



# NetApp HCI Solutions

## HCI

NetApp  
November 04, 2020

This PDF was generated from <https://docs.netapp.com/us-en/hci-solutions/index.html> on November 04, 2020.  
Always check [docs.netapp.com](https://docs.netapp.com) for the latest.



# Table of Contents

NetApp HCI Solutions: Sales and Partner Kits .....	1
Where to find NetApp HCI Kits .....	1
End User Computing Partner Kits .....	1
Private Cloud Partner Kits .....	1
NetApp HCI TV .....	2
Additional information (login required):.....	2
Private Cloud .....	3
End User Computing (EUC).....	4
TR-4861: Hybrid Cloud VDI with Virtual Desktop Service .....	4
End User Computing on NetApp HCI with VMware .....	37
TR-4854: NetApp HCI for Citrix Virtual Apps and Desktops with Citrix Hypervisor .....	38
Infrastructure .....	75
NVA-1148: NetApp HCI with Red Hat Virtualization .....	75
TR-4857: NetApp HCI with Cisco ACI .....	160
Containers and DevOps.....	206
NVA-1141: NetApp HCI with Anthos, Design and Deployment .....	206
NVA-1149: NetApp HCI for Red Hat OpenShift on Red Hat Virtualization .....	285
Database .....	324
Data Fabric.....	325
Data Protection .....	326
TR-4830: NetApp HCI Disaster Recovery with Cleondris .....	326
Artificial Intelligence (AI).....	345
NVA-1144: NetApp HCI AI Inferencing at the Edge Data Center with H615c and NVIDIA T4 .....	345
Security .....	403

# NetApp HCI Solutions: Sales and Partner Kits

NetApp Sales and Partner Kits are rich with information to help you sell more effectively.

Kits for End User Computing (EUC) and Private Cloud show how NetApp HCI solutions deliver differentiation and also describe the use cases addressed by our solutions.

## Where to find NetApp HCI Kits

- EUC and Private Cloud Sales Kits in the [HCS Resource Center](#)
- Partner Kits in the [Partner Portal](#) or use the Partner Kit matrix (below) for rapid access to partner enablement materials

## End User Computing Partner Kits

	Citrix	VMware Horizon
<b>Solution Brief</b>	<a href="#">VDI Citrix</a>	<a href="#">VDI VMware</a>
<b>Customer Presentation</b>	<a href="#">Digital Workplace Transformation (VDI/EUC)</a>	
<b>Technical Presentation</b>	<a href="#">Technical Presentation for EUC Opportunity</a>	
<b>Quick Reference Guide (QRG)</b>	<a href="#">EUC &amp; VDI Solutions with NetApp HCI Partner Marketing</a>	
<b>FAQ</b>	<a href="#">NetApp HCI FAQ for EUC-VDI</a>	
<b>Competitive Resources</b>	<a href="#">Competitive Sales Kit EUC</a>	

## Private Cloud Partner Kits

	VMware	Red Hat
<b>Solution Brief</b>	<a href="#">What is a Private Cloud</a>	
<b>Customer Presentation</b>	<a href="#">BDM Customer Presentation: VMware Private Cloud</a>	<a href="#">RedHat Private Cloud On NetApp HCI</a>
<b>Technical Presentation</b>	<a href="#">Build Your VMware Private Cloud</a>	<a href="#">Private Cloud with Red Hat/KVM</a>
<b>Quick Reference Guide (QRG)</b>	<a href="#">Private Cloud Solutions</a>	
<b>FAQ</b>	<a href="#">FAQ: VMware Private Cloud</a>	
<b>Competitive Resources</b>	<a href="#">Competitive Sales Kit VMware Private Cloud</a>	

# NetApp HCI TV



- Ideas for your customer presentation of EUC and VDI solutions with NetApp HCI  
[Introduction to NetApp HCI EUC Solution](#) (3:24)
- Ideas for your customer presentation of NetApp HCI Private Cloud.  
[Introduction to NetApp HCI Private Cloud](#) (2:05)
- Animated introductions to NetApp HCI differentiation  
[The value for solutions with NetApp disaggregated HCI](#) (2:09)  
[NetApp HCI solution ROI value by eliminating the HCI tax](#) (2:56)

## Additional information (login required):

- [NetApp HCI Solutions Collection](#)
- [NetApp HCI VMware Private Cloud Collection](#)
- [NetApp HCI Red Hat Private Cloud Collection](#)
- [NetApp HCI Red Hat OpenShift Container Platform Collection](#)
- [NetApp HCI End User Computing \(EUC\) Collection](#)
- [NetApp HCI Database Collection](#)
- [NetApp HCI Data Protection Collection](#)

# **Private Cloud**

# End User Computing (EUC)

## TR-4861: Hybrid Cloud VDI with Virtual Desktop Service

Suresh Thoppay, NetApp

The NetApp Virtual Desktop Service (VDS) orchestrates Remote Desktop Services (RDS) in major public clouds as well as on private clouds. VDS supports Windows Virtual Desktop (WVD) on Microsoft Azure. VDS automates many tasks that must be performed after deployment of WVD or RDS, including setting up SMB file shares (for user profiles, shared data, and the user home drive), enabling Windows features, application and agent installation, firewall, and policies, and so on.

Users consume VDS for dedicated desktops, shared desktops, and remote applications. VDS provides scripted events for automating application management for desktops and reduces the number of images to manage.

VDS provides a single management portal for handling deployments across public and private cloud environments.

### Customer Value

The remote workforce explosion of 2020 has changed requirements for business continuity. IT departments are faced with new challenges to rapidly provision virtual desktops and thus require provisioning agility, remote management, and the TCO advantages of a hybrid cloud that makes it easy to provision on-premises and cloud resources. They need a hybrid-cloud solution that:

- Addresses the post-COVID workspace reality to enable flexible work models with global dynamics
- Enables shift work by simplifying and accelerating the deployment of work environments for all employees, from task workers to power users
- Mobilizes your workforce by providing rich, secure VDI resources regardless of the physical location
- Simplifies hybrid-cloud deployment
- Automates and simplifies risk reduction management

[Next: Use Cases](#)

### Use Cases

Hybrid VDI with NetApp VDS allows service providers and enterprise virtual desktop administrators to easily expand resources to other cloud environment without affecting their users. Having on-premises resources on NetApp HCI provides better control of GPU resources and allows you to expand compute or

storage nodes based on demand.

This solution applies to the following use cases:

- Bursting into the cloud for surges in demand for remote desktops and applications
- Reducing TCO for long running remote desktops and applications by hosting them on-premises with flash storage and GPU resources
- Ease of management of remote desktops and applications across cloud environments
- Experience remote desktops and applications by using a software-as-a-service model with on-premises resources

#### Target Audience

The target audience for the solution includes the following groups:

- EUC/VDI architects who want to understand the requirements for a hybrid VDS
- NetApp partners who would like to assist customers with their remote desktop and application needs
- Existing NetApp HCI customers who want to address remote desktop and application demands

[Next: NetApp Virtual Desktop Service Overview](#)

## NetApp Virtual Desktop Service Overview

NetApp offers many cloud services, including the rapid provisioning of virtual desktop with WVD or Remote Applications, including rapid integration with Azure NetApp Files.

Traditionally, it takes weeks to provision and deliver remote desktop services to customers. Apart from provisioning, it can be difficult to manage applications, user profiles, shared data, group policy objects to enforce policies. Firewall rules can increase complexity and require a separate skillset and tools.

With Microsoft Azure Windows Virtual Desktop service, Microsoft takes care of maintenance for Remote Desktop Services components, allowing customers to focus on provisioning workspaces in the cloud. Customers must provision and manage the complete stack which requires special skills to manage VDI environments.

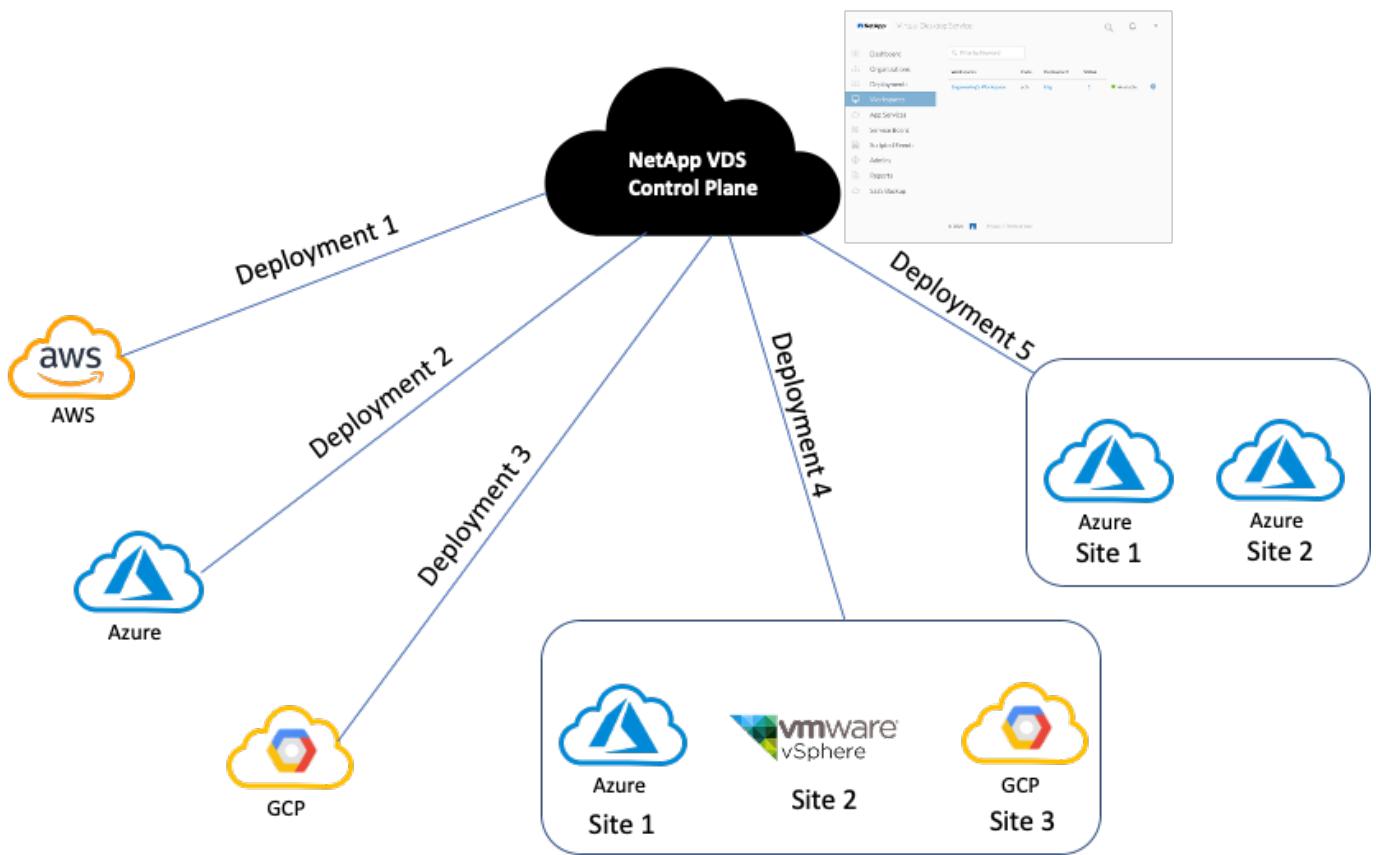
With NetApp VDS, customers can rapidly deploy virtual desktops without worrying about where to install the architecture components like brokers, gateways, agents, and so on. Customers who require complete control of their environment can work with a professional services team to achieve their goals. Customers consume VDS as a service and thus can focus on their key business challenges.

NetApp VDS is a software-as-a-service offering for centrally managing multiple deployments across

AWS, Azure, GCP, or private cloud environments. Microsoft Windows Virtual Desktop is available only on Microsoft Azure. NetApp VDS orchestrates Microsoft Remote Desktop Services in other environments.

Microsoft offers multisession on Windows 10 exclusively for Windows Virtual Desktop environments on Azure. Authentication and identity are handled by the virtual desktop technology; WVD requires Azure Active Directory synced (with AD Connect) to Active Directory and session VMs joined to Active Directory. RDS requires Active Directory for user identity and authentication and VM domain join/management.

A sample deployment topology is shown in the following figure.

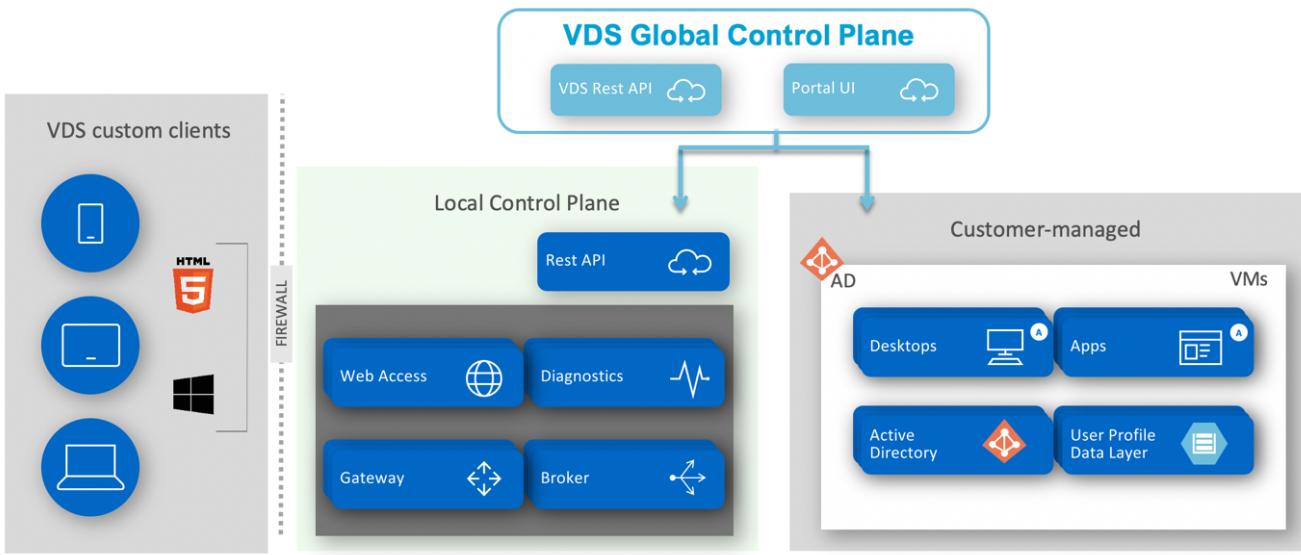


Each deployment is associated with an active directory domain and provides clients with an access entry point for workspaces and applications. A service provider or enterprise that has multiple active directory domains typically has more deployments. A single Active Directory domain that spans multiple regions typically has a single deployment with multiple sites.

For WVD in Azure, Microsoft provides a platform-as-a-service that is consumed by NetApp VDS. For other environments, NetApp VDS orchestrates the deployment and configuration of Microsoft Remote Desktop Services. NetApp VDS supports both WVD Classic and WVD ARM and can also be used to upgrade existing versions.

Each deployment has its own platform services, which consists of Cloud Workspace Manager (REST API endpoint), an HTML 5 Gateway (connect to VMs from a VDS management portal), RDS Gateways

(Access point for clients), and a Domain Controller. The following figure depicts the VDS Control Plane architecture for RDS implementation.



For RDS implementations, NetApp VDS can be readily accessed from Windows and browsers using client software that can be customized to include customer logo and images. Based on user credentials, it provides user access to approved workspaces and applications. There is no need to configure the gateway details.

The following figure shows the NetApp VDS client.



## Virtual Desktop Service

Username

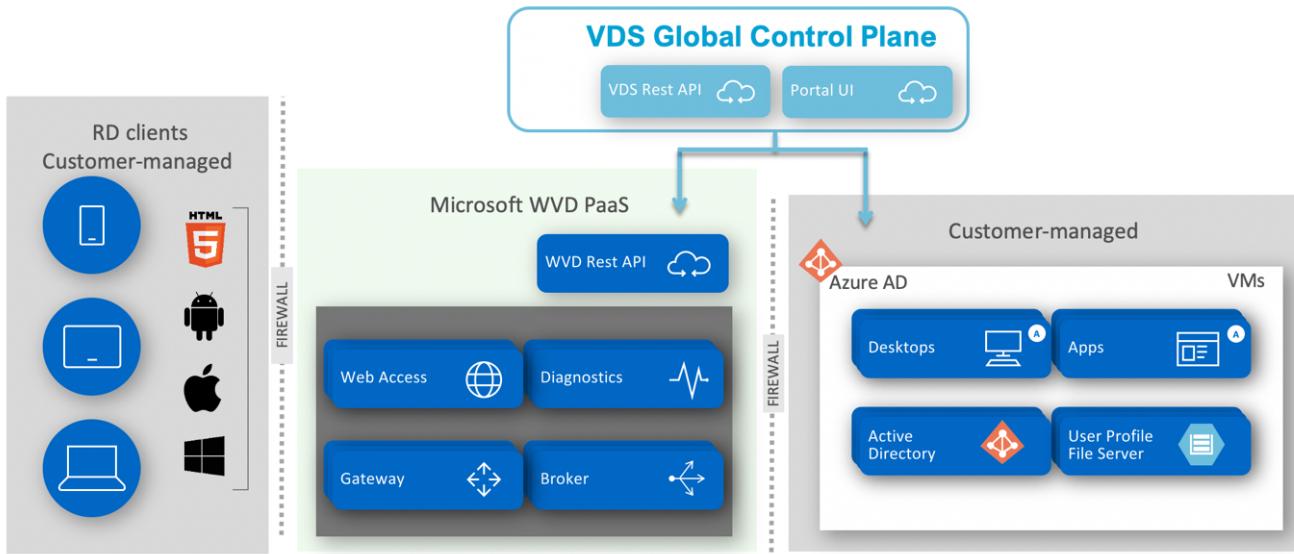
Password

Save Username

[Workspace](#)[Applications](#)

In the Azure WVD implementation, Microsoft handles the access entry point for the clients and can be consumed by Microsoft WVD client available natively for various OS. It can also be accessed from a web-based portal. The configuration of client software must be handled by the Group Policy Object (GPO) or in other ways preferred by customers.

The following figure depicts the VDS Control Plane architecture for Azure WVD implementations.



In addition to the deployment and configuration of required components, NetApp VDS also handles user management, application management, resource scaling, and optimization.

NetApp VDS can create users or grant existing user accounts access to cloud workspace or application services. The portal can also be used for password resets and the delegation of administrating a subset of components. Helpdesk administrators or Level-3 technicians can shadow user sessions for troubleshooting or connect to servers from within the portal.

NetApp VDS can use image templates that you create, or it can use existing ones from the marketplace for cloud-based provisioning. To reduce the number of images to manage, you can use a base image, and any additional applications that you require can be provisioned using the provided framework to include any command-line tools like Chocolatey, MSIX app attach, PowerShell, and so on. Even custom scripts can be used as part of machine lifecycle events.

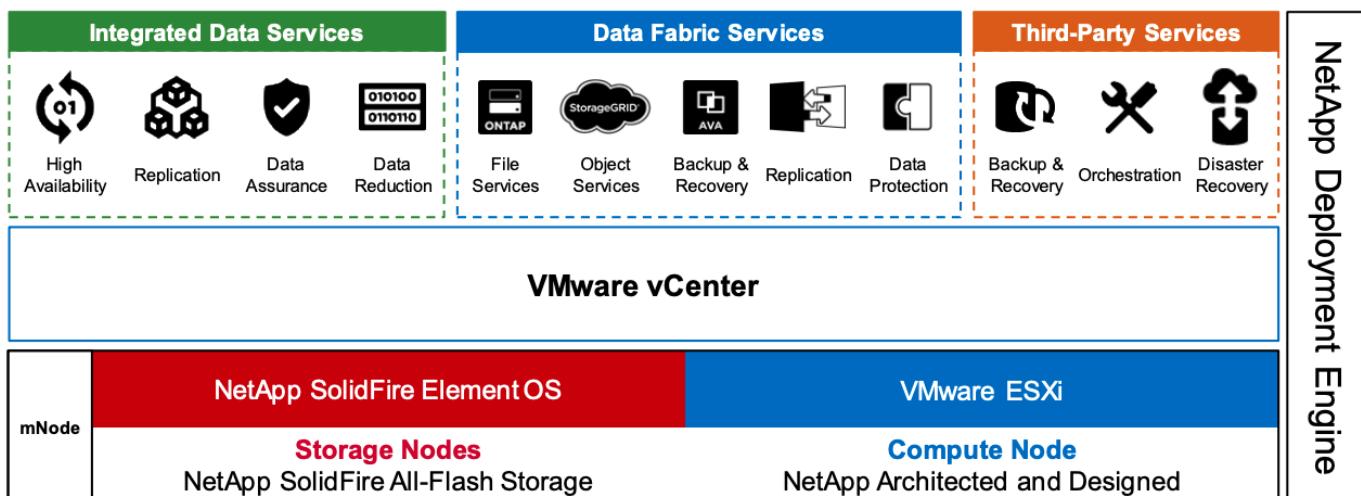
[Next: NetApp HCI Overview](#)

## NetApp HCI Overview

NetApp HCI is a hybrid cloud infrastructure that consists of a mix of storage nodes and compute nodes. It is available as either a two-rack unit or single-rack unit, depending on the model. The installation and configuration required to deploy VMs are automated with the NetApp Deployment Engine (NDE). Compute clusters are managed with VMware vCenter, and storage clusters are managed with the vCenter Plug-in deployed with NDE. A management VM called the mNode is deployed as part of the NDE.

NetApp HCI handles the following functions:

- Version upgrades
  - Pushing events to vCenter
  - vCenter Plug-In management
  - A VPN tunnel for support
  - The NetApp Active IQ collector
  - The extension of NetApp Cloud Services to on-premises, enabling a hybrid cloud infrastructure.
- The following figure depicts HCI components.



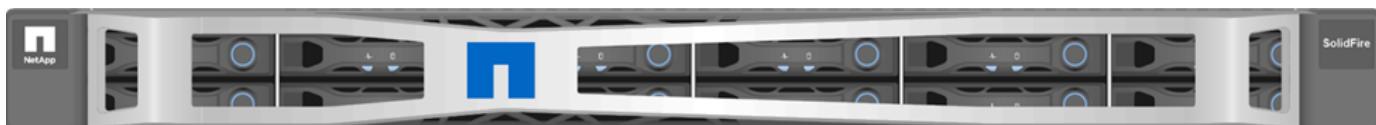
## Storage Nodes

Storage nodes are available as either a half-width or full-width rack unit. A minimum of four storage nodes is required at first, and a cluster can expand to up to 40 nodes. A storage cluster can be shared across multiple compute clusters. All the storage nodes contain a cache controller to improve write performance. A single node provides either 50K or 100K IOPS at a 4K block size.

NetApp HCI storage nodes run NetApp Element software, which provides minimum, maximum, and burst QoS limits. The storage cluster supports a mix of storage nodes, although one storage node cannot exceed one-third of total capacity.

## Compute Nodes

Compute nodes are available in half-width, full-width, and two rack-unit sizes. The NetApp HCI H410C and H610C are based on scalable Intel Skylake processors. The H615C is based on second-generation scalable Intel Cascade Lake processors. There are two compute models that contain GPUs: the H610C contains two NVIDIA M10 cards and the H615C contains three NVIDIA T4 cards.



The NVIDIA T4 has 40 RT cores that provide the computation power needed to deliver real-time ray

tracing. The same server model used by designers and engineers can now also be used by artists to create photorealistic imagery that features light bouncing off surfaces just as it would in real life. This RTX-capable GPU produces real-time ray tracing performance of up to five Giga Rays per second. The NVIDIA T4, when combined with Quadro Virtual Data Center Workstation (Quadro vDWS) software, enables artists to create photorealistic designs with accurate shadows, reflections, and refractions on any device from any location.

Tensor cores enable you to run deep learning inferencing workloads. When running these workloads, an NVIDIA T4 powered with Quadro vDWS can perform up to 25 times faster than a VM driven by a CPU-only server. A NetApp H615C with three NVIDIA T4 cards in one rack unit is an ideal solution for graphics and compute-intensive workloads.

The following figure lists NVIDIA GPU cards and compares their features.

NVIDIA GPUs Recommended for Virtualization				Available on NetApp HCI H615C	Available on NetApp HCI H610C	P6
	V100S	RTX 8000	RTX 6000	T4	M10	
GPU	1 NVIDIA Volta	1 NVIDIA Turing	1 NVIDIA Turing	1 NVIDIA Turing	4 NVIDIA Maxwell	1 NVIDIA Pascal
CUDA Cores	5,120	4,608	4,608	2,560	2,560 (640 per GPU)	2,048
Tensor Cores	640	576	576	320	—	—
RT Cores	—	72	72	40	—	—
Guaranteed QoS (GPU Scheduler)	✓	✓	✓	✓	—	✓
Live Migration	✓	✓	✓	✓	✓	✓
Multi-vGPU	✓	✓	✓	✓	✓	✓
Memory Size	32/16 GB HBM2	48 GB GDDR6	24 GB GDDR6	16 GB GDDR6	32 GB GDDR5 (8 GB per GPU)	16 GB GDDR5
vGPU Profiles	1 GB, 2 GB, 4 GB, 8 GB, 16 GB, 32 GB	1 GB, 2 GB, 3 GB, 4 GB, 6 GB, 8 GB, 12 GB, 16 GB, 24 GB, 48 GB	1 GB, 2 GB, 3 GB, 4 GB, 6 GB, 8 GB, 12 GB, 24 GB	1 GB, 2 GB, 4 GB, 8 GB, 16 GB	0.5 GB, 1 GB, 2 GB, 4 GB, 8 GB	1 GB, 2 GB, 4 GB, 8 GB, 16 GB
Form Factor	PCIe 3.0 dual slot and SXM2	PCIe 3.0 dual slot	PCIe 3.0 dual slot	PCIe 3.0 single slot	PCIe 3.0 dual slot	MXM (blade servers)
Power	250 W / 300 W (SXM2)	250 W	250 W	70 W	225 W	90 W
Thermal	passive	passive	passive	passive	passive	bare board
vGPU Software Support	Quadro vDWS, GRID vPC, GRID vApps, vComputeServer	Quadro vDWS, GRID vPC, GRID vApps, vComputeServer	Quadro vDWS, GRID vPC, GRID vApps, vComputeServer	Quadro vDWS, GRID vPC, GRID vApps, vComputeServer	Quadro vDWS, GRID vPC, GRID vApps	Quadro vDWS, GRID vPC, GRID vApps, vComputeServer
Use Case	Ultra-high-end rendering, simulation, 3D design with Quadro vDWS; ideal upgrade path for V100	High-end rendering, 3D design and creative workflows with Quadro vDWS	Mid-range to high-end rendering, 3D design and creative workflows with Quadro vDWS	Entry-level to highend 3D design and engineering workflows with Quadro vDWS. High-density, low power GPU acceleration for knowledge workers with NVIDIA GRID software.	Knowledge workers using modern productivity apps and Windows 10 requiring best density and total cost of ownership (TCO), multimonitor support with NVIDIA GRID vPC/vApps	For customers requiring GPUs in a blade server form factor; ideal upgrade path for M6

The M10 GPU remains the best TCO solution for knowledge-worker use cases. However, the T4 makes a great alternative when IT wants to standardize on a GPU that can be used across multiple use cases, such as virtual workstations, graphics performance, real-time interactive rendering, and inferencing. With the T4, IT can take advantage of the same GPU resources to run mixed workloads—for example, running VDI during the day and repurposing the resources to run compute workloads at night.

The H610C compute node is two rack units in size; the H615C is one rack unit in size and consumes less power. The H615C supports H.264 and H.265 (High Efficiency Video Coding [HEVC]) 4:4:4 encoding and decoding. It also supports a VP9 decoder, which is becoming more mainstream; even the WebM container package served by YouTube uses the VP9 codec for video.

The number of nodes in a compute cluster is dictated by VMware; currently, it is 96 with VMware vSphere 7.0 Update 1. Mixing different models of compute nodes in a cluster is supported when

Enhanced vMotion Compatibility (EVC) is enabled.

[Next: NVIDIA Licensing](#)

## NVIDIA Licensing

When using an H610C or H615C, the license for the GPU must be procured from NVIDIA partners that are authorized to resell the licenses. You can find NVIDIA partners with the [partner locator](#). Search for competencies such as virtual GPU (vGPU) or Tesla.

NVIDIA vGPU software is available in four editions:

- NVIDIA GRID Virtual PC (GRID vPC)
- NVIDIA GRID Virtual Applications (GRID vApps)
- NVIDIA Quadro Virtual Data Center Workstation (Quadro vDWS)
- NVIDIA Virtual ComputeServer (vComputeServer)

### GRID Virtual PC

This product is ideal for users who want a virtual desktop that provides a great user experience for Microsoft Windows applications, browsers, high-definition video, and multi-monitor support. The NVIDIA GRID Virtual PC delivers a native experience in a virtual environment, allowing you to run all your PC applications at full performance.

### GRID Virtual Applications

GRID vApps are for organizations deploying a Remote Desktop Session Host (RDSH) or other app-streaming or session-based solutions. Designed to deliver Microsoft Windows applications at full performance, Windows Server-hosted RDSH desktops are also supported by GRID vApps.

### Quadro Virtual Data Center Workstation

This edition is ideal for mainstream and high-end designers who use powerful 3D content creation applications like Dassault CATIA, SOLIDWORKS, 3Dexcite, Siemens NX, PTC Creo, Schlumberger Petrel, or Autodesk Maya. NVIDIA Quadro vDWS allows users to access their professional graphics applications with full features and performance anywhere on any device.

### NVIDIA Virtual ComputeServer

Many organizations run compute-intensive server workloads such as artificial intelligence (AI), deep learning (DL), and data science. For these use cases, NVIDIA vComputeServer software virtualizes the NVIDIA GPU, which accelerates compute-intensive server workloads with features such as error correction code, page retirement, peer-to-peer over NVLink, and multi-vGPU.



A Quadro vDWS license enables you to use GRID vPC and NVIDIA vComputeServer.

[Next: Deployment](#)

## Deployment

NetApp VDS can be deployed to Microsoft Azure using a setup App available based on the required codebase. The current release is available at <https://cwasetup.cloudworkspace.com> and the preview release of the upcoming product is available at <https://preview.cwasetup.cloudworkspace.com>.

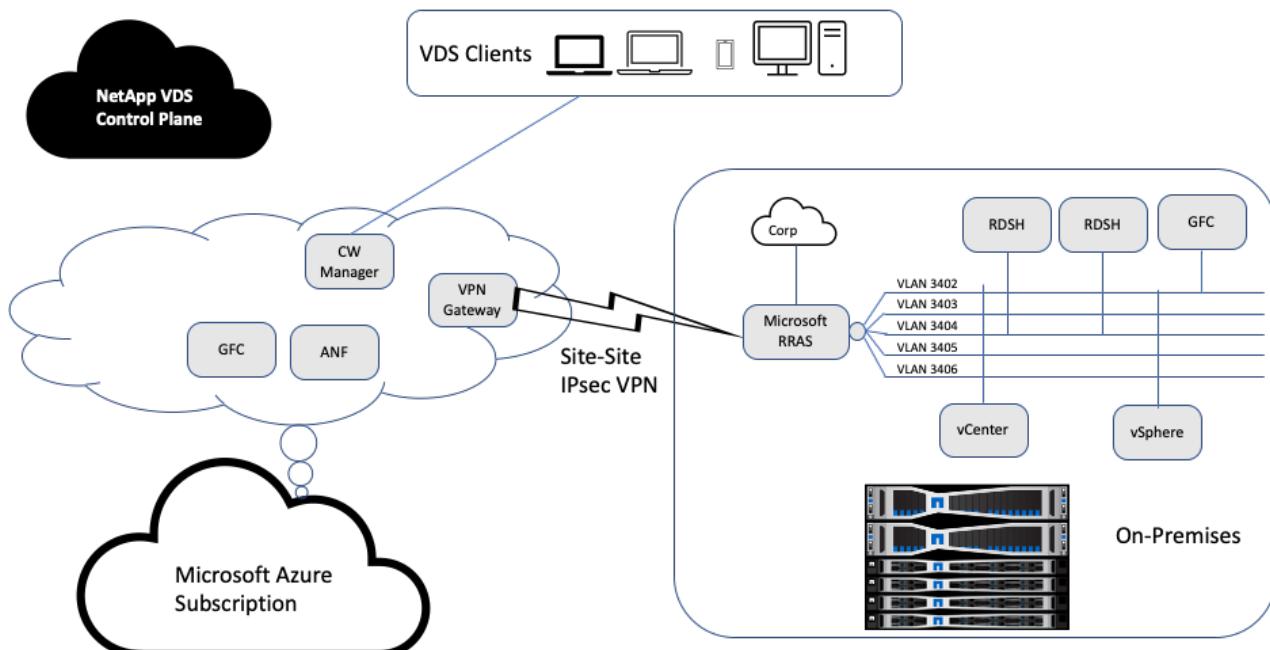
See [this video](#) for deployment instructions.

[Next: Hybrid Cloud Environment](#)

## Hybrid Cloud Environment

NetApp Virtual Desktop Service can be extended to on-premises when connectivity exists between on-premises resources and cloud resources. Enterprises can establish the link to Microsoft Azure using Express Route or a site-to-site IPsec VPN connection. You can also create links to other clouds in a similar way either using a dedicated link or with an IPsec VPN tunnel.

For the solution validation, we used the environment depicted in the following figure.



On-premises, we had multiple VLANs for management, remote-desktop-session hosts, and so on. They were on the 172.21.146-150.0/24 subnet and routed to the corporate network using the Microsoft Remote Routing Access Service. We also performed the following tasks:

1. We noted the public IP of the Microsoft Routing and Remote Access Server (RRAS; identified with IPchicken.com).
2. We created a Virtual Network Gateway resource (route-based VPN) on Azure Subscription.
3. We created the connection providing the local network gateway address for the public IP of the Microsoft RRAS server.
4. We completed VPN configuration on RRAS to create a virtual interface using pre-shared authentication that was provided while creating the VPN gateway. If configured correctly, the VPN should be in the connected state. Instead of Microsoft RRAS, you can also use pfSense or other relevant tools to create the site-to-site IPsec VPN tunnel. Since it is route-based, the tunnel redirects traffic based on the specific subnets configured.

Microsoft Azure Active Directory provides identity authentication based on oAuth. Enterprise client authentications typically require NTLM or Kerberos-based authentication. Microsoft Azure Active Directory Domain Services perform password hash sync between Azure Active Directory and on-prem domain controllers using ADConnect.

For this Hybrid VDS solution validation, we initially deployed to Microsoft Azure and added an additional site with vSphere. The advantage with this approach is that platform services were deployed to Microsoft Azure and were then readily backed up using the portal. Services can then be easily accessed from anywhere, even if the site-site VPN link is down.

To add another site, we used a tool called DCConfig. The shortcut to that application is available on the desktop of the cloud workspace manager (CWMgr) VM. After this application is launched, navigate to the DataCenter Sites tab, add the new datacenter site, and fill in the required info as shown below. The URL points to the vCenter IP. Make sure that the CWMgr VM can communicate with vCenter before adding the configuration.



Make sure that vSphere PowerCLI 5.1 on CloudWorkspace manager is installed to enable communication with VMware vSphere environment.

The following figure depicts on-premises datacenter site configuration.

**Configuration**

DataCenter	Accounts	Email	DatabaseConnection	Exclude	DataCenter Sites	Product Keys	Static IpAddress	Drive Mapping
------------	----------	-------	--------------------	---------	------------------	--------------	------------------	---------------

[Add New DataCenter Site](#)

	DataCenter Site	Type	Is Primary	DataCenter Site Detail	
	Site 1	AzureRM	<input checked="" type="checkbox"/>		<a href="#">Edit</a>
▶	Site 2	vSphere	<input type="checkbox"/>		<a href="#">Edit</a>

To delete DataCenter Site(s), Select it and right click to delete

**DataCenter Site**

DataCenter Site	Site 2	<a href="#">Cancel Edit</a>	<a href="#">Save</a>
Hypervisor	vSphere	<a href="#">Load Hypervisor</a>	<a href="#">Test</a>

**General Settings**

Local VM Account		Hypervisor Account	
Username	Administrator	Username	Administrator@vsphere
Password	*****	Password	*****
URL <input type="text" value="https://172.21.146.150/sdk/"/>			
Vm Name Prefix	<input type="text"/>	Is Primary Hypervisor?	<input checked="" type="radio"/> Yes <input type="radio"/> No
Max Concurrent	<input type="text" value="20"/>	Must Set IpAddress Of VM:	<input checked="" type="radio"/> Yes <input type="radio"/> No
Create Server	<input type="text"/>	Subnet Mask	<input type="text" value="255.255.255.0"/>
Default Gateway	<input type="text" value="172.21.148.250"/>		

**DNS**

Primary DNS:	<input type="text" value="10.67.78.11"/>
Secondary DNS:	<input type="text"/>
Set DNS Address:	<input checked="" type="radio"/> Yes <input type="radio"/> No

**vSphere**

Data Center	<input type="text" value="NetApp-HCI-Datacenter"/>
Cluster	<input type="text"/>
Resource Pool	<input type="text"/>
Host Name	<input type="text"/>
VM Folder	<input type="text" value="VDS"/>
Max VMs In Datastore	<input type="text" value="-1"/>
Min HD Free Space In Datastore GB	<input type="text" value="-1"/>
Min Ram Free GB	<input type="text" value="-1"/>
<a href="#">Exclude VSphere DataStore</a>	
<a href="#">Exclude VSphere ResourcePools</a>	

Note that there are filtering options available for compute resource based on the specific cluster, host name, or free RAM space. Filtering options for storage resource includes the minimum free space on datastores or the maximum VMs per datastore. Datastores can be excluded using regular expressions. Click Save button to save the configuration.

To validate the configuration, click the Test button or click Load Hypervisor and check any dropdown under the vSphere section. It should be populated with appropriate values. It is a best practice to keep the primary hypervisor set to yes for the default provisioning site.

The VM templates created on VMware vSphere are consumed as provisioning collections on VDS. Provisioning collections come in two forms: shared and VDI. The shared provisioning collection type is used for remote desktop services for which a single resource policy is applied to all servers. The VDI type is used for WVD instances for which the resource policy is individually assigned. The servers in a provisioning collection can be assigned one of the following three roles:

- **TSDATA.** Combination of Terminal Services and Data server role.
- **TS.** Terminal Services (Session Host).
- **DATA.** File Server or Database Server. When you define the server role, you must pick the VM

template and storage (datastore). The datastore chosen can be restricted to a specific datastore or you can use the least-used option in which the datastore is chosen based on data usage.

