

Dell EMC PowerVault ME4 Series and Microsoft Hyper-V

Abstract

This document provides best practices for configuring Microsoft® Hyper-V® to perform optimally with Dell EMC™ PowerVault™ ME4 Series storage.

September 2018

Revisions

Date	Description
September 2018	Initial release

Acknowledgements

Author: Marty Glaser

The information in this publication is provided “as is.” Dell Inc. makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose.

Use, copying, and distribution of any software described in this publication requires an applicable software license.

© 2018 Dell Inc. or its subsidiaries. All Rights Reserved. Dell, EMC, Dell EMC and other trademarks are trademarks of Dell Inc. or its subsidiaries. Other trademarks may be trademarks of their respective owners.

Dell believes the information in this document is accurate as of its publication date. The information is subject to change without notice.

Table of contents

Revisions.....	2
Acknowledgements.....	2
Table of contents	3
Executive summary.....	4
Audience	4
1 Introduction.....	5
1.1 ME4 Series overview	5
1.2 Microsoft Hyper-V overview.....	6
1.3 Best practices overview.....	6
1.4 General best practices for Hyper-V	7
2 Design best practices	8
2.1 Right-size the storage array	8
2.2 Linear and virtual disk groups, pools, and RAID configuration	8
2.3 Determine optimal transport and front-end configuration.....	9
3 Administration best practices	11
3.1 Guest integration services	11
3.2 Hyper-V guest VM generations	13
3.3 Virtual hard disks	14
3.4 Present ME4 Series storage to Hyper-V	18
3.5 Optimize format disk wait time for large volumes.....	22
3.6 Placement of page files	22
3.7 Placement of Active Directory domain controllers.....	23
3.8 Queue depth best practices for Hyper-V	23
4 ME4 Series snapshots with Hyper-V.....	25
4.1 Crash-consistent and application-consistent snapshots	25
4.2 Guest VM recovery with ME4 Series snapshots	25
4.3 Create test environment with ME4 Series snapshots.....	29
4.4 Migrate guest VMs with ME4 Series storage	29
A Technical support and additional resources.....	30
A.1 Related resources	30

Executive summary

This document provides best practices for deploying Microsoft® Windows Server® Hyper-V® based solutions with Dell EMC™ PowerVault™ ME4 Series storage systems. It builds upon the resources listed in appendix A.1.

Before configuring an ME4 Series array to work optimally with Hyper-V, review the primary reference documents including the ME4 Series *Administrator's Guide* and *Deployment Guide* on [Dell.com/support](https://dell.com/support). The information in these two guides is supplemented by the best practices in this document.

Audience

This document is intended for Dell EMC customers, partners, and employees who desire to learn more about best practices when configuring Hyper-V with ME4 Series storage systems. It is assumed the reader has working knowledge of ME4 Series storage and Hyper-V.

We welcome your feedback along with any recommendations for improving this document. Send comments to StorageSolutionsFeedback@dell.com.

1 Introduction

Microsoft Hyper-V and Dell EMC PowerVault ME4 Series storage are feature-rich solutions that together present a diverse range of configuration options to solve key business objectives such as storage capacity, performance, and resiliency. This section provides an overview of ME4 Series storage, Microsoft Hyper-V, and general best practices for the solution described in this paper.

1.1 ME4 Series overview

The ME4 Series includes entry-level storage appliances that provide many features found in more advanced storage solutions. The base models include the PowerVault ME4012, ME4024, and ME4084.



Figure 1 Front and rear view of the PowerVault ME4024 array, configured with 24 SSD drives and dual controllers

The ME4 Series 2U-chassis models include the ME4012 array which supports up to twelve 3.5-inch drives, and the ME4024 array which supports up to twenty-four 2.5-inch drives. The ME4084 array (5U chassis) supports up to eighty-four 2.5-inch drives. All three models support additional drive capacity by adding expansion enclosures.

Key features of the ME4 Series include the following:

- Simple, customer-installable design
- Intuitive web-based GUI and CLI tools for system configuration and management
- All-inclusive feature-licensing model
- Support for all-flash, spinning, and hybrid drive configurations
- Linear and virtual disk group and pool configuration options
- Thin-provisioning with virtual disk groups for storage efficiency, along with storage tiering for intelligent real-time data placement for hot and cold data
- A variety of RAID levels and hot spare configurations, including distributed sparing with the new ADAPT RAID option
- Full support for multipath I/O (MPIO) with up to eight front-end (FE) ports per array (four per controller head)
- Active-active controller configuration that permits asymmetrical logical unit access (ALUA) aware hosts to automatically designate MPIO FE paths as optimal or non-optimal
- FE transport options including 12Gb SAS, 8Gb/16Gb Fibre Channel (FC), and 1Gb/10Gb iSCSI

- Support for mixed transport environments (FC and iSCSI)
- Up to nine back-end (BE) expansion enclosures can be added to each ME4 Series array with 12Gb SAS to expand drive capacity
- Support for up to 336 drives with up to 4 petabytes (PB) raw capacity in the ME4084 array
- Direct-attached storage (DAS) support for FE ports (SAS, FC, and iSCSI)
- Storage area network (SAN) support for FE ports connected to FC and iSCSI switches (FE SAS supports DAS only)
- Scheduled and on-demand volume snapshots with rollback and refresh options
- Asynchronous replication over FC or iSCSI to another ME4 Series array for DR protection

Note: Most of these features work seamlessly in the background, regardless of the platform. In most cases, the default settings for these features work well with Hyper-V or at least serve as good configuration starting points. This document highlights additional configuration or tuning steps that may enhance performance, usability, or other factors.

To learn more about these and other ME4 Series features, refer to the ME4 Series *Administrator's Guide* and *Deployment Guide*, and the additional documentation listed in appendix A.

1.2 Microsoft Hyper-V overview

The Windows Server platform leverages Hyper-V for virtualization technology. Initially offered with Windows Server 2008, Hyper-V has matured with each release to include many new features and enhancements. The ME4 Series supports Windows Server 2012 Hyper-V and Windows Server 2016 Hyper-V.

Note: In January 2020, Microsoft will discontinue patches and security updates for Windows Server 2008 R2 (end of support). Customers still running Windows Server 2008 R2 should plan to migrate their Hyper-V environments before support ends.

Microsoft Hyper-V has evolved to become a mature, robust, proven virtualization platform. In simplest terms, it is a layer of software that presents the physical host server hardware resources in an optimized and virtualized manner to guest virtual machines (VMs). Hyper-V hosts (also referred to as nodes when clustered) greatly enhance utilization of physical hardware (such as processors, memory, NICs, and power) by allowing many VMs to share these resources at the same time. Hyper-V Manager and related management tools such as Failover Cluster Manager, Microsoft System Center Virtual Machine Manager (SCVMM), and PowerShell®, offer administrators great control and flexibility for managing host and VM resources.

Note: Many core Hyper-V features (such as dynamic memory) are storage agnostic, and are not covered in detail in this guide. To learn more about core Hyper-V features, functionality, and general best practices, see the [Hyper-V Best Practices Checklist](#) and other resources on [Microsoft TechNet](#).

1.3 Best practices overview

Best practices are typically based on and developed from the collective wisdom and experience of many users over time, and this learning is built into the design of next-generation products. With mature technologies such as Hyper-V or Dell EMC storage arrays, best practices are already factored in to the default configurations, settings, and recommendations.

Because default settings typically incorporate best practices, tuning is often unnecessary (and discouraged) unless a specific design, situation, or workload is known to benefit from a different configuration. For example,

the default queue-depth setting works well for most hosts in a SAN environment. However, increasing the queue depth for a large sequential workload running on a small number of hosts might result in a significant performance increase, while doing the same for a non-sequential workload running on many hosts might have the opposite result, degraded performance. One of the purposes of a best-practices document is to call attention to situations where using a default setting or configuration may not be optimal.

Some common goals of best practices include:

- Minimize complexity and administrative overhead
- Optimize the performance of a workload
- Maximize security
- Ensure resiliency and recoverability
- Maximize return on investment over the life of the hardware

It is important to remember that best practices are baselines that may not be ideal for every environment. Some notable exceptions include the following:

- In some cases, legacy systems that are performing well and have not reached their life expectancy may not adhere to current best practices. The best course of action may be to run legacy configurations until they reach their life expectancy because it is too disruptive or costly to make changes outside of a normal hardware progression or upgrade cycle. Dell EMC recommends upgrading to the latest technologies and adopting current best practices at key opportunities such as when upgrading or replacing infrastructure.
- A common best practices tradeoff is to implement a less-resilient design (to save cost and reduce complexity) in a test or development environment that is not business critical.

Note: While following the best practices in this document is strongly recommended by Dell EMC, some recommendations may not apply to all environments. For questions about the applicability of these guidelines in your environment, contact your Dell EMC representative.

1.4 General best practices for Hyper-V

There are many general best practices for Hyper-V not specific to storage that are not discussed in detail in this document. See resources such as [Microsoft TechNet](#) for guidance on general Hyper-V best practices.

