Clustered File Systems with DE Series Products

Best Practices for Media and Entertainment Customers

Abstract

DE Series storage arrays deliver high-bandwidth performance and extreme reliability for file systems that support multiple host types with access to the same back-end LUNs. This document provides a reference architecture, including complex multipath details that must be considered when setting up a Cluster file system using multiple host types and running a media application.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 Introduction

Lenovo DE Series storage arrays are often deployed in cluster environments that require high-bandwidth and low-latency performance. However, volume ownership thrashing and other multipath/failover issues have been longstanding challenges in clustered environments, especially when hosts running different OSs access the same storage volumes on the array.

Over the past few years, Lenovo has released enhancements in the DE Series firmware that improve interoperability between hosts and DE Series arrays in heterogeneous host OS clustered environments. These changes have culminated in a new failover mode that greatly improves cluster interoperability, and these most recent enhancements are the next step in an ongoing evolution. This includes two enhancements included in Lenovo SAN OS 08.50 that are designed to specifically resist cluster volume ownership thrash.

Options have existed in the past that allow multipath failover to be avoided in deployments where that option made sense, but now these new options exist to build cluster solutions without risk of ownership thrash while still retaining failover and host I/O access redundancy. For some limited configurations, options also exist to provide limited failback in clustered environments. This enables automatic recovery software to move LUN ownership back to preferred paths following the correction of SAN connectivity faults, again without the risk of LUN ownership thrash.

This paper includes an example of a DE Series array deployed using a thrash-resistant, redundant cluster.

2 DE Series Redundant Host Access/Failover: Background Information

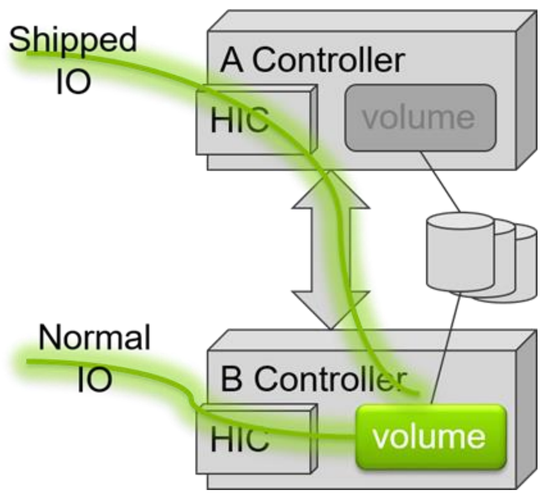
The following information is provided to level-set terminology and to aid in understanding the configuration settings and best practices outlined later in this document. The section presents a significant amount of detailed information, but taking the time to understand the concepts and associated implications of specific multipath driver behavior is useful when following the cluster configuration best practices outlined later in this document. This background can be even more valuable when trying to troubleshoot potential causes of volume ownership thrashing or other multipath issues in an existing clustered environment.

2.1 Volume Ownership Model and ALUA

To achieve consistent, ultralow latency, the DE Series firmware implements a highly optimized write data cache coherency model and a lean I/O path that uses the notion of volume ownership. This means that I/O for a given volume must be processed by the controller that owns that specific volume. This design greatly simplifies the data-caching implementation and hardware design, minimizing I/O latency while keeping the overall product cost low.

Despite using a design that employs an ownership model, DE Series arrays also have multiple target ports on each I/O controller, either on a host interface card (HIC) or on the controller baseboard itself. Both HIC and baseboard ports support an active-active access model. With this model, hosts can access all volumes through any target port on either controller, regardless of volume ownership. This access is accomplished with an “I/O shipping” mechanism, in which I/O received on the nonowning controller is processed by the owning controller using intercontroller messaging. For example, Figure 1 shows the case where a volume is owned by the B controller. I/O directed by the host to the A controller is first shipped through the controller interconnect channels to the B controller, then processed.

Figure 1) ALUA support implementation and I/O shipping example.



This implies that I/O sent to the nonowning controller can result in degraded performance, and so the performance observed by a host is not equal across all the target ports on the array. This is referred to as Asymmetric Logical Unit Access (ALUA), as opposed to a fully symmetric active-active design in which performance is equal across all target ports.

2.2 TPGS Reporting and Legacy Redundant Dual Active Controllers

DE Series arrays also support RDAC (redundant dual active controller, Linux Kernel 3.9 or earlier) T10-SPC (SCSI) standard methods for reporting groups of controller target ports and their associated performance characteristics or asymmetric access states. This method, typically referred to as target port group support (TPGS), allows reporting of the specific target port group or groups that offer optimal performance to the host.

With TPGS, the host multipath driver can obtain information about the groups of target ports on each array controller using the `REPORT TARGET PORT GROUPS` command. DE Series arrays report two groups of target ports, one group for the ports on each controller.

The scope of the `REPORT TARGET PORT GROUPS` command is volume-specific, meaning each volume can report a different asymmetric access state for each of the two groups of target ports. The target port group on the controller that owns a volume is reported as having an asymmetric access state of active/optimized (AO) because that controller can handle I/O requests with maximum/optimal performance. The target port group for the ports on the controller that does not own the volume is reported as having an asymmetric access state of active/nonoptimized (ANO) because directing I/O requests at the nonowning controller implies a performance penalty. Host multipath drivers use this information to route I/O to the target ports that result in optimal performance (that is, I/O to a given volume is routed to the target ports on the controller that owns that volume). Table 1 summarizes the DE Series volume ownership model. Alternating volume ownership continues as additional volumes are provisioned.

Table 1) ALUA model for volume ownership and access.

Volume Name	Volume Owner	Target Port Group Asymmetric Access States Ports on Controller A	Ports on Controller B
Volume 1	Controller A	Active/optimized	Active/nonoptimized
Volume 2	Controller B	Active/nonoptimized	Active/optimized
Volume 3	Controller A	Active/optimized	Active/nonoptimized
Volume 4	Controller B	Active/nonoptimized	Active/optimized

Note: Alternating volume ownership is automatically enabled when using the SAN OS GUI to create volumes. When using command line scripts, volume ownership is specified in the script and does not automatically alternate controller ownership.

Support for TPGS reporting was included in and after the 08.50 firmware release to support for ALUA. Host multipath drivers that utilize these vendor-unique methods are often referred to as legacy redundant dual active controller (RDAC) drivers.

2.3 Preferred and Current Volume Ownership: Failover and Failback

The controller that owns a given volume for data-caching purposes as described earlier is typically referred to as the current owner of the volume. Each volume is also assigned a preferred owner during the provisioning process that is distinct from the current owner. As noted earlier, the preferred owner is assigned by the user through the CLI and is automatically assigned using the GUI at volume creation time. Ownership can be changed by the user after a volume is initially created.

In optimal conditions, the current and preferred owner for a given volume are the same. However, in some conditions where one or more hosts lose access to the preferred owning controller, a failover might occur, resulting in the current and preferred owner being different. After the SAN connectivity fault that resulted in the failover is corrected, the failback process moves current volume ownership back to the preferred controller (more about this later).

`Preferred owner` is reported to the host multipath driver in the data returned from the array in response to the `REPORT TARGET PORT GROUPS` command. Essentially, one of the two target port groups is marked as the preferred (`PREF=1`) group, as defined in the T10-SPC standard, based on which controller is configured as the preferred owner of the volume. This allows the host multipath driver to monitor both current owner (based on the target port group asymmetric access state) and the preferred owner of a given volume.

2.4 Explicit Versus Implicit ALUA/TPGS and Failover

T10-SPC defines two methods of ALUA support: implicit and explicit. Implicit ALUA means the array controller manages the asymmetric access states of the target port groups (that is, the array manages the volume ownership, which in turn determines the asymmetric access state of each target port group). This is accomplished by monitoring all incoming read/write requests to a given volume on the array and determining which controller is receiving the bulk of the incoming requests. In the absence of any SAN connectivity fault, the preferred owner and current owner of a given volume are the same controller. It tends to stay that way because the host multipath driver directs I/O at the current owning controller (active/optimized target port group) in order to maximize performance. When a fault occurs such that hosts with access to the volume lose connection to the preferred/current controller, I/O is directed at the nonowning/nonpreferred controller, and the array decides to move current ownership to the nonpreferred controller to minimize the I/O shipping penalty. By definition, this means more than 75% of the I/O to a given volume over a 5-minute period arrived at the nonowning controller. The result is an implicit failover, and the current and preferred owner are now different.

