NAKIVO®

eBook

The Complete Guide to Hyper-V Clustering

Contents

Overview

This e-book serves as a comprehensive guide to clustering functionality in Hyper-V and how you can leverage this technology in your virtual environment. Beginning with the basics, the material progresses through some essential concepts and includes full walkthroughs for system administrators.

Hyper-V Server 2016 version is considered in this eBook within the framework of failover clustering.

Definition of a Cluster

A **cluster** is a group of independent servers connected over a network that work together as a single system. Dedicated high-speed networks are used to connect servers (called **nodes** in this context) with each other in the cluster. A minimum of two nodes are needed for a cluster to function.

Clusters originated in the 1960s with the idea of connecting multiple computers to perform shared tasks. In the modern world, clusters are popular due to the widespread growth of cloud computing, which ensures horizontal scalability for businesses. The main advantages of clusters are increased computing power, load balancing, fault tolerance, and high availability.

Hyper-V Failover Cluster & How It Works

A Hyper-V failover cluster is a set of Hyper-V servers connected with one another, as well as to shared storage, over a network that is managed by special Hyper-V software to increase scalability and availability. A failover cluster provides **high availability (HA)** of the virtual machines (VMs) to reduce downtime in the event of Hyper-V server failure by executing automatic VM failover. Automatic failover is much faster than manually reloading the VMs or reloading physical servers.

A cluster consists of Hyper-V servers (nodes) that are connected with each other and to a shared storage *(see Figure 1)*. A cluster has systems that continuously monitor all the nodes and VMs within it. Cluster nodes periodically exchange special heartbeat signals through at least two networks, which prevents the problem of having a single point of failure. Dedicated cluster communication networks are usually used for this purpose.

Every node sends a heartbeat once per second. If a cluster node does not provide a steady heartbeat (more specifically, if 5 heartbeats fail within a 10-second period), the system considers the node to have failed. The failed node's workload is then transferred to other nodes within the cluster. Thus, the VMs that were running on the failed node are restarted on healthy nodes *(see Figure 2)*. This is termed **failover**.

The virtual machines use the virtual disks located on a shared storage on **cluster shared volumes (CSVs)** that are accessible by all cluster nodes simultaneously. During the failover process, VMs are unreachable until they are reloaded on different hosts. This is because the high availability that is supported by the Hyper-V failover cluster is not equal to **fault tolerance.** The other Hyper-V hosts that are part of a cluster must have sufficient computing resources to accommodate the VMs that were running on the failed host before failover.

Before going any further into exploring how Hyper-V failover clusters work, it is important to understand the concept of a **quorum** in the context of Hyper-V clusters.

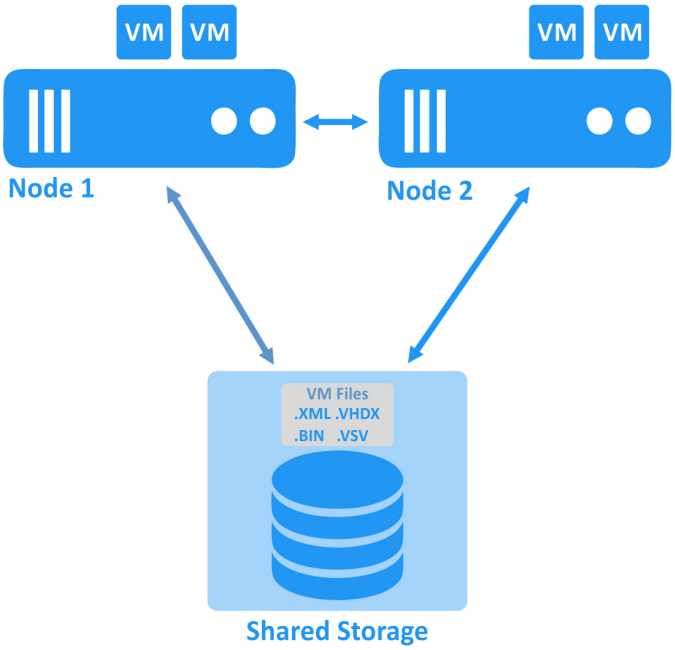

Figure 1. Both nodes work fine.

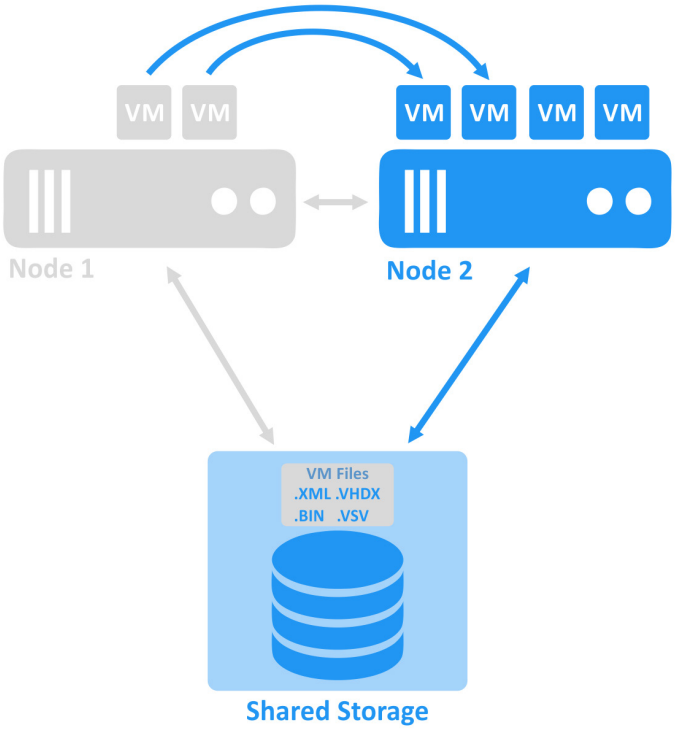

Figure 2. One node has failed. VMs are migrating to the other node.

Quorums

The Hyper-V clustering **quorum** is a feature that helps provide failover cluster protection. A quorum is defined as the majority of votes required for a cluster to function properly according to the Windows clustering models. If any node goes down, a voting event occurs; each node has a vote. If there are enough votes to form a majority, then the cluster continues to operate. If there are not enough votes, the cluster ceases to function. (This approach also helps avoid a **split-brain** situation, wherein the source and failover VMs can run simultaneously while different clients connect and write changes to different VMs.) This section further explores the concept of quorums, the types of quorums, and their particularities, step by step.

Disk Only. This is the oldest quorum model. The cluster can survive losing all nodes but one – this is the node that has a connection to a quorum disk. This disk is called the **disk witness.** In this case, the disk constitutes a "single point of failure" element. If the disk fails, the entire cluster ceases to be functional *(see Figure 3)*.

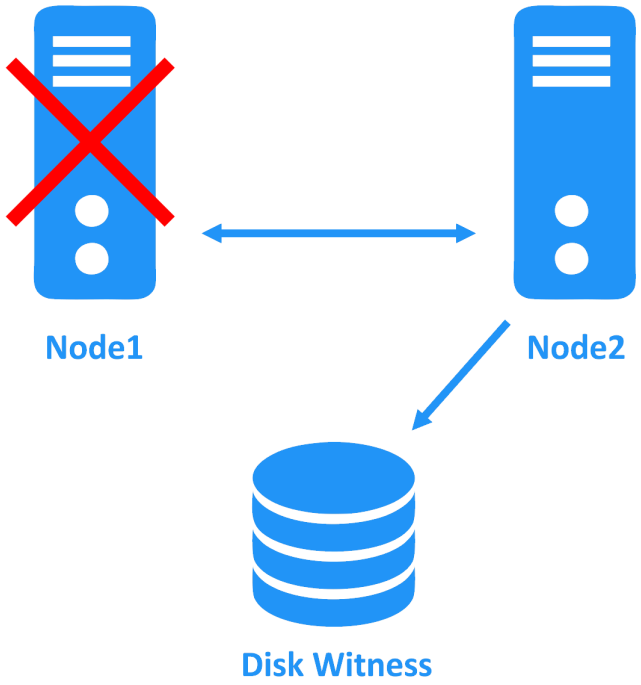

Figure 3. Diagram illustrating the disk-only quorum model. In this scenario, the cluster continues to operate, because the node connected to the disk witness has not failed.

Node Majority. In this system, each node has a vote. The disk witness has no vote. This model is better suited for clusters that include odd numbers of nodes *(see Figures 4, 5, 6, and 7).*

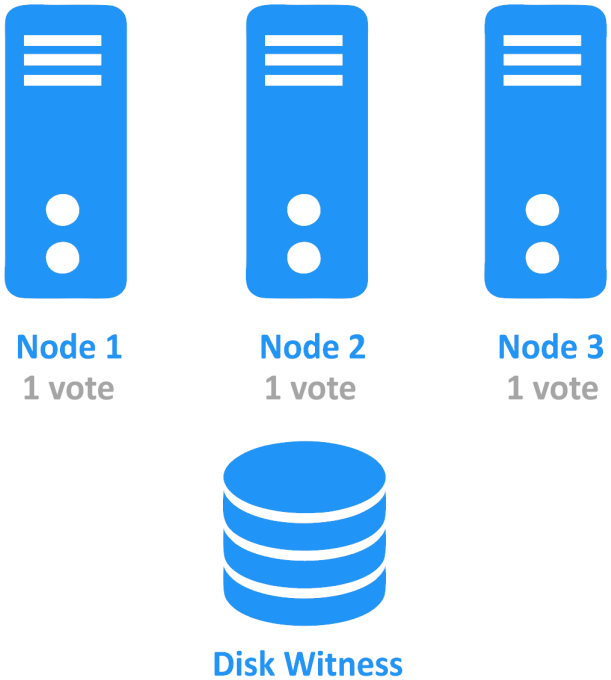

Figure 4. There are three nodes, so 2 votes out of 3 are needed to form a quorum. This fully functional cluster has 3 votes out of 3.

100% > 50%, so the cluster is operational.

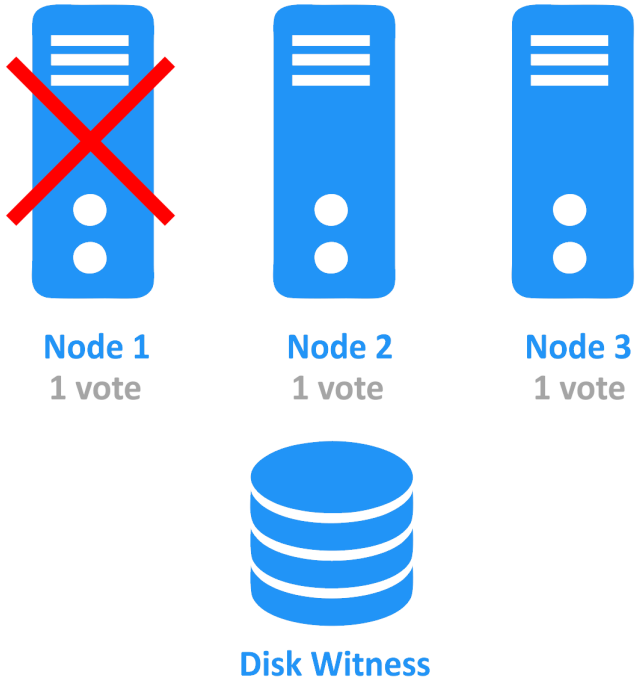

Figure 5. There are three nodes, so 2 votes out of 3 are needed for a quorum. The cluster has 2 votes out of 3.

66% > 50%, so the cluster is functional.

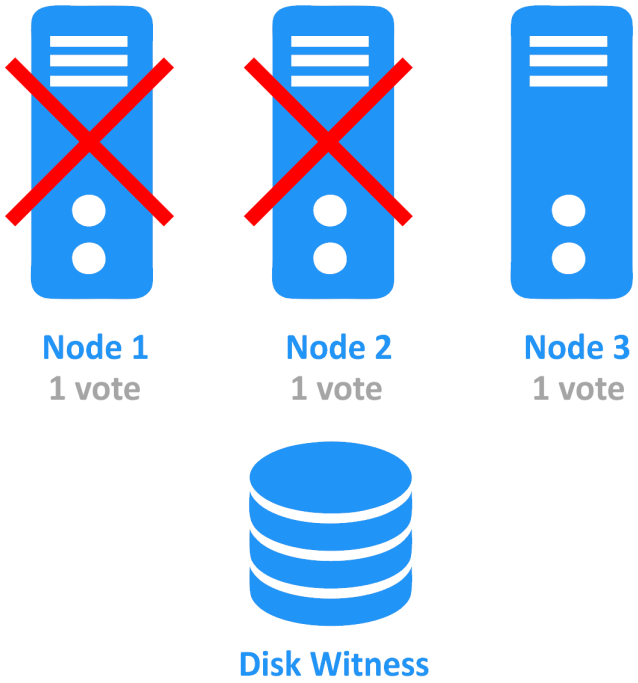

Figure 6. There are three nodes, so 2 votes out of 3 are needed for a quorum. The cluster only has 1 vote out of 3.

33% < 50%, so the cluster is not functional.

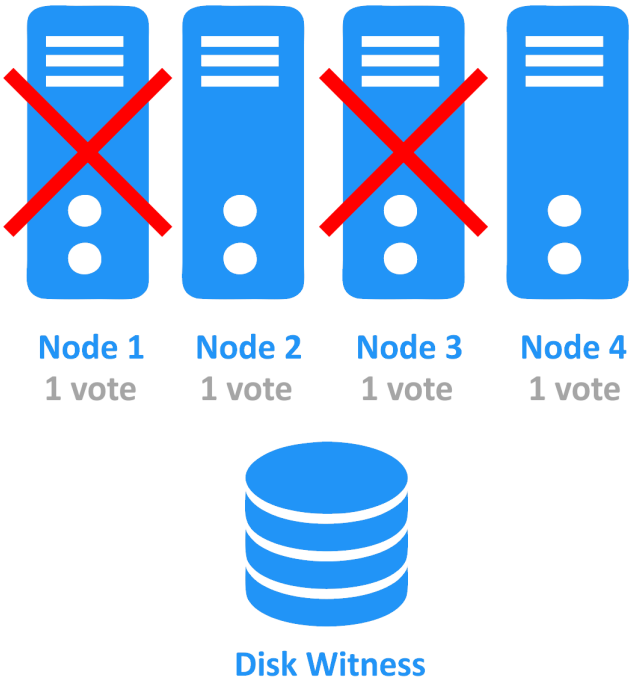

Figure 7. There are four nodes, so 3 votes out of 4 are needed for a quorum. The cluster has 2 votes out of 4, which is equal to 50%. However, to function, the cluster must have greater than 50% functionality.

50% = 50%, so the cluster is not functional.

Node and Disk Majority. Each node of a cluster has one vote, as does the disk witness. This model is better suited for clusters that comprise even numbers of nodes *(see Figures 8, 9, 10).*

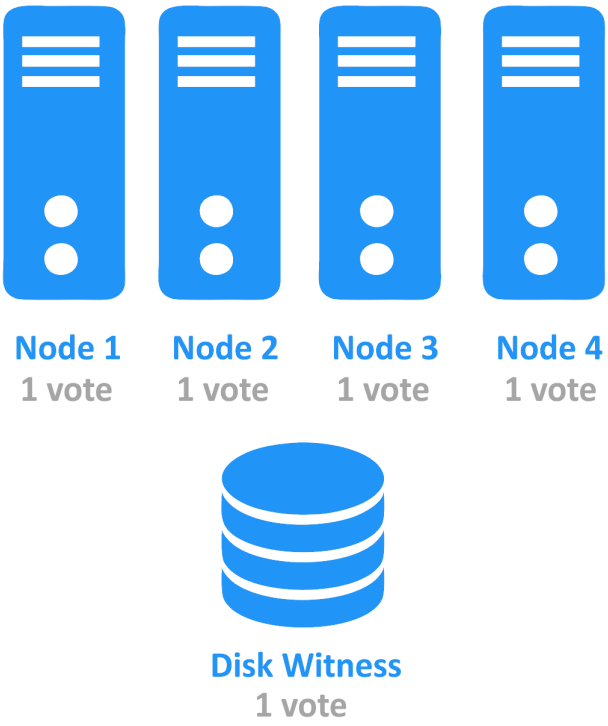

Figure 8. Three votes out of five (four nodes plus the disk witness) are needed to form a quorum. This fully operational cluster has 5 votes out of 5.

100%> 50%, so the cluster is functional.

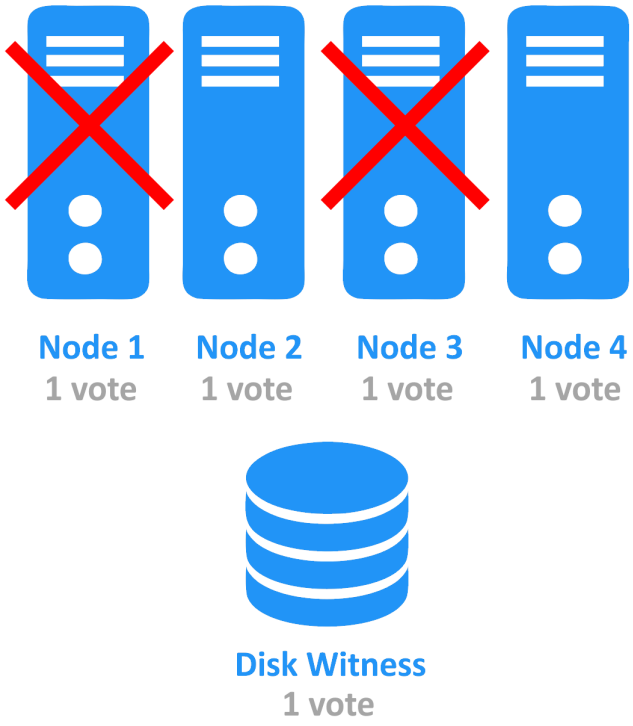

Figure 9. Three votes out of five (four nodes plus the disk witness) are needed to form a quorum. This cluster has 3 votes.

60% > 50%, so the cluster is functional.

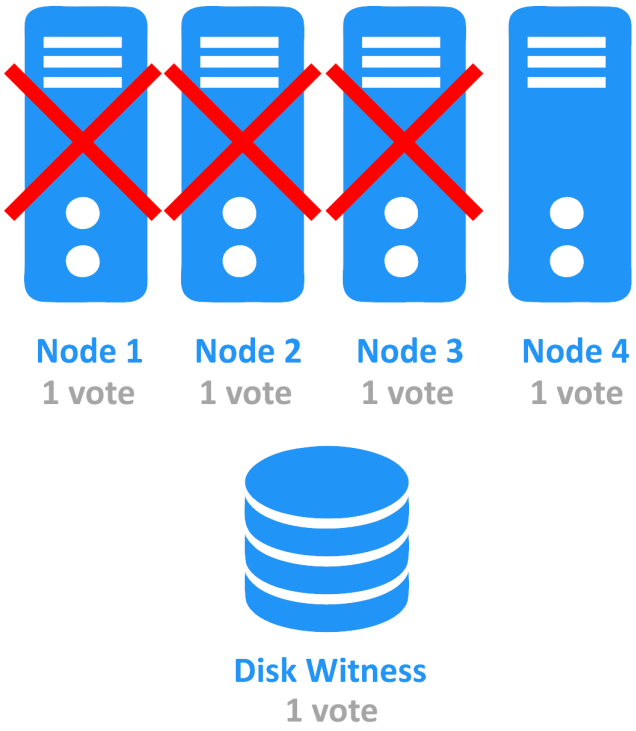

Figure 10. Three votes out of five (four nodes plus the disk witness) are needed to form a quorum. This cluster only has 2 votes.

40% < 50%, so this cluster is not operational.

In both cases (*Node Majority* as well as *Node and Disk Majority*), more than 50% of the votes are needed for the cluster to remain operational. Suppose, for example, there is a three-node cluster using the Node Majority quorum mode. For the cluster to function, more than 50% of the votes are needed – that's two votes in this case. Thus, the cluster can survive the failure of one node *(see Figure 4 and Figure 5)*. However, if two nodes go down, then the whole cluster fails *(see Figure 6).*

Compare *Figure 7* and *Figure 9* to understand the advantages of using the *Node and Disk Majority* quorum mode for clusters with even numbers of nodes. Accordingly, the *Disk Majority* quorum mode is more suitable for clusters with odd numbers of nodes.

Node and File Share Majority. The advantage of this mode over the modes described above is that a **File Share Witness** can be used instead of a disk witness. One or several nodes can be excluded from voting to avoid shutdown of the whole cluster in case of failure of more than 50% of the nodes. A functional node that is excluded from voting node remains a fully working part of cluster and can run VMs. The Node Weight parameter must be set to 0 to eliminate the node from voting.

Example:

Suppose a cluster has five nodes, three of which are located in another office, and a File Share Witness. Normally, if the three nodes located in another office were to fail, the two remaining nodes and File Share Witness would only have three votes between them – which is exactly 50%. This is not enough to form a quorum, so the cluster would fail.

However, if one node from the remote office is eliminated from voting and the three nodes from that office fail, then the two remaining nodes together with the File Share Witness make up 60% (3 out of 5) of the votes, which represents a majority. The cluster continues to operate *(see Figure 11).*

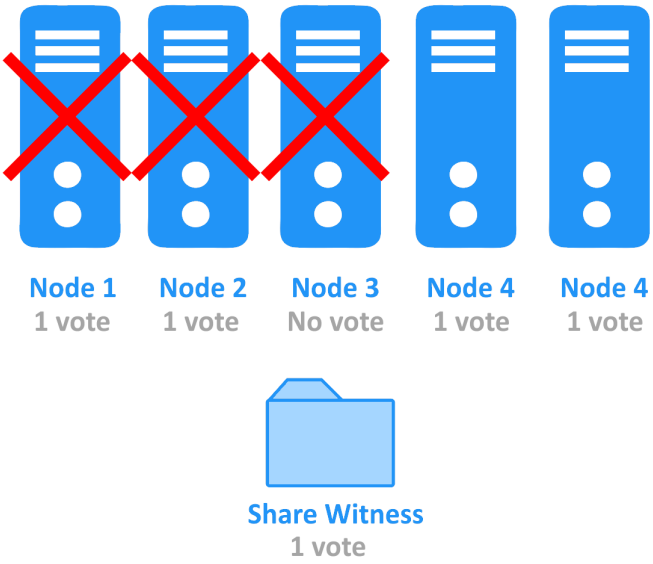

Figure 11. The cluster has 3 votes out of 5, because one of the failed nodes in the remote office has no vote. 60% > 50%, so the cluster continues to function.

Dynamic Quorum. After considering the previous quorum modes, the question arises of how to automate the behavior of the cluster in the event of node failure and keep the cluster in a working state. The Dynamic Quorum addresses this. When using a Dynamic Quorum, all nodes have NodeWeight and DynamicWeight parameters. A Dynamic Quorum operates by altering the DynamicWeight parameters.

If one of the cluster nodes fails, then the Cluster service automatically excludes such host from voting by setting its DynamicWeight parameter to zero. When the node returns to a normal state, it is reassigned its initial DynamicWeight value. By default, every 5 minutes, the system checks the cluster's state and recounts the votes. The votes are also recounted after certain events, including:

- Adding a new node to the cluster. The node adds a vote.
- Shutdown of a node. The node's vote is removed.
- Node failure. The node with the lowest NodeID has its DynamicWeight reset to 0.

Note:

You can check the NodeID parameter in PowerShell by entering the **Get-ClusterNode** command. Don't confuse NodeID and NodeWeight parameters. You can check the NodeWeight and DynamicWeight parameters by entering the following command: ***Get-ClusterNode-Name *|ft NodeName, DynamicWeight, NodeWeight-AutoSize.***

When using a Dynamic Quorum, the cluster can survive even if there is only one node left within it. For this to occur, the nodes must go down one after another, but not simultaneously. See the example illustrated below *(Figures 12 through 15).*

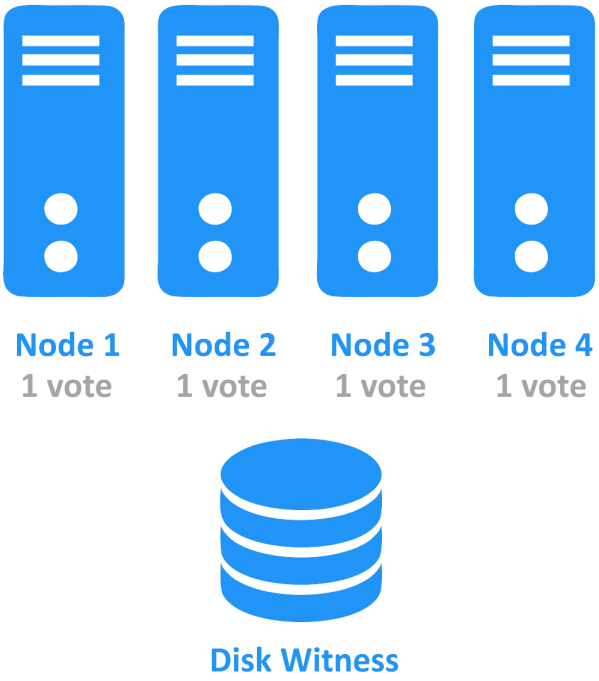

Figure 12. 3 votes are required for a quorum. All nodes are functioning normally. The cluster has 4 votes out of 4, which is 100%, so it continues to operate.

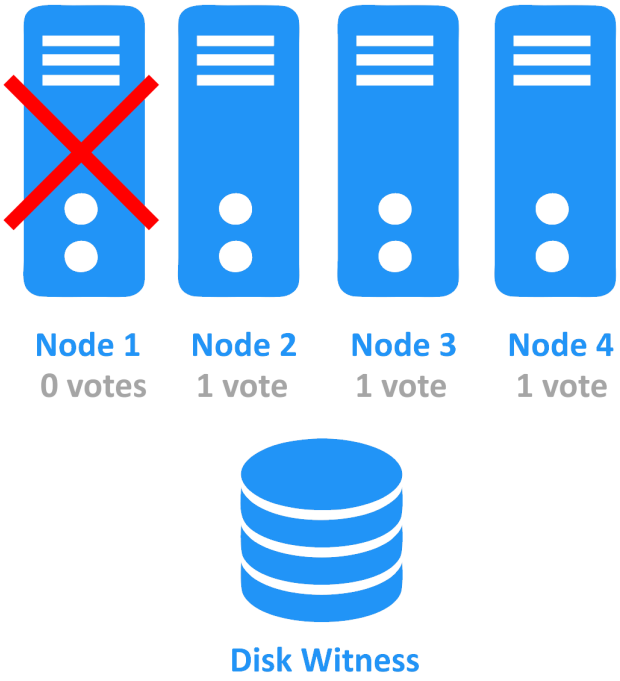

Figure 13. One node is down. 2 votes are required for a quorum. The cluster has 3 votes out of 3, which is 100%, so it continues to operate.

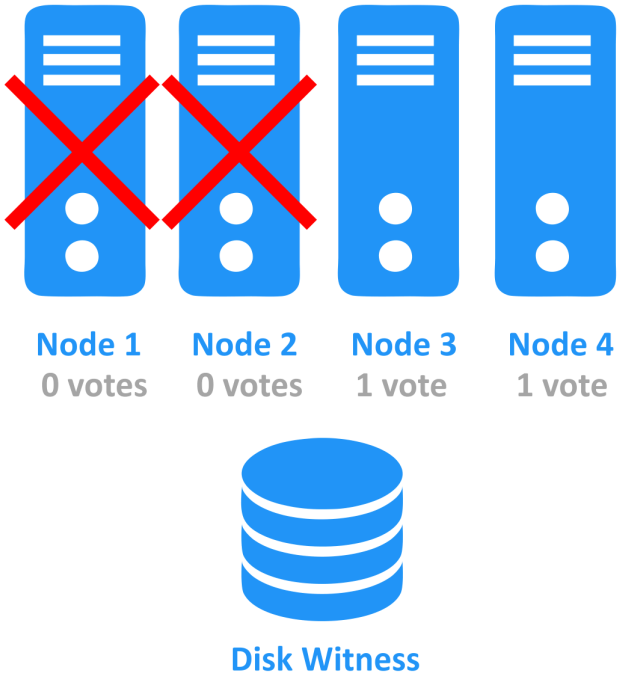

Figure 14. The second node fails. There are now only 2 votes; both are needed to form a quorum. The cluster has 2 out of 2 votes, which is 100%. Thus, it continues to work.

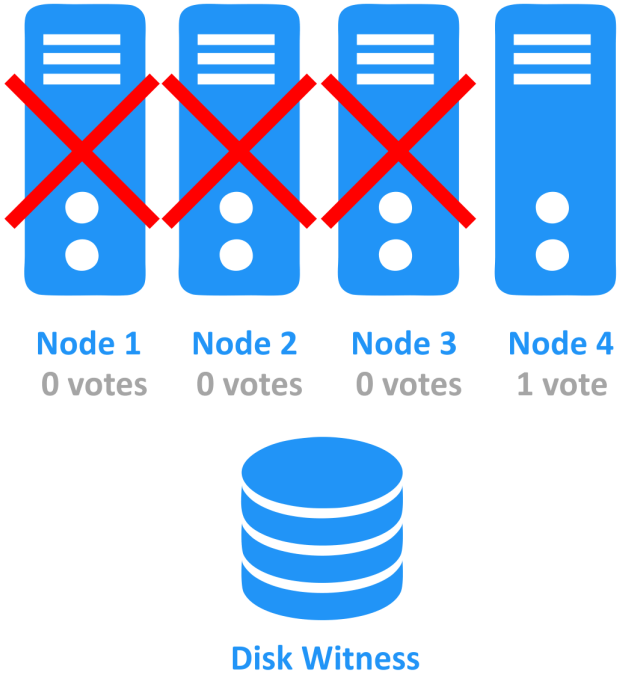

Figure 15. A third node has failed; only one working node is left in the cluster. However, the quorum now only needs 1 vote. The cluster has 1 vote out of 1, which is 100%. Ergo, the cluster is still functional.

Dynamic Witness. A quorum functions best when there is an odd number of votes, as explained above in the discussion of the *Node and Disk Majority* mode. In the event that, after a recount, the Dynamic Quorum yields an even number, a **dynamic witness** can resolve this issue. When using a dynamic witness, *DynamicWeight* is a variable value that depends on the number of working nodes within the cluster. Thus, node vote status can be adjusted automatically.

Example:

In a five-node cluster, the disk witness has a null DynamicWeight *(see Figure 16)*. If one of the nodes fails, then the failed node is assigned a null DynamicWeight and the disk witness receives a DynamicWeight of 1 *(see Figure 17)*. If a second node fails, both failed nodes have null DynamicWeights, and the disk witness also gets a null DynamicWeight *(see Figure 18)*. If a third node fails, then the three failed nodes all get null DynamicWeights and the witness disk's DynamicWeight is once again equal to 1, etc. *(see Figure 19)*. No matter what, there are always an odd number of votes.

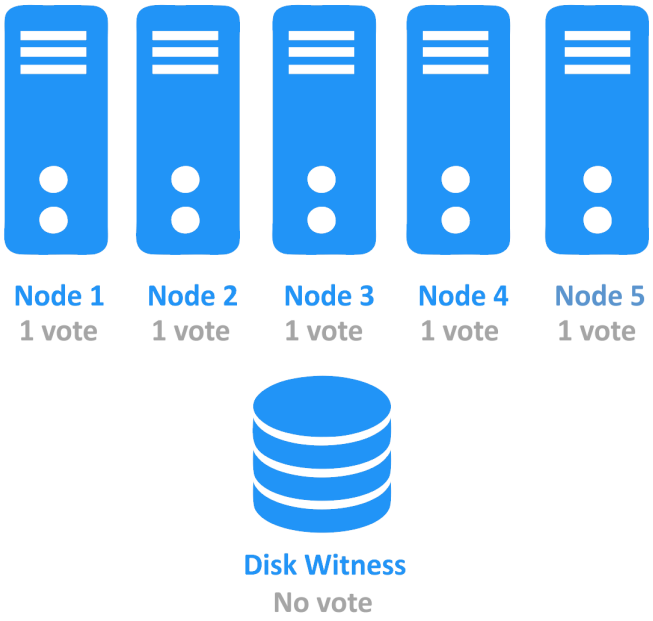

Figure 16. All 5 nodes are functioning properly. Only 3 votes out of 5 are needed for a quorum, but the nodes provide 5 votes out of 5. The cluster functions normally.

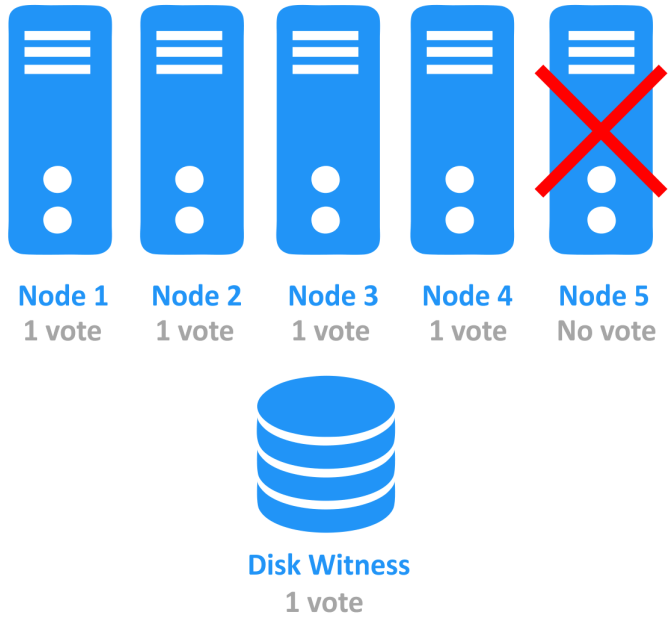

Figure 17. One node fails. The cluster now has an even number of working nodes (4), so the disk witness is given a vote. 3 votes out of 5 are required to constitute a quorum.

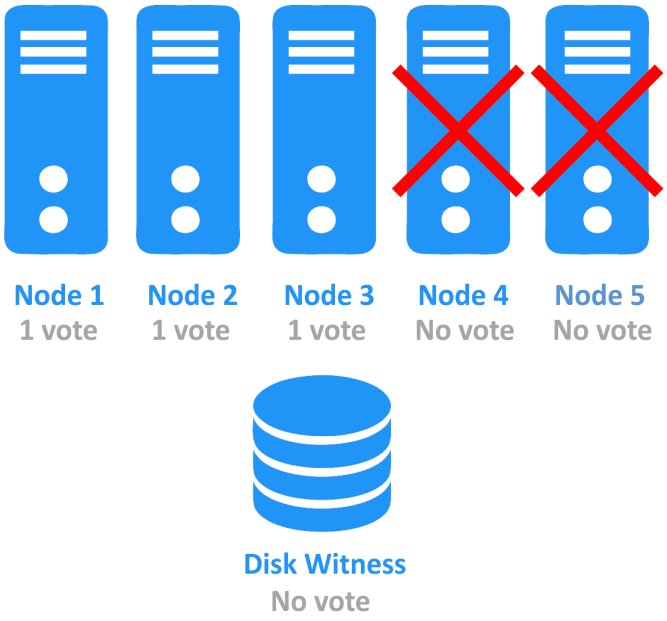

Figure 18. A second node has failed. 2 votes out of 3 are needed for a quorum.

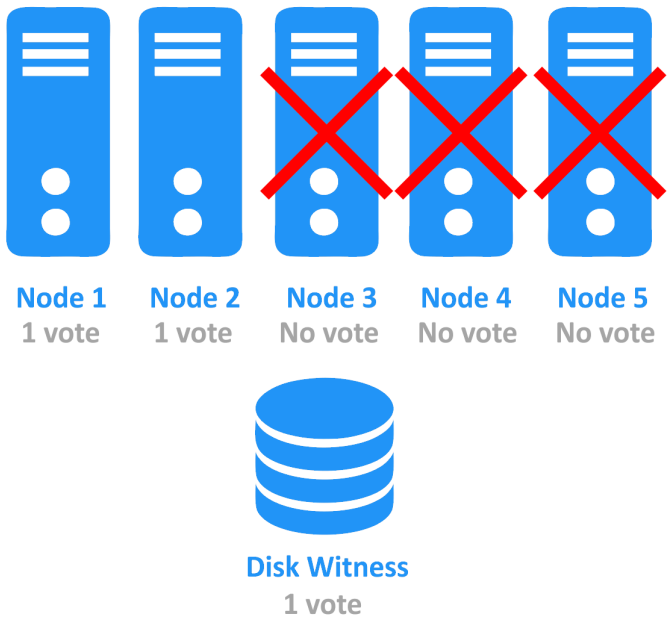

Figure 19. The third node goes down. 2 votes out of 3 are required for a quorum.

The *LowerQuorumPriorityNodeID* parameter allows you to control which nodes remain operational if a cluster consists of nodes located at different sites in a case of, for example, a network interruption that results in the loss of 50% of your nodes. The administrator can decide which nodes would have their DynamicWeights reduced in order for the cluster to resume operation on primary or secondary site after achieving a quorum. Thus, the cluster can continue working with the nodes located at the designated site.

Use Cases for Hyper-V Failover Clusters

Hyper-V failover clusters should be used for Hyper-V virtual environments where little to no downtime of VMs can be afforded. In the event of a single VM failure, there are disadvantages for the company, but if an entire Hyper-V host fails, taking down all the VMs running on the host, the situation can become critical. A Hyper-V failover cluster can help you keep all the advantages of virtualization while eliminating the disadvantages posed by a potential Hyper-V host failure. With Hyper-V failover clustering, companies can achieve high availability and scalability for their Hyper-V virtual environments, preventing the issue of a “single point of failure”.

Additional Features That Can Be Used in Hyper-V Failover Clusters

Dynamic optimization

Dynamic optimization is a load balancing feature that helps you ensure you maintain a balanced cluster. A cluster can have uneven workload distributions: the VMs running on one Hyper-V host might be placing a very high load on the CPU, RAM, disks, and network, while the hardware resources of a second Hyper-V host are almost idle. When Dynamic Optimization is enabled, the system periodically checks the resource utilization (CPU, RAM, disk input/output, network input/output) of the Hyper-V hosts within the cluster and compares the information collected from hosts with configured threshold values. If the workload of one host is significantly higher than the workload of a second host and exceeds the threshold values, then the system decides to optimize the cluster by performing a live migration of VMs *(see Figure 20)*. This process can be performed manually or automatically. In the automatic mode, dynamic optimization is performed with an interval of 10 minutes by default. Before making a decision to migrate a VM, dynamic optimization checks that there would be no warnings or errors triggered by the placement. Particular VMs can be excluded from migration with dynamic optimization.

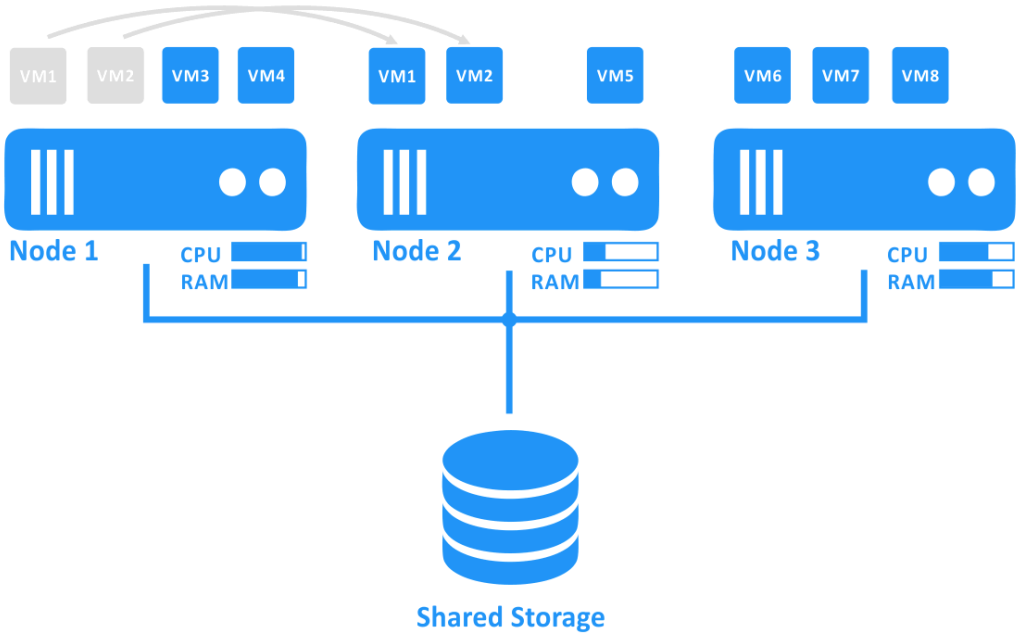

Figure 20. Dynamic optimization.

Power optimization is an optional feature of dynamic optimization that helps you save energy. Power optimization automatically powers off hosts that are not necessary to meet the resource requirements for the VMs running in the cluster *(see Figure 21)*. When the VMs need more resources again, the hosts are powered back on. The feature operates according to a schedule that you set. By default, every 10 minutes, the system considers each host of the cluster and asks: "Are there enough resources in the cluster to run all currently running VMs without negative impact on their performance if this host were powered off?" If there are enough resources between the hosts of the cluster to run all the VMs currently powered on, then the VMs running on the selected host are migrated elsewhere and the host is powered off.

Power optimization can be useful when most of your VMs are only used for part of the day (e.g., during working hours), but some VMs must remain online 24/7. This feature can be scheduled according to demands based on typical workloads. For example, you could schedule power optimization to operate only outside of regular business hours, when many of your VMs do not need to be operational. All hosts are powered back on once the scheduled period of power optimization has expired (in this example, once business hours begin).

Power optimization cannot violate the cluster quorum. The minimum number of nodes is always preserved to ensure that the cluster functions properly. A cluster created with the Virtual Machine Manager can add one vote to the quorum with a disk witness. For example, in a 5-node cluster, power optimization could power off one node. Four nodes would be left to survive a failure of one host, because 3 hosts must work properly (this forms a quorum for the cluster). The more hosts you have within the cluster, the more hosts can be powered

off by the power optimization feature. If you have a 16-node cluster with a disk witness, then up to 7 hosts can be powered off for energy economy. The overall rule is that the number of healthy nodes needed for the cluster to function properly is “quorum plus 2”.