The following provides a high-level summary of some of the most common best practices tuning steps for Hyper-V:

- Minimize or disable unnecessary hardware devices and services to free up host CPU cycles that can be used by other VMs (this also helps to reduce power consumption).
- Schedule tasks such as periodic maintenance, backups, malware scans, and patching to run after hours, and stagger start times when such operations overlap and are CPU or I/O intensive.
- Tune application workloads to reduce or eliminate unnecessary processes or activity.
- Leverage Microsoft PowerShell or other scripting tools to automate step-intensive repeatable tasks to ensure consistency and avoid human error. This can also reduce administration time.

2 Design best practices

This section provides guidance on sizing and configuration options for ME4 Series storage and Hyper-V.

2.1 Right-size the storage array

Before deploying a new ME4 Series storage array, it is important to consider the environmental design factors that impact storage capacity and performance so that new or expanded storage is right-sized for the environment. If the ME4 Series array will be deployed to support an existing Hyper-V workload, metrics such as storage capacity and I/O demands might already be understood. If the environment is new, these factors need to be determined to correctly size the storage array.

Many common short- and long-term problems can be avoided by making sure the storage part of the solution will provide the right capacity and performance in the present and future. Scalability is a key design consideration. For example, Hyper-V clusters can start small with two nodes, and expand one node at a time, up to a maximum of 64 nodes per cluster. Storage including ME4 Series arrays can start with a small number of drives, and expand capacity and I/O performance over time by adding expansion enclosures with more drives as workload demands increase.

Optimizing performance is a process of identifying and mitigating design limitations that cause bottlenecks — the point at which performance begins to be impacted under load because a capacity threshold is reached somewhere within the overall design. The goal is to maintain a balanced configuration that allows the workload to operate at or near peak efficiency.

One common mistake made when sizing a storage array is assuming that total disk capacity translates to disk performance. Installing a small number of large-capacity spinning drives in an array does not automatically translate to high performance just because there is a lot of available storage capacity. There must be enough of the right kind of drives to support the I/O demands of a workload in addition to raw storage capacity.

Where available, customers can confidently use the configuration guidance in Dell EMC storage reference architecture white papers as good baselines to right-size their environments.

Work with your Dell EMC representative to complete a performance evaluation if there are questions about right-sizing an ME4 Series storage solution for your environment and workload.

2.2 Linear and virtual disk groups, pools, and RAID configuration

Choosing the type of disk pools and RAID configurations to use is equally important to right-sizing the ME4 Series storage array for capacity and I/O.

The ME4 Series *Administrator's Guide* provides an in-depth review and comparison of linear and virtual disk groups, pools, the different RAID levels and hot spare configurations available with each, the trade-offs of choosing one over the other, and application (workload) recommendations for each.

One option discussed in the *Administrator's Guide* is the ME4 Series ADAPT option for RAID. ADAPT supports distributed sparing for extremely fast rebuild times, and large-capacity disk groups of up to 128 total drives. However, ADAPT requires a minimum of 12 drives to start with, and all disks must be of the same type and be in the same tier.

From the perspective of Hyper-V, any of the available configurations is supported. Choosing the best type of disk group and RAID option is a function of the workload running on Hyper-V, and the ME4 Series *Administrator's Guide* provides basic guidance.

2.3 Determine optimal transport and front-end configuration

The ME4 Series is configurable as direct-attached storage (DAS) or as part of a storage area network (SAN). Supported transports for DAS include SAS, FC, and iSCSI. Supported transports for SAN include FC and iSCSI, but not SAS. These configuration options offer customers great flexibility when designing their environment.

Before reading further, refer to the ME4 Series *Deployment Guide* to gain a thorough understanding of the different DAS, SAN, host, and replication cabling options available with the ME4 Series.



Figure 2 ME4 Series array with four SAS FE ports per controller



Figure 3 ME4 Series array with four CNC FE ports per controller

The ME4 Series is available with two types of physical FE ports:

- Four SAS ports per controller head (eight ports total)
- Four converged network controller (CNC) ports per controller head (eight CNC ports total)
 - CNC ports are logically configured as FC or iSCSI, or a combination of both, depending on customer preference (the type of SFP transceiver used determines the transport and speed of each CNC port)
 - Changing CNC ports from all FC or iSCSI to a mix of FC and iSCSI can be done using the CLI but requires rebooting the controller heads
 - In a mixed-transport configuration, the first two CNC ports on each controller (0 and 1) must be configured as FC ports, as shown in Figure 3

Hyper-V hosts, nodes, and clusters support all the above configuration options. Consider the following recommendations:

- If a Hyper-V environment is likely to scale beyond four physical hosts or nodes attached to the same ME4 Series array, choose the following:
 - Start with a SAN configuration (FC or iSCSI) (recommended)
 - Start with a DAS configuration (FC or iSCSI), and migrate to a SAN configuration when the fifth node needs to be added (caution: this might be very disruptive to the environment because it will require host down time to reconfigure and re-cable the FE ports)
- If the ME4 Series array is configured to replicate to another ME4 Series array, two of the four FE ports (0 and 1) on each controller head must be dedicated to replication traffic. This will limit the available FE ports for host connectivity to the other two ports (2 and 3) on each controller. If the Hyper-V environment is likely to scale beyond two physical hosts or nodes, choose the following:
 - Start with a SAN configuration (FC or iSCSI) (recommended)
 - Start with a DAS configuration (FC or iSCSI), and migrate to a SAN configuration when the third node needs to be added (caution: as stated previously, this might be very disruptive to the environment)
- SAS FE ports are supported in a DAS configuration only. ME4 Series arrays equipped with SAS FE ports do not support replication to another ME4 Series array. SAS FE ports are a good choice if the ME4 Series array will not need to expand beyond four Hyper-V hosts or nodes, and will not need to be configured for replication.

Other factors to consider include the following:

- With DAS, the hosts must be within reach of the physical cable that is used to directly connect the host to the ME4 Series array. This works well if the hosts are in the same or an adjacent rack that is within easy cabling distance.
- Choosing the type of transport is often a function of what is already in place in the environment or according to personal preference. In cases where the infrastructure to support an FC or iSCSI SAN is already in place, customers can continue using this transport to maximize their return on their investment.

3 Administration best practices

3.1 Guest integration services

Guest integration services are a package of virtualization-aware drivers that are installed on a guest VM to optimize the guest VM virtual hardware for interaction with the physical host hardware and storage. Installing these drivers is typically the first step for optimizing VM performance. If a VM is not performing as expected (due to CPU, disk I/O, or network performance), verify that the VM integration services are current.

Installing and updating integration services is one of the most commonly overlooked steps to ensure overall stability and optimal performance of guest VMs. Although newer Windows-based OSs and some enterprise-class Linux-based OSs come with integration services out of the box, updates may still be required. New versions of integration services may become available as the physical Hyper-V hosts are patched and updated.

With earlier versions of Hyper-V (2012 R2 and prior), during the configuration and deployment of a new VM, the configuration process does not prompt the user to install or update integration services. In addition, the process to install integration services with older versions of Hyper-V (2012 R2 and prior) is a bit obscure and will be explained in this section. With Windows Server 2016 Hyper-V, integration services are updated automatically (in the case of Windows VMs) as a part of Windows updates, requiring less administration to ensure Windows VMs stay current.

One common issue occurs when VMs are migrated from an older physical host or cluster to a newer one (for example, from Windows Server 2008 R2 Hyper-V to Windows Server 2012/R2 Hyper-V). The integration services do not get updated automatically, and degraded performance may be encountered as a result, that may erroneously point the administrator to suspect the storage array as the cause of the problem.

Aside from performance problems, one of the key indications that integration services are outdated or not present on a Windows VM is the presence of unknown devices in Device Manager for the VM.

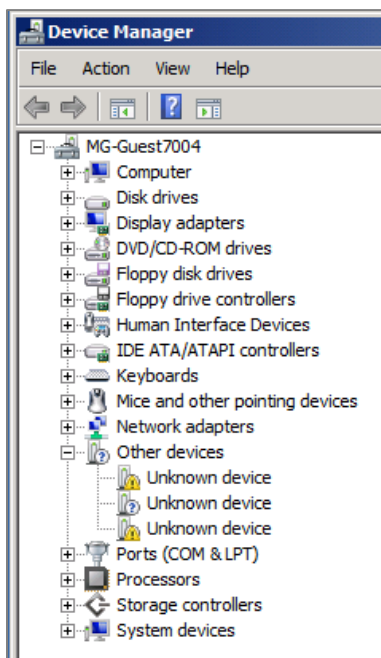


Figure 4 Unknown guest VM devices can indicate missing or outdated integration services

For versions of Hyper-V prior to 2016, use Hyper-V Manager to connect to a VM. Under the **Action** menu, mount the **Integration Services Setup Disk** (an ISO file), and follow the prompts in the guest VM console to complete the installation. Mounting the integration services ISO is no longer supported with Windows Server 2016 Hyper-V because integration services are provided exclusively as part of Windows updates.

