



AHV 6.8

# AHV Administration Guide

May 21, 2024

# Contents

AHV Overview.....	6
Storage Overview.....	6
AHV Turbo.....	7
Acropolis Dynamic Scheduling in AHV.....	8
Disabling Acropolis Dynamic Scheduling.....	11
Enabling Acropolis Dynamic Scheduling.....	11
Virtualization Management Web Console Interface.....	11
Finding the AHV Version on Prism Element.....	12
Finding the AHV Version on Prism Central.....	12
AHV Cluster Power Outage Handling.....	13
Serial Console Redirection to a Telnet Port.....	13
 Node Management.....	15
Nonconfigurable AHV Components.....	15
Nutanix Software.....	15
AHV Settings.....	15
Controller VM Access.....	16
Admin User Access to Controller VM.....	17
Nutanix User Access to Controller VM.....	18
Controller VM Password Complexity Requirements.....	20
AHV Host Access.....	20
Initial Configuration.....	21
Accessing the AHV Host Using the Admin Account.....	22
Changing the Root User Password.....	23
Changing Nutanix User Password.....	23
AHV Host Password Complexity Requirements.....	24
Verifying the Cluster Health.....	24
Node Maintenance Mode.....	26
Putting a Node into Maintenance Mode using Web Console.....	27
Viewing a Node that is in Maintenance Mode.....	29
Exiting a Node from the Maintenance Mode using Web Console.....	29
Guest VM Status when Node is in Maintenance Mode.....	30
Putting a Node into Maintenance Mode using CLI.....	31
Exiting a Node from the Maintenance Mode Using CLI.....	33
Shutting Down a Node in a Cluster (AHV).....	34
Starting a Node in a Cluster (AHV).....	35
Rebooting an AHV Node in a Nutanix Cluster.....	36
Shutting Down an AHV Cluster.....	36
Changing CVM Memory Configuration (AHV).....	40
Renaming an AHV Host.....	40
Changing the Name of the CVM Displayed in the Prism Web Console.....	41
Adding a Never-Schedulable Node (AHV Only).....	42
AHV Compute-Only Node Configuration.....	43
AHV Compute-only with AHV HCI Nodes.....	43
AHV Compute-only with AHV Storage-only Nodes.....	45
Deployment of an AHV CO Node.....	47
Adding an AHV Compute-only Node to an AHV Cluster.....	48

Host Network Management.....	51
Network Types.....	51
Prerequisites for Configuring Networking.....	53
AHV Networking Recommendations.....	53
IP Address Management.....	60
Traffic Marking for Quality of Service.....	61
Layer 2 Network Management.....	64
About Virtual Switch.....	64
Virtual Switch Requirements.....	73
Virtual Switch Limitations.....	74
Virtual Switch Management.....	75
Enabling LACP and LAG (AHV Only).....	84
VLAN Configuration.....	87
IGMP Snooping.....	90
Traffic Mirroring on AHV Hosts.....	92
MAC Address Prefix.....	104
Adding a MAC Address Prefix.....	105
Removing the MAC Address Prefix.....	106
Enabling RSS Virtio-Net Multi-Queue by increasing the Number of VNIC Queues.....	106
Changing the IP Address of an AHV Host.....	109
 Virtual Machine Management.....	 113
Supported Guest VM Types for AHV.....	113
Creating a VM (AHV).....	113
Managing a VM (AHV).....	119
Limitation for vNIC Hot-Unplugging.....	125
VM Migration Specifications.....	127
Live Migration Cases.....	127
Live Migration Restrictions.....	127
Checking Live Migration Status of a VM.....	128
Verifying Affinity Policy Association.....	129
Defining Behaviour for Non-migratable VMs (GPU/CPU/PCI/Host Affinity configured VMs).....	131
Multichannel Support for AHV Live Migration Performance.....	131
Adaptive CPU Management during Live Migration.....	133
On-Demand Cross-Cluster Live Migration.....	134
On-Demand CCLM Requirements.....	135
On-Demand CCLM Limitations.....	137
On-Demand CCLM Best Practices.....	138
Performing On-Demand CCLM.....	138
Virtual Machine Snapshots.....	139
Windows VM Provisioning.....	140
Nutanix VirtIO for Windows.....	140
Installing Windows on a VM.....	151
Windows Defender Credential Guard Support in AHV.....	154
Windows Subsystem for Linux (WSL2) Support on AHV.....	160
Affinity Policies for AHV.....	162
Affinity Policies Defined in Prism Element.....	163
Affinity Policies Defined in Prism Central.....	165
Non-Migratable Hosts.....	176
Affinity Policies Specifications.....	177
Performing Power Operations on VMs by Using Nutanix Guest Tools (aCLI).....	180
UEFI Support for VM.....	181

Creating UEFI VMs by Using aCLI.....	182
Getting Familiar with UEFI Firmware Menu.....	183
Secure Boot Support for VMs.....	187
Secure Boot Considerations.....	187
Creating/Updating a VM with Secure Boot Enabled.....	187
Securing AHV VMs with Virtual Trusted Platform Module (aCLI).....	188
Virtual Machine Network Management.....	190
Virtual Machine Memory and CPU Hot-Plug Configurations.....	190
Hot-Plugging the Memory and CPUs on Virtual Machines (AHV).....	191
Virtual Machine Memory Management (vNUMA).....	191
Enabling and Disabling vNUMA on Virtual Machines.....	192
GPU and vGPU Support.....	194
Supported GPUs.....	194
GPU Pass-Through for Guest VMs.....	194
NVIDIA GRID Virtual GPU Support on AHV.....	195
PXE Configuration for AHV VMs.....	205
Configuring the PXE Environment for AHV VMs.....	206
Configuring a VM to Boot over a Network.....	207
Uploading Files to DSF for Microsoft Windows Users.....	208
Enabling Load Balancing of vDisks in a Volume Group.....	209
VM High Availability in Acropolis.....	210
Enabling High Availability for the Cluster.....	211
Viewing list of restarted VMs after an HA event.....	212
Live vDisk Migration Across Storage Containers.....	214
Migrating a vDisk to Another Container.....	215
Memory Overcommit.....	216
Deployment Workflow.....	217
Requirements for Memory Overcommit.....	219
Limitations of Memory Overcommit.....	220
Total Memory Allocation Mechanism on Linux and Windows VM.....	221
Memory Overcommit Management.....	221
OVA.....	227
OVA Restrictions.....	228
VM Generation UUID and BIOS UUID Support in AHV.....	228
Checking VM Generation UUID and BIOS UUID of a Guest VM.....	229
Automatic Cluster Selection for VM Placement.....	230
Supported Configuration Workflows for Automatic Cluster Selection.....	230
VM Policies with Automatic Cluster Selection.....	230
Automatic Cluster Selection with Self-Service Projects.....	231
Requirements for Automatic Cluster Selection.....	231
Limitations of Automatic Cluster Selection.....	231
Sample Scenarios for Automatic Cluster Selection.....	232
Advanced Processor Compatibility in AHV.....	233
Requirements for Advanced Processor Compatibility.....	235
Limitations and Considerations of Advanced Processor Compatibility.....	235
Supported Configuration Workflows for Advanced Processor Compatibility.....	236
Supported CPU Generation Names for Advanced Processor Compatibility.....	236

VM Template Management.....	239
Limitations of VM Template Feature.....	239
Creating a VM Template.....	239
Deploying VM from a Template.....	242
Managing a VM Template.....	246



Copyright.....250



# AHV OVERVIEW

---

As the default option for Nutanix HCI, the native Nutanix hypervisor, AHV, represents a unique approach to virtualization that offers the powerful virtualization capabilities needed to deploy and manage enterprise applications. AHV complements the HCI value by integrating native virtualization along with networking, infrastructure, and operations management with a single intuitive interface - Nutanix Prism.

Virtualization teams find AHV easy to learn and transition to from legacy virtualization solutions with familiar workflows for VM operations, live migration, VM high availability, and virtual network management. AHV includes resiliency features, including high availability and dynamic scheduling without the need for additional licensing, and security is integral to every aspect of the system from the ground up. AHV also incorporates the optional Flow Security and Networking, allowing easy access to hypervisor-based network microsegmentation and advanced software-defined networking.

See the [Field Installation Guide](#) for information about how to deploy and create a cluster. Once you create the cluster by using Foundation, you can use this guide to perform day-to-day management tasks.

## AOS and AHV Compatibility

For information about the AOS and AHV compatibility with this release, see [Compatibility and Interoperability Matrix](#).

## Minimum Field Requirements for Nutanix Cloud Infrastructure (NCI)

For information about minimum field requirements for NCI, see [Minimum Field Requirements for Nutanix Cloud Infrastructure \(NCI\)](#) topic in *Acropolis Advanced Administration Guide*.

## Limitations

For information about AHV configuration limitations, see [Nutanix Configuration Maximums](#) webpage.

## Nested Virtualization

Nutanix does not support nested virtualization (nested VMs) in an AHV cluster.

## Storage Overview

AHV uses a Distributed Storage Fabric to deliver data services such as storage provisioning, snapshots, clones, and data protection to VMs directly.

In AHV clusters, AOS passes all disks to the VMs as raw SCSI block devices. By that means, the I/O path is lightweight and optimized. Each AHV host runs an iSCSI redirector, which establishes a highly resilient storage path from each VM to storage across the Nutanix cluster.

QEMU is configured with the iSCSI redirector as the iSCSI target portal. Upon a login request, the redirector performs an iSCSI login redirect to a healthy Stargate (preferably the local one).



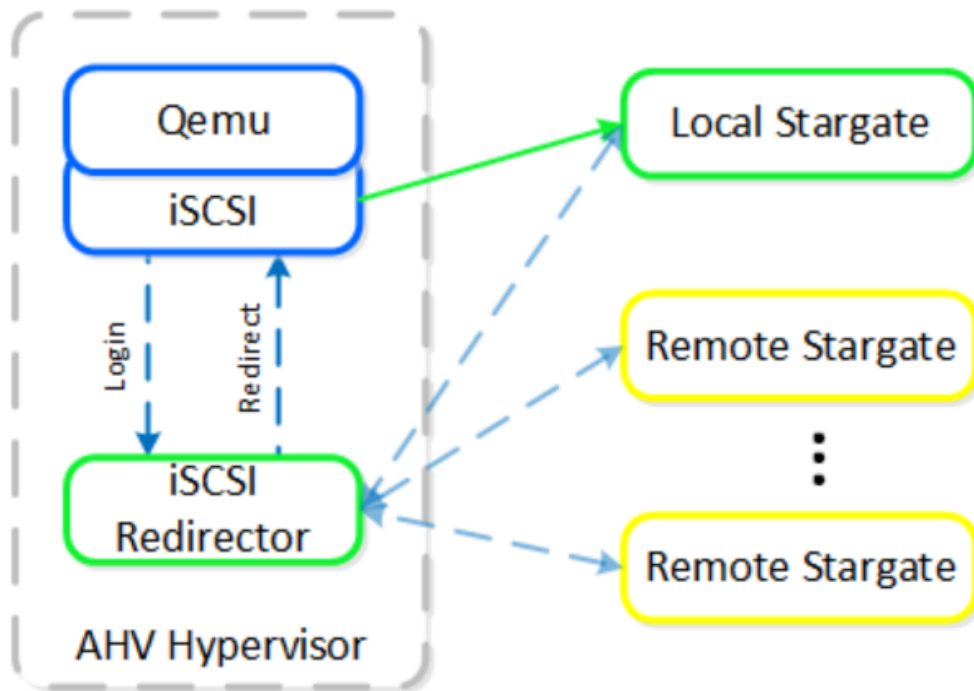


Figure 1: AHV Storage

## AHV Turbo

AHV Turbo represents significant advances to the data path in AHV. AHV Turbo provides an I/O path that bypasses QEMU and services storage I/O requests, which lowers CPU usage and increases the amount of storage I/O available to VMs.

AHV Turbo represents significant advances to the data path in AHV.

When you use QEMU, all I/O travels through a single queue that can impact system performance. AHV Turbo provides an I/O path that uses the multi-queue approach to bypasses QEMU. The multi-queue approach allows the data to flow from a VM to the storage more efficiently. This results in a much higher I/O capacity and lower CPU usage. The storage queues automatically scale out to match the number of vCPUs configured for a given VM, and results in a higher performance as the workload scales up.

AHV Turbo is transparent to VMs and is enabled by default on VMs that runs in AHV clusters. For maximum VM performance, ensure that the following conditions are met:

- The latest Nutanix VirtIO package is installed for Windows VMs. For information on how to download and install the latest VirtIO package, see [Installing or Upgrading Nutanix VirtIO for Windows](#).

Note: No additional configuration is required at this stage.

- The VM has more than one vCPU.
- The workloads are multi-threaded.

Note: Multi-queue is enabled by default in current Linux distributions. For details, refer your vendor-specific documentation for Linux distribution.

In addition to multi-queue approach for storage I/O, you can also achieve the maximum network I/O performance using the multi-queue approach for any vNICs in the system. For information about how to enable multi-queue and set an optimum number of queues, see [Enabling RSS Virtio-Net Multi-Queue by Increasing the Number of vNIC Queues](#).

Note: Ensure that the guest operating system fully supports multi-queue before you enable it. For details, refer your vendor-specific documentation for Linux distribution.

## Acropolis Dynamic Scheduling in AHV

Acropolis Dynamic Scheduling (ADS) proactively monitors your cluster for any compute and storage contentions or hotspots over a period of time. If ADS detects a problem, ADS creates a migration plan that eliminates hotspots in the cluster by migrating VMs from one host to another.

You can monitor the VM migration tasks from the **Tasks** dashboard in both Prism Element and Prism Central.

ADS provides the following advantages:

- ADS performs the runtime optimizations, including moving particular guest VMs and VGs to other AHV hosts to give all workloads the best possible access to resources.
- ADS can enable initial placement of the guest VM in cases when there is not enough space to power it on initially. In this case, the system invokes the ADS and attempts to de-fragment the cluster, which involves moving and rearranging the existing guest VMs to create space for the new guest VM.
- Nutanix Volumes uses ADS to balance sessions of the externally available iSCSI targets.

Note:

- ADS honors all the configured host affinity policies from Prism Central, VM-host affinities from Prism Element, VM-VM anti affinity policies, and HA policies.

The Acropolis Dynamic Scheduling (ADS) always attempts to maintain compliance with the VM-VM anti-affinity policy and ensures that the VM-VM anti-affinity policy is enforced on a best-effort basis. For example, if you manually migrate a VM and the migration leads to non-compliance with the VM-VM anti-affinity policy, ADS performs the following actions:

- Ignores compliance to VM-VM anti-affinity policy, if a host is specified during manual migration.
- Attempts to enforce the policy back into compliance on a best-effort basis, if a host is not specified during manual migration.
- ADS does not consider the nodes that are under maintenance.

By default, ADS is enabled, and Nutanix recommends that you keep this feature enabled. However, see [Disabling Acropolis Dynamic Scheduling](#) on page 11 for information about how to disable the ADS feature. See [Enabling Acropolis Dynamic Scheduling](#) on page 11 for information about how to enable the ADS feature if you previously disabled the feature.

ADS monitors the following resources:

- VM CPU utilization that includes the total CPU usage of each guest VM and storage CPU utilization that includes the storage controller (Stargate) CPU usage per VM or iSCSI target for the last 10 minutes. Based on the last 10 minutes of data, ADS estimates whether a CPU





hotspot exists or is going to exist on a node. This trend analysis ensures that ADS does not migrate the guest VMs when there are sudden spikes in CPU usage that tend to go away quickly. For example, in the case of a boot storm where the CPU usage increases but goes away quickly, ADS does not migrate the guest VM from the host.

- Memory utilization of the guest VMs when memory overcommit is enabled. ADS computes a differential value of the last 2 minutes using statistics such as the amount of accessed memory and guest swaps to identify the below memory overcommit aspects for the guest VMs:
  - Guest VMs needs more memory.
  - Guest VMs already has more memory assigned to them than the actual memory requirement.

Based on these memory overcommit aspects, the host decides the exact memory allocation for the guest VM.

For information on memory overcommit function, see [Memory Overcommit](#) on page 216.

- vGPU profiles to ensure the correct assignment to the compatible GPUs during hotspot mitigation. ADS migrates the guest VMs with virtual GPUs only to the destination node that consists of the compatible GPUs. For more information, see [ADS support for VMs with vGPUs](#) on page 198.

ADS does not monitor the network usage, and memory when memory overcommit is not enabled.

### How Acropolis Dynamic Scheduling Works

ADS performs the following tasks to resolve compute and storage contentions or hotspots:

- Gather statistics from the components it monitors.
- Checks the statistics for potential anomalies and determines how to resolve them, if possible.
- Invokes the tasks (for example, VM migrations) to resolve the situation.

#### Note:

- During initial VM placement, if AHV cannot place the guest VM directly on any node in the cluster, it initiates ADS to check whether a cluster de-fragmentation, which involves migrating or rearranging the existing guest VMs within the cluster, can create enough space for the new guest VM. However, if there are any existing hotspots and not enough space to accommodate the new guest VM in the cluster, the initial placement of the guest VM fails.
- During migration, a guest VM consumes resources on both the source and destination hosts as the High Availability (HA) reservation algorithm must protect the VM on both hosts.
- When multiple hosts are available, ADS checks the availability of attributes such as CPU, memory, storage-controller, and vGPUs usage on each AHV host and selects the AHV host that has the overall highest availability of these attributes for VM placement.

### When Is a Hotspot Detected?

ADS performs the following actions to detect and mitigate the hotspot:



- Runs every 15 minutes to optimize CPU and storage utilization if there is a hotspot. For storage hotspots, the system computes average Stargate CPU usage over the last 10 minutes, and for CPU hotspots, the system computes average host CPU usage over the last 10 minutes. If the resource utilization of an AHV host remains greater than 85% for this span. ADS triggers migration tasks to remove the hotspot.
- Runs every 2 minutes for memory overcommit optimization.
- Maintains the pace period of 21 minutes after completion of a migration plan to ensure that its future decisions only consider statistics resulting from the last migration plan. The pace period is the time duration after which the ADS is run again to capture the current cluster state.

Note:

- If any anomaly is detected and ADS cannot solve the issue (for example, because of limited CPU or storage resources), AHV does not generate the migration plan.
- If the host, firmware, or AOS upgrade is in progress and if any resource contention occurs during the upgrade period, ADS does not perform any resource contention rebalancing. However, if memory overcommit is enabled for the guest VMs, ADS performs the memory overcommit optimization.

During memory overcommit optimization, ADS, if required, increases the amount of memory allocated to guest VMs for which the memory overcommit function is enabled. The memory increase depends on the attributes such as the memory requirements of the other guest VMs on the host for which memory overcommit is not enabled and HA reservations configured in the cluster. For information on memory overcommit, see [Memory Overcommit Management](#) on page 221.

Following are the possible reasons if there is an obvious hotspot, but the guest VMs do not migrate:

- ADS cannot resolve a hotspot. For example, if there is a 3-node cluster with the following specifications:
  - First node: 1 VM with 70% CPU usage and 3 VMs with 10% CPU usage. This node has 100% of CPU usage.
  - Second node: 2 VMs with 40% CPU usage. This node has 80% of CPU usage.
  - Third node: 2 VMs with 40% CPU usage. This node also has 80% of CPU usage.

In this situation, the other hosts (second node and third node) cannot mitigate the hotspot on first node. ADS does not prioritize one host or VM over others for contention, so it leaves the VM they are already hosted.

- The number of all-flash nodes in the cluster is less than the replication factor. If the cluster has an RF2 configuration, the cluster must have a minimum of two all-flash nodes for successful migration of VMs on all the all-flash nodes.

### VM Migration Restrictions with ADS

ADS does not migrate the guest VMs that are configured with the below features as part of the resource contention remediation plan:

- Windows Defender Credential Guard. For more information, see [Windows Defender Credential Guard Support in AHV](#) on page 154.



- Virtual Non-Uniform Memory Access (vNUMA). For more information, see [Virtual Machine Memory Management \(vNUMA\)](#) on page 191
- GPU pass-through. For more information, see [GPU Pass-Through for Guest VMs](#) on page 194.
- CPU pass-through.

For information about VM live migration restrictions, see [Live Migration Restrictions](#) on page 127.

### Migrations Audit

Prism Central displays the list of all the VM migration operations generated by ADS. After you log in to Prism Central, select the **Infrastructure** application from [Application Switcher Function](#), and navigate to **Activity > Audits** to view the VM migrations list. You click **Modify Filters** and select **Migrate** checkbox in the **Operation Type** field in **Filters** pane to filter the list of migration tasks. The list displays all the VM migration tasks created by ADS with details such as the source and target host, VM name, and time of migration.

## Disabling Acropolis Dynamic Scheduling

Perform the procedure described in this topic to disable ADS. Nutanix recommends you keep ADS enabled.

### Procedure

1. Log on to a Controller VM in your cluster with SSH.
2. Disable ADS.

```
nutanix@cvm$ acli ads.update enable=false
```

No action is taken by ADS to solve the contentions after you disable the ADS feature. You must manually take the remedial actions or you can enable the feature.

## Enabling Acropolis Dynamic Scheduling

If you have disabled the ADS feature and want to enable the feature, perform the following procedure.

### Procedure

1. Log onto a Controller VM in your cluster with SSH.
2. Enable ADS.

```
nutanix@cvm$ acli ads.update enable=true
```

## Virtualization Management Web Console Interface

You can manage the virtualization management features by using the Prism GUI (Prism Element and Prism Central web consoles).

You can do the following by using the Prism web consoles:

- Configure network connections
- Create virtual machines



- Manage virtual machines (launch console, start/shut down, take snapshots, migrate, clone, update, and delete)
- Monitor virtual machines
- Enable VM high availability

For more information, see [Prism Element Web Console Guide](#) and [Prism Central Infrastructure Guide](#).

## Finding the AHV Version on Prism Element

You can see the installed AHV version in the Prism Element web console.

### About this task

To view the AHV version installed on the host, do the following.

### Procedure

1. Log in to Prism web console.
2. The **Hypervisor Summary widget** widget on the top left side of the Home page displays the AHV version.

## Finding the AHV Version on Prism Central

### About this task

You can see the installed AHV version in Prism Central.

### Procedure

To view the AHV version installed on any host in the clusters managed by Prism Central, perform the following steps:

1. Log in to Prism Central.
2. Select the **Infrastructure** application from [Application Switcher Function](#), and navigate to **Hardware > Hosts** from the **Navigation Bar**. For information about the **Navigation Bar**, see [Application-specific Navigation Bar](#).  
The system displays the **List** tab by default with all the hosts across registered clusters.
3. Click the target host name for which you want to see the hypervisor version.  
The system displays the [Host Details View](#) with **Summary** tab.



4. Observe the **Hypervisor Version** field in the **Properties** widget to view the hypervisor version.

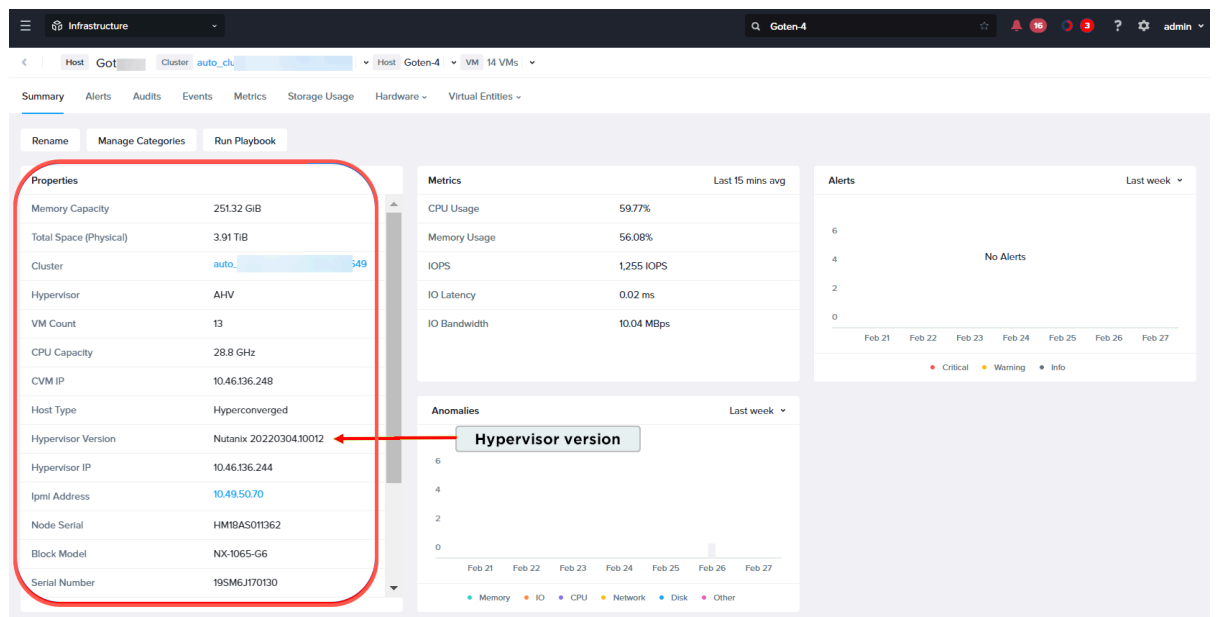


Figure 2: Hypervisor Version in Host Detail View

## AHV Cluster Power Outage Handling

When a power outage occurs, the cluster goes down. After the power is restored, the Nutanix/AHV cluster is recovered by default in the following order of events:

1. Nodes have power restored - which could occur automatically or with manual intervention, depending on BIOS settings.
2. The AHV host automatically starts the CVM.
3. The CVM connects to the host. The CVM can be the local CVM or another CVM in the cluster.
4. The CVM performs a series of checks to identify the VMs that were running on the hosts and applies the appropriate power-on actions.
5. The agent VMs which were previously running on the host are powered on.
6. The remaining user VMs (UVMs) which were previously running on the host are then inspected. If the UVMs were not protected by HA, then they are started. If the UVMs were protected by HA and were started on a different host, then a live migration is triggered to migrate the UVMs back to the recovered host.

### Note:

- The powering on operation across hosts in case of cluster outage may not follow any order, and the checks from the CVM are also unordered. Each of the hosts are handled in parallel.
- There is no option to affect the VM start-up order or keep the VMs in powered down state.

## Serial Console Redirection to a Telnet Port

The serial port is used for console redirection and to output all the AHV kernel messages to the COM port of the hardware.

To manage the serial port, log on to any CVM using SSH as the nutanix user and run the following commands:

Important: Ensure that you power cycle the VM after you add or detach a serial port for the VM, as the changes to the serial port configuration only take effect after the VM power cycle.

Table 1: Commands - Serial Port Management

Command	Description
<code>nutanix@cvm:~\$ acli vm.serial_port_create <i>vm name</i> <i>index=index</i> <i>type=serial port type</i></code>	Used to add a new serial port at COM1 in server mode. For example: <code>nutanix@cvm:~\$ acli vm.serial_port_create my_vm index=0 type=kServer</code>
<code>nutanix@cvm:~\$ acli vm.serial_port_delete <i>vm name</i> <i>index</i></code>	Used to detach a serial port at COM1. For example: <code>nutanix@cvm:~\$ acli vm.serial_port_delete my_vm 0</code>

Replace `vm name` with the VM name, `index` with the serial port index, and `type` with the serial port type.

For information on how to capture the serial output for a Linux VM on AHV, see [KB-4819](#).

Note: The system generates the console log of the CVM at `/var/log/NTNX.serial.out.0`. If a CVM is powered on, but not reachable via SSH or is in a boot loop, you can check this file to view the console output of the CVM.

# NODE MANAGEMENT

---

## Nonconfigurable AHV Components

The components listed here are configured by the Nutanix manufacturing and installation processes. Do not modify any of these components except under the direction of Nutanix Support.

### Nutanix Software

Modifying any of the following Nutanix software settings may inadvertently constrain performance of your Nutanix cluster or render the Nutanix cluster inoperable.

- Local datastore name.
- Configuration and contents of any Controller VM (CVM) (except memory configuration to enable certain features).

Important: Note the following important considerations about CVMs.

- Do not delete the Nutanix CVM.
- Do not take a snapshot of the CVM for backup.
- Do not rename, modify, or delete the **admin** and **nutanix** user accounts of the CVM.
- Do not create additional CVM user accounts.

Use the default accounts (**admin** or **nutanix**), or use **sudo** to elevate to the **root** account.

- Do not decrease CVM memory below recommended minimum amounts required for cluster and add-in features.

Nutanix Cluster Checks (NCC), preupgrade cluster checks, and the AOS upgrade process detect and monitor CVM memory.

- Nutanix does not support the usage of third-party storage on the host part of Nutanix clusters.

Normal cluster operations might be affected if there are connectivity issues with the third-party storage you attach to the hosts in a Nutanix cluster.

- Do not run any commands on a CVM that are not in the Nutanix documentation.

### AHV Settings

Nutanix AHV is a cluster-optimized hypervisor appliance.

Alteration of the hypervisor appliance (unless advised by Nutanix Technical Support) is unsupported and may result in the hypervisor or VMs functioning incorrectly.

Unsupported alterations include (but are not limited to):

- Hypervisor configuration, including installed packages
- Controller VM virtual hardware configuration file (.xml file). Each AOS version and upgrade includes a specific Controller VM virtual hardware configuration. Therefore, do not edit or otherwise modify the Controller VM virtual hardware configuration file.



- iSCSI settings
- Open vSwitch settings
- Installation of third-party software not approved by Nutanix
- Installation or upgrade of software packages from non-Nutanix sources (using yum, rpm, or similar)
- Taking snapshots of the Controller VM
- Creating user accounts on AHV hosts
- Changing the timezone of the AHV hosts. By default, the timezone of an AHV host is set to UTC.
- Joining AHV hosts to Active Directory or OpenLDAP domains

## Controller VM Access

Although each host in a Nutanix cluster runs a hypervisor independent of other hosts in the cluster, some operations affect the entire cluster.

Most administrative functions of a Nutanix cluster can be performed through the web console (Prism), however, there are some management tasks that require access to the Controller VM (CVM) over SSH. Nutanix recommends restricting Controller VM (CVM) SSH access with password or key authentication.

This topic provides information about how to access the CVM as an **admin** user and **nutanix** user.

Note: The direct SSH access to CVM and Prism Central VM (PCVM) is disabled for the **root** user.

### admin User Access

Use the **admin** user access for all tasks and operations that you must perform on the controller VM. As an **admin** user with default credentials, you cannot access nCLI. You must change the default password before you can use nCLI. Nutanix recommends that you do not create additional CVM user accounts. Use the default accounts (**admin** or **nutanix**), or use **sudo** to elevate to the **root** account.

For more information about **admin** user access, see [Admin User Access to Controller VM](#) on page 17.

### nutanix User Access

Nutanix strongly recommends that you do not use the **nutanix** user access unless the procedure (as provided in a Nutanix Knowledge Base article or user guide) specifically requires the use of the **nutanix** user access.

For more information about **nutanix** user access, see [Nutanix User Access to Controller VM](#) on page 18.

You can perform most administrative functions of a Nutanix cluster through the Prism web consoles or REST API. Nutanix recommends using these interfaces whenever possible and disabling Controller VM SSH access with password or key authentication. Some functions, however, require logging on to a Controller VM with SSH. Exercise caution whenever connecting directly to a Controller VM as it increases the risk of causing cluster issues.

Warning: When you connect to a Controller VM with SSH, ensure that the SSH client does not import or change any locale settings. The Nutanix software is not localized, and running the commands with any locale other than en\_US.UTF-8 can cause severe cluster issues.





To check the locale used in an SSH session, run `/usr/bin/locale`. If any environment variables are set to anything other than `en_US.UTF-8`, reconnect with an SSH configuration that does not import or change any locale settings.

## Admin User Access to Controller VM

You can access the Controller VM as the admin user (**admin** user name and password) with SSH. For security reasons, the password of the admin user must meet [Controller VM Password Complexity Requirements](#). When you log on to the Controller VM as the admin user for the first time, you are prompted to change the default password.

See [Controller VM Password Complexity Requirements](#) to set a secure password.

After you have successfully changed the password, the new password is synchronized across all Controller VMs and interfaces (Prism web console, nCLI, and SSH).

Note:

- As an **admin** user, you cannot access nCLI by using the default credentials. If you are logging in as the **admin** user for the first time, you must log on through the Prism web console or SSH to the Controller VM. Also, you cannot change the default password of the **admin** user through nCLI. To change the default password of the **admin** user, you must log on through the Prism web console or SSH to the Controller VM.
- When you make an attempt to log in to the Prism web console for the first time after you upgrade to AOS 5.1 from an earlier AOS version, you can use your existing **admin** user password to log in and then change the existing password (you are prompted) to adhere to the password complexity requirements. However, if you are logging in to the Controller VM with SSH for the first time after the upgrade as the **admin** user, you must use the default **admin** user password (Nutanix/4u) and then change the default password (you are prompted) to adhere to the [Controller VM Password Complexity Requirements](#).
- You cannot delete the **admin** user account.
- The default password expiration age for the **admin** user is 60 days. You can configure the minimum and maximum password expiration days based on your security requirement.
  - `nutanix@cvm$ sudo chage -M MAX-DAYS admin`
  - `nutanix@cvm$ sudo chage -m MIN-DAYS admin`

When you change the **admin** user password, you must update any applications and scripts using the **admin** user credentials for authentication. Nutanix recommends that you create a user assigned with the admin role instead of using the **admin** user for authentication. The [Prism Element Web Console Guide](#) describes authentication and roles.

Following are the default credentials to access a Controller VM.

Table 2: Controller VM Credentials

Interface	Target	User Name	Password
SSH client	Nutanix Controller VM	<b>admin</b>	Nutanix/4u



Interface	Target	User Name	Password
		<b>nutanix</b>	nutanix/4u
Prism web console	Nutanix Controller VM	<b>admin</b>	Nutanix/4u

## Accessing the Controller VM Using the Admin User Account

### About this task

Perform the following procedure to log on to the Controller VM by using the admin user with SSH for the first time.

### Procedure

1. Log on to the Controller VM with SSH by using the management IP address of the Controller VM and the following credentials.

- User name: **admin**
- Password: **Nutanix/4u**

You are now prompted to change the default password.

2. Respond to the prompts, providing the current and new **admin** user password.

```
Changing password for admin.
Old Password:
New password:
Retype new password:
Password changed.
```

See the requirements listed in [Controller VM Password Complexity Requirements](#) to set a secure password.

For information about logging on to a Controller VM by using the **admin** user account through the Prism web console, see [Logging Into The Web Console](#) in the *Prism Element Web Console Guide*.

## Nutanix User Access to Controller VM

You can access the Controller VM as the **nutanix** user (**nutanix** user name and password) with SSH. For security reasons, the password of the **nutanix** user must meet the [Controller VM Password Complexity Requirements](#) on page 20. When you log on to the Controller VM as the **nutanix** user for the first time, you are prompted to change the default password.

See [Controller VM Password Complexity Requirements](#) on page 20 to set a secure password.

After you have successfully changed the password, the new password is synchronized across all Controller VMs and interfaces (Prism web console, nCLI, and SSH).

#### Note:

- As a **nutanix** user, you cannot access nCLI by using the default credentials. If you are logging in as the **nutanix** user for the first time, you must log on through the Prism web console or SSH to the Controller VM. Also, you cannot change the default password of the **nutanix** user through nCLI. To change the default password of the **nutanix** user, you must log on through the Prism web console or SSH to the Controller VM.



- When you make an attempt to log in to the Prism web console for the first time after you upgrade the AOS from an earlier AOS version, you can use your existing **nutanix** user password to log in and then change the existing password (you are prompted) to adhere to the password complexity requirements. However, if you are logging in to the Controller VM with SSH for the first time after the upgrade as the **nutanix** user, you must use the default **nutanix** user password (nutanix/4u) and then change the default password (you are prompted) to adhere to the [Controller VM Password Complexity Requirements](#) on page 20.
- You cannot delete the **nutanix** user account.
- For enhanced access restrictions for the **nutanix** user, consider enabling the **Cluster Lockdown** feature. For more information, see [Controlling Cluster Access](#) section in *Security Guide*.

Important: Nutanix does not recommend changing the password expiry setting for the **nutanix** user account. An expired **nutanix** user account can cause cluster stability issues.

When you change the **nutanix** user password, you must update any applications and scripts using the **nutanix** user credentials for authentication. Nutanix recommends that you create a user assigned with the nutanix role instead of using the **nutanix** user for authentication. The [Prism Element Web Console Guide](#) describes authentication and roles.

Following are the default credentials to access a Controller VM.

Table 3: Controller VM Credentials

Interface	Target	User Name	Password
SSH client	Nutanix Controller VM	<b>admin</b>	Nutanix/4u
		<b>nutanix</b>	nutanix/4u
Prism web console	Nutanix Controller VM	<b>admin</b>	Nutanix/4u

## Accessing the Controller VM Using the Nutanix User Account

### About this task

Perform the following procedure to log on to the Controller VM by using the nutanix user with SSH for the first time.

### Procedure

1. Log on to the Controller VM with SSH by using the management IP address of the Controller VM and the following credentials.
  - User name: **nutanix**
  - Password: **nutanix/4u**

You are now prompted to change the default password.
2. Respond to the prompts, providing the current and new **nutanix** user password.

Changing password for nutanix.



```
Old Password:
New password:
Retype new password:
Password changed.
```

See [Controller VM Password Complexity Requirements](#) on page 20 to set a secure password.

For information about logging on to a Controller VM by using the nutanix user account through the Prism web console, see [Logging Into The Web Console](#) in the *Prism Element Web Console Guide*.

## Controller VM Password Complexity Requirements

The password must meet the following complexity requirements:

- At least eight characters long.
- At least one lowercase letter.
- At least one uppercase letter.
- At least one number.
- At least one special character.

Note: Ensure that the following conditions are met for the special characters usage in the CVM password:

- The special characters are appropriately used while setting up the CVM password. In some cases, for example when you use *! followed by a number* in the CVM password, it leads to a special meaning at the system end, and the system may replace it with a command from the bash history. In this case, you may generate a password string different from the actual password that you intend to set.
- The special character used in the CVM password are ASCII printable characters only. For information about ASCII printable characters, refer *ASCII printable characters (character code 32-127)* article on ASCII code website.

- At least four characters difference from the old password.
- Must not be among the last 5 passwords.
- Must not have more than 2 consecutive occurrences of a character.
- Must not be longer than 199 characters.

If a password for an account (CVM account) is entered five times unsuccessfully within a 15-minute period, the account is locked for 15 minutes.

## AHV Host Access

You can perform most of the administrative functions of a Nutanix cluster using the Prism web consoles or REST API. Nutanix recommends using these interfaces whenever possible. Some functions, however, require logging on to an AHV host with SSH.

Note: From AOS 5.15.5 with AHV 20190916.410 onwards, AHV has two new user accounts—**admin** and **nutanix**.

Nutanix provides the following users to access the AHV host:



- **root**—It is used internally by the AOS. The root user is used for the initial access and configuration of the AHV host.
- **admin**—It is used to log on to an AHV host. The admin user is recommended for accessing the AHV host.
- **nutanix**—It is used internally by the AOS and must not be used for interactive login.

Exercise caution whenever connecting directly to an AHV host as it increases the risk of causing cluster issues.

Following are the default credentials to access an AHV host:

Table 4: AHV Host Credentials

Interface	Target	User Name	Password
SSH client	AHV Host	root	nutanix/4u
		admin	There is no default password for admin. You must set it during the <a href="#">initial configuration</a> .
		nutanix	nutanix/4u

## Initial Configuration

### About this task

The AHV host is shipped with the default password for the **root** and **nutanix** users, which must be changed using SSH when you log on to the AHV host for the first time. After changing the default passwords and the **admin** password, all subsequent logins to the AHV host must be with the **admin** user.

Perform the following procedure to change **admin** user account password for the first time:

Note: Perform this initial configuration on all the AHV hosts.

### Procedure

1. Use SSH and log on to the AHV host using the **root** account.

```
$ ssh root@<AHV Host IP Address>
Nutanix AHV
root@<AHV Host IP Address> password: # default password nutanix/4u
```

2. Change the default **root** user password.

```
root@ahv# passwd root
Changing password for user root.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```



3. Change the default `nutanix` user password.

```
root@ahv# passwd nutanix
Changing password for user nutanix.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

4. Change the `admin` user password.

```
root@ahv# passwd admin
Changing password for user admin.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

## Accessing the AHV Host Using the Admin Account

### About this task

After setting the `admin` password in the [Initial Configuration](#) on page 21, use the `admin` user for all subsequent logins.

Perform the following procedure to access the AHV host using the `admin` account.

### Procedure

1. Log on to the AHV host with SSH using the `admin` account.

```
$ ssh admin@ <AHV Host IP Address>
Nutanix AHV
```

2. Enter the `admin` user password configured in the [Initial Configuration](#) on page 21.

```
admin@<AHV Host IP Address> password:
```

### Changing Admin User Password

### About this task

Perform these steps to change the `admin` password on every AHV host in the cluster:

### Procedure

1. Log on to the AHV host using the `admin` account with SSH.
2. Enter the `admin` user password configured in the [Initial Configuration](#) on page 21.
3. Run the `sudo` command to change to `admin` user password.

```
$ sudo passwd admin
```

4. Respond to the prompts and provide the new password.

```
[sudo] password for admin:
Changing password for user admin.
New password:
Retype new password:
```



```
passwd: all authentication tokens updated successfully.
```

Note: Repeat this step for each AHV host.

See [AHV Host Password Complexity Requirements](#) on page 24 to set a secure password.

## Changing the Root User Password

### About this task

Perform these steps to change the **root** password on every AHV host in the cluster:

### Procedure

1. Log on to the AHV host using the **admin** account with SSH.
2. Run the sudo command to change to **root** user.
3. Change the **root** password.

```
root@ahv# passwd root
```

4. Respond to the prompts and provide the current and new root password.

```
Changing password for root.
```

```
New password:
```

```
Retype new password:
```

```
passwd: all authentication tokens updated successfully.
```

Note: Repeat this step for each AHV host.

See [AHV Host Password Complexity Requirements](#) on page 24 to set a secure password.

## Changing Nutanix User Password

### About this task

Perform these steps to change the **nutanix** password on every AHV host in the cluster:

### Procedure

1. Log on to the AHV host using the **admin** account with SSH.
2. Run the sudo command to change to **root** user.
3. Change the **nutanix** password.

```
root@ahv# passwd nutanix
```

4. Respond to the prompts and provide the current and new **nutanix** password.

```
Changing password for nutanix.
```

```
New password:
```

```
Retype new password:
```

```
passwd: all authentication tokens updated successfully.
```

Note: Repeat this step for each AHV host.

See [AHV Host Password Complexity Requirements](#) on page 24 to set a secure password.



## AHV Host Password Complexity Requirements

The password you choose must meet the following complexity requirements:

- In configurations with high-security requirements, the password must contain:
  - At least 15 characters.
  - At least one upper case letter (A-Z).
  - At least one lower case letter (a-z).
  - At least one digit (0-9).
  - At least one printable ASCII special (non-alphanumeric) character. For example, a tilde (~), exclamation point (!), at sign (@), number sign (#), or dollar sign (\$).
  - At least eight characters different from the previous password.
  - At most three consecutive occurrences of any given character.
  - At most four consecutive occurrences of any given class.

The password cannot be the same as the last five passwords.

- In configurations without high-security requirements, the password must contain:
  - At least eight characters.
  - At least one upper case letter (A-Z).
  - At least one lower case letter (a-z).
  - At least one digit (0-9).
  - At least one printable ASCII special (non-alphanumeric) character. For example, a tilde (~), exclamation point (!), at sign (@), number sign (#), or dollar sign (\$).
  - At least three characters different from the previous password.
  - At most three consecutive occurrences of any given character.

The password cannot be the same as the last five passwords.

In both types of configuration, if a password for an account is entered five times unsuccessfully within a 15-minute period, the account is locked for 15 minutes.

## Verifying the Cluster Health

Before you perform operations such as restarting a CVM or AHV host and putting an AHV host into maintenance mode, check if the cluster can tolerate a single-node failure.

### Before you begin

Ensure that you are running the most recent version of NCC.

### About this task

Note: If you see any critical alerts, resolve the issues by referring to the indicated KB articles. If you are unable to resolve any issues, contact Nutanix Support.

Perform the following steps to avoid unexpected VM downtime or performance issues.





## Procedure

1. Review and resolve any critical alerts. Do one of the following:

- » In the Prism Element web console, go to the **Alerts** page.
- » Log on to a Controller VM (CVM) with SSH and display the alerts.

```
nutanix@cvm$ ncli alert ls
```

Note: If you receive alerts indicating expired encryption certificates or a key manager is not reachable, resolve these issues before you shut down the cluster. If you do not resolve these issues, data loss of the cluster might occur.

2. Verify if the cluster can tolerate a single-node failure. Do one of the following:

- » In the Prism Element web console, in the **Home** page, check the status of the **Data Resiliency Status** dashboard.

Verify that the status is **OK**. If the status is anything other than **OK**, resolve the indicated issues before you perform any maintenance activity.

- » Log on to a Controller VM (CVM) with SSH and check the fault tolerance status of the cluster.

```
nutanix@cvm$ ncli cluster get-domain-fault-tolerance-status type=node
```

An output similar to the following is displayed:

Important:

```
Domain Type          : NODE
Component Type       : STATIC_CONFIGURATION
Current Fault Tolerance : 1
Fault Tolerance Details :
Last Update Time     : Wed Nov 18 14:22:09 GMT+05:00 2015

Domain Type          : NODE
Component Type       : ERASURE_CODE_STRIP_SIZE
Current Fault Tolerance : 1
Fault Tolerance Details :
Last Update Time     : Wed Nov 18 13:19:58 GMT+05:00 2015

Domain Type          : NODE
Component Type       : METADATA
Current Fault Tolerance : 1
Fault Tolerance Details :
Last Update Time     : Mon Sep 28 14:35:25 GMT+05:00 2015

Domain Type          : NODE
Component Type       : ZOOKEEPER
Current Fault Tolerance : 1
Fault Tolerance Details :
Last Update Time     : Thu Sep 17 11:09:39 GMT+05:00 2015

Domain Type          : NODE
Component Type       : EXTENT_GROUPS
Current Fault Tolerance : 1
Fault Tolerance Details :
Last Update Time     : Wed Nov 18 13:19:58 GMT+05:00 2015

Domain Type          : NODE
```



Component Type	: OPLOG
Current Fault Tolerance	: 1
Fault Tolerance Details	:
Last Update Time	: Wed Nov 18 13:19:58 GMT+05:00 2015
Domain Type	: NODE
Component Type	: FREE_SPACE
Current Fault Tolerance	: 1
Fault Tolerance Details	:
Last Update Time	: Wed Nov 18 14:20:57 GMT+05:00 2015

The value of the **Current Fault Tolerance** column must be at least 1 for all the nodes in the cluster.

## Node Maintenance Mode

You are required to gracefully place a node into the maintenance mode or non-operational state for reasons such as making changes to the network configuration of a node, performing manual firmware upgrades or replacements, performing CVM maintenance or any other maintenance operations.

### Entering and Exiting Maintenance Mode

You can only place one node at a time in maintenance mode for each cluster. When a host is in maintenance mode, the CVM is placed in maintenance mode as part of the node maintenance operation and any associated RF1 VMs are powered-off. The cluster marks the host as unschedulable so that no new VM instances are created on it. When a node is placed in the maintenance mode from the Prism web console, an attempt is made to evacuate VMs from the host. If the evacuation attempt fails, the host remains in the *entering maintenance mode* state, where it is marked unschedulable, waiting for user remediation.

When a host is placed in the maintenance mode, the non-migratable VMs (for example, pinned or RF1 VMs which have affinity towards a specific node) are powered-off while live migratable or high availability (HA) VMs are moved from the original host to other hosts in the cluster. After exiting the maintenance mode, all non-migratable guest VMs are powered on again and the live migrated VMs are automatically restored on the original host.

Note: VMs with CPU passthrough or PCI passthrough, pinned VMs (with host affinity policies), and RF1 VMs are not migrated to other hosts in the cluster when a node undergoes maintenance. Click **View these VMs** link to view the list of VMs that cannot be live-migrated.

For information about how to place a node under maintenance, see [Putting a Node into Maintenance Mode using Web Console](#) on page 27.

You can also place an AHV host under maintenance mode or exit an AHV host from maintenance mode through the CLI.

Note: Using the CLI method to place an AHV host under maintenance only places the hypervisor under maintenance mode. The CVM is up running in this method. To place the entire node under maintenance, Nutanix recommends using the UI method (through web console).

- For information about how to use the CLI method to place an AHV host in maintenance mode, see [Putting a Node into Maintenance Mode using CLI](#) on page 31 .
- For information about how to use the CLI method to exit a node from the maintenance mode, see [Exiting a Node from the Maintenance Mode Using CLI](#) on page 33 .

## Exiting a Node from Maintenance Mode

For information about how to remove a node from the maintenance mode, see [Exiting a Node from the Maintenance Mode using Web Console](#) on page 29.

## Viewing a Node under Maintenance Mode

For information about how to view the node under maintenance mode, see [Viewing a Node that is in Maintenance Mode](#) on page 29.

## UVM Status When Node under Maintenance Mode

For information about how to view the status of UVMs when a node is undergoing maintenance operations, see [Guest VM Status when Node is in Maintenance Mode](#) on page 30.

## Best Practices and Recommendations

Nutanix strongly recommends using the **Enter Maintenance Mode** option on the Prism web console to place a node under maintenance.

## Known Issues and Limitations

- With a minimum AOS release of 6.1.2, 6.5.1 or 6.6, you can only place one node at a time in maintenance mode for each cluster.
- Entering or exiting a node under maintenance from the CLI is not equivalent to entering or exiting the node under maintenance from the Prism Element web console. For example, placing a node under maintenance from the CLI places the AHV host and CVM under maintenance while the CVM continues to remain powered on.
- You must exit the node from maintenance mode using the same method that you have used to put the node into maintenance mode. For example, if you used CLI to put the node into maintenance mode, you must use CLI to exit the node from maintenance mode. Similarly, if you used web console to put the node into maintenance mode, you must use the web console to exit the node from maintenance mode.

## Putting a Node into Maintenance Mode using Web Console

### Before you begin

Check the cluster status and resiliency before putting a node under maintenance. You can also verify the status of the UVMs. For more information, see [Guest VM Status when Node is in Maintenance Mode](#) on page 30.

### About this task

As the node enters the maintenance mode, the following high-level tasks are performed internally.

- The AHV host initiates entering the maintenance mode.
- The HA VMs are live migrated.
- The pinned and RF1 VMs are powered-off.
- The AHV host completes entering the maintenance mode.

Note: At this stage, the AHV host is not shut down. For information about how to shut down the AHV host, see [Shutting Down a Node in a Cluster \(AHV\)](#). You can list all the hosts



in the cluster by running `nutanix@cvm$ accli host.list` command, and note the value of *Hypervisor IP* for the node you want to shut down.

- The CVM enters the maintenance mode.
- The CVM is shut down.

Perform the following steps to put the node into maintenance mode.

### Procedure

1. Login to the Prism Element web console.
2. On the home page, select **Hardware** from the drop-down menu.
3. Go to the **Table > Host** view.
4. Select the node which you intend to put under maintenance.
5. Click the **Enter Maintenance Mode** option.  
The Host Maintenance window appears with a prompt to power-off all VMs that cannot be live migrated.

Note: VMs with CPU passthrough, PCI passthrough, pinned VMs (with host affinity policies), and RFI are not migrated to other hosts in the cluster when a node undergoes maintenance. Click **View these VMs** link to view the list of VMs that cannot be live-migrated.

6. Select the **Power-off VMs that can not migrate** checkbox to enable the **Enter Maintenance Mode** button.
7. Click the **Enter Maintenance Mode** button.
  - A revolving icon appears as a tool tip beside the selected node and also in the Host Details view. This indicates that the host is entering the maintenance mode.
  - The revolving icon disappears and the **Exit Maintenance Mode** option is enabled after the node completely enters the maintenance mode.
  - You can also monitor the progress of the node maintenance operation through the newly created Host enter maintenance and Enter maintenance mode tasks which appear in the task tray.

Note: In case of a node maintenance failure, certain rolled-back operations are performed. For example, the CVM is rebooted. But the live migrated are not restored to the original host.

### What to do next

Once the maintenance activity is complete, you can perform any of the following.

- View the nodes under maintenance. For more information, see [Viewing a Node that is in Maintenance Mode](#) on page 29.
- View the status of the UVMs. For more information, see [Guest VM Status when Node is in Maintenance Mode](#) on page 30.
- Remove the node from the maintenance mode. For more information, see [Exiting a Node from the Maintenance Mode using Web Console](#) on page 29.



## Viewing a Node that is in Maintenance Mode

### About this task

Note: This procedure is the same for AHV and ESXi nodes.

Perform the following steps to view a node under maintenance.

### Procedure

1. Login to the Prism Element web console.
2. On the home page, select **Hardware** from the drop-down menu.
3. Go to the **Table > Host** view.
4. Observe the icon along with a tool tip that appears beside the node which is under maintenance. You can also view this icon in the host details view.
5. Alternatively, view the node under maintenance from the **Hardware > Diagram** view.

### What to do next

You can:

- View the status of the guest VMs. For more information, see [Guest VM Status when Node is in Maintenance Mode](#) on page 30.
- Remove the node from the maintenance mode. For more information, see [Exiting a Node from the Maintenance Mode using Web Console](#) on page 29 [Exiting a Node from the Maintenance Mode \(vSphere\)](#).

## Exiting a Node from the Maintenance Mode using Web Console

After you perform any maintenance activity, exit the node from the maintenance mode.

### About this task

As the node exits the maintenance mode, the following high-level tasks are performed internally.

- The CVM is powered on.
- The CVM is taken out of maintenance.
- The host is taken out of maintenance.

Note: The AHV host is shut down during [Putting a Node into Maintenance Mode using Web Console](#) on page 27 and it is required to power on the AHV host. For information about how to power on the AHV host, see [Starting a Node in a Cluster \(AHV\)](#).

After the host exits the maintenance mode, the RF1 VMs continue to be powered on and the VMs migrate to restore host locality.

For more information, see [Guest VM Status when Node is in Maintenance Mode](#) on page 30 to view the status of the UVMs.

Perform the following steps to remove the node into maintenance mode.

### Procedure

1. On the Prism web console home page, select **Hardware** from the drop-down menu.



2. Go to the **Table > Host** view.
3. Select the node which you intend to remove from the maintenance mode.
4. Click the **Exit Maintenance Mode** option.  
The Host Maintenance window appears.
5. Click the **Exit Maintenance Mode** button.
  - A revolving icon appears as a tool tip beside the selected node and also in the Host Details view. This indicates that the host is exiting the maintenance mode.
  - The revolving icon disappears and the **Enter Maintenance Mode** option is enabled after the node completely exits the maintenance mode.
  - You can also monitor the progress of the exit node maintenance operation through the newly created Host exit maintenance and Exit maintenance mode tasks which appear in the task tray.

### What to do next

Once a node exits the maintenance mode, you can perform any of the following.

- View the status of node under maintenance. For more information, see [Viewing a Node that is in Maintenance Mode](#) on page 29.
- View the status of the UVMs. For more information, see [Guest VM Status when Node is in Maintenance Mode](#) on page 30.

### Guest VM Status when Node is in Maintenance Mode

The following scenarios demonstrate the behavior of three guest VM types - high availability (HA) VMs, pinned VMs, and RF1 VMs, when a node enters and exits a maintenance operation. The HA VMs are live VMs that can migrate across nodes if the host server goes down or reboots. The pinned VMs have the host affinity set to a specific node. The RF1 VMs have affinity towards a specific node or a Controller VM (CVM). To view the status of the guest VMs, go to **VM > Table**.

Note: The following scenarios are the same for AHV and ESXi nodes.

#### Scenario 1: Guest VMs before Node Entering Maintenance Mode

All the guest VMs are powered-on and reside on the same host.

#### Scenario 2: Guest VMs during Node Maintenance Mode

As the node enters the maintenance mode, the following high-level tasks are performed internally.

1. The host initiates entering the maintenance mode.
2. The HA VMs are live migrated.
3. The pinned and RF1 VMs are powered-off.
4. The host completes entering the maintenance mode.
5. The CVM enters the maintenance mode.
6. The AHV host completes entering the maintenance mode.
7. The CVM enters the maintenance mode.
8. The CVM is shut down.



### Scenario 3: Guest VMs after Node Exiting Maintenance Mode

As the node exits the maintenance mode, the following high-level tasks are performed internally.

1. The CVM is powered on.
2. The CVM is taken out of maintenance.
3. The host is taken out of maintenance.

After the host exits the maintenance mode, the RF1 VMs continue to be powered on and the VMs migrate to restore host locality.

## Putting a Node into Maintenance Mode using CLI

You are required to put a node into maintenance mode for reasons such as making changes to the network configuration of a node, performing manual firmware upgrades, or any other.

### Before you begin

Caution: Verify the data resiliency status of your cluster. If the cluster only has replication factor 2 (RF2), you can only shut down one node for each cluster. If an RF2 cluster would have more than one node shut down, shut down the entire cluster.

### About this task

When a host is in maintenance mode, AOS marks the host as unschedulable so that no new VM instances are created on it. Next, an attempt is made to evacuate VMs from the host.

If the evacuation attempt fails, the host remains in the "entering maintenance mode" state, where it is marked unschedulable, waiting for user remediation. You can shut down VMs on the host or move them to other nodes. Once the host has no more running VMs, it is in maintenance mode.

When a host is in maintenance mode, VMs are moved from that host to other hosts in the cluster. After exiting maintenance mode, those VMs are automatically returned to the original host, eliminating the need to manually move them.

VMs with GPU, CPU passthrough, PCI passthrough, and host affinity policies are not migrated to other hosts in the cluster. You can choose to shut down such VMs while putting the node into maintenance mode.

Agent VMs are always shut down if you put a node in maintenance mode and are powered on again after exiting maintenance mode.

Perform the following steps to put the node into maintenance mode.

### Procedure

1. Use SSH to log on to a Controller VM in the cluster.
2. Determine the IP address of the node you want to put into maintenance mode.

```
nutanix@cvm$ acli host.list
```

Note the value of **Hypervisor IP** for the node you want to put in maintenance mode.



- Put the node into maintenance mode.

```
nutanix@cvm$ acli host.enter_maintenance_mode hypervisor-IP-address [wait="{ true | false }" ] [non_migratable_vm_action="{ acpi_shutdown | block }" ]
```

Note: Never put Controller VM and AHV hosts into maintenance mode on single-node clusters. It is recommended to shutdown user VMs before proceeding with disruptive changes.

Replace *hypervisor-IP-address* with either the IP address or host name of the AHV host you want to shut down.

The following are optional parameters for running the `acli host.enter_maintenance_mode` command:

- wait:** Set the **wait** parameter to **true** to wait for the host evacuation attempt to finish.
- non\_migratable\_vm\_action:** By default the **non\_migratable\_vm\_action** parameter is set to **block**, which means VMs with GPU, CPU passthrough, PCI passthrough, and host affinity policies are not migrated or shut down when you put a node into maintenance mode.

If you want to automatically shut down such VMs, set the **non\_migratable\_vm\_action** parameter to **acpi\_shutdown**.

- Verify if the host is in the maintenance mode.

```
nutanix@cvm$ acli host.get host-ip
```

In the output that is displayed, ensure that **node\_state** equals to **EnteredMaintenanceMode** and **schedulable** equals to **False**.

Do not continue if the host has failed to enter the maintenance mode.

- See [Verifying the Cluster Health](#) on page 24 to once again check if the cluster can tolerate a single-node failure.
- Put the CVM into the maintenance mode.

```
nutanix@cvm$ ncli host edit id=host-ID enable-maintenance-mode=true
```

Replace *host-ID* with the ID of the host.

This step prevents the CVM services from being affected by any connectivity issues.

- Determine the ID of the host.

```
nutanix@cvm$ ncli host list
```

An output similar to the following is displayed:

```
Id           : aaaaaaaa-bbbb-cccc-dddd-eeeeeeeeeeee::1234
Uuid         : ffffffff-gggg-hhhh-iiii-jjjjjjjjjjjj
Name         : XXXXXXXXXXXX-X
IPMI Address : X.X.Z.3
Controller VM Address : X.X.X.1
Hypervisor Address : X.X.Y.2
```

In this example, the host ID is 1234.

Wait for a few minutes until the CVM is put into the maintenance mode.





8. Verify if the CVM is in the maintenance mode.

Run the following command on the CVM that you put in the maintenance mode.

```
nutanix@cvm$ genesis status | grep -v "[\]"
```

An output similar to the following is displayed:

```
nutanix@cvm$ genesis status | grep -v "[\]"
2021-09-24 05:28:03.827628: Services running on this node:
genesis: [11189, 11390, 11414, 11415, 15671, 15672, 15673, 15676]
scavenger: [27241, 27525, 27526, 27527]
xmount: [25915, 26055, 26056, 26074]
zookeeper: [13053, 13101, 13102, 13103, 13113, 13130]
nutanix@cvm$
```

Only the Genesis, Scavenger, Xmount, and Zookeeper processes must be running (process ID is displayed next to the process name).

Do not continue if the CVM has failed to enter the maintenance mode, because it can cause a service interruption.

### What to do next

Perform the maintenance activity. Once the maintenance activity is complete, remove the node from the maintenance mode. See [Exiting a Node from the Maintenance Mode Using CLI](#) on page 33 for more information.

## Exiting a Node from the Maintenance Mode Using CLI

After you perform any maintenance activity, exit the node from the maintenance mode.

### About this task

Perform the following to exit the host from the maintenance mode.

### Procedure

1. Remove the CVM from the maintenance mode.
  - a. Determine the ID of the host.

```
nutanix@cvm$ ncli host list
```

An output similar to the following is displayed:

```
Id                : aaaaaaaa-bbbb-cccc-dddd-eeeeeeeeeeee:1234
Uuid              : ffffffff-gggg-hhhh-iiii-jjjjjjjjjjjj
Name              : XXXXXXXXXXXX-X
IPMI Address      : X.X.Z.3
Controller VM Address : X.X.X.1
Hypervisor Address : X.X.Y.2
```

In this example, the host ID is 1234.

- a. From any other CVM in the cluster, run the following command to exit the CVM from the maintenance mode.

