v5.9 | March 2023 | BP-2049

BEST PRACTICES

Nutanix Volumes

NUTANIX
YOUR ENTERPRISE CLOUD

Copyright

Contents

1. Executive Summary

The Nutanix Cloud Platform is hypervisor-agnostic and can support any application. To support these key advantages, AOS storage delivers storage using multiple protocols, including Network File System (NFS), Server Message Block (SMB), and Internet Small Computer System Interface (iSCSI). Nutanix Volumes is enterprise-class, software-defined storage that exposes storage resources directly to virtualized guest operating systems or physical hosts using the iSCSI protocol.

Nutanix Volumes facilitates support for several use cases:

- iSCSI for Microsoft Exchange Server
  - Nutanix Volumes allows Microsoft Exchange Server environments to use iSCSI as the primary storage protocol.
- Shared storage for Linux-based clusters and Windows Server Failover Clustering (WSFC)
  - Nutanix Volumes supports SCSI-3 persistent reservations for shared storage-based Windows clusters, which are commonly used with Microsoft SQL Server and clustered file servers.
- Shared storage for Oracle RAC environments
- Bare-metal environments
  - Nutanix Volumes enables server hardware separate from the Nutanix environment to consume Nutanix storage resources, so you can use existing server hardware investments against Nutanix storage. Workloads not targeted for virtualization can also use Nutanix storage.

Nutanix has long supported virtualized workloads and has more recently delivered a native scale-out file serving capability called Nutanix Files and an object-storage feature called Nutanix Objects. With Nutanix Volumes, Files, and Objects, Nutanix supports use cases across virtualized and physical

server workloads, enabling infrastructure consolidation for enterprise cloud environments.

Document Version History

Version Number	Published	Notes
1.0	June 2016	Original publication.
2.0	January 2017	Updated for AOS 5.0.
2.1	February 2017	Updated supported clients.
2.2	May 2017	Updated for AOS 5.1 and added a section on thin provisioning.
3.0	December 2017	Updated for AOS 5.5.
4.0	October 2018	Updated for AOS 5.9. Updated product naming and the Client Tuning Recommendations and SCSI UNMAP sections.
5.0	August 2019	Updated for AOS 5.11. Updated recommendations.
5.1	February 2020	Updated supported clients for AOS 5.11.2.
5.2	June 2020	Updated jumbo frame recommendations.
5.3	May 2021	Updated for SCSI-3 PR and Traffic Isolation IP Pool.
5.4	September 2021	Updated the Clients and Applications Qualified by AOS Release Version table.
5.5	February 2022	Updated Challenge Handshake Authentication Protocol (CHAP) and Best Practices sections.

Version Number	Published	Notes
5.6	May 2022	Updated the Supported Client Environments and Networking Segmentation sections.
5.7	June 2022	Updated the Clients and Applications Qualified by AOS Release Version table.
5.8	January 2023	Updated the Nutanix Volumes Block Storage Overview, Volume Group Connectivity, Networking for iSCSI, and Best Practices sections.
5.9	March 2023	Updated the Nutanix Volumes Block Storage Overview, Volume Group Connectivity, Networking for iSCSI, Best Practices, and Volume Group Data Protection sections.

2. Nutanix Volumes Block Storage Overview

Nutanix Volumes is an enterprise-class, software-defined block storage solution that exposes storage resources directly to virtualized guest operating systems or physical hosts using the iSCSI protocol, or in the case of AHV VMs specifically, through the guest's native storage adaptor.

Nutanix designed Volumes as a scale-out storage solution where every Controller VM (CVM) in a cluster could present storage over iSCSI. This solution allows an individual application to access the entire cluster, if needed, to scale out for high performance. Nutanix Volumes automatically manages high availability to ensure that upgrades or failures are nondisruptive to applications.

Nutanix Volumes implements storage allocation and assignment through volume groups (VGs). A VG is a collection of one or more disks (essentially virtual disks) in a Nutanix storage container. Nutanix Volumes presents these disks to both VMs and physical servers, or hosts.

Nutanix Volumes disks inherit the properties (replication factor, compression, erasure coding, and so on) of the container you create them in. By default, these disks are thin-provisioned.

Once a host has access to a VG, the VG is discovered as one or more iSCSI targets. When it connects to the iSCSI targets, the host discovers the disks as SCSI disk devices. The following figure shows these relationships.

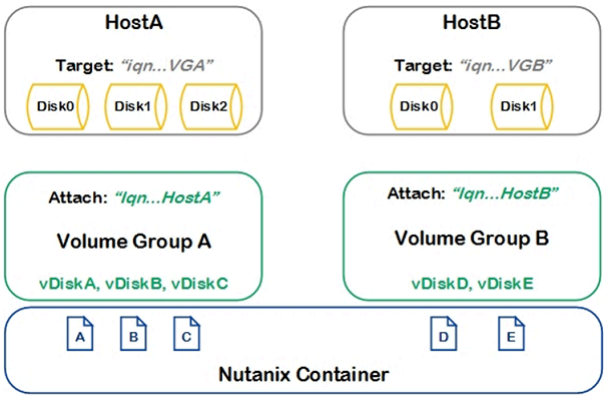

Figure 1: Nutanix Volumes Relationships

You can grow a VG dynamically by adding new disks or expanding existing disks online.

You can use VGs for iSCSI connectivity in your Nutanix cluster, whether the underlying hypervisor is ESXi, Hyper-V, or AHV.

Multiple external hosts can also share disks in a VG for the purposes of shared storage clustering, as shown in the following figure. A common scenario for using shared storage is in WSFC. You can use Prism to attach multiple external initiators to a VG by default. Ensure that you have an appropriate clustering technology (like WSFC) or a clustered file system in place before you share disks across multiple hosts.

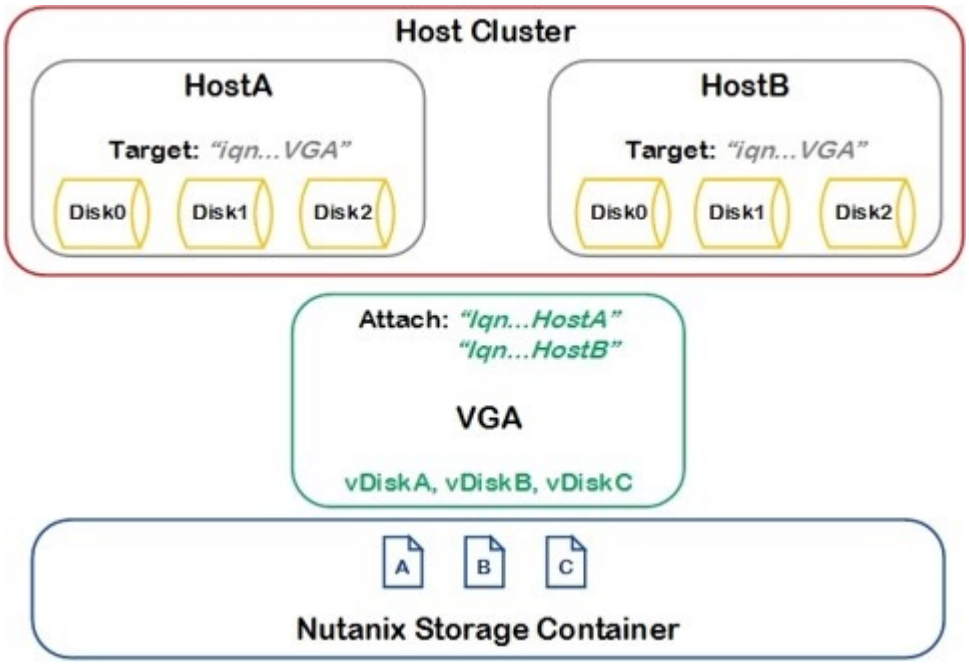

Figure 2: Shared Volume Group

You can use VGs in conjunction with traditional hypervisor vDisks. For example, some VMs in a Nutanix cluster may use .vmdk- or .vhdx-based storage on NFS or SMB, while other hosts use VGs as their primary storage. VMs using VGs have their start and OS drives presented with hypervisor vDisks.

> Note: Nutanix also supports starting an OS from iSCSI using Nutanix Volumes. The Nutanix Support Portal has a list of qualified NICs, HBAs, and operating systems that support starting from iSCSI using Nutanix Volumes.

You can manage VGs from Prism or from a preferred CLI, such as Acropolis command-line interface (aCLI), nCLI, or PowerShell. In Prism, create and monitor VGs from the Storage page, as shown in the following figure.