Note: When an implicit failover occurs, a special notice is sent to all hosts with access to that volume using a `UNIT ATTENTION` check condition, which notifies the host multipath driver that the active/optimized target port group has now changed. A well-behaving host multipath driver simply follows this change and starts directing I/O at the new owning controller (new active/optimized target port group).

Explicit ALUA means that the host multipath driver manages the volume ownership using the `SET TARGET PORT GROUPS` command defined by T10-SPC, which effectively allows the host to change volume ownership by requesting a change to the target port group asymmetric access states. For example, a host could request a change to the controller A target port group asymmetric access state for volume 1 from active/optimized to active/nonoptimized (and vice versa for the controller B target port group), which would tell the array that the volume ownership on volume 1 would have to move from controller A to controller B. An explicit failover would occur when one host loses access to the current owning controller for a given volume. To maintain performance, the host makes an explicit request to the array to change ownership so that the remaining paths retain full performance.

Support for implicit or explicit ALUA (or both) is reported by the array to the host multipath driver using a field in the standard inquiry data reported by the array. If both explicit and implicit modes are reported to the host multipath driver, it is up to the host driver to select which mode is used. Further details about how implicit versus explicit ALUA support is reported to a given host are covered later in this document.

2.5 Host Versus Implicit/Target (Array-Initiated) Failback

Like the failover process, the failback process is when a volume's current ownership is returned to the preferred controller. This can be initiated either by the host multipath driver (explicit failback) or by the array itself (implicit failback, sometimes referred to as target failback).

With explicit failback, the host issues a `SET TARGET PORT GROUP` request (or in the case of a legacy driver, a vendor-unique command), which results in the current volume ownership being returned to the preferred controller. This typically occurs when the host sees that at least one connection to the preferred controller has been reestablished.

Beginning in 08.50.XX.XX firmware, DE Series added support for implicit failback where the array monitors the connectivity to each controller for all hosts that have access to a given volume. The array initiates an ownership change on that volume to move it back to the preferred controller after all hosts have access again using the preferred controller.

2.6 Mixed-Mode Multipath Drivers

Some multipath drivers rely on the array to manage failover (that is, use implicit failover) but make explicit requests to fail back when connectivity returns to the preferred controller (that is, explicit failback). Although this mixed mode seems odd, it was a very common behavior in many mainstream drivers, including our own Windows MPIO DSM (Lenovo DE Series -specific DSM v1.X) and the device handler specific to Lenovo DE

Series for Linux DM-MP (scsi_dh_rdac). This behavior in the drivers specific to Lenovo was essentially an interim phase in the migration from the historically full explicit mode (both explicit failover and failback) to the full implicit mode (both implicit failover and failback) deployed beginning with 08.50.XX.XX firmware and associated multipath drivers.

2.7 Explicit Versus Implicit Failover and Failback: Cluster Implications

In a cluster environment where multiple hosts have access to the same volume, explicit ALUA carries significant risk of volume ownership thrash because individual hosts in the cluster might not agree with each other about the appropriate ownership of a given volume, especially if one host loses access to the preferred controller while one or more of the other hosts in the cluster retain access to the preferred controller. The host that must access the volume using the nonpreferred controller changes ownership to retain full performance, but the other hosts in the cluster see that the volume is off the preferred controller even though they still have connectivity to that controller and initiate a failback operation to change ownership back to the preferred controller. This ownership fight can continue until the connectivity problem is resolved. During this time window, volume ownership can oscillate between controllers very rapidly if the host multipath drivers that conflict are aggressive about initiating explicit failover or failback requests. This thrashing can even happen with mixed-mode drivers using implicit ALUA because one host might direct I/O at the nonowning (but preferred) controller even if they don't issue an explicit ownership transfer request. In this case, the ownership thrash typically is not as rapid. The failover takes at least 5 minutes of heavy I/O to the nonowning controller before the array initiates an implicit failover, followed relatively quickly by another host requesting an explicit failback.

Note: This issue was mitigated by having administrators disable failback in the multipath configuration settings on the host to stop at least one side of the ownership fight. Even if a single host in the cluster was misconfigured and did not have failback disabled, ownership thrash could still result in the entire cluster.

In general, any use of explicit failover or failback is toxic in a clustered environment with shared volume access. This means that any driver using explicit failover must be completely avoided because failover cannot be disabled without losing the ability to have redundant access. Drivers using explicit failback in clustered environments are risky but can be used if failback is disabled in the host driver. This means that resolving ownership issues must be manually triggered by the user after correction of SAN connectivity faults.

Use of implicit ALUA for both failover and failback is highly recommended in a clustered environment with shared volume access. The array has visibility to the current connectivity of all hosts in the cluster from both array controllers as well as visibility to incoming I/O patterns and can therefore select the optimal volume ownership. Essentially, the array has the right vantage point to arbitrate shared volume access fairly and determine when ownership transfers are appropriate. With use of implicit ALUA, it is possible to build a cluster with I/O path redundancy and resiliency without risk of ownership thrash.

Beginning in 08.50.XX.XX firmware, full implicit ALUA (both failover and failback) support is available with Windows MPIO, Linux DM-MP, and VMware NMP/SATP_ALUA multipath drivers. This is accomplished by the array reporting only implicit ALUA support to the host multipath driver in the standard inquiry data to hosts running those multipath solutions, essentially denying the host the ability to use explicit ALUA. The array also hides preferred volume ownership information in `REPORT TARGET PORT GROUPS` responses to these specific OS/drivers to make sure the host does not attempt a failback of any kind. Reporting preferred target port group information is optional in the SCSI standard, so it can be withheld by the array. This essentially builds a cluster-safe multipath solution with full redundancy (failover) and autorecovery (failback).

Having full implicit failover and failback support in the array means the host-side multipath configuration is not as critical in a cluster as long as the host multipath driver adheres to the implicit versus explicit ALUA support reported by the array. This essentially puts the array in charge of all volume ownership and is the direction DE Series engineering is trying to move with all host multipath solutions, given the ever-increasing prevalence of scale-out/clustered solutions.

3 DE Series Storage Partitioning Configuration

During initial device discovery, when the multipath solutions installed on a host are discovering storage devices on the SAN, the standard inquiry data reported by the array controllers and responses to commands such as `REPORT TARGET PORT GROUPS` are key in whether the multipath solution even recognizes the storage, and if it does recognize the storage, what multipath mode (implicit versus explicit ALUA) it selects to use. Proper storage driver operation is critical in making sure of successful interoperability between the array and host.

Note: In some cases, multiple multipath drivers might be installed on a host (that is, in-box OS multipath driver plus vendor-unique drivers installed, and so on). The initial device discovery phase is very critical in making sure the correct multipath solution claims the storage devices associated with DE Series volumes. This primarily depends on the specific inquiry data returned from the array.

Because DE Series supports many different OS/multipath combinations and each of those combinations best operates in different modes requiring different inquiry responses, the host type configuration on the array is critical to make sure of proper responses and successful host/array interoperability.

3.1 Recognizing Host That Initiated a Command (and Associated Host Type)

The DE Series array controller firmware uses the SAN OS storage partitioning configuration information set up by the user (hosts, host ports, host groups) to identify the specific host that sent a command to the array. At the host interface protocol level, the originating worldwide unique port identifier is known for each incoming command, but that port identifier must then be associated with a given host to determine the host type and associated OS/multipath solution. This in turn is used to control some of the host type-specific behavior that must be employed by the array when processing the incoming request. This includes managing access to the specific LUNs/volumes that should be accessible over that host port. Therefore, the mapping of host ports to hosts and the operating system type selected on those hosts in the array configuration are essential configuration parameters that must be correct to make sure of proper responses on incoming device discovery commands received over a given host port.

These configuration parameters are of the utmost importance in a heterogeneous host-type, clustered environment where a single host with a multipath driver operating in explicit ALUA mode could easily trigger volume ownership thrashing in the entire cluster.

Another side effect of an improper setup can be that the installed host multipath drivers cannot claim DE Series volumes during device discovery because the array sends incorrect/unexpected responses to discovery-related commands. This condition is often due to an unknown or misunderstood host operating system type. It can also lead to a condition where host applications have access to the same volume multiple times through multiple device nodes in the OS device stack. By default, the DE Series array exposes each volume in the system on all controller target ports, and the customer's SAN configuration likely exposes each host HBA port to at least one controller target port on each controller in the array. It is the multipath driver that coalesces the multiple paths to the same storage device/volume.

3.2 Common Host Types and Associated Failover Settings

Table 2 outlines a few of the common host types configurable in DE Series arrays and highlights some of their associated multipath-related parameters. The expected multipath solution should claim the storage devices for the DE Series volumes if the associated driver is installed and properly configured on the host.