Each deployment has VM resource defaults for the cloud resource allocation based on Active Users, Fixed, Server Load, or User Count.

[Next: Single Server Load Test with Login VSI](#)

## Single server load test with Login VSI

The NetApp Virtual Desktop Service uses the Microsoft Remote Desktop Protocol to access virtual desktop sessions and applications, and the Login VSI tool determines the maximum number of users that can be hosted on a specific server model. Login VSI simulates user login at specific intervals and performs user operations like opening documents, reading and composing mails, working with Excel and PowerPoint, printing documents, compressing files, and taking random breaks. It then measures response times. User response time is low when server utilization is low and increases when more user sessions are added. Login VSI determines the baseline based on initial user login sessions and it reports the maximum user session when the user response exceeds 2 seconds from the baseline.

The following table contains the hardware used for this validation.

Model	Count	Description
NetApp HCI H610C	4	Three in a cluster for launchers, AD, DHCP, and so on. One server for load testing.
NetApp HCI H615C	1	2x24C Intel Xeon Gold 6282 @2.1GHz. 1.5TB RAM.

The following table contains the software used for this validation.

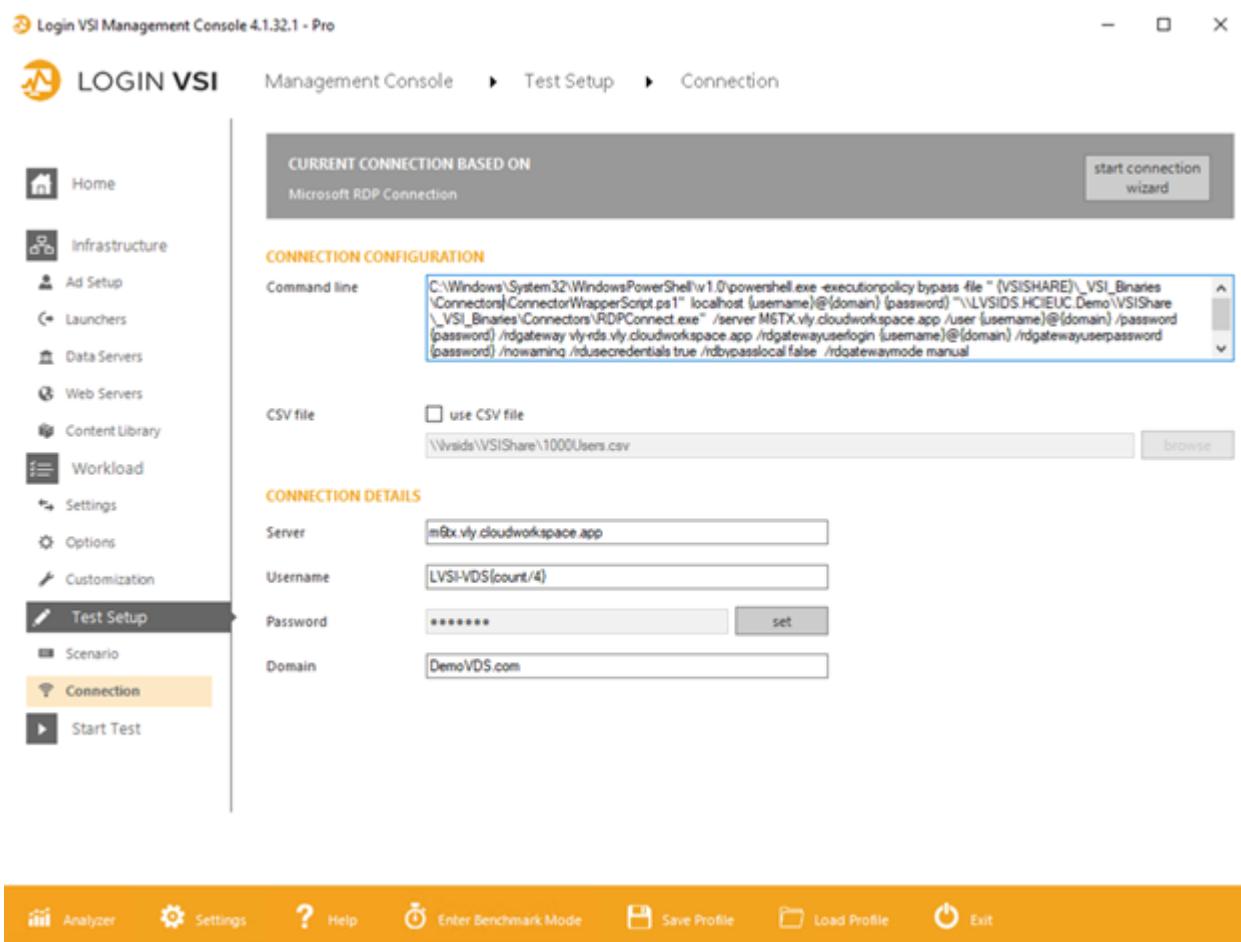
product	Description
NetApp VDS 5.4	Orchestration
VM Template Windows 2019 1809	Server OS for RDSH
Login VSI	4.1.32.1
VMware vSphere 6.7 Update 3	Hypervisor
VMware vCenter 6.7 Update 3f	VMware management tool

The Login VSI test results are as follows:

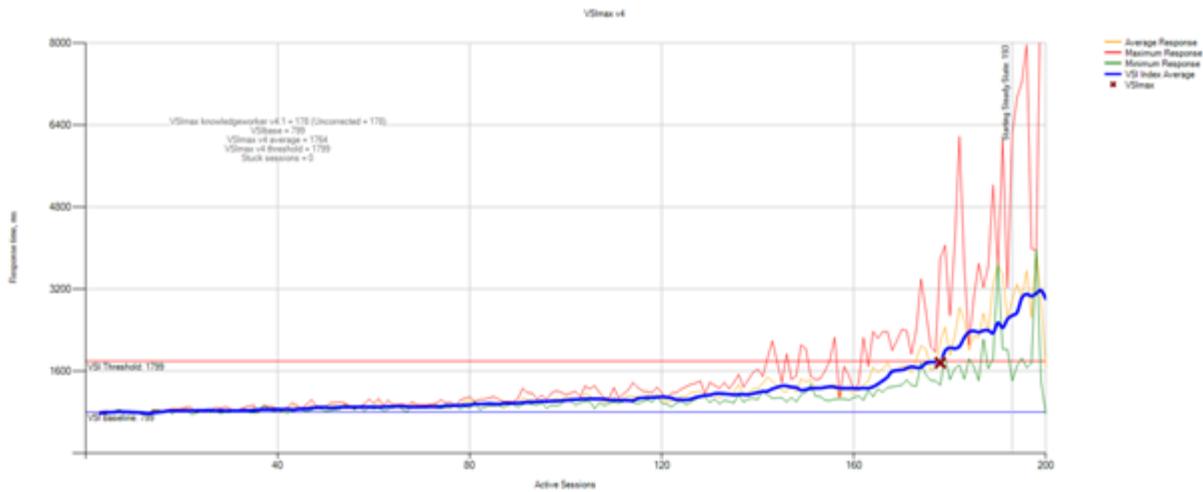
Model	VM configuration	Login VSI baseline	Login VSI Max
H610C	8 vCPU, 48GB RAM, 75GB disk, 8Q vGPU profile	799	178
H615C	12 vCPU, 128GB RAM, 75GB disk	763	272

Considering sub-NUMA boundaries and hyperthreading, the eight VMs chosen for VM testing and configuration depended on the cores available on the host.

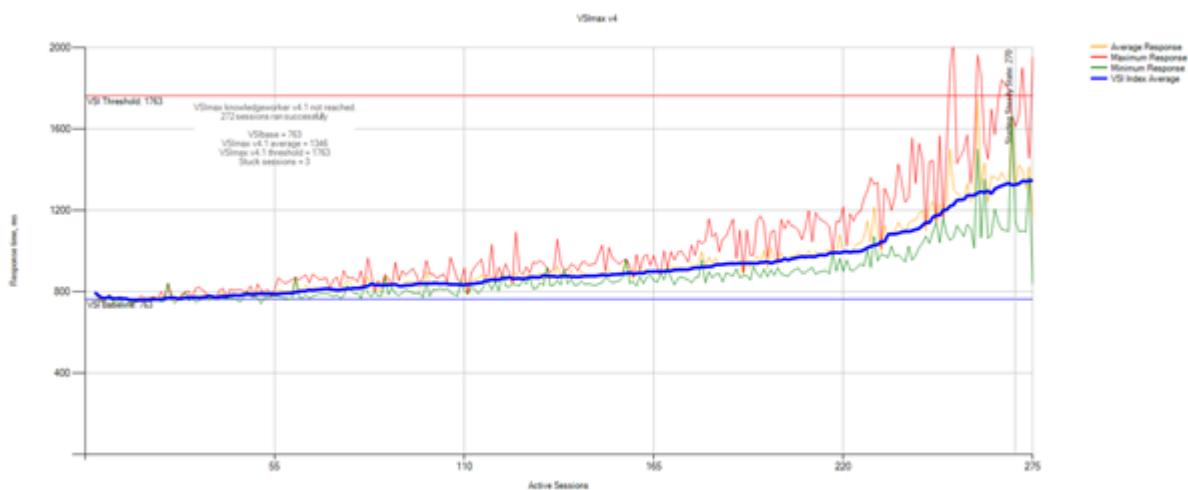
We used 10 launcher VMs on the H610C, which used the RDP protocol to connect to the user session. The following figure depicts the Login VSI connection information.



The following figure displays the Login VSI response time versus the active sessions for the H610C.



The following figure displays the Login VSI response time versus the active sessions for the H615C.



The performance metrics from Cloud Insights during H615C Login VSI testing for vSphere host and VMs is shown in teh following figure.



[Next: Management Portal](#)

## Management Portal

NetApp VDS Cloud Workspace Management Suite portal is available [here](#) and the upcoming version is available [here](#).

The portal allows centralized management for various VDS deployments including one that has sites defined for on-premises, administrative users, the application catalog, and scripted events. The portal is also used by administrative users for the manual provisioning of applications if required and to connect to any machines for troubleshooting.

Service providers can use this portal to add their own channel partners and allow them to manage their own clients.

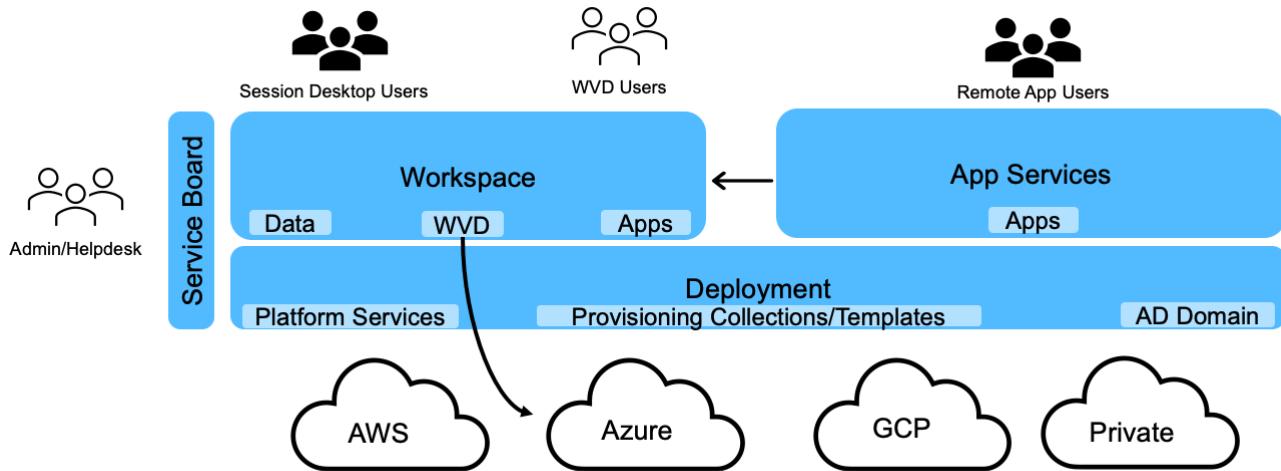
[Next: User Management](#)

## User Management

NetApp VDS uses Azure Active Directory for identity authentication and Azure Active Directory Domain Services for NTLM/Kerberos authentication. The ADConnect tool can be used to sync an on-prem Active Directory domain with Azure Active Directory.

New users can be added from the portal, or you can enable cloud workspace for existing users. Permissions for workspaces and application services can be controlled by individual users or by groups. From the management portal, administrative users can be defined to control permissions for the portal, workspaces, and so on.

The following figure depicts user management in NetApp VDS.



Each workspace resides in its own Active Directory organization unit (OU) under the Cloud Workspace OU as shown in the following figure.

Screenshot of the Active Directory Users and Computers interface:

- File** Action View Help
- Toolbar: Back, Forward, New, Open, Save, Print, Filter, etc.
- Left pane (Navigation):**
  - Active Directory Users and Computers [cwmgr1.vds]
  - Saved Queries
  - vds.demo
  - Builtin
  - Cloud Workspace
  - Cloud Workspace Companies
  - hpyh
  - hpyh-groups
  - ych
  - ych-desktop users
  - ych-groups
  - Cloud Workspace Servers
  - Cloud Workspace Service Accounts
  - Client Service Accounts
  - Infrastructure Service Accounts
  - Cloud Workspace Tech Users
  - Groups
  - Level3 Technicians
  - Computers
  - Domain Controllers
  - ForeignSecurityPrincipals
  - Managed Service Accounts
  - Users
- Right pane (List View):**

Name	Type	Description
87499	Security Group...	Microsoft Access
87500	Security Group...	Microsoft Excel
87501	Security Group...	Google Chrome
87502	Security Group...	Microsoft PowerPoint
87503	Security Group...	Microsoft Word
87517	Security Group...	PuTTY
ych-all users	Security Group...	Company All Users

For more info, see [this video](#) on user permissions and user management in NetApp VDS.

When an Active Directory group is defined as a CRAUserGroup using an API call for the datacenter, all the users in that group are imported into the CloudWorkspace for management using the UI. As the cloud workspace is enabled for the user, VDS creates user home folders, settings permissions, user properties updates, and so on.

If VDI User Enabled is checked, VDS creates a single-session RDS machine dedicated to that user. It prompts for the template and the datastore to provision.

The screenshot shows the 'Security Settings' configuration page. At the top left is a checked checkbox labeled 'VDI User Enabled'. To its right is an unchecked checkbox labeled 'Mobile Drive Enabled'. Below these are two dropdown menus: 'Hypervisor Template' set to 'Windows20192899ver1' and 'Storage Type' set to 'DS02'. In the center of the page is a grid of checkboxes. The first row contains 'Account Expiration Enabled' (unchecked) and 'Local Drive Access Enabled' (checked). The second row contains 'Force Password Reset at Next Login' (unchecked) and 'Wake On Demand Enabled' (unchecked). The third row contains 'Multi-factor Auth Enabled' (unchecked). At the bottom left is a grey 'Update' button.

[Next: Workspace Management](#)

## Workspace Management

A workspace consists of a desktop environment, which can be shared remote desktop sessions hosted on-premises or on any support cloud environment. With Microsoft Azure, the desktop environment can be persistent with Windows Virtual Desktops. Each workspace is associated with a specific organization or client. Options available when creating a new workspace can be seen in the following figure.

## New Workspace

Client & Settings
Choose Applications
Add Users
Review & Provision

Select a Client [Add](#)

No Clients Added.

Workspace Settings

Company Name

Application Settings

- Enable Remote App
- Enable App Locker
- Enable Application Usage Tracking

Primary Notification Email

Device Settings

- Disable Printing Access
- Enable Workspace User Data Storage

Security Settings

- Require Complex User Password
- Enable MFA for All Users
- Permit Access To Task Manager

Cancel
Continue



Each workspace is associated with specific deployment.

Workspaces contain associated apps and app services, shared data folders, servers, and a WVD instance. Each workspace can control security options like enforcing password complexity, multifactor authentication, file audits, and so on.

Workspaces can control the workload schedule to power on extra servers, limit the number of users per server, or set the schedule for the resources available for given period (always on/off). Resources can also be configured to wake up on demand.

Workspace can override the deployment VM resource defaults if required. For WVD, WVD host pools (which contains session hosts and app groups) and WVD workspaces can also be managed from the cloud workspace management suite portal. For more info on the WVD Host Pool, see this [video](#).

[Next: Application Management](#)

## Application Management

Task workers can quickly launch an application from the list of applications made available to them. App services publish applications from the Remote Desktop Services session hosts. With WVD, App Groups provide similar functionality from

multi-session Windows 10 host pools.

For office workers to power users, the applications that they require can be provisioned manually using a service board, or they can be auto-provisioned using the scripted events feature in NetApp VDS.

For more information, see the [NetApp Application Entitlement page](#).

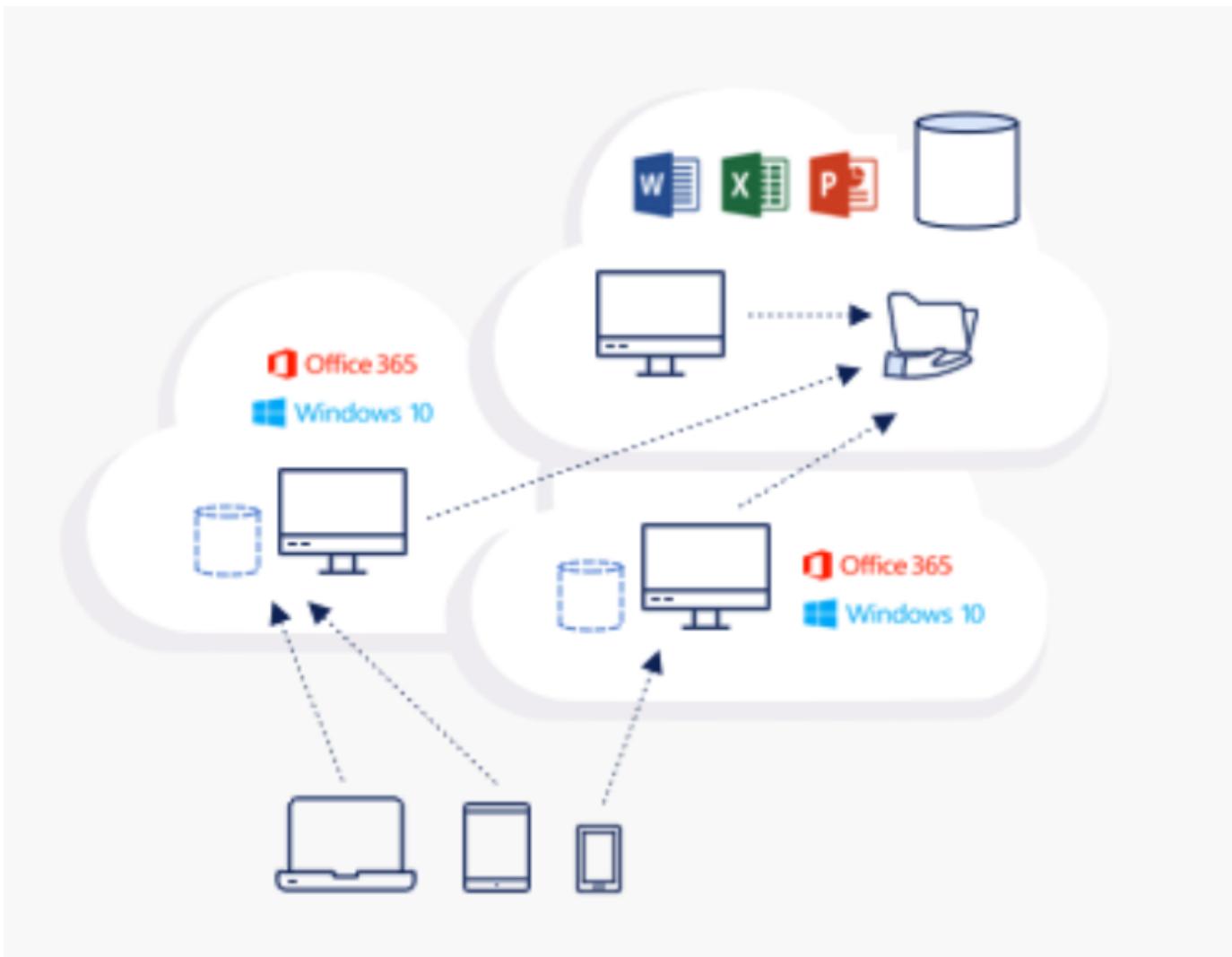
[Next: Data Management](#)

## Data Management

As a part of deployment, you can choose the file-services method to host the user profile, shared data, and the home drive folder. The available options are File Server, Azure Files, or Azure NetApp Files. However, after deployment, you can modify this choice with the TestvDC tool to point to any SMB share. There are various advantages to hosting with NetApp ONTAP. For more information, see the [NetApp Redirecting Storage Platform page](#).

### Global File Cache

When users are spread across multiple sites within a global namespace, Global File Cache can help reduce latency for frequently accessed data. Global File Cache deployment can be automated using a provisioning collection and scripted events. Global File Cache handles the read and write caches locally and maintains file locks across locations. Global File Cache can work with any SMB file servers, including Azure NetApp Files.

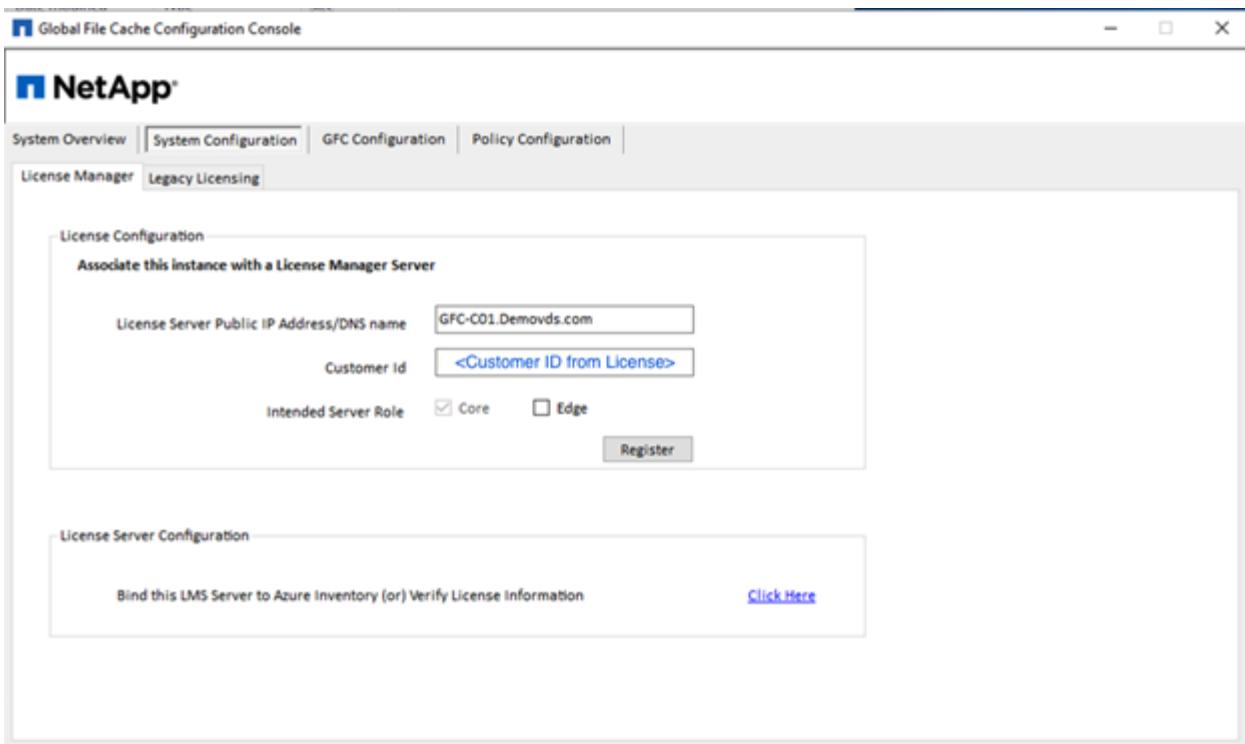


Global File Cache requires the following:

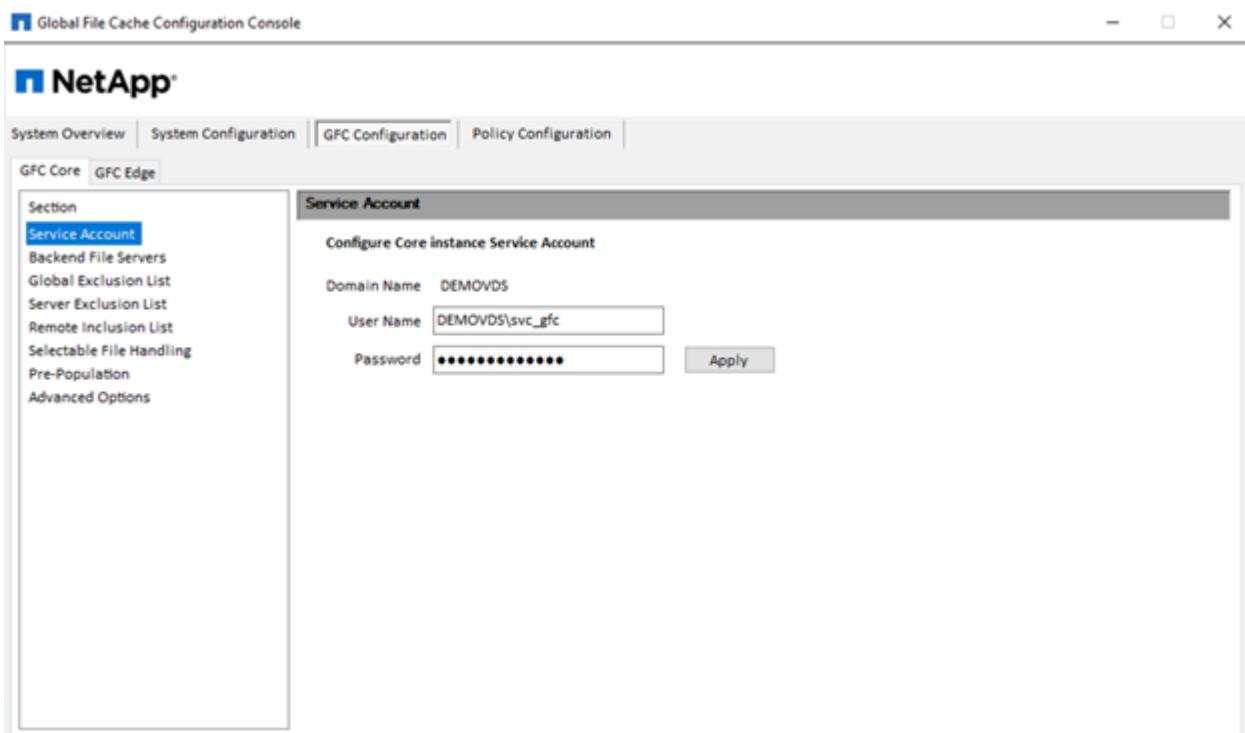
- Management server (License Management Server)
- Core
- Edge with enough disk capacity to cache the data

To download the software and to calculate the disk cache capacity for Edge, see the [GFC documentation](#).

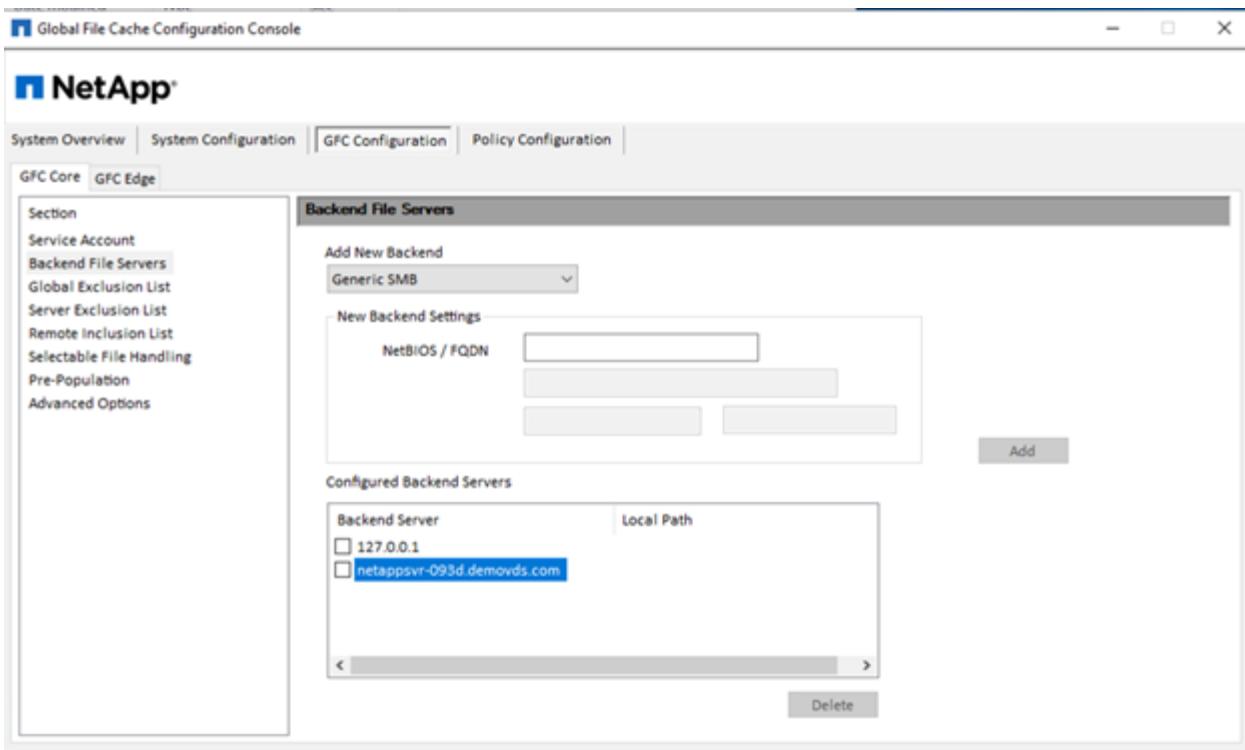
For our validation, we deployed the core and management resources on the same VM at Azure and edge resources on NetApp HCI. Please note that the core is where high-volume data access is required and the edge is a subset of the core. After the software is installed, the license must be activated before use. Under License Configuration section, use the link [Click Here](#) to complete the license activation. Then, register the core.



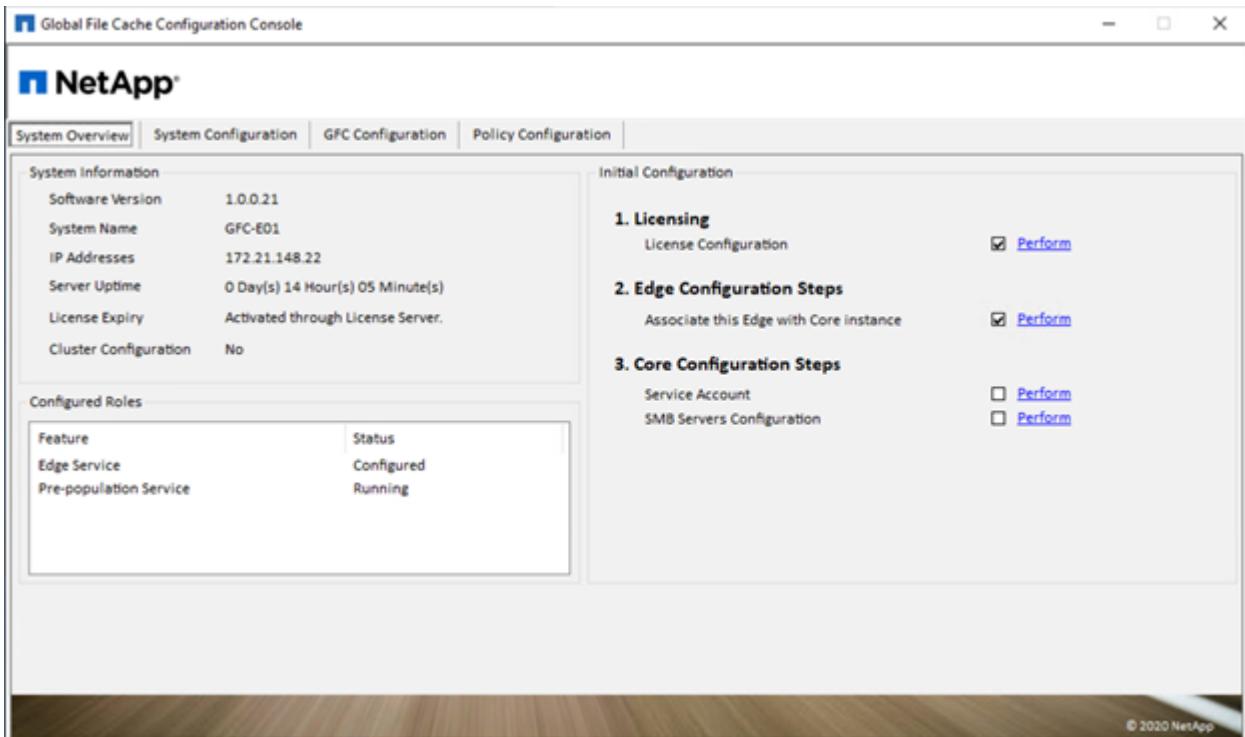
Provide the service account to be used for the Global File Cache. For the required permissions for this account, see the [GFC documentation](#).



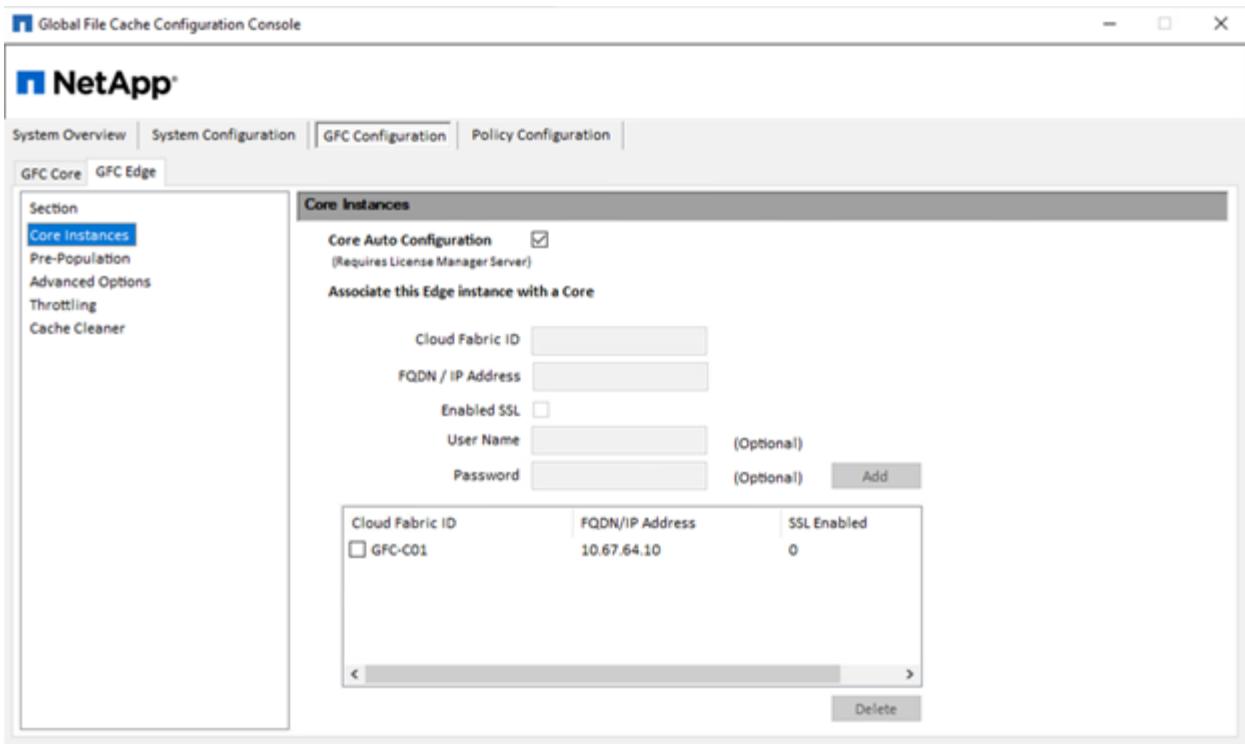
Add a new backend file server and provide the file server name or IP.



On the edge, the cache drive must have the drive letter D. If it does not, use diskpart.exe to select the volume and change drive letter. Register with the license server as edge.



If core auto-configuration is enabled, core information is retrieved from the license management server automatically.



From any client machine, the administrators that use to access the share on file server, can access it via GFC edge using UNC Path `\\\FASTDATA\<core server name>\<backend file server name>\<share name>`. Administrators can include this path in user logonscript or GPO for users drive mapping at the edge location.

To provide transparent access for users across the globe, an administrator can setup the Microsoft Distributed Filesystem (DFS) with links pointing to file server shares and to edge locations.



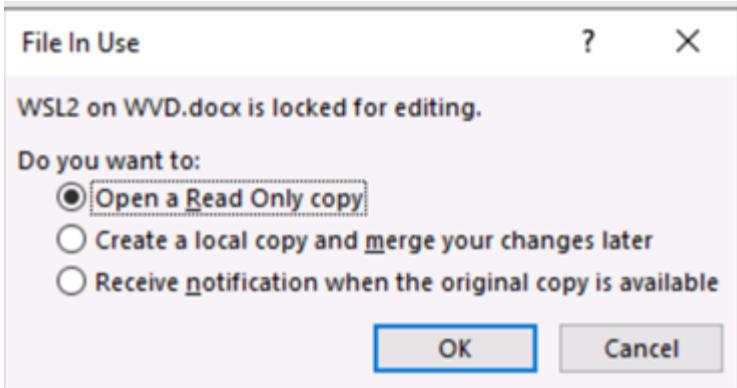
When users log in with Active Directory credentials based on the subnets associated with the site, the appropriate link is utilized by the DFS client to access the data.