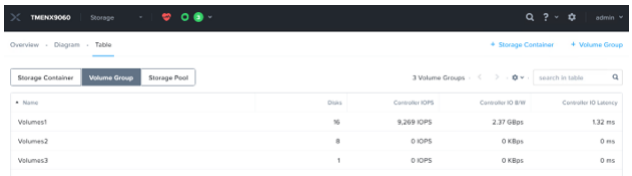

Figure 3: Prism Volume Group Page

3. Supported Client Environments

The following table shows which clients we've qualified and the specific AOS release that introduced that support. For more information refer to the Volumes Guide.

Table: Clients and Applications Qualified by AOS Release Version

Minimum AOS Release Version	Supported Client OS	Qualified Applications
4.7	CentOS 6 and 7, Microsoft Windows Server 2012 R2, Microsoft Windows Server 2008 R2, Oracle Linux 6.x and 7.x, and Red Hat Enterprise Linux (RHEL) 6.7	Oracle RAC, Microsoft SQL Server, and Microsoft Exchange Server
5.0	IBM AIX 7.1 and 7.2 on POWER, RHEL 7.2, and SUSE Linux Enterprise Server (SLES) 11 and 12 (x86 servers)	
5.0.2 and 5.1	RHEL 6.8 (5.1) and Oracle Solaris 11.3 on SPARC	
5.5	Microsoft Windows Server 2016 and RHEL 6.9, 7.3, and 7.4	
5.9		Linux Guest Cluster
5.11	SLES 15 Service Pack 2 (SP2)	
5.11.2	Microsoft Windows Server 2019 and RHEL 7.6 and 7.7	
5.20 and 6.0	RHEL 7.9, 8.2, and 8.4	
5.20.x and 6.0.x	Microsoft Windows Server 2022	

4. Volume Group Connectivity

One of the main advantages of Nutanix Volumes is that it enables a VM or group of VMs to use multiple or even all CVMs in the underlying physical cluster to serve a single VG. By distributing the data in this way, you can harness the power of multiple compute nodes and spread the load across the cluster. This capability, known as volume group load balancing (VGLB), is the default configuration when connecting to a VG over iSCSI.

VMs running on AHV, however, can also directly attach to a VG created on the same cluster. Direct-attached VGs are connected through the guest's native storage adaptor. In this case the CVM local to the client VM serves the VG in its entirety. This method provides simplicity and data locality. Since the CVM hosting the VG is local to the client VM, read traffic doesn't traverse the network, which helps minimize read latency. Although direct-attached VGs don't load balance by default, you can deploy a load-balanced VG to AHV guests by choosing the Load Balance Volume Group option in Prism Central.

If you deployed a VG as direct attached and you later want to convert it to a load-balanced VG, run the following aCLI command:

```
_acli vg.update load_balance_vm_attachments=true_
```

Note: You must remove any VM attachments before running the command.

You can also share direct-attached VGs across VMs for in-guest clustering, like a WSFC, where you need SCSI-3-persistent reservations.

Volume Group Load Balancing

Only one CVM can host any given disk at a time, and all primary storage access for that disk occurs through the hosting CVM, so the host location for the disks can affect network consumption and the balance of CVM usage in a cluster.

VGLB connectivity uses iSCSI redirection to control target path management for disk load balancing and path resilience. The following section explores iSCSI

redirection. Additional information regarding the legacy use of multipath I/O (MPIO) is provided in the appendix.

iSCSI Target Redirection

The preferred method for external connectivity to VGs doesn't involve using MPIO for storage load balancing or storage path resilience. Instead of configuring host iSCSI client sessions to connect directly to each individual CVM, a single iSCSI data services IP (DSIP) address is exposed. This DSIP address acts as a discovery portal and initial connection point. Only one CVM can own the DSIP address at a time. If the owner goes offline, the address moves to another CVM, ensuring that the DSIP address is always available. You can configure the DSIP address in the Cluster Details section of Prism.

Figure 4: Configuring iSCSI Data Services IP Address

Once you've established a connection to a target through the iSCSI DSIP address, the system redirects the iSCSI client to the CVM that hosts the disk for the specified target. The Nutanix cluster uses a heuristic to balance target ownership evenly across CVMs in the cluster. All CVMs are potential targets, including new CVMs added during cluster expansion. The following figure shows the connection process, and we provide a more detailed diagram in the appendix.

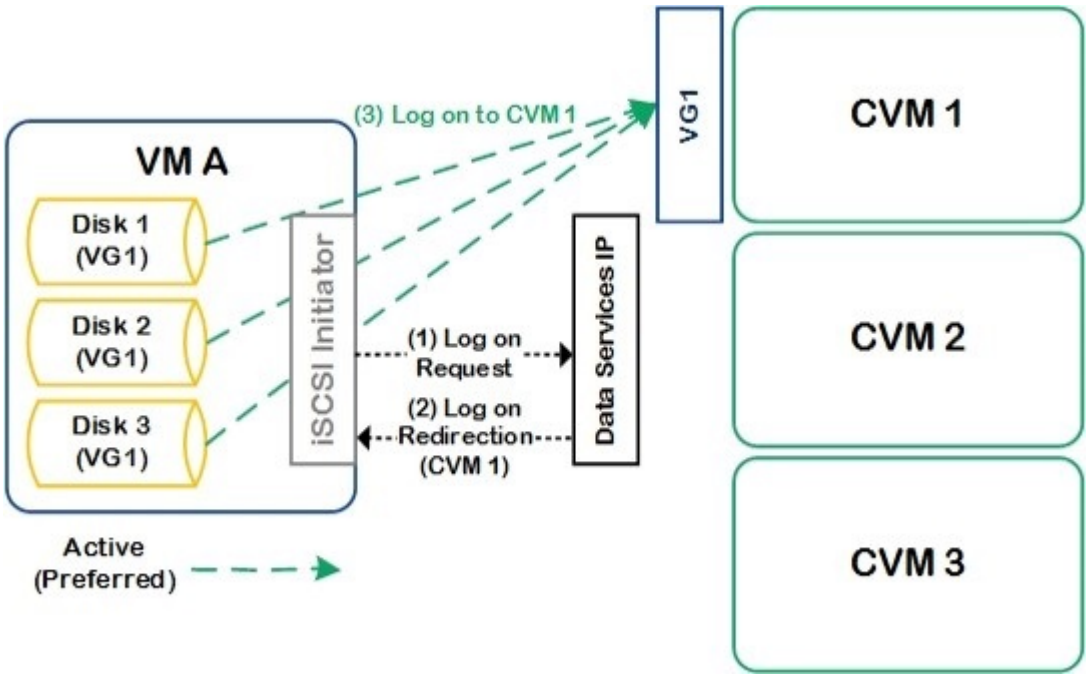

Figure 5: iSCSI Redirection for Initial Connection

The sign-in redirection occurs on a per-target basis. A VG represents one or more virtual targets. Virtual targets enable a single VG to export multiple iSCSI targets that can be redirected to different CVMs.

By default, a VG is configured to have 32 virtual targets. With the default of 32, the number of virtual targets a client sees depends on the number of disks in the VG. The following figure illustrates an example in which 3 disks reside in a VG; therefore, a client sees 3 virtual targets. If a VG has 32 or more disks, then client discovery displays 32 virtual targets. Names for virtual targets start with the VG name and end with a virtual target number. For example, a VG named volumesvg1 with 2 disks could have virtual targets named `volumesvg1-7d64abbe-0e5c-49b6-97e2-449e6db08f54-tgt0` and `volumesvg1-7d64abbe-0e5c-49b6-97e2-449e6db08f54-tgt1`.