Note: Any host type listed in this table that uses explicit failover in the multipath solution is essentially not cluster safe.

Note: Any host type that uses explicit failback is not cluster safe without some mechanism of disabling the failback functionality.

The definition or behavior controlled by these specific host types is encoded in the DE Series NVSRAM build and the controller firmware code. Therefore, it is critical that the correct NVSRAM is installed for a given release of DE Series controller firmware. The following table are based on Lenovo product software builds. LenovoLenovo

Table 2 indicates multipath functionality with SAN OS 08.50.XX.XX and later firmware builds. This later DE Series software added the automatic load balancing feature, so the table also indicates if the new feature is supported for the given host type.

Table 2) Common host types: multipath settings (08.50.XX.XX and later firmware/NVSRAM builds).

Index	Host Type	TPGS Support	ALUA Support	Failover Mode	Failback (Mode + Host or Target Driven)	Automatic Load Balancing (ALB) Support	Expected Multipath Solution
1	Windows	Y	Y	Implicit	Implicit/target	Yes	Windows MPIO with Lenovo DSM 2.0
7	Linux DM-MP (kernel 3.9 or earlier)	N	Y	Implicit	Explicit/host	No	Device mapper multipath (DM-MP) with scsi_dh_rdac device handler
8	Windows clustered	Y	Y	Implicit	Implicit/target	Yes	Windows MPIO with Lenovo DSM 2.0
10	VMware	Y	Y	Implicit	Implicit/target	Yes	VMware NMP/SATP_ALUA
28 (Default)	Linux DM-MP (kernel 3.10 or later)	Y	Y	Implicit	Implicit/target	Yes	Device mapper multipath (DM-MP) with scsi_dh_alua device handler

3.3 Default Host Type

Default Host Operating System Type

The array configuration does have the notion of a default host operating system, which is used to determine proper inquiry responses for such unconfigured hosts. But given that multiple OS/driver combinations are in use in a heterogeneous host-type cluster, applying a single host type to all hosts implies that for at least some of the hosts that default host type is incorrect. It is possible, however, to use the default host group without device discovery or multipath issues if all hosts within the group/cluster are of the same operating system type using the same multipath solution and the default host operating system is set accordingly. It is also possible to use the default host group without multipath issues if the single–target port LUN-mapping feature is being used.

Factory Default Host Type

The factory default host type (host type 0) is the value set for the default host operating system when a new controller is shipped from manufacturing.

Historically, the settings for this host type mimicked the Windows host type behavior. However, as the behavior for Windows host types changed over time, the factory default host type did not keep pace with the changes.

Beginning with 08.50.XX.XX and later firmware, the factory default was updated to enable TPGS reporting of implicit ALUA support, which makes it much more compatible with current generation multipath solutions and would likely result in better interoperability with most hosts.

Note: It is still recommended that the user set the default host operating system type to the most common host type with access to the default host group.

To avoid all issues and limitations associated with the default host group, creating a new DE Series host group specifically used to support cluster access to a given set of volumes is a preferred method for setting up DE Series arrays in any clustered host environment.

3.4 Specific Host Operating Systems and Associated Multipath Solutions

The following information about specific host operating systems and associated multipath solutions is provided for background information and can be used in determining the appropriate host type to use for a given deployment.

Microsoft Windows: Lenovo DE Series Specific MPIO DSM

Lenovo develops a DE Series device-specific module (DSM) for the Windows MPIO multipath architecture. The current DSM revision 2.0 utilizes T10-SPC standard TPGS commands. As with all drivers and host plug-ins, it is essential that the version installed on the host was certified with the particular firmware versions in use on the attached arrays. Table 3 lists changes in the associated host type settings on the Windows host types over recent DE Series firmware releases:

Table 3) Windows MPIO: Lenovo DSM revisions.

Firmware/NVSRAM Version	Expected DSM Version	Failover Method	Failback Method
08.50.XX.XX and later	2.X	Implicit ALUA/TPGS	Implicit/Array-driven (target failback)

Support for legacy failover methods has been retained in both array firmware and the DSM to facilitate backward/forward compatibility of the driver and firmware, so current firmware most likely works with an older driver and vice versa. Nonetheless, the Lenovo Interoperability Matrix should be consulted to determine proper versions for fully certified support because not all combinations are guaranteed to work.

Note that Microsoft also has a generic T10-SPC compliant DSM that might appear to work in most cases to manage multipath support of DE Series storage devices because both that DSM and DE Series firmware are T10-SPC compliant. However, unlike the Lenovo specific DSM, the Microsoft DSM is not tuned to interoperate specifically with DE Series and so does not handle all SAN fault scenarios optimally. Therefore, Lenovo does not test with the generic/Microsoft DSM, and it is not officially supported.

There are two Windows host types defined in DE Series firmware/NVSRAM, one for clustered solutions and one for nonclustered solutions. However, the settings for these two host types are effectively identical, and the reason for two separate host types is only historic at this point.

Linux Multipath Solutions

MPP/RDAC (Kernel 3.9 or earlier)

Historically, Lenovo developed an DE Series -specific multipath driver for Linux using a vendor-unique architecture and legacy/vendor-unique mode and inquiry pages to interact with the DE Series array. This is the Linux MPP driver that is sometimes called RDAC. Support for this driver was officially end of life before 08.50 firmware, although the host type remains in NVSRAM/firmware before 08.51 for the sake of facilitating existing deployments upgrading to later firmware.

Note: After SAN OS is upgraded to 8.52.XX.XX or a later version, a new host type must be selected to use the new multipath solution.

Device Mapper Multipath (DM-MP)

Linux device mapper multipath (DM-MP) uses a plug-in architecture where a device handler understands the specific attributes of a particular storage device. There are two device handlers available in box in most Linux distributions that understand how to manage multipath for DE Series storage: scsi_dh_rdac and scsi_dh_alua. The scsi_dh_rdac device handler uses the DE Series vendor-unique mode and inquiry pages to manage failover and failback, whereas scsi_dh_alua uses T10-SPC TPGS standard methods and is not vendor-specific at all. The host type set in the array configuration determines which device handler claims DE Series storage devices. Therefore, the exact same Linux distribution might select scsi_dh_rdac as the DM-MP device handler if the Linux DM-MP (kernel version 3.9 and earlier) host type is selected in SAN OS during the array configuration, or it might select scsi_dh_alua as the DM-MP device handler if the Linux DM-MP (kernel version 3.10 and later) host type is selected.

Support for scsi_dh_alua and the associated Linux DM-MP (kernel version 3.10 and later) host type was released in SAN OS 08.50.XX.XX firmware/NVSRAM. As the name suggests, Linux distributions based on kernel versions 3.10 and later have the appropriate version of the scsi_dh_alua device handler to properly interoperate with E-Series. Earlier versions of Linux must use the scsi_dh_rdac device handler with DM-MP for proper interoperability, so the Linux DM-MP (kernel version 3.9 and earlier) host type should be selected for hosts running Linux distributions based on kernel version 3.9 or earlier.

Note that sometimes one of the DM-MP device handlers distributed in box on Linux claims DE Series storage devices if the host type is set to another host type that happens to cause the array to return the expected answers for TPGS/ALUA support. Therefore, care must be taken to understand exactly what is installed on the host, what is in box with the Linux distribution in use, and what the host type is set to in the array configuration to fully understand which multipath solution (or lack of any potentially) is in effect.

Cluster Thrash Risk with Linux (DM-MP/scsi_dh_rdac and MPP/RDAC)

It should be noted that the scsi_dh_rdac device handler on DM-MP and legacy MPP/RDAC driver on Linux use explicit host-driven failback. As noted earlier, when using these drivers in a cluster environment, the multipath configuration on the host must be set to disable failback to avoid ownership thrash.

Note that beginning with 08.50.XX.XX , and later firmware builds, support for a special failback disable mechanism within the array firmware can be applied for DM-MP/scsi_dh_rdac to assist with avoiding ownership thrash when DM-MP/scsi_dh_rdac must be used due to specific Linux versions in use. See the Preventing Failback and Cluster Thrash with Non-ATTO Host Types section later for details.

Because the legacy MPP multipath solution uses explicit failover and explicit failback and does not have ALUA support at all, it is not cluster-friendly and should be avoided in clustered solutions.