Name	Site	Location	Type	Description
10.67.64.0/20	Azure-US-East		Subnet	
172.21.146.0/24	RTP		Subnet	
172.21.147.0/24	RTP		Subnet	
172.21.148.0/24	RTP		Subnet	
172.21.149.0/24	RTP		Subnet	
172.21.150.0/24	RTP		Subnet	

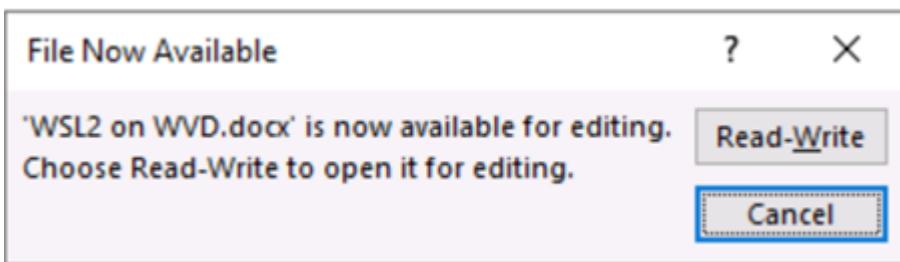
File icons change depending on whether a file is cached; files that are not cached have a grey X on the lower left corner of the icon. After a user in an edge location accesses a file, that file is cached, and the icon changes.

Name	Date modified	Type	Size
Department	10/1/2020 5:28 PM	File folder	
Outlook	10/12/2020 3:05 PM	File folder	
Outlook Files	10/12/2020 6:07 PM	File folder	
Output	10/12/2020 3:12 PM	File folder	
WindowsPowerShell	10/11/2020 6:24 PM	File folder	
FSLogix	10/11/2020 9:11 PM	Registration Entries	2 KB
GFC-1-0-0-21-Release	10/11/2020 10:05 ...	Application	26,869 KB
PDF1.pdf	6/22/2016 9:31 PM	PDF File	1,101 KB
PDF2.pdf	6/22/2016 9:31 PM	PDF File	1,066 KB
Spreadsheet.xlsx	6/22/2016 9:31 PM	XLSX File	298 KB
UserEdit.doc	6/22/2016 9:31 PM	DOC File	1,061 KB
UserEdit1.doc	10/12/2020 3:13 PM	DOC File	1,061 KB
UserEdit2.doc	10/12/2020 3:01 PM	DOC File	1,063 KB
UserMindmap.mm	6/22/2016 9:31 PM	MM File	86 KB
UserPresentation.ppt	6/22/2016 9:31 PM	PPT File	3,071 KB

When a file is open and another user is trying to open the same file from an edge location, the user is prompted with the following selection:



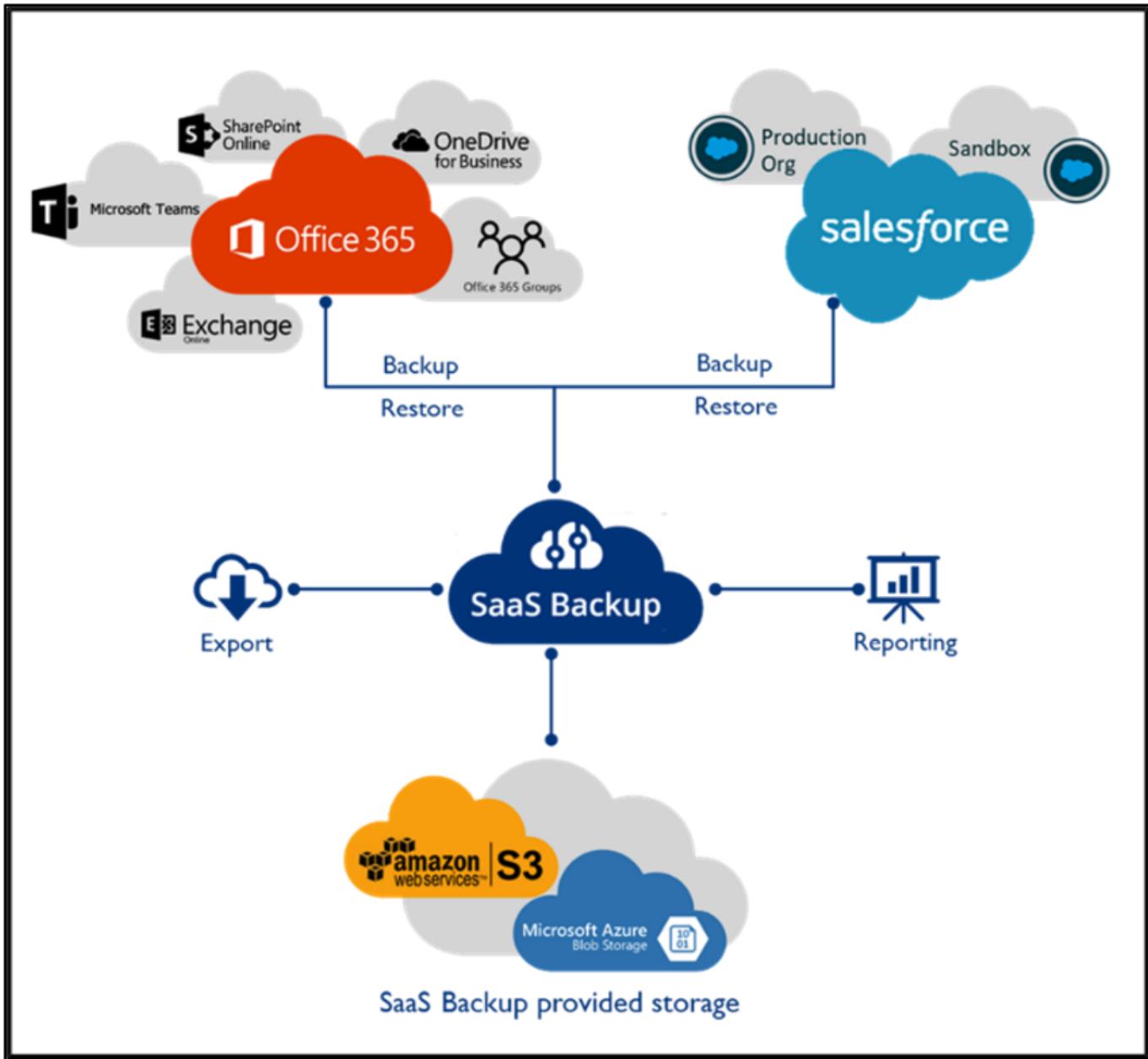
If the user selects the option to receive a notification when the original copy is available, the user is notified as follows:



For more information, see this [video on Talon and Azure NetApp Files Deployment](#).

## SaaS Backup

NetApp VDS provides data protection for Salesforce and Microsoft Office 365, including Exchange, SharePoint, and Microsoft OneDrive. The following figure shows how NetApp VDS provides SaaS Backup for these data services.



For a demonstration of Microsoft Office 365 data protection, see [this video](#).

For demonstration of Salesforce data protection, see [this video](#).

[Next: Operation Management](#)

## Operation Management

With NetApp VDS, administrators can delegate tasks to others. They can connect to deployed servers to troubleshoot, view logs, and run audit reports. While assisting customers, helpdesk or level-3 technicians can shadow user sessions, view process lists, and kill processes if required.

For information on VDS logfiles, see the [Troubleshooting Failed VDA Actions page](#).

For more information on the required minimum permissions, see the [VDA Components and](#)

## Permissions page.

If you would like to manually clone a server, see the [Cloning Virtual Machines page](#).

To automatically increase the VM disk size, see the [Auto-Increase Disk Space Feature page](#).

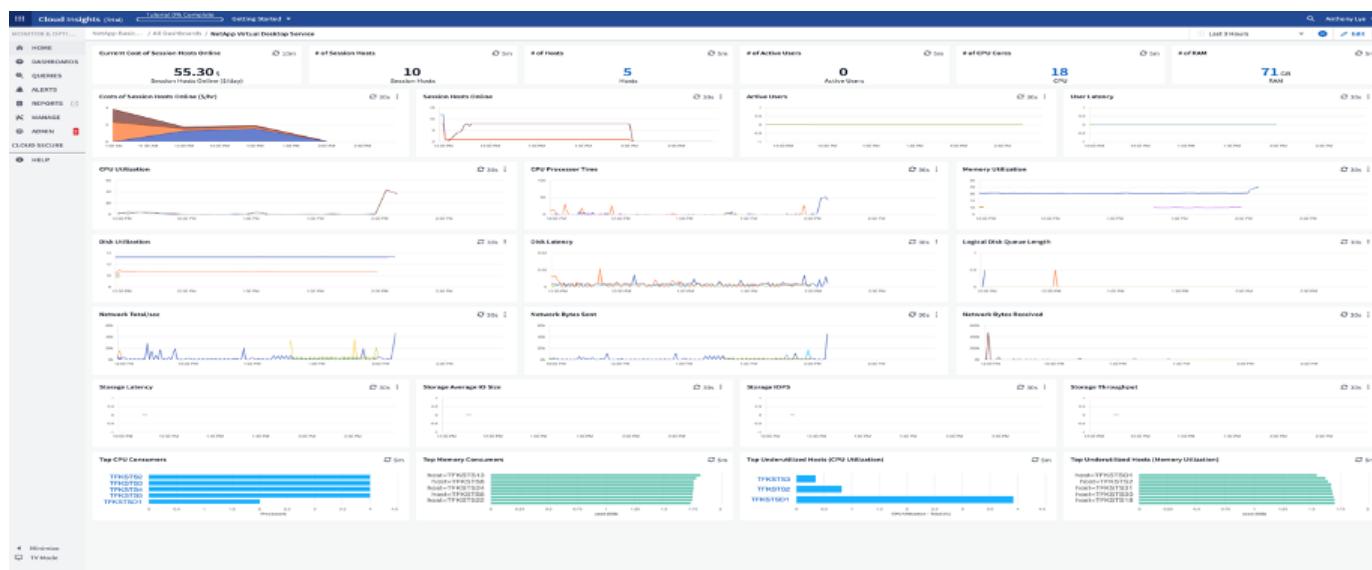
To identify the gateway address to manually configure the client, see the [End User Requirements page](#).

## Cloud Insights

NetApp Cloud Insights is a web-based monitoring tool that gives you complete visibility into infrastructure and applications running on NetApp and other third-party infrastructure components. Cloud Insights supports both private cloud and public clouds for monitoring, troubleshooting, and optimizing resources.

Only the acquisition unit VM (can be Windows or Linux) must be installed on a private cloud to collect metrics from data collectors without the need for agents. Agent-based data collectors allow you to pull custom metrics from Windows Performance Monitor or any input agents that Telegraf supports.

The following figure depicts the Cloud Insights VDS dashboard.



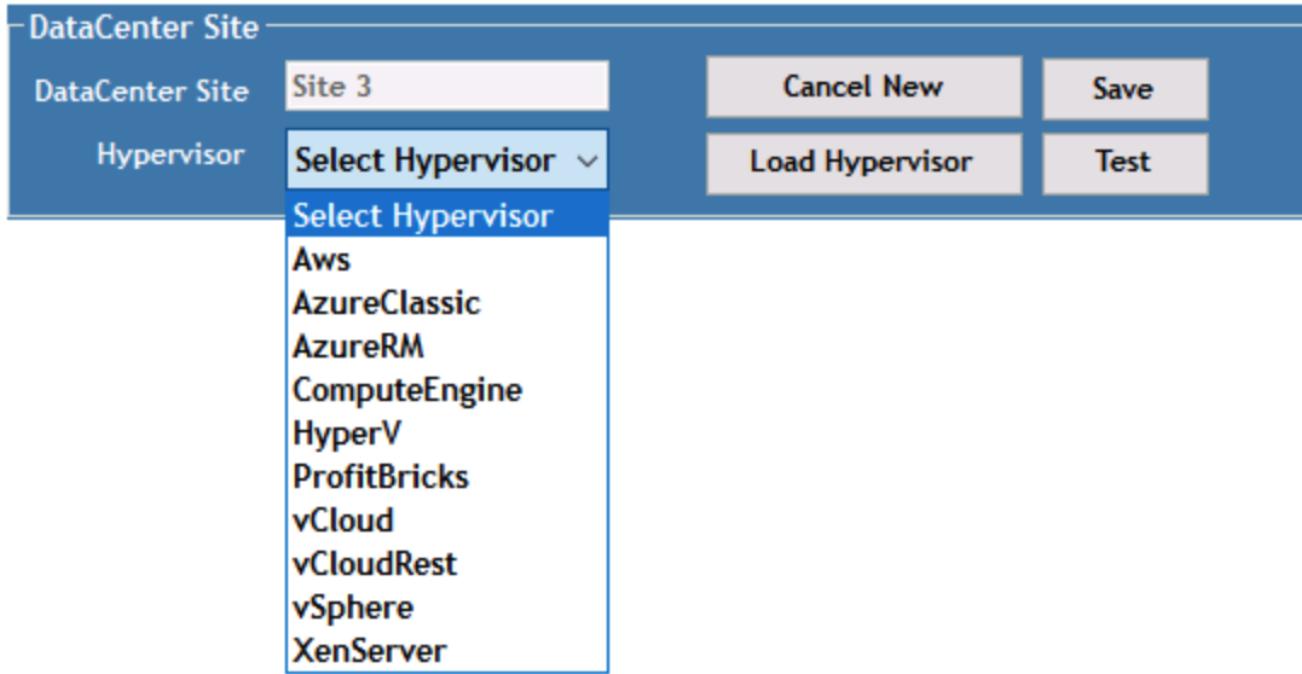
For more info on NetApp Cloud Insights, see [this video](#).

Next: [Tools and Logs](#)

## Tools and Logs

### DCConfig Tool

The DCConfig tool supports the following hypervisor options for adding a site:



The screenshot shows a 'Configuration' interface with a navigation bar at the top containing links for 'DataCenter', 'Accounts', 'Email', 'DatabaseConnection', 'Exclude', 'DataCenter Sites', 'Product Keys', 'Static IpAddress', and 'Drive Mapping'. Below the navigation bar is a table titled 'Drive Mapping' with columns 'Description' and 'DriveLetter'. The table contains three rows: 'Shared Data' (DriveLetter P), 'FTP' (DriveLetter F), and 'User Home' (DriveLetter H, which is highlighted with a blue background). A 'Save' button is located above the table.

	Description	DriveLetter
	Shared Data	P
▶	FTP	F
▶	User Home	H

Workspace-specific drive letter mapping for shared data can be handled using GPO. Professional Services or the support team can use the advanced tab to customize settings like Active Directory OU names, the option to enable or disable deployment of FSLogix, various timeout values, and so on.

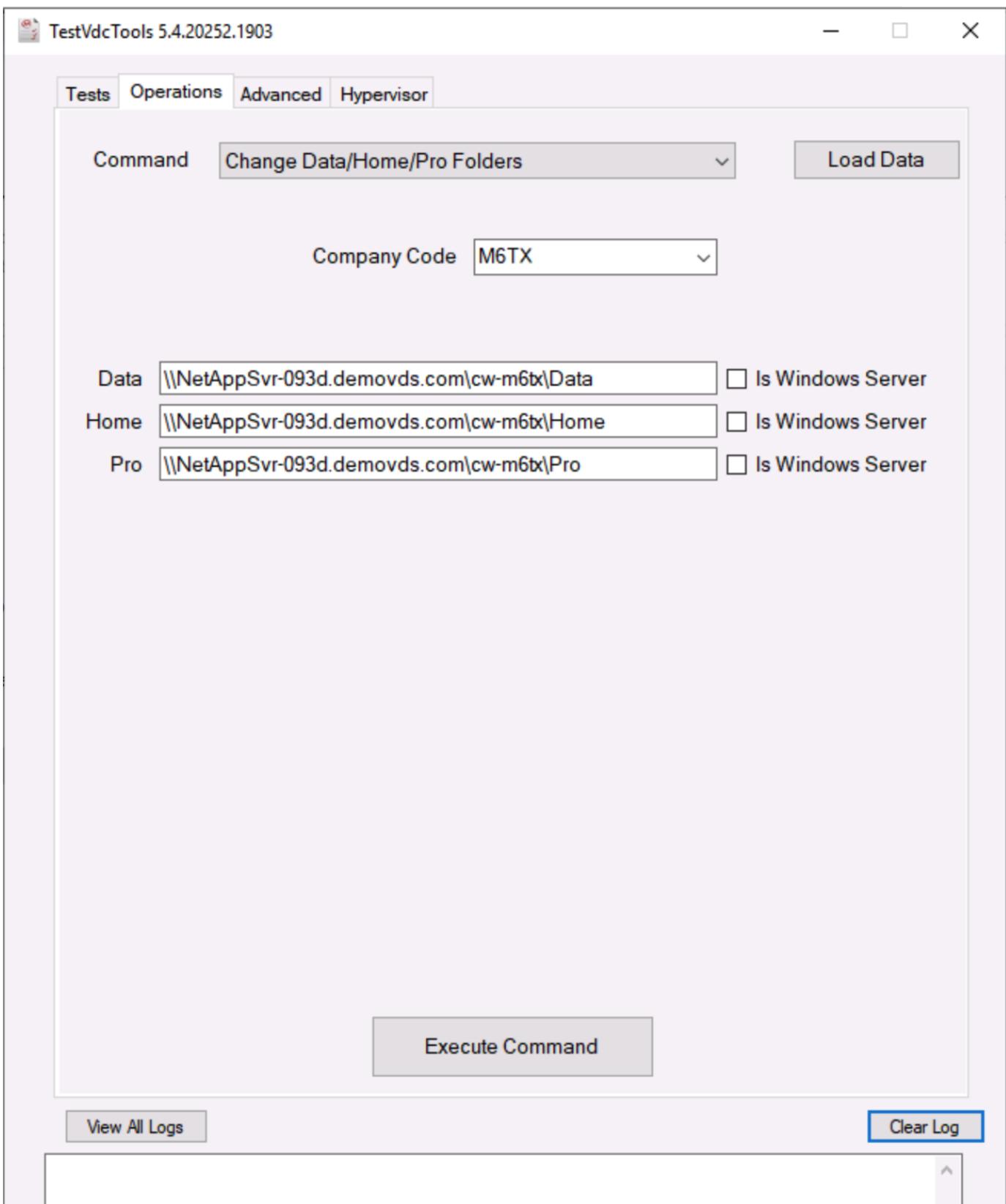
GroupName	FriendlyName	Value
Server Creation	UpdateVMNameWhenRemovedFromCache	<input type="checkbox"/>
Server Creation	UpdateVmIruleRules	<input checked="" type="checkbox"/>
Server Creation	WaitAfterRebootMin	6
Server Creation	WaitAfterHypervisorCreateMin	1
Server Creation	WaitAfterSysPrepMin	10
Server Creation	WaitAfterSysPrepOr2008ServersMin	30
Server Creation	GFI Agent Path	
Server Creation	Automated Cloning Enabled	<input checked="" type="checkbox"/>
Server Creation	CompaniesOU	Cloud Workspace Companies
Server Creation	Install ThinPrint v11	<input checked="" type="checkbox"/>
Server Creation	ServersOU	Cloud Workspace Servers
Server Creation	Install FLogix	<input checked="" type="checkbox"/>
Server Creation	Use Default OUs	<input checked="" type="checkbox"/>
Server Creation	Max Threads	50
Server Creation	Wait for DNS to Update Minutes	15
Check Vdc Tools Version	Run Every X Minutes	5
Daily Actions	Enabled	<input checked="" type="checkbox"/>
Daily Actions	Run at startup	<input checked="" type="checkbox"/>
Generate Reports	Time Of Day	06:00
Daily Maintenance	Enabled	<input checked="" type="checkbox"/>
Daily Maintenance	Time Of Day	00:01
Weekly Maintenance	Enabled	<input checked="" type="checkbox"/>
Weekly Maintenance	Time Of Day	00:01
Automatic Resource Allocation	Day	Sunday
Resource Allocation	Enabled	<input checked="" type="checkbox"/>
EmailReports	Use Data Center Defaults	<input checked="" type="checkbox"/>
EmailReports	IncludeEmailAttachment	<input type="checkbox"/>
Server Heartbeat	Interval Minutes	15

## TestVdc Tools

The TestVdc tool is available in the `C:\Program Files\CloudWorkspace\TestVdcTools\` folder.

The following operations can be performed by Professional Services or an administrator:

- Change the SMB Path for a workspace.



- Change the site for provisioning collection.

[Tests](#) [Operations](#) [Advanced](#) [Hypervisor](#)Command [Edit Provisioning Collection](#) [Load Data](#)Provisioning Collection [Windows2019](#)Description [On vSphere Site 2](#)Share Drive [P](#)Minimum Cache Level [1](#)Operating System [Windows Server 2019](#)Collection Type [Shared](#)

	Data Center Site	Role	Template	Storage
▶	<a href="#">Site 2</a>	<a href="#">TSData</a>	<a href="#">Windows2019</a>	<a href="#">DS01</a>
*				

[Execute Command](#)[View All Logs](#)[Clear Log](#)[Log Files](#)

Name	Date modified	Type	Size
CwAgent	9/19/2020 12:35 PM	File folder	
CWAutomationService	9/19/2020 12:34 PM	File folder	
CWManagerX	9/19/2020 12:53 PM	File folder	
CwVmAutomationService	9/19/2020 12:34 PM	File folder	
TestVdcTools	9/22/2020 8:20 PM	File folder	
report	9/19/2020 12:18 PM	Executable Jar File	705 KB

[Next: Conclusion](#)

## Conclusion

The NetApp Virtual Desktop Service provides an easy-to-consume virtual desktop and application environment with a sharp focus on business challenges. By extending VDS with NetApp HCI, you can use powerful NetApp features in a VDS environment, including in-line deduplication, compaction, thin provisioning, and compression. These features save storage costs and improve performance with all-flash storage. NetApp HCI offers high performance compute, a choice of GPU resources, and with VMware vSphere hypervisor which minimizes the server provisioning time using vSphere API for Array integration. Using the hybrid cloud, customers have the choice to pick the right environment for their demanding workloads and saving expenditure. The desktop session running on-premises can have access to cloud resources based on policy.

[Next: Where to Find Additional Information](#)

## Where to Find Additional Information

To learn more about the information that is described in this document, review the following documents and/or websites:

- NetApp Cloud

<https://cloud.netapp.com/home>

- NetApp VDS Product Documentation

<https://docs.netapp.com/us-en/virtual-desktop-service/index.html>

- Connect your on-premises network to Azure with VPN Gateway

<https://docs.microsoft.com/en-us/learn/modules/connect-on-premises-network-with-vpn-gateway/>

- Azure Portal

<https://portal.azure.com>

- Microsoft Windows Virtual Desktop

<https://azure.microsoft.com/en-us/services/virtual-desktop/>

- Azure NetApp Files Registration

[https://docs.microsoft.com/en-us/azure/azure-netapp-files/azure-netapp-files-register?WT.mc\\_id=Portal-Microsoft\\_Azure\\_NetApp](https://docs.microsoft.com/en-us/azure/azure-netapp-files/azure-netapp-files-register?WT.mc_id=Portal-Microsoft_Azure_NetApp)

## End User Computing on NetApp HCI with VMware

Overview of the End User Computing capabilities on NetApp HCI with VMware Horizon.

[Learn more](#) about NetApp HCI.

### End User Computing (EUC) versus Virtual Desktop Infrastructure (VDI)

Traditionally, the focus on end user computing was centered around the virtualization of desktop infrastructure, or VDI. As VDI evolves, the focus of the conversation has shifted to the accessibility of end user applications and data. To read more about the evolution of VDI and the industry migration to EUC, [read the blog](#) discussing the evolution of VDI and infrastructure to EUC and application accessibility.

### NetApp Validated Architectures and Technical Reports

End User Computing on NetApp HCI with VMware Horizon is a set of fully validated and supported solutions. Details of the design and deployment considerations are documented in the NetApp Validated Architecture (NVA) documents and Technical Reports (TR).

- [NVA: EUC with VMware \(Design Guide\)](#)
- [NVA: EUC with VMware \(Deployment Guide\)](#)
- [NVA: EUC with VMware and NVIDIA GPUs \(Design Guide\)](#)
- [NVA: EUC with VMware and NVIDIA GPUs\(Deployment Guide\)](#)
- [TR: EUC with VMware for 3D Graphics](#)

### Additional Material

# **TR-4854: NetApp HCI for Citrix Virtual Apps and Desktops with Citrix Hypervisor**

Suresh Thoppay, NetApp

NetApp HCI infrastructure allows you to start small and build in small increments to meet the demands of virtual desktop users. Compute or storage nodes can be added or removed to address changing business requirements.

Citrix Virtual Apps and Desktops provides a feature-rich platform for end-user computing that addresses various deployment needs, including support for multiple hypervisors. The premium edition of this software includes tools to manage images and user policies.

Citrix Hypervisor (formerly known as Citrix Xen Hypervisor) provides additional features to Citrix Virtual Apps and Desktops compared to running on other hypervisor platforms. The following are key benefits of running on Citrix Hypervisor:

- A Citrix Hypervisor license is included with all versions of Citrix Virtual Apps and Desktops. This licensing helps to reduce the cost of running the Citrix Virtual Apps and Desktops platform.
- Features like PVS Accelerator and Storage Accelerator are only available with Citrix Hypervisor.
- For Citrix solutions, the Citrix Hypervisor is the preferred workload choice.
- Available in Long Term Service Release (LTSR; aligns with Citrix Virtual Apps and Desktops) and Current Release (CR) options.

## **Abstract**

This document reviews the solution architecture for Citrix Virtual Apps and Desktops with Citrix Hypervisor. It provides best practices and design guidelines for Citrix implementation on NetApp HCI. It also highlights multitenancy features, user profiles, and image management.

## **Solution Overview**

Service providers who deliver the Virtual Apps and Desktops service prefer to host it on Citrix Hypervisor to reduce cost and for better integration. The NetApp Deployment Engine (NDE), which performs automated installation of VMware vSphere on NetApp HCI, currently doesn't support deployment of Citrix Hypervisor. Citrix Hypervisor can be installed on NetApp HCI using PXE boot or installation media or other deployment methods supported by Citrix.

Citrix Virtual Apps and Desktops can automate the provisioning of desktops and session hosts either using Citrix Provisioning (network-based) or by Machine Creation Services (hypervisor storage-based). Both Microsoft Windows-based OSs and popular Linux flavors are supported. Existing physical

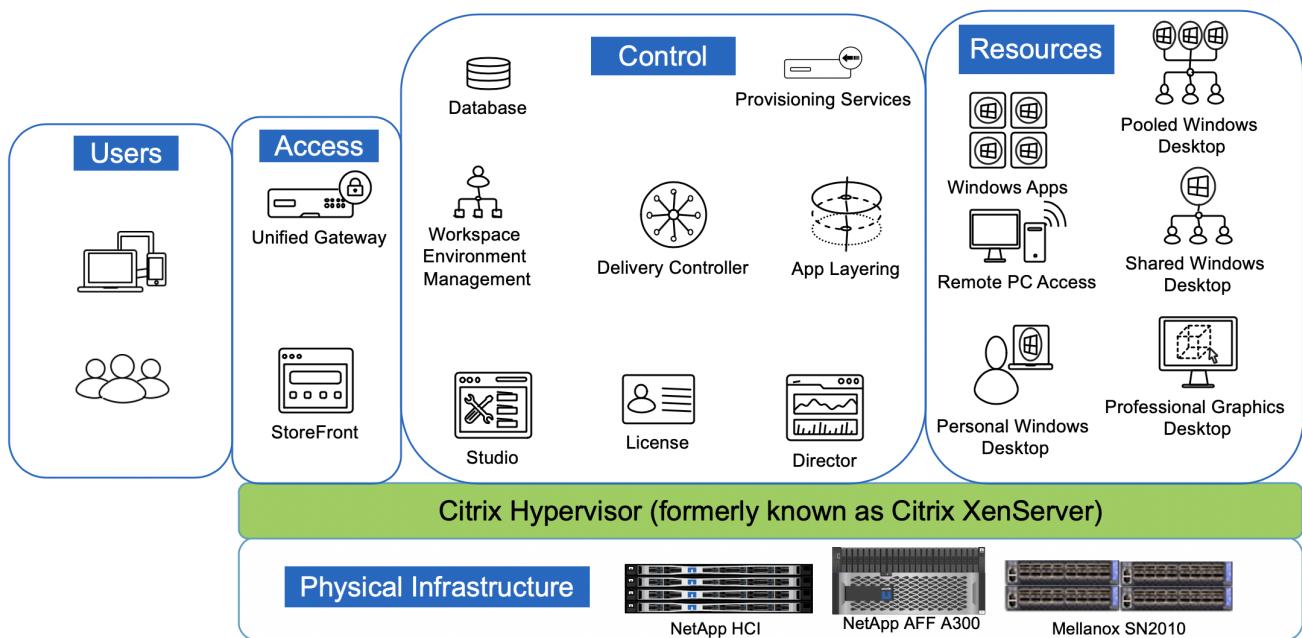
workstations, desktop PCs, and VMs on other hypervisors that are not enabled for auto-provisioning can also be made available for remote access by installing the agents.

The Citrix Workspace Application, a client software used to access Virtual Apps and Desktops, is supported on various devices including tablets and mobile phones. Virtual Apps and Desktops can be accessed using a browser-based HTML5 interface internally or externally to the deployment location.

Based on your business needs, the solution can be extended to multiple sites. However, remember that NetApp HCI storage efficiencies operate on a per-cluster basis.

The following figure shows the high-level architecture of the solution. The access, control, and resource layers are deployed on top of Citrix Hypervisor as virtual machines. Citrix Hypervisor runs on NetApp HCI compute nodes. The virtual disk images are stored in the iSCSI storage repository on NetApp HCI storage nodes.

A NetApp AFF A300 is used in this solution for SMB file shares to store user profiles with FSLogix containers, Citrix profile management (for multisession write-back support), Elastic App Layering images, and so on. We also use SMB file share to mount ISO images on Citrix Hypervisor.



A Mellanox SN2010 switch is used for 10/25/100Gb Ethernet connectivity. Storage nodes use SFP28 transceivers for 25Gb connection, compute nodes use SFP/SFP+ transceivers for 10Gb connection, and interswitch links are QSFP28 transceivers for a 100Gb connection.

Storage ports are configured with multichassis link aggregation (MLAG) to provide total throughput of 50Gb and are configured as trunk ports. Compute node ports are configured as hybrid ports to create a VLAN for iSCSI, XenMotion, and workload VLANs.

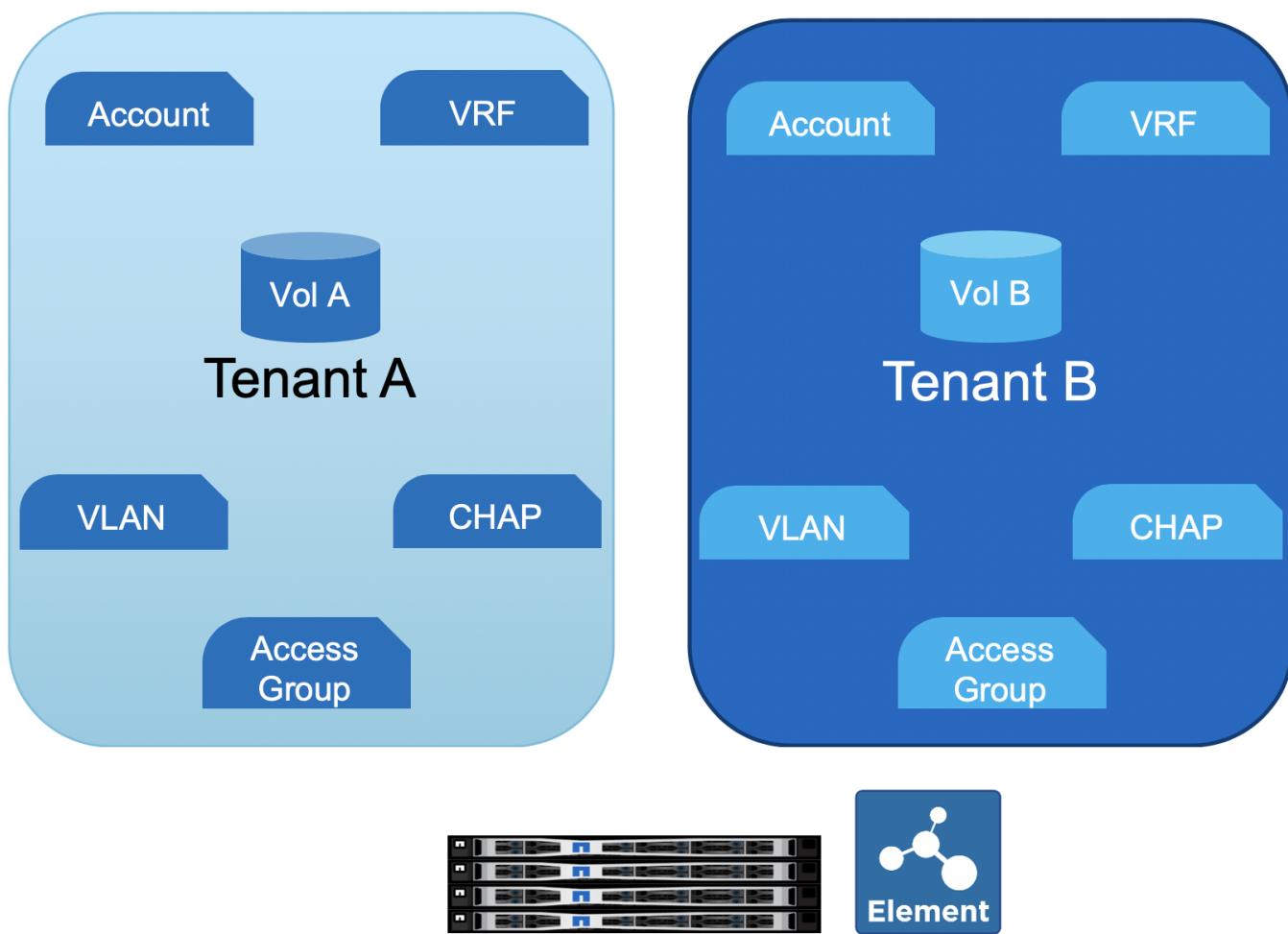
# Physical Infrastructure

## NetApp HCI

NetApp HCI is available as compute nodes or storage nodes. Depending on the storage node model, a minimum of two to four nodes is required to form a cluster. For the compute nodes, a minimum of two nodes are required to provide high availability. Based on demand, nodes can be added one at a time to increase compute or storage capacity.

A management node (mNode) deployed on a compute node runs as a virtual machine on supported hypervisors. The mNode is used for sending data to ActiveIQ (a SaaS-based management portal), to host a hybrid cloud control portal, as a reverse proxy for remote support of NetApp HCI, and so on.

NetApp HCI enables you to have nondistributive rolling upgrades. Even when one node is down, data is serviced from the other nodes. The following figure depicts NetApp HCI storage multitenancy features.



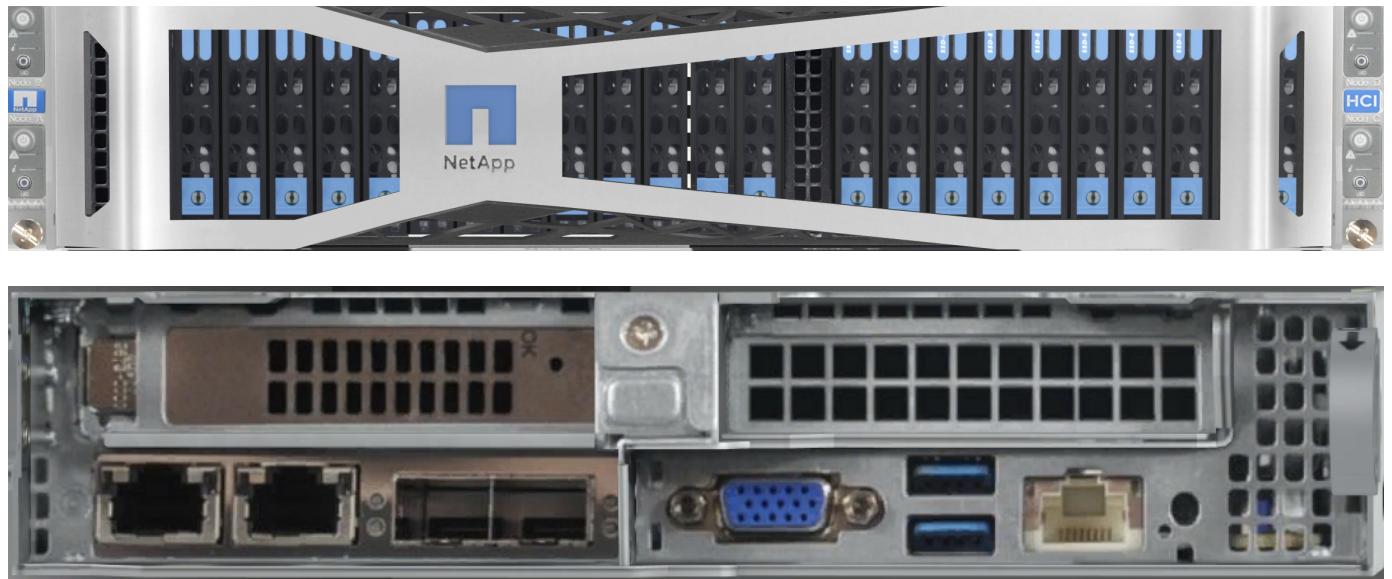
NetApp HCI Storage provides flash storage through iSCSI connection to compute nodes. iSCSI connections can be secured using CHAP credentials or a volume access group. A volume access group

only allows authorized initiators to access the volumes. An account holds a collection of volumes, the CHAP credential, and the volume access group. To provide network-level separation between tenants, different VLANs can be used, and volume access groups also support virtual routing and forwarding (VRF) to ensure the tenants can have same or overlapping IP subnets.

A RESTful web interface is available for custom automation tasks. NetApp HCI has PowerShell and Ansible modules available for automation tasks. For more info, see [NetApp.IO](#).

## Storage Nodes

NetApp HCI supports two storage node models: the H410S and H610S. The H410 series comes in a 2U chassis containing four half-width nodes. Each node has six SSDs of sizes 480GB, 960GB, or 1.92TB with the option of drive encryption. The H410S can start with a minimum of two nodes. Each node delivers 50,000 to 100,000 IOPS with a 4K block size. The following figure presents a front and back view of an H410S storage node.



The H610S is a 1U storage node with 12 NVMe drives of sizes 960GB, 1.92TB, or 3.84TB with the option of drive encryption. A minimum of four H610S nodes are required to form a cluster. It delivers around 100,000 IOPS per node with a 4K block size. The following figure depicts a front and back view of an H610S storage node.



In a single cluster, there can be a mix of storage node models. The capacity of a single node can't exceed 1/3 of the total cluster size. The storage nodes come with two network ports for iSCSI (10/25GbE

– SFP28) and two ports for management (1/10/GbE – RJ45). A single out-of-band 1GbE RJ45 management port is also available.

