vembu

Understanding Windows Server Hyper-V Cluster Configuration, Performance and Security

Author

Brandon Lee

Brandon Lee has been in the IT industry for over 15+ years now and has worked in various IT industries spanning education, manufacturing, hospitality, and consulting for various technology companies including Fortune 500 companies. He is a prolific blogger and contributes to the community through various blog posts and technical documentation primarily at Virtualizationhowto.com

Understanding Hyper-V Cluster Configuration Performance and Security

1. Windows Server Failover Clustering Overview
    - Windows Server Failover Clusters Hyper-V Specific Considerations
2. Hyper-V Configuration Best Practices
    - Use Hyper-V Core installations
    - Sizing the Hyper-V Environment Correctly
    - Network Teaming and Configuration
    - Storage Configuration
3. Hyper-V Networking Best Practices
    - Physical NIC considerations
    - Windows and Virtual Network Considerations
4. What is a Hyper-V Virtual Switch?
    - Hyper-V Virtual Switch Capabilities and Functionality
    - Hyper-V Logical Switches
    - Creating Hyper-V Virtual Switches
5. Hyper-V advanced virtual machine network configuration
    - Virtual Machine Queue (VMQ)
    - IPsec Task Offloading
    - SR-IOV
    - DHCP Guard, Router Guard, Protected Network and Port Mirroring
6. Hyper-V Design Considerations with iSCSI Storage
    - Hyper-V Windows Configuration for iSCSI
    - Verifying Multipathing
7. What is Windows Server 2016 Storage Spaces Direct?
    - Windows Server 2016 Storage Spaces Direct Requirements
    - Windows Server 2016 Storage Spaces Direct Architecture
    - Windows Server 2016 SAN vs Storage Spaces Direct
8. Why Use Hyper-V VHDX File Format?
    - Optimizing VHDX Virtual Disk Files
    - Resizing VHDX Virtual Disk Files
9. Troubleshooting Hyper-V with Event Logs
    - Taking Hyper-V Troubleshooting with Event Viewer Further
    - System Center Virtual Machine Manager Logging
10. System Center Virtual Machine Manager SCVMM – Overview
    - System Center Virtual Machine Manager SCVMM – Features
    - Managing Hyper-V Host and Clusters with SCVMM
    - Is System Center Virtual Machine Manager Required?

What is Windows Server Failover Clustering?

Windows Server Failover Clustering is the mechanism that allows running Windows Roles, Features, and applications to be made highly available across multiple Windows Server hosts. Why is making roles, features, and other applications across multiple hosts important? Clustering helps to ensure that workloads are resilient in the event of a hardware failure. Especially when thinking about virtualized workloads, often multiple virtual machines are running on a single host. If a host fails, it is not simply a single workload that fails, but possibly many production workloads could be taken offline with dense configurations of virtual machines on a single host.

The Windows Server Failover Cluster in the context of Hyper-V, allows bringing together multiple physical Hyper-V hosts into a “cluster” of hosts. This allows aggregating CPU/memory resources attached to shared storage which in turn allows the ability to easily migrate virtual machines between the Hyper-V hosts in the cluster. The shared storage can be in the form of the traditional SAN or in the form of Storage Spaces Direct in Windows Server 2016.

The ability to easily migrate virtual machines between shared storage allows restarting a virtual machine on a different host in the Windows Server Failover Cluster if the original physical host the virtual machine was running on fails. This allows business-critical workloads to be brought back up very quickly even if a host in the cluster has failed.

Windows Server Failover Clustering also has other added benefits as they relate to Hyper-V workloads that are important to consider. In addition to allowing virtual machines to be highly available when hosts fail, the Windows Server Failover Cluster also allows for planned maintenance periods such as patching Hyper-V hosts.

This allows Hyper-V administrators the ability to patch hosts by migrating virtual machines off a host, applying patches, and then rehydrating the host with virtual machines. There is also Cluster Aware Updating that allows this to be done in an automated fashion. Windows Server Failover Clustering also provides the benefit of protecting against corruption if the cluster hosts become separated from one another in the classic “split-brain” scenario. If two hosts attempt to write data to the same virtual disk, corruption can occur.

Windows Server Failover Clusters have a mechanism called quorum that prevents separated Hyper-V hosts in the cluster from inadvertently corrupting data. In Windows Server 2016, new type of quorum has been introduced that can be utilized along with the longstanding quorum mechanisms – the cloud witness.

Windows Server Failover Clustering Basics

Now that we know what Windows Server Failover Cluster is and why it is important, let's take a look at Windows Server Failover Clustering basics to understand a bit deeper how Failover Clustering in Windows Server works. Windows Server Failover Clustering is a feature instead of a role as Windows Server Failover clustering simply helps Windows Servers accomplish their primary role.

It is also included in the Standard Edition version of Windows Server along with the Datacenter version. There is no feature difference between the two Windows versions in the Failover Clustering features and functionality. A Windows Server Failover Cluster is composed of two or more nodes that offer resources to the cluster as a whole. A maximum of 64 nodes per cluster is allowed with Windows Server 2016 Failover Clusters.

Additionally, Windows Server 2016 Failover Clusters can run a total of 8000 virtual machines per cluster. Although in this post we are referencing Hyper-V in general, Windows Server Failover Clusters can house many different types of services including file servers, print servers, DHCP, Exchange, and SQL just to name a few.

One of the primary benefits as already mentioned with Windows Server Failover Clusters is the ability to prevent corruption when cluster nodes become isolated from the rest of the cluster. Cluster nodes communicate via the cluster network to determine if the rest of the cluster is reachable. This is extremely important as it checks to see if the rest of the cluster is reachable. The cluster in general then performs a voting process of sorts that determines which cluster nodes have the node majority or can reach the majority of the cluster resources.

Quorum is the mechanism that validates which cluster nodes have the majority of resources and have the winning vote when it comes to assuming ownership of resources such as in the event of a Hyper-V cluster and virtual machine data. This becomes glaringly important when you think about the case of an even node cluster such as a cluster with (4) nodes. If a network split happens that allows two of the nodes on each side to only see its neighbor, there would be no majority. Starting with Windows Server 2012, by default each node has a vote in the quorum voting process.

A file or share witness allows a tie breaking vote by allowing one side of the partitioned cluster to claim this resource, thus breaking the tie. The cluster hosts that claim the disk or file share witness perform a SCSI lock on the resource, which prevents the other side from obtaining the majority quorum vote. With odd numbered cluster configurations, one side of a partitioned cluster will always have a majority so the file or share witness is not needed.

Quorum received enhancements in Windows Server 2016 with the addition of the cloud witness. This allows using an Azure storage account and its reachability as the witness vote. A "0-byte" blob file is created in the Azure storage account for each cluster that utilizes the account.

Windows Server Failover Clusters Hyper-V Specific Considerations

When using Windows Server Failover Clusters for hosting the Hyper-V role, this opens up many powerful options for running production, business-critical virtual machines. There are a few technologies to be aware of that specifically pertain to Hyper-V and other workloads. These are the following

- Cluster Shared Volumes
- ReFS
- Storage Spaces Direct

Cluster Shared Volumes

Cluster Shared Volumes or CSVs provide specific benefits for Hyper-V virtual machines in allowing more than one Hyper-V host to have read/write access to the volume or LUN where virtual machines are stored. In legacy versions of Hyper-V before CSVs were implemented, only one Windows Server Failover Cluster host could have read/write access to a specific volume at a time. This created complexities when thinking about high availability and other mechanisms that are crucial to running business-critical virtual machines on a Windows Server Failover Cluster.

Cluster Shared Volumes solved this problem by allowing multiple nodes in a failover cluster to simultaneously have read/write access to the same LUN provisioned with NTFS. This allows the advantage of having all Hyper-V hosts provisioned to the various storage LUNs which can then assume compute/memory quickly in the case of a node failure in the Windows Server Failover Cluster.

ReFS

ReFS is short for “Resilient File System” and is the newest file system released from Microsoft speculated to be the replacement for NTFS by many. ReFS touts many advantages when thinking about Hyper-V environments. It is resilient by nature, meaning there is no chkdsk functionality as errors are corrected on the fly.

However, one of the most powerful features of ReFS related to Hyper-V is the block cloning technology. With block cloning the file system merely changes metadata as opposed to moving actual blocks. This means that will typical I/O intensive operations on NTFS such as zeroing out a disk as well as creating and merging checkpoints, the operation is almost instantaneous with ReFS.

ReFS should not be used with SAN/NFS configurations however as the storage operates in I/O redirected mode in this configuration where all I/O is sent to the coordinator node which can lead to severe performance issues. ReFS is recommended however with Storage Spaces Direct which does not see the performance hit that SAN/NFS configurations do with the utilization of RDMA network adapters.

Storage Spaces Direct

Storage Spaces Direct is Microsoft's software defined storage solution that allows creating shared storage by using locally attached drives on the Windows Server Failover Cluster nodes. It was introduced with Windows Server 2016 and allows two configurations:

- Converged
- Hyper-converged

With Storage Spaces Direct you have the ability to utilize caching, storage tiers, and erasure coding to create hardware abstracted storage constructs that allow running Hyper-V virtual machines with scale and performance more cheaply and efficiently than using traditional SAN storage.

Hyper-V Configuration Best Practices

There are several critical configuration areas that you want to take a look at when thinking about Hyper-V configuration best practices with any environment. We will look more closely at the following configuration areas:

- Use Hyper-V Core installations
- Sizing the Hyper-V Environment Correctly
- Network Teaming and Configuration
- Storage Configuration
- Operating System Patches Uniformity

The above configuration areas represent a large portion of potential Hyper-V configuration mistakes that many make in production environments. Let's take a closer look at the above in more detail to explain why they are extremely important to get right in a production environment and what can be done to ensure you do get them right.

Use Hyper-V Core installations

While traditional Windows administrators love the GUI to manage servers, maintaining GUI interfaces on server operating systems is not really a good idea. It leads to much larger installation bases as well as having to maintain patches and other upgrades simply due to the GUI interface and any security and other vulnerabilities that may present as a result.

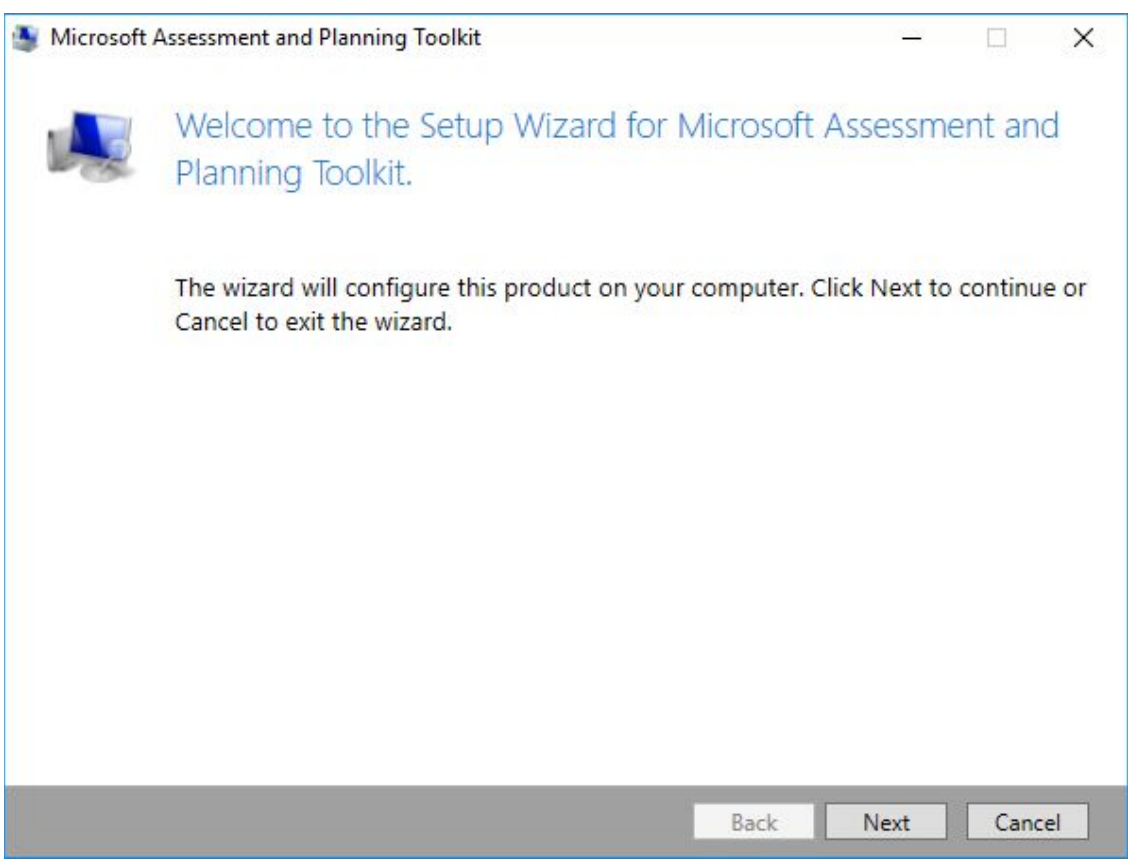
Using the Windows Server 2016 core installation to run the Hyper-V role is certainly the recommended approach to run production workloads on Hyper-V nodes. With the wide range of management tools that can be leveraged with Hyper-V core such as PowerShell remoting, as well as running the GUI Hyper-V manager on another server, it really presents no additional administrative burden to run Hyper-V core with today's tools.

Sizing the Hyper-V Environment Correctly

There are many issues that can come from sizing a Hyper-V environment incorrectly. If a Hyper-V cluster environment is sized too small, performance issues can certainly result due to over-provisioning of resources. Oversizing a Hyper-V environment can certainly be a deterrent from a fiscal standpoint of either being approved for funds on the outset for either a greenfield installation or an upgrade to server resources that are due for a refresh. A final very crucial part of correctly sizing a Hyper-V environment is being able to properly plan for growth in the environment. Every environment in this respect will be different depending on forecast growth.

A great tool that can be utilized to correctly size the needed number of cores, memory, and disk space is the Microsoft Assessment and Planning Toolkit. It can calculate the current cores, memory, and storage being utilized by production workloads in an automated fashion so you can easily gather current workload demands. Then, you can calculate for growth in the environment based on the projected amount of new server resources that will need to be provisioned in the upcoming future.

The Microsoft Assessment and Planning Toolkit can be downloaded here: https://www.microsoft.com/en-us/download/details.aspx?id=7826



The Microsoft Assessment and Planning Toolkit allows sizing new Hyper-V environments based on current workloads

Network Teaming and Configuration

Hyper-V network design is an extremely important part of the Hyper-V Cluster design in a production build out. In fact, if the networking configuration and design is not done properly, you can expect problems to ensue from the outset. Microsoft recommends to design your network configuration with the following goals in mind:

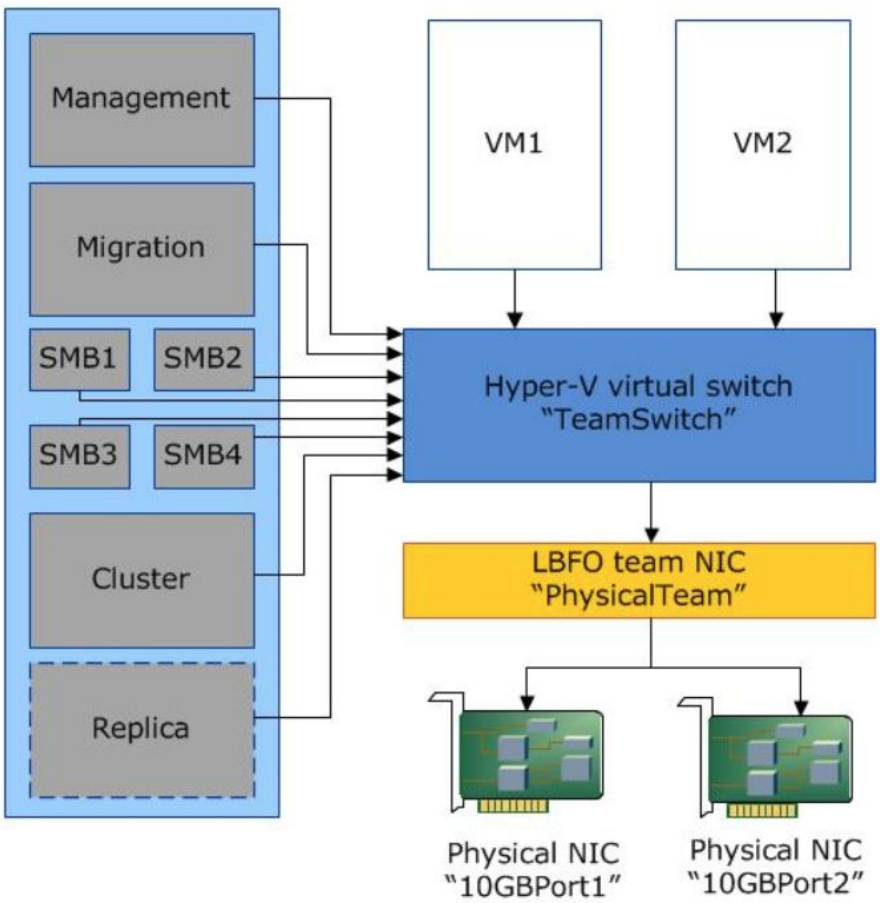
- To ensure network quality of service
- To provide network redundancy
- To isolate traffic to defined networks
- Where applicable, take advantage of Server Message Block (SMB) Multichannel

Proper design of network connections for redundancy generally involves teaming connections together. There are certainly major mistakes that can be made with the Network Teaming configuration that can lead to major problems when either hardware fails or a failover occurs. When cabling and designing network connections on Hyper-V hosts, you want to make sure that the cabling and network adapter connections are "X'ed" out, meaning that there is no single point of failure with the network path. The whole reason that you want to team network adapters is so that if you have a failure with one network card, the other "part" of the team (the other network card) will still be functioning.

Mistakes however can be made when setting up network teams in Hyper-V cluster configurations. A common mistake is to team ports off the same network controller. This issue does not present itself until a hardware failure of the network controller takes both ports from the same network controller offline.

Also, if using different makes/models of network controllers in a physical Hyper-V host, it is not best practice to create a team between those different models of network controllers. There can potentially be issues with the different controllers and how they handle the network traffic with the team. You always want to use the same type of network controller in a team.

Properly setting up your network adapters for redundancy and combining available controller ports together can bring many advantages such as being able to make use of "converged networking". Converged networking with Hyper-V is made possible by combining extremely fast NICs (generally higher than 10 Gbps) and "virtually" splitting traffic on your physical networks inside the hypervisor. So, the same network adapters are used for different kinds of traffic.



Hyper-V Converged Networking logical layout (image courtesy of Microsoft)

Storage Configuration

There is another "teaming" issue as it relates to Hyper-V storage. While teaming is good in other types of network traffic, you do not want to team network controllers with iSCSI traffic. Instead you want to utilize MPIO for load balancing iSCSI traffic. The problem with teaming technologies such as LACP (802.3ad) as relates to iSCSI traffic is that aggregating links via LACP, etc, does not improve the throughput of a single I/O flow. A single flow can only traverse one path. Link aggregation helps traffic flows from different sources. Each flow will then be sent down a different path based on a hash algorithm. MPIO on the other hand works between the hosts and iSCSI initiators and properly load balances the traffic of single flows to the different iSCSI initiators.