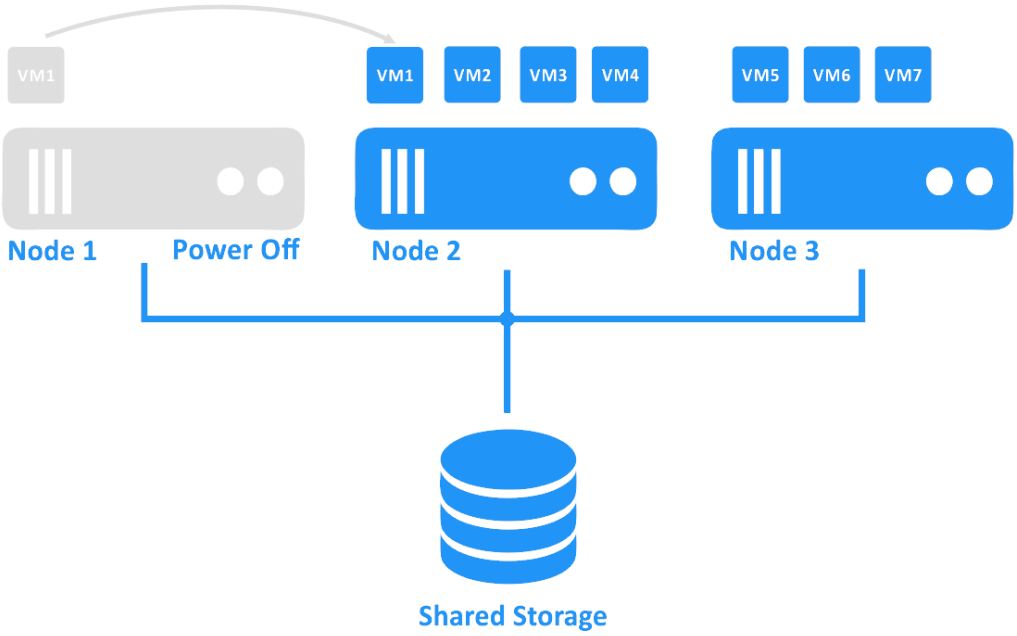

Figure 21. Power optimization.

Quick Migration and Live Migration of VMs

Quick migration is a type of VM migration that can be used for failover (high availability) and requires a brief period of downtime – approximately one minute, depending on the network speed as well as your VM memory settings. Quick migration is used for both planned and unplanned migration of VMs between cluster nodes. The configuration files and virtual disks of the VMs are located on a shared storage that is accessible for all nodes *(see Figure 22)*. The VM is put into a saved state *(see Figure 23)* and memory information is transferred. The VM is then unregistered from the source host and registered on the destination host *(see Figure 24)*, before resuming operation on the new host *(see Figure 25)*. After being put into a saved state and before resuming work on a new host, a VM is unreachable. Accordingly, this migration method is best used in the following cases:

- To migrate running VMs outside of business hours.
- For VMs that are not running at the time (e.g., when the host is under maintenance).
- During failover.

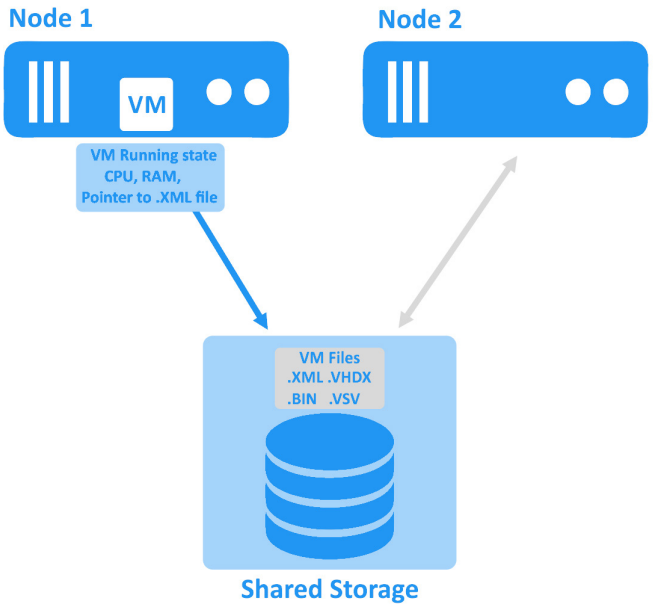

Figure 22

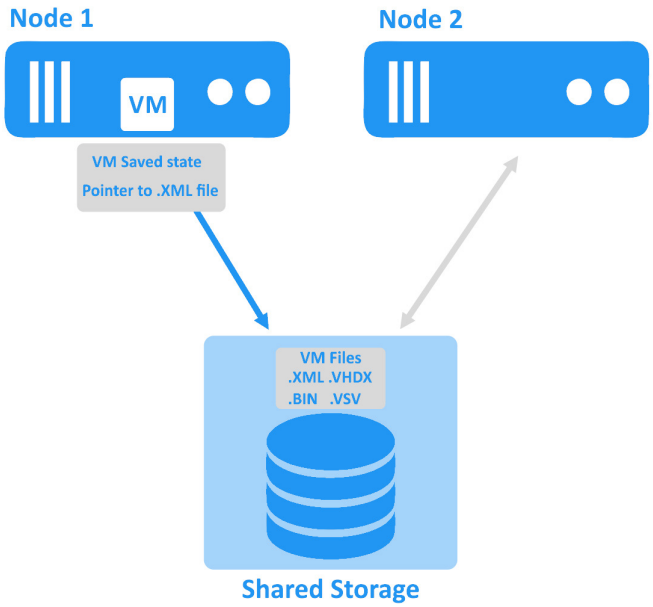

Figure 23

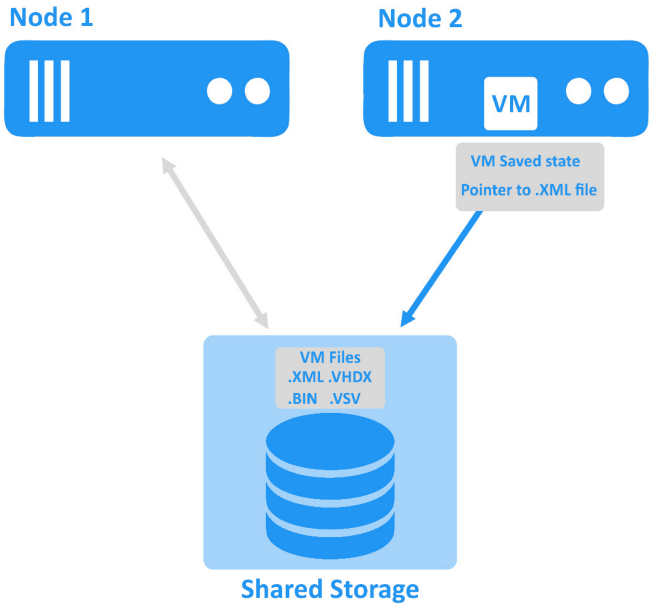

Figure 24

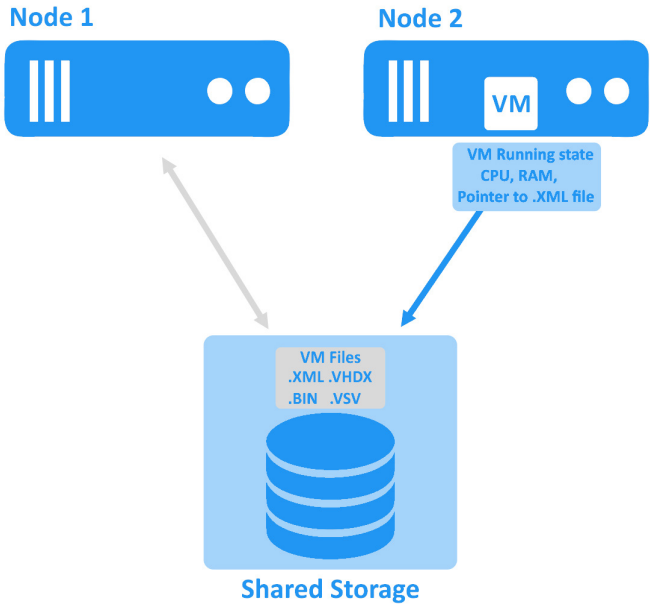

Figure 25

Live migration is a newer feature that was first released for Hyper-V 2.0 and provides greater flexibility than quick migration. This feature is usually used for planned migration of running VMs between hosts without any downtime. To be more precise, live migration results in an unnoticeable moment of downtime by moving VMs without losing availability of service. In detail, the process of live migration consists of the following steps:

1. The administrator initiates the live migration of the VM from one Hyper-V host to another *(Figure 26)*. A copy of source VM's specification (with dependencies) is created on the destination host.
2. The source VM's memory pages are tracked and copied to the destination VM *(Figure 27)*. The first copy is a copy of all the source VM's memory. As the source VM continues running, its memory is modified. Any pages that were changed during the process (termed **dirty pages**) are copied repeatedly over several (increasingly short) iterations, until there are no changed memory pages on the source VM vs. the destination VM *(Figure 28).*
3. The state of the source VM is paused on the source host. The VM becomes temporarily unreachable *(Figure 29).*
4. The VM's state (including the processor's state) is copied from the source to the destination host. The VM copy is now complete.
5. The VM is resumed on the destination host and becomes reachable again *(Figure 30)*. The ARP tables for routing devices are updated.
6. If the VM is running successfully on the destination host, all traces of the VM are deleted from the source host *(Figure 31).*

After pausing in step 3 and before resuming in step 5, the VM is offline. If you ping the VM's IP address during the live migration event, you might notice a packet being lost; this indicates the latency period during which the VM is unavailable. The ping utility uses ICMP protocol for IP network diagnostics. Most applications use protocols that tolerate latencies of a few seconds, allowing you to maintain the established connection state. Thus, with live migration, a VM can transparently migrate without any interruption in service.

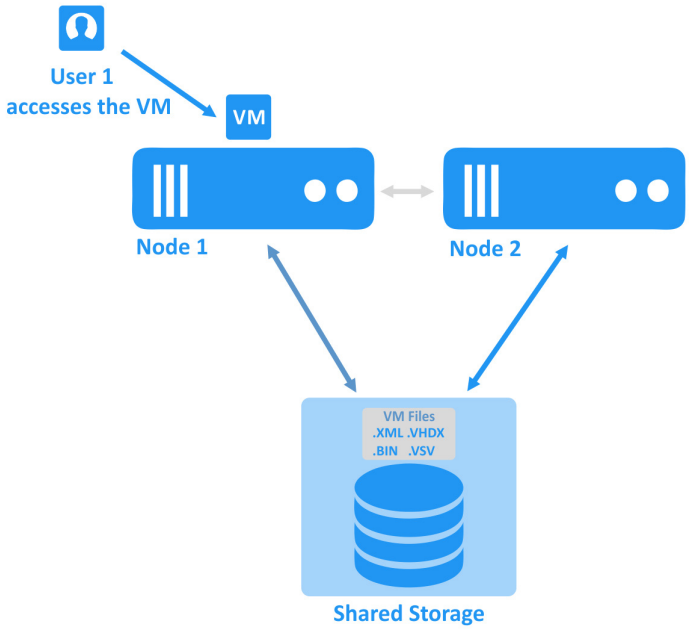

Figure 26

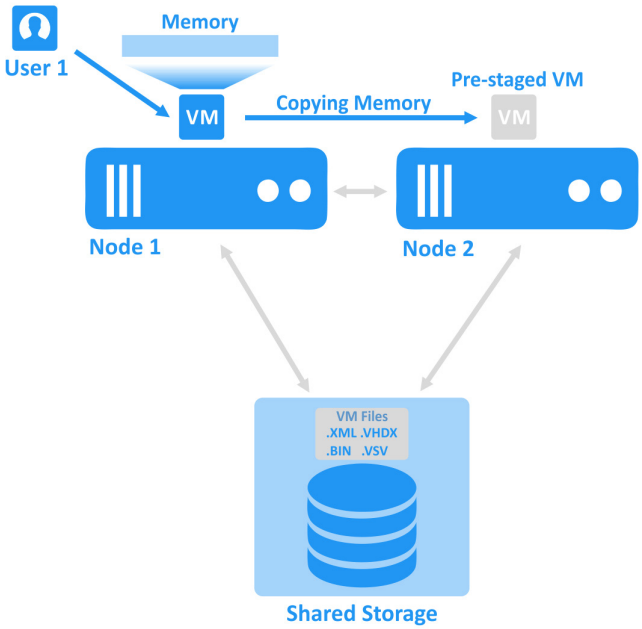

Figure 27

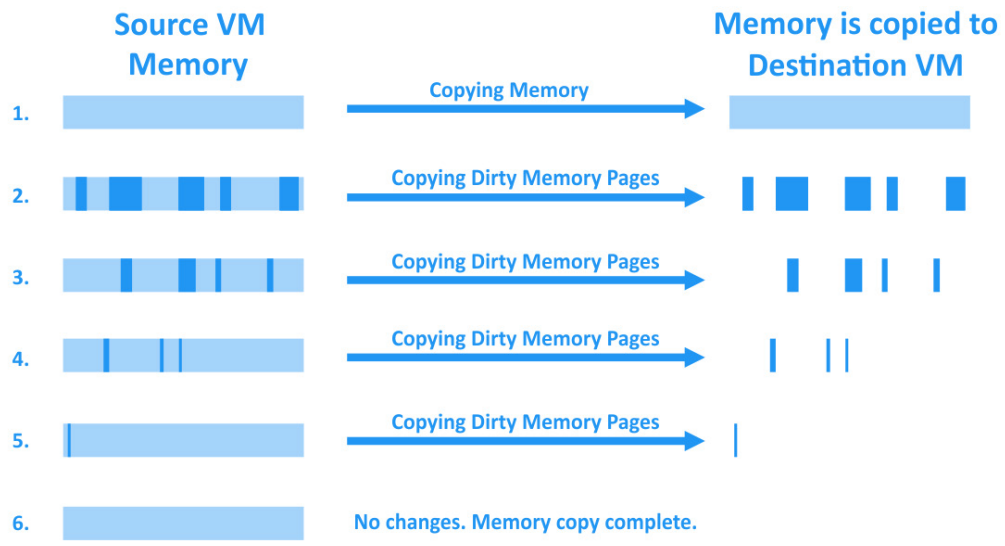

Figure 28

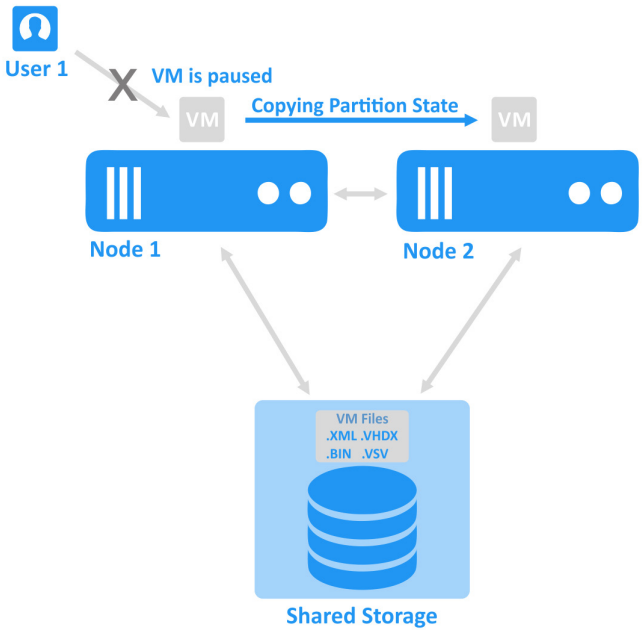

Figure 29

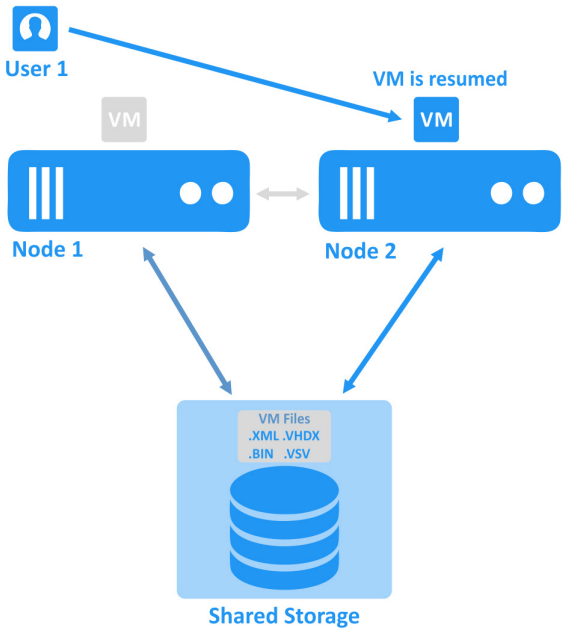

Figure 30

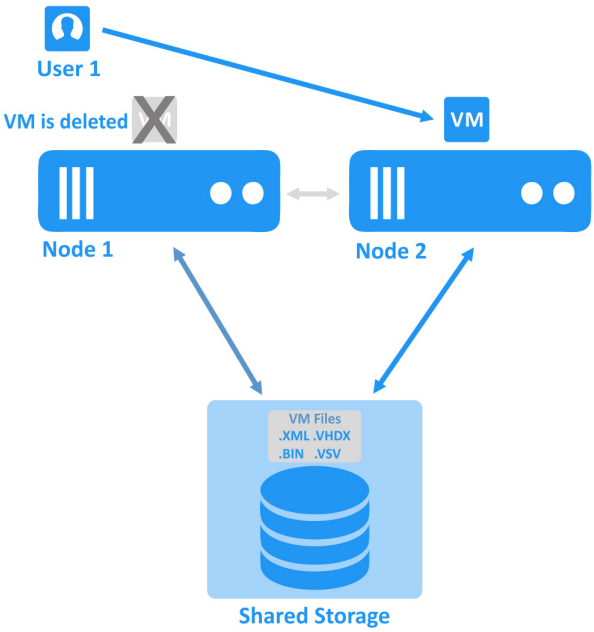

Figure 31

Hyper-V Storage Migration

This feature allows the migration of file-based components of a virtual machine from one shared storage to another without downtime and without changing the host on which the VM is run. This can be useful when you are replacing your NAS (Network Attached Storage) or SAN (Storage Area Network) as well as for redistributing the data between Cluster Shared Volumes after adding more disk space.

The following components of a VM can be migrated with storage migration:

- The VM's virtual hard disks (VHDX files).
- The VM's checkpoints.
- The VM's configuration files (xml, vmcx, bin, csv).
- ISO images that are attached to the VM and can be used as virtual DVD-ROMs.
- The paging file that belongs to the VM.

The running state of a VM cannot be transferred with storage migration; live migration is needed for this purpose.

Cluster Rolling Upgrade

This is a new feature that allows administrators who have a working Hyper-V cluster configured on Windows Server 2012 R2 to upgrade the host's operating system to Windows Server 2016 without interrupting the workloads. Using Hyper-V on Windows Server 2016, you can take advantages of new features.

To execute a Cluster Rolling Upgrade, the following operations must be performed in sequence on each node, one after another:

- The Hyper-V host must be drained (live migration of VMs from a host), put into maintenance mode, and excluded from the cluster.
- A clean installation of Windows Server 2016 must be performed on the selected host.
- If the cluster is operating in a domain, the new installed OS must be added to the appropriate Active Directory (AD) domain and the appropriate users must be created.
- Using the Server Manager or PowerShell, the Hyper-V and Failover Clustering server roles must be added.
- Once the roles have been added, check the virtual switch; its name must be the same as the name before upgrade, for all nodes in cluster. If there is no virtual switch, create a new one.
- After this, the updated node can be added back to the existing cluster.
- At this point, VMs can be migrated to the updated node. Then, the next Hyper-V host must be drained as explained above; repeat the process for each node.

While you execute this process, your Hyper-V failover cluster contains both Windows Server 2012 R2 and Windows Server 2016 nodes. This transitory mode is called **mixed-OS mode**. The core resources of the cluster are moved to the nodes with the newer OS.

See *Figure 32* for a visual representation of the process.

Once all hosts have been upgraded successfully in this way, the cluster's functional level can be upgraded to the Windows Server 2016 functional level, providing you more advanced Hyper-V capabilities.

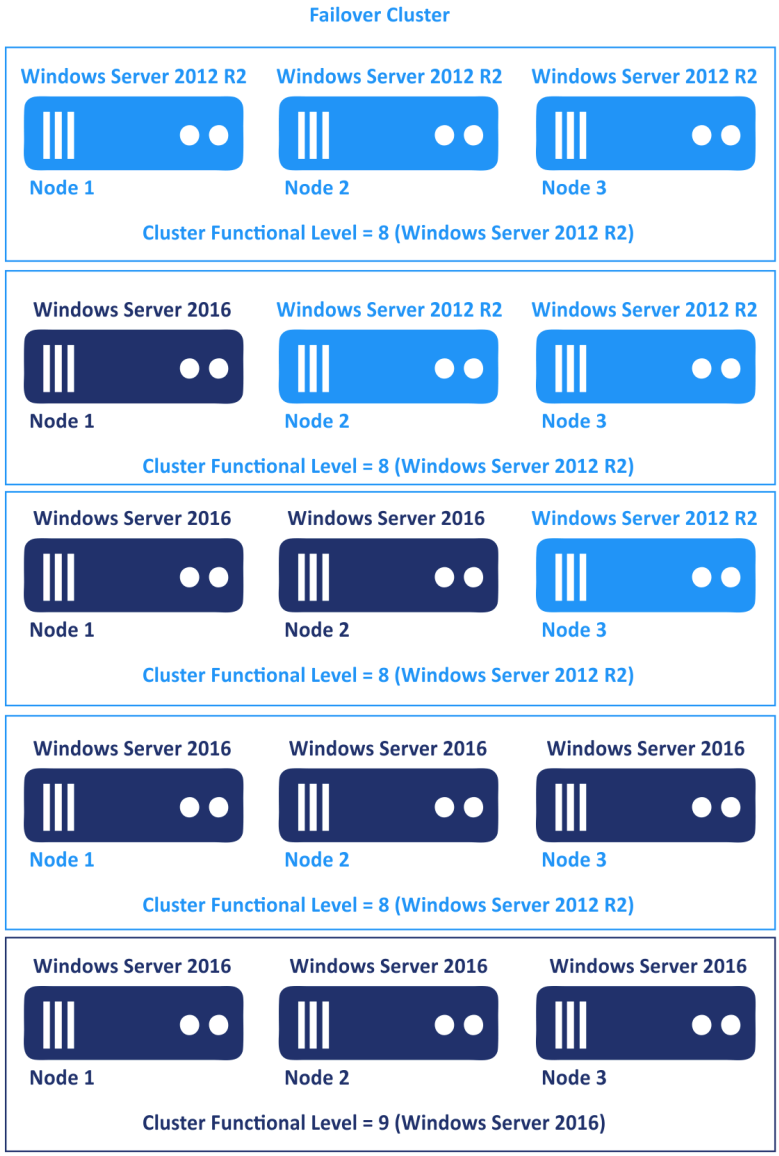

Figure 32

If you would like to have a Hyper-V 2016 Failover cluster, but you don't have a Hyper-V 2012 R2 cluster that can be upgraded, you should consider deploying a Hyper-V 2016 failover cluster from scratch. Start by making sure your system meets the requirements.

Hyper-V Clustering Requirements

The following requirements must be met in order to create a Hyper-V cluster:

- At least two nodes (one active Hyper-V host and one for failover).

- Sufficient CPU power. You must be using exclusively Intel or AMD x64 processors (one or the other – not a mix of both). They should all have hardware virtualization support (which must be also enabled in BIOS) and hardware-based Data Execution Prevention (DEP). Processors must come from the same families with the same instruction sets. Ideally, the processors would be identical. This is important for providing VM migration features.
- At least 1Gigabit network. You should have separate networks for different types of traffic (Hyper-V host management, cluster communications, VM migration network, storage network, etc. – *see Figure 34*) and at least 3 physical network adapters on each host (4 is preferred). Networks can be isolated physically or logically with VLANs.
  - The *Hyper-V host management network* is used for the hosts' operating systems to communicate with Active Directory as well as infrastructure functionality.
  - The *Cluster network* is used for communication between nodes, cluster heartbeats, and Cluster Shared Volumes redirection.
  - The *Live migration network* is used for live migration of virtual machines between hosts.
  - A *Storage network* is used for connecting Hyper-V hosts with shared storage and providing storage traffic such as iSCSI (Internet Small Computer System Interface) or SMB 3.0 (Server Message Block) traffic. A **redundant network connection** is strongly recommended for these purposes (*see Figure 33*).

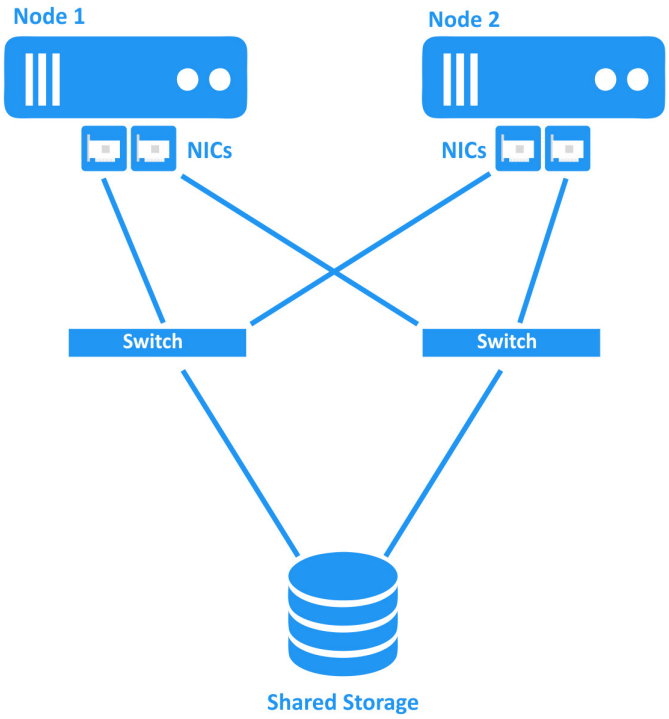

Figure 33. A redundant connection scheme for nodes and shared storage is recommended.

- Shared storage. A failover cluster requires shared storage that is connected to hosts via iSCSI, fibre channel, or SMB 3.0 protocol. Storage media that support Fibre Channel and iSCSI protocols are **block-based** storage systems. **SMB (CIFS)** is a common file-level protocol used by Windows systems. Shared storage that uses SMB 3.0 is classified as **file-level** storage. Using a fibre channel requires a dedicated **Host Bus Adapter (HBA)** card, while with iSCSI, you can use standard Ethernet **Network Interface Controllers (NICs).**

  It is recommended that you configure *Cluster Shared Volumes* (CSVs) for shared storage in the cluster. If there are no CSVs configured in the cluster, then only one node can access a LUN at the same time. In this case, you would need multiple LUNs to perform VM migration. When you have CSVs configured, multiple nodes can access a LUN simultaneously, which greatly simplifies storage access and makes using shared storage more flexible as well as improving reliability. CSVs must be formatted as NTFS partitions.
- Active Directory Domain controller is required for clusters based on Windows Server 2012 R2 and optional for clusters based on Windows Server 2016. A domain controller must not be installed on Hyper-V hosts.

Operating systems and licensing issues

The OS installed on the cluster nodes can be either Hyper-V Server or Windows Server with the Hyper-V role enabled and configured. The Hyper-V Server OS is free and has a small footprint. However, all Windows-based VMs running on Hyper-V Server OS must be licensed. Using Windows Server OS provides some benefits in licensing VMs that have Windows OS. If you use Windows Server Datacenter Edition, you can run an unlimited number of Windows-based VMs on that host. The Enterprise edition of Windows Server includes licenses for 4 Windows-based VMs and the Standard edition includes licenses for 2 Windows-based VMs. Windows Server 2012 uses a per-CPU licensing model while Windows Server 2016 uses a per-core licensing model.

Note:

When it comes to clustering, there are some licensing differences compared to the licensing of standalone Hyper-V hosts. Review Microsoft's licensing policy carefully to ensure you are buying a valid and appropriate license for your Hyper-V cluster.

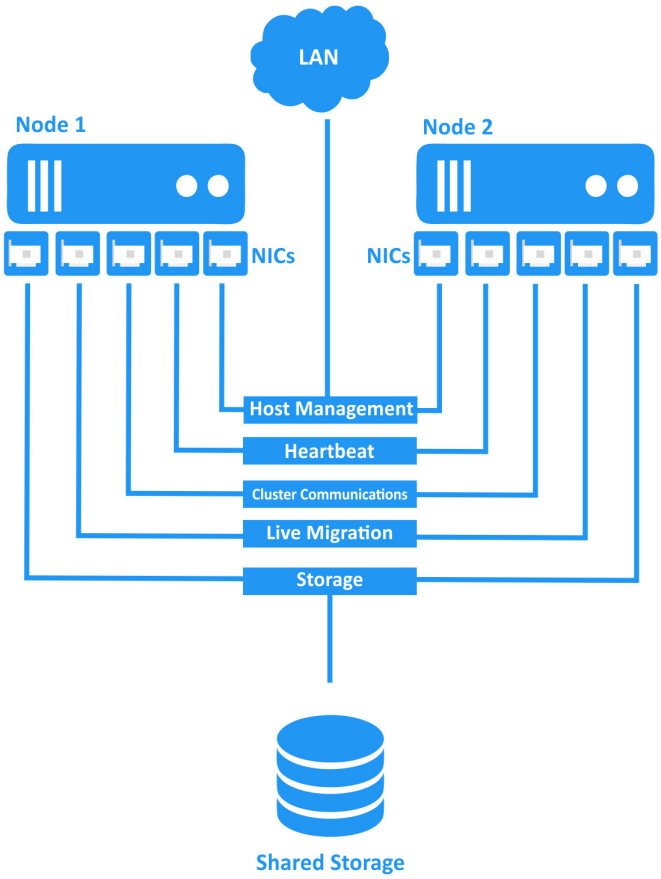

Figure 34. Example of cluster networking.

How to Deploy a Failover Cluster (Comprehensive Walkthrough)

Installing the Hyper-V role

First, the Hyper-V server role must be installed. In order to begin this installation, click **Start** > **Server Manager** *(see Figure 35).*

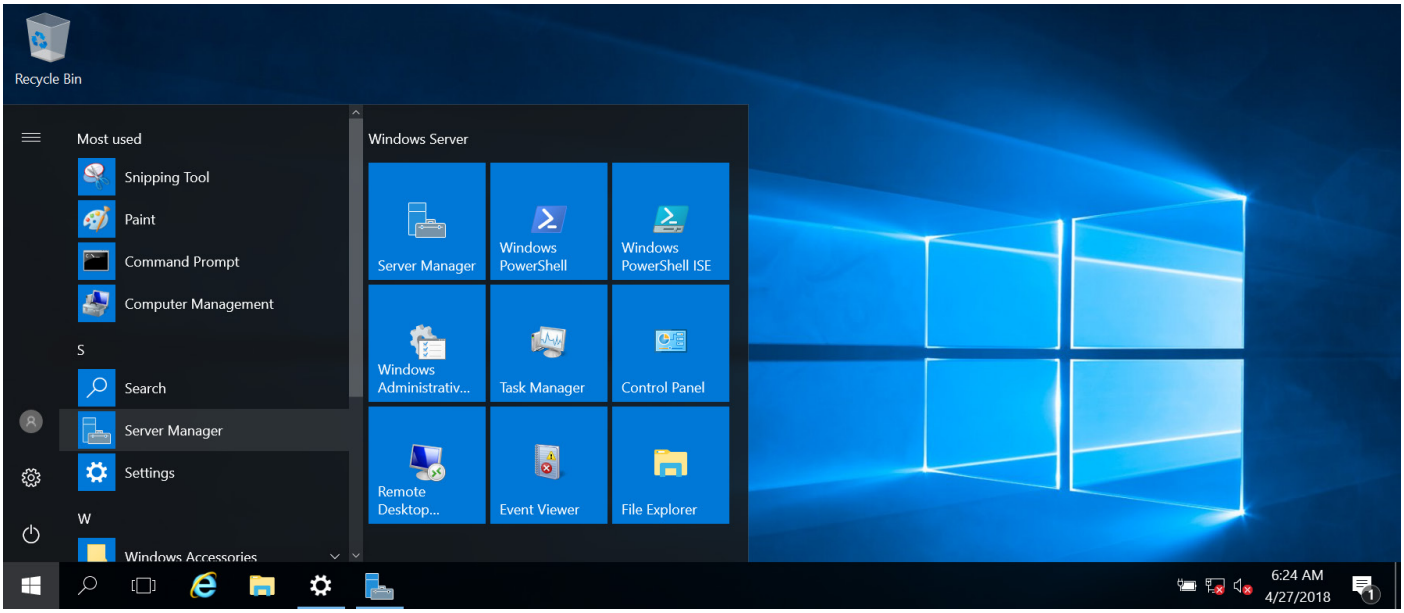

Figure 35. Example of cluster networking.

Click **Add Roles and Features** *(see Figure 36).*

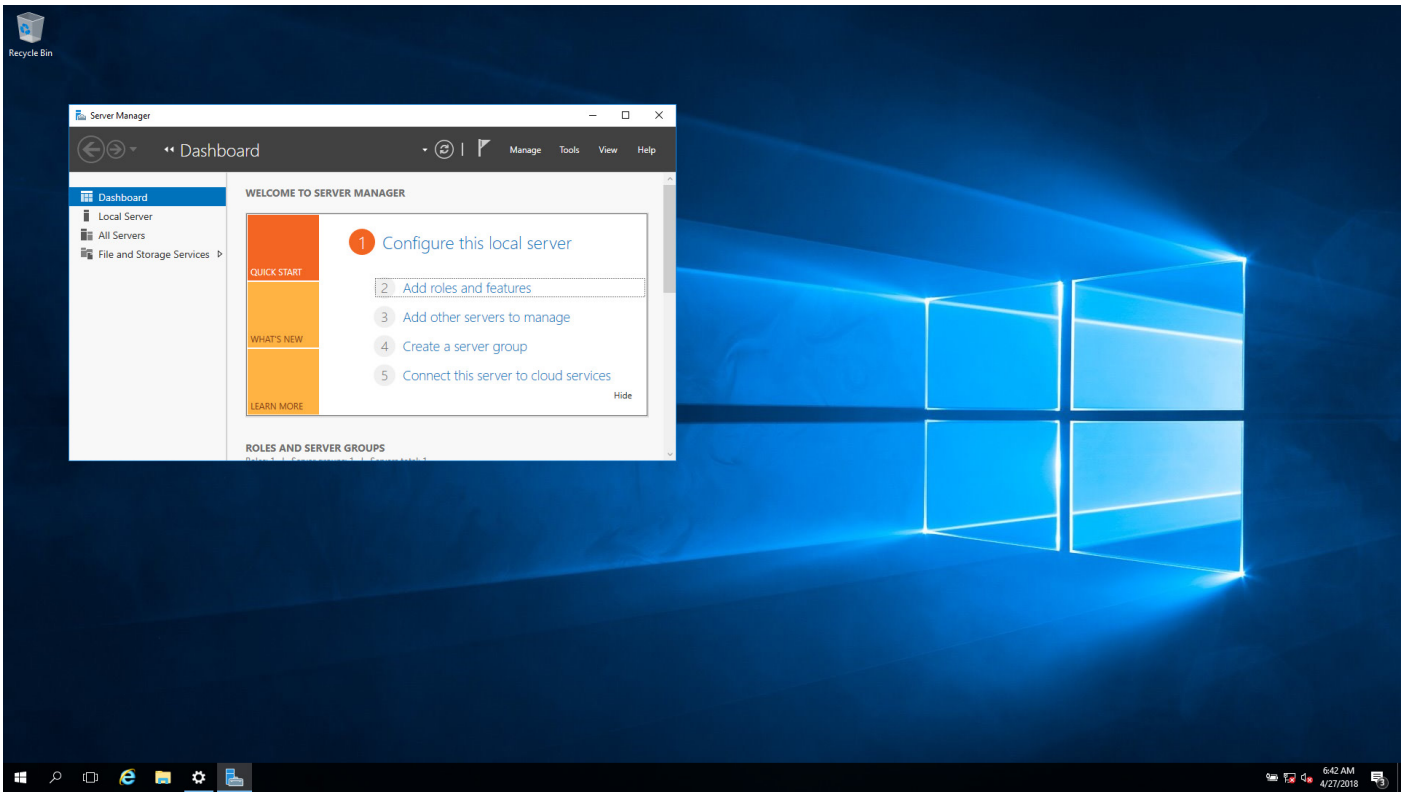

Figure 36

Select the **Role-based** installation type and click **Next** *(see Figure 37).*

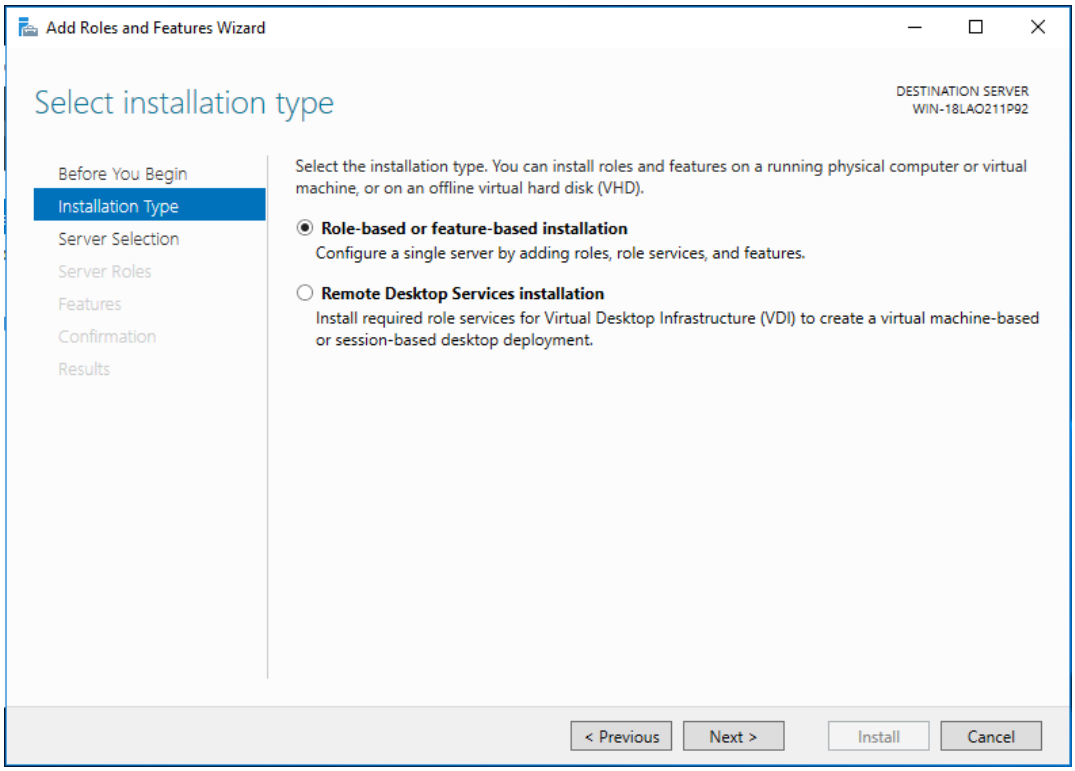

Figure 37

Select your server and click **Next** *(see Figure 38).*

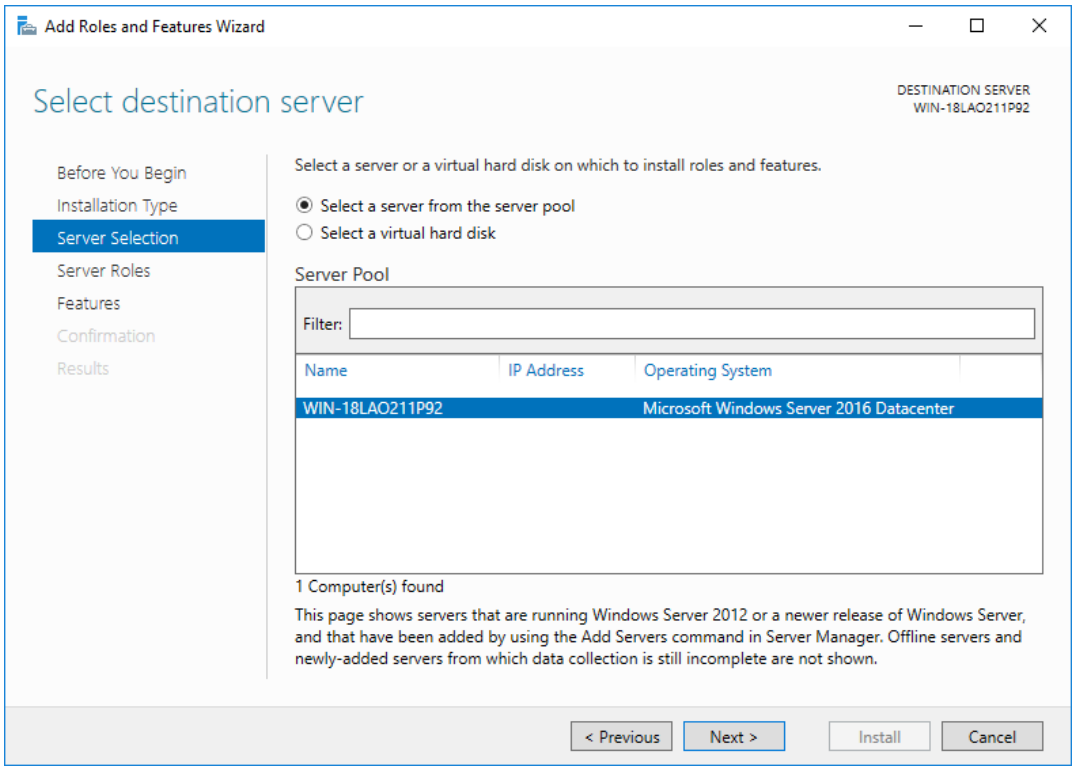

Figure 38

Select the **Hyper-V** server role and click **Next** *(see Figure 39).*

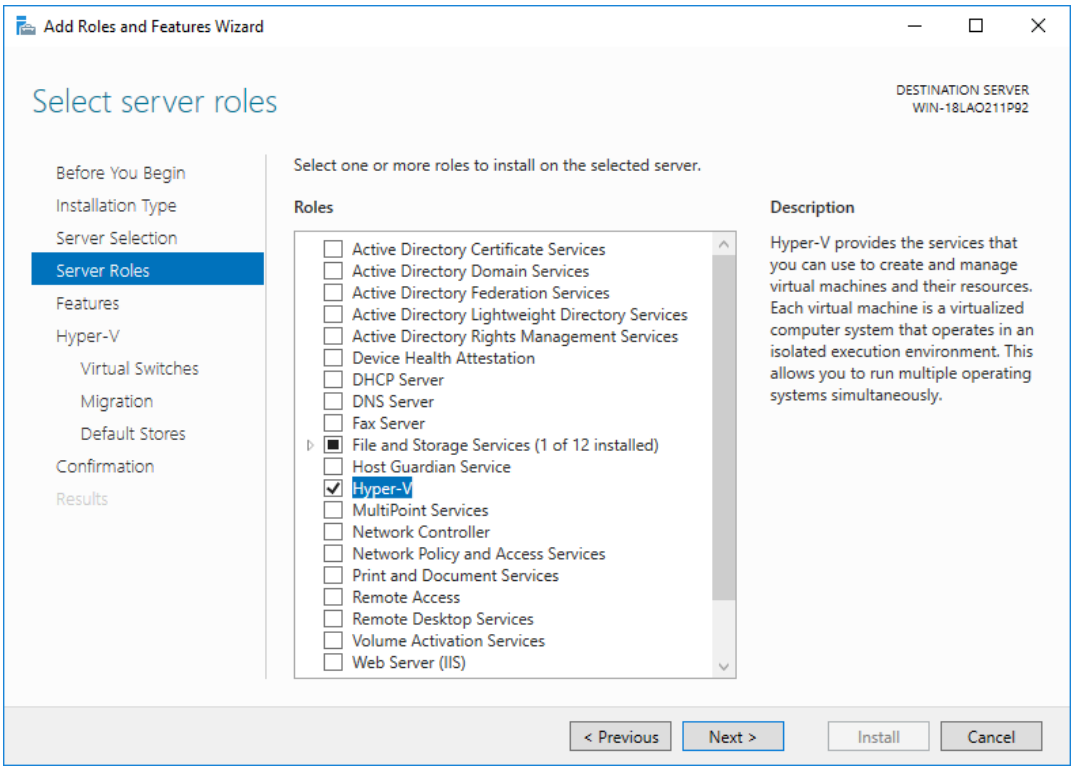

Figure 39

A confirmation message appears. Agree, and click **Add features** to install the additional features that are required for Hyper-V *(see Figure 40).*

