apply hotfixes to all cluster nodes

To specify that hotfixes apply to all cluster nodes, copy them to a folder named **CAUHotfix_All** under the hotfix root folder. In this example, the **HotfixRootFolderPath** plug-in argument is set to `\\MyFileServer\Hotfixes\Root\`. The **CAUHotfix_All** folder contains three updates with the extensions .msu, .msi, and .msp that will be applied to all cluster nodes. The update file names are only for illustration purposes.

NOTE

In this and the following examples, the hotfix configuration file with its default name DefaultHotfixConfig.xml is shown in its required location in the hotfix root folder.

```
\\MyFileServer\Hotfixes\Root\  
  DefaultHotfixConfig.xml  
  CAUHotfix_All\  
    Update1.msu  
    Update2.msi  
    Update3.msp  
    ...
```

Example 2 - Folder structure used to apply certain updates only to a specific node

To specify hotfixes that apply only to a specific node, use a subfolder under the hotfix root folder with the name of the node. Use the NetBIOS name of the cluster node, for example, *ContosoNode1*. Then, move the updates that apply only to this node to this subfolder. In the following example, the **HotfixRootFolderPath** plug-in argument is set to `\\MyFileServer\Hotfixes\Root\`. Updates in the **CAUHotfix_All** folder will be applied to all cluster nodes, and *Node1_Specific_Update.msu* will be applied only to *ContosoNode1*.

```
\\MyFileServer\Hotfixes\Root\  
  DefaultHotfixConfig.xml  
  CAUHotfix_All\  
    Update1.msu  
    Update2.msi  
    Update3.msp  
    ...  
  ContosoNode1\  
    Node1_Specific_Update.msu  
    ...
```

Example 3 - Folder structure used to apply updates other than .msu, .msi, and .msp files

By default, **Microsoft.HotfixPlugin** only applies updates with the .msu, .msi, or .msp extension. However, certain updates might have different extensions and require different installation commands. For example, you might need to apply a firmware update with the extension .exe to a node in a cluster. You can configure the hotfix root folder with a subfolder that indicates a specific, non-default update type should be installed. You must also configure a corresponding folder installation rule that specifies the installation command in the `<FolderRules>` element in the hotfix configuration XML file.

In the following example, the **HotfixRootFolderPath** plug-in argument is set to `\\MyFileServer\Hotfixes\Root\`. Several updates will be applied to all cluster nodes, and a firmware update *SpecialHotfix1.exe* will be applied to *ContosoNode1* by using *FolderRule1*. For information about configuring *FolderRule1* in the hotfix configuration file, see [Customize the hotfix configuration file](#) later in this topic.

```
\\MyFileServer\Hotfixes\Root\  
  DefaultHotfixConfig.xml  
  CAUHotfix_All\  
    Update1.msu  
    Update2.msi  
    Update3.msp  
    ...  
  
  ContosoNode1\  
    FolderRule1\  
      SpecialHotfix1.exe  
    ...
```

Customize the hotfix configuration file

The hotfix configuration file controls how **Microsoft.HotfixPlugin** installs specific hotfix file types in a failover cluster. The XML schema for the configuration file is defined in `HotfixConfigSchema.xsd`, which is located in the following folder on a computer where the CAU tools are installed:

%systemroot%\System32\WindowsPowerShell\v1.0\Modules\ClusterAwareUpdating folder

To customize the hotfix configuration file, copy the sample configuration file `DefaultHotfixConfig.xml` from this location to the hotfix root folder and make appropriate modifications for your scenario.

IMPORTANT

To apply most hotfixes provided by Microsoft and other updates, the default hotfix configuration file can be used without modification. Customization of the hotfix configuration file is a task only in advanced usage scenarios.

By default, the hotfix configuration XML file defines installation rules and exit conditions for the following two categories of hotfixes:

- Hotfix files with extensions that the plug-in can install by default (.msu, .msi, and .msp files).

These are defined as `<ExtensionRules>` elements in the `<DefaultRules>` element. There is one `<Extension>` element for each of the default supported file types. The general XML structure is as follows:

```
<DefaultRules>
  <ExtensionRules>
    <Extension name="MSI">
      <!-- Template and ExitConditions elements for installation of .msi files follow -->
      ...
    </Extension>
    <Extension name="MSU">
      <!-- Template and ExitConditions elements for installation of .msu files follow -->
      ...
    </Extension>
    <Extension name="MSP">
      <!-- Template and ExitConditions elements for installation of .msp files follow -->
      ...
    </Extension>
    ...
  </ExtensionRules>
</DefaultRules>
```

If you need to apply certain update types to all cluster nodes in your environment, you can define additional `<Extension>` elements.

- Hotfix or other update files that are not .msi, .msu, or .msp files, for example, non-Microsoft drivers, firmware, and BIOS updates.

Each non-default file type is configured as a `<Folder>` element in the `<FolderRules>` element. The name attribute of the `<Folder>` element must be identical to the name of a folder in the hotfix root folder that will contain updates of the corresponding type. The folder can be in the **CAUHotfix_All** folder or in a node-specific folder. For example, if *FolderRule1* is configured in the hotfix root folder, configure the following element in the XML file to define an installation template and exit conditions for the updates in that folder:

```
<FolderRules>
  <Folder name="FolderRule1">
    <!-- Template and ExitConditions elements for installation of updates in FolderRule1 follow -->
    ...
  </Folder>
  ...
</FolderRules>
```

The following tables describe the `<Template>` attributes and the possible `<ExitConditions>` subelements.

<code><TEMPLATE></code> ATTRIBUTE	DESCRIPTION
path	<p>The full path to the installation program for the file type that is defined in the <code><Extension name></code> attribute.</p> <p>To specify the path to an update file in the hotfix root folder structure, use <code>\$update\$</code>.</p>
parameters	<p>A string of required and optional parameters for the program that is specified in <code>path</code>.</p> <p>To specify a parameter that is the path to an update file in the hotfix root folder structure, use <code>\$update\$</code>.</p>

<code><EXITCONDITIONS></code> SUBELEMENT	DESCRIPTION
<code><Success></code>	Defines one or more exit codes that indicate the specified update succeeded. This is a required subelement.
<code><Success_RebootRequired></code>	Optionally defines one or more exit codes that indicate the specified update succeeded and the node must restart. Note: Optionally, the <code><Folder></code> element can contain the <code>alwaysReboot</code> attribute. If this attribute is set, it indicates that if a hotfix installed by this rule returns one of the exit codes that is defined in <code><Success></code> , it is interpreted as a <code><Success_RebootRequired></code> exit condition.
<code><Fail_RebootRequired></code>	Optionally defines one or more exit codes that indicate the specified update failed and the node must restart.
<code><AlreadyInstalled></code>	Optionally defines one or more exit codes that indicate the specified update was not applied because it is already installed.
<code><NotApplicable></code>	Optionally defines one or more exit codes that indicate the specified update was not applied because it does not apply to the cluster node.

IMPORTANT

Any exit code that is not explicitly defined in `<ExitConditions>` is interpreted as the update failed, and the node does not restart.

Restrict access to the hotfix root folder

You must perform several steps to configure the SMB file server and file share to help secure the hotfix root folder files and hofix configuration file for access only in the context of **Microsoft.HotfixPlugin**. These steps enable several features that help prevent possible tampering with the hotfix files in a way that might compromise the failover cluster.

The general steps are as follows:

1. Identify the user account that is used for Updating Runs by using the plug-in
2. Configure this user account in the necessary groups on an SMB file server
3. Configure permissions to access the hotfix root folder
4. Configure settings for SMB data integrity
5. Enable a Windows Firewall rule on the SMB server

Step 1. Identify the user account that is used for Updating Runs by using the hotfix plug-in

The account that is used in CAU to check security settings while performing an Updating Run using **Microsoft.HotfixPlugin** depends on whether CAU is used in remote-updating mode or self-updating mode, as follows:

- **Remote-updating mode** The account that has administrative privileges on the cluster to preview and apply updates.
- **Self-updating mode** The name of the virtual computer object that is configured in Active Directory for the CAU clustered role. This is either the name of a prestaged virtual computer object in Active Directory

for the CAU clustered role or the name that is generated by CAU for the clustered role. To obtain the name if it is generated by CAU, run the **Get-CauClusterRole** CAU PowerShell cmdlet. In the output, **ResourceGroupName** is the name of the generated virtual computer object account.

Step 2. Configure this user account in the necessary groups on an SMB file server

IMPORTANT

You must add the account that is used for Updating Runs as a local administrator account on the SMB server. If this is not permitted because of the security policies in your organization, configure this account with the necessary permissions on the SMB server by using the following procedure.

To configure a user account on the SMB server

1. Add the account that is used for Updating Runs to the Distributed COM Users group and to one of the following groups: Power User, Server Operation, or Print Operator.
2. To enable the necessary WMI permissions for the account, start the WMI Management Console on the SMB server. Start PowerShell and then type the following command:

```
wimgmt.msc
```

3. In the console tree, right-click **WMI Control (Local)**, and then click **Properties**.
4. Click **Security**, and then expand **Root**.
5. Click **CIMV2**, and then click **Security**.
6. Add the account that is used for Updating Runs to the **Group or user names** list.
7. Grant the **Execute Methods** and **Remote Enable** permissions to the account that is used for Updating Runs.

Step 3. Configure permissions to access the hotfix root folder

By default, when you attempt to apply updates, the hotfix plug-in checks the configuration of the NTFS file system permissions for access to the hotfix root folder. If the folder access permissions are not configured properly, an Updating Run using the hotfix plug-in might fail.

If you use the default configuration of the hotfix plug-in, ensure that the folder access permissions meet the following requirements.

- The Users group has Read permission.
- If the plug-in will apply updates with the .exe extension, the Users group has Execute permission.
- Only certain security principals are permitted (but are not required) to have Write or Modify permission. The allowed principals are the local Administrators group, SYSTEM, CREATOR OWNER, and TrustedInstaller. Other accounts or groups are not permitted to have Write or Modify permission on the hotfix root folder.

Optionally, you can disable the preceding checks that the plug-in performs by default. You can do this in one of two ways:

- If you are using the CAU PowerShell cmdlets, configure the **DisableAclChecks='True'** argument in the **CauPluginArguments** parameter for the hotfix plug-in.
- If you are using the CAU UI, select the **Disable check for administrator access to the hotfix root folder and configuration file** option on the **Additional Update Options** page of the wizard that is used to configure Updating Run options.

However, as a best practice in many environments, we recommend that you use the default configuration to enforce these checks.

Step 4. Configure settings for SMB data integrity

To check for data integrity in the connections between the cluster nodes and the SMB file share, the hotfix plug-in requires that you enable settings on the SMB file share for SMB signing or SMB Encryption. SMB Encryption, which provides enhanced security and better performance in many environments, is supported starting in Windows Server 2012. You can enable either or both of these settings, as follows:

- To enable SMB signing, see the procedure in the [article 887429](#) in the Microsoft Knowledge Base.
- To enable SMB Encryption for the SMB shared folder, run the following PowerShell cmdlet on the SMB server:

```
Set-SmbShare <ShareName> -EncryptData $true
```

Where *<ShareName>* is the name of the SMB shared folder.

Optionally, to enforce the use of SMB Encryption in the connections to the SMB server, select the **Require SMB Encryption in accessing the hotfix root folder** option in the CAU UI, or configure the **RequireSMBEncryption='True'** plug-in argument by using the CAU PowerShell cmdlets.

IMPORTANT

If you select the option to enforce the use of SMB Encryption, and the hotfix root folder is not configured for connections that use SMB Encryption, the Updating Run will fail.

Step 5. Enable a Windows Firewall rule on the SMB server

You must enable the **File Server Remote Management (SMB-in)** rule in Windows Firewall on the SMB file server. This is enabled by default in Windows Server 2016, Windows Server 2012 R2, and Windows Server 2012.

See also

- [Cluster-Aware Updating Overview](#)
- [Cluster-Aware Updating Windows PowerShell Cmdlets](#)
- [Cluster-Aware Updating Plug-in Reference](#)

Health Service in Windows Server

2/10/2018 • 4 minutes to read • [Edit Online](#)

Applies to Windows Server 2016

The Health Service is a new feature in Windows Server 2016 that improves the day-to-day monitoring and operational experience for clusters running Storage Spaces Direct.

Prerequisites

The Health Service is enabled by default with Storage Spaces Direct. No additional action is required to set it up or start it. To learn more about Storage Spaces Direct, see [Storage Spaces Direct in Windows Server 2016](#).

Reports

See [Health Service reports](#).

Faults

See [Health Service faults](#).

Actions

See [Health Service actions](#).

Automation

This section describes workflows which are automated by the Health Service in the disk lifecycle.

Disk Lifecycle

The Health Service automates most stages of the physical disk lifecycle. Let's say that the initial state of your deployment is in perfect health - which is to say, all physical disks are working properly.

Retirement

Physical disks are automatically retired when they can no longer be used, and a corresponding Fault is raised. There are several cases:

- Media Failure: the physical disk is definitively failed or broken, and must be replaced.
- Lost Communication: the physical disk has lost connectivity for over 15 consecutive minutes.
- Unresponsive: the physical disk has exhibited latency of over 5.0 seconds three or more times within an hour.

NOTE

If connectivity is lost to many physical disks at once, or to an entire node or storage enclosure, the Health Service will *not* retire these disks since they are unlikely to be the root problem.

If the retired disk was serving as the cache for many other physical disks, these will automatically be reassigned to another cache disk if one is available. No special user action is required.

Restoring resiliency

Once a physical disk has been retired, the Health Service immediately begins copying its data onto the remaining physical disks, to restore full resiliency. Once this has completed, the data is completely safe and fault tolerant anew.

NOTE

This immediate restoration requires sufficient available capacity among the remaining physical disks.

Blinking the indicator light

If possible, the Health Service will begin blinking the indicator light on the retired physical disk or its slot. This will continue indefinitely, until the retired disk is replaced.

NOTE

In some cases, the disk may have failed in a way that precludes even its indicator light from functioning - for example, a total loss of power.

Physical replacement

You should replace the retired physical disk when possible. Most often, this consists of a hot-swap - i.e. powering off the node or storage enclosure is not required. See the Fault for helpful location and part information.

Verification

When the replacement disk is inserted, it will be verified against the Supported Components Document (see the next section).

Pooling

If allowed, the replacement disk is automatically substituted into its predecessor's pool to enter use. At this point, the system is returned to its initial state of perfect health, and then the Fault disappears.

Supported Components Document

The Health Service provides an enforcement mechanism to restrict the components used by Storage Spaces Direct to those on a Supported Components Document provided by the administrator or solution vendor. This can be used to prevent mistaken use of unsupported hardware by you or others, which may help with warranty or support contract compliance. This functionality is currently limited to physical disk devices, including SSDs, HDDs, and NVMe drives. The Supported Components Document can restrict on model, manufacturer (optional), and firmware version (optional).

Usage

The Supported Components Document uses an XML-inspired syntax. We recommend using your favorite text editor, such as the free [Visual Studio Code](#) or Notepad, to create an XML document which you can save and reuse.

Sections

The document has two independent sections: `Disks` and `Cache`.

If the `Disks` section is provided, only the drives listed (as `Disk`) are allowed to join pools. Any unlisted drives are prevented from joining pools, which effectively precludes their use in production. If this section is left empty, any drive will be allowed to join pools.

If the `Cache` section is provided, only the drives listed (as `CacheDisk`) are used for caching. If this section is left empty, Storage Spaces Direct attempts to [guess based on media type and bus type](#). Drives listed here should also be listed in `Disks`.

IMPORTANT

The Supported Components Document does not apply retroactively to drives already pooled and in use.

Example

```
<Components>

  <Disks>
    <Disk>
      <Manufacturer>Contoso</Manufacturer>
      <Model>XYZ9000</Model>
      <AllowedFirmware>
        <Version>2.0</Version>
        <Version>2.1</Version>
        <Version>2.2</Version>
      </AllowedFirmware>
      <TargetFirmware>
        <Version>2.1</Version>
        <BinaryPath>\\path\to\image.bin</BinaryPath>
      </TargetFirmware>
    </Disk>
    <Disk>
      <Manufacturer>Fabrikam</Manufacturer>
      <Model>QRSTUV</Model>
    </Disk>
  </Disks>

  <Cache>
    <CacheDisk>
      <Manufacturer>Fabrikam</Manufacturer>
      <Model>QRSTUV</Model>
    </CacheDisk>
  </Cache>

</Components>
```

To list multiple drives, simply add additional `<Disk>` or `<CacheDisk>` tags.

To inject this XML when deploying Storage Spaces Direct, use the `-XML` parameter:

```
$MyXML = Get-Content <Filepath> | Out-String
Enable-ClusterS2D -XML $MyXML
```

To set or modify the Supported Components Document once Storage Spaces Direct has been deployed:

```
$MyXML = Get-Content <Filepath> | Out-String
Get-StorageSubSystem Cluster* | Set-StorageHealthSetting -Name "System.Storage.SupportedComponents.Document" -
Value $MyXML
```

NOTE

The model, manufacturer, and the firmware version properties should exactly match the values that you get using the **Get-PhysicalDisk** cmdlet. This may differ from your "common sense" expectation, depending on your vendor's implementation. For example, rather than "Contoso", the manufacturer may be "CONTOSO-LTD", or it may be blank while the model is "Contoso-XZY9000".

You can verify using the following PowerShell cmdlet:

Settings

See [Health Service settings](#).