Aside from the performance benefits that MPIO brings, it also enables redundancy in that a path may go down between the Hyper-V host and the storage system, and the virtual machine stays online. The Multi**path** I/O that is what MPIO stands for allows for extremely performant and redundant storage paths to service Hyper-V workloads.

As an example, to enable multipath support in Hyper-V for iSCSI storage, run the following command on your Hyper-V host(s):

- Enable-MSDSMAutomaticClaim -BusType iSCSI

To enable round-robin on the paths:

- Set-MSDSMGlobalDefaultLoadBalancePolicy -Policy RR

Set the best-practice disk timeout to **60 seconds**:

- Set-MPIOSetting -NewDiskTimeout 60

Another best practice to keep in mind as relates to storage is always consult your specific vendor when it comes to the Windows storage setting values. This ensures performance is tweaked according to their specific requirements.

Hyper-V Networking Best Practices

There are certainly important considerations that need to be made to ensure Hyper-V Networking Best Practices. This includes the following:

- Physical NIC considerations
  - Firmware and drivers
  - Addressing
  - Enable Virtual Machine Queue or VMQ
  - Jumbo frames
  - Create redundant paths
- Windows and Virtual Network considerations
  - Created Dedicated Networks for traffic types
  - Use NIC teaming except on iSCSI network use MPIO
  - Disable TCP Chimney Offloading, and IPsec Offloading
  - Uncheck management traffic on dedicated Virtual Machine virtual switches

Physical NIC considerations

Starting at the physical NIC layer, this is an extremely important area of scrutiny when designing Hyper-V network architecture. Making sure to have the latest firmware and drivers for the physical NICs loaded, ensures you have the latest features and functionality as well as bug fixes in place. Generally speaking, it is always best practice with any hardware to have the latest firmware and drivers in place. An added benefit is it ensures you are in a supported condition if troubleshooting an issue and contacting a hardware vendor for support. The first question that typically is asked is "do you have the latest firmware and drivers installed?". So be sure you are running the latest firmware and drivers!

When it comes to IP addressing schemes, it goes without saying, never use DHCP for addressing the underlying network layer in a Hyper-V environment. Using automatic addressing schemes can lead to communication issues down the road. A good practice is to design out your IP addresses, subnets, VLANs, and any other network constructs **before** setting up your Hyper-V host or cluster. Putting forethought into the process helps to ensure there are no issues with overlapping IPs, subnets, etc when it comes down to implementing the design. Statically assign addresses to your host or hosts in a cluster.

Today's modern physical NICs inherently have features that dramatically improve performance, especially in virtualized environments. One such technology is **Virtual Machine Queue** or **VMQ** enabled NICs. VMQ enables many hardware virtualization benefits that allow much more efficient network connectivity for TCP/IP, iSCSI, and FCoE. If your physical NICs support VMQ, make sure to enable it.

Use **jumbo frames** with iSCSI, Live Migration, and Clustered Shared Volumes or CSV networks. Jumbo frames are defined as any Ethernet frame that is larger than 1500 bytes. Typically, in a virtualization environment, jumbo frames will be set to a frame size of 9000 bytes or a little larger. This may depend on the hardware you are using such as the network switch connecting devices. By making use of jumbo frames, traffic throughput can be significantly increased with lower CPU cycles. This allows for a much more efficient transmission of frames for the generally high traffic communication of iSCSI, Live Migration, and CSV networks.

Another key consideration when thinking about the physical network cabling of your Hyper-V host/cluster is to always have **redundant paths** so that there is no single point of failure. This is accomplished by using multiple NICs cabled to multiple physical switches which creates redundant paths. This ensures that if one link goes down, critical connected networks such as an ISCSI network still have a connected path.

Windows and Virtual Network Considerations

When creating the virtual network switches that allow critical network communication in a Hyper-V environment, it is best practice to create dedicated networks for each type of network communication. In a Hyper-V cluster, the following networks are generally created to carry each specific type of traffic:

- CSV or Heartbeat
- iSCSI
- Live Migration
- Management
- Virtual Machine Network

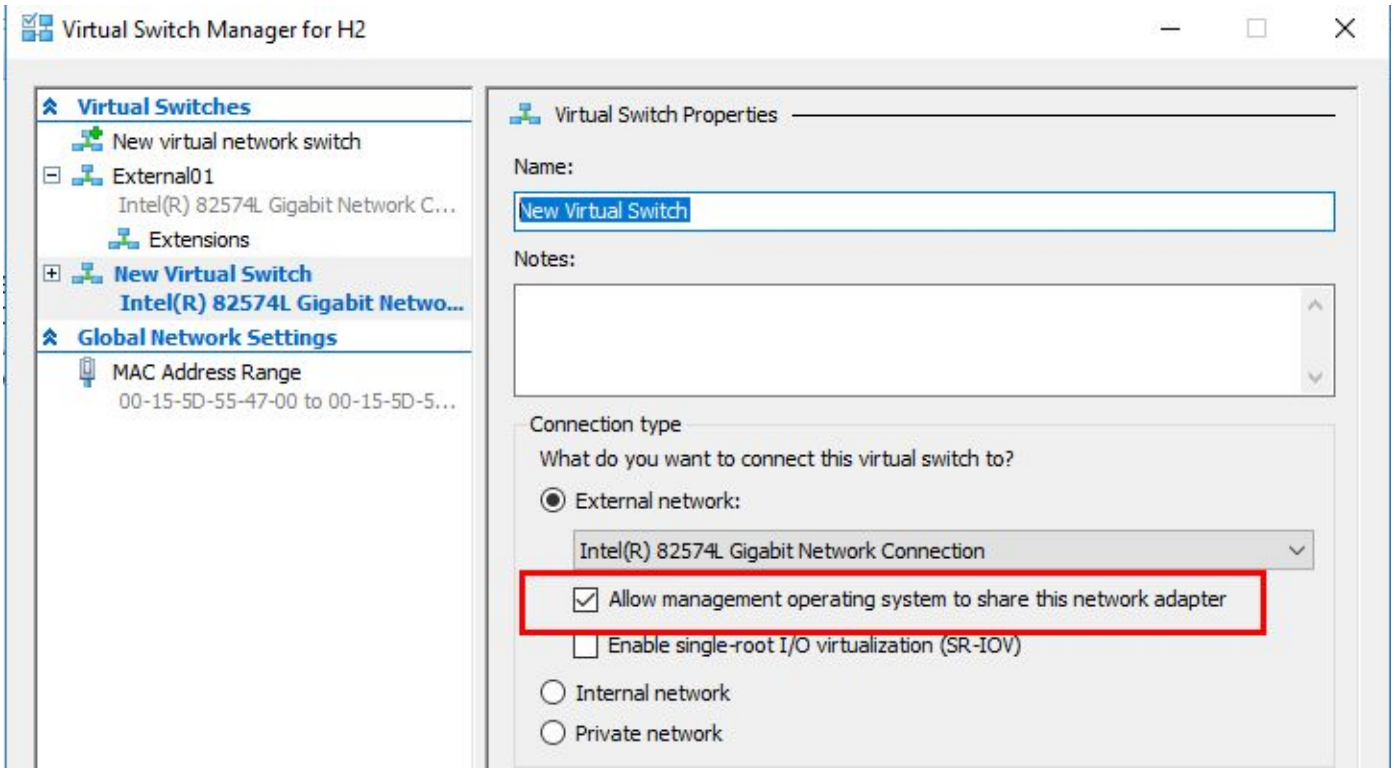
Creating dedicated networks for each type of network communication allows segregating the various types of traffic and is best practice from both a security and performance standpoint. There are various ways of doing this as well. Traffic can either be segregated by using multiple physical NICs or by aggregated multiple NICs and using VLANs to segregate the traffic.

As mentioned in the physical NIC considerations, having redundant paths enables high availability. By **teaming NICs** you are able to take advantage of both increased performance and high availability. A NIC team creates a single "virtual" NIC that Windows is able to utilize as if it were a single NIC. However, it contains multiple physical NICs in the underlying construct of the connection. If one NIC is disconnected, the "team" is still able to operate with the other connected NIC. However, with iSCSI connections, we don't want to use NIC teaming, but rather **Multipath I/O** or **MPIO**. NIC Teams provide increased performance for **unique** traffic flows and does not improve throughput of a single traffic flow as seen with iSCSI. With MPIO, iSCSI traffic is able to take advantage of all the underlying NIC connections for the flows between the hosts and the iSCSI target(s).

Do **not** use TCP Chimney Offloading or IPsec Offloading with Windows Server 2016. These technologies have been deprecated in Windows Server 2016 and can impact server and networking performance. To disable TCP Chimney Offload, from an elevated command prompt run the following commands:

- **Netsh int tcp show global** – This shows the current TCP settings
- **netsh int tcp set global chimney=disabled** – Disables TCP Chimney Offload, if enabled

Hyper-V allows the ability to enable **management traffic** on new virtual switches created. It is best practice to have the management traffic isolated to a dedicated virtual switch and **uncheck** "allow management operating system to share this network adapter" on any dedicated virtual machine virtual switch.



Allow management operating system to share this network adapter setting

What is a Hyper-V Virtual Switch?

The Hyper-V virtual switch is itself a software-based layer 2 Ethernet network switch that is available by default in Hyper-V Manager when you install the Hyper-V role on a server. The Hyper-V virtual switch allows for many different types of management as well as automation via programmatically managed and extensible capabilities. This allows connecting to both virtual networks and the physical network.

In addition to traditional networking in the true sense, Hyper-V virtual switches also allow for and provide policy enforcement for security, isolating resources, and ensuring SLAs. These additional features are powerful tools that allow today's often multi-tenant environments to have the ability to isolate workloads as well as provide traffic shaping. This also assists in protecting against malicious virtual machines.

The Hyper-V virtual switch is highly extensible. Using the Network Device Interface Specification or NDIS filters as well as Windows Filtering Platform or WFP, Hyper-V virtual switches can be extended by plugins written specifically to interact with the Hyper-V virtual switch. These are called Virtual Switch Extensions and can provide enhanced networking and security capabilities.

Hyper-V Virtual Switch Capabilities and Functionality

We have already touched on some of the features and functionality that allows Hyper-V administrators a great deal of control and flexibility in various environments. However, let's look closer at some of the capabilities that are afforded by the Hyper-V virtual switch.

- ARP/ND Poisoning (spoofing) protection – A common method of attack that can be used by a threat actor on the network is MAC spoofing. This allows an attacker to appear to be coming from a source illegitimately. Hyper-V virtual switches prevent this type of behavior by providing MAC address spoofing protection.
- DHCP Guard protection – With DHCP guard, Hyper-V is able to protect against a rogue VM being using for a DHCP server which helps to prevent man-in-the-middle attacks.
- Port ACLs – Port ACLS allow administrators to filter traffic based on MAC or IP addresses or ranges which allows effectively setting up network isolation and microsegmentation
- VLAN trunks to VM – Allows Hyper-V administrators to direct specific VLAN traffic to a specific VM
- Traffic monitoring – Administrators can view traffic that is traversing a Hyper-V virtual switch
- Private VLANs – Private VLANs can effectively microsegment traffic as it is basically a VLAN within a VLAN. VMs can be allowed or prevented from communicating with other VMs within the private VLAN construct

There are three different connectivity configurations for the Hyper-V Virtual Switch that can be configured in Hyper-V. They are:

- Private Virtual Switch
- Internal Virtual Switch
- External Virtual Switch

Private Virtual Switch

With the Private Virtual Switch, the virtual switch only allows communications between the connected virtual machines that are connected to the private virtual switch.

Internal Virtual Switch

With the Internal Virtual Switch, it only allows communication between virtual adapters connected to connected VMs and the management operating system.

External Virtual Switch

External Virtual Switches allows communication between virtual adapters connected to virtual machines and the management operating system. It utilizes the connected physical adapters to the physical switch for communicating externally.

With the external virtual switch, virtual machines can be connected to the outside world without any additional routing mechanism in place. However, with both private and internal switches, there must be some type of routing functionality that allows getting traffic from the internal/private virtual switches to the outside. The primary use case of the internal and private switches is to isolate and secure traffic. When connected to these types of virtual switches, traffic is isolated to only those virtual machines connected to the virtual switch.

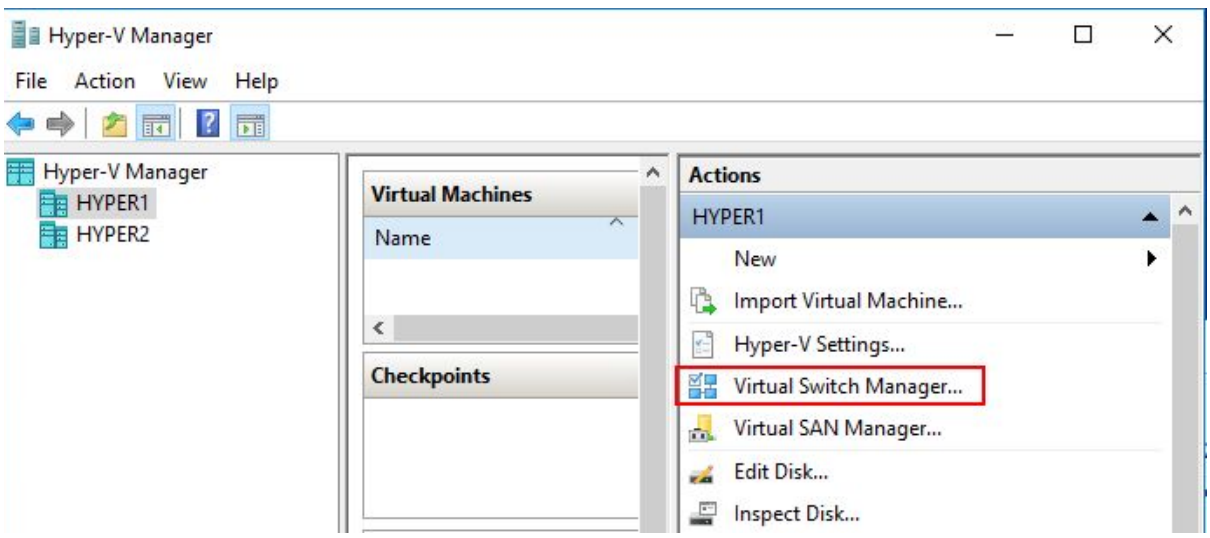
Hyper-V Logical Switches

When utilizing System Center in a Hyper-V environment, the Virtual Machine Manager or VMM fabric enables the use of a different kind of Hyper-V virtual switch – logical switches. A logical switch brings together the virtual switch extensions, port profiles, and port classifications so that network adapters can be consistently configured across multiple hosts. This way, multiple hosts can have the same logical switch and uplink ports associated.

This is similar in feel and function for VMware administrators who have experience with the distributed virtual switch. The configuration for the distributed virtual switch is stored at the vCenter Server level. The configuration is then deployed from vCenter to each host rather than from the host side.

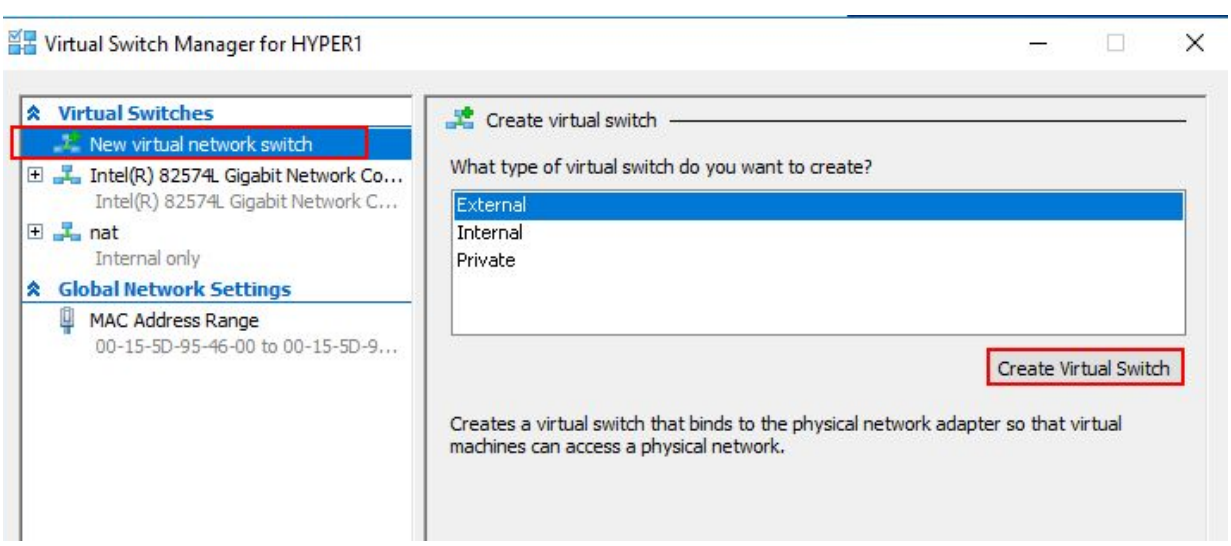
Creating Hyper-V Virtual Switches

Hyper-V standard virtual switches can be created using either the Hyper-V Manager GUI or by using PowerShell. We will take a look at each of these methods of configuration and deployment to see how the standard Hyper-V virtual switch can be deploying using either method.