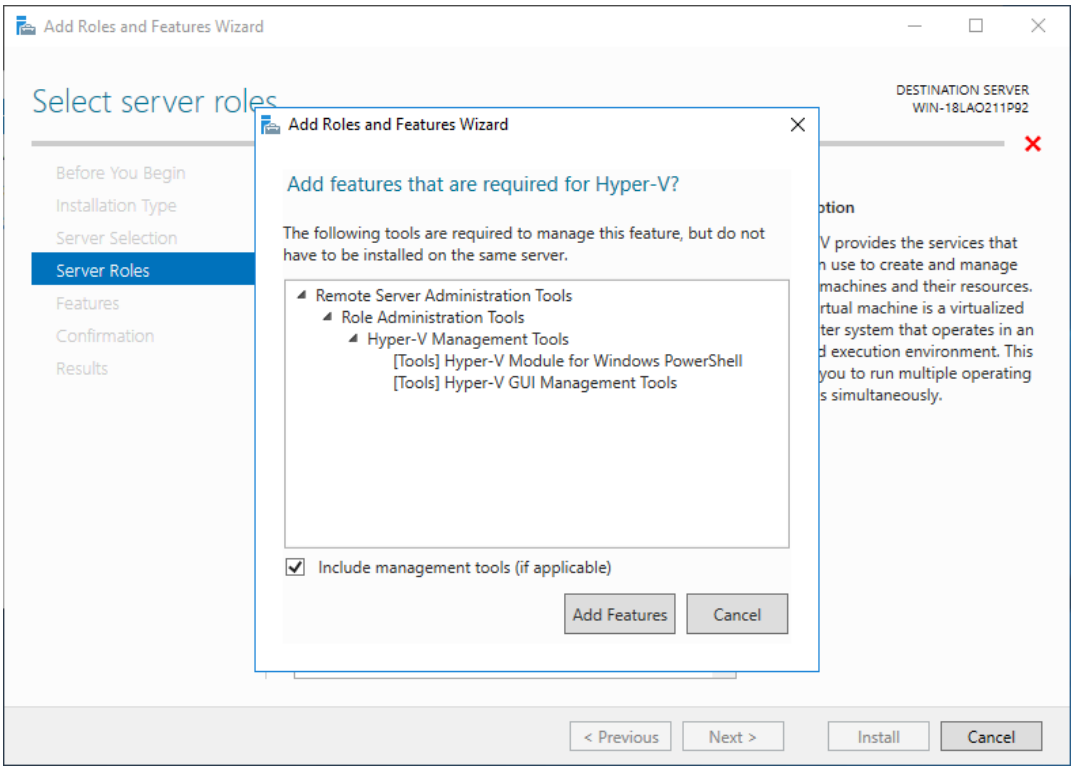

Figure 40

The Wizard that is launched prompts you to configure the Hyper-V role. Click **Next** *(see Figure 41).*

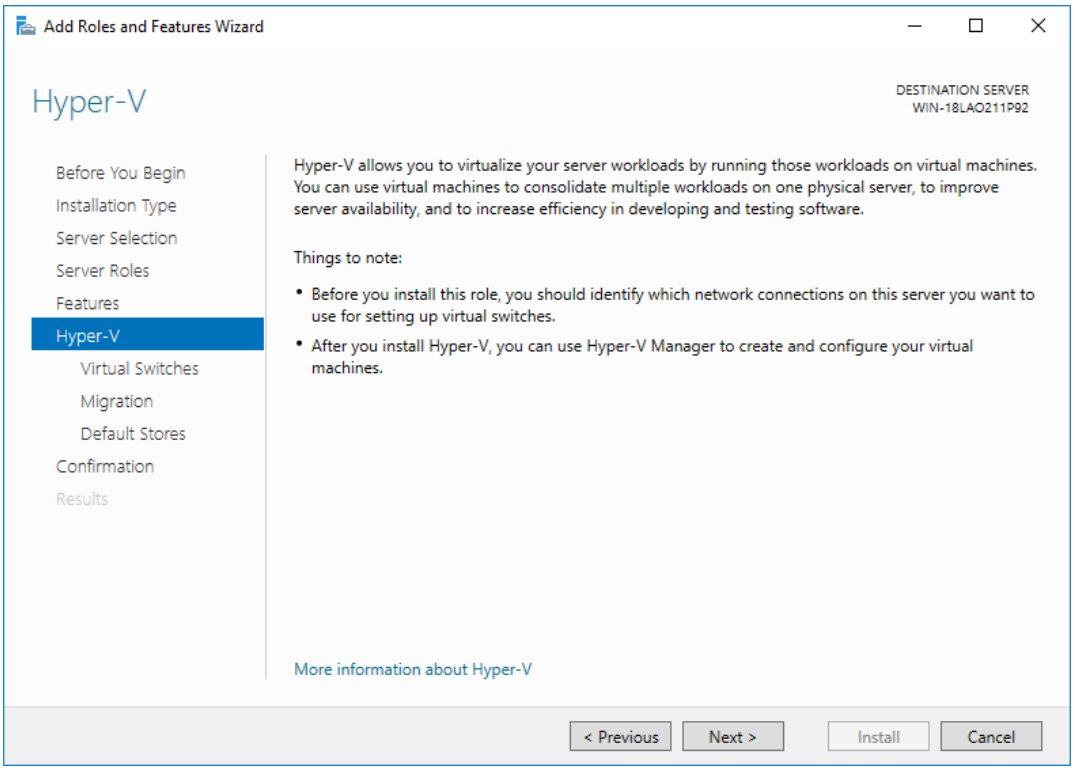

Figure 41

You can add and configure a virtual switch later, after installing the Hyper-V role. Click **Next** *(see Figure 42).*

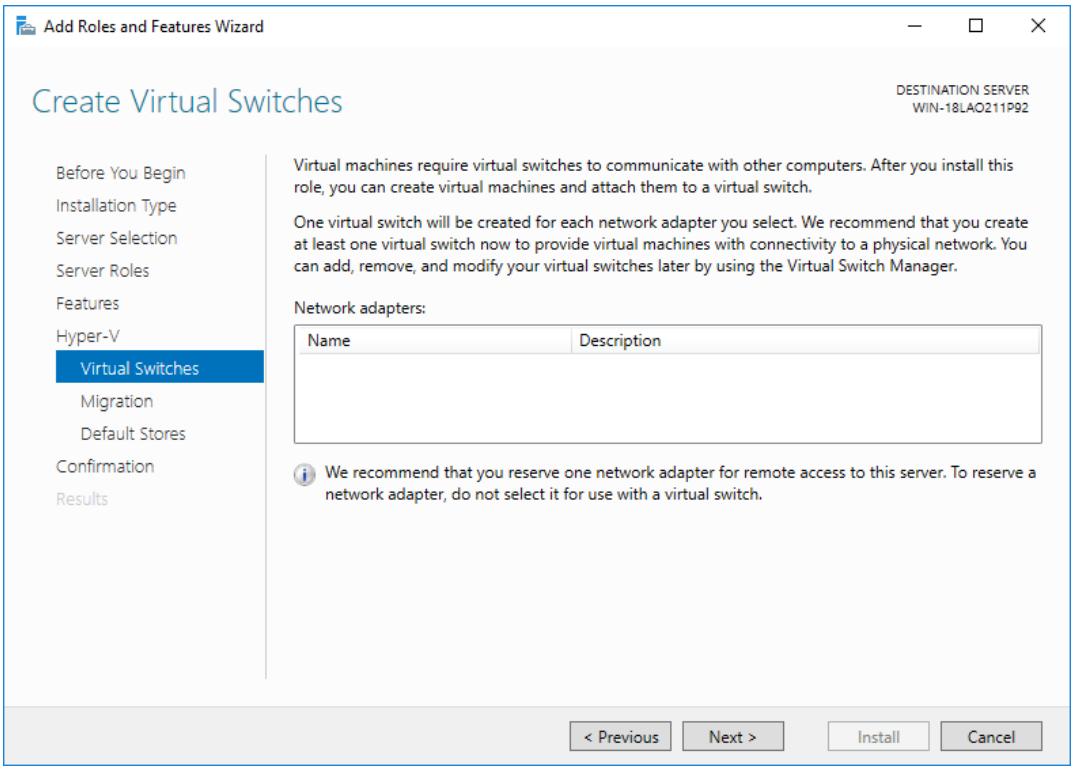

Figure 42

Live migration should be configured after your cluster has been created. Uncheck the box and click **Next** *(see Figure 43).*

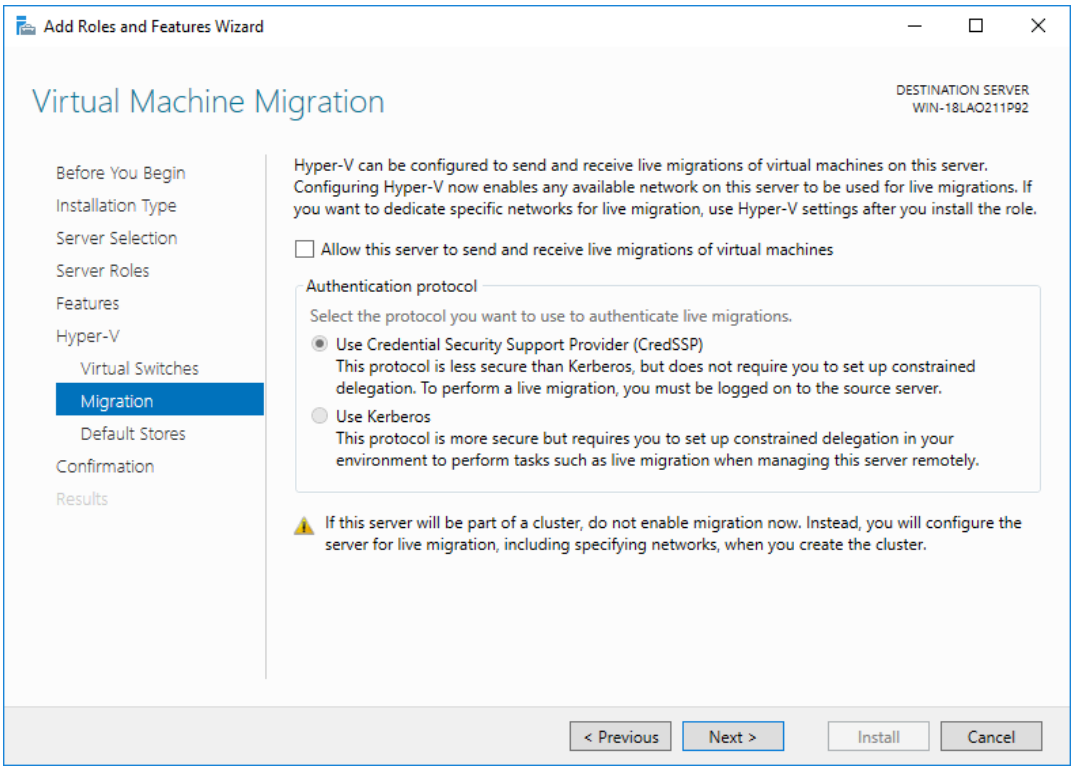

Figure 43

Select a default location for the virtual disk and VM configuration files, then click **Next**. You can select a directory on a separate partition, e.g., D:\VM *(see Figure 44).* Later on, you can select a directory on a shared storage volume as the default location for VMs.

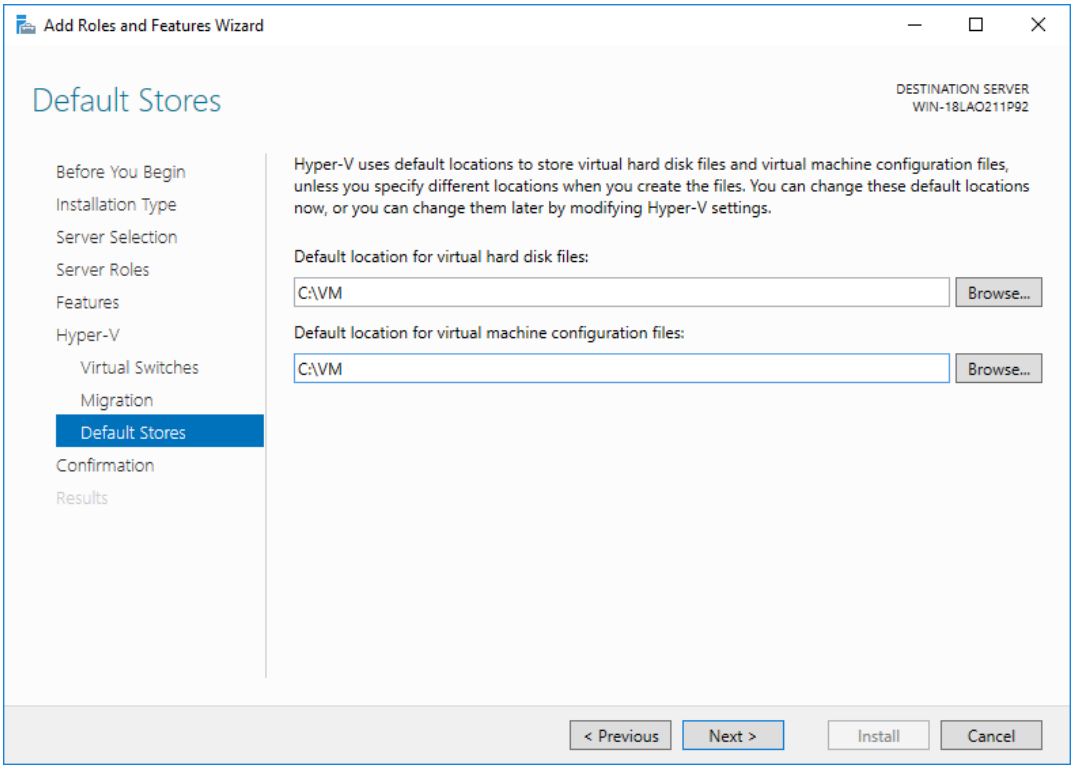

Figure 44

Confirm the selected settings and click **Install** *(see Figure 45).*

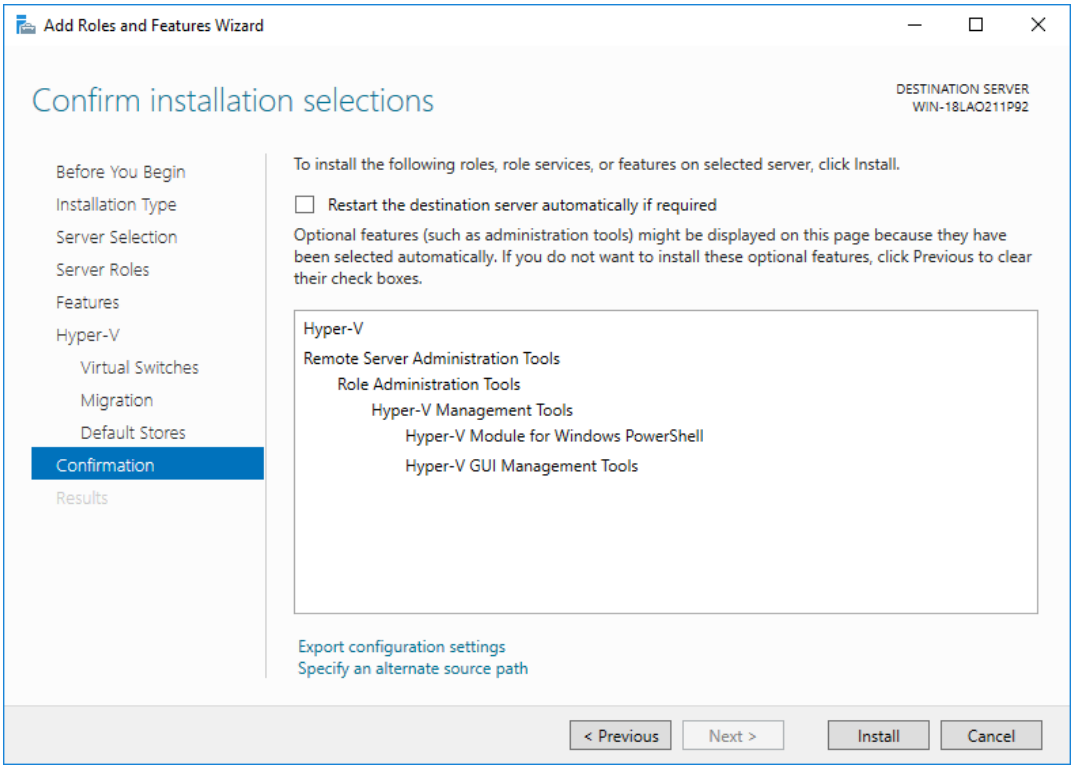

Figure 45

Wait while the features are installed *(see Figure 46)*. A restart is required after the installation.

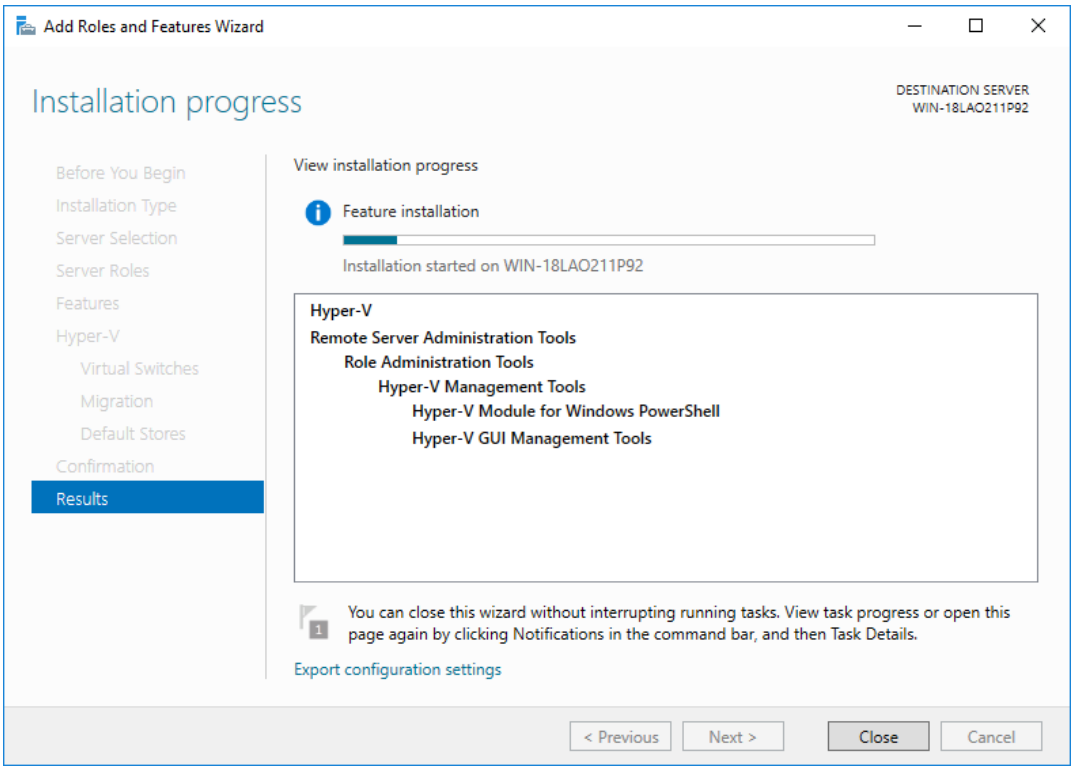

Figure 46

Now that the Hyper-V server role is installed, you can configure your Hyper-V features and options. Go to the Hyper-V Manager. Click **Start > Server Manager.** Select your Hyper-V server, right-click the server name, and select **Hyper-V Manager** in the context menu *(see Figure 47)*.

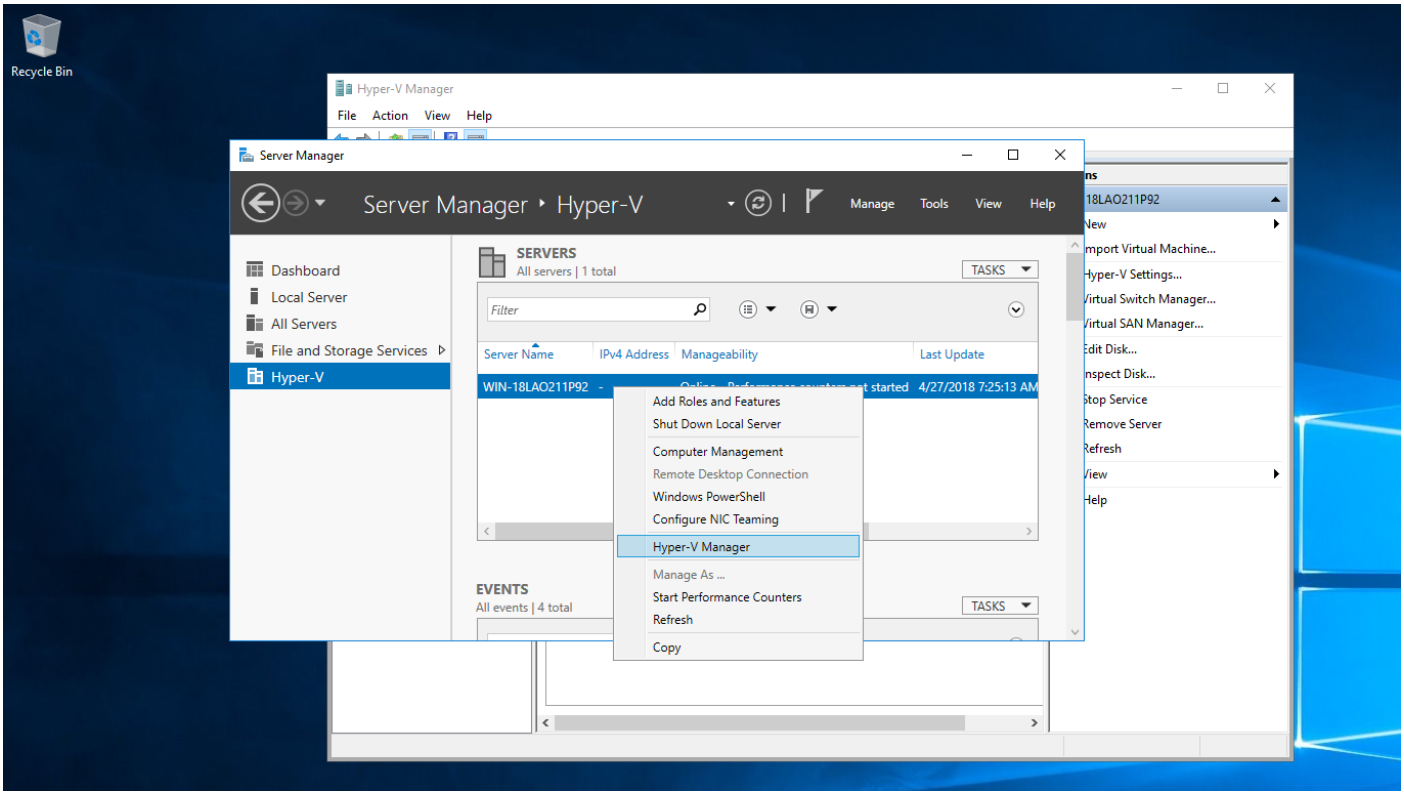

Figure 47

Network Configuration

Configuring virtual switches

A virtual switch is a software program that emulates a switch as a layer-2 network device. A VM's virtual adapters are connected to virtual switches. Virtual switches are used for connecting VMs with each other as well as with other network devices.

There are three connection types for virtual switches in Hyper-V – External, Internal, and Private.

- **External.** The VMs using these virtual adapters can connect with any network devices in the physical network, including with the Hyper-V host on which they are residing and with other VMs running on the Hyper-V host.
- **Internal.** Virtual machines running on the current Hyper-V host can communicate with each other and with the host operating system.
- **Private.** The VMs located on a particular Hyper-V host can communicate only between each other; they cannot communicate with the host operating system or other network devices. They are isolated in a private network. This setup can be useful for testing purposes.

Let's configure a virtual switch. In the Hyper-V Manager, click **Action > Virtual Switch Manager** *(see Figure 48)*.

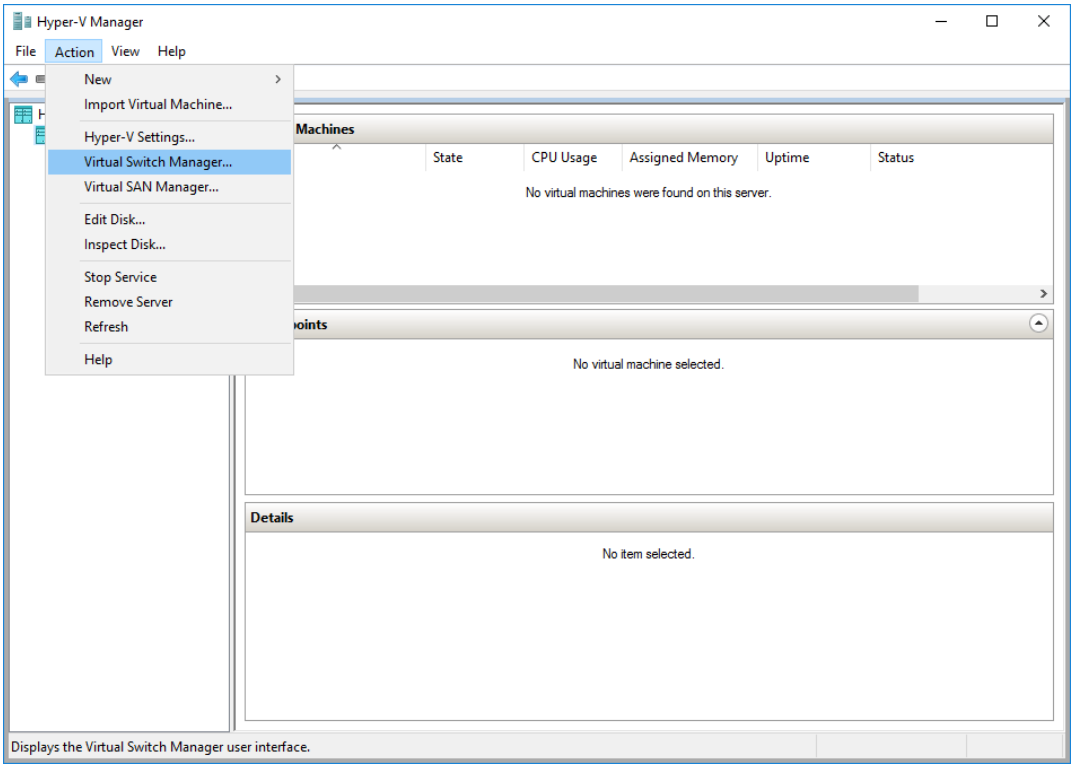

Figure 48

Select the **External** virtual switch type and click the **Create Virtual Switch** button *(see Figure 49)*.

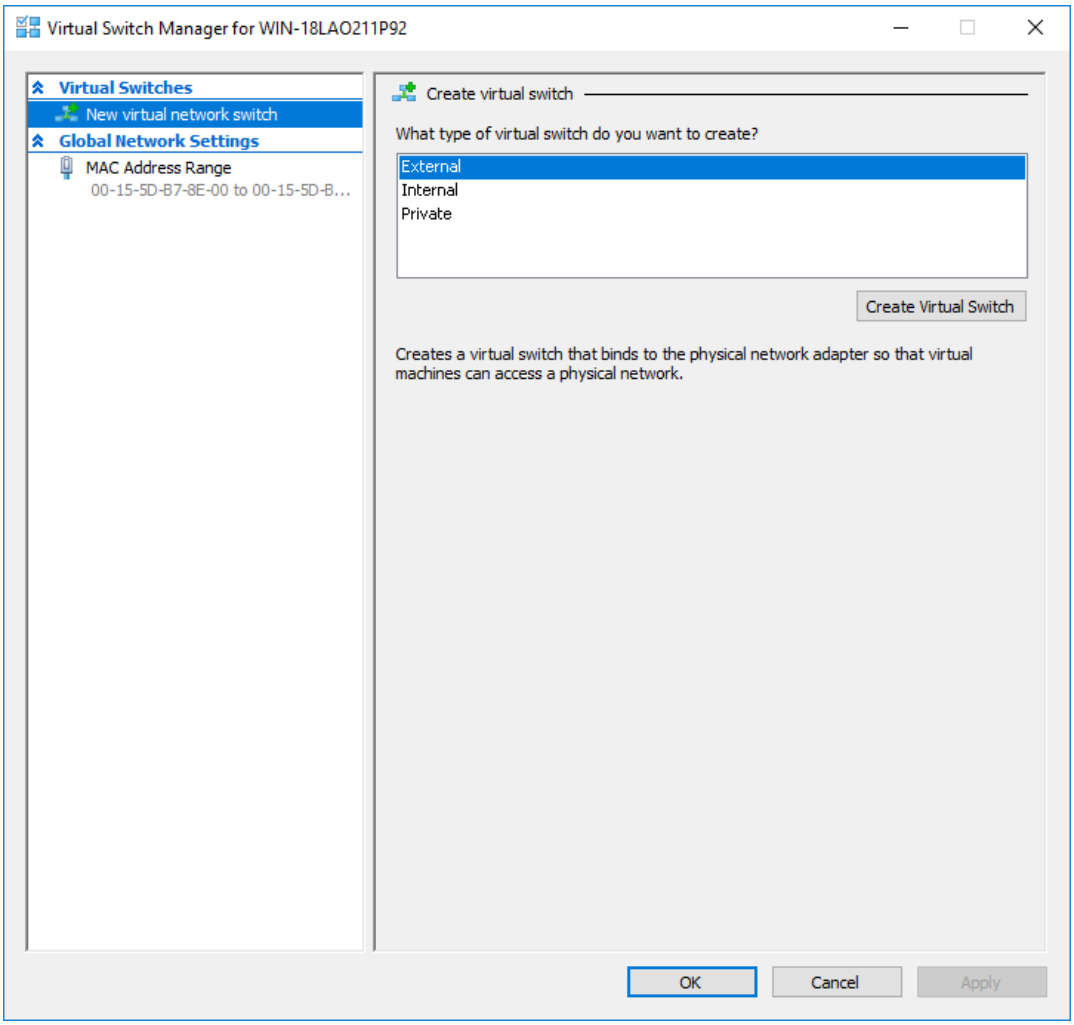

Figure 49

Enter the name of the virtual switch. Select **External network** as the connection type and specify the network adapter for this virtual switch. Tick the checkbox under *VLAN ID* if you want to separate multiple networks logically *(see Figure 50)*.

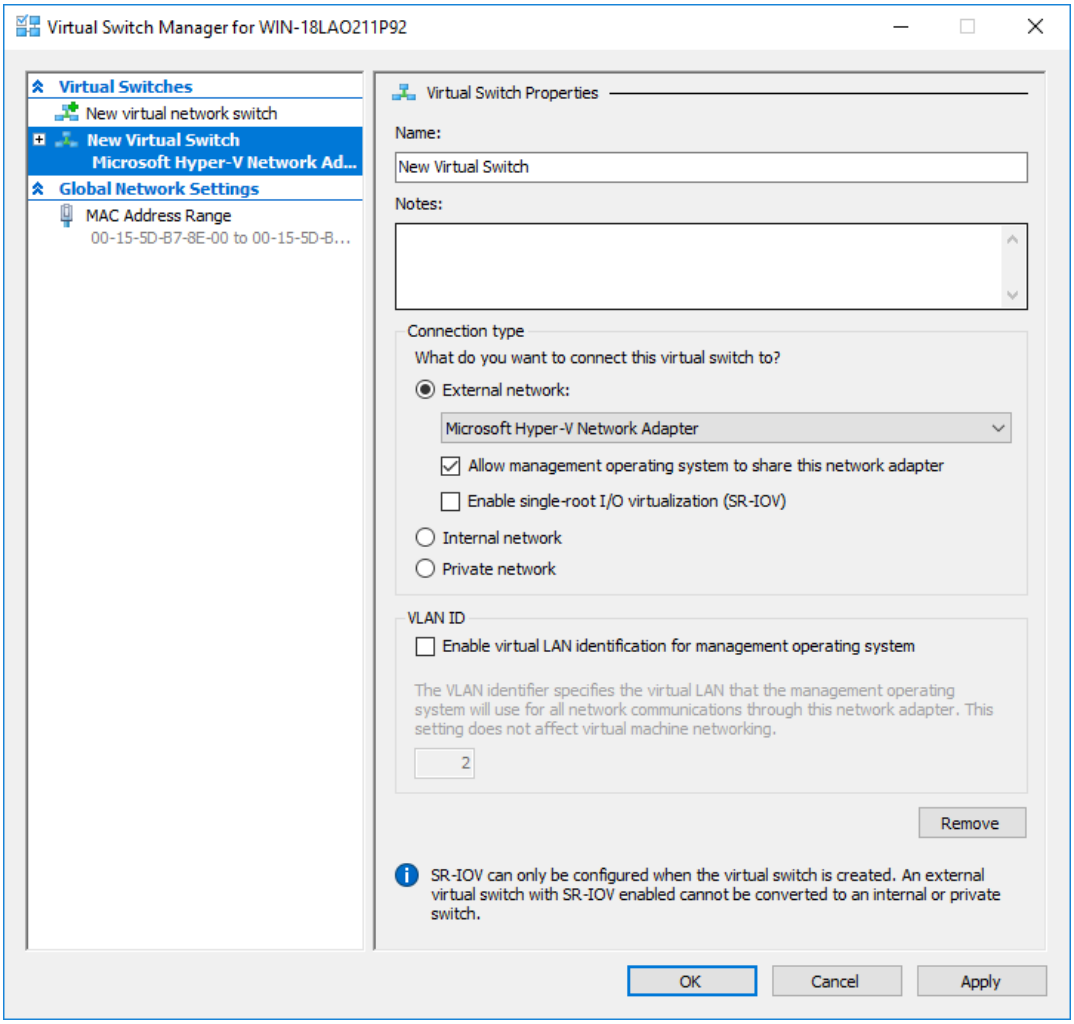

Figure 50

A confirmation message appears. Click **Yes** when you have read the text and are ready to proceed *(see Figure 51).*

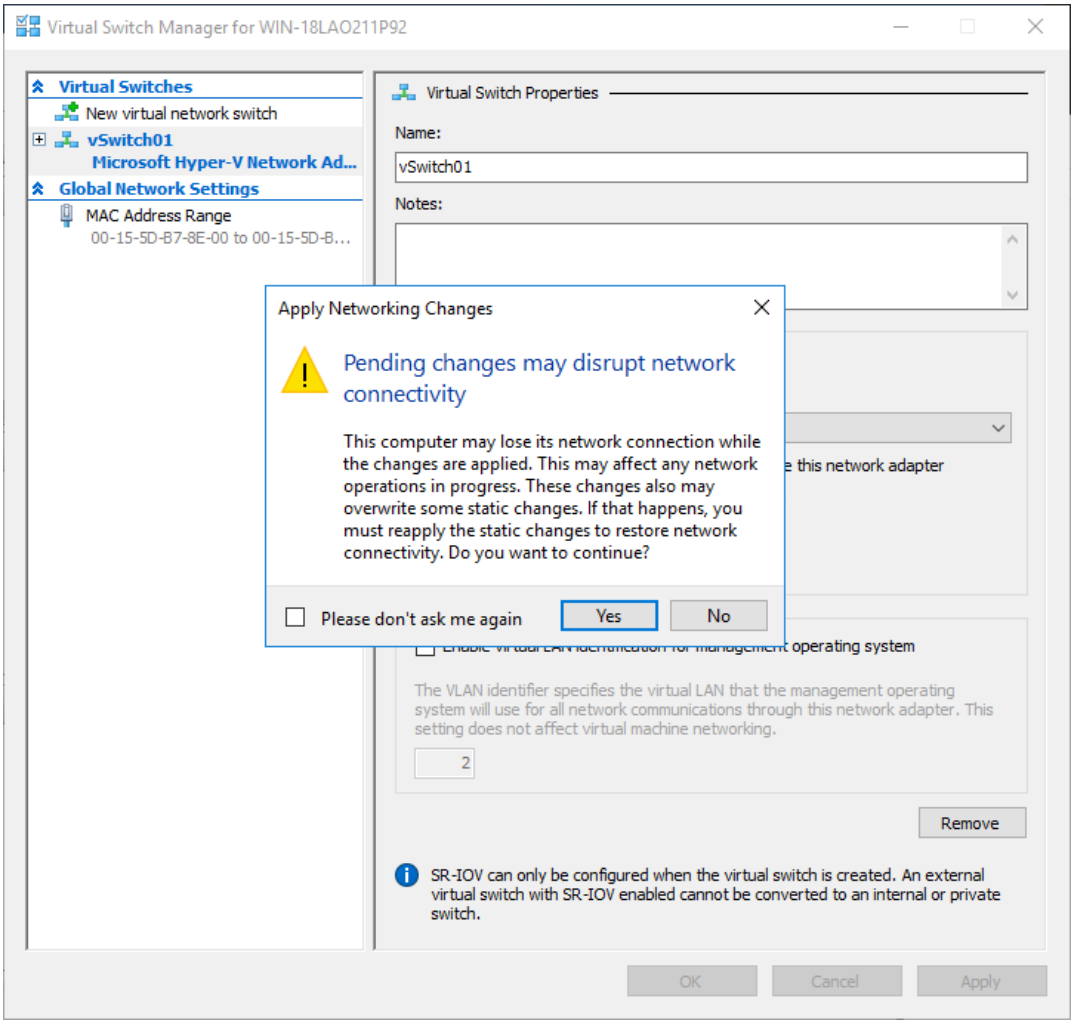

Figure 51

NIC Teaming

NIC Teaming is a feature that allows multiple network adapters to be combined in a group for improved speed and greater redundancy. A NIC "team" with multiple active adapters provides greater speed. A NIC team with active adapters and standby adapters lets you perform failover smoothly.