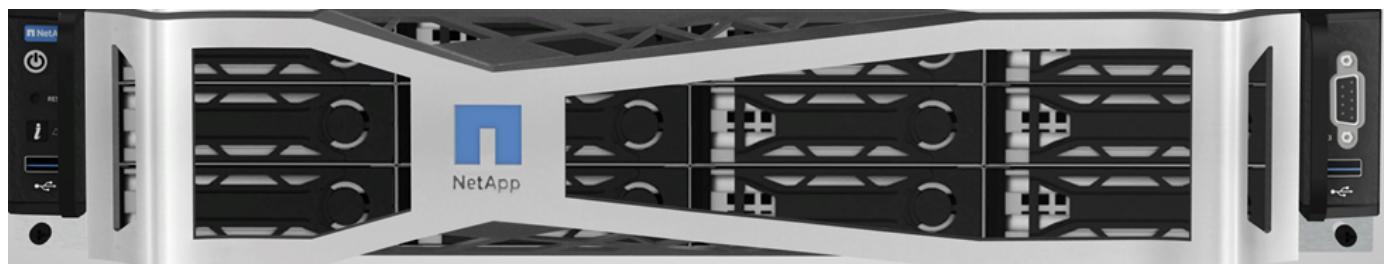
## Compute Nodes

NetApp HCI compute nodes are available in three models: H410C, H610C, and H615C. Compute nodes are all RedFish API-compatible and provide a BIOS option to enable Trusted Platform Module (TPM) and Intel Trusted eXecution Technology (TXT).

The H410C is a half-width node that can be placed in a 2U chassis. The chassis can have a mix of compute and storage nodes. The H410C comes with first-generation Intel Xeon Silver/Gold scalable processors with 4 to 20 cores in dual-socket configurations. The memory size ranges from 384GB to 1TB. There are four 10/25GbE (SFP28) ports and two 1GbE RJ45 ports, with one 1GbE RJ45 port available for out-of-band management. The following figure depicts a front and back view of an H410C compute node.



The H610C is 2RU and has a dual-socket first generation Intel Xeon Gold 6130 scalable processor with 16 cores of 2.1GHz, 512GB RAM and two NVIDIA Tesla M10 GPU cards. This server comes with two 10/25GbE SFP28 ports and two 1GbE RJ45 ports, with one 1GbE RJ45 port available for out-of-band management. The following figure depicts a front and back view of an H610C compute node.





The H610C has two Tesla M10 cards providing a total of 64GB frame buffer memory with a total of 8 GPUs. It can support up to 64 personal virtual desktops with GPU enabled. To host more sessions per server, a shared desktop delivery model is available.

The H615C is a 1RU server with a dual socket for second-generation Intel Xeon Silver/Gold scalable processors with 4 to 24 cores per socket. RAM ranges from 384GB to 1.5TB. One model contains three NVIDIA Tesla T4 cards. The server includes two 10/25GbE (SFP28) and one 1GbE (RJ45) for out-of-band management. The following figure depicts a front and back view of an H615C compute node.



The H615C includes three Tesla T4 cards providing a total of 48GB frame buffer and three GPUs. The T4 card is a general-purpose GPU card that can be used for AI inference workloads as well as for professional graphics. It includes ray tracing cores that can help simulate light reflections.

## Hybrid Cloud Control

The Hybrid Cloud Control portal is often used for scaling out NetApp HCI by adding storage or/and compute nodes. The portal provides an inventory of NetApp HCI compute and storage nodes and a link to the ActiveIQ management portal. See the following screenshot of Hybrid Cloud Control.

The screenshot shows the NetApp Hybrid Cloud Control interface. In the top left, there's a sidebar with a blue header containing the NetApp logo and 'Hybrid Cloud Control'. Below the header, it says 'HCl\_Installation\_01'. The main content area has a header 'Upgrades' with tabs for 'MANAGEMENT SERVICES', 'STORAGE' (which is selected), and 'COMPUTE'. Under 'Upgrade Storage Cluster', it says 'Element is the operating system of your storage cluster. It includes software, NetApp Deployment Engine and firmware.' Below this, it says 'Select a storage cluster to see the latest compatible upgrades packages.' A table lists clusters: 'Storage\_Cluster\_01' (36 nodes, Element 11.5, Upgrades Available) and 'Storage\_Cluster\_02' (6 nodes, Element 11.3, Upgrades Available). For 'Storage\_Cluster\_02', a dropdown menu shows upgrade options: 'Element-Version-12.0' (selected), 'Element-Version-11.7', 'Element-Version-11.5', and 'Element-Version-11.3'. There's also a link 'See release notes'. At the bottom right of the table area is a blue 'Begin Upgrade' button.

## NetApp AFF

NetApp AFF provides an all-flash, scale-out file storage system, which is used as a part of this solution. ONTAP is the storage software that runs on NetApp AFF. Some key benefits of using ONTAP for SMB file storage are as follows:

- Storage Virtual Machines (SVM) for secure multitenancy
- NetApp FlexGroup technology for a scalable, high-performance file system
- NetApp FabricPool technology for capacity tiering. With FabricPool, you can keep hot data local and transfer cold data to cloud storage).
- Adaptive QoS for guaranteed SLAs. You can adjust QoS settings based on allocated or used space.
- Automation features (RESTful APIs, PowerShell, and Ansible modules)
- Data protection and business continuity features including NetApp Snapshot, NetApp SnapMirror, and NetApp MetroCluster technologies

## Mellanox Switch

A Mellanox SN2010 switch is used in this solution. However, you can also use other compatible switches. The following Mellanox switches are frequently used with NetApp HCI.

<b>Model</b>	<b>Rack Unit</b>	<b>SFP28 (10/25GbE) ports</b>	<b>QSFP (40/100GbE) ports</b>	<b>Aggregate Throughput (Tbps)</b>
SN2010	Half-width	18	4	1.7
SN2100	Half-width	–	16	3.2
SN2700	Full-width	–	32	6.4



QSFP ports support 4x25GbE breakout cables.

Mellanox switches are open Ethernet switches that allow you to pick the network operating system. Choices include the Mellanox Onyx OS or various Linux OSs such as Cumulus-Linux, Linux Switch, and so on. Mellanox switches also support the switch software development kit, the switch abstraction interface (SAI; part of the Open Compute Project), and Software for Open Networking in the Cloud (SONIC).

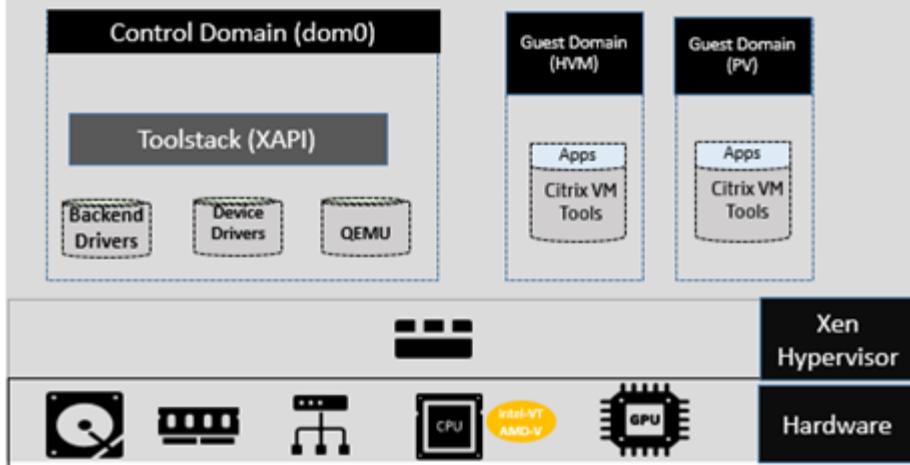
Mellanox switches provide low latency and support traditional data center protocols and tunneling protocols like VXLAN. VXLAN Hardware VTEP is available to function as an L2 gateway. These switches support various certified security standards like UC API, FIPS 140-2 (System Secure Mode), NIST 800-181A (SSH Server Strict Mode), and CoPP (IP Filter).

Mellanox switches support automation tools like Ansible, SALT Stack, Puppet, and so on. The Web Management Interface provides the option to execute multi-line CLI commands.

## Citrix Hypervisor

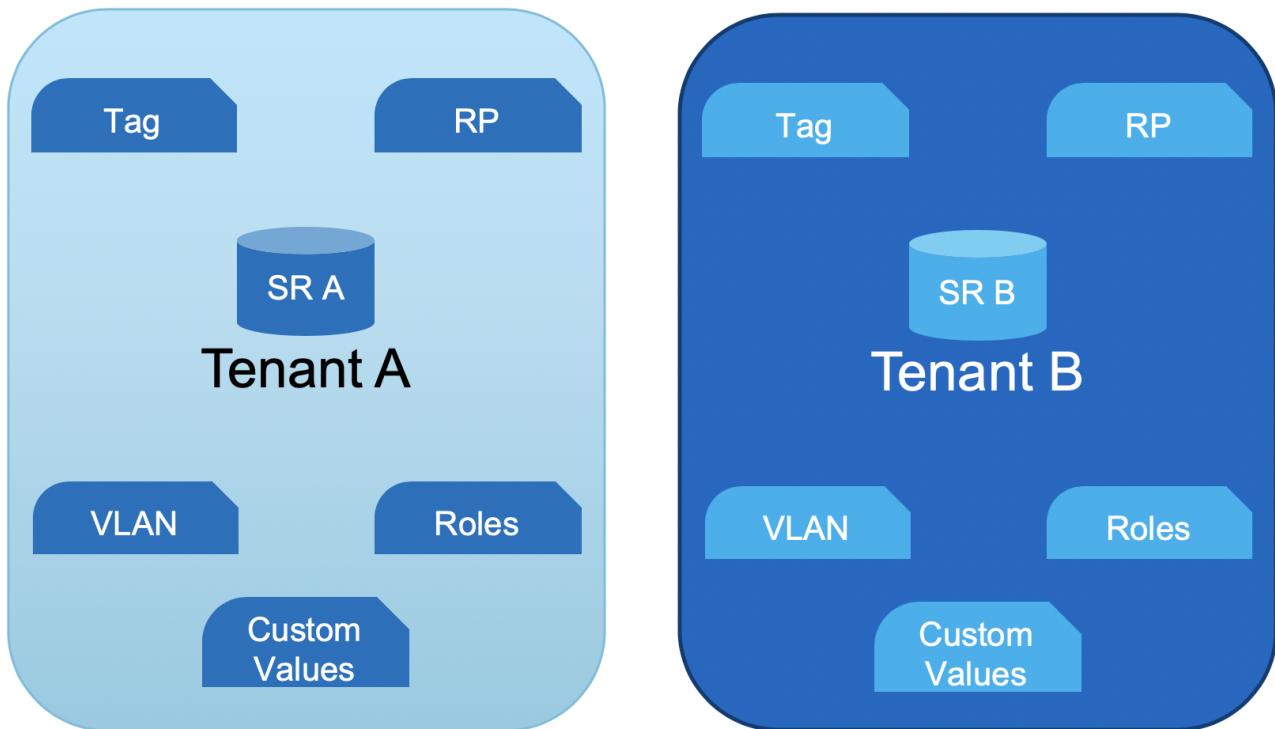
Citrix Hypervisor (formerly known as XenServer) is the industry-leading, cost-effective, open-source platform for desktop virtualization infrastructure. XenCenter is a light-weight graphical management interface for Citrix Hypervisor servers. The following figure presents an overview of the Citrix Hypervisor architecture.

## Architecture Overview



Citrix Hypervisor is a type-1 hypervisor. The control domain (also called Domain 0 or dom0) is a secure, privileged Linux VM that runs the Citrix Hypervisor management tool stack known as XAPI. This Linux VM is based on a CentOS 7.5 distribution. Besides providing Citrix Hypervisor management functions, dom0 also runs the physical device drivers for networking, storage, and so on. The control domain can talk to the hypervisor to instruct it to start or stop guest VMs.

Virtual desktops run in the guest domain, sometimes referred as the user domain or domU, and request resources from the control domain. Hardware-assisted virtualization uses CPU virtualization extensions like Intel VT. The OS kernel doesn't need to be aware that it is running on a virtual machine. Quick Emulator (QEMU) is used for virtualizing the BIOS, the IDE, the graphic adapter, USB, the network adapter, and so on. With paravirtualization (PV), the OS kernel and device drivers are optimized to boost performance in the virtual machine. The following figure presents multitenancy features of Citrix Hypervisor.



Resources from NetApp HCI makes up the hardware layer, which includes compute, storage, network, GPUs, and so on.

### Compute

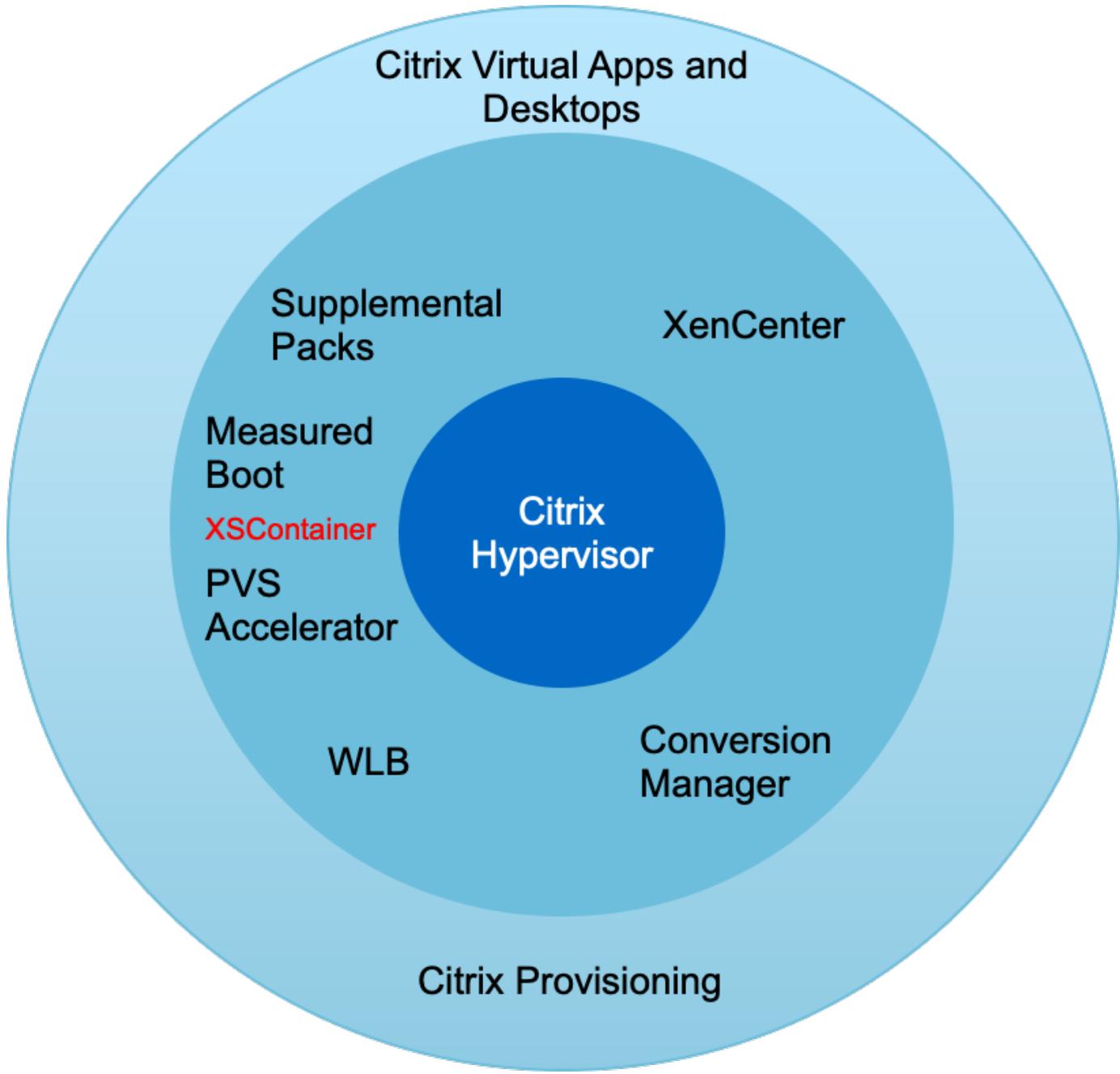
The CPU and memory details of NetApp HCI are covered in the previous section. However, this section focuses on how the compute node is utilized in the Citrix Hypervisor environment.

Each NetApp HCI compute node with Citrix Hypervisor installed is referred as a server. A pool of servers is managed as a resource pool (RP). The resource pools are created with similar model compute nodes to provide similar performance when the workload is moved from one node to another. A resource pool always contains a node designated as master, which exposes the management interface (for XenCenter and the CLI) and which can be routed to other member servers as necessary. When high availability is enabled, master re-election takes place if the master node goes down.

A resource pool can have up to 64 servers (soft limit). However, when clustering is enabled with the GFS2 shared storage resource, the number of servers is restricted to 16.

The resource pool picks a server for hosting the workload and can be migrated to other server using the Live Migration feature. To load balance across the resource pool, the optional WLB management

pack must be installed on Citrix Hypervisor.



Each tenant resource can be hosted on dedicated resource pools or can be differentiated with tags on the same resource pool. Custom values can be defined for operational and reporting purpose.

#### Storage

NetApp HCI compute nodes have local storage that is not recommended for the storage of any persistent data. Such data should be stored on an iSCSI volume created with NetApp HCI storage or can be on NFS datastore on NetApp AFF.

To use NetApp HCI storage, iSCSI must be enabled on Citrix Hypervisor servers. Using the iQN, register the initiators and create access groups on the Element management portal. Create the volumes

(remember to enable 512e block size support for LVM over iSCSI SR) and assign the account ID and access group.

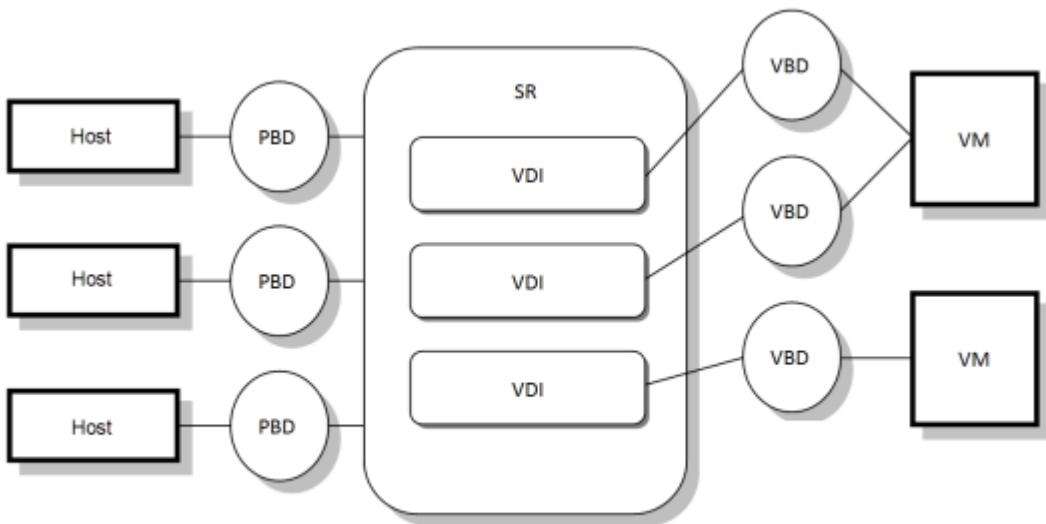


The iSCSI initiator can be customized using the following command on the CLI:

```
xe host-param-set uuid=valid_host_id other-config:iscsi_iqn=new_initiator_iqn
```

Multipathing of iSCSI is supported when multiple iSCSI NICs are configured. iSCSI configuration is performed using XenCenter or by using CLI commands like `iscsiadm` and `multipath`. This configuration can also be performed with the various Citrix Hypervisor CLI tools. For iSCSI multipath for single target storage arrays, see [CTX138429](#).

A storage repository (SR) is the storage target in which virtual machine (VM) virtual disk images (VDIs) are stored. A VDI is a storage abstraction that represents a virtual hard disk drive (HDD). The following figure depicts various Citrix Hypervisor storage objects.



The relationship between the SR and host is handled by a physical block device (PBD), which stores the configuration information required to connect and interact with the given storage target. Similarly, a virtual block device (VBD) maintains the mapping between VDIs and a VM. Apart from that, a VBD is also used for fine tuning the quality of service (QoS) and statistics for a given VDI. The following screenshot presents Citrix Hypervisor storage repository types.

 Choose the type of new storage ?

Type	
Name	<b>Virtual disk storage</b>
Location	<input checked="" type="radio"/> iSCSI <input type="radio"/> Hardware HBA <input type="radio"/> Software FCoE
	<b>Block based storage</b>
	<input type="radio"/> NFS <input type="radio"/> SMB/CIFS
	<b>File based storage</b>
	<input type="radio"/> Windows File Sharing (SMB/CIFS) <input type="radio"/> NFS ISO
	<b>ISO library</b>

**CITRIX**

[< Previous](#) [Next >](#) [Cancel](#)

With NetApp HCI, the following SR types can be created. The following table provides a comparison of features.

Feature	LVM over iSCSI	GFS2
Maximum virtual disk image size	2TiB	16TiB
Disk provisioning method	Thick Provisioned	Thin Provisioned
Read-caching support	No	Yes
Clustered pool support	No	Yes

Feature	LVM over iSCSI	GFS2
Known constraints	<ul style="list-style-type: none"> <li>• Read caching not supported</li> </ul>	<ul style="list-style-type: none"> <li>• VM migration with storage live migration is not supported for VMs whose VDIs are on a GFS2 SR. You also cannot migrate VDIs from another type of SR to a GFS2 SR.</li> <li>• Trim/unmap is not supported on GFS2 SRs.</li> <li>• Performance metrics are not available for GFS2 SRs and disks on these SRs.</li> <li>• Changed block tracking is not supported for VDIs stored on GFS2 SRs.</li> <li>• You cannot export VDIs that are greater than 2TiB as VHD or OVA/OVF. However, you can export VMs with VDIs larger than 2TiB in XVA format.</li> <li>• Clustered pools only support up to 16 hosts per pool.</li> </ul>

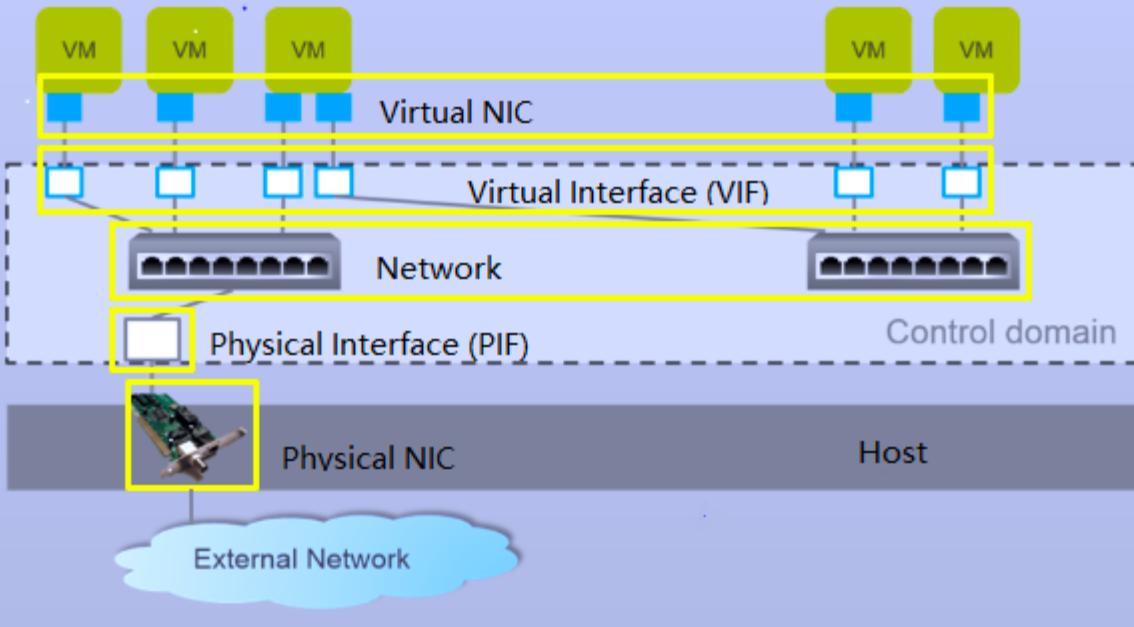
With the current features available in NetApp HCI, the Intellicache feature of Citrix Hypervisor is not of value to NetApp HCI customers. Intellicache improves performance for file-based storage systems by caching data in a local storage repository.

Read caching allows you to improve performance for certain storage repositories by caching data in server memory. GFS2 is the first iSCSI volume to support read caching.

#### Network

Citrix Hypervisor networking is based on Open vSwitch with support for OpenFlow. It supports fine grain security policies to control the traffic sent and receive from a VM. It also provides detailed visibility about the behavior and performance of all traffic sent in the virtual network environment. The following figure presents an overview of Citrix Hypervisor networking.

## Networking Overview



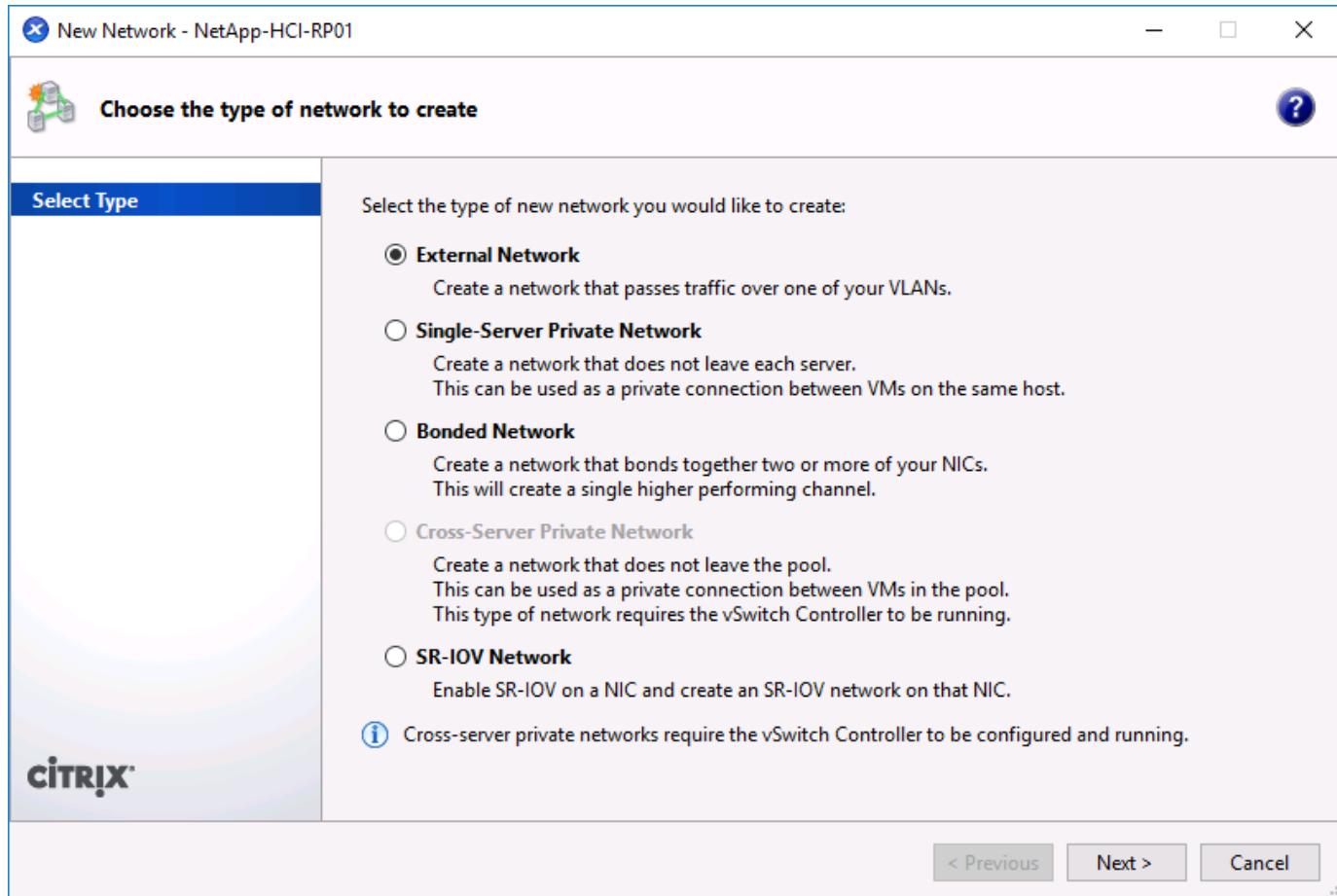
The physical interface (PIF) is associated with a NIC on the server. With Network HCI, up to six NICs are available for use. With the model, which only has two NICs, SR-IOV can be used to add more PIFs. The PIF acts as an uplink port to the virtual switch network. The virtual interface (VIF) connects to a NIC on virtual machines.

Various network options are available:

- An external network with VLANs
- A single server private network with no external connectivity
- Bonded network (active/active – aggregate throughput)
- Bonded network (active/passive – fault tolerant)
- Bonded network (LACP – load balancing based on source and destination IP and port)
- Bonded network (LACP – load balancing based on source and destination mac address)
- Cross-server private network in which the network does not leave the resource pool
- SR-IOV

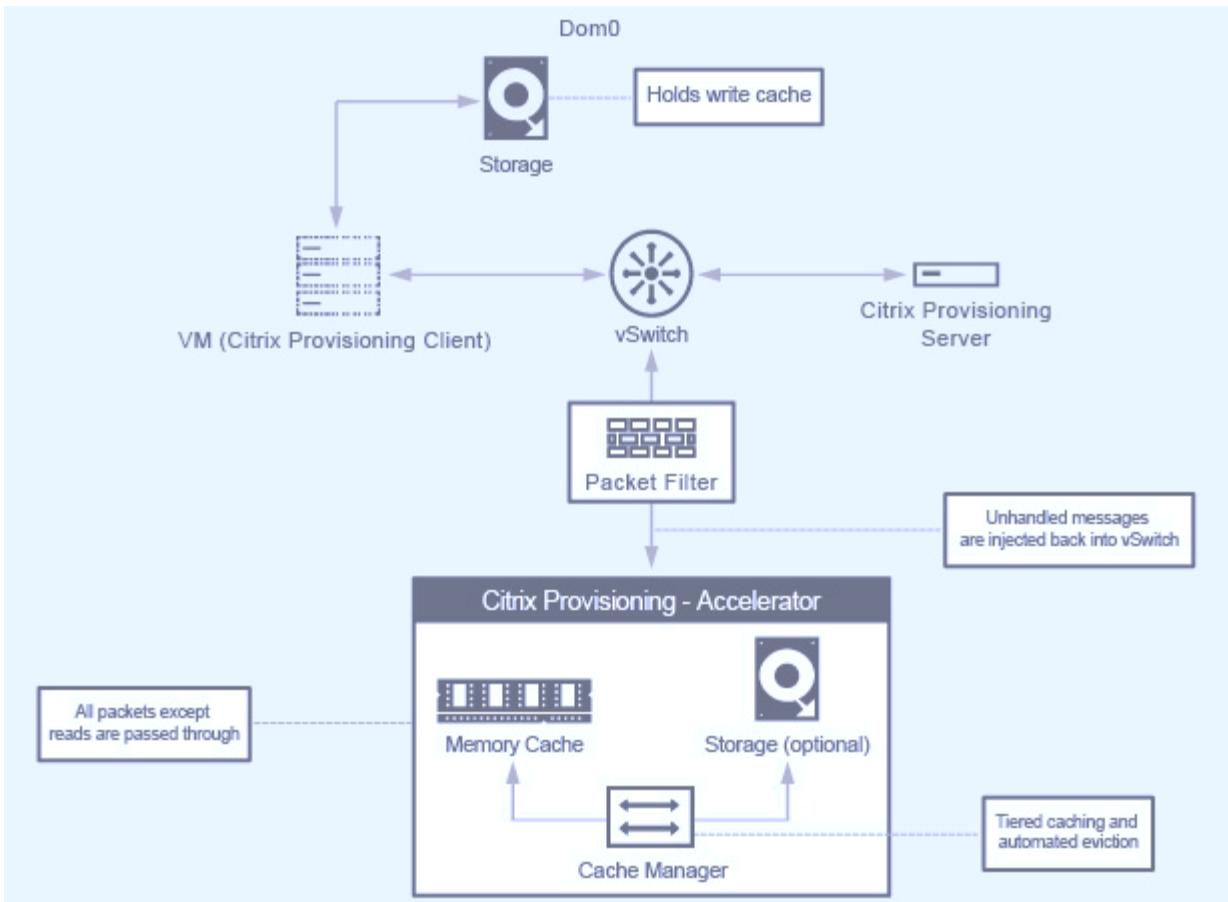
The network configuration created on the master server is replicated to other member servers. Therefore, when a new server is added to the resource pool, its network configuration is replicated from the master.

**i** You can only assign one IP address per VLAN per NIC. For iSCSI multipath, you must have multiple PIFs to assign an IP on the same subnet. For H615C, you can consider SR-IOV for iSCSI.



Because the network on Citrix Hypervisor is based on Open vSwitch, you can manage it with ovs-vsctl and ovs-appctl commands. It also supports NVGRE/VXLAN as an overlay solution for large scale-out environments.

When used with Citrix Provisioning (PVS), PVS Accelerator improves performance by caching Domain 0 memory or by combining memory and a local storage repository.



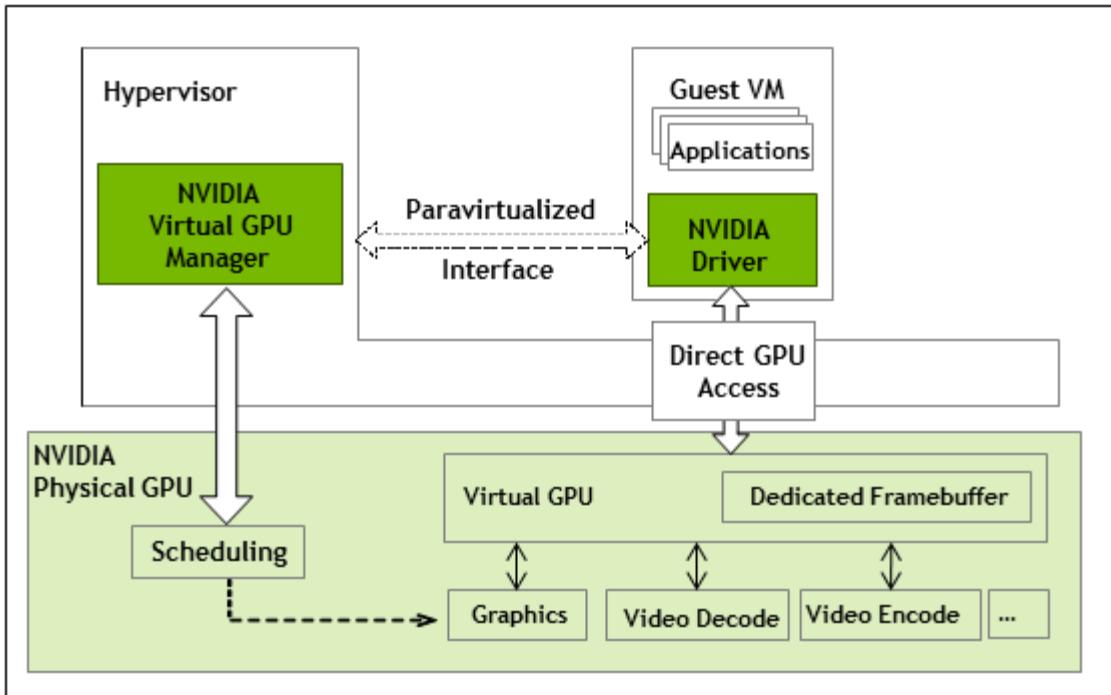
## GPU

Citrix Hypervisor was the first to deploy NVIDIA vGPUs, a virtualization platform for GPUs, enabling the sharing of GPU across multiple virtual machines. NetApp HCI H610C (with NVIDIA Tesla M10 cards) and H615C (with NVIDIA Tesla T4 cards) can provide GPU resources to virtual desktops, providing hardware acceleration to enhance the user experience.

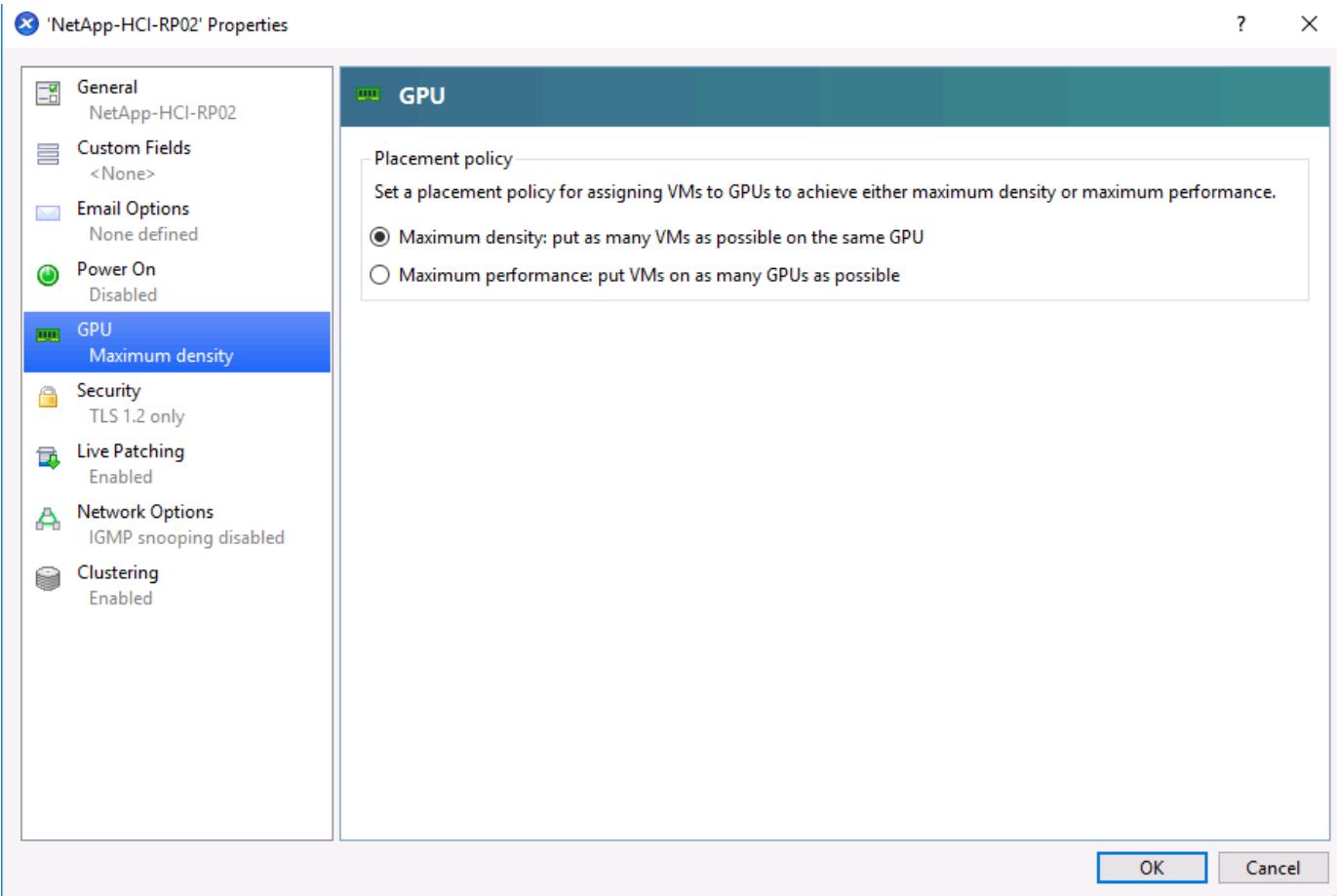
A NetApp HCI GPU can be consumed in a Citrix Hypervisor environment by using pass-through mode, where the whole GPU is presented to a single virtual machine, or it can be consumed using NVIDIA vGPU. Live migration of a VM with GPU pass through is not supported, and therefore NVIDIA vGPU is the preferred choice.