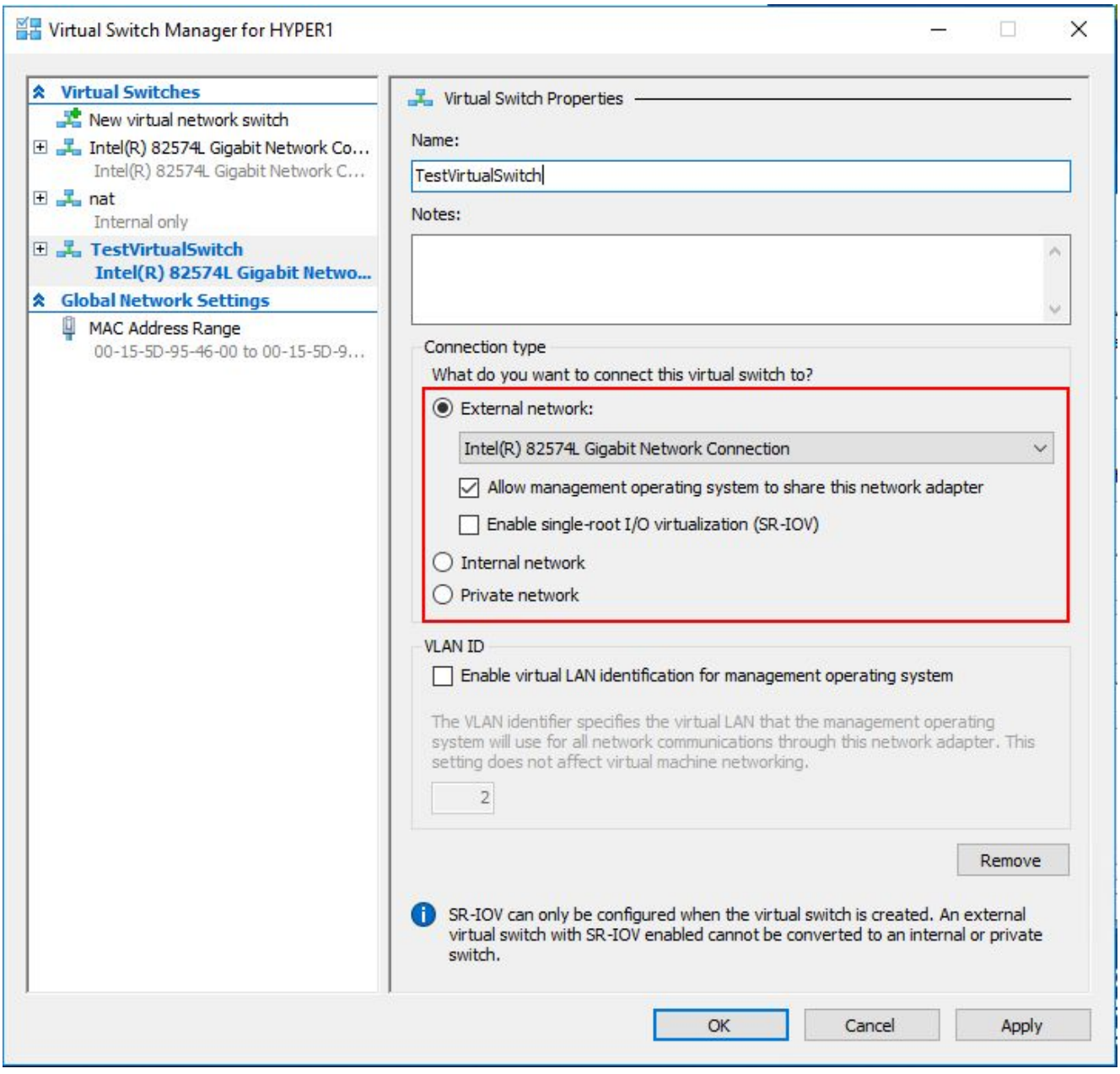
Allow management operating system to share this network adapter setting

Creating a new virtual network switch in the Hyper-V Manager Virtual Switch Manager for Hyper-V.



Creating a new Hyper-V virtual switch

Looking at the Hyper-V virtual switch properties, you can set the Connection type as well as the VLAN ID for the new Hyper-V virtual switch.



Configuring the Hyper-V Virtual Switch properties for a new virtual switch

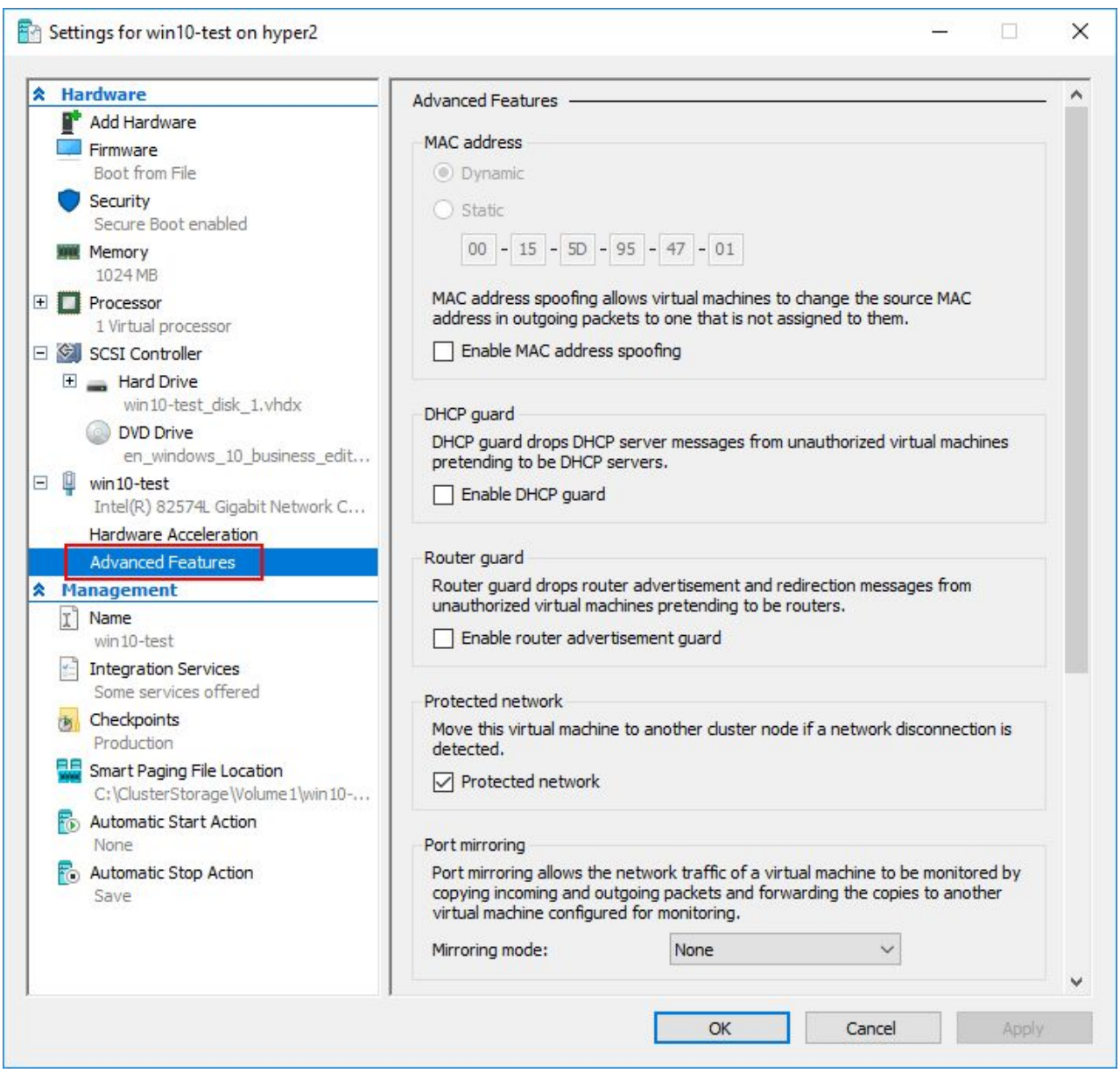
Creating Hyper-V Virtual Switches with PowerShell

Using PowerShell for virtual switch creation is a great way to achieve automation in a Hyper-V environment. PowerShell makes it easy to create new Hyper-V virtual switches in just a few simple one-liner cmdlets.

- **Get-NetAdapter** – make note of the names and network adapters
- **External Switch** - New-VMSwitch -name <switch name> -NetAdapterName <network adapter name> -AllowManagementOS $true
- **Internal Switch** - New-VMSwitch -name <switch name> -SwitchType Internal
- **Private Switch** - New-VMSwitch -name <switch name> -SwitchType Private

While not directly related to the Hyper-V virtual switch configuration, the virtual machine level Advanced Features include several very powerful network features made possible by the Hyper-V virtual switch including:

- DHCP guard – Protects against rogue DHCP servers
- Router guard – Protects against rogue routers
- Protected network – A high availability mechanism that ensures a virtual machine is not disconnected from the network due to a failure on a Hyper-V host
- Port Mirroring – Allows monitoring traffic.



Advanced Virtual Machine Network Configuration settings

Hyper-V advanced virtual machine network configuration

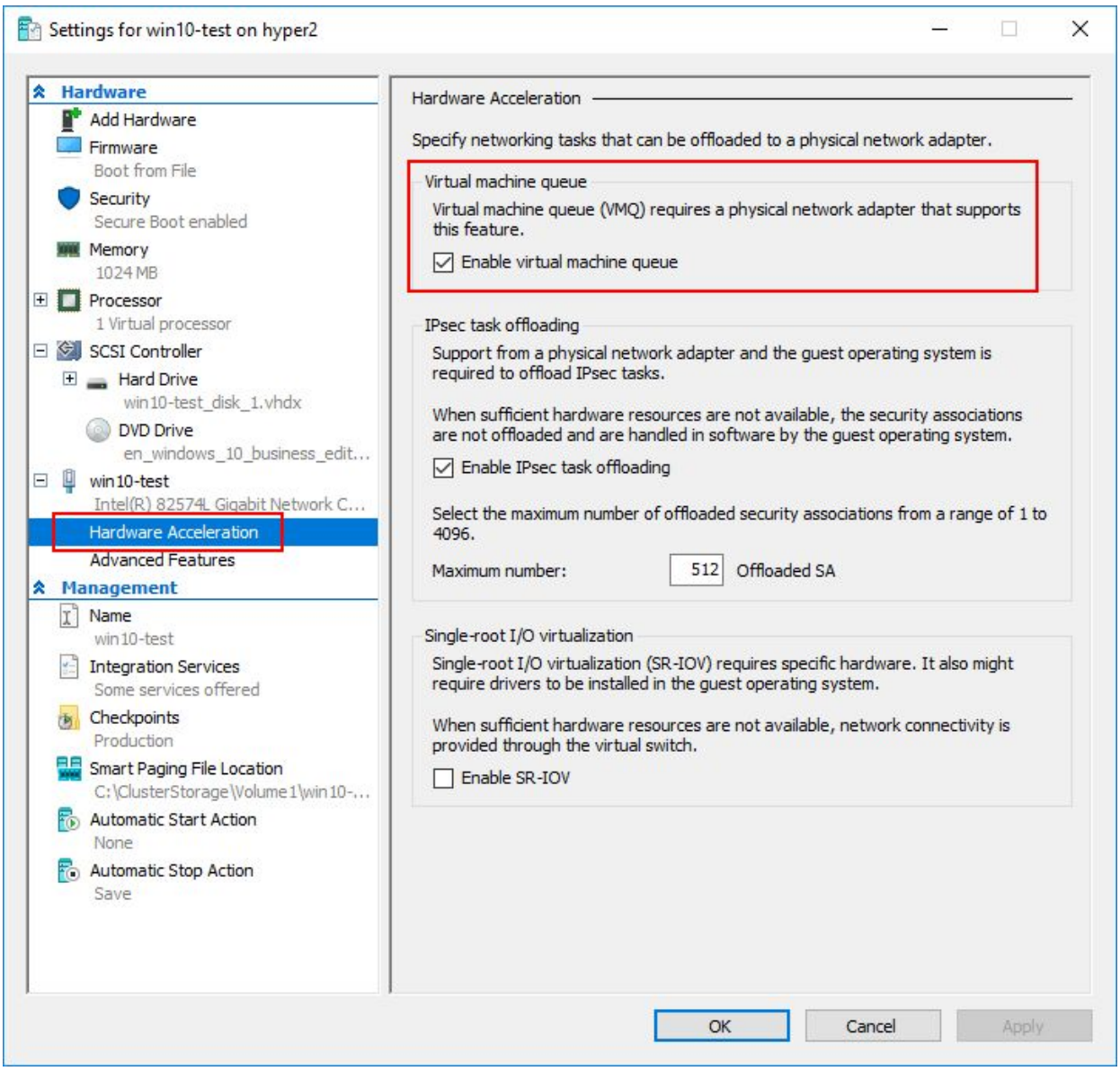
While creating a Hyper-V virtual switch or virtual switches and connecting virtual machines to them is certainly an important and necessary task, it is by no means the only network configuration that can be taken advantage of in a Hyper-V environment. There are many advanced Hyper-V virtual machine networking settings that can be taken advantage of by Hyper-V administrators that serve to strengthen and broaden the control over the Hyper-V network for the administrator. The advanced network configuration settings found in the settings of the Hyper-V virtual machine and Hyper-V in general include:

- Virtual machine queue
- IPsec Task offload
- SR-IOV
- DHCP Guard, Router Guard, Protected Network, and Port Mirroring

Let's take a look at these different Hyper-V advanced network settings configuration and how they can be used and implemented in an organization's Hyper-V infrastructure.

Virtual Machine Queue (VMQ)

- What is Virtual Machine Queue or VMQ and how is it utilized? Virtual Machine Queue or VMQ is a process that allows Hyper-V to improve network performance with virtual machines by expediting the transfer of network traffic from the physical adapter to the virtual machine. VMQ serves to decrease CPU utilization when network traffic utilization is elevated. When it is disabled, the CPU in the Hyper-V host has to utilize its own CPU power to process the multiple I/O streams to various virtual machines.
- ***Note*** There have been known issues with certain network cards, such as Broadcom branded cards, where Virtual Machine Queue being enabled actually has the opposite effect. This seems to have been an issue with earlier versions of Hyper-V and have since been overcome with later Hyper-V releases and firmware updates from network card manufacturers.
- Disabling or Enabling VMQ at the virtual switch level, can be accomplished with the Set-VMNetworkAdapter PowerShell cmdlet:
- Set-VMNetworkAdapter –ManagementOS -Name <VirtualNetworkAdapterName> -VmqWeight 0



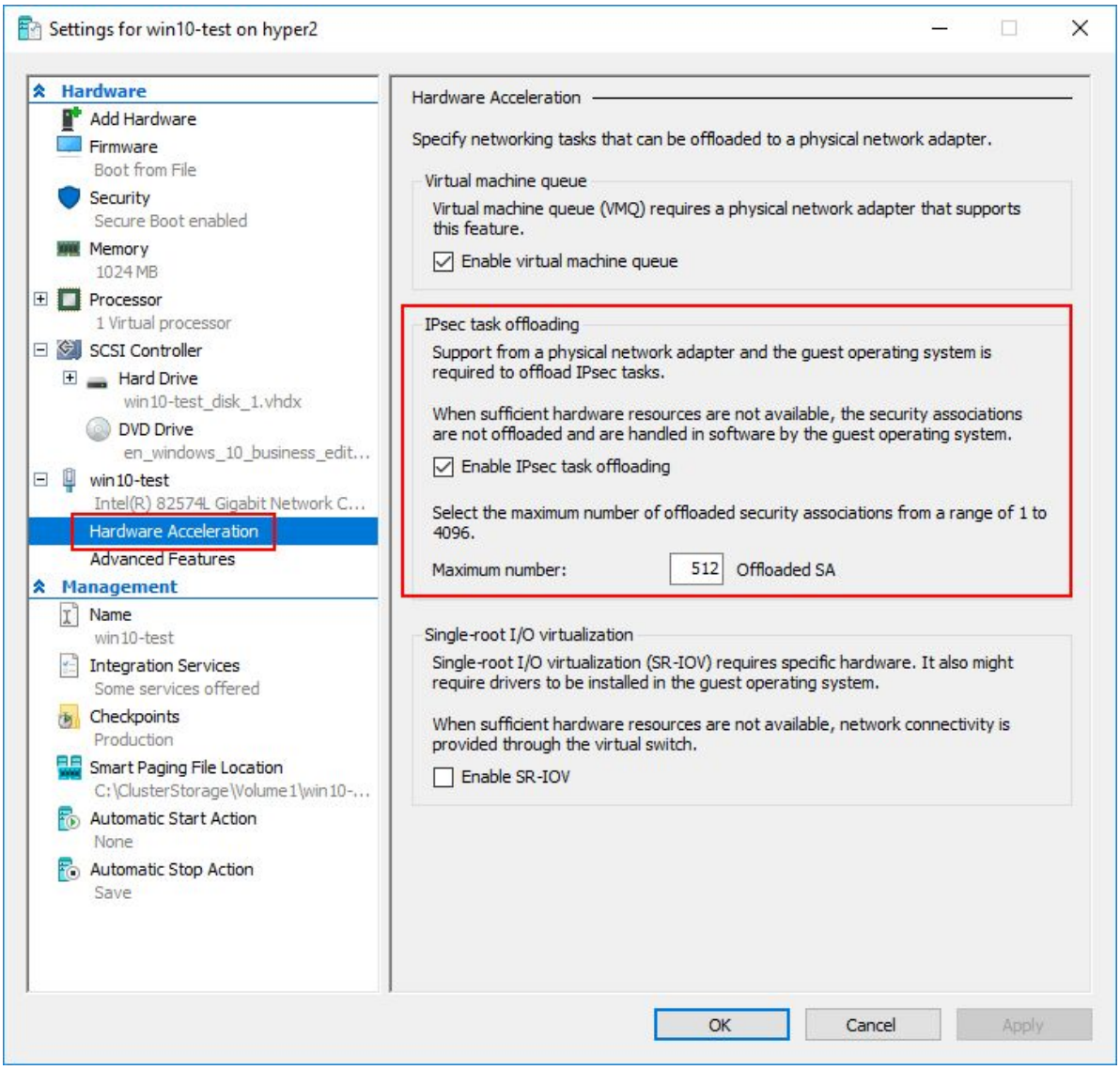
Advanced Virtual Machine Network Configuration settings

IPsec Task Offloading

Another mechanisms to offload network processing to hardware is **IPsec task offloading**. When large IPsec packets are used on the network, the IPsec task offloading feature can lower CPU utilizing on the Hyper-V host. IPsec is very processor intensive due to authenticating and encrypting the contents of packets. This feature in Hyper-V allows offloading this process in virtual machines and not simply the Hyper-V host. This is beneficial from many different perspectives.