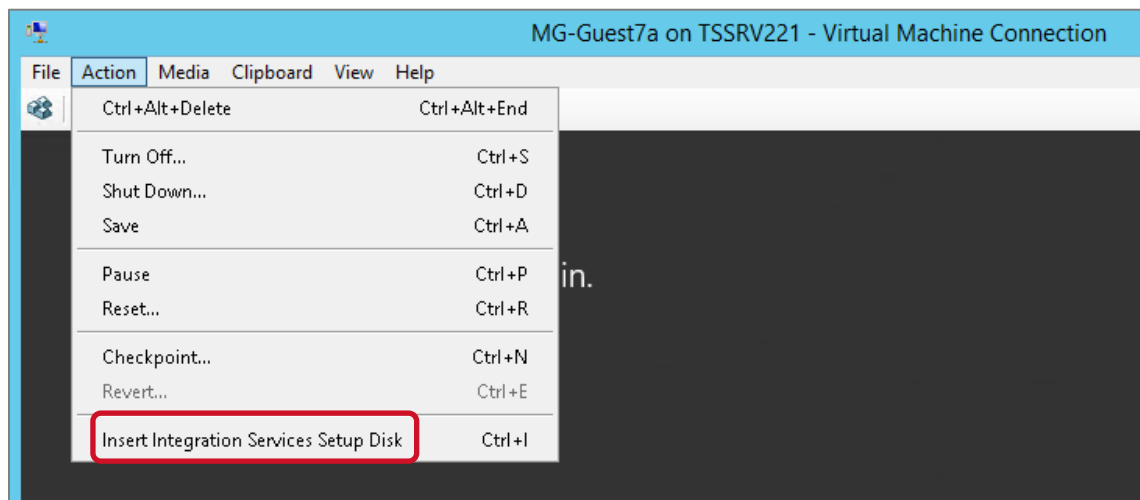


Figure 5 Mount Integration Services Setup Disk in Hyper-V Manager (Hyper-V versions prior to 2016)

To verify the version of integration services, under the **Summary** tab for each VM, select **Failover Cluster Manager**.

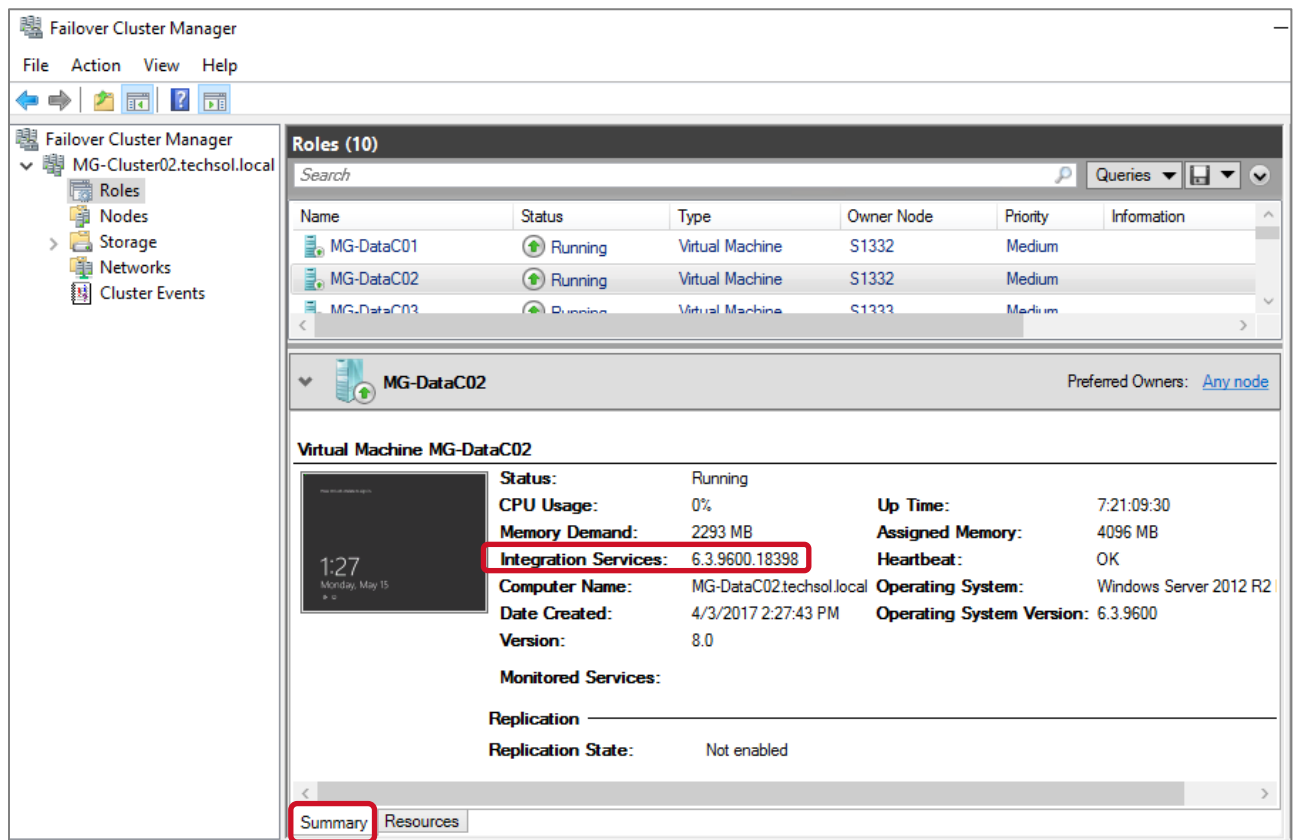


Figure 6 Verify integration services version with Failover Cluster Manager

Verification can also be performed using PowerShell, as shown in the following example:

```
PS C:\Windows\system32> get-VM | Select-Object name, integrationservicesversion
Name      IntegrationServicesVersion
-----
MG-VM12a  6.3.9600.18080
MG-VM12b  6.3.9600.18080
MG-VM12c  6.3.9600.18080
MG-VM12d  6.3.9600.18080
```

3.2 Hyper-V guest VM generations

The generation of a guest VM (generation 1 or generation 2) is an important part of the overall storage strategy because of performance and sizing constraints. When Windows Server 2012 R2 Hyper-V was released, Microsoft designated all existing VMs as generation 1 to differentiate them from a new classification of VMs that could be created as generation 2.

Although generation 1 VMs continue to be supported with Hyper-V, it is a best practice to create new VMs as generation 2 if the host server (Windows Server 2012 R2 Hyper-V and newer) and the guest VM OS support it. Support for generation 1 VMs may eventually be depreciated in future versions of Hyper-V.

Generation 2 guests use Unified Extensible Firmware Interface (UEFI) when booting instead of a legacy BIOS. UEFI provides better security and better interoperability between the OS and the hardware, which offers improved virtual driver support and performance. In addition, one of the most significant changes with generation 2 guests is the elimination of the dependency on virtual IDE for the boot disk. Generation 1 VMs require the boot disk to use a virtual IDE disk controller. Generation 2 guests instead use virtual SCSI controllers for all disks. Virtual IDE is not a supported option with generation 2 VMs.

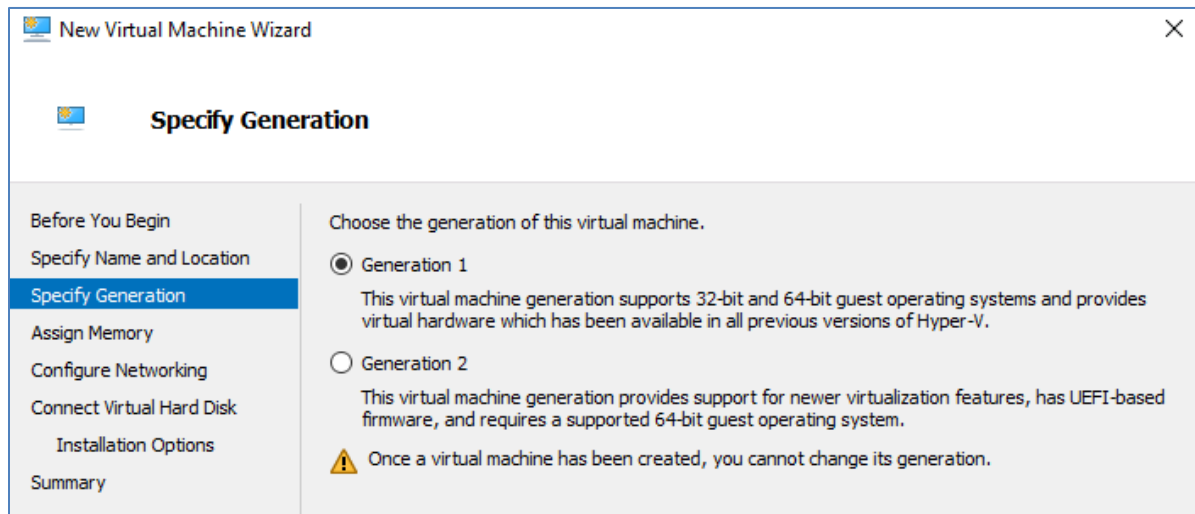


Figure 7 Specify a guest as Generation 1 or Generation 2

For both generations of guest VMs, if there are multiple disks requiring high I/O, each disk can be associated with its own virtual disk controller to further maximize performance.

3.2.1 Convert VMs to a newer generation

Note the warning message in Figure 7 that the VM generation cannot be changed once a VM has been created. However, it is now possible to convert a VM from generation 1 to generation 2. While Microsoft has ongoing efforts to provide tools to perform this action, third-party tools are available (use at your own risk). More information on this topic can be found on [Microsoft TechNet](#).

3.3 Virtual hard disks

A virtual hard disk is a set of data blocks that is stored as a regular Windows file with a .vhd, .vhdx, or .vhds extension, using the host operating system. It is important to understand the different format and type options for virtual hard disks and how this integrates with ME4 Series arrays.