See also

- [Health Service reports](#)
- [Health Service faults](#)
- [Health Service actions](#)
- [Health Service settings](#)
- [Storage Spaces Direct in Windows Server 2016](#)

Health Service reports

10/17/2017 • 5 minutes to read • [Edit Online](#)

Applies to Windows Server 2016

What are reports

The Health Service reduces the work required to get live performance and capacity information from your Storage Spaces Direct cluster. One new cmdlet provides a curated list of essential metrics, which are collected efficiently and aggregated dynamically across nodes, with built-in logic to detect cluster membership. All values are real-time and point-in-time only.

Usage in PowerShell

Use this cmdlet to get metrics for the entire Storage Spaces Direct cluster:

```
Get-StorageSubSystem Cluster* | Get-StorageHealthReport
```

The optional **Count** parameter indicates how many sets of values to return, at one second intervals.

```
Get-StorageSubSystem Cluster* | Get-StorageHealthReport -Count <Count>
```

You can also get metrics for one specific volume or server:

```
Get-Volume -FileSystemLabel <Label> | Get-StorageHealthReport -Count <Count>
```

```
Get-StorageNode -Name <Name> | Get-StorageHealthReport -Count <Count>
```

Usage in .NET and C#

Connect

In order to query the Health Service, you will need to establish a **CimSession** with the cluster. To do so, you will need some things that are only available in full .NET, meaning you cannot readily do this directly from a web or mobile app. These code samples will use C#, the most straightforward choice for this data access layer.

```

...
using System.Security;
using Microsoft.Management.Infrastructure;

public CimSession Connect(string Domain = "...", string Computer = "...", string Username = "...", string
Password = "...")
{
    SecureString PasswordSecureString = new SecureString();
    foreach (char c in Password)
    {
        PasswordSecureString.AppendChar(c);
    }

    CimCredential Credentials = new CimCredential(
        PasswordAuthenticationMechanism.Default, Domain, Username, PasswordSecureString);
    WsmanSessionOptions SessionOptions = new WsmanSessionOptions();
    SessionOptions.AddDestinationCredentials(Credentials);
    Session = CimSession.Create(Computer, SessionOptions);
    return Session;
}

```

The provided Username should be a local Administrator of the target Computer.

It is recommended that you construct the Password **SecureString** directly from user input in real-time, so their password is never stored in memory in cleartext. This helps mitigate a variety of security concerns. But in practice, constructing it as above is common for prototyping purposes.

Discover objects

With the **CimSession** established, you can query Windows Management Instrumentation (WMI) on the cluster.

Before you can get faults or metrics, you will need to get instances of several relevant objects. First, the **MSFT_StorageSubSystem** which represents Storage Spaces Direct on the cluster. Using that, you can get every **MSFT_StorageNode** in the cluster, and every **MSFT_Volume**, the data volumes. Finally, you will need the **MSFT_StorageHealth**, the Health Service itself, too.

```

CimInstance Cluster;
List<CimInstance> Nodes;
List<CimInstance> Volumes;
CimInstance HealthService;

public void DiscoverObjects(CimSession Session)
{
    // Get MSFT_StorageSubSystem for Storage Spaces Direct
    Cluster = Session.QueryInstances(@"root\microsoft\windows\storage", "WQL", "SELECT * FROM
MSFT_StorageSubSystem")
        .First(Instance =>
        (Instance.CimInstanceProperties["FriendlyName"].Value.ToString()).Contains("Cluster"));

    // Get MSFT_StorageNode for each cluster node
    Nodes = Session.EnumerateAssociatedInstances(Cluster.CimSystemProperties.Namespace,
        Cluster, "MSFT_StorageSubSystemToStorageNode", null, "StorageSubSystem", "StorageNode").ToList();

    // Get MSFT_Volumes for each data volume
    Volumes = Session.EnumerateAssociatedInstances(Cluster.CimSystemProperties.Namespace,
        Cluster, "MSFT_StorageSubSystemToVolume", null, "StorageSubSystem", "Volume").ToList();

    // Get MSFT_StorageHealth itself
    HealthService = Session.EnumerateAssociatedInstances(Cluster.CimSystemProperties.Namespace,
        Cluster, "MSFT_StorageSubSystemToStorageHealth", null, "StorageSubSystem", "StorageHealth").First();
}

```

These are the same objects you get in PowerShell using cmdlets like **Get-StorageSubSystem**, **Get-**

StorageNode, and **Get-Volume**.

You can access all the same properties, documented at [Storage Management API Classes](#).

```
...
using System.Diagnostics;

foreach (CimInstance Node in Nodes)
{
    // For illustration, write each node's Name to the console. You could also write State (up/down), or
    // anything else!
    Debug.WriteLine("Discovered Node " + Node.CimInstanceProperties["Name"].Value.ToString());
}
```

Invoke **GetReport** to begin streaming samples of an expert-curated list of essential metrics, which are collected efficiently and aggregated dynamically across nodes, with built-in logic to detect cluster membership. Samples will arrive every second thereafter. All values are real-time and point-in-time only.

Metrics can be streamed for three scopes: the cluster, any node, or any volume.

The complete list of metrics available at each scope in Windows Server 2016 is documented below.

IObserver.OnNext()

This sample code uses the [Observer Design Pattern](#) to implement an Observer whose **OnNext()** method will be invoked when each new sample of metrics arrives. Its **OnCompleted()** method will be called if/when streaming ends. For example, you might use it to reinitiate streaming, so it continues indefinitely.

```

class MetricsObserver<T> : IObservable<T>
{
    public void OnNext(T Result)
    {
        // Cast
        CimMethodStreamedResult StreamedResult = Result as CimMethodStreamedResult;

        if (StreamedResult != null)
        {
            // For illustration, you could store the metrics in this dictionary
            Dictionary<string, string> Metrics = new Dictionary<string, string>();

            // Unpack
            CimInstance Report = (CimInstance)StreamedResult.ItemValue;
            IEnumerable<CimInstance> Records =
            (IEnumerable<CimInstance>)Report.CimInstanceProperties["Records"].Value;
            foreach (CimInstance Record in Records)
            {
                /// Each Record has "Name", "Value", and "Units"
                Metrics.Add(
                    Record.CimInstanceProperties["Name"].Value.ToString(),
                    Record.CimInstanceProperties["Value"].Value.ToString()
                );
            }

            // TODO: Whatever you want!
        }
    }
    public void OnError(Exception e)
    {
        // Handle Exceptions
    }
    public void OnCompleted()
    {
        // Reinvoke BeginStreamingMetrics(), defined in the next section
    }
}

```

Begin streaming

With the Observer defined, you can begin streaming.

Specify the target **CimInstance** to which you want the metrics scoped. It can be the cluster, any node, or any volume.

The count parameter is the number of samples before streaming ends.

```

CimInstance Target = Cluster; // From among the objects discovered in DiscoverObjects()

public void BeginStreamingMetrics(CimSession Session, CimInstance HealthService, CimInstance Target)
{
    // Set Parameters
    CimMethodParametersCollection MetricsParams = new CimMethodParametersCollection();
    MetricsParams.Add(CimMethodParameter.Create("TargetObject", Target, CimType.Instance, CimFlags.In));
    MetricsParams.Add(CimMethodParameter.Create("Count", 999, CimType.UInt32, CimFlags.In));
    // Enable WMI Streaming
    CimOperationOptions Options = new CimOperationOptions();
    Options.EnableMethodResultStreaming = true;
    // Invoke API
    CimAsyncMultipleResults<CimMethodResultBase> InvokeHandler;
    InvokeHandler = Session.InvokeMethodAsync(
        HealthService.CimSystemProperties.Namespace, HealthService, "GetReport", MetricsParams, Options
    );
    // Subscribe the Observer
    MetricsObserver<CimMethodResultBase> Observer = new MetricsObserver<CimMethodResultBase>(this);
    IDisposable Disposable = InvokeHandler.Subscribe(Observer);
}

```

Needless to say, these metrics can be visualized, stored in a database, or used in whatever way you see fit.

Properties of reports

Every sample of metrics is one "report" which contains many "records" corresponding to individual metrics.

For the full schema, inspect the **MSFT_StorageHealthReport** and **MSFT_HealthRecord** classes in *storagewmi.mof*.

Each metric has just three properties, per this table.

PROPERTY	EXAMPLE
Name	IOLatencyAverage
Value	0.00021
Units	3

Units = { 0, 1, 2, 3, 4 }, where 0 = "Bytes", 1 = "BytesPerSecond", 2 = "CountPerSecond", 3 = "Seconds", or 4 = "Percentage".

Coverage

Below are the metrics available for each scope in Windows Server 2016.

MSFT_StorageSubSystem

NAME	UNITS
CPUUsage	4
CapacityPhysicalPooledAvailable	0
CapacityPhysicalPooledTotal	0
CapacityPhysicalTotal	0

NAME	UNITS
CapacityPhysicalUnpooled	0
CapacityVolumesAvailable	0
CapacityVolumesTotal	0
IOLatencyAverage	3
IOLatencyRead	3
IOLatencyWrite	3
IOPSRead	2
IOPSTotal	2
IOPSWrite	2
IOThroughputRead	1
IOThroughputTotal	1
IOThroughputWrite	1
MemoryAvailable	0
MemoryTotal	0

MSFT_StorageNode

NAME	UNITS
CPUUsage	4
IOLatencyAverage	3
IOLatencyRead	3
IOLatencyWrite	3
IOPSRead	2
IOPSTotal	2
IOPSWrite	2
IOThroughputRead	1
IOThroughputTotal	1

NAME	UNITS
IOThroughputWrite	1
MemoryAvailable	0
MemoryTotal	0

MSFT_Volume

NAME	UNITS
CapacityAvailable	0
CapacityTotal	0
IOLatencyAverage	3
IOLatencyRead	3
IOLatencyWrite	3
IOPSRead	2
IOPSTotal	2
IOPSWrite	2
IOThroughputRead	1
IOThroughputTotal	1
IOThroughputWrite	1

See also

- [Health Service in Windows Server 2016](#)

Health Service faults

2/20/2018 • 13 minutes to read • [Edit Online](#)

Applies to Windows Server 2016

What are faults

The Health Service constantly monitors your Storage Spaces Direct cluster to detect problems and generate "faults". One new cmdlet displays any current faults, allowing you to easily verify the health of your deployment without looking at every entity or feature in turn. Faults are designed to be precise, easy to understand, and actionable.

Each fault contains five important fields:

- Severity
- Description of the problem
- Recommended next step(s) to address the problem
- Identifying information for the faulting entity
- Its physical location (if applicable)

For example, here is a typical fault:

```
Severity: MINOR
Reason: Connectivity has been lost to the physical disk.
Recommendation: Check that the physical disk is working and properly connected.
Part: Manufacturer Contoso, Model XYZ9000, Serial 123456789
Location: Seattle DC, Rack B07, Node 4, Slot 11
```

NOTE

The physical location is derived from your fault domain configuration. For more information about fault domains, see [Fault Domains in Windows Server 2016](#). If you do not provide this information, the location field will be less helpful - for example, it may only show the slot number.

Root cause analysis

The Health Service can assess the potential causality among faulting entities to identify and combine faults which are consequences of the same underlying problem. By recognizing chains of effect, this makes for less chatty reporting. For example, if a server is down, it is expected that any drives within the server will also be without connectivity. Therefore, only one fault will be raised for the root cause - in this case, the server.

Usage in PowerShell

To see any current faults in PowerShell, run this cmdlet:

```
Get-StorageSubSystem Cluster* | Debug-StorageSubSystem
```

This returns any faults which affect the overall Storage Spaces Direct cluster. Most often, these faults relate to hardware or configuration. If there are no faults, this cmdlet will return nothing.

NOTE

In a non-production environment, and at your own risk, you can experiment with this feature by triggering faults yourself - for example, by removing one physical disk or shutting down one node. Once the fault has appeared, re-insert the physical disk or restart the node and the fault will disappear again.

You can also view faults that are affecting only specific volumes or file shares with the following cmdlets:

```
Get-Volume -FileSystemLabel <Label> | Debug-Volume
```

```
Get-FileShare -Name <Name> | Debug-FileShare
```

This returns any faults that affect only the specific volume or file share. Most often, these faults relate to capacity planning, data resiliency, or features like Storage Quality-of-Service or Storage Replica.

Usage in .NET and C#

Connect

In order to query the Health Service, you will need to establish a **CimSession** with the cluster. To do so, you will need some things that are only available in full .NET, meaning you cannot readily do this directly from a web or mobile app. These code samples will use C#, the most straightforward choice for this data access layer.

```
...
using System.Security;
using Microsoft.Management.Infrastructure;

public CimSession Connect(string Domain = "...", string Computer = "...", string Username = "...", string
Password = "...")
{
    SecureString PasswordSecureString = new SecureString();
    foreach (char c in Password)
    {
        PasswordSecureString.AppendChar(c);
    }

    CimCredential Credentials = new CimCredential(
        PasswordAuthenticationMechanism.Default, Domain, Username, PasswordSecureString);
    WsManSessionOptions SessionOptions = new WsManSessionOptions();
    SessionOptions.AddDestinationCredentials(Credentials);
    Session = CimSession.Create(Computer, SessionOptions);
    return Session;
}
```

The provided Username should be a local Administrator of the target Computer.

It is recommended that you construct the Password **SecureString** directly from user input in real-time, so their password is never stored in memory in cleartext. This helps mitigate a variety of security concerns. But in practice, constructing it as above is common for prototyping purposes.

Discover objects

With the **CimSession** established, you can query Windows Management Instrumentation (WMI) on the cluster.

Before you can get Faults or Metrics, you will need to get instances of several relevant objects. First, the **MSFT_StorageSubSystem** which represents Storage Spaces Direct on the cluster. Using that, you can get every **MSFT_StorageNode** in the cluster, and every **MSFT_Volume**, the data volumes. Finally, you will need the **MSFT_StorageHealth**, the Health Service itself, too.

```

CimInstance Cluster;
List<CimInstance> Nodes;
List<CimInstance> Volumes;
CimInstance HealthService;

public void DiscoverObjects(CimSession Session)
{
    // Get MSFT_StorageSubSystem for Storage Spaces Direct
    Cluster = Session.QueryInstances(@"root\microsoft\windows\storage", "WQL", "SELECT * FROM
MSFT_StorageSubSystem")
        .First(Instance =>
(InInstance.CimInstanceProperties["FriendlyName"].Value.ToString()).Contains("Cluster"));

    // Get MSFT_StorageNode for each cluster node
    Nodes = Session.EnumerateAssociatedInstances(Cluster.CimSystemProperties.Namespace,
        Cluster, "MSFT_StorageSubSystemToStorageNode", null, "StorageSubSystem", "StorageNode").ToList();

    // Get MSFT_Volumes for each data volume
    Volumes = Session.EnumerateAssociatedInstances(Cluster.CimSystemProperties.Namespace,
        Cluster, "MSFT_StorageSubSystemToVolume", null, "StorageSubSystem", "Volume").ToList();

    // Get MSFT_StorageHealth itself
    HealthService = Session.EnumerateAssociatedInstances(Cluster.CimSystemProperties.Namespace,
        Cluster, "MSFT_StorageSubSystemToStorageHealth", null, "StorageSubSystem", "StorageHealth").First();
}

```

These are the same objects you get in PowerShell using cmdlets like **Get-StorageSubSystem**, **Get-StorageNode**, and **Get-Volume**.

You can access all the same properties, documented at [Storage Management API Classes](#).

```

...
using System.Diagnostics;

foreach (CimInstance Node in Nodes)
{
    // For illustration, write each node's Name to the console. You could also write State (up/down), or
    // anything else!
    Debug.WriteLine("Discovered Node " + Node.CimInstanceProperties["Name"].Value.ToString());
}

```

Query faults

Invoke **Diagnose** to get any current faults scoped to the target **CimInstance**, which be the cluster or any volume.

The complete list of faults available at each scope in Windows Server 2016 is documented below.

```

public void GetFaults(CimSession Session, CimInstance Target)
{
    // Set Parameters (None)
    CimMethodParametersCollection FaultsParams = new CimMethodParametersCollection();
    // Invoke API
    CimMethodResult Result = Session.InvokeMethod(Target, "Diagnose", FaultsParams);
    IEnumerable<CimInstance> DiagnoseResults =
    (IEnumerable<CimInstance>)Result.OutParameters["DiagnoseResults"].Value;
    // Unpack
    if (DiagnoseResults != null)
    {
        foreach (CimInstance DiagnoseResult in DiagnoseResults)
        {
            // TODO: Whatever you want!
        }
    }
}

```

Optional: MyFault class

It may make sense for you to construct and persist your own representation of faults. For example, this **MyFault** class stores several key properties of faults, including the **FaultId**, which can be used later to associate update or remove notifications, or to deduplicate in the event that the same fault is detected multiple times, for whatever reason.

```

public class MyFault {
    public String FaultId { get; set; }
    public String Reason { get; set; }
    public String Severity { get; set; }
    public String Description { get; set; }
    public String Location { get; set; }

    // Constructor
    public MyFault(CimInstance DiagnoseResult)
    {
        CimKeyedCollection<CimProperty> Properties = DiagnoseResult.CimInstanceProperties;
        FaultId    = Properties["FaultId"].Value.ToString();
        Reason     = Properties["Reason"].Value.ToString();
        Severity   = Properties["PerceivedSeverity"].Value.ToString();
        Description = Properties["FaultingObjectDescription"].Value.ToString();
        Location   = Properties["FaultingObjectLocation"].Value.ToString();
    }
}

```

```

List<MyFault> Faults = new List<MyFault>;

foreach (CimInstance DiagnoseResult in DiagnoseResults)
{
    Faults.Add(new Fault(DiagnoseResult));
}

```

The complete list of properties in each fault (**DiagnoseResult**) is documented below.