You can set the number of maximum number of security associations that can be offloaded to the physical adapter in PowerShell:

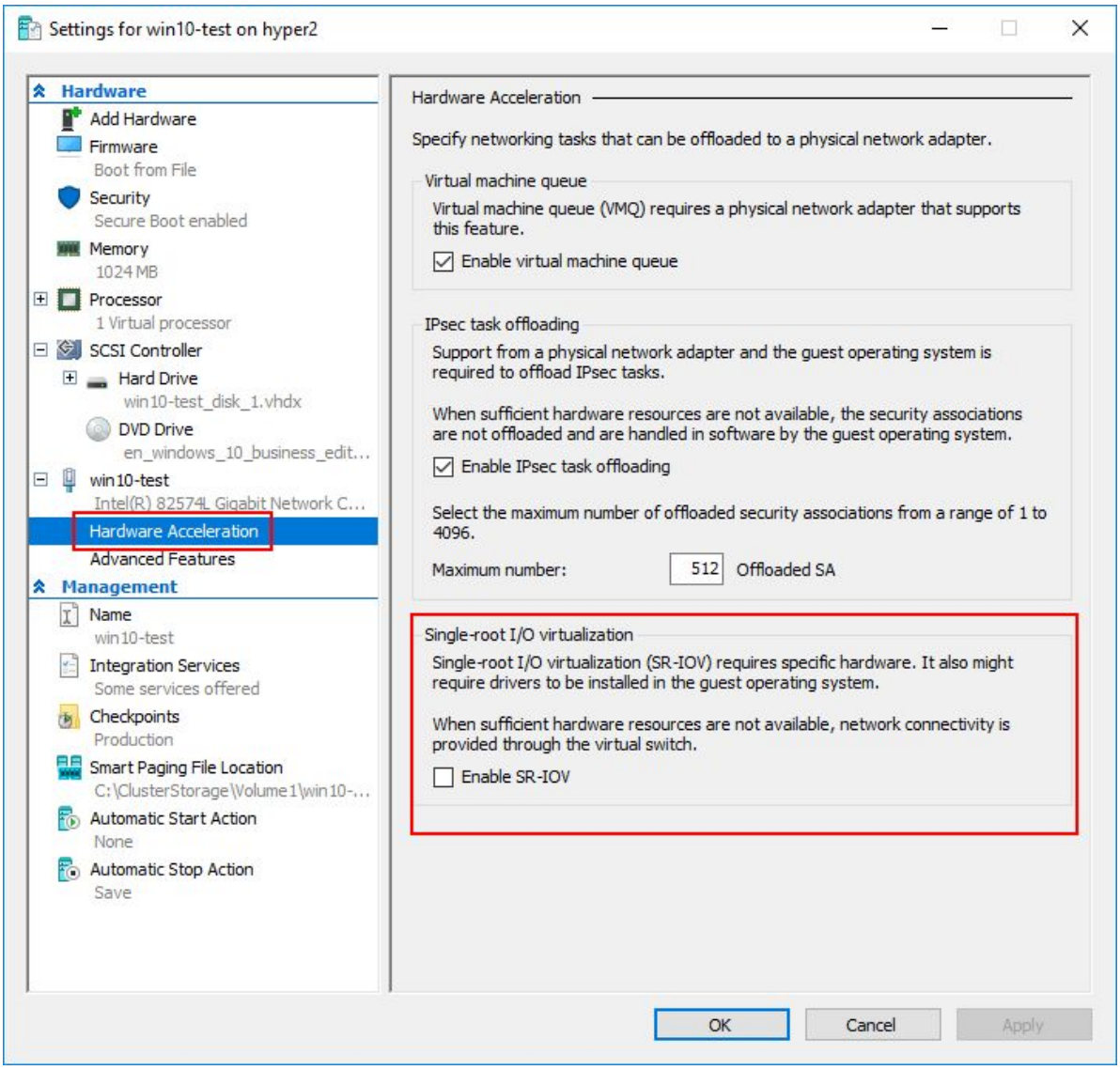
- Set-VMNetworkAdapter -IPsecOffloadMaximumSecurityAssociation<UInt32>



Configuring IPsec task offloading in virtual machine properties

SR-IOV

Windows Server 2016 Hyper-V introduces **Single-root I/O virtualization** or **SR-IOV**. What is SR-IOV? Again this is network performance feature that allows network traffic the ability to completely bypass the software switch layer of Hyper-V and allows SR-IOV devices to be assigned directly to a virtual machine. This is accomplished by some slick remapping of resources to the virtual machine such as interrupts and DMA. This feature is extremely well-suited for virtual machines that heavily utilize the network. Hyper-V is able to pass network traffic directly from the virtual machine to the physical network card and in doing such, doesn't manage the network traffic from the virtual machine to the physical network. This feature is compatible with many of the core Hyper-V features and virtual machine capabilities such as snapshotting, live migration, etc. Note that SR-IOV is not compatible with NIC teaming or extensible switch features.



Configuring Single-root I/O virtualization or SR-IOV on a virtual machine

DHCP Guard, Router Guard, Protected Network and Port Mirroring

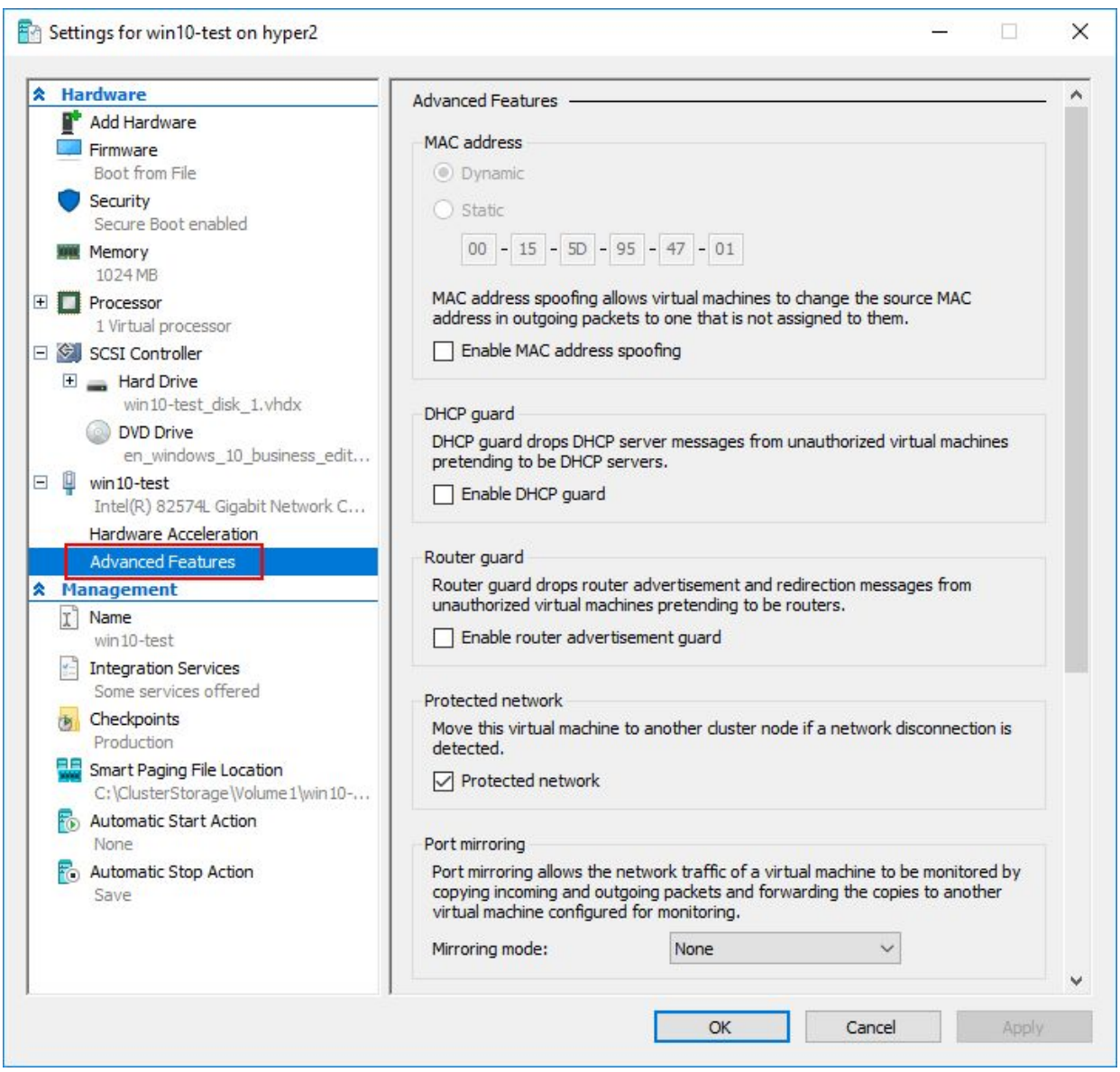
Under the **Advanced Features** of a Hyper-V virtual machine, there are a number of extremely powerful and interesting settings to take advantage of including **DHCP Guard**, **Router Guard**, **Protected Network**, and **Port Mirroring**. From a security and high availability standpoint, these settings provide some really great features for the Hyper-V administrator to control potential network issues as well as monitor network traffic.

The **DHCP guard** feature is a great way to ensure that a virtual machine is not enabled as a DHCP server accidentally or intentionally without authorization. When the DHCP guard feature is turned on, the Hyper-V host drops DHCP server messages from unauthorized virtual machines that are attempting to act as a DHCP server on the network.

With the **Router guard** feature, the Hyper-V host prevents virtual machines from advertising themselves on the network as a router and possibly causing routing loops or wreaking other havoc on the network.

Protected Network is a great feature for high availability. When set, this feature proactively moves the virtual machine to another cluster node if a network disconnection condition is detected on the virtual machine. This is enabled by default.

Port mirroring is a great way to either troubleshoot a network issue or perhaps perform security reconnaissance on the network. It typically mirrors traffic from one "port" to another "port" allowing a TAP device to be installed on the mirrored port to record all network traffic. With the **Port mirroring** virtual machine setting, Hyper-V administrators can mirror the network traffic of a virtual machine to another virtual machine that has monitoring utilities installed.



Advanced Hyper-V network settings that allow powerful functionality to control virtual machine network traffic

Overview of iSCSI Architecture

Most who are familiar with SAN storage over the past decade are well accustomed to iSCSI enabled SANs. What is iSCSI? The term iSCSI stands for Internet Small Computer Systems Interface and allows for IP based storage that enables block-level access to storage devices. The iSCSI commands are encapsulated in the TCP/IP packets. One of the huge advantages of using iSCSI for storage is it allows using the traditional Ethernet constructs that already exist in most enterprise datacenters. This means that iSCSI can be transmitted over existing switches and cabling, even alongside other types of network traffic. Aside from LAN connectivity, iSCSI commands can even be transmitted over WANs and even the Internet.

SANs that are enabled with iSCSI present storage **targets** to clients who are **initiators**. In a virtualization environment, the clients or initiators are the hypervisor hosts. The targets are the LUNs that are presented to the hypervisor hosts for storage. The iSCSI LUNs act as if they are local storage to the hypervisor host.

Hyper-V Design Considerations with iSCSI Storage

When thinking about properly designing any Hyper-V cluster, high availability and redundancy of resources should always be part of the design. This includes having multiples – Hyper-V hosts, SAN switches, cabling paths, SAN devices with multiple controllers, etc. With that being said, for iSCSI physical hardware:

- **Multiple Hyper-V Hosts** - Configure at least two Hyper-V hosts in a cluster with three or more being recommended for increased HA
- **Redundant network cards** - Have two network cards dedicated to iSCSI traffic
- **Redundant Ethernet switches** - Two Ethernet switches dedicated to iSCSI traffic. Cabling from redundant network cards should be "X-ed" out with no one single path of failure to storage from each Hyper-V host.
- **SAN with redundant controllers** - A SAN with multiple controllers (most enterprise ready SANs today are configured with at least (2) controllers). This protects against failures caused by a failed storage controller. When one fails, it "fails over" to the secondary controller.

With Windows Server 2016, the network convergence model allows aggregating various types of network traffic across the same physical hardware. The Hyper-V networks suited for the network convergence model include the management, cluster, Live Migration, and VM networks. However, iSCSI storage traffic needs to be separated from the other types of network traffic found in the network convergence model as per Microsoft best practice regarding storage traffic.

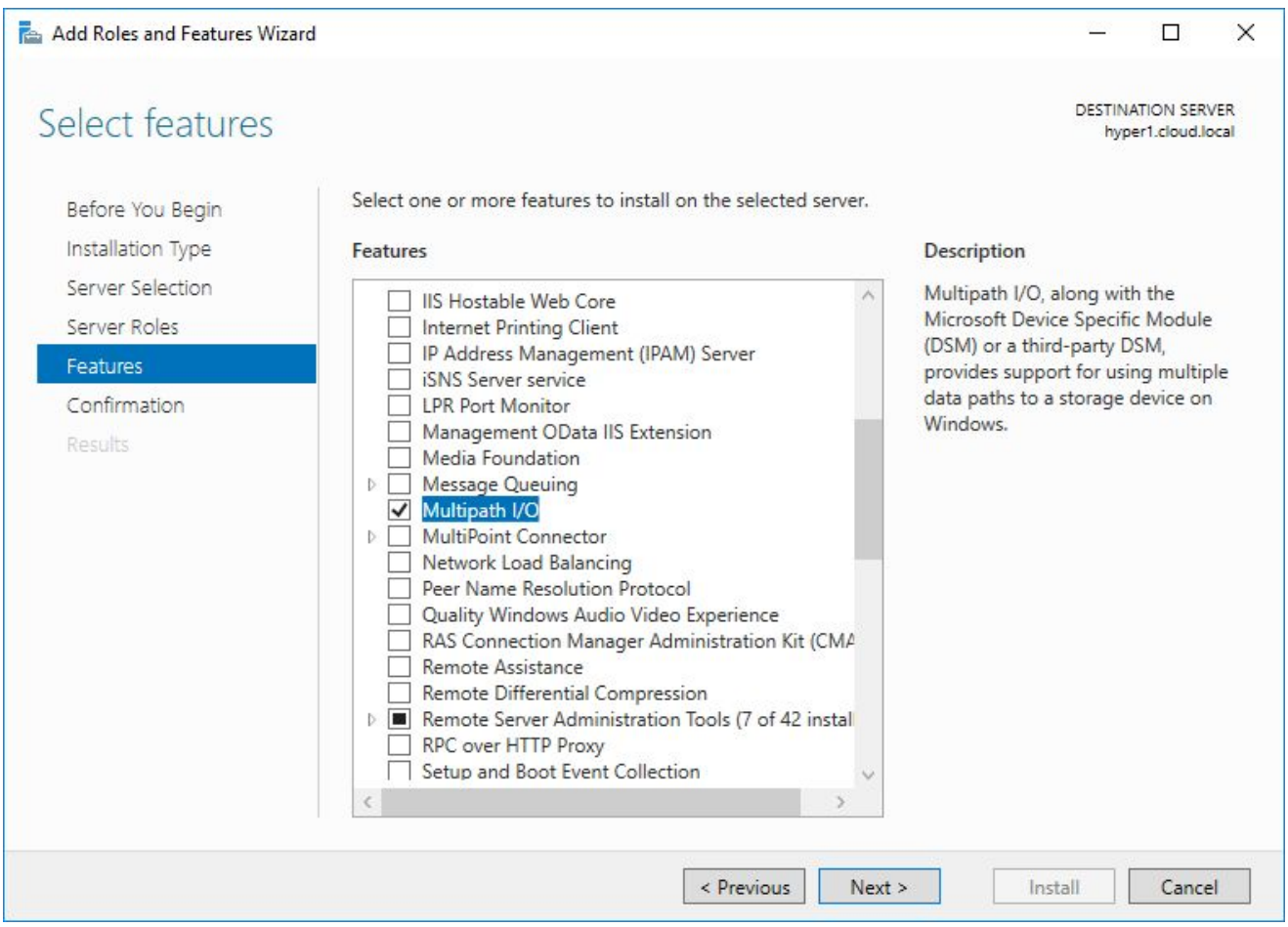
With network configuration, you will want to configure your two network adapters for iSCSI traffic with unique IPs that will communicate with the storage controller(s) on the SAN. There are a few other considerations when configuring the iSCSI network cards on the Hyper-V host including:

- **Use Jumbo frames where possible** – Jumbo frames allow a larger data size to be transmitted before the packet is fragmented. This generally increases performance for iSCSI traffic and lowers the CPU overhead for the Hyper-V hosts. However, it does require that all network hardware used in the storage area network is capable of utilizing jumbo frames.
- **Use MPIO** – MPIO or Multipath I/O is used with accessing storage rather than using port aggregation technology such as LACP on the switch or Switch Embedded Teaming for network convergence. Link aggregation technologies such as LACP only improve the throughput of **multiple** I/O flows coming from different sources. Since the flows for iSCSI will not appear to be unique, it will not improve the performance of iSCSI traffic. MPIO on the other hand works from the perspective of the initiator and target so can improve the performance of iSCSI.
- **Use dedicated iSCSI networks** – While part of the appeal of iSCSI is the fact that it can run alongside other types of network traffic, for the most performance, running iSCSI storage traffic on dedicated switch fabric is certainly best practice

On the storage side of things, if your SAN supports Offloaded Data Transfer or ODX, this can greatly increase storage performance as well. Microsoft's Offloaded Data Transfer is also called **copy offload** and enables direct data transfers within or between a storage device(s) without involving the host. Comparatively, without ODX, the data is read from the source and transferred across the network to the host. Then the host transfers the data back over the network to the destination. The ODX transfer, again, eliminates the host as the middle party and significantly improves the performance of copying data. This also lowers the host CPU utilization since the host no longer has to process this traffic. Network bandwidth is also saved since the network is no longer needed to copy the data back and forth.

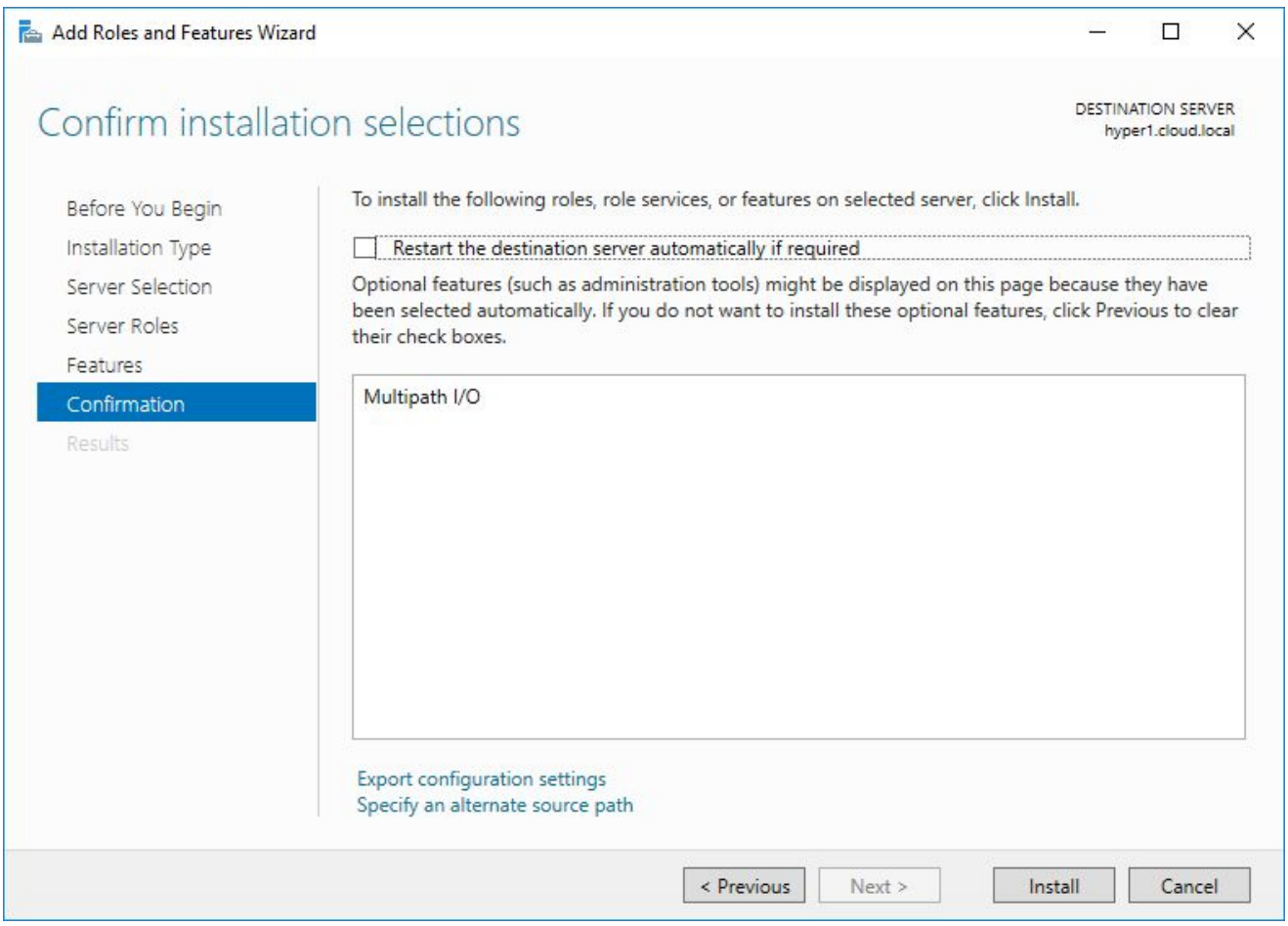
Hyper-V Windows Configuration for iSCSI

There are a few items to note in configuring iSCSI connections in Windows Server. You need to add the needed component to a Windows Server installation to properly handle MPIO connections. First, you add the MPIO feature to the Windows Server installation.