3.3.1 Virtual hard disk format

There are three kinds of virtual hard disk formats that are supported with either VM generation:

- **VHD** is supported with all Hyper-V versions and is limited to a maximum size of 2 TB. This is now considered a legacy format (use VHDX instead for new VM deployments).
- **VHDX** is supported with Windows Server 2012 Hyper-V and newer. The VHDX format offers better resiliency in the event of a power loss, better performance, and supports a maximum size of 64 TB. VHD files can be converted to the VHDX format using tools such as Hyper-V Manager or PowerShell.
- **VHDS (VHD Set)** is supported on Windows Server 2016 Hyper-V and newer. VHDS is for virtual hard disks that are shared by two or more guest VMs in support of highly-available (HA) guest VM clustering configurations.

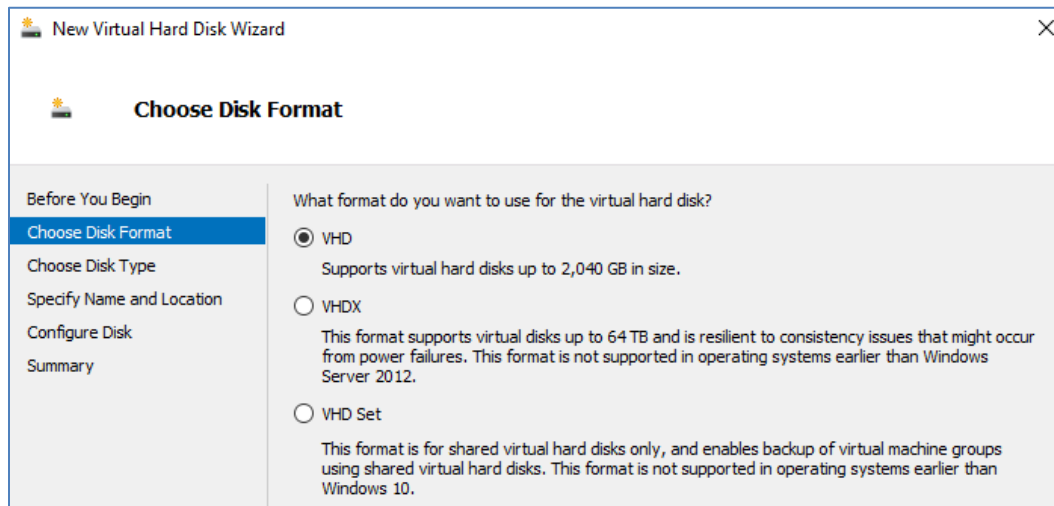


Figure 8 Virtual hard disk format options

3.3.2 Virtual hard disk type

In addition to the format, a virtual hard disk can be designated as fixed, dynamically expanding, or differencing.

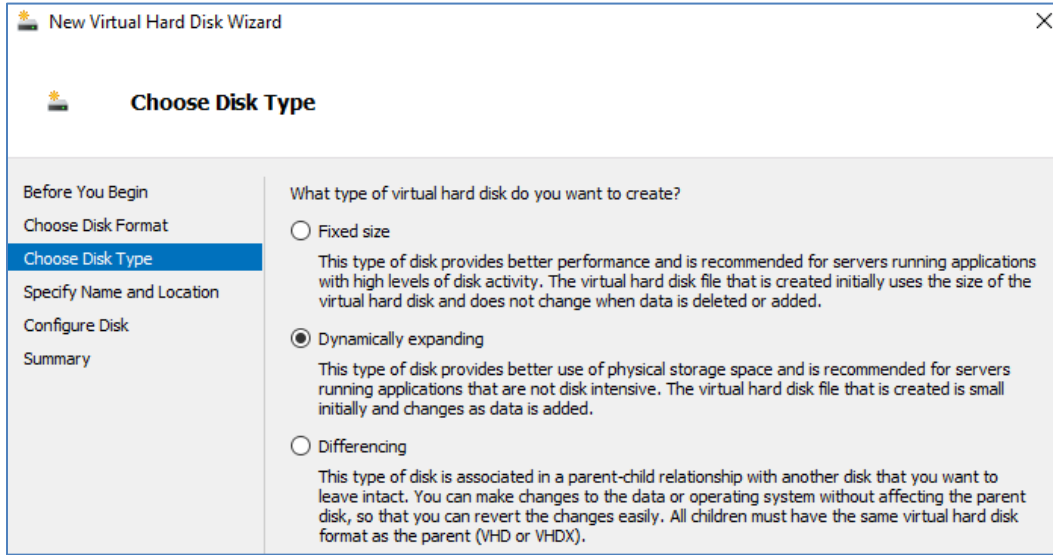


Figure 9 Virtual hard disk type options

The dynamically expanding disk type will work well for most workloads on ME4 Series arrays. If the array is configured to use virtual disk groups and pools which take advantage of thin provisioning, only data that is actually written to a virtual hard disk, regardless of the disk type (fixed, dynamic, or differencing), will consume space on the array. As a result, determining the best disk type is mostly a function of the workload as opposed to how it will impact storage utilization. For workloads generating very high I/O, such as Microsoft SQL Server® databases, Microsoft recommends using the **fixed size** virtual hard disk type for optimal performance.

As shown in Figure 10, a fixed virtual hard disk consumes the full amount of space from the perspective of the host server. For a dynamic virtual hard disk, the space consumed is equal to amount of data on the virtual disk (plus some metadata overhead), and is more space efficient from the perspective of the host. From the perspective of the guest VM, either type of virtual hard disk shown in this example will present a full 60 GB of available space to the guest.

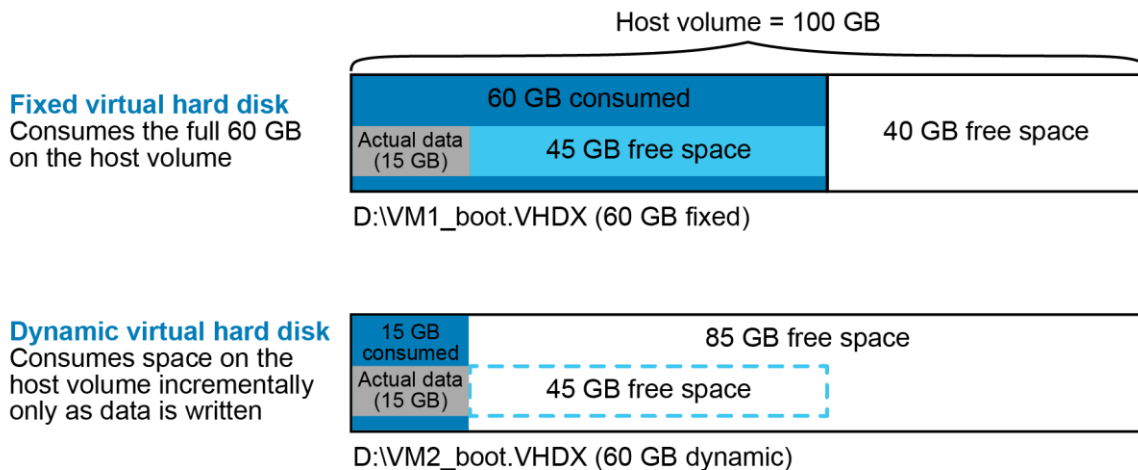


Figure 10 Fixed and dynamic virtual hard disk comparison

There are some performance and management best practices to keep in mind when choosing the right kind of virtual hard disk type for your environment.

- Fixed-size virtual hard disks:
 - Are recommended for virtual hard disks that experience a high level of disk activity, such as Microsoft SQL Server, Microsoft Exchange, or OS page or swap files. For many workloads, the performance difference between fixed and dynamic will be negligible. When formatted, they take up the full amount of space on the host server volume.
 - Are less susceptible to fragmentation at the host level.
 - Take longer to copy (for example, from one host server to another over the network) because the file size is the same as the formatted size.
- Dynamically expanding virtual hard disks:
 - Are recommended for most virtual hard disks, except in cases of workloads with very high disk I/O.
 - Require slightly more CPU and I/O overhead as they grow compared to fixed virtual hard disks. This usually does not impact the workload except in cases where I/O demand is very high. This is minimized on ME4 Series arrays when configuring one or two SSDs to function as read-cache (in a disk pool with spinning drives) or when using all flash drives.
 - Are more susceptible to fragmentation at the host level.
 - Consume very little space (for some metadata) when initially formatted, and expand as new data is written to them by the guest VM.
 - Take less time to copy to other locations than a fixed disk because only the actual data is copied. For example, if a 500 GB dynamically expanding virtual hard disk contains 20 GB of actual data, 20 GB is copied instead of 500 GB.
 - Allow the host server volume to be over-provisioned from the perspective of the host server. In this case, it is an important best practice to configure alerting on the host server to avoid running the volume out of space unintentionally.
- Differencing virtual hard disks:
 - Are used in limited use cases, such as a virtual desktop infrastructure (VDI) deployment.
 - Offer some storage savings by allowing multiple Hyper-V guest VMs with identical operating systems share a common boot virtual hard disk.
 - Require all children to use the same virtual hard disk format as the parent.
 - Are referenced by reads of unchanged data. Unchanged data that is read infrequently may reside in a lower classification of storage on systems that support tiering. When a Dell EMC ME4 array is configured with a disk pool that supports tiering, cold data will typically be allowed to move to a lower tier as a best practice.
 - Require new data to be written to the child virtual hard disk.
 - Are created for each native Hyper-V based snapshot of a Hyper-V guest VM to freeze the changed data since the last snapshot, and allow new data to be written to a new virtual hard disk file. Creating native Hyper-V based snapshots of a Hyper-V guest VM can elevate the CPU usage of storage I/O, but will probably not affect performance noticeably unless the guest VM experiences very high I/O demands.
 - Can result in performance impacts to the Hyper-V guest VM because maintaining a long chain of native Hyper-V based snapshots of the guest VM requires reading from the virtual hard disk and checking for the requested blocks in a chain of many different differencing virtual hard disks.
 - Should not be used with native Hyper-V based snapshots of Hyper-V guests, or should be kept at a minimum to maintain optimal disk I/O performance. With ME4 Series arrays, native Hyper-V snapshots can be minimized or even avoided altogether by leveraging array-based storage snapshots. Administrators can leverage array-based snapshots to recover VMs and replicate data to other locations for archive or recovery.