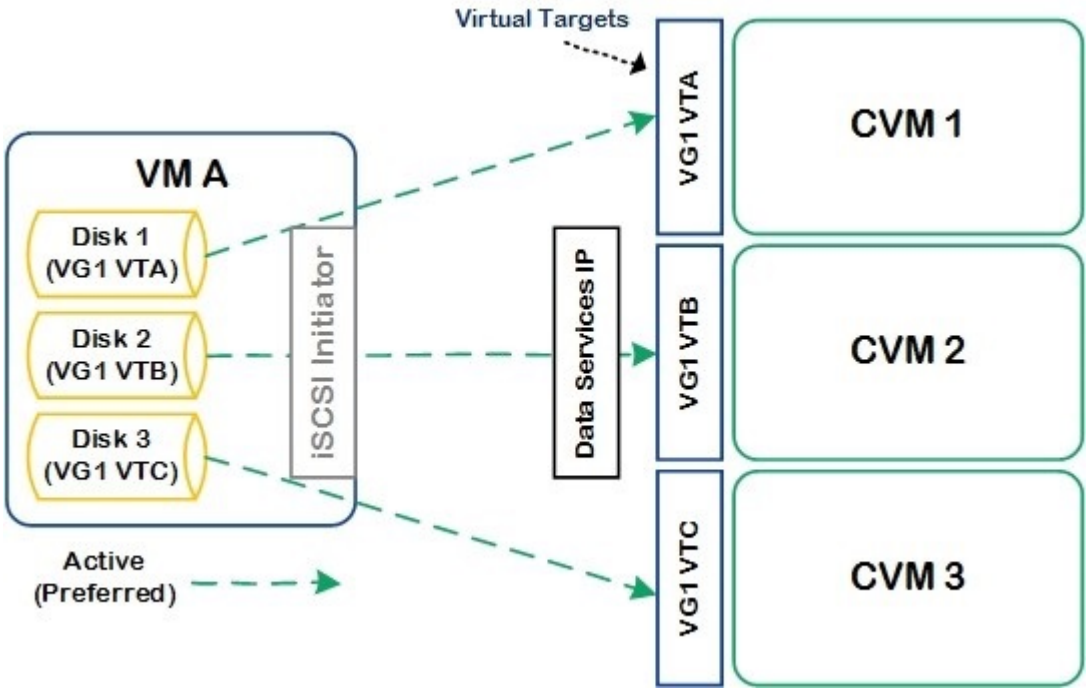

Figure 6: VG Virtual Targets

If an active CVM goes offline because of a failure or planned maintenance, the system disconnects any active sessions against that CVM, which triggers the iSCSI client to sign in again. The new sign-in occurs through the iSCSI DSIP address, which redirects the session to a healthy CVM. The next figure shows this process, and we provide a more detailed diagram in the appendix.

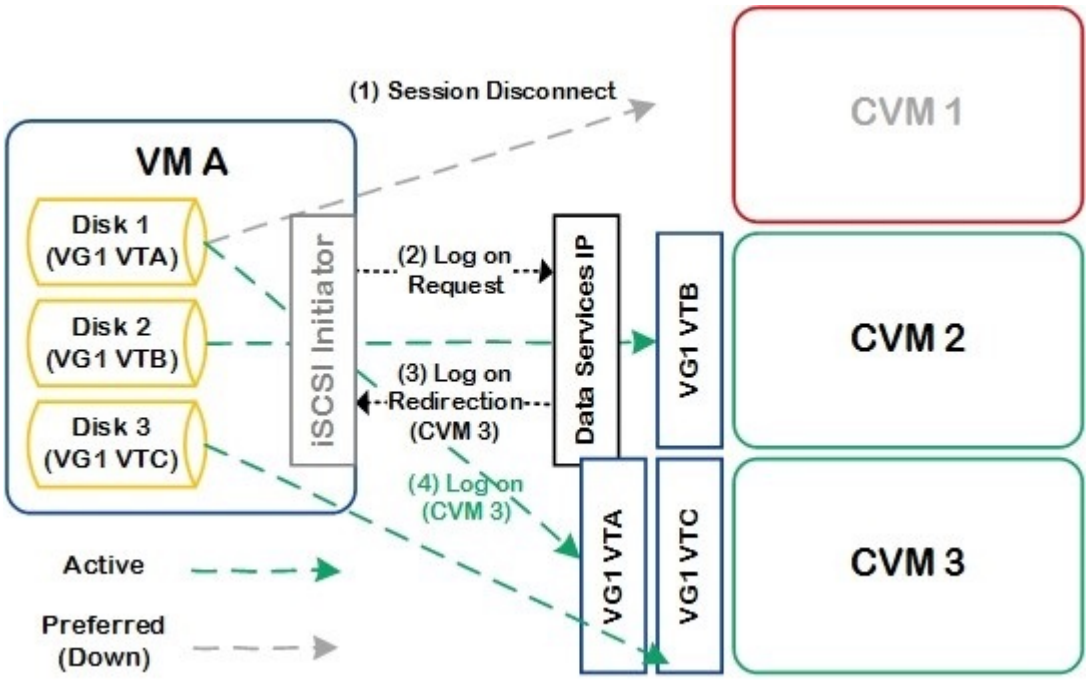

Figure 7: iSCSI Redirection on Failure

Each iSCSI target has a preferred CVM. The preferred CVM is configured automatically as part of the load-balancing process. If a preferred CVM goes offline and then returns to operation, the iSCSI session may fail back. In the case of a failback, the system signs the client out and redirects it to the appropriate CVM.

If multiple clients share an iSCSI target, the system directs each client to the same CVM. The following figure shows a clustered configuration where multiple VMs share disks and are redirected to the same preferred CVM.

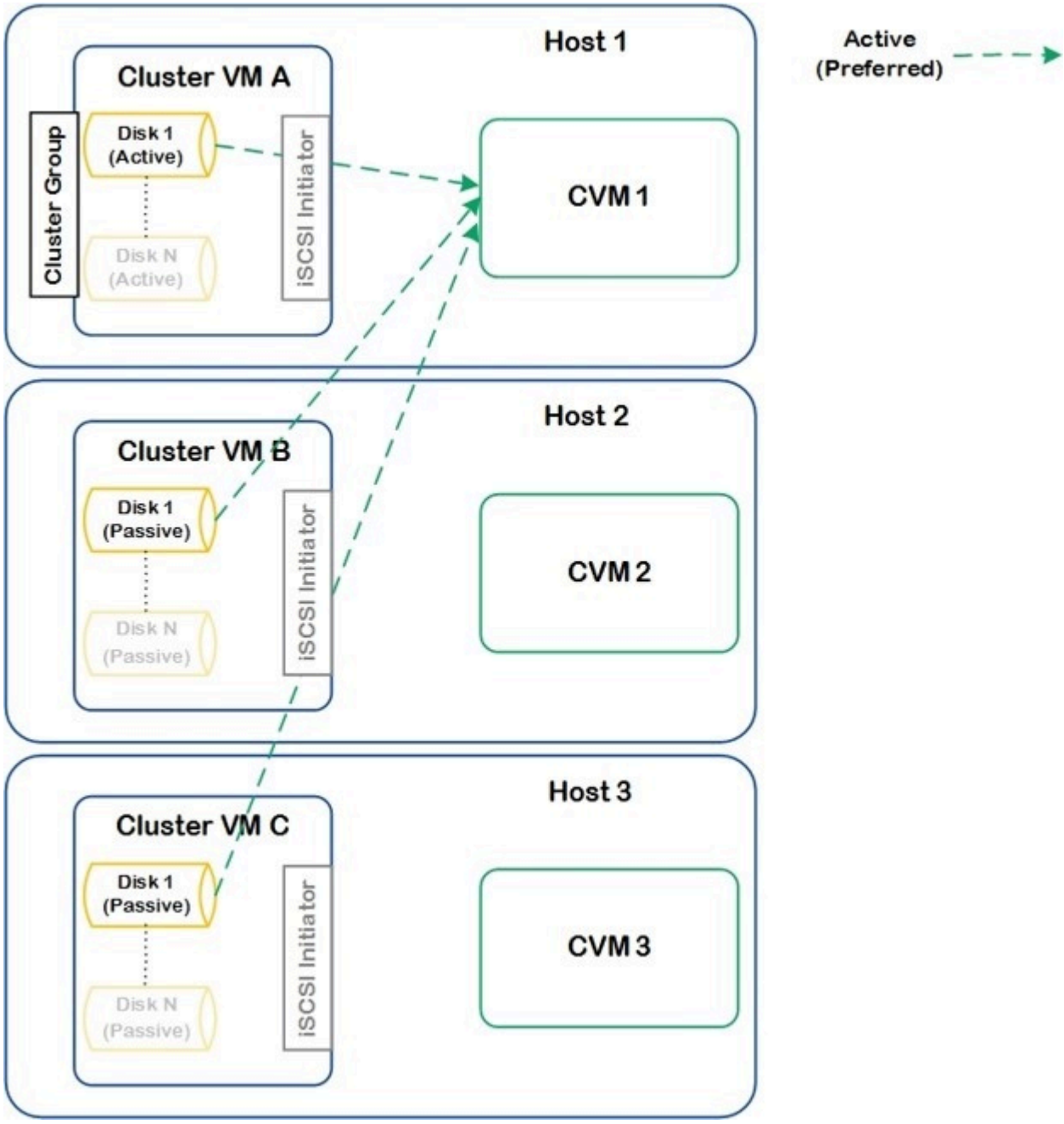

Figure 8: Cluster Redirected to Common CVM

Challenge Handshake Authentication Protocol (CHAP)

CHAP is a peer authentication protocol that allows an iSCSI client and target to authenticate based on a password, also referred to as a secret. CHAP includes both one-way and mutual authentication. With one-way CHAP, a target authenticates a client initiator when it connects. With mutual CHAP, the client and the target authenticate each other based on their respective passwords. Nutanix Volumes supports using both one-way and mutual CHAPs.