Fault events

When Faults are created, removed, or updated, the Health Service generates WMI events. These are essential to keeping your application state in sync without frequent polling, and can help with things like determining when to send email alerts, for example. To subscribe to these events, this sample code uses the Observer Design Pattern again.

First, subscribe to **MSFT_StorageFaultEvent** events.

```

public void ListenForFaultEvents()
{
    IObservable<CimSubscriptionResult> Events = Session.SubscribeAsync(
        @"root\microsoft\windows\storage", "WQL", "SELECT * FROM MSFT_StorageFaultEvent");
    // Subscribe the Observer
    FaultsObserver<CimSubscriptionResult> Observer = new FaultsObserver<CimSubscriptionResult>(this);
    IDisposable Disposable = Events.Subscribe(Observer);
}

```

Next, implement an Observer whose **OnNext()** method will be invoked whenever a new event is generated.

Each event contains **ChangeType** indicating whether a fault is being created, removed, or updated, and the relevant **FaultId**.

In addition, they contain all the properties of the fault itself.

```

class FaultsObserver : IObservable
{
    public void OnNext(T Event)
    {
        // Cast
        CimSubscriptionResult SubscriptionResult = Event as CimSubscriptionResult;

        if (SubscriptionResult != null)
        {
            // Unpack
            CimKeyedCollection<CimProperty> Properties = SubscriptionResult.Instance.CimInstanceProperties;
            String ChangeType = Properties["ChangeType"].Value.ToString();
            String FaultId = Properties["FaultId"].Value.ToString();

            // Create
            if (ChangeType == "0")
            {
                Fault MyNewFault = new MyFault(SubscriptionResult.Instance);
                // TODO: Whatever you want!
            }
            // Remove
            if (ChangeType == "1")
            {
                // TODO: Use FaultId to find and delete whatever representation you have...
            }
            // Update
            if (ChangeType == "2")
            {
                // TODO: Use FaultId to find and modify whatever representation you have...
            }
        }
    }
    public void OnError(Exception e)
    {
        // Handle Exceptions
    }
    public void OnCompleted()
    {
        // Nothing
    }
}

```

Understand fault lifecycle

Faults are not intended to be marked "seen" or resolved by the user. They are created when the Health Service observes a problem, and they are removed automatically and only when the Health Service can no longer observe the problem. In general, this reflects that the problem has been fixed.

However, in some cases, faults may be rediscovered by the Health Service (e.g. after failover, or due to intermittent connectivity, etc.). For this reason, it may makes sense to persist your own representation of faults, so you can easily deduplicate. This is especially important if you send email alerts or equivalent.

Properties of faults

This table presents several key properties of the fault object. For the full schema, inspect the **MSFT_StorageDiagnoseResult** class in *storagewmi.mof*.

PROPERTY	EXAMPLE
FaultId	{12345-12345-12345-12345-12345}
FaultType	Microsoft.Health.FaultType.Volume.Capacity
Reason	"The volume is running out of available space."
PerceivedSeverity	5
FaultingObjectDescription	Contoso XYZ9000 S.N. 123456789
FaultingObjectLocation	Rack A06, RU 25, Slot 11
RecommendedActions	{"Expand the volume.", "Migrate workloads to other volumes."}

FaultId Unique within the scope of one cluster.

PerceivedSeverity PerceivedSeverity = { 4, 5, 6 } = { "Informational", "Warning", and "Error" }, or equivalent colors such as blue, yellow, and red.

FaultingObjectDescription Part information for hardware, typically blank for software objects.

FaultingObjectLocation Location information for hardware, typically blank for software objects.

RecommendedActions List of recommended actions, which are independent and in no particular order. Today, this list is often of length 1.

Properties of fault events

This table presents several key properties of the fault event. For the full schema, inspect the **MSFT_StorageFaultEvent** class in *storagewmi.mof*.

Note the **ChangeType**, which indicates whether a fault is being created, removed, or updated, and the **FaultId**. An event also contains all the properties of the affected fault.

PROPERTY	EXAMPLE
ChangeType	0
FaultId	{12345-12345-12345-12345-12345}
FaultType	Microsoft.Health.FaultType.Volume.Capacity
Reason	"The volume is running out of available space."
PerceivedSeverity	5

PROPERTY	EXAMPLE
FaultingObjectDescription	Contoso XYZ9000 S.N. 123456789
FaultingObjectLocation	Rack A06, RU 25, Slot 11
RecommendedActions	{"Expand the volume.", "Migrate workloads to other volumes."}

ChangeType ChangeType = { 0, 1, 2 } = { "Create", "Remove", "Update" }.

Coverage

In Windows Server 2016, the Health Service provides the following fault coverage:

PhysicalDisk (8)

FaultType: Microsoft.Health.FaultType.PhysicalDisk.FailedMedia

- Severity: Warning
- Reason: *"The physical disk has failed."*
- RecommendedAction: *"Replace the physical disk."*

FaultType: Microsoft.Health.FaultType.PhysicalDisk.LostCommunication

- Severity: Warning
- Reason: *"Connectivity has been lost to the physical disk."*
- RecommendedAction: *"Check that the physical disk is working and properly connected."*

FaultType: Microsoft.Health.FaultType.PhysicalDisk.Unresponsive

- Severity: Warning
- Reason: *"The physical disk is exhibiting recurring unresponsiveness."*
- RecommendedAction: *"Replace the physical disk."*

FaultType: Microsoft.Health.FaultType.PhysicalDisk.PredictiveFailure

- Severity: Warning
- Reason: *"A failure of the physical disk is predicted to occur soon."*
- RecommendedAction: *"Replace the physical disk."*

FaultType: Microsoft.Health.FaultType.PhysicalDisk.UnsupportedHardware

- Severity: Warning
- Reason: *"The physical disk is quarantined because it is not supported by your solution vendor."*
- RecommendedAction: *"Replace the physical disk with supported hardware."*

FaultType: Microsoft.Health.FaultType.PhysicalDisk.UnsupportedFirmware

- Severity: Warning
- Reason: *"The physical disk is in quarantine because its firmware version is not supported by your solution vendor."*
- RecommendedAction: *"Update the firmware on the physical disk to the target version."*

FaultType: Microsoft.Health.FaultType.PhysicalDisk.UnrecognizedMetadata

- Severity: Warning
- Reason: *"The physical disk has unrecognised meta data."*
- RecommendedAction: *"This disk may contain data from an unknown storage pool. First make sure there is no useful data on this disk, then reset the disk."*

FaultType: Microsoft.Health.FaultType.PhysicalDisk.FailedFirmwareUpdate

- Severity: Warning

- Reason: *"Failed attempt to update firmware on the physical disk."*
- RecommendedAction: *"Try using a different firmware binary."*

Virtual Disk (2)

FaultType: `Microsoft.Health.FaultType.VirtualDisks.NeedsRepair`

- Severity: Informational
- Reason: *"Some data on this volume is not fully resilient. It remains accessible."*
- RecommendedAction: *"Restoring resiliency of the data."*

FaultType: `Microsoft.Health.FaultType.VirtualDisks.Detached`

- Severity: Critical
- Reason: *"The volume is inaccessible. Some data may be lost."*
- RecommendedAction: *"Check the physical and/or network connectivity of all storage devices. You may need to restore from backup."*

Pool Capacity (1)

FaultType: `Microsoft.Health.FaultType.StoragePool.InsufficientReserveCapacityFault`

- Severity: Warning
- Reason: *"The storage pool does not have the minimum recommended reserve capacity. This may limit your ability to restore data resiliency in the event of drive failure(s)."*
- RecommendedAction: *"Add additional capacity to the storage pool, or free up capacity. The minimum recommended reserve varies by deployment, but is approximately 2 drives' worth of capacity."*

Volume Capacity (2)¹

FaultType: `Microsoft.Health.FaultType.Volume.Capacity`

- Severity: Warning
- Reason: *"The volume is running out of available space."*
- RecommendedAction: *"Expand the volume or migrate workloads to other volumes."*

FaultType: `Microsoft.Health.FaultType.Volume.Capacity`

- Severity: Critical
- Reason: *"The volume is running out of available space."*
- RecommendedAction: *"Expand the volume or migrate workloads to other volumes."*

Server (3)

FaultType: `Microsoft.Health.FaultType.Server.Down`

- Severity: Critical
- Reason: *"The server cannot be reached."*
- RecommendedAction: *"Start or replace server."*

FaultType: `Microsoft.Health.FaultType.Server.Isolated`

- Severity: Critical
- Reason: *"The server is isolated from the cluster due to connectivity issues."*
- RecommendedAction: *"If isolation persists, check the network(s) or migrate workloads to other nodes."*

FaultType: `Microsoft.Health.FaultType.Server.Quarantined`

- Severity: Critical
- Reason: *"The server is quarantined by the cluster due to recurring failures."*
- RecommendedAction: *"Replace the server or fix the network."*

Cluster (1)

FaultType: `Microsoft.Health.FaultType.ClusterQuorumWitness.Error`

- Severity: Critical

- Reason: *"The cluster is one server failure away from going down."*
- RecommendedAction: *"Check the witness resource, and restart as needed. Start or replace failed servers."*

Network Adapter/Interface (4)

FaultType: Microsoft.Health.FaultType.NetworkAdapter.Disconnected

- Severity: Warning
- Reason: *"The network interface has become disconnected."*
- RecommendedAction: *"Reconnect the network cable."*

FaultType: Microsoft.Health.FaultType.NetworkInterface.Missing

- Severity: Warning
- Reason: *"The server {server} has missing network adapter(s) connected to cluster network {cluster network}."*
- RecommendedAction: *"Connect the server to the missing cluster network."*

FaultType: Microsoft.Health.FaultType.NetworkAdapter.Hardware

- Severity: Warning
- Reason: *"The network interface has had a hardware failure."*
- RecommendedAction: *"Replace the network interface adapter."*

FaultType: Microsoft.Health.FaultType.NetworkAdapter.Disabled

- Severity: Warning
- Reason: *"The network interface {network interface} is not enabled and is not being used."*
- RecommendedAction: *"Enable the network interface."*

Enclosure (6)

FaultType: Microsoft.Health.FaultType.StorageEnclosure.LostCommunication

- Severity: Warning
- Reason: *"Communication has been lost to the storage enclosure."*
- RecommendedAction: *"Start or replace the storage enclosure."*

FaultType: Microsoft.Health.FaultType.StorageEnclosure.FanError

- Severity: Warning
- Reason: *"The fan at position {position} of the storage enclosure has failed."*
- RecommendedAction: *"Replace the fan in the storage enclosure."*

FaultType: Microsoft.Health.FaultType.StorageEnclosure.CurrentSensorError

- Severity: Warning
- Reason: *"The current sensor at position {position} of the storage enclosure has failed."*
- RecommendedAction: *"Replace a current sensor in the storage enclosure."*

FaultType: Microsoft.Health.FaultType.StorageEnclosure.VoltageSensorError

- Severity: Warning
- Reason: *"The voltage sensor at position {position} of the storage enclosure has failed."*
- RecommendedAction: *"Replace a voltage sensor in the storage enclosure."*

FaultType: Microsoft.Health.FaultType.StorageEnclosure.IoControllerError

- Severity: Warning
- Reason: *"The IO controller at position {position} of the storage enclosure has failed."*
- RecommendedAction: *"Replace an IO controller in the storage enclosure."*

FaultType: Microsoft.Health.FaultType.StorageEnclosure.TemperatureSensorError

- Severity: Warning
- Reason: *"The temperature sensor at position {position} of the storage enclosure has failed."*
- RecommendedAction: *"Replace a temperature sensor in the storage enclosure."*

Firmware Rollout (3)

FaultType: Microsoft.Health.FaultType.FaultDomain.FailedMaintenanceMode

- Severity: Warning
- Reason: *"Currently unable to make progress while performing firmware roll out."*
- RecommendedAction: *"Verify all storage spaces are healthy, and that no fault domain is currently in maintenance mode."*

FaultType: Microsoft.Health.FaultType.FaultDomain.FirmwareVerifyVersionFaile

- Severity: Warning
- Reason: *"Firmware roll out was cancelled due to unreadable or unexpected firmware version information after applying a firmware update."*
- RecommendedAction: *"Restart firmware roll out once the firmware issue has been resolved."*

FaultType: Microsoft.Health.FaultType.FaultDomain.TooManyFailedUpdates

- Severity: Warning
- Reason: *"Firmware roll out was cancelled due to too many physical disks failing a firmware update attempt."*
- RecommendedAction: *"Restart firmware roll out once the firmware issue has been resolved."*

Storage QoS (3)²

FaultType: Microsoft.Health.FaultType.StorQos.InsufficientThroughput

- Severity: Warning
- Reason: *"Storage throughput is insufficient to satisfy reserves."*
- RecommendedAction: *"Reconfigure Storage QoS policies."*

FaultType: Microsoft.Health.FaultType.StorQos.LostCommunication

- Severity: Warning
- Reason: *"The Storage QoS policy manager has lost communication with the volume."*
- RecommendedAction: *"Please reboot nodes {nodes}"*

FaultType: Microsoft.Health.FaultType.StorQos.MisconfiguredFlow

- Severity: Warning
- Reason: *"One or more storage consumers (usually Virtual Machines) are using a non-existent policy with id {id}."*
- RecommendedAction: *"Recreate any missing Storage QoS policies."*

¹ Indicates the volume has reached 80% full (minor severity) or 90% full (major severity).

² Indicates some .vhd(s) on the volume have not met their Minimum IOPS for over 10% (minor), 30% (major), or 50% (critical) of rolling 24-hour window.

NOTE

The health of storage enclosure components such as fans, power supplies, and sensors is derived from SCSI Enclosure Services (SES). If your vendor does not provide this information, the Health Service cannot display it.

See also

- [Health Service in Windows Server 2016](#)

Health Service actions

10/17/2017 • 2 minutes to read • [Edit Online](#)

Applies to Windows Server 2016

The Health Service is a new feature in Windows Server 2016 that improves the day-to-day monitoring and operational experience for clusters running Storage Spaces Direct.

Actions

The next section describes workflows which are automated by the Health Service. To verify that an action is indeed being taken autonomously, or to track its progress or outcome, the Health Service generates "Actions". Unlike logs, Actions disappear shortly after they have completed, and are intended primarily to provide insight into ongoing activity which may impact performance or capacity (e.g. restoring resiliency or rebalancing data).

Usage

One new PowerShell cmdlet displays all Actions:

```
Get-StorageHealthAction
```

Coverage

In Windows Server 2016, the **Get-StorageHealthAction** cmdlet can return any of the following information:

- Retiring failed, lost connectivity, or unresponsive physical disk
- Switching storage pool to use replacement physical disk
- Restoring full resiliency to data
- Rebalancing storage pool

See also

- [Health Service in Windows Server 2016](#)
- [Developer documentation, sample code, and API reference on MSDN](#)

Health Service settings

8/14/2017 • 2 minutes to read • [Edit Online](#)

Applies to Windows Server 2016

The Health Service is a new feature in Windows Server 2016 that improves the day-to-day monitoring and operational experience for clusters running Storage Spaces Direct.

Many of the parameters which govern the behavior of the Health Service are exposed as settings. You can modify these to tune the aggressiveness of faults or actions, turn certain behaviors on/off, and more.

Use the following PowerShell cmdlet to set or modify settings.

Usage

```
Get-StorageSubSystem Cluster* | Set-StorageHealthSetting -Name <SettingName> -Value <Value>
```

Example

```
Get-StorageSubSystem Cluster* | Set-StorageHealthSetting -Name  
"System.Storage.Volume.CapacityThreshold.Warning" -Value 70
```

Common settings

Some commonly modified settings are listed below, along with their default values.

Volume Capacity Threshold

```
"System.Storage.Volume.CapacityThreshold.Enabled" = True  
"System.Storage.Volume.CapacityThreshold.Warning" = 80  
"System.Storage.Volume.CapacityThreshold.Critical" = 90
```

Pool Reserve Capacity Threshold

```
"System.Storage.StoragePool.CheckPoolReserveCapacity.Enabled" = True
```

Physical Disk Lifecycle

```
"System.Storage.PhysicalDisk.AutoPool.Enabled" = True  
"System.Storage.PhysicalDisk.AutoRetire.OnLostCommunication.Enabled" = True  
"System.Storage.PhysicalDisk.AutoRetire.OnUnresponsive.Enabled" = True  
"System.Storage.PhysicalDisk.AutoRetire.DelayMs" = 900000 (i.e. 15 minutes)  
"System.Storage.PhysicalDisk.Unresponsive.Reset.CountResetIntervalSeconds" = 360 (i.e. 60 minutes)  
"System.Storage.PhysicalDisk.Unresponsive.Reset.CountAllowed" = 3
```

Supported Components Document

See the previous section.

Firmware Rollout

```
"System.Storage.PhysicalDisk.AutoFirmwareUpdate.SingleDrive.Enabled"      = True
"System.Storage.PhysicalDisk.AutoFirmwareUpdate.RollOut.Enabled"          = True
"System.Storage.PhysicalDisk.AutoFirmwareUpdate.RollOut.LongDelaySeconds" = 604800 (i.e. 7 days)
"System.Storage.PhysicalDisk.AutoFirmwareUpdate.RollOut.ShortDelaySeconds" = 86400 (i.e. 1 day)
"System.Storage.PhysicalDisk.AutoFirmwareUpdate.RollOut.LongDelayCount"    = 1
"System.Storage.PhysicalDisk.AutoFirmwareUpdate.RollOut.FailureTolerance"  = 3
```

Platform / Quiescence

```
"Platform.Quiescence.MinDelaySeconds" = 120 (i.e. 2 minutes)
"Platform.Quiescence.MaxDelaySeconds" = 420 (i.e. 7 minutes)
```

Metrics

```
"System.Reports.ReportingPeriodSeconds" = 1
```

Debugging

```
"System.LogLevel" = 4
```

See also

- [Health Service in Windows Server 2016](#)
- [Storage Spaces Direct in Windows Server 2016](#)

Configure and manage quorum

6/20/2018 • 20 minutes to read • [Edit Online](#)

Applies to: Windows Server 2016, Windows Server 2012 R2, Windows Server 2012

This topic provides background and steps to configure and manage the quorum in a Windows Server failover cluster.