Adding Multipath I/O to Windows Server 2016

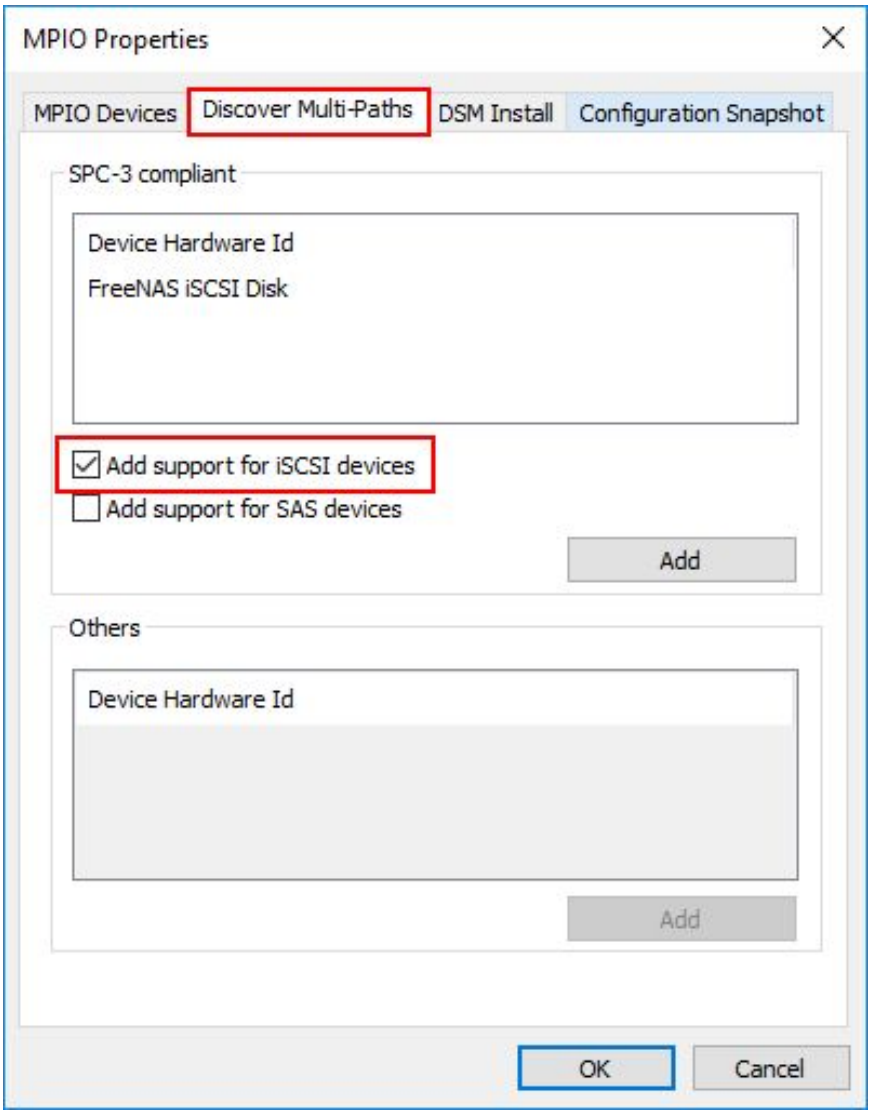
The installation of MPIO in Windows Server 2016 installs quickly but will require a reboot.



About to begin the installation of MPIO in Windows Server 2016

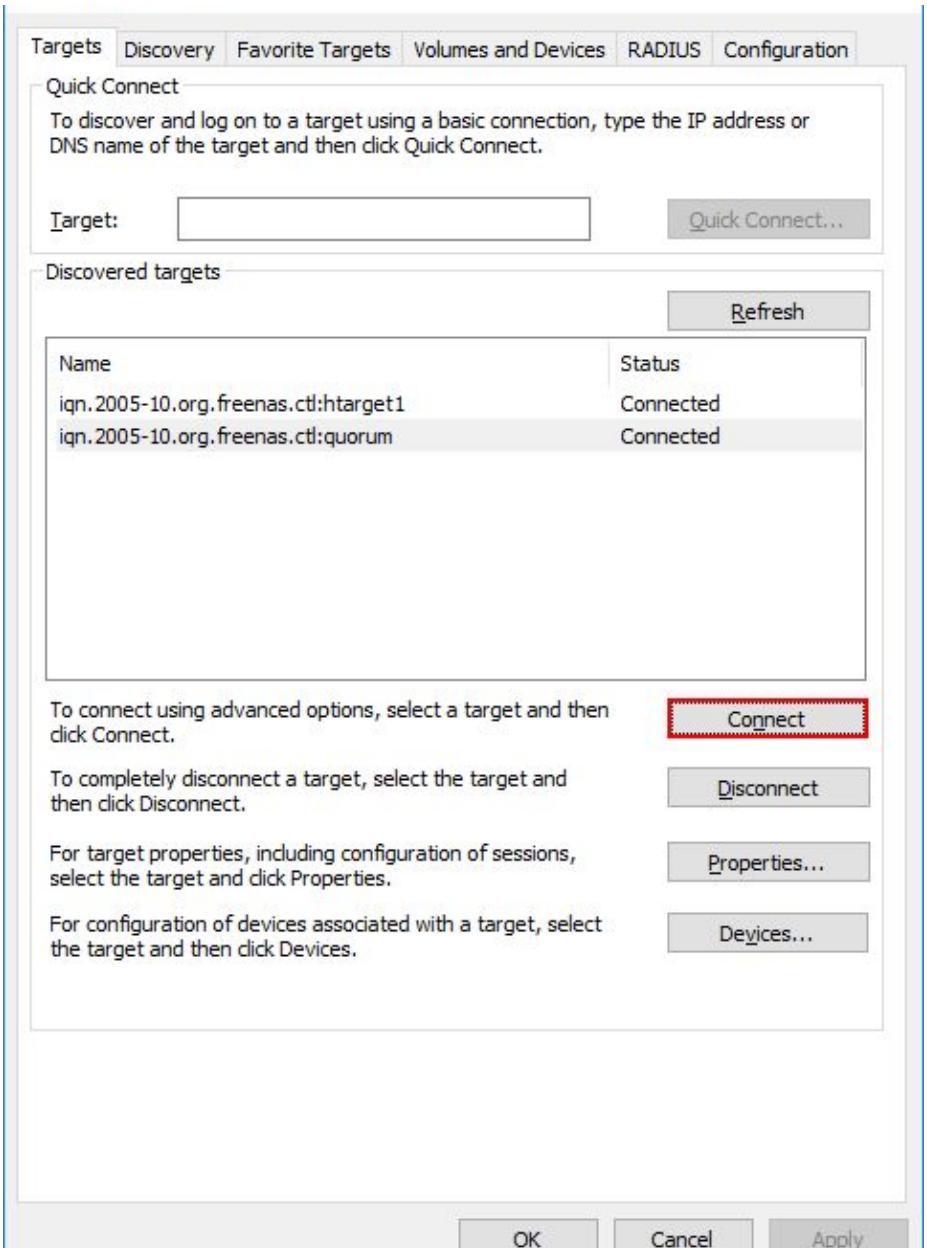
Once MPIO is installed and the server has been rebooted, you can now configure MPIO for iSCSI connections.

1. Launch the MPIO utility by typing **mpiocpl** at a run menu. This will launch the **MPIO Properties** configuration dialog.
2. Under the **Discover Multi-Paths** tab, check the **Add support for iSCSI devices** check box.
3. Click OK



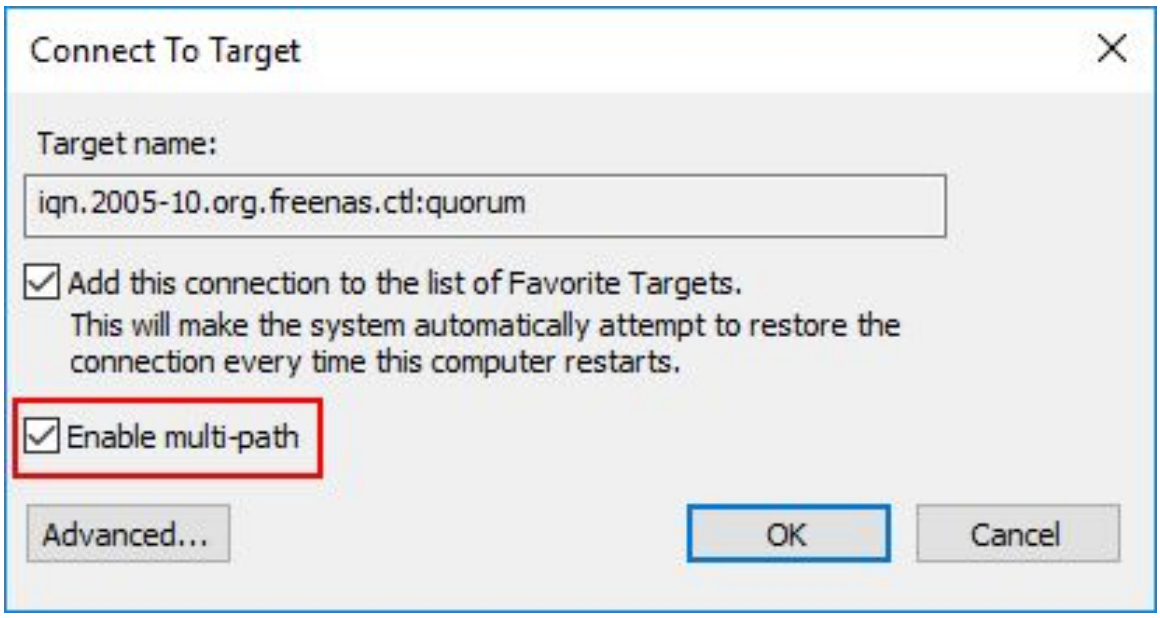
Configuring MPIO for iSCSI connections

Under the iSCSI configuration, launched by typing iscsicpl, you can Connect to an iSCSI target



Click the Connect button to setup multipath for an iSCSI target

The **Connect to Target** dialog box allows selecting the **Enable multi-path** checkbox to enable multipathing for iSCSI targets.



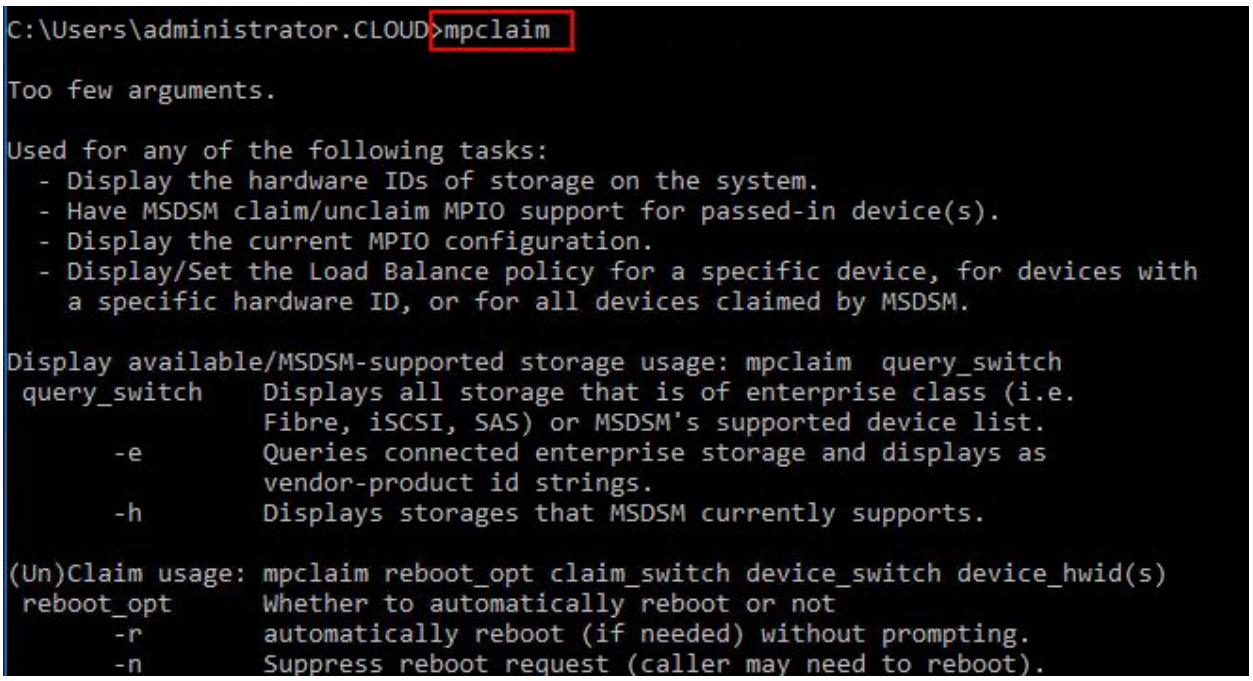
Click the Enable multi-path check box

You will need to do this for every volume the Hyper-V host is connected to. Additionally, in a configuration where you have two IP addresses bound to two different network cards in your Hyper-V server and two available IPs for the iSCSI targets on your SAN, you would create paths for each IP address connected to the respective IP address of the iSCSI targets. This will create an iSCSI network configuration that is not only fault tolerant but also able to use all available connections for maximum performance.

Verifying Multipathing

There is a powerful little command line utility that allows pulling a tremendous amount of information regarding multipath disk connections – **mpclaim**.

Launch mpclaim from the command line to see the various options available.



The mpclaim utility displays information regarding multipathing

To check your current policy for your iSCSI volumes:

- mpclaim -s -d

To verify paths to a specific device number:

- mpclaim -s -d <device number>

What is Windows Server 2016 Storage Spaces Direct

Windows Server Storage Spaces Direct was introduced with Windows Server 2016. Again, drawing a comparison here with VMware vSAN that many are familiar with, Storage Spaces Direct works on similar hardware architecture ideas. With Storage Spaces Direct, locally attached drives are used to create software-defined storage in a converged or hyper-converged manner. It includes creating storage tiers including caching and capacity tiers. Using erasure coding, Storage Spaces Direct is able to provide fault tolerance between nodes. Converged networking utilizing RDMA is also able to deliver very good network performance. What are the requirements for Storage Spaces Direct? There are several requirements for configuring Storage Spaces Direct across both the physical server hardware and operating system version.

Windows Server 2016 Storage Spaces Direct Requirements

There are quite a few hardware considerations to be made when considering Windows Server 2016 Storage Spaces Direct. The following are requirements in building out compatible hardware for Windows Server 2016 Storage Spaces Direct.

Windows Server 2016 Certified

Hardware needs to be certified to run with Microsoft Windows Server 2016. A great resource for testing whether or not specific hardware is tested and approved for use with Microsoft Windows Server 2016 is the Windows Server Catalog.

- https://www.windowsservercatalog.com/

Windows Server Catalog allows seeing if specific hardware his certified

Physical Servers

- Storage Spaces Direct needs a minimum of (2) servers and can contain a maximum of (16) servers
- It is best practice to use the same make/model of servers

CPU

- Intel/AMD procs – Nehalem/EPYC or newer

Memory

- You need enough memory for Windows Server 2016 itself
- Recommended (4) GB of memory for every 1 TB of cache drive capacity on each server

Boot

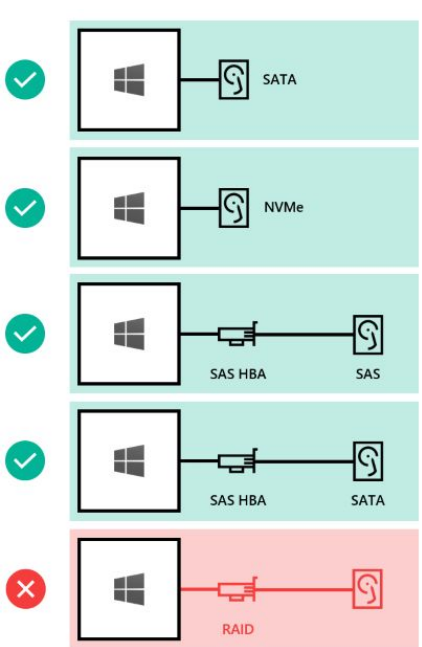
- Any Windows supported boot device
- RAID 1 mirror for boot drive is supported
- 200 GB minimum boot drive size

Networking

- (1) 10 Gbps network adapter per server
- Recommended at least 2 NICs for redundancy
- A (2) server configuration supports a “switchless” configuration with a direct cable connection

Drives

- SATA, SAS, and NVMe supported
- 512n, 512e, and 4K native drives all supported
- SSDs need to have power loss protection
- Direct attached SATA drives supported
- Direct attached NVMe drives
- It is **not supported** to have a RAID controller card or SAN storage. HBA cards must be configured in pass-through mode



Supported Drive configurations for Storage Spaces Direct (courtesy of Microsoft)

Windows Server 2016 Operating System Requirements

- Windows Server 2016 Datacenter License is required for Storage Spaces Direct

Windows Server 2016 Storage Spaces Direct Architecture

Typically, in Hyper-V environments you may want to utilize a hyper-converged solution. Storage Spaces Direct will automatically use the fastest storage (SSD or NVMe) for caching. The caching mechanism is dynamic meaning it can change the drives that serves the caching mechanism due to changes in storage traffic or even an SSD failure. Microsoft recommends at least (2) caching devices per node utilizing either SSD or NVMe drives. Microsoft also recommends making use of the new Resilient File System with the fast-cloning technology and resilient nature.

As far as fault tolerance, if you have three or more nodes in your cluster, Storage Spaces Direct is resilient to up to two drive losses or losing two hosts. With two-node clusters, the hit on disk space is quite high since the cluster utilizes a two-way mirroring mechanism for fault tolerance. This means you essentially lose 50% of your disk capacity.

When utilizing 4 node clusters and larger, you can take advantage of erasure coding similar to RAID 5 which is much more efficient from a capacity standpoint (60-80%). However, this erasure coding is heavier on writes. Microsoft has worked on this problem with Multi Resilient Volumes (MRVs) using ReFS. This creates a three-way mirror with erasure coding that acts as a sort of a write cache that works extremely well with Hyper-V virtual machines. Data is safely stored on three different drives on different servers.