3.5 Host Context Agent (HCA)

Given both the complexity and importance of properly configuring hosts and host ports, selecting the correct host type when configuring the array is the first step to a successful multipath implementation. The SAN OS host context agent software was developed to assist and automate much of the host setup process. This agent, packaged with Storage Manager, detects when new storage arrays are attached to a host. The agent gathers information from the OS to determine which drivers are installed, and it gathers information from the newly attached array to confirm the supported host types and the port worldwide unique identifiers that connect the host to the new array. This information is then used to automatically configure the host and host ports in the array as well as set the host type correctly.

Note that the HCA does not update an existing configuration, so if the host port worldwide unique identifier has already been configured/registered in the array configuration, it is not reregistered if the host is removed and then reattached. This is intentional to preserve any configuration changes the user applied manually after the initial configuration. Similarly, if the host port worldwide unique identifier and host had previously been manually configured by the user, the HCA does not modify that manual configuration.

In some cases, it might not be desirable or feasible to install SAN OS software packages on the I/O attach host to enable HCA support, but if it is possible, the HCA might provide one alternative to simplify the initial setup.

3.6 Volume Not on Preferred Path Needs Attention Condition

Whenever volume ownership is transferred away from the preferred controller, the DE Series firmware posts a volume not on preferred path (VNOPP) alert to the user. This is intended to indicate that some sort of SAN connectivity fault resulted in one or more hosts in the cluster directing I/O at the nonowning controller, which in turn resulted in a failover and volume ownership changed from the preferred DE Series controller to the nonpreferred controller.

However, in cluster configurations where the multipath driver or the host interoperability with the array is suboptimal in some way (such as use of an explicit ALUA driver or lack of any multipath driver at all on one or more hosts in the cluster), the VNOPP alert can be triggered if hosts begin directing I/O at the nonowning controller or an explicit ALUA driver specifically requests an ownership change on a volume. This can occur without any connectivity fault in the SAN. The result is a phantom alert from the user's viewpoint because there appears to be no reason for the issue and no issue that can be resolved to make the alert go away.

The key to eliminating these phantom VNOPP alerts is to make sure of proper array/host interoperability, and that starts with building a cluster that avoids using any explicit ALUA drivers (or at least employs implicit failover and disables any explicit failback). The mechanisms and best practices in this document are intended to reduce phantom VNOPP alerts such that any alert should be a true indicator of SAN connectivity issues that need to be addressed.

Note that because failback is disabled in most cluster environments, when a real SAN connectivity fault causes a legitimate VNOPP alert, the alert is not automatically cleared after the connectivity issue is corrected unless the user takes manual action to realign volume ownership back with the preferred controller. Long term, as solutions are built that can allow implicit failback within a cluster environment, the need to manually clear VNOPP alerts might eventually be eliminated.

4 Media Specific Features and Enhancements

The following DE Series features and enhancements were implemented with the specific needs of the media and entertainment user, targeting use cases where storage configurations are more dynamic than would usually be the case in a typical enterprise data center. In these environments, host access redundancy is often less of a requirement.

4.1 Preventing Failback and Cluster Thrash with Non-ATTO Host Types

Suppressing reporting of preferred path information from the DE Series array is required when using host types that utilize explicit failover and explicit failback in a cluster configuration. To enable the suppression, an SMcli command is used to set a parameter on a per–host type basis. This is intended to be used primarily if the cluster has Linux hosts with kernel version 3.9 or earlier and DM-MP is being used as the failover solution (utilizing the scsi_dh_rdac device handler, which uses explicit failback), or if the cluster has Windows hosts and pre-08.50.XX.XX firmware on the array, which implies the Windows MPIO DSM is operating in explicit failback mode.

Note: This method is not intended for use on any host type/driver mode that already supports implicit failback (or no failback).

Use of this mechanism is only supported on a per customer basis, consult your Lenovo Marketing representative. This is to make sure the specific host OS/drivers and HBAs in use on the customer configuration have been tested and are known to react favorably to the firmware behavior of not reporting any preferred ownership information.

To enable this for the Windows host types, issue the following commands to the DE Series array using SMcli:

```
set controller[a] HostNVSRAMbyte[1, 61]=1,1;
set controller[a] HostNVSRAMbyte[8, 61]=1,1;
set controller[b] HostNVSRAMbyte[1, 61]=1,1;
set controller[b] HostNVSRAMbyte[8, 61]=1,1;
```

To enable this for the Linux DM-MP (kernel 3.9 and earlier) host type, issue the following commands to the DE Series array using SMcli:

```
set controller[a] HostNVSRAMbyte[7, 61]=1,1;
set controller[b] HostNVSRAMbyte[7, 61]=1,1;
```

Note: All hosts must be forced to rescan/rediscover storage volumes mapped from the array after executing these CLI commands for the changes to be picked up by the host multipath drivers.

The DE Series controller firmware persists this new setting enabled using the CLI in such a way as to apply it to all hosts of the same type, so it only should be enabled once for each of the unique host types that have a need to use it. This does disable all failback behavior for the associated hosts of that type, so the user must expect that failback following recovery of a SAN fault does not work after this option is enabled. Given that in a cluster, the user should have been disabling failback from the host settings anyway, this should be expected behavior. This setting also persists through subsequent updates of NVSRAM or controller firmware.

Important note for Linux DM-MP: If all arrays accessible by a given host have this option enabled to suppress preferred ownership on the Linux OS with kernel version 3.9 or earlier host type or in cases where the Linux kernel version 3.10 or later host type is being used (which suppresses preferred ownership information reporting to the host by default), then the DM-MP configuration on that host should be set to enable failback `immediate`, even in a cluster environment. This forces DM-MP to follow target port group asymmetric access state changes more aggressively, which is desirable if the host is not provided with preferred ownership information and the path routing decision is based solely on target port group asymmetric access states. However, if the host has access to any array that does not have this new option set to suppress preferred ownership information reporting (or is potentially reconnected at a later time to such an array), then failback should be set to `manual` in the DM-MP configuration on all cluster nodes to avoid ownership thrash.

5 Recommended Configuration Best Practices

The following procedures are intended to assist in building a cluster that is resistant to volume ownership thrashing. Although many of these concepts are applicable to any clustered solution where multiple hosts have shared access to the same storage volumes, this is specifically targeted at deployments utilizing heterogeneous host-type clustered file systems.

5.1 Determine Requirement for Host Access Redundancy and Solution Design

First, the requirement for host access redundancy (failover) must be understood. If loss of access due to SAN connectivity faults or array controller faults/failures cannot be tolerated in the customer environment, then some sort of multipath solution must be deployed on the hosts in the cluster, and the array must be configured accordingly. Note that with the addition of support to DE Series firmware for more implicit mode ALUA multipath solutions and new mechanisms available for disabling failback using the array configuration, it is very possible to support this redundancy requirement with minimal risk of ownership thrash. These new solution options should be considered when discussing the redundancy requirements during the design work for a new deployment.

Explicit Setup of Host Group/Host/Host Ports: Best Practice

Regardless of whether a multipath solution is deployed or not, it is a best practice to inventory all hosts in the cluster accessing the clustered file system and set up a host group. Next, individually define the hosts with their associated host type based on the OS and multipath solution in use on each and assign the host port worldwide unique identifiers to each host. This allows the array to properly interoperate with the OS/multipath driver on each host in the cluster.

Design Option: Ownership Thrash-Resistant Multipath Cluster

If a multipath solution is selected for the configuration design to meet the solution redundancy requirement or if there is not a strict redundancy requirement, then the proper cluster configuration must be set up in order to create a thrash-resistant configuration. This setup simply involves configuring the hosts in the array to have the appropriate host type and host group association. On the hosts, it’s a matter of making sure the appropriate Lenovo certified multipath solution is installed (if not already in box on the OS) and configured on each host. In some cases, there might not be any host-side installation or setup at all if the driver is in box. You might just have to validate that a certified OS version is in use and the appropriate host type in the array configuration was selected.

Assuming the hosts combined in the cluster are using one of the cluster-compatible multipath solutions listed in Table 9, risk of ownership thrash is greatly reduced. Clusters configured this way with properly installed (or in-box) multipath solutions should be able to fail over when fault conditions occur and provide redundant host access to the DE Series storage.

Table 4) Cluster-compatible multipath solutions.