Understanding quorum

The quorum for a cluster is determined by the number of voting elements that must be part of active cluster membership for that cluster to start properly or continue running. For a more detailed explanation, see the [understanding cluster and pool quorum doc](#).

Quorum configuration options

The quorum model in Windows Server is flexible. If you need to modify the quorum configuration for your cluster, you can use the Configure Cluster Quorum Wizard or the Failover Clusters Windows PowerShell cmdlets. For steps and considerations to configure the quorum, see [Configure the cluster quorum](#) later in this topic.

The following table lists the three quorum configuration options that are available in the Configure Cluster Quorum Wizard.

OPTION	DESCRIPTION
Use typical settings	The cluster automatically assigns a vote to each node and dynamically manages the node votes. If it is suitable for your cluster, and there is cluster shared storage available, the cluster selects a disk witness. This option is recommended in most cases, because the cluster software automatically chooses a quorum and witness configuration that provides the highest availability for your cluster.
Add or change the quorum witness	You can add, change, or remove a witness resource. You can configure a file share or disk witness. The cluster automatically assigns a vote to each node and dynamically manages the node votes.
Advanced quorum configuration and witness selection	You should select this option only when you have application-specific or site-specific requirements for configuring the quorum. You can modify the quorum witness, add or remove node votes, and choose whether the cluster dynamically manages node votes. By default, votes are assigned to all nodes, and the node votes are dynamically managed.

Depending on the quorum configuration option that you choose and your specific settings, the cluster will be configured in one of the following quorum modes:

MODE	DESCRIPTION
------	-------------

MODE	DESCRIPTION
Node majority (no witness)	Only nodes have votes. No quorum witness is configured. The cluster quorum is the majority of voting nodes in the active cluster membership.
Node majority with witness (disk or file share)	<p>Nodes have votes. In addition, a quorum witness has a vote. The cluster quorum is the majority of voting nodes in the active cluster membership plus a witness vote.</p> <p>A quorum witness can be a designated disk witness or a designated file share witness.</p>
No majority (disk witness only)	<p>No nodes have votes. Only a disk witness has a vote. The cluster quorum is determined by the state of the disk witness.</p> <p>The cluster has quorum if one node is available and communicating with a specific disk in the cluster storage. Generally, this mode is not recommended, and it should not be selected because it creates a single point of failure for the cluster.</p>

The following subsections will give you more information about advanced quorum configuration settings.

Witness configuration

As a general rule when you configure a quorum, the voting elements in the cluster should be an odd number. Therefore, if the cluster contains an even number of voting nodes, you should configure a disk witness or a file share witness. The cluster will be able to sustain one additional node down. In addition, adding a witness vote enables the cluster to continue running if half the cluster nodes simultaneously go down or are disconnected.

A disk witness is usually recommended if all nodes can see the disk. A file share witness is recommended when you need to consider multisite disaster recovery with replicated storage. Configuring a disk witness with replicated storage is possible only if the storage vendor supports read-write access from all sites to the replicated storage. **A Disk Witness isn't supported with Storage Spaces Direct.**

The following table provides additional information and considerations about the quorum witness types.

WITNESS TYPE	DESCRIPTION	REQUIREMENTS AND RECOMMENDATIONS
Disk witness	<ul style="list-style-type: none"> - Dedicated LUN that stores a copy of the cluster database - Most useful for clusters with shared (not replicated) storage 	<ul style="list-style-type: none"> - Size of LUN must be at least 512 MB - Must be dedicated to cluster use and not assigned to a clustered role - Must be included in clustered storage and pass storage validation tests - Cannot be a disk that is a Cluster Shared Volume (CSV) - Basic disk with a single volume - Does not need to have a drive letter - Can be formatted with NTFS or ReFS - Can be optionally configured with hardware RAID for fault tolerance - Should be excluded from backups and antivirus scanning - A Disk Witness isn't supported with Storage Spaces Direct

WITNESS TYPE	DESCRIPTION	REQUIREMENTS AND RECOMMENDATIONS
File share witness	<ul style="list-style-type: none"> - SMB file share that is configured on a file server running Windows Server - Does not store a copy of the cluster database - Maintains cluster information only in a witness.log file - Most useful for multisite clusters with replicated storage 	<ul style="list-style-type: none"> - Must have a minimum of 5 MB of free space - Must be dedicated to the single cluster and not used to store user or application data - Must have write permissions enabled for the computer object for the cluster name <p>The following are additional considerations for a file server that hosts the file share witness:</p> <ul style="list-style-type: none"> - A single file server can be configured with file share witnesses for multiple clusters. - The file server must be on a site that is separate from the cluster workload. This allows equal opportunity for any cluster site to survive if site-to-site network communication is lost. If the file server is on the same site, that site becomes the primary site, and it is the only site that can reach the file share. - The file server can run on a virtual machine if the virtual machine is not hosted on the same cluster that uses the file share witness. - For high availability, the file server can be configured on a separate failover cluster.

Node vote assignment

As an advanced quorum configuration option, you can choose to assign or remove quorum votes on a per-node basis. By default, all nodes are assigned votes. Regardless of vote assignment, all nodes continue to function in the cluster, receive cluster database updates, and can host applications.

You might want to remove votes from nodes in certain disaster recovery configurations. For example, in a multisite cluster, you could remove votes from the nodes in a backup site so that those nodes do not affect quorum calculations. This configuration is recommended only for manual failover across sites. For more information, see [Quorum considerations for disaster recovery configurations](#) later in this topic.

The configured vote of a node can be verified by looking up the **NodeWeight** common property of the cluster node by using the [Get-ClusterNodeWindows](#) PowerShell cmdlet. A value of 0 indicates that the node does not have a quorum vote configured. A value of 1 indicates that the quorum vote of the node is assigned, and it is managed by the cluster. For more information about management of node votes, see [Dynamic quorum management](#) later in this topic.

The vote assignment for all cluster nodes can be verified by using the **Validate Cluster Quorum** validation test.

Additional considerations for node vote assignment

- Node vote assignment is not recommended to enforce an odd number of voting nodes. Instead, you should configure a disk witness or file share witness. For more information, see [Witness configuration](#) later in this topic.
- If dynamic quorum management is enabled, only the nodes that are configured to have node votes assigned can have their votes assigned or removed dynamically. For more information, see [Dynamic quorum management](#) later in this topic.

Dynamic quorum management

In Windows Server 2012, as an advanced quorum configuration option, you can choose to enable dynamic quorum management by cluster. For more details on how dynamic quorum works, see [this explanation](#).

With dynamic quorum management, it is also possible for a cluster to run on the last surviving cluster node. By dynamically adjusting the quorum majority requirement, the cluster can sustain sequential node shutdowns to a single node.

The cluster-assigned dynamic vote of a node can be verified with the **DynamicWeight** common property of the cluster node by using the [Get-ClusterNode](#) Windows PowerShell cmdlet. A value of 0 indicates that the node does not have a quorum vote. A value of 1 indicates that the node has a quorum vote.

The vote assignment for all cluster nodes can be verified by using the **Validate Cluster Quorum** validation test.

Additional considerations for dynamic quorum management

- Dynamic quorum management does not allow the cluster to sustain a simultaneous failure of a majority of voting members. To continue running, the cluster must always have a quorum majority at the time of a node shutdown or failure.
- If you have explicitly removed the vote of a node, the cluster cannot dynamically add or remove that vote.
- When Storage Spaces Direct is enabled, the cluster can only support two node failures. This is explained more in the [pool quorum section](#)

General recommendations for quorum configuration

The cluster software automatically configures the quorum for a new cluster, based on the number of nodes configured and the availability of shared storage. This is usually the most appropriate quorum configuration for that cluster. However, it is a good idea to review the quorum configuration after the cluster is created, before placing the cluster into production. To view the detailed cluster quorum configuration, you can use the Validate a Configuration Wizard, or the [Test-Cluster](#) Windows PowerShell cmdlet, to run the **Validate Quorum Configuration** test. In Failover Cluster Manager, the basic quorum configuration is displayed in the summary information for the selected cluster, or you can review the information about quorum resources that returns when you run the [Get-ClusterQuorum](#) Windows PowerShell cmdlet.

At any time, you can run the **Validate Quorum Configuration** test to validate that the quorum configuration is optimal for your cluster. The test output indicates if a change to the quorum configuration is recommended and the settings that are optimal. If a change is recommended, you can use the Configure Cluster Quorum Wizard to apply the recommended settings.

After the cluster is in production, do not change the quorum configuration unless you have determined that the change is appropriate for your cluster. You might want to consider changing the quorum configuration in the following situations:

- Adding or evicting nodes
- Adding or removing storage
- A long-term node or witness failure
- Recovering a cluster in a multisite disaster recovery scenario

For more information about validating a failover cluster, see [Validate Hardware for a Failover Cluster](#).

Configure the cluster quorum

You can configure the cluster quorum settings by using Failover Cluster Manager or the Failover Clusters Windows PowerShell cmdlets.

IMPORTANT

It is usually best to use the quorum configuration that is recommended by the Configure Cluster Quorum Wizard. We recommend customizing the quorum configuration only if you have determined that the change is appropriate for your cluster. For more information, see [General recommendations for quorum configuration](#) in this topic.

Configure the cluster quorum settings

Membership in the local **Administrators** group on each clustered server, or equivalent, is the minimum permissions required to complete this procedure. Also, the account you use must be a domain user account.

NOTE

You can change the cluster quorum configuration without stopping the cluster or taking cluster resources offline.

Change the quorum configuration in a failover cluster by using Failover Cluster Manager

1. In Failover Cluster Manager, select or specify the cluster that you want to change.
2. With the cluster selected, under **Actions**, select **More Actions**, and then select **Configure Cluster Quorum Settings**. The Configure Cluster Quorum Wizard appears. Select **Next**.
3. On the **Select Quorum Configuration Option** page, select one of the three configuration options and complete the steps for that option. Before you configure the quorum settings, you can review your choices. For more information about the options, see [Overview of the quorum in a failover cluster](#), earlier in this topic.
 - To allow the cluster to automatically reset the quorum settings that are optimal for your current cluster configuration, select **Use typical settings** and then complete the wizard.
 - To add or change the quorum witness, select **Add or change the quorum witness**, and then complete the following steps. For information and considerations about configuring a quorum witness, see [Witness configuration](#) earlier in this topic.
 - a. On the **Select Quorum Witness** page, select an option to configure a disk witness or a file share witness. The wizard indicates the witness selection options that are recommended for your cluster.

NOTE

You can also select **Do not configure a quorum witness** and then complete the wizard. If you have an even number of voting nodes in your cluster, this may not be a recommended configuration.

- b. If you select the option to configure a disk witness, on the **Configure Storage Witness** page, select the storage volume that you want to assign as the disk witness, and then complete the wizard.
 - c. If you select the option to configure a file share witness, on the **Configure File Share Witness** page, type or browse to a file share that will be used as the witness resource, and then complete the wizard.
- To configure quorum management settings and to add or change the quorum witness, select **Advanced quorum configuration and witness selection**, and then complete the following steps. For information and considerations about the advanced quorum configuration settings, see [Node vote assignment](#) and [Dynamic quorum management](#) earlier in this topic.
 - a. On the **Select Voting Configuration** page, select an option to assign votes to nodes. By default, all nodes are assigned a vote. However, for certain scenarios, you can assign votes

only to a subset of the nodes.

NOTE

You can also select **No Nodes**. This is generally not recommended, because it does not allow nodes to participate in quorum voting, and it requires configuring a disk witness. This disk witness becomes the single point of failure for the cluster.

- b. On the **Configure Quorum Management** page, you can enable or disable the **Allow cluster to dynamically manage the assignment of node votes** option. Selecting this option generally increases the availability of the cluster. By default the option is enabled, and it is strongly recommended to not disable this option. This option allows the cluster to continue running in failure scenarios that are not possible when this option is disabled.
- c. On the **Select Quorum Witness** page, select an option to configure a disk witness or a file share witness. The wizard indicates the witness selection options that are recommended for your cluster.

NOTE

You can also select **Do not configure a quorum witness**, and then complete the wizard. If you have an even number of voting nodes in your cluster, this may not be a recommended configuration.

- d. If you select the option to configure a disk witness, on the **Configure Storage Witness** page, select the storage volume that you want to assign as the disk witness, and then complete the wizard.
 - e. If you select the option to configure a file share witness, on the **Configure File Share Witness** page, type or browse to a file share that will be used as the witness resource, and then complete the wizard.
4. Select **Next**. Confirm your selections on the confirmation page that appears, and then select **Next**.

After the wizard runs and the **Summary** page appears, if you want to view a report of the tasks that the wizard performed, select **View Report**. The most recent report will remain in the `systemroot\Cluster\Reports` folder with the name **QuorumConfiguration.mht**.

NOTE

After you configure the cluster quorum, we recommend that you run the **Validate Quorum Configuration** test to verify the updated quorum settings.

Windows PowerShell equivalent commands

The following examples show how to use the `Set-ClusterQuorum` cmdlet and other Windows PowerShell cmdlets to configure the cluster quorum.

The following example changes the quorum configuration on cluster `CONTOSO-FC1` to a simple node majority configuration with no quorum witness.

```
Set-ClusterQuorum -Cluster CONTOSO-FC1 -NodeMajority
```

The following example changes the quorum configuration on the local cluster to a node majority with witness configuration. The disk resource named `Cluster Disk 2` is configured as a disk witness.

```
Set-ClusterQuorum -NodeAndDiskMajority "Cluster Disk 2"
```

The following example changes the quorum configuration on the local cluster to a node majority with witness configuration. The file share resource named `\\CONTOSO-FS\fsw` is configured as a file share witness.

```
Set-ClusterQuorum -NodeAndFileShareMajority "\\fileservers\fsw"
```

The following example removes the quorum vote from node *ContosoFCNode1* on the local cluster.

```
(Get-ClusterNode ContosoFCNode1).NodeWeight=0
```

The following example adds the quorum vote to node *ContosoFCNode1* on the local cluster.

```
(Get-ClusterNode ContosoFCNode1).NodeWeight=1
```

The following example enables the **DynamicQuorum** property of the cluster *CONTOSO-FC1* (if it was previously disabled):

```
(Get-Cluster CONTOSO-FC1).DynamicQuorum=1
```

Recover a cluster by starting without quorum

A cluster that does not have enough quorum votes will not start. As a first step, you should always confirm the cluster quorum configuration and investigate why the cluster no longer has quorum. This might happen if you have nodes that stopped responding, or if the primary site is not reachable in a multisite cluster. After you identify the root cause for the cluster failure, you can use the recovery steps described in this section.

NOTE

- If the Cluster service stops because quorum is lost, Event ID 1177 appears in the system log.
- It is always necessary to investigate why the cluster quorum was lost.
- It is always preferable to bring a node or quorum witness to a healthy state (join the cluster) rather than starting the cluster without quorum.

Force start cluster nodes

After you determine that you cannot recover your cluster by bringing the nodes or quorum witness to a healthy state, forcing your cluster to start becomes necessary. Forcing the cluster to start overrides your cluster quorum configuration settings and starts the cluster in **ForceQuorum** mode.

Forcing a cluster to start when it does not have quorum may be especially useful in a multisite cluster. Consider a disaster recovery scenario with a cluster that contains separately located primary and backup sites, *SiteA* and *SiteB*. If there is a genuine disaster at *SiteA*, it could take a significant amount of time for the site to come back online. You would likely want to force *SiteB* to come online, even though it does not have quorum.

When a cluster is started in **ForceQuorum** mode, and after it regains sufficient quorum votes, the cluster automatically leaves the forced state, and it behaves normally. Hence, it is not necessary to start the cluster again normally. If the cluster loses a node and it loses quorum, it goes offline again because it is no longer in the forced state. To bring it back online when it does not have quorum requires forcing the cluster to start without quorum.

IMPORTANT

- After a cluster is force started, the administrator is in full control of the cluster.
- The cluster uses the cluster configuration on the node where the cluster is force started, and replicates it to all other nodes that are available.
- If you force the cluster to start without quorum, all quorum configuration settings are ignored while the cluster remains in **ForceQuorum** mode. This includes specific node vote assignments and dynamic quorum management settings.

Prevent quorum on remaining cluster nodes

After you have force started the cluster on a node, it is necessary to start any remaining nodes in your cluster with a setting to prevent quorum. A node started with a setting that prevents quorum indicates to the Cluster service to join an existing running cluster instead of forming a new cluster instance. This prevents the remaining nodes from forming a split cluster that contains two competing instances.

This becomes necessary when you need to recover your cluster in some multisite disaster recovery scenarios after you have force started the cluster on your backup site, *SiteB*. To join the force started cluster in *SiteB*, the nodes in your primary site, *SiteA*, need to be started with the quorum prevented.

IMPORTANT

After a cluster is force started on a node, we recommend that you always start the remaining nodes with the quorum prevented.

Here's how to recover the cluster with Failover Cluster Manager:

1. In Failover Cluster Manager, select or specify the cluster you want to recover.
2. With the cluster selected, under **Actions**, select **Force Cluster Start**.