For Linux clients running CentOS 8 and later, Nutanix provides support for the newer digest algorithms SHA1 and SHA256 in addition to the default MD5 for CHAP authentication. SHA1 and SHA256 support was introduced in AOS 6.1. Nutanix recommends using these newer CHAP algorithms instead of the default MD5 authentication for a more secure sign-in and iSCSI target discovery, which is important for environments that must comply with FIPS.

5. Acropolis Dynamic Scheduling

The Nutanix cluster uses a heuristic to balance client connectivity (via target ownership) evenly across CVMs in the cluster. Despite this balance, some targets might have higher performance requirements than others and can saturate a CVM's processors. If a given CVM uses more than 85 percent of its CPU for storage traffic (measured by the Nutanix Stargate process), a feature called Acropolis Dynamic Scheduling (ADS), which is enabled by default, can automatically move specific iSCSI sessions to other CVMs in the cluster to help avoid hotspots. ADS works in conjunction with Nutanix Volumes on any hypervisor.

6. Networking for iSCSI

iSCSI is a TCP/IP-based protocol, so TCP/IP performance directly affects storage I/O performance. Nutanix provides networking best practices based on the underlying hypervisor. The base recommendations are described in the VMware vSphere Networking, AHV Networking, and Hyper-V Windows Server 2016 Networking guides.

When you use Nutanix Volumes with the iSCSI DSIP address, the eth0 network of the CVM services iSCSI traffic by default. Also by default, the system uses the eth0 CVM network for Nutanix management and communication between the CVMs in the cluster.

Note: Place hosts that use Nutanix Volumes on the same subnet as the iSCSI DSIP address.

Networking Segmentation

Some organizations require iSCSI traffic to be on its own network segment. You can separate the CVM-to-CVM communication onto a backplane LAN. Using this LAN, operations such as remote writes and remote reads for distributed storage occur over an optional eth2 backplane network. The eth0 network interface continues to service iSCSI traffic.

You can also separate the iSCSI traffic for Nutanix Volumes onto a dedicated virtual network interface on the CVMs. AOS supports having two segmented networks dedicated to Nutanix Volumes. Both AHV and ESXi hypervisors support iSCSI network segmentation. There are various scenarios when you might want two Nutanix Volumes networks (for example, situations where you need to separate production and nonproduction environments). Two Nutanix Volumes networks can also provide faster high availability in the event of CVM failure where the failed CVM happens to be hosting a DSIP address. If you enable MPIO in the client with paths set up to both DSIP addresses, failover between CVMs is faster in the event that a DSIP address owner that also hosts

vDisks from a VG fails, because it doesn't have to wait for the DSIP address to become available on another node.

The virtual network interface can use a shared or dedicated bridge (AHV) or virtual switch (ESXi). The bridge or virtual switch specified has physical NICs configured as uplinks. The following diagram shows an ESXi environment where CVM interface ntnx0 is the dedicated Nutanix Volumes iSCSI network.

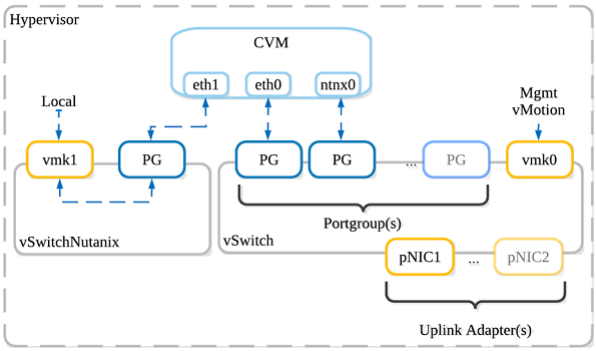

Figure 9: iSCSI Network Segmentation

You configure the dedicated iSCSI network by creating a new internal interface in the Network Configuration menu under the cluster settings.

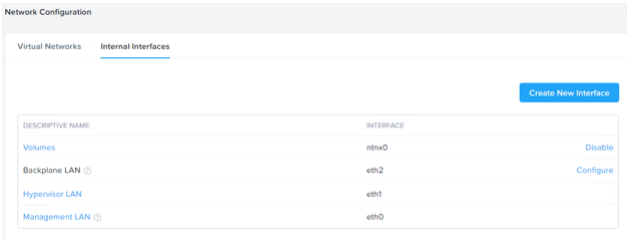

Figure 10: Internal Interfaces

When you create the internal interface, you create an IP address pool that's used to assign IP addresses to the new vNIC on each CVM. You can also specify a virtual IP address that represents the DSIP address. Add at least n + 1 IP addresses, where n is the number of nodes in the cluster.

The new interface workflow creates the iSCSI interface on each CVM, assigns IP addresses from the IP address pool, and associates the vNIC with the specified port group (ESXi) or bridge (AHV). This process runs on one CVM at a time in a rolling fashion, followed by a restart of the CVM. While the CVM is offline,

data resilience in the cluster is reduced. Once this process finishes, the Nutanix Volumes service is configured to use the new network. You can then start to use the new DSIP address as the discovery portal for your hosts.

If you need a second Nutanix Volumes network, repeat the process. The second Nutanix Volumes network must be on a different subnet than the first. The second network can use the same VLAN and bridge or virtual switch as the first or you can assign a separate VLAN and bridge or virtual switch. The second Nutanix Volumes network has its own DSIP address and associated IP address pool. You can find more information in the Nutanix Volumes section of the AOS security guide.

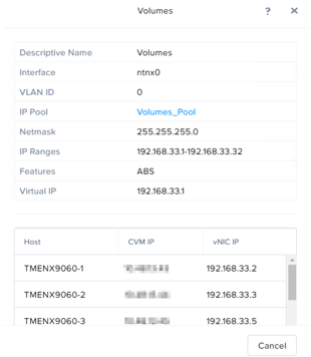

Figure 11: Volumes Network Segmentation

7. Configuring Windows Clients for Connectivity

The following steps provide detailed guidance for configuring iSCSI-based VGs with Nutanix using Windows 2008 or Windows 2012 (R2).

Start the iSCSI Service and Set the Startup Type to Automatic

1. Open the Services control panel (services.msc).
2. Find the Microsoft iSCSI Initiator Service.
3. Start the service.
4. Set the startup type to automatic.

Set Firewall Rules to Allow iSCSI Traffic

1. Open the Firewall control panel (firewall.cpl).
2. Click Allow an app or feature through Windows Firewall.
3. Select the appropriate networks for the iSCSI service.
4. Open ports 3260 and 3205.

Get the IQN from the Virtual Machines

1. Start the iSCSI control panel (iscsicpl.exe).
2. Go to the Configuration tab.
   a. Copy the full string under Initiator Name. For example:
   `iqn.1991-05.com.microsoft:csva.tme.local`

Create the Nutanix Volume Group

You can perform VG configuration from Prism or by using aCLI from a CVM. The following example uses aCLI. You only need to perform these steps from one CVM.

1. Connect to any CVM using SSH.
2. Run the aCLI command.
   a. Use the Tab key to get command syntax and existing objects (like VGs, containers, and so on) while in the aCLI shell.
3. Create a VG.
   a. `<acropolis> vg.create vg_name`
   b. Example: `vg.create mssql1 shared=true`
   c. Specify shared if the disks are shared across hosts.
4. Add one or more disks to the VG.
   a. `<acropolis> vg.disk_create vg_name container=container_name create_size=disk_size`
   b. Example: `vg.disk_create mssql1 container=ctr1 create_size=100G`
5. Attach the initiator IQN to the VG. Repeat for each host participating in the cluster.
   a. `<acropolis> vg.attach_external vg_name initiator_name=initiator_iqn`
   b. Example: `vg.attach_external mssql1 initiator_name=iqn.1991-5.com.microsoft:csva.tme.local`

Create an iSCSI Discovery Portal

The discovery portal allows the Windows host to discover the available iSCSI targets in the Nutanix cluster.

1. Start the iSCSI control panel (iscsicpl.exe).
2. Go to the Discovery tab.
3. Click Discover Portal.
4. Enter the iSCSI DSIP address and click OK.

Connect to Discovered Targets and Persist the Connections

1. Start the iSCSI control panel (iscsicpl.exe).
2. Go to the Targets tab.
3. Select a discovered target in the list.
   a. The target names should contain the name of the Nutanix VG.
   b. Click Refresh if the discovered targets list is empty.
4. Click Connect.
5. In the Connect to Target dialog box, select Add this connection to the list of Favorite Targets....
6. Click Advanced.
7. Under the General tab:
   a. For Local adapter, select Microsoft iSCSI Initiator.
   b. For Initiator IP, select the IP address associated with an iSCSI NIC.
   c. For Target portal IP, select the iSCSI DSIP address.
8. Click OK to exit from the General tab.
9. Click OK to exit the Connect to Target screen.