Windows Server 2016 SAN vs Storage Spaces Direct

With the new age of software defined storage and hyper-converged infrastructure capabilities supported in today's hypervisors, the choice for storage today is not so clear cut as it has been in years past. Hyper-V administrators today with the new Storage Spaces Direct capabilities built inside of Hyper-V have the same question – stay with the familiar traditional SAN storage or utilize Storage Spaces Direct? Both architectures will provide great platforms for Hyper-V virtual machines. Even though S2D is the new comer on the scene, it is already a very stable and reliable solution for running production workloads in a Hyper-V virtual environment. However, there may be certain features or functionality, as well as use cases that may dictate one solution over the other. Let's consider some potential characteristics of each solution that may help choose one over the other.

SAN based storage is tried and proven and most vendors today have Hyper-V compatible solutions. Storage Spaces technology has been in existence since Windows Server 2012 and not before. Don't think that just because you are using Windows Server 2016 Hyper-V that you must use Storage Spaces Direct. Often times, procedures and processes are in place that organizations have used successfully with SAN storage in their virtualized environments that work very well and efficiently. If you are comfortable with these process, procedures, and vendor solutions in your environment, this can certainly be a major reason to stick with SAN storage for Windows Server 2016 Hyper-V.

Storage Spaces Direct is also a great solution and often is much cheaper than high end SAN storages from various vendors. Additionally, the storage provisioning and management becomes part of the Windows Server 2016 operating system and not a separate entity that must be managed with disparate tooling and vendor utilizes. Storage Spaces Direct can be fully managed and configured from within PowerShell which is certainly a trusted and powerful solution that is baked into today's Windows Server OS.

One of the reasons you might pick SAN storage over Storage Spaces Direct today is if you have the need for deduplication. Currently, Storage Spaces Direct does not support deduplication or other more advanced storage features and capabilities that you may get with third-party vendor SAN storage. There is no doubt that in future versions of Storage Spaces Direct, deduplication features will be included into the solution as well as other expanded storage functionality.

Performance for many is a driving factor when it comes to choosing a storage solution. Can Storage Spaces Direct perform adequately for running production workloads? A published article from Microsoft shows Storage Spaces Direct enabled servers providing 60GB/sec in aggregate throughput using (4) Dell PowerEdge R730XD servers. The solution can certainly perform and perform well!

Why Use Hyper-V VHDX File format?

To begin with, let's take a step back and look at basic features that VHDX provides and why Hyper-V administrators would choose to use the newer VHDX file format as opposed to the VHD virtual disk. The VHDX virtual disk file format was introduced with Windows Server 2012 and provides a much more powerful virtual disk format that helps to solve some of the scalability and performance constraints that exist with the VHD file format. What are the new configuration maximums for the VHDX file format?

- Supports 64 TB virtual hard disk size
- Improved logging mechanisms in VHDX
- Automatic disk alignment
- Dynamic resizing
- Virtual disk sharing

New disk sizes

The 64 TB virtual hard disk size certain opens up some pretty interesting use cases. However, for most, there will be no disk that will not fall within the boundaries of this new disk size and most will not even come close to this new configuration maximum. This also will negate the need to perform pass-through storage provisioning if this was necessary for size reasons.

Improved Logging

With the improved logging features that are contained within the VHDX virtual disk metadata, the VHDX virtual disk is further protected from corruption that could happen due to unexpected power failure or power loss. This also opens up the possibility to store custom metadata about a file. Users may want to capture notes about the specific VHDX file such as the operating system contained or patches that have been applied.

Automatic Disk Alignment

Aligning the virtual hard disk format to the disk sector size provides performance improvements. VHDX files automatically align to the physical disk structure. VHDX files also leverage larger block sizes for both the dynamic and differencing disk formats. This greatly improves the performance of dynamic sized VHDX files, making the difference in performance negligible between fixed and dynamic. The dynamic sizing option is the option that is preferred when creating VHDX files.

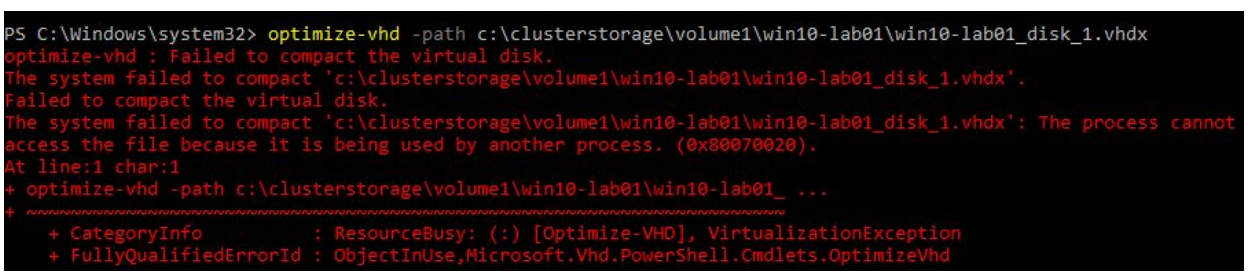
Shared VHDX

There is a new option starting in Windows Server 2012 to share virtual VHDX hard disks between virtual machines. Why would you do this? Guest clustering is an interesting option to run a clustered Windows Server configuration on top of a physical Hyper-V cluster to allow application high availability on top of virtual machine high availability. If a virtual machine fails, you still suffer the downtime it takes to restart the virtual machine on another Hyper-V host. When running a cluster inside a Hyper-V cluster, when one virtual machine fails, the second VM in the cluster assumes the role of servicing the application. A shared VHDX allows utilizing a VHDX virtual disk as a shared cluster disk between guest cluster nodes.

Optimizing VHDX Virtual Disk Files

Optimizing VHDX Virtual Disk Files allows optimizing the space in a dynamic sized virtual hard disk file. This is accomplished with the **optimize-vhd** cmdlet. The **Compact** operation is used to optimize the files. This option reclaims unused blocks and rearranges the blocks to be more efficiently packed which reduces the overall size of the VHDX virtual hard disk file.

The **optimize-vhd** operation can only be performed with the virtual hard disk detached or attached in read-only mode if the virtual machine is running. If the disk is not attached properly for the operation specified or in use, you will see the following:



Error received when trying to optimize a VHDX file that is in use

PowerShell options available with the **optimize-vhd** cmdlet:

- Optimize-vhd -Path <your VHD path> -Mode **Full** – This option runs the compact operation in Full mode which scans for zero blocks and reclaims unused blocks. This is only allowed if the virtual hard disk is mounted in read only mode.
- Optimize-vhd -Path <your VHD path> -Mode **Pretrimmed** – Performs the same as Quick mode but does not require the hard disk to be mounted in read only mode.
- Optimize-vhd -Path <your VHD path> -Mode **Quick** – The virtual hard disk is mounted in read-only and reclaims unused blocks but does not scan for zero blocks.
- Optimize-vhd -Path <your VHD path> -Mode **Retrim** – Sends retrims without scanning for zero blocks or reclaiming unused blocks.
- Optimize-vhd -Path <your VHD path> -Mode **Prezeroed** – performs as Quick mode but does not require the virtual disk to be read only. The unused space detection will be less effective than the read only scan. This is useful if a tol has been run to zero all the free space on the virtual disk as this mode then can reclaim the space for subsequent block allocations.

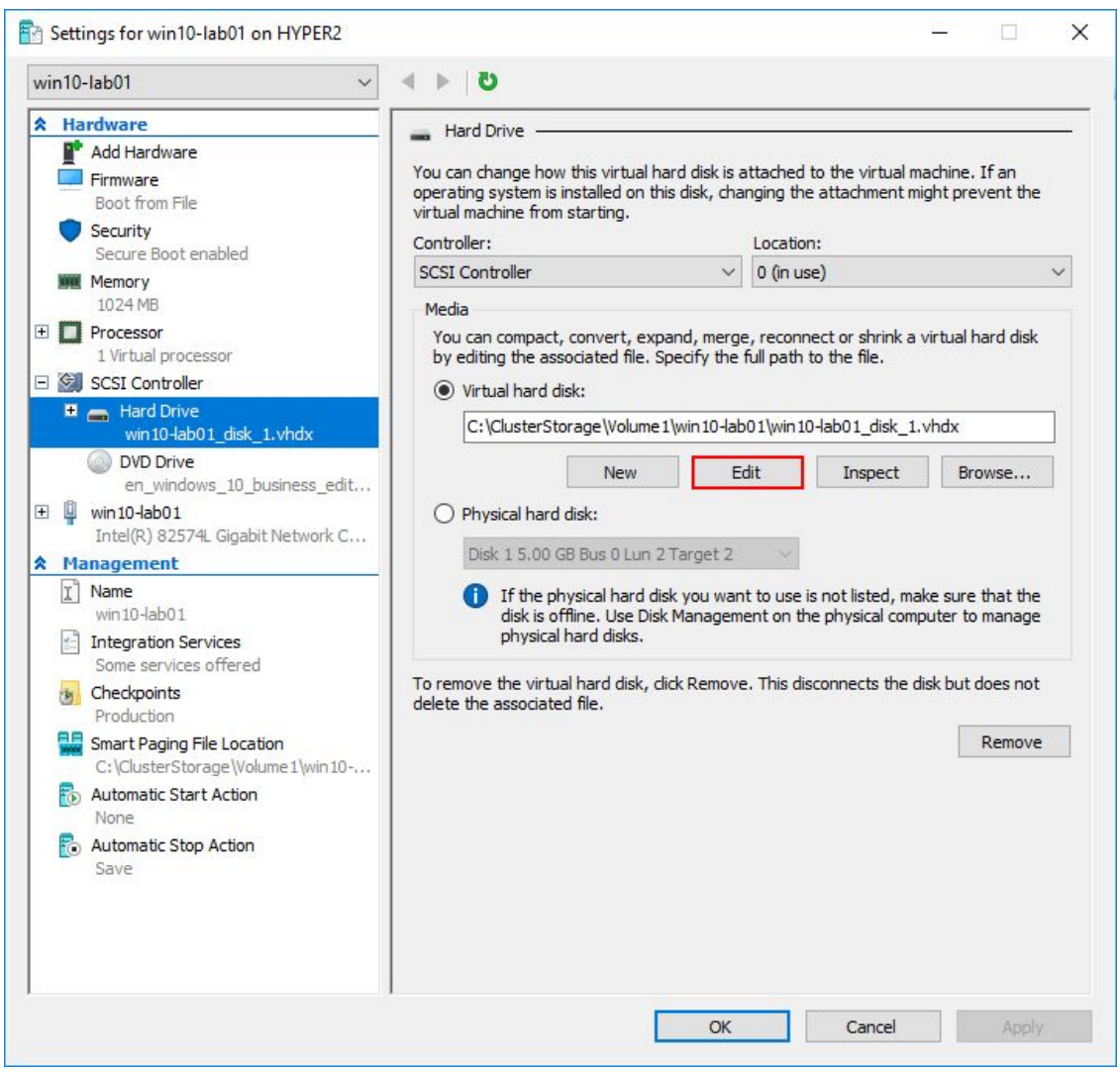
Resizing VHDX Virtual Disk Files

Starting with Windows Server 2012 R2, you can no perform a resize operation on a virtual hard disk of a running virtual machine in Hyper-V. This was not possible with previous versions of Hyper-V as the virtual machine had to be powered off. The new functionality is called **dynamic resize** which allows increasing and decreasing the size of a file while virtual machines are running which has opened up a good deal of possibilities for organizations to do maintenance operations while production virtual machines are running.

What are the requirements for resizing VHDX files?

- Must be VHDX, this is not available for VHD files
- Must be attached to SCSI controller

This can be done via the GUI with Hyper-V manager or using PowerShell. Choose the **Edit** option for the virtual disk file and then you can choose to Expand, Shrink, or Compact.



Choose to Edit the virtual disk to expand or shrink the VHDX file

With PowerShell, you run the **Resize-vhd** cmdlet to resize. You can easily see the information regarding the virtual hard disk with the **get-vhd** cmdlet.

```
PS C:\Windows\system32> get-vhd c:\clusterstorage\volume1\win10-lab01\win10-lab01_disk_1.vhdx


ComputerName            : HYPER2
Path                    : c:\clusterstorage\volume1\win10-lab01\win10-lab01_disk_1.vhdx
VhdFormat               : VHDX
VhdType                 : Dynamic
FileSize                : 15776874496
Size                    : 42949672960
MinimumSize             : 42948641280
LogicalSectorSize       : 512
PhysicalSectorSize      : 4096
BlockSize               : 2097152
ParentPath              :
DiskIdentifier          : 0064AC32-57AF-45A3-98D5-48F226E260FD
FragmentationPercentage : 100
Alignment               : 1
Attached                : True
DiskNumber              :
Number                  :
```

Using get-vhd cmdlet to see the FileSize, Size, and MinimumSize parameters of the virtual disk

Below we are using the **resize-vhd** cmdlet to resize the file to the minimum size. You can see the file size has indeed changed when comparing the above cmdlet return for information compared to the below returned file size information. The minimumsize parameter will resize the VHDX to the smallest possible size. Again, this can be done while the virtual machine is powered on.

```
PS C:\Windows\system32> resize-vhd c:\clusterstorage\volume1\win10-lab01\win10-lab01_disk_1.vhdx -tominimumsize
PS C:\Windows\system32> get-vhd c:\clusterstorage\volume1\win10-lab01\win10-lab01_disk_1.vhdx


ComputerName            : HYPER2
Path                    : c:\clusterstorage\volume1\win10-lab01\win10-lab01_disk_1.vhdx
VhdFormat               : VHDX
VhdType                 : Dynamic
FileSize                : 15279849472
Size                    : 42948641280
MinimumSize             : 42948641280
LogicalSectorSize       : 512
PhysicalSectorSize      : 4096
BlockSize               : 2097152
ParentPath              :
DiskIdentifier          : 0064AC32-57AF-45A3-98D5-48F226E260FD
FragmentationPercentage : 100
Alignment               : 1
Attached                : False
DiskNumber              :
Number                  :
```

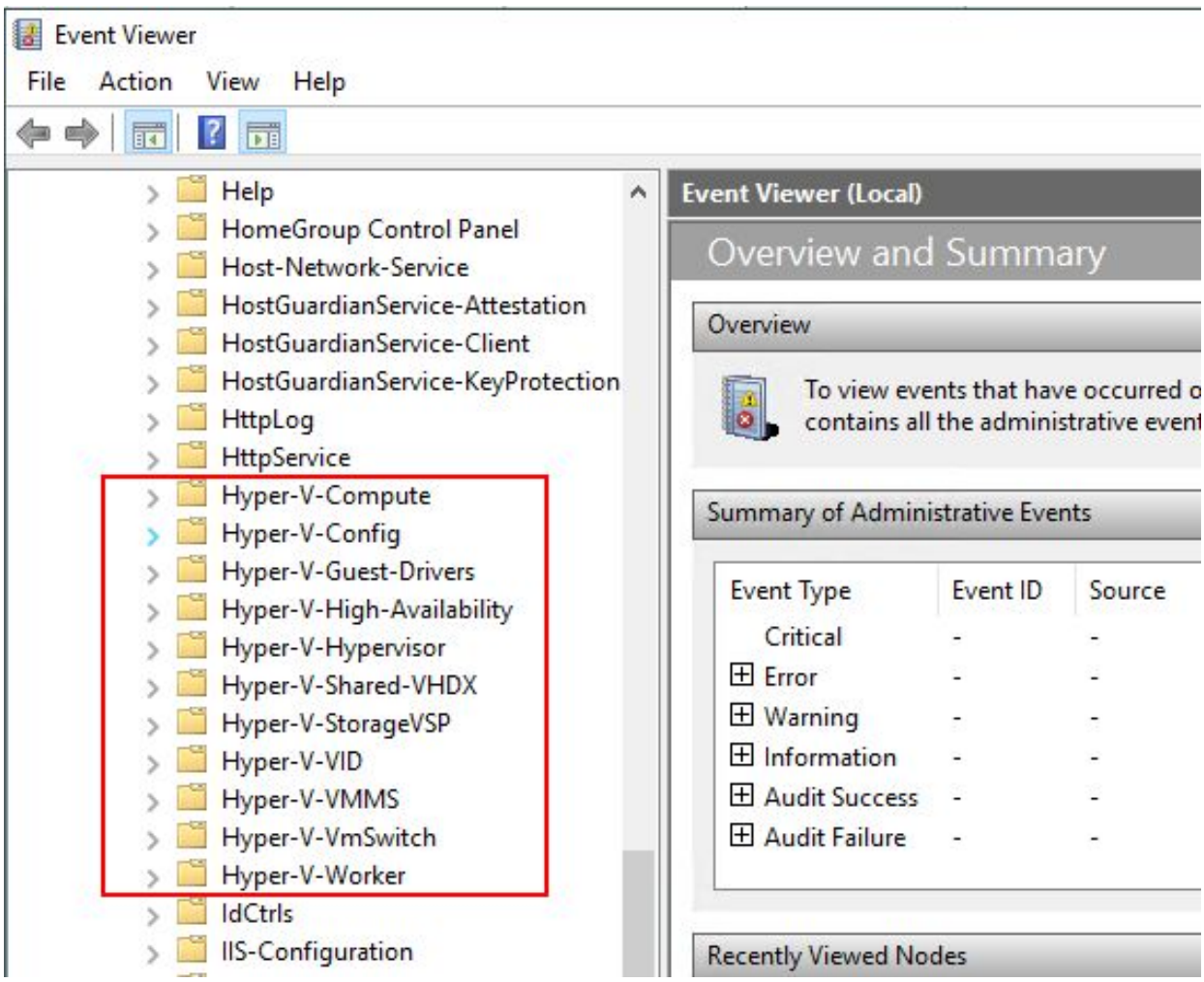
Resizing the virtual disk to the smallest possible size using the -tominimumsize parameter

Troubleshooting Hyper-V with Event Logs

The event logs in Windows Server has typically not received the most welcomed reaction from administrators. Typically, most administrators do not like scouring through log entries trying to find the source of the issue. However, with Windows Server hosting Hyper-V, Microsoft has done a much better job of specifically capturing Hyper-V events and organizing the Hyper-V specific logs in such a way that they make sense and are much more intuitive. There are (11) different log files that are used to capture Hyper-V information in typical event viewer fashion, albeit in a much more useful manner. Windows Server 2016 contains the following log file groupings to help with troubleshooting Hyper-V environments:

- **Hyper-V-Compute** – Captures information regarding the container management API known as the Host Compute Service (HCS) and serves as a low-level management API
- **Hyper-V-Config** – Captures events related to the virtual machine configuration files. Errors that involve virtual machine configuration files either missing, corrupt, or otherwise inaccessible will be logged here
- **Hyper-V-Guest-Drivers** – Log file that contains information regarding the Hyper-V integration services components and provides valuable information in regards to troubleshooting issues with the integration components.
- **Hyper-V-High-Availability** – Events related to Hyper-V Windows Server Failover Clusters
- **Hyper-V-Hypervisor** – Events related to the Hyper-V hypervisor itself. If Hyper-V fails to start, look here. Also, informational messages such as Hyper-V partitions created or deleted will be logged here
- **Hyper-V-Shared-VHDX** – Information specific to shared VHDX virtual disks between virtual machines are found in this log
- **Hyper-V-StorageVSP** – Captures information regarding the Storage Virtualization Service Provider. This contains low-level troubleshooting information for virtual machine storage.
- **Hyper-V-VID** – Logs events from the Virtualization Infrastructure Driver regarding memory assignment, dynamic memory, or changing static memory with a running virtual machine
- **Hyper-V-VMMS** – Virtual Machine Management Service events which are valuable in troubleshooting a virtual machine that won't start or a failed Live Migration operation
- **Hyper-V-VmSwitch** – Contains events from the virtual network switches
- **Hyper-V-Worker** – the log that captures Hyper-V worker process information which is responsible for the actual running of the virtual machine.