3.3.3 Virtual hard disks and thin provisioning with ME4 Series arrays

It does not matter which type of virtual hard disk is used to in order maximize the space utilization on ME4 Series storage when leveraging thin provisioning at the array level. Regardless of the virtual hard disk type, only the actual data written by a guest VM will consume space on the storage array due to the advantages of thin provisioning.

The example shown in Figure 11 illustrates an ME4 Series 100 GB volume presented to a Hyper-V host that contains two 60 GB virtual hard disks (overprovisioned in this case to demonstrate behavior, but not as a general best practice). One disk is fixed, and the other is dynamic. Each virtual hard disk contains 15 GB of actual data. From the perspective of the host server, a total of 75 GB of space is consumed and can be described as follows:

Example: 60 GB fixed disk + 15 GB of used space on the dynamic disk = 75 GB total

Note: The host server reports the entire size of a fixed virtual hard disk as consumed.

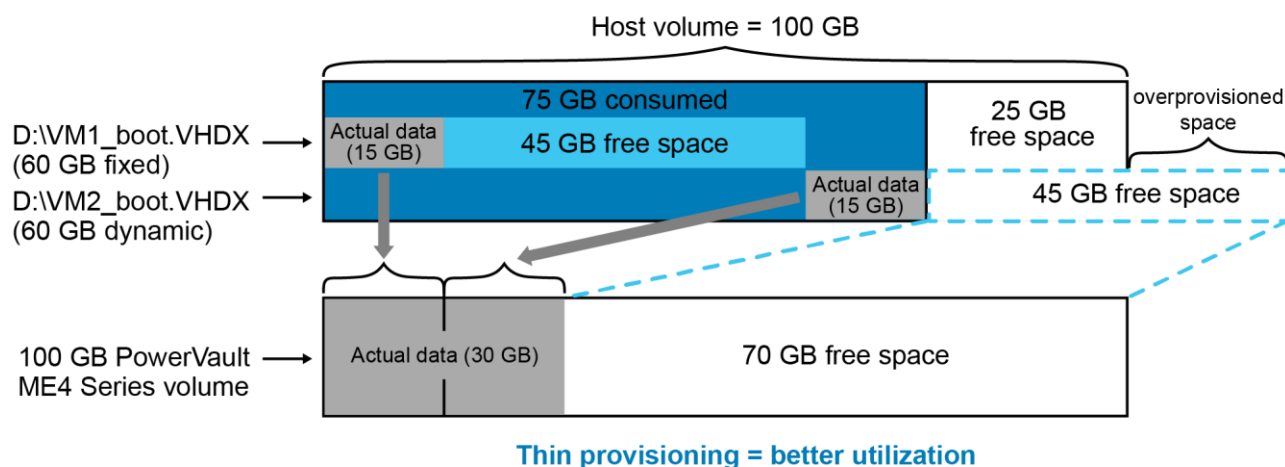


Figure 11 Thin provisioning with ME4 Series array

Comparatively, this is how the ME4 Series array reports storage utilization on this same volume:

Example: 15 GB of used space on the fixed disk + 15 GB of used space on the dynamic disk = 30 GB

Note: Both types of virtual hard disks (dynamic and fixed) will use the same amount of space utilization on Dell EMC ME4 arrays when using thin provisioning. Other factors such as the I/O performance of the workload would be primary considerations when determining the type of virtual hard disk in your environment.

3.3.4 Overprovisioning with dynamic virtual hard disks

With dynamic virtual hard disks and thin provisioning, there is an inherent risk of either the host volume or a storage group or pool on the ME4 Series array running out of space. Figure 11 shows an example of this. If the dynamic disk used by VM2 on the host volume expands far enough, it would fill up the underlying host volume and negatively impact both VM1 and VM2. From the perspective of VM2, it would still see 20 GB of free space, but would not be able to use it because the underlying physical host volume is full. To resolve this, an administrator would move the virtual hard disk for VM1 or VM2 elsewhere to free up space, or expand the 100 GB host volume. In either case, identifying the problem may not be obvious, and resolving the problem might incur a service outage.

To mitigate risks, consider the following recommendations:

- Create a Hyper-V physical host volume that is large enough so that current and future expanding dynamic virtual hard disks will not fill the host volume to capacity. Creating large Hyper-V host volumes will not waste space on ME4 Series arrays that leverage thin provisioning.
 - If Hyper-V based snapshots are used (which create differencing virtual hard disks on the same physical volume), allow adequate overhead on the host volume for the extra space consumed by the differencing virtual hard disks.
 - Expand existing host volumes as needed to avoid the risks associated with overprovisioning.
 - If a physical host volume that hosts virtual hard disks is overprovisioned, set up monitoring so that if a percent-full threshold is exceeded (such as 90 percent), an alert is generated with enough lead time to allow for remediation.
- Monitor alerts on ME4 Series storage so that warnings about disk group and pool capacity thresholds are remediated before they reach capacity.

3.4 Present ME4 Series storage to Hyper-V

There are several ways to present ME4 Series storage to Windows Server Hyper-V hosts, nodes, and VMs. A summary is provided in the following bullet points. For more information about cabling guidance, see the *ME4 Series Administrator's Guide* and *Deployment Guide*.

- ME4 Series storage can be presented to physical Hyper-V hosts and cluster nodes using FC, iSCSI, or SAS in either a direct-attached configuration (SAS, FC, iSCSI) or as part of a SAN (FC or iSCSI).
- ME4 Series storage can also be presented directly to Hyper-V guest VMs using the following:
 - In-guest iSCSI
 - Pass-through disks (this is a legacy configuration option introduced with Hyper-V 2008 that Dell EMC and Microsoft discourage using with Hyper-V 2012 and 2016)

3.4.1 Transport options

ME4 Series storage can be presented to Hyper-V environments using SAS, FC, or iSCSI, and will typically include an MPIO configuration for load balancing and failover protection.

Typically, an environment is configured to use a preferred transport when it is built and will be part of the infrastructure core design. When deploying Hyper-V to existing environments, the existing transport is typically used. Deciding which transport to use is usually based on customer preference and factors such as size of the environment, cost of the hardware, and the required support expertise.

It is not uncommon, especially in larger environments, to have more than one transport available. This might be required to support collocated but diverse platforms with different transport requirements. When this is the case, administrators might be able to choose between different transport options.

Regardless of the transport chosen, it is a best practice to ensure redundant paths to both ME4 Series controller heads A and B. Refer to section 2.3 and the *ME4 Series Deployment Guide* for more information. While the ME4 Series array permits front-end cabling that does not include redundancy, it is a best practice in a production environment to configure cabling for redundancy. For test or development environments that can accommodate down time without business impact, a less-costly, less-resilient design may be acceptable to the business.

3.4.2 Mixed transports

There is limited Microsoft support for using mixed transports to present an ME4 Series LUN to a Windows Server Hyper-V host or cluster node, and this is not recommended as a best practice. In each a Hyper-V cluster environment, all nodes should be configured to use a common transport (FC, iSCSI, or SAS).

There are some use cases where using mixed transports may be necessary, such as when migrating the overall environment from one type of transport to another, and both transports need to be available to a host during a transition period. If mixed transports must be used, use a single transport for each LUN mapped to a Hyper-V host or node.

The following mapping is recommended:

- LUN1 is mapped to host 1 using FC only (with MPIO)
- LUN2 is mapped to host 1 using iSCSI only (with MPIO)

The following mapping is not recommended:

- LUN1 is mapped to host 1 using both FC and iSCSI paths

Each transport by itself should provide adequate bandwidth and MPIO resiliency. Configuring a LUN to use multiple transports increases design complexity unnecessarily and can introduce unpredictable service-affecting I/O behavior in path failure scenarios.

3.4.3 MPIO best practices

Windows Server and Hyper-V natively support MPIO by means of the built-in Device Specific Module (DSM) that is bundled with the OS. This DSM is fully supported with ME4 Series arrays.

Windows and Hyper-V hosts default to the Round Robin with Subset policy with ME4 Series storage, unless a different default MPIO policy is set on the host by the administrator. Round Robin with Subset is typically the best MPIO policy for Hyper-V environments.