The target should now appear as Connected. Repeat the steps in this section for any additional targets. You can run a scan from diskmgmt.msc or diskpart.exe to discover the disks.

Configure Red Hat Enterprise Linux (RHEL) Clients for Connectivity

1. Install the iSCSI initiator package.
   a. `sudo yum install iscsi-initiator-utils`
2. Check to ensure that the iSCSI daemon is running on the host.
   a. RHEL 6: `sudo /etc/init.d/iscsi status`
   b. RHEL 7: `sudo service iscsid status`
   c. Replace `status` with `start` if the status isn't returned as running.
3. Get the iSCSI initiator name from the host.
   a. `cat /etc/iscsi/initiatorname.iscsi`

4. Create a Nutanix VG.
   a. Connect to any CVM using SSH.
   b. Run the aCLI command.
   c. Use the Tab key to get command syntax and existing objects (like VGs, containers, and so on) while in the aCLI shell.
5. Create a VG.
   a. `<acropolis> vg.create vg_name`
   b. Example: `vg.create rhel1 shared=true`
   c. Specify `shared` if the disks are shared across hosts.
6. Add one or more disks to the VG.
   a. `<acropolis> vg.disk_create vg_name container=container_name create_size=disk_size`
   b. Example: `vg.disk_create rhel1 container=ctr1 create_size=100G`
7. Attach the initiator IQN to the VG. Do this step for each host participating in the cluster.
   a. `<acropolis> vg.attach_external vg_name initiator_name=initiator_iqn`
   b. Example: `vg.attach_external rhel1 initiator_name=iqn.1994-05.com.redhat:d7965a15d6e`
8. Discover the available targets on the host.
   a. `iscsiadm -m discovery -t st -p <external_data_services_ip_address>`, where `-p` is the iSCSI DSIP address.
   b. Take note of the listed targets.
9. Connect to the discovered targets on the host.
   a. `iscsiadm -m node --login`
10. Verify that the disk is attached.
    a. `ls --l /dev/disk/by-path`

The output shows the target name and LUN device mapping.

```
lrwxrwxrwx. 1 root root 9 May 2 20:18 ip-10.4.57.40:3260-iscsi-
iqn.2010-06.com.nutanix:rhelvg1-9d59957c-e1c7-4094-8049-bcd4bff0bce6-tgt0-lun-0 -
> ../../sdb
```

RHEL automatically maintains the mapping from the WWID-based device name to a current `/dev/sd` name on the system. Applications can use the `/dev/disk/by-`

`id/` name to have a persistent device identifier for the data on the disk based on WWID. Applications can also use `/dev/disk/by-uuid/` to have a persistent file system identifier for the data.

8. Best Practices

- Use the DSIP address method for external host connectivity to VGs. Don't use the cluster virtual IP.
- For backward compatibility, you can upgrade existing environments nondisruptively and continue to use MPIO for load balancing and path resilience.
- For security, use at least one-way CHAP.
- Implement either the SHA1 or SHA256 CHAP algorithm for increased security.
- Leave ADS enabled. (Enabled is the default setting.)
- Use multiple disks rather than a single large disk for an application. Consider using a minimum of one disk per Nutanix node to distribute the workload across all nodes in a cluster. Multiple disks per Nutanix node may also improve an application's performance.
- For performance-intensive environments, use between four and eight disks per CVM for a given workload.
- Populate VG allowlists with IQNs rather than IP addresses.
- Use dedicated network interfaces for iSCSI traffic in your hosts.
- Place hosts that use Nutanix Volumes on the same subnet as the iSCSI DSIP address.
- Use a single subnet (broadcast domain) for iSCSI traffic. Avoid routing between the client initiators and CVM targets.
- With receive-side scaling (RSS) enabled, the system can use multiple CPU cores to process network traffic, preventing a single CPU core from becoming a bottleneck. Enabling RSS in hosts can be beneficial for heavy iSCSI workloads. For VMs running in ESXi environments, RSS requires VMXNET3

vNICs. For Hyper-V environments, enable VMQ to take full advantage of Virtual RSS.

- Keep the Nutanix CVM default ethernet MTU of 1,500 bytes for all the network interfaces, as it delivers excellent performance and stability.
  - Nutanix doesn't support configuring the MTU on a CVM's network interfaces to higher values, so it doesn't make sense to enable jumbo frames on the user VM's iSCSI NIC.
- For Linux environments, ensure that the SCSI device timeout is 60 seconds. See Red Hat's documentation for an example of checking and modifying this setting.
- For Linux environments, use persistent file system or device naming identifiers to ensure that applications reference storage devices correctly across system reboots. See Red Hat's documentation on persistent naming attributes for more details.
- For Windows environments, set the TcpAckFrequency value to 1 for the NIC connecting to the Nutanix Volumes iSCSI targets, so that every packet is acknowledged immediately. See Microsoft Support's documentation for more details.
- When you use the iSCSI DSIP address:
  - In general, use the default iSCSI client timer settings except when you use MPIO. Tests with the default iSCSI timeout and timer settings have shown path failover to take 15 to 20 seconds. These results are well within the Windows default disk timeout, which is 60 seconds.
  - In physical server environments that require NIC redundancy, you can use either NIC teaming (also called bonding) or MPIO.
    - When you use MPIO for NIC redundancy, use an active-active load balance policy such as round robin.
  - When you use MPIO, set the Windows iSCSI LinkDownTime setting to 60 seconds.

Client Tuning Recommendations

Not all environments require tuning, but there are additional iSCSI settings that can benefit performance in some environments.

- For large block sequential workloads with I/O sizes of 1 MB or larger, it's beneficial to increase the iSCSI MaxTransferLength from 256 KB to 1 MB.
  - Windows: Details on the MaxTransferLength setting are available in the Microsoft iSCSI Software Initiator and iSNS Server timers quick reference guide.
  - Linux: Settings in the `/etc/iscsi/iscsid.conf file; node.conn[0].iscsi.MaxRecvDataSegmentLength`
- For workloads with large storage queue depth requirements, it can be beneficial to increase the initiator and device iSCSI client queue depths.
  - Windows: Details on the MaxPendingRequests setting are available in the Microsoft iSCSI Software Initiator and iSNS Server timers quick reference guide.
  - Linux: Settings in the `/etc/iscsi/iscsid.conf file; Initiator limit: node.session.cmds_max (default: 128); Device limit: node.session.queue_depth (default: 32)`

The default Nutanix iSCSI target values are as follows:

- `Iscsi_max_recv_data_segment_length`
  - Maximum number of bytes allowed in a single PDU data segment.
  - Default: 1048576
- `Iscsi_desired_first_burst_length`
  - Maximum amount of unsolicited data in bytes an iSCSI initiator may send to the target for a single SCSI command.
  - Default: 16777216

- `Iscsi_desired_max_burst_length`
  - Desired value for MaxBurstLength if negotiated.
  - Default: 16777216
- `Iscsi_session_queue_size`
  - Maximum number of outstanding requests an initiator can have on a given iSCSI session.
  - Default: 512

Linux Client Tuning Example

Configure the following iSCSI settings on the guest OS in the `/etc/iscsi/iscsid.conf` file and restart the iscsid process.

```
node.session.timeo.replacement_timeout = 120
node.conn[0].timeo.noop_out_interval = 5
node.conn[0].timeo.noop_out_timeout = 10
node.session.cmds_max = 2048
node.session.queue_depth = 1024
node.session.iscsi.ImmediateData = Yes
node.session.iscsi.FirstBurstLength = 1048576
node.session.iscsi.MaxBurstLength = 16776192
node.conn[0].iscsi.MaxRecvDataSegmentLength = 1048576
discovery.sendtargets.iscsi.MaxRecvDataSegmentLength = 1048576
```


9. Thin Provisioning

Thin provisioning provides efficient storage space usage because the system presents storage devices without allocating space until data is written. Nutanix storage pools and containers are thin-provisioned by default as a core feature of AOS storage. Disks created in VGs are thin-provisioned. Over time, as the system writes data, it allocates storage in the disks. When you delete data from a disk, AOS storage reclaims the space consumed by the deleted file in the background.

SCSI UNMAP

VGs support the SCSI UNMAP command as defined in the SCSI T10 specification. By issuing the SCSI UNMAP command, a host application or OS specifies that a given range of storage is no longer in use and can be reclaimed. This capability is useful when you delete data from a disk that Nutanix Volumes presents. Windows and Linux operating systems provide native support for issuing UNMAP commands to storage.