To find the various Hyper-V specific events logs in the Windows Event Viewer, navigate to **Windows Logs >> Applications and Services Logs >> Microsoft >> Windows**



Hyper-V specific event viewer logs useful in troubleshooting

Taking Hyper-V Troubleshooting with Event Viewer Further

Even though Microsoft has organized the event viewer groups into fairly logical and intuitive channels, some may desire to take the event viewer a step further in the direction of consolidating all the logs into a single view for more easily piecing together issues or troubleshooting an underlying problem. Switching between the different logs may be a bit cumbersome, especially if more than one Hyper-V component is at play in the issue or various parts of the overall problem picture are found in different logs. There is a GitHub PowerShell module that can be downloaded that allows enabling all the important Windows event channels into a single **evtx** file to help with troubleshooting.

There are a couple of steps to take advantage of the PowerShell module from GitHub. First you need to download and import the PowerShell module, then you reproduce the issue which should capture the relevant information in the logs.

Below is a synopsis of the steps found here: https://blogs.technet.microsoft.com/virtualization/2017/10/27/a-great-way-to-collect-logs-for-troubleshooting/

Download the PowerShell module and import it

 # Download the current module from GitHub

Invoke-WebRequest "https://github.com/MicrosoftDocs/Virtualization-Documentation/raw/live/hyperv-tools/HyperVLogs/HyperVLogs.psm1" -OutFile "HyperVLogs.psm1"

 # Import the module

Import-Module .\HyperVLogs.psm1

Reproduce the Issue and Capture the Logs

 # Enable Hyper-V event channels to assist in troubleshooting

Enable-EventChannels -HyperVChannels VMMS, Config, Worker, Compute, VID

 # Capture the current time to a variable

$startTime = [System.DateTime]::Now

 # Reproduce the issue here

 # Write events that happened after "startTime" for the defined channels to a named directory

Save-EventChannels -HyperVChannels VMMS, Config, Worker, Compute, VID -StartTime $startTime

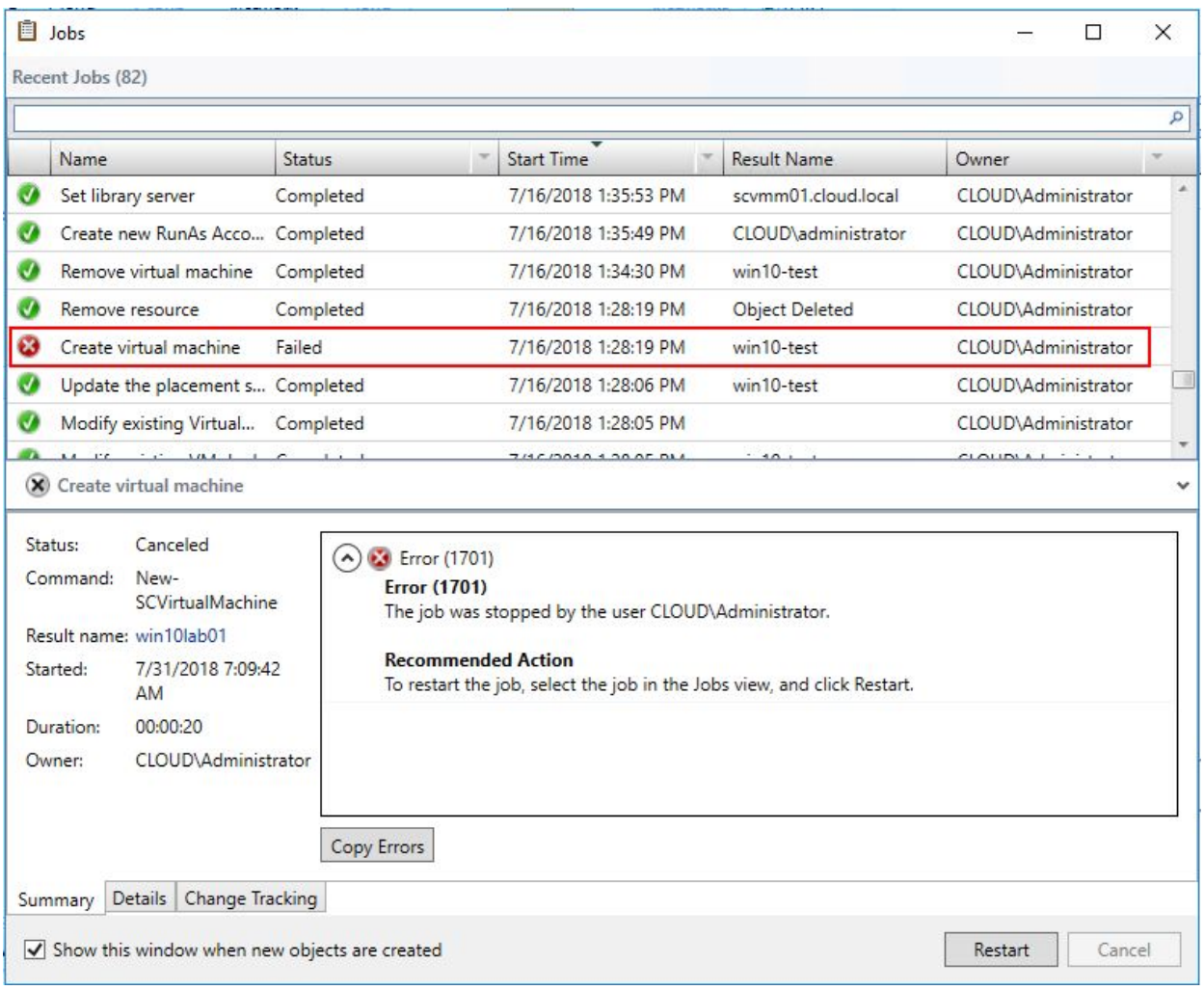
 # Disable the analytical and operational logs -- by default admin logs are left enabled

Disable-EventChannels -HyperVChannels VMMS, Config, Worker, Compute, VID

System Center Virtual Machine Manager Logging

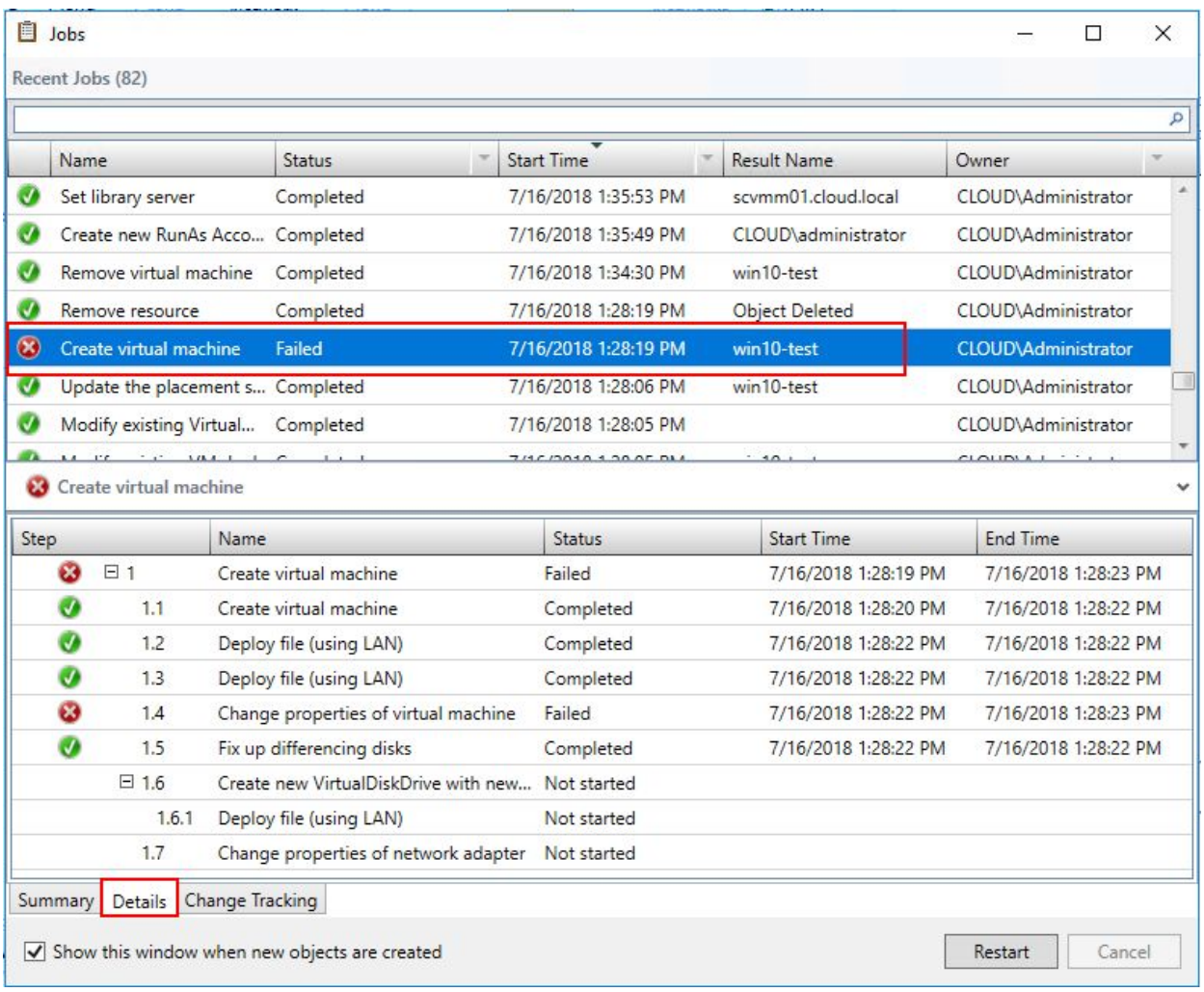
System Center Virtual Machine Manager or SCVMM provides a powerful interface to managing and troubleshooting Hyper-V environments. In the Hyper-V world, SCVMM provides a "vCenter-like" experience with centralized management of your hosts and clusters. When using System Center Virtual Machine Manager with the central point of management for Hyper-V, administrators have the ability to have a single pane of glass look at multiple Hyper-V hosts. Particularly, the **Jobs** view in System Center Virtual Machine Manager provides a view of all actions in the Hyper-V environment. Taking it a step further, the **Details** tab of the Jobs view provides a step-by-step overview of the action and any sub component part of a task that failed.

Below, a failed create virtual machine task shows the status of **Failed**. What caused the job to fail? The **Details** view allows digging further.



Looking at Recent Jobs tasks

On the **Details** tab, System Center Virtual Machine Manager provides a detailed step-by-step overview of all the steps involved in the particular task executed in SCVMM. Note, below, how SCVMM enumerates the individual steps, and shows the exact point the task presented with a failure – "change properties of virtual machine". This extremely helpful when you are looking to detail exactly what is causing a global task to fail.



A look at the Details tab in the Jobs view in SCVMM providing details of tasks and failures in Hyper-V

System Center Virtual Machine Manager SCVMM – Overview

- Most Windows Server administrators are familiar with the System Center suite of products. System Center Virtual Machine Manager or SCVMM simplifies the administration, configuration, and management of Windows Server Hyper-V environments. It does this by bringing all the tools, management, and administration of Hyper-V hosts and clusters under a single management tool. Without SCVMM, Hyper-V administrators manage their Hyper-V environment by using a combination of tools with no one tool being the all-encompassing tool for administration. These include the well-known Hyper-V manager, Failover Cluster Manager, and PowerShell. Typically, these tools complement one another and none of the tools mentioned are used solely in and of themselves for managing Hyper-V. Hyper-V administrators may use Hyper-V manager to configure their Hyper-V host virtual switches, but then use Failover Cluster Manager to configure a highly available virtual machine. Then, they may use PowerShell to configure affinity rules.
- With System Center Virtual Machine Manager, all of these various features and functionality and much more are brought under a single pane-of-glass dashboard and allows for consolidated management of the entire Hyper-V infrastructure. System Center 2016 brings to the mix cloud support, enabling seamless management of complex hybrid cloud environments with both on-prem and Azure public cloud workloads.
- As mentioned, System Center Virtual Machine Manager is a pay for product and is licensed and integrated along with the System Center family of products. You cannot license System Center Virtual Machine Manager as a standalone product. For further details on buying/licensing System Center, take a look at the following link:
- https://www.microsoft.com/en-us/cloud-platform/system-center-pricing

Pricing and licensing overview for System Center

Datacenter and Standard Edition overview	Datacenter Edition for managing virtual servers	Standard Edition for managing physical servers
Operating system environments / Hyper-V Containers	Unlimited	2[1]
Windows Server Containers	Unlimited	Unlimited
Open NL L&SA 2-year price (licensed by cores)	$3,607	$1,323

Pricing Overview for System Center (image courtesy of Microsoft)

System Center Virtual Machine Manager SCVMM – Features

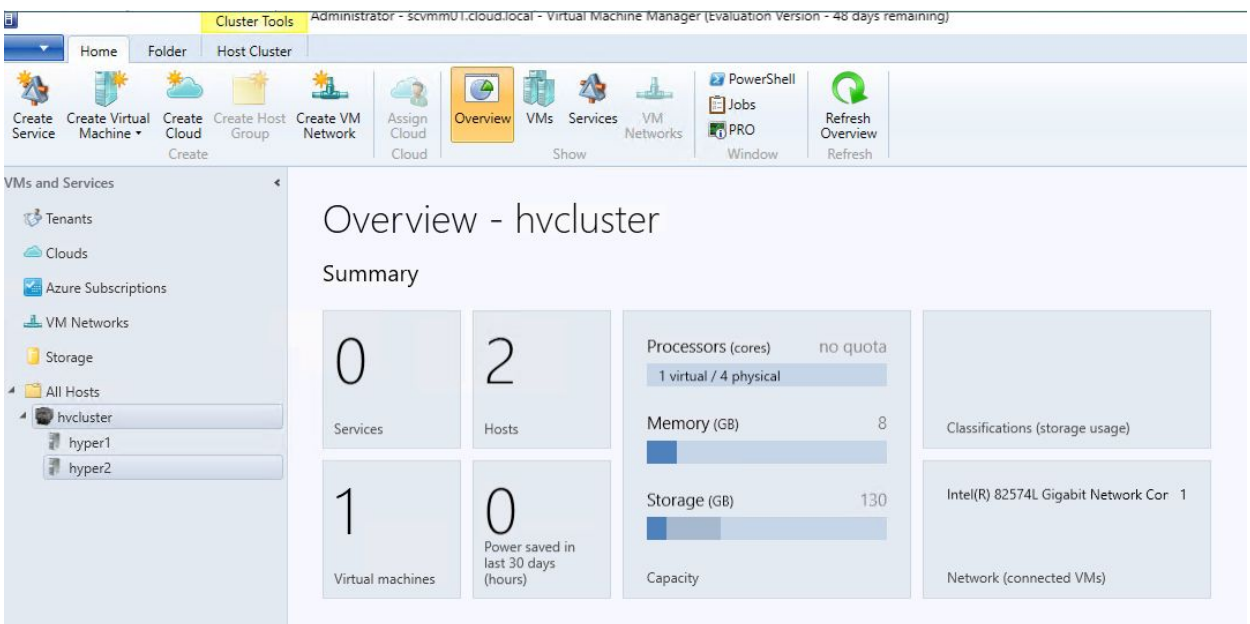
System Center Virtual Machine Manager contains really great management features for Hyper-V environments, allowing administrators to successfully manage Hyper-V at scale in the enterprise datacenter. Among the features and functionality afforded Hyper-V administrators by SCVMM are the following:

- **Windows PowerShell** – Windows PowerShell is the premier scripting language for use by Windows Server administrators today. SCVMM allows IT administrator to take advantage of the fully scriptable capabilities of SCVMM and run scripts against multiple VMs.
- **Integrated Physical to Virtual (P2V) Conversions** – Most organizations today are looking to virtualize physical resources if they still have physical servers around. SCVMM allows easily performing P2V operations.
- **Virtual Machines Intelligent Placement** – SCVMM allows automatically profiling Hyper-V hosts in the environment and placing VMs on the host that has the best fit for hosting those resources.
- **Centralized Resource Management and Optimization** – One of the primary advantages of SCVMM is the centralized management it offers. Hyper-V administrators have all the tools and management for Hyper-V hosts and clusters in a single location.
- **Virtual Machine Rapid Deployment and Migration** – SCVMM allows creating virtual machine templates that allow rapidly deploying virtual machines from master VM templates. Service templates allow creating complete groups of virtual machines and deploys them together as a single object that provides an application(s).
- **Centralized Resource Library** – This component of SCVMM allows building a library of all resources required to build virtual machines, including ISO images, scripts, profiles, guest operating system profiles, and virtual machine templates. The Centralized Library facilitates the rapid deployment of virtual machines.
- **Centralized Monitoring and Reporting** – Centralized monitoring and reporting of the entire Hyper-V infrastructure allows administrators to quickly and easily monitor the environment.
- **Self-service Provisioning** – SCVMM administrators can delegate controlled access to end users for specific virtual machines, templates, and other resources through a web-based portal. This is especially powerful in DEV/TEST where developers may need to quickly provision new VMs for themselves according to the controls
- **Existing SAN Networks** – SCVMM allows taking advantage of existing SAN networks for use in the environment. SCVMM can automatically detect and use existing SAN infrastructure to transfer virtual machine files.

Managing Hyper-V Host and Clusters with SCVMM

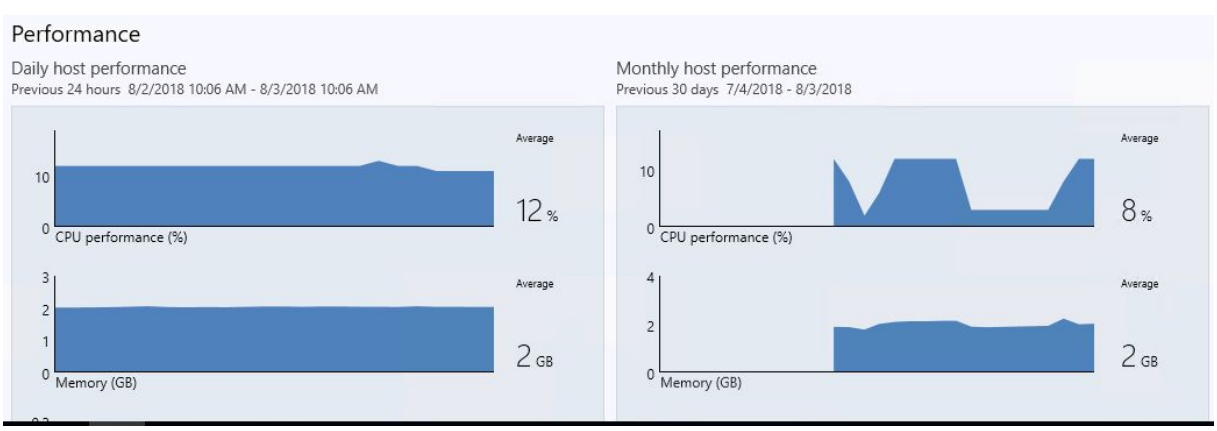
SCVMM makes managing Hyper-V environments much easier and streamlined for Hyper-V administrators. Below, let's take a look at a few screenshots of managing Hyper-V environments with SCVMM, including many of the above-mentioned capabilities.