```
nutanix@cvm$ ncli host edit id=host-ID enable-maintenance-mode=false
```

Replace ***host-ID*** with the ID of the host.

Note: The command fails if you run the command from the CVM that is in the maintenance mode.



- b. Verify if all the processes on all the CVMs are in the UP state.

```
nutanix@cvm$ cluster status | grep -v UP
```

Do not continue if the CVM has failed to exit the maintenance mode.

2. Remove the AHV host from the maintenance mode.

- a. From any CVM in the cluster, run the following command to exit the AHV host from the maintenance mode.

```
nutanix@cvm$ acli host.exit_maintenance_mode hypervisor-IP-address
```

Replace *hypervisor-IP-address* with the new IP address of the host.

This command migrates (live migration) all the VMs that were previously running on the host back to the host.

- b. Verify if the host has exited the maintenance mode.

```
nutanix@cvm$ acli host.get hypervisor-IP-address
```

In the output that is displayed, ensure that **node\_state** equals to **kAcropolisNormal** or **AcropolisNormal** and **schedulable** equals to **True**.

Contact Nutanix Support if any of the steps described in this document produce unexpected results.

## Shutting Down a Node in a Cluster (AHV)

### Before you begin

Caution: Verify the data resiliency status of your cluster. If the cluster only has replication factor 2 (RF2), you can only shut down one node for each cluster. If an RF2 cluster would have more than one node shut down, shut down the entire cluster.

See [Verifying the Cluster Health](#) on page 24 to check if the cluster can tolerate a single-node failure. Do not proceed if the cluster cannot tolerate a single-node failure.

### About this task

Perform the following procedure to shut down a node.

### Procedure

1. Put the node into maintenance mode as described in [Putting a Node into Maintenance Mode using Web Console](#) on page 27.
2. Log on to the AHV host with SSH.
3. Shut down the host.

```
root@ahv# shutdown -h now
```

### What to do next

See [Starting a Node in a Cluster \(AHV\)](#) on page 35 for instructions about how to start a node, including how to start a CVM and how to exit a node from maintenance mode.



# Starting a Node in a Cluster (AHV)

## About this task

### Procedure

1. On the hardware appliance, power on the node. The CVM starts automatically when you reboot the node.
2. If the node is in maintenance mode, log on to Prism Web Console and remove the node from the maintenance mode.  
See [Exiting a Node from the Maintenance Mode using Web Console](#) on page 29 for more information.
3. Log on to another CVM in the Nutanix cluster with SSH.
4. Verify that the status of all services on all the CVMs are Up.

```
nutanix@cvm$ cluster status
```

If the Nutanix cluster is running properly, output similar to the following is displayed for each node in the Nutanix cluster.

CVM:host IP-Address Up

	Zeus	UP	[9935, 9980, 9981, 9994, 10015, 10037]
	Scavenger	UP	[25880, 26061, 26062]
	Xmount	UP	[21170, 21208]
	SysStatCollector	UP	[22272, 22330, 22331]
	IkatProxy	UP	[23213, 23262]
	IkatControlPlane	UP	[23487, 23565]
	SSLTerminator	UP	[23490, 23620]
	SecureFileSync	UP	[23496, 23645, 23646]
	Medusa	UP	[23912, 23944, 23945, 23946, 24176]
	DynamicRingChanger	UP	[24314, 24404, 24405, 24558]
	Pithos	UP	[24317, 24555, 24556, 24593]
	InsightsDB	UP	[24322, 24472, 24473, 24583]
	Athena	UP	[24329, 24504, 24505]
	Mercury	UP	[24338, 24515, 24516, 24614]
	Mantle	UP	[24344, 24572, 24573, 24634]
	VipMonitor	UP	[18387, 18464, 18465, 18466, 18474]
	Stargate	UP	[24993, 25032]
	InsightsDataTransfer	UP	[25258, 25348, 25349, 25388, 25391, 25393, 25396]
	Ergon	UP	[25263, 25414, 25415]
	Cerebro	UP	[25272, 25462, 25464, 25581]
	Chronos	UP	[25281, 25488, 25489, 25547]
	Curator	UP	[25294, 25528, 25529, 25585]
	Prism	UP	[25718, 25801, 25802, 25899, 25901, 25906, 25941, 25942]
	CIM	UP	[25721, 25829, 25830, 25856]
	AlertManager	UP	[25727, 25862, 25863, 25990]
	Arithmos	UP	[25737, 25896, 25897, 26040]
	Catalog	UP	[25749, 25989, 25991]
	Acropolis	UP	[26011, 26118, 26119]
	Uhura	UP	[26037, 26165, 26166]
	Snmp	UP	[26057, 26214, 26215]
	NutanixGuestTools	UP	[26105, 26282, 26283, 26299]
	MinervaCVM	UP	[27343, 27465, 27466, 27730]
	ClusterConfig	UP	[27358, 27509, 27510]
	Aequitas	UP	[27368, 27567, 27568, 27600]
	APLOSEngine	UP	[27399, 27580, 27581]



APLOS	UP	[27853, 27946, 27947]
Lazan	UP	[27865, 27997, 27999]
Delphi	UP	[27880, 28058, 28060]
Flow	UP	[27896, 28121, 28124]
Anduril	UP	[27913, 28143, 28145]
XTrim	UP	[27956, 28171, 28172]
ClusterHealth	UP	[7102, 7103, 27995, 28209, 28495, 28496,
		28503, 28510,
		28573, 28574, 28577, 28594, 28595, 28597, 28598, 28602, 28603, 28604, 28607, 28645, 28646,
		28648, 28792,
		28793, 28837, 28838, 28840, 28841, 28858, 28859, 29123, 29124, 29127, 29133, 29135, 29142,
		29146, 29150,
		29161, 29162, 29163, 29179, 29187, 29219, 29268, 29273]

## Rebooting an AHV Node in a Nutanix Cluster

### About this task

The **Request Reboot** operation in the Prism web console gracefully restarts the selected nodes one after the other.

Note: Reboot host is a graceful restart workflow. Hosts are automatically put into maintenance mode and all the user VMs are migrated to another host when you perform a reboot operation for a host. There is no impact on the user workload due to the reboot operation. Reboot fails if the AHV node is already in maintenance mode.

### Procedure

To reboot the nodes in the cluster, perform the following steps:

1. Log on to the Prism Element web console.
2. Click the gear icon in the main menu and then select **Reboot** in the **Settings** page.
3. In the Request Reboot window, select the checkbox associated with the nodes you want to restart, and click **Reboot**.

A progress bar is displayed that indicates the progress of the restart of each node.

## Shutting Down an AHV Cluster

You might need to shut down an AHV cluster to perform a maintenance activity or tasks such as relocating the hardware.

### Before you begin

Ensure that you meet the following prerequisites before you shut down the cluster:

- IPMI password is accessible as it is required to perform power actions through IPMI.
- The NCC is upgraded to the most recent version.

- The NCC health check is performed on the cluster.

Log on to a Controller VM (CVM) with SSH and run the complete NCC health check.

```
nutanix@cvm$ ncc health_checks run_all
```

If you receive any failure or error messages, resolve those issues by referring to the KB articles indicated in the output of the NCC check results. If you are unable to resolve these issues, contact Nutanix Support.

Warning: If you receive alerts indicating expired encryption certificates or a key manager is not reachable, resolve these issues before you shut down the cluster. If you do not resolve these issues, data loss of the cluster might occur.

## About this task

Shut down an AHV cluster in the following sequence.

## Procedure

1. Shut down the services or VMs associated with AOS features or Nutanix products. For example, shut down all the Nutanix file server VMs (FSVMs). See the documentation of those features or products for more information.
2. Shut down all the guest VMs in the cluster in one of the following ways.
  - » Shut down the guest VMs from within the guest OS.
  - » Shut down the guest VMs by using the Prism Element web console.
  - » If you are running many VMs, shut down the VMs by using aCLI:
  - a. Log on to a CVM in the cluster with SSH.
  - b. Shut down all the guest VMs in the cluster.

```
nutanix@cvm$ for i in $(accli vm.list power_state=on | grep -v NTNX | awk 'NR!=1 {print $NF}');do accli vm.shutdown $i ; done
```

- c. Verify if all the guest VMs are shut down.

```
nutanix@cvm$ accli vm.list power_state=on
```

- d. If any VMs are on, consider powering off the VMs from within the guest OS. To force shut down through AHV, run the following command:

```
nutanix@cvm$ accli vm.off vm-name
```

Replace `vm-name` with the name of the VM you want to shut down.



3. Stop the Nutanix cluster.
  - a. Log on to any CVM in the cluster with SSH.
  - b. Stop the cluster.

```
nutanix@cvm$ cluster stop
```

- c. Verify if the cluster services have stopped.

```
nutanix@cvm$ cluster status
```

The output displays the message **The state of the cluster: stop**, which confirms that the cluster has stopped.

Note: The following system services continue to run even after the cluster has stopped successfully:

- Zeus
- Scavenger
- Xmount
- VIPMonitor

You can observe the status of these system services in the output logs:

```
The state of the cluster: stop
Lockdown mode: Disabled  CVM: 10.xx.x.xxx Up
    Zeus UP [13130, 13326, 13327, 13347]
    Scavenger UP [15015, 15141, 15142, 15143]
    Xmount UP [15012, 15121, 15122, 15147]
SysStatCollector DOWN []
IkatProxy DOWN []
IkatControlPlane DOWN []
SSLTerminator DOWN []
SecureFileSync DOWN []
    Medusa DOWN []
DynamicRingChanger DOWN []
    Pithos DOWN []
    InsightsDB DOWN []
    Athena DOWN []
    Mercury DOWN []
    Mantle DOWN []
    VipMonitor UP [25898, 25899, 25900, 25901, 25904]
    Stargate DOWN []
InsightsDataTransfer DOWN []
    Ergon DOWN []
    GoErgon DOWN []
    Cerebro DOWN []
    Chronos DOWN []
    Curator DOWN []
    Prism DOWN []
    Hera DOWN []
    CIM DOWN []
AlertManager DOWN []
Arithmos DOWN []
Catalog DOWN []
Acropolis DOWN []
    Uhura DOWN []
NutanixGuestTools DOWN []
    MinervaCVM DOWN []
ClusterConfig DOWN []
```



```
APLOSEngine DOWN []
APLOS DOWN []
PlacementSolver DOWN []
Lazan DOWN []
Polaris DOWN []
Delphi DOWN []
Security DOWN []
Flow DOWN []
Anduril DOWN []
XTrim DOWN []
ClusterHealth DOWN []
```

4. Shut down all the CVMs in the cluster. Log on to each CVM in the cluster with SSH and shut down that CVM.

```
nutanix@cvm$ sudo shutdown -P now
```

5. Shut down each node in the cluster. Perform the following steps for each node in the cluster.
  - a. Log on to the IPMI web console of each node.
  - b. Under **Remote Control** > **Power Control**, select **Power Off Server - Orderly Shutdown** to gracefully shut down the node.

Note: The navigation path, tabs, and UI layout in IPMI web console can vary based on the hardware used at your site.

- c. Ping each host to verify that all AHV hosts are shut down.
6. Complete the maintenance activity or any other tasks.
7. Start all the nodes in the cluster.
  - a. Press the power button on the front of the block for each node.
  - b. Log on to the IPMI web console of each node.
  - c. On the **System** tab, check the **Power Control** status to verify if the node is powered on.

Note: The navigation path, tabs, and UI layout in IPMI web console can vary based on the hardware used at your site.

8. Start the cluster.

a. Wait for approximately 5 minutes after you start the last node to allow the cluster services to start.

All CVMs start automatically after you start all the nodes.

b. Log on to any CVM in the cluster with SSH.

c. Start the cluster.

```
nutanix@cvm$ cluster start
```

d. Verify that all the cluster services are in the UP state.

```
nutanix@cvm$ cluster status
```

e. Start the guest VMs from within the guest OS or use the Prism Element web console.

If you are running many VMs, start the VMs by using aCLI:

```
nutanix@cvm$ for i in $(acli vm.list power_state=off | grep -v NTNX | awk 'NR!=1 {print $NF}');do acli vm.on $i ; done
```

f. Start the services or VMs associated with AOS features or Nutanix products. For example, start all the FSVMs. See the documentation of those features or products for more information.

g. Verify if all guest VMs are powered on by using the Prism Element web console.

## Changing CVM Memory Configuration (AHV)

### About this task

You can increase the memory reserved for each Controller VM in your cluster by using the 1-click Controller VM Memory Upgrade available from the Prism Element web console. Increase memory size depending on the workload type or to enable certain AOS features. See the [Increasing the Controller VM Memory Size](#) topic in the *Prism Element Web Console Guide* for CVM memory sizing recommendations and instructions about how to increase the CVM memory.

## Renaming an AHV Host

You can rename an AHV host from Prism Central.

### About this task

The minimum supported versions to rename an AHV host from Prism Central are version 6.6 for Prism Element (with recommended AHV version) and version 2022.9 for Prism Central. For information about the recommended AHV version based on the AOS release and hardware model, see [Compatibility and Interoperability Matrix](#).

To rename an AHV host from Prism Central, perform the following steps:

### Procedure

1. Log in to Prism Central.

2. Select the **Infrastructure** application from [Application Switcher Function](#), and navigate to **Hardware > Hosts**.

The system displays the **List** tab by default with all the hosts across registered clusters.





3. Select the target host checkbox, and choose **Rename** from the **Actions** dropdown menu. The system displays the **Rename Host** window.
4. Enter a new name for the host in the **New Host Name** field, and click **Save**.

You must adhere to the following rules for the host name:

- Allowed characters are: uppercase letters (A-Z), lowercase letters (a-z), decimal digits (0-9), dots (.), and hyphens (-).
- The host name must start and end with a number or letter.
- A minimum of one character and a maximum of 63 characters are allowed.
- Successive dots are not allowed. Each dot-separated string must follow the first two rules.

Note: Once the AHV host rename task is complete, it might take a few minutes for the changes to propagate from the host to Prism Central.

## Changing the Name of the CVM Displayed in the Prism Web Console

You can change the CVM name that is displayed in the Prism web console. The procedure described in this document does not change the CVM name that is displayed in the terminal or console of an SSH session.

### About this task

You can change the CVM name by using the `change_cvm_display_name` script. Run this script from a CVM other than the CVM whose name you want to change. When you run the `change_cvm_display_name` script, AOS performs the following steps:

1. Checks if the new name starts with **NTNX-** and ends with **-CVM**. The CVM name must have only letters, numbers, and dashes (-).
2. Checks if the CVM has received a shutdown token.
3. Powers off the CVM. The script does not put the CVM or host into maintenance mode. Therefore, the VMs are not migrated from the host and continue to run with the I/O operations redirected to another CVM while the current CVM is in a powered off state.
4. Changes the CVM name, enables auto-start, and powers on the CVM.

Perform the following to change the CVM name displayed in the Prism web console.

### Procedure

1. Use SSH to log on to a CVM other than the CVM whose name you want to change.
2. Change the name of the CVM.

```
nutanix@cvm$ change_cvm_display_name --cvm_ip=CVM-IP --cvm_name=new-name
```

Replace **CVM-IP** with the IP address of the CVM whose name you want to change and **new-name** with the new name for the CVM.

The CVM name must have only letters, numbers, and dashes (-), and must start with **NTNX-** and end with **-CVM**.

Note: Do not run this command from the CVM whose name you want to change, because the script powers off the CVM. In this case, when the CVM is powered off, you lose connectivity to the CVM from the SSH console and the script abruptly ends.



## Adding a Never-Schedulable Node (AHV Only)

Add a never-schedulable node if you want to add a node to increase data storage on your Nutanix cluster, but do not want any AHV VMs to run on that node.

### About this task

AOS never schedules any VMs on a never-schedulable node. Therefore, a never-schedulable node configuration ensures that no additional compute resources such as CPUs are consumed from the Nutanix cluster. In this way, you can meet the compliance and licensing requirements of your virtual applications.

Note the following points about a never-schedulable node configuration.

Note:

- Ensure that at any given time, the cluster has a minimum of three nodes (never-schedulable or otherwise) in function. To add your first never-schedulable node to your Nutanix cluster, the cluster must comprise of at least three schedulable nodes.
- You can add any number of never-schedulable nodes to your Nutanix cluster.
- If you want a node that is already a part of the cluster to work as a never-schedulable node, remove that node from the cluster and then add that node as a never-schedulable node.
- If you no longer need a node to work as a never-schedulable node, remove the node from the cluster.
- Foundation allocates the maximum resources to Controller VM (CVM) of a Storage-only or Never-schedulable node as follows:
  - CVM vCPU = Number of physical host CPUs minus 2, limited to a maximum of 22 vCPUs.

Note: This behavior is applicable till Foundation version 5.3.x. From Foundation version 5.4 onwards, the capping of maximum 22 vCPUs is not applicable.

- CVM memory = Available RAM minus 16 GiB, limited to a maximum of 256 GiB.

Note:

- This behavior is applicable from Foundation version 5.3 and above. In the earlier Foundation versions, the memory allocation happens without capping to 256 GiB.
- A capping of maximum 256 GiB is applied, and Foundation allocates the maximum possible vRAM to CVM. For example, if the available RAM is 512 GiB, the system allocates a maximum of 256 GiB and never considers the  $512 - 16 = 496$  GiB value. However, if you change the system allocated vRAM, the vRAM gets overridden with the supplied value.

Note: Minimum Foundation version of 5.3 supports these limits with NUMA pinnings or alignments. Earlier Foundation versions with a minimum version of 5.0 support these limits but not NUMA pinnings or alignments.



## Procedure

You can add a never-schedulable node (storage-only node) to a cluster using the **Expand Cluster** operation from Prism Web Console.

For information about how to add a never-schedulable node to a cluster, see the [Expanding a Cluster](#) topic in *Prism Element Web Console Guide*.

## AHV Compute-Only Node Configuration

The Nutanix cluster uses the resources (CPUs and memory) of a compute-only (CO) node exclusively for computing purposes. A compute-only (CO) node allows you to seamlessly and efficiently expand the computing capacity (CPU and memory) of your AHV cluster.

Note: Clusters that have compute-only nodes do not support virtual switches. Instead, use bridge configurations for network connections. For more information, see [Virtual Switch Limitations](#).

### Use Cases of AHV Compute-Only Nodes

CO nodes enable you to achieve more control and value from restrictive licenses such as Oracle. There is no Controller VM (CVM) running on the CO node (VMs use CVMs running on the HCI nodes to access disks). When a CO node is part of a Nutanix Hyperconverged Infrastructure (HCI) cluster, then license cores of the CO nodes are used only for the application VMs.

Applications or databases that are licensed on a per CPU core basis require the entire node to be licensed and that also includes the cores on which the CVM runs. With CO nodes, you get a much higher ROI on the purchase of your database licenses (such as Oracle and Microsoft SQL Server) since the CVM does not consume any compute resources.

AHV CO nodes support the following two types of deployments:

- AHV CO node + AHV HCI node. For more information, see [AHV Compute-only with AHV HCI Nodes](#) on page 43.
- AHV CO node + AHV Storage-only (SO) node. For more information, see [AHV Compute-only with AHV Storage-only Nodes](#) on page 45.

### AHV Compute-only with AHV HCI Nodes

You can deploy AHV CO nodes that use AHV hyperconverged (HCI) nodes for storage in a cluster.

#### Key Features of AHV Compute-only Node

Following are the key features of CO nodes that use AHV HCI nodes for storage in a cluster:

- AHV CO nodes support AHV HCI nodes for storage in a cluster.
- AHV CO nodes do not have a Controller VM (CVM) and local storage.  
AOS sources the storage for vDisks associated with VMs running on AHV CO nodes from the HCI nodes (AHV CO + AHV HCI deployment) in the cluster.
- AHV CO nodes use the Prism Element web console to let you seamlessly manage your VMs (CRUD operations, ADS, and HA).
- AHV CO node deployments with AHV HCI nodes support Network segmentation only for Controller VM backplane network and volume networks.
- AHV runs on the local storage media of the CO node.



- Upgrade the AHV version on any AHV CO node using the Life Cycle Manager. For more information, see [Firmware and Software Updates Management](#) section in the *Life Cycle Manager Guide*.

### Minimum Cluster Requirements

Following are the minimum cluster requirements for AHV CO nodes.

- The Nutanix cluster must have a minimum of three nodes before you add an AHV CO node.
- Nutanix recommends that the following ratio of AHV CO nodes to AHV HCI nodes in a cluster must not exceed the following:  
1 AHV CO : 2 AHV HCI
- All the AHV HCI nodes in the cluster must be all-flash nodes.
- The number of vCPUs assigned to Controller VMs on the HCI nodes must be greater than or equal to the total number of available cores on all the AHV CO nodes in the cluster. The Controller VM requires a minimum of 12 vCPUs. For more information about how Foundation allocates memory and vCPUs to your platform model, see [CVM vCPU and vRAM Allocation](#) in the *Field Installation Guide*.
- Nutanix recommends that you use dual 25 GbE on AHV CO nodes and quad 25 GbE on AHV HCI nodes.
- The total amount of NIC bandwidth allocated to all the HCI nodes must be twice the amount of the total NIC bandwidth allocated to all the compute-only nodes in the cluster.
- The AHV version of the AHV CO node must be the same as the other AHV nodes (AHV HCI) in the cluster.

When you add an AHV CO node to the cluster, AOS checks if the AHV version of the node matches with the AHV version of the existing AHV nodes in the cluster. If there is a mismatch, the node addition operation fails.

For general requirements about adding a node to a Nutanix cluster, see [Expanding a Cluster](#) topic in *Prism Web Console Guide*.

### Restrictions

Nutanix does not support the following features or tasks on an AHV CO node:

1. Host boot disk replacement
2. Rolling restart of AHV CO nodes
3. More than 32 nodes in the cluster
4. Virtual Switch configuration: Use bridge configurations instead.

### Licensing

AHV compute-only nodes with AHV HCI nodes deployment uses NCI licenses on a per-core basis. For more information about NCI licences, see NCI section in [Nutanix Cloud Platform Software Options](#).

### Supported AOS Versions

Nutanix supports AHV compute-only nodes with AHV HCI nodes on AOS releases 5.11 or later.

### Supported Hardware Platforms

Compute-only nodes with AHV HCI Nodes are supported on the following hardware platforms.



- For SO node: NX8170-G8, NX-8170-G9, NX-8155-G9, NX-8150-G9, Dell XC750/ XC650
- For CO node: NX8170-G8, NX1175S-G8, NX-8170-G9, NX-1175S-G9, NX-8155-G9, NX-8150-G9, Dell XC750/XC650

## Networking Configuration

To perform network tasks on an AHV CO node such as creating or modifying bridges or uplink bonds or uplink load balancing, use the **manage\_ovs** commands and add the **--host** flag to the **manage\_ovs** commands as shown in the following example:

```
nutanix@cvm$ manage_ovs --host IP_address_of_co_node --bridge_name bridge_name
create_single_bridge
```

Replace **IP\_address\_of\_co\_node** with the IP address of the CO node and **bridge\_name** with the name of bridge you want to create.

Note: Run the **manage\_ovs** commands for an AHV CO node from any Controller VM running on an AHV HCI node.

Perform the networking tasks for each AHV CO node in the cluster separately.

For more information about networking configuration of the AHV hosts, see [Host Network Management](#) on page 51.

## AHV Compute-only with AHV Storage-only Nodes

You can deploy an AHV compute-only node in a AHV storage-only node cluster running a minimum AOS version of 6.7 with the a minimum AHV version of 20230302.204.

### Key Features of AHV Compute-only with AHV Storage-only Nodes

Following are the key features of AHV Compute-only used with AHV Storage-only Nodes:

- AHV CO nodes do not have a Controller VM (CVM) and local storage. The vDisk/Volumes associated with guest VMs on the compute-only (CO) nodes source their storage from storage-only (SO) nodes in the cluster.
- AHV CO nodes use the Prism Element web console to let you seamlessly manage your VMs (CRUD operations, ADS, and HA).
- AHV CO node deployed with AHV SO nodes support Network segmentation only for Controller VM backplane network and volume networks.
- AHV runs on the local storage media of the CO node.
- Upgrade the AHV version and the firmware upgrade modules on any AHV CO node using the Life Cycle Manager. For more information, see [Firmware and Software Updates Management](#) section in the *Life Cycle Manager Guide*.

### Minimum Cluster Requirements

Following are the minimum cluster requirements for AHV CO nodes.

- The Nutanix cluster must have a minimum of three SO nodes (three CVMs) to ensure a healthy and active cluster. The Cluster must have a minimum of 2 AHV CO nodes to maintain high availability.



- Nutanix recommends the following ratio of AHV CO nodes to AHV SO nodes in a cluster:
  - Even nodes - one AHV CO and one AHV SO.
  - Odd Nodes - Nutanix recommends the difference of one between the total number of CO and SO nodes in a cluster.

Note: The above ratio is only a recommendation. You can deploy different combination of CO and SO nodes, provided the combinations comply with minimum 5 nodes and maximum 32 nodes in a cluster and meet your workload requirements.

- All the AHV SO nodes in the cluster must be NVMe nodes.
- Since all the CPUs and memory are allocated to the AHV SO node, there is no necessity to balance the vCPU assignment between the CO and SO nodes.
- Nutanix recommends that you use dual 25 GbE or above on AHV CO and AHV SO nodes.
- The AHV CO node must run the same AHV version as the AHV SO nodes in the cluster.

When you add an AHV CO node to the cluster, AOS checks if the AHV version of the node matches with the AHV version of the existing AHV nodes in the cluster. If there is a mismatch, the node addition fails.

For general requirements about adding a node to a Nutanix cluster, see [Expanding a Cluster](#) in the *Prism Web Console Guide*.

## Restrictions

Nutanix does not support the following features or tasks on a cluster with AHV CO node:

- More than 500 VMs for cluster with ESXi CO nodes
- More than 32 nodes in the cluster
- Addition of HCI node to AHV CO + AHV SO cluster
- Rolling restarts for clusters with AHV CO nodes
- RDMA support for clusters with AHV CO nodes
- Host boot disk replacement for AHV CO nodes
- Virtual Switch configuration: Use bridge configurations instead
- Network segmentation for disaster recovery
- Automatic discovery of ESXi CO nodes as part of *Expand cluster* workflow. You must initiate the manual host discovery workflow to add compute-only node when you use the [Expanding a Cluster](#) workflow.
- It is mandatory to deploy NDB for the combination of AHV SO and AHV CO nodes since NDB Platform licenses are required for the AHV CO nodes.

Host boot disk replacement, rolling restarts, and discovery of AHV CO nodes to expand the cluster must be performed manually. Contact Nutanix Support if required.

## Licensing

The Optimized DB solution uses a combination of NCI Ultimate or NCI Pro licenses for AHV storage-only nodes and NDB Platform licenses for AHV compute-only nodes. Both NCI and NDB platforms are licensed on a per-core basis.



NCI Ultimate on storage-only AHV nodes is the preferred licensing model to get the most functional Optimized DB Solution. When you use the NCI Pro license on the storage-only AHV nodes, the entire cluster functions at the Pro level feature set, and the NDB disaster recovery feature and other advanced functionalities are not available.

For more information about NCI Ultimate and NDB feature set licenses, see <https://www.nutanix.com/products/cloud-platform/software-options>.

### Supported AOS Versions

Nutanix supports Optimized DB solution deployments with AHV CO and AHV SO nodes on AOS 6.7 or later.

### Supported Hardware Platforms

Compute-only nodes are supported on the following hardware platforms.

- For SO node: NX8170-G8, NX-8170-G9, NX-8155-G9, NX-8150-G9, Dell XC750/ XC650
- For CO node: NX8170-G8, NX1175S-G8, NX-8170-G9, NX-1175S-G9, NX-8155-G9, NX-8150-G9, Dell XC750/XC650

### Networking Configuration

To perform network tasks on an AHV CO node such as creating or modifying bridges or uplink bonds or uplink load balancing, use the **manage\_ovs** commands and add the **--host** flag to the **manage\_ovs** commands as shown in the following example:

Note: If deployment of default virtual switch vs0 fails for AHV CO node with AHV SO node, the Prism Element, Prism Central or CLI workflows for virtual switch management are unavailable to manage the bridges and bonds. Use the **manage\_ovs** command options to manage the bridges and bonds.

```
nutanix@cvm$ manage_ovs --host IP_address_of_co_node --bridge_name bridge_name  
create_single_bridge
```

Replace **IP\_address\_of\_co\_node** with the IP address of the CO node and **bridge\_name** with the name of bridge you want to create.

Note: Run the **manage\_ovs** commands for an AHV CO node from any Controller VM running on an AHV HCI node.

Perform the networking tasks for each AHV CO node in the cluster separately.

For more information about networking configuration of the AHV hosts, see [Host Network Management](#) on page 51.

## Deployment of an AHV CO Node

You can use a supported server or an existing AHV Hyperconverged Infrastructure (HCI) node as an AHV CO node.

### New Node as CO Node

To add a new node as a CO node to the cluster, you must:

- Image the node as CO by using Foundation. For more information about how to image a node as a CO node, see the [Field Installation Guide](#).
- Add that node to the cluster by using the Prism Element web console.



## Existing HCI Node as CO Node

To add an existing HCI node, that is already a part of the cluster, as a CO node to the cluster, you must:

- Remove that node from the cluster. For more information about how to remove a node, see [Modifying a Cluster](#) section in *Prism Web Console Guide*.
- Image that node as CO by using Foundation.
- Add that node back to the cluster.

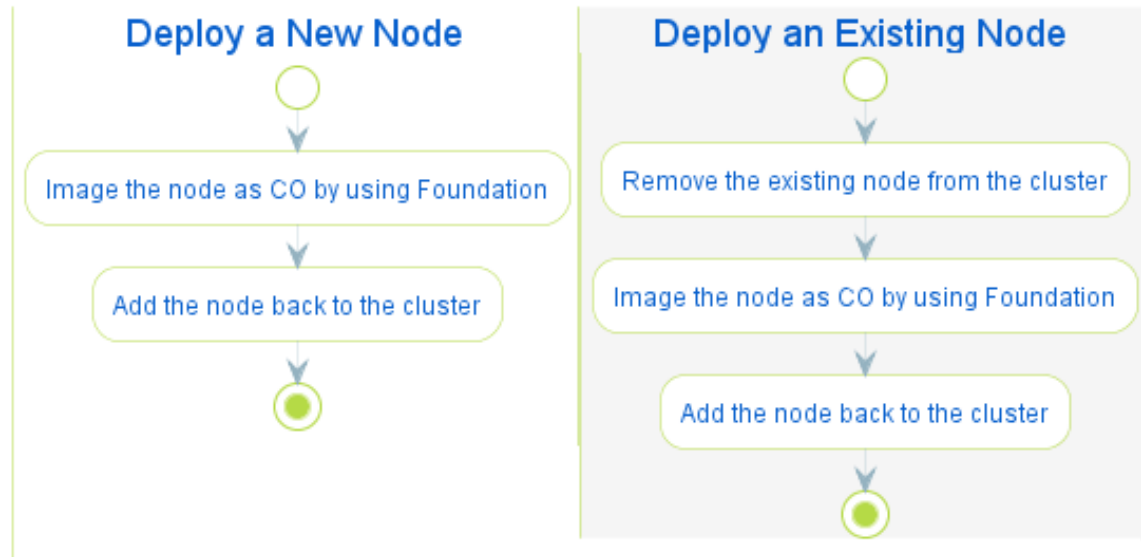


Figure 3: Deployment Workflow of an AHV CO Node

## Adding an AHV Compute-only Node to an AHV Cluster

### About this task

Perform the following procedure to add a Compute-Only (CO) node to an AHV cluster.

### Before you begin

Ensure that the following prerequisites are met before you add a compute-only node to an AHV cluster:

- Observe the requirements and restrictions listed in [AHV Compute-only with AHV HCI Nodes](#) on page 43 and [AHV Compute-only with AHV Storage-only Nodes](#) on page 45.
- Log on to CVM using SSH, and disable all the virtual switches including the default virtual switch (vs0), using the following command:

```
nutanix@cvm:~$ acli net.disable_virtual_switch virtual_switch=<virtual-switch-name>
```

In the above command, replace `<virtual-switch-name>` with the actual virtual switch name in your network.

### Procedure

To add a CO node to an AHV cluster, perform the following steps:



1. Log on to the Prism Element web console.
2. Do one of the following:
  - » Click the gear icon in the main menu and select **Expand Cluster** on the **Settings** page.
  - » Go to the hardware dashboard and click **Expand Cluster**.

The system displays the **Expand Cluster** window:

3. Select **Expand Cluster** to expand the cluster with CO node.

Note: To expand a cluster with CO node, do not select **Prepare Now and Expand Later**. This option is used to only prepare the nodes and expand the cluster at a later point in time. For CO nodes, node preparation is not supported.

The system displays the error **Compute only nodes cannot be prepared** in the **Configure Host** tab, if you proceed with **Prepare Now and Expand Later** option:

4. In the **Select Host** tab, scroll down and, under **Manual Host Discovery**, click **Discover Hosts Manually**.
5. Click **Add Host**.
6. Under **Host or CVM IP**, type the IP address of the AHV host and click **Save**.

Note: The CO node does not have a Controller VM and you must therefore provide the IP address of the AHV host.

7. Click **Discover and Add Hosts**.  
Prism Element discovers the CO node and the CO node appears in the list of nodes in the **Select Host** tab.
8. Select a CO node to view the node details, and click **Next**.  
The system displays the **Choose Node Type** tab.
9. Click **Next** in the **Choose Node Type** tab.  
The system prompts you to skip host networking.
10. Click **Skip Host Networking**.  
The system prompts you to run checks and expand the cluster with the selected CO node.



11. Click either of the following options in the **Configure Host** tab:

- **Run Checks** - Used to only run pre-checks required for cluster expansion. Once all pre-checks are successful, you can click the **Expand Cluster** to add the CO node to the cluster.
- **Expand Cluster** - Used to run both; pre-checks required for cluster expansion and expand cluster operation together.

The add-node process begins, and Prism Element performs a set of checks before the node is added to the cluster. Once all checks are completed and the node is added successfully, the system displays the completion states for the tasks as **100%**.

Note:

- You can check the progress of the operation in the **Tasks** menu of the Prism Element web console. The operation takes approximately five to seven minutes to complete.
- If you have not disabled the virtual switch as specified in [Prerequisites](#), the system displays the multiple errors during cluster expansion.

Check the progress of the operation in the **Tasks** menu of the Prism Element web console. The operation takes approximately five to seven minutes to complete.

12. Check the **Hardware Diagram** view to verify if the CO node is added to the cluster.

You can identify a node as a CO node if the Prism Element web console displays **N/A** in the **CVM IP** field.

Important: Virtual switch configuration is not supported when there are CO nodes in the cluster. The system displays the error message if you attempt to reconfigure the virtual switch for the cluster, using the following command:

```
nutanix@cvm:~$ acli net.migrate_br_to_virtual_switch br0 vs_name=vs0
```

Virtual switch configuration is not supported when there are Compute-Only nodes in the cluster.

# HOST NETWORK MANAGEMENT

---

Network management in an AHV cluster consists of the following tasks:

- Configuring Layer 2 switching through virtual switch and Open vSwitch bridges. When configuring virtual switch vSwitch, you configure bridges, bonds, and VLANs.
- Optionally changing the IP address, netmask, and default gateway that were specified for the hosts during the imaging process.

Flow Virtual Networking, enabled and managed in Prism Central instance that manages the AHV clusters, also provides networking support to the AHV clusters. Powered by the Network Controller and Network Gateway appliance, Flow Virtual Networking drives network virtualization to offer a seamless network experience with enhanced security. It is a software-defined network virtualization solution providing overlay capabilities for the on-premises AHV clusters. It integrates tools to deploy networking features like Virtual LANs, Virtual Private Cloud (VPC), Virtual Private Network (VPN), Layer 2 Virtual Network Extension using VPN or VTEP, Border Gateway Protocol sessions to support flexible app-driven networking that focuses on VMs and applications.

## **Network Controller**

The Network Controller is defined as networking component of Prism Central that manages and controls configuration, monitoring and optimization of network resources for Flow Virtual Networking. It provides programmability, automation, and centralized control for configuring and managing network flows.

Network Controller is necessary to use centralized VLAN management, Flow Virtual Networking and .Flow Network Security 2.0.

## **Network Gateway**

The Network Gateway appliance is deployed along with the Network Controller when you install or upgrade Prism Central. It is used to create VPN, VTEP, or BGP gateways to connect subnets using VPN connections, Layer 2 subnet extensions over VPN or VTEP, or over BGP sessions.

For more information about Flow Virtual Networking, see the [Flow Virtual Networking Guide](#).

## Network Types

### **AHV Networks**

#### **VLAN Basic Subnets (or Basic VLANs)**

VLAN Basic Subnets, also known as Basic VLANs, refer to the AHV networking based VLANs that Acropolis creates while creating the AHV clusters (VLAN0 - default VLAN that is used to network the Controller VMs and AHV hosts) or the AHV networking based VLANs that you create to network the guest VMs using the Network Configuration page in Prism Element Web Console.

VLAN Basic Subnets (Virtual LANs or networks) supported by AHV are of two types:

#### **Virtual Networks (Layer 2)**

Each VM network interface is bound to a virtual network. Each virtual network is bound to a single VLAN; trunking VLANs to a virtual network is not supported. Networks are designated by the Layer 2 type (**vlan**) and the VLAN number.



By default, each virtual network maps to virtual switch such as the default virtual switch **vs0**. However, you can change this setting to map a virtual network to a custom virtual switch. The user is responsible for ensuring that the specified virtual switch exists on all hosts, and that the physical switch ports for the virtual switch uplinks are properly configured to receive VLAN-tagged traffic.

For more information about virtual switches, see [About Virtual Switch](#) on page 64.

A VM NIC must be associated with a virtual network. You can change the virtual network of a vNIC without deleting and recreating the vNIC.

### Managed Networks (Layer 3)

A virtual network can have an IPv4 configuration, but it is not required. A virtual network with an IPv4 configuration is a managed network; one without an IPv4 configuration is an unmanaged network. A VLAN can have at most one managed network defined. If a virtual network is managed, every NIC is assigned an IPv4 address at creation time.

A managed network can optionally have one or more non-overlapping DHCP pools. Each pool must be entirely contained within the network's managed subnet.

If the managed network has a DHCP pool, the NIC automatically gets assigned an IPv4 address from one of the pools at creation time, provided at least one address is available. Addresses in the DHCP pool are not reserved. That is, you can manually specify an address belonging to the pool when creating a virtual adapter. If the network has no DHCP pool, you must specify the IPv4 address manually.

All DHCP traffic on the network is rerouted to an internal DHCP server, which allocates IPv4 addresses. DHCP traffic on the virtual network (that is, between the guest VMs and the Controller VM) does not reach the physical network, and vice versa.

A network must be configured as managed or unmanaged when it is created. It is not possible to convert one to the other.

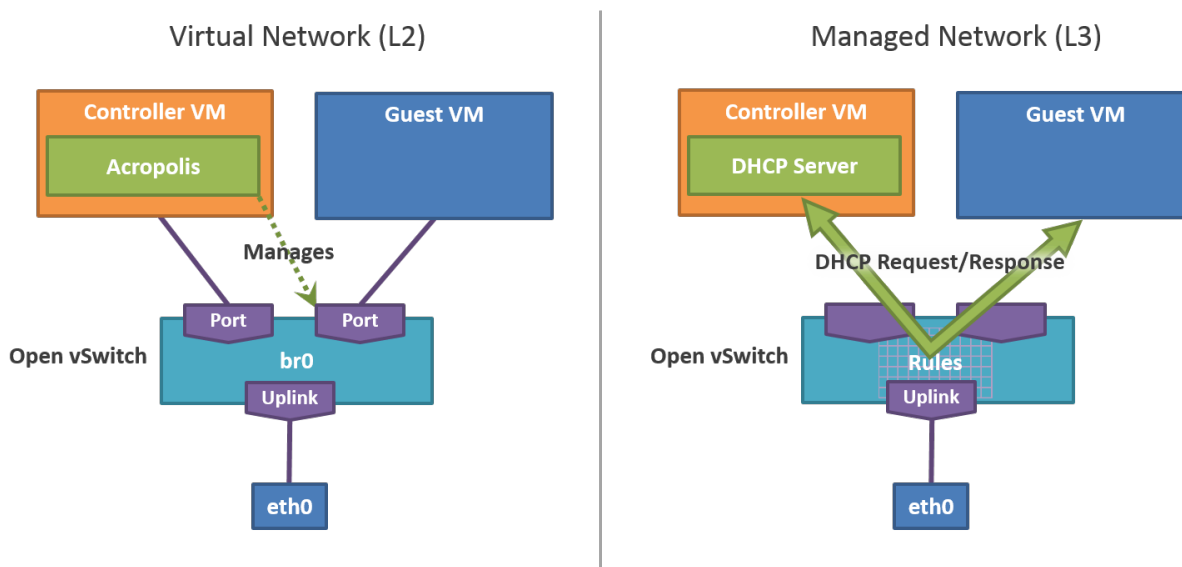


Figure 4: AHV Networking Architecture

### Network Controller based Networks on Prism Central

## Overlay Networks

You can create an IP-based Overlay subnet for a VPC. An Overlay network is a virtualized network that is configured on top of an underlying virtual or physical network. Examples of Overlay networks are:

- You can create a special purpose multicast network as an Overlay network within an existing network.
- A peer-to-peer network or a VPN.

An important assumption for an Overlay network is that the underlying network is fully connected. Nutanix provides the capability to create Overlay network-based VPCs.

For more information, see *Overlay networks* in [Essential Concepts](#).

## VLAN Networks

Starting with Prism Central pc.2023.3 with AOS 6.7 and AHV 20230302.198, Network Controller 3.0.0 and later versions support the creation of VLANs (VLAN Subnets) on the Flow Virtual Networking Network Controller. The Network Controller also supports migration of VLAN Basic Subnets to VLAN Subnets subject to support and limitations information provided in the *VLAN Subnets Support* section.

### VLAN Subnets (VLANs)

Create or manage the VLAN Subnets (VLANs) or Network Controller managed VLANs using the Network Controller on Prism Central. You can only create or manage these VLAN Subnets in Prism Central. You cannot use Prism Element Web Console to create or manage these VLAN Subnets.

Note: Clusters with CO nodes do not support the creation of VLAN Subnets.

For more information, see *VLANs (or VLAN Subnets)* in the [Flow Virtual Networking Guide](#).

## Prerequisites for Configuring Networking

Change the configuration from the factory default to the recommended configuration. See [AHV Networking Recommendations](#) on page 53.

## AHV Networking Recommendations

Nutanix recommends that you perform the following OVS configuration tasks from the Controller VM, as described in this documentation:

- Viewing the network configuration
- Configuring uplink bonds with desired interfaces using the Virtual Switch (VS) configurations.
- Assigning the Controller VM to a VLAN

For performing other network configuration tasks such as adding an interface to a bridge and configuring LACP for the interfaces in a bond, follow the procedures described in the [AHV Networking](#) best practices documentation.

Nutanix recommends that you configure the network as follows:



Table 5: Recommended Network Configuration

Network Component	Best Practice
Virtual Switch	<p>Do not modify the OpenFlow tables of any bridges configured in any VS configurations in the AHV hosts.</p> <p>Do not rename default virtual switch vs0. You cannot delete the default virtual switch vs0.</p> <p>Do not delete or rename OVS bridge br0.</p> <p>Do not modify the native Linux bridge virbr0.</p>
Switch Hops	<p>Nutanix nodes send storage replication traffic to each other in a distributed fashion over the top-of-rack network. One Nutanix node can, therefore, send replication traffic to any other Nutanix node in the cluster. The network should provide low and predictable latency for this traffic. Ensure that there are no more than three switches between any two Nutanix nodes in the same cluster.</p>
Switch Fabric	<p>A switch fabric is a single leaf-spine topology or all switches connected to the same switch aggregation layer. The Nutanix VLAN shares a common broadcast domain within the fabric. Connect all Nutanix nodes that form a cluster to the same switch fabric. Do not stretch a single Nutanix cluster across multiple, disconnected switch fabrics.</p> <p>Every Nutanix node in a cluster should therefore be in the same L2 broadcast domain and share the same IP subnet.</p>
WAN Links	<p>A WAN (wide area network) or metro link connects different physical sites over a distance. As an extension of the switch fabric requirement, do not place Nutanix nodes in the same cluster if they are separated by a WAN.</p>

Network Component	Best Practice
VLANs	<p>Add the Controller VM and the AHV host to the same VLAN. Place all CVMs and AHV hosts in a cluster in the same VLAN. By default the CVM and AHV host are untagged, shown as VLAN 0, which effectively places them on the native VLAN configured on the upstream physical switch.</p> <p>Note: Do not add any other device (including guest VMs) to the VLAN to which the CVM and hypervisor host are assigned. Isolate guest VMs on one or more separate VLANs.</p> <p>Nutanix recommends configuring the CVM and hypervisor host VLAN as the native, or untagged, VLAN on the connected switch ports. This native VLAN configuration allows for easy node addition and cluster expansion. By default, new Nutanix nodes send and receive untagged traffic. If you use a tagged VLAN for the CVM and hypervisor hosts instead, you must configure that VLAN while provisioning the new node, before adding that node to the Nutanix cluster.</p> <p>Use tagged VLANs for all guest VM traffic and add the required guest VM VLANs to all connected switch ports for hosts in the Nutanix cluster. Limit guest VLANs for guest VM traffic to the smallest number of physical switches and switch ports possible to reduce broadcast network traffic load. If a VLAN is no longer needed, remove it.</p>
Default VS bonded port (br0-up)	<p>Aggregate the fastest links of the same speed on the physical host to a VS bond on the default vs0 and provision VLAN trunking for these interfaces on the physical switch.</p> <p>By default, interfaces in the bond in the virtual switch operate in the recommended active-backup mode.</p> <p>Note: The mixing of bond modes across AHV hosts in the same cluster is not recommended and not supported.</p>

Network Component	Best Practice
1 GbE and 10 GbE interfaces (physical host)	<p>Ensure you do not form a NIC combination or mix NICs in any of the following ways in the same bond:</p> <ul style="list-style-type: none"> <li>NIC models from different vendors.</li> <li>NICs operating at different speeds.</li> <li>NICs with different drivers. To verify the NIC driver in use, run the following command on the AHV host for each NIC:</li> </ul> <pre>root@ahv# ethtool -i &lt;nic-name&gt;</pre> <p>In the above command, replace <b>&lt;nic-name&gt;</b> with the NIC name available at your site.</p> <p>If 10 GbE or faster uplinks are available, Nutanix recommends that you use them instead of 1 GbE uplinks.</p> <p>Recommendations for 1 GbE uplinks are as follows:</p> <ul style="list-style-type: none"> <li>If you plan to use 1 GbE uplinks, do not include them in the same bond as the 10 GbE interfaces.</li> <li>If you choose to configure only 1 GbE uplinks, when migration of memory-intensive VMs becomes necessary, power off and power on in a new host instead of using live migration. In this context, memory-intensive VMs are VMs whose memory changes at a rate that exceeds the bandwidth offered by the 1 GbE uplinks.</li> </ul> <p>Nutanix recommends the manual procedure for memory-intensive VMs because live migration, which you initiate either manually or by placing the host in maintenance mode, might appear prolonged or unresponsive and might eventually fail.</p> <p>Use the aCLI on any CVM in the cluster to start the VMs on another AHV host:</p> <pre>nutanix@cvm\$ acli vm.on vm_list host=host</pre> <p>Replace <b>vm_list</b> with a comma-delimited list of VM names and replace <b>host</b> with the IP address or UUID of the target host.</p> <ul style="list-style-type: none"> <li>If you must use only 1GbE uplinks, add them into a bond to increase bandwidth and use the balance-TCP (LACP) or balance-SLB bond mode.</li> </ul>
IPMI port on the hypervisor host	<p>Do not use VLAN trunking on switch ports that connect to the IPMI interface. Configure the switch ports as access ports for management simplicity.</p>



Network Component	Best Practice
Upstream physical switch	<p>Nutanix does not recommend the use of Fabric Extenders (FEX) or similar technologies for production use cases. While initial, low-load implementations might run smoothly with such technologies, poor performance, VM lockups, and other issues might occur as implementations scale upward (see Knowledge Base article <a href="#">KB1612</a>). Nutanix recommends the use of 10Gbps, line-rate, non-blocking switches with larger buffers for production workloads.</p> <p>Cut-through versus store-and-forward selection depends on network design. In designs with no oversubscription and no speed mismatches you can use low-latency cut-through switches. If you have any oversubscription or any speed mismatch in the network design, then use a switch with larger buffers. Port-to-port latency should be no higher than 2 microseconds.</p> <p>Use fast-convergence technologies (such as Cisco PortFast) on switch ports that are connected to the hypervisor host.</p>
Physical Network Layout	<p>Use redundant top-of-rack switches in a traditional leaf-spine architecture. This simple, flat network design is well suited for a highly distributed, shared-nothing compute and storage architecture.</p> <p>Add all the nodes that belong to a given cluster to the same Layer-2 network segment.</p> <p>Other network layouts are supported as long as all other Nutanix recommendations are followed.</p>
Jumbo Frames	<p>The Nutanix CVM uses the standard Ethernet MTU (maximum transmission unit) of 1,500 bytes for all the network interfaces by default. The standard 1,500 byte MTU delivers excellent performance and stability. Nutanix does not support configuring the MTU on network interfaces of a CVM to higher values.</p> <p>You can enable jumbo frames (MTU of 9,000 bytes) on the physical network interfaces of AHV hosts and guest VMs if the applications on your guest VMs require them. If you choose to use jumbo frames on hypervisor hosts, be sure to enable them end to end in the desired network and consider both the physical and virtual network infrastructure impacted by the change.</p> <p>If you try to configure, on the default virtual switch vs0, an MTU value that does not fall within the range of 1500 ~ 9000 bytes, Prism displays an error and fails to apply the configuration.</p>
Controller VM	<p>Do not remove the Controller VM from either the OVS bridge br0 or the native Linux bridge virbr0.</p>



Network Component	Best Practice
Rack Awareness and Block Awareness	Block awareness and rack awareness provide smart placement of Nutanix cluster services, metadata, and VM data to help maintain data availability, even when you lose an entire block or rack. The same network requirements for low latency and high throughput between servers in the same cluster still apply when using block and rack awareness.
<p>Note: Do not use features like block or rack awareness to stretch a Nutanix cluster between different physical sites.</p>	

Oversubscription	<p>Oversubscription occurs when an intermediate network device or link does not have enough capacity to allow line rate communication between the systems connected to it. For example, if a 10 Gbps link connects two switches and four hosts connect to each switch at 10 Gbps, the connecting link is oversubscribed. Oversubscription is often expressed as a ratio—in this case 4:1, as the environment could potentially attempt to transmit 40 Gbps between the switches with only 10 Gbps available. Achieving a ratio of 1:1 is not always feasible. However, you should keep the ratio as small as possible based on budget and available capacity. If there is any oversubscription, choose a switch with larger buffers.</p> <p>In a typical deployment where Nutanix nodes connect to redundant top-of-rack switches, storage replication traffic between CVMs traverses multiple devices. To avoid packet loss due to link oversubscription, ensure that the switch uplinks consist of multiple interfaces operating at a faster speed than the Nutanix host interfaces. For example, for nodes connected at 10 Gbps, the inter-switch connection should consist of multiple 10 Gbps or 40 Gbps links.</p>
------------------	--

The following diagrams show sample network configurations using Open vSwitch and Virtual Switch.

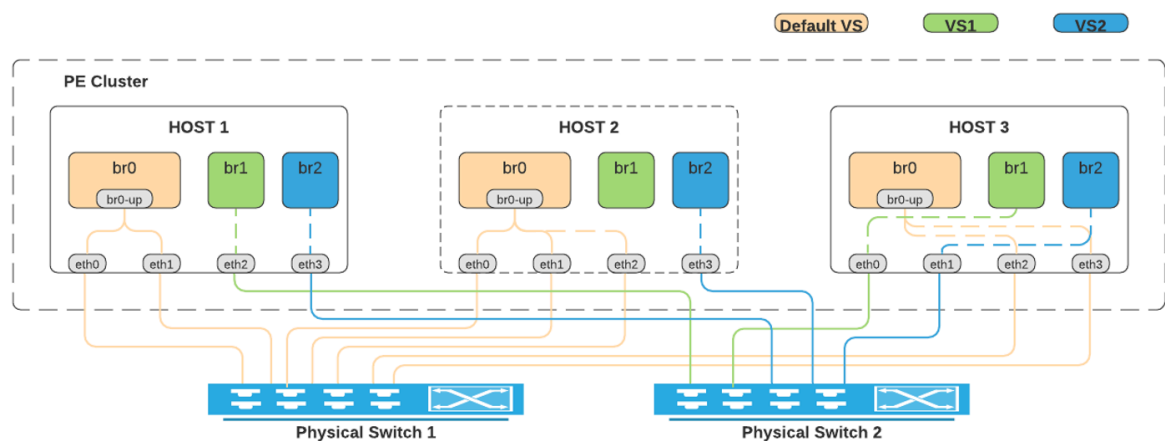


Figure 5: Virtual Switch

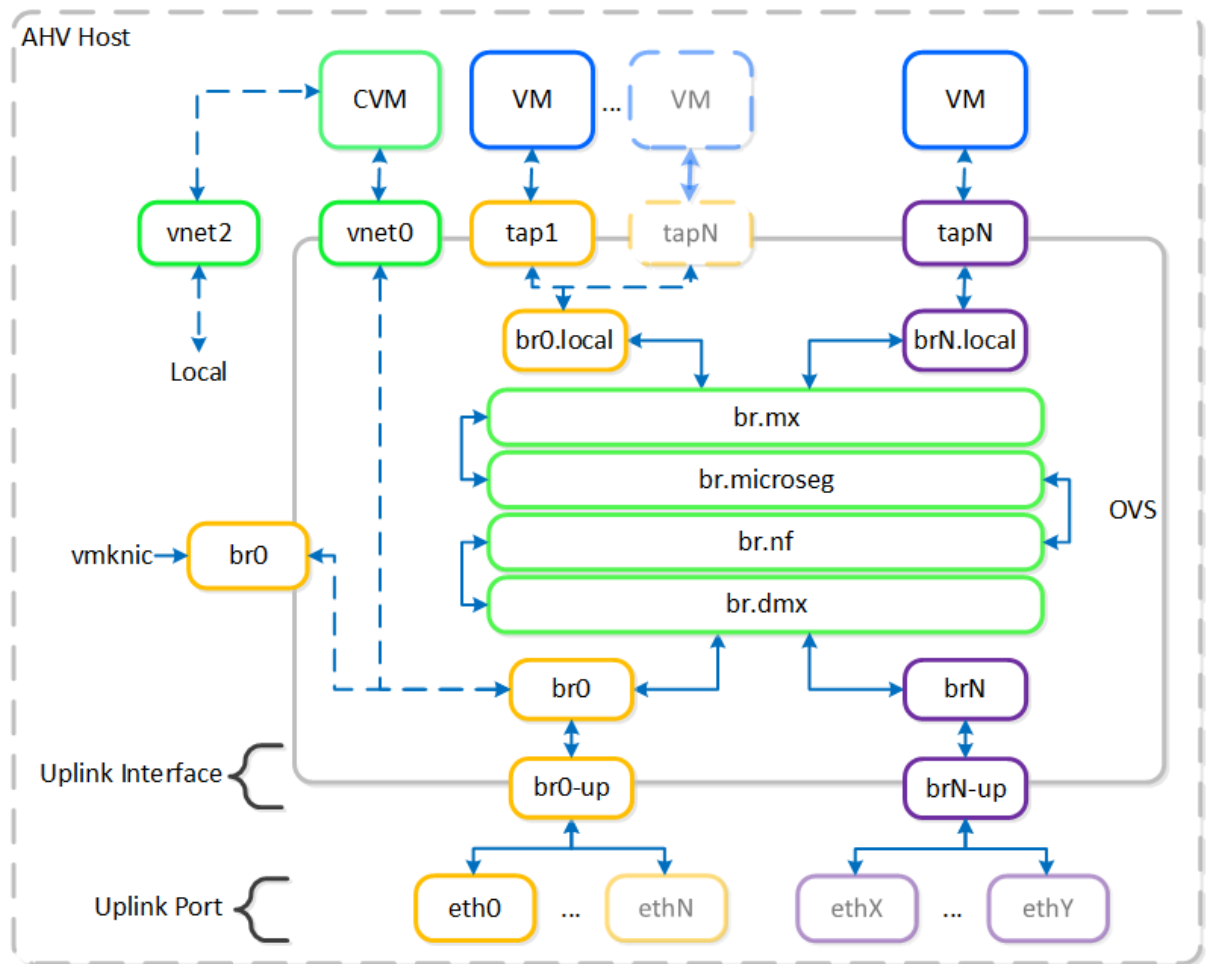


Figure 6: AHV Bridge Chain

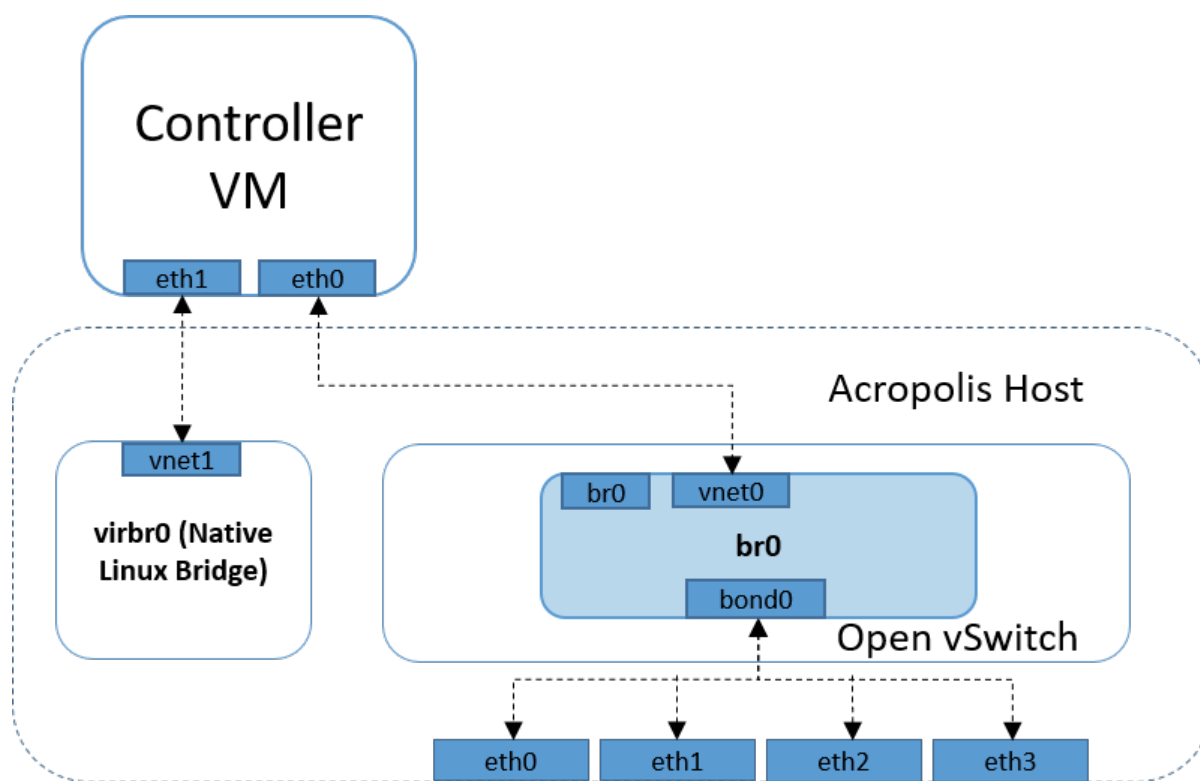


Figure 7: Default factory configuration of Open vSwitch in AHV

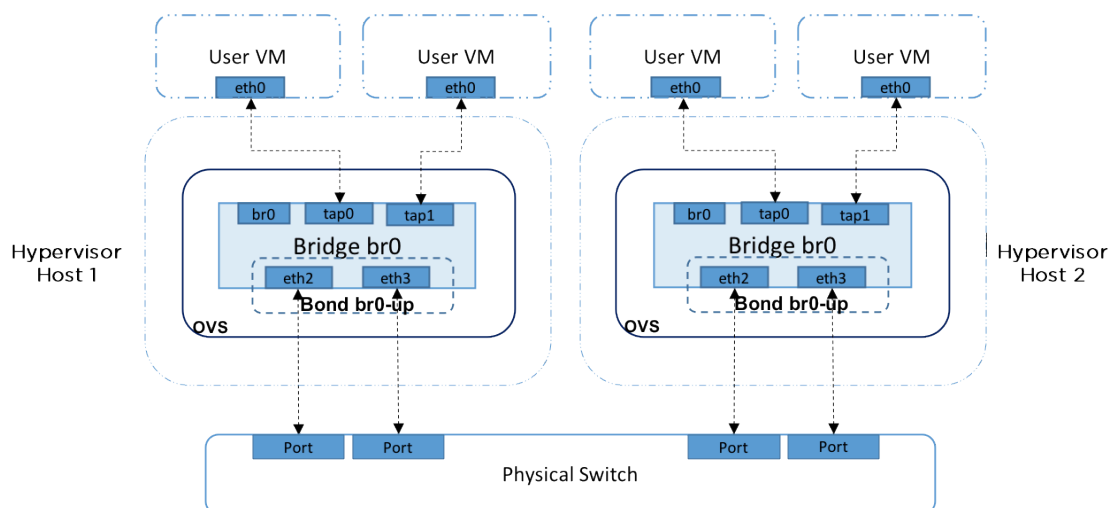


Figure 8: Open vSwitch Configuration

## IP Address Management

IP Address Management (IPAM) is a feature of AHV that allows it to assign IP addresses automatically to VMs by using DHCP. You can configure each virtual network with a specific IP address subnet, associated domain settings, and IP address pools available for assignment to VMs.

An AHV network is defined as a managed network or an unmanaged network based on the IPAM setting.

## Managed Network

Managed network refers to an AHV network in which IPAM is enabled.

## Unmanaged Network

Unmanaged network refers to an AHV network in which IPAM is not enabled or is disabled.

IPAM is enabled, or not, in the Create Network dialog box when you create a virtual network for Guest VMs. See [Configuring a Virtual Network for Guest VM Interfaces](#) topic in the *Prism Element Web Console Guide*.

Note: You can enable IPAM only when you are creating a virtual network. You cannot enable or disable IPAM for an existing virtual network.

IPAM enabled or disabled status has implications. For example, when you want to reconfigure the IP address of a Prism Central VM, the procedure to do so may involve additional steps for managed networks (that is, networks with IPAM enabled) where the new IP address belongs to an IP address range different from the previous IP address range. See [Reconfiguring the IP Address and Gateway of Prism Central VMs](#) in *Prism Central Infrastructure Guide*.

## Traffic Marking for Quality of Service

To prioritize outgoing (or egress) traffic as required, you can configure quality of service on the traffic for a cluster.

There are two distinct types of outgoing or egress traffic:

- Management traffic (mgmt)
- Data services (data-svc)

Data services traffic consists of the following protocols:

Table 6: Data Services Protocols

Protocol	Port	Nutanix Services
NFS	Source ports (TCP): 445, 2049, 20048, 20049, 20050, and 7508.	-Nutanix Files-
	Source ports (UDP): 2049, 20048, 20049, 20050, and 7508.	
	Source and Destination ports (TCP for Replicator-dr): 7515.	
SMB	Source ports (TCP): 445, 2049, 20048, 20049, 20050, and 7508.	-Nutanix Files-
	Source ports (UDP): 2049, 20048, 20049, 20050, and 7508.	
	Source and Destination ports (TCP for Replicator-dr): 7515.	



Protocol	Port	Nutanix Services
Cluster-to-cluster replications (external or inter-site)	Destination Ports: 2009 and 2020 on CVM.	Stargate and Cerebro
Node-to-node replications (internal or intra-site)	Destination Ports: 2009 and 2020 on CVM.	Stargate and Cerebro
iSCSI	Source Ports: 3260,3261,3205 on CVM.  Destination Ports: 3260,3261,3205 on AHV.	Nutanix Files and Volumes

Traffic other than data services traffic is management traffic. Traffic marking for QoS is disabled by default.

When you enable QoS, you can mark both the types of traffic with QoS values. AOS considers the values in hexadecimal even if you provide the values in decimal. When you view or get the QoS configuration enabled on the cluster, nCLI provides the QoS values in hexadecimal format (0xXX where XX is hexadecimal value in the range 00–3f).

Note: Set any QoS value in the range 0x0–0x3f. The default QoS values for the traffic are as follows:

- Management traffic (mgmt) = 0x10
- Data services (data-svc) = 0xa

### Configuring Traffic Marking for QoS

Configure Quality of Service (QoS) for management and data services traffic using nCLI.

#### About this task

To perform the following operations for QoS on the egress traffic of a cluster, use the nCLI commands in this section:

- Enable traffic marking for QoS on the cluster. QoS traffic marking is disabled by default.
- View or get the QoS configuration enabled on the cluster.
- Set QoS values for all traffic types or specific traffic types.
- Disable QoS on the cluster.

When you run any of the QoS configuration commands and the command succeeds, the console displays the following output indicating the successful command run:

```
QoSUpdateStatusDTO(status=true, message=null)
```

Where:

- **status=true** indicates that the command succeeded.
- **message=null** indicates that there is no error.

When you run any of the QoS configuration commands and the command fails, the console displays the following sample output indicating the failure:

```
QoSUpdateStatusDTO(status=false, message=QoS is already enabled.)
```



Where:

- **status=false** indicates that the command failed.
- **message=QoS is already enabled.** indicates why the command failed. This sample error message indicates that the **net enable-qos** command failed because QoS enable command was run again when QoS is already enabled.

## Procedure

- To enable QoS on a cluster, run the following command:

```
ncli> net enable-qos [data-svc="data-svc value"][mgmt="mgmt value"]
```

If you run the command as **net enable-qos** without the options, AOS enables QoS with the default values (**mgmt=0x10** and **data-svc=0xa**).

Note: After you run the **net enable-qos** command, if you run it again, the command fails and AOS displays the following output:

```
QoSUpdateStatusDT0(status=false, message=QoS is already enabled.)
```

Note: If you need to change the QoS values after you enable it, run the **net edit-qos** command with the option (**data-svc** or **mgmt** or both as necessary).

Note: Set any QoS value in the range 0x0-0x3f.

- To view or get the QoS configuration enabled on a cluster, run the following command:

```
ncli> net get-qos
```

Note: When you get the QoS configuration enabled on the cluster, nCLI provides the QoS values in hexadecimal format (0xXX where XX is hexadecimal value in the range 00-3f).

A sample output on the console is as follows:

```
QoSDT0(status=true, isEnabled=true, mgmt=0x10, dataSvc=0xa, message=null)
```

Note:

Where:

- **status=true** indicates that the **net get-qos** command passed. **status=false** indicates that the **net get-qos** command failed. See the **message=** value for the failure error message.
- **isEnabled=true** indicates that QoS is enabled. **isEnabled=false** indicates that QoS is not enabled.
- **mgmt=0x10** indicates that QoS value for Management traffic (**mgmt** option) is set to 0x10 (represented in hexadecimal value as **0x10**. If you disabled QoS, then this parameter is displayed as **mgmt=null**.
- **dataSvc=0xa** indicates that QoS value for data services traffic (**data-svc** option) is set to 0xa (represented in hexadecimal value as **0xa**. If you disabled QoS, then this parameter is displayed as **dataSvc=null**.
- **message=null** indicates there is no error message. **message=** parameter provides the command failure error message if the command fails.



- To set the QoS values for the traffic types on a cluster after you enabled QoS on the cluster, run the following command:

```
ncli> net edit-qos [data-svc="data-svc value"][mgmt="mgmt value"]
```

You can provide QoS values between 0x0-0x3f for one or both the options. The value is hexadecimal representation of a value between decimal 0-63 both inclusive.

- To disable QoS on a cluster, run the following command:

```
ncli> net disable-qos
```

```
QoSDTO(status=true, isEnabled=false, mgmt=null, dataSvc=null, message=null)
```

## Layer 2 Network Management

AHV uses virtual switch (VS) to connect the Controller VM, the hypervisor, and the guest VMs to each other and to the physical network. Virtual switch is configured by default on each AHV node and the VS services start automatically when you start a node.

To configure virtual networking in an AHV cluster, you need to be familiar with virtual switch. This documentation gives you a brief overview of virtual switch and the networking components that you need to configure to enable the hypervisor, Controller VM, and guest VMs to connect to each other and to the physical network.

### About Virtual Switch

Virtual switches or VS are used to manage multiple bridges and uplinks.

The VS configuration is designed to provide flexibility in configuring virtual bridge connections. A virtual switch (VS) defines a collection of AHV nodes and the uplink ports on each node. It is an aggregation of the same OVS bridge on all the compute nodes in a cluster. For example, vs0 is the default virtual switch is an aggregation of the br0 bridge and br0-up uplinks of all the nodes.

After you configure a VS, you can use the VS as reference for physical network management instead of using the bridge names as reference.

For overview about Virtual Switch, see [Virtual Switch Considerations](#) on page 67.

For information about OVS, see [About Open vSwitch](#) on page 71.

#### Virtual Switch Workflow

A virtual switch (VS) defines a collection of AHV compute nodes and the uplink ports on each node. It is an aggregation of the same OVS bridge on all the compute nodes in a cluster. For example, vs0 is the default virtual switch is an aggregation of the br0 bridge of all the nodes.

The system creates the default virtual switch vs0 connecting the default bridge br0 on all the hosts in the cluster during installation of or upgrade to the compatible versions of AOS and AHV. Default virtual switch vs0 has the following characteristics:

- The default virtual switch cannot be deleted.
- The default bridges br0 on all the nodes in the cluster map to vs0. thus, vs0 is not empty. It has at least one uplink configured.
- The default management connectivity to a node is mapped to default bridge br0 that is mapped to vs0.
- The default parameter values of vs0 - Name, Description, MTU and Bond Type - can be modified subject to aforesaid characteristics.





- The default virtual switch is configured with the **Active-Backup** uplink bond type.

For more information about bond types, see the [Bond Type](#) table.

The virtual switch aggregates the same bridges on all nodes in the cluster. The bridge (for example, br1) connects to the physical port such as eth3 (Ethernet port) via the corresponding uplink (for example, br1-up). The uplink ports of the bridges are connected to the same physical network. For example, the following illustration shows that vs0 is mapped to the br0 bridge, in turn connected via uplink br0-up to various (physical) Ethernet ports on different nodes.

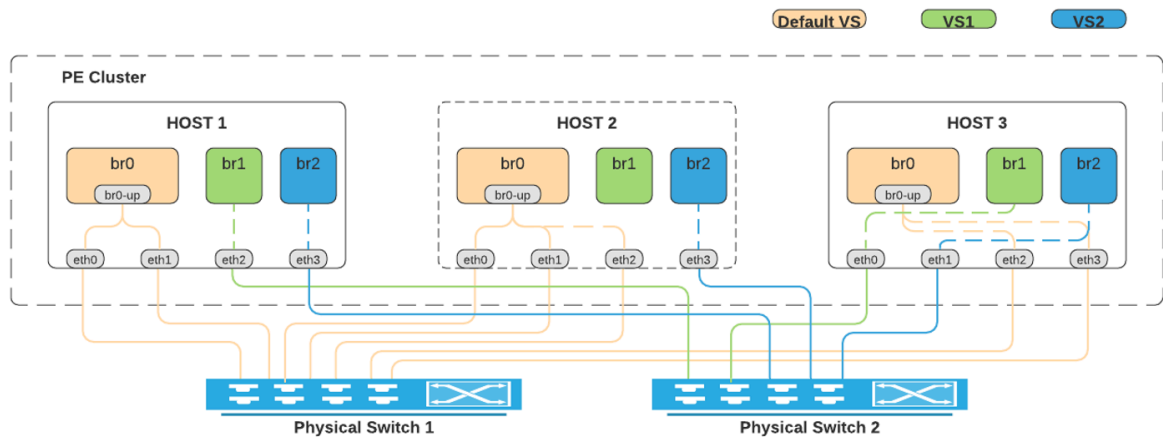


Figure 9: Virtual Switch

Uplink configuration uses bonds to improve traffic management. The bond types are defined for the aggregated OVS bridges. A new bond type - No uplink bond - provides a no-bonding option. A virtual switch configured with the No uplink bond uplink bond type has 0 or 1 uplinks.

When you configure a virtual switch with any other bond type, you must select at least two uplink ports on every node.

If you change the uplink configuration of vs0, AOS applies the updated settings to all the nodes in the cluster one after the other (the rolling update process). To update the settings in a cluster, AOS performs the following tasks when configuration method applied is **Standard**:

1. Puts the node in maintenance mode (migrates VMs out of the node)
2. Applies the updated settings
3. Checks connectivity with the default gateway
4. Exits maintenance mode
5. Proceeds to apply the updated settings to the next node

AOS does not put the nodes in maintenance mode when the **Quick** configuration method is applied.

Note:

Do not create, update or delete any virtual switch when the AOS or AHV upgrade process is running.

Table 7: Bond Types

Bond Type	Use Case	Maximum VM NIC Throughput	Maximum Host Throughput
Active-Backup	Recommended. Default configuration, which transmits all traffic over a single active adapter.	10 GB	10 GB
Active-Active with MAC pinning Also known as balance-slb	Works with caveats for multicast traffic. Increases host bandwidth utilization beyond a single 10 GB adapter. Places each VM NIC on a single adapter at a time. Do not use this bond type with link aggregation protocols such as LACP.	10 GB	20 GB
Active-Active Also known as LACP with balance-tcp	LACP and link aggregation required. Increases host and VM bandwidth utilization beyond a single 10 GB adapter by balancing VM NIC TCP and UDP sessions among adapters. Also used when network switches require LACP negotiation.  The default LACP settings are: <ul style="list-style-type: none"> <li>• Speed—Fast (1s)</li> <li>• Mode—Active fallback-active-backup</li> <li>• Priority—Default. This is not configurable.</li> </ul>	20 GB	20 GB
No Uplink Bond	No uplink or a single uplink on each host.  Virtual switch configured with the No uplink bond uplink bond type has 0 or 1 uplinks. When you configure a virtual switch with any other bond type, you must select at least two uplink ports on every node.	-	-

Note the following points about the uplink configuration.

- Virtual switches are not enabled in a cluster that has one or more compute-only nodes. See [Virtual Switch Limitations](#) on page 74 and [Virtual Switch Requirements](#) on page 73.



- If you select the **Active-Active** policy, you must manually enable LAG and LACP on the corresponding ToR switch for each node in the cluster.
- If you reimage a cluster with the **Active-Active** policy enabled, the default virtual switch (vs0) on the reimaged cluster is once again the **Active-Backup** policy. The other virtual switches are removed during reimage.
- Nutanix recommends configuring LACP with fallback to active-backup or individual mode on the ToR switches. The configuration and behavior varies based on the switch vendor. Use a switch configuration that allows both switch interfaces to pass traffic after LACP negotiation fails.

## Virtual Switch Considerations

### Virtual Switch Deployment

A VS configuration is deployed using rolling update of the clusters. After the VS configuration (creation or update) is received and execution starts, every node is first put into maintenance mode before the VS configuration is made or modified on the node. This is called the **Standard** recommended default method of configuring a VS.

You can select the **Quick** method of configuration also where the rolling update does not put the clusters in maintenance mode. The VS configuration task is marked as successful when the configuration is successful on the first node. Any configuration failure on successive nodes triggers corresponding NCC alerts. There is no change to the task status.

Note: If you are modifying an existing bond, AHV removes the bond and then re-creates the bond with the specified interfaces.

Ensure that the interfaces you want to include in the bond are physically connected to the Nutanix appliance before you run the command described in this topic. If the interfaces are not physically connected to the Nutanix appliance, the interfaces are not added to the bond.

Ensure that the pre-checks listed in *LCM Prechecks* section of the [Life Cycle Manager Guide](#) and the Always and Host Disruptive Upgrades types of pre-checks listed [KB-4584](#) pass for Virtual Switch deployments.

The VS configuration is stored and re-enforced at system reboot.

The VM NIC configuration also displays the VS details. When you Update VM configuration or Create NIC for a VM, the NIC details show the virtual switches that can be associated. This view allows you to change a virtual network and the associated virtual switch.

To change the virtual network, select the virtual network in the **Subnet Name** dropdown list in the Create NIC or Update NIC dialog box.



Create NIC?×

Subnet Name

dhcp

VLAN ID

59

Virtual Switch

vs0

Network Connection State

Connected

Private IP Assignment

Network address / prefix

NONE

Cancel

Add

Figure 10: Create VM - VS Details

Update VM?×

You haven't added any volume groups yet.

+ Add Volume Group

Network Adapters (NIC)+ Add New NIC

VLAN ID / VPC	VIRTUAL SWITCH	PRIVATE IP	MAC	
59 dhcp	vs0	-	... 98:23:73	

VM Host Affinity

You haven't pinned the VM to any hosts yet.

+ Set Affinity

Close

Save

Figure 11: VM NIC - VS Details

### Impact of Installation of or Upgrade to Compatible AOS and AHV Versions

See [Virtual Switch Requirements](#) on page 73 for information about minimum and compatible AOS and AHV versions.

When you upgrade the AOS to a compatible version from an older version, the upgrade process:



- Triggers the creation of the default virtual switch vs0, which is mapped to bridge br0 on all the nodes.
- Validates bridge br0 and its uplinks for consistency in terms of MTU and bond-type on every node.

If valid, it adds the bridge br0 of each node to the virtual switch vs0.

If br0 configuration is not consistent, the system generates an NCC alert which provides the failure reason and necessary details about it.

The system migrates only the bridge br0 on each node to the default virtual switch vs0 because the connectivity of bridge br0 is guaranteed.

- Does not migrate any other bridges to any other virtual switches during upgrade. You need to manually migrate the other bridges after install or upgrade is complete.

Note:

Do not create, update or delete any virtual switch when the AOS or AHV upgrade process is running.

## Bridge Migration

After upgrading to a compatible version of AOS, you can migrate bridges other than br0 that existed on the nodes. When you migrate the bridges, the system converts the bridges to virtual switches.

See *Virtual Switch Migration Requirements* in [Virtual Switch Requirements](#) on page 73.

Note: You can migrate only those bridges that are present on every compute node in the cluster. See [Migrating Bridges after Upgrade](#) topic in *Prism Element Web Console Guide*.

## Cluster Scaling Impact

VS management for cluster scaling (addition or removal of nodes) is seamless.

### Node Removal

When you remove a node, the system detects the removal and automatically removes the node from all the VS configurations that include the node and generates an internal system update. For example, a node has two virtual switches, vs1 and vs2, configured apart from the default vs0. When you remove the node from the cluster, the system removes the node for the vs1 and vs2 configurations automatically with internal system update.

### Node Addition

When you add a new node or host to a cluster, the bridges or virtual switches on the new node are treated in the following manner:

Note: If a host already included in a cluster is removed and then added back, it is treated as a new host.

- The system validates the default bridge br0 and uplink bond br0-up to check if it conforms to the default virtual switch vs0 already present on the cluster.

If br0 and br0-up conform, the system includes the new host and its uplinks in vs0.

If br0 and br0-up do not conform, then the system generates an NCC alert.



- The system does not automatically add any other bridge configured on the new host to any other virtual switch in the cluster.

It generates NCC alerts for all the other non-default virtual switches.

- You can manually include the host in the required non-default virtual switches. Update a non-default virtual switch to include the host.

For information about updating a virtual switch in Prism Element Web Console, see the [Configuring a Virtual Network for Guest VM Interfaces](#) section in *Prism Element Web Console Guide*.

For information about updating a virtual switch in Prism Central, see the [Network Connections](#) section in the *Prism Central Infrastructure Guide*.

## VS Management

You can manage virtual switches from Prism Central or Prism Web Console. You can also use aCLI or REST APIs to manage them. See the [Acropolis API Reference](#) and [Command Reference](#) guides for more information.

You can also use the appropriate aCLI commands for virtual switches from the following list:

- `net.create_virtual_switch`
- `net.list_virtual_switch`
- `net.get_virtual_switch`
- `net.update_virtual_switch`
- `net.delete_virtual_switch`
- `net.migrate_br_to_virtual_switch`
- `net.disable_virtual_switch`

## About Open vSwitch

Open vSwitch (OVS) is an open-source software switch implemented in the Linux kernel and designed to work in a multiserver virtualization environment. By default, OVS behaves like a Layer 2 learning switch that maintains a MAC address learning table. The hypervisor host and VMs connect to virtual ports on the switch.

Each hypervisor hosts an OVS instance, and all OVS instances combine to form a single switch. As an example, the following diagram shows OVS instances running on two hypervisor hosts.



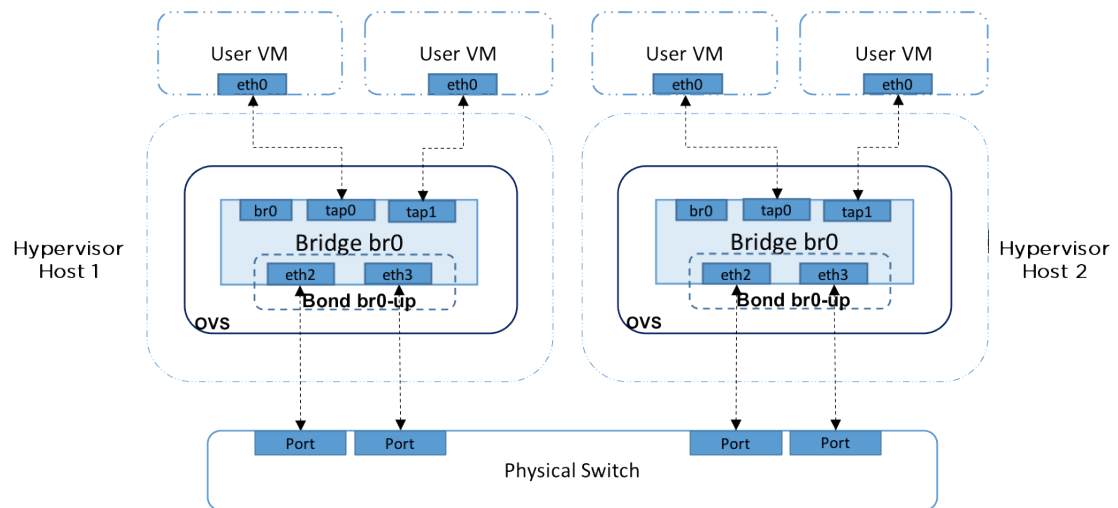


Figure 12: Open vSwitch

### Default Factory Configuration

The factory configuration of an AHV host includes a default OVS bridge named br0 (configured with the default virtual switch vs0) and a native Linux bridge called virbr0.

Bridge br0 includes the following ports by default:

- An internal port with the same name as the default bridge; that is, an internal port named br0. This is the access port for the hypervisor host.
- A bonded port named br0-up. The bonded port aggregates all the physical interfaces available on the node. For example, if the node has two 10 GbE interfaces and two 1 GbE interfaces, all four interfaces are aggregated on br0-up. This configuration is necessary for Foundation to successfully image the node regardless of which interfaces are connected to the network.

#### Note:

Before you begin configuring a virtual network on a node, you must disassociate the 1 GbE interfaces from the br0-up port. This disassociation occurs when you modify the default virtual switch (vs0) and create new virtual switches. Nutanix recommends that you aggregate only the 10 GbE or faster interfaces on br0-up and use the 1 GbE interfaces on a separate OVS bridge deployed in a separate virtual switch.

See [Virtual Switch Management](#) on page 75 for information about virtual switch management.

The following diagram illustrates the default factory configuration of OVS on an AHV node:



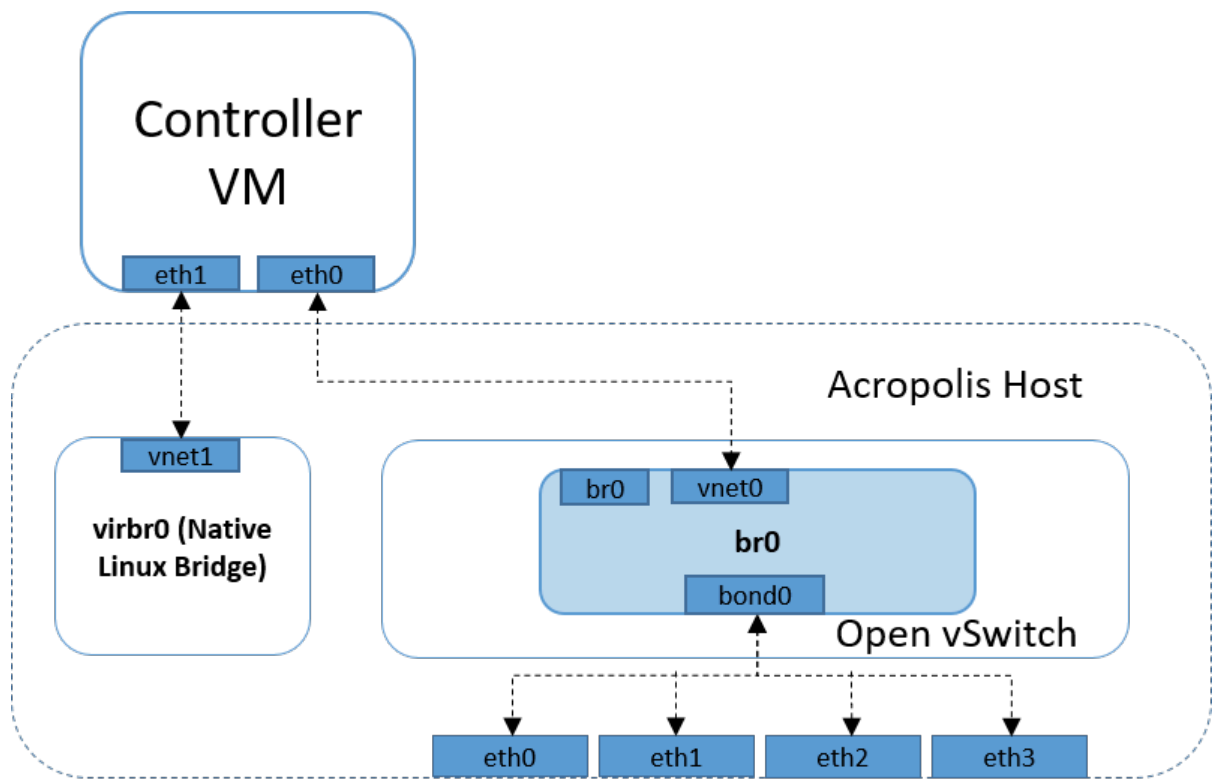


Figure 13: Default factory configuration of Open vSwitch in AHV

The Controller VM has two network interfaces by default. As shown in the diagram, one network interface connects to bridge br0. The other network interface connects to a port on virbr0. The Controller VM uses this bridge to communicate with the hypervisor host.

## Virtual Switch Requirements

The requirements to deploy virtual switches are as follows:

1. Virtual switches are supported on AOS 5.19 or later with AHV 20201105.12 or later. Therefore you must install or upgrade to AOS 5.19 or later, with AHV 20201105.12 or later, to use virtual switches in your deployments.
2. Virtual bridges used for a VS on all the nodes must have the same specification such as name, MTU and uplink bond type. For example, if vs1 is mapped to br1 (virtual or OVS bridge 1) on a node, it must be mapped to br1 on all the other nodes of the same cluster.

## Virtual Switch Migration Requirements

The AOS upgrade process initiates the virtual switch migration. The virtual switch migration is successful only when you meet the following requirements:

- Before migrating to Virtual Switch,
  - All bridge brO bond interfaces should have the same bond type on all hosts in the cluster. For example, all hosts should use the Active-backup bond type or balance-tcp. If some hosts use Active-backup and other hosts use balance-tcp, virtual switch migration fails.
  - If using LACP, confirm that the LACP speed on the physical switch is set to **fast** or **1** second. Also ensure that the switch ports are ready to fallback to individual mode if LACP negotiation fails due to a configuration such as **no lacp suspend-individual**.
  - Confirm that the upstream physical switch is set to **spanning-tree portfast** or **spanning-tree port type edge trunk**. Failure to do so might lead to a 30-second network timeout and the virtual switch migration might fail because it uses 20-second non-modifiable timer.
- Ensure that the pre-checks listed in *LCM Prechecks* section of the [Life Cycle Manager Guide](#) and the Always and Host Disruptive Upgrades types of pre-checks listed [KB-4584](#) pass for Virtual Switch deployments.
- For the default virtual switch vs0,
  - All configured uplink ports must be available for connecting the network. In Active-Backup bond type, the active port is selected from any configured uplink port that is linked. Therefore, the virtual switch vs0 can use all the linked ports for communication with other CVMs/hosts.
  - All the host IP addresses in the virtual switch vs0 must be resolvable to the configured gateway using ARP.

## Virtual Switch Limitations

### Virtual Switch Operations During Upgrade

Do not create, update or delete any virtual switch when the AOS or AHV upgrade process is running.

### MTU Restriction

The Nutanix Controller VM uses the standard Ethernet MTU (maximum transmission unit) of 1,500 bytes for all the network interfaces by default. The standard 1,500-byte MTU delivers excellent performance and stability. Nutanix does not support configuring higher values of MTU on the network interfaces of a Controller VM.

You can enable jumbo frames (MTU of 9,000 bytes) on the physical network interfaces of AHV, ESXi, or Hyper-V hosts and guest VMs if the applications on your guest VMs require such higher MTU values. If you choose to use jumbo frames on the hypervisor hosts, enable the jumbo frames end to end in the specified network, considering both the physical and virtual network infrastructure impacted by the change.

If you try to configure, on the default virtual switch vs0, an MTU value that does not fall within the range of 1500 ~ 9000 bytes, Prism displays an error and fails to apply the configuration.

### Single node and Two-node cluster configuration.

Virtual switch cannot be deployed if your single-node or two-node cluster has any instantiated user VMs. The virtual switch creation or update process involves a rolling restart, which checks for maintenance mode and whether you can migrate the VMs. On a single-node or two-node cluster, any instantiated user VMs cannot be migrated and the virtual switch operation fails.



Therefore, power down all user VMs for virtual switch operations in a single-node or two-node cluster.

### **Compute-only node is not supported.**

Virtual switch is not compatible with Compute-only (CO) nodes. If a CO node is present in the cluster, then the virtual switches are not deployed (including the default virtual switch). You need to use the `net.disable_virtual_switch` aCLI command to disable the virtual switch workflow if you want to expand a cluster which has virtual switches and includes a CO node.

The `net.disable_virtual_switch` aCLI command cleans up all the virtual switch entries from the IDF. All the bridges mapped to the virtual switch or switches are retained as they are.

See [AHV Compute-Only Node Configuration](#) on page 43.

### **Including a storage-only node in a VS is not necessary.**

Virtual switch is compatible with Storage-only (SO) nodes but you do not need to include an SO node in any virtual switch, including the default virtual switch.

### **Mixed-mode Clusters with AHV Storage-only Nodes**

Consider that you have deployed a mixed-node cluster where the compute-only nodes are ESXi or Hyper-V nodes and the storage-only nodes are AHV nodes. In such a case, the default virtual switch deployment fails.

Without the default VS, the Prism Element, Prism Central and CLI workflows for virtual switch required to manage the bridges and bonds are not available. You need to use the `manage_ovs` command options to update the bridge and bond configurations on the AHV hosts.

## **Virtual Switch Management**

Virtual Switch can be viewed, created, updated or deleted from both Prism Web Console as well as Prism Central.

### **Virtual Switch Views and Visualization**

For information on the virtual switch network visualization in Prism Element Web Console, see the [Network Visualization](#) topic in the *Prism Web Console Guide*.

### **Virtual Switch Create, Update and Delete Operations**

For information about the procedures to create, update and delete a virtual switch in Prism Element Web Console, see the [Configuring a Virtual Network for Guest VM Interfaces](#) information in the *Prism Element Web Console Guide*.

For information about the procedures to create, update and delete a virtual switch in Prism Central, see the [Network Connections](#) information in the *Prism Central Infrastructure Guide*.

#### **Note:**

Do not create, update or delete any virtual switch when the AOS or AHV upgrade process is running.

### **Uplinks for Virtual Private Cloud Traffic**

Starting with a minimum AOS version 6.1.1 with Prism Central version pc.2022.4 and Flow networking controller version 2.1.1, you can use virtual switches to separate traffic of the guest VMs that are networked using Flow Networking Virtual Private Cloud (VPC) configurations.

AHV uses the default virtual switch for the management and other Controller VM traffic (unless if you have configured network segmentation to route the Controller VM traffic on another



virtual switch). When you enable Flow networking in a cluster, Prism Central with the Flow networking controller and network gateway allows you to deploy Virtual Private Clouds (VPCs) that network guest VMs on hosts within the cluster and on other clusters. By default, AHV uses the default virtual switch vs0 for the VPC (Flow networking) traffic as well.

You can configure AHV to route the VPC (Flow networking) traffic on a different virtual switch, other than the default virtual switch.

### Conditions for VPC Uplinks

Certain conditions apply to the use of virtual switches to separate the Controller VM traffic and traffic of the guest VMs that are networked using Virtual Private Cloud (VPC) configurations.

### Host IP Addresses in Virtual Switch

The virtual switch selected for Flow networking VPC traffic must have IP addresses configured on the hosts. If the selected virtual switch does not have IP addresses configured on the hosts, then the following error is displayed:

**Bridge interface IP address is not configured for host: <host-UUID> on virtual\_switch <name-of-selected-virtual-switch>**

Configure IP addresses from this subnet on the hosts in the virtual switch.

**Note:**

Do not create, update or delete any virtual switch when the AOS or AHV upgrade process is running.

### Requirements

Ensure that the default virtual switch vs0 is enabled.

The following conditions apply to the IP addresses that you configure:

- Ensure that the host IP addresses in the subnet do not overlap with the primary IP addresses of the host configured during installation or the IP addresses used in any other configured virtual switches.
- Ensure that the host IP addresses in the subnet do not overlap with the IP addresses configured for the backplane operations (using network segmentation).
- Ensure that the host IP addresses configured on the hosts in the virtual switch is not a network IP address. For example, in a subnet 10.10.10.0/24, the network IP address of the subnet is 10.10.10.0. Ensure that this IP address (10.10.10.0) is not configured as a host IP address in the virtual switch. Failure message is as follows:

**Host IP address cannot be assigned equal to the subnet.**

- Ensure that the host IP addresses configured on the hosts in the virtual switch is not the broadcast IP address of the subnet. For example, in a subnet 10.10.10.0/24, the broadcast IP address of the subnet is 10.10.10.255. Ensure that this IP address (10.10.10.255) is not configured as a host IP address in the virtual switch. Failure message is as follows:

**Host IP address cannot be assigned equal to the subnet broadcast address.**

- Ensure that the subnet configured in the virtual switch has a prefix of /30 or less. For example, you can configure a subnet with a prefix of /30 such as 10.10.10.0/30, but not a subnet with prefix of /31 or /32 such as 10.10.10.0/31 or 10.10.10.0/32. Any subnet that you



configure in a virtual switch must have not less than 2 usable IP addresses. Failure message is as follows:

**Prefix length cannot be greater than 30.**

- Ensure that the host IP addresses configured in a virtual switch belongs to the same subnet. In other words, you cannot configure host IP addresses from two or more different subnets. For example, one host IP address is 10.10.10.10 from the subnet 10.10.10.0/24 and another host IP address is 10.100.10.10 from the subnet 10.100.10.0/24. This configuration fails. Both the hosts must have IP addresses from the 10.10.10.0/24 subnet (or both IP addresses must be from 10.100.10.0/24 subnet). Failure message is as follows:

**Different host IP address subnets found.**

- Ensure that the gateway IP address for the host IP addresses configured in a virtual switch belongs to the same subnet as host IP addresses. In other words, you cannot configure host IP addresses from one subnet while the gateway IP address of any of those host IP address is in a different subnet. Failure message is as follows:

**Gateway IP address is not in the same subnet.**

## Network Traffic Types

The network traffic is classified into the following types:

- *East/West (Intra-VPC) traffic* - Network traffic that is sent and received on the AHV host internal port br0 by default. The intra-VPC traffic is Geneve encapsulated and stays within the VPC. The intra-VPC traffic is also called East/West traffic because it is sent between nodes within the compute clusters.
- *North/South (ingress/egress) traffic*: Network traffic that enters or exits the VPC. The external network determines the virtual switch and VLAN for this traffic type.

The instructions specified in the [Configuring Virtual Switch for VPC Traffic Types](#) on page 77 section can isolate the network traffic to another virtual switch and VLAN if traffic separation from the default AHV and CVM VLAN is desired.

## Configuring Virtual Switch for VPC Traffic Types

Configure a new or existing non-default virtual switch for VPC traffic.

### About this task

**Important:** Perform this task only if you require physical network separation of VPC East/West (intra-VPC) or North/South (ingress/egress) traffic. If your site already meets this criteria, then do not perform the configuration mentioned in this section.

The default configuration places all VPC East/West traffic in the AHV VLAN, and all North/South traffic in the external network VLAN. For more information about network traffic types, see [Network Traffic Types](#) on page 77.

After you create a new virtual switch, the system automatically creates a new interface on all AHV hosts in the new virtual switch. Allocate one new IP address per AHV host to assign to the new interface.

**Note:** The assumption is to allocate the IP addresses (IP address that is allocated per AHV host to assign to the new interface) from the default VLAN available in the uplinks of the specific non-default bridge (br1). If you want to allocate the IP addresses from a different VLAN, then tag the required VLAN to the specific non-default bridge (br1).



Log on to the AHV host using the **root** account with SSH, and run the following command to tag the VLAN to the non-default bridge (br1):

```
root@ahv# ovs-vsctl set port brX tag= <VLAN-ID>
```

For more information, see [Assigning an AHV Host to a VLAN](#) on page 87.

When network segmentation is enabled for virtual switch vs1, the AHV internal interface terminates the Geneve encapsulated East/West traffic on internal port br1. The following figure shows the example where a new IP address is assigned to the port br1 on Virtual Switch vs1:

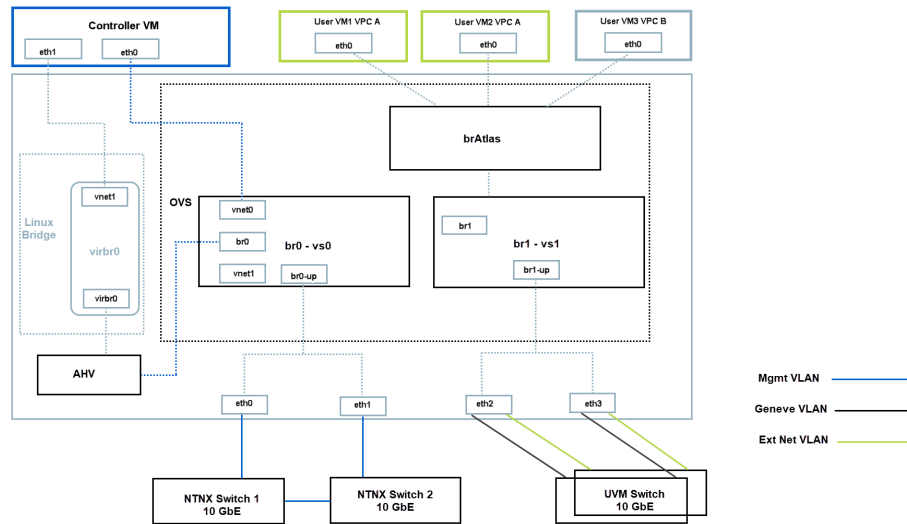


Figure 14: VPC Network Segmentation

## Before you begin

You need a virtual switch, other than the default virtual switch vs0, that can be used to route the VPC traffic to achieve optional physical traffic separation. Create a separate virtual switch that you can use to route the VPC traffic.

For information about the procedures to create or update a virtual switch in Prism Element web console, see the [Configuring a Virtual Network for Guest VMs](#) section in the *Prism Element Web Console Guide*.

For information about the procedures to create or update a virtual switch in Prism Central, see [Network Connections](#) in the *Prism Central Infrastructure Guide*.

Note: Do not create, update or delete any virtual switch when the AOS or AHV upgrade process is running.

## Procedure

To configure the uplinks for guest VMs networked by Flow networking VPCs, perform the following steps:

1. Create the virtual switch you want to use for VPC traffic. For example, create **vs1** as a virtual switch for the VPC traffic.

For more information, see [Creating or Updating a Virtual Switch](#) section in the *Prism Element Web Console Guide*.

- Using `acli`, run the following command to configure the IP addresses for the hosts that you have included in the virtual switch and a gateway IP address for the network:

```
nutanix@cvm$ acli net.update_virtual_switch virtual-switch-name host_ip_addr_config={host-  
uuid1:host_ip_address/prefix;host-uuid2:host_ip_address/prefix;host-uuid3:host_ip_address/  
prefix} gateway_ip_address=IP_address
```

For example, to update the host IP addresses and gateway IP address for virtual switch `vs1`, the sample command is as follows:

```
nutanix@cvm$ acli net.update_virtual_switch vs1 host_ip_addr_config={ebeae8d8-47cb-40d0-87f9-  
d03a762ffad7:10.XX.XX.15/24;731e24e0-d7a7-11ed-afa1-0242ac120002:10.XX.XX.16/24;88b1746a-  
d7a7-11ed-afa1-0242ac120002:10.XX.XX.17/24} gateway_ip_address=10.XX.XX.1
```

The options are:

- host\_ip\_addr\_config**=: Provide the host UUID and associated IP address with prefix as follows:

```
host_ip_addr_config={host-uuid1:host_ip_address/prefix}
```

Note: To retrieve the host UUID, run the following command from any CVM:

```
nutanix@cvm$ acli host.list
```

When there are more than one host on the virtual switch, use a semicolon separated list as follows:

```
host_ip_addr_config={host-uuid1:host_ip_address/prefix;host-uuid2:host_ip_address/  
prefix;host-uuid3:host_ip_address/prefix}
```

- gateway\_ip\_address**=: Provide the gateway IP address as follows:

```
gateway_ip_address=IP_address
```

Note: You can get the gateway IP from the network team at your site. The gateway IP is the same for all the nodes in the cluster.

- Set the virtual switch for use with Flow Virtual Networking VPCs using the following command:

```
nutanix@cvm$ acli net.set_vpc_east_west_traffic_config virtual_switch=virtual-switch-name
```

Note: When you run this command, if the virtual switch does not have IP addresses configured for the hosts, the command fails with an error message. For more information, see [Conditions for VPC Uplinks](#) on page 76.

For example, to set `vs1` as the virtual switch for VPC traffic, the sample command is as follows:

```
nutanix@cvm$ acli net.set_vpc_east_west_traffic_config virtual_switch=vs1
```

Note: You can set the value for the **permit\_all\_traffic**= option in the **net.set\_vpc\_east\_west\_traffic\_config** command to **true**, if you want to allow this virtual switch to pass traffic other than VPC traffic to the secondary IP of AHV host. For example, `ssh` and `snmp` are allowed through the selected virtual switch. The default value for this option is **false** which allows only GENEVE and ICMP traffic to pass through the selected virtual switch and blocks all traffic to the secondary IPs of the AHV Host.

Leave the default **permit\_all\_traffic=false** option if you only want to use the virtual switch for VPC traffic. Configure the **permit\_all\_traffic**= option with the value



**true** only when you want to allow traffic to the secondary IPs of the AHV host. For example, ssh and snmp are allowed through the selected virtual switch.

To update the virtual switch after it is already set, run the following command:

```
nutanix@cvm$ acli net.update_vpc_east_west_traffic_config virtual_switch=vs1
```

Configure the **permit\_all\_traffic=** option with the value **true** only when you want to allow all traffic types, including the non-VPC traffic, through the selected virtual switch. The value **false** blocks all traffic to the secondary IPs of the AHV host and allows only GENEVE and ICMP pings. To update the **permit\_all\_traffic=** option with the value **true**, run the following command:

```
nutanix@cvm$ acli net.update_vpc_east_west_traffic_config permit_all_traffic=true
```

4. Update the subnet to use the new virtual switch for the external traffic (or North/South traffic), on the Prism Central VM, using the following command:

```
nutanix@pcvm$ atlas_cli subnet.update external_subnet_name  
virtual_switch_uuid=virtual_switch_uuid
```

## What to do next

- To verify if the settings are made as required, run the **atlas\_config.get** command and check the output:

```
nutanix@cvm$ acli atlas_config.get
```

```
config {  
  anc_domain_name_server_list: "10.xxx.xxx.xxx"  
  dvs_physnet_mapping_list {  
    dvs_uuid: "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"  
    physnet: "physnet1"  
  }  
  enable_atlas_networking: True  
  logical_timestamp: 54  
  minimum_ahv_version: "20201105.2016"  
  ovn_cacert_path: "/home/certs/OvnController/ca.pem"  
  ovn_certificate_path: "/home/certs/OvnController/OvnController.crt"  
  ovn_privkey_path: "/home/certs/OvnController/OvnController.key"  
  ovn_remote_address: "ssl:anc-ovn-external.default.xxx.nutanix.com:6652"  
  vpc_east_west_traffic_config {  
    dvs_uuid: "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"  
    permit_all_traffic: True  
  }  
}
```

Where:

- **dvs\_physnet\_mapping\_list** provides details of the virtual switch.
- **vpc\_east\_west\_traffic\_config** provides the configuration for traffic with **permit\_all\_traffic** being **true**. It also provides the UUID of the virtual switch being used for traffic as **dvs\_uuid: "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"**.
- To verify the UUIDs of all virtual switches, run the following command:

```
nutanix@cvm$ acli net.get_virtual_switch *
```

Observe the following log information to check the uuids of the virtual switches:

```
default: True  
description: "Default Virtual Switch"
```





```
lACP_config {
lACP: True
lACP_fallback: True
lACP_timeout: "kFast"
}
mtu: 9000
name: "vs0"
nic_team_policy: "kBalanceTcp"
partially_done: False
update_in_progress: False
vswitch_uuid: "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
```

For more information about commands related to managed networks, see [Command Reference](#).

- VLAN tagging. See [VLAN Configuration](#) on page 87.

## Clearing, Disabling and Deleting the Virtual Switch

You can disable and delete a non-default virtual switch used for Flow networking VPC traffic.

### About this task

#### Before you begin

Before you delete a virtual switch that allows Flow networking VPC traffic, you must clear the virtual switch configuration that assigns the VPC traffic to that virtual switch. Use the `net.clear_vpc_east_west_traffic_config` command.

#### Note:

Do not create, update or delete any virtual switch when the AOS or AHV upgrade process is running.

### About this task

To disable or delete a virtual switch configured to manage Flow networking VPC traffic, do the following:

Note: After you clear the virtual switch settings using [step 1](#) on page 81, you can disable and delete the virtual switch in Prism Central.

### Procedure

1. Use the `net.clear_vpc_east_west_traffic_config` command to remove the settings on the virtual switch or switches (vs1 per example) configured for Flow networking VPC traffic.
2. Use the `net.disable_virtual_switch virtual_switch=<virtual-switch-name>` option to disable the virtual switch.
3. Use the `net.delete_virtual_switch virtual_switch=<virtual-switch-name>` option to delete the virtual switch.

#### Re-Configuring Bonds Across Hosts Manually

If you are upgrading AOS to 5.20, 6.0, or later versions, you need to migrate the existing bridges to virtual switches. If there are inconsistent bond configurations across hosts before the migration of the bridges, then after the migration of bridges, the virtual switches might not



be properly deployed. To resolve such issues, you must manually configure the bonds to make them consistent.

## Before you begin

Ensure that you meet the following prerequisites before you reconfigure the bonds:

- Place the affected AHV host where you want to reconfigure the bonds into maintenance mode.

Log on to any CVM using SSH, and run the following command:

```
nutanix@cvm$ acli host.enter_maintenance_mode hypervisor-IP-address [wait="{ true | false }" ] [non_migratable_vm_action="{ acpi_shutdown | block }" ]
```

Replace *hypervisor-IP-address* with either the IP address or host name of the AHV host you want to shut down.

The following are optional parameters for running the `acli host.enter_maintenance_mode` command:

- **wait:** Set the **wait** parameter to **true** to wait for the host evacuation attempt to finish.
- **non\_migratable\_vm\_action:** By default the **non\_migratable\_vm\_action** parameter is set to **block**, which means VMs with GPU, CPU passthrough, PCI passthrough, and host affinity policies are not migrated or shut down when you put a node into maintenance mode.

If you want to automatically shut down such VMs, set the **non\_migratable\_vm\_action** parameter to **acpi\_shutdown**.

For more information, see [Putting a Node into Maintenance Mode using CLI](#) on page 31.

- Check the Data Resiliency Status of the cluster to ensure the cluster is healthy and resilient to any brief interruptions to network connectivity during uplink changes. For more information, see [Home Dashboard](#) section in *Prism Web Console Guide*.

## About this task

Important:

- Perform the bond changes only on one host at a time. Ensure that you get the completed host out of maintenance mode before you proceed to work on any other affected hosts.
- Use this procedure only when you need to modify the inconsistent bonds in a migrated bridge across hosts in a cluster, that is preventing Acropolis (AOS) from deploying the virtual switch for the migrated bridge.

Do not use **ovs-vsctl** commands to make the bridge level changes. Use the **manage\_ovs** commands, instead.

The **manage\_ovs** command allows you to update the cluster configuration. The changes are applied and retained across host restarts. The **ovs-vsctl** command allows you to update the live running host configuration but does not update the AOS cluster configuration and the changes are lost at host restart. This behavior of



**ovs-vsctl** introduces connectivity issues during maintenance, such as upgrades or hardware replacements.

**ovs-vsctl** is typically used in cases where a host might be isolated on the network and requires a workaround to gain connectivity before the cluster configuration can actually be updated using **manage\_ovs**.

Note: Disable the virtual switch before you attempt to change the bonds or bridge.

If you face an issue where the virtual switch is automatically re-created after it is disabled (with AOS versions 5.20.0 or 5.20.1), follow steps 1 and 2 below to disable such an automatically re-created virtual switch again before migrating the bridges. For more information, see [KB-3263](#).

Be cautious when using the **disable\_virtual\_switch** command because it deletes all the configurations from the IDF, not only for the default virtual switch vs0, but also any virtual switches that you may have created (such as vs1 or vs2). Therefore, before you use the **disable\_virtual\_switch** command, ensure that you check a list of existing virtual switches, that you can get using the **acli net.get\_virtual\_switch** command.

Complete this procedure on each host Controller VM that is sharing the bridge that needs to be migrated to a virtual switch.

### Procedure

1. To list the virtual switch, use the following command.

```
nutanix@cvm$ acli net.list_virtual_switch
```

2. Disable the identified virtual switch.

```
nutanix@cvm$ acli net.disable_virtual_switch vs_name
```

Where **vs\_name** is the name of the virtual switch.

3. Change the bond type to align with the same bond type on all the hosts for the specified virtual switch

```
nutanix@cvm$ manage_ovs --bridge_name bridge-name --bond_name bond_name --bond_mode bond-type update_uplinks
```

Where:

- **bridge-name**: Provide the name of the bridge, such as **br0** for the virtual switch on which you want to set the uplink bond mode.
- **bond-name**: Provide the name of the uplink port such as **br0-up** for which you want to set the bond mode.
- **bond-type**: Provide the bond mode that you require to be used uniformly across the hosts on the named bridge.

Use the `manage_ovs --help` command for help on this command.

Note: To disable LACP, change the bond type from LACP Active-Active (balance-tcp) to Active-Backup/Active-Active with MAC pinning (balance-slb) by setting the **bond\_mode** using this command as **active-backup** or **balance-slb**.

Ensure that you turn off LACP on the connected ToR switch port as well. To avoid blocking of the bond uplinks during the bond type change on the host, ensure that you follow the ToR switch best practices to enable LACP fallback or passive mode.

To enable LACP, configure **bond-type** as **balance-tcp** (Active-Active) with additional variables **--lACP\_mode fast** and **--lACP\_fallback true**.

4. Exit the host from maintenance mode, using the following command:

```
nutanix@cvm$ acli host.exit_maintenance_mode hypervisor-IP-address
```

Replace **hypervisor-IP-address** with the IP address of the AHV host.

This command migrates (live migration) all the VMs that were previously running on the host back to the host. For more information, see [Exiting a Node from the Maintenance Mode Using CLI](#) on page 33.

5. Repeat the above steps for each host for which you intend to reconfigure bonds.
6. (If migrating to AOS version earlier than 5.20.2) Check if the issue in the [note](#) and disable the virtual switch.

## What to do next

After making the bonds consistent across all the hosts configured in the bridge, migrate the bridge or enable the virtual switch. For more information, see:

- [Configuring a Virtual Network for Guest VM Interfaces](#) in the *Prism Element Web Console Guide*.
- [Network Configuration](#) in the *Prism Central Infrastructure Guide*.

To check whether LACP is enabled or disabled, use the following command.

```
nutanix@cvm$ manage_ovs show_uplinks
```

## Enabling LACP and LAG (AHV Only)

This section describes the procedure to enable LAG and LACP in AHV nodes and the Top-of-Rack (ToR) switch or any switch that is directly connected to the Nutanix node.



## Procedure

To enable LACP and LAG, perform the following steps:

1. Login to Prism element and navigate to **Settings > Network Configuration > Virtual Switch**.  
You can also login to Prism Central, select the **Infrastructure** application from [Application Switcher Function](#), and navigate to **Network & Security > Subnets > Network Configuration > Virtual Switch** from the navigation bar.

The system displays the **Virtual Switch** tab.

2. Click the Edit icon (✎) for the target virtual switch on which you want to configure LACP and LAG.

The system displays the **Edit Virtual Switch** window.

3. In the **General** tab, choose **Standard (Recommended)** option in the **Select Configuration Method** field, and click **Next**.

Note: The **Standard** configuration method puts each node in maintenance mode before applying the updated settings. After applying the updated settings, the node exits from maintenance mode. For more information, see [Virtual Switch Workflow](#).

4. In the **Uplink** Configuration tab, select **Active-Active** in the **Bond Type** field, and click **Save**.

Note: The **Active-Active** bond type configures all AHV hosts with the fast setting for LACP speed, causing the AHV host to request LACP control packets at the rate of one per second from the physical switch. In addition, the **Active-Active** bond type configuration sets LACP fallback to **Active-Backup** on all AHV hosts. You cannot modify these default settings after you have configured them in Prism, even by using the CLI.

This completes the LAG and LACP configuration on the cluster. At this stage, cluster starts the Rolling Reboot operation for all the AHV hosts. Wait for the reboot operation to complete before you put the node and CVM in maintenance mode and change the switch ports.

For more information about how to manually perform the rolling reboot operation for an AHV host, see [Rebooting an AHV Node in a Nutanix Cluster](#).

Perform the following steps on each node, one at a time:

5. Put the node and the Controller VM into maintenance mode.

Note: Before you put a node in maintenance mode, see [Verifying the Cluster Health](#) and carry out the necessary checks.

The Step 6 in [Putting a Node into Maintenance Mode using Web Console](#) section puts the Controller VM in maintenance mode.

6. Change the settings for the interface on the switch that is directly connected to the Nutanix node to match the LACP and LAG settings made in the Edit Virtual Switch window above.

For more information about how to change the LACP settings of the switch that is directly connected to the node, refer to the vendor-specific documentation of the deployed switch.

Nutanix recommends you perform the following configurations for LACP settings on the switch:



Table 8: Nutanix Recommendations for LACP Settings

Nutanix Recommendations	Description
<p>Enable LACP fallback</p>	<p>Nutanix recommends you enable LACP fallback to set up a workaround for the port, using which the port establishes a link before the switch receives the LACP Bridge Protocol Data Units (BPDUs).</p> <p>The LACP fallback helps avoid link failures if either AHV host or switch that is connected to the AHV node does not negotiate LACP.</p> <p>LACP fallback provides seamless discovery of new nodes in an active or passive capacity setup and reduces the impact on the node operation. When LACP fallback is enabled, you can have a minimal business impact as VMs and applications remain healthy in case of an LACP status mismatch between the AHV host and the ToR switch port.</p> <p>As LACP fallback ensures connectivity during initial deployment, so it is crucial when you do not have LACP in discoveryOS.</p> <div data-bbox="800 905 1437 1123"> <p>Caution: When LACP fallback occurs, the port runs in fallback mode, and this might lead to an unbalanced utilization of ports and lack of redundancy in your site deployment. Based on your internal networking policies, you can decide whether LACP fallback is helpful for you, and enable or disable it.</p> </div>
<p>Consider the LACP time options (<i>slow</i> and <i>fast</i>)</p>	<p>If the switch has a fast configuration, Nutnaix recommends you set the LACP time to fast on AHV host.</p> <p>Nutanix recommends the LACP time to match on both; switch and AHV host, for L2 failure detection at the same time on the switch and AHV host. If the switch has a fast configuration, set the LACP time to fast on AHV host.</p> <p>When the LACP time setting matches on AHV host and switch, the detachment of a failed interface occurs at the same time, and both switch and the AHV host do not use the failed interface.</p> <p>When the LACP time is set to:</p> <ul style="list-style-type: none"> <li>• <i>fast</i> - Failure detection occurs faster within 3 seconds</li> <li>• <i>slow</i> - Failure detection occurs slowly and takes up to 90 seconds</li> </ul> <p>The matching LACP time helps to prevent the outage.</p>

7. Verify that LACP negotiation status is **Negotiated**.

Perform the SSH to the CVM as a nutanix user, and run the following commands:

```
nutanix@cvm$ ssh root@[AHV host IP] "ovs-appctl bond/show bond-name"
```

```
nutanix@cvm$ ssh root@[AHV host IP] "ovs-appctl lacp/show bond-name"
```

- Replace the following attributes in the above commands:
    - **bond-name** with the actual name of the uplink port such as br0-up in the above commands.
    - **[AHV host IP]** with the actual AHV host IP at your site.
  - Search for the string **negotiated** in the status lines.
8. Remove the node and Controller VM from maintenance mode. For more information, see [Exiting a Node from the Maintenance Mode using Web Console](#).

The Controller VM exits maintenance mode during the same process.

### What to do next

Do the following after completing the procedure to enable LAG and LACP in all the AHV nodes the connected ToR switches:

- Verify that the status of all services on all the CVMs are Up. Run the following command and check if the status of the services is displayed as **Up** in the output:

```
nutanix@cvm$ cluster status
```

- Log in to the Prism Element web console of the node and check the **Data Resiliency Status** widget displays **OK**.

## VLAN Configuration

You can set up a VLAN-based segmented virtual network on an AHV node by assigning the ports on virtual bridges managed by virtual switches to different VLANs. VLAN port assignments are configured from the Controller VM that runs on each node.

For best practices associated with VLAN assignments, see [AHV Networking Recommendations](#) on page 53. For information about assigning guest VMs to a virtual switch and VLAN, see [Network Connections](#) in the *Prism Central Infrastructure Guide*.

### Assigning an AHV Host to a VLAN

#### About this task

Note: Perform the following procedure during a scheduled maintenance window. Before you begin, stop the cluster. Once the process begins, hosts and CVMs partially lose network access to each other and VM data or storage containers become unavailable until the process completes.

To assign an AHV host to a VLAN, do the following on every AHV host in the cluster:

#### Procedure

1. Log on to the AHV host with SSH.



2. Put the AHV host and the CVM in maintenance mode.

See [Putting a Node into Maintenance Mode using CLI](#) on page 31 for instructions about how to put a node into maintenance mode.

3. Assign port br0 (the internal port on the default OVS bridge, br0 on default virtual switch vs0) to the VLAN that you want the host be on.

```
root@ahv# ovs-vsctl set port br0 tag=host_vlan_tag
```

Replace `host_vlan_tag` with the VLAN tag for hosts.

4. Confirm VLAN tagging on port br0.

```
root@ahv# ovs-vsctl list port br0
```

5. Check the value of the tag parameter that is shown.

6. Verify connectivity to the IP address of the AHV host by performing a ping test.

7. Exit the AHV host and the CVM from the maintenance mode.

See [Exiting a Node from the Maintenance Mode Using CLI](#) on page 33 for more information.

## What to do next

To remove the VLAN configuration for AHV Host, see [Removing VLAN Configuration for CVM and AHV Host](#) on page 90.

### Assigning the Controller VM to a VLAN

By default, the public interface of a Controller VM is assigned to VLAN 0. To assign the Controller VM to a different VLAN, change the VLAN ID of its public interface. After the change, you can access the public interface from a device that is on the new VLAN.

## About this task

Note: Perform the following procedure during a scheduled maintenance window. Before you begin, stop the cluster. Once the process begins, hosts and CVMs partially lose network access to each other and VM data or storage containers become unavailable until the process completes.

Note: To avoid losing connectivity to the Controller VM, do not change the VLAN ID when you are logged on to the Controller VM through its public interface. To change the VLAN ID, log on to the internal interface that has IP address 192.168.5.254.

Perform these steps on every Controller VM in the cluster. To assign the Controller VM to a VLAN, do the following:

## Procedure

1. Log on to the AHV host with SSH.
2. Put the AHV host and the Controller VM in maintenance mode.  
See [Putting a Node into Maintenance Mode using CLI](#) on page 31 for instructions about how to put a node into maintenance mode.

3. Check the Controller VM status on the host.

```
root@host# virsh list
```

An output similar to the following is displayed:

```
root@host# virsh list
```





Id	Name	State
1	NTNX-CLUSTER_NAME-3-CVM	running
3	3197bf4a-5e9c-4d87-915e-59d4aff3096a	running
4	c624da77-945e-41fd-a6be-80abf06527b9	running

```
root@host# logout
```

- Log on to the Controller VM (CVM).

```
root@host# ssh nutanix@192.168.5.254
```

Accept the host authenticity warning if prompted, and enter the CVM nutanix password.

- Assign the public interface of the Controller VM to a VLAN.

```
nutanix@cvm$ change_cvm_vlan vlan_id
```

Replace *vlan\_id* with the ID of the VLAN to which you want to assign the Controller VM.

For example, add the Controller VM to VLAN 201.

```
nutanix@cvm$ change_cvm_vlan 201
```

- Confirm VLAN tagging on the Controller VM.

```
root@host# virsh dumpxml cvm_name
```

Replace *cvm\_name* with the CVM name or CVM ID to view the VLAN tagging information.

Note: Refer to step 3 for Controller VM name and Controller VM ID.

An output similar to the following is displayed:

```
root@host# virsh dumpxml 1 | grep "tag id" -C10 --color
<target dev='vnet2'/>
<model type='virtio'/>
<driver name='vhost' queues='4'/>
<alias name='net2'/>
<address type='pci' domain='0x0000' bus='0x00' slot='0x05' function='0x00'/>
</interface>
<interface type='bridge'>
  <mac address='50:6b:8d:b9:0a:18'/>
  <source bridge='br0'/>
  <vlan>
    <tag id='201'/>
  </vlan>
  <virtualport type='openvswitch'>
    <parameters interfaceid='c46374e4-c5b3-4e6b-86c6-bfd6408178b5'/>
  </virtualport>
  <target dev='vnet0'/>
  <model type='virtio'/>
  <driver name='vhost' queues='4'/>
  <alias name='net3'/>
  <address type='pci' domain='0x0000' bus='0x00' slot='0x03' function='0x00'/>
</interface>
root@host#
```

- Check the value of the tag parameter that is shown.
- Restart the network service.