Failover Cluster Manager force starts the cluster on all nodes that are reachable. The cluster uses the current cluster configuration when starting.

NOTE

- To force the cluster to start on a specific node that contains a cluster configuration that you want to use, you must use the Windows PowerShell cmdlets or equivalent command-line tools as presented after this procedure.
- If you use Failover Cluster Manager to connect to a cluster that is force started, and you use the **Start Cluster Service** action to start a node, the node is automatically started with the setting that prevents quorum.

Windows PowerShell equivalent commands (Start-ClusterNode)

The following example shows how to use the **Start-ClusterNode** cmdlet to force start the cluster on node *ContosoFCNode1*.

```
Start-ClusterNode -Node ContosoFCNode1 -FQ
```

Alternatively, you can type the following command locally on the node:

```
Net Start ClusSvc /FQ
```

The following example shows how to use the **Start-ClusterNode** cmdlet to start the Cluster service with the quorum prevented on node *ContosoFCNode1*.

```
Start-ClusterNode -Node ContosoFCNode1 -PQ
```

Alternatively, you can type the following command locally on the node:

```
Net Start ClusSvc /PQ
```

Quorum considerations for disaster recovery configurations

This section summarizes characteristics and quorum configurations for two multisite cluster configurations in disaster recovery deployments. The quorum configuration guidelines differ depending on if you need automatic failover or manual failover for workloads between the sites. Your configuration usually depends on the service level agreements (SLAs) that are in place in your organization to provide and support clustered workloads in the event of a failure or disaster at a site.

Automatic failover

In this configuration, the cluster consists of two or more sites that can host clustered roles. If a failure occurs at any site, the clustered roles are expected to automatically fail over to the remaining sites. Therefore, the cluster quorum must be configured so that any site can sustain a complete site failure.

The following table summarizes considerations and recommendations for this configuration.

ITEM	DESCRIPTION
Number of node votes per site	Should be equal
Node vote assignment	Node votes should not be removed because all nodes are equally important
Dynamic quorum management	Should be enabled
Witness configuration	File share witness is recommended, configured in a site that is separate from the cluster sites
Workloads	Workloads can be configured on any of the sites

Additional considerations for automatic failover

- Configuring the file share witness in a separate site is necessary to give each site an equal opportunity to survive. For more information, see [Witness configuration](#) earlier in this topic.

Manual failover

In this configuration, the cluster consists of a primary site, *SiteA*, and a backup (recovery) site, *SiteB*. Clustered roles are hosted on *SiteA*. Because of the cluster quorum configuration, if a failure occurs at all nodes in *SiteA*, the cluster stops functioning. In this scenario the administrator must manually fail over the cluster services to *SiteB* and perform additional steps to recover the cluster.

The following table summarizes considerations and recommendations for this configuration.

ITEM	DESCRIPTION
Number of node votes per site	Can differ

ITEM	DESCRIPTION
Node vote assignment	<ul style="list-style-type: none"> - Node votes should not be removed from nodes at the primary site, <i>SiteA</i> - Node votes should be removed from nodes at the backup site, <i>SiteB</i> - If a long-term outage occurs at <i>SiteA</i>, votes must be assigned to nodes at <i>SiteB</i> to enable a quorum majority at that site as part of recovery
Dynamic quorum management	Should be enabled
Witness configuration	<ul style="list-style-type: none"> - Configure a witness if there is an even number of nodes at <i>SiteA</i> - If a witness is needed, configure either a file share witness or a disk witness that is accessible only to nodes in <i>SiteA</i> (sometimes called an asymmetric disk witness)
Workloads	Use preferred owners to keep workloads running on nodes at <i>SiteA</i>

Additional considerations for manual failover

- Only the nodes at *SiteA* are initially configured with quorum votes. This is necessary to ensure that the state of nodes at *SiteB* does not affect the cluster quorum.
- Recovery steps can vary depending on if *SiteA* sustains a temporary failure or a long-term failure.

More information

- [Failover Clustering](#)
- [Failover Clusters Windows PowerShell cmdlets](#)
- [Understanding Cluster and Pool Quorum](#)

Troubleshooting a Failover Cluster using Windows Error Reporting

4/11/2018 • 9 minutes to read • [Edit Online](#)

Applies to: Windows Server 2016, Windows Server

Windows Error Reporting (WER) is a flexible event-based feedback infrastructure designed to help advanced administrators or Tier 3 support gather information about the hardware and software problems that Windows can detect, report the information to Microsoft, and provide users with any available solutions. This [reference](#) provides descriptions and syntax for all WindowsErrorReporting cmdlets.

The information on troubleshooting presented below will be helpful for troubleshooting advanced issues that have been escalated and that may require data to be sent to Microsoft for triaging.

Enabling event channels

When Windows Server is installed, many event channels are enabled by default. But sometimes when diagnosing an issue, we want to be able to enable some of these event channels since it will help in triaging and diagnosing system issues.

You could enable additional event channels on each server node in your cluster as needed; however, this approach presents two problems:

1. You have to remember to enable the same event channels on every new server node that you add to your cluster.
2. When diagnosing, it can be tedious to enable specific event channels, reproduce the error, and repeat this process until you root cause.

To avoid these issues, you can enable event channels on cluster startup. The list of enabled event channels on your cluster can be configured using the public property **EnabledEventLogs**. By default, the following event channels are enabled:

```
PS C:\Windows\system32> (get-cluster).EnabledEventLogs
```

Here's an example of the output:

```
Microsoft-Windows-Hyper-V-VmSwitch-Diagnostic,4,0xFFFFFFFFD
Microsoft-Windows-SMBDirect/Debug,4
Microsoft-Windows-SMBServer/Analytic
Microsoft-Windows-Kernel-LiveDump/Analytic
```

The **EnabledEventLogs** property is a multistring, where each string is in the form: **channel-name**, **log-level**, **keyword-mask**. The **keyword-mask** can be a hexadecimal (prefix 0x), octal (prefix 0), or decimal number (no prefix) number. For instance, to add a new event channel to the list and to configure both **log-level** and **keyword-mask** you can run:

```
(get-cluster).EnabledEventLogs += "Microsoft-Windows-WinINet/Analytic,2,321"
```

If you want to set the **log-level** but keep the **keyword-mask** at its default value, you can use either of the

following commands:

```
(get-cluster).EnabledEventLogs += "Microsoft-Windows-WinINet/Analytic,2"
(get-cluster).EnabledEventLogs += "Microsoft-Windows-WinINet/Analytic,2,"
```

If you want to keep the **log-level** at its default value, but set the **keyword-mask** you can run the following command:

```
(get-cluster).EnabledEventLogs += "Microsoft-Windows-WinINet/Analytic,,0xf1"
```

If you want to keep both the **log-level** and the **keyword-mask** at their default values, you can run any of the following commands:

```
(get-cluster).EnabledEventLogs += "Microsoft-Windows-WinINet/Analytic"
(get-cluster).EnabledEventLogs += "Microsoft-Windows-WinINet/Analytic,"
(get-cluster).EnabledEventLogs += "Microsoft-Windows-WinINet/Analytic,"
```

These event channels will be enabled on every cluster node when the cluster service starts or whenever the **EnabledEventLogs** property is changed.

Gathering Logs

After you have enabled event channels, you can use the **DumpLogQuery** to gather logs. The public resource type property **DumpLogQuery** is a multistring value. Each string is an [XPath query as described here](#).

When troubleshooting, if you need to collect additional event channels, you can modify the **DumpLogQuery** property by adding additional queries or modifying the list.

To do this, first test your XPath query using the [get-WinEvent](#) PowerShell cmdlet:

```
get-WinEvent -FilterXML "<QueryList><Query><Select Path='Microsoft-Windows-GroupPolicy/Operational'>*[System[TimeCreated[timediff(@SystemTime) &gt;= 600000]]]</Select></Query></QueryList>"
```

Next, append your query to the **DumpLogQuery** property of the resource:

```
(Get-ClusterResourceType -Name "Physical Disk").DumpLogQuery += "<QueryList><Query><Select Path='Microsoft-Windows-GroupPolicy/Operational'>*[System[TimeCreated[timediff(@SystemTime) &gt;= 600000]]]</Select></Query></QueryList>"
```

And if you want to get a list of queries to use, run:

```
(Get-ClusterResourceType -Name "Physical Disk").DumpLogQuery
```

Gathering Windows Error Reporting reports

Windows Error Reporting Reports are stored in **%ProgramData%\Microsoft\Windows\WER**

Inside the **WER** folder, the **ReportsQueue** folder contains reports that are waiting to be uploaded to Watson.

```
PS C:\Windows\system32> dir c:\ProgramData\Microsoft\Windows\WER\ReportQueue
```

Here's an example of the output:

Volume in drive C is INSTALLTO
Volume Serial Number is 4031-E397

Directory of C:\ProgramData\Microsoft\Windows\WER\ReportQueue

```
<date> <time> <DIR>      .
<date> <time> <DIR>      ..
<date> <time> <DIR>      Critical_Physical
Disk_1cbd8ffecbc8a1a0e7819e4262e3ece2909a157a_00000000_02d10a3f
<date> <time> <DIR>      Critical_Physical
Disk_1cbd8ffecbc8a1a0e7819e4262e3ece2909a157a_00000000_0588dd06
<date> <time> <DIR>      Critical_Physical
Disk_1cbd8ffecbc8a1a0e7819e4262e3ece2909a157a_00000000_10d55ef5
<date> <time> <DIR>      Critical_Physical
Disk_1cbd8ffecbc8a1a0e7819e4262e3ece2909a157a_00000000_13258c8c
<date> <time> <DIR>      Critical_Physical
Disk_1cbd8ffecbc8a1a0e7819e4262e3ece2909a157a_00000000_13a8c4ac
<date> <time> <DIR>      Critical_Physical
Disk_1cbd8ffecbc8a1a0e7819e4262e3ece2909a157a_00000000_13dcf4d3
<date> <time> <DIR>      Critical_Physical
Disk_1cbd8ffecbc8a1a0e7819e4262e3ece2909a157a_00000000_1721a0b0
<date> <time> <DIR>      Critical_Physical
Disk_1cbd8ffecbc8a1a0e7819e4262e3ece2909a157a_00000000_1839758a
<date> <time> <DIR>      Critical_Physical
Disk_1cbd8ffecbc8a1a0e7819e4262e3ece2909a157a_00000000_1d4131cb
<date> <time> <DIR>      Critical_Physical
Disk_1cbd8ffecbc8a1a0e7819e4262e3ece2909a157a_00000000_23551d79
<date> <time> <DIR>      Critical_Physical
Disk_1cbd8ffecbc8a1a0e7819e4262e3ece2909a157a_00000000_2468ad4c
<date> <time> <DIR>      Critical_Physical
Disk_1cbd8ffecbc8a1a0e7819e4262e3ece2909a157a_00000000_255d4d61
<date> <time> <DIR>      Critical_Physical
Disk_1cbd8ffecbc8a1a0e7819e4262e3ece2909a157a_00000000_cab_08289734
<date> <time> <DIR>      Critical_Physical
Disk_64acaf7e4590828ae8a3ac3c8b31da9a789586d4_00000000_cab_1d94712e
<date> <time> <DIR>      Critical_Physical
Disk_ae39f5243a104f21ac5b04a39efec4c126754_00000000_003359cb
<date> <time> <DIR>      Critical_Physical
Disk_ae39f5243a104f21ac5b04a39efec4c126754_00000000_cab_1b293b17
<date> <time> <DIR>      Critical_Physical
Disk_b46b8883d892cfa8a26263afca228b17df8133d_00000000_cab_08abc39c
<date> <time> <DIR>      Kernel_166_1234dacd2d1a219a3696b6e64a736408fc785cc_00000000_cab_19c8a127
      0 File(s)              0 bytes
      20 Dir(s)  23,291,658,240 bytes free
```

Inside the **WER** folder, the **ReportsArchive** folder contains reports that have already been uploaded to Watson. Data in these reports is deleted, but the **Report.wer** file persists.

```
PS C:\Windows\system32> dir C:\ProgramData\Microsoft\Windows\WER\ReportArchive
```

Here's an example of the output:

```
Volume in drive C is INSTALLTO
Volume Serial Number is 4031-E397
```

```
Directory of c:\ProgramData\Microsoft\Windows\WER\ReportArchive
```

```
<date> <time> <DIR>      .
<date> <time> <DIR>      ..
<date> <time> <DIR>
Critical_powershell.exe_7dd54f49935ce48b2dd99d1c64df29a5cfb73db_00000000_cab_096cc802
          0 File(s)              0 bytes
          3 Dir(s)  23,291,658,240 bytes free
```

Windows Error Reporting provides many settings to customize the problem reporting experience. For further information, please refer to the Windows Error Reporting [documentation](#).

Troubleshooting using Windows Error Reporting reports

Physical disk failed to come online

To diagnose this issue, navigate to the WER report folder:

```
PS C:\Windows\system32> dir
C:\ProgramData\Microsoft\Windows\WER\ReportArchive\Critical_PhysicalDisk_b46b8883d892cfa8a26263afca228b17df8133
d_00000000_cab_08abc39c
```

Here's an example of the output:

Volume in drive C is INSTALLTO
Volume Serial Number is 4031-E397

```
<date> <time> <DIR> .
<date> <time> <DIR> ..
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_1.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_10.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_11.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_12.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_13.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_14.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_15.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_16.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_17.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_18.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_19.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_2.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_20.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_21.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_22.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_23.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_24.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_25.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_26.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_27.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_28.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_29.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_3.evtx
<date> <time> 1,118,208 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_30.evtx
<date> <time> 1,118,208 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_31.evtx
<date> <time> 1,118,208 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_32.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_33.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_34.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_35.evtx
<date> <time> 2,166,784 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_36.evtx
<date> <time> 1,118,208 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_37.evtx
<date> <time> 33,194 Report.wer
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_38.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_39.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_4.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_40.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_41.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_5.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_6.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_7.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_8.evtx
<date> <time> 69,632 CLUSWER_RHS_ERROR_8d06c544-47a4-4396-96ec-af644f45c70a_9.evtx
<date> <time> 7,382 WERC263.tmp.WERInternalMetadata.xml
<date> <time> 59,202 WERC36D.tmp.csv
<date> <time> 13,340 WERC38D.tmp.txt
```

Next, start triaging from the **Report.wer** file — this will tell you what failed.

```
EventType=Failover_clustering_resource_error
<skip>
Sig[0].Name=ResourceType
Sig[0].Value=Physical Disk
Sig[1].Name=CallType
Sig[1].Value=ONLINERESOURCE
Sig[2].Name=RHSCallResult
Sig[2].Value=5018
Sig[3].Name=ApplicationCallResult
Sig[3].Value=999
Sig[4].Name=DumpPolicy
Sig[4].Value=5225058577
DynamicSig[1].Name=OS Version
DynamicSig[1].Value=10.0.17051.2.0.0.400.8
DynamicSig[2].Name=Locale ID
DynamicSig[2].Value=1033
DynamicSig[27].Name=ResourceName
DynamicSig[27].Value=Cluster Disk 10
DynamicSig[28].Name=ReportId
DynamicSig[28].Value=8d06c544-47a4-4396-96ec-af644f45c70a
DynamicSig[29].Name=FailureTime
DynamicSig[29].Value=2017//12//12-22:38:05.485
```

Since the resource failed to come online, no dumps were collected, but the Windows Error Reporting report did collect logs. If you open all **.evtx** files using Microsoft Message Analyzer, you will see all of the information that was collected using the following queries through the system channel, application channel, failover cluster diagnostic channels, and a few other generic channels.

```
PS C:\Windows\system32> (Get-ClusterResourceType -Name "Physical Disk").DumpLogQuery
```

Here's an example of the output:

```

<QueryList><Query Id="0"><Select Path="Microsoft-Windows-Kernel-PnP/Configuration">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-ReFS/Operational">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-Ntfs/Operational">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-Ntfs/WHC">*[System[TimeCreated[timediff(@SystemTime)
&lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-Storage-Storport/Operational">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-Storage-Storport/Health">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-Storage-Storport/Admin">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-Storage-ClassPnP/Operational">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-Storage-ClassPnP/Admin">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-PersistentMemory-ScmBus/Certification">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 86400000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-PersistentMemory-ScmBus/Operational">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-PersistentMemory-PmemDisk/Operational">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-PersistentMemory-NvdimM/Operational">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-PersistentMemory-INvdimM/Operational">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-PersistentMemory-VirtualNvdimM/Operational">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-Storage-Disk/Admin">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-Storage-Disk/Operational">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-ScmDisk0101/Operational">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-Partition/Diagnostic">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-Volume/Diagnostic">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-VolumeSnapshot-Driver/Operational">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-FailoverClustering-Clusport/Operational">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-FailoverClustering-ClusBflt/Operational">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-StorageSpaces-Driver/Diagnostic">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-StorageManagement/Operational">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 86400000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-StorageSpaces-Driver/Operational">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-Storage-Tiering/Admin">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-Hyper-V-VmSwitch-Operational">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>
<QueryList><Query Id="0"><Select Path="Microsoft-Windows-Hyper-V-VmSwitch-Diagnostic">*
[System[TimeCreated[timediff(@SystemTime) &lt;= 600000]]]</Select></Query></QueryList>

```

Message Analyzer enables you to capture, display, and analyze protocol messaging traffic. It also lets you trace and assess system events and other messages from Windows components. You can download [Microsoft Message Analyzer from here](#). When you load the logs into Message Analyzer, you will see the following providers and messages from the log channels.