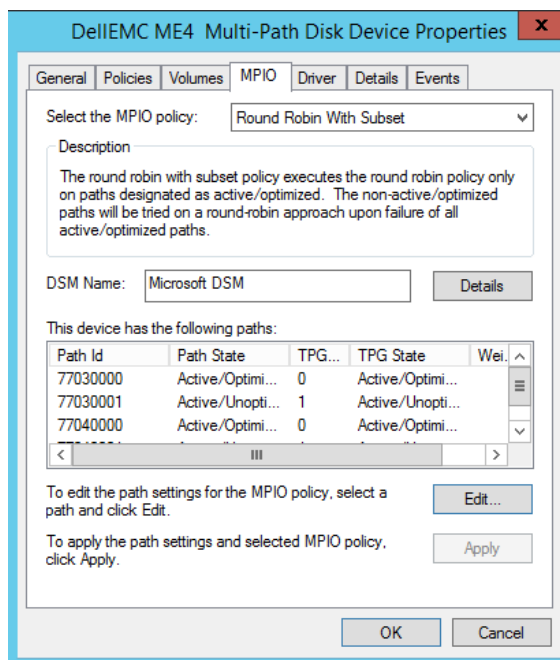


Figure 12 Verify MPIO settings (Microsoft DSM shown)

Note the following:

- The active/optimized paths are associated with the ME4 Series storage controller head that owns the volume. The active/unoptimized paths are associated with the other controller head.
- If each controller has four FE transport paths configured (shown in Figure 12), each volume that is mapped should list eight total paths: four that are optimized, and four that are unoptimized.

Best practices recommendations include the following:

- Changes to MPIO registry settings on the Windows or Hyper-V host (such as time-out values) should not be made unless directed by ME4 Series documentation, or unless directed by Dell EMC support to solve a specific problem.
- Configure all available FE ports on an ME4 Series array (when it is connected to a SAN) to use your preferred transport to optimize throughput and maximize performance.
- If using a direct-connect option for iSCSI, SAS or FC, configure each host to use at least two matching ports (one from each controller head) to provide MPIO and failover protection against a single-path or controller-head failure.
- Verify that current versions of software are installed (such as OS, boot code, firmware, and drivers) for all components in the data path:
 - ME4 Series arrays
 - Data switches
 - HBAs, NICs, converged network adapters (CNAs)
- Verify that all hardware is supported per the latest version of the [Dell EMC hardware Compatibility Matrix](#).

3.4.4 Guest VMs and in-guest iSCSI

ME4 Series storage supports in-guest iSCSI to present block storage volumes directly to guest VMs. The setup and configuration are essentially the same as for a physical host server, except that the VM is using virtual hardware. Follow the guidance in the ME4 Series *Administrator's Guide* to optimize iSCSI settings, such as Jumbo frames.

3.4.5 Direct-attached in-guest iSCSI storage use cases

Although ME4 Series arrays support in-guest iSCSI volumes mapped to guest VMs, direct-attached storage for guest VMs is not recommended as a best practice unless there is a specific use case that requires it.

Typical use cases include:

- Situations where a workload has very high I/O requirements, and the performance gain over using a virtual hard disk is important. Direct-attached disks bypass the host server file system. This reduces host CPU overhead for managing guest VM I/O. For many workloads, there will be no notable difference in performance between direct-attached and virtual hard disks.
- VM clustering on legacy platforms prior to support for shared virtual hard disks, which became available with the 2012 R2 release of Hyper-V, and enhanced with Hyper-V 2016.
- When needing to troubleshoot I/O performance on a volume and it must be isolated from all other servers and workloads.
- When there is a need to create custom snapshot or replication policies or profiles on ME4 Series storage for a specific data volume.
- When a single data volume presented to a guest VM will exceed the maximum size for a VHD (2 TB) or VHDX (64 TB).

There are also disadvantages to using direct-attached storage for guest VMs:

- The ability to perform native Hyper-V snapshots is lost. However, the ability to leverage ME4 Series snapshots of the underlying volume is unaffected.
- Complexity increases, requiring more management overhead to support.
- VM mobility is reduced due to creating a physical hardware layer dependency.

Note: Legacy environments that are using direct-attached disks for guest VM clustering should consider switching to shared virtual hard disks, particularly when migrating to Windows Server 2016 Hyper-V.

3.4.6 Guest VMs and pass-through disks

A block-based pass-through disk is a special type of Hyper-V disk that is mapped to a Hyper-V host or cluster, and then is passed through directly to a Hyper-V guest VM. The Hyper-V host or cluster has visibility to a pass-through disk but does not have I/O access. Hyper-V keeps it in a reserved state because only the guest VM has I/O access.

Using pass-through disks is a legacy design that is discouraged unless there is a specific use case that requires it. They are no longer necessary in most cases because of the feature enhancements with newer releases of Hyper-V (generation 2 guest VMs, VHDX format, and shared VHDs in Windows Server 2016 Hyper-V.) Use cases for pass-through disks are similar to the list provided for direct-attached iSCSI storage in section 3.4.5.

Reasons to avoid using pass-through disks include the following:

- The ability to perform native Hyper-V snapshots is lost, which is similar to direct-attached storage.
- The use of a pass-through disk as a boot volume on a guest VM prevents the use of a differencing disk.
- VM mobility is reduced by creating a dependency on the physical layer.
- This can result in many LUNs presented to hosts or cluster nodes which can become unmanageable and impractical at larger scale.

3.4.7 ME4 Series and Hyper-V server clusters

When mapping shared volumes (quorum disks, cluster disks, or cluster shared volumes) to multiple hosts, make sure that the volume is mapped to all nodes in the cluster using a consistent LUN number. Leverage host groups on the ME4 Series array to simplify the task of mapping a consistent LUN number to multiple hosts.

As a best practice and a time-saving tip, configure the nodes in a cluster so that they are identical with regard to the number of disks and LUNs. In this way, when mapping new storage LUNs, the next available LUN ID will be the same on all hosts. By doing this, having to change LUN IDs later to make them consistent can be avoided.

3.4.8 Volume design considerations for ME4 Series storage

One design consideration that is often unclear is choosing the number of guest VMs to place on an ME4 Series Hyper-V volume, or cluster shared volume (CSV), when Hyper-V hosts are clustered. While many-to-one and one-to-one strategies both have advantages, a many-to-one strategy presents a good design starting point in most scenarios, and can be adjusted for specific uses cases.

Some advantages for a many-to-one strategy include the following:

- Fewer ME4 Series array volumes to create and administer (avoids volume sprawl)
- Quicker VM deployment because creating additional guest VMs does not require creation of a new volume on the ME4 Series array

Some advantages for a one-to-one strategy include the following:

- Easier to isolate and monitor disk I/O patterns for a specific Hyper-V guest VM
- Ability to quickly restore a guest VM by simply recovering the ME4 Series volume from a snapshot
- Gives administrators more granular control over what data gets replicated if ME4 Series volumes are replicated to another location
- Makes it faster to move a guest VM from one host or cluster to another by remapping the volume rather than copying large virtual hard disk files from one volume to another over the network

Other strategies include placing all boot virtual hard disks on a common CSV, and data virtual hard disks on one or more data CSVs.

3.5 Optimize format disk wait time for large volumes

Formatting a large SAN volume mapped to a Windows host (this is not specific to the Hyper-V role) should complete in a few seconds. If long format wait times are experienced for larger volumes (minutes instead of seconds), disable the file system **Delete Notify** attribute on the host by completing the following steps:

1. Access a command prompt on the host server with elevated (administrator) rights.
2. To verify the state of the attribute, type **fsutil behavior query disabledeletenotify** and press **[Enter]**. A result of zero means the attribute is enabled.
3. To disable the attribute, type **fsutil behavior set disabledeletenotify 1** and press **[Enter]**.

The result should display an attribute value of one. To test the result, map a large temporary volume (several TB) from the ME4 Series array to the host and format the volume. It should complete in a few seconds.

3.6 Placement of page files

Windows Servers and VMs typically place the page file on the boot volume by default, and automatically manage page file and memory settings without user intervention. In most cases, these settings should not be changed, unless, for example, an application vendor provides specific guidance on how to tune the page file and memory settings to optimize the performance of a specific workload. Ultimately, each customer will need to decide on the best strategy as they consider variables that are unique to their environment.

With ME4 Series storage, there can be some advantages to placing a page file on a separate volume from the perspective of the storage array. The following reasons may not be sufficiently advantageous by themselves to justify changing the defaults, but in cases where a vendor recommends making changes to optimize a workload, consider the following tips as part of the overall page-file strategy.

- Moving the page file to a separate dedicated volume reduces the amount of data that is changing on the system (boot) volume. This can help reduce the size of ME4 Series snapshots of boot volumes which will conserve space in the disk pool.
- Volumes or virtual hard disks dedicated to page files typically do not require snapshot protection, and therefore do not need to be replicated to a remote site as part of a DR plan. This is especially beneficial in cases where there is limited bandwidth for replication of volumes and snapshots to another ME4 Series array.

3.7 Placement of Active Directory domain controllers

It is a best practice to avoid configuring a Microsoft Active Directory® (AD) domain controller as a Hyper-V guest VM on a Hyper-V cluster when the cluster service requires AD authentication to start.

Consider this scenario: A service outage takes the cluster offline (including the domain controller VM). When attempting to recover, unless there is another domain controller available outside of the affected cluster, the cluster service will not start because it cannot authenticate.