When the guests send UNMAP commands to the Nutanix storage layer, Prism accurately reflects the amount of available storage as AOS storage background scans complete. If the guest doesn't send UNMAP commands, freed space isn't made available for other guests and doesn't display as available in Prism.

Windows

Windows Server 2012 or later can issue industry-standard UNMAP commands to tell the Nutanix cluster to free storage associated with unused space. Windows supports reclaim operations against both NTFS and ReFS formatted volumes; these operations are enabled by default.

With Windows Server 2012 or later, UNMAP commands are issued under the following conditions:

- When you delete files from a file system, Windows automatically issues reclaim commands for the area of the file system you freed.

- When you format a volume residing on a thin-provisioned drive with the quick option, Windows reclaims the entire volume.
- When a regularly scheduled operation selects the Optimize option for a volume or when you manually select this option, either from the Optimize Drives console or when you use the optimize-volume PowerShell command with the Retrim option, Windows issues an UNMAP command.

To check the current Windows configuration, which is a global setting for the host, you can use the `fsutil` command:

```
fsutil behavior query DisableDeleteNotify
DisableDeleteNotify=0 <---- enabled (default)
DisableDeleteNotify=1 <---- disabled
```

To disable the feature:

```
fsutil behavior set DisableDeleteNotify 1
```

To enable the feature:

```
fsutil behavior set DisableDeleteNotify 0
```

Nutanix recommends using the default Windows configuration, in which the guest file system reports freed space back to the storage layer as you delete files.

Windows Disk Formatting with UNMAP

Windows guests send UNMAP commands by default during disk format operations, which can increase the amount of time required to complete the format. Set DisableDeleteNotify to 1, temporarily disabling the feature during format to finish the format operation faster. Enable the feature after completing the format operation.

Linux

Most Linux guests don't send UNMAP operations by default for mounted disks. Administrators can enable UNMAP operations in the Linux guest, either by mounting the disk with the discard option or by performing fstrim operations. Nutanix recommends following RHEL documentation and using a periodic fstrim operation instead of the discard mount option if you need guest free space reporting in Linux.

To check the current Linux configuration, which is a setting for each disk, you can view the value of `discard_zeroes_data`:

```
cat /sys/block/<device>/queue/discard_zeroes_data
0 => TRIM (UNMAP) disabled
1 => TRIM (UNMAP) enabled
```

Some Linux distributions, such as Ubuntu and SUSE, perform periodic fstrim operations for you in a weekly cron task. Nutanix recommends using the following configuration in Linux guests. If possible, randomize the start time among guests.

```
cat /etc/cron.weekly/fstrim
/sbin/fstrim --all || true
```

Linux Disk Formatting with UNMAP

Linux file systems such as Ext4 send UNMAP commands during the format operation, which can increase the amount of time required to finish formatting. To finish the format operation faster, use the nodiscard option:

```
mkfs.ext4 -E nodiscard /dev/sdXY
```


10. Volume Group Data Protection

Nutanix snapshots can use asynchronous protection domains to protect VGs. You can add VGs to protection domains and replicate them the same way you replicate VMs.

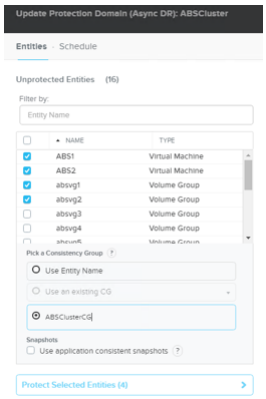

Figure 12: Protecting Volume Groups

When selected, all disks in a VG are automatically included for replication. You can place VMs and VGs in the same consistency group, which we recommend for dependent applications. For example, you can put the VMs in a Windows failover cluster in the same consistency group as the VG-based disks the clustered application uses.

The same protection domain workflows used with VMs can also restore, clone, and recover protected VGs between Nutanix clusters. External bare-metal attachments require a few additional steps to fully restore the VGs following a protection domain recovery workflow:

- Following a local VG restore operation, update the VG with the initiator IQNs of the externally attached hosts.

- Following VG activation or a migration of a VG between clusters:
  - Update the VG with the initiator IQN of the externally attached hosts.
  - Update host iSCSI client discovery and connection information to reflect the iSCSI data services IP address of the new target cluster.

For environments using in-guest iSCSI for VMs running on the Nutanix cluster, AOS provides a functionality that helps protect and restore VMs and VGs. You can select automatic protection for related entities when you work with VMs that have the Nutanix Guest Tools (NGT) software installed and enabled. As an example, when you create a protection domain and select a VM, the protection domain automatically includes any VGs being used by that VM as well.

With Nutanix, restoring, cloning, or migrating VMs with related VGs triggers a workflow that assigns a new IQN for the guest. The Nutanix guest agent automatically reconfigures the guest's iSCSI client to connect to the appropriate data services IP address using the new IQN. Note that iSCSI attachments aren't automatically updated when you segment the iSCSI network.

Also regarding NGT-enabled VMs using in-guest iSCSI, you should use the IQN rather than the IP address when populating VG allowlists. The allowlist is responsible for maintaining the association between a VM and a VG, and accepts both IQNs and IP addresses. However, when you restore a VG from a protection domain snapshot, the allowlist retains only IQN entries and discards IP address entries. The VG's association with VMs identified by IP address is therefore lost and you must repeat the initial setup process. For this reason, Nutanix recommends populating VG allowlists with IQNs rather than IP addresses.

> Note: Nutanix doesn't currently support VGs for replication with Metro Availability or synchronous replication.

11. Conclusion

Whether you have applications that require shared storage access or environments with separate storage and compute needs, Nutanix Volumes simplifies deployment and highlights the dynamic scale-out, extreme performance, and high availability of the Nutanix platform. Nutanix Volumes seamlessly manages failure events and automatically load-balances iSCSI clients to take advantage of all cluster resources. The same upgrade, snapshot, and asynchronous replication workflows that customers use with VMs today work consistently with VGs as well. By enabling VM, file, object, and block services, Nutanix offers a single platform to consolidate workloads and ease administration, reducing risk and empowering you to simplify your infrastructure.

12. Appendix

Exclude Nodes for Nutanix Volumes Placement

You can manually select which CVMs in a cluster are available for Nutanix Volumes to use. By default, all nodes—including storage nodes—are automatically available to host iSCSI sessions. We recommend keeping Nutanix Volumes at this default placement setting.

However, if you must exclude specific CVMs from placement (for example, if you need to isolate workloads), you can change the setting from the iSCSI 2009 page: `http://<cvm_ip>:2009/iscsi/lb`. This page also directs you to the CVM you can manage additional changes from.

Sign-In Redirection

Initial Connection

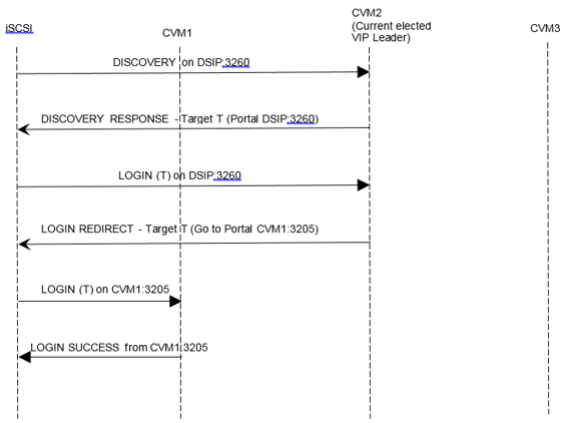

Figure 13: Initial iSCSI Client Connection

Initial Connection and CVM Offline Workflow

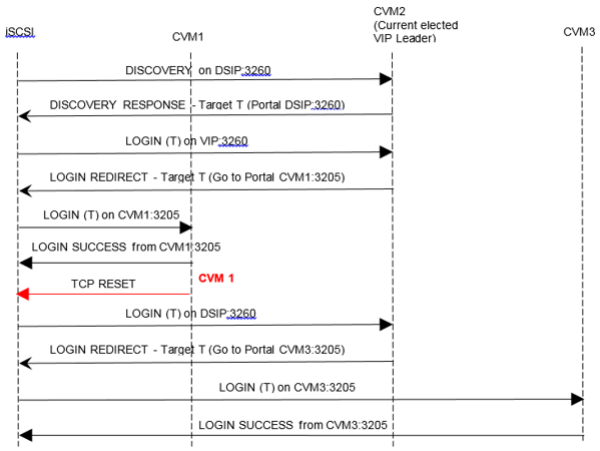

Figure 14: Initial Connection and Target Failover

iSCSI and Multipath I/O (MPIO)

In releases prior to AOS 4.7, external host connectivity to VGs depended on a combination of software iSCSI initiators and MPIO. MPIO controls the active, preferred, and standby paths for disks. MPIO also controls path failover if a CVM becomes unavailable because of planned maintenance, host failure, or network failure. The following figure provides an example of a VM accessing three disks using CVM 1 as the active and preferred path and CVM 2 and CVM 3 as standby paths.