NVIDIA Virtual GPU Manager for Citrix Hypervisor can be deployed along with other management packs by using XenCenter or it can be installed using an SSH session with the server. The virtual GPU gets its own dedicated frame buffers, while sharing the streaming processors, encoder, decoder and so on. It can also be controlled using a scheduler.

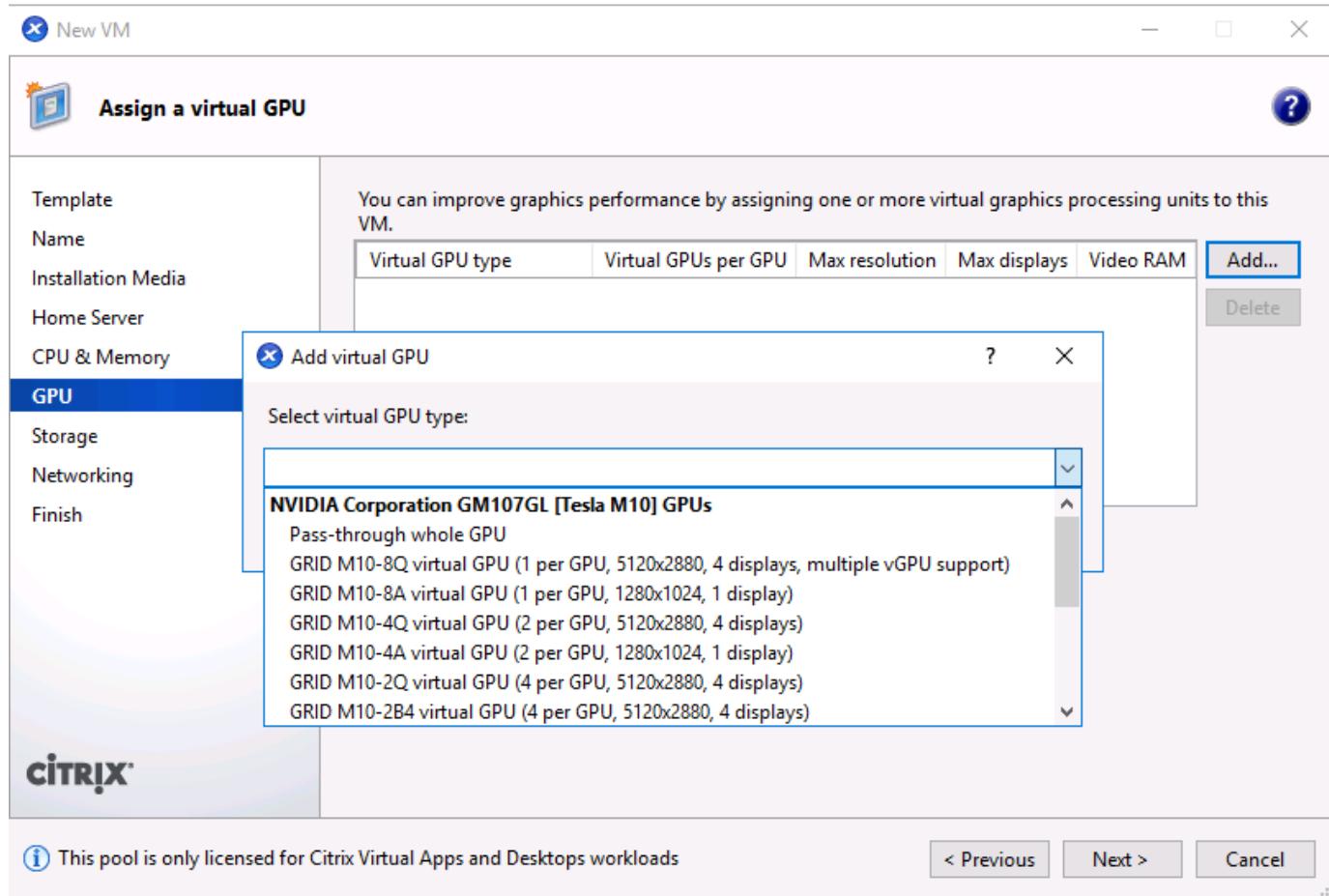
The H610C has two Tesla M10 graphic cards, each with 4 GPUs per card. Each GPU has 8GB of frame buffer memory with a total of 8 GPUs and 64GB of memory per server. H615C has three Tesla T4 cards, each with its own GPU and 16GB frame buffer memory with a total of 3 GPUs and 48GB of graphic memory per server. The following figure presents an overview of the NVIDIA vGPU architecture.



NVIDIA vGPU supports homogenous profiles for each GPU. The placement of virtual machines on a GPU is controlled by a policy that sets either maximum density or maximum performance in response to demand.



When creating a VM, you can set a virtual GPU profile. The vGPU profile you chose is based on the frame buffer memory level needed, the number of displays, and the resolution requirement. You can also set the purpose of a virtual machine, whether it be virtual apps (A), virtual desktops (B), a professional Quadro virtual workstation (Q), or compute workloads (C) for AI inferencing applications.



Independently from XenCenter, the CLI utility on the Citrix Hypervisor nvidia-smi can be used to troubleshoot and for monitoring the performance.

The NVIDIA driver on a virtual machine is required to access the virtual GPU. Typically, the hypervisor driver version and the VM guest driver should have the same vGPU release version. But, starting with vGPU release 10, the hypervisor can have the latest version while the VM driver can be the n-1 version.

## Security

Citrix Hypervisor supports authentication, authorization, and audit controls. Authentication is controlled by local accounts as well as by Active Directory. Users and groups can be assigned to roles that control permission to resources. Events and logging can be stored remotely in addition to on the local server.

Citrix Hypervisor supports Transport Layer Security (TLS) 1.2 to encrypt the traffic using SSL certificates.

Because most configuration is stored locally in an XML database, some of the contents, like SMB

passwords, are in clear text, so you must protect access to the hypervisor.

## Data Protection

Virtual machines can be exported as OVA files, which can be used to import them to other hypervisors. Virtual machines can also be exported in the native XVA format and imported to any other Citrix Hypervisor. For disaster recovery, this second option is also available along with storage-based replication handled by SnapMirror or native Element OS synchronous or asynchronous replication. With NetApp, HCI storage can also be paired with ONTAP storage for replication.

Storage-based snapshot and cloning features are available to provide crash-consistent image backups. Hypervisor-based snapshots can be used to provide point-in-time snapshots and can also be used as templates to provision new virtual machines.

## Resource Layer

### Compute

To host virtual apps and desktop resources, a connection to a hypervisor and resource details should be configured in Citrix Studio or with PowerShell. In the case of Citrix Hypervisor, a resource pool master node DNS or IP address is required. For a secure connection, use HTTPS with SSL certificates installed on the server. Resources are defined with selection the of storage resources and networks.

The screenshot shows the Citrix Studio interface for managing a resource pool named DS02. The left sidebar navigation includes options like Search, Machine Catalogs, AppDisks, Delivery Groups, Applications, Policies, Logging, Configuration (Administrators, Controllers), Hosting, Licensing, StoreFront, App-V Publishing, AppDNA, and Zones. The main central area displays a table of resources:

Name	Type	Address	State
Infra	Citrix Hypervisor®	http://172.21.146.40	Enabled
RP-01	Citrix Hypervisor®	http://E13U07.HCIEUC.Demo	Enabled
RP02 - GPU	Citrix Hypervisor®	http://172.21.146.38	Enabled
DS02			
NFS02			
RDSH			
RP03	Citrix Hypervisor®	http://172.21.146.38	Enabled
Demo			
H410C			

Below the table, a "Details - DS02" pane shows the configuration for the DS02 resource pool:

Resources		Storage	
Name:	DS02	Standard storage	XSE-DS02
Networks:	VLAN-3403 VLAN-3404 VLAN-3405 VLAN-3406	Personal vDisk storage	XSE-DS02
Graphics virtualization:	On	Temporary storage	Local storage on e13u07.hcieu.demo
GPU group:	Group of NVIDIA Corporation GM107GL [Tesla M10]... GRID M10-8Q (7616MB video RAM per VM)	IntelliCache:	Enabled

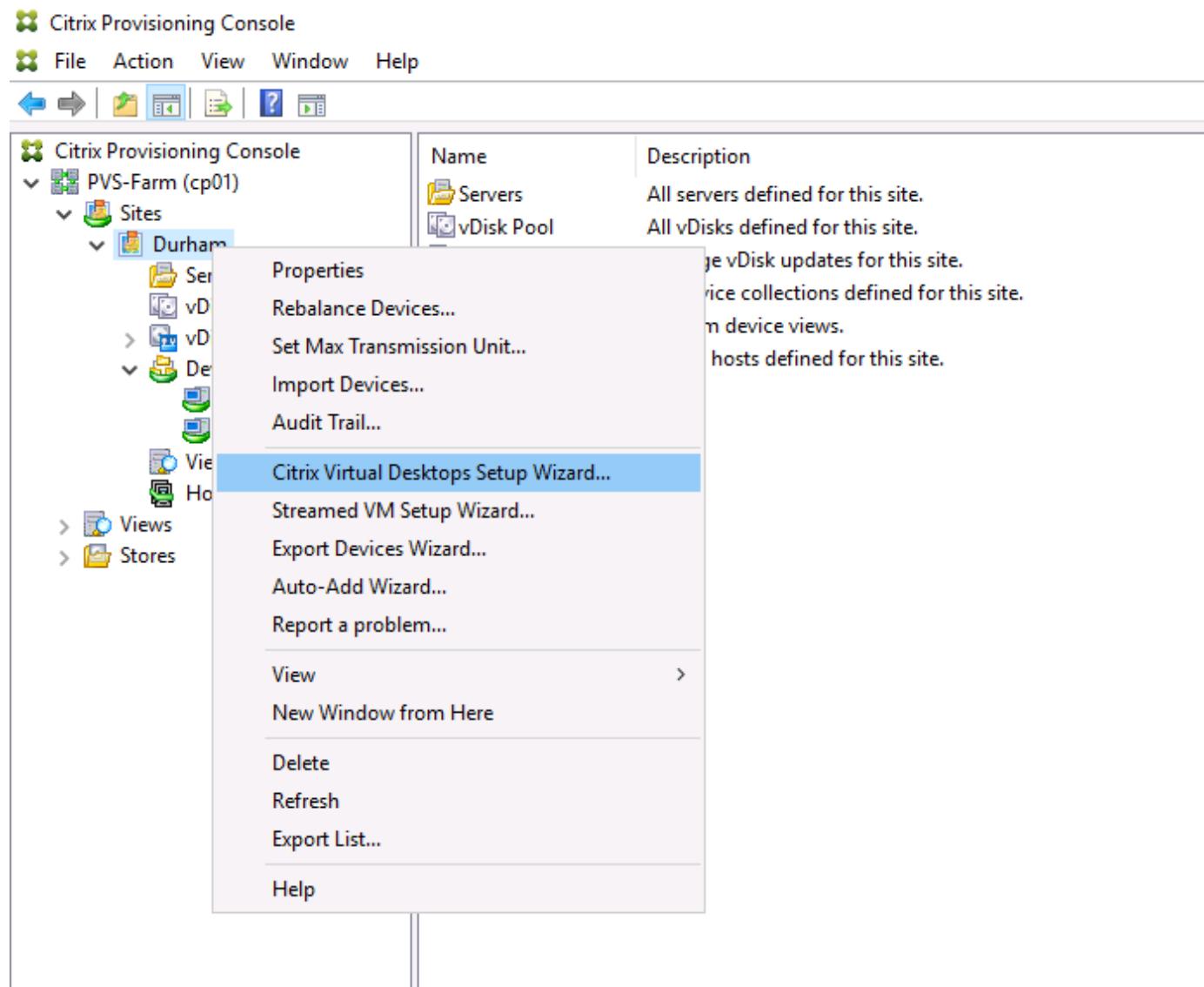
The right-hand Actions pane for DS02 lists the following options:

- Add Connection and Resources
- View
- Refresh
- Help
- Edit Storage
- Delete Resources
- Rename Resources
- Test Resources
- Help

When additional compute capacity is required, a hypervisor server can be added to existing resource pool. Whenever you add a new resource pool and you need to make it available for hosting virtual apps and desktops, you must define a new connection.

A site is where the SQL database resides and is known as the primary zone. Additional zones are added to address users in different geographic locations to provide better response time by hosting on local resources. A satellite zone is a remote zone that only has hypervisor components to host virtual apps or desktops with optional delivery controllers.

Citrix Provisioning also uses the connection and resources information when using the Citrix Virtual Desktops Setup Wizard.



## Storage

The storage repository for Virtual Apps and Desktops is controlled using the connection and resources covered in the section [Compute](#). When you define the resource, you have the option to pick the shared storage and enable Intellicache with Citrix Hypervisor.

**Studio**

✓ Connection

**Storage Management**

Storage Selection

Network

Summary

**Storage Management**

Configure virtual machine storage resources for this connection.

Select an optimization method for available site storage.

Use storage **shared** by hypervisors

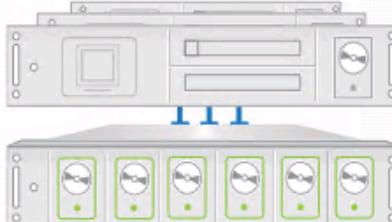
Optimize **temporary** data on available local storage

Use storage **local** to the hypervisor

Manage personal data centrally on shared storage

**Optimization technology (optional):**

Use intellicache to reduce load on the shared storage device



Back    **Next**    Cancel

There is also an option to pick resources for the OS, the personal vDisk, and temporary data. When multiple resources are selected, Citrix Virtual Apps and Desktops automatically spreads the load. In a multitenant environment, a dedicated resource selection can be made for each tenant resource.

**Studio**

- ✓ Connection
- ✓ Storage Management
- Storage Selection**
- Network
- Summary

**Storage Selection**

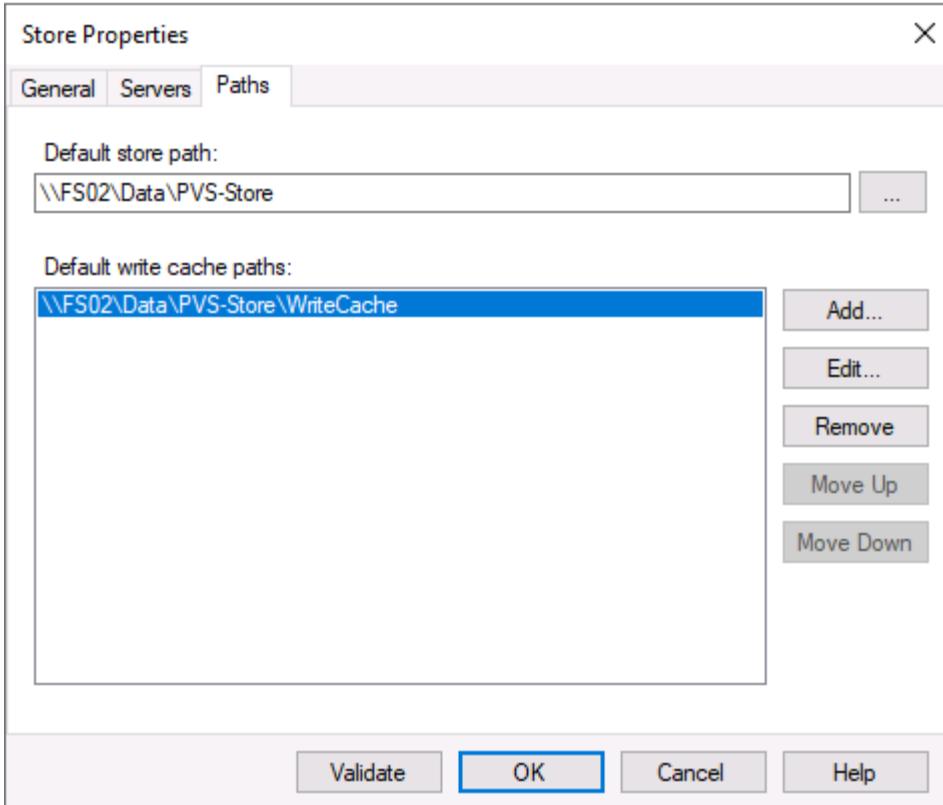
When using shared storage, you must select the type of data to store on each shared storage device; machine operating system data, personal user data, and if not storing temporary data locally, temporary data. At least one device must be selected for each data type.

Select data storage locations:

Name	OS	Personal vDisk	Temporary
XSE-DS02	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
NFS02	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
VDI02	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

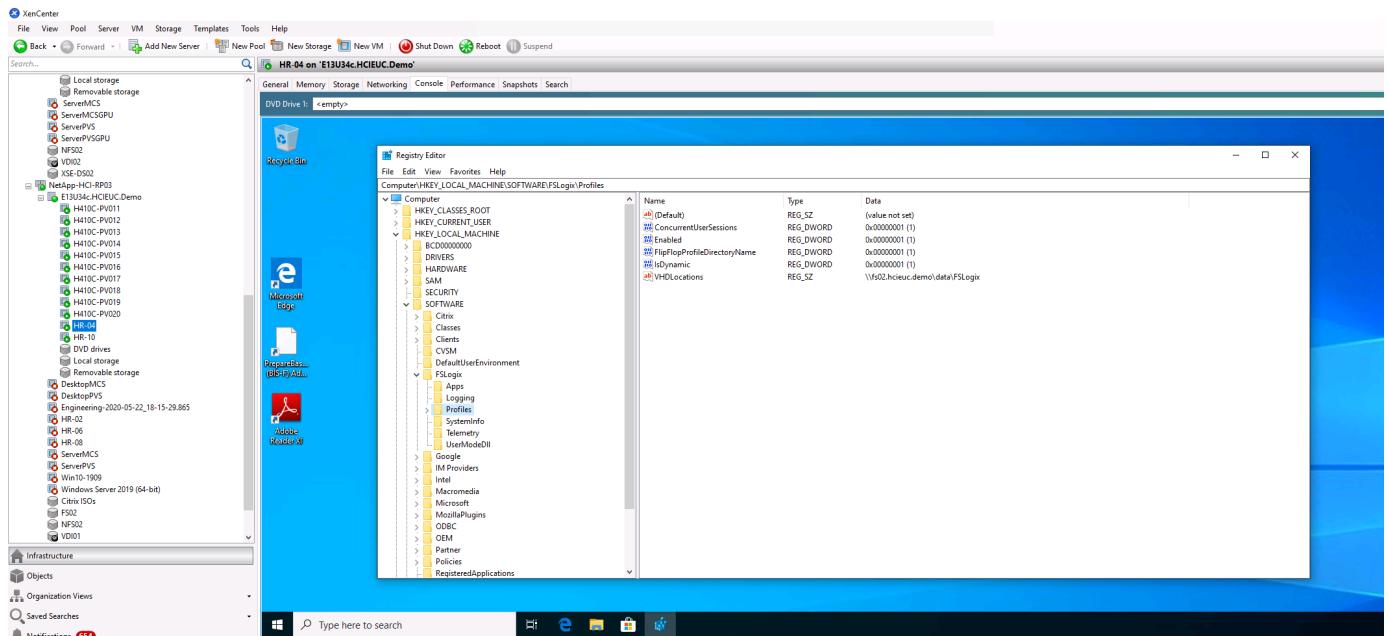
**Back** **Next** **Cancel**

Citrix Provisioning requires an SMB file share to host the vDisks for the devices. We recommend hosting this SMB share on a FlexGroup volume to improve availability, performance, and capacity scaling.



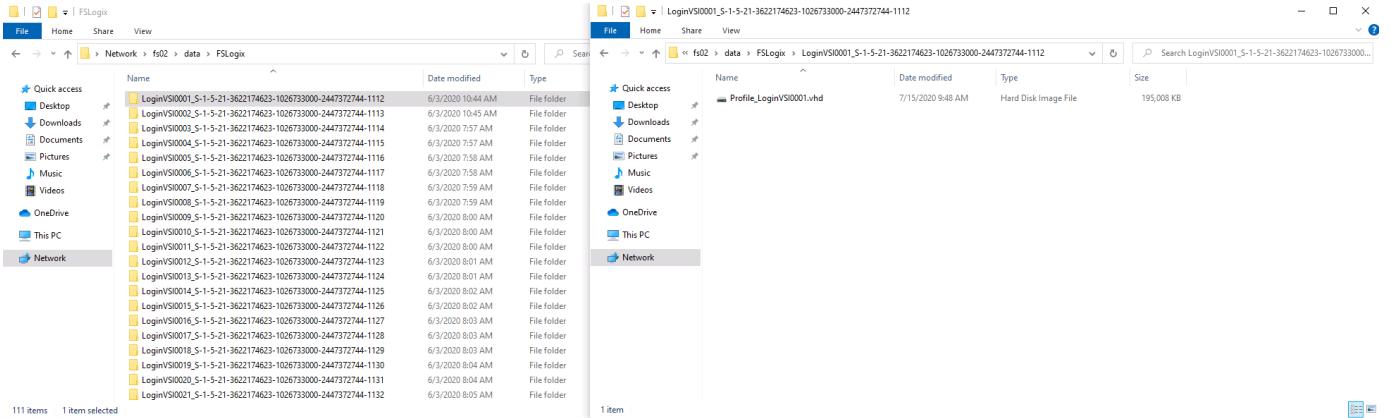
## FSLogix

FSLogix allows users to have a persistent experience even in non-persistent environments like pooled desktop deployment scenarios. It optimizes file I/O between the virtual desktops and the SMB file store and reduces login time. A native (local) profile experience minimizes the tasks required on the master image to set up user profiles.



FSLogix keeps user settings and personal data in its own container (VHD file). The SMB file share to store the FSLogix user profile container is configured on a registry that is controlled by group policy

object. Citrix User Profile Management can be used along with FSLogix to support concurrent sessions with virtual desktops at the same time on virtual apps.



This figure shows the content of the FSLogix SMB location. Note that we switched the directory name to show the username before the security identifier (sid).

## Network

Virtual Apps and Desktops require a connection and resources to host, as covered in the section [Compute](#). When defining the resource, pick the VLANs that must be associated with the resource. During machine catalog deployment, you are prompted to associate the VM NIC to the corresponding network.

## Add Connection and Resources

Studio

Network

Name for these resources: All

The resources name helps identify this storage and network combination in Studio.

Select one or more networks for the virtual machines to use:

Name
VLAN-3403
VLAN-3404
VLAN-3405
VLAN-3406

Do you want to use graphics virtualization?

No

Yes

Select a GPU type and group:

GRID M10-8Q (Group of NVIDIA Corporation GM10)

7616MB video RAM per virtual machine.  
This group allocates GPU resources on-demand.

Back    Next    Cancel

## GPU

As indicated in the previous section, when you determine whether the hypervisor server has a GPU resource, you are prompted to enable graphics virtualization and pick the vGPU profile.

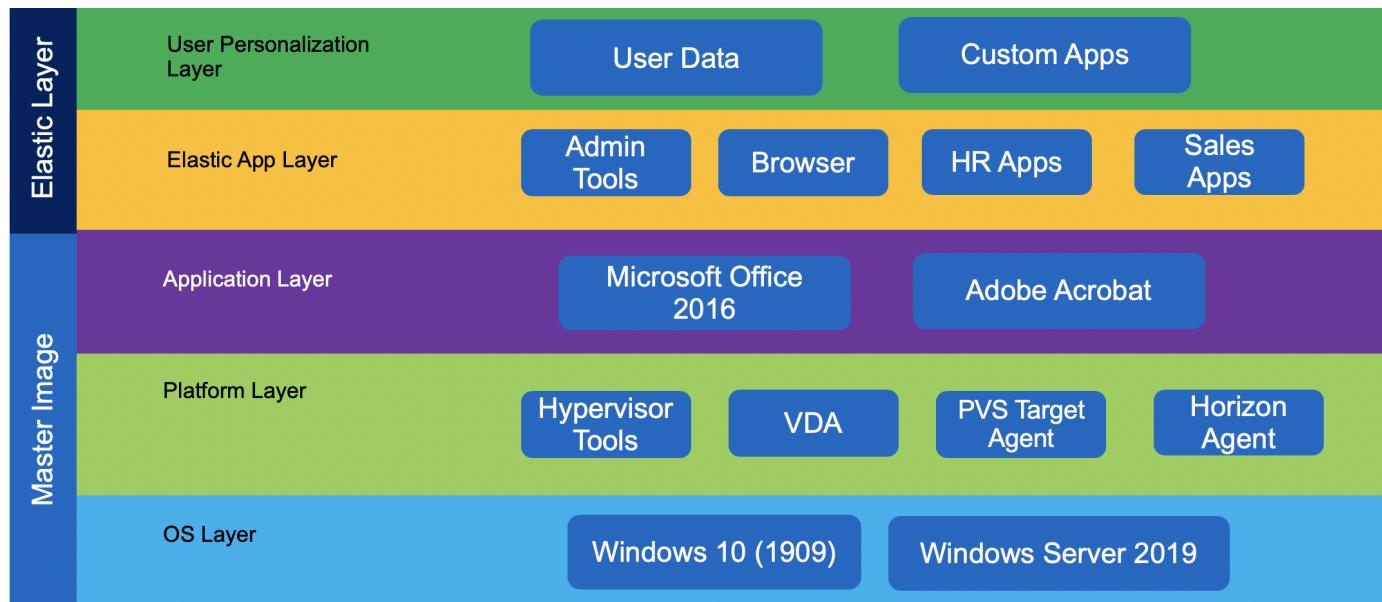
## Control Layer

### App Layering

Layering is a technology to separate the OS, applications, and user settings and data, each hosted on its own virtual disks or group of virtual disks. These components are then merged with the OS as if they were all on same machine image. Users can continue with their work without any additional training. Layers make it easy to assign, patch, and update. A layer is simply a container for file system and registry entries unique to that layer.

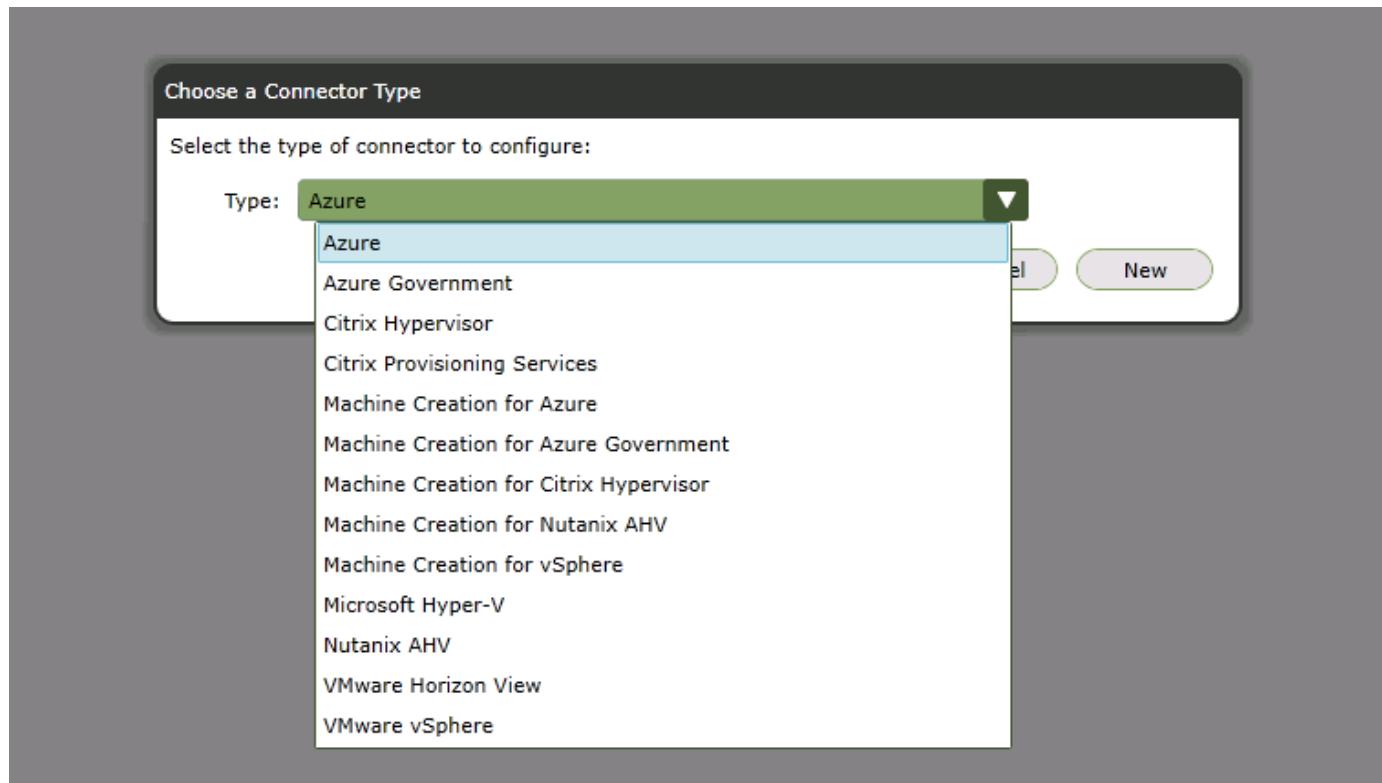
Citrix App Layering allows you to manage master images for Citrix Virtual Apps and Desktops as well as for the VMware Horizon environment. App layering also allows you to provision applications to users on demand; these apps are attached while logging in. The user personalization layer allows users to install custom apps and store the data on their dedicated layer. Therefore, you can have a personal desktop experience even when you are using a shared desktop model.

Citrix App Layering creates merged layers to create the master image and does not have any additional performance penalty. With Elastic Layers, the user login time increases.



Citrix App Layering uses a single virtual appliance to manage the layers and hands off using the image and application delivery to another platform. The Citrix Enterprise Layer Manager (ELM) portal must be accessed from web browsers that supports Microsoft Silverlight 4.0. A cloud-based management portal is also available if local management interface requirements cannot be met.

Initial configuration includes the creation of platform connectors of two types; the first is a platform connector for layer creation, and the other is a platform connector for image publishing.



A layer repository is an SMB file share configured with ELM where Elastic Layers are stored. A layer work disk is where all the layers created by ELM are stored. The disk is attached to the appliance and is consumed as a block device on which a local Linux file system is used. The layer work disk is used as scratch area where the layer images are put together. After the master image is created, it is pushed to the provisioning platform.

The screenshot shows the Citrix ELM System interface. The top navigation bar includes tabs for Images, Layers, Users, and System, with System being the active tab. Below the navigation bar is a toolbar with icons for refresh, search, and other system functions. A sub-navigation bar below the toolbar has tabs for Manage Appliance, Connectors, User Layer Storage Locations, and Settings and Configuration, with User Layer Storage Locations being the active tab. The main content area is titled "App Layering Services" and contains a table with two rows. The table columns are Name, Status, and Local Storage. The first row shows "Management Service" as Running with no visible storage information. The second row shows "Layering Service" as Running, with a progress bar indicating "216.9 GB free of 299.8 GB".

When there are common or shared files on multiple layers, by default the high priority layer ID wins. Layer ID is incremented whenever a new layer is created. If you would like to control layer priority, use the support utility on the [Citrix LayerPriority Utility page](#).

ELM also supports authentication and role-based access control with integration with Active Directory and LDAP.

#### Delivery Controller

The delivery controller is responsible for user access, brokering, and optimizing connections. It also provides Machine Creation Services (MCS) for provisioning virtual machines in an effective manner. At least one delivery controller is required per site, and typically additional controllers are added for redundancy and scalability.

Virtual desktop agents (VDA) must register with the delivery controller to make it available to users. During VDA deployment, the initial registration options can be provided manually through GPO based

on the Active Directory OU. This process can also be handled with MCS.

Delivery controllers keep a local host cache in case a controller loses its connectivity to database server.

## Database

A SQL Server database is used for site configuration data, logging, and monitoring. There should be at least one database per site. To provide high availability, use Microsoft SQL Server features like AlwaysOn availability groups, database mirroring, or SQL clustering. At a minimum, consider using the hypervisor high-availability feature for a SQL VM.

Even though the controller has a local host cache, it doesn't affect any existing connections. However, for new connections, NetApp recommends database connectivity.

## Director

Citrix Director provides a monitoring solution for Citrix Virtual Apps and Desktops. Help Desk users can search for a specific user session and get a complete picture for troubleshooting. When Citrix Virtual Apps and Desktop Resources are hosted on Citrix Hypervisor, Help Desk users have the option to launch a console session from the Director portal.

The screenshot shows the Citrix Director web interface. The top navigation bar includes links for Dashboard, Trends, Filters, Alerts, Applications, Configuration, and Analytics. The main header displays "Admin Connected" and the location "HR [HCIEUC\HR-04]". On the left, the "Activity Manager" sidebar lists "Applications" and "Processes". Under "Applications", there is a single entry for "Microsoft Edge" with the status "Running". The main content area is divided into two sections: "Machine Details" and "Session Details".

**Machine Details:**

Machine name	HR\DCUHR-04
Display name	HR
Delivery Group	HR
Machine Catalog	No
Remote PC Access	No
Site name	Durham
Registration state	Registered
OS type	Windows 10
Address type	Static
Machine IP	172.21.148.140
Organizational unit	CN=HR-04\OU=berg_C=HCIEUC\DC=Demo
VDA version	2003.0.2.50506
Hosting Connection Name	RPO3
Host Name	E13U94c.HCIEUC.Demo
VM name	HR-04_Console
vCPU	2
Memory	4092 MB
Hard disk	49 GB
Avg. disk sec/transfer	0
Current disk queue length	0
VDA hotfixes	- none -

**Session Details:**

Session State	Active
Application State	Desktop
Anonymous	No
Time in state	533 hours 51 minutes
Endpoint name	n/a
Endpoint IP	172.21.0.1
Connection type	ICP
Protocol	n/a
Citrix Workspace App Version	n/a
ICA RTT	n/a
ICA Latency	n/a
Launched via	n/a
Connected via	172.21.0.1

## License

The Citrix license server manages the repository of all Citrix licenses so that licenses can be easily consumed by applications. The license server provides a management portal for advanced troubleshooting. For regular operations, Citrix Studio can also be used.

## Provisioning Services

Provisioning services enable the provisioning of desktop images even to bare metal workstations by using PXE boot. An ISO or CDROM-based boot option is also available to support environments in which network changes aren't allowed for PXE boot. The DHCP server options that we used in our lab is provided in the following figure. CP.HCIEUC.Demo and PVS.HCIEUC.Demo are the load balancer virtual IPs that point to two provisioning servers. When option 011 and 017 are available, options 066 and 067 are ignored.

The screenshot shows the Citrix Studio interface for managing a DHCP server. On the left, a tree view shows the structure: 'DHCP' > 'jumphost01.h' > 'IPv4'. Under 'IPv4', there are several scopes and their properties. One scope is expanded to show its options. The table below lists the options and their values:

Option Name	Vendor	Value	Policy Name
006 DNS Servers	Standard	172.21.146.10, 172.21.146.11	None
011 Resource Location Servers	Standard	172.21.146.67, 172.21.146.66, 172.21.146.195, 172.21.146.196	None
015 DNS Domain Name	Standard	HCIEUC.Demo	None
017 Root Path	Standard	pvs:[cp.hcieuc.demo]:17:6910	None
042 NTP Servers	Standard	10.54.17.30	None
066 Boot Server Host Name	Standard	PVS.HCIEUC.Demo	None
067 Bootfile Name	Standard	PvsNbpX64.efi	None

The high-level operation to create a machine catalog based on Citrix provisioning is as follows:

1. On the template VM, install the target agent before installing VDA.
2. Assign an additional disk for caching and format it with MBR. This step is optional. At least verify that the PVS store has a write cache path.
3. Start the Target Image Wizard and respond to its questions. Remember to provide a single Citrix Provisioning server when prompted.
4. The device boots with PXE or with ISO. The Imaging wizard continues to capture the image.
5. Select the vDisk that is created and right click to select Load Balancing and enable it.
6. For vDisk Properties, change the access mode to Standard and the Cache Type to Cache in Device RAM with Overflow on Hard Disk.
7. Right click on the site to pick the Create Virtual Desktops Setup Wizard and respond to the questions.