You can configure NIC teaming on the Hyper-V hosts that you are going to include in the Failover cluster. Go to the **Server Manager**, select your Hyper-V server, and right-click your server name. From the context menu, select the **Configure NIC Teaming** option *(see Figure 52).*

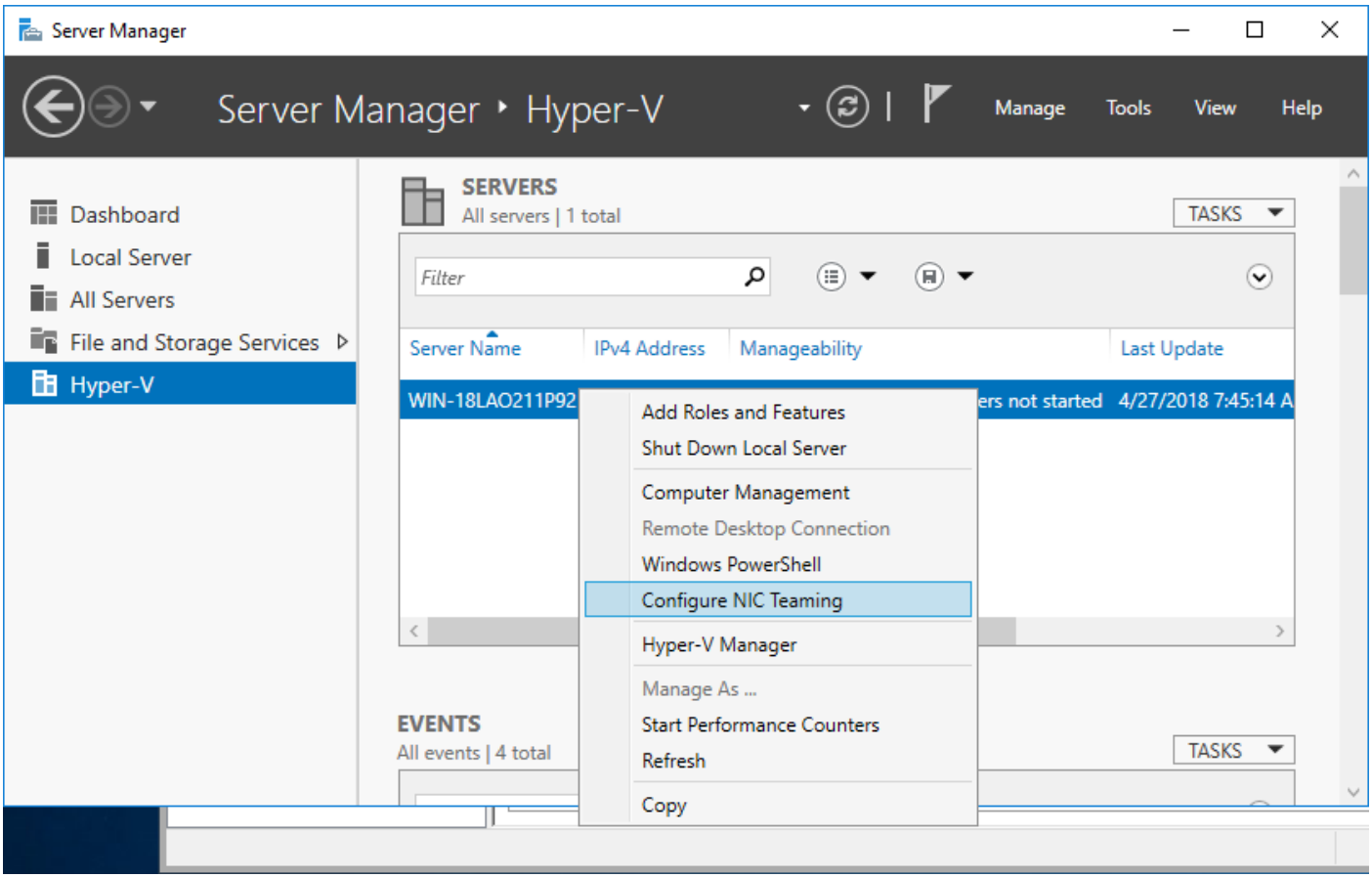

Figure 52

In the **Adapters and Interfaces** pane, select at least two network adapters by right-clicking them. From the context menu, select the **Add to New Team** option *(see Figure 53).*

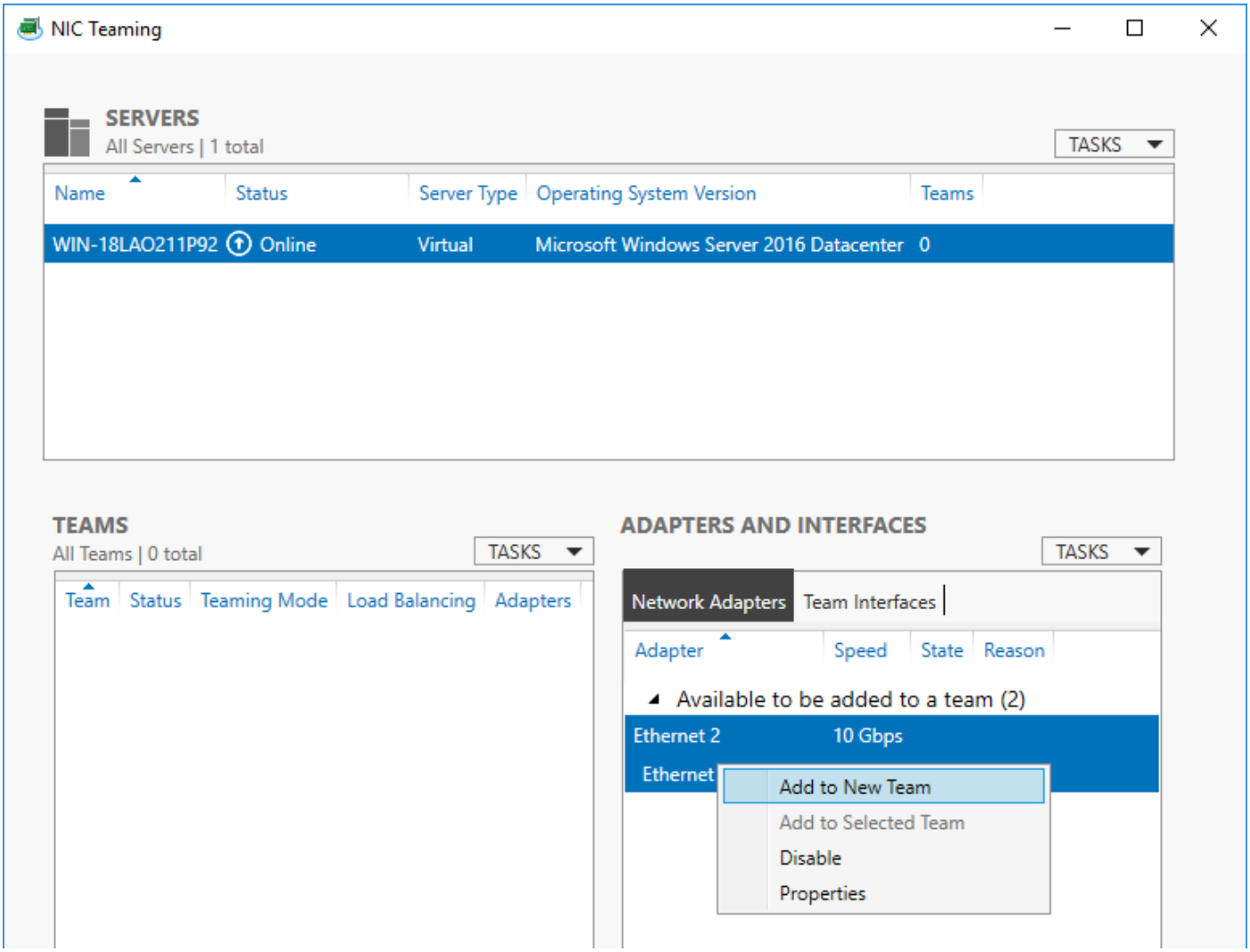

Figure 53

Specify a team name, tick the checkboxes next to the member adapters you want to include, select the **Switch Independent** teaming mode, and select the **all adapters Active** option *(see Figure 54).*

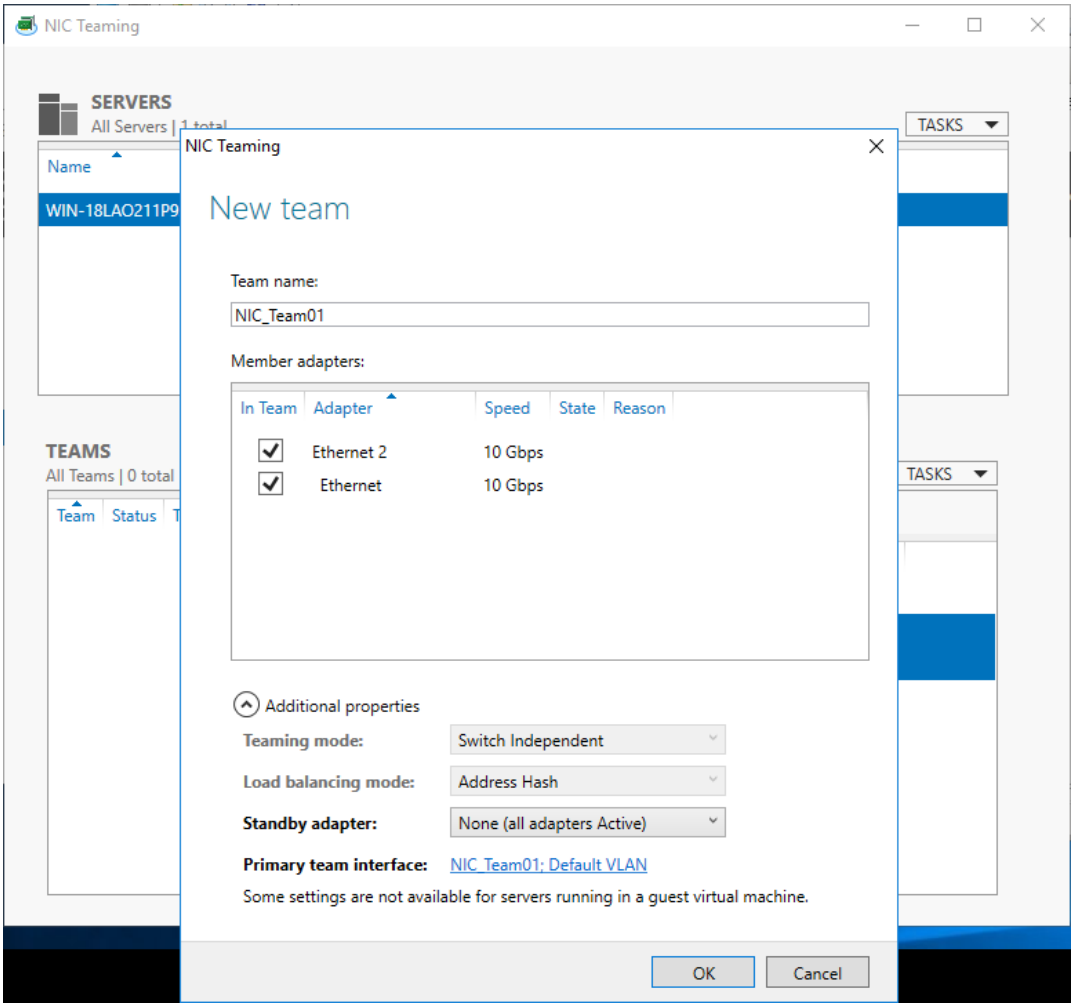

Figure 54

Teaming mode offers the following two options:

- *Switch Independent.* In this configuration, the switches to which teamed network adapters are connected don't know anything about the NIC team. Thus, it cannot distribute the traffic to adapters within the NIC team. The NIC team distributes the inbound network traffic among its members.
- *Switch Dependent.* In this configuration, the switches to which teamed network adapters are connected are aware of the NIC team. The switches determine how to distribute inbound traffic between NIC team members. All team members must be connected to the same physical switch.

Load Balancing mode has the following options:

- *Address Hash.* A hash is created based on the address components of the packet. Then this load balancing mode assigns packets with specified hash values to one of teamed adapters.

- *Hyper-V port.* Every VM running on Hyper-V is tied to a particular NIC in the team. The virtual adapter of each VM, which is connected to a virtual switch, has its own MAC address. This method can be effectively used for Hyper-V hosts running many VMs. This mode cannot be used for NIC teams created inside the VMs by using virtual adapters (use the Address Hash mode for these cases).
- *Dynamic mode.* This option gives you the advantages of both Address Hash and Hyper-V port modes. Address hashing is used for outbound traffic and Hyper-V port balancing is used for inbound traffic.

Standby adapter: None (All adapters are Active) or a specified teamed adapter.

Your NIC team is now online *(see Figure 55).*

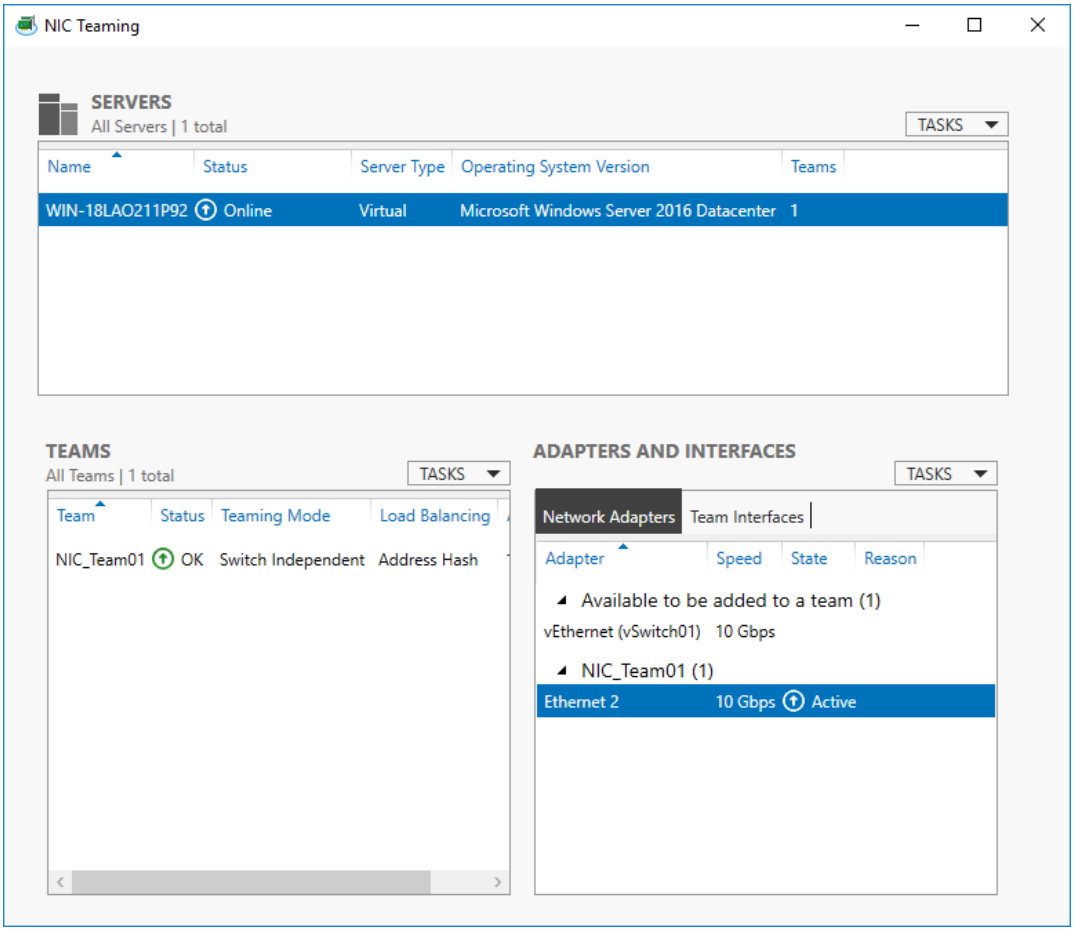

Figure 55

You can view your recently created NIC Team through the *Network Adapters* options panel *(see Figure 56).*

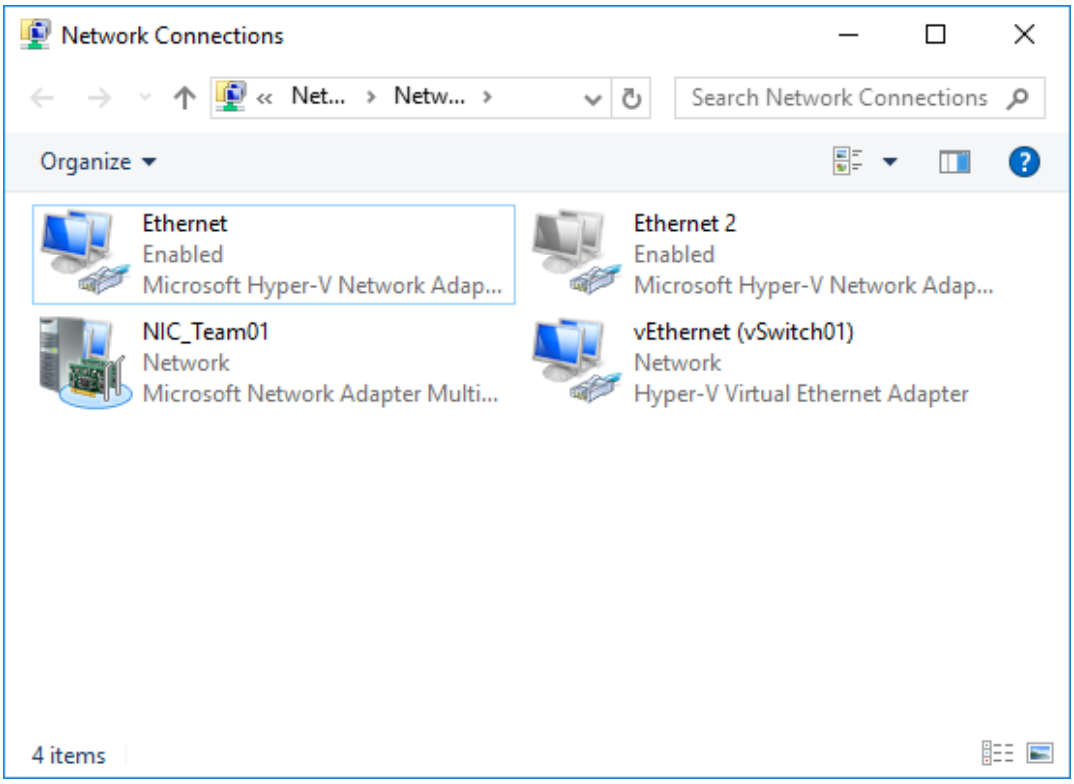

Figure 56

Repeat these steps for each network (storage, live migration, cluster communications network, etc.) and for each Hyper-V host you want add to the cluster. Assign subnets for each network (e.g., 10.10.10.0/24 for the heartbeat network, 10.10.11.0/24 for the migration network, etc.) Configure the network settings for each network interface manually and use static IP addresses *(see Figure 57)*. Do not allow network interfaces to obtain IP addresses automatically via DHCP.

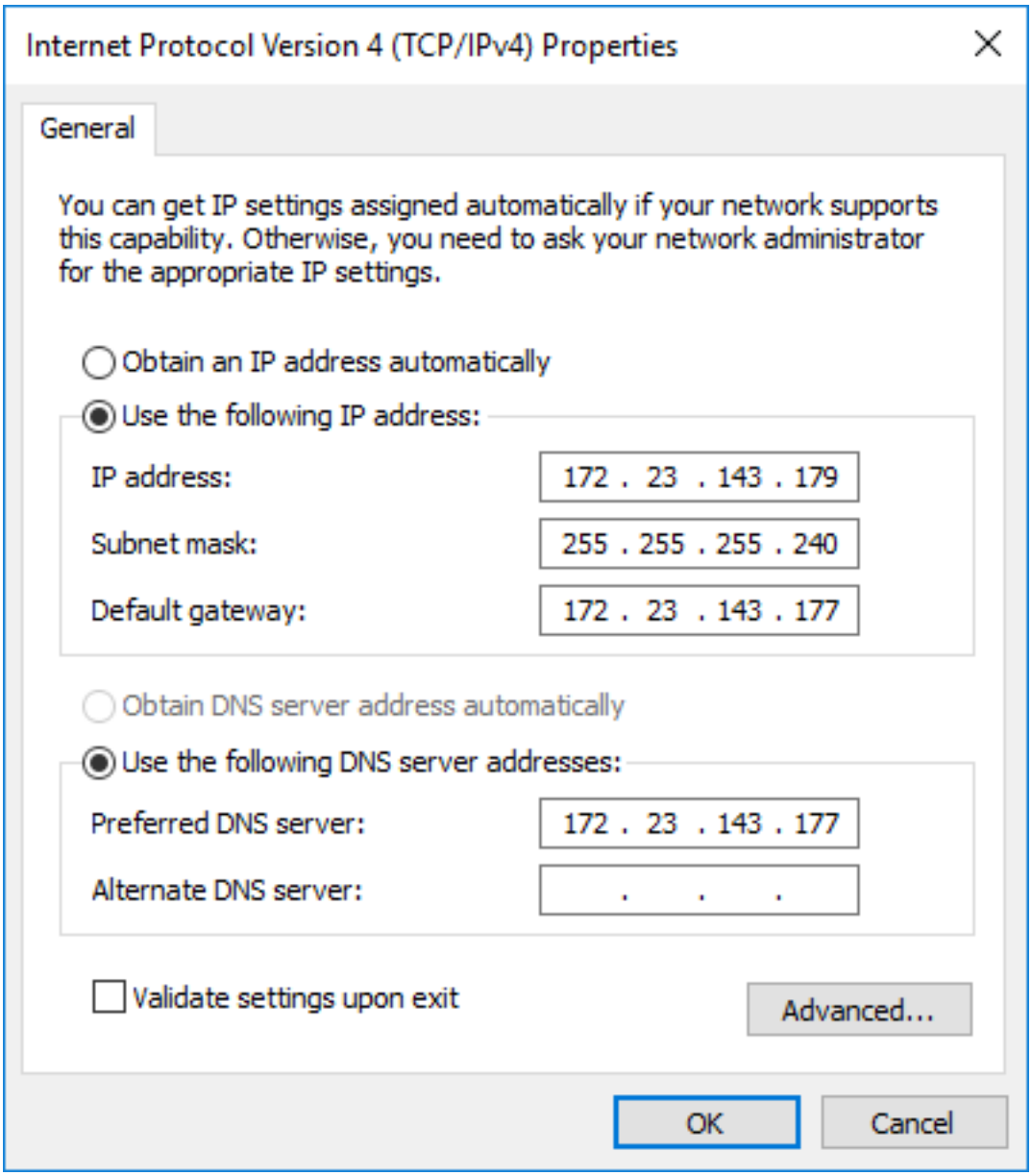

Figure 57

Shared Storage Configuration

Shared storage can be configured on enterprise-level SAN or NAS devices. It is also possible to configure a physical server for use as shared storage. If you have a dedicated server with Windows Server OS installed, you can configure shared storage on the server. Windows Server 2016 with Hyper-V supports iSCSI, Fibre Channel, and SMB v3 connections to shared storage. For the purposes of this example, iSCSI block-level shared storage is configured on Windows Server 2016.

Note:

Storage Space Direct is a new scalable technology from Microsoft that was created for shared storage and supports SMB v3 protocol. This technology allows you to create shared storage based on standard servers with local disks (HDD and SSD) that can also be used for Hyper-V clustering. The Resilient File System **(ReFS)** that is integrated into Storage Space Direct is more resistant against data corruption than the NTFS file system. Combining HDD with SSD allows Storage Space Direct to work more efficiently; it can use the faster SSD for frequently changed data and the slower HDD for rarely changed data. Storage Space Direct is classified as a **software-defined storage solution.**

First, prepare a separate disk to share via iSCSI protocol. **RAID 1** or **RAID 10** is preferred to ensure greater reliability. The example below shows a 30 GB disk being used to create the shared storage *(see Figure 58).*

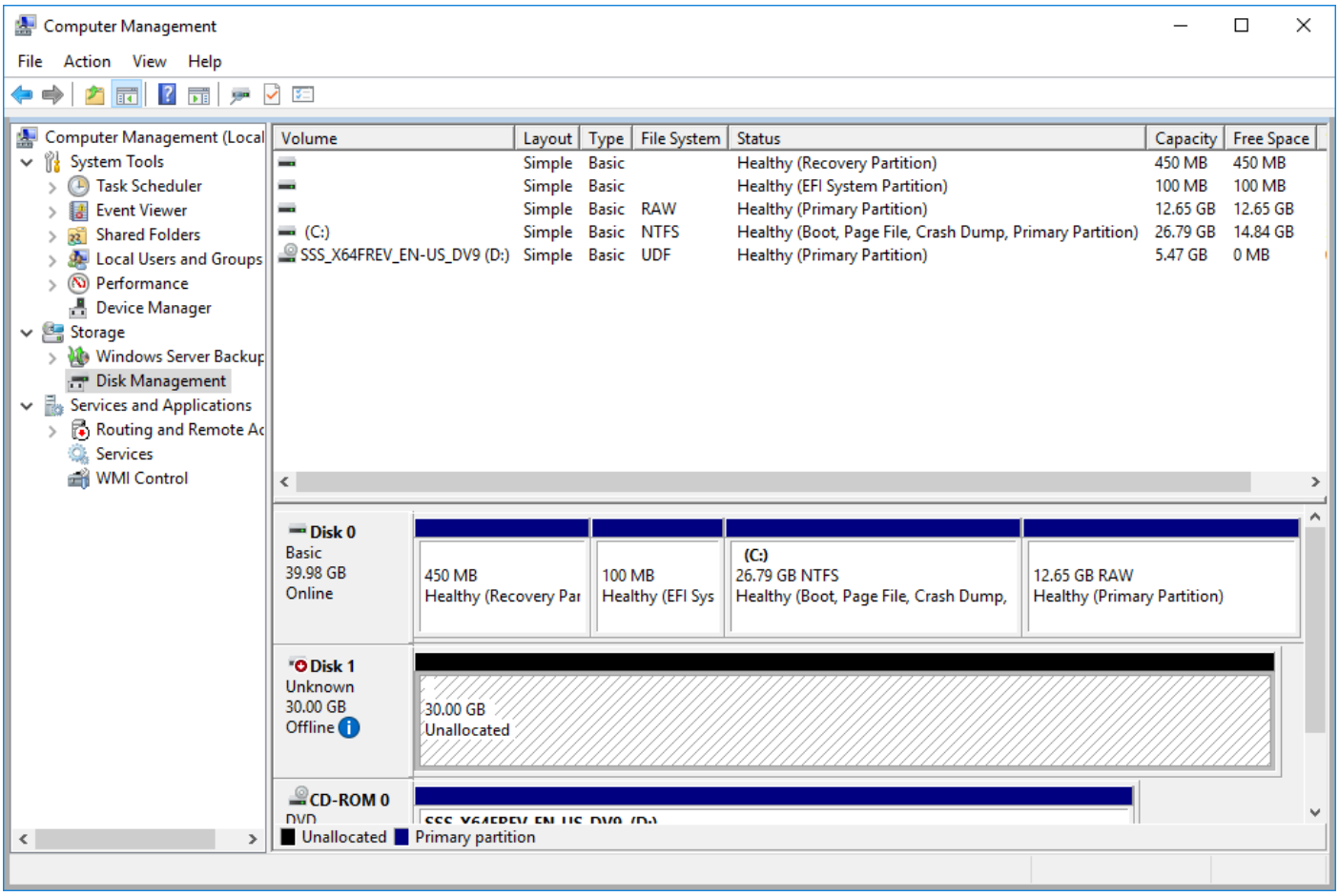

Figure 58

Install the iSCSI Target Server role

Open the Server Manager (**Start > Server Manager**). Click **Add roles and features** (similarly as described in the section above on adding the Hyper-V role). In the Wizard window, under the Server Roles section, select **iSCSI Target Server** *(see Figure 59)*. The Wizard asks you also to install a File Server role. Agree, click **Add Features**, click **Nex**t and, finally, click **Install.**

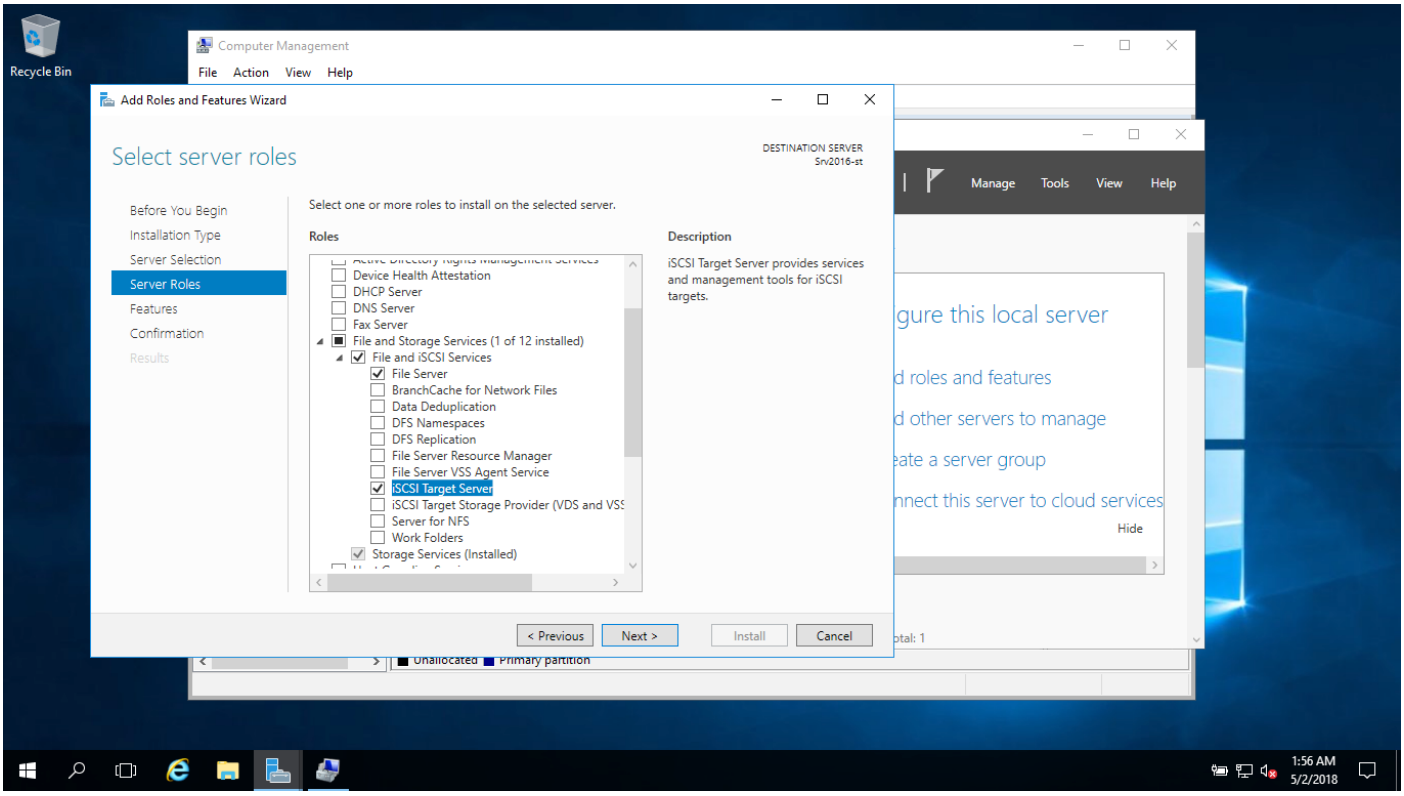

Figure 59

Wait while the server role is installed *(see Figure 60).*

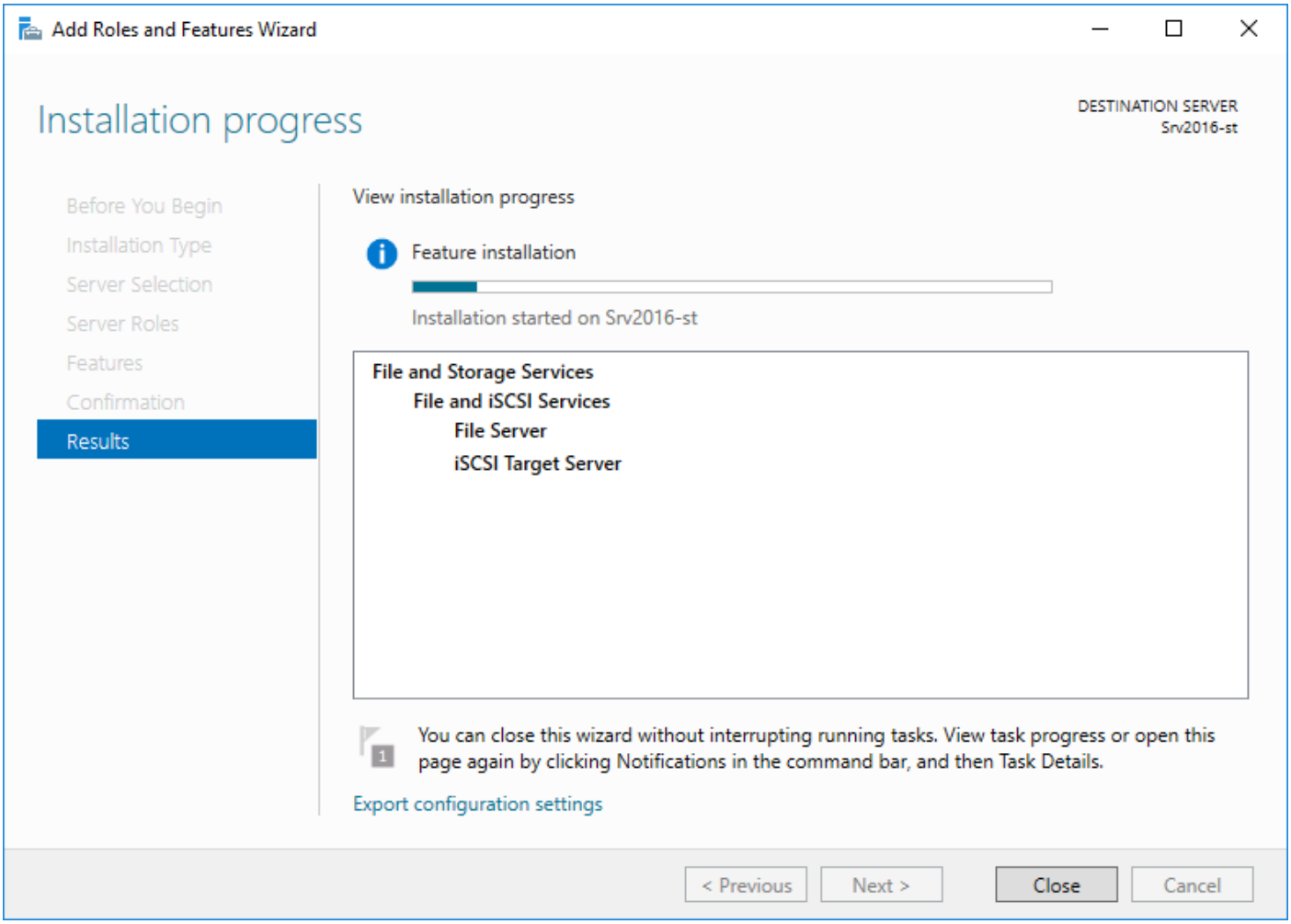

Figure 60

iSCSI Virtual Disk Setup

You can create two partitions in the disk manager – one partition for storing VMs and the second partition for the disk witness.

Once the partitions are created, go to **Server Manager > File and Storage Services**, then select the partition you created for VM storage. Right-click the partition and select **New iSCSI virtual disk** from the context menu *(see Figure 61).*

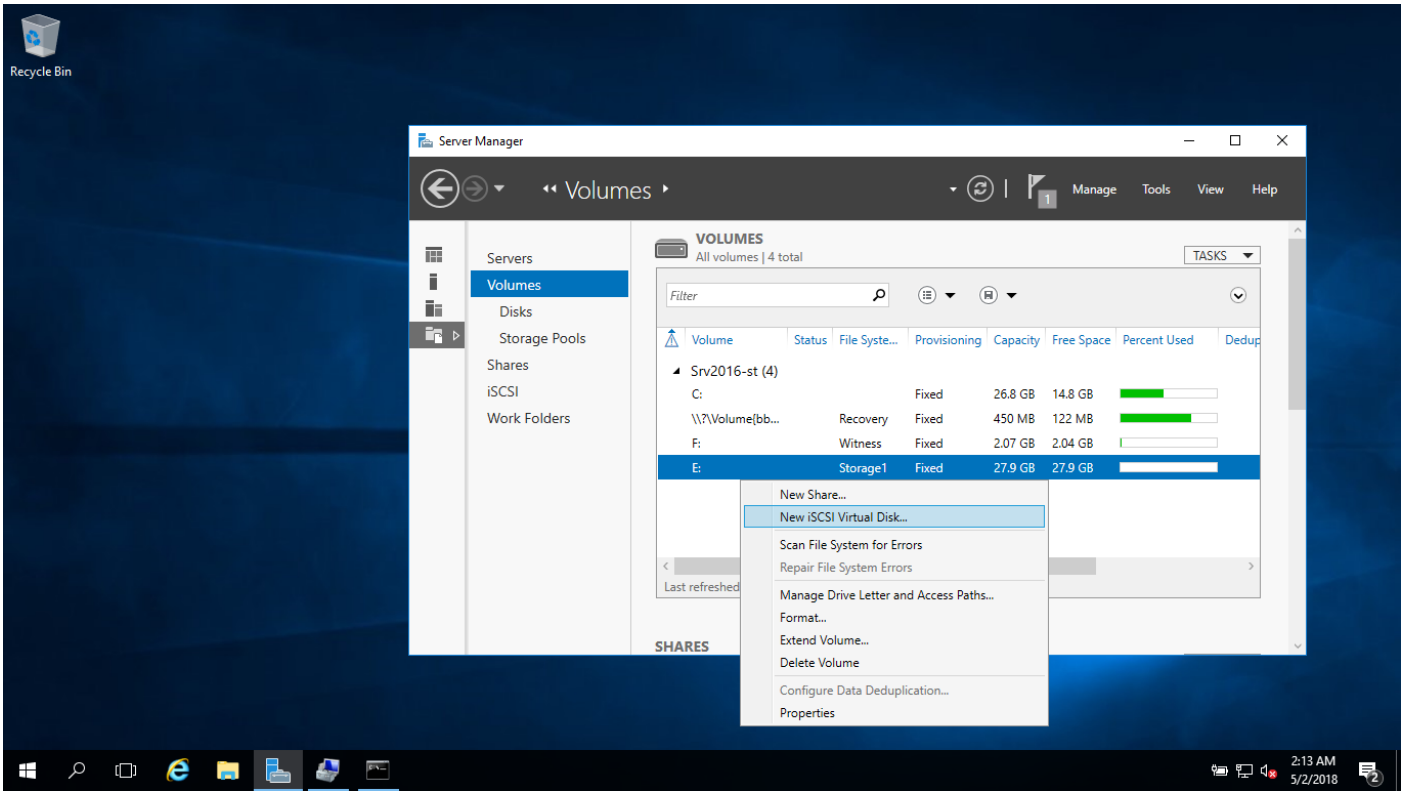

Figure 61

The *New iSCSI Virtual Disk Wizard* is launched. Select the appropriate volume and click **Next** *(see Figure 62).*

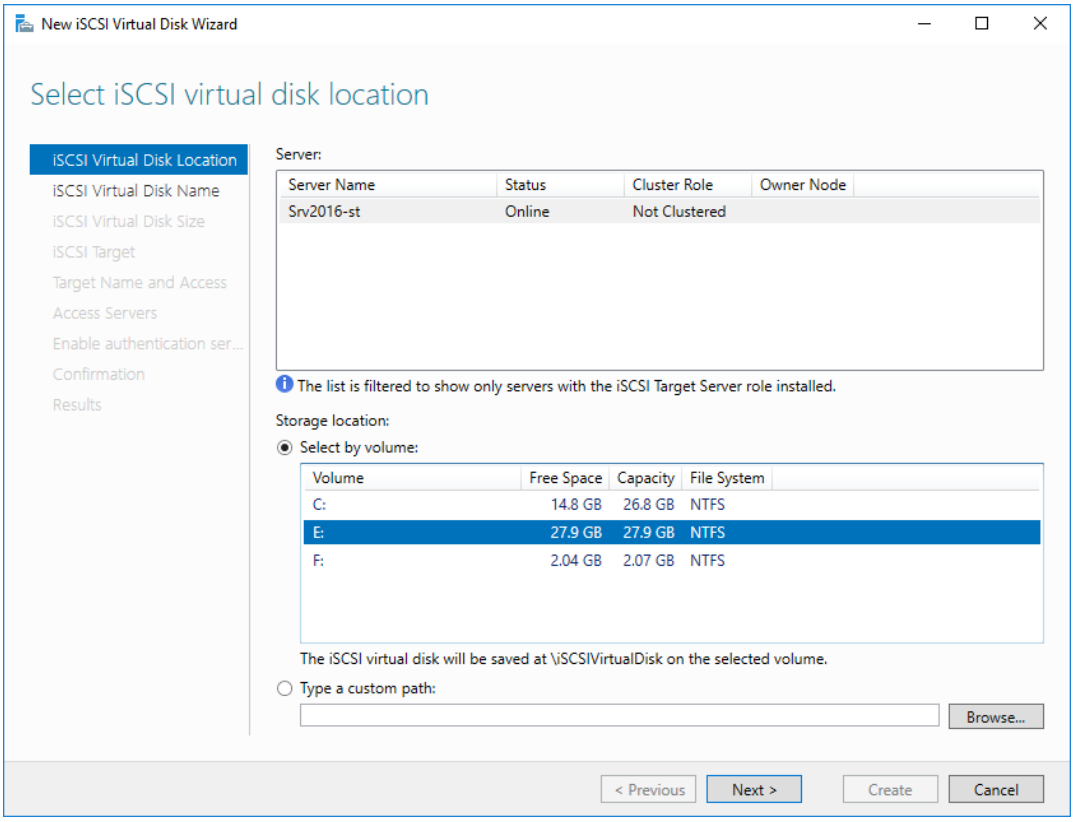

Figure 62

Type the name and description *(see Figure 63)*. Click **Next**.