```
nutanix@cvm$ sudo service network restart
```



9. Verify connectivity to the Controller VMs external IP address by performing a ping test from the same subnet. For example, perform a ping from another Controller VM or directly from the host itself.
10. Exit the AHV host and the Controller VM from the maintenance mode.  
See [Exiting a Node from the Maintenance Mode Using CLI](#) on page 33 for more information.

### What to do next

To remove the VLAN configuration for CVM, see [Removing VLAN Configuration for CVM and AHV Host](#) on page 90.

#### Removing VLAN Configuration for CVM and AHV Host

This section describes how to remove the VLAN tags from AHV host and CVM, and revert to the default untagged configuration.

#### About this task

Perform the following procedure during a scheduled maintenance window. Before you begin, stop the cluster. Once the process begins, hosts and CVMs partially lose network access to each other and VM data or storage containers become unavailable until the process completes.

Note: To avoid losing connectivity to the Controller VM, do not remove the VLAN ID when you are logged on to the Controller VM through its public interface. To change the VLAN ID, log on to the internal interface that has IP address 192.168.5.254.

Important: Ensure that you have access to the out of band management interface (IPMI, ILO, iDRAC, etc.) for the hosts which helps you to recover the hosts more easily in case of any misconfiguration.

### Procedure

To remove the VLAN tagging from CVM and AHV host, perform the following steps:

1. Put the AHV host and the CVM in maintenance mode.  
See [Putting a Node into Maintenance Mode using CLI](#) on page 31 for instructions about how to put a node into maintenance mode.
2. Log on to the Controller VM using SSH.
3. Run the following command to remove the VLAN configuration for the Controller VM:  

```
nutanix@cvm$ change_cvm_vlan --remove
```
4. Log on to the AHV host with SSH.
5. Run the following command to remove the VLAN configuration for the AHV host:  

```
root@ahv# ovs-vsctl set port br0 tag=0
```
6. Exit the AHV host and the CVM from the maintenance mode.  
See [Exiting a Node from the Maintenance Mode Using CLI](#) on page 33 for more information.

### IGMP Snooping

On an AHV host, when multicast traffic flows to a virtual switch the host floods the Mcast traffic to all the VMs on the specific VLAN. This mechanism is inefficient when many of the VMs on the VLAN do not need that multicast traffic. IGMP snooping allows the host to track which VMs



on the VLAN need the multicast traffic and send the multicast traffic to only those VMs. For example, assume there are 50 VMs on VLAN 100 on virtual switch vs1 and only 25 VMs need to receive (hence, receiver VMs) the multicast traffic. Turn on IGMP snooping to help the AHV host track the 25 receiver VMs and deliver the multicast traffic to only the 25 receiver VMs instead of pushing the multicast traffic to all 50 VMs.

When IGMP snooping is enabled in a virtual switch on a VLAN, the ToR switch or router queries the VMs about the Mcast traffic that the VMs are interested in. When the switch receives a join request from a VM in response to the query, it adds the VM to the multicast list for that source entry as a receiver VM. When the switch sends a query, only the VMs that require the multicast traffic respond to the switch. The VMs that do not need the traffic do not respond at all. So, the switch does not add a VM to a multicast group or list unless it receives a response from that VM for the query.

Typically, in a multicast scenario, there is a source entity that casts the multicast traffic. This source may be another VM in this target cluster (that contains the target VMs that need to receive the multicast traffic) or another cluster connected to the target cluster. The host in the target cluster acts as the multicast router. Enable IGMP snooping in the virtual switch that hosts the VLAN connecting the VMs. You must also enable either the native Acropolis IGMP querier on the host or a separate third party querier must exist in the network. Check the documentation of your network vendor to enable an IGMP querier in the appropriate networks. The native Acropolis IGMP querier sends IGMP v2 query packets to the VMs.

The IGMP Querier sends out queries periodically to keep the multicast groups or lists updated. The periodicity of the query is determined by the IGMP snooping timeout value that you must specify when you enable IGMP snooping. For example, if you have configured the IGMP snooping timeout as 30 seconds then the IGMP Querier sends out a query every 15 seconds.

When you enable IGMP snooping and are using the native Acropolis IGMP querier, you must configure the IGMP VLAN list. The IGMP VLAN list is a list of VLANs that the native IGMP Querier must send the query out to. This list value is a comma-separated list of the VLAN IDs that the query needs to be sent to. If you do not provide a list of VLANs, then the native IGMP Querier sends the query to all the VLANs in the switch.

When a VM needs to receive the multicast traffic from specific multicast source, configure the multicast application on the VM to listen to the queries received by the VM from the IGMP Querier. Also, configure The multicast application on the VM to respond to the relevant query, that is, the query for the specific multicast source. The response that the application sends is logged by the virtual switch which then sends the multicast traffic to that VM instead of flooding it to all the VMs on the VLAN.

A multicast source always sends multicast traffic to a multicast group or list that is indicated by a multicast group IP address.

### **Enabling or Disabling IGMP Snooping**

IGMP snooping helps you manage multicast traffic to specific VMs configured on a VLAN.

### **About this task**

You can only enable IGMP snooping using aCLI.



## Procedure

- Run the following command:

```
net.update_virtual_switch virtual-switch-name enable_igmp_snooping=true
enable_igmp_querier=[true | false] igmp_query_vlan_list=VLAN IDs
igmp_snooping_timeout=timeout
```

Provide:

- virtual-switch-name**—The name of the virtual switch in which the VLANs are configured. For example, the name of the default virtual switch is **vs0**. Provide the name of the virtual switch exactly as it is configured.
- enable\_igmp\_snooping=[true | false]**—**true** to enable IGMP snooping. Provide **false** to disable IGMP snooping. The default setting is **false**.
- enable\_igmp\_querier=[true | false]**—**true** to enable the native IGMP querier. Provide **false** to disable the native IGMP querier. The default setting is **false**.
- igmp\_query\_vlan\_list=VLAN IDs**—List of VLAN IDs mapped to the virtual switch for which IGMP querier is enabled. When it's not set or set as an empty list, querier is enabled for all VLANs of the virtual switch.
- igmp\_snooping\_timeout=timeout**—An integer indicating time in seconds. For example, you can provide **30** to indicate IGMP snooping timeout of 30 seconds.

The default timeout is **300** seconds.

You can set the timeout in the range of **15** - **3600** seconds.

## What to do next

You can verify whether IGMP snooping is enabled or disabled by running the following command:

```
net.get_virtual_switch virtual-switch-name
```

The output of this command includes the following sample configuration:

```
igmp_config {
  enable_querier: True
  enable_snooping: True
}
```

The above sample shows that IGMP snooping and the native acropolis IGMP querier are enabled.

## Traffic Mirroring on AHV Hosts

Traffic Mirroring enables you to mirror traffic from the interfaces of the AHV hosts to the virtual NIC (vNIC) of guest VMs. Traffic Mirroring mirrors the packets from a set of source ports to a set of destination ports. You can mirror inbound, outbound, or bidirectional traffic flowing on a set of source ports. You can then use the mirrored traffic for security analysis and to gain visibility of traffic flowing through the set of source ports. Traffic Mirroring is a useful tool for troubleshooting packets and necessary for compliance.

Important: Configuring the Traffic Mirroring session on an AHV host using aCLI is disabled if the cluster is registered to Prism Central. Nutanix recommends that you create the Traffic Mirroring session on Prism Central. For more information, see [Creating a Traffic Mirroring Session](#) section in *Prism Central Infrastructure Guide*.



VLAN Subnets (VLAN constructs on the Network Controller stack for Flow Virtual Networking on Prism Central) support Traffic Mirroring. For information about VLAN Subnets, see [Network Types](#) in the *Flow Virtual Networking Guide*.

AHV supports the following types of source ports in a Traffic Mirroring session:

1. A bond port that is already mapped to a virtual switch (VS) such as vs0, vs1, or any other VS you have created.
2. A non-bond port that is already mapped to a VS such as vs0, vs1, or any other VS you have created.
3. An uplink port that is not assigned to any VS or bridge on the host.
4. A virtual port attached to a VM.

### Important Considerations

Consider the following before you configure Traffic Mirroring on AHV hosts:

- AHV supports mirroring of traffic from physical and virtual interfaces. Therefore, a source port could be physical or virtual interfaces.
- AHV supports Traffic Mirroring to destination guest VMs running on the same AHV host (to which the source ports belong) or on a remote AHV host.
- The Traffic Mirroring sessions and traffic forwarding are independent of the security policies configured on the Traffic Mirroring source and destination VMs.
- AHV supports different types of source ports in one session. For example, you can create a session with br0-up (bond port) and eth5 (single uplink port) on the same host as two different source ports in the same session. You can even have two different bond ports in the same session.
- AHV does not support Traffic Mirroring from a member of a bond port. For example, if you have a bond with eth0 and eth1 in virtual switch vs0, you cannot create a Traffic Mirroring session with either eth0 or eth1 as the source port.
- You cannot create a Traffic Mirroring session on an AHV host that is in the maintenance mode.
- Delete the Traffic Mirroring session before you delete the Traffic Mirroring destination VM or vNIC. Otherwise, the state of the Traffic Mirroring session is displayed as an error.
- If you move the uplink interface to another virtual switch, the Traffic Mirroring session fails. The system does not generate an alert in this situation.
- TCP Segmentation Offload can coalesce multiple packets belonging to the same TCP stream into a single one before being delivered to the Traffic Mirroring destination VM. With TCP Segmentation Offload enabled, there can be a difference between the number of packets received on the uplink interface and packets forwarded to the Traffic Mirroring destination VM (session packet count <= uplink interface packet count). However, the byte count at the Traffic Mirroring destination VM is closer to the number at the uplink interface.
- All the source and destination entities must run on the same AHV host. If you migrate a destination VM, delete a vNIC, or a physical port is down, the Traffic Mirroring session displays an error status.
- If all vNICs are migrated to the same host, the Traffic Mirroring session displays active status. However, if the session has physical NICs (pNICs), the Traffic Mirroring session displays an error status.
- You cannot enable or disable the Traffic Mirror sessions using aCLI.



- You cannot configure the direction of the traffic if the ports are not managed by Open vSwitch (OVS).
- Traffic mirroring is disabled if a cluster is connected to AWS and Azure.
- You can select source VM vNIC for a single Traffic Mirroring session.
- You can select destination VM vNIC for a single Traffic Mirroring session.

### Traffic Mirroring Session Scale

Traffic Mirroring session supports the following scale:

Table 9: Traffic Mirroring Session Scale

Entities	Scale
Source ports or entities	4 entities per Traffic Mirroring session
Destination ports or entities	2 entities per Traffic Mirroring session
Maximum number of Traffic Mirror sessions	2,000 per cluster
Maximum number of Active Sessions	2 per host

### Nutanix Recommendations for Traffic Mirroring

The following table provides the Nutanix recommendations for a successful Traffic Mirroring session based on the Traffic Mirroring setup criteria:

Table 10: Nutanix Recommendations - Traffic Mirroring

Traffic Mirroring Setup Criteria	Applicable AOS Release	Condition for successful Traffic Mirroring session	Nutanix Recommendations
<i>Source:</i> Host NICs and bonds  <i>Destination:</i> VM (Traffic Mirroring Destination VM/ Monitoring VM)	AOS 6.0 and later	The Traffic Mirroring destination VM (monitoring VM) must run on the same AHV host where the source ports are located.	Create or modify the Traffic Mirroring destination VM as an agent VM so that it is not migrated from the host. For information about how to create or modify a VM as an agent VM, see <a href="#">Creating a VM (AHV)</a> on page 113 or <a href="#">Managing a VM (AHV)</a> on page 119.

Traffic Mirroring Setup Criteria	Applicable AOS Release	Condition for successful Traffic Mirroring session	Nutanix Recommendations
<p><i>Source:</i> VM (Source VMs / Guest VMs)</p> <p><i>Destination:</i> VM (Traffic Mirroring Destination VM/ Monitoring VM)</p> <div> <p>Note:</p> <ul style="list-style-type: none"> <li>VM, Host NICs, and bonds can co-exist as sources, however the system supports up to a maximum of 4 sources and 2 Traffic Mirroring destination VMs in a Traffic Mirroring session.</li> <li>VM with only <i>kHostNic</i> and <i>kVmNic</i> types are supported and VM with <i>kDirectNIC</i> type is not supported.</li> </ul> </div>	AOS 6.5.2 and later	The Traffic Mirroring destination VM (monitoring VM) must run on the same AHV host where the source VMs are located.	<p>Set the VM-Host affinity policy for both source VMs (guest VMs) and Traffic Mirroring destination VM with the same host.</p> <p>For information about affinity policies, see <a href="#">Affinity Policies for AHV</a> on page 162.</p> <p>For information about how to create a VM-Host affinity policy, see <a href="#">Creating a VM (AHV)</a> on page 113 or <a href="#">Managing a VM (AHV)</a> on page 119.</p> <p>This helps you to have both guest VMs and Traffic Mirroring destination VM always reside on the same host when any of the following VM migration scenario occurs:</p> <ul style="list-style-type: none"> <li>VM migration to another host due to an HA event.</li> <li>VM migration to another host due to Nutanix admin-initiated change in host in VM-Host affinity policies for both source and Traffic Mirroring destination VM.</li> </ul> <p>After both source VMs and destination VMs migrate to the same host, the system restores the Traffic Mirroring session.</p>

Traffic Mirroring Setup Criteria	Applicable AOS Release	Condition for successful Traffic Mirroring session	Nutanix Recommendations
<p><i>Source:</i> VM (Source VMs / Guest VMs)</p> <p><i>Destination:</i> VM (Traffic Mirroring Destination VM/ Monitoring VM)</p> <div> <p>Note:</p> <ul style="list-style-type: none"> <li>• VM, Host NICs, and bonds can co-exist as sources, however the system supports up to a maximum of 4 sources and 2 Traffic Mirroring destination VMs in a Traffic Mirroring session.</li> <li>• VM with only <i>kHostNic</i> and <i>kVmNic</i> types are supported and VM with <i>kDirectNIC</i> type is not supported.</li> </ul> </div>	AOS 6.8 and later	The Traffic Mirroring destination VM (monitoring VM) might run on the same or different AHV hosts, from the AHV host(s) that the source VMs are located. If the list of sources includes Host NICs and bonds, then only the destination VMs that are located on the same AHV host where the source host NICs/bonds are located, receive the mirrored traffic.	None

### Configuring Traffic Mirroring on an AHV Host

To configure Traffic Mirroring on an AHV host, create a Traffic Mirroring destination vNIC where you assign that vNIC to a guest VM (Traffic Mirroring destination VM). After you create the vNIC, create a Traffic Mirroring session specifying the source and destination ports between which you want to run the Traffic Mirroring session.

### Before you begin

Ensure that the following prerequisites are met before you configure Traffic Mirroring on AHV host:

- Configuring the Traffic Mirroring session on an AHV host using aCLI is disabled if the cluster is registered to Prism Central. Nutanix recommends that you create the Traffic Mirroring session on Prism Central. For more information, see [Creating a Traffic Mirroring Session](#) section in *Prism Central Infrastructure Guide*.
- The guest VM that you want to configure as the Traffic Mirroring destination VM (monitoring VM) is created.
- The Nutanix recommendations described in [Nutanix Recommendations for Traffic Mirroring](#) on page 94 are followed.





## Procedure

To configure Traffic Mirroring on an AHV host, perform the following steps:

1. Log on to a Controller VM in the cluster with SSH.
2. Determine the name and UUID of the guest VM that you want to configure as the Traffic Mirroring destination VM.

```
nutanix@cvm$ acli vm.list
```

Example:

```
nutanix@cvm$ acli vm.list
VM name          VM UUID
traffic-mirror-dest-VM  85abfdd5-7419-4f7c-bffa-8f961660e516
```

In this example, **traffic-mirror-dest-VM** is the name and **85abfdd5-7419-4f7c-bffa-8f961660e516** is the UUID of the guest VM.

Note: If you delete the Traffic Mirroring destination VM without deleting the Traffic Mirroring session that you created with this Traffic Mirroring destination VM, the Traffic Mirroring session **State** displays **kError**.

3. Create a Traffic Mirroring destination vNIC for the guest VM.

```
nutanix@cvm$ acli vm.nic_create vm-name type=kSpanDestinationNic
```

Replace **vm-name** with the name of the guest VM on which you want to configure Traffic Mirroring session.

Note: Do not include any other parameter when you are creating a Traffic Mirroring destination vNIC.

Example:

```
nutanix@cvm$ acli vm.nic_create traffic-mirror-dest-VM type=kSpanDestinationNic
NicCreate: complete
```

Note: If you delete the Traffic Mirroring destination vNIC without deleting the Traffic Mirroring session that you created with this Traffic Mirroring destination vNIC, the Traffic Mirroring session **State** displays **kError**.

4. Determine the MAC address of the vNIC.

```
nutanix@cvm$ acli vm.nic_get vm-name
```

Replace **vm-name** with the name of the guest VM to which you assigned the vNIC.

Example:

```
nutanix@cvm$ acli vm.nic_get traffic-mirror-VM
50:6b:8d:de:c6:44 {
  connected: True
  mac_addr: "50:6b:8d:de:c6:44"
  network_type: "kNativeNetwork"
  type: "kSpanDestinationNic"
  uuid: "75e1582f-cb33-4b69-bd00-c0657cbc716a"
}
50:6b:8d:8b:2c:94 {
  connected: True
  mac_addr: "50:6b:8d:8b:2c:94"
  network_name: "vlan.124"
```



```

network_type: "kNativeNetwork"
network_uuid: "5128e6bf-3253-4db7-ab3b-c197a1b881eb"
type: "kNormalNic"
uuid: "ac752ec9-be79-4774-8ea9-a9ea7ea5e956"
vlan_mode: "kAccess"
}

```

Note that the MAC address (value of **mac\_addr**) of the vNIC for which **type** is set to **kSpanDestinationNic**.

- Determine the UUID of the host whose traffic you want to monitor by using Traffic Mirroring.

```
nutanix@cvm$ acli host.list
```

- Create a Traffic Mirroring session based on the following Traffic Mirroring setup criteria:

Note: VM, host NICs, and bonds can co-exist as sources, however the system supports a maximum of 4 sources and 2 Traffic Mirroring destination VMs in a Traffic Mirroring session.

Table 11: Traffic Mirroring Session Creation

Traffic Mirroring Setup Criteria	Traffic Mirroring Session Creation Command
<p><i>Source:</i> Host NICs and bonds</p> <p><i>Destination:</i> VM (Traffic Mirroring Destination VM/ Monitoring VM)</p>	<p>Run the following command:</p> <pre>nutanix@cvm\$ acli net.create_traffic_mirror traffic-mirror-session-name description="description-text" source_list=\{uuid=host-uuid,type=kHostNic,identifier=source-port-name,direction=traffic-type} dest_list=\{uuid=vm-uuid,type=kVmNic,identifier=vnic-mac-address}</pre> <p>For example,</p> <pre>nutanix@cvm\$ acli net.create_traffic_mirror traffic-mirror-1 description="traffic mirror session 1" source_list=\{uuid=492a2bda-ffc0-486a-8bc0-8ae929471714,type=kHostNic,identifier=br0-up,direction=kBiDir} dest_list=\{uuid=85abfdd5-7419-4f7c-bffa-8f961660e516,type=kVmNic,identifier=50:6b:8d:de:c6:44}</pre>

Traffic Mirroring Setup Criteria	Traffic Mirroring Session Creation Command
Source: VM (Source VMs / Guest VMs)	Run the following command:
Destination: VM (Traffic Mirroring Destination VM/ Monitoring VM)	<pre>nutanix@cvm\$ acli net.create_traffic_mirror traffic-mirror-session-name description="description-text" source_list=\{uuid=source-vm-uuid,type=kVmNic,identifier=source-vnic-mac-address,direction=traffic-type} dest_list=\{uuid=vm-uuid,type=kVmNic,identifier=vnic-mac-address}</pre>
	For example,
	<pre>nutanix@cvm\$ acli net.create_traffic_mirror traffic-mirror-1 description="traffic mirror session 1" source_list=\{uuid= dc2aef46-c326-4bdd-a9da-9f87e972ea3d,type=kVmNic,identifier= 50:6b:8d:d1:93:ba,direction=kBiDir} dest_list=\{uuid=85abfdd5-7419-4f7c-bffa-8f961660e516,type=kVmNic,identifier=50:6b:8d:de:c6:44}</pre>

Replace the variables mentioned in the command for the following parameters with their appropriate values:

- **traffic-mirror-session-name:** Replace `traffic-mirror-session-name` with a name for the session.
- **description** (Optional): Replace `description-text` with a description for the session.

Note:

All `source_list` and `dest_list` parameters are mandatory inputs. These parameters do not have default values. Provide an appropriate value for each parameter.

Source list parameters:

- **uuid:**
  - Replace `host-uuid` with the UUID of the host whose traffic you want to monitor by using Traffic Mirroring. (determined in *step 5*).
  - Replace `source-vm-uuid` with the UUID of the VM whose traffic you want to monitor by using Traffic Mirroring. You can use the same command as indicated in *Step 2* using the source VM name.
- **type:**
  - Specify `kHostNic` as the type if the source is Host NIC or bond.
  - Specify `kVmNic` as the type if source is a VM
- **identifier:**
  - Replace `source-port-name` with name of the source port whose traffic you want to mirror. For example, `br0-up`, `eth0`, or `eth1`.
  - Replace `source-vnic-mac-address` with the MAC address of the source VM for which you want to mirror the traffic. You can use the same command as described in *step 4* using the source VM name.
- **direction:** Replace `traffic-type` with **kIngress** if you want to mirror inbound traffic, **kEgress** for outbound traffic, or **kBiDir** for bidirectional traffic.



Destination list parameters:

- **uuid:** Replace `vm-uuid` with the UUID of the guest VM that you want to configure as the Traffic Mirroring destination VM. (determined in *step 2*).
- **type:** Specify `kVmNic` as the type. Only the `kVmNic` type is supported in this release.
- **identifier:** Replace `vnic-mac-address` with the MAC address of the destination port where you want to mirror the traffic (determined in *step 4*).

Note: The syntax for **source\_list** and **dest\_list** is as follows:

```
source_list/dest_list=[{key1=value1,key2=value2,...}]
```

Each pair of curly brackets includes the details of one source or destination port with a comma-separated list of the key-value pairs. There must not be any space between two key-value pairs and commas in-between the two key-value pairs.

One Traffic Mirroring session supports up to 2 source and 2 destination ports. If you want to include an extra port, separate the curly brackets with a semicolon (no space) and list the key-value pairs of the second port in the other curly bracket.

7. Display the list of all Traffic Mirroring sessions running on a host.

```
nutanix@cvm$ accli net.list_traffic_mirror
```

Example:

```
nutanix@cvm$ accli net.list_traffic_mirror
Name          UUID                               State
traffic-mirror-1 69252eb5-8047-4e3a-8adc-91664a7104af kActive
```

Possible values for **State** are:

- *kActive*: Denotes that the Traffic Mirroring session is active.
- *kError*: denotes that there is an error and the configuration is not working. For example, if there are two sources and one source is down, the **State** of the session is displayed as *kError*.

8. Display the details of a Traffic Mirroring session.

```
nutanix@cvm$ accli net.get_traffic_mirror traffic-mirroring-session-name
```

Replace `traffic-mirroring-session-name` with the name of the Traffic Mirroring session whose details you want to view.

See the following example where source NIC is a physical interface (**source\_list** has **nic\_type** listed as `"kHostNic"`):

```
nutanix@cvm$ accli net.get_traffic_mirror traffic-mirror-1
traffic-mirror-1 {
  config {
    datapath_name: "s6925"
    description: "traffic mirror session 1"
    destination_list {
      nic_type: "kVmNic"
      port_identifier: "50:6b:8d:de:c6:44"
      uuid: "85abfdd5-7419-4f7c-bffa-8f961660e516"
    }
  }
  name: "traffic-mirror-1"
  session_uuid: "69252eb5-8047-4e3a-8adc-91664a7104af"
  source_list {
```



```

        direction: "kBiDir"
        nic_type: "kHostNic"
        port_identifier: "br0-up"
        uuid: "492a2bda-ffc0-486a-8bc0-8ae929471714"
    }
}
stats {
    name: "traffic-mirror-1"
    session_uuid: "69252eb5-8047-4e3a-8adc-91664a7104af"
    state: "kActive"
    stats_list {
        tx_byte_cnt: 67498
        tx_pkt_cnt: 436
    }
}
}

```

See the following example where source NIC is a virtual interface (`source_list` has `nic_type` listed as `"kVmNic"`):

```

nutanix@cvm$ accli net.get_traffic_mirror traffic-brAtlas
traffic-session-name {
    config {
        datapath_name: "s5812"
        destination_list {
            nic_type: "kVmNic"
            port_identifier: "50:6b:8d:95:11:00"
            uuid: "c6234cbf-c6e8-4a9c-1234-b8ade24f"
        }
        name: "traffic-brAtlas"
        session_uuid: "58122427-0302-4e61-8185-13abaad2a8d8"
        source_list {
            direction: "kBiDir"
            inactive: False
            nic_type: "kVmNic"
            port_identifier: "50:6b:8d:ab:cd:21"
            uuid: "1a70d859-253b-43d6-ac17-de78d854d925"
        }
    }
    stats {
        name: "traffic-brAtlas"
        session_uuid: "58122427-0302-4e61-8185-13abaad2a8d8"
        state: "kActive"
        state_message: ""
        stats_list {
            tx_byte_cnt: 0
            tx_pkt_cnt: 0
        }
    }
}

```

Note: The value of the **datapath\_name** field in the Traffic Mirroring session configuration is a unique key that identifies the Traffic Mirroring session. You need the unique key to correctly identify the Traffic Mirroring session for troubleshooting reasons.

### Updating a Traffic Mirroring Session

You can update any of the details of a Traffic Mirroring session.

### About this task

When you are updating a Traffic Mirroring session:



- Specify the values of the parameters you want to update.
- Add the remaining parameters again as specified originally when you created the Traffic Mirroring session.

For example, if you want to change only the name and description, specify the updated name and description and then add the **source\_list** and **dest\_list** parameters exactly as they were before updating.

Important: Updating the Traffic Mirroring session on an AHV host through aCLI is disabled if the cluster is registered to Prism Central. Nutanix recommends creating Traffic Mirroring sessions from Prism Central user interface or v4 API.

Perform the following procedure to update a Traffic Mirroring session:

### Procedure

1. Log on to a Controller VM in the cluster with SSH.
2. Update the Traffic Mirroring session.

```
nutanix@cvm$ acli net.update_traffic_mirror traffic-mirroring-session-name
description="description-text" source_list=\{uuid=host-uuid,type=kHostNic,identifier=source-
port-name,direction=traffic-type} dest_list=\{uuid=vm-UUID,type=kVmNic,identifier=vNIC-mac-
address}
```

The update command includes the same parameters as the create command. See [Configuring Traffic Mirroring on an AHV Host](#) on page 96 for more information.

Example:

```
nutanix@cvm$ acli net.update_traffic_mirror traffic-mirror-1 name=traffic-
mirror_br0_to_traffic-mirror_dest description="traffic mirror from
br0-up to traffic-mirror-dest VM" source_list=\{uuid=492a2bda-
ffc0-486a-8bc0-8ae929471714,type=kHostNic,identifier=br0-up,direction=kBiDir} dest_list=
\{uuid=85abfdd5-7419-4f7c-bffa-8f961660e516,type=kVmNic,identifier=50:6b:8d:de:c6:44}
SpanUpdate: complete
```

```
nutanix@cvm$ acli net.list_traffic_mirror
Name                               UUID                               State
traffic-mirror_br0_to_traffic-mirror_dest  69252eb5-8047-4e3a-8adc-91664a7104af  kActive
```

```
nutanix@cvm$ acli net.get_traffic_mirror traffic_mirror_br0_to_traffic_mirror_dest
traffic-mirror_br0_to_traffic-mirror_dest {
  config {
    datapath_name: "s6925"
    description: "traffic mirror from br0-up to traffic-mirror-dest VM"
    destination_list {
      nic_type: "kVmNic"
      port_identifier: "50:6b:8d:de:c6:44"
      uuid: "85abfdd5-7419-4f7c-bffa-8f961660e516"
    }
    name: "traffic-mirror_br0_to_traffic-mirror_dest"
    session_uuid: "69252eb5-8047-4e3a-8adc-91664a7104af"
    source_list {
      direction: "kBiDir"
      nic_type: "kHostNic"
      port_identifier: "br0-up"
      uuid: "492a2bda-ffc0-486a-8bc0-8ae929471714"
    }
  }
  stats {
    name: "traffic-mirror_br0_to_traffic-mirror_dest"
```



```

    session_uuid: "69252eb5-8047-4e3a-8adc-91664a7104af"
    state: "kActive"
    stats_list {
      tx_byte_cnt: 805705
      tx_pkt_cnt: 4792
    }
  }
}

```

In this example, only the name and description are updated. However, complete details of the source and destination ports are included in the command exactly as they were before the update.

If you want to change the name of a Traffic Mirroring session, specify the existing name first, and then include the new name by using the “name=” parameter as shown in this example.

### Deleting a Traffic Mirroring Session

Delete the Traffic Mirroring session if you want to disable Traffic Mirroring on an AHV host. Nutanix recommends that you delete the Traffic Mirroring session associated with a Traffic Mirroring destination VM or Traffic Mirroring destination VNIC.

### About this task

Important: Deleting the Traffic Mirroring session on an AHV host through aCLI is disabled if the cluster is registered to Prism Central. Nutanix recommends creating Traffic Mirroring sessions from Prism Central user interface or v4 API.

To delete a Traffic Mirroring session, do the following:

### Procedure

1. Log on to a Controller VM in the cluster with SSH.
2. Delete the Traffic Mirroring session.

```
nutanix@cvm$ acli net.delete_traffic_mirror traffic-mirror-session-name
```

Replace **traffic-mirror-session-name** with the name of the Traffic Mirroring session you want to delete.

### Traffic Mirroring Session Alerts

Prism Element web console generates the Traffic Mirroring session alerts for the following inconsistent state scenarios:

Note: For more information about the Alerts generated in Prism Element web console, see [Prism Web Console Alerts and Events Reference Guide](#).

Table 12: Traffic Mirror Session Alerts List

Alert Title	Possible Cause
Inconsistent SPAN Session State Detected	This alert occurs when a VM NIC is removed from the destination VM.
Inconsistent SPAN Session State Detected	This alert occurs when a destination VM is migrated to another host.



Alert Title	Possible Cause
Inconsistent SPAN Session State Detected	This alert occurs when a destination VM is powered off.

## MAC Address Prefix

You can avoid duplicate IP addresses in a single-cluster or multi-cluster environment by implementing one of two possible configurations:

1. Nutanix recommends that you assign a set of unique VLANs for guest VMs on each AHV cluster. Ensure these VLANs do not overlap with the VLANs on other AHV clusters. Assigning unique VLAN ranges for each cluster reduces the risk of MAC address conflicts. Such an assignment also ensures compliance with the general best practice of maintaining small Layer 2 broadcast domains with limited numbers of endpoints.
2. If multiple AHV clusters need to share the same VLAN, or when guest VM MAC addresses need be globally unique among multiple AHV clusters, configure a predefined MAC address prefix for each AHV cluster.

Nutanix does not guarantee unique MAC address assignment by default between Nutanix clusters with VLAN networks.

### Sample Design Scenario with Multiple Sites and Clusters

Using locally administered MAC addresses, you can ensure unique MAC addresses for VMs in an environment made up of multiple sites and clusters.

A MAC address is usually a 6-octet hexadecimal address block. The notation for a MAC address is **xx:xx:xx:xx:xx:xx**. In this 6-octet address, the first 3 octets are the organizationally unique identifier (OUI) octets or the MAC address prefix OUI. The second bit of the 1st octet (the first hexadecimal number **xx**) is set to 1 to make the MAC address a locally administered MAC address. The next 3 octets are NIC specific octets, **xx:xx:xx**, represents the useable hexadecimal range for endpoints within each AHV cluster.

Note: Nutanix AHV clusters use the MAC address prefix OUI **50:6B:8D** by default.

By default, Acropolis leader generates MAC address for a VM on AHV. The first 24 bits of the MAC address (OUI) is set to **50-6b-8d (0101 0000 0110 1101 1000 1101)** and are reserved by Nutanix, the 25th bit is set to **1** (reserved by Acropolis leader), the 26th bit to 48th bits are auto generated random numbers.

Consider this sample design of a deployment with three sites and five clusters in each site. Define a unique MAC address prefix for Site1-Cluster1 such as 02:01:01, where:

- 02—Defines the MAC address as a unicast address that is locally administered. This value could be a hexadecimal number defined by X2, X3, X6, X7, XA, XB, or XE series, where X is any valid hexadecimal value such that the second binary bit (binary bits being counted from right to left, right most is the first bit) of the binary equivalent of this hexadecimal number **xx** is set to 1 to make the MAC address a locally administered MAC address.
- 01—Used to identify, for example, the site number. This value could be any valid hexadecimal value.
- 01—Used to identify, for example, the AHV cluster within the site. This value could be any valid hexadecimal value

The NIC specific octets, **xx:xx:xx**, are auto-assigned to the VM NICs or the endpoints within each AHV cluster. Thus, for Site1, the clusters would have the following prefixes:





- Site1-Cluster1: 02:01:01
- Site1-Cluster2: 02:01:02
- Site1-Cluster3: 02:01:03
- Site1-Cluster4: 02:01:04
- Site1-Cluster5: 02:01:05

... and so on for the other clusters at Site1.

Similarly for Site2, if you define, for example 02:02:01 as the MAC address prefix for the first cluster - Cluster1, you get the series of predefined MAC address prefixes for the clusters and VMs or endpoints in Site2, Cluster1.

- Site2-Cluster1-VM1: 02:02:01:00:00:01
- Site2-Cluster1-VM2: 02:02:01:00:00:02
- ...
- Site2-Cluster1-VM10: 02:02:01:00:00:0A

... and so on for the other clusters at Site2.

## Adding a MAC Address Prefix

### Before you begin

Ensure that the VMs in the cluster do not have any NICs that have MAC addresses with the default prefix.

### About this task

Use aCLI to configure the MAC address prefix for a cluster.

### Procedure

1. Log on to a Controller VM in your cluster with SSH.
2. Access Acropolis CLI using the `acli` command.

```
nutanix@cvm$ acli
```

The prompt changes to `<acropolis>`.

3. Add the MAC address prefix for the cluster using the following command.

```
<acropolis> net.set_mac_prefix XX:XX:XX
```

Note: Ensure that the VMs in the cluster do not have any NICs that have MAC addresses with the default prefix.

Replace `XX:XX:XX` by the MAC address prefix for the cluster.

Using the example discussed in [MAC Address Prefix](#) on page 104, the following sample command adds the `02:01:01` as the MAC address prefix for Site1-Cluster1

```
<acropolis> net.set_mac_prefix 02:01:01
```

### What to do next

Verify if the MAC address prefix is configured using the `net.get_mac_prefix` command.



The output displays the hexadecimal prefix that you configured.

Using the configuration example, the output would show `"02:01:01"` as follows:

```
<acropolis> net.get_mac_prefix  
"02:01:01"  
<acropolis>
```

Repeat this procedure to add MAC address prefixes to other clusters that share the same VLAN (defining the common broadcast domain) that you want to avoid duplicate MAC addresses in.

## Removing the MAC Address Prefix

You can remove the MAC address prefix. The cluster then uses the default MAC prefix, `"50:6b:8d"`.

### Before you begin

Remove the MAC addresses with the configured prefix from the VM NICs in the cluster.

### About this task

Use aCLI commands to remove the MAC address prefix for a cluster.

### Procedure

1. Log on to a Controller VM in your cluster with SSH.
2. Access Acropolis CLI using the `acli` command.

```
nutanix@cvm$ acli
```

The prompt changes to `<acropolis>`.

3. Remove the MAC address prefix for the cluster using the following command.

```
<acropolis> net.clear_mac_prefix
```

Note: Ensure that the VMs in the cluster do not have any NICs that have MAC addresses with the configured prefix.

### What to do next

Verify that the MAC address prefix is removed. When you use the `net.get_mac_prefix` command, the output displays the default MAC address prefix, `"50:6b:8d"`.

```
<acropolis> net.get_mac_prefix  
"50:6b:8d"  
<acropolis>
```

## Enabling RSS Virtio-Net Multi-Queue by increasing the Number of VNIC Queues

Multi-Queue in VirtIO-net enables you to improve network performance for network I/O-intensive guest VMs or applications running on AHV hosts.

### About this task

You can enable VirtIO-net multi-queue by increasing the number of VNIC queues. If an application uses many distinct streams of traffic, Receive Side Scaling (RSS) can distribute the streams across multiple VNIC DMA rings. This increases the amount of RX buffer space by the number of VNIC queues (N). Also, most guest operating systems pin each ring to a particular vCPU, handling the interrupts and ring-walking on that vCPU, by that means achieving N-way



parallelism in RX processing. However, if you increase the number of queues beyond the number of vCPUs, you cannot achieve extra parallelism.

Following workloads have the greatest performance benefit of VirtIO-net multi-queue:

- VMs where traffic packets are relatively large
- VMs with many concurrent connections
- VMs with network traffic moving:
  - Among VMs on the same host
  - Among VMs across hosts
  - From VMs to the hosts
  - From VMs to an external system
- VMs with high VNIC RX packet drop rate if CPU contention is not the cause

You can increase the number of queues of the AHV VM VNIC to allow the guest OS to use multi-queue VirtIO-net on guest VMs with intensive network I/O. Multi-Queue VirtIO-net scales the network performance by transferring packets through more than one Tx/Rx queue pair at a time as the number of vCPUs increases.

Nutanix recommends that you be conservative when increasing the number of queues. Do not set the number of queues larger than the total number of vCPUs assigned to a VM. Packet reordering and TCP retransmissions increase if the number of queues is larger than the number of vCPUs assigned to a VM. For this reason, start by increasing the queue size to 2. The default queue size is 1. After making this change, monitor the guest VM and network performance. Before you increase the queue size further, verify that the vCPU usage has not dramatically or unreasonably increased.

Perform the following steps to make more VNIC queues available to a guest VM. See your guest OS documentation to verify if you must perform extra steps on the guest OS to apply the additional VNIC queues.

Note: You must shut down the guest VM to change the number of VNIC queues. Therefore, make this change during a planned maintenance window. The VNIC status might change from Up->Down->Up or a restart of the guest OS might be required to finalize the settings depending on the guest OS implementation requirements.

### Before you begin

(Optional) Nutanix recommends that you perform the following checks before you enable RSS Virtio-Net Multi-Queue by increasing the number of VNIC queues:

- AHV and AOS are running the latest version.
- AHV guest VMs are running the latest version of the Nutanix VirtIO driver package.

For RSS support, ensure you are running Nutanix VirtIO 1.1.6 or later. See [Nutanix VirtIO for Windows](#) on page 140 for more information about Nutanix VirtIO.

### Procedure

To set up a multi-queue virtio-net connection, perform the following steps:



1. Determine the exact name of the guest VM for which you want to change the number of VNIC queues using the following command:

```
nutanix@cvm$ acli vm.list
```

An output similar to the following is displayed:

```
nutanix@cvm$ acli vm.list
VM name      VM UUID
ExampleVM1    a91a683a-4440-45d9-8dbe-xxxxxxxxxxxx
ExampleVM2    fda89db5-4695-4055-a3d4-xxxxxxxxxxxx
...
```

2. Determine the MAC address of the VNIC and confirm the current number of VNIC queues using the following command:

```
nutanix@cvm$ acli vm.nic_get VM-name
```

Replace **VM-name** with the name of the VM.

An output similar to the following is displayed:

```
nutanix@cvm$ acli vm.nic_get VM-name
...
mac_addr: "50:6b:8d:2f:zz:zz"
...
(queues: 2)    <- If there is no output of 'queues', the setting is default (1 queue).
```

Note: AOS defines queues as the maximum number of Tx/Rx queue pairs (default is 1).

3. Determine the total count of vCPUs assigned to the VM using the following command:

```
nutanix@cvm$ acli vm.get VM-name | grep num.*vcpu
```

Replace **VM-name** with the name of the VM.

An output similar to the following is displayed:

```
num_cores_per_vcpu: 4
num_vcpus: 1
```

The total count of vCPUs assigned to the VM is calculated as -  $num\_vcpus * num\_cores\_per\_vcpu$ . In the above output, the total count of vCPUs assigned to the VM is  $1 * 4 = 4$ .

4. Shut down the guest VM using the following command:

```
nutanix@cvm$ acli vm.shutdown VM-name
```

Replace **VM-name** with the name of the VM.

5. Increase the number of VNIC queues.

```
nutanix@cvm$ acli vm.nic_update VM-name vNIC-MAC-address queues=N
```

Replace **VM-name** with the name of the guest VM, **vNIC-MAC-address** with the MAC address of the VNIC, and **N** with the number of queues.

Note: **N** must be less than or equal to the total count of vCPUs assigned to the guest VM.

6. Start the guest VM using the following command:

```
nutanix@cvm$ acli vm.on VM-name
```

Replace **VM-name** with the name of the VM.



7. Confirm in the guest OS documentation if any additional steps are required to enable multi-queue in VirtIO-net.

Note: Microsoft Windows has RSS enabled by default.

For example, for RHEL and CentOS VMs, perform the following steps:

- a. Log on to the guest VM.
- b. Confirm if `irqbalance.service` is active or not using the following command:

```
uservm# systemctl status irqbalance.service
```

An output similar to the following is displayed:

```
irqbalance.service - irqbalance daemon
Loaded: loaded (/usr/lib/systemd/system/irqbalance.service; enabled; vendor preset:
enabled)
Active: active (running) since Tue 2020-04-07 10:28:29 AEST; Ns ago
```

- c. Start `irqbalance.service` if it is not active using the following command:

Note: It is active by default on CentOS VMs. You might have to start it on RHEL VMs.

```
uservm# systemctl start irqbalance.service
```

- d. Run the following command:

```
uservm# ethtool -L ethX combined M
```

Replace:

- **M** with the number of VNIC queues.
- **ethX** with the required ethernet interface. For example, `eth1`.

Note the following caveat from the RHEL 7 *Virtualization Tuning and Optimization Guide: 5.4. NETWORK TUNING TECHNIQUES* document:

Currently, setting up a multi-queue virtio-net connection can have a negative effect on the performance of outgoing traffic. Specifically, this might occur while sending packets under 1,500 bytes over the Transmission Control Protocol (TCP) stream.

8. Monitor the VM performance to make sure that the expected network performance increase is observed and that the guest VM vCPU usage is not dramatically increased to impact the application on the guest VM.

For assistance with the steps described in this document, or if these steps do not resolve your guest VM network performance issues, contact Nutanix Support.

## Changing the IP Address of an AHV Host

Change the IP address, network mask, or gateway of an AHV host.

### Before you begin

Ensure that you perform the following actions before you change the IP address, network mask, or gateway of an AHV host:

Caution: All Controller VMs and hypervisor hosts must be on the same subnet.

Warning: Ensure that you perform the steps in the exact order as indicated in this document.



1. Verify the cluster health by following the instructions in [Verifying the Cluster Health](#).

Do not proceed if the cluster cannot tolerate failure of at least one node.

2. Shut down all the guest VMs in the cluster from within the guest OS or use the Prism Element web console.
3. Put the node into the maintenance mode:

1. Use SSH to log on to a Controller VM in the cluster.
2. Determine the IP address of the node you want to put into the maintenance mode:

```
nutanix@cvm$ acli host.list
```

Note the value of Hypervisor IP (Host IP) for the node you want to put in the maintenance mode.

3. Run the following command to put the node in maintenance mode:

```
nutanix@cvm$ acli host.enter_maintenance_mode <Host_IP>
```

Replace **<Host\_IP>** with either the IP address or hostname of the AHV host you want to shut down.

4. Verify if the host is in maintenance mode, using the following command:

```
nutanix@cvm$ acli host.get <Host_IP>
```

Replace **<Host\_IP>** with either the IP address or hostname of the AHV host for which you want to verify the maintenance mode exit status.

In the output that is displayed, ensure that **node\_state** equals to **EnteredMaintenanceMode** and **schedulable** equals **False**.

4. Stop the Acropolis service on all the CVMs in the cluster.

```
nutanix@cvm$ allssh genesis stop acropolis
```

### About this task

Perform the following procedure to change the IP address, netmask, or gateway of an AHV host.

### Procedure

1. Edit the settings of port br0, which is the internal port on the default bridge br0.

- a. Log on to the host console as root.

You can access the hypervisor host console either through IPMI or by attaching a keyboard and monitor to the node.

- b. Open the network interface configuration file for port br0 in a text editor.

```
root@ahv# vi /etc/sysconfig/network-scripts/ifcfg-br0
```

- c. Update entries for host IP address, netmask, and gateway.

The block of configuration information that includes these entries is similar to the following:

```
ONBOOT="yes"
NM_CONTROLLED="no"
PERSISTENT_DHCLIENT=1
NETMASK="subnet_mask"
IPADDR="host_ip_addr"
DEVICE="br0"
TYPE="OVSIIntPort"
GATEWAY="gateway_ip_addr"
```



```
BOOTPROTO="none"
```

- Replace `host_ip_addr` with the IP address for the hypervisor host.
- Replace `subnet_mask` with the subnet mask for `host_ip_addr`.
- Replace `gateway_ip_addr` with the gateway address for `host_ip_addr`.

d. Save your changes.

e. Restart network services.

```
systemctl restart network.service
```

f. Assign the impacted host to a VLAN and confirm VLAN tagging on port `br0`, using the following commands:

```
root@ahv# ovs-vsctl set port br0 tag=host_vlan_tag
```

Replace `host_vlan_tag` with the VLAN tag for the host.

```
root@ahv# ovs-vsctl list port br0
```

g. Verify network connectivity by pinging the gateway, other CVMs, and AHV hosts.

2. Log on to the Controller VM that is running on the AHV host whose IP address you changed and restart genesis.

```
nutanix@cvm$ genesis restart
```

If the restart is successful, output similar to the following is displayed:

```
Stopping Genesis pids [1933, 30217, 30218, 30219, 30241]  
Genesis started on pids [30378, 30379, 30380, 30381, 30403]
```

See [Controller VM Access](#) on page 16 for information about how to log on to a Controller VM.

Genesis takes a few minutes to restart.

3. From any CVM in the cluster, restart the Acropolis service.

```
nutanix@cvm$ cluster start
```

4. Run the following command to get the updated IP address of the node (AHV host):

```
nutanix@cvm$ acli host.list
```

5. Exit the node from the maintenance mode using the following command:

```
nutanix@cvm$ acli host.exit_maintenance_mode <Host_IP>
```

Replace `<Host_IP>` with either the updated IP address or host name of the AHV host you want to exit from maintenance mode.

6. Verify if the host is exited from maintenance mode, using the following command:

```
nutanix@cvm$ acli host.get <Host_IP>
```

Replace `<Host_IP>` with either the IP address or hostname of the AHV host for which you want to verify the maintenance mode exit status.

In the output that is displayed, ensure that `node_state` equals `kAcropolisNormal` or `AcropolisNormal` and `schedulable` equals `True`.



7. Verify if the IP address of the hypervisor host has changed. Run the following nCLI command from any CVM other than the one in the maintenance mode.

```
nutanix@cvm$ ncli host list
```

An output similar to the following is displayed:

```
nutanix@cvm$ ncli host list
  Id           : aaaaaaaa-bbbb-cccc-dddd-eeeeeeeeeeee::1234
  Uuid         : ffffffff-gggg-hhhh-iiii-jjjjjjjjjjjj
  Name        : XXXXXXXXXXXX-X
  IPMI Address : X.X.Z.3
  Controller VM Address : X.X.X.1
  Hypervisor Address : X.X.Y.4 <- New IP Address
...
```

8. Power on the guest VMs from the Prism Element web console.



# VIRTUAL MACHINE MANAGEMENT

---

The following topics describe various aspects of virtual machine management in an AHV cluster.

## Supported Guest VM Types for AHV

The [compatibility matrix](#) available on the Nutanix Support portal includes the latest supported AHV guest VM OSes.

### AHV Configuration Maximums

The [Nutanix configuration maximums](#) available on the Nutanix support portal includes all the latest configuration limits applicable to AHV. Select the appropriate AHV version to view version specific information.

## Creating a VM (AHV)

In AHV clusters, you can create a new virtual machine (VM) through the Prism Element web console.

### About this task

Note: Use Prism Central to create a VM with the memory overcommit feature enabled. Prism Element web console does not allow you to enable memory overcommit while creating a VM. If you create a VM using the Prism Element web console and want to enable memory overcommit for it, update the VM using Prism Central and enable memory overcommit in the **Update VM** page in Prism Central. For more information, see [Updating a VM through Prism Central](#) information in *Prism Central Infrastructure Guide*.

When creating a VM, you can configure all of its components, such as number of vCPUs and memory, but you cannot attach a volume group to the VM. Attaching a volume group is possible only when you are modifying a VM.

To create a VM, do the following:

### Procedure

1. Log in to Prism Element web console.
2. In the VM dashboard, click the **Create VM** button.

Note: This option does not appear in clusters that do not support this feature.

The Create VM dialog box appears.



3. Do the following in the indicated fields:
  - a. **Name:** Enter a name for the VM.
  - b. **Description** (optional): Enter a description for the VM.
  - c. **Timezone:** Select the timezone that you want the VM to use. If you are creating a Linux VM, select **(UTC) UTC**.

Note:

The RTC of Linux VMs must be in UTC, so select the UTC timezone if you are creating a Linux VM.

Windows VMs preserve the RTC in the local timezone, so set up the Windows VM with the hardware clock pointing to the desired timezone.

- d. **Use this VM as an agent VM:** Select this option to make this VM as an agent VM.

You can use this option for the VMs that must be powered on before the rest of the VMs (for example, to provide network functions before the rest of the VMs are powered on the host) and must be powered off after the rest of the VMs are powered off (for example, during maintenance mode operations). Agent VMs are never migrated to any other host in the cluster. If an HA event occurs or the host is put in maintenance mode, agent VMs are powered off and are powered on the same host once that host comes back to a normal state.

If an agent VM is powered off, you can manually start that agent VM on another host and the agent VM now permanently resides on the new host. The agent VM is never migrated back to the original host. Note that you cannot migrate an agent VM to another host while the agent VM is powered on.
  - e. **vCPU(s):** Enter the number of virtual CPUs to allocate to this VM.
  - f. **Number of Cores per vCPU:** Enter the number of cores assigned to each virtual CPU.
  - g. **Memory:** Enter the amount of memory (in GiB) to allocate to this VM.
4. (For GPU-enabled AHV clusters only) To configure GPU access, click **Add GPU** in the **Graphics** section, and then do the following in the Add GPU dialog box:

For more information, see [GPU and vGPU Support](#) on page 194.

  - a. To configure GPU pass-through, in **GPU Mode**, click **Passthrough**, select the GPU that you want to allocate, and then click **Add**.

If you want to allocate additional GPUs to the VM, repeat the procedure as many times as you need to. Make sure that all the allocated pass-through GPUs are on the same host. If all specified GPUs of the type that you want to allocate are in use, you can proceed to



allocate the GPU to the VM, but you cannot power on the VM until a VM that is using the specified GPU type is powered off.

For more information, see [GPU and vGPU Support](#) on page 194.

- b. To configure virtual GPU access, in **GPU Mode**, click **virtual GPU**, select a GRID license, and then select a virtual GPU profile from the list.

Note: This option is available only if you have installed the GRID host driver on the GPU hosts in the cluster.

For more information about the NVIDIA GRID host driver installation instructions, see the [NVIDIA Grid Host Driver for Nutanix AHV Installation Guide](#).

You can assign multiple virtual GPU to a VM. A vGPU is assigned to the VM only if a vGPU is available when the VM is starting up.

Before you add multiple vGPUs to the VM, see [Multiple Virtual GPU Support](#) on page 198 and [Restrictions for Multiple vGPU Support](#) on page 199.

Note: Multiple vGPUs are supported on the same VM only if you select the highest vGPU profile type.

After you add the first vGPU, to add multiple vGPUs, see [Adding Multiple vGPUs to the Same VM](#) on page 202.

5. Select one of the following firmware to boot the VM.

- » **Legacy BIOS:** Select legacy BIOS to boot the VM with legacy BIOS firmware.
- » **UEFI:** Select UEFI to boot the VM with UEFI firmware. UEFI firmware supports larger hard drives, faster boot time, and provides more security features. For more information about UEFI firmware, see [UEFI Support for VM](#) on page 181.

If you select UEFI, you can enable the following features:

- **Secure Boot:** Select this option to enable UEFI secure boot policies for your guest VMs. For more information about Secure Boot, see [Secure Boot Support for VMs](#) on page 187.
- **Windows Defender Credential Guard:** Select this option to enable the Windows Defender Credential Guard feature of Microsoft Windows operating systems that allows you to securely isolate user credentials from the rest of the operating system. Follow the detailed instructions described in [Windows Defender Credential Guard Support in AHV](#) on page 154 to enable this feature.

Note: To add virtual TPM, see [Securing AHV VMs with Virtual Trusted Platform Module](#).



6. To attach a disk to the VM, click the **Add New Disk** button.

The Add Disk dialog box appears. Do the following in the indicated fields:

- a. **Type:** Select the type of storage device, **DISK** or **CD-ROM**, from the drop-down list. The following fields and options vary depending on whether you choose **DISK** or **CD-ROM**.
- b. **Operation:** Specify the device contents from the drop-down list.
- Select **Clone from ADSF file** to copy any file from the cluster that can be used as an image onto the disk.
  - Select **Empty CD-ROM** to create a blank CD-ROM device. (This option appears only when **CD-ROM** is selected in the previous field.) A CD-ROM device is needed when you intend to provide a system image from CD-ROM.
  - Select **Allocate on Storage Container** to allocate space without specifying an image. (This option appears only when **DISK** is selected in the previous field.) Selecting this option means you are allocating space only. You have to provide a system image later from a CD-ROM or other source.
  - Select **Clone from Image Service** to copy an image that you have imported by using image service feature onto the disk. For more information about the Image Service feature, see [Configuring Images](#) and [Image Management](#) in the *Prism Self Service Administration Guide*.
- c. **Bus Type:** Select the bus type from the pull-down list. The choices are **IDE**, **SCSI**, or **SATA**.

The options displayed in the **Bus Type** drop-down list varies based on the storage device Type selected in Step a.

- For device **Disk**, select from **SCSI**, **SATA**, **PCI**, or **IDE** bus type.
- For device **CD-ROM**, you can select either **IDE**, or **SATA** bus type.

Note:

- SCSI bus is the preferred bus type and it is used in most cases. Ensure you have installed the VirtIO drivers in the guest OS. For more information about VirtIO drivers, see [Nutanix VirtIO for Windows](#) in *AHV Administration Guide*.
- For AHV 5.16 and later, you cannot use an IDE device if **Secured Boot** is enabled for **UEFI Mode** boot configuration.

Caution: Use SATA, PCI, IDE for compatibility purpose when the guest OS does not have VirtIO drivers to support SCSI devices. This may have performance implications. For more information about VirtIO drivers, see [Nutanix VirtIO for Windows](#) in *AHV Administration Guide*.

- d. **ADSF Path:** Enter the path to the desired system image.

This field appears only when **Clone from ADSF file** is selected. It specifies the image to copy. Enter the path name as `/storage_container_name/iso_name.iso`. For example to clone an image from `myos.iso` in a storage container named `crt1`, enter `/crt1/myos.iso`. When a user types the storage container name (`/storage_container_name/`), a



list appears of the ISO files in that storage container (assuming one or more ISO files had previously been copied to that storage container).

- e. **Image:** Select the image that you have created by using the image service feature. This field appears only when **Clone from Image Service** is selected. It specifies the image to copy.
- f. **Storage Container:** Select the storage container to use from the drop-down list. This field appears only when **Allocate on Storage Container** is selected. The list includes all storage containers created for this cluster.
- g. **Logical Size (GiB):** Enter the disk size in GiB.
- h. **Index:** Displays **Next Available** by default.
- i. When all the field entries are correct, click the **Add** button to attach the disk to the VM and return to the Create VM dialog box.
- j. Repeat this step to attach additional devices to the VM.

7. To create a network interface for the VM, click the **Add New NIC** button.  
Prism console displays the Create NIC dialog box.

Note: To create or update a Traffic Mirroring destination type VM or vNIC, use command line interface. For more information, see [Traffic Mirroring on AHV Hosts](#) on page 92.

Do the following in the indicated fields:

- a. **Subnet Name:** Select the target virtual LAN from the drop-down list.  
The list includes all defined networks.

Note: Selecting IPAM enabled subnet from the drop-down list displays the **Private IP Assignment** information that provides information about the number of free IP addresses available in the subnet and in the IP pool.

- b. **Network Connection State:** Select the state for the network that you want it to operate in after VM creation. The options are **Connected** or **Disconnected**.
- c. **Private IP Assignment:** This is a read-only field and displays the following:
  - **Network Address/Prefix:** The network IP address and prefix.
  - **Free IPs (Subnet):** The number of free IP addresses in the subnet.
  - **Free IPs (Pool):** The number of free IP addresses available in the IP pools for the subnet.
- d. **Assignment Type:** This is for IPAM enabled network. Select **Assign with DHCP** to assign IP address automatically to the VM using DHCP. For more information, see [IP Address Management](#) on page 60 .
- e. When all the field entries are correct, click the **Add** button to create a network interface for the VM and return to the Create VM dialog box.
- f. Repeat this step to create additional network interfaces for the VM.

Note:

- Nutanix does not recommend configuring multiple clusters to use the same broadcast domain (the same VLAN network), but if you do, configure MAC address prefixes for each cluster to avoid duplicate MAC addresses. For information about configuring or removing pre-defined prefix of MAC addresses for each cluster, see [MAC Address Prefix](#) on page 104
- Nutanix AHV clusters use the MAC address prefix OUI **50:6B:8D** by default.

8. To configure affinity policy for this VM, click **Set Affinity**.  
The Set VM Host Affinity dialog box appears.
  - a. Select the host or hosts on which you want configure the affinity for this VM.
  - b. Click **Save**.  
The selected host or hosts are listed. This configuration is permanent. The VM will not be moved from this host or hosts even in case of HA event and will take effect once the VM starts.



9. To customize the VM by using Cloud-init (for Linux VMs) or Sysprep (for Windows VMs), select the **Custom Script** check box.  
Fields required for configuring Cloud-init and Sysprep, such as options for specifying a configuration script or answer file and text boxes for specifying paths to required files, appear below the check box.
10. To specify a user data file (Linux VMs) or answer file (Windows VMs) for unattended provisioning, do one of the following:
  - » If you uploaded the file to a storage container on the cluster, click **ADSF path**, and then enter the path to the file.  
Enter the ADSF prefix (**adsf://**) followed by the absolute path to the file. For example, if the user data is in `/home/my_dir/cloud.cfg`, enter **adsf:///home/my\_dir/cloud.cfg**. Note the use of three slashes.
  - » If the file is available on your local computer, click **Upload a file**, click **Choose File**, and then upload the file.
  - » If you want to create or paste the contents of the file, click **Type or paste script**, and then use the text box that is provided.
11. To copy one or more files to a location on the VM (Linux VMs) or to a location in the ISO file (Windows VMs) during initialization, do the following:
  - a. In **Source File ADSF Path**, enter the absolute path to the file.
  - b. In **Destination Path in VM**, enter the absolute path to the target directory and the file name.  
For example, if the source file entry is `/home/my_dir/myfile.txt` then the entry for the **Destination Path in VM** should be `/<directory_name>/copy_destination` i.e. `/mnt/myfile.txt`.
  - c. To add another file or directory, click the button beside the destination path field. In the new row that appears, specify the source and target details.
12. When all the field entries are correct, click the **Save** button to create the VM and close the Create VM dialog box.  
The new VM appears in the VM table view.

## Managing a VM (AHV)

You can use the web console to manage virtual machines (VMs) in AHV managed clusters.

### About this task

Note: Use Prism Central to update a VM if you want to enable memory overcommit for it. Prism Element web console does not allow you to enable memory overcommit while updating a VM. You can enable memory overcommit in the **Update VM** page in Prism Central. For more information, see [Updating a VM through Prism Central](#) information in *Prism Central Infrastructure Guide*.

After creating a VM, you can use the web console to start or shut down the VM, launch a console window, update the VM configuration, take a snapshot, attach a volume group, migrate the VM, clone the VM, or delete the VM.



Note: Your available options depend on the VM status, type, and permissions. Unavailable options are grayed out.

To accomplish one or more of these tasks, do the following:

### Procedure

1. Log in to Prism Element web console.
2. In the VM dashboard, click the **Table** view.
3. Select the target VM in the table (top section of screen).  
The Summary line (middle of screen) displays the VM name with a set of relevant action links on the right. You can also right-click on a VM to select a relevant action.

The possible actions are **Manage Guest Tools**, **Launch Console**, **Power on** (or **Power off**), **Take Snapshot**, **Migrate**, **Clone**, **Update**, and **Delete**.

Note: VM pause and resume feature is not supported on AHV.

The following steps describe how to perform each action.

4. To manage guest tools as follows, click **Manage Guest Tools**.  
You can also enable NGT applications (self-service restore, Volume Snapshot Service and application-consistent snapshots) also as part of manage guest tools.
  - a. Select the **Enable Nutanix Guest Tools** checkbox to enable NGT on the selected VM.
  - b. Select the **Mount Nutanix Guest Tools** checkbox to mount NGT on the selected VM.  
Ensure that VM must have at least one empty IDE CD-ROM slot to attach the ISO.  
The VM is registered with the NGT service. NGT is enabled and mounted on the selected virtual machine. A CD with volume label NUTANIX\_TOOLS gets attached to the VM.
  - c. To enable self-service restore feature for Windows VMs, select the **Self Service Restore (SSR)** checkbox.  
The Self-Service Restore feature is enabled on the VM. The guest VM administrator can restore the desired file or files from the VM. For more information about self-service restore feature, see [Self-Service Restore](#) in the *Data Protection and Recovery with Prism Element* guide.
  - d. After you select the **Enable Nutanix Guest Tools** checkbox the VSS snapshot feature is enabled by default.  
After this feature is enabled, Nutanix native in-guest VmQuiesced Snapshot Service (VSS) agent takes snapshots for VMs that support VSS.

Note: The AHV VM snapshots are not application consistent. The AHV snapshots are taken from the **VM** entity menu by selecting a VM and clicking **Take Snapshot**.

The application consistent snapshots feature is available with Protection Domain based snapshots and Recovery Points in Prism Central. For more information,





see [Conditions for Application-consistent Snapshots](#) in the *Data Protection and Recovery with Prism Element* guide.

e. Click **Submit**.

The VM is registered with the NGT service. NGT is enabled and mounted on the selected virtual machine. A CD with volume label NUTANIX\_TOOLS gets attached to the VM.

Note:

- If you clone a VM, by default NGT is not enabled on the cloned VM. If the cloned VM is powered off, enable NGT from the UI and power on the VM. If cloned VM is powered on, enable NGT from the UI and restart the nutanix guest agent service.
- If you want to enable NGT on multiple VMs simultaneously, see [Enabling NGT and Mounting the NGT Installer Simultaneously on Multiple Cloned VMs](#).

If you eject the CD, you can mount the CD back again by logging into the Controller VM and running the following nCLI command.