OS	HBA	Multipath Solution	Recommended Host Type
Linux kernel version 3.9 or earlier[2]	Lenovo certified non-ATTO HBA	In-box, DM-MP (scsi_dh_rdac)	Linux (kernel 3.9 or earlier)
Linux kernel version 3.10 or later[2,3]	Lenovo certified non-ATTO HBA	In-box, DM-MP (scsi_dh_alua)	Linux (kernel 3.10 or later)
Windows Server 2012/2016 cluster (v2.0 or later Lenovo DE Series DSM)	Lenovo certified non-ATTO HBA	DE Series MPIO DSM v2.0 or later	Windows clustered

1. Limited to revisions of the OS and HBAs certified by Lenovo Interoperability Matrix.

2. Requires 08.50.XX.XX firmware and compatible NVSRAM or later.

In cases where 08.50.XX.XX and later firmware is in use on the DE Series array and hosts in the cluster are only using non-ATTO HBAs and running either Linux with kernel versions 3.10 and later or Windows with version 2.X of the Lenovo MPIO DSM, it is also possible to provide autofailback functionality without risk of ownership thrash.

6 Multipath-Related Features in 08.50.XX.XX Firmware

The following information is provided as additional background on some of the multipath-related features in the SAN OS 08.50.XX.XX firmware code. Apart from using the implicit or target failback feature for a few specific host types, most of the following feature descriptions do not apply directly to media customers. However, understanding the details of how these features operate (and how they are disabled if necessary) can be useful for anyone assisting in deploying DE Series arrays running 08.50.XX.XX or later firmware into a media and entertainment environment.

6.1 Automatic Load Balancing

The automatic load balancing (ALB) feature takes advantage of the behavior of current generation host multipath drivers and implicit ALUA to rebalance dynamic workloads across the two array controllers automatically. The incoming I/O workload to each volume is monitored, and periodic rebalances occur with the array controller triggering volume ownership transfers as needed to shift workload from one controller to the other. This uses the fact that the host multipath drivers are notified of the implicit target port group asymmetric access state change when an ownership transfer occurs, and drivers that operate in implicit ALUA mode then follow that change and start directing I/O at the new active/optimized target port group (new owning controller for a given volume). This is especially useful in environments where the workload distribution across volumes is highly dynamic and a simple fixed balance of ownership between the two controllers does not achieve an acceptable workload balance.

In the media and entertainment use cases where a clustered file system is striped across a small number of volumes on the array, ALB likely does not provide much added value. The real benefit of this feature in the media space is the shift to support for implicit ALUA within DE Series, which was a necessary move to support the array-initiated ownership transfers for rebalancing workloads. Without implicit ALUA support, there would be no opportunity to provide thrash-resistant failover capabilities within a clustered environment.

ALB is sensitive to the fact that some environments cannot tolerate regular volume ownership transfers, and the design encompasses several aspects to deal with that:

- ALB has a global switch that allows the user to enable or disable the functionality. ALB is enabled by default.
- ALB does not transfer ownership of a volume if any host with access to that volume is not running a multipath solution that can tolerate implicit ownership transfers. See the ALB column in Table 2 for details of which host types/multipath solutions support ALB.
  - This also means that environments using explicit ALUA that would be sensitive to ownership thrash are automatically disqualified for ALB functionality.
- ALB has a built-in thrash avoidance mechanism to avoid the same volume from being transferred too often for the purpose of rebalancing load. There is also a mechanism that monitors host-side behavior and disables ALB transfers if a host multipath driver does not follow an implicit ownership transfer for some reason.
- The ownership transfer process used during ALB was highly optimized to minimize any I/O latency spikes when the process is activated.

Implicit/Target Failback

Support for implicit failback (also known as target failback or array-initiated failback) was added with ALB in 08.50.XX.XX firmware but is not enabled/disabled with ALB. Like ALB, the functionality is only applied to volumes accessible to hosts that are running a multipath solution that can tolerate implicit ownership transfers. See Table 2 for the specific host types and associated multipath solutions that can support implicit or target failback.

Target failback support provides failback functionality for hosts with multipath drivers that do not support a failback process at all or drivers that fail back by simply redirecting I/O to the nonowning (but preferred) controller upon recovery of a SAN connectivity fault. This implies a performance penalty for I/O shipping for some period of time.

The implicit failback feature is a side benefit from the ALB feature. In order for ALB to succeed in rebalancing workloads between peer controllers or get hosts to redirect I/O to the desired controller, it must be certain that all hosts with access to that volume are connected to the controller that takes ownership of the volume. Additionally, the multipath driver must have discovered paths to the new owning controller as well. That means ALB has implemented tracking within the controller firmware of all host-to-controller connectivity and multipath discovery. This combines transport protocol–level connectivity status information for all connections between the hosts and controllers and command-level protocol monitoring that watches for incoming commands that are indicative of host multipath driver discovery of a given path. The net result is that the array controller is now aware of connectivity faults within the SAN and when those faults recover such that all hosts with access to a given volume have regained access to the preferred controller. The controller uses this tracking data to initiate a failback ownership transfer.

Perhaps the most important aspect of this is the fact that in a clustered environment, the array controller has visibility to all hosts in the cluster and their associated connectivity status, which allows it to determine when an appropriate time is to initiate a failback without creating any ownership thrash (no “fights” for setting ownership between the hosts in the cluster).

Given that the DE Series host type controls this mechanism as noted in Table 2, there is no need to explicitly disable the target failback capability in a media environment as long as host types are properly configured.

6.2 Connectivity Reporting

Another benefit of the connectivity tracking added for ALB support in 08.50.XX.XX and later firmware is new reporting mechanisms to augment the VNOPP alert.

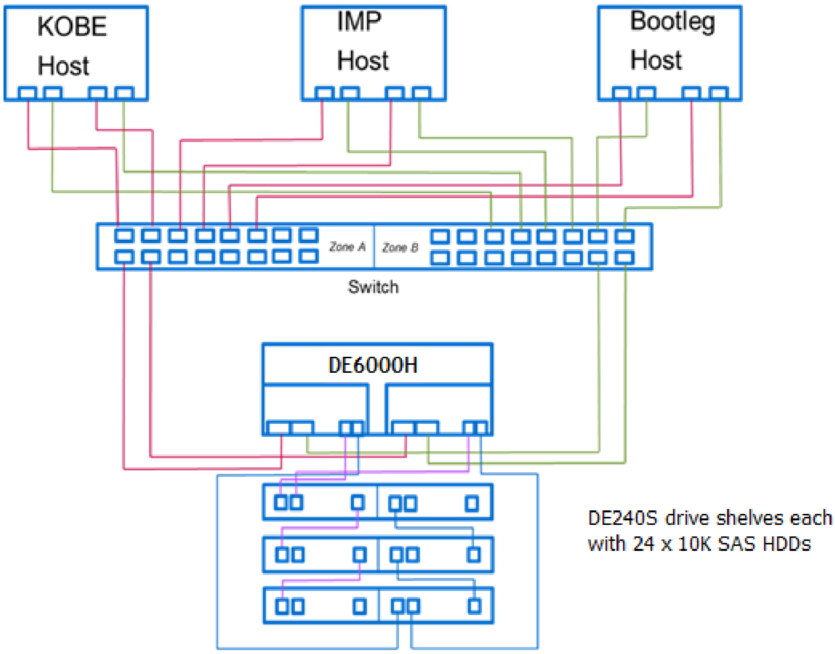
Rather than waiting for a volume to be transferred off the preferred controller before raising an alert, the array uses the internal SAN connectivity tracking logic to detect and report loss of redundant host connectivity as soon as it happens. This means that an alert is raised even if connectivity is lost to the nonpreferred path, which previously would not have raised the VNOPP alert. The new alert for host redundancy loss is also more specific in indicating the host that lost connectivity and to which array controller. This was not certain previously in cluster environments when multiple hosts were sharing access to the volume that reported VNOPP because the alert was only reported on a volume and not a specific host or hosts.

Note: This new connectivity reporting is only enabled if ALB is enabled. It was added primarily because ALB cannot function properly in environments where SAN connectivity faults have occurred.

7 Example Configuration

For this report, we used a DE6000H shelf (RBOD) with 24 x 10k SAS drives. This RBOD was connected to 3 DE240S expansion shelves (EBODs), each with 24 x 10k SAS drives. The four shelves were connected as shown in Figure 2.

Figure 2) Drive tray configuration: cluster with DE Series.



7.1 Storage Configuration Steps

The storage array system was configured by using the following steps:

1. Established IP addresses for both controllers in the RBOD and specified the storage array name.
2. Connect the Fibre Channel (FC) HIC of the controllers in the RBOD to the two FC switches using fiber cables. The switches were connected to three hosts, namely, IMP, Bootleg, and Kobe, using FC cables. Each host had two FC HBAs installed, and the hosts had the following characteristics:
   a. ThinkSystem SR650
   b. 64G Memory
3. FC HBA listed in Lenovo Interoperability Matrix for DE-Series
4. Used ThinkSystem System Manager to manage DE the storage array system
5. Created eight volume groups with one volume per volume group using 8+2 RAID 6 (total capacity: 104.450TB usable, 104.450TB used).