## Studio

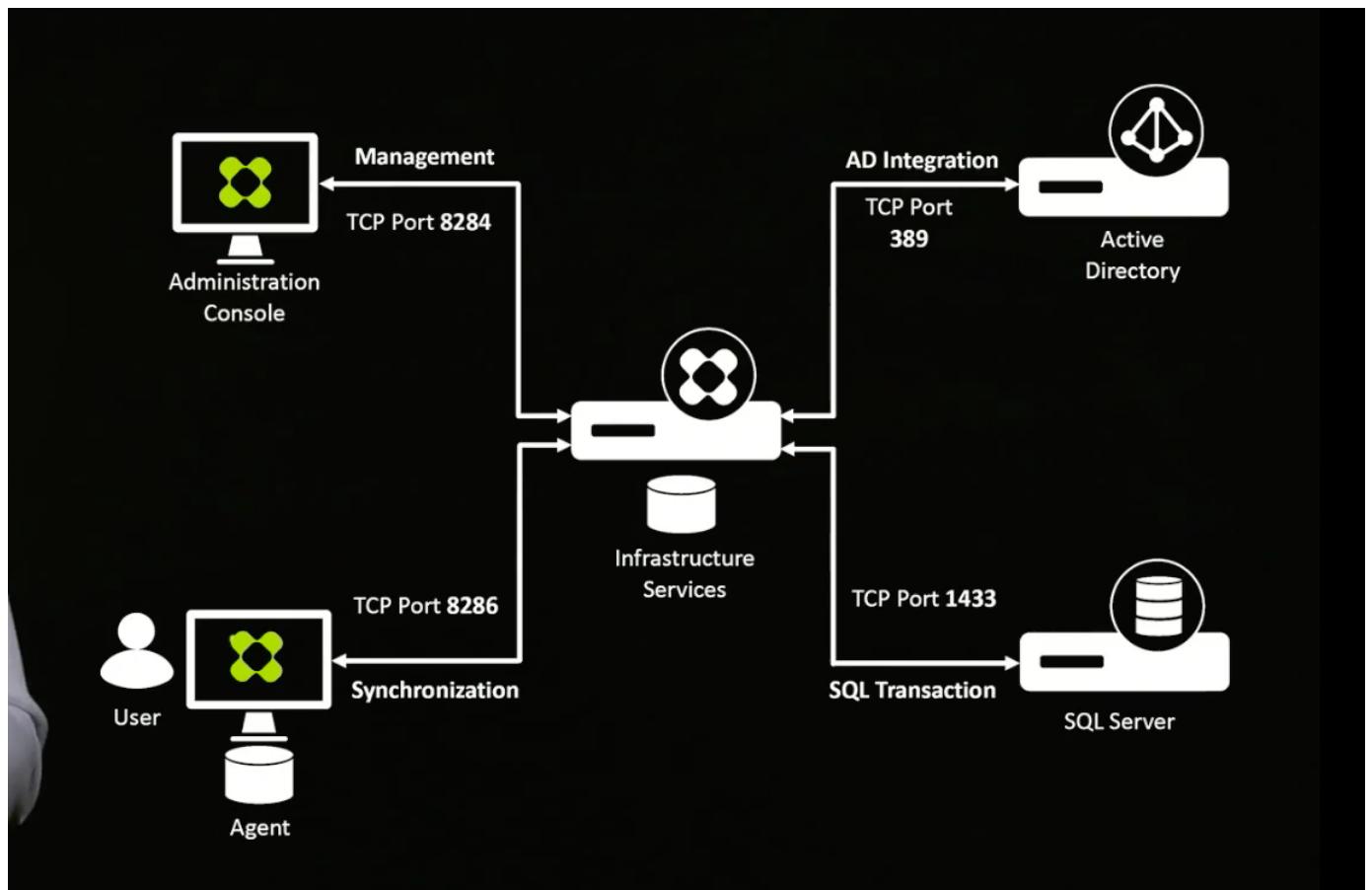
Citrix Studio is the central management console used by the Citrix Virtual Apps and Desktops. The management of machine catalogs, delivery groups, applications, policies, and the configuration of resource hosting, licenses, zones, roles, and scopes are handled by the Citrix Studio. Citrix Studio also provides PowerShell snap-ins to manage Citrix Virtual Apps and Desktops.

## Workspace Environment Management

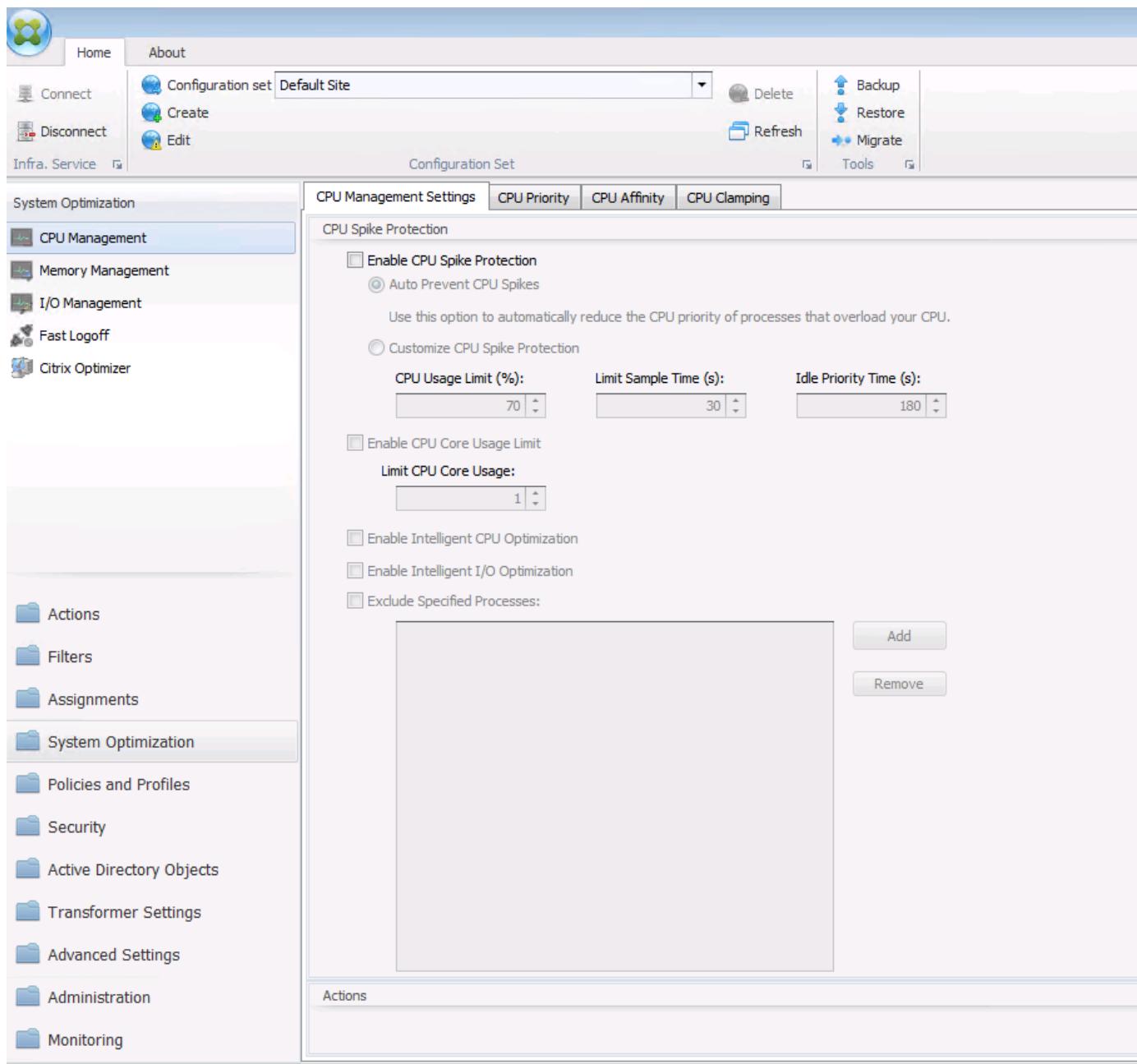
Workspace Environment Management (WEM) provides intelligent resource management and profile management technologies to deliver the best possible performance, desktop login, and application response times for Citrix Virtual Apps and Desktops in a software-only, driver-free solution.

WEM requires a SQL database to store configuration information. To provide high availability to

infrastructure services, multiple instances are used with a load balancer virtual server connection. The following figure depicts the WEM architecture.



The following figure depicts the WEM console.



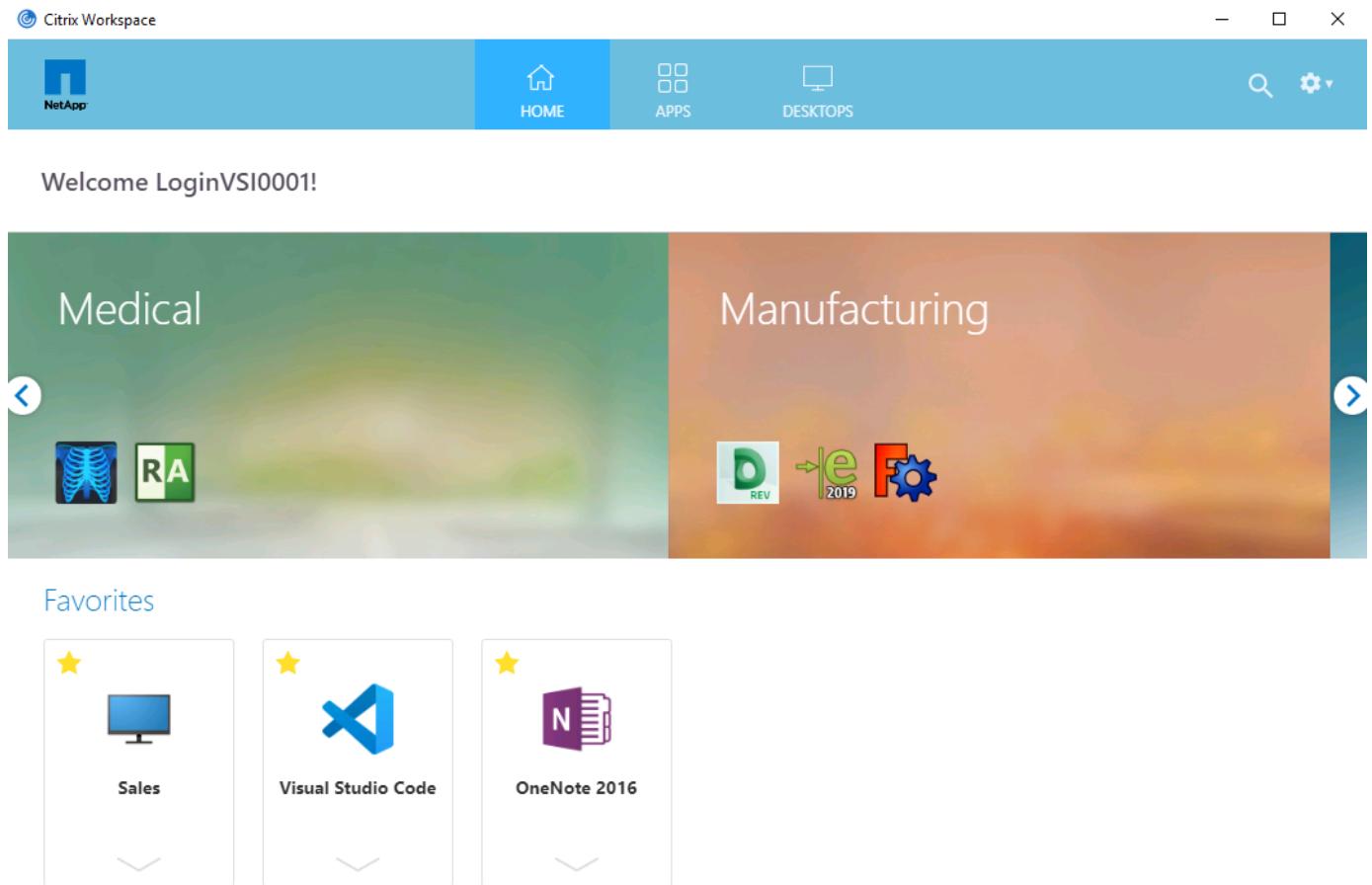
The key features of WEM are as follows:

- The ability to control resources for certain tasks or applications
- An easy interface to manage windows icons, network drives, start menu items, and so on
- The ability to reuse an old machine and manage it as a thin client
- Role-based access control
- Control policies based on various filters

## Access Layer

## StoreFront

StoreFront consolidates resources published from multiple delivery controllers and presents unique items to users. Users connect to StoreFront and hides the infrastructure changes on the backend.



Users connect to StoreFront with the Citrix Workspace application or with a web browser. The user experience remains the same. An administrator can manage StoreFront using Microsoft Management Console. The StoreFront portal can be customized to meet customer branding demands. Applications can be grouped into categories to promote new applications. Desktops and applications can be marked as favorites for easy access. Administrators can also use tags for ease of troubleshooting and to keep track of resources in multitenant environments. The following screenshot depicts featured app groups.

Edit Receiver for Web site - /Citrix/SFWeb

**StoreFront**

**Customize Appearance**

**Featured App Groups**

**Authentication Methods**

**Website Shortcuts**

**Deploy Citrix Receiver/ Workspace app**

**Session Settings**

**Workspace Control**

**Client Interface Settings**

**Advanced Settings**

**Manage Featured App Groups**

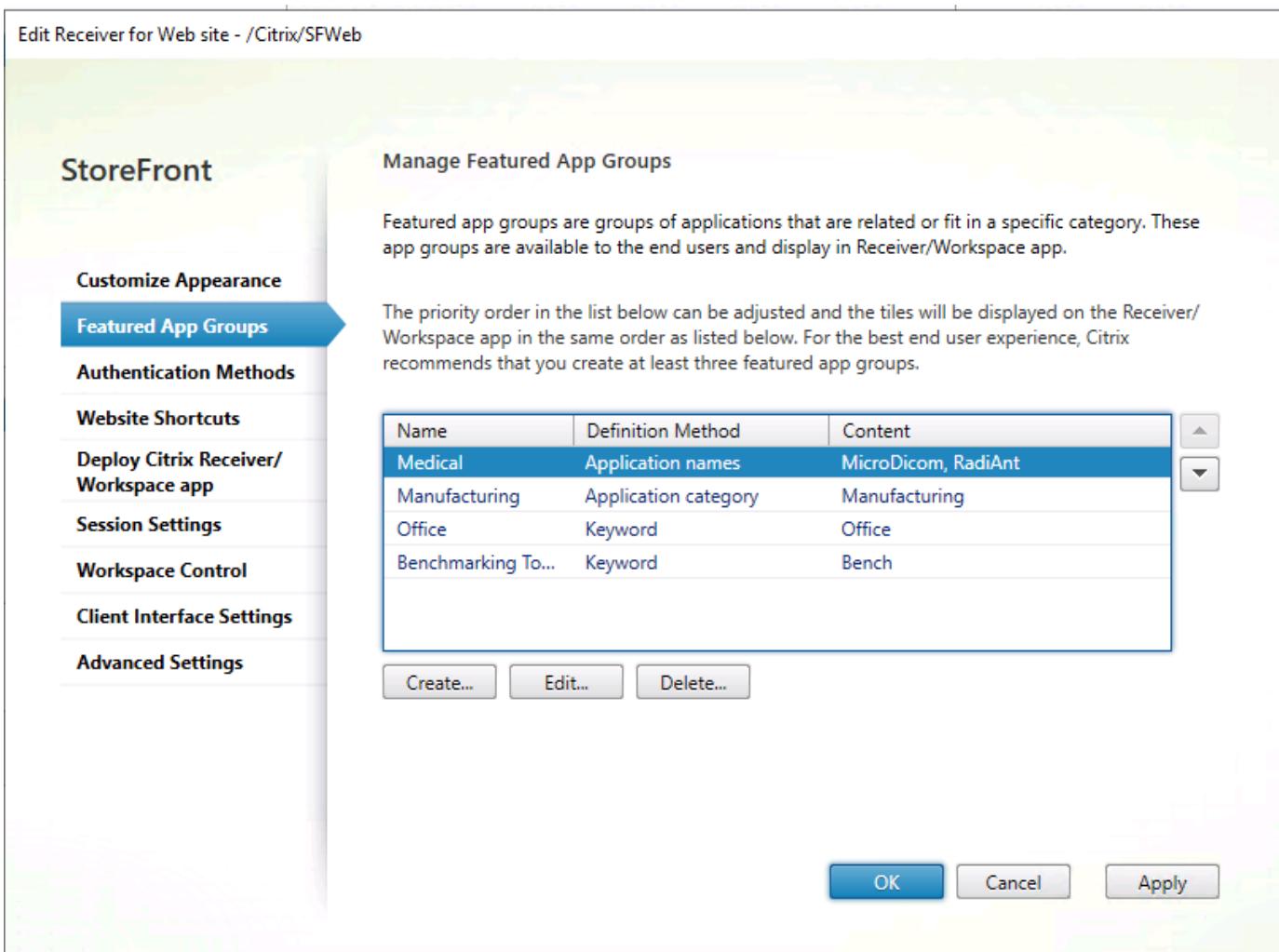
Featured app groups are groups of applications that are related or fit in a specific category. These app groups are available to the end users and display in Receiver/Workspace app.

The priority order in the list below can be adjusted and the tiles will be displayed on the Receiver/ Workspace app in the same order as listed below. For the best end user experience, Citrix recommends that you create at least three featured app groups.

Name	Definition Method	Content
Medical	Application names	MicroDicom, RadiAnt
Manufacturing	Application category	Manufacturing
Office	Keyword	Office
Benchmarking To...	Keyword	Bench

**Create...** **Edit...** **Delete...**

**OK** **Cancel** **Apply**



## Unified Gateway

To provide secure access to Citrix Virtual Apps and Desktops from the public internet to resources hosted behind a corporate firewall, Unified Gateway is deployed in a DMZ network. Unified Gateway provides access to multiple services like an SSL VPN, a reverse proxy to intranet resources, load balancer and so on by using a single IP address or URL.

Users have the same experience whether they are accessing the resources internally or externally to an organization. Application Delivery Controller (ADC) provides enhanced networking features for Virtual Apps and Desktops, and HDX Network Insights enhances HDX monitoring information with Citrix Director.

## User Layer

Citrix Virtual Apps and Desktops enables users to access their workspace environment from anywhere with internet access and from any device with a web browser that has HTML5 support or with the Citrix Workspace application.

Users can be categorized as task workers, office workers, knowledge workers, and power users. Task workers primarily use predefined applications throughout the day for their work. Hosted Windows

Apps can serve their needs. Office workers require desktop interfaces that run office applications, a web browser, and so on. Typically, they are not allowed to install applications on their workspace. They are best served by either a shared desktop with multi-session on server OS or with pooled desktops.

Knowledge workers typically require a desktop experience working with multiple applications simultaneously and must be able to persist the applications that they installed on their workspace. Static desktops (also referred to as personal desktops) allow this. Power users typically work on graphic-intensive applications or other applications that require more hardware resources. Static desktops created with an appropriate master image address the needs of power users.

## NetApp Value

### Data Fabric

Infrastructure built with the data fabric powered by NetApp allows you to migrate data or perform disaster recovery from one site to another (including the cloud). The data in Citrix Virtual Apps and Desktops can be categorized as follows:

- Infrastructure components
- Machine images
- Applications
- User profiles
- User data

Based on your needs, sites can be configured as active/active or active/passive. Infrastructure components can be on-premises or in the cloud and accessed as a service. VM templates must be distributed to each site to provision desktop and application pools. Application layers, user profiles, and data are stored in SMB file shares that must be available on each site.

You can create a global namespace using Azure NetApp Files, NetApp Cloud Volumes ONTAP, and FlexGroup volumes at the location where most of your users reside. Other locations can use Global FileCache to cache the content locally on a file server. If Citrix ShareFile is preferred, NetApp StorageGRID provides high-performance, S3-compatible storage to host data on-premises with NAS gateway access.

### Cloud Insights

Cloud Insights allows you to monitor, optimize, and troubleshoot resources deployed in the public cloud as well as on private datacenters.

Cloud Insights helps you in the following ways:

- **Reduce the mean time to resolution by as much as 90%.** Stop lengthy log hunting and failing to manually correlate infrastructure; use our dynamic topology and correlation analysis to pinpoint

the problem area immediately.

- **Reduce cloud infrastructure costs by an average of 33%.** Remove inefficiencies by identifying abandoned and unused resources and right-size workloads to optimized performance and cost tiers.
- **Prevent as much as 80% of cloud issues from affecting end users.** Stop searching through vast amounts of data to find the relevant item by using advanced analytics and machine learning to identify issues before they become critical outages.

## Appendix iSCSI Device Configuration

Edit the multipath configuration file at `/etc/multipath.conf` as follows:

```
# This is a basic configuration file with some examples, for device mapper
# multipath.
## Use user friendly names, instead of using WWIDs as names.
defaults {
    user_friendly_names yes
}
##
devices {
    device {
        vendor "SolidFir"
        product "SSD SAN"
        path_grouping_policy multibus path_selector "round-robin 0"
        path_checker tur hardware_handler "0"
        fallback immediate rr_weight uniform rr_min_io 10 rr_min_io_rq 10
        features "0"
        no_path_retry 24
        prio const
    }
}
## Device black list
## Enter devices you do NOT want to be controlled by multipathd
## Example: internal drives
#blacklist {
#}
```

## Where to Find Additional Information

To learn more about the information that is described in this document, review the following documents and/or websites:

- NetApp Cloud Central  
<https://cloud.netapp.com/home>

- NetApp Element Software Configuration for Linux  
<https://www.netapp.com/us/media/tr-4639.pdf>
- NetApp Product Documentation  
<https://docs.netapp.com>
- Citrix Security Recommendations  
[https://www.citrix.com/content/dam/citrix/en\\_us/documents/white-paper/security-recommendations-when-deploying-citrix-xenserver.pdf](https://www.citrix.com/content/dam/citrix/en_us/documents/white-paper/security-recommendations-when-deploying-citrix-xenserver.pdf)
- Citrix Monitoring in Healthcare Environment with Goliath  
<https://goliathtechnologies.com/webinar/on-demand/>
- Citrix User Profile and FSLogix Integration  
<https://youtu.be/dFpWdXIytJI>
- Citrix App Layering Login VSI Test Results  
<https://youtu.be/rWF5e84To4E>
- Citrix App Layering FAQ  
<https://www.citrix.com/blogs/2020/03/02/citrix-tips-citrix-app-layering-webinar-qa/>
- Citrix App Layering Reference Architecture  
<https://docs.citrix.com/en-us/tech-zone/design/reference-architectures/app-layering.html>
- Citrix App Layering  
<https://docs.citrix.com/en-us/citrix-app-layering/4/app-layering.pdf>
- Multi-session write back to FSLogix Profile Container  
[https://www.deyda.net/index.php/en/2020/03/27/citrix-virtual-apps-and-desktops-wem-2003-is-released/-MultiSession\\_writeback\\_for\\_FSLogix\\_Profile\\_Container](https://www.deyda.net/index.php/en/2020/03/27/citrix-virtual-apps-and-desktops-wem-2003-is-released/-MultiSession_writeback_for_FSLogix_Profile_Container)
- Citrix XAPI Backup  
<https://support.citrix.com/article/CTX217618>

# Infrastructure

## NVA-1148: NetApp HCI with Red Hat Virtualization

Alan Cowles, Nikhil M Kulkarni, NetApp

NetApp HCI with Red Hat Virtualization is a verified, best-practice architecture for the deployment of an on-premises virtual datacenter environment in a reliable and dependable manner.

This architecture reference document serves as both a design guide and a deployment validation of the Red Hat Virtualization solution on NetApp HCI. The architecture described in this document has been validated by subject matter experts at NetApp and Red Hat to provide a best-practice implementation for an enterprise virtual datacenter deployment using Red Hat Virtualization on NetApp HCI within your own enterprise datacenter environment.

### Use Cases

The NetApp HCI for Red Hat OpenShift on Red Hat Virtualization solution is architected to deliver exceptional value for customers with the following use cases:

1. Infrastructure to scale on demand with NetApp HCI
2. Enterprise virtualized workloads in Red Hat Virtualization

### Value Proposition and Differentiation of NetApp HCI with Red Hat Virtualization

NetApp HCI provides the following advantages with this virtual infrastructure solution:

- A disaggregated architecture that allows for independent scaling of compute and storage.
- The elimination of virtualization licensing costs and a performance tax on independent NetApp HCI storage nodes.
- NetApp Element storage provides quality of service (QoS) per storage volume and allows for guaranteed storage performance for workloads on NetApp HCI, preventing adjacent workloads from negatively affecting performance.
- The data fabric powered by NetApp allows data to be replicated from an on-premise to on-premise location or replicated to the cloud to move the data closer to where the application needs the data.
- Support through NetApp Support or Red Hat Support.

### NetApp HCI Design

NetApp HCI, is the industry's first and leading disaggregated hybrid cloud infrastructure, providing the widely recognized benefits of hyperconverged solutions. Benefits include lower TCO and ease of acquisition, deployment, and management for virtualized workloads, while also allowing enterprise customers to independently scale compute and storage resources as needed. NetApp HCI with Red Hat

Virtualization provides an open source, enterprise virtualization environment based on Red Hat Enterprise Linux.

By providing an agile turnkey infrastructure platform, NetApp HCI enables you to run enterprise-class virtualized and containerized workloads in an accelerated manner. At its core, NetApp HCI is designed to provide predictable performance, linear scalability of both compute and storage resources, and a simple deployment and management experience.

### **Predictable**

One of the biggest challenges in a multitenant environment is delivering consistent, predictable performance for all your workloads. Running multiple enterprise-grade workloads can result in resource contention, where one workload interferes with the performance of another. NetApp HCI alleviates this concern with storage quality-of-service (QoS) limits that are available natively with NetApp Element software. Element enables the granular control of every application and volume, helps to eliminate noisy neighbors, and satisfies enterprise performance SLAs. NetApp HCI multitenancy capabilities can help eliminate many traditional performance-related problems.

### **Flexible**

Previous generations of hyperconverged infrastructure typically required fixed resource ratios, limiting deployments to four-node and eight-node configurations. NetApp HCI is a disaggregated hyper-converged infrastructure that can scale compute and storage resources independently. Independent scaling prevents costly and inefficient overprovisioning, eliminates the 10% to 30% HCI tax from controller virtual machine (VM) overhead, and simplifies capacity and performance planning. NetApp HCI is available in mix-and-match, small, medium, and large storage and compute configurations.

The architectural design choices offered enable you to confidently scale on your terms, making HCI viable for core Tier-1 data center applications and platforms. NetApp HCI is architected in building blocks at either the chassis or the node level. Each chassis can hold four nodes in a mixed configuration of storage or compute nodes.

### **Simple**

A driving imperative within the IT community is to simplify deployment and automate routine tasks, eliminating the risk of user error while freeing up resources to focus on more interesting, higher-value projects. NetApp HCI can help your IT department become more agile and responsive by both simplifying deployment and ongoing management.

### **Business Value**

Enterprises that perform virtualization in an open-source data center with Red Hat products can realize the value of this solution by following the recommended design, deployment, and best practices described in this document. The detailed setup of RHV on NetApp HCI provides several benefits when deployed as part of an enterprise virtualization solution:

- High availability at all layers of the stack
- Thoroughly documented deployment procedures
- Nondisruptive operations and upgrades to hypervisors and the manager VM
- API-driven, programmable infrastructure to facilitate management
- Multitenancy with performance guarantees
- The ability to run virtualized workloads based on KVM with enterprise-grade features and support
- The ability to scale infrastructure independently based on workload demands

NetApp HCI with Red Hat Virtualization acknowledges these challenges and helps address each concern by implementing a verified architecture for solution deployment.

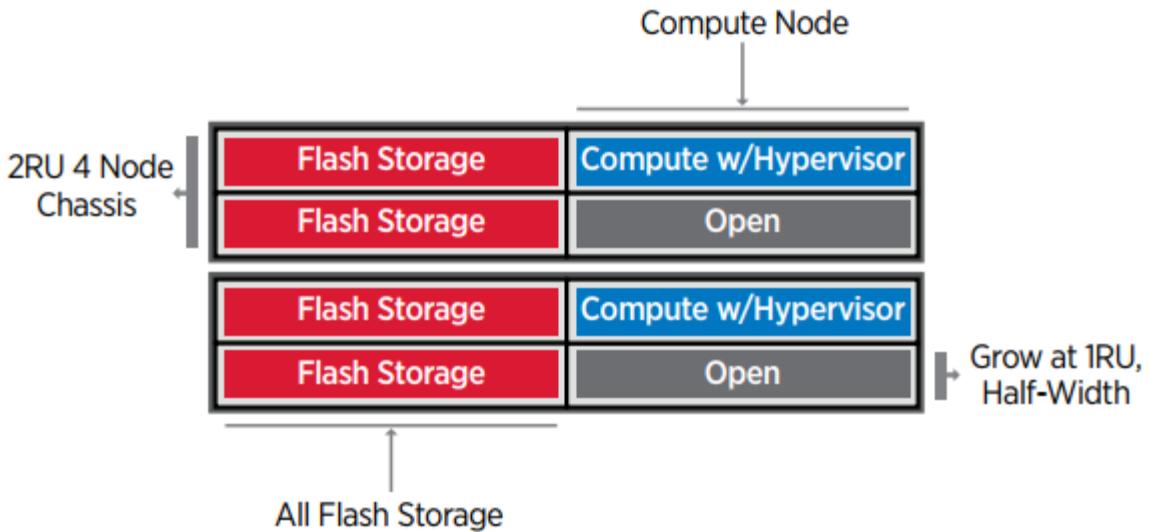
## **Technology Overview**

With NetApp HCI for Red Hat Virtualization, you can deploy a fully integrated, production-grade virtual data center that allows you to take advantage of the following features:

- NetApp HCI compute and storage nodes
  - Enterprise-grade hyperconverged infrastructure designed for hybrid cloud workloads
  - NetApp Element storage software
  - Intel-based server compute nodes, including options for NVIDIA GPUs
- Red Hat Virtualization
  - Enterprise hypervisor solution for deployment and management of virtual infrastructures

## **NetApp HCI**

NetApp HCI is an enterprise-scale disaggregated hybrid cloud infrastructure (HCI) solution that delivers compute and storage resources in an agile, scalable, and easy-to-manage two-rack unit (2RU) four-node building block. It can also be configured with 1RU compute and server nodes. The minimum deployment consists of four NetApp HCI storage nodes and two NetApp HCI compute nodes. The compute nodes are installed as RHV-H hypervisors in an HA cluster. This minimum deployment can be easily scaled to fit customer enterprise workload demands by adding additional NetApp HCI storage or compute nodes to expand available resources.



The design for NetApp HCI for Red Hat Virtualization consists of the following components in a minimum starting configuration:

- NetApp H-Series all-flash storage nodes running NetApp Element software
- NetApp H-Series compute nodes running the Red Hat Virtualization RHV-H hypervisor

For more information about compute and storage nodes in NetApp HCI, see the [NetApp HCI Datasheet](#).

### **NetApp Element Software**

NetApp Element software provides modular, scalable performance, with each storage node delivering guaranteed capacity and throughput to the environment. You can also specify per-volume storage QoS policies to support dedicated performance levels for even the most demanding workloads.

### **iSCSI Login Redirection and Self-Healing Capabilities**

NetApp Element software uses the iSCSI storage protocol, a standard way to encapsulate SCSI commands on a traditional TCP/IP network. When SCSI standards change or when Ethernet network performance improves, the iSCSI storage protocol benefits without the need for any changes.

Although all storage nodes have a management IP and a storage IP, NetApp Element software advertises a single storage virtual IP address (SVIP address) for all storage traffic in the cluster. As a part of the iSCSI login process, storage can respond that the target volume has been moved to a different address, and therefore it cannot proceed with the negotiation process. The host then reissues the login request to the new address in a process that requires no host-side reconfiguration. This process is known as iSCSI login redirection.

iSCSI login redirection is a key part of the NetApp Element software cluster. When a host login request is received, the node decides which member of the cluster should handle the traffic based on IOPS and the capacity requirements for the volume. Volumes are distributed across the NetApp Element software cluster and are redistributed if a single node is handling too much traffic for its volumes or if

a new node is added. Multiple copies of a given volume are allocated across the array. In this manner, if a node failure is followed by volume redistribution, there is no effect on host connectivity beyond a logout and login with redirection to the new location. With iSCSI login redirection, a NetApp Element software cluster is a self-healing, scale-out architecture that is capable of non-disruptive upgrades and operations.

## NetApp Element Software Cluster QoS

A NetApp Element software cluster allows QoS to be dynamically configured on a per-volume basis. You can use per-volume QoS settings to control storage performance based on SLAs that you define. The following three configurable parameters define the QoS:

- **Minimum IOPS.** The minimum number of sustained IOPS that the NetApp Element software cluster provides to a volume. The minimum IOPS configured for a volume is the guaranteed level of performance for a volume. Per-volume performance does not drop below this level.
- **Maximum IOPS.** The maximum number of sustained IOPS that the NetApp Element software cluster provides to a specific volume.
- **Burst IOPS.** The maximum number of IOPS allowed in a short burst scenario. The burst duration setting is configurable, with a default of 1 minute. If a volume has been running below the maximum IOPS level, burst credits are accumulated. When performance levels become very high and are pushed, short bursts of IOPS beyond the maximum IOPS are allowed on the volume.

## Multitenancy

Secure multitenancy is achieved with the following features:

- **Secure authentication.** The Challenge-Handshake Authentication Protocol (CHAP) is used for secure volume access. The Lightweight Directory Access Protocol (LDAP) is used for secure access to the cluster for management and reporting.
- **Volume access groups (VAGs).** Optionally, VAGs can be used in lieu of authentication, mapping any number of iSCSI initiator-specific iSCSI Qualified Names (IQNs) to one or more volumes. To access a volume in a VAG, the initiator's IQN must be in the allowed IQN list for the group of volumes.
- **Tenant virtual LANs (VLANS).** At the network level, end-to-end network security between iSCSI initiators and the NetApp Element software cluster is facilitated by using VLANs. For any VLAN that is created to isolate a workload or a tenant, Element software creates a separate iSCSI target SVIP address that is accessible only through the specific VLAN.
- **VPN routing/forwarding (VRF)-enabled VLANs.** To further support security and scalability in the data center, Element software allows you to enable any tenant VLAN for VRF-like functionality. This feature adds these two key capabilities:
  - **L3 routing to a tenant SVIP address.** This feature allows you to situate iSCSI initiators on a separate network or VLAN from that of the NetApp Element software cluster.
  - **Overlapping or duplicate IP subnets.** This feature enables you to add a template to tenant

environments, allowing each respective tenant VLAN to be assigned IP addresses from the same IP subnet. This capability can be useful for service provider environments where scale and preservation of IP-space are important.

## Enterprise Storage Efficiencies

The NetApp Element software cluster increases overall storage efficiency and performance. The following features are performed inline, are always on, and require no manual configuration by the user:

- **Deduplication.** The system only stores unique 4K blocks. Any duplicate 4K blocks are automatically associated with an already stored version of the data. Data is on block drives and is mirrored with Element Helix data protection. This system significantly reduces capacity consumption and write operations within the system.
- **Compression.** Compression is performed inline before data is written to NVRAM. Data is compressed, stored in 4K blocks, and remains compressed in the system. This compression significantly reduces capacity consumption, write operations, and bandwidth consumption across the cluster.
- **Thin provisioning.** This capability provides the right amount of storage at the time that you need it, eliminating capacity consumption that caused by overprovisioned volumes or underutilized volumes.
- **Helix.** The metadata for an individual volume is stored on a metadata drive and is replicated to a secondary metadata drive for redundancy.



Element was designed for automation. All the storage features mentioned above can be managed with APIs. These APIs are the only method that the UI uses to control the system and can be incorporated into user workflows to ease the management of the solution.

## Red Hat Virtualization

Red Hat Virtualization (RHV) is an enterprise virtual data center platform that runs on Red Hat Enterprise Linux using the KVM hypervisor.

For more information about Red Hat Virtualization, see the website located [here](#).

RHV provides the following features:

- **Centralized management of VMs and hosts.** The RHV manager runs as a physical or VM in the deployment and provides a web-based GUI for the management of the solution from a central interface.
- **Self-Hosted Engine.** To minimize the hardware requirements, RHV allows RHV Manager to be deployed as a VM on the same hosts that run guest VMs.
- **High Availability.** To avoid disruption from host failures, RHV allows VMs to be configured for

high availability. The highly available VMs are controlled at the cluster level using resiliency policies.

- **High Scalability.** A single RHV cluster can have up to 200 hypervisor hosts, enabling it to support the requirements of massive VMs to hold resource-greedy enterprise-class workloads.
- **Enhanced security.** Inherited from RHEL, Secure Virtualization (sVirt) and Security Enhanced Linux (SELinux) technologies are employed by RHV for the purposes of elevated security and hardening for the hosts and VMs. The key advantage from these features is logical isolation of a VM and its associated resources.

## Red Hat Virtualization Manager

Red Hat Virtualization Manager (RHV-M) provides centralized enterprise-grade management for the physical and logical resources within the RHV virtualized environment. A web-based GUI with different role-based portals is provided to access RHV-M features.

RHV-M exposes configuration and management of RHV resources with open-source, community-driven RESTful APIs. It also supports full-fledged integration with Red Hat CloudForms and Red Hat Ansible for automation and orchestration.

## Red Hat Virtualization Hosts

Hosts (also called hypervisors) are the physical servers that provide hardware resources for the VMs to run on. A kernel-based virtual machine (KVM) provides full virtualization support, and Virtual Desktop Server Manager (VDSM) is the host agent that is responsible for host communication with the RHV-M.

The two types of hosts supported in Red Hat Virtualization are Red Hat Virtualization Hosts (RHV-H) and Red Hat Enterprise Linux hosts (RHEL).

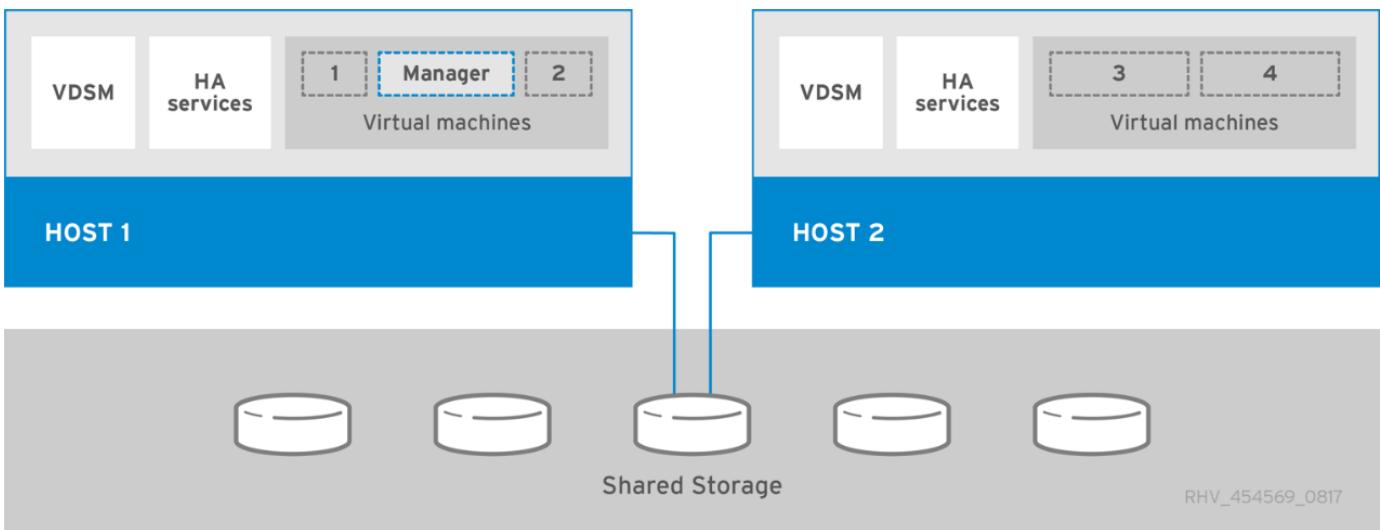
RHV-H is a minimal, light-weight operating system based on Red Hat Enterprise Linux that is optimized for the ease of setting up physical servers as RHV hypervisors.