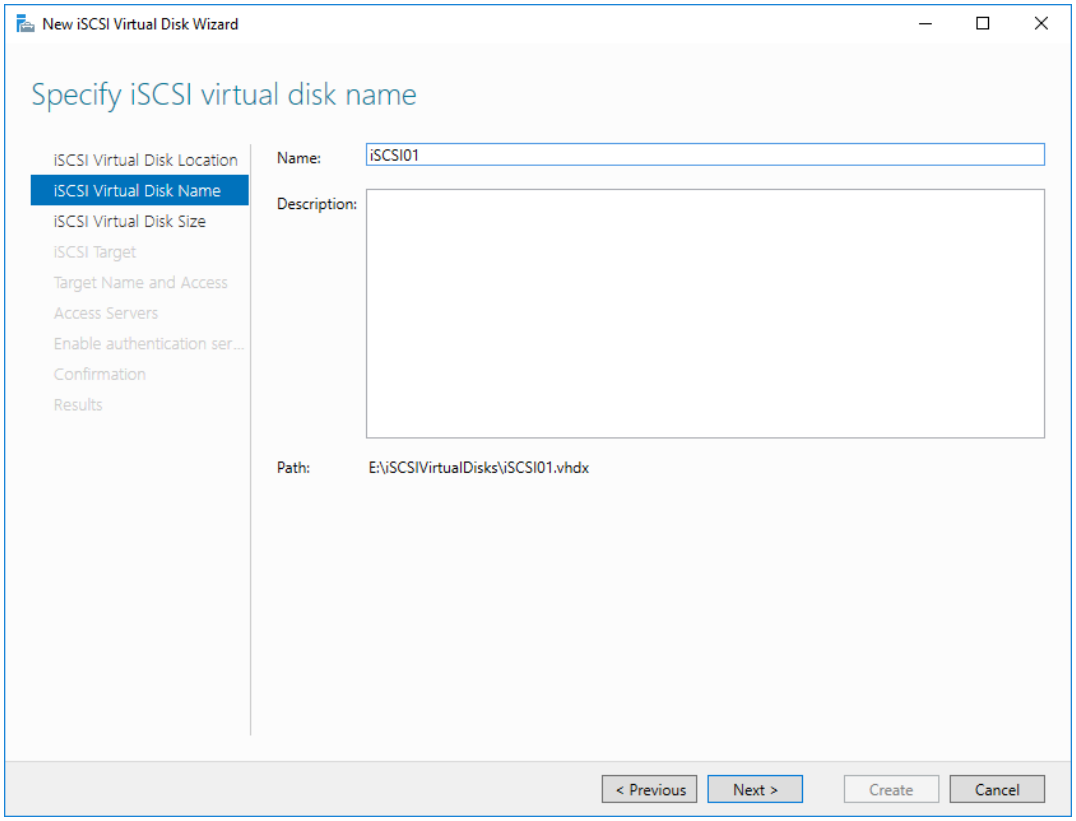

Figure 63

Set the iSCSI virtual disk size. There are useful tips that can help you select the best options. In this example, the *dynamically expanding* disk type is used *(see Figure 64)*. Click **Next**.

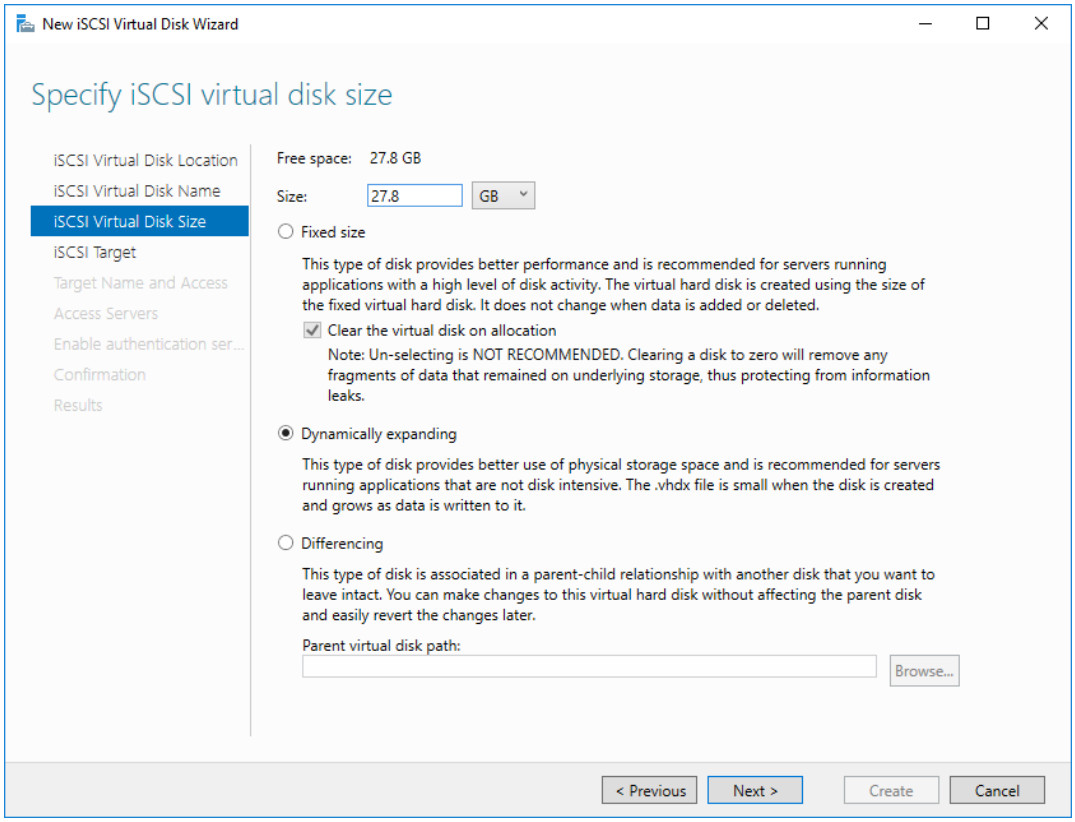

Figure 64

Create a new iSCSI target *(see Figure 65)*. Click **Next**.

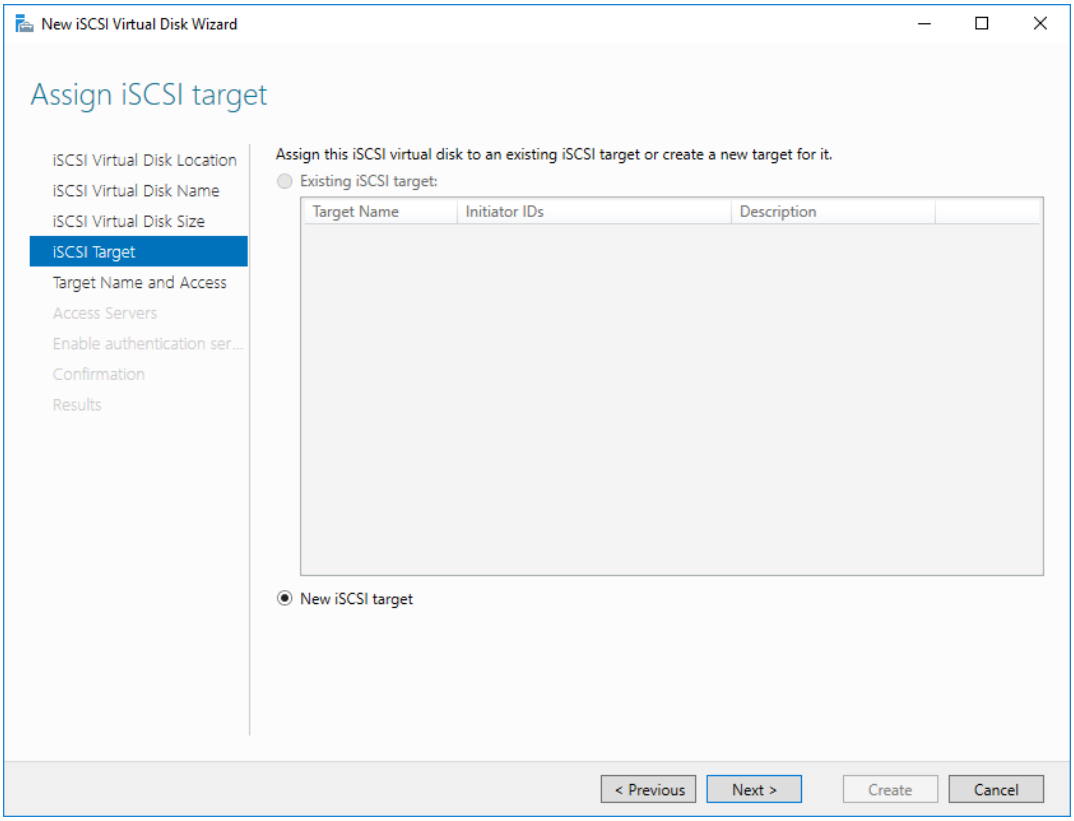

Figure 65

Set the iSCSI target name *(see Figure 66)*. Click **Next**.

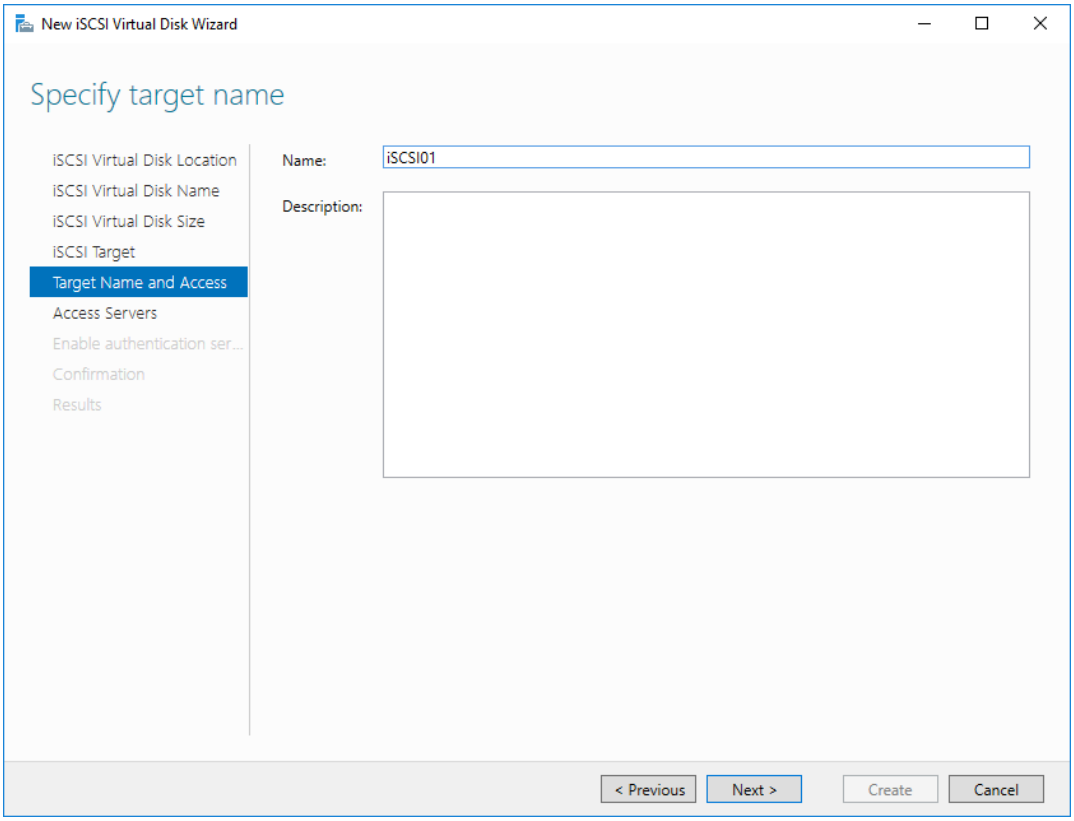

Figure 66

Select the servers that should be allowed access to the iSCSI target *(see Figure 67)*.

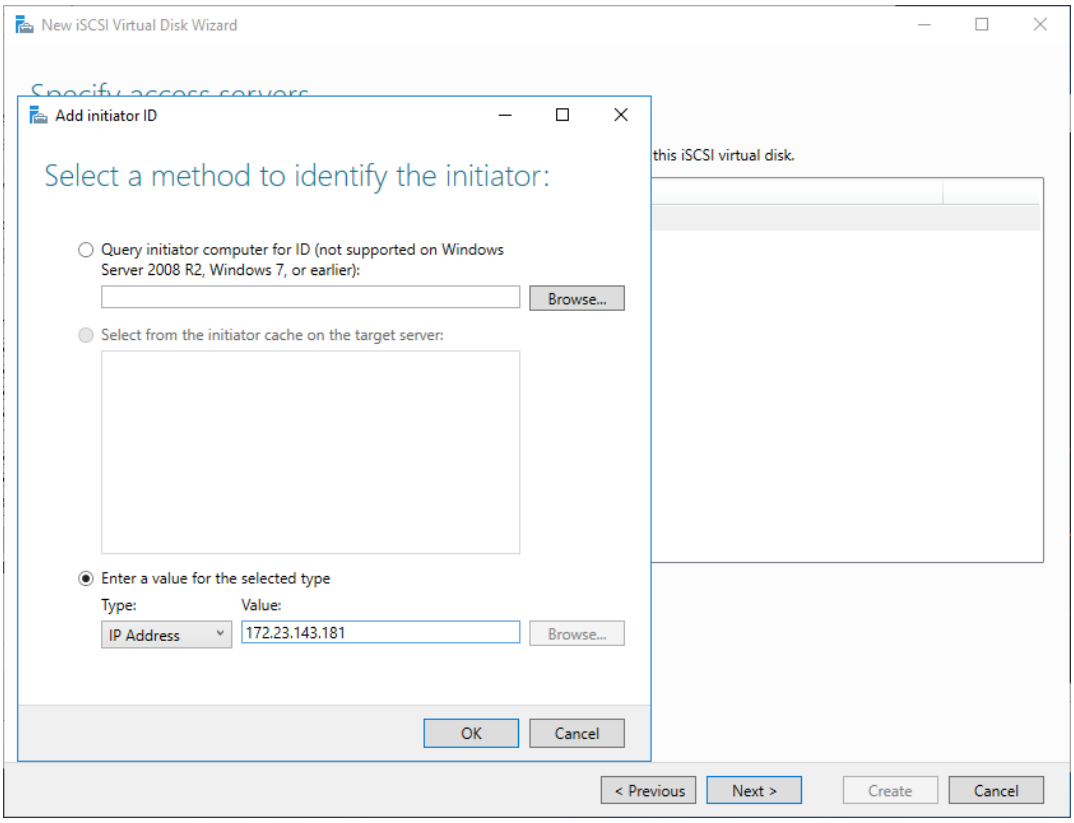

Figure 67

Make sure that you add all servers that should use the shared storage to the access list. Click **Create**. Authentication is not used in this example. Click **Next**, check the settings on the confirmation screen *(see Figure 68)*, then click **Create**.

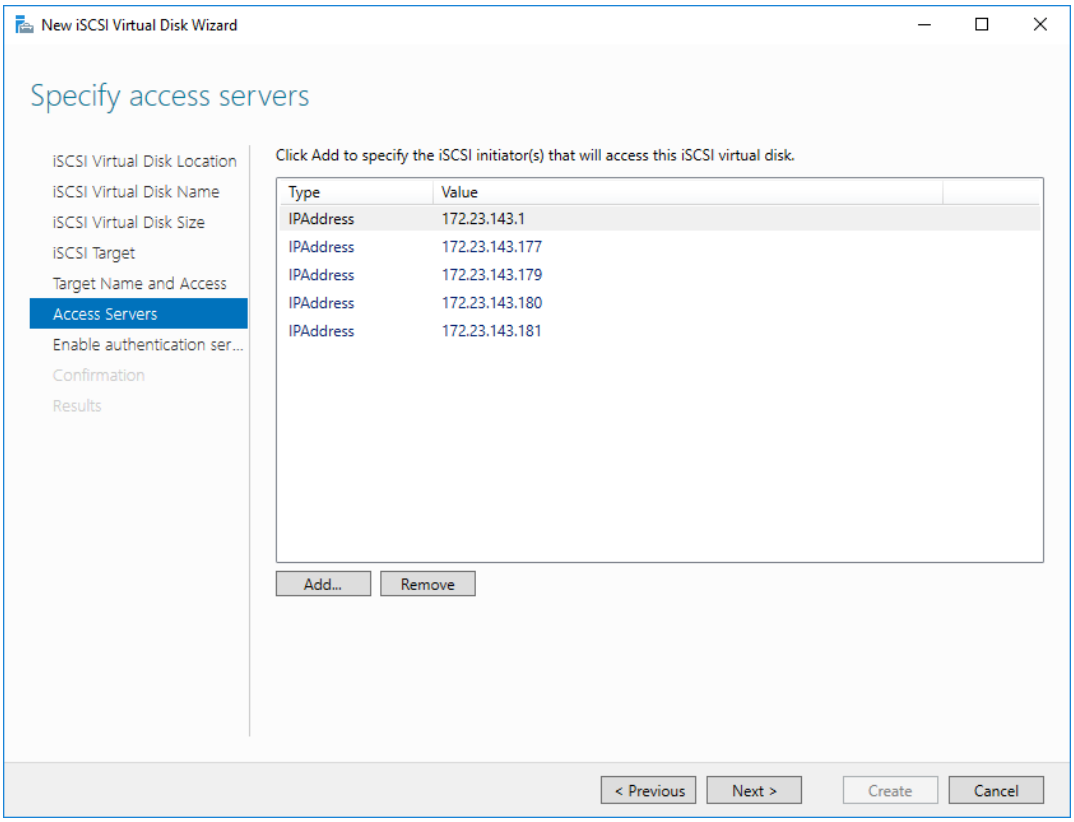

Figure 68

Wait while the iSCSI virtual disk is created *(see Figure 69).*

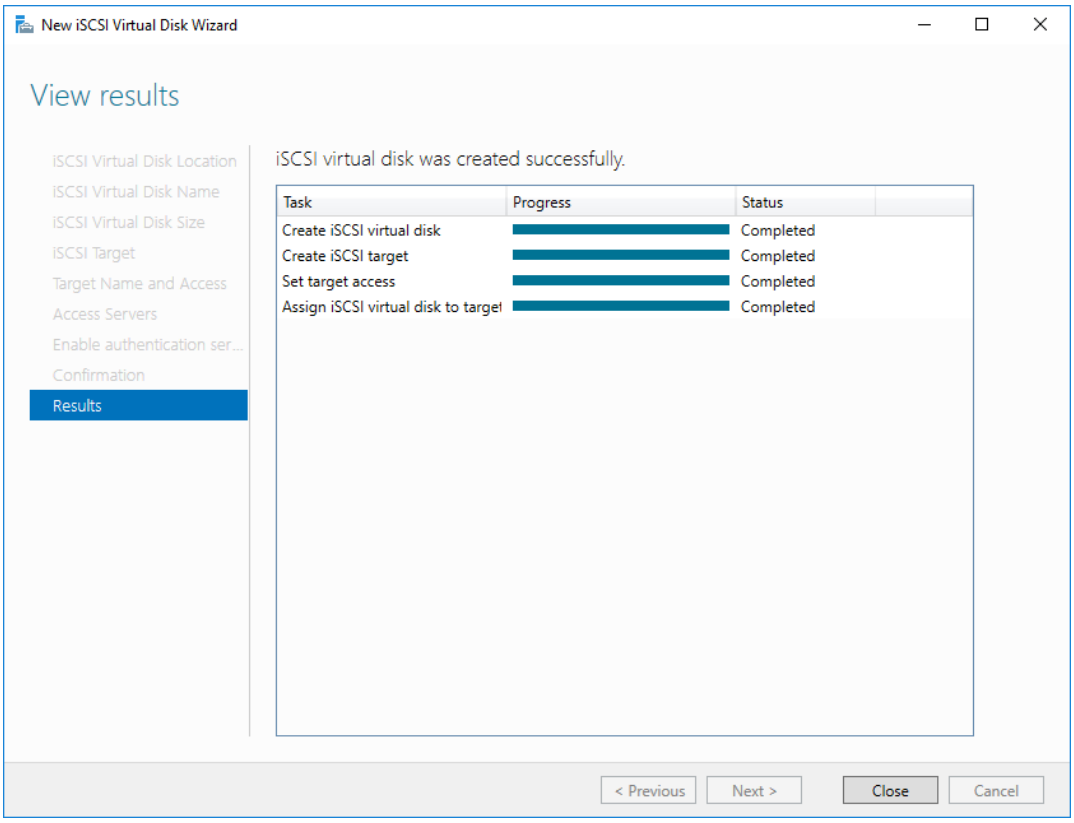

Figure 69

The iSCSI target is now ready for shared storage. Click the **Close** button to close the wizard window.

Repeat these steps and create a shared iSCSI disk for the witness. A 2 GB disk is sufficient for a witness.

Connecting nodes to the iSCSI target

Now that you have created the iSCSI target for shared storage, you can connect the first Hyper-V server to the iSCSI target.

Open the **Server Manager**. Click **Tools** and select **iSCSI initiator** *(see Figure 70).*

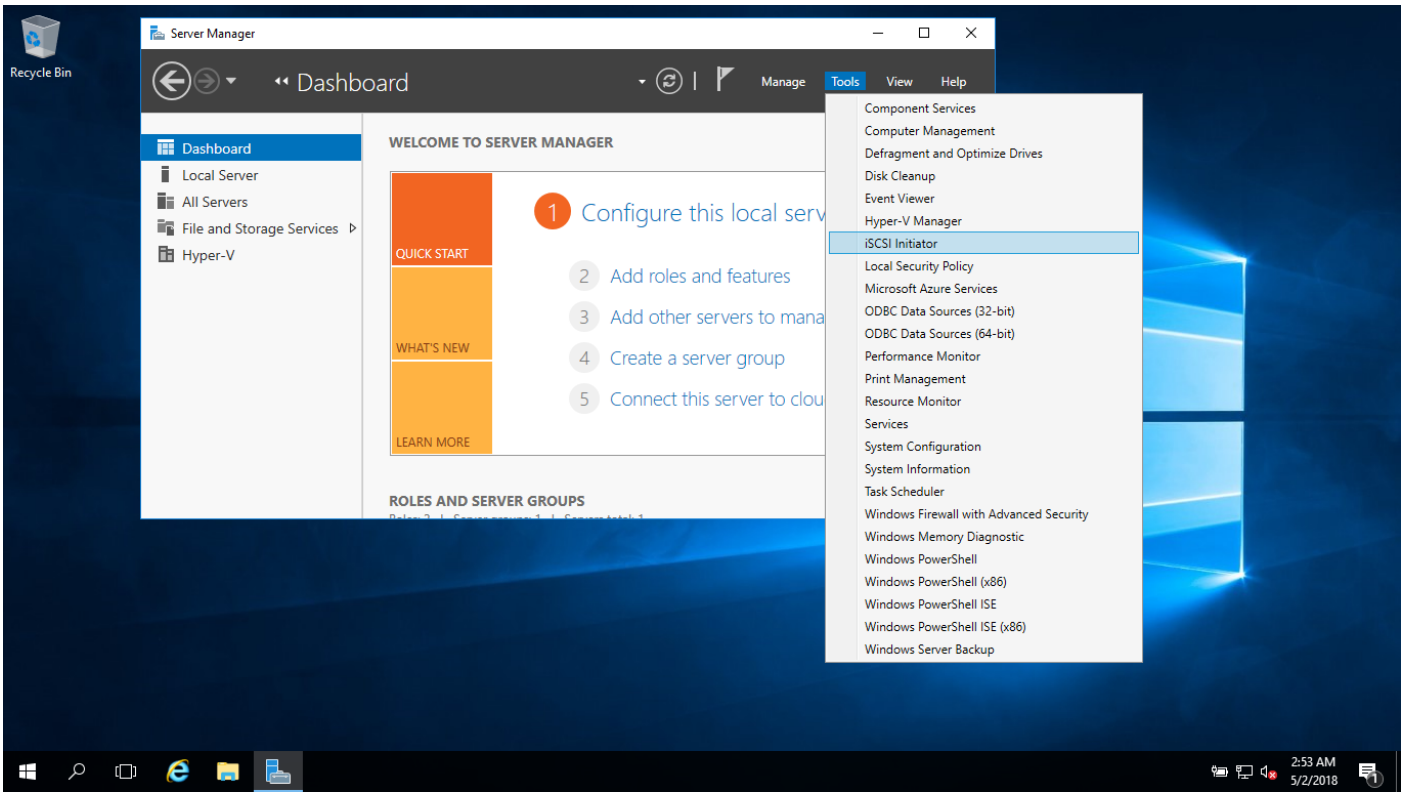

Figure 70

If the iSCSI service is not running, Windows asks you to start this service. Read the warning and click the **Yes** button *(see Figure 71).*

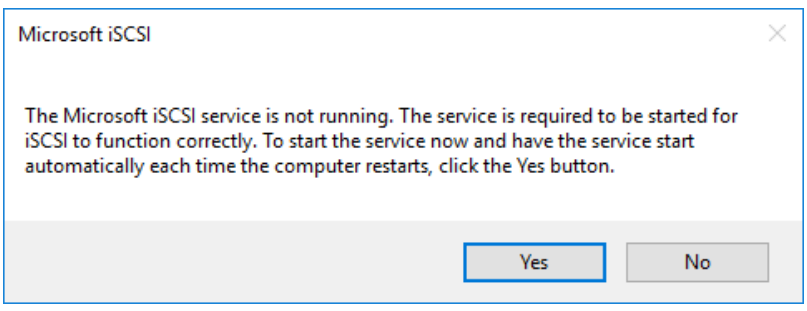

Figure 71

Under the *iSCSI Initiator Properties*, go to the **Discovery** tab, click the **Discover Portal** button, and select the IP address of iSCSI target. In this example, the IP address of shared storage is 172.23.143.178 (*see Figure 72)*. Your IP address and network are likely to differ.

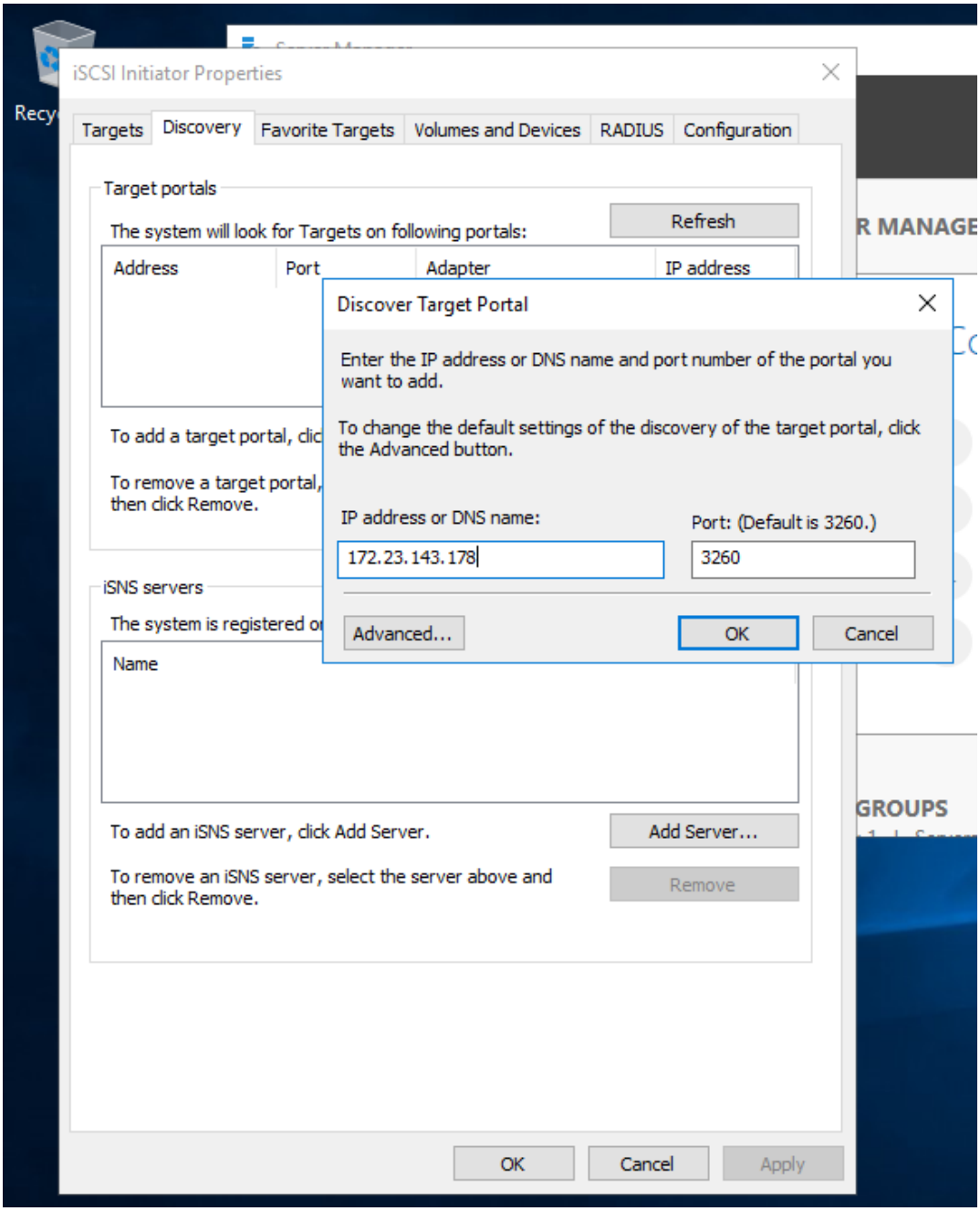

Figure 72

Your target portals appear in the list *(see Figure 73).*

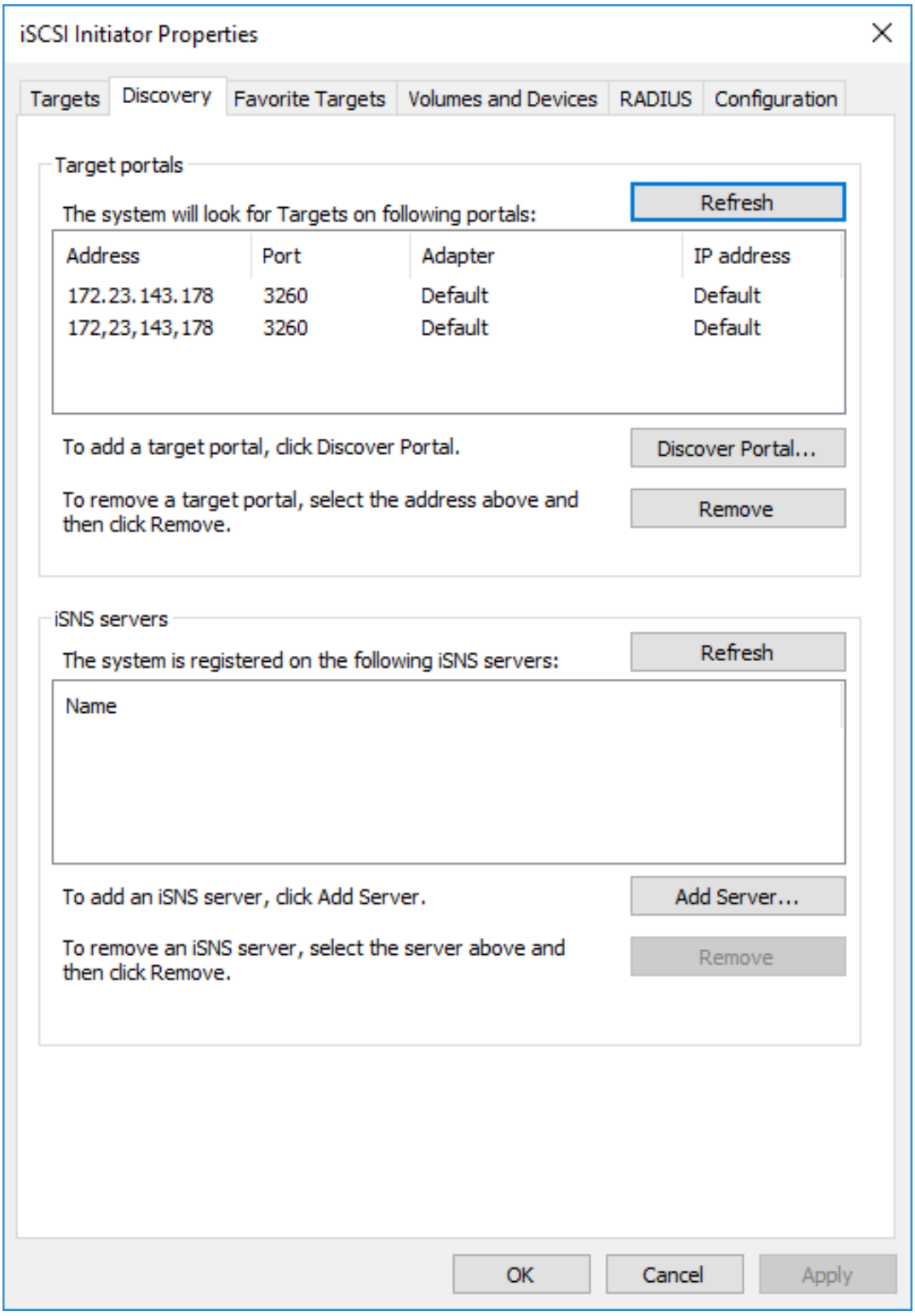

Figure 73

Go to the **Targets** tab, select the iSCSI targets, and click the **Connect** button *(see Figure 74).*

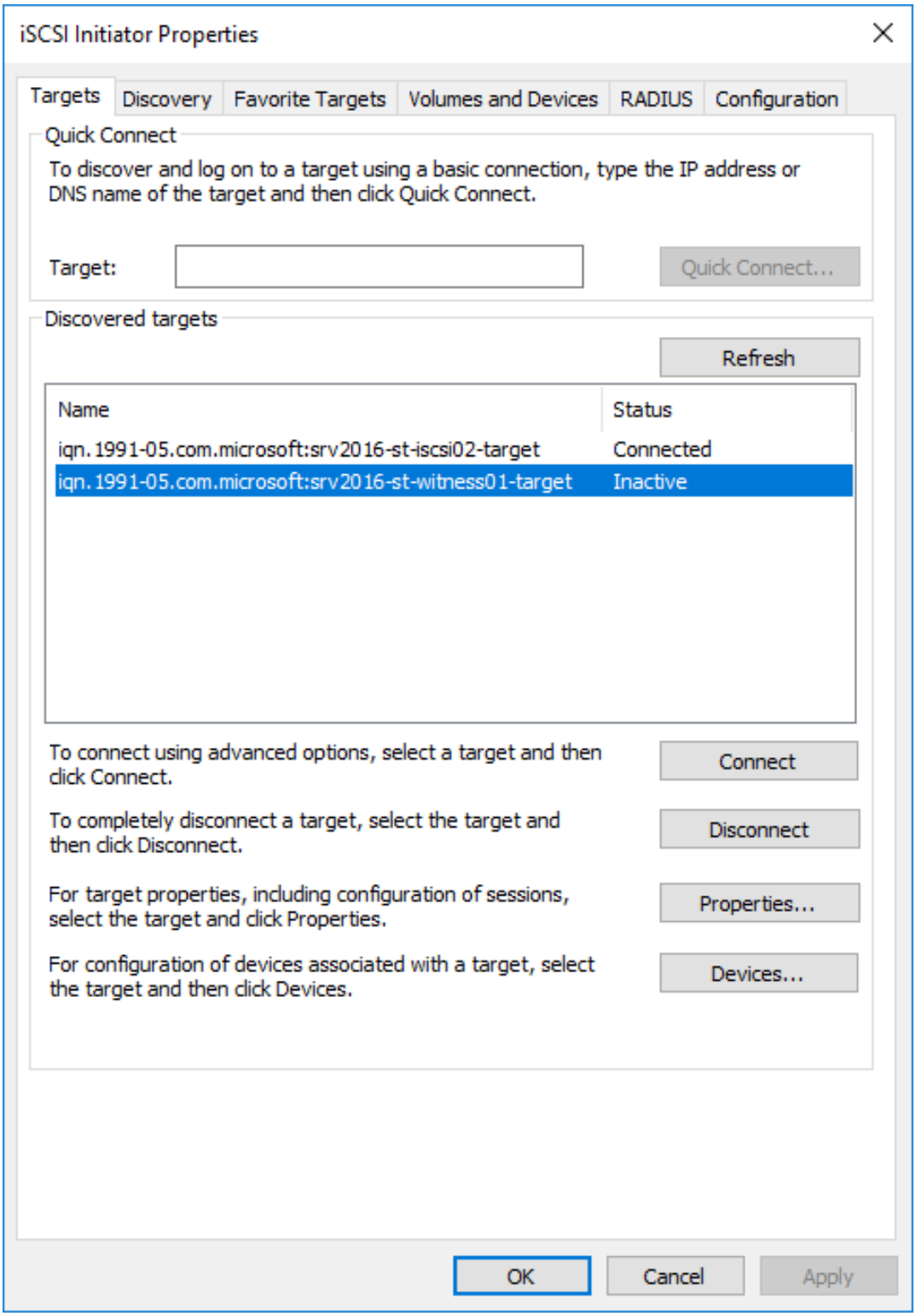

Figure 74

Now you can initialize the connected disks. On your Hyper-V host, open **Disk Management** (**Start** > **Windows Administrative Tools** > **Computer Management**). The connected disks are offline. Right-click the disks and select the **Online** option *(see Figure 75)*.

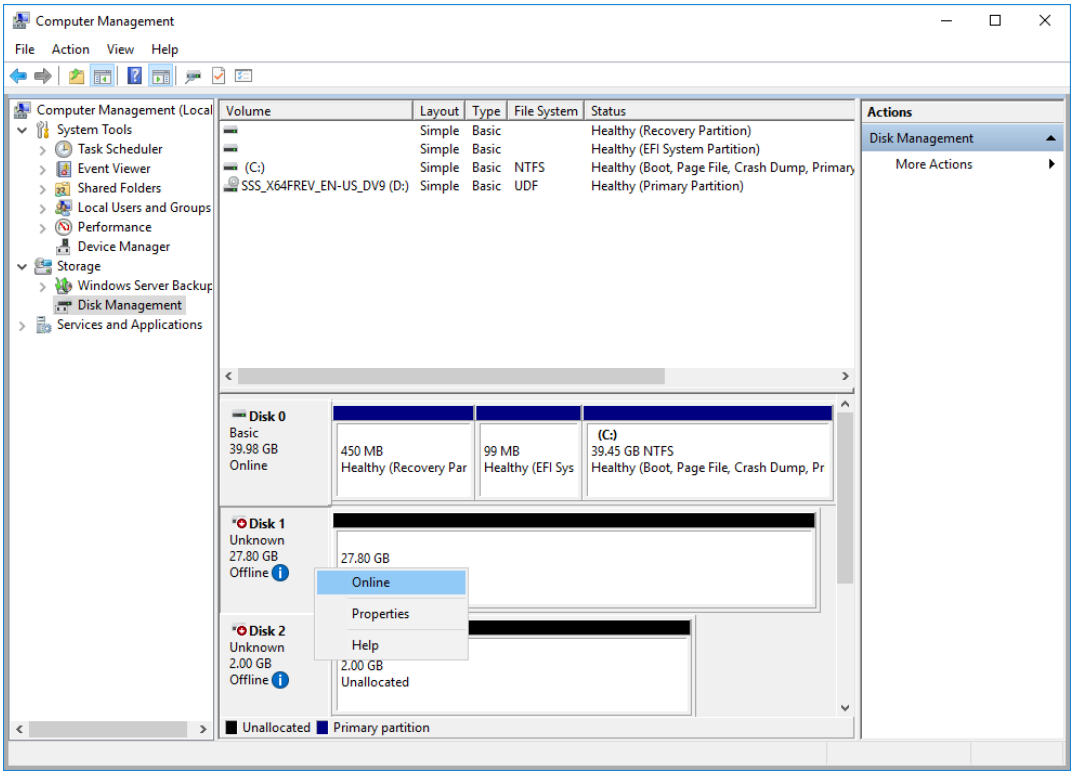

Figure 75

Figure 75

Then right-click the disk and select the **Initialize Disk** option *(see Figure 76)*.

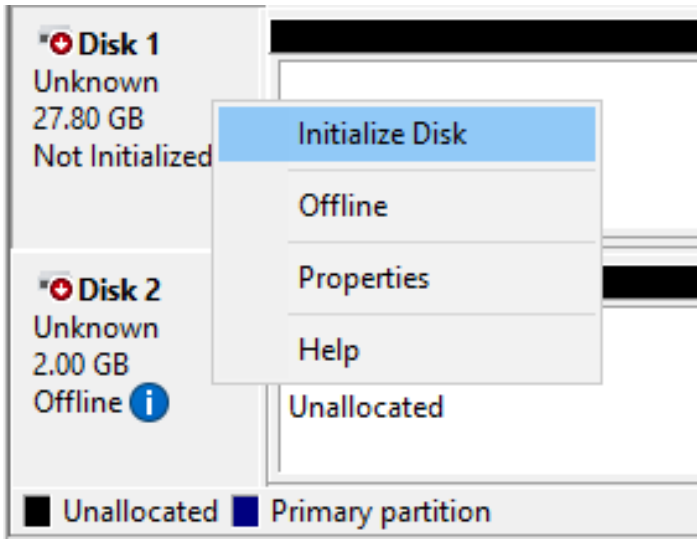

Figure 76

After that, right-click the disk and select **New Simple Volume** *(see Figure 77)*.