MessageNu	ProcessId	Thread	Level	ActivityId	LevelDisplayNa	Timestamp
Channel (258):					Application	
Channel (2979):					Microsoft-Windows-FailoverClustering/Diagnostic	
Channel (1329):					Microsoft-Windows-FailoverClustering/DiagnosticVerbose	
Channel (370):					Microsoft-Windows-FailoverClustering/Operational	
Channel (20):					Microsoft-Windows-FailoverClustering-CsvFs/Operational	
Channel (108):					Microsoft-Windows-FailoverClustering-NetFt/Operational	
Channel (25):					Microsoft-Windows-Kernel-LiveDump/Analytic	
Channel (6):					Microsoft-Windows-Kernel-PnP/Configuration	
Channel (4):					Microsoft-Windows-Ntfs/Operational	
Channel (8):					Microsoft-Windows-Storage-ClassPnP/Operational	
Channel (4):					Microsoft-Windows-StorageManagement/Operational	
Channel (12):					Microsoft-Windows-Storage-Storport/Operational	
Channel (9):					Microsoft-Windows-VolumeSnapshot-Driver/Operational	
Channel (832):					System	

You can also group by providers to get the following view:

ProviderName (1):
ProviderName (2): BROWSER
ProviderName (3): Desktop Window Manager
ProviderName (3): DFSR
ProviderName (44): ESENT
ProviderName (9): EventLog
ProviderName (11): iScsiPrt
ProviderName (2): LsaSrv
ProviderName (4): Microsoft-Windows-DfsSvc
ProviderName (12): Microsoft-Windows-Dhcp-Client
ProviderName (10): Microsoft-Windows-DHCPv6-Client
ProviderName (2): Microsoft-Windows-Directory-Services-SAM
ProviderName (14): Microsoft-Windows-DNS-Client
ProviderName (1): Microsoft-Windows-Eventlog
ProviderName (2): Microsoft-Windows-EventSystem
ProviderName (4685): Microsoft-Windows-FailoverClustering
ProviderName (20): Microsoft-Windows-FailoverClustering-CsvFs-Diagnostic
ProviderName (108): Microsoft-Windows-FailoverClustering-NetFt
ProviderName (32): Microsoft-Windows-FilterManager
ProviderName (36): Microsoft-Windows-Hyper-V-Netvsc
ProviderName (12): Microsoft-Windows-Kernel-Boot
ProviderName (148): Microsoft-Windows-Kernel-General
ProviderName (24): Microsoft-Windows-Kernel-LiveDump
ProviderName (6): Microsoft-Windows-Kernel-PnP
ProviderName (4): Microsoft-Windows-Kernel-Power
ProviderName (4): Microsoft-Windows-Kernel-Processor-Power
ProviderName (1): Microsoft-Windows-LoadPerf
ProviderName (4): Microsoft-Windows-MSDTC
ProviderName (2): Microsoft-Windows-MSDTC 2
ProviderName (4): Microsoft-Windows-MSMQ
ProviderName (29): Microsoft-Windows-Ntfs
ProviderName (3): Microsoft-Windows-Perflib
ProviderName (40): Microsoft-Windows-Security-SPP
ProviderName (4): Microsoft-Windows-StorageManagement-WSP-Spaces
ProviderName (8): Microsoft-Windows-StorDiag
ProviderName (12): Microsoft-Windows-StorPort
ProviderName (12): Microsoft-Windows-Time-Service
ProviderName (7): Microsoft-Windows-User Profiles Service
ProviderName (9): Microsoft-Windows-VolumeSnapshot-Driver
ProviderName (9): Microsoft-Windows-WindowsUpdateClient
ProviderName (2): Microsoft-Windows-Wininit
ProviderName (4): Microsoft-Windows-Winlogon
ProviderName (3): Microsoft-Windows-WinRM
ProviderName (4): Microsoft-Windows-WMI
ProviderName (2): MSiSCSI
ProviderName (2): MSiSNS
ProviderName (1): NETLOGON
ProviderName (1): SceCli
ProviderName (441): Service Control Manager
ProviderName (104): SideBySide
ProviderName (2): User32
ProviderName (8): VSS
ProviderName (2): Windows Error Reporting
ProviderName (19): Wins
ProviderName (26): WTT

To identify why the disk failed, navigate to the events under **FailoverClustering/Diagnostic** and **FailoverClustering/DiagnosticVerbose**. Then run the following query: **EventLog.EventData["LogString"] contains "Cluster Disk 10"**. This will give you the following output:

MessageId	ProcessId	Thread	Level	ActivityId	LevelDisplay	Name	Timestamp	Message	Provider	EventId	Version	Op	Sub	EventData["LogString"]
746	7576	7972	4	132dee88...	Information		2017-12-12T22:38:06.0386228	E.. Mic.. Micro.. 2049 0 In.. U.. m.. [RES] Physical Disk <Cluster Disk 10>: ResHardDiskOnlineV2: Online request.	E.. Mic.. Micro.. 2049 0 In.. U.. m.. [RES] Physical Disk <Cluster Disk 10>: ResHardDiskOnlineV2: Online request.	2049	0	In..	U.. m..	[RES] Physical Disk <Cluster Disk 10>: ResHardDiskOnlineV2: Online request.
747	3860	104	5	nothing	Verbose		2017-12-12T22:38:06.0799345	E.. Mic.. Micro.. 2052 0 In.. U.. m.. [RCH] rcn::RcmResource::Control: (Cluster Disk 10, GET_ID)	E.. Mic.. Micro.. 2052 0 In.. U.. m.. [RCH] rcn::RcmResource::Control: (Cluster Disk 10, GET_ID)	2052	0	In..	U.. m..	[RCH] rcn::RcmResource::Control: (Cluster Disk 10, GET_ID)
748	7576	2740	5	132dee88...	Verbose		2017-12-12T22:38:06.0806403	E.. Mic.. Micro.. 2052 0 In.. U.. m.. [RHS] Resource Cluster Disk 10 called SetResourceStatusEx: checkpoint 1. Old state OnlinePending, new state Onli..	E.. Mic.. Micro.. 2052 0 In.. U.. m.. [RHS] Resource Cluster Disk 10 called SetResourceStatusEx: checkpoint 1. Old state OnlinePending, new state Onli..	2052	0	In..	U.. m..	[RHS] Resource Cluster Disk 10 called SetResourceStatusEx: checkpoint 1. Old state OnlinePending, new state Onli..
749	7576	2740	2	132dee88...	Error		2017-12-12T22:38:06.0807718	E.. Mic.. Micro.. 2051 0 In.. U.. m.. [RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating failure. g.FailureCode: 999.	E.. Mic.. Micro.. 2051 0 In.. U.. m.. [RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating failure. g.FailureCode: 999.	2051	0	In..	U.. m..	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating failure. g.FailureCode: 999.
750	7576	2740	2	132dee88...	Error		2017-12-12T22:38:06.0810469	E.. Mic.. Micro.. 2051 0 In.. U.. m.. [RES] Physical Disk <Cluster Disk 10>: OnlineThread: Error 999 bringing resource online.	E.. Mic.. Micro.. 2051 0 In.. U.. m.. [RES] Physical Disk <Cluster Disk 10>: OnlineThread: Error 999 bringing resource online.	2051	0	In..	U.. m..	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Error 999 bringing resource online.
751	7576	2740	5	132dee88...	Verbose		2017-12-12T22:38:06.0811786	E.. Mic.. Micro.. 2052 0 In.. U.. m.. [RHS] Resource Cluster Disk 10 called SetResourceStatusEx: checkpoint 1. Old state OnlinePending, new state Fail..	E.. Mic.. Micro.. 2052 0 In.. U.. m.. [RHS] Resource Cluster Disk 10 called SetResourceStatusEx: checkpoint 1. Old state OnlinePending, new state Fail..	2052	0	In..	U.. m..	[RHS] Resource Cluster Disk 10 called SetResourceStatusEx: checkpoint 1. Old state OnlinePending, new state Fail..
752	7576	2740	5	132dee88...	Verbose		2017-12-12T22:38:06.0812237	E.. Mic.. Micro.. 2052 0 In.. U.. m.. [RHS] OnlineResCall::OnResourceStatusChanged(Cluster Disk 10)	E.. Mic.. Micro.. 2052 0 In.. U.. m.. [RHS] OnlineResCall::OnResourceStatusChanged(Cluster Disk 10)	2052	0	In..	U.. m..	[RHS] OnlineResCall::OnResourceStatusChanged(Cluster Disk 10)

Physical disk timed out

To diagnose this issue, navigate to the WER report folder. The folder contains log files and dump files for **RHS**, **clussvc.exe**, and of the process that hosts the “**smphost**” service, as shown below:

```
PS C:\Windows\system32> dir
C:\ProgramData\Microsoft\Windows\WER\ReportArchive\Critical_PhysicalDisk_64acaf7e4590828ae8a3ac3c8b31da9a789586
d4_00000000_cab_1d94712e
```

Here's an example of the output:

```
Volume in drive C is INSTALLTO
Volume Serial Number is 4031-E397

<date> <time> <DIR> .
<date> <time> <DIR> ..
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_1.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_10.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_11.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_12.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_13.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_14.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_15.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_16.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_17.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_18.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_19.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_2.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_20.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_21.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_22.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_23.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_24.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_25.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_26.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_27.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_28.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_29.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_3.evtx
<date> <time> 1,118,208 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_30.evtx
<date> <time> 1,118,208 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_31.evtx
<date> <time> 1,118,208 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_32.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_33.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_34.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_35.evtx
<date> <time> 2,166,784 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_36.evtx
<date> <time> 1,118,208 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_37.evtx
<date> <time> 28,340,500 memory.hdump
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_38.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_39.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_4.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_40.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_41.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_5.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_6.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_7.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_8.evtx
<date> <time> 69,632 CLUSWER_RHS_HANG_75e60318-50c9-41e4-94d9-fb0f589cd224_9.evtx
<date> <time> 4,466,943 minidump.0f14.mdmp
<date> <time> 1,735,776 minidump.2200.mdmp
<date> <time> 33,890 Report.wer
<date> <time> 49,267 WER69FA.tmp.mdmp
<date> <time> 5,706 WER70A2.tmp.WERInternalMetadata.xml
<date> <time> 63,206 WER70E0.tmp.csv
<date> <time> 13,340 WER7100.tmp.txt
```

Next, start triaging from the **Report.wer** file — this will tell you what call or resource is hanging.

```
EventType=Failover_clustering_resource_timeout_2
<skip>
Sig[0].Name=ResourceType
Sig[0].Value=Physical Disk
Sig[1].Name=CallType
Sig[1].Value=ONLINERESOURCE
Sig[2].Name=DumpPolicy
Sig[2].Value=5225058577
Sig[3].Name=ControlCode
Sig[3].Value=18
DynamicSig[1].Name=OS Version
DynamicSig[1].Value=10.0.17051.2.0.0.400.8
DynamicSig[2].Name=Locale ID
DynamicSig[2].Value=1033
DynamicSig[26].Name=ResourceName
DynamicSig[26].Value=Cluster Disk 10
DynamicSig[27].Name=ReportId
DynamicSig[27].Value=75e60318-50c9-41e4-94d9-fb0f589cd224
DynamicSig[29].Name=HangThreadId
DynamicSig[29].Value=10008
```

The list of services and processes that we collect in a dump is controlled by the following property: **PS C:\Windows\system32> (Get-ClusterResourceType -Name "Physical Disk").DumpServicesSmphost**

To identify why the hang happened, open the dum files. Then run the following query: **EventLog.EventData["LogString"] contains "Cluster Disk 10"** This will give you give you the following output:

MessageNo	ProcessId	Thread	Level	ActivityId	LevelDisplayNa	Timestamp	Mi	Chan	Provider	Event	Ver	Op	Su	Ev	EventData["LogString"]
1820	3860	5584	4	nothing	Information	2017-12-12T22:41:57.7579775	E.	Mic.	Micro...	2049	0	In.	U.	m.	[RCM] Res Cluster Disk 10: OnlineCallIssued -> OnlinePending(StateUnknown)
177	3860	5584	4	nothing	Information	2017-12-12T22:41:57.7580195	E.	Mic.	Micro...	5399	0	In.	U.	m.	[RCM] TransitionToState(Cluster Disk 10) OnlineCallIssued->OnlinePending.
1821	3860	5584	4	nothing	Information	2017-12-12T22:41:57.7580206	E.	Mic.	Micro...	2049	0	In.	U.	m.	[RCM] TransitionToState(Cluster Disk 10) OnlineCallIssued->OnlinePending.
182	3860	4476	5	nothing	Verbose	2017-12-12T22:41:57.7746874	E.	Mic.	Micro...	5408	0	In.	U.	m.	[RCM] rcm:RcmResource:Control: (Cluster Disk 10, GET_ID)
1826	3860	4476	5	nothing	Verbose	2017-12-12T22:41:57.7746887	E.	Mic.	Micro...	2052	0	In.	U.	m.	[RCM] rcm:RcmResource:Control: (Cluster Disk 10, GET_ID)
239	7576	10008	2	75e60318..	Error	2017-12-12T22:42:06.0653000	E.	Mic.	Micro...	5401	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 1.
1883	7576	10008	2	75e60318..	Error	2017-12-12T22:42:06.0653000	E.	Mic.	Micro...	2051	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 1.
297	7576	10008	2	75e60318..	Error	2017-12-12T22:42:13.0160722	E.	Mic.	Micro...	5401	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 2.
1941	7576	10008	2	75e60318..	Error	2017-12-12T22:42:13.0160749	E.	Mic.	Micro...	2051	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 2.
335	7576	10008	2	75e60318..	Error	2017-12-12T22:42:23.0169812	E.	Mic.	Micro...	5401	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 3.
1979	7576	10008	2	75e60318..	Error	2017-12-12T22:42:23.0169839	E.	Mic.	Micro...	2051	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 3.
416	7576	10008	2	75e60318..	Error	2017-12-12T22:42:33.1598682	E.	Mic.	Micro...	5401	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 4.
2060	7576	10008	2	75e60318..	Error	2017-12-12T22:42:33.1598709	E.	Mic.	Micro...	2051	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 4.
466	7576	10008	2	75e60318..	Error	2017-12-12T22:42:43.4320410	E.	Mic.	Micro...	5401	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 5.
2110	7576	10008	2	75e60318..	Error	2017-12-12T22:42:43.4320437	E.	Mic.	Micro...	2051	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 5.
523	7576	10008	2	75e60318..	Error	2017-12-12T22:42:53.4365993	E.	Mic.	Micro...	5401	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 6.
2167	7576	10008	2	75e60318..	Error	2017-12-12T22:42:53.4366018	E.	Mic.	Micro...	2051	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 6.
635	7576	10008	2	75e60318..	Error	2017-12-12T22:43:03.4352304	E.	Mic.	Micro...	5401	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 7.
2285	7576	10008	2	75e60318..	Error	2017-12-12T22:43:03.4352328	E.	Mic.	Micro...	2051	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 7.
685	7576	10008	2	75e60318..	Error	2017-12-12T22:43:13.4357504	E.	Mic.	Micro...	5401	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 8.
2335	7576	10008	2	75e60318..	Error	2017-12-12T22:43:13.4357531	E.	Mic.	Micro...	2051	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 8.
729	7576	10008	2	75e60318..	Error	2017-12-12T22:43:23.4437011	E.	Mic.	Micro...	5401	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 9.
2379	7576	10008	2	75e60318..	Error	2017-12-12T22:43:23.4437037	E.	Mic.	Micro...	2051	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 9.
785	7576	10008	2	75e60318..	Error	2017-12-12T22:43:33.4421265	E.	Mic.	Micro...	5401	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 10.
2435	7576	10008	2	75e60318..	Error	2017-12-12T22:43:33.4421290	E.	Mic.	Micro...	2051	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 10.
824	7576	10008	2	75e60318..	Error	2017-12-12T22:43:43.4412089	E.	Mic.	Micro...	5401	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 11.
2474	7576	10008	2	75e60318..	Error	2017-12-12T22:43:43.4412837	E.	Mic.	Micro...	2051	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 11.
881	7576	10008	2	75e60318..	Error	2017-12-12T22:43:53.4418088	E.	Mic.	Micro...	5401	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 12.
2531	7576	10008	2	75e60318..	Error	2017-12-12T22:43:53.4418911	E.	Mic.	Micro...	2051	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 12.
928	7576	10008	2	75e60318..	Error	2017-12-12T22:44:03.4439773	E.	Mic.	Micro...	5401	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 13.
2578	7576	10008	2	75e60318..	Error	2017-12-12T22:44:03.4439798	E.	Mic.	Micro...	2051	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 13.
972	7576	10008	2	75e60318..	Error	2017-12-12T22:44:13.4446826	E.	Mic.	Micro...	5401	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 14.
2622	7576	10008	2	75e60318..	Error	2017-12-12T22:44:13.4446852	E.	Mic.	Micro...	2051	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 14.
1022	7576	10008	2	75e60318..	Error	2017-12-12T22:44:23.4461055	E.	Mic.	Micro...	5401	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 15.
2672	7576	10008	2	75e60318..	Error	2017-12-12T22:44:23.4461079	E.	Mic.	Micro...	2051	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 15.
1083	7576	10008	2	75e60318..	Error	2017-12-12T22:44:33.4469696	E.	Mic.	Micro...	5401	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 16.
2733	7576	10008	2	75e60318..	Error	2017-12-12T22:44:33.4469695	E.	Mic.	Micro...	2051	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 16.
1122	7576	10008	2	75e60318..	Error	2017-12-12T22:44:43.4474174	E.	Mic.	Micro...	5401	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 17.
2772	7576	10008	2	75e60318..	Error	2017-12-12T22:44:43.4474201	E.	Mic.	Micro...	2051	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 17.
1174	7576	10008	2	75e60318..	Error	2017-12-12T22:44:53.4478159	E.	Mic.	Micro...	5401	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 18.
2824	7576	10008	2	75e60318..	Error	2017-12-12T22:44:53.4478178	E.	Mic.	Micro...	2051	0	In.	U.	m.	[RES] Physical Disk <Cluster Disk 10>: OnlineThread: Simulating hang. Sleeping for 10 seconds. Iteration 18.
1176	7576	7644	3	5b123e07..	Warning	2017-12-12T22:44:54.1274438	E.	Mic.	Micro...	5400	0	In.	U.	m.	[RMS - Timeout] Resource 'Cluster Disk 10' has not responded to the call ONLINERESOURCE:18. The timeout to respo.
2826	7576	7644	3	5b123e07..	Warning	2017-12-12T22:44:54.1274454	E.	Mic.	Micro...	2050	0	In.	U.	m.	[RMS - Timeout] Resource 'Cluster Disk 10' has not responded to the call ONLINERESOURCE:18. The timeout to respo.
1186	7576	7644	3	5b123e07..	Warning	2017-12-12T22:46:14.3576085	E.	Mic.	Micro...	5400	0	In.	U.	m.	[RMS-Timeout] Health Monitoring Failure: Resource Cluster Disk 10 is not functioning as expected. Cancelling cur.
2836	7576	7644	3	5b123e07..	Warning	2017-12-12T22:46:14.3576103	E.	Mic.	Micro...	2050	0	In.	U.	m.	[RMS-Timeout] Health Monitoring Failure: Resource Cluster Disk 10 is not functioning as expected. Cancelling cur.
1187	7576	7644	5	5b123e07..	Verbose	2017-12-12T22:46:14.3576415	E.	Mic.	Micro...	5402	0	In.	U.	m.	[RMS] SetOnlineOrOfflineCall: Stopping pending state timer for Resource Cluster Disk 10 CurrentState: OnlinePend.
2837	7576	7644	5	5b123e07..	Verbose	2017-12-12T22:46:14.3576427	E.	Mic.	Micro...	2052	0	In.	U.	m.	[RMS] SetOnlineOrOfflineCall: Stopping pending state timer for Resource Cluster Disk 10 CurrentState: OnlinePend.
1188	7576	7644	5	5b123e07..	Verbose	2017-12-12T22:46:14.3576913	E.	Mic.	Micro...	5402	0	In.	U.	m.	[RMS] OnlineResCall: OnDeadlock(Cluster Disk 10)
2838	7576	7644	5	5b123e07..	Verbose	2017-12-12T22:46:14.3576924	E.	Mic.	Micro...	2052	0	In.	U.	m.	[RMS] OnlineResCall: OnDeadlock(Cluster Disk 10)
1190	3860	6224	3	nothing	Warning	2017-12-12T22:46:14.3591303	E.	Mic.	Micro...	5400	0	In.	U.	m.	[RCM] HandleMonitorReply: FAILURENOTIFICATION for 'Cluster Disk 10', gen(2) result 5018/0.
2840	3860	6224	3	nothing	Warning	2017-12-12T22:46:14.3591321	E.	Mic.	Micro...	2050	0	In.	U.	m.	[RCM] HandleMonitorReply: FAILURENOTIFICATION for 'Cluster Disk 10', gen(2) result 5018/0.