Name	Status	Usable Capacity	Used Capacity	Free Capacity	RAID Level	Drive/Media Type	Volumes	Secure Capable	DA Capable
fbc_5624_03_vg00	Optimal	13.056 TB	13.056 TB	0.000 MB	6	Serial Attached SCSI (SAS), Hard Disk Drive	1	Yes (Non Secure)	Yes
fbc_5624_03_vg01	Optimal	13.056 TB	13.056 TB	0.000 MB	6	Serial Attached SCSI (SAS), Hard Disk Drive	1	Yes (Non Secure)	Yes
fbc_5624_03_vg02	Optimal	13.056 TB	13.056 TB	0.000 MB	6	Serial Attached SCSI (SAS), Hard Disk Drive	1	Yes (Non Secure)	Yes
fbc_5624_03_vg03	Optimal	13.056 TB	13.056 TB	0.000 MB	6	Serial Attached SCSI (SAS), Hard Disk Drive	1	Yes (Non Secure)	Yes
fbc_5624_03_vg04	Optimal	13.056 TB	13.056 TB	0.000 MB	6	Serial Attached SCSI (SAS), Hard Disk Drive	1	Yes (Non Secure)	Yes
fbc_5624_03_vg05	Optimal	13.056 TB	13.056 TB	0.000 MB	6	Serial Attached SCSI (SAS), Hard Disk Drive	1	Yes (Non Secure)	Yes
fbc_5624_03_vg06	Optimal	13.056 TB	13.056 TB	0.000 MB	6	Serial Attached SCSI (SAS), Hard Disk Drive	1	Yes (Non Secure)	Yes
fbc_5624_03_vg07	Optimal	13.056 TB	13.056 TB	0.000 MB	6	Serial Attached SCSI (SAS), Hard Disk Drive	1	Yes (Non Secure)	Yes

6. The eight volumes were confirmed to have the following characteristics:
    a. Tray loss protection: No
    b. Data assurance (DA) capable: Yes
    c. DA-enabled volume present: No
    d. Read cache: Enabled
    e. Write cache: Enabled
    f. Write cache without batteries: Disabled
    g. Write cache with mirroring: Enabled
    h. Flush write cache after (in seconds): 10.00
    i. Dynamic cache read prefetch: Enabled
    j. Enable background media scan: Enabled
    k. Media scan with redundancy check: Disabled
    l. Preread redundancy check: Disabled
7. Following is an example of a 10-drive distribution for a RAID 6 volume group and, in this case, its associated single volume between trays (shelves) 99, 0, 1, and 2. All eight volume groups were striped across the 4 shelves in a similar manner.

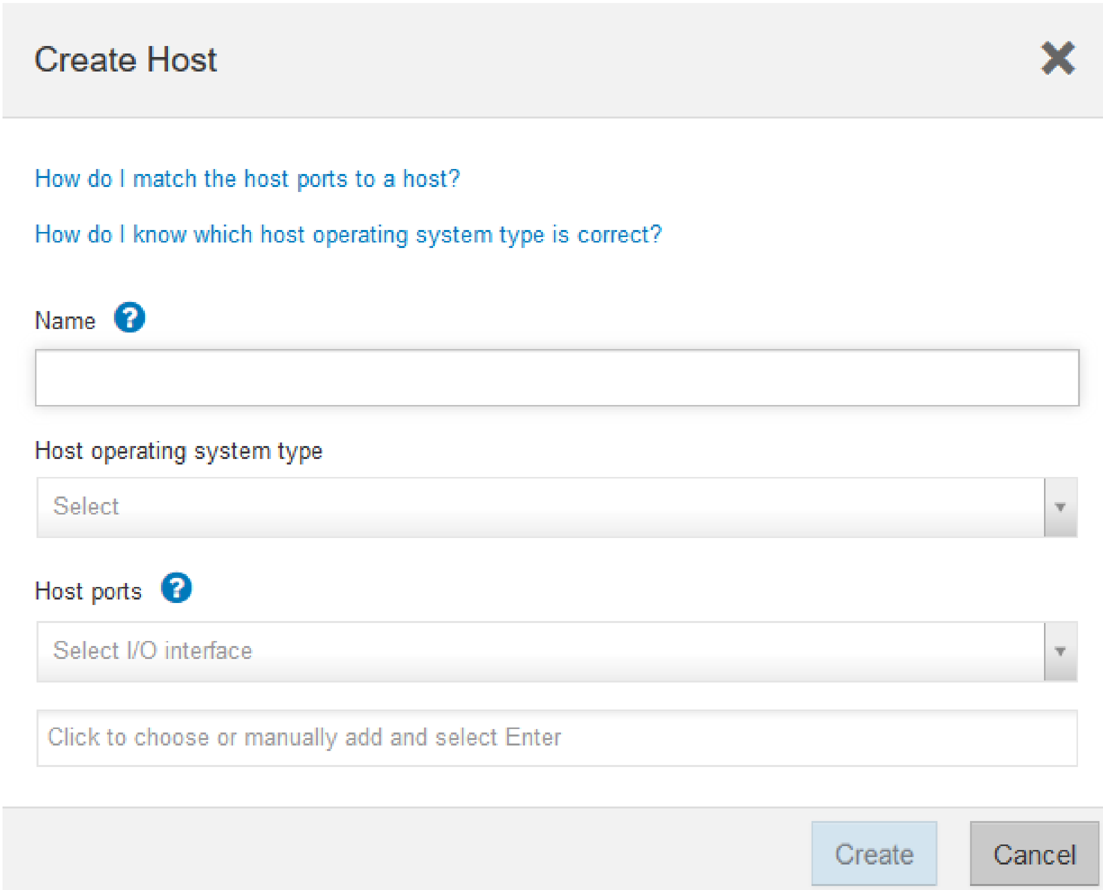
Tray	Slot
1	3
2	3
99	4
0	4
1	4
2	4
99	5
0	5
1	5
2	5

8. Defined the zoning on switch A and switch B. The zones are defined such that any HBA in each host can access one target port from one array controller from each switch.