Note: This order dependency can be avoided with Windows Server 2016 Hyper-V because the cluster service uses certificates to authenticate instead of AD. With Windows Server 2016, Hyper-V clusters can also be comprised of nodes that are in workgroups or domains.

Encountering this scenario may be a service-affecting event depending on how long it takes to recover. It may be necessary to manually recover the domain controller VM to a standalone Hyper-V host outside of the cluster, or to another cluster.

This situation can be avoided by doing the following:

- Configure at least one domain controller as a physical server booting from local disk.
- Place virtualized domain controllers on standalone Hyper-V hosts or on individual cluster nodes if there is an AD dependency for cluster services.
- Use Hyper-V Replica (2012 and newer) to ensure that the guest VM can be recovered on another host.
- Place virtualized backup domain controllers on separate clusters, so that a service-affecting event with any one cluster does not result in all domain controllers becoming unavailable. This does not protect against cases where there is a site outage that takes all the clusters (and therefore all the virtualized AD servers) offline.
- Leverage Windows Server 2016 Hyper-V, which does not have an AD dependency to authenticate cluster services.

3.8 Queue depth best practices for Hyper-V

Queue depth is defined as the total number of disk transactions that can be transmitting between an initiator (a port on the host server) and a target (a port on the storage array). The initiator is typically a Windows Server HBA FC port or iSCSI initiator, and the target is an FC or iSCSI port on the SAN array (in this case, the ME4 Series array). Since any given target port can have multiple initiator ports sending it data, the initiator queue depth is generally used to throttle the number of transactions any given initiator can send to a target from a host to keep the target from becoming flooded. When flooding happens, the transactions are queued, which can cause higher latencies and degraded performance for the affected workloads.

3.8.1 When to change queue depth

One issue that is commonly considered is the best practices for queue depth settings for Windows Server hosts and nodes with Hyper-V. On a Windows Server host, queue depth is a function of the Microsoft storport.sys driver and the vendor-specific miniport driver for the FC HBA, iSCSI NIC, or CNA.

In many cases, there is no need to change the default queue depth, unless there is a specific use where changing the queue depth is known to improve performance. For example, if a storage array is connected to a small number of Windows Server Hyper-V cluster nodes hosting a large block sequential read application workload, increasing the queue depth setting may be very beneficial. However, if the storage array has many

hosts all competing for a few target ports, increasing the queue depth on a few hosts might overdrive the target ports and negatively impact the performance of all connected hosts.

While increasing the queue depth can sometimes increase performance significantly for specific workloads, if it is set too high, there is an increased risk of overdriving the target ports on the storage array. Generally, if transactions are being queued and performance is being impacted, and increasing the queue depth results in saturation of the target ports, then increasing the number of initiators and targets (if available) to spread out I/O can be an effective remediation.

3.8.2 Vendor-specific HBA and CNA queue depth settings

It is very important to understand the firmware and miniport driver registry settings for the host server FC HBA, iSCSI NIC, or CNA adapter and how these settings affect queue depth. In the case of QLogic® FC HBAs, for example, the execution throttle setting can be adjusted to control queue depth.

For direction on adjusting firmware or registry settings to modify queue depth, see the documentation for your FC HBA, iSCSI NIC, or CNA.

Note: Changes to FC HBA, iSCSI NIC, or CNA firmware or registry settings that affect queue depth should be evaluated in a test environment prior to implementation on production workloads.

4 ME4 Series snapshots with Hyper-V

ME4 Series snapshots can be used to protect Hyper-V workloads. They are space-efficient snapshots, meaning they consume no additional space unless they are mapped to a host and new data is written. For general use cases and best practices regarding snapshots, see the ME4 Series *Administrator's Guide*.

ME4 Series snapshots allow administrators to do the following in Hyper-V environments:

- Provision an isolated test environment that matches the production environment
- Provision new servers from a snapshot that is designated as a gold image source

ME4 Series snapshots can be taken of volumes mapped as LUNs to a Hyper-V environment regardless of content. This applies to data volumes, cluster shared volumes (CSV), pass-through disks, and in-guest iSCSI volumes. These volumes can also be replicated to another ME4 Series array for DR or other purposes.

4.1 Crash-consistent and application-consistent snapshots

Unless a server is powered off at the time a snapshot is taken, or first put into a consistent state (for example, by leveraging the Microsoft volume shadow copy service (VSS) to gracefully pause a server or application), ME4 Series snapshots are considered crash-consistent. When recovering a server using a crash-consistent snapshot, it is like having the server recover from a power outage at that point in time. In most cases, servers and applications are resilient enough to recover to a crash-consistent state without any complications, whether the cause is an unexpected power outage, or the server is being recovered to a previous point in time to recover from an event such as a malware infection. An exception to this is when the Hyper-V environment hosts a transactional workload such as Microsoft Exchange or SQL Server. With transactional workloads, the risk of data loss or corruption is higher when attempting to recover to a crash-consistent state.

Some examples for how to configure and use Dell EMC ME4 snapshots for a Hyper-V environment are provided in the following sections.

4.2 Guest VM recovery with ME4 Series snapshots

Hyper-V guest VMs can be recovered to a previous point in time by using crash-consistent snapshots. Snapshots can be used to create copies of VMs in an isolated environment at the same or a different location when replication between ME4 Series arrays is used. This section provides guidance and best practices for several different recovery options using snapshots.

4.2.1 Recover a guest VM on a standalone Hyper-V host

In this scenario, the virtual hard disk and configuration files for a VM reside on a data volume that is mapped to a Hyper-V host.

Option 1: Recover the existing data volume on the host that contains the VM configuration and virtual hard disks by using an ME4 Series snapshot rollback.

- This may only be practical if the data volume contains only one VM. If the data volume contains multiple VMs, it will still work if all the VMs are being recovered to the same point in time. Otherwise, option 2 or 3 would be necessary if needing to recover just one VM.
- This will allow the VM being recovered to power up without any additional configuration or recovery steps required.
- It is essential to document the LUN number, disk letter, or mount-point information for the volume to be recovered, before starting the recovery.

Option 2: Map a snapshot containing the VM configuration and virtual hard disks to the host as a new volume, in a side-by-side fashion using a new drive letter or mount point. The VM can be recovered by manually copying the virtual hard disks from the recovery snapshot to the original location.

- This involves deleting, moving, or renaming the original virtual hard disks.
- After copying the recovered virtual hard disks to their original location, they must be renamed and Hyper-V manager must be used to re-associate them with the guest VM. This is necessary to allow the guest VM to start without permissions errors.
- This may not be practical if the virtual hard disks are extremely large. In this case, the original VM can be deleted, and the recovery VM imported or created as a new VM directly from the recovery volume. After the recovery, the original data volume can be unmapped from the host if no longer needed.
- This method also facilitates recovery of a subset of data from a VM by mounting a recovery virtual hard disk as a volume on the host server temporarily.

Option 3: Map the recovery snapshot to a different Hyper-V host and recover the VM there by importing the VM configuration or creating a new VM that points to the virtual hard disks on the recovery volume.

- This is common in situations where the original VM and the recovery VM both need to be online at the same time, but be isolated from each other to avoid name or IP conflicts, or avoid a split-brain situation with data reads/writes.
- This is a good recovery method when the original host server is no longer available due to a host failure.

If possible, before beginning any VM recovery, record essential details about the VM hardware configuration (such as number of virtual CPUs, RAM, virtual networks, and IP addresses) if importing a VM configuration fails.

4.2.2 Recover guest VM on a cluster shared volume

The process of using ME4 Series snapshots to recover guest VMs that reside on a CSV is like the process of recovering a guest VM to a standalone host, as detailed in section 4.2.1. However, recovering a VM from a snapshot of a CSV may require changing the disk signature first.

Windows Servers assign each volume a unique disk ID (or signature). For example, the disk ID for an MBR disk is an 8-character hexadecimal number such as 045C3E2F4. No two volumes mapped to a server can have the same disk ID.

When an ME4 Series snapshot is taken of a Windows or Hyper-V volume, the snapshot is an exact point-in-time copy, which includes the Windows disk ID. Therefore, recovery volumes based on snapshots will also have the same disk ID.

With standalone Windows or Hyper-V servers, disk ID conflicts are avoided because standalone servers can automatically detect duplicate disk IDs and change them dynamically on the offending disk with no user intervention. However, host servers are not able to dynamically change conflicting disk IDs when disks are configured as CSVs, because the disks are mapped to multiple nodes at the same time.

When attempting to map a copy (snapshot) of a CSV back to any server in that same cluster, the recovery volume will cause a disk ID conflict, which may be service-affecting.