```
nutanix@cvm$ ncli ngt mount vm-id=virtual_machine_id
```

For example, to mount the NGT on the VM with VM\_ID=00051a34-066f-72ed-0000-000000005400::38dc7bf2-a345-4e52-9af6-c1601e759987, type the following command.

```
nutanix@cvm$ ncli ngt mount vm-id=00051a34-066f-72ed-0000-000000005400::38dc7bf2-a345-4e52-9af6-c1601e759987
```

5. To launch a console window, click the **Launch Console** action link.

This opens a Virtual Network Computing (VNC) client and displays the console in a new tab or window. This option is available only when the VM is powered on. The console window includes the following menu options (top right):

- Clicking the **Mount ISO** button displays a window that allows you to mount an ISO image to the VM. To mount an image, select the desired image and CD-ROM drive from the drop-down lists and then click the **Mount** button.

Note: For information about how to select CD-ROM as the storage device when you intent to provide a system image from CD-ROM, see *Add New Disk* in [Creating a VM \(AHV\)](#) on page 113.

- Clicking the **Unmount ISO** button unmounts the ISO from the console.
- Clicking the **C-A-D** icon button sends a **CtrlAltDel** command to the VM.
- Clicking the camera icon button takes a screenshot of the console window.
- Clicking the power icon button allows you to power on/off the VM. These are the same options that you can access from the **Power On Actions** or **Power Off Actions** action link below the VM table (see next step).



6. To start or shut down the VM, click the **Power on** (or **Power off**) action link.

Power on begins immediately. If you want to power off the VMs, you are prompted to select one of the following options:

- **Power Off:** Hypervisor performs a hard power off action on the VM.
- **Power Cycle:** Hypervisor performs a hard restart action on the VM.
- **Reset:** Hypervisor performs an ACPI reset action through the BIOS on the VM.
- **Guest Shutdown:** Operating system of the VM performs a graceful shutdown.
- **Guest Reboot:** Operating system of the VM performs a graceful restart.

Select the option you want and click **Submit**.

Note: If you perform power operations such as Guest Reboot or Guest Shutdown by using the Prism Element web console or API on Windows VMs, these operations might silently fail without any error messages if at that time a screen saver is running in the Windows VM. Perform the same power operations again immediately, so that they succeed.

7. To make a snapshot of the VM, click the **Take Snapshot** action link.

For more information, see [Virtual Machine Snapshots](#) on page 139.

8. To migrate the VM to another host, click the **Migrate** action link.

This displays the **Migrate VM** dialog box. Select the target host from the drop-down list (or select the **System will automatically select a host** option to let the system choose the host) and then click the **Migrate** button to start the migration.

Note: Nutanix recommends to live migrate VMs when they are under light load. If they are migrated while heavily utilized, migration may fail because of limited bandwidth.

9. To clone the VM, click the **Clone** action link.

This displays the Clone VM dialog box, which includes the same fields as the Create VM dialog box. A cloned VM inherits the most the configurations (except the name) of the source VM. Enter a name for the clone and then click the **Save** button to create the clone. You can optionally override some of the configurations before clicking the **Save** button. For example, you can override the number of vCPUs, memory size, boot priority, NICs, or the guest customization.

Note:

- You can clone up to 250 VMs at a time.
- You cannot override the secure boot setting while cloning a VM, unless the source VM already had secure boot setting enabled.

10. To modify the VM configuration, click the **Update** action link.

The Update VM dialog box appears, which includes the same fields as the Create VM dialog box. Modify the configuration as needed, and then save the configuration. In addition to modifying the configuration, you can attach a volume group to the VM and enable flash mode on the VM. If you attach a volume group to a VM that is part of a protection

domain, the VM is not protected automatically. Add the VM to the same Consistency Group manually.

(For GPU-enabled AHV clusters only) You can add pass-through GPUs if a VM is already using GPU pass-through. You can also change the GPU configuration from pass-through to vGPU or vGPU to pass-through, change the vGPU profile, add more vGPUs, and change the specified vGPU license. However, you need to power off the VM before you perform these operations.

- Before you add multiple vGPUs to the VM, see [Multiple Virtual GPU Support](#) and [Restrictions for Multiple vGPU Support](#) in the *AHV Administration Guide*.
- Multiple vGPUs are supported on the same VM only if you select the highest vGPU profile type.
- For more information on vGPU profile selection, see:
  - *Virtual GPU Types for Supported GPUs* in the *NVIDIA Virtual GPU Software User Guide* in the NVIDIA's *Virtual GPU Software Documentation* web-page, and
  - [GPU and vGPU Support](#) in the *AHV Administration Guide*.

- After you add the first vGPU, to add multiple vGPUs, see [Adding Multiple vGPUs to the Same VM](#) in the *AHV Administration Guide*.

You can add new network adapters or NICs using the **Add New NIC** option. For more information, see Step 7 in [Creating a VM \(AHV\)](#) on page 113.

You can also modify the network used by an existing NIC. Before you modify the NIC network observe the limitations in [Limitation for vNIC Hot-Unplugging](#) in the *AHV Administration Guide*.

Note:

- To create or update a Traffic Mirroring destination type VM or vNIC, use command line interface. For more information, see [Traffic Mirroring on AHV Hosts](#) on page 92.
- If you delete a vDisk attached to a VM and snapshots associated with this VM exist, space associated with that vDisk is not reclaimed unless you also delete the VM snapshots.

To increase the memory allocation and the number of vCPUs on your VMs while the VMs are powered on (hot-pluggable), do the following:

- a. In the **vCPUs** field, you can increase the number of vCPUs on your VMs while the VMs are powered on.
- b. In the **Number of Cores Per vCPU** field, you can change the number of cores per vCPU only if the VMs are powered off.

Note: This is not a hot-pluggable feature.

- c. In the **Memory** field, you can increase the memory allocation on your VMs while the VMs are powered on.

For more information about hot-pluggable vCPUs and memory, see [Virtual Machine Memory and CPU Hot-Plug Configurations](#) in the *AHV Administration Guide*.

To attach a volume group to the VM, do the following:

- a. In the **Volume Groups** section, click **Add volume group**, and then do one of the following:
  - » From the **Available Volume Groups** list, select the volume group that you want to attach to the VM.
  - » Click **Create new volume group**, and then, in the **Create Volume Group** dialog box, create a volume group. After you create a volume group, select it from the **Available Volume Groups** list.

Repeat these steps until you have added all the volume groups that you want to attach to the VM.

- b. Click **Add**.



11. To enable flash mode on the VM, click the **Enable Flash Mode** check box.
  - » After you enable this feature on the VM, the status is updated in the VM table view. To view the status of individual virtual disks (disks that are flashed to the SSD), click the update disk icon in the **Disks** pane in the Update VM window.
  - » You can disable the flash mode feature for individual virtual disks. To update the flash mode for individual virtual disks, click the update disk icon in the **Disks** pane and deselect the **Enable Flash Mode** check box.
12. To delete the VM, click the **Delete** action link. A window prompt appears; click the **OK** button to delete the VM.

The deleted VM disappears from the list of VMs in the table.

### Limitation for vNIC Hot-Unplugging

If you detach (hot-unplug) the vNIC for the VM with guest OS installed on it, the AOS displays the detach result as successful, but the actual detach success depends on the status of the ACPI mechanism in guest OS.

The following table describes the vNIC detach observations and workaround applicable based on guest OS response to ACPI request:

Table 13: vNIC Detach - Observations and Workaround

Detach Procedure Followed	Guest OS responds to ACPI request (Yes/No)	AOS Behavior	Actual Detach Result	Workaround
vNIC Detach (hot-unplug)	Yes	vNIC detach is Successful.	vNIC detach is successful.	<i>No action needed</i>
<ul style="list-style-type: none"> <li>Using Prism Central: See <a href="#">Managing a VM (AHV)</a> topic in <i>Prism Central Infrastructure Guide</i>.</li> <li>Using Prism Element web console: See <a href="#">Managing a VM (AHV)</a> on page 119.</li> <li>Using <i>acli</i>: Log on to the CVM with SSH and run the following command:   <pre>nutanix@cvm\$ acli vm.nic_delete &lt;vm_name&gt; &lt;nic mac address&gt;</pre> or,   <pre>nutanix@cvm\$ acli vm.nic_update &lt;vm_name&gt; &lt;nic mac address&gt; connected=false</pre> Replace the following attributes in the above commands: <ul style="list-style-type: none"> <li>&lt;vm_name&gt; with the name of the guest VM for which the vNIC is to be detached.</li> <li>&lt;nic mac address&gt; with the vNIC MAC address that needs to be detached.</li> </ul> </li> </ul>	No	Observe the following logs:  <b>Device detached successfully</b>	vNIC detach is not successful.	Power cycle the VM for successful vNIC detach.

Note: In most cases, it is observed that the ACPI mechanism failure occurs when no guest OS is installed on the VM.

## VM Migration Specifications

VM migration can be performed in the following ways:

- *Live Migration* - This method is used to transfer the VM from one physical host to another host without affecting the normal functions and operations of the VM, and with minimal and often no interruption to the users of the VM.

Live migration is supported for the following scenarios:

- Intra-cluster live migration of guest VMs from one host to another host. For more information, see [Migrating Within the Cluster](#).
- Cross-cluster live migration (CCLM) of guest VMs protected with synchronous replication schedules. For more information, see [Cross-Cluster Live Migration](#) in the *Nutanix Disaster Recovery Guide*.
- On-demand cross-cluster live migration of guest VMs across AHV clusters registered to the same or different Prism Central AZs. For more information, see [On-Demand Cross-Cluster Live Migration](#) on page 134.
- *Cold Migration* - This method is used to transfer the VM from one physical host to another host with minimal VM downtime. You can use Nutanix Move for Inter-cluster VM migration <Third-Party Hypervisor> to the AHV cluster. For more information, see [Move User Guide](#).

Note: The <Third-party hypervisors> are Xen (Amazon EC2), Microsoft Hyper-V, and VMware ESXi.

## Live Migration Cases

VM live migration is applicable for the following cases:

- Maintenance activity required on the cluster - This scenario is applicable for a planned event (for example, scheduled maintenance of guest VMs) at the primary AZ when you put node into maintenance mode. For information about how to configure a planned failover, see [Cross-Cluster Live Migration](#) section in *Nutanix Disaster Recovery Guide*.
- AHV or Firmware Upgrades - When you upgrade the AHV or Firmware, the VMs need to be live migrated to another AHV cluster.
- Load balancing or to isolate specific VMs on hosts.
- Changes in Host affinity policy.
- Acropolis Dynamic Scheduling (ADS)- If ADS detects a resource contention in the cluster, it creates a migration plan to eliminate the hotspots in the cluster by migrating VMs from one host to another. For more information about ADS, see [Acropolis Dynamic Scheduling in AHV](#).
- Defragmentation activity to create resources for VM power-on operation. This activity is performed when the cluster can accommodate resources for VM power-on operation after moving some VMs to another host.

## Live Migration Restrictions

The following VM condition (or properties) impacts the VM Live migration.

- VM is powered off.



- VM is configured as an agent VM.

You can select or clear the **Use this VM as an agent VM** checkbox to enable or disable this function when you create or update a VM. For more information, see [Creating a VM through Prism Central \(AHV\)](#) section in *Prism Central Infrastructure Guide*.

- WSL2 is enabled. For more information about WSL2, see [Windows Subsystem for Linux \(WSL2\) Support on AHV](#).

- GPU passthrough is enabled.

For information about how to verify if GPU passthrough is enabled for the VM, see [Checking Live Migration Status of a VM](#) on page 128.

- CPU passthrough is enabled.

- VM-Host affinity is set from the Prism Element web console with one host.

For information about how to verify the VM-Host affinity policy from Prism Element web console, see [Verifying Affinity Policy Association](#) on page 129.

- VM Affinity policies are defined from Prism Central using VM Categories and Host categories.

For information about how to verify the affinity policies mapped to a VM from Prism Central, see [Checking Affinity Policies of a VM From Prism Central](#) on page 130.

- Defragmentation activity to create resources for VM-power-on is in progress.

Note: Starting with AOS 6.8 release, AHV supports the live migration of the guest VMs that have windows credential guard enabled. When you upgrade to AOS 6.8 or later release and enable HA reservation for the cluster, the system reserves the resources to accommodate the HA event. However, if you have the existing VMs in the cluster that have Windows Credential guard enabled, some guest VMs (both with or without windows credential guard enabled) might fail to start due to the unavailability of sufficient resources in the cluster. Ensure that you have sufficient resources available in the cluster to accommodate the HA reservation.

For more information on windows credential guard, see [Windows Defender Credential Guard Support in AHV](#) in *AHV Administration Guide*.

For information on how to enable HA reservation, see [Enabling High Availability for the Cluster](#) section in *Prism Web Console Guide*.

## Checking Live Migration Status of a VM

This section describes how to check whether the live migration is allowed for the VM.

### Procedure

To check if a VM can be live migrated, perform the following steps:

1. Log on to the CVM as a nutanix user using SSH.
2. Run the following command:

```
nutanix@cvm$ aclcli vm.get <VM_NAME> | grep 'allow_live_migrate'
```

Replace **<VM\_NAME>** with the actual VM name at your site.

The following attribute value confirms the live migration is allowed:

```
allow_live_migrate: True
```





What to do next

If `allow_live_migrate: False` is returned for the above command, then check the status of the VM properties that restricts live migration.

The following table provides the information about the verification procedure to check the status of VM properties:

VM Properties	Verification Procedure
WSL2	<p>Log on to the CVM as a Nutanix user using SSH, and run the following command:</p> <pre>nutanix@cvm\$ acli vm.get &lt;VM_NAME&gt;   grep 'hardware_virtualization'</pre> <p>Replace <code>&lt;VM_NAME&gt;</code> with the actual VM name at your site.</p> <p>The following attribute value confirms WSL2 is enabled:</p> <pre>hardware_virtualization: True</pre>
GPU	<ol style="list-style-type: none"><li>Log in to Prism Central.</li><li>Select the <b>Infrastructure</b> application from <a href="#">Application Switcher Function</a>, and navigate to <b>Compute &amp; Storage &gt; VMs</b> from the <b>Navigation Bar</b>. For information about the <b>Navigation Bar</b>, see <a href="#">Application-specific Navigation Bar</a>.</li></ol> <p>The system displays the <b>List</b> tab by default with all the VMs across registered clusters in <b>Nutanix</b> environment.</p> <ol style="list-style-type: none"><li>Apply the <b>GPU TYPE</b> filter from the <b>Modify Filter</b> option.</li></ol> <p>The system displays the VMs based on the selected GPU criteria.</p>

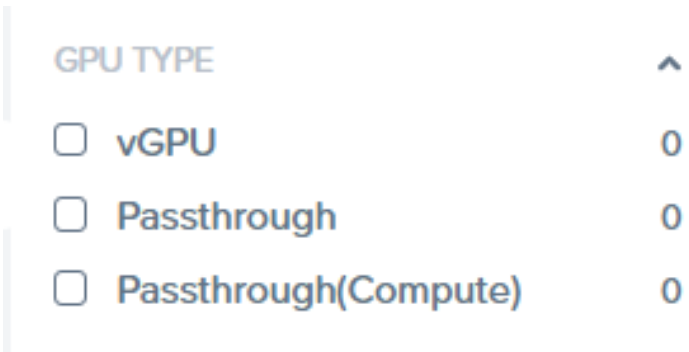


Figure 15: GPU TYPE Filter

Verifying Affinity Policy Association

This section describes how to verify the VM-Host affinity from Prism Element web console and affinity policies mapped to the VM from Prism Central.

Procedure

To verify affinity policy association for a VM, perform the following checks:

- [Checking VM-Host Affinity From Prism Element](#) on page 130.
- [Checking Affinity Policies of a VM From Prism Central](#) on page 130.



## Checking VM-Host Affinity From Prism Element

### About this task

Using Prism Element, you can define the VM-host affinity policy that controls the placement of a VM. You can use this policy to specify that a selected VM can only run on the members of the affinity host list.

### Procedure

To verify the VM-Host affinity, perform the following steps:

1. Log in to Prism element web console.
2. Navigate to **VM > Table**, and double-click the target VM name. The system displays the **Update VM** window.
3. Under **VM Host Affinity**, check the hosts specified for the VM.

#### Note:

VMs with Host affinity policies can only be migrated to the hosts specified in the affinity policy during an HA event. If only one host is specified, the VMs cannot be migrated.

## Checking Affinity Policies of a VM From Prism Central

### About this task

Using Prism Central, you can define the affinity policy based on VM Categories and Host categories.

### Procedure

To verify the affinity policy associated with a VM, perform the following steps:

1. Log in to Prism Central.
2. Select the **Infrastructure** application from [Application Switcher Function](#), and navigate to **Compute & Storage > VMs** from the **Navigation Bar**. For information about the **Navigation Bar**, see [Application-specific Navigation Bar](#).

The system displays the **List** tab by default with all the VMs across registered clusters in **Nutanix** environment.

3. Click the target VM name.  
The system displays the details view of the target VM.
4. Click the **Categories** tab in the target VM's details view.  
The system displays the Categories and Affinity Policy associated with the VM.



## Defining Behaviour for Non-migratable VMs (GPU/CPU/PCI/Host Affinity configured VMs)

### About this task

When an attempt is made for a host to enter maintenance mode, the VMs with GPU passthrough, CPU passthrough, PCI passthrough, and host affinity policies are not migrated to the other hosts in the cluster and therefore may block the attempt to enter maintenance mode.

To overcome this situation, Nutanix recommends you define either of the following behavior for these non-migratable VMs when host enters the maintenance mode:

- Automatically shutdown these non-migratable VMs.
- Set a block migration indication for these non-migratable VM.

### Procedure

To define the above behaviors for non-migratable VMs, perform the following steps:

1. Log on to CVM with SSH.
2. Run the following command to put the host into maintenance mode with default action set for non-migratable VMs:

```
nutanix@cvm$ acli host.enter_maintenance_mode <Hypervisor-IP-address> [wait="{ true | false }" ] [non_migratable_vm_action="{ acpi_shutdown | block }"]
```

Replace:

- **<Hypervisor-IP-address>** - with the actual host name at your site.
- **wait**: Set the wait parameter to true to wait for the host evacuation attempt to finish.
- **non\_migratable\_vm\_action**: By default, this parameter is set to *block*, which means VMs with GPU, CPU passthrough, PCI passthrough, and host affinity policies are not migrated or shut down when you put a node into maintenance mode. If you want to automatically shut down such VMs for the duration of the maintenance mode, set this parameter to *acpi\_shutdown*.

## Multichannel Support for AHV Live Migration Performance

Multichannel live migration facilitates the creation of multiple channels (or data streams) between the source and the destination host during live migration. For every channel, there is a separate TCP/unix socket and a thread that manages the data processing (sending or receiving the data) on that socket. The live migration feature then uses multiple send and receive threads to completely use the higher network bandwidths.

Live migration with multiple send and receive threads provides the following benefits:

- Support for the usage of multiple NICs available in uplinks to transfer the data during live migration.

Note: If you have NICs configured in Active-Standby mode and only one NIC is used at a time, Nutanix recommends you perform link aggregation using LACP to use the combined network bandwidth of all the NICs. For more information about how to enable LACP, see [Enabling LACP and LAG \(AHV Only\)](#).

- Live migration with an optimum number of channels or data streams and utilization of the maximum network bandwidth between source and destination host.



For example, for a four-node cluster with an available bandwidth of 25 Gbps per node, a single live migration with multiple channels utilizes the complete network and improves the migration speed. The following table provides a brief comparison of the measurement attributes between live migration with multiple channels and live migration without multiple channels:

**Table 14: Measurement Attributes - Multichannel Live Migration**

Measurement Attributes	Live migration with Multiple Channels	Live Migration without Multiple Channels
Bandwidth Utilization at the source and destination host.	Utilizes more than 50 to 55 % bandwidth and offers approximately 55.54 Gbps bandwidth at both source and destination hosts.	Utilizes only 10 to 11 % bandwidth and offers approximately 10.57 to 11.1 Gbps bandwidth at both source and destination hosts.
Enter Maintenance Mode (EMM) task time	Significant reduction in EMM task time to approximately six times when compared to the single live migration without multiple channels.	

**Note:**

- The Multichannel functionality is supported with AHV version 20220304.10055 and AOS version 6.7 or above.
- Multichannel Live migration is only supported for host *Enter Maintenance Mode* and *Restore Locality* migration tasks. These migration tasks run in parallel during host evacuation and restore operations.
- Based on the workloads available, the system allocates the source host bandwidth during *Enter Maintenance Mode* migration task and destination host bandwidth during the *Restore Locality* migration task

During serial live migration, the CPU is unable to use the complete bandwidth if a single migration is in progress on a 10 Gbps or faster network. However, with multichannel support, the system performs the following actions:

- Uses multiple streams.
- Selects the right number of channels for transfer.
- Utilizes the complete network bandwidth, and reduces the host evacuation time during live migration.

**Note:** For reliability improvements, the system performs the internal serial retry for each migration that fails due to insufficient bandwidth at the source host.

The following table describes the system behavior when the multichannel support for live migration is enabled based on AOS and AHV releases:



Table 15: Multichannel Support - System Behavior

AOS	AHV Version	Is Multichannel Support Enabled?	System Behavior
AOS 6.6 or before.	Bundled AHV version	No	The system performs live migration using a single data stream for host evacuation and restore operations. In this case, a single CPU is not able to use the network bandwidth completely. The system can only use 16 Gbps for host evacuation, which means only two live migrations at a time, as 8 Gbps is available for single live migration.
AOS 6.7 or later	20220304.10055 or recommended AHV version for later AOS releases.	Yes	<p>The system utilizes the complete network bandwidth using multiple channels or data streams for the following migration tasks:</p> <ul style="list-style-type: none"> <li>• <i>Enter Maintenance Mode</i> task for host evacuation at a source host.</li> <li>• <i>Restore Locality</i> task for restore at the destination host.</li> </ul> <p>Based on the network workload, the system assigns the maximum required bandwidth to the data streams for live migration.</p>

## Adaptive CPU Management during Live Migration

VM live migration involves migrating a VM from one host to another host without any noticeable VM downtime. During this live migration process, the system transfers the VM memory from the source host (where the VM is currently running) to the destination host. Since the VM is still running on the source host, a part or the entire VM's memory might get written again. The rewritten memory is called the dirty memory that needs to be transferred again to the destination host. The dirty memory transfer process goes on for iterations until we have a very small size of dirty memory that can be transferred in a smaller downtime window. The high dirty



memory rate impacts the VM migration time and success rate, especially when the network bandwidth is limited for VM Live migration.

AHV provides a seamless mechanism to handle these dirty memory build-ups. AHV intelligently detects the vCPUs that are involved in actively dirtying memory and manages their utilization dynamically without any impact on the vCPUs that are not dirtying the memory. The adaptive CPU management in AHV helps you to automatically optimize the network bandwidth consumption, increase the success rate for VM migrations, and reduce the VM migration time significantly.

## On-Demand Cross-Cluster Live Migration

On-demand cross-cluster live migration (CCLM) enables you to migrate guest VMs (and all of their associated metadata like VM categories) across AHV clusters registered to the same or different Prism Central availability zones (AZs). You can live migrate (which ensures zero VM downtime) the guest VMs without the need to protect them with synchronous replication schedules or set up Disaster Recovery from the Prism Central web console.

Note: AHV Metro Availability also provides CCLM capability that is based on the Disaster Recovery workflows. With AHV Metro CCLM, you can live migrate only the guest VMs protected with synchronous replication schedules. For more information on AHV Metro CCLM, see [Cross-Cluster Live Migration](#) in the *Nutanix Disaster Recovery Guide*.

You can live migrate a VM in the following situations:

- In an overlay subnet in a Prism Central AZ to an overlay subnet in another Prism Central AZ
- In an overlay subnet in a virtual private cloud (VPC) in a Prism Central AZ to an overlay subnet in another VPC in the same Prism Central AZ
- Within the same VPC in a Prism Central AZ
- In a VLAN subnet in a Prism Central AZ to a VLAN subnet in another Prism Central AZ
- In a VLAN subnet in a Prism Central AZ to another VLAN subnet in the same Prism Central AZ

A Layer 2 (L2) network extension creates a tunnel between the source and destination subnets so that the VMs in these subnets can communicate with each other. To ensure that dependent VMs on the source cluster and the destination cluster can communicate with the migrating VM throughout the live migration process, connect the source and the destination subnets using an L2 network extension before you live migrate a VM in the following scenarios:

- From an overlay subnet in a Prism Central AZ to an overlay subnet in another Prism Central AZ
- From an overlay subnet in a VPC in a Prism Central AZ to an overlay subnet in another VPC in the same Prism Central AZ
- From a VLAN subnet in a Prism Central AZ to a VLAN subnet in another Prism Central AZ
- From a VLAN subnet in a Prism Central AZ to another VLAN subnet in the same Prism Central AZ

Tip: An L2 network extension is not required between the source and the destination subnets if you live migrate a guest VM within the same VPC subnet.

For successful live migration, Nutanix recommends that the round-trip latency between the clusters be 40 ms or less.



## On-Demand CCLM Requirements

This section describes the requirements for live migrating guest VMs.

Table 16: Nutanix Software Requirements

Nutanix Software	Software Version Required
AOS	<ul style="list-style-type: none"><li>6.7 or later.</li></ul> <p>While upgrading the AOS version from 6.6 to 6.7, ensure that the upgrade is completed before performing on-demand migration of the guest VM.</p> <ul style="list-style-type: none"><li>For OD-CCLM of vTPM-enabled guest VMs, 6.8 or later.</li></ul> <p>Both the primary and recovery clusters must be running AOS 6.8 and later.</p>
Prism Central	<ul style="list-style-type: none"><li>pc.2023.3 or later.</li><li>For OD-CCLM of vTPM-enabled guest VMs pc.2024.1 or later.</li></ul>

Table 17: Prism Central User Requirements

Prism Central User	Permissions
Administrator	Only the administrator can perform on-demand live migration between the clusters registered to different Prism Central deployments.
Non-administrator	<p>Non-administrators can perform on-demand live migration only between the clusters registered to the Prism Central deployment. The user must have the following permissions:</p> <ul style="list-style-type: none"><li>View_Prism_Central</li><li>View_Virtual_Machine</li><li>View_Subnet</li><li>View_Availability_Zone</li><li>Allow_Cross_Cluster_VM_Migration</li><li>View_Cluster</li></ul>

- Both the source and the destination subnet must have the same IP network prefix.
- If you did not enable the advanced processor compatibility feature for the guest VM, ensure that both the source and the destination cluster have identical CPU feature sets (set of



CPU flags). Otherwise, live migration fails. For more information, see [Advanced Processor Compatibility in AHV](#) in the *AHV Administration Guide*.

Live migration can happen even when the CPU types of the source and destination clusters do not match exactly. The destination cluster must support the superset of the CPU features of the source cluster.

- Both the primary and the recovery clusters must run on the same AHV version.
- Both the primary and the recovery clusters must have the same storage container name for the guest VMs.

A storage container with the same name as the one on the primary cluster must exist on the recovery cluster. For example, if a *SelfServiceContainer* storage container exists on the source cluster, the destination cluster must also have a *SelfServiceContainer* storage container.

- If the primary and the recovery cluster (Prism Element) are in different subnets, open the ports listed on [Ports and Protocols](#) for communication.

Tip: If the primary and the recovery cluster (Prism Element) are in the same subnet, you do not need to open the ports manually.

- To open the ports for communication from the primary cluster to the recovery cluster, run the following command on all CVMs of the source cluster:

```
nutanix@cvm$ allssh 'modify_firewall -f -r remote_cvm_ip,remote_virtual_ip -p 2030,2036,2073,2090 -i eth0'
```

Replace `remote_cvm_ip` with the IP address of the destination cluster CVM. For multiple CVMs, replace `remote_cvm_ip` with the IP addresses of the CVMs separated by comma.

Replace `remote_virtual_ip` with the virtual IP address of the recovery cluster.

- To open the ports for communication from the recovery cluster to the primary cluster, run the following command on all CVMs of the destination cluster:

```
nutanix@cvm$ allssh 'modify_firewall -f -r source_cvm_ip,source_virtual_ip -p 2030,2036,2073,2090 -i eth0'
```

Replace `source_cvm_ip` with the IP address of the source cluster CVM. For multiple CVMs, replace `source_cvm_ip` with the IP addresses of the CVMs separated by comma.

Replace `source_virtual_ip` with the virtual IP address of the primary cluster.

Note:

- Use the `eth0` interface only. `eth0` is the default CVM interface that shows up when you install AOS.
- For network segmentation enabled Nutanix cluster, use the `ntnx0` interface and run the following command:

```
nutanix@cvm$ allssh 'modify_firewall -f -r remote_cvm_ip,remote_virtual_ip -p 2030,2036,2073,2090 -i ntnx0'
```

Replace `remote_cvm_ip` with the IP address of the cluster CVM where you want to migrate the guest VMs. For multiple CVMs, replace `remote_virtual_ip` with the IP addresses of the CVMs separated by comma.

- The recovery cluster must have sufficient storage capacity to host the migrating guest VMs.
- The recovery cluster must be reachable from the source cluster.





- IP address management must be manually set up on both the primary and the recovery cluster.
- The guest VM to be live migrated must be powered on.

The Prism Central web console filters out the guest VMs that are powered off.

## On-Demand CCLM Limitations

Consider the following limitations before performing on-demand live migration of your guest VMs:

- On-demand live migration to Nutanix Cloud AZ (DRaaS) is not supported.
- On-demand live migration to Nutanix Cloud Cluster AZ (NC2) is not supported.
- On-demand live migration of guest VMs fails in the following scenarios:
  - Guest VMs are part of Flow Network Security policies.
  - Guest VMs are part of consistency groups.
  - Guest VMs have redundancy factor 1 (RF1).
  - GPU passthrough is enabled on guest VMs. For information on GPU passthrough verification, see [Checking Live Migration Status of a VM](#) in the AHV Administration Guide.
  - CPU passthrough is enabled on guest VMs. For information on GPU passthrough verification using acropolis CLI (accli), see [vm](#) in the [Command Reference Guide](#).
  - Guest VMs are vNUMA VMs.
  - Guest VMs are in a paused state or have NVMe disks attached.
  - Memory overcommit is enabled on guest VMs.
  - Guest VMs are connected to a managed VLAN (IPAM-enabled).
  - Guest VMs are upgraded before on-demand live migration starts.

When on-demand live migration is in progress, do not perform hotplug upgrades to the memory or CPU. Updates to NIC, disk, memory, and CPU are fatal while data is copied during migration.

  - Prism Central upgrade is in progress.

- AHV upgrade is in progress.

Similarly, if on-demand live migration is in progress, upgrading AHV fails.

- LCM upgrade is in progress.
- After migrating a guest VM in an overlay subnet in a virtual private cloud (VPC) to an overlay subnet in another VPC, Prism Central does not retain the floating IP address of the guest VM if the VPCs are attached to different external subnets. Therefore, you must reassign the floating IP address to the migrated guest VM to access the guest VM from the external network.

If the VPCs share the same external subnets, Prism Central retains the floating IP address of the guest VM.

- After migration to the recovery cluster, VM categories are not preserved if those categories do not exist on the recovery cluster.



- Data over the wire is not encrypted during the live migration.
- Automatic defragmentation on the recovery cluster is not supported.
- Guest VMs running on ESXi are not supported.

## On-Demand CCLM Best Practices

Nutanix recommends the following best practices for live migrating your guest VMs (and all of their associated metadata like VM categories):

- Ensure that you set up the relevant user permissions. For more information, see [On-Demand CCLM Requirements](#) on page 135.
- Enable the advanced processor compatibility feature for the guest VM and select the oldest CPU generation to maximize the migration capability of the guest VM. For more information, see [Advanced Processor Compatibility in AHV](#) in the *AHV Administration Guide*.
- When on-demand live migration is in progress, do not protect the guest VM with a DR protection policy.
- When on-demand live migration is in progress, do not perform Nutanix software upgrades. For more information, see [On-Demand CCLM Limitations](#) on page 137.
- Do not perform on-demand live migration when Nutanix software upgrades are in progress. For more information, see [On-Demand CCLM Limitations](#) on page 137.
- When on-demand live migration is in progress, do not change IP addresses or other network configurations of the guest VMs.
- Nutanix recommends migrating up to 10 large-size and 60 small-size guest VMs in parallel. While no limit exists to the number of parallel migrations, the time required for successful migration depends on the guest VM size.
- When an administrator user s the guest VMs created by a non-administrator to a Nutanix cluster registered to a different Prism Central deployment, the administrator user becomes the owner of the guest VM after the VM migration. The ownership remains the same irrespective of who migrated the guest VM when the live migration happens between the Nutanix clusters registered to the same Prism Central deployment.

## Performing On-Demand CCLM

If, due to a failure event (for example, scheduled maintenance of guest VMs) at the primary cluster, you must migrate your applications to another AHV cluster without VM downtime, perform an on-demand live migration to the recovery cluster.

### Before you begin

See [On-Demand CCLM Requirements](#) on page 135, [On-Demand CCLM Limitations](#) on page 137, and [On-Demand CCLM Best Practices](#) on page 138.

### About this task

To live migrate the guest VMs, follow these steps at the primary AZ:

### Procedure

1. Log in to the Prism Central web console.



2. From **Application Switcher Function**, select the **Infrastructure** application, and navigate to **Compute & Storage > VMs** from the **Navigation Bar**.
3. Select the guest VMs to live migrate.

Warning: To see the guest VM configurations that cannot be migrated, see [On-Demand CCLM Limitations](#) on page 137.

4. Click **Migrate Across Clusters** from the **Actions** drop-down menu.
5. Do the following in the Destination Cluster tab:
  - a. From the dropdown menu, under **Destination Location**, select the AZ to live migrate your guest VMs.
  - b. From the dropdown menu, under **Destination Cluster**, select the AHV cluster to live migrate your guest VMs.
  - c. Click **Next**.
6. Do the following in the Network tab:
  - a. From the dropdown menu under **Destination Subnets**, select the network to map the subnet of the migrated guest VMs.
  - b. Click **Next**.
7. Do the following in the Migration Checks tab:
  - a. Review the status of the selected guest VMs after the prechecks run automatically. If any precheck fails, resolve the issue that is causing the failure and click **check again**.
  - b. When all the **Checks** under the **Selected VMs** show OK, click **Migrate**.  
The selected guest VMs migrate to the recovery cluster with zero VM downtime.

## Virtual Machine Snapshots

You can generate snapshots of virtual machines or VMs. You can generate snapshots of VMs manually or automatically. Some of the purposes that VM snapshots serve are as follows:

- Disaster recovery
- Testing - as a safe restoration point in case something went wrong during testing.
- Migrate VMs
- Create multiple instances of a VM.

Snapshot is a point-in-time state of entities such as VM and Volume Groups, and used for restoration and replication of data.. You can generate snapshots and store them locally or remotely. Snapshots are mechanism to capture the delta changes that has occurred over time. Snapshots are primarily used for data protection and disaster recovery. Snapshots are not autonomous like backup, in the sense that they depend on the underlying VM infrastructure and other snapshots to restore the VM. Snapshots consume less resources compared to a full autonomous backup. Typically, a VM snapshot captures the following:

- The state including the power state (for example, powered-on, powered-off, suspended) of the VMs.



- The data includes all the files that make up the VM. This data also includes the data from disks, configurations, and devices, such as virtual network interface cards.

### VM Snapshots and Snapshots for Disaster Recovery

The VM Dashboard only allows you to generate VM snapshots manually. You cannot select VMs and schedule snapshots of the VMs using the VM dashboard. The snapshots generated manually have very limited utility.

Note: These snapshots (stored locally) cannot be replicated to other sites.

You can schedule and generate snapshots as a part of the disaster recovery process using Nutanix DR solutions. AOS generates snapshots when you protect a VM with a protection domain using the Data Protection dashboard in Prism Web Console. For more information, see [Snapshots](#) in the *Data Protection and Recovery with Prism Element Guide*. Similarly, AOS generates recovery points (snapshots are called recovery points in Prism Central) when you protect a VM with a protection policy. For more information about protection policies, see [Protection Policies View](#) in *Nutanix Disaster Recovery Guide*.

For example, in the Data Protection dashboard in Prism Web Console, you can create schedules to generate snapshots using various RPO schemes such as asynchronous replication with frequency intervals of 60 minutes or more, or NearSync replication with frequency intervals of as less as 20 seconds up to 15 minutes. These schemes create snapshots in addition to the ones generated by the schedules, for example, asynchronous replication schedules generate snapshots according to the configured schedule and, in addition, an extra snapshot every 6 hours. Similarly, NearSync generates snapshots according to the configured schedule and also generates one extra snapshot every hour.

Similarly, you can use the options in the [Data Protection](#) entity of Prism Central to generate recovery points using the same RPO schemes.

## Windows VM Provisioning

This section provides information about the windows OS specifications, installation, and configurations that you can perform on a guest VM.

### Nutanix VirtIO for Windows

Nutanix VirtIO is a collection of drivers for para-virtual devices that enhance the stability and performance of virtual machines on AHV.

Nutanix VirtIO is available in two formats:

- *VirtIO ISO file* - Use it when VM does not yet have a Windows OS installed. For more information, see [Installing or Upgrading Nutanix VirtIO for Windows](#) on page 142.
- *VirtIO MSI installer file* - Use it to install or upgrade VirtIO when Windows OS is installed and running on a VM or VM already has VirtIO installed.

The VirtIO MSI installer file is also bundled with Nutanix Guest Tools (NGT). Install NGT on a VM to install the Nutanix VirtIO Package on the VM. For information about how to install NGT, see [NGT Installation](#) in the *Prism Web Console Guide*.

The following table describes the NGT behavior based on AOS release:



Table 18: NGT Behavior - AOS Release

AOS Release	NGT Behavior
Prior to AOS 6.6	<p>NGT includes the VM Mobility package which is a re-packaging of VirtIO. The repackaging is done with additional changes to enable a built-in driver that is pre-installed in Windows but not enabled by default. This driver is used to enable the SCSI controller required by some Windows editions for seamless mobility between different types of hypervisors.</p> <p>In this case, the VM Mobility package uses the same version numbering as VirtIO.</p>
AOS 6.6 and above	<p>NGT contains both a VirtIO and a VM Mobility packages. The Nutanix VirtIO contains all VirtIO drivers, and the VM Mobility package is no longer re-packaged with VirtIO drivers and contains only the change to enable SCSI controller.</p> <p>The NGT release is aligned with the AOS release and the NGT contains the latest VirtIO package available at the time of NGT release. However, there is no back-porting of NGT release alignment with the previous AOS releases.</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>Caution: If you install an older version of NGT, the latest VirtIO version, even if installed, is replaced by the older VirtIO version.</p> </div> <p>The VirtIO release is not aligned with the AOS release. To ensure that you have the latest VirtIO drivers, either install the latest NGT version or update the drivers using the latest VirtIO package available on the Nutanix Support portal. For more information, see <a href="#">Installing or Upgrading Nutanix VirtIO for Windows</a> on page 142.</p>

### VirtIO Requirements

Requirements for Nutanix VirtIO for Windows.

VirtIO supports the following operating systems:

- Microsoft Windows server version: Windows 2008 R2 or later
- Microsoft Windows client version: Windows 7 or later

Note: On Windows 7 and Windows Server 2008 R2, install Microsoft KB3033929 or update the operating system with the latest Windows Update to enable support for SHA2 certificates.

Caution: The VirtIO installation or upgrade may fail if multiple Windows VSS snapshots are present in the guest VM. The installation or upgrade failure is due to the timeout that occurs during installation of *Nutanix VirtIO SCSI pass-through controller driver*.

It is recommended to clean up the VSS snapshots or temporarily disconnect the drive that contains the snapshots. Ensure that you only delete the snapshots that are no longer needed. For more information about how to observe the VirtIO installation or upgrade failure that occurs due to availability of multiple Windows VSS snapshots, see [KB-12374](#).

## Installing or Upgrading Nutanix VirtIO for Windows

Download Nutanix VirtIO and the Nutanix VirtIO Microsoft installer (MSI). The MSI installs and upgrades the Nutanix VirtIO drivers.

### Before you begin

Make sure that your system meets the VirtIO requirements described in [VirtIO Requirements](#) on page 141.

### About this task

If you have already installed Nutanix VirtIO, use the following procedure to upgrade VirtIO to a latest version. If you have not yet installed Nutanix VirtIO, use the following procedure to install Nutanix VirtIO.

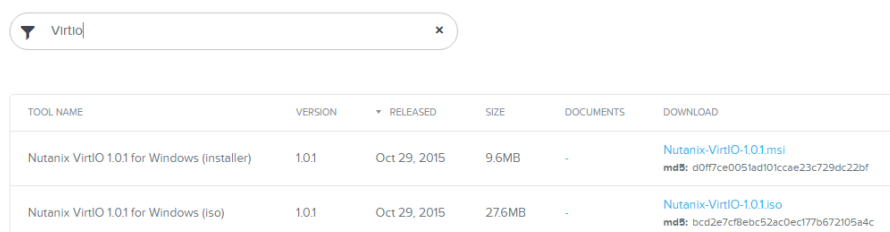
### Procedure

1. Go to the [Downloads page of VirtIO package](#) on Nutanix Support portal.

Alternatively, you can also click **AHV** in the [Downloads](#) page on Nutanix Support portal, and select **VirtIO** from the dropdown list.

2. Select the appropriate VirtIO package.

- » If you are creating a new Windows VM, download the ISO file. The installer is available on the ISO if your VM does not have Internet access.
- » If you are updating drivers in a Windows VM, download the MSI installer file.



The screenshot shows a search filter bar with the text 'Virtio' and a dropdown arrow. Below it is a table with two rows of search results. The table has columns for Tool Name, Version, Released, Size, Documents, and Download. The first row is for 'Nutanix VirtIO 1.0.1 for Windows (installer)' and the second row is for 'Nutanix VirtIO 1.0.1 for Windows (iso)'.

TOOL NAME	VERSION	RELEASED	SIZE	DOCUMENTS	DOWNLOAD
Nutanix VirtIO 1.0.1 for Windows (installer)	1.0.1	Oct 29, 2015	9.6MB	-	<a href="#">Nutanix-VirtIO-1.0.1.msi</a> md5: d0f7ce0051ad101ccee23c729dc22bf
Nutanix VirtIO 1.0.1 for Windows (iso)	1.0.1	Oct 29, 2015	27.6MB	-	<a href="#">Nutanix-VirtIO-1.0.1.iso</a> md5: bcd2e7cf8ebc52ac0ec177b672105a4c

Figure 16: Search filter and VirtIO options

3. Run the selected package.

- » For the ISO: Upload the ISO to the cluster, as described in the [Configuring Images](#) topic in *Prism Element Web Console Guide*.
- » For the MSI: open the download file to run the MSI.

4. Read and accept the Nutanix VirtIO license agreement. Click **Install**.

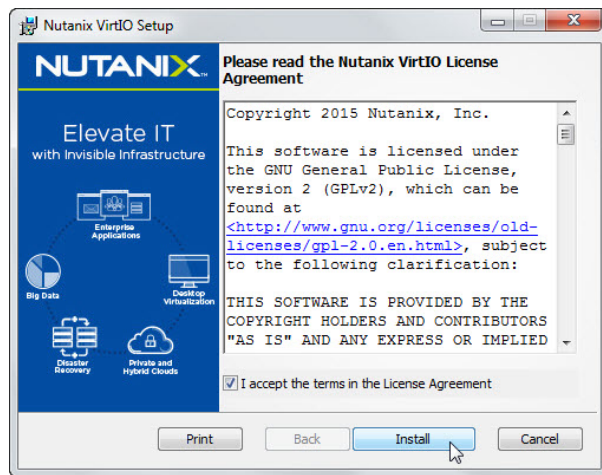


Figure 17: Nutanix VirtIO Windows Setup Wizard

The Nutanix VirtIO setup wizard shows a status bar and completes installation.

### Manually Installing or Upgrading Nutanix VirtIO

Manually install or upgrade Nutanix VirtIO.

#### Before you begin

Make sure that your system meets the VirtIO requirements described in [VirtIO Requirements](#) on page 141.

#### About this task

Note: To automatically install Nutanix VirtIO, see [Installing or Upgrading Nutanix VirtIO for Windows](#) on page 142.

If you have already installed Nutanix VirtIO, use the following procedure to upgrade VirtIO to a latest version. If you have not yet installed Nutanix VirtIO, use the following procedure to install Nutanix VirtIO.

#### Procedure

1. Go to the [Downloads page of VirtIO package](#) on Nutanix Support portal.

Alternatively, you can also click **AHV** in the [Downloads](#) page on Nutanix Support portal, and select **VirtIO** from the dropdown list.

2. Do one of the following:

- » Extract the VirtIO ISO into the same VM where you load Nutanix VirtIO, for easier installation.

If you choose this option, proceed directly to step 7.

- » Download the VirtIO ISO for Windows to your local machine.

If you choose this option, proceed to step 3.

3. Upload the ISO to the cluster, as described in the [Configuring Images](#) topic of Prism Element Web Console Guide.
4. Locate the VM where you want to install the Nutanix VirtIO ISO and update the VM.
5. Add the Nutanix VirtIO ISO by clicking **Add New Disk** and complete the indicated fields.
  - **TYPE:** CD-ROM
  - **OPERATION:** CLONE FROM IMAGE SERVICE
  - **BUS TYPE:** IDE
  - **IMAGE:** Select the Nutanix VirtIO ISO
6. Click **Add**.
7. Log on to the VM and browse to **Control Panel > Device Manager**.



8.

Note: Select the **x86** subdirectory for 32-bit Windows, or the **amd64** for 64-bit Windows.

Open the devices and select the specific Nutanix drivers for download. For each device, right-click and **Update Driver Software** into the drive containing the VirtIO ISO. For each device, follow the wizard instructions until you receive installation confirmation.

a. **System Devices > Nutanix VirtIO Balloon Drivers**

b. **Network Adapter > Nutanix VirtIO Ethernet Adapter.**

c. **Processors > Storage Controllers > Nutanix VirtIO SCSI pass through Controller**

The Nutanix VirtIO SCSI pass-through controller prompts you to restart your system. Restart at any time to install the controller.



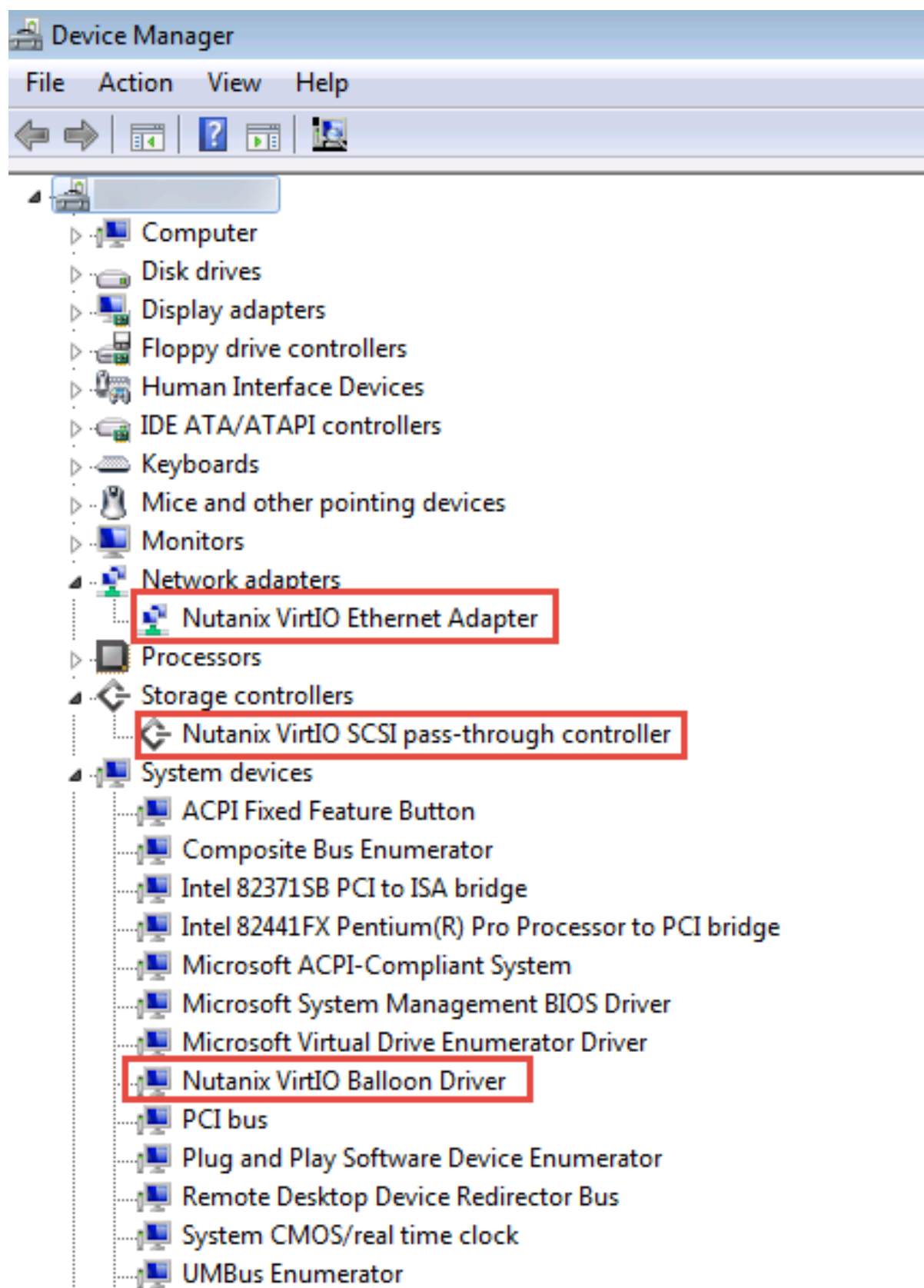


Figure 18: List of Nutanix VirtIO downloads

## Creating a Windows VM on AHV with Nutanix VirtIO

Create a Windows VM in AHV, or migrate a Windows VM from a non-Nutanix source to AHV, with the Nutanix VirtIO drivers.

### Before you begin

- Upload the Windows installer ISO to your cluster as described in the [Configuring Images](#) topic in *Web Console Guide*.
- Upload the Nutanix VirtIO ISO to your cluster as described in the [Configuring Images](#) topic in *Web Console Guide*.

### About this task

To install a new or migrated Windows VM with Nutanix VirtIO, complete the following.

### Procedure

1. Log on to the Prism web console using your Nutanix credentials.
2. At the top-left corner, click **Home > VM**.  
The VM page appears.

3. Click **+ Create VM** in the corner of the page.  
The Create VM dialog box appears.

Create VM

?

×

General Configuration

Name

VM-site1

Description

VM for site 1

Timezone

(UTC) UTC

Cluster

Use this VM as an agent VM

Compute Details

vCPU(s)

1

Number Of Cores Per vCPU

1

Memory

1

GIB

Boot Configuration

Legacy BIOS

Set Boot Priority

Default Boot Order (CD-ROM, Disk, Network)

UEFI

Disks

Add New Disk

Type	Address	Parameters	
CD-ROM	sata.0	EMPTY=true; BUS=sata	

Volume Groups

Please create a VM before you can add a volume group.

Add Volume Group

Network Adapters (NIC)

Add New NIC

VLAN ID	VIRTUAL SWITCH	PRIVATE IP	MAC	
890	vs0	-	-	
ossNet				

VM Host Affinity

You haven't pinned the VM to any hosts yet.

Set Affinity

Custom Script



4. Complete the indicated fields.
  - a. **NAME:** Enter a name for the VM.
  - b. **Description** (optional): Enter a description for the VM.
  - c. **Timezone:** Select the timezone that you want the VM to use. If you are creating a Linux VM, select **(UTC) UTC**.

Note:

The RTC of Linux VMs must be in UTC, so select the UTC timezone if you are creating a Linux VM.

Windows VMs preserve the RTC in the local timezone, so set up the Windows VM with the hardware clock pointing to the desired timezone.

- d. **Number of Cores per vCPU:** Enter the number of cores assigned to each virtual CPU.
  - e. **MEMORY:** Enter the amount of memory for the VM (in GiB).
5. If you are creating a Windows VM, add a Windows CD-ROM to the VM.
  - a. Click the pencil icon next to the CD-ROM that is already present and fill out the indicated fields.

- **OPERATION: CLONE FROM IMAGE SERVICE**
- **BUS TYPE: SATA**

Note: If **Secured Boot** is enabled for **UEFI** boot configuration, the **IDE** interface is not supported for CD-ROM. You must use **SATA** as the **BUS TYPE** for CD-ROM. For more information, see [Secure Boot Support for VMs](#) on page 187.

- **IMAGE:** Select the Windows OS install ISO.
- b. Click **Update**.  
The current CD-ROM opens in a new window.

6. Add the Nutanix VirtIO ISO.

- a. Click **Add New Disk** and complete the indicated fields.

- **TYPE: CD-ROM**
- **OPERATION: CLONE FROM IMAGE SERVICE**
- **BUS TYPE: SATA**

Note: If **Secured Boot** is enabled for **UEFI** boot configuration, the **IDE** interface is not supported for Nutanix VirtIO ISO. You must use **SATA** as the **BUS TYPE** for Nutanix VirtIO ISO. For more information, see [Secure Boot Support for VMs](#) on page 187.

- **IMAGE:** Select the Nutanix VirtIO ISO.
- b. Click **Add**.



7. Add a new disk for the hard drive.

a. Click **Add New Disk** and complete the indicated fields.

- **TYPE: DISK**
- **OPERATION: ALLOCATE ON STORAGE CONTAINER**
- **BUS TYPE: SCSI**

Note: SCSI bus is the preferred bus type and it is used in most cases. For more information, see [Compatibility and Interoperability Matrix for AHV Guest OS](#).

- **STORAGE CONTAINER:** Select the appropriate storage container.
- **SIZE:** Enter the number for the size of the hard drive (in GiB).

b. Click **Add** to add the disk driver.

8. If you are migrating a VM, create a disk from the disk image.

a. Click **Add New Disk** and complete the indicated fields.

- **TYPE: DISK**
- **OPERATION: CLONE FROM IMAGE**
- **BUS TYPE: SCSI**

Note: SCSI bus is the preferred bus type and it is used in most cases. For more information, see [Compatibility and Interoperability Matrix for AHV Guest OS](#).

- **CLONE FROM IMAGE SERVICE:** Click the drop-down menu and choose the image you created previously.

b. Click **Add** to add the disk driver.

9. Optionally, after you have migrated or created a VM, add a network interface card (NIC).

a. Click **Add New NIC**.

b. In the **VLAN ID** field, choose the VLAN ID according to network requirements and enter the IP address, if necessary.

c. Click **Add**.

10. Click **Save**.

### What to do next

Install Windows by following [Installing Windows on a VM](#) on page 151.

## Installing Windows on a VM

Install a Windows virtual machine.

### Before you begin

Create a Windows VM.



## Procedure

1. Log on to the web console.
2. Click **Home > VM** to open the VM dashboard.
3. Select the Windows VM.
4. In the center of the VM page, click **Power On**.
5. Click **Launch Console**.  
The Windows console opens in a new window.
6. Select the desired language, time and currency format, and keyboard information.
7. Click **Next > Install Now**.  
The Windows setup dialog box shows the operating systems to install.
8. Select the Windows OS you want to install.
9. Click **Next** and accept the license terms.
10. Click **Next > Custom: Install Windows only (advanced) > Load Driver > OK > Browse**.





11. Choose the Nutanix VirtIO driver.
  - a. Select the Nutanix VirtIO CD drive.
  - b. Expand the Windows OS folder and click **OK**.

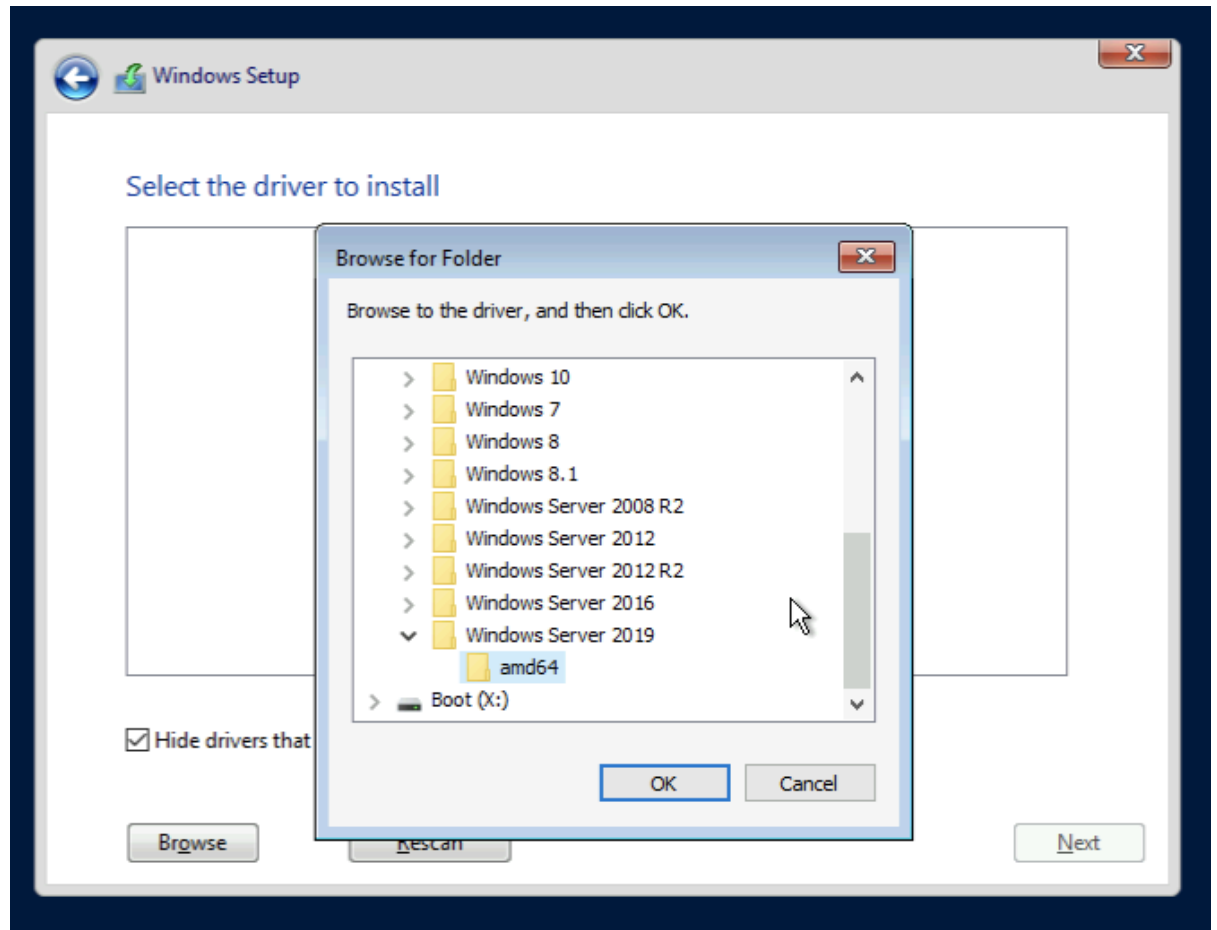


Figure 20: Select the Nutanix VirtIO drivers for your OS

The Select the driver to install window appears.

12. Select the VirtIO SCSI driver (`vioscsi.inf`) and click **Next**.

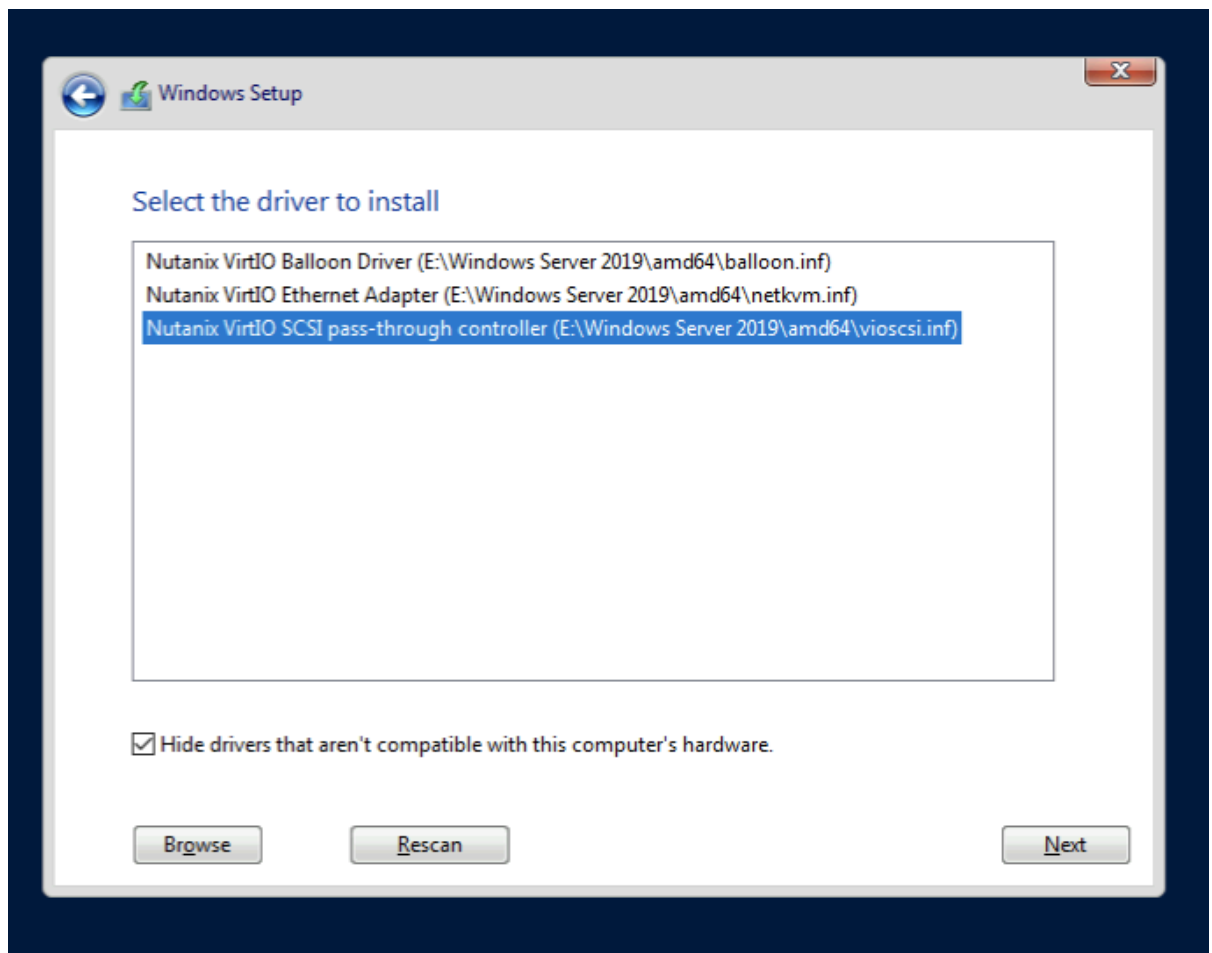


Figure 21: Select the Driver for Installing Windows on a VM

The **amd64** folder contains drivers for 64-bit operating systems. The **x86** folder contains drivers for 32-bit operating systems.

Note: From Nutanix VirtIO driver version 1.1.5, the driver package contains Windows Hardware Quality Lab (WHQL) certified driver for Windows.

13. Select the allocated disk space for the VM and click **Next**.  
Windows shows the installation progress, which can take several minutes.
14. Enter your user name and password information and click **Finish**.  
Installation can take several minutes.  
Once you complete the login information, Windows setup completes installation.
15. Follow the instructions in [Installing or Upgrading Nutanix VirtIO for Windows](#) on page 142 to install other drivers which are part of Nutanix VirtIO package.

## Windows Defender Credential Guard Support in AHV

AHV enables you to use the Windows Defender Credential Guard security feature on Windows guest VMs.

Windows Defender Credential Guard feature of Microsoft Windows operating systems allows you to securely isolate user credentials from the rest of the operating system. By that means, you can protect guest VMs from credential theft attacks such as Pass-the-Hash or Pass-The-Ticket.

See the Microsoft documentation for more information about the Windows Defender Credential Guard security feature.

### Windows Defender Credential Guard Architecture in AHV

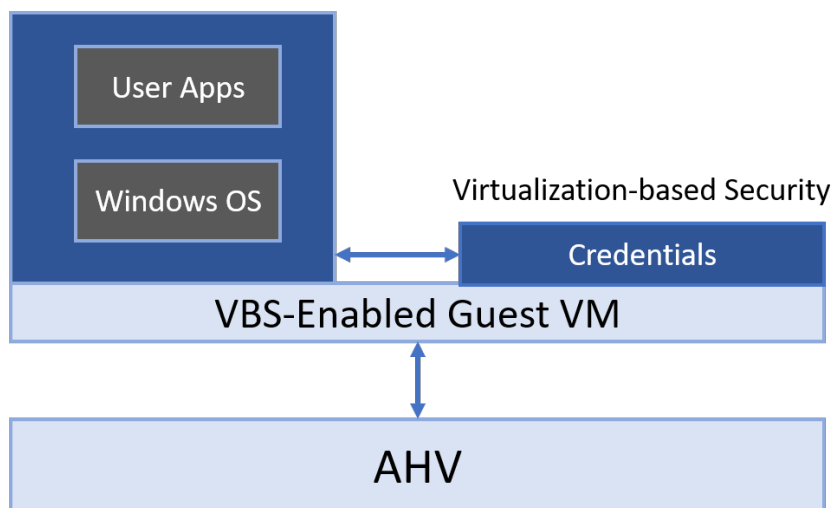


Figure 22: Architecture

Windows Defender Credential Guard uses Microsoft virtualization-based security to isolate user credentials in the virtualization-based security (VBS) module in AHV. When you enable Windows Defender Credential Guard on an AHV guest VM, the guest VM runs on top of AHV running both the Windows OS and VBS. Each Windows OS guest VM, which has credential guard enabled, has a VBS to securely store credentials.

### Windows Defender Credential Guard Requirements

Ensure the following to enable Windows Defender Credential Guard:

1. AOS, AHV, and Windows Server versions support Windows Defender Credential Guard:
  - AOS version must be 5.19 or later
  - AHV version must be AHV 20201007.1 or later
  - Windows Server version must be Windows server 2016 or later, Windows 10 Enterprise or later and Windows Server 2019 or later
2. UEFI, Secure Boot, and machine type q35 are enabled in the Windows VM from AOS.

The Prism Element workflow to enable Windows Defender Credential Guard includes the workflow to enable these features.

## Limitations

- If you enable Windows Defender Credential Guard for your AHV guest VMs, the following optional configurations are not supported:
  - Virtual GPU configurations.
  - vTPM (Virtual Trusted Platform Modules) to store MS policies.

Note: vTPM is supported with AOS 6.5.1 or later and AHV 20220304.242 or later release versions only.

- DMA protection (vIOMMU).
- Cross hypervisor DR of Credential Guard VMs.
- Starting with AOS 6.8 release, AHV supports the live migration of the guest VMs that have windows credential guard enabled. When you upgrade to AOS 6.8 or later release and enable HA reservation for the cluster, the system reserves the resources to accommodate the HA event. However, if you have the existing VMs in the cluster that have Windows Credential guard enabled, some guest VMs (both with or without windows credential guard enabled) might fail to start due to the unavailability of sufficient resources in the cluster. Ensure that you have sufficient resources available in the cluster to accommodate the HA reservation.

For more information on windows credential guard, see [Windows Defender Credential Guard Support in AHV](#).

For information on how to enable HA reservation, see [Enabling High Availability for the Cluster](#) section in *Prism Web Console Guide*

Caution: Use of Windows Defender Credential Guard in your AHV clusters impacts VM performance. If you enable Windows Defender Credential Guard on AHV guest VMs, VM density drops by ~15–20%. This expected performance impact is due to nested virtualization overhead added as a result of enabling credential guard.

## Enabling Windows Defender Credential Guard Support in AHV Guest VMs

You can enable Windows Defender Credential Guard when you are either creating a VM or updating a VM.

### About this task

Perform the following procedure to enable Windows Defender Credential Guard:

### Procedure

1. Enable Windows Defender Credential Guard when you are either creating a VM or updating a VM. Do one of the following:
  - » If you are creating a VM, see step 2.
  - » If you are updating a VM, see step 3.



2. If you are creating a Windows VM, do the following:
  - a. Log on to the Prism Element web console.
  - b. In the VM dashboard, click **Create VM**.
  - c. Fill in the mandatory fields to configure a VM.
  - d. Under **Boot Configuration**, select **UEFI**, and then select the **Secure Boot** and **Windows Defender Credential Guard** options.

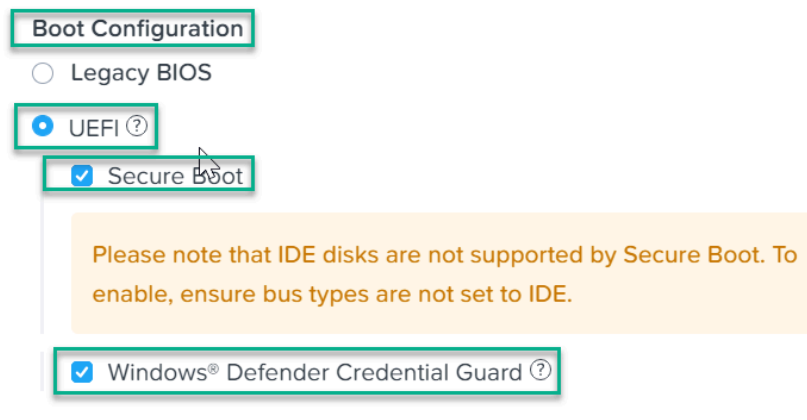


Figure 23: Enable Windows Defender Credential Guard

See [UEFI Support for VM](#) on page 181 and [Secure Boot Support for VMs](#) on page 187 for more information about these features.

- e. Proceed to configure other attributes for your Windows VM.  
See [Creating a Windows VM on AHV with Nutanix VirtIO](#) on page 147 for more information.
- f. Click **Save**.
- g. Turn on the VM.

3. If you are updating an existing VM, do the following:
  - a. Log on to the Prism Element web console.
  - b. In the VM dashboard, click the **Table** view, select the VM, and click **Update**.
  - c. Under **Boot Configuration**, select **UEFI**, and then select the **Secure Boot** and **Windows Defender Credential Guard** options.

Note:

If the VM is configured to use BIOS, install the guest OS again.

If the VM is already configured to use UEFI, skip the step to select Secure Boot.

See [UEFI Support for VM](#) on page 181 and [Secure Boot Support for VMs](#) on page 187 for more information about these features.

- d. Click **Save**.
  - e. Turn on the VM.
4. Enable Windows Defender Credential Guard in the Windows VM by using group policy.  
See the *Enable Windows Defender Credential Guard by using the Group Policy* procedure of the *Manage Windows Defender Credential Guard* topic in the Microsoft documentation to enable VBS, Secure Boot, and Windows Defender Credential Guard for the Windows VM.
5. Open command prompt in the Windows VM and apply the **Group Policy** settings:

```
> gpupdate /force
```

If you have not enabled Windows Defender Credential Guard (step 4) and perform this step (step 5), a warning similar to the following is displayed:

Updating policy...

Computer Policy update has completed successfully.

The following warnings were encountered during computer policy processing:

Windows failed to apply the {F312195E-3D9D-447A-A3F5-08DFFA24735E} settings.

{F312195E-3D9D-447A-A3F5-08DFFA24735E} settings might have its own log file. Please click on the "More information" link.

User Policy update has completed successfully.

For more detailed information, review the event log or run GPRESULT /H GPRReport.html from the command line to access information about Group Policy results.

Event Viewer displays a warning for the group policy with an error message that indicates Secure Boot is not enabled on the VM.

To view the warning message in Event Viewer, do the following:

- In the Windows VM, open **Event Viewer**.

- Go to **Windows Logs** -> **System** and click the warning with the **Source** as **GroupPolicy (Microsoft-Windows-GroupPolicy)** and **Event ID** as **1085**.

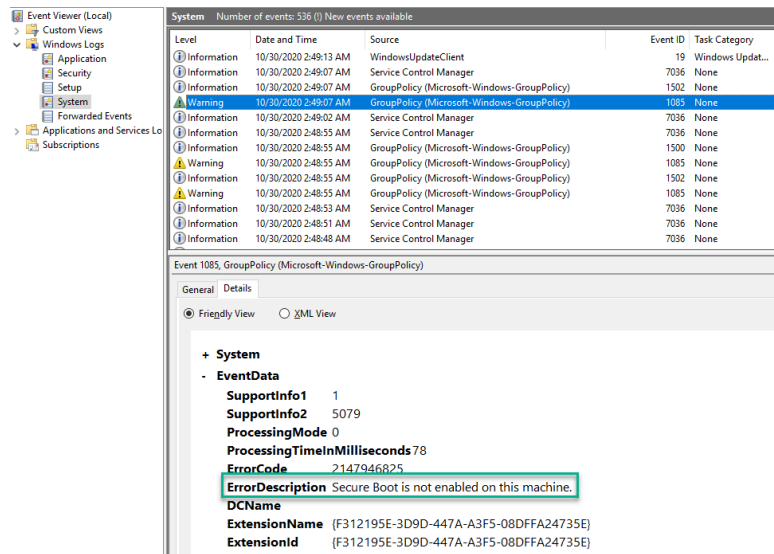


Figure 24: Warning in Event Viewer

Note: Ensure that you follow the steps in the order that is stated in this document to successfully enable Windows Defender Credential Guard.

6. Restart the VM.
7. Verify if Windows Defender Credential Guard is enabled in your Windows VM.
  - a. Start a Windows PowerShell terminal.
  - b. Run the following command.

```
PS > Get-CimInstance -ClassName Win32_DeviceGuard -Namespace 'root\Microsoft\Windows\DeviceGuard'
```

An output similar to the following is displayed.

```
PS > Get-CimInstance -ClassName Win32_DeviceGuard -Namespace 'root\Microsoft\Windows\DeviceGuard'
AvailableSecurityProperties           : {1, 2, 3, 5}
CodeIntegrityPolicyEnforcementStatus : 0
InstanceIdIdentifier                 : 4ff40742-2649-41b8-bdd1-e80fad1cce80
RequiredSecurityProperties           : {1, 2}
SecurityServicesConfigured           : {1}
SecurityServicesRunning              : {1}
UsermodeCodeIntegrityPolicyEnforcementStatus : 0
Version                             : 1.0
VirtualizationBasedSecurityStatus    : 2
```

Confirm that both **SecurityServicesConfigured** and **SecurityServicesRunning** have the value **{ 1 }**.

Alternatively, you can verify if Windows Defender Credential Guard is enabled by using System Information (msinfo32):

- In the Windows VM, open **System Information** by typing **msinfo32** in the search field next to the **Start** menu.
- Verify if the values of the parameters are as indicated in the following screen shot:

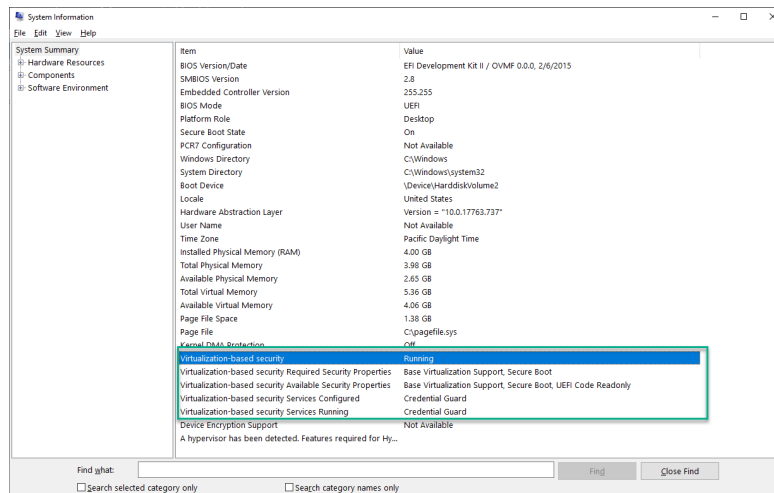


Figure 25: Verify Windows Defender Credential Guard

## Windows Subsystem for Linux (WSL2) Support on AHV

AHV supports WSL2 that enables you to run a Linux setup on a Windows OS without a dedicated VM and dual-boot environment.

For more information about WSL, refer to *What is the Windows Subsystem for Linux?* topic in Microsoft Technical Documentation.

### Note:

- Both hardware and software support are required to enable a guest VM to communicate with its nested guest VMs in a WSL2 setup.
- The system performance gets impacted in a WSL2 environment due to specific workloads and lack of hardware features in the processors. These attributes are required to enhance the virtualization environment.
- VM live migration is currently not supported for WSL. You must power off the VM during any AOS or AHV upgrades.

## Limitations

The following table provides the information about the limitations that are applicable for WSL2 based on the AOS and AHV version:



Table 19: Limitations for WSL2

AOS Version	AHV Version	Limitations for WSL2
AOS 6.5.1	AHV 20201105.30411 (default bundled AHV with AOS 6.5.1)	<p>The following optional configurations are not supported:</p> <ul style="list-style-type: none"> <li>• vTPM (Virtual Trusted Platform Modules) to store MS policies</li> <li>• Hosts with AMD CPUs</li> <li>• DMA protection (vIOMMU).</li> <li>• Nutanix Live Migration.</li> <li>• Cross hypervisor DR of WSL2 VM</li> </ul>
AOS 6.5.1 to AOS 6.6.2.8	AHV 20220304.242 to AHV-20220304.10057	<p>The following optional configurations are not supported:</p> <ul style="list-style-type: none"> <li>• Hosts with AMD CPUs</li> <li>• DMA protection (vIOMMU).</li> <li>• Nutanix Live Migration.</li> <li>• Cross hypervisor DR of WSL2 VM</li> </ul>
AOS 6.7 or later	<p>AHV-20230302.207 (default bundled AHV with AOS 6.7) or recommended AHV version for later AOS releases.</p> <div> <p>Note: From AOS 6.8 release onwards, AOS does not include the AHV installation bundle. Download the AHV installer bundle from the <a href="#">Downloads</a> page on Nutanix portal.</p> </div>	<p>The following optional configurations are not supported:</p> <ul style="list-style-type: none"> <li>• DMA protection (vIOMMU).</li> <li>• Nutanix Live Migration.</li> <li>• Cross hypervisor DR of WSL2 VM</li> </ul>

### Enabling WSL2 on AHV

This section provides the information on how to enable the WSL2 on AHV.

#### Before you begin

Ensure that the AOS 6.5.1 or later along with AHV 20220304.242 or later release versions are deployed at your site.

#### About this task

Note:



In the following procedure, ensure that you replace the <VM\_name> with the actual guest VM name.

To configure WSL2 on AHV:

### Procedure

1. Power off the guest VM on which you want to configure WSL2.
2. Log on to any CVM in the cluster with SSH.
3. Run the following command to enable the guest VM to support WSL2:

```
nutanix@CVM~ $ acli vm.update <VM_name> hardware_virtualization=true
```

Note: In case you need to create a new guest VM (Windows VM) on AHV with Nutanix VirtIO, see [Creating a Windows VM on AHV with Nutanix VirtIO](#) on page 147.

4. (Optional) Retrieve the guest VM details using the following command to check whether all the attributes are correctly set for the guest VM:

```
nutanix@CVM~ $ acli vm.get <VM_name>
```

Observe the following log attributes to verify whether the infrastructure to support WSL2 is configured successfully in the guest VM:

```
hardware_virtualization: True
```

5. Power on the guest VM using the following command:

```
nutanix@CVM~ $ acli vm.on <VM_name>
```

6. Enable WSL2 on Windows OS. For information on how to install WSL, refer [Install WSL](#) topic in Microsoft Technical Documentation.

## Affinity Policies for AHV

As an administrator of an AHV cluster, you can create VM-Host affinity policies for virtual machines on an AHV cluster. By defining these policies, you can control the placement of virtual machines on the hosts within a cluster.

Note: VMs with Host affinity policies can only be migrated to the hosts specified in the affinity policy. If only one host is specified, the VM cannot be migrated or started on another host during an HA event. For more information, see [Non-Migratable Hosts](#) on page 176.

You can define affinity policies for a VM at two levels:

### Affinity Policies defined in Prism Element

In Prism Element, you can define affinity policies at VM level during the VM create or update operation. You can use an affinity policy to specify that a particular VM can only run on the members of the affinity host list.

### Affinity Policies defined in Prism Central

In Prism Central, you can define category-based VM-Host affinity policies, where a set of VMs can be affinity to run only on a particular set of hosts. Category-based affinity policy enables you to easily manage affinities for a large number of VMs.



## Affinity Policies Defined in Prism Element

In Prism Element, you can define scheduling policies for virtual machines on an AHV cluster at a VM level. By defining these policies, you can control the placement of a virtual machine on specific hosts within a cluster.

For information about how the protection domain-based VM-Host affinity policies are handled during disaster recovery, see [Affinity Policies Handling - Protection-Domain Based DR Solution with On-prem Clusters Only](#).

You can define two types of affinity policies in Prism Element.

### VM-Host Affinity Policy

The VM-host affinity policy controls the placement of a VM. You can use this policy to specify that a selected VM can only run on the members of the affinity host list. This policy checks and enforces where a VM can be hosted when you power on or migrate the VM.

#### Note:

- If you choose to apply the VM-host affinity policy, it limits Acropolis HA and Acropolis Dynamic Scheduling (ADS) in such a way that a virtual machine cannot be powered on or migrated to a host that does not conform to requirements of the affinity policy as this policy is enforced mandatorily.
- The VM-host anti-affinity policy is not supported.
- VMs configured with host affinity settings retain these settings if the VM is migrated to a new cluster. Remove the VM-Host affinity policies applied to a VM that you want to migrate to another cluster, as the UUID of the host is retained by the VM and it does not allow the VM to restart on the destination cluster. When you attempt to protect such VMs, it is successful. However, some disaster recovery operations like migration fail and attempts to power on these VMs also fail.

You can define the VM-Host affinity policies by using Prism Element during the VM create or update operation. For more information, see [Creating a VM \(AHV\)](#).

### VM-VM Anti-Affinity Policy

You can use this policy to specify anti-affinity between the virtual machines. The VM-VM anti-affinity policy keeps the specified virtual machines apart in such a way that when a problem occurs with one host, you should not lose both the virtual machines.

#### Important:

The VM-VM anti-affinity policy is a preferential policy. The system does not block any VM operation, such as VM maintenance mode or manual live migration of the VM, even if there is a policy violation. For example, when you manually migrate one VM of a VM-VM pair with an anti-affinity policy, the policy is applied on a best-effort basis only.

The Acropolis Dynamic Scheduling (ADS) always attempts to maintain compliance with the VM-VM anti-affinity policy and ensures that the VM-VM anti-affinity policy is enforced on a best-effort basis. For example, if you manually migrate a VM and the migration leads to non-compliance with the VM-VM anti-affinity policy, ADS performs the following actions:

- Ignores compliance to VM-VM anti-affinity policy, if a host is specified during manual migration.



- Attempts to enforce the policy back into compliance on a best-effort basis, if a host is not specified during manual migration.

For more information on ADS, see [Acropolis Dynamic Scheduling in AHV](#) on page 8.

Note:

- Currently, you can only define VM-VM anti-affinity policy by using aCLI. For more information, see [Configuring VM-VM Anti-Affinity Policy](#) on page 164.
- The VM-VM affinity policy is not supported.
- If a VM is cloned that has the affinity policies configured, then the policies are not automatically applied to the cloned VM. However, if a VM is restored from a DR snapshot, the policies are automatically applied to the VM.

## Limitations of Affinity Rules

Even though if a host is removed from a cluster, the host UUID is not removed from the host-affinity list for a VM.

### Configuring VM-VM Anti-Affinity Policy

To configure VM-VM anti-affinity policies, you must first define a group and then add all the VMs on which you want to define VM-VM anti-affinity policy.

### About this task

Note: Currently, the VM-VM affinity policy is not supported.

Perform the following procedure to configure the VM-VM anti-affinity policy.

### Procedure

1. Log on to the Controller VM with SSH session.
2. Create a group.

```
nutanix@cvm$ acli vm_group.create group_name
```

Replace `group_name` with the name of the group.

3. Add the VMs on which you want to define anti-affinity to the group.

```
nutanix@cvm$ acli vm_group.add_vms group_name vm_list=vm_name
```

Replace `group_name` with the name of the group. Replace `vm_name` with the name of the VMs that you want to define anti-affinity on. In case of multiple VMs, you can specify comma-separated list of VM names.

#### 4. Configure VM-VM anti-affinity policy.

```
nutanix@cvm$ acli vm_group.antiaffinity_set group_name
```

Replace `group_name` with the name of the group.

After you configure the group and then power on the VMs, the VMs that are part of the group are started (attempt to start) on the different hosts.

##### Important:

The VM-VM anti-affinity policy is a preferential policy. The system does not block any VM operation, such as VM maintenance mode or manual live migration of the VM, even if there is a policy violation. For example, when you manually migrate one VM of a VM-VM pair with an anti-affinity policy, the policy is applied on a best-effort basis only.

The Acropolis Dynamic Scheduling (ADS) always attempts to maintain compliance with the VM-VM anti-affinity policy and ensures that the VM-VM anti-affinity policy is enforced on a best-effort basis. For example, if you manually migrate a VM and the migration leads to non-compliance with the VM-VM anti-affinity policy, the ADS checks if the host is specified in manual migration, and performs the following actions:

- Ignores compliance to VM-VM anti-affinity policy, if a host is specified during manual migration.
- Attempts to enforce the policy back into compliance on a best-effort basis, if a host is not specified during manual migration.

For more information on ADS, see [Acropolis Dynamic Scheduling in AHV](#) on page 8.

#### Removing VM-VM Anti-Affinity Policy

Perform the following procedure to remove the VM-VM anti-affinity policy.

##### Procedure

1. Log on to the Controller VM with SSH session.
2. Remove the VM-VM anti-affinity policy.

```
nutanix@cvm$ acli vm_group.antiaffinity_unset group_name
```

Replace `group_name` with the name of the group.

The VM-VM anti-affinity policy is removed for the VMs that are present in the group, and they can start on any host during the next power on operation (as necessitated by the ADS feature).

## Affinity Policies Defined in Prism Central

In Prism Central, you can define the category-based VM-Host affinity policies, where a set of VMs can be affinity (correlated) to run only on a particular set of hosts. Category-based affinity policy enables you to easily manage affinities for a large number of VMs. In case of any changes to the affinity hosts, you only need to update the category of the host, and it updates the affinity policy for all the affected VMs.

This policy checks and hard enforces where a VM can be hosted when you start or migrate the VM. If there are no resources available on any of the affinity hosts, the VM is not started.

Note:



If you create a VM-Host affinity policy for a VM that is configured for asynchronous replication, you must create similar categories and corresponding policies on the remote site as well. If you define similar categories and policies on the remote site, the system applies the affinity policies when the VMs are migrated to the remote site.

For information about how the category-based VM-Host affinity policies are handled during disaster recovery, see [Affinity Policies Handling - PC Based DR Solution with Physically Isolated Locations](#).

### Limitations of Affinity Policies

Affinity policies created in Prism Central have the following limitations:

- The minimum supported versions for VM-Host affinity policies are version 6.1 for Prism Element and version 2022.1 for Prism Central.
- Host category attach or detach takes around five minutes to get reflected in the applicable affinity policies.

When you assign a category to a host and map the host category to the affinity policy, you can observe that the host count gets updated immediately in the [Entities](#) tab. However, the system takes approximately 5 minutes to update the host count in [Affinity Policies Summary View](#) on page 170.

The delay in host count update is due to the usage of different APIs to derive the host count in [Entities](#) tab and [Affinity Policies Summary View](#) on page 170.

For information about how to create a category, see [Creating a Category](#) information in *Prism Central Infrastructure Guide*.

For information about how to assign a category to host, see [Associating Hosts with Categories](#) on page 167

For information about how to create the affinity policy and map the host category to the affinity policy, see [Creating an Affinity Policy](#) on page 168.

### Affinity Policy Configuration Workflow

#### About this task

To set up an affinity policy, perform the following steps:

#### Procedure

1. Create categories for the following entities:

- a. VMs
- b. Hosts

For information about how to create a category, see [Creating a Category](#) information in *Prism Central Infrastructure Guide*.

2. Apply the VM categories to the VMs and host categories to the hosts.

For information about how to associate a category to the VMs, see [Associating VMs with Categories](#) information in *Prism Central Infrastructure Guide* . For information about associating categories with hosts, see [Associating hosts with Categories](#).

3. Create the affinity policy. For details, see [Creating an Affinity Policy](#) information in *Prism Central Infrastructure Guide*.



## Associating VMs with Categories

### About this task

To associate categories to VMs, perform the following steps:

#### Procedure

1. Log in to Prism Central.
2. Select the **Infrastructure** application from [Application Switcher Function](#), and navigate to **Compute & Storage > VMs** from the **Navigation Bar**. For information about the **Navigation Bar**, see [Application-specific Navigation Bar](#).  
The system displays the **List** tab by default.
3. Select the target VMs checkboxes that you want to associate with a category, and choose **Manage Categories** from the **Actions** dropdown menu.  
The system displays the **Manage VM Categories** window.
4. Type the name of the category in **Set Categories** field.  
The system displays the list of matching categories based on the typed entry.
5. Use the Add icon and Remove Icon explained in [Prism Central Landing Page](#) and to add and remove the required categories.

Note: If the VM category you selected is already part of any affinity policy, the system displays the associated affinity policy details.

6. Click **Save**.

## Associating Hosts with Categories

### About this task

To associate categories with hosts, perform the following steps:

#### Procedure

1. Log in to Prism Central.
2. Select the **Infrastructure** application from [Application Switcher Function](#), and navigate to **Hardware > Hosts** from the **Navigation Bar**. For information about the **Navigation Bar**, see [Application-specific Navigation Bar](#).  
The system displays the **List** tab by default.
3. Select the target host checkboxes that you want to associate with a category, and choose **Manage Categories** from the **Actions** dropdown menu.  
The system displays the **Manage Host Categories** window.
4. Type the name of the category in **Set Categories** field.  
The system displays the list of matching categories based on the typed entry.
5. Use the Add icon and Remove Icon explained in [Prism Central Landing Page](#) and to add and remove the required categories.
6. Click **Save**.



## Creating an Affinity Policy

### About this task

This section describes how to create an affinity policy in Prism Central.

### Before you begin

Ensure that the following prerequisites are met before you create an affinity policy:

- VM and Host categories are created. For information about how to create a category, see [Creating a Category](#) in *Prism Central Infrastructure Guide*.
- VMs are associated with VM categories, and hosts with host categories. For details, see [Associating VMs with Categories](#) on page 167 and [Associating Hosts with Categories](#) on page 167.

#### Note:

- The system also allows you to associate VMs with VM category and hosts with host category after creation of affinity policy.
- If you have configured any legacy affinity policy (non-category-based affinity policy) associated with the VMs, you must first remove those legacy affinity policies to allow the creation of category-based affinity policies associated with the same VMs.

### Procedure

To create an affinity policy, perform the following steps:

1. Log in to Prism Central.
2. Select the **Infrastructure** application from [Application Switcher Function](#), and navigate to **Compute & Storage > VMs** from the **Navigation Bar**. For information about the **Navigation Bar**, see [Application-specific Navigation Bar](#).  
The system displays the **List** tab by default.
3. Select **Affinity Policies** in the **Policies** tab dropdown menu, and click **Create Affinity Policy**.  
The system displays the **Create Affinity Policy** window.





4. Specify the following information in the **Create Affinity Policy** window:

- **Name:** Enter the affinity policy name.
- (optional) **Description:** Enter the description for the affinity policy.
- **VM Categories:**Type the name of the VM category. The system displays the list of matching categories based on the typed entry. Use the Add icon and Remove Icon explained in [Prism Central Landing Page](#) and to add and remove the required categories.
- **Host Categories:**Type the name of the host category. The system displays the list of matching categories based on the typed entry. Use the Add icon and Remove Icon explained in [Prism Central Landing Page](#) and to add and remove the required categories.

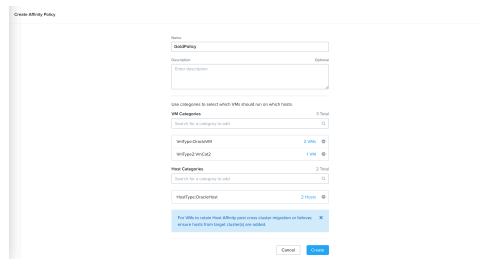


Figure 26: Create Affinity Policy

5. Click **Create**.

### Managing Affinity Policies

This section describes how to manage the existing affinity policies in Prism Central.

#### About this task

You can perform the following actions to manage the existing affinity policies in Prism Central:

- Update an affinity policy.
- Delete an affinity policy.
- Re-enforce an affinity policy.

#### Procedure

To manage the affinity policies, perform the following steps:

1. Log in to Prism Central.
2. Select the **Infrastructure** application from [Application Switcher Function](#), and navigate to **Compute & Storage > VMs** from the **Navigation Bar**. For information about the **Navigation Bar**, see [Application-specific Navigation Bar](#). The system displays the **List** tab by default.
3. Select **Affinity Policies** in the **Policies** tab dropdown menu. The system displays the affinity policies created across registered clusters.
4. Select the target affinity policy checkbox, and choose **Update** from the **Actions** dropdown menu.

- Update the fields in the **Update Affinity Policy** window as per your requirement, and click **Save**. For field details, see [Creating an Affinity Policy](#) on page 168.

Figure 27: Update Affinity Policy

To delete the affinity policy, select the target affinity policy checkbox and choose **Delete** from the **Actions** dropdown menu. The system prompts you to confirm the delete action. Click **Delete** to confirm the delete affinity policy action.

To re-enforce the affinity policy, select the target affinity policy checkbox and choose **Re-enforce** from the **Actions** dropdown menu. The system prompts you to confirm the action. Click **Re-enforce** to apply the updated affinity policy.

### Affinity Policies Summary View

The affinity policies summary view enables you to access a list of all the user-defined affinity policies across registered clusters.

To access the summary view of all affinity policies:

- Log in to Prism Central.
- Select the **Infrastructure** application from [Application Switcher Function](#), and navigate to **Compute & Storage > VMs** from the **Navigation Bar**. For information about the **Navigation Bar**, see [Application-specific Navigation Bar](#).

The system displays the **List** tab by default.

- Select **Affinity Policies** in the **Policies** tab dropdown menu.

The system displays a summary view of affinity policies across all registered clusters.

<input type="checkbox"/>	Name	VMs	Hosts	VM Compliance Status	Modified By	Last Modified
<input type="checkbox"/>	Affinity-Site-A	2	1	✓ 2 Compliant	admin	Feb 3, 2023, 12:17 PM

Figure 28: Summary View - All Affinity Policies

Table 20: Affinity Policies Page - Field Description

Parameter	Description	Values
Name	Displays the affinity policy name.	(name)
VMs	Displays the count of VMs associated with the selected affinity policy.	(number of VMs)
Hosts	Displays the count of hosts associated with the selected affinity policy.	(number of hosts)
VM Compliance Status	Displays the compliance status of the VMs associated with this policy. If the policy is being applied and the compliance status is not yet known, the status is displayed as Pending.  If a VM is part of multiple VM-Host affinity policies, the oldest policy is applied on the VM. For rest of the policies, the VM is displayed as non-compliant.	(number of VMs Compliant/Non Compliant/Pending)
Modified By	Displays the name of the user who modified the selected affinity policy last time.	(user)
Last Modified	Displays the date and time when the selected affinity policy is modified last time.	(date & time)

You can perform the following actions for the affinity policies in the **Affinity Policies** summary view:

- Access the detailed information about an individual affinity policy. For more information, see [Affinity Policies Details View](#) on page 171.
- Create an affinity policy. For more information, see [Creating an Affinity Policy](#) on page 168.
- Use the **Actions** dropdown menu to update, delete, or re-enforce an affinity policy. For more information, see [Managing Affinity Policies](#) on page 169.

### Affinity Policies Details View

The affinity policy details view allows you access the detailed information about an individual affinity policy. It includes four tabs: **Summary**, **Categories**, **Entities** and **Audit**.

To access the details view of an individual affinity policy, perform the following steps:

1. Log in to Prism Central.
2. Select the **Infrastructure** application from [Application Switcher Function](#), and navigate to **Compute & Storage > VMs** from the **Navigation Bar**. For information about the **Navigation Bar**, see [Application-specific Navigation Bar](#).

The system displays the **List** tab by default.

3. Select **Affinity Policies** in the **Policies** tab.

The system displays a summary view of affinity policies across all registered clusters.



- Click the target `<Affinity_Policy_Name>` to view the **Summary** tab of an individual affinity policy.

Note: Replace `<Affinity_Policy_Name>` with the actual affinity policy name at your site.

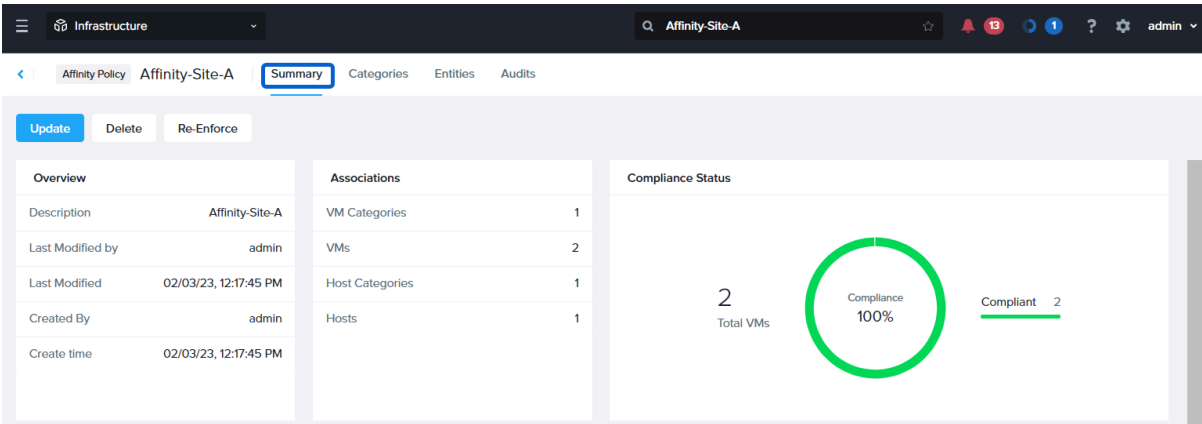


Figure 29: Affinity Policies Details view

The **Summary** tab of an individual affinity policy provides the following widgets:

- Overview** - Displays an overview information about the affinity policy.
- Associations** - Displays the associated VM and Host entities and their categories.
- Compliance Status** - Displays the compliance status of the VMs associated with this policy. If the policy is being applied and the compliance status is not yet known, the status is displayed as Pending. If a VM is part of multiple VM-Host affinity policies, the oldest policy is applied on the VM. For rest of the policies, the VM is displayed as non-compliant.

For information about the fields available in the widgets, see [Affinity Policy Widgets - Field Details](#) on page 172.

`<Action>` available above the widgets. Click the appropriate `<Action>` to run that administrative action on the affinity policy. For more information about how to perform any `<Action>`, see [Managing Affinity Policies](#) on page 169.

### Affinity Policy Widgets - Field Details

The following table describes the fields in the **Overview**, **Associations**, and **Compliance Status** widgets.

Table 21: Affinity Policy Widgets - Field Description

Field	Description	Values
<b>Overview</b> widget		
Description	Displays the affinity policy description specified while creating the affinity policy.	(description)
Last Modified By	Displays the name of the user who modified the affinity policy last time.	(user)

Field	Description	Values
Last Modified	Displays the date and time when the policy is modified last time.	(date & time)
Created By	Displays the name of the user who created the affinity policy	(user)
Create time	Displays the date and time when the affinity policy is created.	(date & time)
<b>Associations</b> widget		
VM Categories	Displays the count of VM categories mapped to the selected affinity policy.	(number of VM categories)
VMs	Displays the count of VMs associated with the selected affinity policy.	(number of VMs)
Host Categories	Displays the count of host categories mapped to the selected affinity policy.	(number of host categories)
Hosts	Displays the count of hosts associated with the selected affinity policy.	(number of hosts)
<b>Compliance Status</b> widget		
Total VMs	Displays the total number of VMs in the selected affinity policy.	(total number of VMs)
Compliance	Displays the compliance status in percentage .	Integer %
Compliant, In Progress, or Non compliant	Displays the number of VMs that are compliant and non-compliant with the affinity policy. If the compliance check is in progress, the system displays the compliance status as <i>In progress</i> .	Compliant, In Progress, or Non compliant

## Categories Tab

The **Categories** tab displays the VMs and hosts categories information.

Viewing 1 Associated VM Category			Export	1 - 1 of 1
<input type="checkbox"/> Name	VMs	Status		
<input type="checkbox"/> VM-Cat-Site-A: Site-A-VMs	2	✓ 2 Compliant		

Figure 30: Affinity Policies - Categories (VMs)

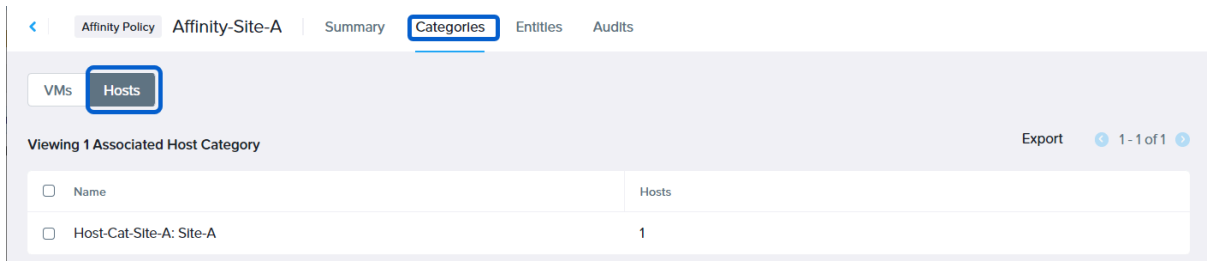


Figure 31: Affinity Policies - Categories (Hosts)

The following table describes the information displayed in the **Categories** tab

Table 22: Categories Tab -Field Description

Field	Description	Values
<b>VMs tab</b>		
Name	Displays the VM category name and its value.	(VM Category Name: Value)
VMs	Displays the count of VMs associated with the selected affinity policy.	(number of VMs)
Status	Displays the compliance status.	(number of VMs Compliant Non-compliant Pending)
<b>Host tab</b>		
Name	Displays the host category name and its value.	(host category Name: Value)
Hosts	Displays the count of hosts associated with the selected affinity policy.	(number of hosts)

You can export the table that contains the list of VM categories and host categories and their information to a file in a CSV format. For more information about **Export** option, see [Prism Central GUI Organization](#) in *Prism Central Infrastructure Guide*.

### Entities Tab

The **Entities** tab displays the VMs and hosts entity-related information.

Name	Host	Cluster	Associated via Categories	VM Compliance Status
TestJlt	Goten-4	auto_cluster_prod_f38293eb...	VM-Cat-Site-A: Site-A-VMs	✓ Compliant
Trial-ToDelete-0	Goten-4	auto_cluster_prod_f38293eb...	VM-Cat-Site-A: Site-A-VMs	✓ Compliant

Figure 32: Affinity Policies - Entities (VMs)

Host	Cluster	Associated via Categories
Goten-4	auto_cluster_prod_f38293eb9649	Host-Cat-Site-A: Site-A

Figure 33: Affinity Policies - Entities (Hosts)

The following table describes the information displayed in the **Entities** tab

Table 23: Entities Tab -Field Description

Field	Description	Values
<b>VMs tab</b>		
Name	Displays the VM name associated with the selected affinity policy.	(VM name)
Host	Displays the host name associated with the selected affinity policy.	(host name)
Cluster	Displays the name of the cluster (on which the host resides) associated with the selected affinity policy.	(cluster name)
Associated via Categories	Displays the VM category name and its value.	(VM category name: Value)
Compliance Status	Displays the VM compliance status with the selected affinity policy.	Compliant Non-compliant Pending
<b>Host tab</b>		
Host	Displays the name of the host associated with the selected affinity policy.	(host name)

Field	Description	Values
Cluster	Displays the name of the cluster (on which the host resides) associated with the selected affinity policy.	(cluster name)
Associated via Categories	Displays the host Category name and its value.	(host category name: Value)

You can export the table that contains the list of VM entities and host entities related information to a file in a CSV format. For more information about **Export** option, see [Prism Central GUI Organization](#) in *Prism Central Infrastructure Guide*.

## Audits Tab

The **Audits** tab displays the user action-related information for the affinity policies.

Field	Description	Values
Action Description	Displays the user-action for affinity policies.	(action description)
User Name	Displays the name of the user who performed the action.	(host name)
Operation Type	Displays the type of operation performed by the user for affinity policies. For example, Create	(cluster name)
Request Time	Displays the date and time of the user action.	(date and time)

Figure 34: Affinity Policies - Audits

The following table describes the information displayed in the **Audits** tab

Table 24: Audits Tab -Field Description

Field	Description	Values
Action Description	Displays the user-action for affinity policies.	(action description)
User Name	Displays the name of the user who performed the action.	(host name)
Operation Type	Displays the type of operation performed by the user for affinity policies. For example, Create	(cluster name)
Request Time	Displays the date and time of the user action.	(date and time)

You can export the table that contains the list of user actions related information to a file in a CSV format. For more information about **Export** option, see [Prism Central GUI Organization](#) in *Prism Central Infrastructure Guide*.

## Non-Migratable Hosts

VMs with GPU, CPU passthrough, PCI passthrough, and host affinity policies are not migrated to other hosts in the cluster. Such VMs are treated in a different manner in scenarios where VMs are required to migrate to other hosts in the cluster.



Table 25: Scenarios Where VMs Are Required to Migrate to Other Hosts

Scenario	Behavior
One-click upgrade	VM is powered off.
Life-cycle management (LCM)	Pre-check for LCM fails and the VMs are not migrated.
Rolling restart	VM is powered off.
AHV host maintenance mode	Use the tunable option to shut down the VMs while putting the node in maintenance mode. For more information, see <a href="#">Putting a Node into Maintenance Mode using CLI</a> on page 31.

## Affinity Policies Specifications

This section defines the specifications that apply to affinity policies defined in Prism Central and VM-Host affinity policies defined in Prism Element. These specifications cover the Disaster Recovery and RBAC-related aspects of affinity policies.

### Disaster Recovery Specifications for Affinity Policies

This section describes how affinity policies are handled in the Disaster Recovery scenario.

#### Affinity Policies Handling: PC-based DR Solution

This section provides the information about how to handle the affinity policies at the recovery site in a Prism Central-based DR setup.

#### Failover Phase

After the failover is completed, the VM categories created on the primary site (primary AZ) are synchronized to the recovery site (recovery AZ). The synchronized VM categories are also reflected in the Prism Central instance of the recovery site. The system applies any host affinity policy that exists on the recovery site with synchronized VM categories.

To set up the affinity policies at the recovery site, perform the following actions before a disaster occurs:

1. Create the host categories at the recovery site. For more information, see [Creating a Category](#) in *Prism Central Infrastructure Guide*.
2. Create the VM categories at the recovery site with the same name as defined at the primary site. For more information, see [Creating a Category](#) in *Prism Central Infrastructure Guide*.
3. Reconfigure the VM affinity policies at the recovery site using the VM categories created in Step 2 and new host categories created in Step 1. For more information, see [Creating an Affinity Policy](#) on page 168.

Important: If VM-Host affinity policies are defined using Prism element, the VMs never get the affinity after failover, and you need to manually redefine the affinity policies at the recovery site after failover.

#### Failback Phase

After you perform failback, the VM categories are synchronized back with the primary site, and the affinity policies created on the primary site (primary AZ) are enabled for the VMs.



## Synchronous Replication Case

You can also configure the host affinity at the recovery site for the VMs/VM categories that are protected with synchronous replication schedule in the primary site using the following workflow:

1. Create the host category at the recovery site. For more information, see [Creating a Category](#) in *Prism Central Infrastructure Guide*.
2. Create the VM categories at the recovery site with the same name as defined at the primary site. For more information, see [Creating a Category](#) in *Prism Central Infrastructure Guide*.
3. Create the VM affinity policies at the recovery site using the VM categories created in Step 2 and new host categories created in Step 1. For more information, see [Creating an Affinity Policy](#) on page 168.

The following table provides information about host-affinity support for the VMs/VM categories that are protected with synchronous replication:

Table 26: VM-Host Affinity with Synchronous Replication RPO

AOS Release deployed at Primary Site	AOS Release deployed at Recovery Site	Is host affinity with Synchronous replication supported?	Workaround
6.7 or later	6.7 or later	Supported	None
6.6	6.7 or later	Not supported	Upgrade primary site to AOS 6.7 or above release and then configure host affinity with synchronous replication at the primary site.
6.7 or later	6.6	Not supported	Upgrade recovery site to AOS 6.7 or above release and then configure host affinity with synchronous replication at the primary site.

For more information about how to configure the protection policy with a synchronous replication schedule, see [Creating a Protection Policy](#) in *Nutanix Disaster Recovery Guide*.

### Affinity Policies Handling: PD-based DR Solution

At the recovery site (recovery cluster), when you recover a guest VM protected with asynchronous or nearsync replication schedule using local snapshots, the affinity policies also get applied after you restore and never go stale.

## Failover and Failback Phase

When you perform the failover or failback action, the system clears all the VM-Host affinity policies defined using Prism element. The guest VMs can start using any host at the current (migrated) site.

Important:



- If all clusters are registered to one Prism Central instance and the VM-Host affinity policies are defined using Prism Central, the system periodically reconciles the affinity policies and enables you to have consistency in the affinity policies defined at the local site.
- If clusters are registered to multiple Prism Central instances and the VM-Host affinity policies are defined using Prism Central, the system clears the affinity policies, and the guest VMs start using any host at the local site.

## Role Based Access Control for Affinity Policies

This section describes the Role Based Access Control (RBAC) requirements and limitations for affinity policies. For information about RBAC, see [Controlling User Access \(RBAC\)](#) in *Security Guide*.

### Role Based Access Control Requirements for Affinity Policies

Before you allocate the [Affinity Policy-Related Permissions](#) on page 179, ensure that the following basic permissions are allocated to the admin user or a custom user to access the affinity policies framework in Prism Central:

- View VM
- View category

### Affinity Policy-Related Permissions

In addition to the basic permissions to view VM and categories as specified in [Role Based Access Control Requirements for Affinity Policies](#) on page 179, you can also assign the following permissions to the admin or custom user to manage the affinity policies in Prism Central:

Table 27: Affinity Policy-Related Permissions

Permission	Description
Create_VM_Host_Affinity_Policy	Admin or custom user can create Affinity Policies.
View_VM_Host_Affinity_Policy	Admin or custom user can view affinity policies.
Update_VM_Host_Affinity_Policy	Admin or custom user can update the existing affinity policies.
Reenforce_VM_Host_Affinity_Policy	Admin or custom user can re-enforce the existing affinity policies.
Delete_VM_Host_Affinity_Policy	Admin or custom user can delete affinity policies.

These permissions also facilitate the admin or custom user to manage the legacy affinity policies (affinity policies that are defined in the Prism Element) from Prism Central.

### Role-Based Access Control Limitations for Affinity Policies

The following table describes the limitations that apply to RBAC for affinity policies:



Table 28: RBAC - Limitations for Affinity Policies

Permission Assigned to the Admin or Custom user	Admin or Custom User Action	System Behavior	Limitation
Create_VM_Host_Affinity_Policy Update_VM_Host_Affinity_Policy	Admin or custom user creates or updates the affinity policy.	The system applies the affinity policy to all the VMs associated with the mapped VM categories.	The system does not check whether the admin or the custom user has permission to access the VMs or hosts mapped in the affinity policy.
Update VM Categories	Admin or custom user updates a VM category (for example, attaches a VM to a VM category).	The system applies the affinity policy to which the VM category is assigned.	The system does not check whether the admin or the custom user has permission to create or update the affinity policy.
Update Host Categories	Admin or custom user updates a Host category (for example, attaches a host to a Host category).	The system applies the affinity policy to which the host category is assigned.	The system does not check whether the admin or the custom user has permission to create or update the affinity policy.

## Performing Power Operations on VMs by Using Nutanix Guest Tools (aCLI)

You can initiate safe and graceful power operations such as soft shutdown and restart of the VMs running on the AHV hosts by using the aCLI. Nutanix Guest Tools (NGT) initiates and performs the soft shutdown and restart operations within the VM. This workflow ensures a safe and graceful shutdown or restart of the VM. You can create a pre-shutdown script that you can choose to run before a shutdown or restart of the VM. In the pre-shutdown script, include any tasks or checks that you want to run before a VM is shut down or restarted. You can choose to cancel the power operation if the pre-shutdown script fails. If the script fails, an alert (guest\_agent\_alert) is generated in the Prism web console.

### Before you begin

Ensure that you have met the following prerequisites before you initiate the power operations:

1. NGT is enabled on the VM. All operating systems that NGT supports are supported for this feature.
2. NGT version running on the Controller VM and guest VM is the same.
3. (Optional) If you want to run a pre-shutdown script, place the script in the following locations depending on your VMs:

- Windows VMs: installed\_dir\scripts\power\_off.bat  
The file name of the script must be **power\_off.bat**.
- Linux VMs: installed\_dir/scripts/power\_off  
The file name of the script must be **power\_off**.



## About this task

Note: You can also perform these power operations by using the V3 API calls. For more information, see [developer.nutanix.com](https://developer.nutanix.com).

Perform the following steps to initiate the power operations:

### Procedure

1. Log on to a Controller VM with SSH.
2. Do one of the following:

» Soft shut down the VM.

```
nutanix@cvm$ acli vm.guest_shutdown vm_name enable_script_exec=[true or false]  
fail_on_script_failure=[true or false]
```

Replace `vm_name` with the name of the VM.

» Restart the VM.

```
nutanix@cvm$ acli vm.guest_reboot vm_name enable_script_exec=[true or false]  
fail_on_script_failure=[true or false]
```

Replace `vm_name` with the name of the VM.

Set the value of `enable_script_exec` to **true** to run your pre-shutdown script and set the value of `fail_on_script_failure` to **true** to cancel the power operation if the pre-shutdown script fails.

## UEFI Support for VM

UEFI firmware is a successor to legacy BIOS firmware that supports larger hard drives, faster boot time, and provides more security features.

VMs with UEFI firmware have the following advantages:

- Boot faster
- Avoid legacy option ROM address constraints
- Include robust reliability and fault management
- Use UEFI drivers

Note:

- Nutanix supports the starting of VMs with UEFI firmware in an AHV cluster. However, if a VM is added to a protection domain and later restored on a different cluster, the VM loses boot configuration. To restore the lost boot configuration, see [Setting up Boot Device](#) on page 184.
- Nutanix also provides limited support for VMs migrated from a Hyper-V cluster.

You can create or update VMs with UEFI firmware by using `accli` commands, Prism Element web console, or Prism Central.

- For more information about how to create a VM using the Prism Element web console, see [Creating a VM \(AHV\)](#) on page 113.



- For more information about how to create a VM using Prism Central, see [Creating a VM through Prism Central \(AHV\)](#) information in *Prism Central Infrastructure Guide*.
- For information about how to create a VM by using aCLI, see [Creating UEFI VMs by Using aCLI](#) on page 182.

Note: If you create a VM using aCLI commands, you can define the location of the storage container for UEFI firmware and variables. Prism Element web console or Prism Central does not provide the option to define the storage container to store UEFI firmware and variables.

For more information about the supported OSes for the guest VMs, see the *AHV Guest OS* section in the [Compatibility and Interoperability Matrix](#) document.

## Creating UEFI VMs by Using aCLI

In AHV clusters, you can create a virtual machine (VM) to start with UEFI firmware by using Acropolis CLI (aCLI). This topic describes the procedure to create a VM by using aCLI. See the "Creating a VM (AHV)" topic for information about how to create a VM by using the Prism Element web console.

### Before you begin

Ensure that the VM has an empty vDisk.

### About this task

Perform the following procedure to create a UEFI VM by using aCLI:

### Procedure

1. Log on to any Controller VM in the cluster with SSH.
2. Create a UEFI VM.

```
nutanix@cvm$ acli vm.create vm-name uefi_boot=true
```

A VM is created with UEFI firmware. Replace **vm-name** with a name of your choice for the VM. By default, the UEFI firmware and variables are stored in an NVRAM container. If you would like to specify a location of the NVRAM storage container to store the UEFI firmware and variables, do so by running the following command.

```
nutanix@cvm$ acli vm.create vm-name uefi_boot=true nvram_container=NutanixManagementShare
```

Replace **NutanixManagementShare** with a storage container in which you want to store the UEFI variables.

The UEFI variables are stored in a default NVRAM container. Nutanix recommends you to choose a storage container with at least RF2 storage policy to ensure the VM high availability for node failure scenarios. For more information about RF2 storage policy, see [Failure and Recovery Scenarios](#) in the *Prism Element Web Console Guide* document.

Note: When you update the location of the storage container, clear the UEFI configuration and update the location of **nvram\_container** to a container of your choice.

### What to do next

Go to the UEFI BIOS menu and configure the UEFI firmware settings. For more information about accessing and setting the UEFI firmware, see [Getting Familiar with UEFI Firmware Menu](#) on page 183.



## Getting Familiar with UEFI Firmware Menu

After you launch a VM console from the Prism Element web console, the UEFI firmware menu allows you to do the following tasks for the VM.

- Changing default boot resolution
- Setting up boot device
- Changing boot-time value

### Changing Boot Resolution

You can change the default boot resolution of your Windows VM from the UEFI firmware menu.

### Before you begin

Ensure that the VM is in powered on state.

### About this task

Perform the following procedure to change the default boot resolution of your Windows VM by using the UEFI firmware menu.

### Procedure

1. Log on to the Prism Element web console.
2. Launch the console for the VM.  
For more details about launching console for the VM, see [Managing a VM \(AHV\)](#) section in *Prism Element Web Console Guide*.
3. To go to the UEFI firmware menu, press the **F2** keys on your keyboard.

Tip: To enter UEFI menu, open the VM console, select **Reset** in the **Power off/Reset VM** dialog box, and immediately press **F2** when the VM starts to boot.

Important: Resetting the VM results in VM downtime. We suggest that you reset the VM only during off-production hours or during a maintenance period.

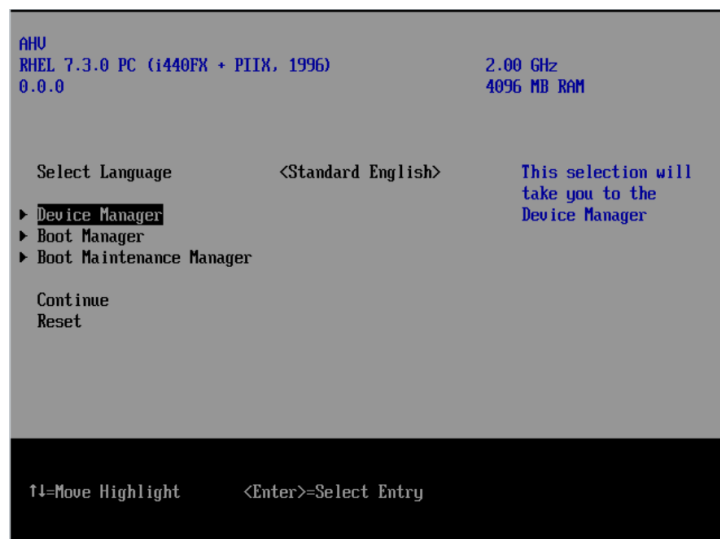


Figure 35: UEFI Firmware Menu

4. Use the up or down arrow key to go to **Device Manager** and press **Enter**.  
The **Device Manager** page appears.
5. In the **Device Manager** screen, use the up or down arrow key to go to **OVMF Platform Configuration** and press **Enter**.



Figure 36: OVMF Settings

The **OVMF Settings** page appears.

6. In the **OVMF Settings** page, use the up or down arrow key to go to the **Change Preferred** field and use the right or left arrow key to increase or decrease the boot resolution.  
The default boot resolution is 1280X1024.
7. Do one of the following.
  - » To save the changed resolution, press the **F10** key.
  - » To go back to the previous screen, press the **Esc** key.
8. Select **Reset** and click **Submit** in the Power off/Reset dialog box to restart the VM.  
After you restart the VM, the OS displays the changed resolution.

### Setting up Boot Device

This section describes how to set up the boot device for a UEFI VM.

#### About this task

You cannot set the boot order for UEFI VMs by using the aCLI, Prism Central web console, or Prism Element web console. You can change the boot device for a UEFI VM by using the UEFI firmware menu only.

#### Before you begin

Ensure that the following prerequisites are met before you set up or change the boot order for the VM:

- VM is in powered on state.



- The system behavior associated with following VM conditions is noted:

Table 29: System Behavior based on VM condition

Condition	System Behavior
VM is installed with UEFI and the EFI boot partition exists.	Any change made to the boot order persists and the changes are saved in the nvVars file in the EFI partition.
No guest OS is installed on the VM but the EFI boot partition exists.	
VM with no EFI boot partition.	Any change made to the boot order persists only while the VM is on (or rebooted), but a power off/on action reverts the boot order change to the previous setting.
Secure boot-enabled VM (with or without EFI partition present)	

## Procedure

To set up the boot device for a UEFI VM, perform the following steps:

1. Log on to the Prism Element web console.
2. Launch the console for the VM.  
For more details about launching console for the VM, see [Managing a VM \(AHV\)](#) section in *Prism Element Web Console Guide*.
3. To go to the UEFI firmware menu, press the **F2** keys on your keyboard.

Tip: To enter UEFI menu, open the VM console, select **Reset** in the **Power off/Reset VM** dialog box, and immediately press **F2** when the VM starts to boot.

Important: Resetting the VM results in VM downtime. We suggest that you reset the VM only during off-production hours or during a maintenance period.

4. Use the up or down arrow key to go to **Boot Manager** and press **Enter**.  
The **Boot Manager** screen displays the list of available boot devices in the cluster based on the VM disks configuration.
5. In the **Boot Manager** screen, use the up or down arrow key to select the boot device and press **Enter**.  
The boot device is saved. After you select and save the boot device, the VM boots up with the new boot device.
6. To go back to the previous screen, press **Esc**.

## Changing Boot Time-Out Value

The boot time-out value determines how long the boot menu is displayed (in seconds) before the default boot entry is loaded to the VM. This topic describes the procedure to change the default boot-time value of 0 seconds.



## About this task

Ensure that the VM is in powered on state.

## Procedure

1. Log on to the Prism Element web console.

2. Launch the console for the VM.

For more details about launching console for the VM, see [Managing a VM \(AHV\)](#) section in *Prism Element Web Console Guide*.

3. To go to the UEFI firmware menu, press the **F2** keys on your keyboard.

Tip: To enter UEFI menu, open the VM console, select **Reset** in the **Power off/Reset VM** dialog box, and immediately press **F2** when the VM starts to boot.

Important: Resetting the VM results in VM downtime. We suggest that you reset the VM only during off-production hours or during a maintenance period.

4. Use the up or down arrow key to go to **Boot Maintenance Manager** and press **Enter**.



Figure 37: Boot Maintenance Manager

5. In the **Boot Maintenance Manager** screen, use the up or down arrow key to go to the **Auto Boot Time-out** field.

The default boot-time value is 0 seconds.

6. In the **Auto Boot Time-out** field, enter the boot-time value and press **Enter**.

Note: The valid boot-time value ranges from 1 second to 9 seconds.

The boot-time value is changed. The VM starts after the defined boot-time value.

7. To go back to the previous screen, press **Esc**.

## Secure Boot Support for VMs

The pre-operating system environment is vulnerable to attacks by possible malicious loaders. Secure boot addresses this vulnerability with UEFI secure boot using policies present in the firmware along with certificates, to ensure that only properly signed and authenticated components are allowed to execute.

### Supported Operating Systems

For more information about the supported OSes for the guest VMs, see the *AHV Guest OS* section in the [Compatibility and Interoperability Matrix](#) document.

## Secure Boot Considerations

This section provides the limitations and requirements to use Secure Boot.

### Limitations

Secure Boot for guest VMs has the following limitations:

- Nutanix does not support converting a VM that uses IDE disks or legacy BIOS to VMs that use Secure Boot.
- The minimum supported version of the Nutanix VirtIO package for Secure boot-enabled VMs is 1.1.6.
- Secure boot VMs do not permit CPU, memory, or PCI disk hot plug.

### Requirements

Secure Boot is supported only on the Q35 machine type.

## Creating/Updating a VM with Secure Boot Enabled

You can enable Secure Boot with UEFI firmware, either while creating a VM or while updating a VM by using aCLI commands or Prism Element web console.

See [Creating a VM \(AHV\)](#) on page 113 for instructions about how to enable Secure Boot by using the Prism Element web console.

### Creating a VM with Secure Boot Enabled

#### About this task

To create a VM with Secure Boot enabled:

#### Procedure

1. Log on to any Controller VM in the cluster with SSH.
2. To create a VM with Secure Boot enabled:

```
nutanix@cvm$ acli vm.create <vm_name> secure_boot=true machine_type=q35
```

Note: Specifying the machine type is required to enable the secure boot feature. UEFI is enabled by default when the Secure Boot feature is enabled.

### Updating a VM to Enable Secure Boot

#### About this task

To update a VM to enable Secure Boot:



## Procedure

1. Log on to any Controller VM in the cluster with SSH.
2. To update a VM to enable Secure Boot, ensure that the VM is powered off.