We can cross-examine this with the thread from the **memory.hdmp** file:

```
# 21 Id: 1d98.2718 Suspend: 0 Teb: 0000000b`f1f7b000 Unfrozen
# Child-SP          RetAddr          Call Site
00 0000000b`f3c7ec38 00007ff8`455d25ca ntdll!ZwDelayExecution+0x14
01 0000000b`f3c7ec40 00007ff8`2ef19710 KERNELBASE!SleepEx+0x9a
02 0000000b`f3c7ece0 00007ff8`3bdf7fbf clusres!ResHardDiskOnlineOrTurnOffMMThread+0x2b0
03 0000000b`f3c7f960 00007ff8`391eed34 resutils!ClusWorkerStart+0x5f
04 0000000b`f3c7f9d0 00000000`00000000 vfbasics+0xed34
```

Cluster sets

7/3/2018 • 23 minutes to read • [Edit Online](#)

Applies To: Windows Server Insider Preview build 17650 and later

Cluster sets is the new cloud scale-out technology in this preview release that increases cluster node count in a single Software Defined Data Center (SDDC) cloud by orders of magnitude. A cluster set is a loosely-coupled grouping of multiple Failover Clusters: compute, storage or hyper-converged. Cluster sets technology enables virtual machine fluidity across member clusters within a cluster set and a unified storage namespace across the set in support of virtual machine fluidity.

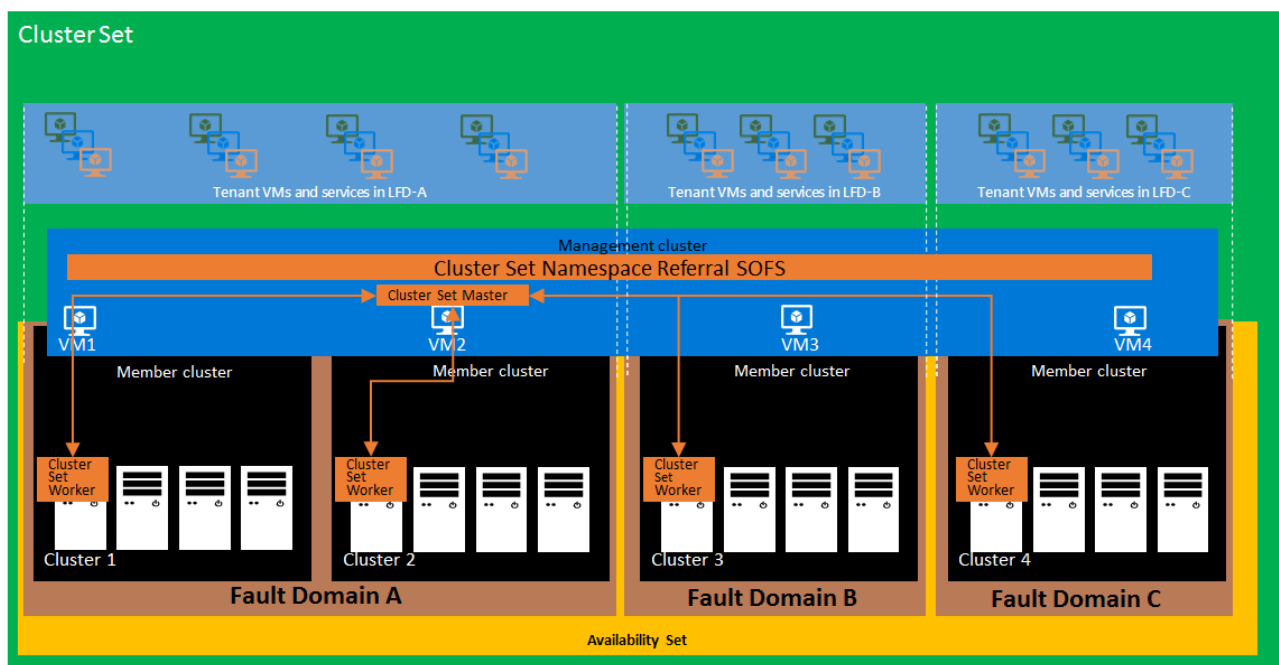
While preserving existing Failover Cluster management experiences on member clusters, a cluster set instance additionally offers key use cases around lifecycle management at the aggregate. This Windows Server Preview Scenario Evaluation Guide provides you the necessary background information along with step-by-step instructions to evaluate cluster sets technology using PowerShell.

Technology introduction

Cluster sets technology is developed to meet specific customer requests operating Software Defined Datacenter (SDDC) clouds at scale. Cluster sets value proposition in this Preview release may be summarized as the following:

- Significantly increase the supported SDDC cloud scale for running highly available virtual machines by combining multiple smaller clusters into a single large fabric, even while keeping the software fault boundary to a single cluster
- Manage entire Failover Cluster lifecycle including onboarding and retiring clusters, without impacting tenant virtual machine availability, via fluidly migrating virtual machines across this large fabric
- Easily change the compute-to-storage ratio in your hyper-converged I
- Benefit from [Azure-like Fault Domains and Availability sets](#) across clusters in initial virtual machine placement and subsequent virtual machine migration
- Mix-and-match different generations of CPU hardware into the same cluster set fabric, even while keeping individual fault domains homogenous for maximum efficiency. Please note that the recommendation of same hardware is still present within each individual cluster as well as the entire cluster set.

From a high level view, this is what cluster sets can look like.



The following provides a quick summary of each of the elements in the above image:

Management cluster

Management cluster in a cluster set is a Failover Cluster that hosts the highly-available management plane of the entire cluster set and the unified storage namespace (Cluster Set Namespace) referral Scale-Out File Server (SOFS). A management cluster is logically decoupled from member clusters that run the virtual machine workloads. This makes the cluster set management plane resilient to any localized cluster-wide failures, e.g. loss of power of a member cluster.

Member cluster

A member cluster in a cluster set is typically a traditional hyper-converged cluster running virtual machine and Storage Spaces Direct workloads. Multiple member clusters participate in a single cluster set deployment, forming the larger SDDC cloud fabric. Member clusters differ from a management cluster in two key aspects: member clusters participate in fault domain and availability set constructs, and member clusters are also sized to host virtual machine and Storage Spaces Direct workloads. Cluster set virtual machines that move across cluster boundaries in a cluster set must not be hosted on the management cluster for this reason.

Cluster set namespace referral SOFS

A cluster set namespace referral (Cluster Set Namespace) SOFS is a Scale-Out File Server wherein each SMB Share on the Cluster Set Namespace SOFS is a referral share – of type 'SimpleReferral' newly introduced in this Preview release. This referral allows Server Message Block (SMB) clients access to the target SMB share hosted on the member cluster SOFS. The cluster set namespace referral SOFS is a light-weight referral mechanism and as such, does not participate in the I/O path. The SMB referrals are cached perpetually on the each of the client nodes and the cluster sets namespace dynamically updates automatically these referrals as needed.

Cluster set master

In a cluster set, the communication between the member clusters is loosely coupled, and is coordinated by a new cluster resource called "Cluster Set Master" (CS-Master). Like any other cluster resource, CS-Master is highly available and resilient to individual member cluster failures and/or the management cluster node failures. Through a new Cluster Set WMI provider, CS-Master provides the management endpoint for all Cluster Set manageability interactions.

Cluster set worker

In a Cluster Set deployment, the CS-Master interacts with a new cluster resource on the member Clusters called

“Cluster Set Worker” (CS-Worker). CS-Worker acts as the only liaison on the cluster to orchestrate the local cluster interactions as requested by the CS-Master. Examples of such interactions include virtual machine placement and cluster-local resource inventorying. There is only one CS-Worker instance for each of the member clusters in a cluster set.

Fault domain

A fault domain is the grouping of software and hardware artifacts that the administrator determines could fail together when a failure does occur. While an administrator could designate one or more clusters together as a fault domain, each node could participate in a fault domain in an availability set. Cluster sets by design leaves the decision of fault domain boundary determination to the administrator who is well-versed with data center topology considerations – e.g. PDU, networking – that member clusters share.

Availability set

An availability set helps the administrator configure desired redundancy of clustered workloads across fault domains, by organizing those into an availability set and deploying workloads into that availability set. Let's say if you are deploying a two-tier application, we recommend that you configure at least two virtual machines in an availability set for each tier which will ensure that when one fault domain in that availability set goes down, your application will at least have one virtual machine in each tier hosted on a different fault domain of that same availability set.

Why use cluster sets

Cluster sets provides the benefit of scale without sacrificing resiliency.

Cluster sets allows for clustering multiple clusters together to create a large fabric, while each cluster remains independent for resiliency. For example, you have a several 4-node HCI clusters running virtual machines. Each cluster provides the resiliency needed for itself. If the storage or memory starts to fill up, scaling up is your next step. With scaling up, there are some options and considerations.

1. Add more storage to the current cluster. With Storage Spaces Direct, this may be tricky as the exact same model/firmware drives may not be available. The consideration of rebuild times also need to be taken into account.
2. Add more memory. What if you are maxed out on the memory the machines can handle? What if all available memory slots are full?
3. Add additional compute nodes with drives into the current cluster. This takes us back to Option 1 needing to be considered.
4. Purchase a whole new cluster

This is where cluster sets provides the benefit of scaling. If I add my clusters into a cluster set, I can take advantage of storage or memory that may be available on another cluster without any additional purchases. From a resiliency perspective, adding additional nodes to a Storage Spaces Direct is not going to provide additional votes for quorum. As mentioned [here](#), a Storage Spaces Direct Cluster can survive the loss of 2 nodes before going down. If you have a 4-node HCI cluster, 3 nodes go down will take the entire cluster down. If you have an 8-node cluster, 3 nodes go down will take the entire cluster down. With Cluster sets that has two 4-node HCI clusters in the set, 2 nodes in one HCI go down and 1 node in the other HCI go down, both clusters remain up. Is it better to create one large 16-node Storage Spaces Direct cluster or break it down into four 4-node clusters and use cluster sets? Having four 4-node clusters with cluster sets gives the same scale, but better resiliency in that multiple compute nodes can go down (unexpectedly or for maintenance) and production remains.

Considerations for deploying cluster sets

When considering if cluster sets is something you need to use, consider these questions:

- Do you need to go beyond the current HCI compute and storage scale limits?
- Are all compute and storage not identically the same?
- Do you live migrate virtual machines between clusters?
- Would you like Azure-like computer availability sets and fault domains across multiple clusters?
- Do you need to take the time to look at all your clusters to determine where any new virtual machines need to be placed?

If your answer is yes, then cluster sets is what you need.

There are a few other items to consider where a larger SDDC might change your overall data center strategies. SQL Server is a good example. Does moving SQL Server virtual machines between clusters require licensing SQL to run on additional nodes?

Scale-out file server and cluster sets

In Windows Server 2019, there is a new scale-out file server role called Infrastructure Scale-Out File Server (SOFS).

The following considerations apply to an Infrastructure SOFS role:

1. There can be at most only one Infrastructure SOFS cluster role on a Failover Cluster. Infrastructure SOFS role is created by specifying the **"-Infrastructure"** switch parameter to the **Add-ClusterScaleOutFileServerRole** cmdlet. For example:

```
Add-ClusterScaleoutFileServerRole -Name "my_infra_sofs_name" -Infrastructure
```

2. Each CSV volume created in the failover automatically triggers the creation of an SMB Share with an auto-generated name based on the CSV volume name. An administrator cannot directly create or modify SMB shares under an SOFS role, other than via CSV volume create/modify operations.
3. In hyper-converged configurations, an Infrastructure SOFS allows an SMB client (Hyper-V host) to communicate with guaranteed Continuous Availability (CA) to the Infrastructure SOFS SMB server. This hyper-converged SMB loopback CA is achieved via virtual machines accessing their virtual disk (VHDx) files where the owning virtual machine identity is forwarded between the client and server. This identity forwarding allows ACL-ing VHDx files just as in standard hyper-converged cluster configurations as before.

Once a cluster set is created, the cluster set namespace relies on an Infrastructure SOFS on each of the member clusters, and additionally an Infrastructure SOFS in the management cluster.

At the time a member cluster is added to a cluster set, the administrator specifies the name of an Infrastructure SOFS on that cluster if one already exists. If the Infrastructure SOFS does not exist, a new Infrastructure SOFS role on the new member cluster is created by this operation. If an Infrastructure SOFS role already exists on the member cluster, the Add operation implicitly renames it to the specified name as needed. Any existing singleton SMB servers, or non-Infrastructure SOFS roles on the member clusters are left unutilized by the cluster set.

At the time the cluster set is created, the administrator has the option to use an already-existing AD computer object as the namespace root on the management cluster. Cluster set creation operations create the Infrastructure SOFS cluster role on the management cluster or renames the existing Infrastructure SOFS role just as previously described for member clusters. The Infrastructure SOFS on the management cluster is used as the cluster set namespace referral (Cluster Set Namespace) SOFS. It simply means that each SMB Share on the cluster set namespace SOFS is a referral share – of type 'SimpleReferral' - newly introduced in this Preview release. This referral allows SMB clients access to the target SMB share hosted on the member cluster SOFS. The cluster set namespace referral SOFS is a light-weight referral mechanism and as such, does not participate in the I/O path. The SMB referrals are cached perpetually on each of the client nodes and the cluster sets namespace dynamically updates automatically these referrals as needed.

Creating a Cluster Set

Prerequisites

When creating a cluster set, you following prerequisites are recommended:

1. Configure a management client running the latest Windows Server Insider release.
2. Install the Failover Cluster tools on this management server.
3. Create cluster members (at least two clusters with at least two Cluster Shared Volumes on each cluster)
4. Create a management cluster (physical or guest) that straddles the member clusters. This approach ensures that the Cluster setsagement plane continues to be available despite possible member cluster failures.

Steps

1. Create a new cluster set from three clusters as defined in the prerequisites. The below chart gives an example of clusters to create. The name of the cluster set in this example will be **CSMASTER**.

Cluster Name	Infrastructure SOFS Name to be used later
SET-CLUSTER	SOFS-CLUSTERSET
CLUSTER1	SOFS-CLUSTER1
CLUSTER2	SOFS-CLUSTER2

2. Once all cluster have been created, use the following commands to create the cluster set master.

```
New-ClusterSet -Name CSMaster -NamespaceRoot SOFS-CLUSTERSET -CimSession SET-CLUSTER
```

3. To add a Cluster Server to the cluster set, the below would be used.

```
Add-ClusterSetMember -ClusterName CLUSTER1 -CimSession CSMaster -InfraSOFSName SOFS-CLUSTER1
Add-ClusterSetMember -ClusterName CLUSTER2 -CimSession CSMaster -InfraSOFSName SOFS-CLUSTER2
```

NOTE: If you are using a static IP Address scheme, you must include *-StaticAddress x.x.x.x* on the **New-ClusterSet** command.