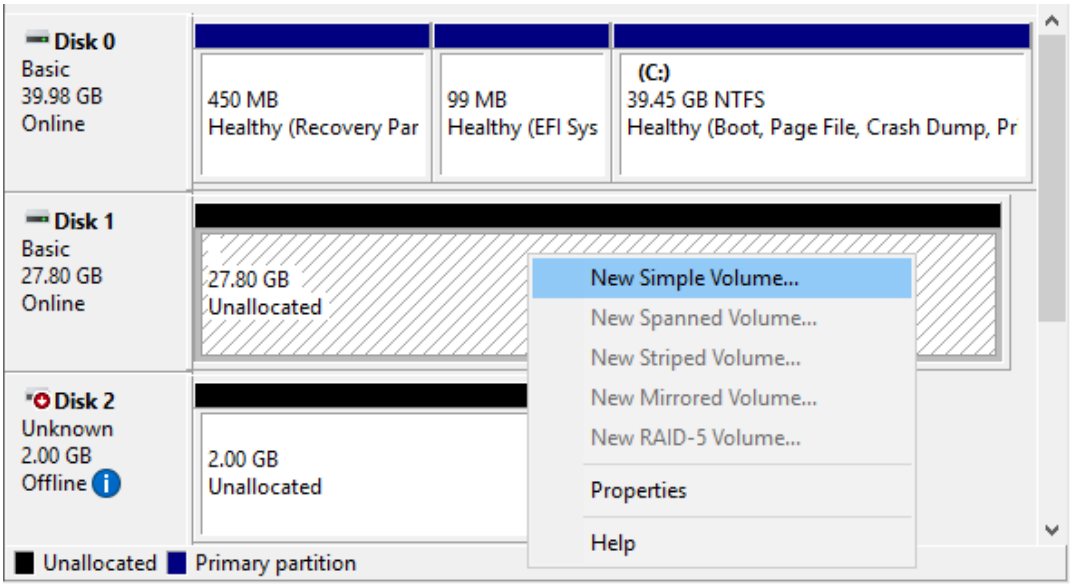

Figure 77

Set a volume label. Volume labels must be the same for all Hyper-V hosts in a cluster *(see Figure 78).*

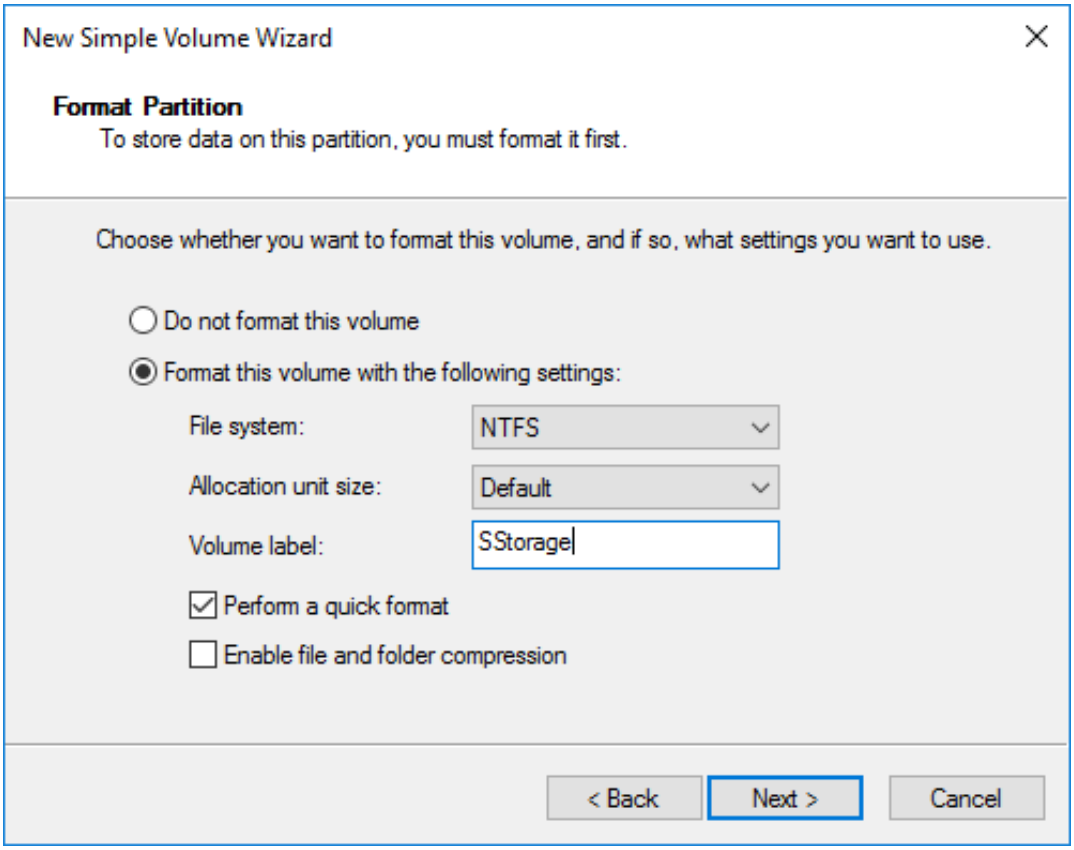

Figure 78

Carry out the steps above for each disk connected to the Hyper-V Server as an iSCSI target *(see Figure 79).*

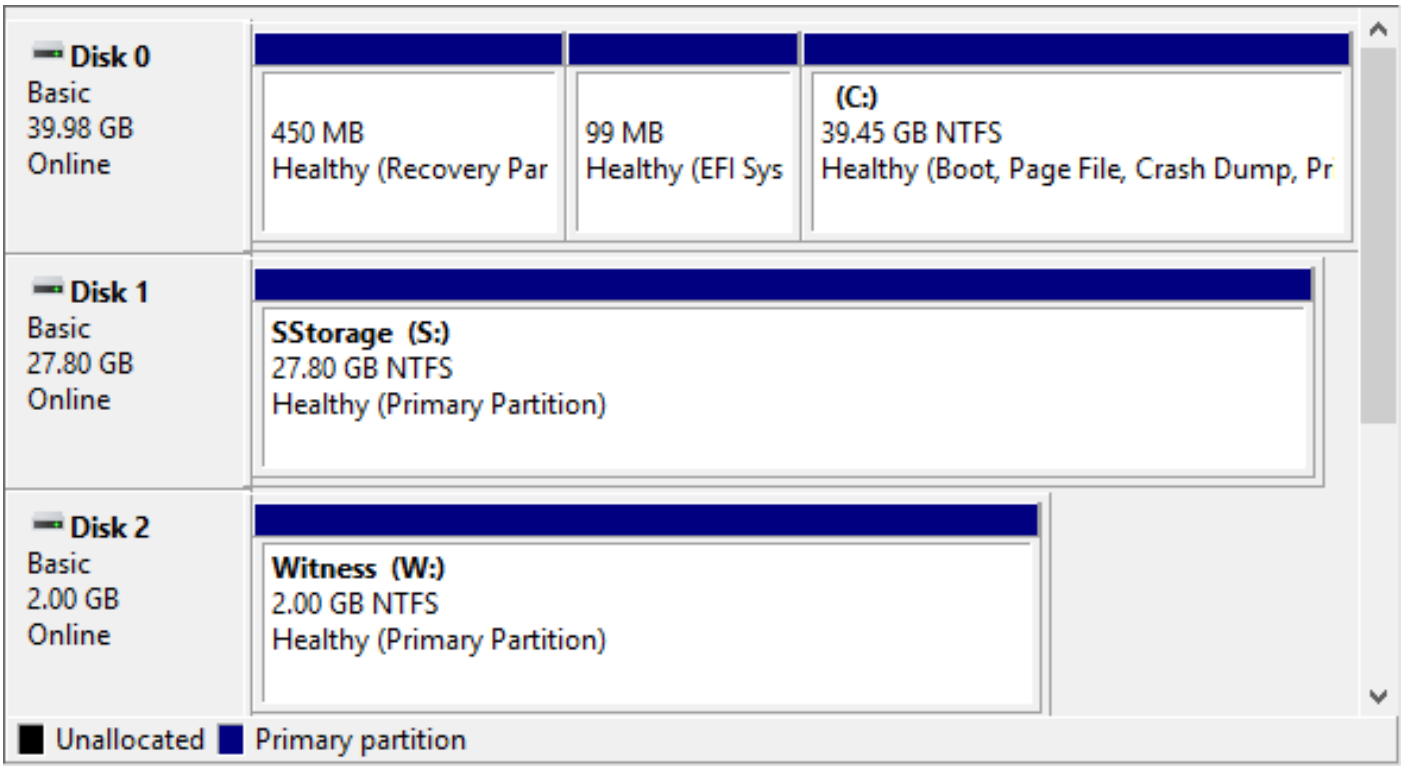

Figure 79

Repeat these steps for the other Hyper-V hosts.

Note:

After setting the status of the shared disks on your other Hyper-V hosts to Online, you can see the parameters configured above, read from the shared storage (disk size, names of volumes etc.).

Installing the Failover Clustering role on a Hyper-V host

Open the Server Manager, select **Add roles and features** (follow the Wizard, as described in the section on Hyper-V server role installation). In the **Features** section, tick the **Failover Clustering** checkbox, click **Next**, then click **Install** *(see Figure 80).*

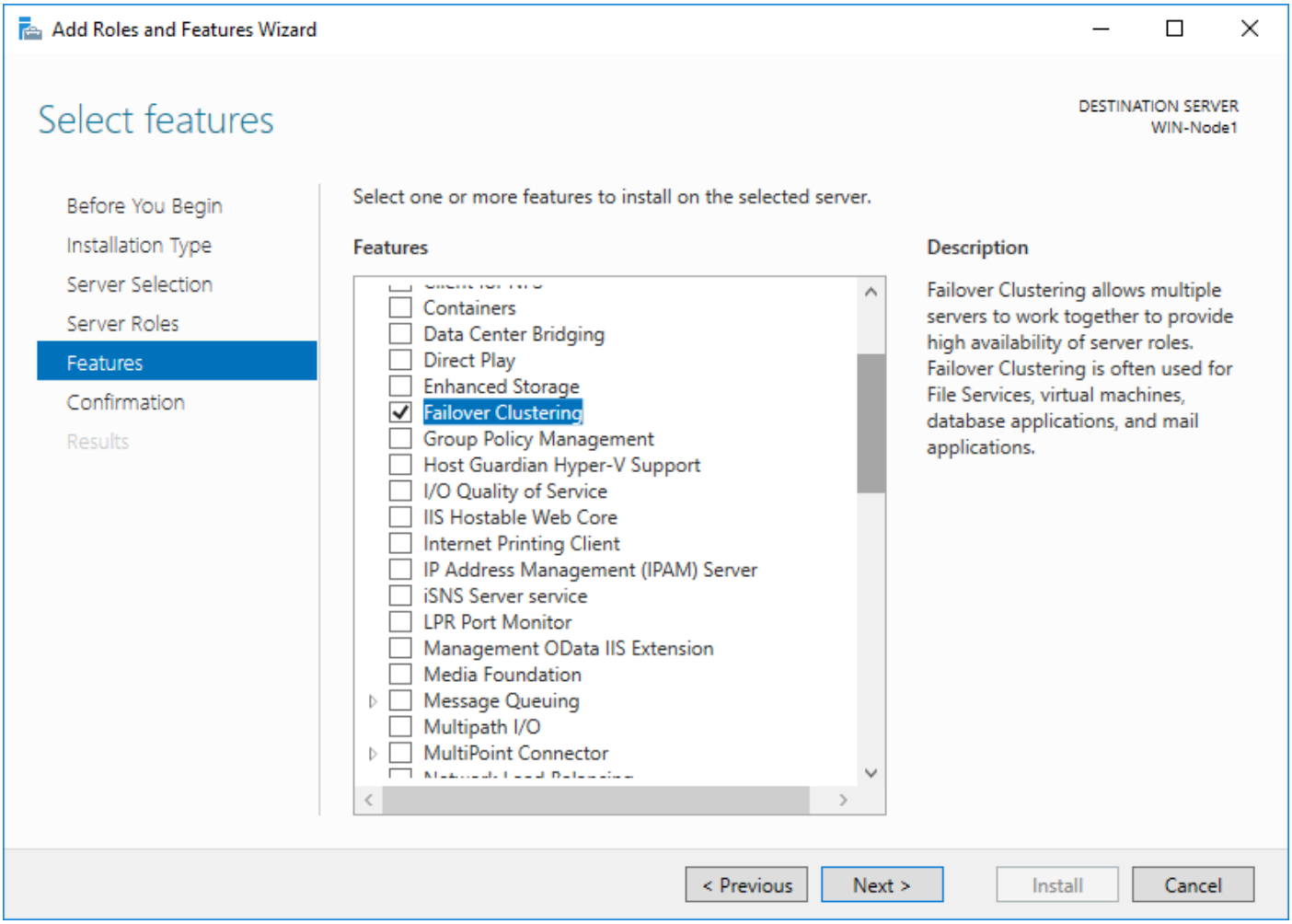

Figure 80

Wait while the Failover Clustering feature is installed *(see Figure 81).*

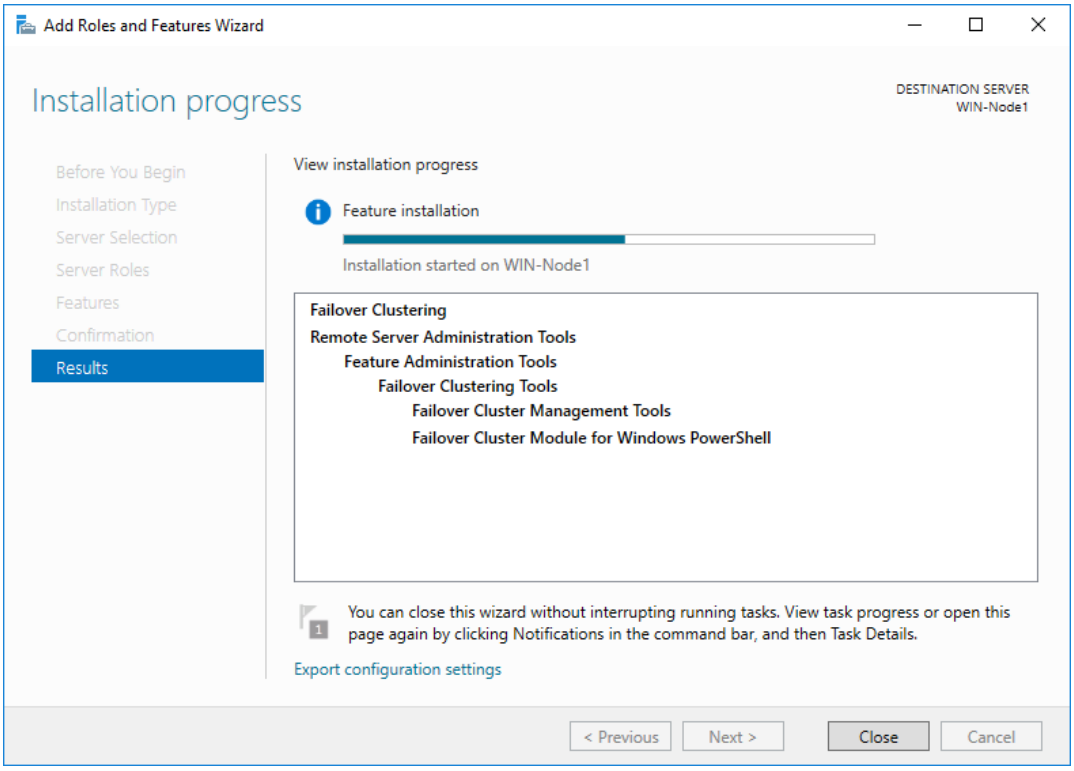

Figure 81

Go to the Server Manager. Click **Tools** > **Failover Cluster Manager** *(see Figure 82).*

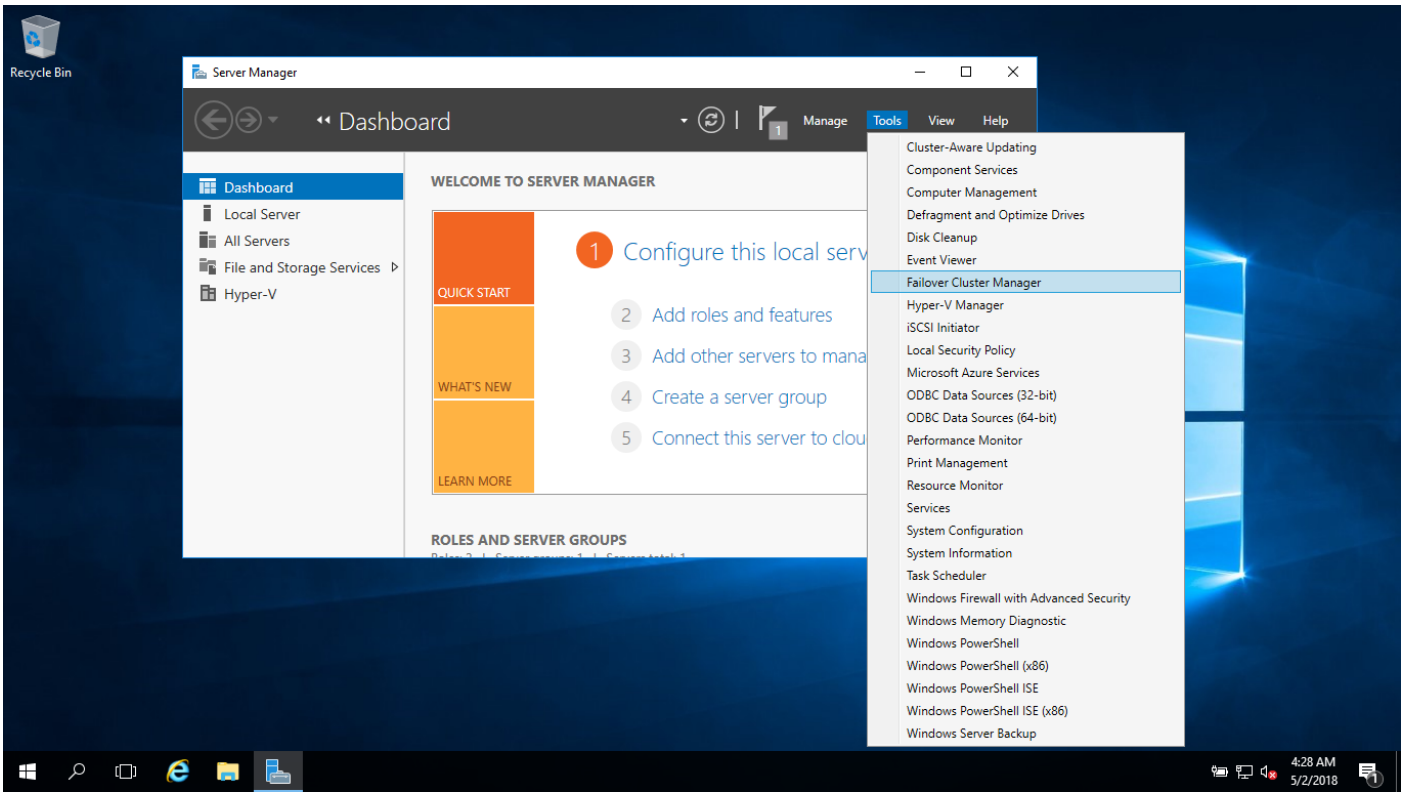

Figure 82

The *Failover Cluster Manager* is launched in a new window *(see Figure 83)*. It is recommended that you first validate the configuration to make sure that the cluster can be successfully deployed. In order to do this, click **Validate Configuration** in the *Management* section of the interface.

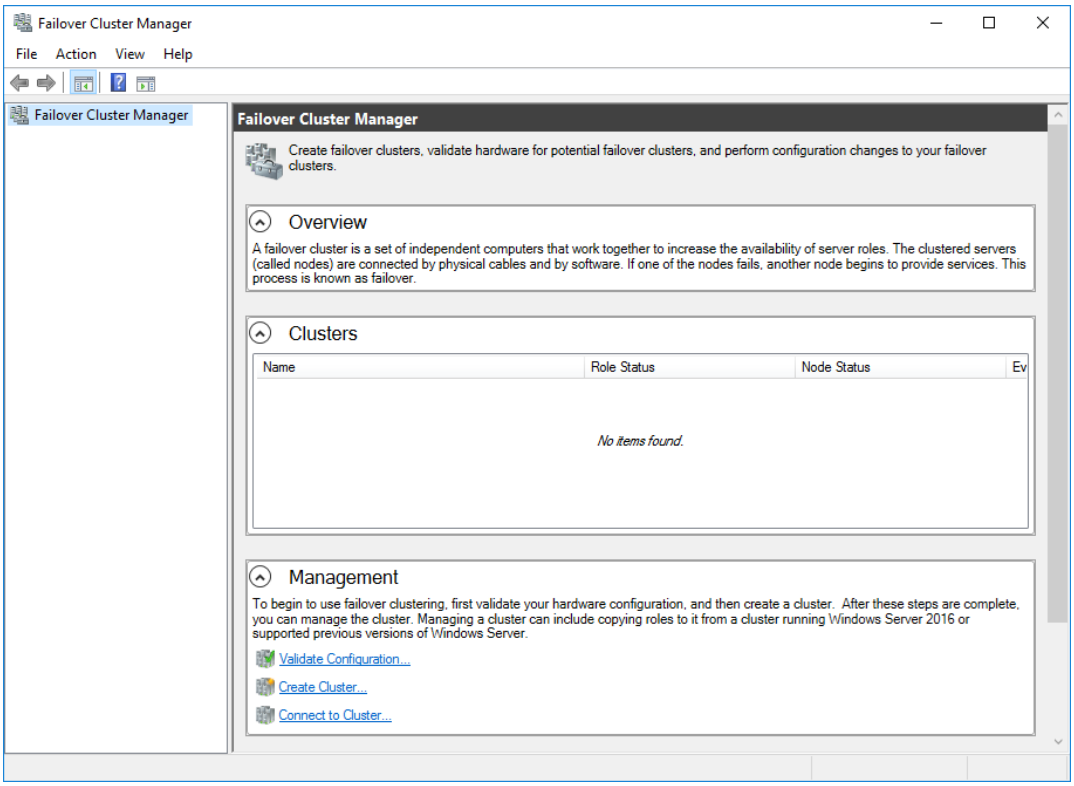

Figure 83

Select the names of the servers you want added to the cluster (these names are set in the computer name settings of your Windows system properties). Enter a server name and click the **Add** button. In the example below *(see Figure 84)*, the *WIN-Node0* and *WIN-Node1* servers have been added for validation. Click **Next**.

Note:

If you use Active Directory, add Hyper-V hosts to the same domain controller (with the appropriate permissions) before creating a cluster.

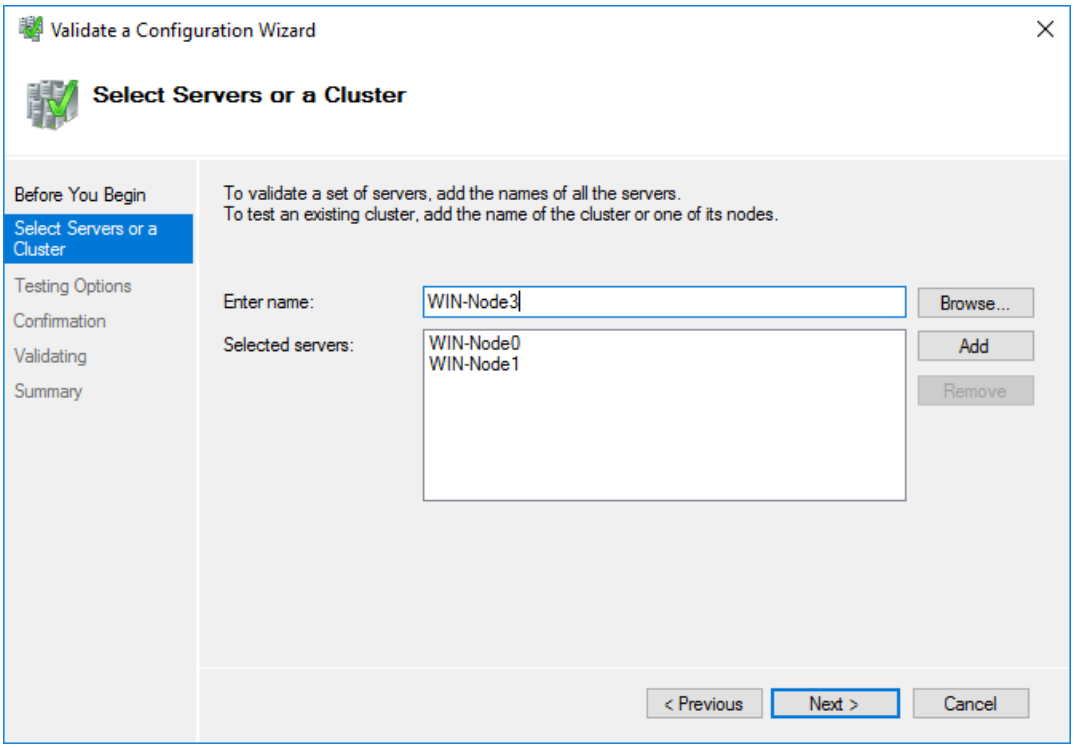

Figure 84

At the next step, you can select which tests to run. You can run all the tests, or only selected ones *(see Figure 85)*. Click **Next** when you are ready to proceed.

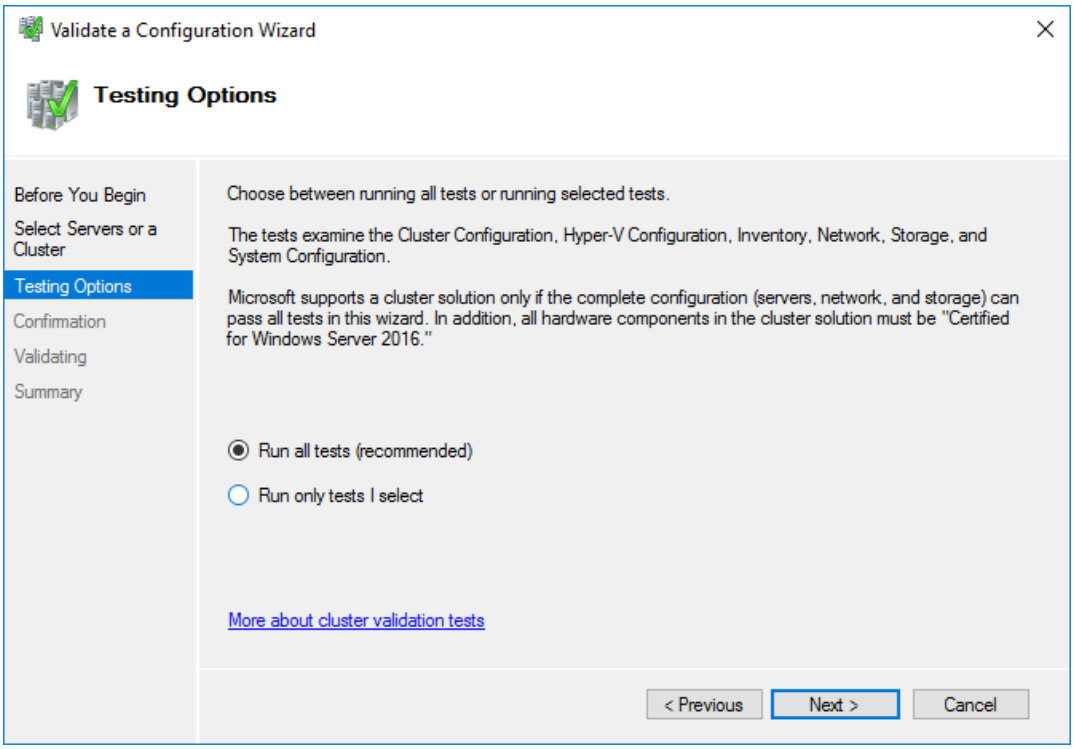

Figure 85

At the Confirmation screen, click **Next** and wait until the tests are complete *(see Figure 86)*.

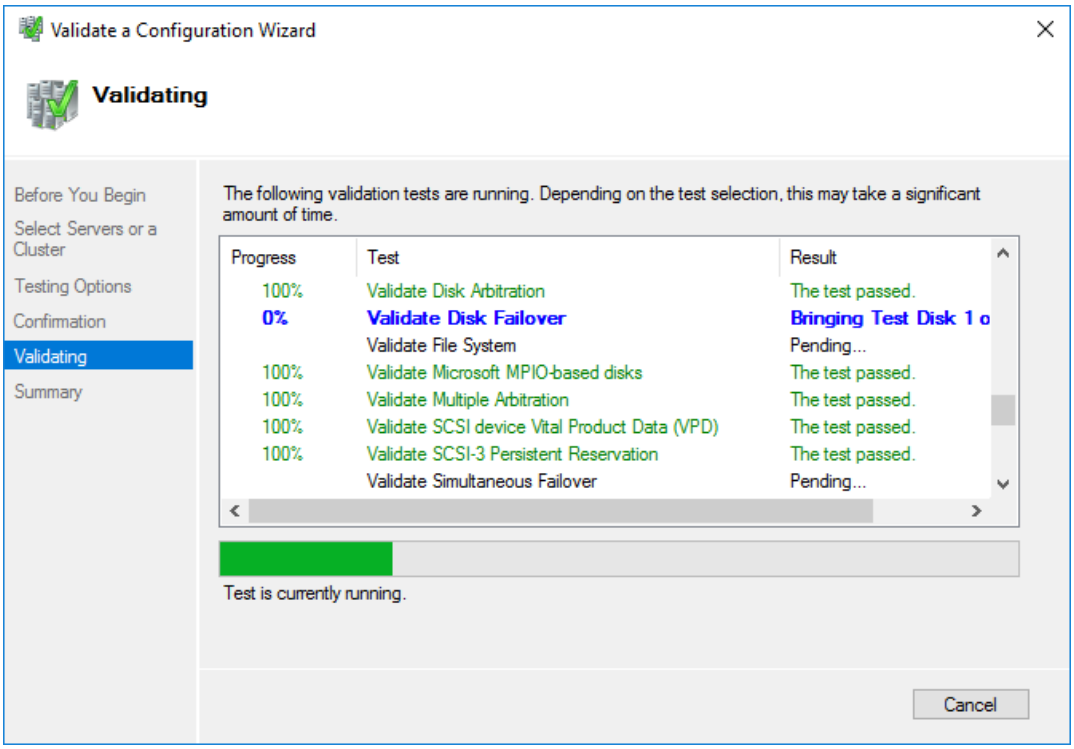

Figure 86

Review the validation report and click **Finish** *(see Figure 87)*.

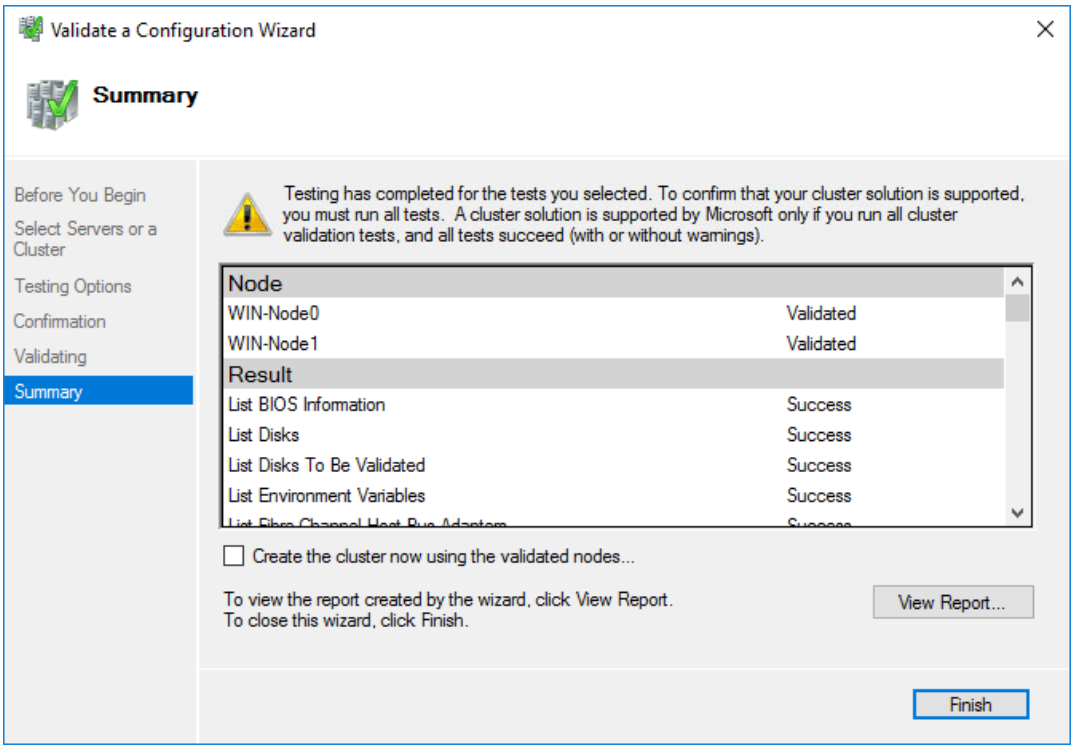

Figure 87

Now you are ready to create a cluster.

Creating a Cluster

In order to create a cluster, click **Create Cluster** in the **Management** section of the Failover Cluster Manager window. The *Create Cluster Wizard* is launched in a new window. Add the servers you want to use in the cluster by entering each server name then clicking **Add**. When all servers have been added, click **Next** *(see Figure 88).*

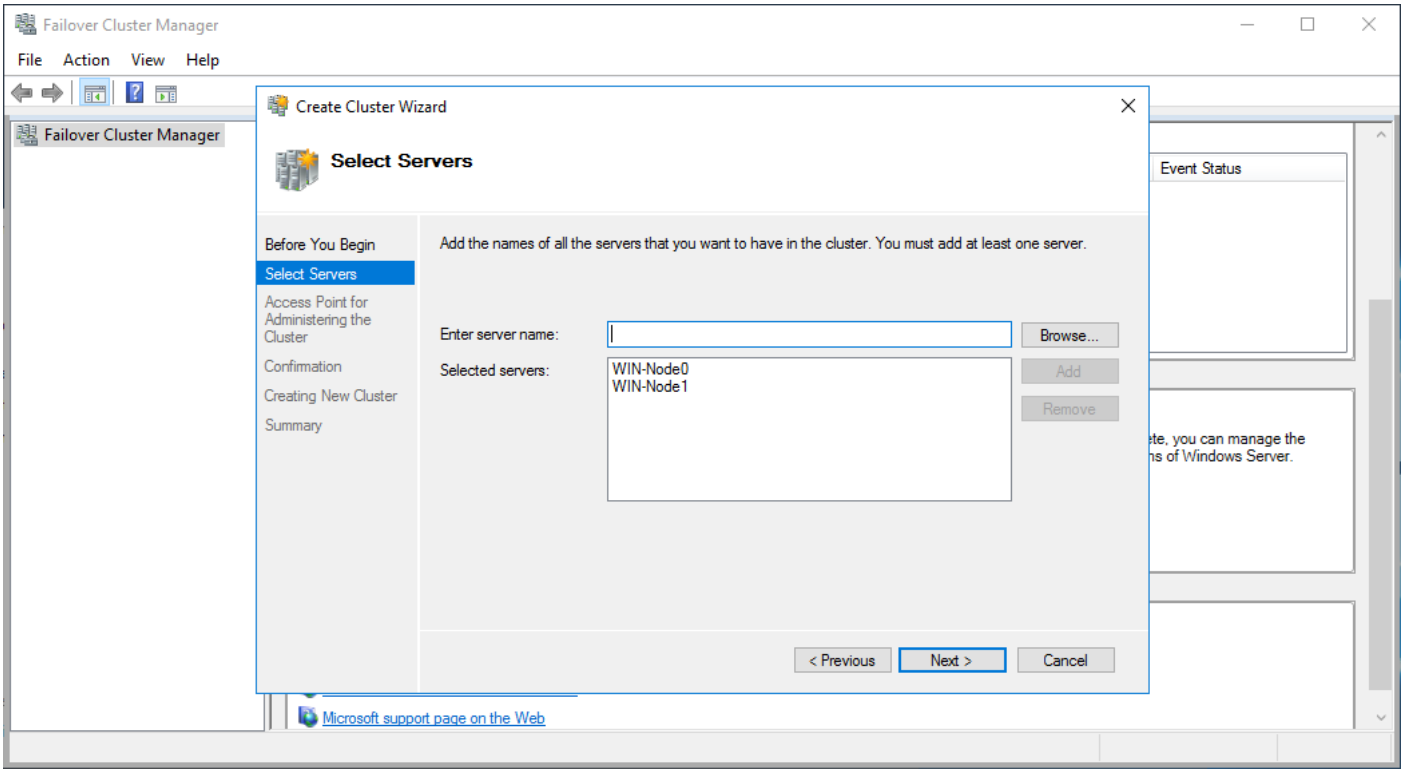

Figure 88

Set the access point for cluster administration. Define the cluster name and IP address *(see Figure 89)*. Click **Next.**

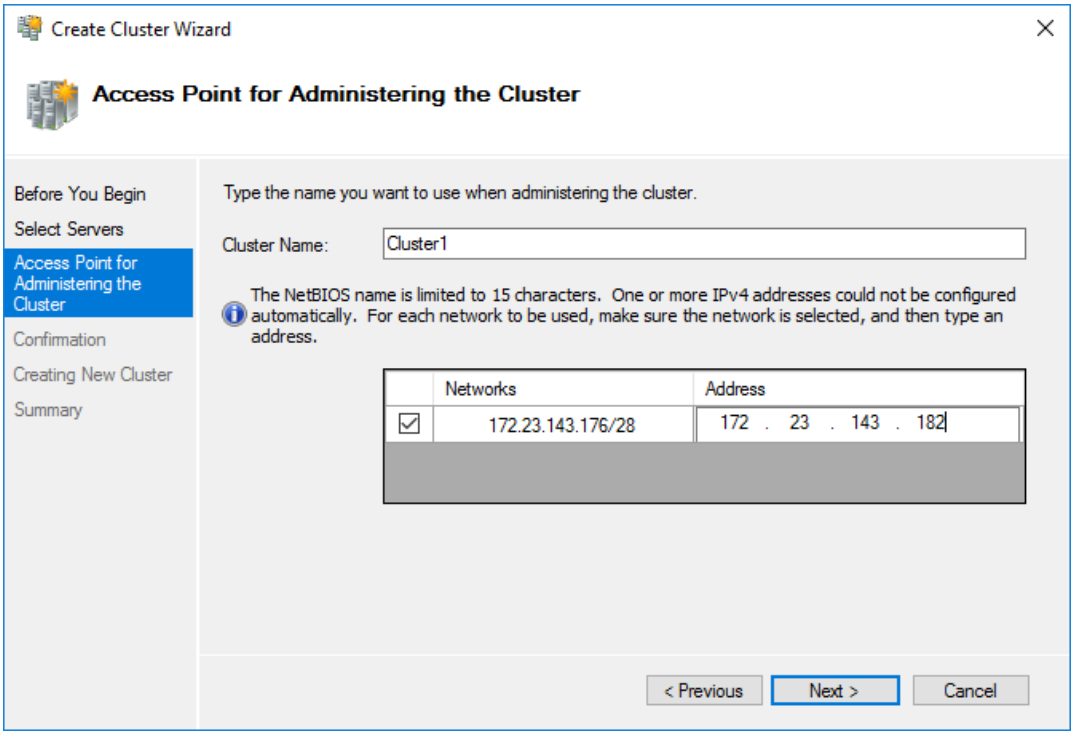

Figure 89

At the confirmation screen, check the settings and click **Next** when you are satisfied *(see Figure 90).*

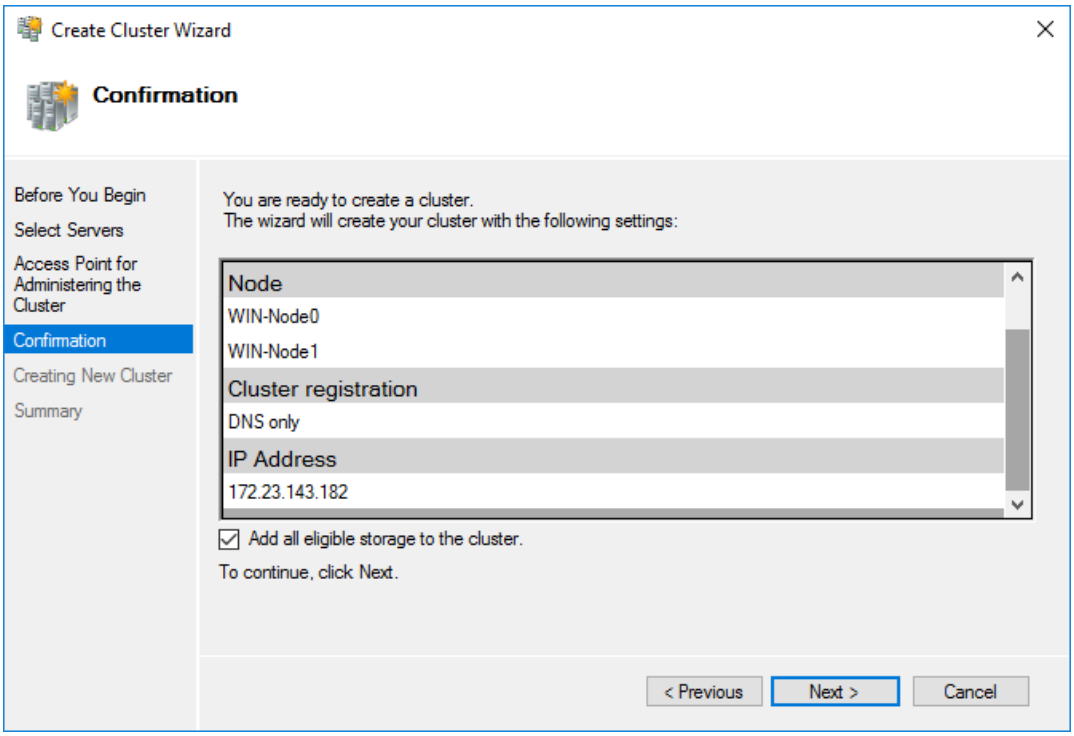

Figure 90

Wait while the cluster is configured *(see Figure 91).*

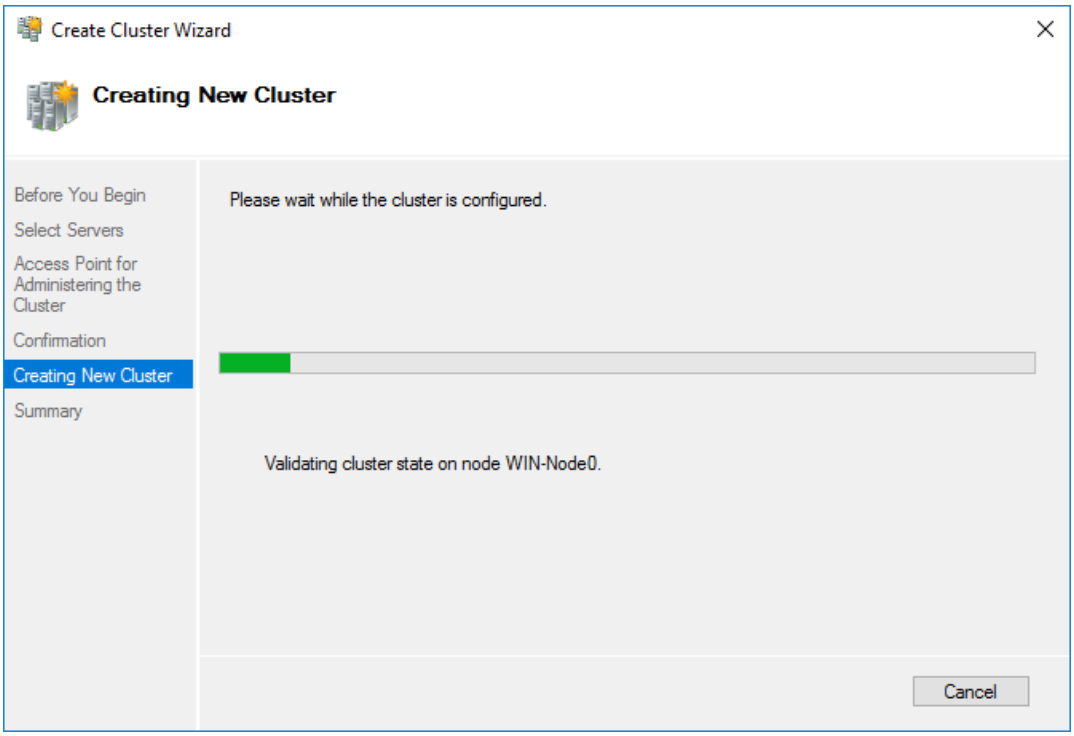

Figure 91

The cluster has now been successfully created. You can see the summary: cluster nodes, cluster name, quorum type, and IP address for cluster management *(see Figure 92)*. Click **Finish** to close the wizard.