```
nutanix@cvm$ acli vm.update <vm_name> secure_boot=true machine_type=q35
```

### Note:

- If you disable the secure boot flag alone, the machine type remains q35, unless you disable that flag explicitly.
- UEFI is enabled by default when the Secure Boot feature is enabled. Disabling Secure Boot does not revert the UEFI flags.

## Securing AHV VMs with Virtual Trusted Platform Module (aCLI)

### Overview

A Trusted Platform Module (TPM) is used to manage cryptographic keys for security services like encryption and hardware (and software) integrity protection. AHV vTPM is software-based emulation of the TPM 2.0 specification that works as a virtual device.

You can use the AHV vTPM feature to secure virtual machines running on AHV.

### Note:

- You can enable vTPM using aCLI or Prism Central. To enable vTPM using Prism Central, see [Securing AHV VMs with Virtual Trusted Platform Module](#) in the *Security Guide*.
- AHV vTPM does NOT require OR use a hardware TPM.

### vTPM Use Cases

AHV vTPM provides virtualization-based security support for the following primary use cases.

- Support for storing cryptographic keys and certificates for Microsoft Windows BitLocker
- TPM protection for storing VBS encryption keys for Windows Defender Credential Guard

See *Microsoft Documentation* for details on Microsoft Windows Defender Credential Guard and Microsoft Windows BitLocker.

Tip: Windows 11 installation requires both TPM 2.0 and secure boot enabled for the guest VM. For more information, see Microsoft website for Windows 11 specs, features, and computer requirements.

For information on how to create or update a guest VM with secure boot enabled, see [Creating/Updating a VM with Secure Boot Enabled](#) on page 187.

### Considerations for Enabling vTPM in AHV VMs (aCLI)

### Requirements

Supported Software Versions:

- AHV version 20220304.242 or above



- AOS version 6.5.1 or above

VM Requirements:

- You must enable UEFI on the VM on which you want to enable vTPM, see [UEFI Support for VM](#).
- You must enable Secure Boot (applicable if using Microsoft Windows BitLocker). To enable Secure Boot, see [Creating a VM with Secure Boot Enabled](#) on page 187.

## Limitations

- All [Secure Boot limitations](#) apply to vTPM VM.
- [Disaster Recovery limitations](#) apply when protecting vTPM-enabled VMs.

Creating AHV VMs with vTPM (aCLI)

## About this task

You can create a virtual machine with the vTPM configuration enabled using the following aCLI procedure.

## Procedure

1. Log on to any Controller VM in the cluster with SSH.
2. At the CVM prompt, type `accli` to enter the Acropolis CLI mode.
3. Create a VM with the [required configuration](#) using one of the following methods.
  - » Create a VM using Prism Element or Prism Central web console. If you choose to create the VM using Prism Element or Prism Central, proceed to *Step 4*.

Note: For simplicity, it is recommended to use Prism Element or Prism Central web console to create VMs, see [Creating a VM](#).

- » Create a VM using aCLI. You can enable vTPM at the time of creating a VM. To enable vTPM during VM creation, do the following and proceed to *step 5* (skip *step 4*).

Use the "vm.create" command with required arguments to create a VM. For details on VM creation command ("vm.create") and supported arguments using aCLI, see "vm" in the [Command Reference Guide](#).

```
accli> vm.create <vm-name> machine_type=q35 uefi_boot=true secure_boot=true
virtual_tpm=true <argument(s)>
```

Replace <vm-name> with the name of the VM and <argument(s)> with one or more arguments as needed for your VM.

4. Enable vTPM.

```
accli> vm.update <vm-name> virtual_tpm=true
```

In the above command, replace "<vm-name>" with name of the newly created VM.

5. Start the VM.

```
accli> vm.on <vm-name>
```

Replace <vm-name> with the name of the VM.  
vTPM is enabled on the VM.



## Enabling vTPM for Existing AHV VMs (aCLI)

### About this task

You can update the settings of an existing virtual machine (that satisfies [vTPM requirements](#)) to enable vTPM using the following aCLI procedure.

### Procedure

1. Log on to any Controller VM in the cluster with SSH.
2. At the CVM prompt, type `accli` to enter the acropolis CLI mode.
3. Shut down the VM to enforce an update on the VM.

```
accli> vm.shutdown <vm-name>
```

Replace <vm-name> with the name of the VM.

4. Enable vTPM.

```
accli> vm.update <vm-name> virtual_tpm=true
```

Replace <vm-name> with the name of the VM.

5. Start the VM.

```
accli> vm.on <vm-name>
```

Replace <vm-name> with the name of the VM.

vTPM is enabled on the VM.

## Virtual Machine Network Management

Virtual machine network management involves configuring connectivity for guest VMs through virtual switches, VLANs, and VPCs.

For information about how to create or update a virtual switch and other VM network options, see [Network & Security Management](#) information in *Prism Central Infrastructure Guide*. Virtual switch creation and updates are also covered in [Network Management](#) in *Prism Web Console Guide*.

## Virtual Machine Memory and CPU Hot-Plug Configurations

Memory and CPUs are hot-pluggable on guest VMs running on AHV. You can increase the memory allocation and the number of CPUs on your VMs while the VMs are powered on. You can change the number of vCPUs (sockets) while the VMs are powered on. However, you cannot change the number of cores per socket while the VMs are powered on.

Note: You cannot decrease the memory allocation and the number of CPUs on your VMs while the VMs are powered on.

You can change the memory and CPU configuration of your VMs by using the Acropolis CLI (aCLI) (see [Managing a VM \(AHV\)](#) in the Prism Element Web Console Guide or see [Managing a VM \(AHV\)](#) and [Managing a VM \(Self Service\)](#) in the Prism Central Infrastructure Guide).

See the [AHV Guest OS Compatibility Matrix](#) for information about operating systems on which you can hot plug memory and CPUs.



## Memory OS Limitations

1. On Linux operating systems, the Linux kernel might not make the hot-plugged memory online. If the memory is not online, you cannot use the new memory. Perform the following procedure to make the memory online.

1. Identify the memory block that is offline.

Display the status of all of the memory.

```
$ cat /sys/devices/system/memory/memoryXXX/state
```

Display the state of a specific memory block.

```
$ grep line /sys/devices/system/memory/*/state
```

2. Make the memory online.

```
$ echo online > /sys/devices/system/memory/memoryXXX/state
```

2. If your VM has CentOS 7.2 as the guest OS and less than 3 GB memory, hot plugging more memory to that VM so that the final memory is greater than 3 GB, results in a memory-overflow condition. To resolve the issue, restart the guest OS (CentOS 7.2) with the following setting:

```
swiotlb=force
```

## CPU OS Limitation

On CentOS operating systems, if the hot-plugged CPUs are not displayed in `/proc/cpuinfo`, you might have to bring the CPUs online. For each hot-plugged CPU, run the following command to bring the CPU online.

```
$ echo 1 > /sys/devices/system/cpu/cpu<n>/online
```

Replace `<n>` with the number of the hot plugged CPU.

## Hot-Plugging the Memory and CPUs on Virtual Machines (AHV)

### About this task

Perform the following procedure to hot plug the memory and CPUs on the AHV VMs.

### Procedure

1. Log on the Controller VM with SSH.
2. Update the memory allocation for the VM.

```
nutanix@cvm$ acli vm.update vm-name memory=new_memory_size
```

Replace `vm-name` with the name of the VM and `new_memory_size` with the memory size.

3. Update the number of CPUs on the VM.

```
nutanix@cvm$ acli vm.update vm-name num_vcpus=n
```

Replace `vm-name` with the name of the VM and `n` with the number of CPUs.

## Virtual Machine Memory Management (vNUMA)

AHV hosts support Virtual Non-uniform Memory Access (vNUMA) on virtual machines. You can enable vNUMA on VMs when you create or modify the VMs to optimize memory performance.



## Non-uniform Memory Access (NUMA)

In a NUMA topology, the memory access times of a VM depend on the memory location relative to a processor. A VM accesses memory local to a processor faster than the non-local memory. If the VM uses both CPU and memory from the same physical NUMA node, you can achieve optimal resource utilization. If you are running the CPU on one NUMA node (for example, node 0) and the VM accesses the memory from another node (node 1) then memory latency occurs. Ensure that the virtual topology of VMs matches the physical hardware topology to achieve minimum memory latency.

## Virtual Non-uniform Memory Access (vNUMA)

vNUMA optimizes the memory performance of virtual machines that require more vCPUs or memory than the capacity of a single physical NUMA node. In a vNUMA topology, you can create multiple vNUMA nodes where each vNUMA node includes vCPUs and virtual RAM. When you assign a vNUMA node to a physical NUMA node, the vCPUs can intelligently determine the memory latency (high or low). Low memory latency within a vNUMA node results in low latency in the physical NUMA node as well.

## Enabling and Disabling vNUMA on Virtual Machines

### Before you begin

Before you enable vNUMA, see [AHV Best Practices Guide](#) under *Solutions Documentation*.

### About this task

Perform the following procedure to enable vNUMA on your VMs running on the AHV hosts.

### Procedure

1. Log on to a Controller VM with SSH.
2. Check how many NUMA nodes are available on each AHV host in the cluster.

```
nutanix@cvm$ hostssh "numactl --hardware"
```

The console displays an output similar to the following:

```
===== 10.x.x.x =====
available: 2 nodes (0-1)
node 0 cpus: 0 1 2 3 4 5 6 7 16 17 18 19 20 21 22 23
node 0 size: 128837 MB
node 0 free: 862 MB
node 1 cpus: 8 9 10 11 12 13 14 15 24 25 26 27 28 29 30 31
node 1 size: 129021 MB
node 1 free: 352 MB
node distances:
node  0  1
   0: 10 21
   1: 21 10
===== 10.x.x.x =====
available: 2 nodes (0-1)
node 0 cpus: 0 1 2 3 4 5 12 13 14 15 16 17
node 0 size: 128859 MB
node 0 free: 1076 MB
node 1 cpus: 6 7 8 9 10 11 18 19 20 21 22 23
node 1 size: 129000 MB
node 1 free: 436 MB
node distances:
node  0  1
   0: 10 21
```





```

1: 21 10
===== 10.x.x.x =====
available: 2 nodes (0-1)
node 0 cpus: 0 1 2 3 4 5 12 13 14 15 16 17
node 0 size: 128859 MB
node 0 free: 701 MB
node 1 cpus: 6 7 8 9 10 11 18 19 20 21 22 23
node 1 size: 129000 MB
node 1 free: 357 MB
node distances:
node 0 1
0: 10 21
1: 21 10
===== 10.x.x.x =====
available: 2 nodes (0-1)
node 0 cpus: 0 1 2 3 4 5 12 13 14 15 16 17
node 0 size: 128838 MB
node 0 free: 1274 MB
node 1 cpus: 6 7 8 9 10 11 18 19 20 21 22 23
node 1 size: 129021 MB
node 1 free: 424 MB
node distances:
node 0 1
0: 10 21
1: 21 10
===== 10.x.x.x =====
available: 2 nodes (0-1)
node 0 cpus: 0 1 2 3 4 5 12 13 14 15 16 17
node 0 size: 128837 MB
node 0 free: 577 MB
node 1 cpus: 6 7 8 9 10 11 18 19 20 21 22 23
node 1 size: 129021 MB
node 1 free: 612 MB
node distances:
node 0 1
0: 10 21
1: 21 10

```

The example output shows that each AHV host has two NUMA nodes.

### 3. Do one of the following:

- » Enable vNUMA if you are creating a VM.

```

nutanix@cvm$ acli vm.create <vm_name> num_vcpus=x \
num_cores_per_vcpu=x memory=xG \
num_vnuma_nodes=x

```

- » Enable vNUMA if you are modifying an existing VM.

```

nutanix@cvm$ acli vm.update <vm_name> \
num_vnuma_nodes=x

```

- » Disable vNUMA if you are modifying an existing VM.

```

nutanix@cvm$ acli vm.update <vm_name> num_vnuma_nodes=0

```

Replace `<vm_name>` with the name of the VM on which you want to enable vNUMA or vUMA.  
Replace `x` with the values for the following indicated parameters:

- `num_vcpus`: Type the number of vCPUs for the VM.
- `num_cores_per_vcpu`: Type the number of cores per vCPU.



- memory: Type the memory in GB for the VM.
- num\_vnuma\_nodes: Type the number of vNUMA nodes for the VM.

For example:

```
nutanix@cvm$ acli vm.create test_vm num_vcpus=20 memory=150G num_vnuma_nodes=2
```

This command creates a VM with 2 vNUMA nodes, 10 vCPUs and 75 GB memory for each vNUMA node.

## GPU and vGPU Support

AHV supports GPU-accelerated computing for guest VMs. You can configure either GPU pass-through or a virtual GPU.

Note: You can configure either pass-through or a vGPU for a guest VM but not both.

This guide describes the concepts related to the GPU and vGPU support in AHV. For the configuration procedures, see the [Prism Element Web Console Guide](#).

For driver installation instructions, see the [NVIDIA Grid Host Driver for Nutanix AHV Installation Guide](#).

Note: VMs with GPU are not migrated to other hosts in the cluster. For more information, see [Non-Migratable Hosts](#) on page 176.

## Supported GPUs

For information about supported GPUs based on AOS release and AHV version, see [NVIDIA Drivers](#) on the *Compatibility and Interoperability* page.

## GPU Pass-Through for Guest VMs

AHV hosts support GPU pass-through for guest VMs, allowing applications on VMs direct access to GPU resources. The Nutanix user interfaces provide a cluster-wide view of GPUs, allowing you to allocate any available GPU to a VM. You can also allocate multiple GPUs to a VM. However, in a pass-through configuration, only one VM can use a GPU at any given time.

Note: AHV does not support multiple GPU pass-through to a single VM. If you host any application that requires multiple GPUs to pass through to a single VM, it might fail.

### Host Selection Criteria for VMs with GPU Pass-Through

When you power on a VM with GPU pass-through, the VM is started on the host that has the specified GPU, provided that the Acropolis Dynamic Scheduler determines that the host has sufficient resources to run the VM. If the specified GPU is available on more than one host, the Acropolis Dynamic Scheduler ensures that a host with sufficient resources is selected. If sufficient resources are not available on any host with the specified GPU, the VM is not powered on.

If you allocate multiple GPUs to a VM, the VM is started on a host if, in addition to satisfying Acropolis Dynamic Scheduler requirements, the host has all of the GPUs that are specified for the VM.

If you want a VM to always use a GPU on a specific host, configure host affinity for the VM.



## Support for Graphics and Compute Modes

AHV supports running GPU cards in either graphics mode or compute mode. If a GPU is running in compute mode, Nutanix user interfaces indicate the mode by appending the string `compute` to the model name. No string is appended if a GPU is running in the default graphics mode.

## Switching Between Graphics and Compute Modes

If you want to change the mode of the firmware on a GPU, put the host in maintenance mode, and then flash the GPU manually by logging on to the AHV host and performing standard procedures as documented for Linux VMs by the vendor of the GPU card.

Typically, you restart the host immediately after you flash the GPU. After restarting the host, redo the GPU configuration on the affected VM, and then start the VM. For example, consider that you want to re-flash an NVIDIA Tesla® M60 GPU that is running in graphics mode. The Prism web console identifies the card as an `NVIDIA Tesla M60` GPU. After you re-flash the GPU to run in compute mode and restart the host, redo the GPU configuration on the affected VMs by adding back the GPU, which is now identified as an `NVIDIA Tesla M60.compute` GPU, and then start the VM.

## Supported GPU Cards

For a list of supported GPUs, see [Supported GPUs](#) on page 194.

## Limitations

GPU pass-through support has the following limitations:

- Live migration of VMs with a GPU configuration is not supported. Live migration of VMs is necessary when the BIOS, BMC, and the hypervisor on the host are being upgraded. During these upgrades, VMs that have a GPU configuration are powered off and then powered on automatically when the node is back up.
- VM pause and resume are not supported.
- You cannot hot add VM memory if the VM is using a GPU.
- Hot add and hot remove support is not available for GPUs.
- You can change the GPU configuration of a VM only when the VM is turned off.
- The Prism web console does not support console access for VMs that are configured with GPU pass-through. Before you configure GPU pass-through for a VM, set up an alternative means to access the VM. For example, enable remote access over RDP.

Removing GPU pass-through from a VM restores console access to the VM through the Prism web console.

## Configuring GPU Pass-Through

For information about configuring GPU pass-through for guest VMs, see *Creating a VM (AHV)* in the "Virtual Machine Management" chapter of the *Prism Element Web Console Guide*.

## NVIDIA GRID Virtual GPU Support on AHV

AHV supports NVIDIA GRID technology, which enables multiple guest VMs to use the same physical GPU concurrently. Concurrent use is made possible by dividing a physical GPU into discrete virtual GPUs (vGPUs) and allocating those vGPUs to guest VMs. Each vGPU is allocated a fixed range of the physical GPU's framebuffer and uses all the GPU processing cores in a time-sliced manner.



Virtual GPUs are of different types (vGPU types are also called vGPU profiles) and differ by the amount of physical GPU resources allocated to them and the class of workload that they target. The number of vGPUs into which a single physical GPU can be divided therefore depends on the vGPU profile that is used on a physical GPU.

Each physical GPU supports more than one vGPU profile, but a physical GPU cannot run multiple vGPU profiles concurrently. After a vGPU of a given profile is created on a physical GPU (that is, after a vGPU is allocated to a VM that is powered on), the GPU is restricted to that vGPU profile until it is freed up completely. To understand this behavior, consider that you configure a VM to use an M60-1Q vGPU. When the VM is powering on, it is allocated an M60-1Q vGPU instance only if a physical GPU that supports M60-1Q is either unused or already running the M60-1Q profile and can accommodate the requested vGPU.

If an entire physical GPU that supports M60-1Q is free at the time the VM is powering on, an M60-1Q vGPU instance is created for the VM on the GPU, and that profile is locked on the GPU. In other words, until the physical GPU is completely freed up again, only M60-1Q vGPU instances can be created on that physical GPU (that is, only VMs configured with M60-1Q vGPUs can use that physical GPU).

Note: NVIDIA does not support Windows Guest VMs on the C-series NVIDIA vGPU types. See the NVIDIA documentation on Virtual GPU software for more information.

## NVIDIA Grid Host Drivers and License Installation

To enable guest VMs to use vGPUs on AHV, you must install NVIDIA drivers on the guest VMs, install the NVIDIA GRID host driver on the hypervisor, and set up an NVIDIA GRID License Server.

See the [NVIDIA Grid Host Driver for Nutanix AHV Installation Guide](#) for details about the workflow to enable guest VMs to use vGPUs on AHV and the NVIDIA GRID host driver installation instructions.

## vGPU Profile Licensing

vGPU profiles are licensed through an NVIDIA GRID license server. The choice of license depends on the type of vGPU that the applications running on the VM require. Licenses are available in various editions, and the vGPU profile that you want might be supported by more than one license edition.

Note: If the specified license is not available on the licensing server, the VM starts up and functions normally, but the vGPU runs with reduced capability.

You must determine the vGPU profile that the VM requires, install an appropriate license on the licensing server, and configure the VM to use that license and vGPU type. For information about licensing for different vGPU types, see the NVIDIA GRID licensing documentation.

Guest VMs check out a license over the network when starting up and return the license when shutting down. As the VM is powering on, it checks out the license from the licensing server. When a license is checked back in, the vGPU is returned to the vGPU resource pool.

When powered on, guest VMs use a vGPU in the same way that they use a physical GPU that is passed through.

## Supported GPU Cards

For a list of supported GPUs, see [Supported GPUs](#) on page 194.



## High Availability Support for VMs with vGPUs

Nutanix conditionally supports high availability (HA) of VMs that have NVIDIA GRID vGPUs configured. The cluster does not reserve any specific resources to guarantee High Availability for the VMs with vGPUs. The vGPU VMs are restarted on best effort basis in the event of a node failure. You can restart a VM with vGPUs on another (failover) host which has compatible or identical vGPU resources available. The vGPU profile available on the failover host must be identical to the vGPU profile configured on the VM that needs HA. The system attempts to restart the VM after an event. If the failover host has insufficient memory and vGPU resources for the VM to start, the VM fails to start after failover.

The following conditions are applicable to HA of VMs with vGPUs:

- Memory is not reserved for the VM on the failover host by the HA process. When the VM fails over, if sufficient memory is not available, the VM cannot power on.
- vGPU resource is not reserved on the failover host. When the VM fails over, if the required vGPU resources are not available on the failover host, the VM cannot power on.

## Limitations for vGPU Support

vGPU support on AHV has the following limitations:

- You cannot hot-add memory to VMs that have a vGPU.
- The Prism web console does not support console access for a VM that is configured with multiple vGPUs. The Prism web console supports console access for a VM that is configured with a single vGPU only.

Before you add multiple vGPUs to a VM, set up an alternative means to access the VM. For example, enable remote access over RDP. For Linux VMs, instead of RDP, use Virtual Network Computing (VNC) or equivalent.

- The vGPU on AHV supports only the guest operating systems that are supported by NVIDIA.

## Console Support for VMs with vGPU

Like other VMs, you can access a VMs with a single vGPU using the console. Enable or disable console support for a VM with only one vGPU configured. Enabling console support for a VM with multiple vGPUs is not supported. By default, console support for a vGPU VM is disabled.

## Recovery of vGPU Console-enabled VMs

With AHV, you can recover vGPU console-enabled guest VMs efficiently. When you perform DR of vGPU console-enabled guest VMs, the VMs recovers with the vGPU console. The guest VMs fail to recover when you perform cross-hypervisor disaster recovery (CHDR).

For AHV with minimum AOS versions 6.1, 6.0.2.4 and 5.20:

- vGPU-enabled VMs can be recovered when protected by protection domains in PD-based DR or protection policies in Leap based solutions using asynchronous, NearSync, or Synchronous (Leap only) replications.

Note: GPU Passthrough is not supported.

- If both site A and site B have the same GPU boards (and the same assignable vGPU profiles), failovers work seamlessly. However, with protection domains, no additional steps are required. GPU profiles are restored correctly and vGPU console settings persist after recovery. With Leap DR, vGPU console settings do not persist after recovery.



- If site A and site B have different GPU boards and vGPU profiles, you must manually remove the vGPU profile before you power on the VM in site B.

The vGPU console settings are persistent after recovery and all failovers are supported for the following:

**Table 30: Persistent vGPU Console Settings with Failover Support**

Recovery using	For vGPU enabled AHV VMs
Protection domain based DR	Yes
VMware SRM with Nutanix SRA	Not applicable

For information about the behavior See the [Recovery of vGPU-enabled VMs](#) topic in the *Data Protection and Recovery with Prism Element* guide.

See [Enabling or Disabling Console Support for vGPU VMs](#) on page 204 for more information about configuring the support.

For SRA and SRM support, see the [Nutanix SRA documentation](#).

### ADS support for VMs with vGPUs

AHV supports Acropolis Dynamic Scheduling (ADS) for VMs with vGPUs.

**Note:** ADS support requires live migration of VMs with vGPU be operational in the cluster. See *Live Migration of VMs with vGPUs* above for minimum NVIDIA and AOS versions that support live migration of VMs with vGPUs.

When a number of VMs with vGPUs are running on a host and you enable ADS support for the cluster, the Lazan manager invokes VM migration tasks to resolve resource hotspots or fragmentation in the cluster to power on incoming vGPU VMs. The Lazan manager can migrate vGPU-enabled VMs to other hosts in the cluster only if:

- The other hosts support compatible or identical vGPU resources as the source host (hosting the vGPU-enabled VMs).
- The host affinity is not set for the vGPU-enabled VM.

For more information about limitations, see [Live Migration of vGPU-enabled VMs](#) on page 202 and [Limitations of Live Migration Support](#) on page 204.

For more information about ADS, see [Acropolis Dynamic Scheduling in AHV](#) on page 8.

### Multiple Virtual GPU Support

Prism Central and Prism Element Web Console can deploy VMs with multiple virtual GPU instances. This support harnesses the capabilities of NVIDIA GRID virtual GPU (vGPU) support for multiple vGPU instances for a single VM.

**Note:** Multiple vGPUs on the same VM are supported on NVIDIA Virtual GPU software version 10.1 (440.53) or later.

You can deploy virtual GPUs of different types. A single physical GPU can be divided into the number of vGPUs depending on the type of vGPU profile that is used on the physical GPU. Each physical GPU on a GPU board supports more than one type of vGPU profile. For example, a Tesla® M60 GPU device provides different types of vGPU profiles like M60-0Q, M60-1Q, M60-2Q, M60-4Q, and M60-8Q.



You can only add multiple vGPUs of the same type of vGPU profile to a single VM. For example, consider that you configure a VM on a Node that has one NVIDIA Tesla® M60 GPU board. Tesla® M60 provides two physical GPUs, each supporting one M60-8Q (profile) vGPU, thus supporting a total of two M60-8Q vGPUs for the entire host.

For restrictions on configuring multiple vGPUs on the same VM, see [Restrictions for Multiple vGPU Support](#) on page 199.

For steps to add multiple vGPUs to the same VM, see [Creating a VM \(AHV\)](#) and [Adding Multiple vGPUs to a VM](#) information in *Prism Element Web Console Guide* or [Creating a VM through Prism Central \(AHV\)](#) and [Adding Multiple vGPUs to a VM](#) information in *Prism Central Infrastructure Guide*.

## Restrictions for Multiple vGPU Support

You can configure multiple vGPUs subject to the following restrictions:

- All the vGPUs that you assign to one VM must be of the same type. In the aforesaid example, with the Tesla® M60 GPU device, you can assign multiple M60-8Q vGPU profiles. You cannot assign one vGPU of the M60-1Q type and another vGPU of the M60-8Q type.

Note: You can configure any number of vGPUs of the same type on a VM. However, the cluster calculates a maximum number of vGPUs of the same type per VM. This number is defined as `max_instances_per_vm`. This number is variable and changes based on the GPU resources available in the cluster and the number of VMs deployed. If the number of vGPUs of a specific type that you configured on a VM exceeds the `max_instances_per_vm` number, then the VM fails to power on and the following error message is displayed:

```
Operation failed: NoHostResources: No host has enough available GPU for VM <name of VM>(UUID of VM).  
You could try reducing the GPU allotment...
```

When you configure multiple vGPUs on a VM, after you select the appropriate vGPU type for the first vGPU assignment, Prism (Prism Central and Prism Element Web Console)

automatically restricts the selection of vGPU type for subsequent vGPU assignments to the same VM.

Add GPU?×

Some vGPU profiles are hidden because you can only add vGPUs with the same profile.×

GPU Mode

☒ vGPU

☐ Passthrough

NVIDIA Virtual GPU License ?

GRID-Virtual-WS,2.0 ▼

VGPU Profile

GPU resources from the selected profile will be assigned to this VM while it is powered on.

NAME	VIRTUAL SLICE <span>?</span>	FRAMEBUFFER	VMS ASSIGNED	DETAILS
<div><input checked="" type="radio"/></div> GRID M60-8Q	1/1 GPU	8 GiB	0	<span>?</span>

Close

Add

Figure 38: vGPU Type Restriction Message

Note:



You can use CLI (accli) to configure multiple vGPUs of multiple types to the same VM. See *Acropolis Command-Line Interface (aCLI)* for information about aCLI. Use the **vm.gpu\_assign <vm.name> gpu=<gpu-type>** command multiple times, once for each vGPU, to add multiple vGPUs of multiple types to the same VM.

See the GPU board and software documentation for information about the combinations of the number and types of vGPUs profiles supported by the GPU resources installed in the cluster. For example, see the NVIDIA Virtual GPU Software Documentation for the vGPU type and number combinations on the Tesla® M60 board.

- Configure multiple vGPUs only of the highest type using Prism. The highest type of vGPU profile is based on the driver deployed in the cluster. In the aforesaid example, on a Tesla® M60 device, you can only configure multiple vGPUs of the M60-8Q type. Prism prevents you from configuring multiple vGPUs of any other type such as M60-2Q.



Figure 39: vGPU Type Restriction Message

**Note:**

You can use CLI (accli) to configure multiple vGPUs of other available types. See *Acropolis Command-Line Interface (aCLI)* for the aCLI information. Use the **vm.gpu\_assign <vm.name> gpu=<gpu-type>** command multiple times, once for each vGPU, to configure multiple vGPUs of other available types.

See the GPU board and software documentation for more information.

- Configure either a passthrough GPU or vGPUs on the same VM. You cannot configure both passthrough GPU and vGPUs. Prism automatically disallows such configurations after the first GPU is configured.
- The VM powers on only if the requested type and number of vGPUs are available in the host. In the aforesaid example, the VM, which is configured with two M60-8Q vGPUs, fails to power on if another VM sharing the same GPU board is already using one M60-8Q vGPU. This is because the Tesla® M60 GPU board allows only two M60-8Q vGPUs. Of these, one is already used by another VM. Thus, the VM configured with two M60-8Q vGPUs fails to power on due to unavailability of required vGPUs.

- Multiple vGPUs on the same VM are supported on NVIDIA Virtual GPU software version 10.1 (440.53) or later. Ensure that the relevant GRID version license is installed and select it when you configure multiple vGPUs.

## Adding Multiple vGPUs to the Same VM

### About this task

You can add multiple vGPUs of the same vGPU type when you create a new VM or update an existing VM.

- For information about how to create a VM, see [Creating a VM through Prism Central \(AHV\)](#).
- For information about how to update an existing VM, see [Updating a VM through Prism Central \(AHV\)](#).

For more information about multiple vGPU support, see [Multiple Virtual GPU Support](#) information in the *AHV Administration Guide*.

### Before you begin

Ensure that the following prerequisites are met before you add multiple vGPUs to the VM:

- Select the license for NVIDIA Virtual GPU software version 10.1 (440.53) or later.
- Observe the guidelines and restrictions specified in [Multiple Virtual GPU Support](#) and [Restrictions for Multiple vGPU Support](#).

### Procedure

To add multiple vGPUs to the same VM, perform the following steps:

1. Click **Add GPU** in the **Resources** step of create VM workflow or update VM workflow. For more information, see [Creating a VM through Prism Central \(AHV\)](#) or [Updating a VM through Prism Central \(AHV\)](#)

2. In the Add GPU window, click **Add**.

The License field is grayed out and you cannot select a different license when you add a vGPU for the same VM.

The **VGPU Profile** is auto-selected. The system allows you to select additional vGPU of the same vGPU type as indicated by the message at the top of the **Add GPU** window.

The newly added vGPU appears in the **Create VM** or **Update VM** window.

3. Repeat the steps for each vGPU addition to the VM.

### Live Migration of vGPU-enabled VMs

You can perform the live migration of VMs enabled with virtual GPUs (vGPU-enabled VMs) only on best effort basis, if the destination node is equipped to provide enough resources to the vGPU-enabled VMs. However, if the destination node is not equipped with the enough resources, the vGPU-enabled VMs are shut down and the VMs might experience downtime.

In a successful migration case, the vGPUs can continue to run while the VMs that are running the vGPUs are seamlessly migrated in the background.

When you perform the LCM update, the vGPU-enabled VMs are listed as *Non-HA-protected VMs*. LCM also migrates the non-HA-protected VMs on best effort basis to the destination node if the following requirements are met:



- Destination node is equipped with the required resources for the VM.
- The VM GPU drivers are compatible with the AHV host GPU drivers.

If the destination node is not equipped with the enough resources or there is any compatibility issue between the VM GPU drivers and AHV host GPU drivers, the LCM forcibly shuts down the non-HA-protected VMs.

Note: Live migration of VMs with vGPUs is supported for vGPUs created with minimum NVIDIA Virtual GPU software version 10.1 (440.53).

Table 31: Minimum Versions

Component	Supports	With Minimum Version
AOS	Live migration within the same cluster	5.18.1
AHV	Live migration within the same cluster	20190916.294
AOS	Live migration across cluster	6.1
AHV	Live migration across cluster	20201105.30142

Important: In an HA event involving any GPU node, the node locality of the affected vGPU-enabled VMs is not restored after GPU node recovery. The affected vGPU-enabled VMs are not migrated back to their original GPU host intentionally to avoid extended VM stun time expected during migration. If vGPU-enabled VM node locality is required, migrate the affected vGPU-enabled VMs to the desired host manually.

## Virtual GPU Limitations

Important frame buffer and VM stun time considerations are:

- The GPU board (for example, NVIDIA Tesla M60) vendor provides the information for maximum frame buffer size of vGPU types (for example, M60-8Q type) that can be configured on VMs. However, the actual frame buffer usage may be lower than the maximum sizes.
- The VM stun time depends on the number of vGPUs configured on the VM being migrated. Stun time may be longer in case of multiple vGPUs operating on the VM.

The stun time also depends on the network factors such bandwidth available for use during the migration.

## Live Migration Workflows

You can live migrate a vGPU-enabled VM to the following destinations:

- To another host within the same cluster. Both Prism Web Console (Prism Element) and Prism Central allow you to live migrate a vGPU-enabled VM to another host within the same cluster.
- To a host outside the cluster, that is, a host in another cluster. Only Prism Central allows you to migrate a vGPU-enabled VM to a host outside the cluster.)

For information about the steps to live migrate a vGPU-enabled VM, see the following:

- [Migrating Live a vGPU-enabled VM Within the Cluster](#) in the *Prism Web Console Guide*.
- [Migrating Within the Cluster](#) in the *Prism Central Infrastructure Guide*.



## Limitations of Live Migration Support

- Live migration is supported only for VMs configured with single or multiple virtual GPUs. It is not supported for VMs configured with passthrough GPUs.
- Live migration to a host in another cluster is supported only if the VM is protected by protection policy with Synchronous replication schedule.
- The target host for the migration must have adequate and available GPU resources, with the same vGPU types as configured for the VMs to be migrated, to support the vGPUs on the VMs that need to be migrated.

See [Restrictions for Multiple vGPU Support](#) on page 199 for more details.

- The vGPU-enabled VMs that need to be migrated live cannot be protected with high availability.
- Ensure that the VM is not powered off.
- Ensure that you have the right GPU software license(an appropriate license of NVIDIA GRID software version) that supports live migration of vGPUs. The source and target hosts must have the same license type.

## Enabling or Disabling Console Support for vGPU VMs

### About this task

Enable or disable console support for a VM with only one vGPU configured. Enabling console support for a VM with multiple vGPUs is not supported. By default, console support for a vGPU VM is disabled.

To enable or disable console support for each VM with vGPUs, do the following:

### Procedure

1. Run the following aCLI command to check if console support is enabled or disabled for the VM with vGPUs.

```
accli> vm.get vm-name
```

Where **vm-name** is the name of the VM for which you want to check the console support status.

The step result includes the following parameter for the specified VM:

```
gpu_console=False
```

Where **False** indicates that console support is not enabled for the VM. This parameter is displayed as **True** when you enable console support for the VM. The default value for **gpu\_console** is **False** since console support is disabled by default.

Note: The console may not display the **gpu\_console** parameter in the output of the **vm.get** command if the **gpu\_console** parameter was not previously enabled.



2. Run the following aCLI command to enable or disable console support for the VM with vGPU:

```
vm.update vm-name gpu_console=true | false
```

Where:

- **true**—indicates that you are enabling console support for the VM with vGPU.
- **false**—indicates that you are disabling console support for the VM with vGPU.

3. Run the `vm.get` command to check if `gpu_console` value is **true** indicating that console support is enabled or **false** indicating that console support is disabled as you configured it.

If the value indicated in the `vm.get` command output is not what is expected, then perform **Guest Shutdown** of the VM with vGPU. Next, run the `vm.on vm-name` aCLI command to turn the VM on again. Then run `vm.get` command and check the `gpu_console=` value.

4. Click a VM name in the VM table view to open the VM details page. Click **Launch Console**. The Console opens but only a black screen is displayed.

5. Click on the console screen. Click one of the following key combinations based on the operating system you are accessing the cluster from.

» For Apple Mac OS: **Control+Command+2**

» For MS Windows: **Ctrl+Alt+2**

The console is fully enabled and displays the content.

## PXE Configuration for AHV VMs

You can configure a VM to boot over the network in a Preboot eXecution Environment (PXE). Booting over the network is called PXE booting and does not require the use of installation media. When starting up, a PXE-enabled VM communicates with a DHCP server to obtain information about the boot file it requires.

Configuring PXE boot for an AHV VM involves performing the following steps:

- Configuring the VM to boot over the network.
- Configuring the PXE environment.

The procedure for configuring a VM to boot over the network is the same for managed and unmanaged networks. The procedure for configuring the PXE environment differs for the two network types, as follows:

- An unmanaged network does not perform IPAM functions and gives VMs direct access to an external Ethernet network. Therefore, the procedure for configuring the PXE environment for AHV VMs is the same as for a physical machine or a VM that is running on any other hypervisor. VMs obtain boot file information from the DHCP or PXE server on the external network.
- A managed network intercepts DHCP requests from AHV VMs and performs IP address management (IPAM) functions for the VMs. Therefore, you must add a TFTP server and the required boot file information to the configuration of the managed network. VMs obtain boot file information from this configuration.

A VM that is configured to use PXE boot boots over the network on subsequent restarts until the boot order of the VM is changed.



## Configuring the PXE Environment for AHV VMs

The procedure for configuring the PXE environment for a VM on an unmanaged network is similar to the procedure for configuring a PXE environment for a physical machine on the external network and is beyond the scope of this document. This procedure configures a PXE environment for a VM in a managed network on an AHV host.

### About this task

To configure a PXE environment for a VM on a managed network on an AHV host, do the following:

### Procedure

1. Log on to the Prism web console, click the gear icon, and then click **Network Configuration** in the menu.
2. On **Network Configuration** > **Subnets** tab, click the **Edit** action link of the network for which you want to configure a PXE environment.  
The VMs that require the PXE boot information must be on this network.
3. In the **Update Subnet** dialog box:
  - a. Select the **Enable IP address management** check box and complete the following configurations:
    - In the **Network IP Prefix** field, enter the network IP address, with prefix, of the subnet that you are updating.
    - In the **Gateway IP Address** field, enter the gateway IP address of the subnet that you are updating.
    - To provide DHCP settings for the VM, select the **DHCP Settings** check box and provide the following information.

Fields	Description and Values
Domain Name Servers	Provide a comma-separated list of DNS IP addresses. Example: 8.8.8.8, or 9.9.9.9
Domain Search	Enter the VLAN domain name. Use only the domain name format. Example: nutanix.com
TFTP Server Name	Enter a valid TFTP host server name of the TFTP server where you host the host boot file. The IP address of the TFTP server must be accessible to the virtual machines to download a boot file. Example: tftp_vlan103
Boot File Name	The name of the boot file that the VMs need to download from the TFTP host server. Example: boot_ahv202010



4. Under **IP Address Pools**, click **Create Pool** to add IP address pools for the subnet.  
(Mandatory for Overlay type subnets) This section provides the **Network IP Prefix** and **Gateway IP** fields for the subnet.  
(Optional for VLAN type subnet) Check this box to display the **Network IP Prefix** and **Gateway IP** fields and configure the IP address details.
5. (Optional and for VLAN networks only) Check the **Override DHCP Server** dialog box and enter an IP address in the **DHCP Server IP Address** field.  
  
You can configure a DHCP server using the **Override DHCP Server** option only in case of VLAN networks.  
  
The DHCP Server IP address (reserved IP address for the Acropolis DHCP server) is visible only to VMs on this network and responds only to DHCP requests. If this box is not checked, the DHCP Server IP Address field is not displayed and the DHCP server IP address is generated automatically. The automatically generated address is `network_IP_address_subnet.254`, or if the default gateway is using that address, `network_IP_address_subnet.253`.  
  
Usually the default DHCP server IP is configured as the last usable IP in the subnet (For example, its 10.0.0.254 for 10.0.0.0/24 subnet). If you want to use a different IP address in the subnet as the DHCP server IP, use the override option.
6. Click **Close**.

## Configuring a VM to Boot over a Network

To enable a VM to boot over the network, update the VM's boot device setting. Currently, the only user interface that enables you to perform this task is the Acropolis CLI (aCLI).

### About this task

To configure a VM to boot from the network, do the following:

### Procedure

1. Log on to any CVM in the cluster using SSH.
2. Create a VM.

```
nutanix@cvm$ acli vm.create vm num_vcpus=num_vcpus memory=memory
```

Replace `vm` with a name for the VM, and replace `num_vcpus` and `memory` with the number of vCPUs and amount of memory that you want to assign to the VM, respectively.

For example, create a VM named nw-boot-vm.

```
nutanix@cvm$ acli vm.create nw-boot-vm num_vcpus=1 memory=512
```

3. Create a virtual interface for the VM and place it on a network.

```
nutanix@cvm$ acli vm.nic_create vm network=network
```

Replace `vm` with the name of the VM and replace `network` with the name of the network. If the network is an unmanaged network, make sure that a DHCP server and the boot file that the VM requires are available on the network. If the network is a managed network, configure the

DHCP server to provide TFTP server and boot file information to the VM. See [Configuring the PXE Environment for AHV VMs](#) on page 206.

For example, create a virtual interface for VM nw-boot-vm and place it on a network named network1.

```
nutanix@cvm$ accli vm.nic_create nw-boot-vm network=network1
```

4. Obtain the MAC address of the virtual interface.

```
nutanix@cvm$ accli vm.nic_list vm
```

Replace **vm** with the name of the VM.

For example, obtain the MAC address of VM nw-boot-vm.

```
nutanix@cvm$ accli vm.nic_list nw-boot-vm
00-00-5E-00-53-FF
```

5. Update the boot device setting so that the VM boots over the network.

```
nutanix@cvm$ accli vm.update_boot_device vm mac_addr=mac_addr
```

Replace **vm** with the name of the VM and **mac\_addr** with the MAC address of the virtual interface that the VM must use to boot over the network.

For example, update the boot device setting of the VM named nw-boot-vm so that the VM uses the virtual interface with MAC address 00-00-5E-00-53-FF.

```
nutanix@cvm$ accli vm.update_boot_device nw-boot-vm mac_addr=00-00-5E-00-53-FF
```

6. Power on the VM.

```
nutanix@cvm$ accli vm.on vm_list [host="host"]
```

Replace **vm\_list** with the name of the VM. Replace **host** with the name of the host on which you want to start the VM.

For example, start the VM named nw-boot-vm on a host named host-1.

```
nutanix@cvm$ accli vm.on nw-boot-vm host="host-1"
```

## Uploading Files to DSF for Microsoft Windows Users

If you are a Microsoft Windows user, you can securely upload files to DSF by using the following procedure.

### Procedure

1. Use WinSCP, with SFTP selected, to connect to Controller VM through port 2222 and start browsing the DSF datastore.

Note: The root directory displays storage containers and you cannot change it. You can only upload files to one of the storage containers and not directly to the root directory. To create or delete storage containers, you can use the Prism user interface.

2. Authenticate by using Prism username and password or, for advanced users, use the public key that is managed through the Prism cluster lockdown user interface.





## Enabling Load Balancing of vDisks in a Volume Group

AHV hosts support load balancing of vDisks in a volume group for guest VMs. Load balancing of vDisks in a volume group enables IO-intensive VMs to use the storage capabilities of multiple Controller VMs (CVMs).

### About this task

If you enable load balancing on a volume group, the guest VM communicates directly with each CVM hosting a vDisk. Each vDisk is served by a single CVM. Therefore, to use the storage capabilities of multiple CVMs, create more than one vDisk for a file system and use the OS-level striped volumes to spread the workload. This configuration improves performance and prevents storage bottlenecks.

#### Note:

- vDisk load balancing is disabled by default for volume groups that are directly attached to VMs.  
However, vDisk load balancing is enabled by default for volume groups that are attached to VMs by using a data services IP address.
- If you use web console to clone a volume group that has load balancing enabled, the volume group clone does not have load balancing enabled by default. To enable load balancing on the volume group clone, you must set the `load_balance_vm_attachments` parameter to true using acli or Rest API.
- You can attach a maximum number of 10 load balanced volume groups per guest VM.
- For Linux VMs, ensure that the SCSI device timeout is 60 seconds. For information about how to check and modify the SCSI device timeout, see the Red Hat documentation at [https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/5/html/online\\_storage\\_reconfiguration\\_guide/task\\_controlling-scsi-command-timer-onlining-devices](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/5/html/online_storage_reconfiguration_guide/task_controlling-scsi-command-timer-onlining-devices).

Perform the following procedure to enable load balancing of vDisks by using aCLI.

### Procedure

1. Log on to a Controller VM with SSH.

2. Do one of the following:

- » Enable vDisk load balancing if you are creating a volume group.

```
nutanix@cvm$ acli vg.create vg_name load_balance_vm_attachments=true
```

Replace `vg_name` with the name of the volume group.

- » Enable vDisk load balancing if you are updating an existing volume group.

```
nutanix@cvm$ acli vg.update vg_name load_balance_vm_attachments=true
```

Replace `vg_name` with the name of the volume group.

Note: To modify an existing volume group, you must first detach all the VMs that are attached to that volume group before you enable vDisk load balancing.



3. Verify if vDisk load balancing is enabled.

```
nutanix@cvm$ acli vg.get vg_name
```

An output similar to the following is displayed:

```
nutanix@cvm$ acli vg.get ERA_DB_VG_XXXXXXX
ERA_DB_VG_XXXXXXX {
  attachment_list {
    vm_uuid: "XXXXXX"
  }
  .
  .
  .
  .
  iscsi_target_name: "XXXXXX"
  load_balance_vm_attachments: True
  logical_timestamp: 4
  name: "ERA_DB_VG_XXXXXXX"
  shared: True
  uuid: "XXXXXX"
}
```

If vDisk load balancing is enabled on a volume group, `load_balance_vm_attachments: True` is displayed in the output. The output does not display the `load_balance_vm_attachments` parameter at all if vDisk load balancing is disabled.

4. (Optional) Disable vDisk load balancing.

```
nutanix@cvm$ acli vg.update vg_name load_balance_vm_attachments=false
```

Replace `vg_name` with the name of the volume group.

## VM High Availability in Acropolis

Acropolis uses the segment-based reservation method to enable VM high availability. The host-based reservation method is deprecated, and not supported.

If you have not enabled High Availability, in case of host failure, the VMs are restarted from the failed host to any available space on the other hosts in the cluster. Once the failed host joins the cluster again, VMs are migrated back to the host. This type of VM high availability is implemented without reserving any resources. Admission control is not enforced and hence there may not be sufficient capacity available to start all the VMs.

Note:

- Nutanix does not support VMs that are running with 100% remote storage for High Availability. The VMs must have at least one local disk that is present on the cluster.
- The VM HA does not reserve the memory for the non-migratable VMs. For information on how to check the non-migratable VMs, see [Checking Live Migration Status of a VM](#) in the *Prism Central Guide*.

The VM HA uses the Guarantee mode with segment-based reservation method.

In segment-based reservation, the cluster is divided into segments to ensure that enough space is reserved for any host failure. Each segment corresponds to the largest VM that is guaranteed to be restarted in case the failure occurs. The other factor is the number of host failures that can be tolerated. Using these inputs, the scheduler implements admission control to always have enough resources reserved so that the VMs can be restarted upon failure of any host in the cluster.



The segment size ensures that the largest VM can be powered on in HA failover when cluster is fully loaded (if the cluster is fully used except the reserved segments). The number of segments that is reserved is such a way that for each host enough resources are reserved to ensure that any host failure in the cluster is tolerated. Multiple VMs may fit into a segment. If anything changes in the cluster, the reservation is computed again. The total resources reserved for segments can be more than the resources used by running VMs. This implementation guarantees successful failover even in the case of fragmentation of segments. The actual number of reserved resources depends on the current load of the cluster, but it is typically at 1 to 1.25 times the resource usage on the most loaded host.

If the host enters maintenance mode (in case of host upgrade), you might not be protected against further host failures. Maintenance mode uses reservations made for HA for migrating VMs from the host. Although you are not protected against host failure if you have reservation for HA, hypervisor upgrade occurs without any difficulty because from the perspective of a user it is the same as host failure except that the VMs are migrated (instead of restarted) and hence no runtime state is lost. The HA status goes through the same states as it goes when the host failure had occurred.

### **Fault Detection for High Availability**

Acropolis version 6.1 (with minimum supported AHV version 20201105.2229) onwards, the fault detection mechanism for High Availability checks for the heartbeat of the management services on the host. If any one of the services are down, then the host is marked as disconnected.

In addition to this, the fault detection mechanism also performs the following health checks for the host:

- Root file system corruption
- Read-only root file system

If any of the above-mentioned health checks are affirmative, then the host is marked as disconnected.

If the host remains in the disconnected status for 40 seconds, the VMs running on the affected host are automatically restarted (based on the resource availability) .

You can view the alerts raised for any of the above-mentioned checks in the **Activity > Alerts** view in Prism UI.

## **Enabling High Availability for the Cluster**

In Acropolis managed clusters, you can enable high availability for the cluster to ensure that VMs can be migrated and restarted on another host in case of failure.

### **About this task**

After you enable high availability for the cluster, if a host failure occurs the cluster goes through following changes.

- OK: This state implies that the cluster is protected against a host failure.
- Healing: Healing period is the time that Acropolis brings the cluster to the protected state. There are two phases to this state. The first phase occurs when the host fails. The VMs are restarted on the available host. After restarting all the VMs if there are enough resources to protect the VM, the HA status of the cluster comes back to OK state. If this does not occur, the cluster goes into critical state. The second phase occurs when the host comes back from the failure. Once the host comes back from failure, no VMs are present on the host and hence during this healing phase restore locality task occurs (VMs are migrated back). Apart from restoring the locality of the VMs, the restore locality task ensures that the cluster is back to the same state before the HA failure. Once it is finished, the HA status is back to OK state.



- Critical: If the host is down, the HA status of the cluster goes into Critical state. This happens because the cluster cannot tolerate any more host failures. You have to ensure that you bring back the host so that your cluster is protected against any further host failures.

Note: On a less loaded cluster, it is possible for HA to go directly back to OK state if enough resources is reserved to protect another host failure. The start and migrate operations on the VMs are restricted in the Critical state because Acropolis continuously tries to ensure that the HA status is back to the OK state.

### Procedure

1. Log in to the Prism Element web console.
2. Click the gear icon in the main menu and then select **Manage VM High Availability** in the **Settings** page.

Note: This option does not appear in clusters that do not support this feature.

The Manage VM High Availability dialog box appears.

3. Check the **Enable HA Reservation** box and then click the **Save** button to enable.

## Viewing list of restarted VMs after an HA event

This section provides the information about how to view the list of VMs that are restarted after an HA event in the AHV cluster.

### About this task

If an AHV host becomes inaccessible or fails due to some unplanned event, the AOS restarts the VMs across the remaining hosts in the cluster.

### Procedure

To view the list of restarted VMs after an HA event:

1. Log in to Prism Central or Prism web console.

2. View the list of restarted VMs on either of the following pages:

- **Events** page:

1. Select the **Infrastructure** application from [Application Switcher Function](#), and navigate to **Activity > Events** in Prism Central. For more information, see [Prism Central Alerts and Events Reference Guide](#).

Navigate to **Alerts > Events** from the main menu to access the **Events** page in the *Prism web console*.

2. Locate or search for the following string, and hover over or click the string:

**VMs restarted due to HA failover**

The system displays the list of restarted VMs in the **Summary** page and as a hover text for the selected event.

For example:

VMs restarted due to HA failover: <VM\_Name1>, <VM\_Name2>, <VM\_Name3>, <VM\_Name4>. VMs were running on host X.X.X.1 prior to HA.

Observe <VM\_Name1>, <VM\_Name2>, <VM\_Name3>, and <VM\_Name4> as the actual VMs in your cluster.

- **Tasks** page:

1. Select the **Infrastructure** application from [Application Switcher Function](#), and navigate to **Activity > Tasks** in Prism Central.

Navigate to **Tasks** from the main menu to access the **Tasks** page in the Prism web console.

2. Locate or search for the following task, and click **Details**:

**HA failover**

The system displays a list of related tasks for the HA failover event.

3. Locate or search for the following related task, and click **Details**:

**Host restart all VMs**

The system displays **Restart VM group** task for the HA failover event.

4. In the **Entity Affected** column, click **Details**, or hover over the **VMs** text for **Restart VM group** task:

The system displays the list of restarted VMs:



Figure 40: List of restarted VMs

## Live vDisk Migration Across Storage Containers

vDisk migration allows you to change the container of a vDisk. You can migrate vDisks across storage containers while they are attached to guest VMs without the need to shut down or delete VMs (live migration). You can either migrate all vDisks attached to a VM or migrate specific vDisks to another container.

In a Nutanix solution, group the vDisks into storage containers and attach vDisks to guest VMs. AOS applies storage attributes such as replication factor, encryption, compression, deduplication, and erasure coding at the storage container level. If you apply a storage policy to a storage container, AOS enables that policy on all the vDisks of the container. If you want to change the policies of the vDisks (for example, from RF2 to RF3), create another container with a different policy and move the vDisk to that container. With live migration of vDisks across containers, you can migrate vDisk across containers even if those vDisks are attached to a live VM. Thus, live migration of vDisks across storage containers enables you to efficiently manage storage policies for guest VMs.

### General Considerations

You cannot migrate images or volume groups.

You cannot perform the following operations during an ongoing vDisk migration:

- Clone the VM
- Resize the VM
- Take a snapshot

Note: During vDisk migration, the logical usage of a vDisk is more than the total capacity of the vDisk. The issue occurs because the logical usage of the vDisk includes the space occupied in both the source and destination containers. Once the migration is complete, the logical usage of the vDisk returns to its normal value.

Migration of vDisks stalls if sufficient storage space is not available in the target storage container. Ensure that the target container has sufficient storage space before you begin migration.

### Disaster Recovery Considerations

Consider the following points if you have a disaster recovery and backup setup:

- You cannot migrate vDisks of a VM that is protected by a protection domain or protection policy. When you start the migration, ensure that the VM is not protected by a protection domain or protection policy. If you want to migrate vDisks of such a VM, do the following:
  - Remove the VM from the protection domain or protection policy.
  - Migrate the vDisks to the target container.
  - Add the VM back to the protection domain or protection policy.
  - Configure the remote site with the details of the new container.
- vDisk migration fails if the VM is protected by a protection domain or protection policy.
- If you are using a third-party backup solution, AOS temporarily blocks snapshot operations for a VM if vDisk migration is in progress for that VM.

## Migrating vdisk from Prism Central

Starting with Prism Central 2023.4, the individual VM disks can be migrated using Prism Central UI. For more information, see [Updating a VM through Prism Central \(AHV\)](#) section in *Prism Central Infrastructure Guide*. If more granular control is required, perform the steps through acli as described in [Migrating a vDisk to Another Container](#) on page 215.

## Migrating a vDisk to Another Container

You can either migrate all vDisks attached to a VM or migrate specific vDisks to another container.

### About this task

Note: Starting with Prism Central 2023.4, the individual VM disks can be migrated using Prism Central UI. For more information, see [Updating a VM through Prism Central \(AHV\)](#) section in *Prism Central Infrastructure Guide*. If more granular control is required, perform the steps through acli as described in this section.

### Procedure

Perform the following procedure to migrate vDisks across storage containers:

1. Log on to a CVM in the cluster with SSH.
2. Do one of the following:
  - » Migrate all vDisks of a VM to the target storage container.

```
nutanix@cvm$ acli vm.update_container vm-name container=target-container wait=false
```

Replace **vm-name** with the name of the VM whose vDisks you want to migrate and **target-container** with the name of the target container.

- » Migrate specific vDisks by using either the UUID of the vDisk or address of the vDisk.

Migrate specific vDisks by using the UUID of the vDisk.

```
nutanix@cvm$ acli vm.update_container vm-name device_uuid_list=device_uuid  
container=target-container wait=false
```

Replace **vm-name** with the name of VM, **device\_uuid** with the device UUID of the vDisk, and **target-container** with the name of the target storage container.

Run **nutanix@cvm\$ acli vm.get <vm-name>** to determine the device UUID of the vDisk.

You can migrate multiple vDisks at a time by specifying a comma-separated list of device UUIDs of the vDisks.

Alternatively, you can migrate vDisks by using the address of the vDisk.

```
nutanix@cvm$ acli vm.update_container vm-name disk_addr_list=disk-address  
container=target-container wait=false
```

Replace **vm-name** with the name of VM, **disk-address** with the address of the disk, and **target-container** with the name of the target storage container.

Run **nutanix@cvm\$ acli vm.get <vm-name>** to determine the address of the vDisk.

Following is the format of the vDisk address:

```
bus.index
```

Following is a section of the output of the **acli vm.get vm-name** command:

```
disk_list {
```



```
addr {  
  bus: "scsi"  
  index: 0  
}
```

Combine the values of bus and index as shown in the following example:

```
nutanix@cvm$ acli vm.update_container TestUVM_1 disk_addr_list=scsi.0 container=test-  
container-17475
```

You can migrate multiple vDisks at a time by specifying a comma-separated list of vDisk addresses.

3. Check the status of the migration in the **Tasks** menu of the Prism Element web console.
4. (Optional) Cancel the migration if you no longer want to proceed with it.