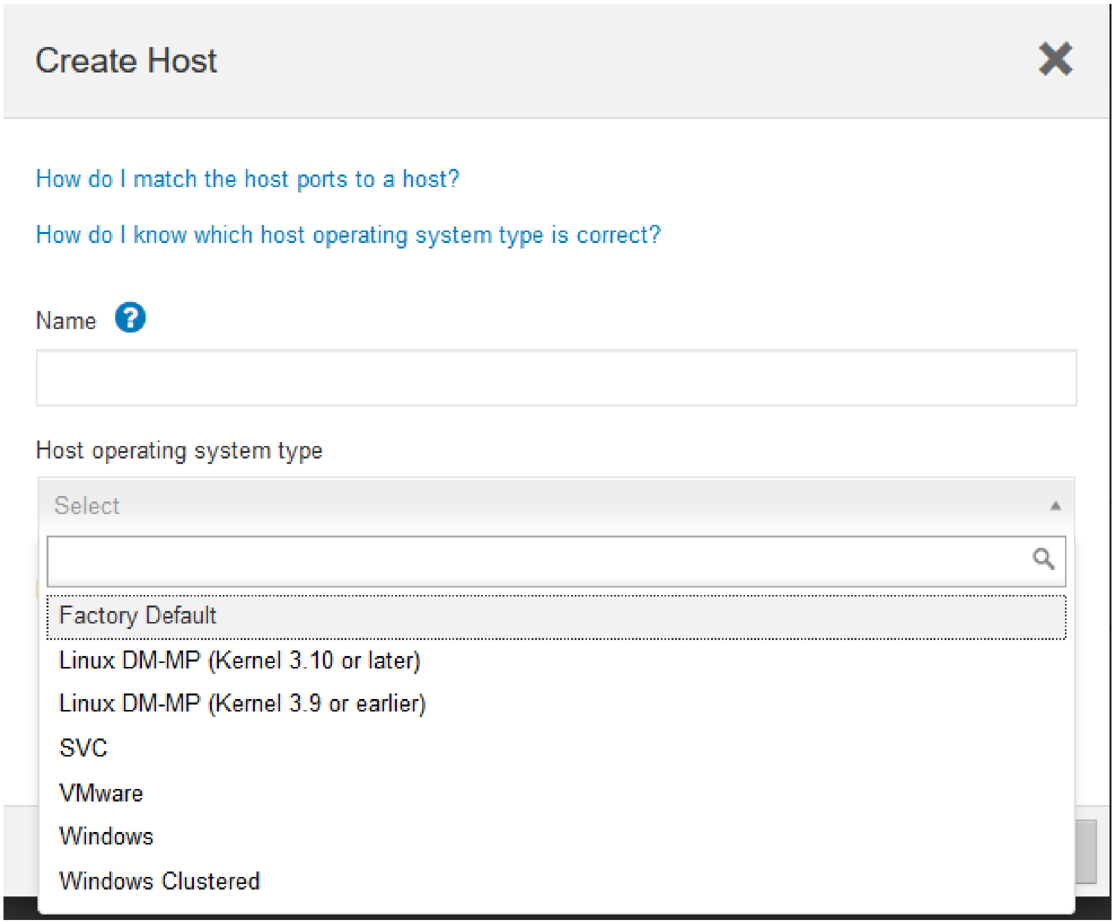
Switch A	Switch B
Defined configuration: cfg: Cluster Bootleg_DE6000A0; IMP_ DE6000A1; Kobe_ DE6000B1; MDC1_ DE6000; MDC2_ DE6000 zone: Bootleg_ DE6000A0 DE6000A0; Bootleg zone: IMP_ DE6000A1 DE6000A1; IMP zone: Kobe_ DE6000B1 DE6000B1; Kobe zone: MDC1_ DE6000 DE6000A0; DE6000A1; DE6000B0; DE6000B1; MDC1 zone: MDC2_ DE6000 DE6000A0; DE6000A1; DE6000B0; DE6000B1; MDC2 alias: Bootleg 1,13 alias: DE6000A0 1,0 alias: DE6000A1 1,4 alias: DE6000B0 1,1 alias: DE6000B1 1,5 alias: IMP 1,14 alias: Kobe 1,16 alias: MDC1 1,3 alias: MDC2 1,7 Effective configuration: cfg: Cluster zone: Bootleg_ DE6000A0 1,0	Defined configuration: cfg: Cluster Bootleg_DE6000B2; IMP_DE6000B3; Kobe_DE6000A3; MDC1_DE6000; MDC2_DE6000 zone: Bootleg_DE6000B2 DE6000B2; Bootleg zone: IMP_DE6000B3 DE6000B3; IMP zone: Kobe_DE6000A3 DE6000A3; Kobe zone: MDC1_DE6000 DE6000A2; DE6000A3; DE6000B2; DE6000B3; MDC1 zone: MDC2_DE6000 DE6000A2; DE6000A3; DE6000B2; DE6000B3; MDC2 alias: Bootleg 1,13 alias: DE6000A2 1,0 alias: DE6000A3 1,4 alias: DE6000B2 1,1 alias: DE6000B3 1,5 alias: IMP 1,14 alias: Kobe 1,16 alias: MDC1 1,3 alias: MDC2 1,7 Effective configuration: cfg: Cluster zone: Bootleg_ DE6000B2 1,1 1,13 zone: IMP_DE6000B3 1,5
1,13 zone: IMP_DE6000A1 1,4 1,14 zone: Kobe_DE6000B1 1,5 1,16 zone: MDC1_DE6000 1,0 1,4 1,1 1,5 1,3 zone: MDC2_DE6000 1,0 1,4 1,1 1,5 1,7	1,14 zone: Kobe_DE6000A3 1,4 1,16 zone: MDC1_DE6000 1,0 1,4 1,1 1,5 1,3 zone: MDC2_DE6000 1,0 1,4 1,1 1,5 1,7

9. Defined hosts under Host icon in Storage menu

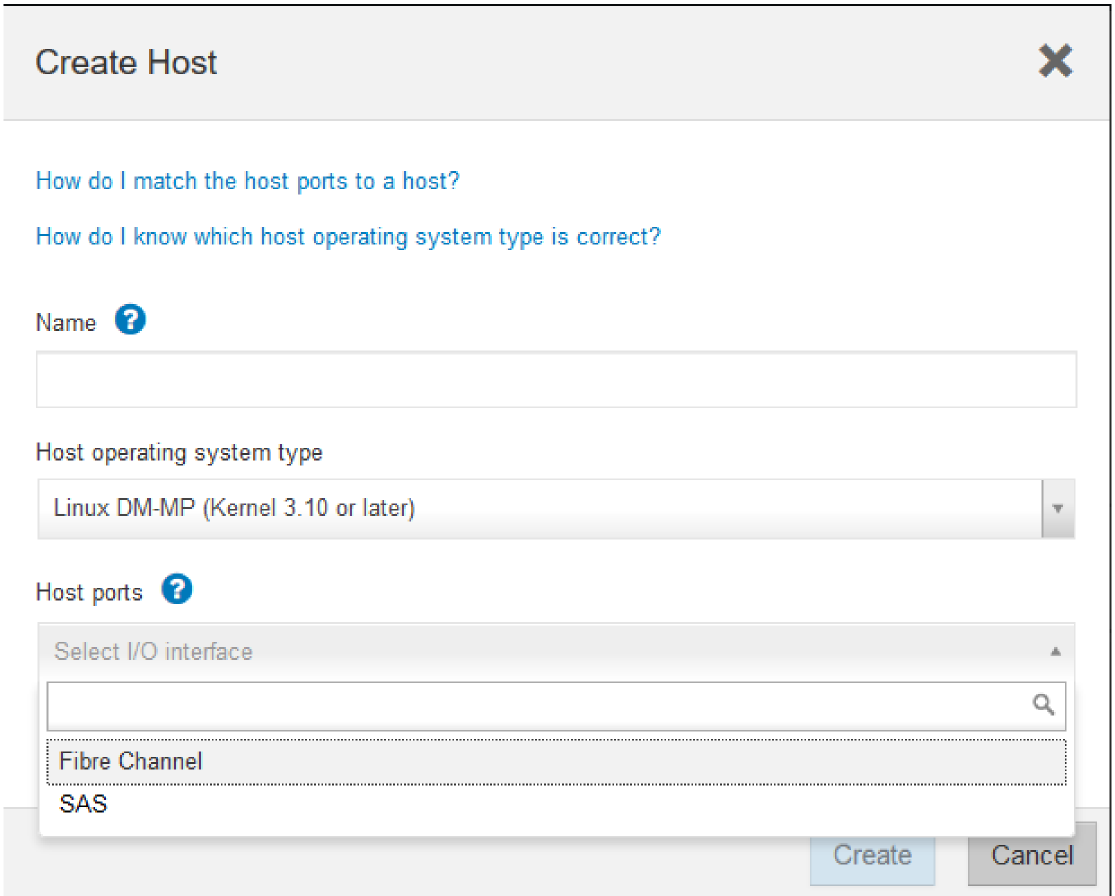
m. Select Storage menu item
n. Select Hosts icon
o. Click Create -> Host.



p. Input the host or client name.
q. Select Host operating system type



r. Select I/O interface under Host Ports.



s. Specify the HBA port identifiers that belong to the host being defined:
   i. The WWPN of the active initiator port in same zoning will listed in the dropdown
   ii. Select the WWPN
   iii. Repeat steps ii through iii for the remaining HBA ports.

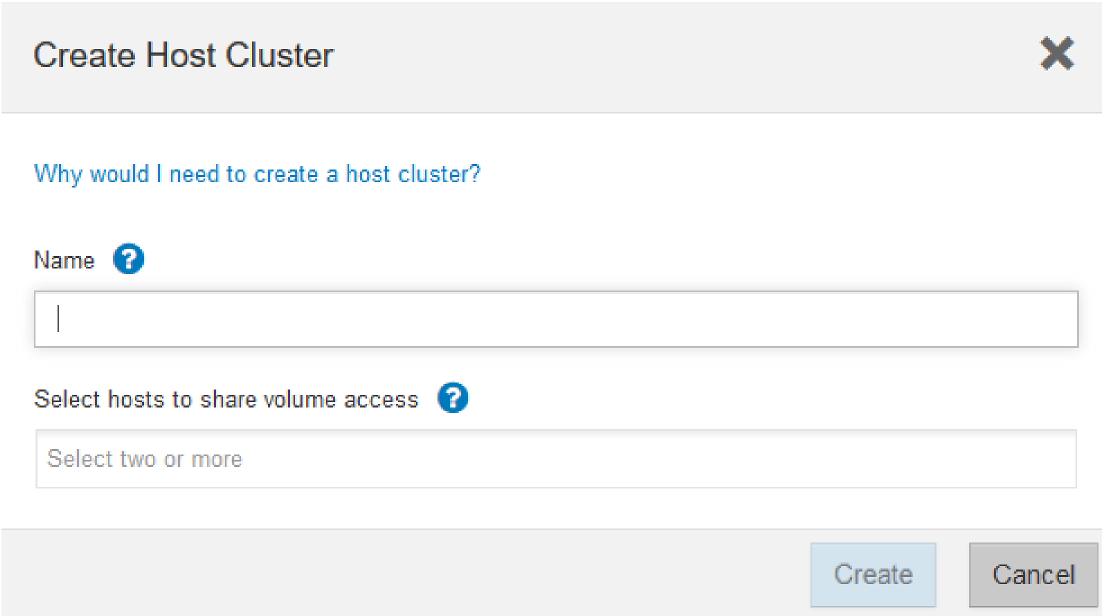
t. Select Create.

u. Repeat the steps for all Host

10. Defined host groups under Host icon in Storage menu

v. Select HOSTS icon under Storage menu.