System Center Virtual Machine Manager allows seeing an overview of current performance not only across Hyper-V hosts, but also Hyper-V clusters.



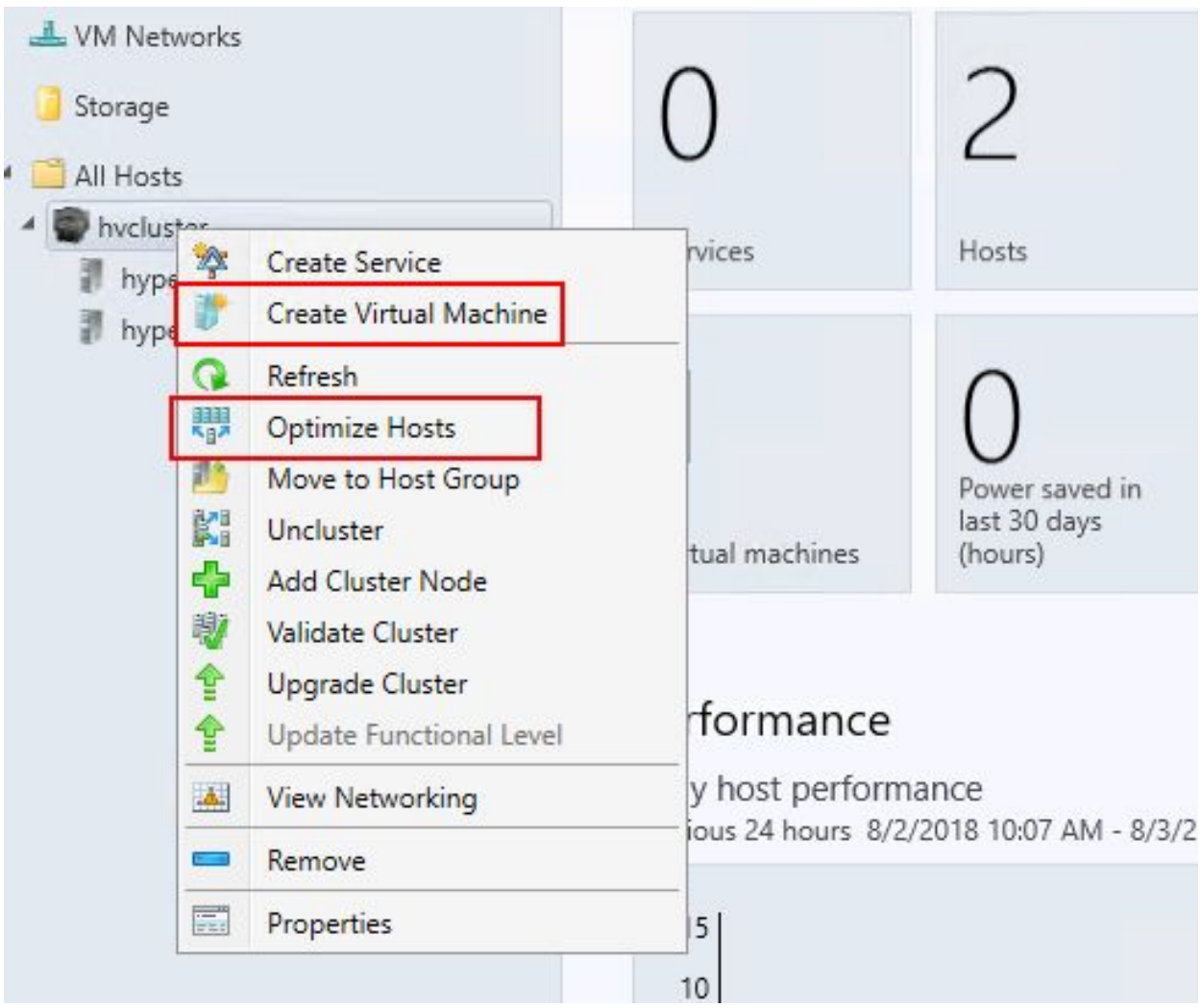
SCVMM provides a centralized view of Hyper-V clusters including individual hosts

A deeper look at **Performance**, including Daily host performance and Monthly host performance metrics.



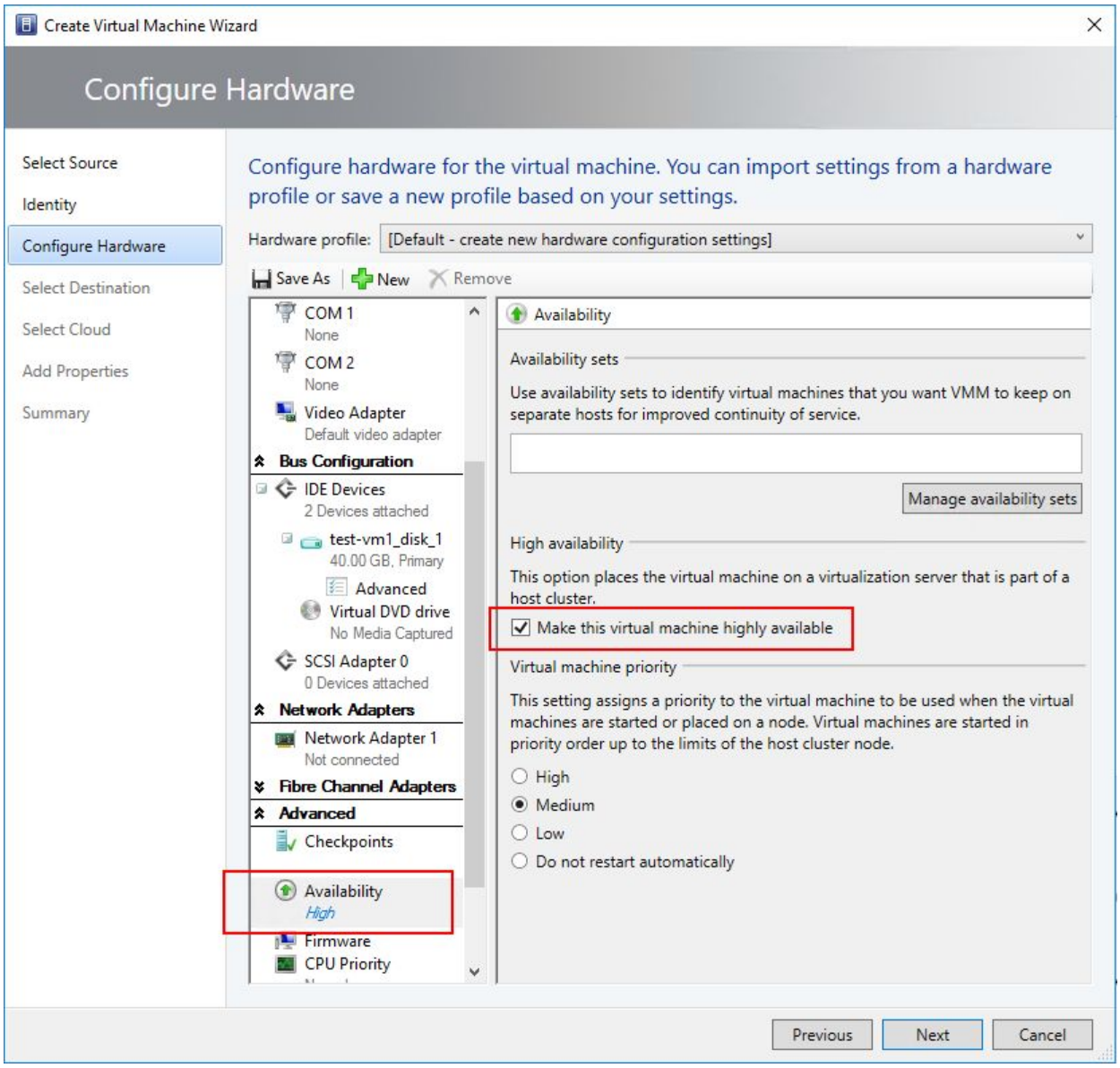
A look at daily and monthly performance metrics

Using System Center Virtual Machine Manager, you can easily create **High Availability** virtual machines. Additionally, hosts can easily be optimized.



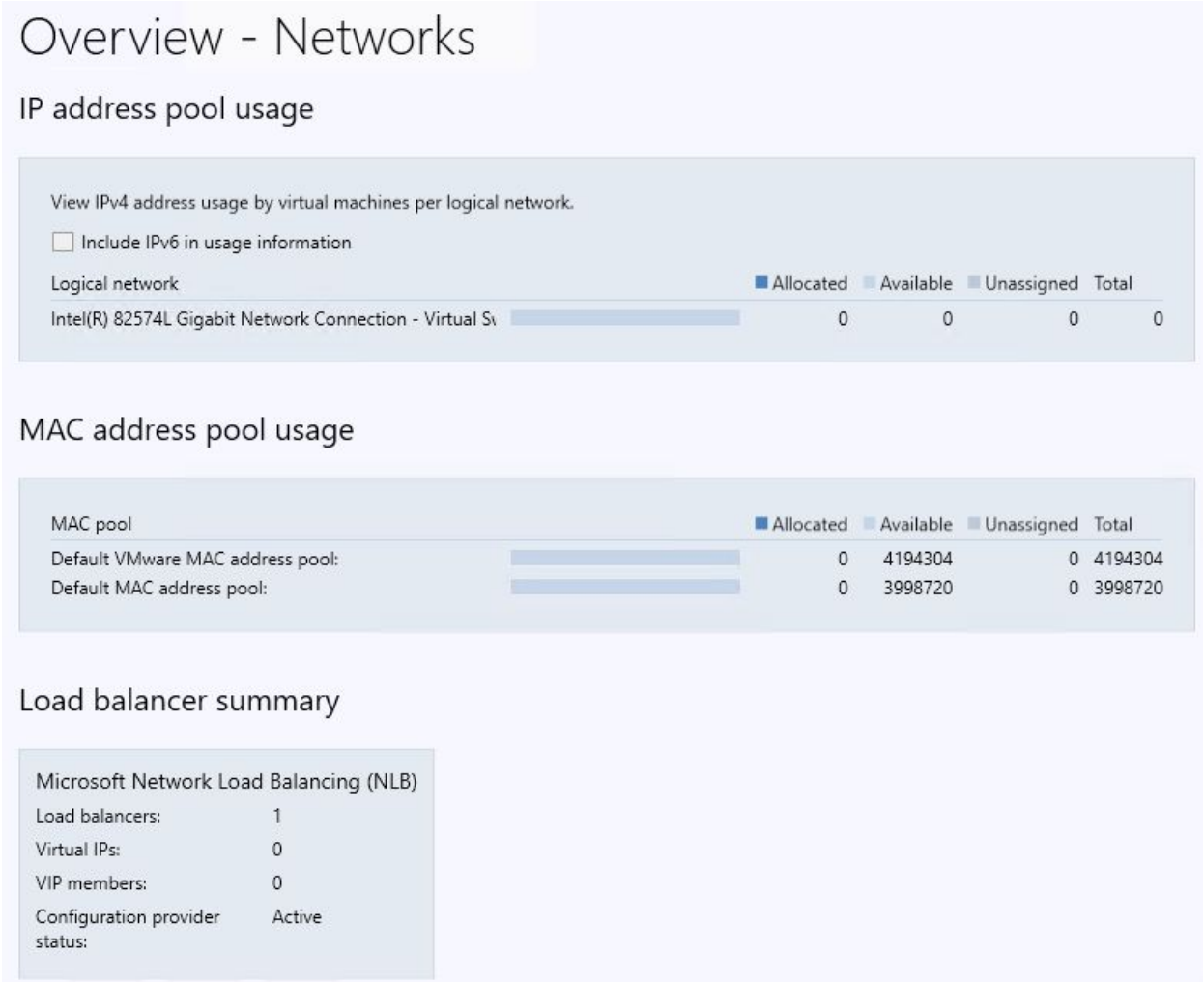
Creating High Availability Virtual Machines using SCVMM

In the **Create Virtual Machine Wizard**, SCVMM allows configuring the **Availability** of the virtual machine that is housed on a Hyper-V cluster as well as the **Start** priority.



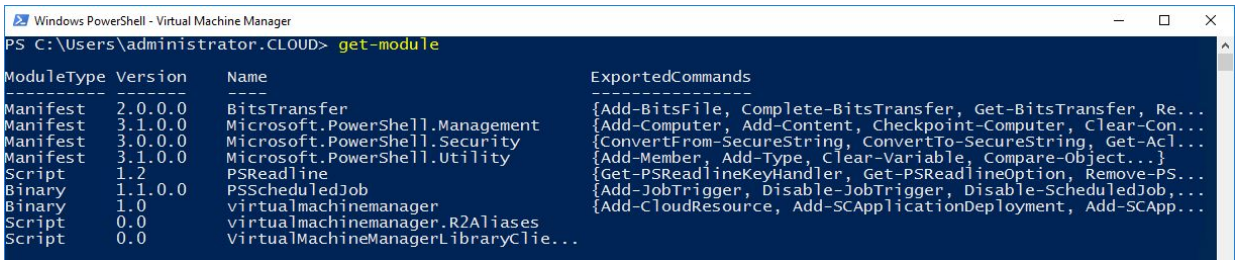
Configuring a new virtual machine using SCVMM

Configuring, monitoring, and managing Hyper-V networks is easily done with SCVMM. It allows administrators to easily see all network adapters on hosts, logical networks, virtual switches, MAC addresses, etc.



SCVMM network overview inside a Hyper-V cluster

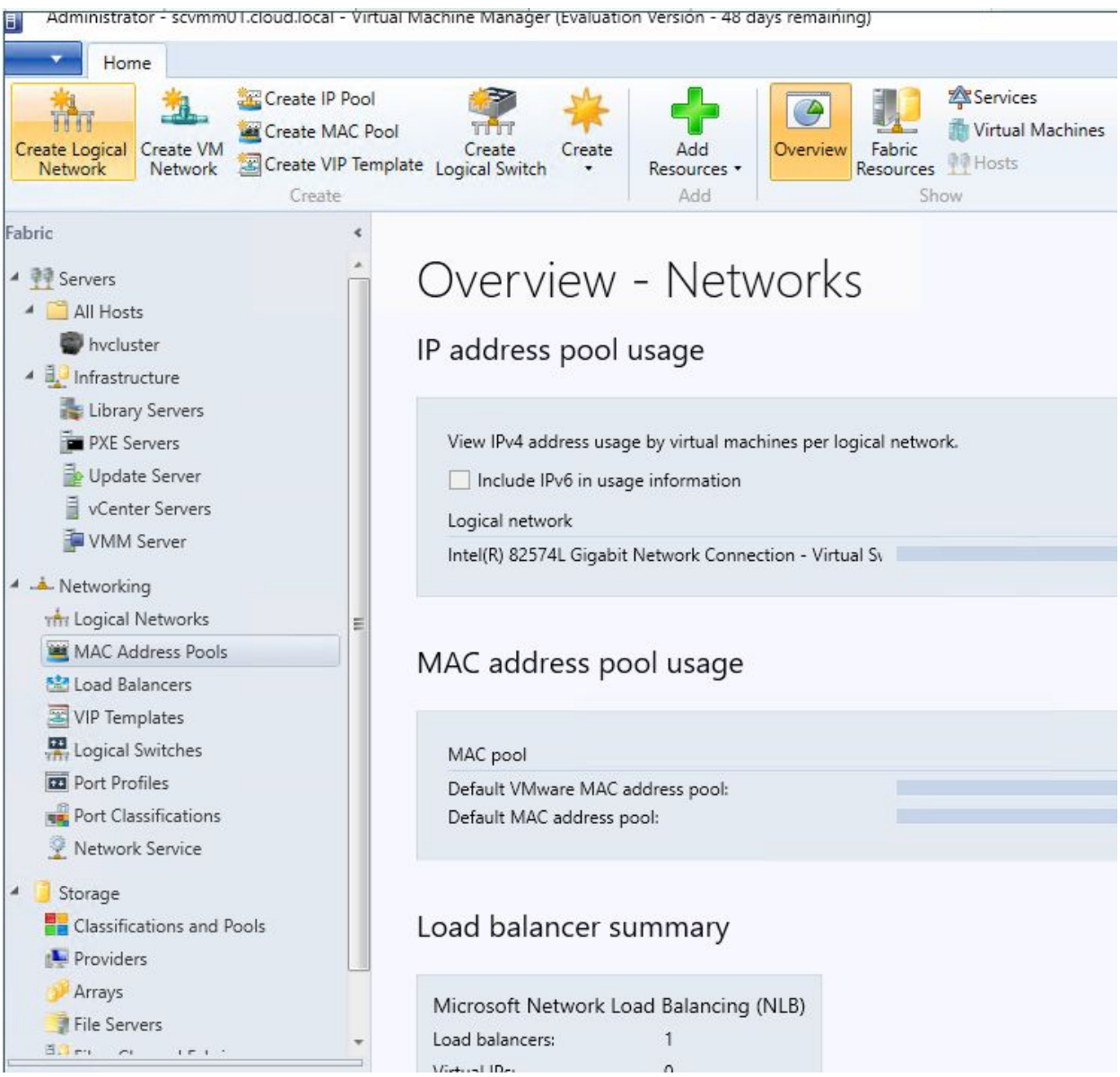
Launching PowerShell from Virtual Machine Manager yields a number of modules that allow programmatically interacting with SCVMM and a Hyper-V environment.



Getting modules in a Virtual Machine Manager PowerShell environment

SCVMM allows a wealth of visibility and configurability from a **Fabric** standpoint. Take a look at the configuration options for Networking as an example. Administrators can configure:

- Logical Networks
- MAC Address Pools
- Load Balancers
- VIP Templates
- Logical Switches
- Port Profiles
- Port Classifications
- Network Service



Network Fabric Configuration in SCVMM

Is System Center Virtual Machine Manager Required?

When thinking about System Center Virtual Machine Manager and managing production Hyper-V environments, one may ask, is it required to have SCVMM? The answer to that question is most likely “No”. However, would it be extremely valuable to use SCVMM in managing your Hyper-V environment? The answer to that question is probably “Yes”. SCVMM makes a lot of sense for organizations who may scale beyond just a few hosts and clusters. For larger environments with even moderate to large deployments of Hyper-V, SCVMM becomes more necessary to maintain a management layer of the environment that is efficient and consolidated. Without SCVMM, the manual processes and workload can increase dramatically. The overall answer then is “it depends”. Businesses will have to analyze their individual environments and use cases to justify the cost of SCVMM. However, SCVMM can quickly offset the initial cost with the return yielded by much more efficient management and operational expenses recouped in administration time.