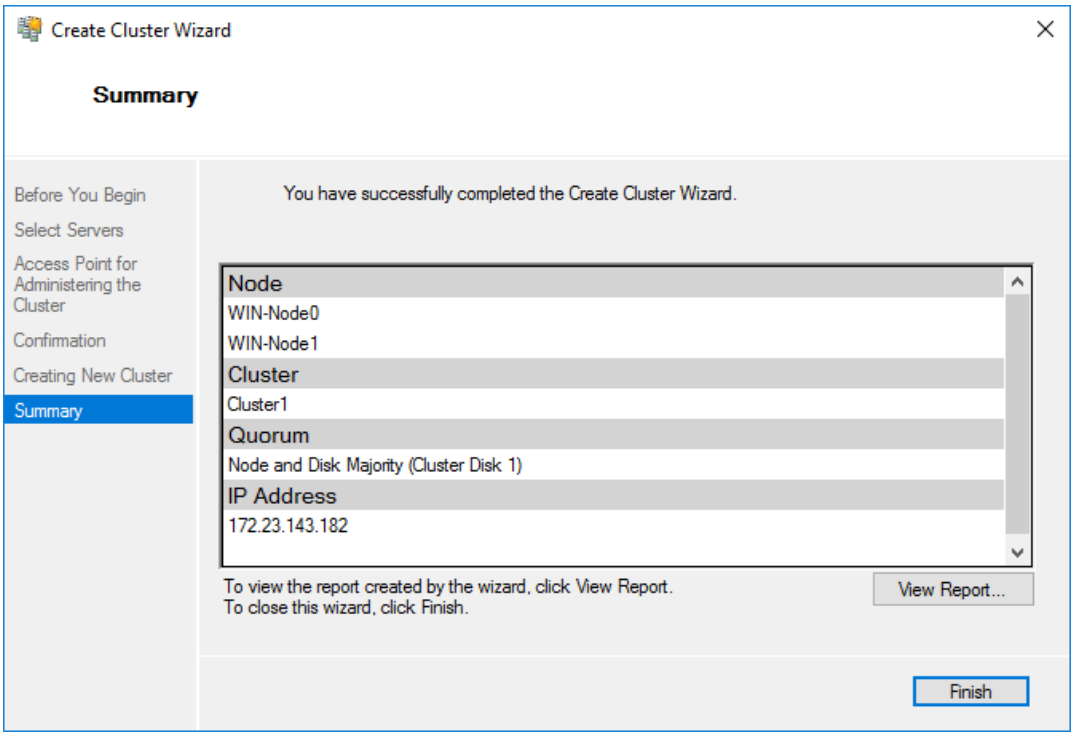

Figure 92

You should now be able to see your recently created cluster (*Cluster1* in this example) in the Failover Cluster Manager *(see Figure 93).*

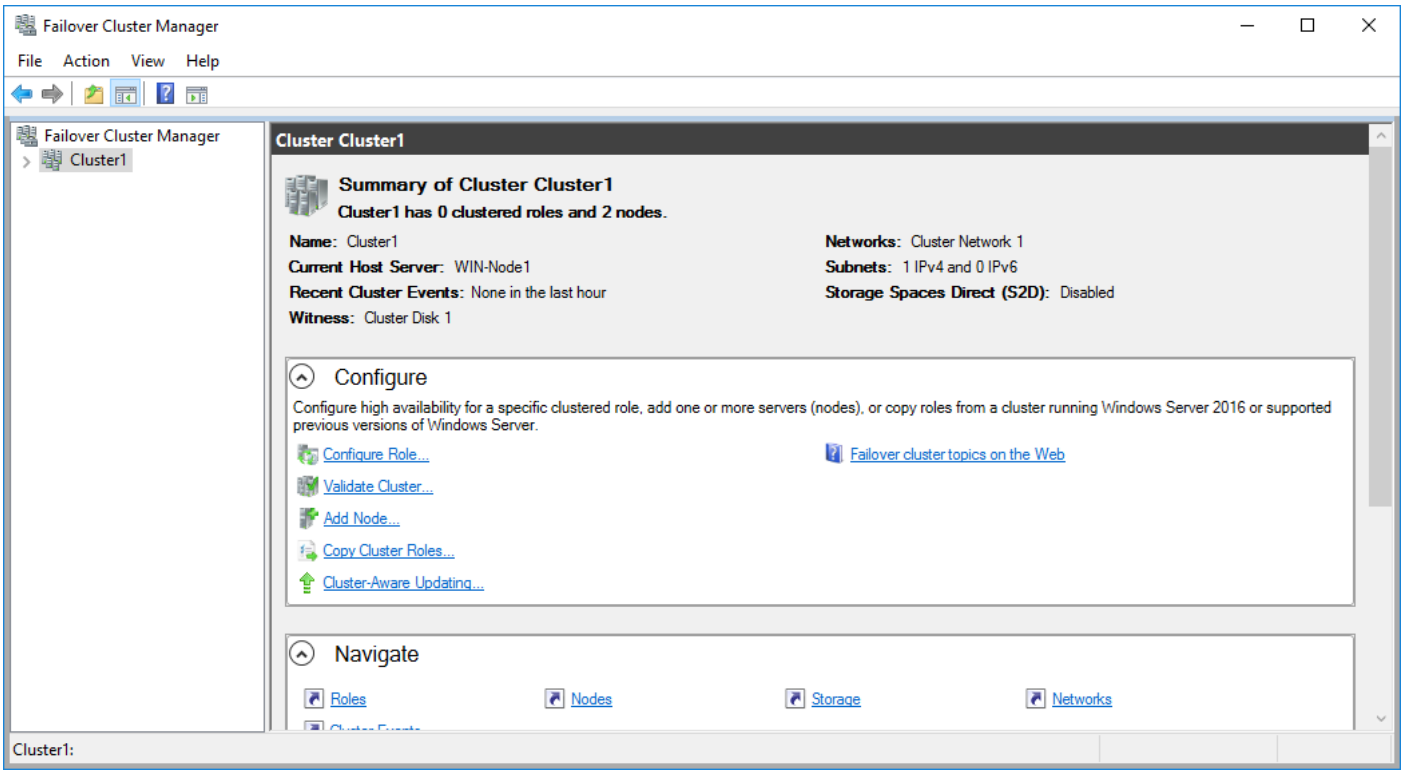

Figure 93

You can click on your cluster and expand the objects therein, such as Roles, Nodes, Disks, Pools, etc., to view or configure them. Cluster disks that are online are shown in the screenshot below *(see Figure 94).*

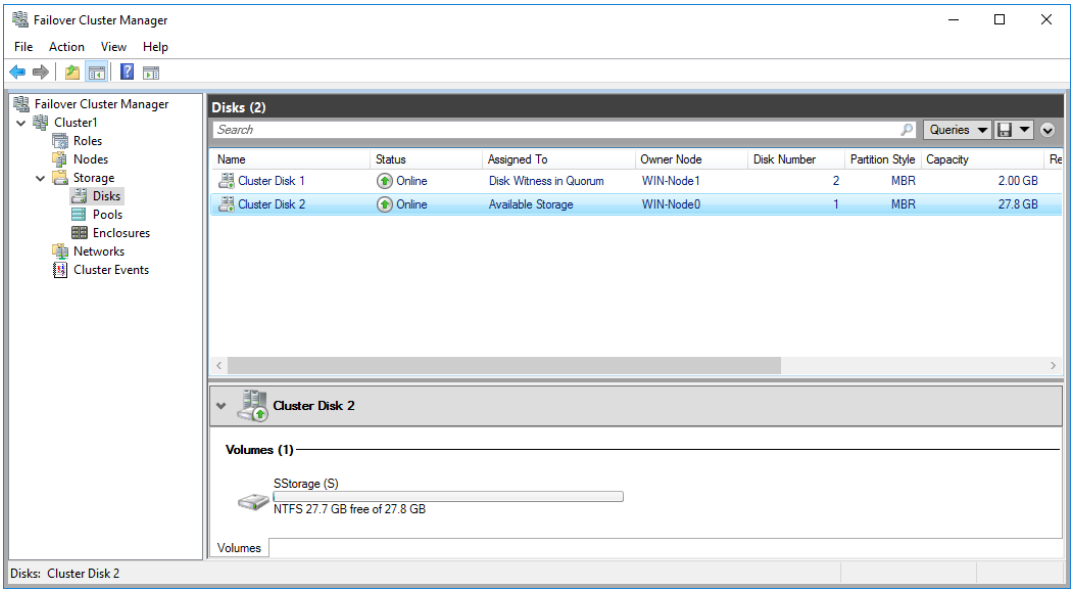

Figure 94

Now add a cluster disk to be used as storage to Cluster Shared Volumes (CSV). Using CSV, all nodes can access the disk simultaneously. Right-click your cluster disk and select the **Add to Cluster Shared Volumes** option in the context menu *(see Figure 95).*

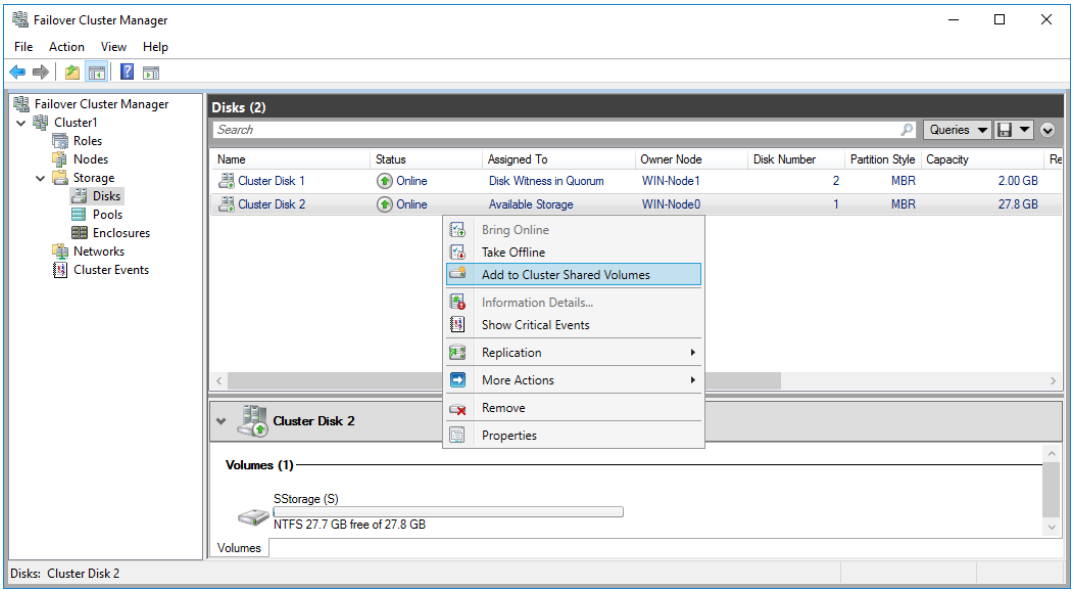

Figure 95

Your CSV is now created *(see Figure 96).*

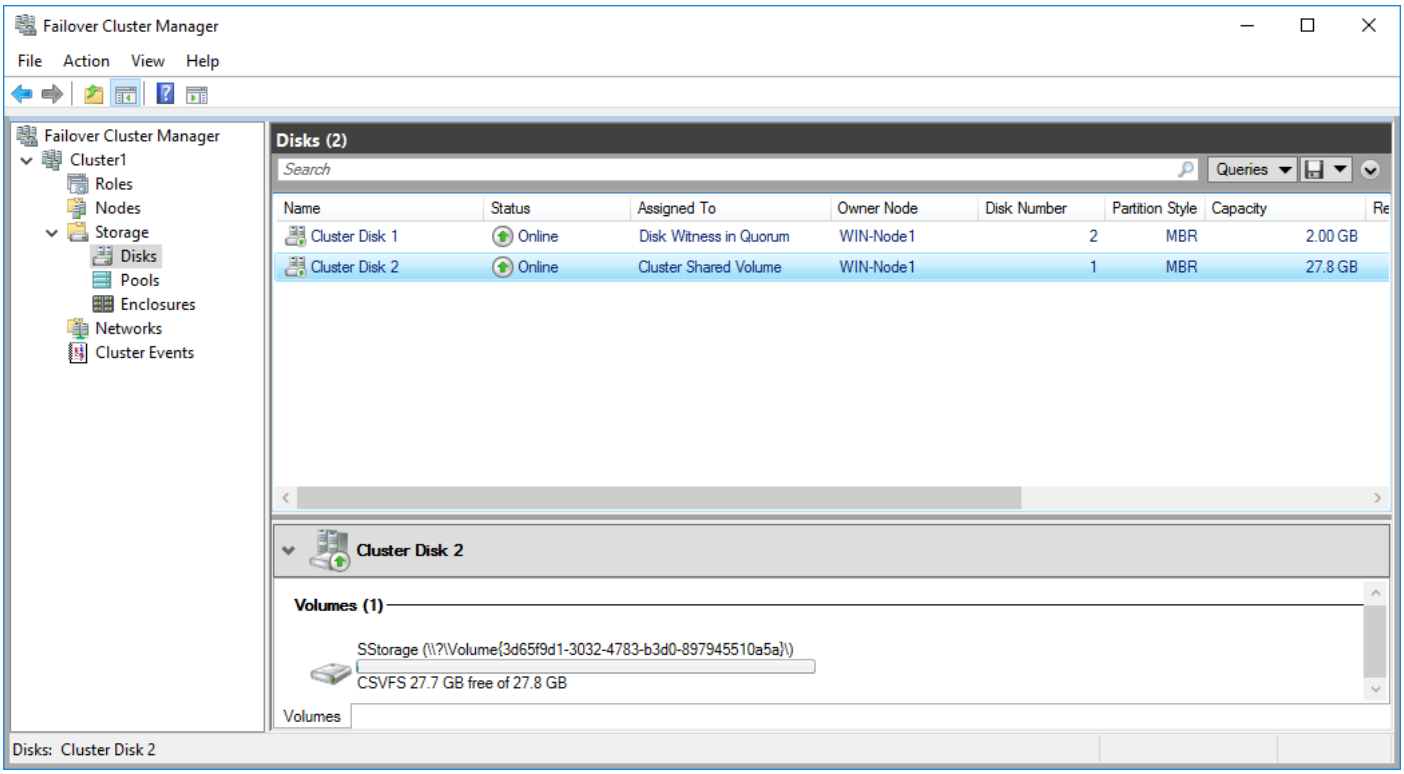

Figure 96

Click **Nodes** to view the status of your Hyper-V hosts *(see Figure 97).*

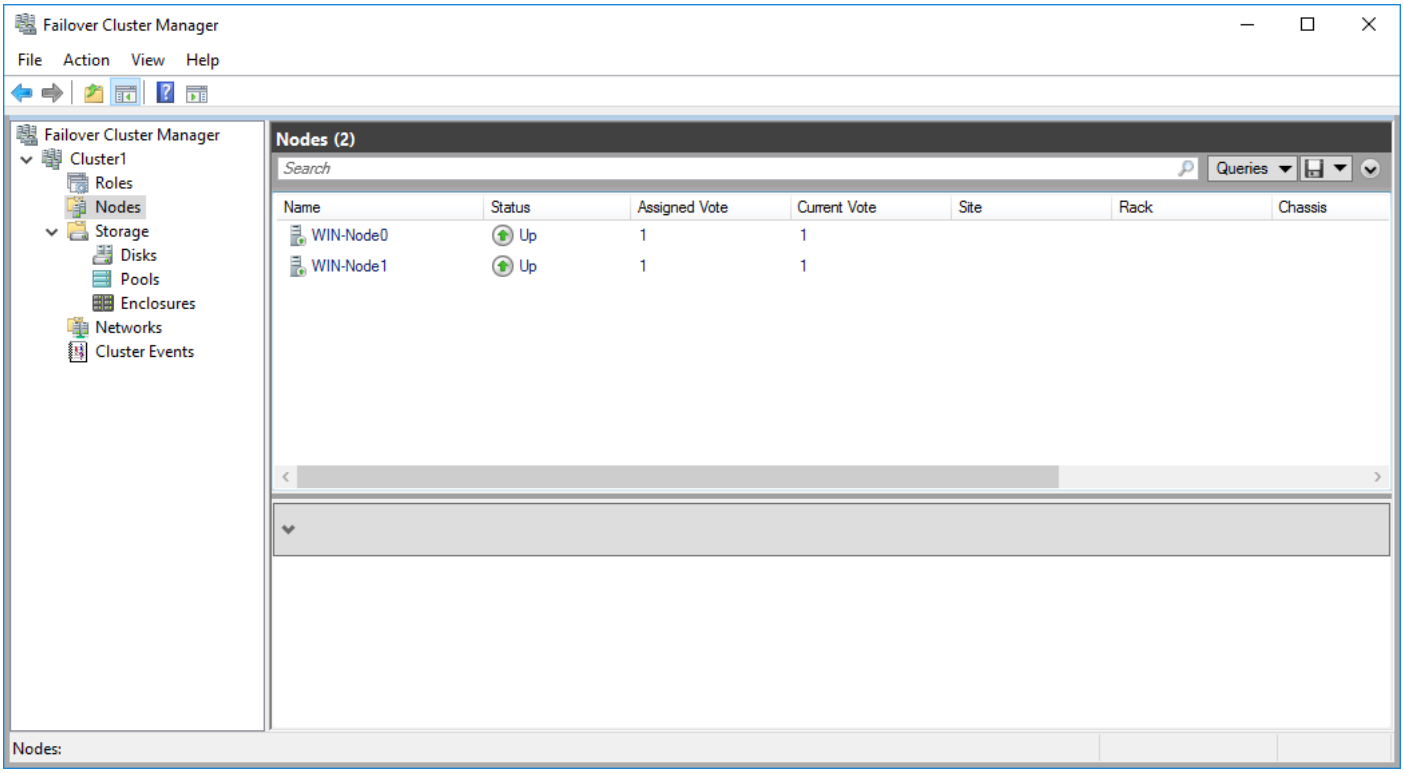

Figure 97

Adding VMs

Now, let's create a highly available virtual machine in the failover cluster. Right-click **Roles** and select **Virtual Machines** > **New Virtual Machine** *(see Figure 98).*

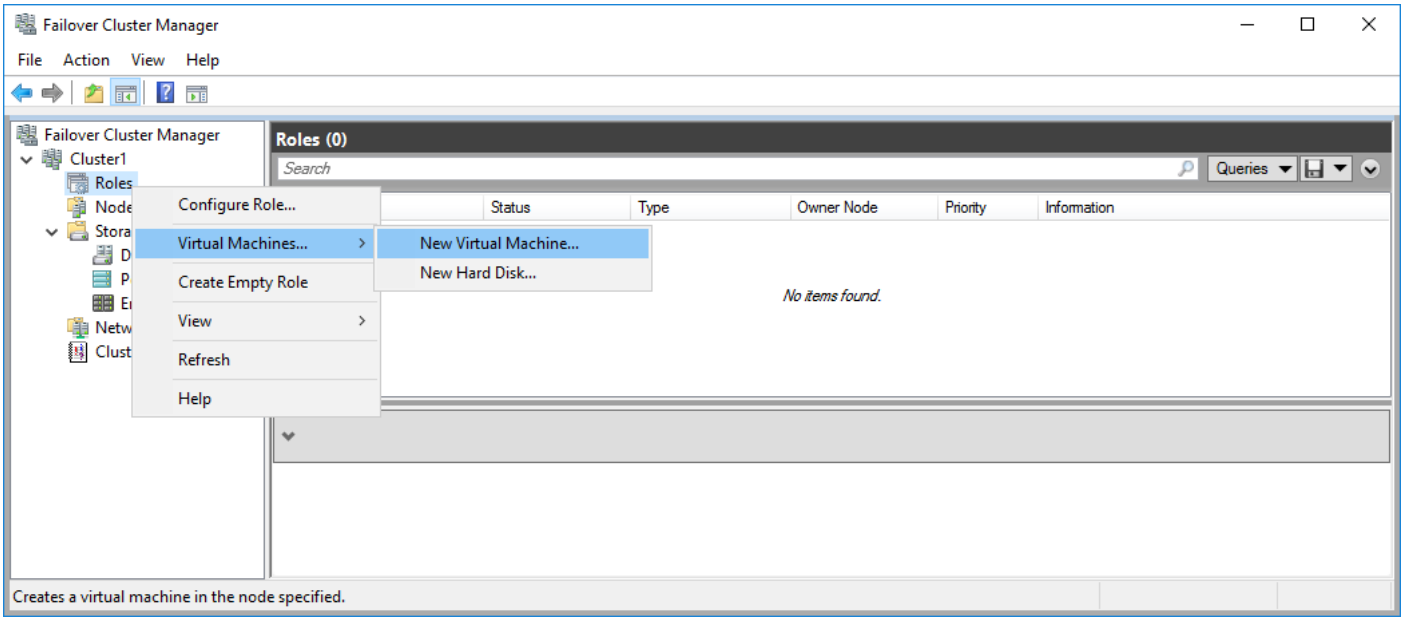

Figure 98

Select the node on which you want the VM to run *(see Figure 99).*

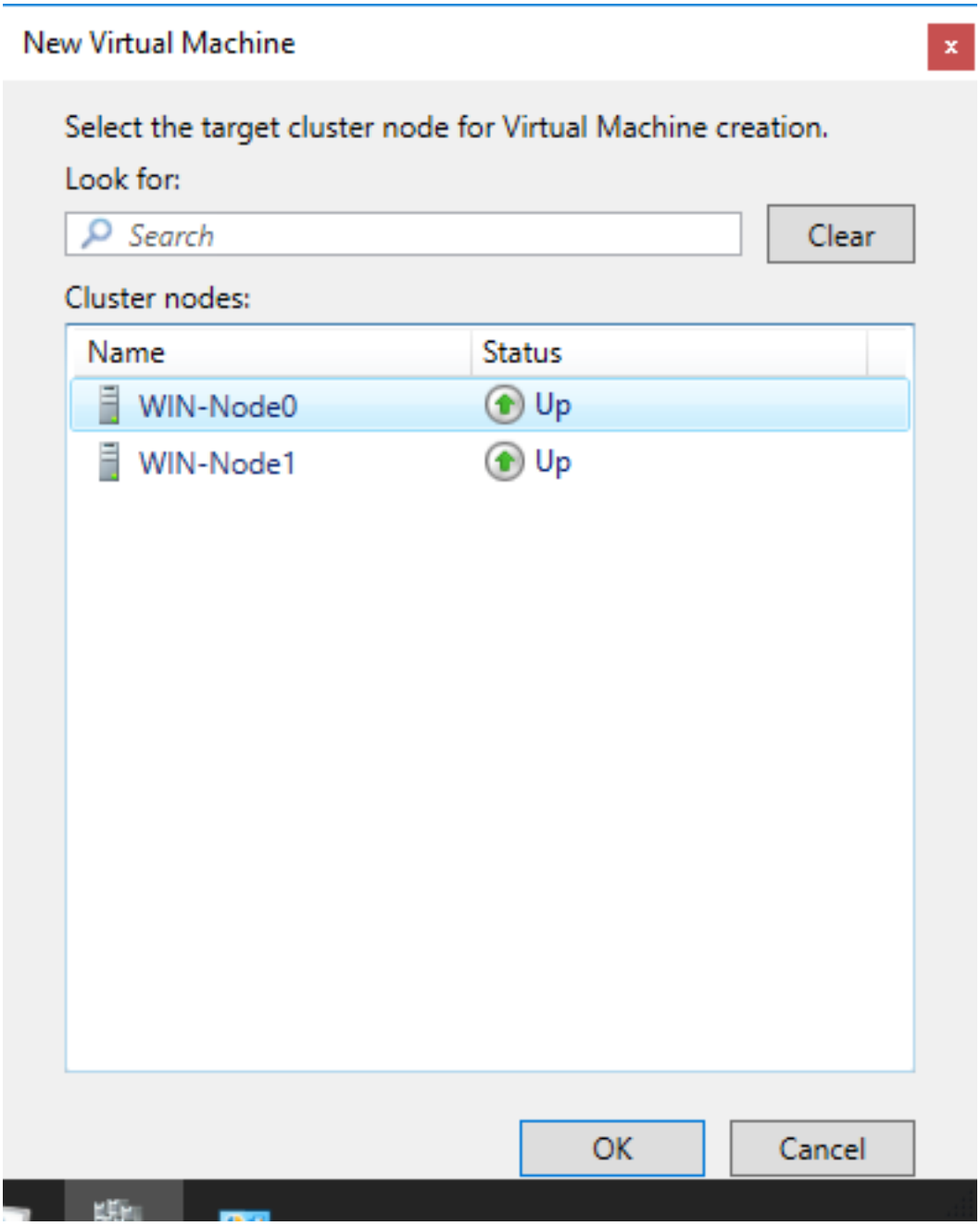

Figure 99

The New Virtual Machine Wizard is launched. Specify the name and location of the VM. Store the VM on the CSV that you created earlier (following the instructions in the previous chapter of this walkthrough). In this example, the path to CSV is *C:\ClusterStorage\Volume1 (see Figure 100).*

Click **Next** when you are finished.

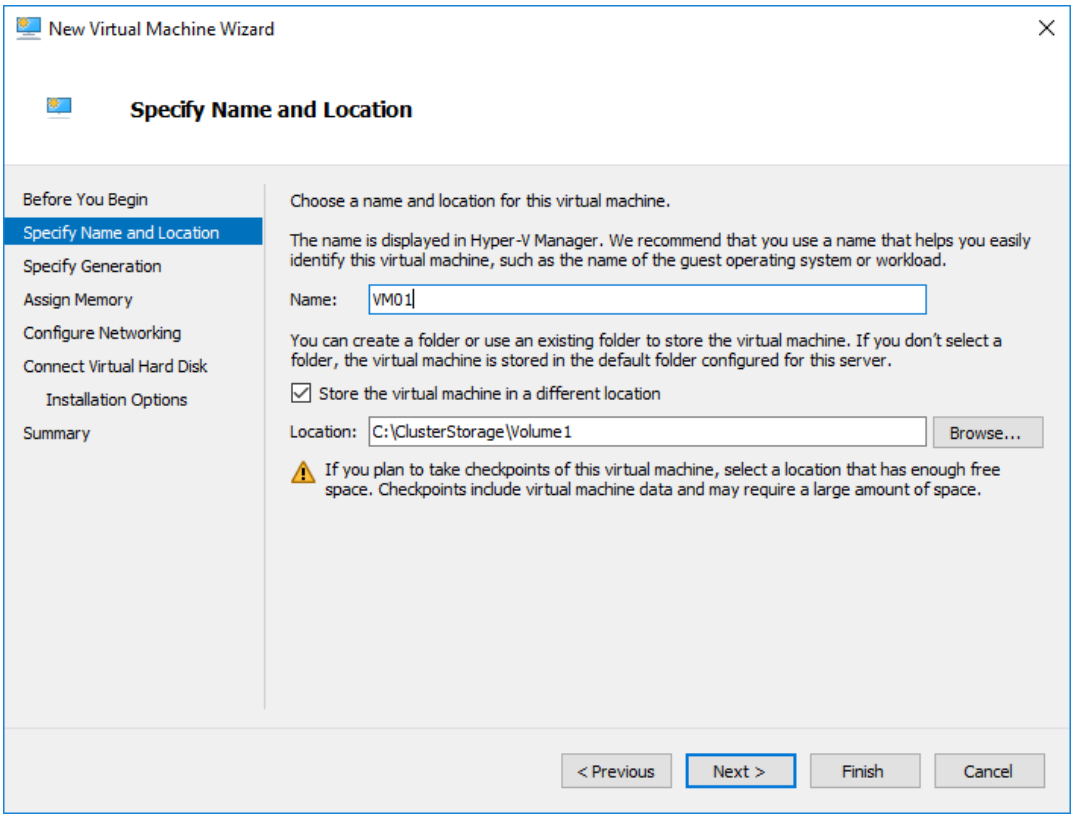

Figure 100

You can set a default location for VMs in the Hyper-V Manager settings for greater convenience. In order to do this, open the Hyper-V settings, then edit the *Virtual Hard Disks* and *Virtual Machines* options as shown in the screenshot below *(see Figure 101).* In this example, the old location was C:\VM and the new one specified is *C:\ClusterStorage\Volume1.*

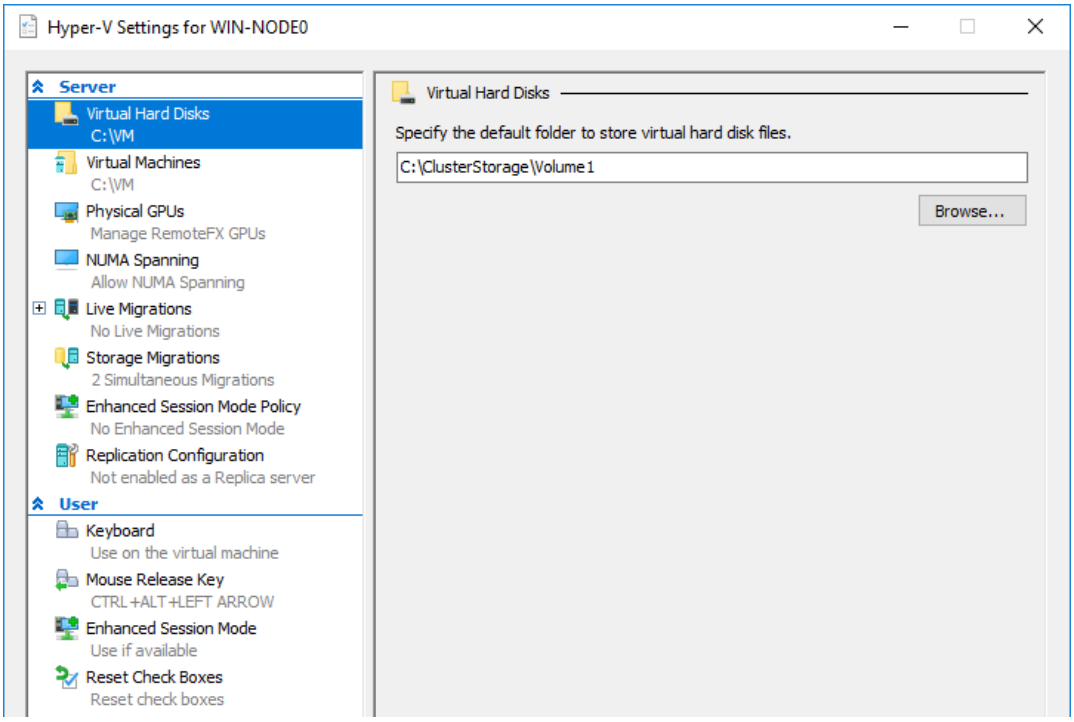

Figure 101

Specify the VM generation *(see Figure 102)*. Then click **Next.**

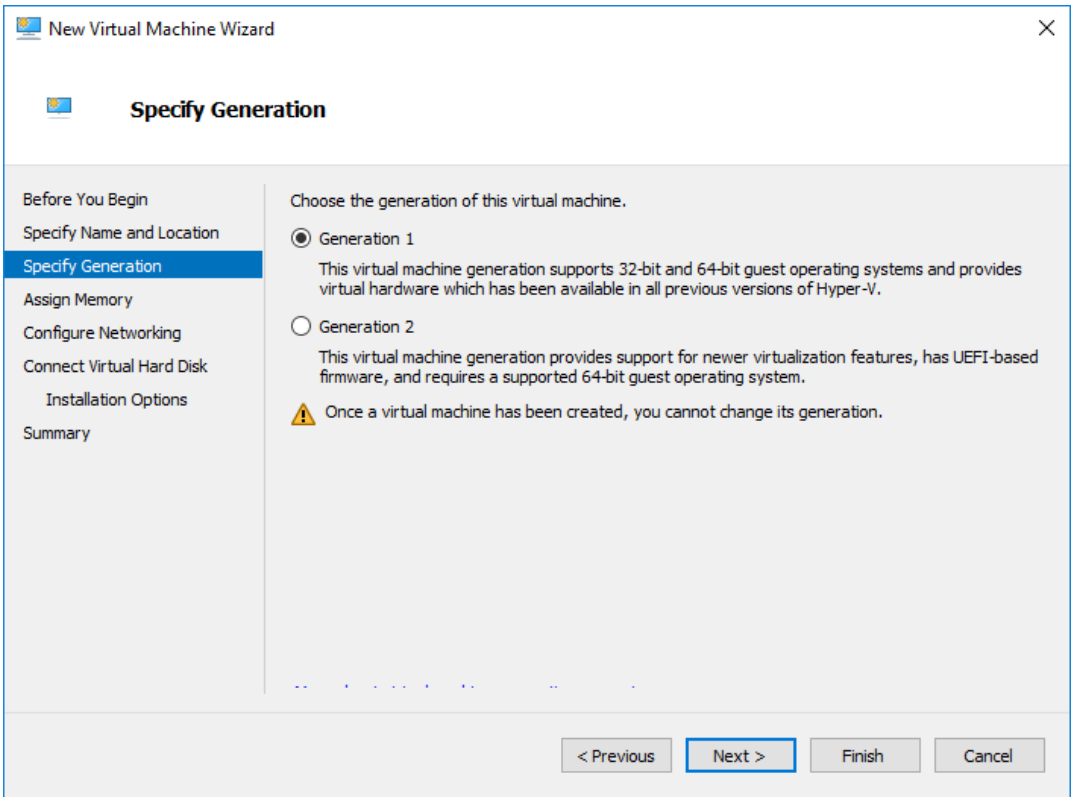

Figure 102

Input the amount of memory that you wish to allocate and click **Next** *(see Figure 103).*

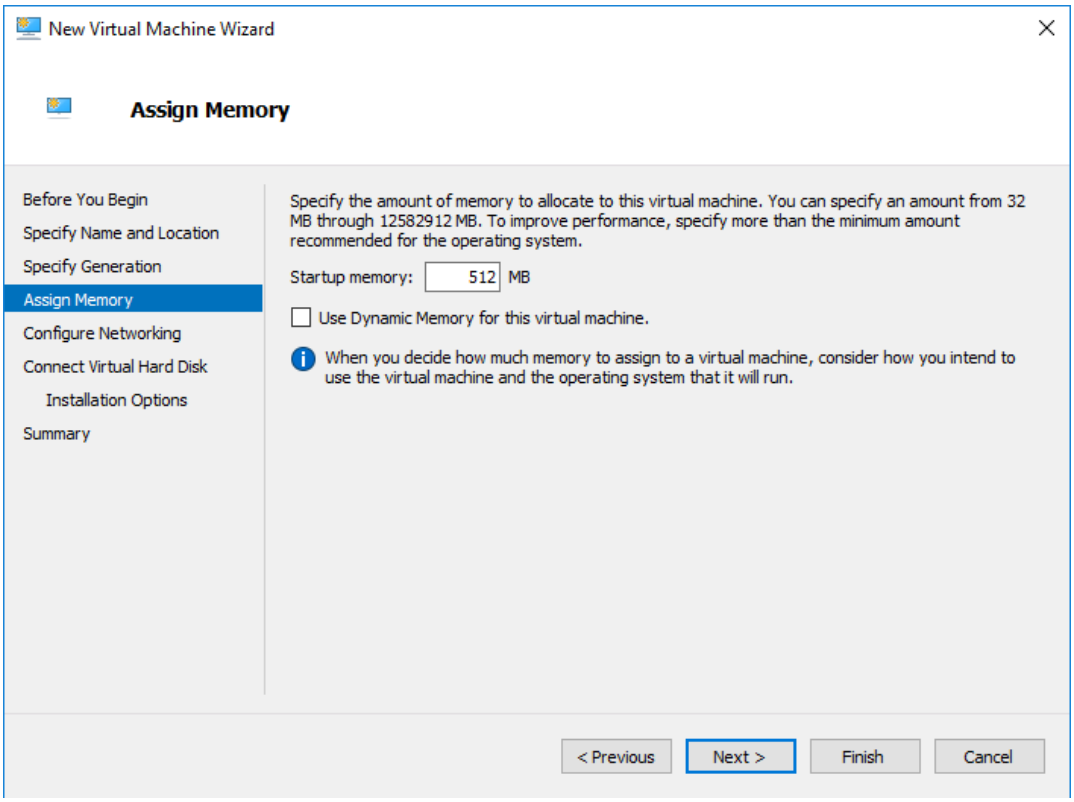

Figure 103

Select the virtual switch to which your VM should be connected *(see Figure 104).*

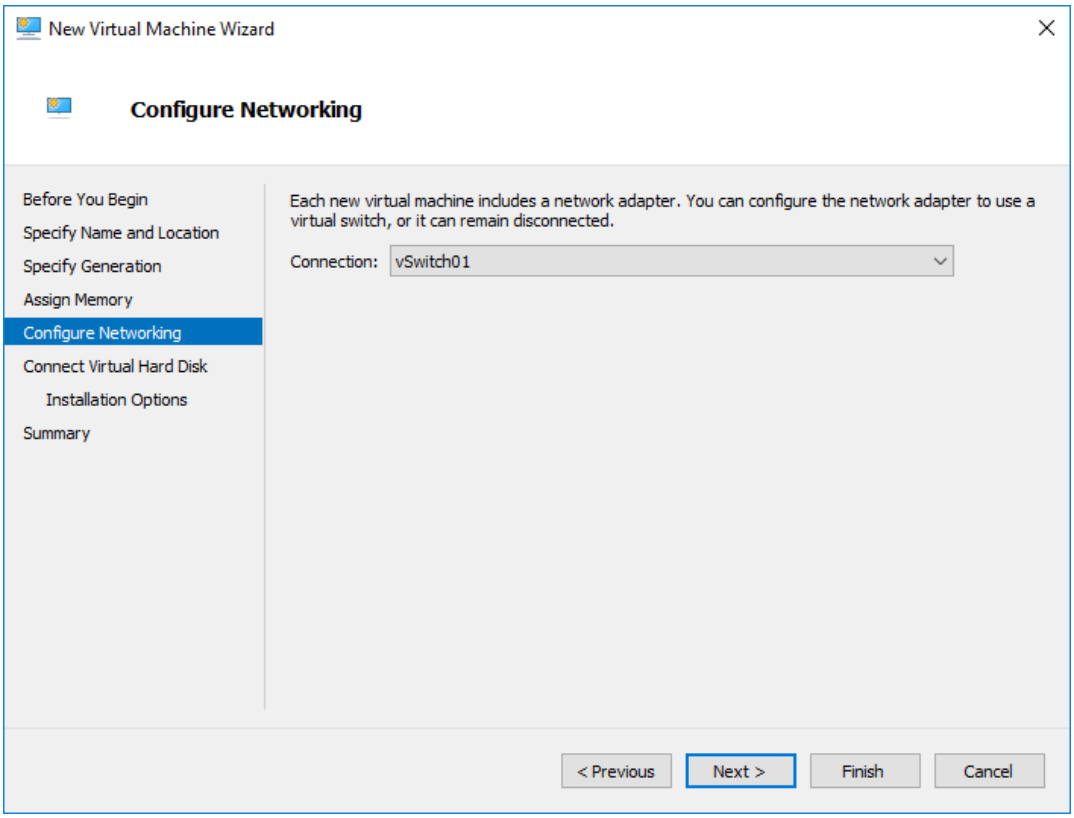

Figure 104

Set the name, location, and size of the virtual disk *(see Figure 105).* The virtual disk must be located on the CSV.