RHEL hosts are servers that run the standard Red Hat Enterprise Linux operating system. They can then be configured with the required subscriptions to install the packages required to permit the physical servers to be used as RHV hosts.

## Red Hat Virtualization Architecture

Red Hat Virtualization can be deployed in two different architectures, with the RHV-M as a physical server in the infrastructure or with the RHV-M configured as a self-hosted engine. NetApp recommends using the self-hosted engine deployment, in which the RHV-M is a VM hosted in the same environment as other VMs, as we do in this guide.

A minimum of two self-hosted nodes are required for high availability of guest VMs and RHV-M. To provide high availability for the manager VM, HA services are enabled and run on all the self-hosted engine nodes.



## Architecture Overview: NetApp HCI with RHV

### Hardware Requirements

The following table lists the minimum number of hardware components that are required to implement the solution. The hardware components that are used in specific implementations of the solution might vary based on customer requirements.

Hardware	Model	Quantity
NetApp HCI compute nodes	NetApp H410C	2
NetApp HCI storage nodes	NetApp H410S	4
Data switches	Mellanox SN2010	2
Management switches	Cisco Nexus 3048	2

### Software Requirements

The following table lists the software components that are required to implement the solution. The software components that are used in any implementation of the solution might vary based on customer requirements.

Software	Purpose	Version
NetApp HCI	Infrastructure (compute/storage)	1.8
NetApp Element	Storage	12.0
Red Hat Virtualization	Virtualization	4.3.9

## Design Considerations: NetApp HCI with RHV

Review the following design considerations when developing your deployment

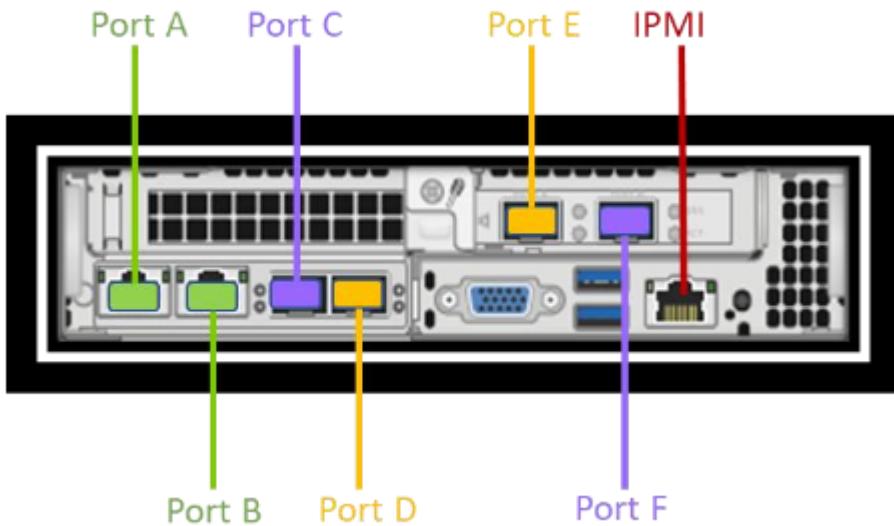
strategy.

## Networking Requirements

This section describes the networking requirements for the deployment of Red Hat Virtualization on NetApp HCI as a validated solution. It provides physical diagrams of the network ports on both the NetApp HCI compute nodes and the switches deployed in the solution. This section also describes the arrangement and purpose of each virtual network segment used in the solution.

## Port Identification

NetApp HCI consists of NetApp H-Series nodes dedicated to either compute or storage. Both node configurations are available with two 1GbE ports (ports A and B) and two 10/25GbE ports (ports C and D) on board. The compute nodes have additional 10/25GbE ports (ports E and F) available in the first mezzanine slot. Each node also has an additional out-of-band management port that supports Intelligent Platform Management Interface (IPMI) functionality. Each of these ports on the rear of an H410C node can be seen in the following figure.



## Network Design

The NetApp HCI with Red Hat Virtualization solution uses two data switches to provide primary data connectivity at 25Gbps. It also uses two additional management switches that provide connectivity at 1Gbps for in-band management for the storage nodes and out-of-band management for IPMI functionality.

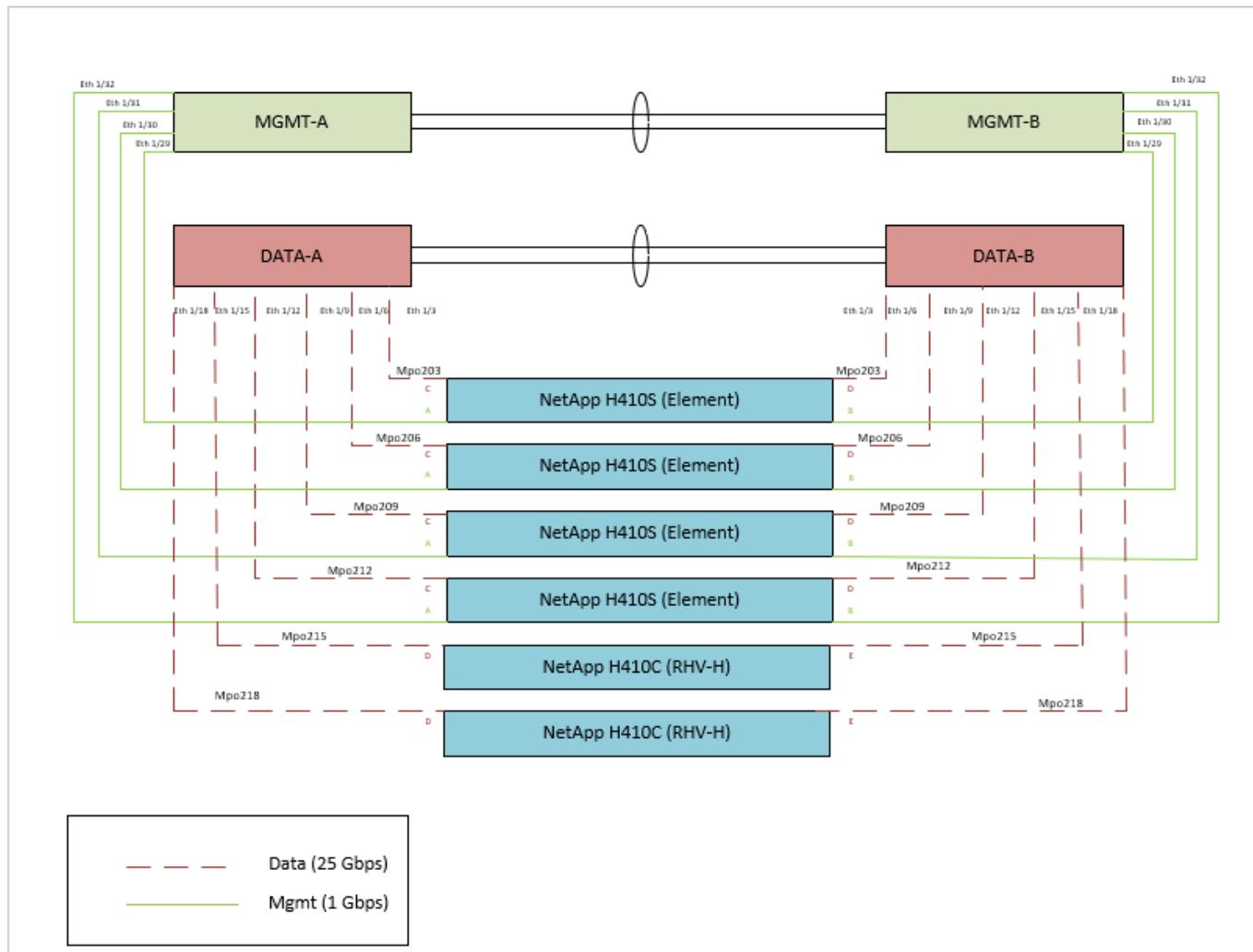
## Cabling Storage Nodes

The management ports A and B must be active on each storage node to configure the NetApp HCI cluster, and provide management accessibility to Element after the solution is deployed. The two 25Gbps ports (C and D) should be connected, one to each data switch, to provide physical fault tolerance. The switch ports should be configured for multi-chassis link aggregation (MLAG) and the data ports on the node should be configured for LACP with jumbo-frames support enabled. The IPMI

ports on each node can be used to remotely manage the node after it is installed in a data center. With IPMI, the node can be accessed with a web-browser-based console to run the initial installation, run diagnostics, and reboot or shut down the node if necessary.

## Cabling Compute Nodes

The two 25Gbps ports (C and E) should be connected, one to each data switch, to provide physical fault tolerance. The switch ports should be configured for multi-chassis link aggregation (MLAG), and the data ports on the node should be configured for LACP with jumbo-frames support enabled. The IPMI ports can also be used to remotely manage the node after it is installed in a data center. With IPMI, the node can be accessed with a web-browser- based console to run the initial installation, run diagnostics, and reboot or shut down the node if necessary.



## VLAN Requirements

The solution is designed to logically separate network traffic for different purposes by using Virtual Local Area Networks (VLANs). NetApp HCI requires a minimum of three network segments. However, this configuration can be scaled to meet customer demands or to provide further isolation for specific network services. The following table lists the VLANs that are required to implement the solution, as

well as the specific VLAN IDs that are used later in the validated architecture deployment.

VLANs	Purpose	VLAN Used
Out-of-band management network	Management for HCI nodes / IPMI	16
In-band management network	Management for HCI nodes / ovirtmgmt	1172
Storage network	Storage network for NetApp Element.	3343
Migration network	Network for virtual guest migration.	3345
VM network	Network for virtual guests.	3346

### **Network Infrastructure Support Resources**

The following infrastructure should be in place prior to the deployment of the Red Hat Virtualization on NetApp HCI solution:

- At least one DNS server providing full host-name resolution that is accessible from the in-band management network and the VM network.
- At least one NTP server that is accessible from the in-band management network and the VM network.
- Outbound internet connectivity is recommended, but not required, for both the in-band management network and the VM network.

## **Deployment Procedures NetApp HCI with RHV**

### **Deployment Summary: NetApp HCI with RHV**

The detailed steps provided in this section provide a validation for the minimum hardware and software configuration required to deploy and validate the NetApp HCI with Red Hat Virtualization solution.

Deploying Red Hat Virtualization for NetApp HCI involves the following high-level tasks:

1. Configure Management Switches
2. Configure Data Switches
3. Deploy Element Storage System on HCI Storage Nodes
4. Install RHV-H to HCI Compute Nodes
5. Deploy RHV Manager as a Self-hosted Engine

6. Deploy Test VMs
7. Test HA Functionality

## 1. Configure Management Switches: NetApp HCI with RHV

Cisco Nexus 3048 switches are used in this deployment procedure to provide 1Gbps connectivity for in and out-of-band management of the compute and storage nodes. These steps begin after the switches have been racked, powered, and put through the initial setup process. To configure the switches to provide management connectivity to the infrastructure, complete the following steps:

### Enable Advanced Features for Cisco Nexus

Run the following commands on each Cisco Nexus 3048 switch to configure advanced features:

1. Enter configuration mode.

```
Switch-01# configure terminal
```

2. Enable VLAN functionality.

```
Switch-01(config)# feature interface-vlan
```

3. Enable LACP.

```
Switch-01(config)# feature lacp
```

4. Enable virtual port channels (vPCs).

```
Switch-01(config)# feature vpc
```

5. Set the global port-channel load-balancing configuration.

```
Switch-01(config)# port-channel load-balance src-dst ip-l4port
```

6. Perform global spanning-tree configuration.

```
Switch-01(config)# spanning-tree port type network default
Switch-01(config)# spanning-tree port type edge bpduguard default
```

## Configure Ports on the Switch for In-Band Management

1. Run the following commands to create VLANs for management purposes:

```
Switch-01(config)# vlan 2
Switch-01(config-vlan)# Name Native_VLAN
Switch-01(config-vlan)# vlan 16
Switch-01(config-vlan)# Name OOB_Network
Switch-01(config-vlan)# vlan 1172
Switch-01(config-vlan)# Name MGMT_Network
Switch-01(config-vlan)# exit
```

2. Configure the ports ETH1/29-32 as VLAN trunk ports that connect to management interfaces on each HCI storage node.

```
Switch-01(config)# int eth 1/29
Switch-01(config-if)# description HCI-STG-01 PortA
Switch-01(config-if)# switchport mode trunk
Switch-01(config-if)# switchport trunk native vlan 2
Switch-01(config-if)# switchport trunk allowed vlan 1172
Switch-01(config-if)# spanning tree port type edge trunk
Switch-01(config-if)# int eth 1/30
Switch-01(config-if)# description HCI-STG-02 PortA
Switch-01(config-if)# switchport mode trunk
Switch-01(config-if)# switchport trunk native vlan 2
Switch-01(config-if)# switchport trunk allowed vlan 1172
Switch-01(config-if)# spanning tree port type edge trunk
Switch-01(config-if)# int eth 1/31
Switch-01(config-if)# description HCI-STG-03 PortA
Switch-01(config-if)# switchport mode trunk
Switch-01(config-if)# switchport trunk native vlan 2
Switch-01(config-if)# switchport trunk allowed vlan 1172
Switch-01(config-if)# spanning tree port type edge trunk
Switch-01(config-if)# int eth 1/32
Switch-01(config-if)# description HCI-STG-04 PortA
Switch-01(config-if)# switchport mode trunk
Switch-01(config-if)# switchport trunk native vlan 2
Switch-01(config-if)# switchport trunk allowed vlan 1172
Switch-01(config-if)# spanning tree port type edge trunk
Switch-01(config-if)# exit
```

## Configure Ports on the Switch for Out-of-Band Management

Run the following commands to configure the ports for cabling the IPMI interfaces on each HCI node.

```
Switch-01(config)# int eth 1/13
Switch-01(config-if)# description HCI-CMP-01 IPMI
Switch-01(config-if)# switchport mode access
Switch-01(config-if)# switchport access vlan 16
Switch-01(config-if)# spanning-tree port type edge
Switch-01(config-if)# int eth 1/14
Switch-01(config-if)# description HCI-STG-01 IPMI
Switch-01(config-if)# switchport mode access
Switch-01(config-if)# switchport access vlan 16
Switch-01(config-if)# spanning-tree port type edge
Switch-01(config-if)# int eth 1/15
Switch-01(config-if)# description HCI-STG-03 IPMI
Switch-01(config-if)# switchport mode access
Switch-01(config-if)# switchport access vlan 16
Switch-01(config-if)# spanning-tree port type edge
Switch-01(config-if)# exit
```



In the validated configuration, we cabled odd-node IPMI interfaces to Switch-01 and even-node IPMI interfaces to Switch-02.

## Create a vPC Domain to Ensure Fault Tolerance

1. Activate the ports used for the vPC peer-link between the two switches.

```
Switch-01(config)# int eth 1/1
Switch-01(config-if)# description vPC peer-link Switch-02 1/1
Switch-01(config-if)# int eth 1/2
Switch-01(config-if)# description vPC peer-link Switch-02 1/2
Switch-01(config-if)# exit
```

2. Perform the vPC global configuration.

```
Switch-01(config)# vpc domain 1
Switch-01(config-vpc-domain)# role priority 10
Switch-01(config-vpc-domain)# peer-keepalive destination <switch-02_mgmt_address> source
<switch-01_mgmt_address> vrf management
Switch-01(config-vpc-domain)# peer-gateway
Switch-01(config-vpc-domain)# auto recovery
Switch-01(config-vpc-domain)# ip arp synchronize
Switch-01(config-vpc-domain)# int eth 1/1-2
Switch-01(config-vpc-domain)# channel-group 10 mode active
Switch-01(config-vpc-domain)# int Po10
Switch-01(config-if)# description vPC peer-link
Switch-01(config-if)# switchport mode trunk
Switch-01(config-if)# switchport trunk native vlan 2
Switch-01(config-if)# switchport trunk allowed vlan 16, 1172
Switch-01(config-if)# spanning-tree port type network
Switch-01(config-if)# vpc peer-link
Switch-01(config-if)# exit
```

## 2. Configure Data Switches: NetApp HCI with RHV

Mellanox SN2010 switches are used in this deployment procedure to provide 25Gbps connectivity for the data plane of the compute and storage nodes. These steps begin after the switches have been racked, cabled, and put through the initial setup process. To configure the switches to provide data connectivity to the infrastructure, complete the following steps:

### Create MLAG Cluster to Provide Fault Tolerance

1. Run the following commands on each Mellanox SN210 switch for general configuration:

- Enter configuration mode.

```
Switch-01 enable
Switch-01 configure terminal
```

- Enable the LACP required for the Inter-Peer Link (IPL).

```
Switch-01 (config) # lacp
```

- Enable the Link Layer Discovery Protocol (LLDP).

```
Switch-01 (config) # lldp
```

- d. Enable IP routing.

```
Switch-01 (config) # ip routing
```

- e. Enable the MLAG protocol.

```
Switch-01 (config) # protocol mlag
```

- f. Enable global QoS.

```
Switch-01 (config) # dcb priority-flow-control enable force
```

2. For MLAG to function, the switches must be made peers to each other through an IPL. This should consist of two or more physical links for redundancy. The MTU for the IPL is set for jumbo frames (9216), and all VLANs are enabled by default. Run the following commands on each switch in the domain:

- a. Create port channel 10 for the IPL.

```
Switch-01 (config) # interface port-channel 10
Switch-01 (config interface port-channel 10) # description IPL
Switch-01 (config interface port-channel 10) # exit
```

- b. Add interfaces ETH 1/20 and 1/22 to the port channel.

```
Switch-01 (config) # interface ethernet 1/20 channel-group 10 mode active
Switch-01 (config) # interface ethernet 1/20 description ISL-SWB_01
Switch-01 (config) # interface ethernet 1/22 channel-group 10 mode active
Switch-01 (config) # interface ethernet 1/22 description ISL-SWB_02
```

- c. Create a VLAN outside of the standard range dedicated to IPL traffic.

```
Switch-01 (config) # vlan 4000
Switch-01 (config vlan 4000) # name IPL VLAN
Switch-01 (config vlan 4000) # exit
```

- d. Define the port channel as the IPL.

```
Switch-01 (config) # interface port-channel 10 ipl 1
Switch-01 (config) # interface port-channel 10 dcb priority-flow-control mode on
force
```

- e. Set an IP for each IPL member (non-routable; it is not advertised outside of the switch).

```
Switch-01 (config) # interface vlan 4000
Switch-01 (config vlan 4000) # ip address 10.0.0.1 255.255.255.0
Switch-01 (config vlan 4000) # ipl 1 peer-address 10.0.0.2
Switch-01 (config vlan 4000) # exit
```

3. Create a unique MLAG domain name for the two switches and assign a MLAG virtual IP (VIP). This IP is used for keep-alive heartbeat messages between the two switches. Run these commands on each switch in the domain:

- a. Create the MLAG domain and set the IP address and subnet.

```
Switch-01 (config) # mlag-vip MLAG-VIP-DOM ip a.b.c.d /24 force
```

- b. Create a virtual MAC address for the system MLAG.

```
Switch-01 (config) # mlag system-mac AA:BB:CC:DD:EE:FF
```

- c. Configure the MLAG domain so that it is active globally.

```
Switch-01 (config) # no mlag shutdown
```

The IP used for the MLAG VIP must be in the same subnet as the switch management network (mgmt0). Also, The MAC address used can be any unicast MAC address and must be set to the same value on both switches in the MLAG domain.

## Configure Ports to Connect to Storage and Compute Hosts

1. Create each of the VLANs needed to support the services for NetApp HCI. Run these commands on each switch in the domain:
- a. Create the VLANs.

```
Switch-01 (config) # vlan 1172
Switch-01 (config vlan 1172) exit
Switch-01 (config) # vlan 3343
Switch-01 (config vlan 3343) exit
Switch-01 (config) # vlan 3344
Switch-01 (config vlan 3345) exit
Switch-01 (config) # vlan 3345
Switch-01 (config vlan 3346) exit
```

- b. Create names for each VLAN for easier accounting.

```
Switch-01 (config) # vlan 1172 name "MGMT_Network"
Switch-01 (config) # vlan 3343 name "Storage_Network"
Switch-01 (config) # vlan 3345 name "Migration_Network"
Switch-01 (config) # vlan 3346 name "VM_Network"
```

2. Create MLAG interfaces and hybrid VLANs on ports identified so that you can distribute connectivity between the switches and tag the appropriate VLANs for the NetApp HCI compute nodes.

- a. Select the ports you want to work with.

```
Switch-01 (config) # interface ethernet 1/15
```

- b. Set the MTU for each port.

```
Switch-01 (config interface ethernet 1/15) # mtu 9216 force
```

- c. Modify spanning-tree settings for each port.

```
Switch-01 (config interface ethernet 1/15) # spanning-tree bpdufilter enable
Switch-01 (config interface ethernet 1/15) # spanning-tree port type edge
Switch-01 (config interface ethernet 1/15) # spanning-tree bpduguard enable
```

- d. Set the switchport mode to hybrid.

```
Switch-01 (config interface ethernet 1/15) # switchport mode hybrid
Switch-01 (config interface ethernet 1/15) # exit
```

- e. Create descriptions for each port being modified.

```
Switch-01 (config) # interface ethernet 1/15 description HCI-CMP-01 PortD
```

- f. Create and configure the MLAG port channels.

```
Switch-01 (config) # interface mlag-port-channel 215
Switch-01 (config) interface mlag-port-channel 215) # exit
Switch-01 (config) # interface mlag-port-channel 215 no shutdown
Switch-01 (config) # interface mlag-port-channel 215 mtu 9216 force
Switch-01 (config) # interface ethernet 1/15 lacp port-priority 10
Switch-01 (config) # interface ethernet 1/15 lacp rate fast
Switch-01 (config) # interface ethernet 1/15 mlag-channel-group 215 mode active
```

- g. Tag the appropriate VLANs for the NetApp HCI environment.

```
Switch-01 (config) # interface mlag-port-channel 215 switchport hybrid
Switch-01 (config) # interface mlag-port-channel 215 switchport hybrid allowed-vlan
add 1172
Switch-01 (config) # interface mlag-port-channel 215 switchport hybrid allowed-vlan
add 3343
Switch-01 (config) # interface mlag-port-channel 215 switchport hybrid allowed-vlan
add 3345
Switch-01 (config) # interface mlag-port-channel 215 switchport hybrid allowed-vlan
add 3346
```

3. Create MLAG interfaces and hybrid VLAN ports identified so that you can distribute connectivity between the switches and tag the appropriate VLANs for the NetApp HCI storage nodes.

- a. Select the ports that you want to work with.

```
Switch-01 (config) # interface ethernet 1/3
```

- b. Set the MTU for each port.

```
Switch-01 (config) interface ethernet 1/3) # mtu 9216 force
```

- c. Modify spanning tree settings for each port.

```
Switch-01 (config interface ethernet 1/3) # spanning-tree bpduguard enable
Switch-01 (config interface ethernet 1/3) # spanning-tree port type edge
Switch-01 (config interface ethernet 1/3) # spanning-tree bpduguard enable
```

- d. Set the switchport mode to hybrid.

```
Switch-01 (config interface ethernet 1/3) # switchport mode hybrid  
Switch-01 (config interface ethernet 1/3) # exit
```

- e. Create descriptions for each port being modified.

```
Switch-01 (config) # interface ethernet 1/3 description HCI-STG-01 PortD
```

- f. Create and configure the MLAG port channels.

```
Switch-01 (config) # interface mlag-port-channel 203  
Switch-01 (config interface mlag-port-channel 203) # exit  
Switch-01 (config) # interface mlag-port-channel 203 no shutdown  
Switch-01 (config) # interface mlag-port-channel 203 mtu 9216 force  
Switch-01 (config) # interface mlag-port-channel 203 lacp-individual enable force  
Switch-01 (config) # interface ethernet 203 lacp port-priority 10  
Switch-01 (config) # interface ethernet 203 lacp rate fast  
Switch-01 (config) # interface ethernet 1/3 mlag-channel-group 203 mode active
```

- g. Tag the appropriate VLANs for the storage environment.

```
Switch-01 (config) # interface mlag-port-channel 203 switchport mode hybrid  
Switch-01 (config) # interface mlag-port-channel 203 switchport hybrid allowed-vlan  
add 1172  
Switch-01 (config) # interface mlag-port-channel 203 switchport hybrid allowed-vlan  
add 3343
```

The configurations in this section show the configuration for a single port as example. They must also be run for each additional port connected in the solution, as well as on the associated port of the second switch in the MLAG domain. NetApp recommends that the descriptions for each port are updated to reflect the device ports that are being cabled and configured on the other switch.

## Create Uplink Ports for the Switches

1. Create an MLAG interface to provide uplinks to both Mellanox SN2010 switches from the core network.

```
Switch-01 (config) # interface mlag port-channel 201
Switch-01 (config interface mlag port-channel) # description Uplink CORE-SWITCH port
PORT
Switch-01 (config interface mlag port-channel) # exit
```

## 2. Configure the MLAG members.

```
Switch-01 (config) # interface ethernet 1/1 description Uplink to CORE-SWITCH port
PORT
Switch-01 (config) # interface ethernet 1/1 speed 10000 force
Switch-01 (config) # interface mlag-port-channel 201 mtu 9216 force
Switch-01 (config) # interface ethernet 1/1 mlag-channel-group 201 mode active
```

## 3. Set the switchport mode to hybrid and allow all VLANs from the core uplink switches.

```
Switch-01 (config) # interface mlag-port-channel switchport mode hybrid
Switch-01 (config) # interface mlag-port-channel switchport hybrid allowed-vlan all
```

## 4. Verify that the MLAG interface is up.

```
Switch-01 (config) # interface mlag-port-channel 201 no shutdown
Switch-01 (config) # exit
```



The configurations in this section must also be run on the second switch in the MLAG domain. NetApp recommends that the descriptions for each port are updated to reflect the device ports that are being cabled and configured on the other switch.

## 3. Deploy the Element Storage System on the HCI Storage Nodes: NetApp HCI with RHV

### Basic NetApp Element Storage Setup

NetApp Element cluster setup is performed in a manner similar to a standalone NetApp SolidFire storage setup. These steps begin after the nodes have been racked, and cabled, and the IPMI port has been configured on each node using the console. To setup a storage cluster, complete the following steps:

1. Access the out-of-band management console for the storage nodes in the cluster and log in with the default credentials ADMIN/ADMIN.



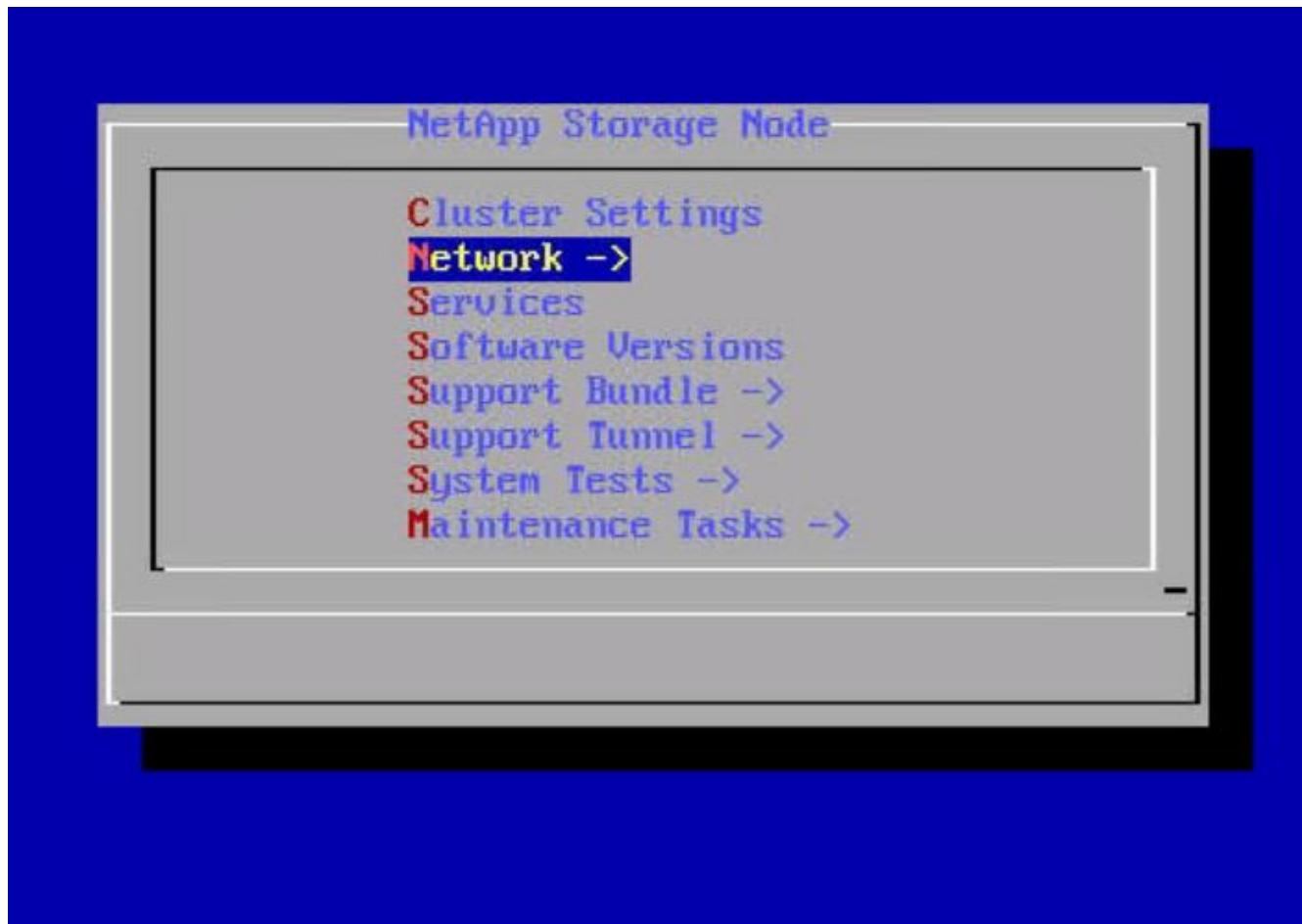
**Please Login**

**Username**

**Password**

**login**

2. Click the Remote Console Preview image in the center of the screen to download a JNLP file launched by Java Web Start, which launches an interactive console to the system.



3. Navigate to Network > Network Config > Bond1G (Management) and configure the Bond1G interface. The Bond1G interface should be in ActivePassive bond mode and must have an IP, a netmask, and a gateway set statically. Its VLAN must correspond to IB Management network and DNS servers defined for the environment. Then click OK.

NetApp Storage Node -> Network -> Network Config -> Bond1G

Hit 'tab' to navigate between the form and buttons. Use ↑↓ to navigate between fields. Start typing or hit ←→ to enter the field to make changes. Press 'enter' with a field selected, or hit 'tab' then 'enter' to submit all pending changes.

\* denotes required fields.

Method:	static
Link speed:	1000
*IPv4 Address:	10.63.172.136
*IPv4 Subnet_Mask:	255.255.255.0
*IPv4 Gateway:	10.63.172.1
Mtu:	1500
Dns:	10.61.184.251, 10.61.184.252
Domains:	cie.netapp.com
IPv6 Address:	
IPv6 Gateway:	
*Bond mode:	ActivePassive
*Status:	UpAndRunning
Vlan:	1172

< **OK** >

<**Cancel**>

< **Help** >

4. Select Bond10G (Storage) and configure the Bond10G interface. The Bond 10G interface must be in LACP bonding mode and have the MTU set to 9000 to enable jumbo frames. It must be assigned an IP address and netmask that are available on the defined storage VLAN. Click OK after entering the details.

NetApp Storage Node -> Network -> Network Config -> Bond10G

Hit 'tab' to navigate between the form and buttons. Use ↑↓ to navigate between fields. Start typing or hit ←→ to enter the field to make changes. Press 'enter' with a field selected, or hit 'tab' then 'enter' to submit all pending changes.

\* denotes required fields.

Method:	static
Link speed:	50000
*IPv4 Address:	172.21.87.130
*IPv4 Subnet_Mask:	255.255.255.0
IPv4 Gateway:	
Mtu:	9000
*Bond mode:	LACP
*Status:	UpAndRunning
Vlan:	3343

< **OK** >

<**Cancel**>

< **Help** >

5. Go back to the initial screen, navigate to Cluster Settings, and click Change Settings. Enter the Cluster Name of your choice and click OK.

**Change Cluster Settings**

Hit 'tab' to navigate between the form and buttons. Use ↑/↓ to navigate between fields. Start typing or hit ←/→ to enter the field to make changes. Press 'enter' with a field selected, or hit 'tab' then 'enter' to submit all pending changes.  
\* denotes required fields.

*Hostname:	SF-1A94
Cluster:	RHV-Store
*Management Interface:	Bond1G

< [OK](#) >      <[Cancel](#)>

6. Repeat steps 1 to 5 for all HCI storage nodes.
7. After all the storage nodes are configured, use a web browser to log into the IB Management IP of one of the storage nodes. This presents the setup page with the Create a New Cluster dialog. Management VIP, storage VIP, and other details of the Element cluster are configured on this page. The storage nodes that were configured in the previous step are automatically detected. Make sure that any nodes that you do not want in the cluster are unchecked before proceeding. Accept the End User License Agreement and click Create New Cluster to begin the cluster creation process. It takes a few minutes to get the cluster up.



In some cases, visiting the IB management address automatically connects on port 442 and launches the NDE setup wizard. If this happens, delete the port specification from the URL and reconnect to the page.

**Create a New Cluster**

**Node:** SF-1A94      **Status:** Searching for cluster RHV-Store

Management VIP :	10.63.172.140
iSCSI (Storage) VIP :	172.21.87.140
Data Protection :	Double Helix (2 replicas)
Create Username :	admin
Create Password :	*****
Repeat Password :	*****

**Nodes**

IP Address	Version	Include
172.21.87.30	12.0.0.333	<input checked="" type="checkbox"/>
172.21.87.32	12.0.0.333	<input checked="" type="checkbox"/>
172.21.87.130	12.0.0.333	<input checked="" type="checkbox"/>
172.21.87.132	12.0.0.333	<input checked="" type="checkbox"/>

8. After the cluster is created, it redirects to the Element cluster management interface available at the assigned MVIP address. Log in with the credentials provided in the previous step.
9. After you log in, the cluster automatically detects the number of available drives and requests for confirmation to add all drives. Click Add Drives to add all drives at once.
10. The Element cluster is ready to use. Navigate to Cluster > Nodes, and all four nodes should be in a healthy state with active drives.

The screenshot shows the NetApp Element Cluster Management interface. At the top, there's a navigation bar with tabs for Reporting, Management, Data Protection, Users, and Cluster. The Cluster tab is selected. Below the navigation bar, there's a sub-navigation bar with tabs for Settings, SNMP, LDAP, Drives, Nodes (which is highlighted), FC Ports, and Network. On the right side of the header, there are buttons for RHV-Store, API Log, and a bell icon. The main content area is titled "Nodes" and shows a table with four rows of node information. The columns are: Node ID, Node Name, Node Role, Node Type, Active Drives, Management IP, Cluster IP, Storage IP, Management VLAN ID, and Storage VLAN ID. The nodes listed are SF-1D1B, SF-1A94, SF-34F7, and SF-1FA7, each with 6 active drives and specific IP and VLAN details. At the bottom of the table, it says "Showing 1 - 4 of 4 Nodes".

Active	Pending	PendingActive									0 Selected	Bulk Actions					
Node ID	Node Name	Node Role	Node Type	Active Drives	Management IP	Cluster IP	Storage IP	Management VLAN ID	Storage VLAN ID								
4	SF-1D1B	Ensemble Node	H410S-1	6	10.63.172.138	172.21.87.132	172.21.87.132	1172	3343								
3	SF-1A94	Ensemble Node	H410S-1	6	10.63.172.136	172.21.87.130	172.21.87.130	1172	3343								
2	SF-34F7	Cluster Master, Ensemble Node	H410S-1	6	10.63.172.139	172.21.87.32	172.21.87.32	1172	3343								
1	SF-1FA7	-	H410S-1	6	10.63.172.137	172.21.87.30	172.21.87.30	1172	3343								

Showing 1 - 4 of 4 Nodes

## Element Storage Configuration to Support RHV Deployment

In our NetApp HCI for Red Hat Virtualization solution, we use a NetApp Element storage system to provide the backend storage support for RHV's requirement of shared storage domains. The self-hosted engine architecture of RHV deployment requires two storage domains at a minimum—one for the hosted engine storage domain and one for the guest VM data domain.

For this part of deployment, you must configure an account, two volumes of appropriate size, and the associated initiators. Then map these components to an access group that allows the RHV hosts to map the block volumes for use. Each of these actions can be performed through the web user interface or through the native API for the Element system. For this deployment guide, we go through the steps with the GUI.

Log in to the NetApp Element cluster GUI at its MVIP address using a web browser. Navigate to the Management tab and complete the following steps:

1. To create accounts, go to the Accounts sub-tab and click Create Account. Enter the name of your choice and click Create Account.

**Create a New Account** X

---

**Account Details**

Username

**CHAP Settings**

Initiator Secret

Target Secret

---

**Create Account** **Cancel**

2. To create volumes, complete the following steps:

- a. Navigate to the Volumes sub-tab and click Create Volume.
- b. To create the volume for the self-hosted engine storage domain, enter the name of your choice, select the account you created in the last step, enter the size of the volume for the self-hosted engine storage domain, configure the QoS setting, and click Create Volume.