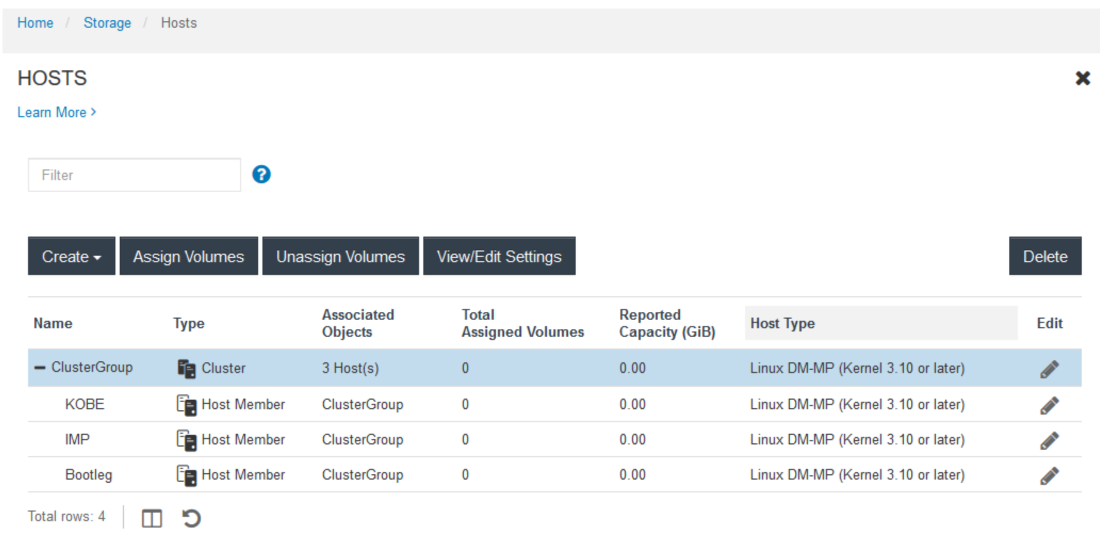
w. Click Create -> Host cluster.



x. Enter the host group name (for example, ClusterGroup).

y. Select Hosts defined in previous step

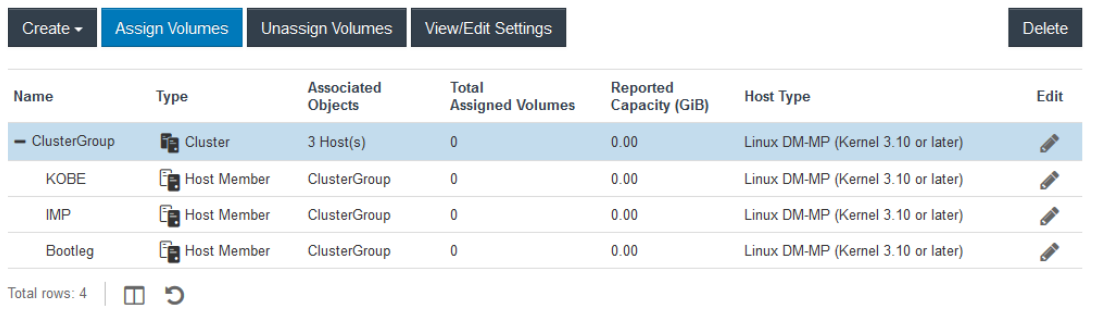
z. Select OK.



11. Assign volumes to the host cluster:

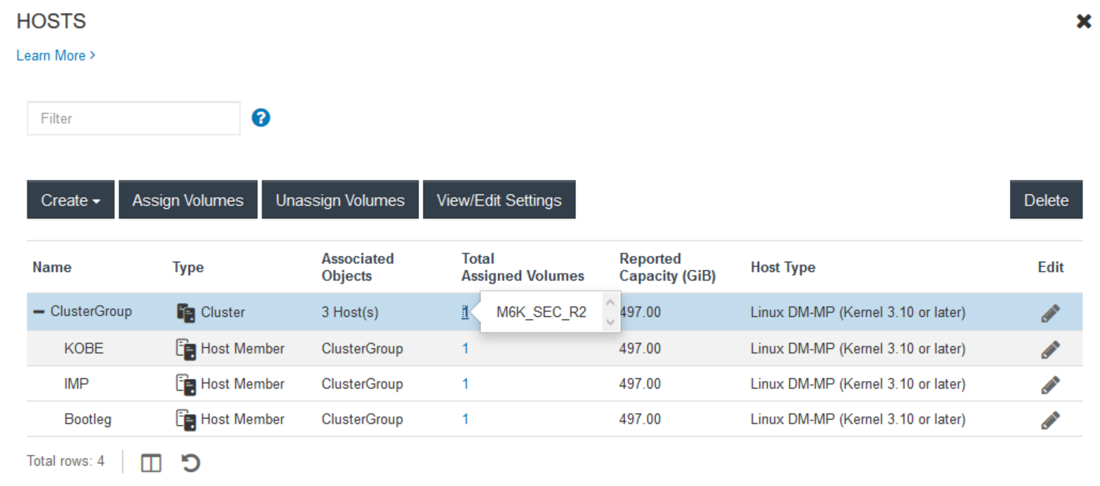
    aa. Select Host cluster (for example, ClusterGroup).

    bb. Click the Assign Volumes



cc. Check volumes from the volume list.

dd. And click Assign.



8 Conclusion

Lenovo engineering continues to improve host/array interoperability and compatibility with host multipath solutions to better support clustered host configurations with shared access to the same set of storage volumes.

DE Series

08.50.XX.XX DE Series design also continues to evolve in a direction that is more cluster-friendly, emphasizing implicit ALUA to support failover in shared volume access environments today and identifying potential opportunities to continue to expand these capabilities going forward.

Where to Find Additional Information

To learn more about the information that is described in this document, review the following documents on Lenovo DataCenter Support, https://datacentersupport.lenovo.com/products/storage/DEseries:

- DE Series Software Deployment and Configuration Guide
- DE Series 2U/4U Hardware Installation and Maintenance Guide

Contacting Support

You can contact Support to obtain help for your issue.

You can receive hardware service through a Lenovo Authorized Service Provider. To locate a service provider authorized by Lenovo to provide warranty service, go to https://datacentersupport.lenovo.com/serviceprovider and use filter searching for different countries. For Lenovo support telephone numbers, see https://datacentersupport.lenovo.com/supportphonelist for your region support details.

Notices

Lenovo may not offer the products, services, or features discussed in this document in all countries. Consult your local Lenovo representative for information on the products and services currently available in your area.

Any reference to a Lenovo product, program, or service is not intended to state or imply that only that Lenovo product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any Lenovo intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any other product, program, or service.

Lenovo may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document is not an offer and does not provide a license under any patents or patent applications. You can send inquiries in writing to the following:

Lenovo (United States), Inc.
8001 Development Drive

Morrisville, NC 27560
U.S.A.

Attention: Lenovo Director of Licensing

LENOVO PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT,

MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. Lenovo may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

The products described in this document are not intended for use in implantation or other life support applications where malfunction may result in injury or death to persons. The information contained in this document does not affect or change Lenovo product specifications or warranties. Nothing in this document shall operate as an express or implied license or indemnity under the intellectual property rights of Lenovo or third parties. All information contained in this document was obtained in specific environments and is presented as an illustration. The result obtained in other operating environments may vary.

Lenovo may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any references in this publication to non-Lenovo Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this Lenovo product, and use of those Web sites is at your own risk.

Any performance data contained herein was determined in a controlled environment. Therefore, the result obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Trademarks

LENOVO, LENOVO logo, and THINKSYSTEM are trademarks of Lenovo. All other trademarks are the property of their respective owners.