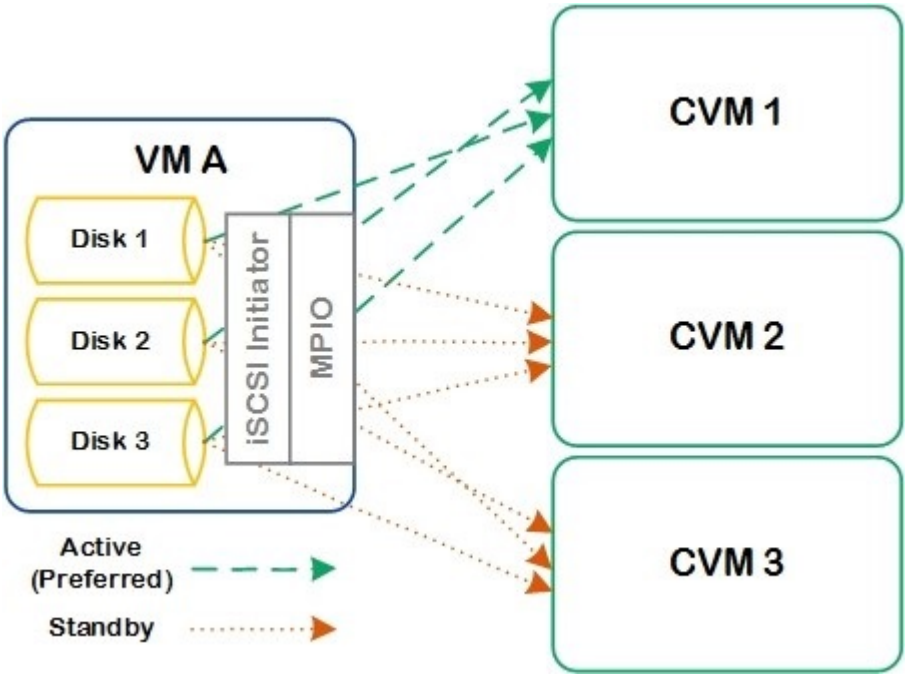

Figure 15: Legacy MPIO Connectivity

A common configuration in virtualized environments is for the VM to access its vDisks using the local CVM, which is the CVM running on the same server as the VM. This structure runs iSCSI network traffic over the virtual switch in the host so it doesn't need to use the physical network. The following figure provides an example of such a configuration.

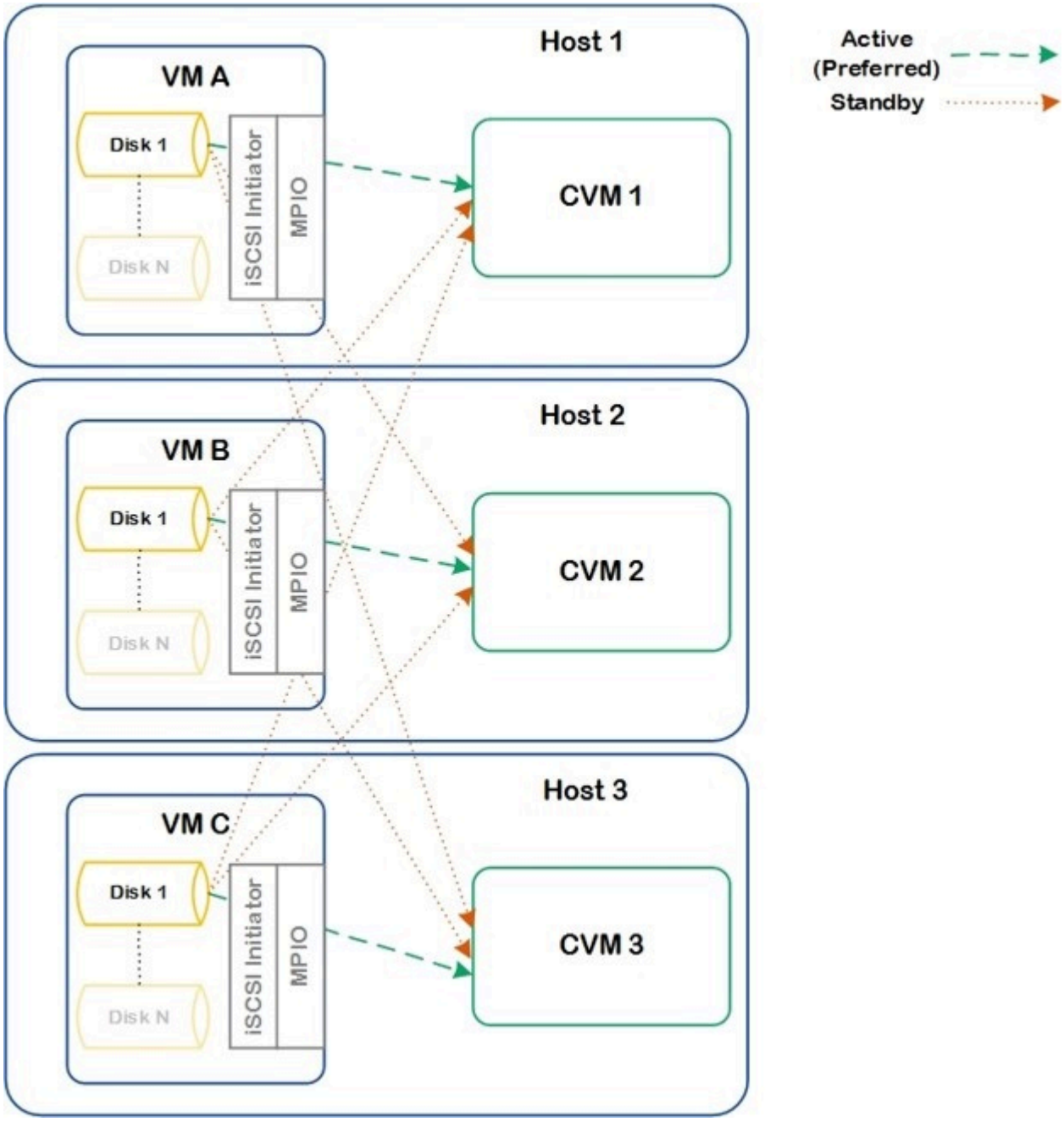

Figure 16: Local CVM Access

If the active CVM becomes unavailable, an MPIO path failover to a standby path occurs.

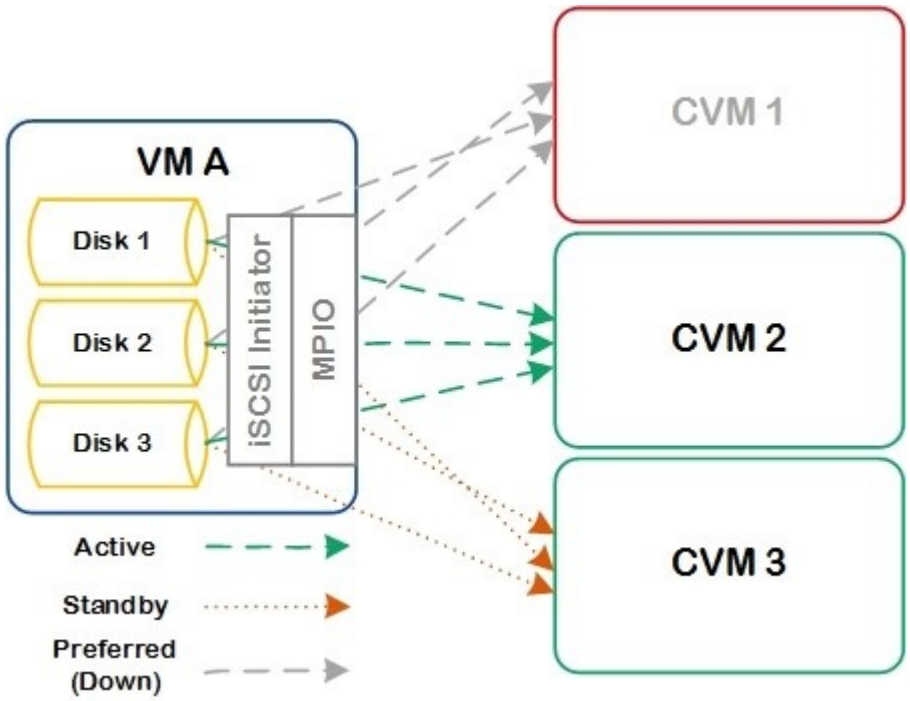

Figure 17: MPIO Path Failover

In this example, because CVM 1 is marked as a preferred path through MPIO, it automatically returns to being the active path upon its recovery.