There are two methods to work around the duplicate disk ID issue:

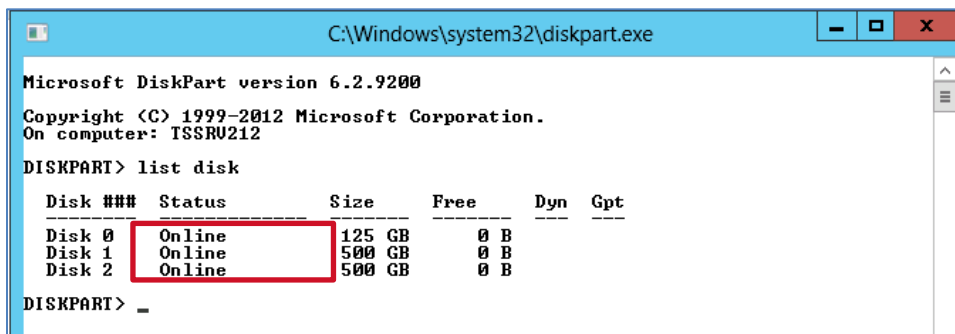
Option 1: Map the recovery volume (snapshot) containing the CSV to another host that is outside of the cluster and copy the guest VM files over the network to recover the guest VM.

Option 2: Map the recovery volume to another Windows host outside of the cluster and use **Diskpart.exe** or PowerShell to change the disk ID. Once the ID has been changed, remap the recovery volume to the cluster. The steps to use Diskpart.exe to change the disk ID are detailed in section 4.2.3.

4.2.3 Change a CSV disk ID with Diskpart

Follow these steps to change a volume disk ID. PowerShell can also be used.

1. Access the standalone Windows host that the recovery volume (snapshot) containing the CSV will be mapped to.
2. Open a command window with administrator rights.
3. Type **diskpart.exe** and press **[Enter]**.
4. Type **list disk** and press **[Enter]**.
5. Make note of the current list of disks (in this example, Disk 0, Disk 1, and Disk 2).



```

Microsoft DiskPart version 6.2.9200
Copyright (C) 1999-2012 Microsoft Corporation.
On computer: TSSRU212

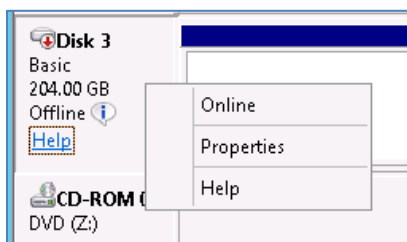
DISKPART> list disk

   Disk ###    Status         Size      Free      Dyn  Gpt
   -----    -
   Disk 0      Online         125 GB     0 B
   Disk 1      Online         500 GB     0 B
   Disk 2      Online         500 GB     0 B

DISKPART> _

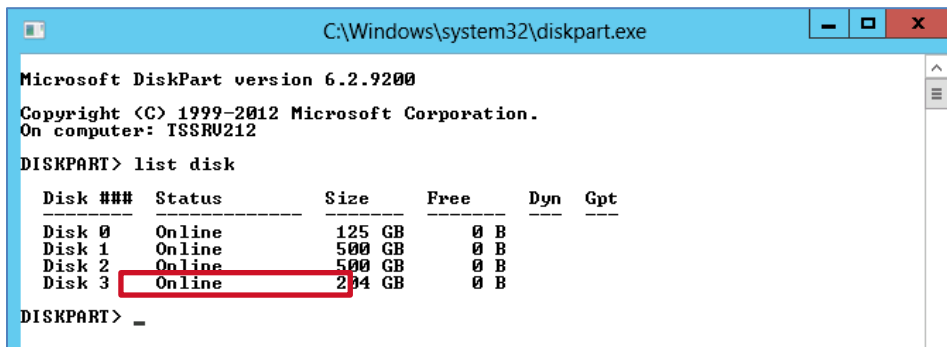
```

6. Map the recovery volume containing the CSV to this host.
7. From the **Diskpart** command prompt, type **rescan** and press **[Enter]**.
8. Use **Disk Management** on the host server to bring the recovery volume online.



9. Return to the **Diskpart** command prompt window, type **list disk**, and press **[Enter]**.

The new disk (Disk 3 in this example) should now be listed. Usually, the bottom disk will be the one most recently added.



```

C:\Windows\system32\diskpart.exe
Microsoft DiskPart version 6.2.9200
Copyright (C) 1999-2012 Microsoft Corporation.
On computer: TSSRU212
DISKPART> list disk

   Disk ###  Status         Size           Free           Dyn  Gpt
   -----  -
   Disk 0    Online         125 GB          0 B
   Disk 1    Online         500 GB          0 B
   Disk 2    Online         500 GB          0 B
   Disk 3    Online         204 GB          0 B

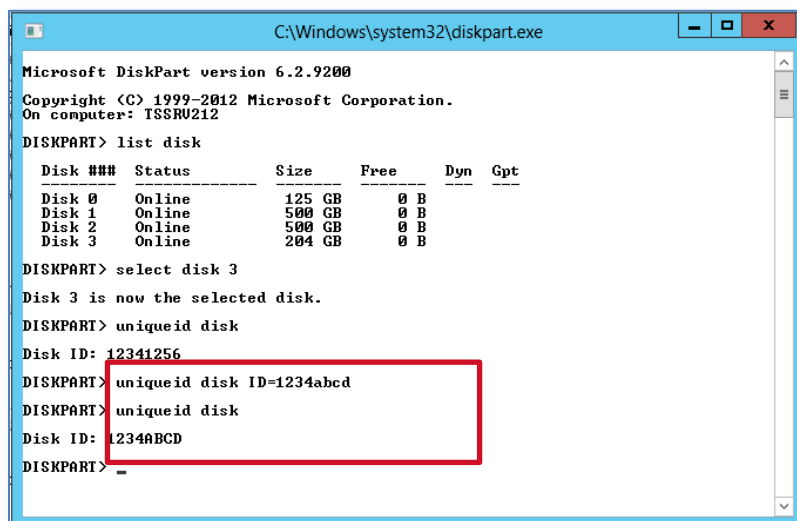
DISKPART> _

```

10. Type **select disk #** (# represents the number of the new disk, Disk 3 in this example) and press **[Enter]**.
11. Type **uniqueid disk** and press **[Enter]** to view the current ID for the disk.
12. To change the disk ID, type **uniqueid disk ID=<newid>** and press **[Enter]**.
 - For **<newid>**, provide a random ID of your choice. For an MBR disk, the new ID must be an eight-character string in hexadecimal format using a mix of the numbers 0–9 and the letters A–F.
 - For a GPT disk, the new ID must be a Globally Unique Identifier (GUID).
13. Type **uniqueid disk** again and press **[Enter]** to verify the ID is now changed.

Now that the disk has a new signature, it can be unmapped from the standalone host server and re-mapped to the cluster without causing a disk ID conflict.

14. Recover the guest VM.



```

C:\Windows\system32\diskpart.exe
Microsoft DiskPart version 6.2.9200
Copyright (C) 1999-2012 Microsoft Corporation.
On computer: TSSRU212
DISKPART> list disk

   Disk ###  Status         Size           Free           Dyn  Gpt
   -----  -
   Disk 0    Online         125 GB          0 B
   Disk 1    Online         500 GB          0 B
   Disk 2    Online         500 GB          0 B
   Disk 3    Online         204 GB          0 B

DISKPART> select disk 3
Disk 3 is now the selected disk.
DISKPART> uniqueid disk
Disk ID: 12341256
DISKPART> uniqueid disk ID=1234abcd
DISKPART> uniqueid disk
Disk ID: 1234ABCD
DISKPART> _

```

4.3 Create test environment with ME4 Series snapshots

In addition to VM recovery, ME4 Series snapshots can be used to quickly create test or development environments that mirror a production environment. When volumes containing VMs are replicated to another location, this makes it very easy to do this at a different location.

Note: To avoid IP, MAC address, or server name conflicts, copies of existing VMs that are brought online should be isolated from the original VMs.

The procedure to use a snapshot to create a test environment from an existing Hyper-V guest VM is very similar to VM recovery. The main difference is that the original VM continues operation, and the VM copy is configured so that it is isolated from the original VM.

4.4 Migrate guest VMs with ME4 Series storage

Microsoft provides native tools to move or migrate VMs with Windows Server 2012 and 2016 Hyper-V, so there are fewer use cases for using SAN-based snapshots to move VMs. When a guest VM is live migrated from one node to another node within the same Hyper-V cluster configuration, no data needs to be copied or moved because all nodes in that cluster have shared access to the underlying cluster shared volumes (CSV).

However, when an administrator needs to migrate a guest VM from one host or cluster to another host or cluster, the data (the virtual hard disks) must be copied to the target host or cluster, and this will consume network bandwidth and may require significant time if the virtual hard disks are extremely large. This can also consume additional storage space unnecessarily because another copy of the data is created.

When moving VMs to another host or cluster, it may be much quicker to leverage ME4 Series storage to unmap the host volume containing the VM configuration and virtual hard disks and map the volume to the new target host or cluster. This can also be done using a using a snapshot.

While this might involve a small amount of downtime for the VM being moved during a maintenance window, it might be a more practical approach than waiting for a large amount of VM data to copy over the network, consuming additional array space unnecessarily.

A Technical support and additional resources

[Dell.com/support](https://www.dell.com/support) is focused on meeting customer needs with proven services and support.

[Storage solutions technical documents](#) provide expertise that helps to ensure customer success on Dell EMC storage platforms.

A.1 Related resources

The following ME4 Series publications and additional resources are available at [Dell.com/support](https://www.dell.com/support).

- Administrator's Guide
- Deployment Guide
- CLI Guide
- Owner's Manual
- Support Matrix

Additionally, see the following third-party referenced or recommended publications and articles:

- [Microsoft TechNet Library](#)
- [Microsoft PowerShell Developer Network \(MSDN\)](#)