```
nutanix@cvm$ ecli task.cancel task_list=task-ID
```

Replace **task-ID** with the ID of the migration task.

Determine the task ID as follows:

```
nutanix@cvm$ ecli task.list
```

In the **Type** column of the tasks list, look for **VmChangeDiskContainer**.

**VmChangeDiskContainer** indicates that it is a vDisk migration task. Note the ID of such a task.

Note: Note the following points about canceling migration:

- If you cancel an ongoing migration, AOS retains the vDisks that have not yet been migrated in the source container. AOS does not migrate vDisks that have already been migrated to the target container back to the source container.
- If sufficient storage space is not available in the original storage container, migration of vDisks back to the original container stalls. To resolve the issue, ensure that the source container has sufficient storage space.

## Memory Overcommit

The Memory Overcommit feature enables you to optimize the physical memory utilization of a VM. It allows the host to automatically detect whether the memory is under-utilized or over-utilized for a VM. Using Prism Central, you can enable the Memory Overcommit feature. For information on how to enable the Memory Overcommit feature, see [Memory Overcommit Management](#) on page 221.

After you enable the Memory Overcommit feature, the host performs the following actions:

- Automatically combines and uses the excess or unutilized memory of a VM, or even uses the disk space to swap out VM memory.
- Uses the reclaimed physical memory to provision the memory requirements of another VM, or to create another VM.

Note:

- Memory overcommit enables the host to optimize and overcommit memory for any existing or new VM within the host. It does not support the utilization of memory for VMs across the hosts. For example, overcommitted memory of VM1 on Host1 cannot be utilized on VMs on Host2. It can only be utilized for another VM on Host1 itself.





- The reclaiming of unutilized VM memory requires the system to observe multiple memory usage cycles of the VM to conclude on the memory usage pattern of the VM. The system might take up to 3 minutes for memory accounting process to reclaim the unutilized memory.

Caution: AHV does not support a memory overcommit of more than four times the allocated physical memory. For High Availability - Reserved Segments (HA-RS) setup, a memory overcommit beyond 33% of allocated physical memory leads to performance issues, and the system disallows the VM power on operation.

With the Memory Overcommit feature enabled on the host, the total memory assigned to all the VMs by the host can be greater than the total physical memory installed on the host. For more information on VM memory allocation with Memory Overcommit feature, see the deployment example described in [Memory Overcommit Deployment Example](#) on page 218.

Note: The Memory Overcommit feature is only available with AOS version 6.0.2 with a recommended minimum AHV version of 20201105.30142. The AHV version 20201105.30007 is the absolute minimum version with some Memory Overcommit support. For more information, see the *Acropolis Family Release Notes*.

## Deployment Workflow

You can enable or disable the Memory Overcommit feature using Prism Central on new or existing VMs.

Note: Ensure the VMs are shut down before you enable or disable the Memory Overcommit feature.

Memory Overcommit is useful in test and development environments to verify the performance of the VM. Based on the memory usage of the VM, AHV calculates an appropriate memory size for each VM enabled with Memory Overcommit.

### Essential Concepts

#### Over Provisioned VM

A VM with a memory allocation that is more than the actual requirement to run the application on it is called an Over Provisioned VM. For example, a VM with an allocation of 10 GB memory to run an SQL server that only requires 7 GB of memory to operate is categorized as an Over Provisioned VM. In this case, the VM is over provisioned by 3 GB. This over provisioned memory is unused by the VM. The host can use this unutilized memory to add a new VM or allocate more memory to an existing VM, even if the physical memory is not available on the host.

#### Memory Overcommitted VM

A memory overcommitted VM is a VM that has the Memory Overcommit feature enabled and shares its unutilized memory with the host. By default, each VM is guaranteed to have at least 25% of its memory provisioned as physical memory. The remaining 75% is either unutilized in the VM and reclaimed for use in other VMs or swapped out to the host swap disks.

Memory Overcommit is required for the following types of VMs:

- A VM with an unutilized memory due to over provisioning.
- A VM with workloads which are not sensitive to frequent memory access. The frequency of memory access is increased due to swapping of VM memory to host swap disks.



## Host Swap

After installing or upgrading the current AOS version of the cluster to AOS 6.0.2 with AHV 20201105.30007 or later, Memory Overcommit is available for the cluster. Swap disks are created on the ADSF for every host in the cluster.

Memory Overcommit reclaims memory from a guest VM that has unutilized memory by default. The host swap memory is utilized when the guest VM does not permit the reclamation of memory due to the amount of memory utilized in the guest VM.

Note: The performance of the swap memory depends on the hardware, workload, and the guest VM. The swap memory defined in the operating system of the guest VM (guest VM swap) may result in better performance characteristics than host swap. It is always recommended to increase guest VM swap instead of relying on host swap. For example, you can increase the guest VM swap up to RAM size of the guest VM.

### Memory Overcommit Deployment Example

*Example: A 128 GB host with three VMs (64 GB + 32 GB + 32 GB)*

#### *Observations Before Memory Overcommit*

Observe the following attributes before you enable Memory Overcommit:

- The physical memory of the host is equal to the total memory of all the VMs.
- The host cannot add a new VM as the host's physical memory is fully committed.

#### *Action*

Enable the Memory Overcommit feature on all the three VMs.

#### *Observations After Memory Overcommit*

If the VMs are not fully utilizing their allocated memory, the host allows you to create a new VM with a memory size that is equal to or less than the amount of unutilized memory. When the new VM is started, the host attempts to reclaim the unutilized memory from the running VMs and makes it available for the new VM.

The following table shows the example of a 128 GB host with the total utilized and unutilized memory:

**Table 32: Memory Overcommit Example**

Host 1 (128 GB)			
VMs	VM Memory (GB)	Utilized Memory (GB)	Unutilized Memory (GB)
VM1	64 GB	48 GB	16 GB
VM2	32 GB	20 GB	12 GB
VM3	32 GB	24 GB	8 GB
Total	128 GB	92 GB	36 GB

In this example, the host can combine the 36 GB of unutilized physical memory to create a new VM up to an aggregate size of 36 GB. Even though the unutilized memory can be reclaimed from a running VM, all the new VMs memory must be allocated from physical memory. In this case, you can start a new VM only with less than or equal to the total memory that is reclaimable from other VMs.



In the following table, you can observe the following attributes:

- VMs 1,2 and 3 are over-committed by 16 GB, 12 GB, and 8 GB respectively (thus over-committed by 36 GB cumulatively) which can be reclaimed by the host.
- The information on the memory reclaim process is also indicated in the *Utilized Memory (GB)* column.
- The host creates a new VM - VM4 with an allocated memory of 32 GB.

In this scenario, if the VM4 is only utilizing 28 GB of the allocated 32 GB, it is possible to start a new VM with up to 8GB of memory allocated with reclamation of unutilized memory of 4 GB.

Table 33: Memory Overcommit Example

Host 1 (128 GB)			
VMs	VM Memory (GB)	Utilized Memory (GB)	Unutilized Memory (GB)
VM1	64 GB	48 GB	4 GB
VM2	32 GB	20 GB	
VM3	32 GB	24 GB	
VM4	32 GB	28 GB	4 GB
Total	160 GB	120 GB	8 GB

Note: Before the Memory Overcommit feature is enabled, the physical memory of the host is equal to the total memory of all the VMs. After the memory overcommit feature is enabled on the host, the total of all the memory assigned to VMs by the host can *appear* to be greater than the total physical memory installed on the host.

*Important:* The above example is subject to the considerations mentioned in [Requirements](#) and [Limitations](#) sections. The guest VM utilizes the memory, which is reclaimed to permit the starting of a new VM, only when it experiences a greater memory pressure. In this case, the system identifies the requirement of additional memory for the guest VM and attempts to correctly balance the memory in other VMs based on their memory utilization. If the system is not able to reduce the memory pressure with unutilized memory, it falls back to utilize the host swap to reduce the memory pressure of the guest VM. For example, if VM1 listed in the above table requires an additional 10GB of memory, the system attempts to recover the unutilized 1GB from VM2, 1GB from VM3 and 4GB from VM4. You can observe that a shortfall of 4GB occurs to reduce the memory pressure in VM1, even when the system applies this action. In this case, the system allocates the shortfall memory of 4GB from host swap.

Note: ADS tracks the memory usage on the host and, if the host is in an over-committed state for memory, it is required to migrate VMs to mitigate over-committed memory hotspots.

## Requirements for Memory Overcommit

Memory Overcommit is only available with AOS 6.0.2 and later, and has the following requirements:

- Ensure that you have an AOS version with AHV 20201105.30007 or later. See *Release Notes*.



- Install the balloon drivers. For example, for Windows VM, install the latest [VirtIO driver package](#).

The balloon drivers enable the system to get visibility of the OS memory usage and reclaim the unutilized memory.

The latest VirtIO drivers ensure minimal impact on the performance of the Windows VM enabled with Memory Overcommit.

- With a minimum Prism Central version of pc.2022.4, you can enable or disable the Memory Overcommit feature using Prism Central on new or existing VMs.

## Limitations of Memory Overcommit

Memory overcommit has the following limitations:

- You can enable or disable Memory Overcommit only while the VM is powered off.
- Power off the VM enabled with memory overcommit before you change the memory allocation for the VM.

For example, you cannot update the memory of a VM that is enabled with memory overcommit when it is still running. The system displays the following alert: **InvalidVmState: Cannot complete request in state on.**

- Memory overcommit is not supported with VMs that use GPU configurations (both GPU passthrough and vGPU) and vNUMA.

For example, you cannot update a VM to a vNUMA VM when it is enabled with memory overcommit. The system displays the following alert: **InvalidArgument: Cannot use memory overcommit feature for a vNUMA VM error.**

- Memory overcommit feature can slow down the performance and the predictable performance of the VM.

For example, migrating a VM enabled with Memory Overcommit takes longer than migrating a VM not enabled with Memory Overcommit.

- There may be a temporary spike in the aggregate memory usage in the cluster during the migration of a VM enabled with Memory Overcommit from one node to another.

For example, when you migrate a VM from Node A to Node B, the total memory used in the cluster during migration is greater than the memory usage before the migration.

The memory usage of the cluster eventually drops back to pre-migration levels when the cluster reclaims the memory for other VM operations.

- Using Memory Overcommit heavily can cause a spike in the disk space utilization in the cluster. This spike is caused because the Host Swap uses some of the disk space in the cluster.

If the VMs do not have a swap disk, then in case of memory pressure, AHV uses space from the swap disk created on ADSF to provide memory to the VM. This can lead to an increase in disk space consumption on the cluster.

- All DR operations except Cross Cluster Live Migration (CCLM) are supported.

On the destination side, if a VM fails when you enable Memory Overcommit, the failed VM fails over (creating the VM on the remote site) as a fixed size VM. You can enable Memory Overcommit on this VM after the failover is complete.

- When Memory Overcommit is enabled, you cannot perform *Memory hot add* to include additional memory resources for a VM.



## Total Memory Allocation Mechanism on Linux and Windows VM

The guest OS for the VM should have the balloon drivers installed. For example, VirtIO drivers are needed for the Windows VM. This enables the system to get a visibility of the OS memory usage and reclaim the unutilized memory. For more information, see [Requirements for Memory Overcommit](#) on page 219.

When you enable the Memory Overcommit on a Linux VM and Windows VM with the same memory size, you can observe the following difference in memory allocation mechanism between Linux VM and Windows VM:

- In the *Linux VM* case, the system adjusts the *Total Memory* to accommodate the balloon driver memory consumption.
- In the *Windows VM* case, the system adjusts the *Used Memory* (memory usage) to accommodate the balloon driver memory consumption. The *Total Memory* always remains the same as assigned during initial provisioning.

The following table provides the information about the memory allocation observations made on the Linux VM and Windows VM:

Note:

- The cache, shared, and buffer memories are intentionally not shown in the following table. The system adjusts the cache memory in the Free Memory.
- The following attributes are used in the following table:
  - X - Indicates the Total Memory assigned to the VM during initial provisioning.
  - Y - Indicates the memory consumed by the Linux or Windows Balloon Drivers
  - Z - Indicates the Used Memory (VM memory usage)

Table 34: Memory Allocation - Linux VM and Windows VM

Total Memory assigned to the VM (in GB)	VM OS	Balloon Drivers Memory Consumption (GB)	Memory Allocation Observation on VM		
			Used Memory (in GB)	Free Memory (in GB)	Total Memory (in GB)
X	Linux	Y	Z	X - Z	X - Y
	Windows		Z + Y	X - (Y + Z)	X
<i>For example, if X = 16, Y = 0.5, and Z = 2.5</i>					
16	Linux	0.5	2.5	13.5	15.5
	Windows		3	13	16

## Memory Overcommit Management

### Enabling Memory Overcommit While Creating a VM

This procedure helps you enable memory overcommit in a VM that you create in Prism Central. You need a minimum Prism Central version of pc.2022.4 for this procedure.



## Before you begin

You cannot enable the memory overcommit feature using the Prism Element Web Console. Enable memory overcommit on a VM that you create by using Prism Central. If you create a VM using Prism Web Console, then you can enable memory overcommit on that VM using Prism Central. For the procedure to enable memory overcommit on a VM that you have already created, see [Enabling/Disabling Memory Overcommit on Existing VMs](#) on page 224.

## About this task

When you are creating a VM, you can enable memory overcommit for it.

## Procedure

To enable memory commit while creating a VM, perform the following steps:

1. Log in to Prism Central.
2. Select the **Infrastructure** application from [Application Switcher Function](#), and navigate to **Compute & Storage > VMs** from the **Navigation Bar**. For information about the **Navigation Bar**, see [Application-Specific Navigation Bar](#).  
The system displays the page with **Lists** tab by default with all the VMs across registered clusters.

3. Click **Create VM**.  
The system displays the Create VM window.

### Create VM

1

Configuration

2

Resources

3

Management

4

Review

Name

test-vm

Description

test vm

Cluster

auto\_cluster\_

Number of VMs

1

VM Properties

CPU

1

vCPU

Cores Per CPU

1

Cores

Memory

4

GIB

☐ Enable Memory Overcommit

Cancel

Next

Figure 41: Create VM - Enable Memory Overcommit

4. Enter all the details necessary to create the VM, and select the **Enable Memory Overcommit** checkbox.
5. Click **Next**.
6. Provide the details necessary for the VM creation in the Resources and Management steps, and click **Next**.

For more information, see [Creating a VM through Prism Central \(AHV\)](#) information in the *Prism Central Infrastructure Guide* .

7. In the Review step, ensure that the **Memory Overcommit** configuration is displayed as **Enabled**.

If **Memory Overcommit** configuration is not displayed as **Enabled**, click **Edit** to go back to the Configuration step, and select the **Enable Memory Overcommit** checkbox.

8. Click **Save**.

### What to do next

The Tasks page displays the VM creation task. After it is successfully completed, check the VMs dashboard to verify that the VM is created and the value in the **Memory Overcommit** column displays **Enabled**.

Note: The default **General** list view does not provide the **Memory Overcommit** column. Create your own customized view and add the **Memory Overcommit** column to that view. You can also add other columns to your customized view.

Click the newly created VM to open the VM details page. In the Summary tab, the Properties widget displays **Enabled** for the **Memory Overcommit** property.

Properties	
Efficiency	Good
IP Addresses	-
Description	techpubs feat doc
Cluster	<a href="#">auto_cluster_nested_6239fd0f73101a2ef03763ee</a>
Host	-
Host IP	-
vCPU	1
Memory	4 GiB
Memory Overcommit	Enabled
Power State	Off

Figure 42: VM Details Page - Memory Overcommit Enabled

### Enabling/Disabling Memory Overcommit on Existing VMs

The procedures in this section enable you to enable memory overcommit in one or more existing VMs. You need a minimum Prism Central version of pc.2022.4 for this procedure.

### Before you begin

Ensure that the VM on which you want to enable memory overcommit is powered off. If the VM is powered on or is in **Soft Shutdown** state, the **Enable Memory Overcommit** action is not available in the **Actions** dropdown list.



## Procedure

To enable memory overcommit in one or more existing VM, perform the following steps:

1. Log in to Prism Central.
2. Select the **Infrastructure** application from [Application Switcher Function](#), and navigate to **Compute & Storage > VMs** from the **Navigation Bar**. For information about the **Navigation Bar**, see [Application-Specific Navigation Bar](#).  
The system displays the page with **Lists** tab by default with all the VMs across registered clusters.
3. Select the target VM checkbox, and choose **Enable Memory Overcommit** from the actions dropdown menu..  
You can select one VM or multiple VMs at a time as a bulk update to enable memory overcommit.
4. In the **Actions** dropdown list, click the **Enable Memory Overcommit** action.

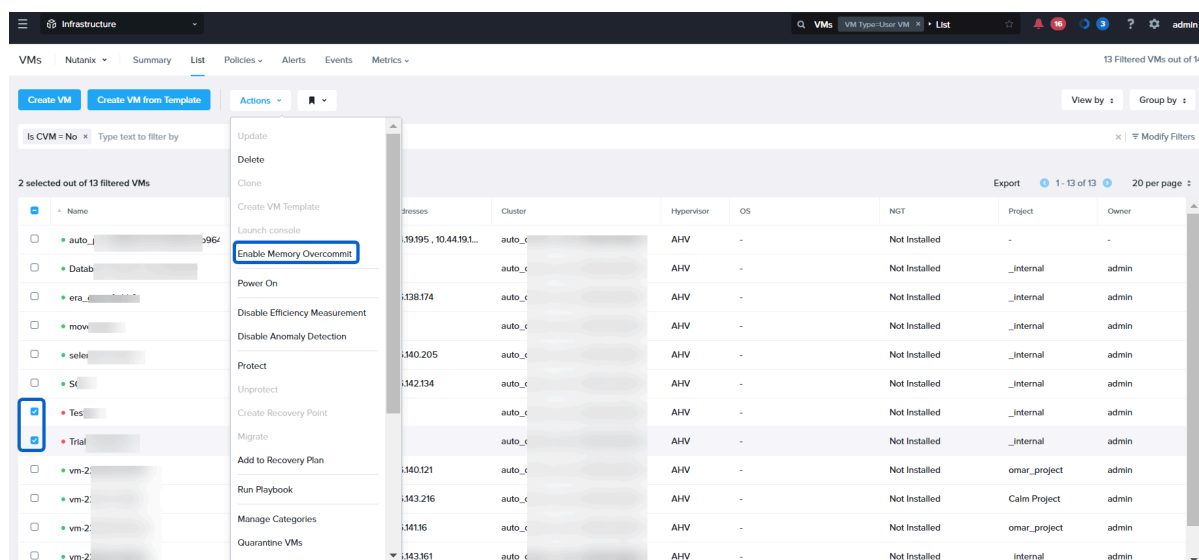


Figure 43: VM(s) Selection to Enable Memory Overcommit

Note: For an individual VM, you can also enable the memory overcommit from the update VM workflow. For more information, see [Updating a VM through Prism Central \(AHV\)](#) information in *Prism Central Infrastructure Guide*.

To disable the memory overcommit, select the target VM checkbox, and choose **Disable Memory Overcommit** from the **Actions** dropdown menu.

## What to do next

You can check the update tasks in the **Tasks** page. If you selected multiple VMs to enable or disable memory overcommit, the Task page displays the update for each VM as a separate **Update VM** task. For more information, see [Tasks View](#) information in *Prism Central Infrastructure Guide*.

You can verify the memory overcommit status as **Enabled** or **Disabled** using the following methods:



- Select **Performance** from the **View By** option in [VMs Summary View](#).
- Navigate to the [VM Details View](#) , and observe the **Memory Overcommit** field in the **Properties** widget.

### Enabling Memory Overcommit using CLI

You can configure memory overcommit on a new VM or on an existing VM after shutting it down and enabling memory overcommit on it.

### About this task

Perform the following procedure to enable memory overcommit on an existing VM after shutting it down:

### Procedure

1. Log on to the Controller VM using SSH.
2. At the CVM prompt, type **accli** to enter the acropolis CLI mode.
3. Shut down the VM to enforce an update on the VM.

```
accli> vm.shutdown vm-name
```

Replace **vm-name** with the name of the VM.

4. Update the VM to enable memory overcommit.

```
accli> vm.update vm-name memory_overcommit=True
```

Replace **vm-name** with the name of the VM.

Note: Use the wildcard \* to list out VMs of similar naming scheme and enable memory overcommit on them.

```
accli> vm.update vm-name* memory_overcommit=True
```

Use **vm-name\*** to update multiple VMs that follow identical naming scheme.

For example, Host 1 has six VMs (VM1, VM2, VM3, FileServer, SQL server, VM4), then **vm.update VM\* memory\_overcommit=True** CLI command will only update VM1, VM2, VM3, and VM4. It will not update the FileServer and SQL Server VMs on the host.

Note: Use your discretion while using the wildcard \* without mentioning the VM naming scheme. Doing so will update all the VMs in the cluster, including the VMs that you may not want to configure with memory overcommit.

5. Verify memory overcommit configuration.

The **memory\_overcommit** parameter should be set to **True**.

```
accli> vm.get vm-name
```

Replace **vm-name** with the name of the VM.

6. Start the VM.

```
accli> vm.on vm-name
```

Replace **vm-name** with the name of the VM.



## Disabling Memory Overcommit using CLI

You can disable memory overcommit on an existing VM after shutting it down and changing the memory overcommit configuration of the VM.

### About this task

Perform the following procedure to disable memory overcommit on an existing VM:

#### Procedure

1. Log on to the Controller VM using SSH.
2. At the CVM prompt, type **accli** to enter the acropolis CLI mode.
3. Shut down the VM.

```
accli> vm.shutdown vm-name
```

Replace **vm-name** with the name of the VM.

4. Update the VM to disable memory overcommit.

```
accli> vm.update vm_name memory_overcommit=False
```

Replace **vm-name** with the name of the VM.

Note: Use the wildcard \* to list out VMs of similar naming scheme and disable memory overcommit on them.

```
accli> vm.update vm-name* memory_overcommit=False
```

Use **vm-name\*** to update multiple VMs that follow identical naming scheme.

For example, Host 1 has six VMs (VM1, VM2, VM3, File Server, SQL server, VM4), then **vm.update VM\* memory\_overcommit=False** CLI command will only update VM1, VM2, VM3, and VM4. It will not update the File Server and SQL Server VMs on the host.

Note: Use your discretion while using the wildcard \* without mentioning the VM naming scheme. Doing so will update all the VMs in the cluster, including the VMs that you may not want to configure with memory overcommit.

5. Start the VM.

```
accli> vm.on vm-name
```

Replace **vm-name** with the name of the VM.

## OVA's

An Open Virtual Appliance (OVA) file is a tar archive file created by converting a virtual machine (VM) into an Open Virtualization Format (OVF) package for easy distribution and deployment. OVA helps you to quickly create, move or deploy VMs on different hypervisors.

Prism Central helps you perform the following operations with OVA's:

- Export an AHV VM as an OVA file.
- Upload OVA's of VMs or virtual appliances (vApps). You can import (upload) an OVA file with the QCOW2 or VMDK disk formats from a URL or the local machine.
- Deploy an OVA file as a VM.
- Download an OVA file to your local machine.



- Rename an OVA file.
- Delete an OVA file.
- Track or monitor the tasks associated with OVA operations in Tasks.

The access to OVA operations is based on your role. See *Role Details View* in the [Prism Central Admin Center Guide](#) to check if your role allows you to perform the OVA operations.

For information about:

- Restrictions applicable to OVA operations, see [OVA Restrictions](#) on page 228.
- The OVAs dashboard, see [OVAs View](#) in the *Prism Central Infrastructure Guide*.
- Exporting a VM as an OVA, see [Exporting a VM as an OVA](#) in the *Prism Central Infrastructure Guide*.
- Other OVA operations, see [OVA Management](#) in the *Prism Central Infrastructure Guide*.

## OVA Restrictions

You can perform the OVA operations subject to the following restrictions:

- Export to or upload OVAs with one of the following disk formats:
  - QCOW2: Default disk format auto-selected in the Export as OVA window. For more information, see [Exporting a VM as an OVA](#) information in *Prism Central Infrastructure Guide*.
  - VMDK: Deselect QCOW2 and select VMDK, if required, before you submit the VM export request when you export a VM.
  - When you export a VM or upload an OVA and the VM or OVA does not have any disks, the disk format is irrelevant.
- Upload an OVA to multiple clusters using a URL as the source for the OVA. You can upload an OVA only to a single cluster when you use the local OVA File source. For more information, see [Uploading an OVA](#) information in *Prism Central Infrastructure Guide*.
- Perform the OVA operations only with appropriate permissions. You can run the OVA operations that you have permissions for, based on your assigned user role.
- The OVA that results from exporting a VM on AHV is compatible with any AHV version 5.18 or later.
- The minimum supported versions for performing OVA operations are AOS 5.18, Prism Central 2020.8, and AHV-20190916.253.
- OVAs are not supported for vTPM and credential guard.

## VM Generation UUID and BIOS UUID Support in AHV

This section provides the information about VM Generation UUID and BIOS UUID.

The VM Generation UUID is a 64-bit unique virtual machine identifier that is used to identify the occurrence of any time shift event for the VM. The BIOS UUID is a 64-bit unique hardware identifier that helps you to identify any underlying hardware change for the VM. BIOS UUID is only relevant to the VM, and does not qualify as a VM identity for any external communication.

The AHV allocates the Generation UUID and BIOS UUID to the guest VM.



The VM Generation UUID enables you to identify any change to the VM's place in time and apply necessary safety measures to protect the application environment during VM rollback, recovery, or any operation that affects the VM's place in time.

For more information about Virtual Machine Generation Identifier, see *Virtual machine generation identifier* topic in *Microsoft Technical Documentation*.

The following table describes the scenarios in which a new Generation UUID and BIOS UUID is allocated to the guest VM or the existing Generation UUID and BIOS UUID are retained.

Scenarios	New Generation UUID
VM Creation	Yes
VM Creation using Template	
VM Creation using OVA	
VM Update	No
VM Reboot	No
VM Host Reboot	No
VM Clone from another VM	Yes
VM Clone, copy, or import from VM Snapshot	Yes
VM Pause or resume	No
High Availability (HA) restart case	No
VM Migration within a cluster	No
Recover VM from backup	Yes
VM Restore from AHV snapshot, recovery point, or Cerebro snapshots	Yes
Unplanned failover with Asynchronous Replication Schedule	Yes
Planned failover with Asynchronous Replication Schedule	No
Unplanned failover with Synchronous Replication Schedule	Yes
Planned Failover with Synchronous Replication Schedule	No
Cross-Cluster Live Migration (CCLM)	The system persists the existing Generation UUID and BIOS UUID. The recovery (remote) site supports Generation UUID and BIOS UUID.
On-Demand CCLM	
Cold Migration using Move	
Cross Hypervisor Disaster Recovery (CHDR)	

## Checking VM Generation UUID and BIOS UUID of a Guest VM

This section describes how to check the VM Generation UUID and BIOS UUID of a guest VM using `acli`.

### Before you begin

Ensure you meet the following requirements before you check the VM Generation UUID and BIOS UUID of a guest VM:

- Deploy the AOS 6.6.1 version or later at your site.



- **Power Cycle** the guest VMs if you have upgraded to AOS 6.6.1 release or later. For more information, see [Managing a VM \(AHV\)](#) on page 119.

## Procedure

To check the Generation UUID and BIOS UUID of a guest VM, perform the following steps:

1. Log on to any CVM in the cluster using SSH.
2. Retrieve the guest VM details using the following command:

```
nutanix@CVM ~ $ acli vm.get <VM_Name>
```

Observe the following log attributes to check the BIOS UUID and VM Generation UUID:

**bios\_uuid:** "cba7dc40-a45f-4823-8a8d-7b2e7119be64"

**generation\_uuid:** "0304dbc1-b1e0-452a-a104-ee2100d21c10"

## Automatic Cluster Selection for VM Placement

Automatic cluster selection in Prism Central (PC) enables you to place guest VMs on a system-selected cluster without manually selecting the cluster for guest VM placement. The system intelligently performs load balancing to identify the target cluster where sufficient resources are available for the functioning of the guest VM and places the guest VM on it.

Note: Automatic cluster selection performs load balancing only during the initial VM deployment, and Acropolis Dynamic Scheduling (ADS) then manages individual host load within the cluster. For more information on ADS, see [Acropolis Dynamic Scheduling in AHV](#) on page 8.

For more information on how the system performs the load balancing with automatic cluster selection, see [Sample Scenarios for Automatic Cluster Selection](#) on page 232.

## Supported Configuration Workflows for Automatic Cluster Selection

You can configure automatic cluster selection by using any of the following methods (workflows):

- Creating a VM from Prism Central. For more information, see [Creating a VM through Prism Central \(AHV\)](#) in *Prism Central Infrastructure Guide*.
- Deploying a VM from OVA. For more information, see [Deploying an OVA as VM](#) in *Prism Central Infrastructure Guide*.
- Creating a VM or deploying a VM from OVA using v3 API. For more information, see [Nutanix Dev](#) page.

Note: The system also supports the existing functionality of manual cluster selection while creating a VM, deploying a VM using OVA, or deploying a VM using a VM template.

## VM Policies with Automatic Cluster Selection

The automatic cluster selection function ensures that the VM is auto-placed only in one of the clusters that can satisfy the hardware requirements of the VM and the following policies that you define for VM functioning.

- Storage policies: The automatic cluster selection honors all aspects of storage policies, such as data replication (replication factor), data security (encryption), data reduction (compression), and QoS (IOPS or throughput throttling).



- Host affinity policies that you define using Prism Central, and VM-VM anti-affinity policy that you define using acli in Prism Element: For example, if the host affinity policy is configured for the VM with hosts in a single cluster or across clusters, the system considers the host affinity policy and appropriately places the VM only on the cluster that is aligned to the host affinity policy and the load balancing result.

Note: You can use the host affinity policies to restrict the VM auto-placement in the clusters that are designated for test workloads, licensed for specific workloads, in a specific location, or any other restrictions around where a workload can run.

- Image management policies: The automatic cluster selection honors the image placement policies.

For more information on VM Policies, see [Policies in Infrastructure](#) section in *Prism Central Infrastructure Guide*.

For more information on how the system considers the policies with automatic cluster selection, see [Sample Scenarios](#).

## Automatic Cluster Selection with Self-Service Projects

When the automatic cluster selection function is enabled, the system considers the cluster quota defined using Nutanix Self-Service (formerly Nutanix Calm) while placing the VMs in the cluster. The system places the VMs only in the clusters that can offer the vCPU, memory, and disk requirements to the VM within the defined quota limits.

For more information, see the [Setting Up Quota Defaults](#) section in the *Self-Service Administration and Operations Guide*.

## Requirements for Automatic Cluster Selection

Ensure that you meet the following requirements before you use automatic cluster selection:

- All the intended clusters are on AOS 6.7 or later release.
- Prism central version should be upgraded to PC 2024.1 or later release.
- All the intended clusters must be hosted with AHV hypervisor. The ESXi hypervisor and mixed hypervisor clusters are not supported.
- If you use APIs directly, use only V3 APIs to configure VM auto-cluster scheduling. For information on v3 documentation, see [Nutanix Dev](#) page.
- Only a super admin user with permission to create a VM and management privileges for all intended clusters can configure automatic cluster selection. For more information, see the [Controlling User Access \(RBAC\)](#) section in the *Security Guide*.

## Limitations of Automatic Cluster Selection

The following limitations apply to automatic cluster selection:

- While placing a VM based on an existing image with automatic cluster selection, the system always places the VM on one of the registered clusters where the VM image is present.
- Automatic cluster selection configuration for VM deployments using VM templates is not supported.



- Automatic cluster selection configuration for VMs with GPU specifications or advanced processor compatibility settings is not supported.

For information on advanced processor compatibility, see [Advanced Processor Compatibility in AHV](#) on page 233.

- When you configure automatic cluster selection from Prism Central, you can select overlay subnets only. However, if you use API to configure automatic cluster selection, the system supports both VLAN and overlay subnet.

## Sample Scenarios for Automatic Cluster Selection

### Assumption

A sample Prism Central deployment with the following specifications:

- Total number of clusters registered with Prism Central: 4
- Cluster Specifications:

Cluster Name	Cluster A	Cluster B	Cluster C
Number of Nodes	5	3	1
Aggregate Memory Capacity	600 GB	753 GB	62 GB
Aggregate CPU capacity	100	100	100
Storage	RF3 capable	RF2 capable	-
The total capacity of all clusters is: 600 + 753 + 62 + 450 = 1865 GB			

- Image placement policies with multiple images placed across clusters
- Host affinity policies with hosts across clusters

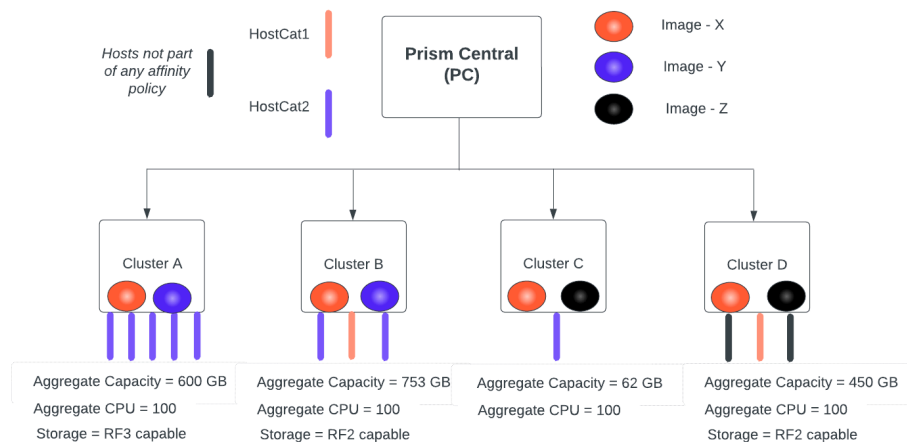


Figure 44: Sample Prism Central Deployment

### Scenario 1

Placement of VMs when Image placement, host affinity, and storage policies are defined for the VM.



Create a VM		Configured Policies at the Site		
Image Used	Configured Host Category for Host Affinity	Image placement	Host Affinity	Storage
X	HostCat1	X image in Cluster A, Cluster B, Cluster C, and Cluster D	HostCat1 with hosts used across Cluster B and Cluster D	RF2 config
X	HostCat2			
		Y image in Cluster A and Cluster B	HostCat2 with hosts used across Cluster A, Cluster B, and Cluster C	
Y	HostCat1	Z image in Cluster C and Cluster D		
Y	HostCat2			
Z	HostCat1			
Z	HostCat2			

## Scenario 2

Load balancing with automatic cluster selection with image placement policies consideration.

Total VMs created using X image	Image Placement Policy Configured	VM approximate Distribution Count	
		Cluster A	Cluster B
100	X image in Cluster A, Cluster B, Cluster C, and Cluster D	32	41

## Advanced Processor Compatibility in AHV

Acropolis, by default, automatically assigns a CPU generation to all the guest VMs. The automatically assigned CPU generation represents the highest subset of CPU features available on all nodes in the PE cluster.

Note: The automatically assigned CPU generation name might change when you add a node from a different CPU generation to a cluster. A power cycle may be required to ensure VM mobility for ADS and maintenance scenarios. The system provides a notification when a power cycle is required for individual VMs after the cluster expansion operation.

You can migrate a guest VM to a destination cluster only if the CPU capabilities of the guest VM are present in the destination cluster.

AHV provides an advanced processor compatibility feature that enables you to customize the CPU capabilities for the guest VMs. You can associate the required CPU capabilities (CPU generation name) to the guest VM from the supported CPU capabilities presented by the cluster and can enhance the ability of the VM to migrate between nodes or clusters with different CPU generations. For example, you can successfully migrate the guest VMs from the source cluster with higher CPU capabilities to the destination cluster with lower CPU capabilities.

This feature also provides the following additional benefits:



- Provides a distinct set of CPU features to any guest VM, irrespective of changes in the AHV version. This prevents the guest VM from being exposed to advanced CPU features with which it might not function correctly.
- Associate a distinct CPU family with a guest VM. This helps you to configure guest VMs as requiring specific CPU capabilities requirements. For example, the guest VM might require AES-NI to mitigate timing-based attacks against cryptographic algorithms or to configure a guest VM with Windows 11 to boot on Skylake CPUs only.
- Reduces the number of guest VMs in the cluster that require a power cycle when you add a node with a different CPU generation to the cluster during cluster expansion, which changes the supported CPU generations of the cluster.

The guest VMs with advanced processor compatibility enabled might be configured to use a CPU generation that is supported on the expanded cluster and, hence, does not require a restart after cluster expansion. However, the guest VMs that use the automatically assigned CPU generation might require a restart.

You can select the CPU generation based on the considerations required at your site:

- Select the oldest CPU generation to maximize the cross-cluster live migration (CCLM) capability of the guest VM. For example, you can select Sandybridge on Intel and Naples on AMD processors.
- Select the latest CPU generation to maximize the performance of the guest VM with less consideration for cross-cluster live migration (CCLM) capability.

Note:

- By default, Acropolis selects the CPU generation, which is compatible with a wide range of hardware when you enable advanced processor compatibility for the guest VM. For more information, see [Creating a VM through Prism Central \(AHV\)](#) or [Updating a VM through Prism Central \(AHV\)](#) in the *Prism Central Infrastructure Guide*.
- Acropolis only allows the selection of a CPU generation, which is supported by all nodes in the cluster.
- The choice of CPU generation with advanced processor compatibility varies between CPU vendors. For example, with AMD CPU, the system provides you the option to select either Naples or Rome, but with Intel CPU, the system provides you the option to select Sandybridge, Ivybridge, Haswell, Broadwell, Skylake, Cascadelake, or Icelake. For more information, refer to the CPU manufacturer documentation.
- The list of CPU generation names available when you enable advanced processor compatibility, depends on the cluster (PE registered with Prism Central) you select to host the guest VM during the VM creation workflow. If the cluster that you select uses Intel CPU, the options for CPU generation name selection show only the supported Intel CPU generation names.
- For successful cross-cluster VM migration, the destination cluster must have all the CPU capabilities that are available with the guest VMs in the source cluster after advanced processor compatibility is applied to them.
- The CPU generation that you directly associate with the guest VM using advanced processor compatibility persists after the VM power cycle, VM migration, or VM restart in a high-availability setup. Acropolis does not update the CPU generation associated with the guest VM, even if the destination cluster supports a higher CPU generation level.

Using Prism Central, you can change the automatically assigned CPU family (CPU generation name) to any of the cluster-supported CPU families. For more information, see [Supported Configuration Workflows for Advanced Processor Compatibility](#) on page 236.

If advanced processor compatibility is not enabled, and no CPU generation name is associated with the guest VM, the Acropolis performs the following actions:

- Automatically selects the most recent CPU generation supported on the cluster each time the VM is powered on and provides those CPU capabilities to the guest VM to maximize the system performance.
- Permits live migration of the guest VM to any cluster that can support the automatically selected CPU generation.

For information on the supported CPU families and the CPU capabilities exposed to the guest, see [Supported CPU Generation Names for Advanced Processor Compatibility](#) on page 236.

## Requirements for Advanced Processor Compatibility

Ensure that you meet the following prerequisites to enable Advanced processor compatibility:

- Upgrade the cluster to AOS 6.8 or later release.
- Host the cluster with AHV and upgrade to AHV 20230302.100173 or later release.
- Prism central version should be upgraded to PC 2024.1 or later release.

## Limitations and Considerations of Advanced Processor Compatibility

The following limitations apply to advanced processor compatibility:

- Only AHV clusters are supported.
- During Disaster Recovery, the following behavior applies when advanced processor compatibility is enabled for the guest VM:
  - For the guest VM that is protected with an asynchronous or nearsync replication schedule, the advanced processor compatibility configuration of guest VMs is not restored from the recovery point on the destination cluster, and the guest VM can start on the destination cluster without the advanced processor compatibility configuration.
  - For the guest VM that is protected with a synchronous replication schedule, the advanced processor compatibility configuration of the guest VM is replicated to the destination cluster, and the Acropolis permits the live migration only at the destination cluster that supports the directly assigned or automatically assigned CPU generation of the running guest VM.

Note: In case of unplanned failover of the guest VMs replicated using synchronous replication, the guest VM might fail to start at the destination cluster if the destination cluster does not support the CPU capabilities (CPU generation name) associated with the guest VM. However, you can manually edit the advanced processor compatibility configuration of the guest VMs at the destination cluster according to the CPU capabilities of the destination cluster to start the guest VM.

Nutanix recommends that you reduce the selected CPU generation to one supported by both source and destination clusters to avoid interruptions during unplanned failover of VMs using advanced processor compatibility.

- If you manually create a recovery point for the guest VM that has advanced processor compatibility enabled and attempt to either restore the guest VM or create a new VM using



the same recovery point, the advanced processor compatibility configuration of the guest VM is not restored. To re-enable the advanced processor compatibility configuration for the guest VM, you must shut down the guest VM, reconfigure the advanced processor compatibility configuration, and start the guest VM again. For more information, see the [Creating Recovery Points Manually \(Out-of-Band Snapshots\)](#) section in the *Nutanix Disaster Recovery Guide*.

- When you trigger an on-demand cross-cluster live migration for the guest VM with advanced processor compatibility enabled, the Acropolis permits the live migration only at the destination cluster that supports the directly assigned or automatically assigned CPU generation of the running guest VM.
- You can migrate the guest VMs only to the clusters that can provide all features of the CPU generation name associated with the guest VMs through advanced processor compatibility.
- You cannot enable the advanced processor compatibility functionality for guest VMs configured to use CPU pass-through.
- You cannot configure CPU pass-through for the guest VMs for which the advanced processor compatibility functionality is enabled.
- Advanced processor compatibility is not currently enabled for use with VM Templates or with automatic cluster selection.

For information on VM templates, see [VM Template Management](#) section in *Prism Central Infrastructure Guide*.

For information on automatic cluster selection, see [Automatic Cluster Selection for VM Placement](#) on page 230.

## Supported Configuration Workflows for Advanced Processor Compatibility

You can configure advanced processor compatibility for a VM using the following workflows:

- Creating a VM from Prism Central. For more information, see [Creating a VM through Prism Central \(AHV\)](#) in *Prism Central Infrastructure Guide*.
- Updating a VM from Prism Central. For more information, see [Updating a VM through Prism Central \(AHV\)](#) in *Prism Central Infrastructure Guide*.

Note: You must shut down the guest VM to configure advanced processor compatibility.

## Supported CPU Generation Names for Advanced Processor Compatibility

This section provides information about the supported CPU generation names and their capabilities for advanced processor compatibility.

### Intel CPU

The following table provides information about the supported Intel CPU generation name and their capabilities for advanced processor compatibility.



CPU Generation Name	Available Features
Intel Sandy Bridge	<p>This CPU generation exposes the following capabilities:</p> <ul style="list-style-type: none"> <li>• Base features of Intel Sandy Bridge CPUs, including AVX and SSE4.2.</li> <li>• Support of additional nested virtualization using Intel VMX for virtualization enabled.</li> <li>• Hyper-V enlightenments for Microsoft Windows VMs that include RUNTIME, SPINLOCK, TLBFLUSH, and IPI.</li> </ul> <p>Note: This CPU generation is the maximum Intel CPU generation supported by AHV.</p>
Intel Ivy Bridge	<p>This CPU generation exposes the following capabilities:</p> <ul style="list-style-type: none"> <li>• All CPU features from Intel Sandy Bridge plus additional CPU features including FSGSBASE, and ERMS.</li> <li>• Additional CPU features for nested virtualization that include VMX-PL2, RDRAND-EXIT.</li> </ul>
Intel Haswell	<p>This CPU generation exposes the following capabilities:</p> <ul style="list-style-type: none"> <li>• All CPU features from Intel Ivy Bridge plus additional CPU features including BMI1, FMA, and MOVBE.</li> <li>• Additional CPU features for nested virtualization that include VMWRITE-VMEXIT-FIELDS, and SHADOW-VMCS.</li> </ul>
Intel Broadwell	<p>This CPU generation exposes the following capabilities:</p> <ul style="list-style-type: none"> <li>• All CPU features from Intel Haswell plus additional CPU features including AVX512, and MD_CLEAR, SSBD.</li> <li>• Additional CPU features for nested virtualization that include VMX-PL3.</li> </ul> <p>Note: This is the default Intel CPU generation when no CPU generation is specified. It provides CPU features sufficient to run most guests (For Windows guests, this requires a very wide range of hardware).</p>
Intel Skylake	<p>This CPU generation exposes the following capabilities:</p> <ul style="list-style-type: none"> <li>• All CPU features from Intel Broadwell plus additional CPU features including XSAVEC, PKU, CLWB, PDPE1GB, AVX512BW, AVX512VL, AVX512VBQ.</li> <li>• Additional CPU features for nested virtualization that include VMX-PL4.</li> </ul> <p>Note: This CPU generation is the minimum Intel CPU generation supported by AHV.</p>
Intel Cascade Lake	<p>This CPU generation exposes all CPU features from Intel Skylake including AVX512VBQ, CAPABILITIES, SKIP-LIDFL-VMENTRY, IBRS-ALL, MDS-NO, RDSEED.</p>

CPU Generation Name	Available Features
Intel Ice Lake	<p>This CPU generation exposes the following capabilities:</p> <ul style="list-style-type: none"> <li>• All CPU features from Intel Cascade Lake plus additional CPU features including UMIP, TAA-NO, LA57, AVX512BITALG, GFNI, PSCHANGE-MC-N, AVX512VBMI, SHA-NI, AVX512IFMA, RDPID, and FSRM.</li> <li>• Additional CPU features for nested virtualization that include P</li> </ul>

## AMD CPU

The following table provides information about the supported AMD CPU generation names and their capabilities for advanced processor compatibility.

CPU Generation Name	Available Features
AMD Naples	<p>This CPU generation exposes the following capabilities:</p> <ul style="list-style-type: none"> <li>• Base features of AMD Naples CPUs that include AVX, SSE, and</li> <li>• Additional nested virtualization using AMD SVM for VMs with</li> <li>• Hyper-V enlightenments for Microsoft Windows VMs that include RUNTIME, SPINLOCK, TLBFLUSH, IPI</li> </ul> <p>Note: This is the default AMD CPU generation when no CPU generation is specified. It provides the minimum set of CPU features sufficient to run most guests (For Windows 11, see <a href="#">AMD documentation</a> for a wide range of hardware).</p>
AMD Rome	<p>This CPU generation exposes all CPU features from AMD Naples including CLZERO, UMIP, PERFCTR_CORE, AMD-STIBP, IBPB, WBNOINVD</p> <p>Note: This CPU generation is the minimum AMD CPU generation</p>



# VM TEMPLATE MANAGEMENT

---

In Prism Central, you can create VM templates to manage the golden image of a VM. A VM template can be considered as a master copy of a virtual machine. It captures the virtual machine configuration and the contents of the VM including the guest operating system, and the applications installed on the VM. You can use this template to deploy multiple VMs across clusters.

Note: You can create or manage a VM template only as an admin user.

## Limitations of VM Template Feature

The current implementation of the VM template feature has the following limitations:

- You cannot create a VM template if any of the following conditions is applicable at your site:
  - VM is not on AHV.
  - VM is an agent or a PC VM.
  - Any volume group is attached to the VM.
  - VM is undergoing vDisk migration.
  - VM has disks located on RF1 containers.
  - VM is protected by PD-based DR.
- VM templates do not copy the following attributes from the source VMs:
  - Host affinity attributes
  - HA priority attributes
  - Nutanix Guest Tools (NGT) installation
  - Quality of service (QoS) configuration

You must reconfigure the above-listed attributes at the deployed VMs.

- VM Template does not protect the underlying VM recovery point from deletion by a user having delete permission. VM Template deployment fails if you delete the associated VM recovery point.

## Creating a VM Template

This section describes how to create a VM template in Prism Central.

### About this task

To create a VM template, perform the following steps:

### Procedure

1. Log in to Prism Central.
2. Select the **Infrastructure** application from [Application Switcher Function](#), and navigate to **Compute & Storage > VMs** from the **Navigation Bar**. For information about the **Navigation**



**Bar**, see [Application-specific Navigation Bar](#) information in the *Prism Central Infrastructure Guide*.

The system displays the **List** tab as default

3. In List tab, select the VM to create a VM template.

Note: Before you select a VM to create a VM template, ensure that the VM is powered off.

4. In the **Actions** dropdown menu, select **Create VM Template**.

You can also click **Create VM Template** in the **Summary** page of an individual VM. For information about how to access the **Summary** page of an individual VM, see [VM Details View](#) information in *Prism Central Infrastructure Guide*.

The system displays the Create template from VM window.

Create Template from VM

VM Template Name

win10-template

Description

Guest Customization

Specify guest OS customization during VM Deployment

Sysprep (Windows) Custom Script

Upload Script

Upload or paste script here

Allow users to override at VM Deployment ? ☐ Yes

Next

Figure 45: Create Template from VM window



5. In the **Create Template from VM** window, enter the following information:

- a. **Name** - Name of the VM Template.
- b. **Description** - Description for the VM template. Description is an optional field.
- c. For the **Guest Customization** fields, select the following options for guest operating system (OS) for the VMs that you deploy using this VM template:

- In the **Script Type** field, select **Sysprep (Windows)** to customize the Windows OS, and **Cloud-init (Linux)** to customize the Linux OS.

Note: If you select **No Customization** at the time of creating the template and allow the users to override the guest customization settings using **Allow users to override at VM Deployment?** toggle field, it gives the maximum customization control to the users. In this case, the users can customize the script type and the configuration method.

- In the **Configuration Method** field, for each of these script types selected in **Script Type** field, select either upload a custom script or opt for a guided setup in the field.

Note:

- If you select **Custom Script**, you can either upload a script to customize the guest OS of the VMs, or you can copy-paste the script in the text box.
- If you select **Guided Script**, enter the following information:
  - **Authentication Type**: Select one of the radio button to set the authentication type:
    - **Password**: Set a username and password for the user who uses this template to deploy the VM.
    - **SSH Key** [Cloudinit (Linux) only]: Set the SSH key for the user who use this template to deploy the VM.
  - **Locale**: Select the locale (language) from the dropdown list.
  - **Hostname**: Enter the hostname for the user who uses this template to deploy a VM
  - **License Key**: Enter the license key.

Important: The information that you enter is used to customize the OS of the VMs that are deployed using this template.

Note: If you opt for **Guest Customization** setup with script (custom or guided script), ensure that the script is in a valid format. The system does not validate the guest customization scripts. In the script is not in valid format, the VM deployment might succeed



but the guest customization script may not work as expected. You can observe the discrepancies (if any) only after the VM gets deployed.

- d. Set the **Allow users to override at VM Deployment?** toggle field as per your requirement. This toggle field is used to enable or disable the override permission for guest customization settings.

- If this toggle field is:
  - *Enabled*: The system allows the users (who use this VM template to deploy VM) to modify the guest OS customization settings in VM template. The users can modify the settings only for the **Configuration Method** field. For example, the users can change the authentication information at the time of deploying a VM from VM template, or they can change from a guided setup to a custom script.
  - *Disabled*: The settings that are already provided in the VM template, can be used for VM deployment.

6. Click **Next**.

On the next page, you can review the template details.

The system prompts you to review the configuration details, resource details, network details, and management details.

7. Click **Save** to save the inputs and create a VM template.

The new template appears in the **Templates** page list.

Note: Once you create a VM template, the template metadata is available in Prism Central. The template data is stored as a VM recovery point and is co-located with the source VM. If you use the VM template to deploy a VM to another cluster, the recovery point is copied to the destination cluster before deployment.

## Deploying VM from a Template

### Before you begin

Ensure that the VM template is available. For information about how to create a VM template, see [Creating a VM Template](#) on page 239.

### About this task

After you create a VM template, you can use the VM template to deploy any number of VMs across clusters.

To deploy a VM using a VM template, perform the following steps:

### Procedure

1. Log in to Prism Central.
2. Select the **Infrastructure** application from [Application Switcher Function](#), and navigate to **Compute & Storage > Templates** from the **Navigation Bar**. For information about the **Navigation Bar**, see [Application-specific Navigation Bar](#) information in *Prism Central Infrastructure Guide*.



3. Select the target VM template to deploy VMs using either of the following methods:

- In the **Templates** page, select the target template checkbox, and click **Deploy VMs**. For more information about **Templates** page, see [VM Template Summary View](#) information in *Prism Central Infrastructure Guide*.
- In the **Summary** page of an individual VM template, click **Deploy VMs**. For more information about **Summary** page of an individual VM template, see [VM Template Details View](#) information in *Prism Central Infrastructure Guide*.
- In the **Versions** page of an individual VM template, select the target VM template version, and click **Deploy VMs**. For more information about **Versions** page of an individual VM template, see [VM Template Details View](#) information in *Prism Central Infrastructure Guide*.

Note: In the **Versions** page, you can select any active or non-active VM template version for VM deployment.

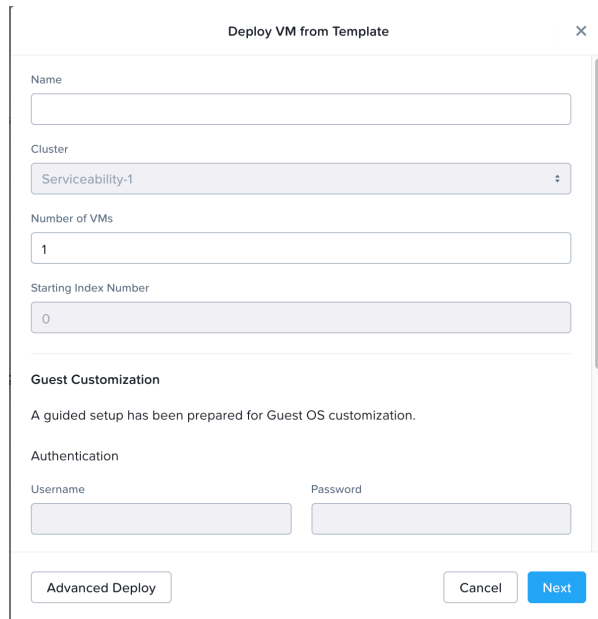
Note: By default, the active version of the VM template is used for VM deployment.

The **Deploy VM from Template** window appears. By default, you see a **Quick Deploy** method. You can click **Advanced Deploy** to access the **Advanced Deploy** method in Deploy VM from Template window. In **Advanced Deploy** method, you can view and modify some VM properties and network settings.

4. Deploy VM using either of the following deployment methods:

- **Quick Deploy** method:

There is an example showing the **Quick Deploy** method:



The screenshot shows a dialog box titled "Deploy VM from Template" with a close button (X) in the top right corner. The dialog contains several input fields and sections:

- Name:** A text input field.
- Cluster:** A dropdown menu showing "Serviceability-1".
- Number of VMs:** A text input field with the value "1".
- Starting Index Number:** A text input field with the value "0".
- Guest Customization:** A section with the text "A guided setup has been prepared for Guest OS customization."
- Authentication:** A section with two input fields: "Username" and "Password".
- Buttons:** At the bottom, there are three buttons: "Advanced Deploy", "Cancel", and "Next".

Figure 46: Deploy VM from Template - Quick Deploy Method

1. Enter the following information:

- **Name:** Enter a name for the VM.
- **Cluster:** Select the cluster where you want to deploy the VM.
- **Number of VMs:** Enter the number of VMs that you want to deploy.
- **Starting Index Number:** Enter the starting index number for the VMs when you are deploying multiple VMs simultaneously. These index numbers are used in the VM names. For example, if you are deploying two VMs and specify the starting index number as 5, the VMs are named as `vm_name-5` and `vm_name-6`.
- **Guest Customization:** The template can have any one of the following options for guest OS customization:
  - **No Customization**
  - **Sysprep (Windows), or Cloud-init (Linux).** For **Sysprep (Windows)**, or **Cloud-init (Linux)**, you can choose to either upload a custom script or opt for a guided setup.

Note: These fields are enabled for modification only if the VM template allows you to override its guest customization settings while deploying the VM. The **Allow users to override at VM Deployment?** toggle field is used to enable or disable the override for guest customization settings. For information about how to set this toggle field during VM template creation, see [Creating a VM Template](#) on page 239. If the override permission for the guest OS

customization is disabled, the settings that are already provided in the template can be used for VM deployment.

2. Click **Next** to verify the configuration details of the VMs to be deployed.
3. Click **Deploy** to deploy the VMs.

- **Advanced Deploy** method:

The following is an example showing the **Advanced Deploy** method:

The screenshot shows the 'Deploy VM from Template' wizard with the 'Configuration' tab selected. The wizard has four steps: 1. Configuration, 2. Resources, 3. Management, and 4. Review. The Configuration tab contains the following fields: Name (text input), Description (text input with '(Optional)' placeholder), Cluster (dropdown menu showing 'auto\_cluster\_nested\_61afc38057f2f30dec6788bc'), Number of VMs (text input with '1'), and Starting Index Number (text input with '0'). Below these fields is the 'VM Properties' section, which includes three columns: CPU (with a text input '1' and a dropdown 'vCPU'), Cores Per CPU (with a text input '1' and a dropdown 'Cores'), and Memory (with a text input '2' and a dropdown 'GB'). There is also a checkbox labeled 'Enable Memory Overcommit' which is currently unchecked. At the bottom of the form are three buttons: 'Back to Quick Deploy', 'Cancel', and 'Next'.

Figure 47: Deploy VM from Template - Advanced Deploy Method

1. Enter the following information:

- **Configuration:** Provide inputs for name and description (optional) of the VM, cluster where you want to deploy the VM, Number of VMs to be deployed, and starting index number (only if deploying multiple VMs). In this tab, you can also view and modify the VM properties such as CPU, core per CPU, and memory.
- **Resources:** Review the configuration settings for the VM resources such as disks, networks, and boot configuration. Here, you can modify the network settings but cannot modify any other settings.
- **Management:** The fields in this tab are enabled for modification only if the VM template allows you to override its guest customization settings while deploying the VM. The **Allow users to override at VM Deployment?** toggle field is used to enable or disable the override for guest customization settings. For information about how to set this toggle field during VM template creation, see [Creating a VM Template](#) on

page 239. If the override permission for the guest OS customization is disabled, the settings that are already provided in the template can be used for VM deployment.

If guest customization has been enabled, provide inputs for authentication type, username, password, locale, and Hostname.

Note: Hostname of the VM is automatically generated based on the VM names that you provide. If you are deploying a single VM, you can override the automatic generation of the hostname by specifying a hostname. If you are deploying multiple VMs, you cannot override the automatic hostname generation.

- **Review:** Review the information displayed in this tab.
2. Click **Deploy** to deploy the VMs.

## Managing a VM Template

### About this task

After you create a VM template, you can perform the following actions to manage the VM Template:

- Update the guest OS of the source VM specified in the VM template
- Complete guest OS update
- Cancel guest OS update
- Update the configuration of the template to create a new VM template version
- Delete the VM template.

For information about how to create a VM template, see [Creating a VM Template](#) on page 239.

To manage a VM template, perform the following steps:

### Procedure

1. Log in to Prism Central.
2. Select the **Infrastructure** application from [Application Switcher Function](#), and navigate to **Compute & Storage > Templates** from the **Navigation Bar**. For information about the **Navigation Bar**, see [Application-specific Navigation Bar](#) information in *Prism Central Infrastructure Guide*.

Select the target VM template using either of the following methods:

- Select the target VM template checkbox in the **Templates** page. For more information about **Templates** page, see [VM Template Summary View](#) information in *Prism Central Infrastructure Guide*.
  - Click the target template to view the **Summary** page of an individual VM template. For more information about **Summary** page of an individual VM template, see [VM Template Details View](#) information in *Prism Central Infrastructure Guide*.
3. Select the required action from the **Actions** dropdown menu:



Note: The available actions appear in bold and the unavailable actions are greyed out. The available actions depend on the current state of the template and user permissions.

The following actions are available in **Actions** drop down menu:

- **Update Guest OS**

To update the guest OS in a VM Template, perform the following steps:

1. Select the target VM template version on which the guest OS is to be updated in **Select a version to Update** window, and click **Proceed**.

Note: By default the active version of the VM Template is selected for VM Template update, however the system provides you an option to select the VM Template version on which the guest OS is to be updated.

2. Review the information displayed in the **Update Guest OS** window, and click **Proceed**.

At this stage, the system deploys a temporary VM from the selected VM template version, and provides you an option to access the VM. You can also access the new VM from VMs

**List** tab. For information about VMs **List** tab, see [VMs Summary View](#) information in *Prism Central Infrastructure Guide*.

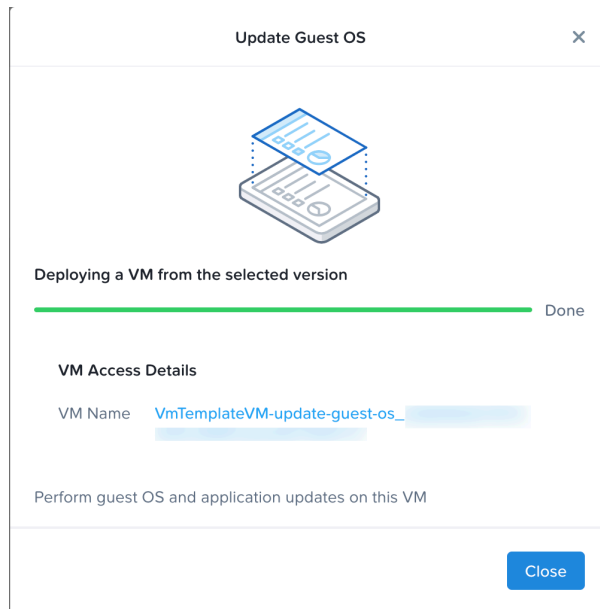


Figure 48: Update Guest OS

3. Start the temporary VM, log on to the VM, and update the guest OS of the temporary VM. For information on how to update a VM, see [Managing a VM through Prism Central \(AHV\)](#) information in *Prism Central Infrastructure Guide*.
  4. Complete the guest OS update using the **Complete Guest OS Update** option from the **Actions** dropdown menu.
- **Complete Guest OS Update** to complete the process initiated for guest OS update. You must select this option only after successful update of the guest OS of the temporary VM.

Note: At this stage, the system prompts you to create a new version of the VM Template with updated guest OS. Specify the details for the new version, and click **Complete Update**.

The temporary VM automatically gets deleted after completion of the guest OS upgrade process.

- **Cancel Guest OS Update** to cancel the process initiated for guest OS update.  
The temporary VM automatically gets deleted after cancellation of the guest OS upgrade process.
- **Update Configuration** to modify the VM template configuration. In Select a version to Update window, select the VM template version that you want to modify, and create a version on top of it.

Perform the following steps to update the VM template configuration in Update Template Configuration window:



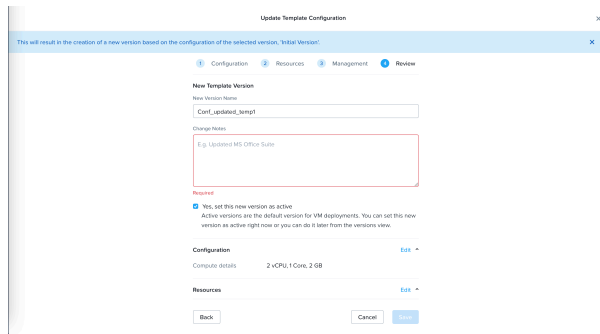


Figure 49: Update Template Configuration

1. In the **Configuration** step, view the name of the base version that you want to update, change notes for that version, cluster name, VM properties (CPU, cores per CPU), and memory). In this section, you can modify only VM properties.
2. In the **Resources** step, view the information about disks, networks, and boot configuration. In this section, you can modify only network resources.
3. In the **Management** step, modify the guest customization settings.
4. In the **Review** step, review and modify the configuration settings that you are allowed to modify. You must provide a name and change notes for the new version. You can also choose to set this new version as active version.

Note: An active version is the version of the template gets deployed by default when you click **Deploy VMs** after VM template configuration update.

5. Click **Save** to save the settings and create a new VM template version.
- **Delete Template** to delete a template. The system prompts you to confirm the delete action. Click **OK** to delete the VM template.
4. To only manage a VM template version:
    - a. Select the VM template version from the **Versions** page. For information about how to access the **Versions** page of an individual VM template, see [VM Template Details View](#) information in *Prism Central Infrastructure Guide*.
    - b. Select either of the following options from the **Actions** dropdown menu:
      - **Set as Active** to make the selected VM template version as the active version.
      - **Delete** to delete the selected VM template version.

Note: You cannot delete the active version of a VM template.

# COPYRIGHT

---

Copyright 2024 Nutanix, Inc.

Nutanix, Inc.

1740 Technology Drive, Suite 150

San Jose, CA 95110

All rights reserved. This product is protected by U.S. and international copyright and intellectual property laws. Nutanix and the Nutanix logo are registered trademarks of Nutanix, Inc. in the United States and/or other jurisdictions. All other brand and product names mentioned herein are for identification purposes only and may be trademarks of their respective holders.