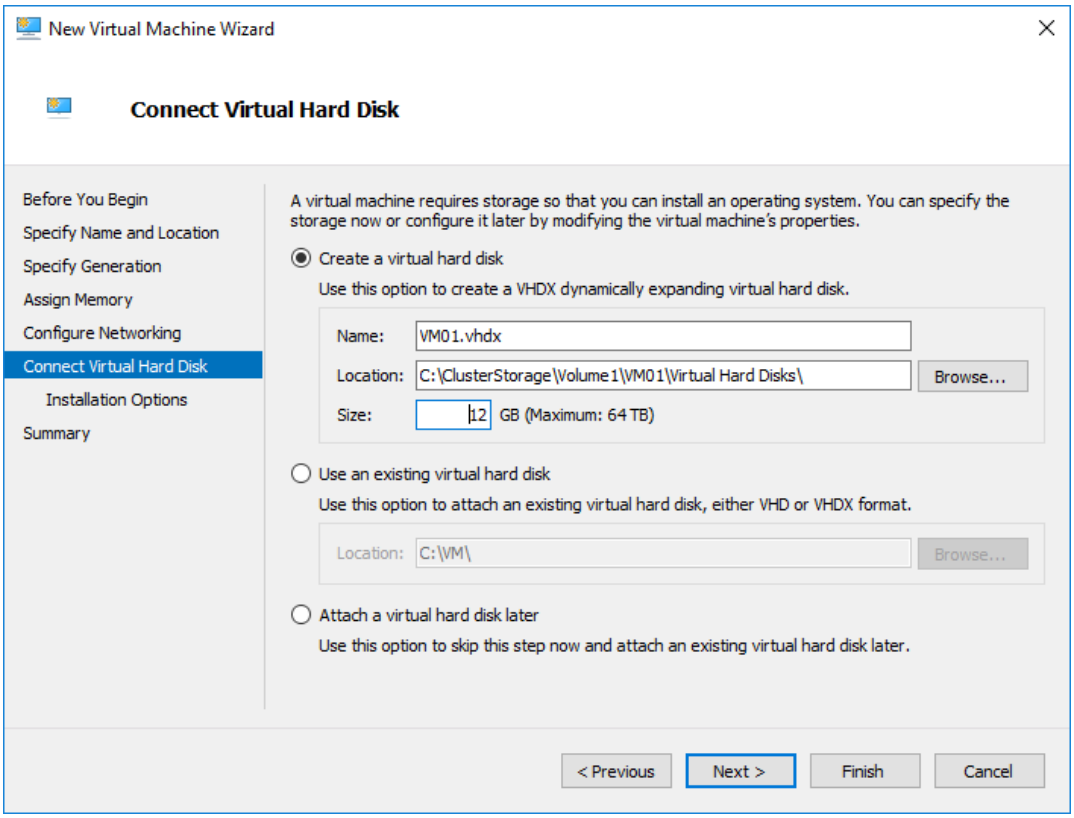

Figure 105

Select the installation options for the VM you have created *(see Figure 106)*. Click **Next.**

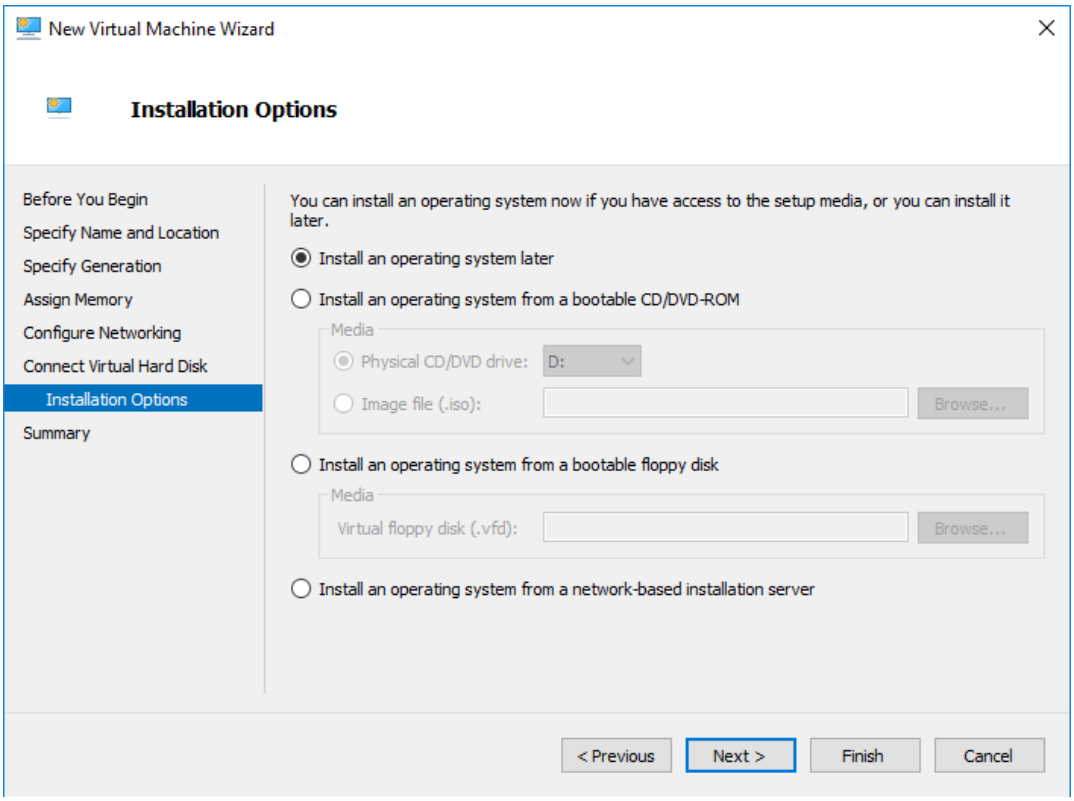

Figure 106

Check the summary and click **Finish** if everything is suitable *(see Figure 107).*

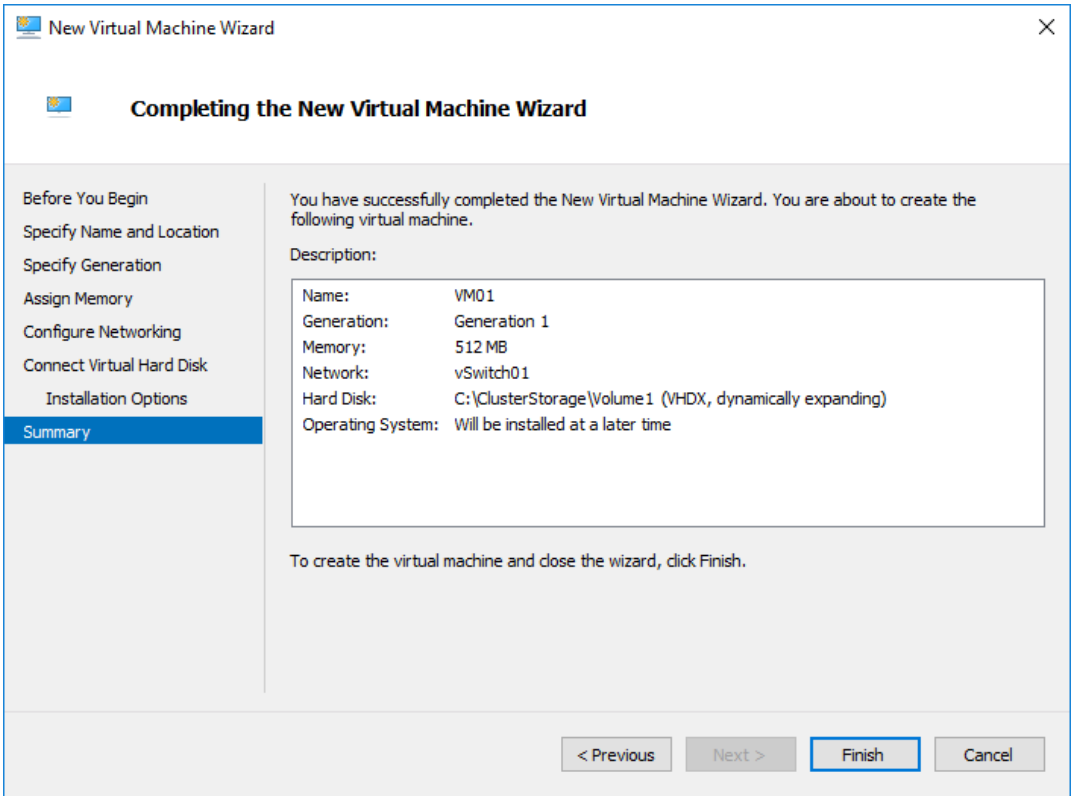

Figure 107

Now your VM can be migrated (failed over) if the host on which the VM is running fails. The following screenshot shows two VMs. One of them is running and the other is stopped *(see Figure 108).*

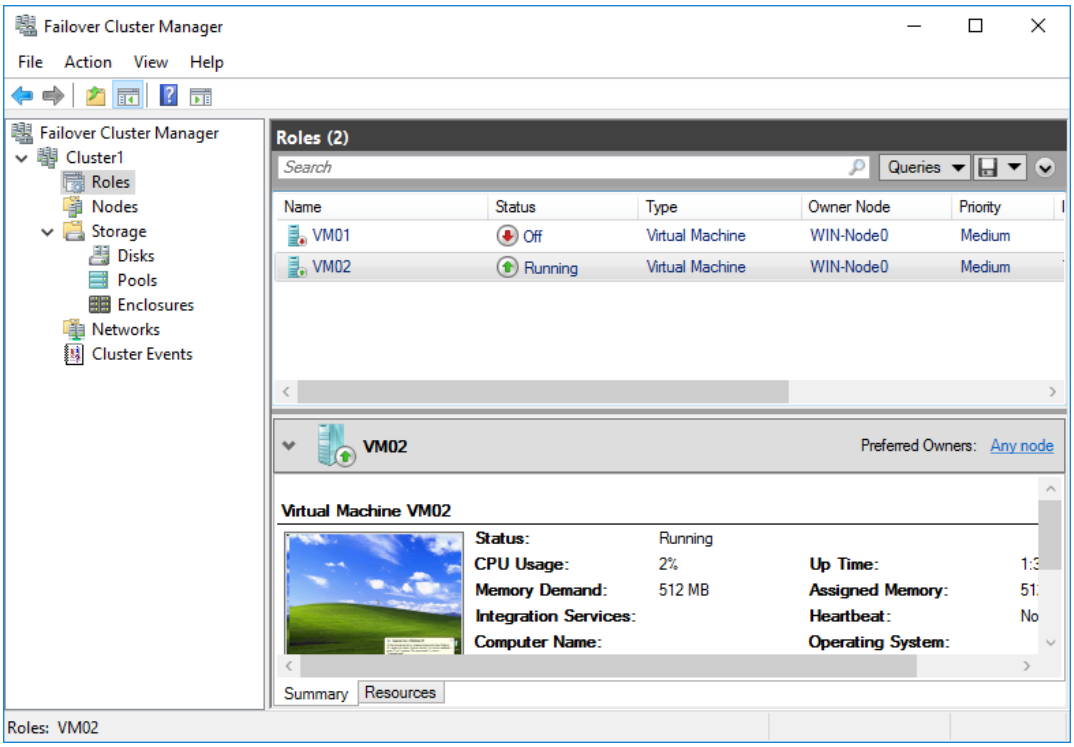

Figure 108

You can also migrate the VMs between the hosts manually. To do this, right-click a VM and select **Move**. Live migration, quick migration or VM storage migration can be performed *(see Figure 109).*

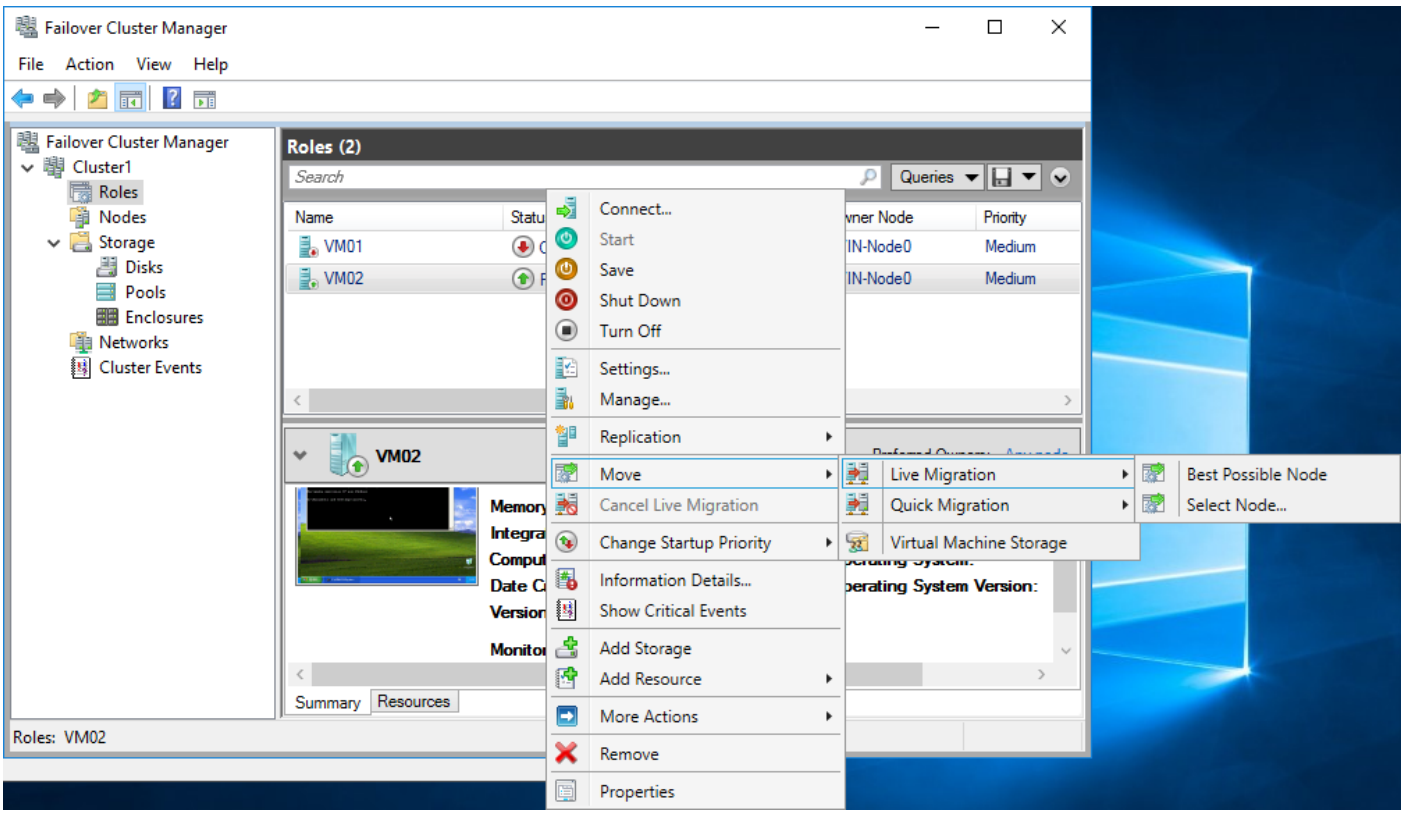

Figure 109

Your Hyper-V Failover Cluster is now functional and ready to use. You can change the settings at any time, as needed.

You can configure the network settings for any of your networks used in the cluster. Open the Failover Cluster Manager, select your cluster, and click **Networks**. You can right-click one of the networks and open the *Properties* menu *(see Figure 110)*. There, you can allow or deny cluster communications and connections to clients. In the right pane, you can click **Live Migration Settings** if you want to select networks for live migration and manage the priority settings.

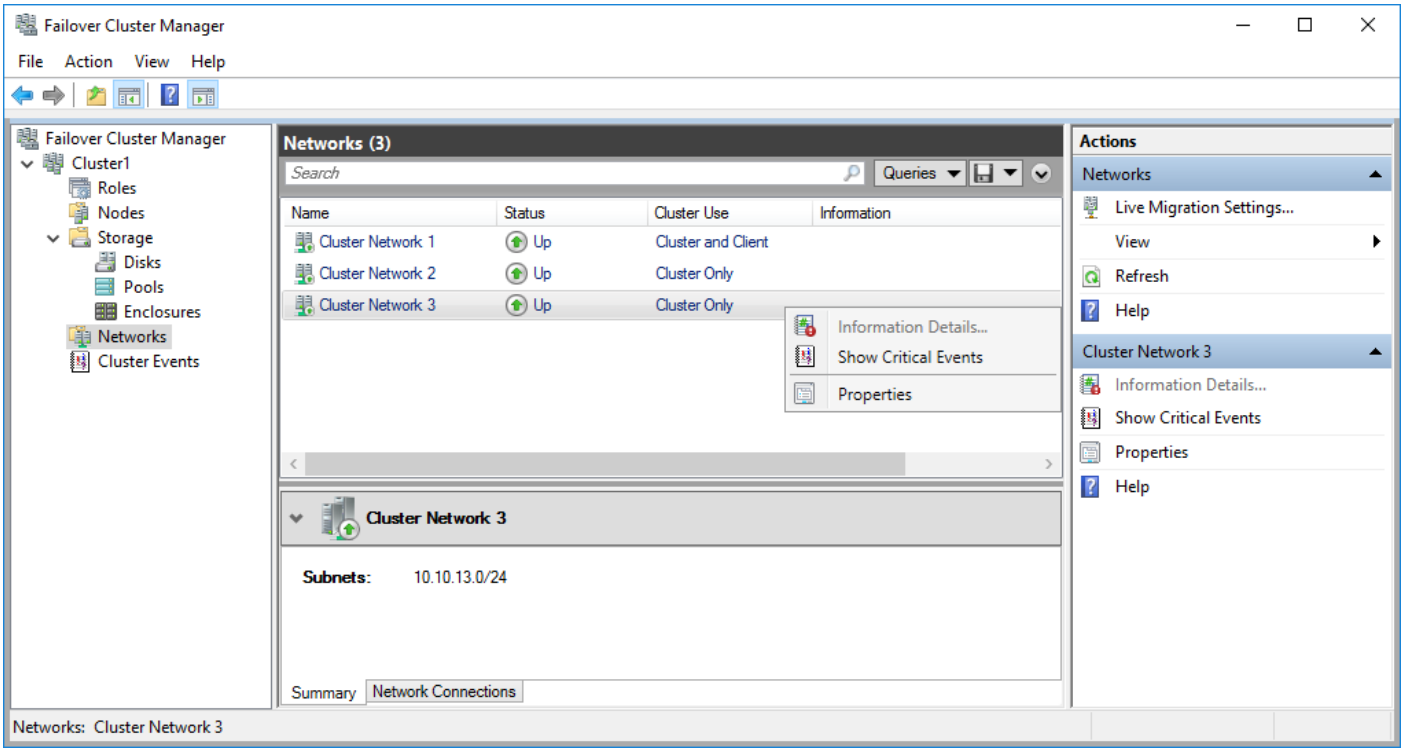

Figure 110

The quorum settings can also be changed. In order to change the quorum settings, right-click your cluster, and from the context menu select **More Actions** > **Configure Cluster Quorum Settings** *(see Figure 111).*

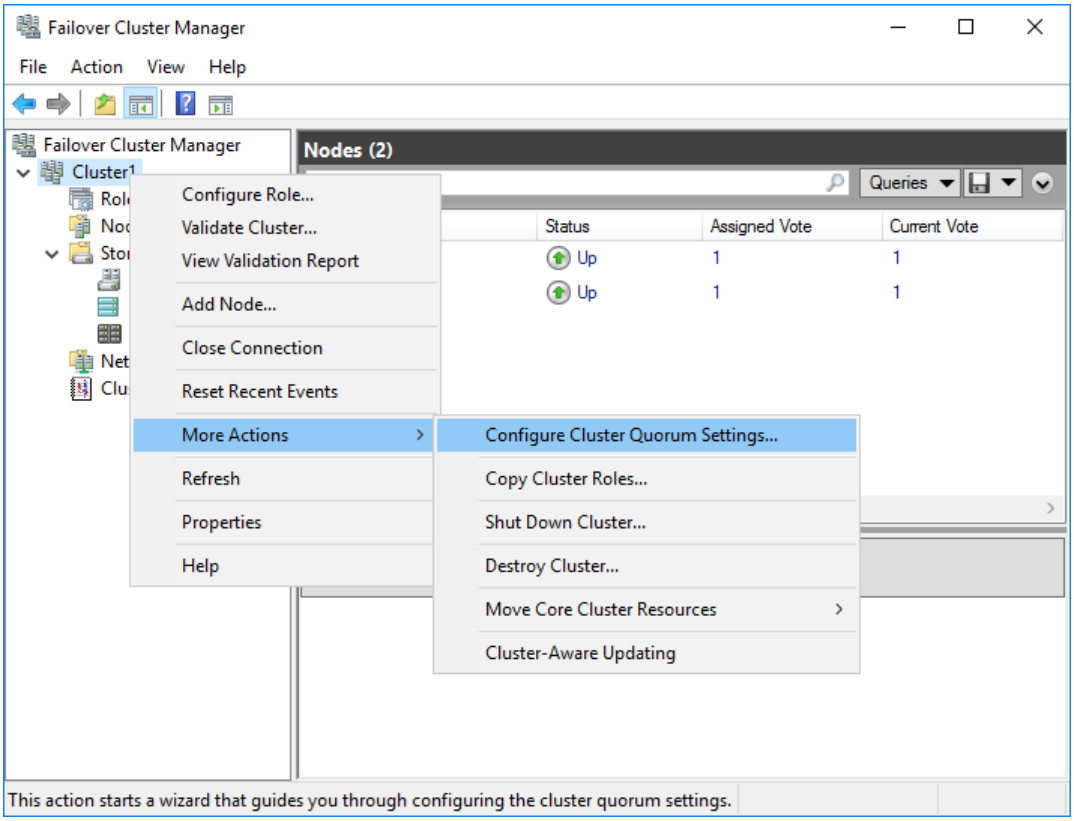

Figure 111

For remote connection to the cluster from another machine, open the Failover Cluster Manager (open **Server Manager**, click **Tools** > **Failover Cluster Manager**), right-click the Failover Cluster Manager, and select **Connect to Cluster** *(see Figure 112).*

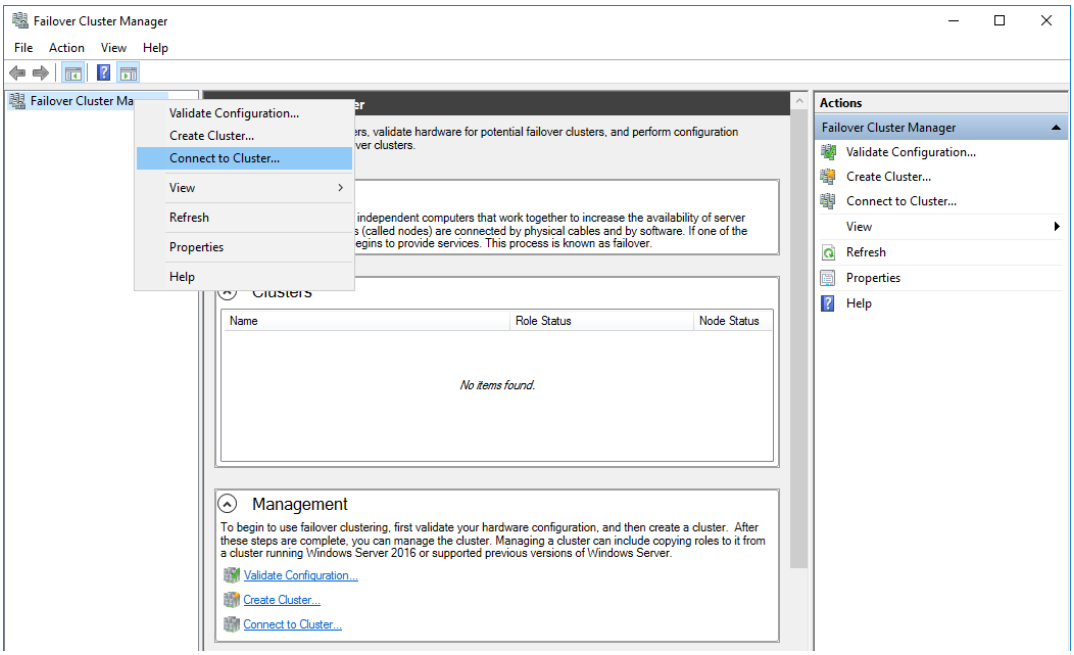

Figure 112

Enter the name of your cluster or the IP address that you specified for cluster administration *(see Figure 113).*

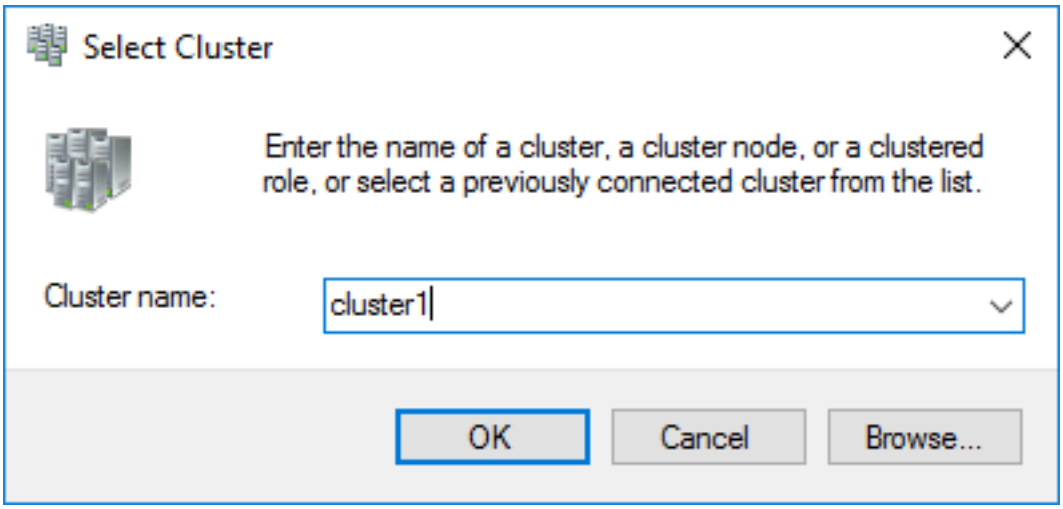

Figure 113

Cluster management using the Failover Cluster Manager was explained above, but there is an alternative way to manage Hyper-V clusters, which is explained in the following section.

System Center Virtual Machine Manager (SCVMM)

What is SCVMM?

System Center Virtual Machine Manager (SCVMM) is a Microsoft tool for centralized management of Hyper-V virtual environments *(see Figure 114)*. You can add Hyper-V Hosts and clusters (including the VMs) to what is called the **Fabric**. With SCVMM, you can manage hosts, clusters, virtual machines, virtual networks, and other infrastructure objects. A special agent needs to be installed on Hyper-V hosts that are managed by SCVMM.

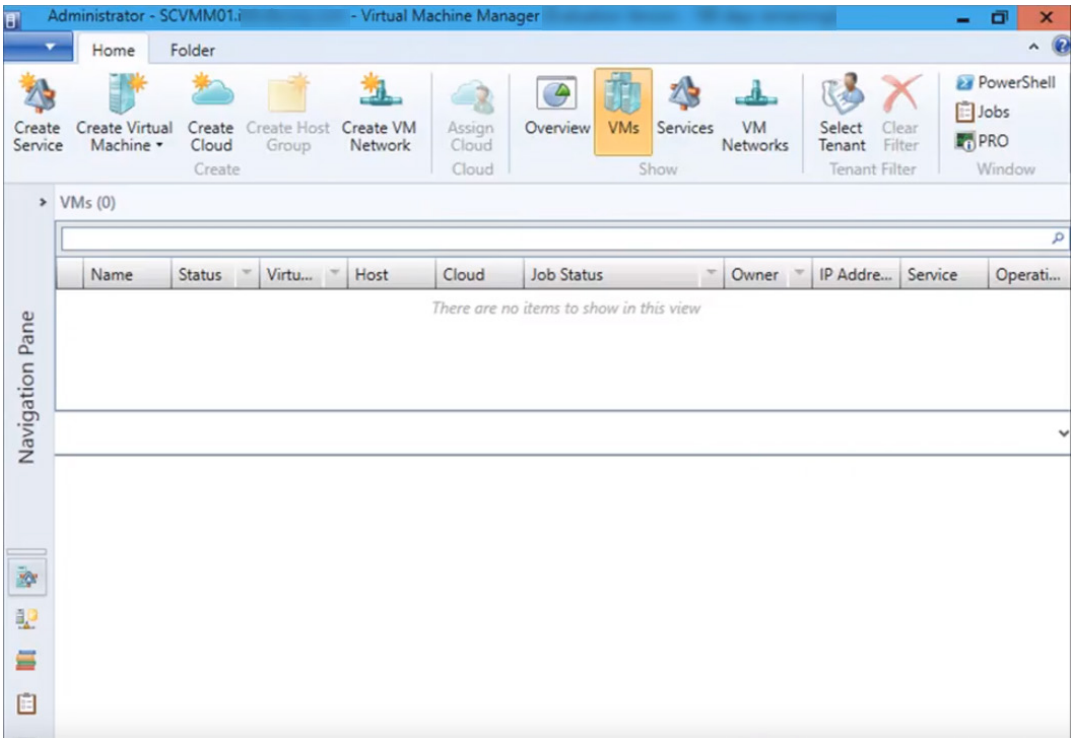

Figure 114

Using SCVMM

Using System Center Virtual Machine Manager for centralized VM management offers several advantages for large virtual environments and optimizes performance as well as resources. VM orchestration and cluster rolling upgrades are convenient with SCVMM; the platform has an administration console with a graphical user interface (GUI). Integration with PowerShell helps automate the processes.

How to deploy SCVMM as a Hyper-V VM

SCVMM can be deployed as a virtual machine. First, download an archived VM from Microsoft's site (the archive is divided into multiple files – *see Figure 115*). Then run *SC2016_SCVMM_VHD.exe* to extract the VHD file.

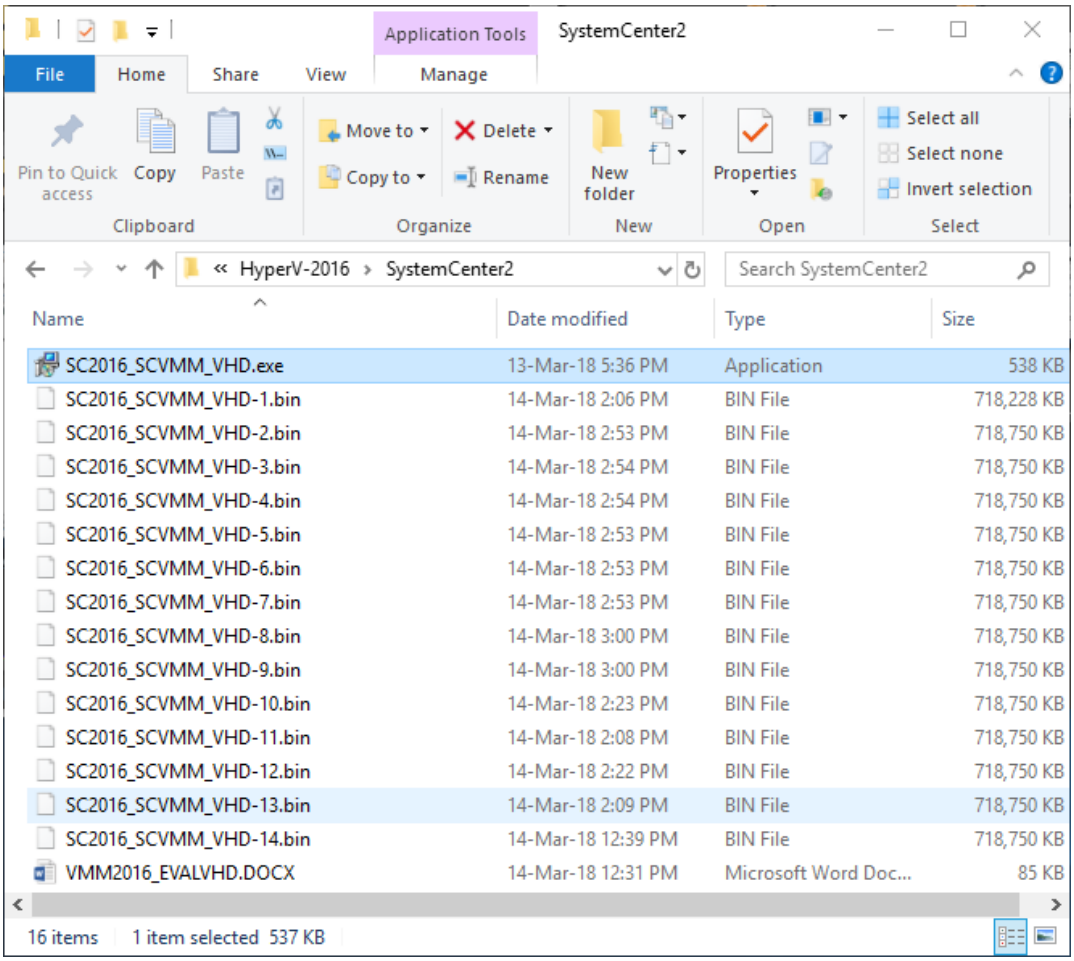

Figure 115

The main steps for deploying SCVMM are:

- Creating a new VM using the downloaded and extracted VHD file.
- Configuring MS SQL Server inside the VM.
- Configuring Virtual Machine Manager (VMM) inside the VM.

The *VMM2016_EVALHD.DOCX* file that you downloaded with parts of archive includes requirements and a detailed, step-by step installation manual from Microsoft. Follow the procedures outlined in that manual in order to install and configure SCVMM.

Now you know how to deploy, configure, and manage a Hyper-V failover cluster, which ensures high availability for your VMs. These powerful clustering features can protect your virtual infrastructure from long periods of downtime. What they cannot help with is recovery of information when undesired changes happen with a VM: accidental deletion of file, corruption of data by malware, etc. In these cases, you can recover the intact data from backups if you have created them in advance. If you are thinking about ensuring the highest protection level for your VMs, consider performing backup and replication in addition to implementing clustering features.

Protection of a Hyper-V Cluster With NAKIVO Backup & Replication

NAKIVO Backup & Replication is a software product developed especially for making backup and replication in virtual infrastructures fast, easy, and reliable. NAKIVO Backup & Replication provides the following advantages to help you perform backup and replication of your particular VMs as well as entire clusters:

- Image-based, host-level backup of each VM in the cluster. No agents need to be installed, because backup is performed at the hypervisor level. Configuring MS SQL Server inside the VM.
- Application-aware backup, which means you can create transactionally consistent backups of your VMs with MS SQL, MS Exchange, or the Active Directory Domain controller running.
- Support for Hyper-V Cluster Shared Volumes (CSVs).
- The ability to add an entire cluster (with all its VMs) to your inventory in just a few clicks. Clusters can include high numbers of VMs, and with NAKIVO Backup & Replication you don't need to add each one manually. This eliminates the risk of forgetting one. Furthermore, when new VMs are added to the cluster, they are automatically protected under the relevant rules. *(Note: you can still manually exclude VMs in the cluster that should not be backed up.)*
- Automatic tracking of VMs that migrate between cluster nodes. VMs can change hosts after migrations caused by failover or load balancing events. NAKIVO Backup & Replication tracks such VMs to ensure they are always backed up or replicated.
- Powerful features that help save storage space:
  - Incremental backup copies only the data changed on the VM since the previous backup.
  - Repository-wide deduplication excludes redundant blocks from backups. Compression then further reduces the backup size.
  - Swap files, which are unnecessary for backups, are skipped.
- Backup-to-cloud, including Microsoft Azure, helps you store remote copies of your backups and increases your total level of data protection.
- Automated backup verification lets you rest assured that your backed up data is consistent. With this feature, you can don't have to worry about backup data being corrupted; you can receive automated email reports with screenshots of test-booted OSes from your backups.
- Full VM recovery as well as instant file and object recovery. You can recover a whole VM, but sometimes you may need to recover particular files or objects such as MS SQL, MS Exchange, or Active Directory objects. NAKIVO Backup & Replication offers you granular recovery as well as full VM recovery. You can also recover VMs across platforms (e.g., Hyper-V to VMware, or vice versa).
- Site Recovery. Create custom site recovery workflows for automated disaster recovery of your virtual infrastructure, including your clusters.

Conclusion

This eBook has covered the methods of Hyper-V virtual machine protection, one of which is to use a Hyper-V failover cluster. These clusters ensure high availability (HA) for your VMs, significantly reducing VM downtime in the event of Hyper-V host failure by migrating the failed host's VMs to other hosts.

To deploy a cluster, you need to configure at least two hosts, block-level or file-level shared storage, separate networks, a quorum, and the VMs that you want to protect with high availability. Hyper-V Server or Windows Server operating system must be used on the hosts.

Clustering provides scalability, flexibility, and rational usage of resources. Backup and/or replication can complement the high availability that clusters offer, helping you recover deleted or corrupted VM data quickly in case of an emergency. Use the advantages of Hyper-V failover clusters and NAKIVO Backup & Replication to maximize the protection level of your VMs.

Glossary

Cluster rolling upgrade is a process of upgrading a Hyper-V cluster from version 2012 to version 2016 without interrupting cluster operation. Nodes must be disconnected, upgraded and connected to a cluster one by one.

Cluster shared volume (CSV) is a feature in Windows Server OS that allows shared disks to be simultaneously accessible for all nodes within a failover cluster.

Fabric (SCVMM) is a couple of infrastructure elements intended for running Hyper-V virtual machines.

Fault tolerance (FT) is a capability of a computer system (cluster, network, cloud etc.) to continue operating without interruption if some of the system components fail.

High availability (HA) is a quality of infrastructure that allows to minimize downtime of computers, systems, and services. High availability ensures a high operational performance level for a period of time.

Internet Control Message Protocol (ICMP) is a supporting protocol in the Internet protocol suite. It is used by network devices, including routers, to send error messages and operational information indicating, for example, that a requested service is not available or that a host or router could not be reached.

Internet Small Computer System Interface (iSCSI) is a transport layer protocol developed for transporting SCSI commands over TCP/IP networks.

Network interface controller (NIC) is a network card used for connecting a device to the network. NIC can be built into the motherboard or represent a separate compatible circuit board that is attached to a motherboard via compatible interfaces.

Node is a host added to a cluster.

PowerShell is a task automation and configuration management framework from Microsoft, consisting of a command-line shell and associated scripting language.

Quorum is defined as the majority of votes required for a cluster to function properly according to the Windows clustering models.

Redundant Array of Independent Disks (RAID)

– RAID 1 provides disk mirroring. Data on one disk is the same as data on another disk. Protects against one disk failure.

– RAID 10 is a mix of RAID 1 and RAID 0 (RAID 1+0). Disks are mirrored and then striped. At least 4 disks are required to build RAID 10. Protects against failure of one disk or multiple disks if they belong to different mirrors.

System Center Virtual Machine Manager (SCVMM) is Microsoft's multi-server Hyper-V management tool.

Split-brain situation is a situation in which the source and failover VMs run simultaneously while different clients connect and write changes to different VMs. This is problematic because, after losing connectivity, each node can consider itself a master node as well as the nodes might not be able to recover a healthy cluster state after restoring connectivity. This is prevented with the help of quorums.

Server message block (SMB) or **Common Internet File System (CIFS)** is a common file-level protocol used by Windows systems and shared storage. SMB 3.0 is the latest version of this protocol that is used for file-level storage.

Host Bus Adapter (HBA) card is a circuit board that must be inserted into the appropriate slot (PCI Express, for example) of the motherboard for physical connectivity between a host/server and storage/network device. Different compatible interfaces and technologies can be used (Fibre Channel, SCSI, SAS, SATA, Ethernet, etc.).

SMB v3 protocol is the latest version of SMB protocol lets you create shared storage based on standard servers with local disks (HDD and SSD) which can also be used for Hyper-V clustering.

Resilient File System (ReFS) is a Microsoft proprietary file system introduced in Windows Server 2012, intended to be the "next generation" file system after NTFS.

New Technology File System (NTFS) is a file system created by Microsoft and widely used in Windows-based systems.

Address Resolution Protocol (ARP) is a protocol for mapping an IP address to a physical network adapter address that is recognized in the local network (e.g., a MAC or Media Access Control address).

NAKIVO Backup & Replication at a Glance

NAKIVO Backup & Replication is a fast, reliable, and affordable VM backup solution. The product protects VMware, Hyper-V, and AWS EC2 environments. NAKIVO Backup & Replication offers advanced features that increase backup performance, improve reliability, and speed up recovery. As a result, you can save time and money.

Deploy in under 1 minute
Pre-configured VMware VA and AWS AMI; 1-click deployment on ASUSTOR, QNAP, Synology, NETGEAR, and WD NAS; 1-click Windows installer, 1-command Linux installer

Reduce backup size
Exclusion of swap files and partitions, global backup deduplication, adjustable backup compression, support for deduplication applianes

Protect VMs automatically
Native, agentless, policy-based backup and replication for VMware, Hyper-V, and AWS VMs

Ensure recoverability
Instant backup and replica verification with screenshots of test-recovered VMs, backup copy offsite/to the cloud, cross-platform recovery

Increase backup speed
Incremental backup with CBT/RCT, LAN-free data transfer, network acceleration, up to 2X performance increase when installed on NAS

Decrease recovery time
Instant recovery of VMs, files, application objects back to source; site recovery for DR orchestration and automation

About NAKIVO

The winner of a “Best of VMworld 2018” and the Gold Award for Data Protection, NAKIVO is a US-based corporation dedicated to developing the ultimate VM backup and site recovery solution. With 20 consecutive quarters of double-digit growth, 5-star online community reviews, 97.3% customer satisfaction with support, and more than 10,000 deployments worldwide, NAKIVO delivers an unprecedented level of protection for VMware, Hyper-V, and Amazon EC2 environments.

As a unique feature, NAKIVO Backup & Replication runs natively on leading storage systems including QNAP, Synology, ASUSTOR, Western Digital, and NETGEAR to deliver up to 2X performance advantage. The product also offers support for high-end deduplication appliances including Dell/EMC Data Domain and NEC HYDRAstor. Being one of the fastest-growing data protection software vendors in the industry, NAKIVO provides a data protection solution for major companies such as Coca-Cola, Honda, and China Airlines, as well as working with over 3,300 channel partners in 140 countries worldwide.

© 2019 NAKIVO, Inc. All rights reserved. All trademarks are the property of their respective owners.