You can also balance the paths for disks across CVMs in the cluster.

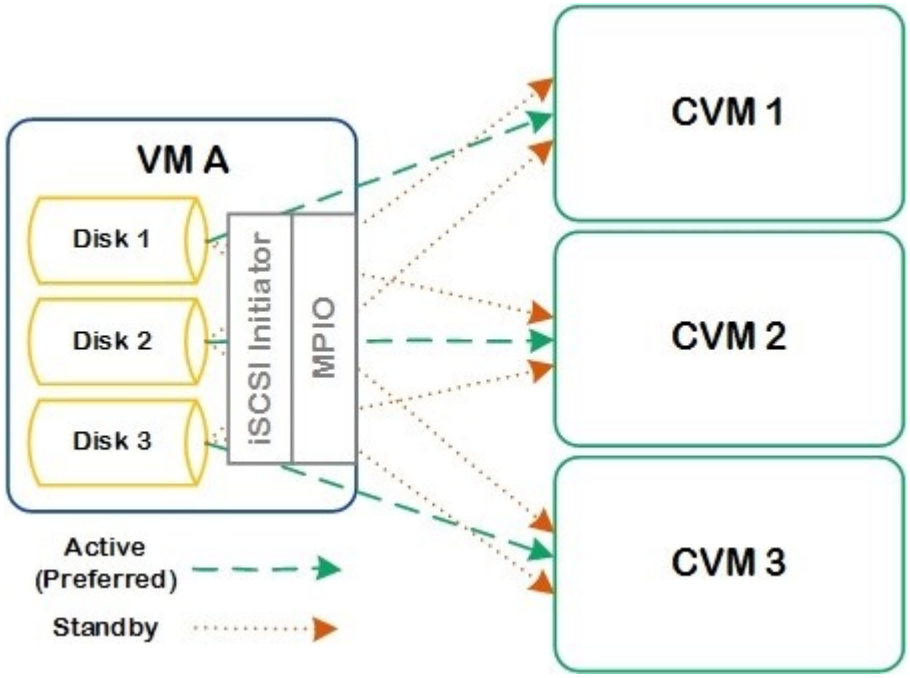

Figure 18: Distributed Disk Access

This configuration distributes the workload across more CVMs, which can improve performance. The tradeoff for such a design is that iSCSI traffic traverses the physical network.

Configuring Volume Groups for Windows with MPIO

The following steps provide detailed guidance for configuring iSCSI-based VGs with Nutanix using Windows 2008 or Windows 2012 GUI tools. Some steps require the CLI, and we include those commands. Perform the Windows-specific steps on each VM participating in the cluster.

Start the iSCSI Service and Set the Startup Type to Automatic with MPIO

1. Open the Services control panel (services.msc).
2. Find the Microsoft iSCSI Initiator Service.
3. Start the service.
4. Set the startup type to automatic.

Set Firewall Rules to Allow iSCSI Traffic with MPIO

1. Open the Firewall control panel (firewall.cpl).
2. Select Allow an app or feature through Windows Firewall.
3. Select the appropriate networks for the iSCSI Service.

Install MPIO

1. Open Server Manager (servermanager.exe).
   a. Click Add Roles and Features (Add Features if using Windows 2008).
   b. Follow the steps and select the multipath I/O feature to install.

Configure MPIO to Automatically Claim iSCSI Devices

1. Open the MPIO control panel (mpiocpl.exe).
2. Go to the Discover Multipaths tab.
3. Select the Add support for iSCSI devices box and click Add.
4. Restart the host if prompted.

Configure MPIO Global Load Balance Policy

Windows 2008 CLI:

- For failover-only: `mpclaim -l -m 1`

Windows 2012 R2 PowerShell:

- For failover-only: `Set-MSDSMGlobalDefaultLoadBalancePolicy -Policy FOO`

Get the IQN from the VMs with MPIO

1. Start the iSCSI control panel (iscsicpl.exe).
2. Go to the Configuration tab.
3. Copy the full string under Initiator Name.
   a. For example: `iqn.1991-05.com.microsoft:csva.tme.local`

Create the Nutanix Volume Group with MPIO

You can use Nutanix Prism to configure the VG or use aCLI from a CVM. You only need to perform these steps from one CVM.

1. Connect to any CVM using SSH.

2. Run the aCLI command.
   a. Use the Tab key to get command syntax and existing objects (like VGs, containers, and so on) while in the aCLI shell.
3. Create a VG.
   a. `<acropolis> vg.create vg_name`
   b. Example: `vg.create mssql1 shared=true`
   c. Specify shared if the disks are shared across hosts.
4. Add one or more disks to the VG.
   a. `<acropolis> vg.disk_create vg_name container=container_name create_size=disk_size`
   b. Example: `vg.disk_create mssql1 container=ctr1 create_size=100G`
5. Attach the initiator IQN to the VG.
   a. `<acropolis> vg.attach_external vg_name_initiator_iqn`
   b. Example: `vg.attach_external mssql1 iqn.1991-5.com.microsoft:csva.tme.local`

Create an iSCSI Discovery Portal with MPIO

The discovery portal allows the Windows host to discover the available iSCSI targets in the Nutanix cluster.

GUI: iSCSI control panel (iscsicpl.exe)

1. Start the iSCSI control panel.
2. Go to the Discovery tab.
3. Select Discover Portal.
4. Enter the IP address of one Nutanix CVM and click OK.

Connect to Discovered Targets and Persist the Connections with MPIO

GUI: iSCSI control panel (iscsicpl.exe)

1. Start the iSCSI control panel.
2. Go to the Targets tab.
3. Select the discovered target in the list.
   a. The target name should reflect the name of the Nutanix VG.
   b. Click Refresh if the discovered targets list is empty.
4. Click Connect.

5. On the Connect to Target dialog box, select both options:
   a. Add this connection to the list of Favorite Targets....
   b. Enable multipath.
6. Click Advanced.
7. Under the General tab:
   a. For Local adapter, select Microsoft iSCSI Initiator.
   b. For Initiator IP, select the IP address associated with an iSCSI NIC.
   c. For Target portal IP, select the desired CVM IP address.
   d. Use the intended active path for this first connection.
8. Click OK to exit from the General tab.
9. Click OK to exit the Connect to Target screen.

The target should now show as Connected. If this connection is the first to a given target, you can run a scan from diskmgmt.msc or diskpart.exe to discover the disks and establish the active path. Repeat this process to create additional connections between the iSCSI NICs and CVMs intended as standby connections.

Setting Specific Targets as Both Active and Preferred (Failover-Only)

If the active paths weren't set as expected while connecting to the intended targets, you can modify them using the following steps.

To correlate the Path ID with the CVM IP address:

1. Start the iSCSI control panel (iscsicpl.exe).
2. Go to the Targets tab.
3. Select a target and choose Devices.
4. From the Devices menu, select a disk and choose MPIO....
5. From the MPIO menu, note the Path ID.
6. Select a device path and choose Details.
7. The CVM IP shows under the Target Portal connection.

To set the active path:

1. From the MPIO menu, select a device path and choose Edit.
2. Change the Path type to Active.

To set the active path and preferred path:

1. Start the Disk Management GUI (diskmgmt.msc).
2. Right-click the appropriate disk and select properties.
3. Select the MPIO tab.
4. Select a device path and choose Edit.
5. Choose the path state and select preferred for the desired active or optimized path.

Linux multipath.conf Example

```
devices {
 device {
 vendor "NUTANIX"
 product "Server"
 path_grouping_policy multibus
 path_selector "round-robin 0"
 features "1 queue_if_no_path"
 path_checker tur
 rr_min_io_rq 20
 rr_weight priorities
 failback immediate
 }
}
multipaths {
 multipath {
 wwid 1NUTANIX_NFS_3_0_264_1b0f8465_7e6d_4b7f_a517_29b7ec986216
 alias <name>
 }
}
```


About Nutanix

Nutanix is a global leader in cloud software and a pioneer in hyperconverged infrastructure solutions, making clouds invisible and freeing customers to focus on their business outcomes. Organizations around the world use Nutanix software to leverage a single platform to manage any app at any location for their hybrid multicloud environments. Learn more at www.nutanix.com or follow us on Twitter @nutanix.

List of Figures