1. Once you have created the cluster set out of cluster members, you can list the nodes set and its properties. To enumerate all the member clusters in the cluster set:

```
Get-ClusterSetMember -CimSession CSMaster
```

2. To enumerate all the member clusters in the cluster set including the management cluster nodes:

```
Get-ClusterSet -CimSession CSMaster | Get-Cluster | Get-ClusterNode
```

3. To list all the nodes from the member clusters:

```
Get-ClusterSetNode -CimSession CSMaster
```

4. To list all the resource groups across the cluster set:

```
Get-ClusterSet -CimSession CSMaster | Get-Cluster | Get-ClusterGroup
```

5. To verify the cluster set creation process created one SMB share (identified as Volume1 or whatever the CSV

folder is labeled with the ScopeName being the name of the Infrastructure File Server and the path as both) on the Infrastructure SOFS for each cluster member's CSV volume:

```
Get-SmbShare -CimSession CSMaster
```

6. Cluster sets has debug logs that can be collected for review. Both the cluster set and cluster debug logs can be gathered for all members and the management cluster.

```
Get-ClusterSetLog -ClusterSetCimSession CSMaster -IncludeClusterLog -IncludeManagementClusterLog -  
DestinationFolderPath <path>
```

7. Configure kerberos [constrained delegation](#) between all cluster set members.
8. Configure the cross-cluster virtual machine live migration authentication type to Kerberos on each node in the Cluster Set.

```
foreach($h in $hosts){ Set-VMHost -VirtualMachineMigrationAuthenticationType Kerberos -ComputerName $h }
```

9. Add the management cluster to the local administrators group on each node in the cluster set.

```
foreach($h in $hosts){ Invoke-Command -ComputerName $h -ScriptBlock {Net localgroup administrators /add  
<management_cluster_name>}} }
```

Creating new virtual machines and adding to cluster sets

After creating the cluster set, the next step is to create new virtual machines. Normally, when it is time to create virtual machines and add them to a cluster, you need to do some checks on the clusters to see which it may be best to run on. These checks could include:

- How much memory is available on the cluster nodes?
- How much disk space is available on the cluster nodes?
- Does the virtual machine require specific storage requirements (i.e. I want my SQL Server virtual machines to go to a cluster running faster drives; or, my infrastructure virtual machine is not as critical and can run on slower drives).

Once these questions are answered, you create the virtual machine on the cluster you need it to be. One of the benefits of cluster sets is that cluster sets do those checks for you and place the virtual machine on the most optimal node.

The below commands will both identify the optimal cluster and deploy the virtual machine to it. In the below example, a new virtual machine is created specifying that at least 4 gigabytes of memory is available for the virtual machine and that it will need to utilize 1 virtual processor.

- ensure that 4gb is available for the virtual machine
- set the virtual processor used at 1
- check to ensure there is at least 10% CPU available for the virtual machine

```
# Identify the optimal node to create a new virtual machine
$memoryinMB=4096
$vpcount = 1
$targetnode = Get-ClusterSetOptimalNodeForVM -CimSession CSMaster -VMMemory $memoryinMB -
VMVirtualCoreCount $vpcount -VMCpuReservation 10
$secure_string_pwd = convertto-securestring "<password>" -asplaintext -force
$cred = new-object -typename System.Management.Automation.PSCredential ("
<domain\account>", $secure_string_pwd)

# Deploy the virtual machine on the optimal node
Invoke-Command -ComputerName $targetnode.name -scriptblock { param([String]$storagepath); New-VM CSV1
-MemoryStartupBytes 3072MB -path $storagepath -NewVHDPATH CSV1.vhdx -NewVHDSizeBytes 4194304 } -
ArgumentList @("&SOFs-CLUSTER1\VOLUME1") -Credential $cred | Out-Null
Start-VM CSV1 -ComputerName $targetnode.name | Out-Null
Get-VM CSV1 -ComputerName $targetnode.name | fl State, ComputerName
```

When it completes, you will be given the information about the virtual machine and where it was placed. In the above example, it would show as:

```
State          : Running
ComputerName    : 1-S2D2
```

If you were to not have enough memory, cpu, or disk space to add the virtual machine, you will receive the error:

```
Get-ClusterSetOptimalNodeForVM : A cluster node is not available for this operation.
```

Once the virtual machine has been created, it will be displayed in Hyper-V manager on the specific node specified. To add it as a cluster set virtual machine and into the cluster, the command is below.

```
Register-ClusterSetVM -CimSession CSMaster -MemberName $targetnode.Member -VMName CSV1
```

When it completes, the output will be:

Id	VMName	State	MemberName	PSComputerName
1	CSV1	On	CLUSTER1	CSMASTER

If you have added a cluster with existing virtual machines, the virtual machines will also need to be registered with Cluster set so register all the virtual machines at once, the command to use is:

```
Get-ClusterSetMember -name CLUSTER3 -CimSession CSMaster | Register-ClusterSetVM -RegisterAll -CimSession CSMaster
```

However, the process is not complete as the path to the virtual machine needs to be added to the cluster set namespace.

So for example, an existing cluster is added and it has pre-configured virtual machines that reside on the local Cluster Shared Volume (CSV), the path for the VHDX would be something similar to "C:\ClusterStorage\Volume1\MYVM\Virtual Hard Disks\MYVM.vhdx. A storage migration would need to be accomplished as CSV paths are by design local to a single member cluster. Thus, will not be accessible to the virtual machine once they are live migrated across member clusters.

In this example, CLUSTER3 was added to the cluster set using Add-ClusterSetMember with the Infrastructure Scale-Out File Server as SOFS-CLUSTER3. To move the virtual machine configuration and storage, the command

is:

```
Move-VMStorage -DestinationStoragePath \\SOFS-CLUSTER3\Volume1 -Name MYVM
```

Once it completes, you will receive a warning:

```
WARNING: There were issues updating the virtual machine configuration that may prevent the virtual machine
from running. For more information view the report file below.
WARNING: Report file location: C:\Windows\Cluster\Reports\Update-ClusterVirtualMachineConfiguration '' on
date at time.htm.
```

This warning can be ignored as the warning is "No changes in the virtual machine role storage configuration were detected". The reason for the warning as the actual physical location does not change; only the configuration paths.

For more information on Move-VMStorage, please review this [link](#).

Live migrating a virtual machine between different cluster set clusters is not the same as in the past. In non-cluster set scenarios, the steps would be:

1. remove the virtual machine role from the Cluster.
2. live migrate the virtual machine to a member node of a different cluster.
3. add the virtual machine into the cluster as a new virtual machine role.

With Cluster sets these steps are not necessary and only one command is needed. For example, I want to move a Cluster Set virtual machine from CLUSTER1 to CLUSTER3. The single command would be:

```
Move-ClusterSetVM -CimSession CSMaster -VMName CSV1 -ClusterName CLUSTER3
```

Please note that this does not move the virtual machine storage or configuration files. This is not necessary as the path to the virtual machine remains as \SOFS-CLUSTER1\VOLUME1. Once a virtual machine has been registered with cluster sets has the Infrastructure File Server share path, the drives and virtual machine do not require being on the same machine as the virtual machine.

Creating Availability sets Fault Domains

As described in the introduction, Azure-like fault domains and availability sets can be configured in a cluster set. This is beneficial for initial virtual machine placements and migrations between clusters.

In the example below, there are four clusters participating in the cluster set. Within the set, a logical fault domain will be created with two of the clusters and a fault domain created with the other two clusters. These two fault domains will comprise the Availability Set.

In the example below, CLUSTER1 and CLUSTER2 will be in a fault domain called **FD1** while CLUSTER3 and CLUSTER4 will be in a fault domain called **FD2**. The availability set will be called **CSMASTER-AS** and be comprised of the two fault domains.

To create the fault domains, the commands are:

```
New-ClusterSetFaultDomain -Name FD1 -FdType Logical -CimSession CSMaster -MemberCluster CLUSTER1,CLUSTER2 -
Description "This is my first fault domain"

New-ClusterSetFaultDomain -Name FD2 -FdType Logical -CimSession CSMaster -MemberCluster CLUSTER3,CLUSTER4 -
Description "This is my second fault domain"
```

To ensure they have been created successfully, Get-ClusterSetFaultDomain can be run with its output shown.

```
PS C:\> Get-ClusterSetFaultDomain -CimSession CSMaster -FdName FD1 | fl *
```

```
PSShowComputerName : True
FaultDomainType    : Logical
ClusterName        : {CLUSTER1, CLUSTER2}
Description         : This is my first fault domain
FDName             : FD1
Id                 : 1
PSComputerName     : CSMaster
```

Now that the fault domains have been created, the availability set needs to be created.

```
New-ClusterSetAvailabilitySet -Name CSMaster-AS -FdType Logical -CimSession CSMaster -ParticipantName
FD1,FD2
```

To validate it has been created, then use:

```
Get-ClusterSetAvailabilitySet -AvailabilitySetName CSMaster-AS -CimSession CSMaster
```

When creating new virtual machines, you would then need to use the `-AvailabilitySet` parameter as part of determining the optimal node. So it would then look something like this:

```
# Identify the optimal node to create a new virtual machine
$memoryinMB=4096
$vpcount = 1
$av = Get-ClusterSetAvailabilitySet -Name CSMaster-AS -CimSession CSMaster
$targetnode = Get-ClusterSetOptimalNodeForVM -CimSession CSMaster -VMMemory $memoryinMB -VMVirtualCoreCount
$vpcount -VMCpuReservation 10 -AvailabilitySet $av
$secure_string_pwd = convertto-securestring "<password>" -asplaintext -force
$cred = new-object -typename System.Management.Automation.PSCredential ("
<domain\account>",$secure_string_pwd)
```

Removing a cluster from cluster sets due to various life cycles. There are times when a cluster needs to be removed from a cluster set. As a best practice, all cluster set virtual machines should be moved out of the cluster. This can be accomplished using the **Move-ClusterSetVM** and **Move-VMStorage** commands.

However, if the virtual machines will not be moved as well, cluster sets runs a series of actions to provide an intuitive outcome to the administrator. When the cluster is removed from the set, all remaining cluster set virtual machines hosted on the cluster being removed will simply become highly available virtual machines bound to that cluster, assuming they have access to their storage. Cluster sets will also automatically update its inventory by:

- No longer tracking the health of the now-removed cluster and the virtual machines running on it
- Removes from cluster set namespace and all references to shares hosted on the now-removed cluster

For example, the command to remove the CLUSTER1 cluster from cluster sets would be:

```
Remove-ClusterSetMember -ClusterName CLUSTER1 -CimSession CSMaster
```

Frequently asked questions (FAQ)

Question: In my cluster set, am I limited to only using hyper-converged clusters?

Answer: No. You can mix Storage Spaces Direct with traditional clusters.

Question: Can I manage my Cluster Set via System Center Virtual Machine Manager?

Answer: System Center Virtual Machine Manager does not currently support Cluster sets

Question: Can Windows Server 2012 R2 or 2016 clusters co-exist in the same cluster set?

Question: Can I migrate workloads off Windows Server 2012 R2 or 2016 clusters by simply having those clusters join the same Cluster Set?

Answer: Cluster sets is a new technology being introduced in Windows Server Preview builds, so as such, does not exist in previous releases. Down-level OS-based clusters cannot join a cluster set. However, Cluster Operating System rolling upgrades technology should provide the migration functionality that you are looking for by upgrading these clusters to Windows Server 2019.

Question: Can Cluster sets allow me to scale storage or compute (alone)?

Answer: Yes, by simply adding a Storage Space Direct or traditional Hyper-V cluster. With cluster sets, it is a straightforward change of Compute-to-Storage ratio even in a hyper-converged cluster set.

Question: What is the management tooling for cluster sets

Answer: PowerShell or WMI in this release.

Question: How will the cross-cluster live migration work with processors of different generations?

Answer: Cluster sets does not work around processor differences and supercede what Hyper-V currently supports. Therefore, processor compatibility mode must be used with quick migrations. The recommendation for Cluster sets is to use the same processor hardware within each individual Cluster as well as the entire Cluster Set for live migrations between clusters to occur.

Question: Can my cluster set virtual machines automatically failover on a cluster failure?

Answer: In this release, cluster set virtual machines can only be manually live-migrated across clusters; but cannot automatically failover.

Question: How do we ensure storage is resilient to cluster failures?

Answer: Use cross-cluster Storage Replica (SR) solution across member clusters to realize the storage resiliency to cluster failures.

Question: I use Storage Replica (SR) to replicate across member clusters. Do cluster set namespace storage UNC paths change on SR failover to the replica target Storage Spaces Direct cluster?

Answer: In this release, such a cluster set namespace referral change does not occur with SR failover. Please let Microsoft know if this scenario is critical to you and how you plan to use it.

Question: Is it possible to failover virtual machines across fault domains in a disaster recovery situation (say the entire fault domain went down)?

Answer: No, note that cross-cluster failover within a logical fault domain is not yet supported.

Question: Can my cluster set span clusters in multiple sites (or DNS domains)?

Answer: This is an untested scenario and not immediately planned for production support. Please let Microsoft know if this scenario is critical to you and how you plan to use it.

Question: Does cluster set work with IPv6?

Answer: Both IPv4 and IPv6 are supported with cluster sets as with Failover Clusters.

Question: What are the Active Directory Forest requirements for cluster sets

Answer: All member clusters must be in the same AD forest.

Question: How many clusters or nodes can be part of a single cluster Set?

Answer: In preview, cluster sets been tested and supported up to 64 total cluster nodes. However, cluster sets architecture scales to much larger limits and is not something that is hardcoded for a limit. Please let Microsoft know if larger scale is critical to you and how you plan to use it.

Question: Will all Storage Spaces Direct clusters in a cluster set form a single storage pool?

Answer: No. Storage Spaces Direct technology still operates within a single cluster and not across member clusters in a cluster set.

Question: Is the cluster set namespace highly available?

Answer: Yes, the cluster set namespace is provided via a Continuously Available (CA) referral SOFS namespace server running on the management cluster. Microsoft recommends having enough number of virtual machines from member clusters to make it resilient to localized cluster-wide failures. However, to account for unforeseen catastrophic failures – e.g. all virtual machines in the management cluster going down at the same time – the referral information is additionally persistently cached in each cluster set node, even across reboots.

Question: Does the cluster set namespace-based storage access slow down storage performance in a cluster set?

Answer: No. Cluster set namespace offers an overlay referral namespace within a cluster set – conceptually like Distributed File System Namespaces (DFSN). And unlike DFSN, all cluster set namespace referral metadata is auto-populated and auto-updated on all nodes without any administrator intervention, so there is almost no performance overhead in the storage access path.

Question: How can I backup cluster set metadata?

Answer: This guidance is the same as that of Failover Cluster. The System State Backup will backup the cluster state as well. Through Windows Server Backup, you can do restores of just a node's cluster database (which should never be needed because of a bunch of self-healing logic we have) or do an authoritative restore to roll back the entire cluster database across all nodes. In the case of cluster sets, Microsoft recommends doing such an authoritative restore first on the member cluster and then the management cluster if needed.

Change history for Failover Clustering topics in Windows Server 2016

6/29/2018 • 2 minutes to read • [Edit Online](#)

Applies to: Windows Server (Semi-Annual Channel), Windows Server 2016

This topic lists new and updated topics in the [Failover Clustering](#) documentation for Windows Server 2016.

If you're looking for update history for Windows Server 2016, see [Windows 10 and Windows Server 2016 update history](#).

June 2018

NEW OR CHANGED TOPIC	DESCRIPTION
Cluster sets	New topic

May 2018

NEW OR CHANGED TOPIC	DESCRIPTION
Configure and manage quorum	Migrated from the Previous Versions library.

April 2018

NEW OR CHANGED TOPIC	DESCRIPTION
Troubleshooting a Failover Cluster using Windows Error Reporting	New topic.
Scale-Out File Server for application data	Migrated from the Previous Versions library.
Hardware requirements	Migrated from the Previous Versions library.
Use Cluster Shared Volumes (CSVs)	Migrated from the Previous Versions library.
Create a failover cluster	Migrated from the Previous Versions library.
Prestage a cluster in AD DS	Migrated from the Previous Versions library.
Deploy a Cloud Witness for a Failover Cluster	Migrated from the Previous Versions library.

June 2017

NEW OR CHANGED TOPIC	DESCRIPTION
Cluster-Aware Updating advanced options	Added info about using run profile paths that include spaces.

April 2017

NEW OR CHANGED TOPIC	DESCRIPTION
Cluster-Aware Updating overview	New topic.
Cluster-Aware Updating requirements and best practices	New topic.
Cluster-Aware Updating advanced options	New topic.
Cluster-Aware Updating FAQ	New topic.
Cluster-Aware Updating plug-ins	New topic.
Deploy a cloud witness for a Failover Cluster	Clarified the type of storage account that's required (you can't use Azure Premium Storage or Blob storage accounts).

March 2017

NEW OR CHANGED TOPIC	DESCRIPTION
Deploy a cloud witness for a Failover Cluster	Updated screenshots to match changes to Microsoft Azure.

February 2017

NEW OR CHANGED TOPIC	DESCRIPTION
Cluster operating system rolling upgrade	Removed an unnecessary Caution note and added a link.