### Volume Details

Volume Name

Volume Size

Gi ▾

Block Size

512e  4k

Account

▼

### Quality of Service

Policy

Custom Settings

IO Size	Min IOPS	Max IOPS	Burst IOPS
4 KB	50	15000	15000

8 KB	31 IOPS	9375 IOPS	9375 IOPS
------	---------	-----------	-----------

16 KB	19 IOPS	5556 IOPS	5556 IOPS
-------	---------	-----------	-----------

262 KB	1 IOPS	385 IOPS	385 IOPS
--------	--------	----------	----------

Max Bandwidth	104.86 MB/sec	104.86 MB/sec
---------------	---------------	---------------

**Create Volume**

**Cancel**

The minimum size for the hosted engine volume is 75GB. In our design, we added additional space to allow for future extents to be added to the RHV-M VM if necessary.

- c. To create the volume for the guest VMs data storage domain, enter the name of your choice, select the account you created in the last step, enter the size of the volume for the data storage domain, configure the QoS setting and click Create Volume.

### Volume Details

Volume Name

Volume Size

Block Size

 512e 4k

Account

### Quality of Service

Policy

Custom Settings

IO Size	Min IOPS	Max IOPS	Burst IOPS
4 KB	<input type="text" value="50"/>	<input type="text" value="15000"/>	<input type="text" value="15000"/>
8 KB	31 IOPS	9375 IOPS	9375 IOPS
16 KB	19 IOPS	5556 IOPS	5556 IOPS
262 KB	1 IOPS	385 IOPS	385 IOPS
Max Bandwidth		104.86 MB/sec	104.86 MB/sec

The size of the data domain depends on the kind of VMs run in the environment and the space required to support them. Adjust the size of this volume to meet the needs of your environment.

3. To create initiators, complete the following steps:
  - a. Go to the Initiators sub-tab and click Create Initiator.

- b. Select the Bulk Create Initiators radio button and enter the initiators' details of both the RHV-H nodes with comma separated values. Then click Add Initiators, enter the aliases for the initiators, and click the tick button. Verify the details and click Create Initiators.

**Create a New Initiator** X

---

Create a Single Initiator

IQN/WWPN  
[Empty input field]

Alias  
[Empty input field]

Bulk Create Initiators

Initiators	2
Name	Alias (optional)
iqn.1994-05.com.redhat:rhv-host-node-01	RHV-H01 <span style="float: right;">X</span>
iqn.1994-05.com.redhat:rhv-host-node-02	RHV-H02 <span style="float: right;">X</span>

---

**Create Initiators** **Cancel**

4. To create access groups, complete the following steps:
- Go to the Access Groups sub-tab and click Create Access Groups.
  - Enter the name of your choice, select the initiators for both RHV-H nodes that were created in the previous step, select the volumes, and click Create Access Group.

## Volume Access Group Details

Name

### Add Initiators

Initiators

[Create Initiator?](#)

Initiators			2
ID	Name	Alias	
3	iqn.1994-05.com.redhat:rhv-host-node-01	RHV-H01	
4	iqn.1994-05.com.redhat:rhv-host-node-02	RHV-H02	

Delete orphan initiators

### Attach Volumes

Volumes

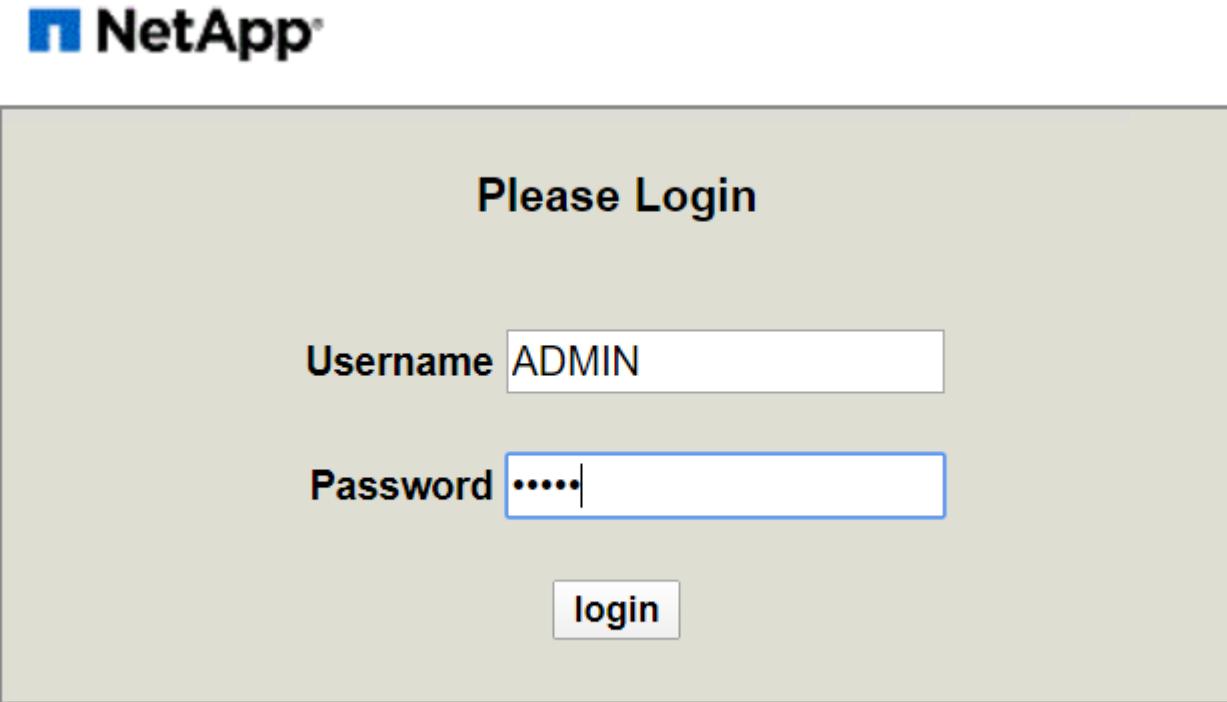
Attached Volumes			2
ID	Name		
1	RHV-HostedEngine		
2	RHV-DataDomain		

## 4. Deploy the RHV-H Hypervisor on the HCI Compute Nodes: NetApp HCI with RHV

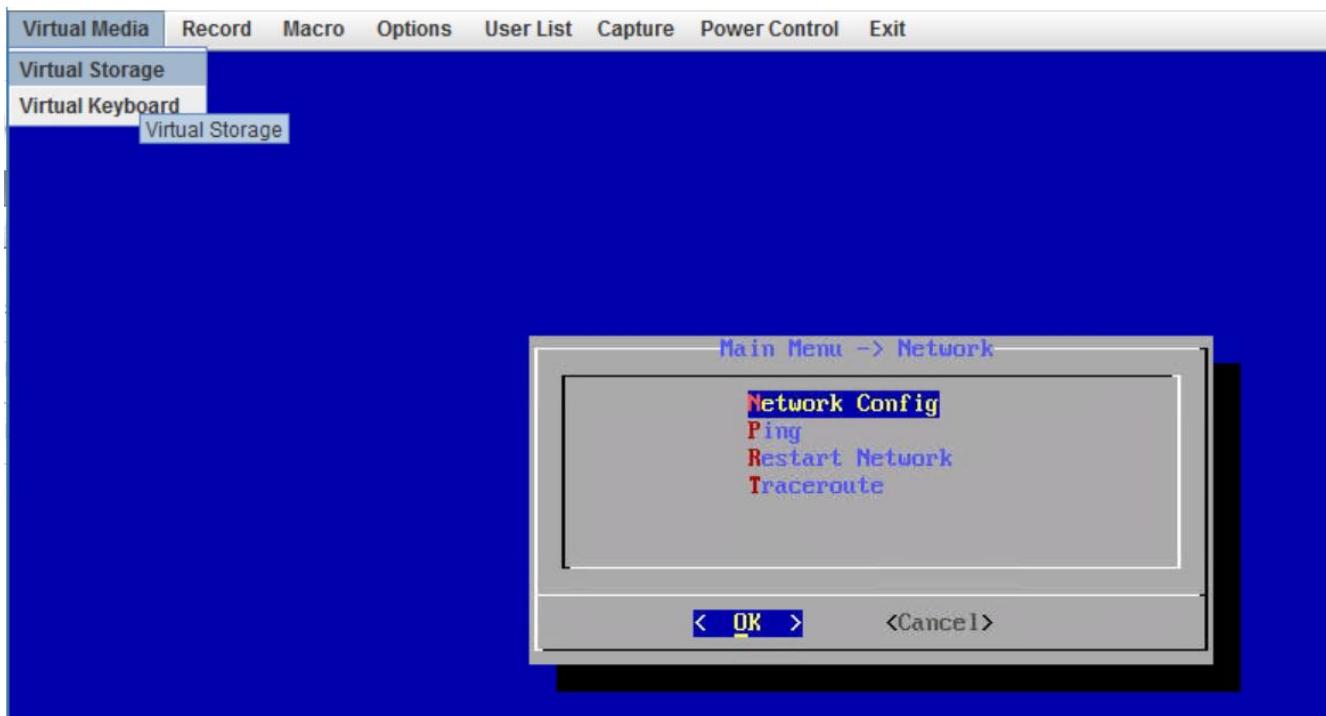
This solution employs the recommended self-hosted engine architecture of RHV deployment with the minimum setup (two self-hosted engine nodes). These steps begin after the nodes have been racked and cabled and the IPMI port has been

configured on each node for using the console. To deploy the RHV-H hypervisor on HCI compute nodes, complete the following steps:

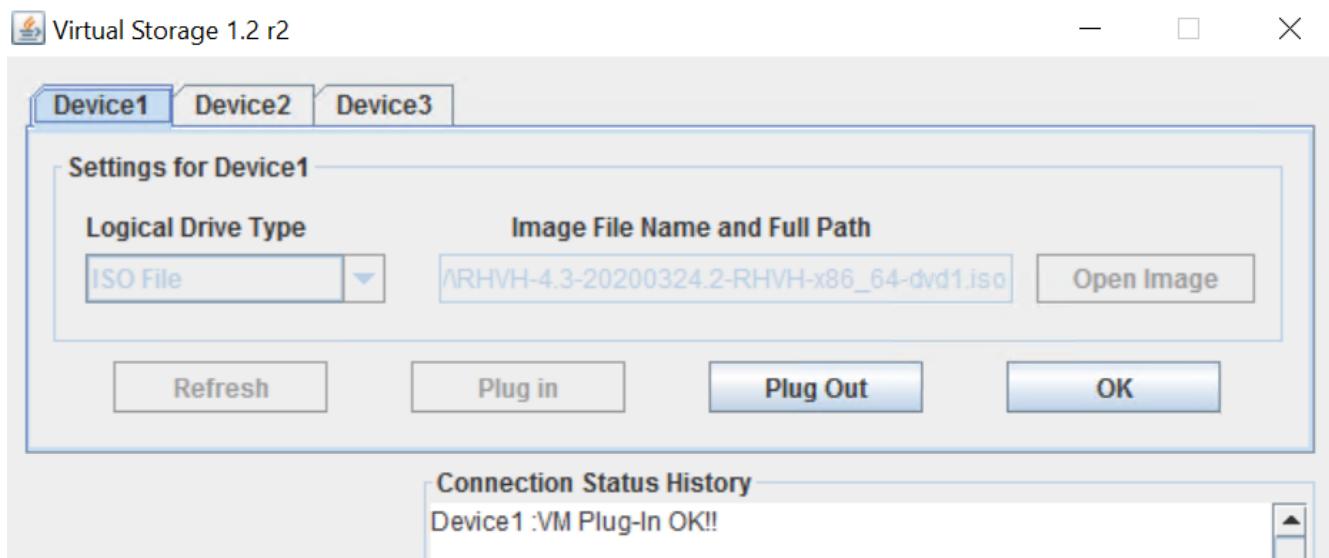
1. Access the out-of-band management console for the compute nodes in the cluster and log in with the default credentials ADMIN/ADMIN.



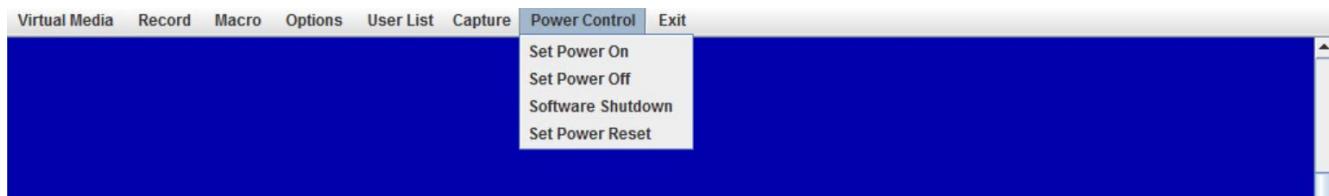
2. Click the Remote Console Preview image in the center of the screen to download a JNLP file launched by Java Web Start, which launches an interactive console to the system.
3. After the virtual console launches, attach the RHV-H 4.3.9 ISO by navigating to and clicking Virtual Media > Virtual Storage.



4. For Logical Drive Type, select ISO File from the drop down. Provide the full path and full name of the RHV-H 4.3.9 ISO file or attach it by clicking the Open Image button. Then click Plug In.



5. Reboot the server so that it boots using RHV-H 4.3.9 ISO by navigating and clicking Power Control > Set Power Reset.



6. When the node reboots and the initial screen appears, press F11 to enter the boot menu. From the boot menu, navigate to and click ATEN Virtual CDROM YSOJ.



7. On the next screen, navigate to and click Install RHV 4.3. This loads the image, runs the pre-installation scripts, and starts Anaconda, the Red Hat Enterprise Linux system installer.

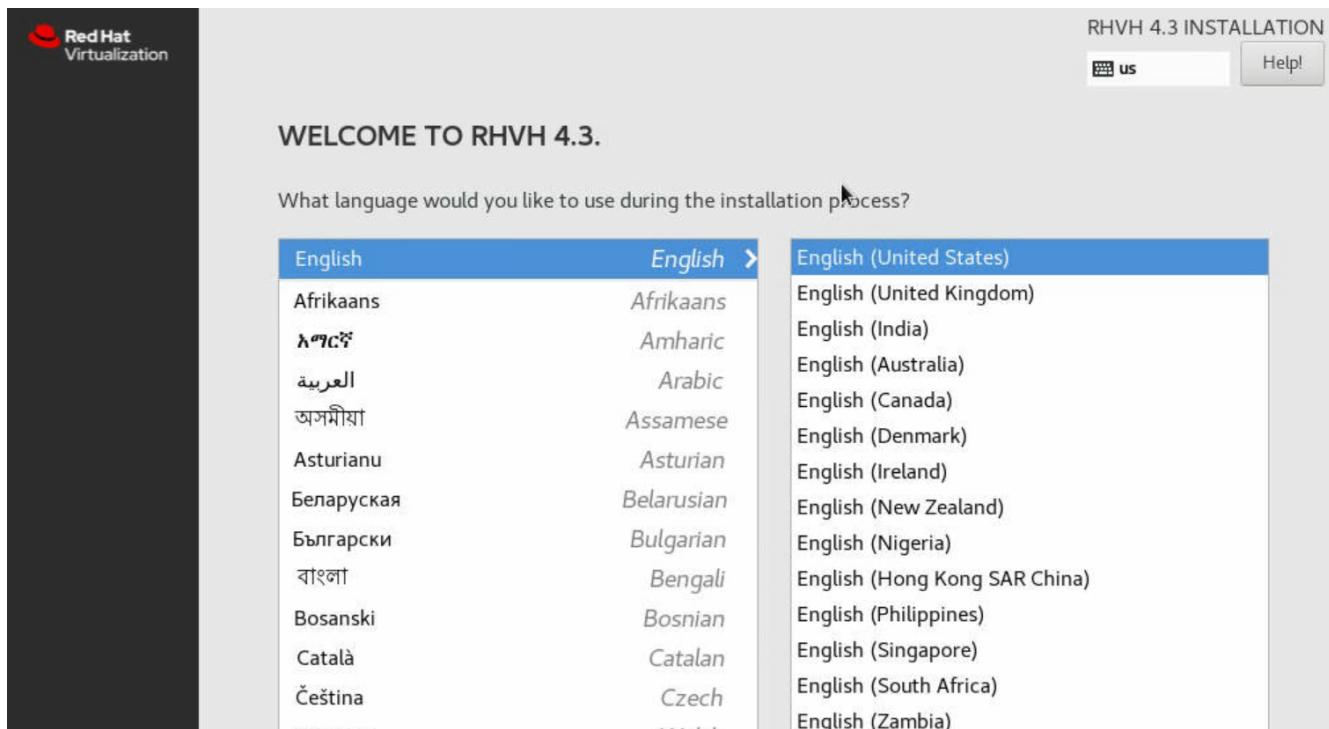
RHVH 4.3

Install RHVH 4.3  
Test this media & install RHVH 4.3

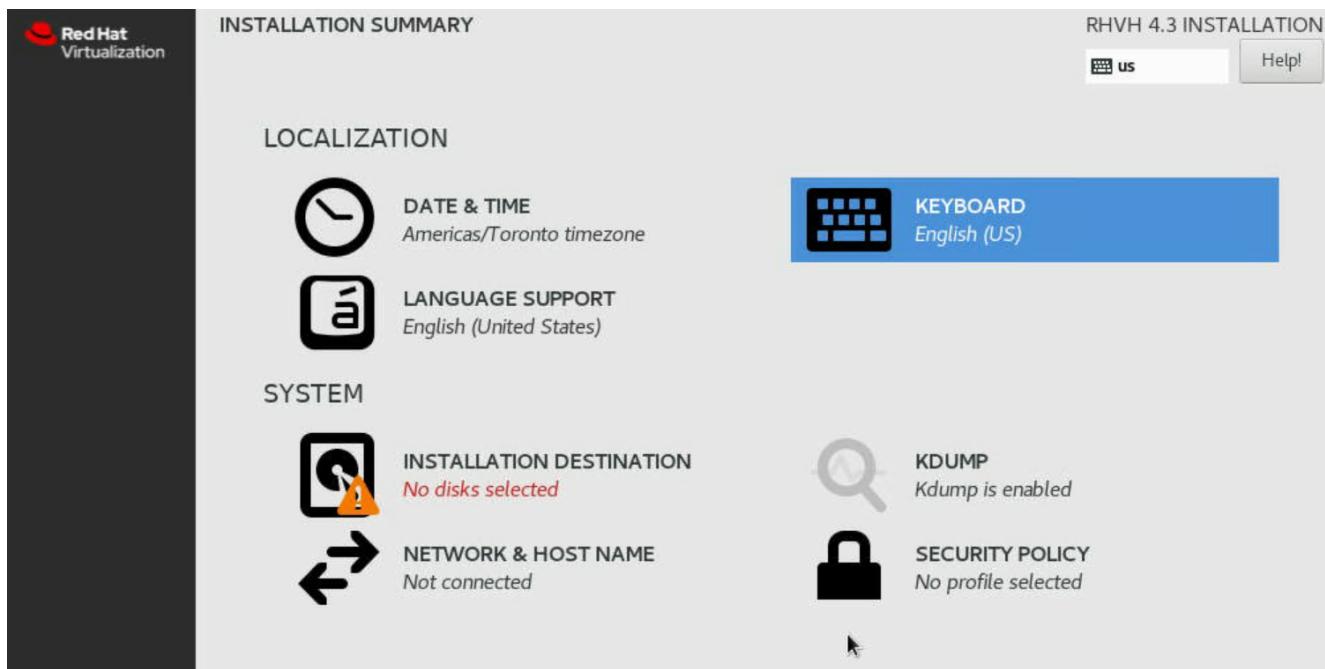
Troubleshooting >

Press Tab for full configuration options on menu items.

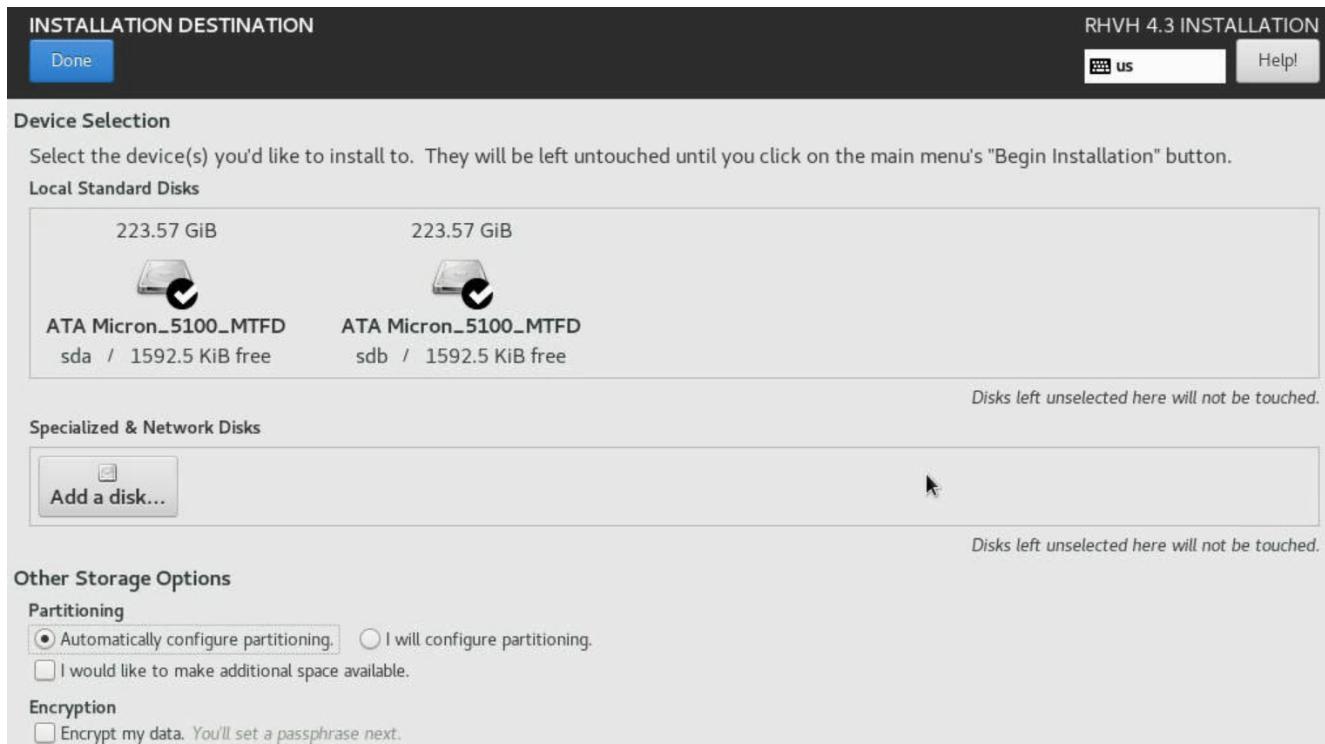
8. The installation welcome screen appears. Select the preferred language and click Next.



9. In the next screen, select your time zone under Date & Time. The default is UTC. However, NetApp recommends that you configure NTP servers for your environment on this screen. Then select the keyboard language and click Done.



10. Next, click Installation Destination. In the Installation Destination screen, select the drives on which you want to install RHV-H. Verify that Automatically Configure Partitioning is selected in the Partitioning section. Optionally, you can enable encryption by checking the box next to Encrypt My Data. Click Done to confirm the settings.



11. Click Network & Host Name. Provide the desired host name at the bottom of the screen. Then click

the (+) button at the bottom. Select the Bond from the drop down and click Add.



12. Next, in the bond configuration screen, click Add to add the member interfaces to the bond interface.

## Editing Bond connection 1

Connection name: **Bond connection 1**

General    **Bond**    Proxy    IPv4 Settings    IPv6 Settings

Interface name: **bond0**

Bonded connections:

	<b>Add</b>
	<b>Edit</b>
	<b>Delete</b>

Mode: **Round-robin**

Link Monitoring: **MII (recommended)**

Monitoring frequency: **1**   ms

Link up delay: **0**   ms

Link down delay: **0**   ms

MTU: **automatic**   bytes

**Cancel**    **Save**

13. Select Ethernet from the drop down, indicating that the Ethernet interface is added as a member to the bond interface. Click Create.



## Choose a Connection Type

Select the type of connection you wish to create.

If you are creating a VPN, and the VPN connection you wish to create does not appear in the list, you may not have the correct VPN plugin installed.

Ethernet

Cancel

Create...

- From the Device dropdown in the slave 1 configuration screen, select the Ethernet interface. Verify that the MTU is set to 9000. Click Save.

## Editing bond0 slave 1

Connection name: **bond0 slave 1**

General   **Ethernet**   802.1X Security   DCB

Device: en01 (AC:1F:6B:8D:85:28)

Cloned MAC address:

MTU: 9000 - + bytes

Wake on LAN:  
 Default    Phy    Unicast    Multicast  
 Ignore    Broadcast    Arp    Magic

Wake on LAN password:

Link negotiation: Automatic

Speed: 100 Mb/s

Duplex: Full

**Cancel**   **Save**

15. Repeat steps 12, 13, and 14 to add the other Ethernet port to the bond0 interface.
16. From the Mode dropdown in the bond configuration screen, select 802.3ad for LACP. Verify that the MTU is set to 9000. Then click Save.

Editing Bond connection 1

Connection name: Bond connection 1

General Bond Proxy IPv4 Settings IPv6 Settings

Interface name: bond0

Bonded connections:

- bond0 slave 1
- bond0 slave 2

Add Edit Delete

Mode: 802.3ad

Link Monitoring: MII (recommended)

Monitoring frequency: 1 ms

Link up delay: 0 ms

Link down delay: 0 ms

MTU: 9000 bytes

Cancel Save

17. Create the VLAN interface for the in-band management network. Click the (+) button again, select VLAN from the dropdown and click Create.



18. In the Editing VLAN connection screen, select bond0 in the Parent Interface dropdown, enter the VLAN ID of the in-band management network. Provide the name of the VLAN interface in `bond 0.<vlan_id>` format.

**Editing VLAN connection 1**

Connection name **VLAN connection 1**

**VLAN** **General** **Proxy** **IPv4 Settings** **IPv6 Settings**

Parent interface	bond0 (via "Bond connection 1")	▼
VLAN id	1172	- +
VLAN interface name	bond0.1172	
Cloned MAC address		▼
MTU	automatic	- + bytes
Flags	<input checked="" type="checkbox"/> Reorder headers <input type="checkbox"/> GVRP <input type="checkbox"/> Loose binding <input type="checkbox"/> MVRP	

**Cancel** **Save**

19. In the Editing VLAN connection screen, click the IPv4 Settings sub-tab. In the IPv4 Settings sub-tab, configure the network address, netmask, gateway, and DNS servers corresponding to the in-band management network. Click Save to confirm the settings.

**Editing VLAN connection 1**

Connection name: **VLAN connection 1**

General    VLAN    Proxy    **IPv4 Settings**    IPv6 Settings

Method: **Manual**

**Addresses**

Address	Netmask	Gateway	
10.63.172.151	24	10.63.172.1	<b>Add</b>

**DNS servers:** 10.61.184.251, 10.61.184.252

**Search domains:** cie.netapp.com

**DHCP client ID:** [empty field]

**Require IPv4 addressing for this connection to complete**

**Routes...**

**Cancel**    **Save**

20. Create the VLAN interface for the storage network. Click the (+) button again, select VLAN from the dropdown, and click Create. In the Editing VLAN Connection screen, select bond0 in the Parent Interface dropdown, enter the VLAN ID of the storage network, provide the name of the VLAN interface in the **bond 0.< vlan\_id >** format. Adjust the MTU to 9000 to allow jumbo frame support. Click Save.

Editing VLAN connection 2

Connection name: **VLAN connection 2**

General **VLAN** Proxy IPv4 Settings IPv6 Settings

Parent interface: bond0 (via “Bond connection 1”)

VLAN id: 3343 - +

VLAN interface name: bond0.3343

Cloned MAC address:

MTU: 9000 - + bytes

Flags:  Reorder headers  GVRP  Loose binding  MVRP

Cancel Save

21. In the Editing VLAN Connection screen, click the IPv4 Settings sub-tab. In the IPv4 Settings sub-tab, configure the network address and the netmask corresponding to the storage network. Click Save to confirm the settings.

Editing VLAN connection 2 (on localhost.localdomain) X

Connection name **VLAN connection 2**

General VLAN Proxy **IPv4 Settings** IPv6 Settings

Method **Manual** ▾

**Addresses**

Address	Netmask	Gateway	
172.21.87.31	255.255.255.0		<b>Add</b>
			<b>Delete</b>

DNS servers

Search domains

DHCP client ID

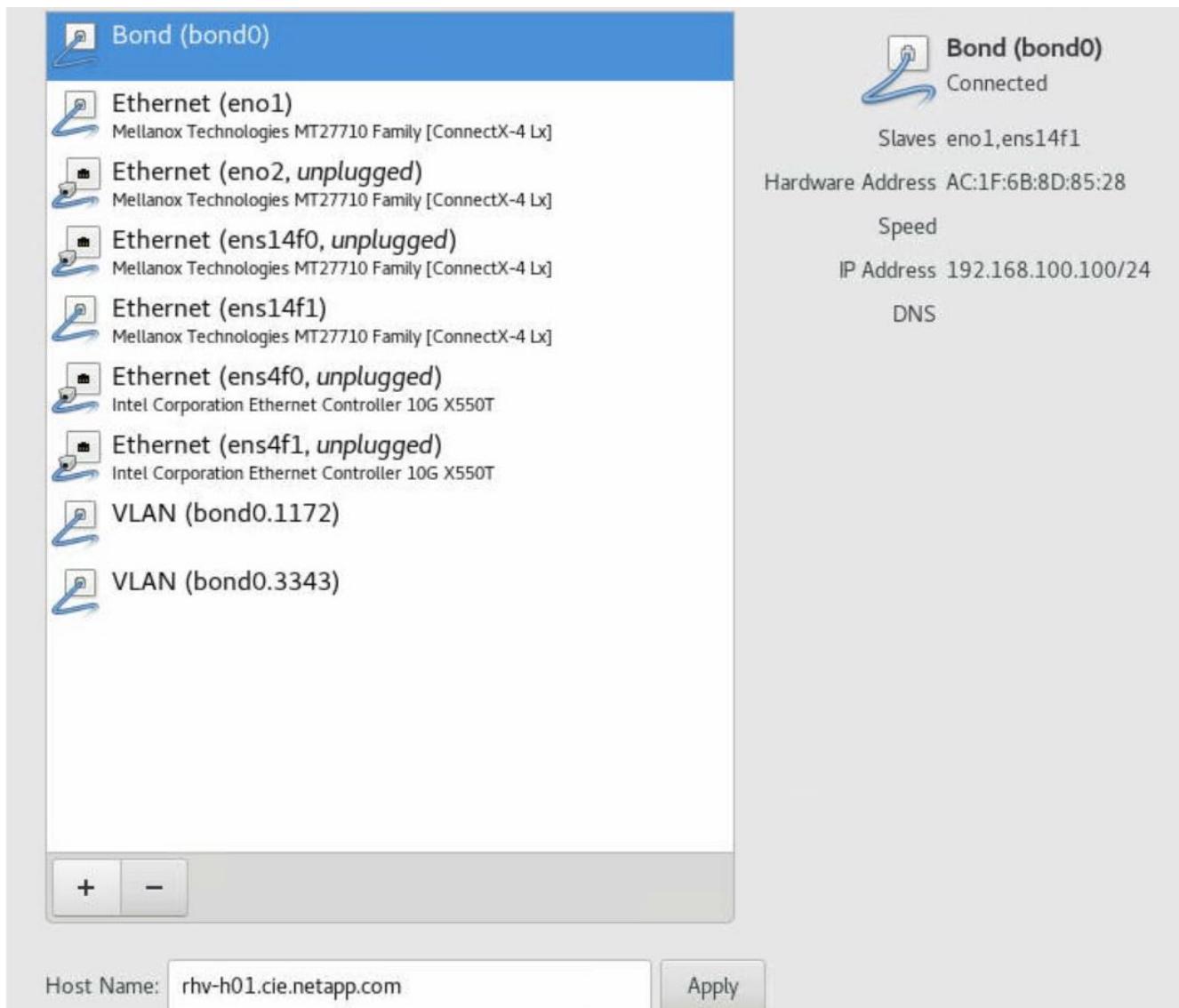
**Require IPv4 addressing for this connection to complete**

**Routes...**

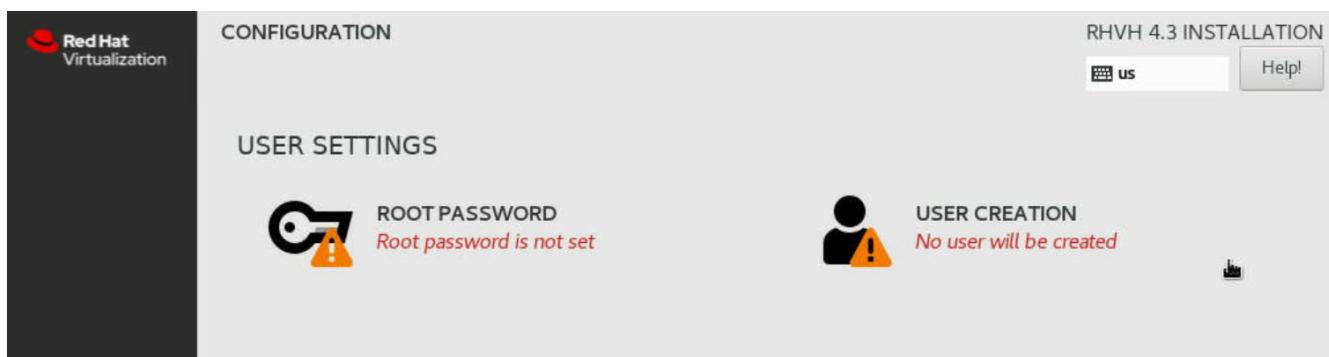
**Cancel** **Save**

Address	Netmask	Gateway	
172.21.87.31	255.255.255.0		<b>Add</b>
			<b>Delete</b>

22. Confirm that the network interfaces are up and click Done.



23. After the wizard navigates back to the configuration page, click Begin Installation. The next screen prompts you to configure the root password and optionally to create another user for logging into RHV-H.



24. After the installation completes, unmount the ISO file by navigating to Virtual media > Virtual Storage in the virtual console and click Plug Out. Then click Reboot on the Anaconda GUI to complete the installation process. The node then reboots.



After the node comes up, it displays the login screen.

```
Red Hat Virtualization Host 4.3.9 (el7.8)
Kernel 3.10.0-1127.el7.x86_64 on an x86_64
rhv-h01 login:
```

- Now that the installation is complete, you must then register RHV-H and enable the required repositories. Open a browser and log in to the Cockpit user interface at <https://<HostFQDN/IP>:9090> using the root credentials provided during the installation.

RED HAT VIRTUALIZATION HOST 4.3.9 (EL7.8)

Virtualization

Node Status

Health **ok** (green circle)

Current Layer [rhvh-4.3.9.2-0.20200324.0+1](#)

[Rollback](#)

System

Networking Information: [View](#)

System Logs: [View](#)

Storage: [View](#)

SSH Host Key: [View](#)

Virtual Machines 0 Running

26. Navigate to localhost > Subscriptions and click Register. Enter your Red Hat Portal username and password, click the check box Connect this System to Red Hat Insights, and click Register. The system automatically subscribes to the Red Hat Virtualization Host entitlement.

Red Hat Insights provide continuous analysis of registered systems to proactively recognize threats to availability, security, performance, and stability across physical, virtual, and cloud environments.

### Register system

URL

Proxy  Use proxy server

Login

Password

Activation Key

Organization

Insights  Connect this system to [Red Hat Insights](#).

[Cancel](#) [Register](#)

27. Navigate to localhost > Terminal to display the CLI. Optionally you can use any SSH client to log in to the RHV-H CLI. Confirm that the required subscription is attached, and then enable the Red Hat Virtualization Host 7 repository to allow further updates and make sure that all other repositories

are disabled.

```
# subscription-manager list
+-----+
   Installed Product Status
+-----+
Product Name: Red Hat Virtualization Host
Product ID: 328
Version: 4.3
Arch: x86_64
Status: Subscribed
# subscription-manager repos --disable=
Repository 'rhel-7-server- rhvh-4-source-rpms' is disabled for this system.
Repository 'rhvh-4-build-beta-for-rhel-8-x86_64-source-rpms' is disabled for this
system.
Repository 'rhel-7-server- rhvh-4-beta-debug-rpms' is disabled for this system.
Repository 'rhvh-4-beta-for-rhel-8-x86_64-debug-rpms' is disabled for this system.
Repository 'jb-eap-textonly-1-for-middleware-rpms' is disabled for this system.
Repository 'rhvh-4-build-beta-for-rhel-8-x86_64-rpms' is disabled for this system.
Repository 'rhvh-4-beta-for-rhel-8-x86_64-source-rpms' is disabled for this system.
Repository 'rhel-7-server- rhvh-4-debug-rpms' is disabled for this system.
Repository 'rhvh-4-build-beta-for-rhel-8-x86_64-debug-rpms' is disabled for this
system.
Repository 'rhel-7-server- rhvh-4-beta-source-rpms' is disabled for this system.
Repository 'rhel-7-server- rhvh-4-rpms' is disabled for this system.
Repository 'jb-coreservices-textonly-1-for-middleware-rpms' is disabled for this
system.
Repository 'rhvh-4-beta-for-rhel-8-x86_64-rpms' is disabled for this system.
Repository 'rhel-7-server- rhvh-4-beta-rpms' is disabled for this system.
# subscription-manager repos --enable=rhel-7-server- rhvh-4-rpms
Repository 'rhel-7-server- rhvh-4-rpms' is enabled for this system.
```

28. From the console, modify the iSCSI initiator ID to match the one you set in the Element access group previously by running the following command.

```
rhv-h01 # echo InitiatorName=iqn.1994-05.com.redhat:rhv-host-node- 01 >
/etc/iscsi/initiatorname.iscsi
```

29. Enable and restart the iscsid service.

```

# systemctl enable iscsid
Created symlink from /etc/systemd/system/multi-user.target.wants/iscsid.service to
/usr/lib/systemd/system/iscsid.service
# systemctl start iscsid
# systemctl status iscsid
● iscsid.service - Open-iSCSI
   Loaded: loaded (/usr/lib/systemd/system/iscsid.service; enabled; vendor preset:
disabled)
   Active: active (running) since Thu 2020-05-14 16:08:52 EDT; 3 days ago
     Docs: man:iscsid(8)
           man:iscsiuio(8)
           man:iscsiadm(8)
 Main PID: 5422 (iscsid)
   Status: "Syncing existing session(s)"
   CGroup: /system.slice/iscsid.service
           └─5422 /sbin/iscsid -f
              └─5423 /sbin/iscsid -f

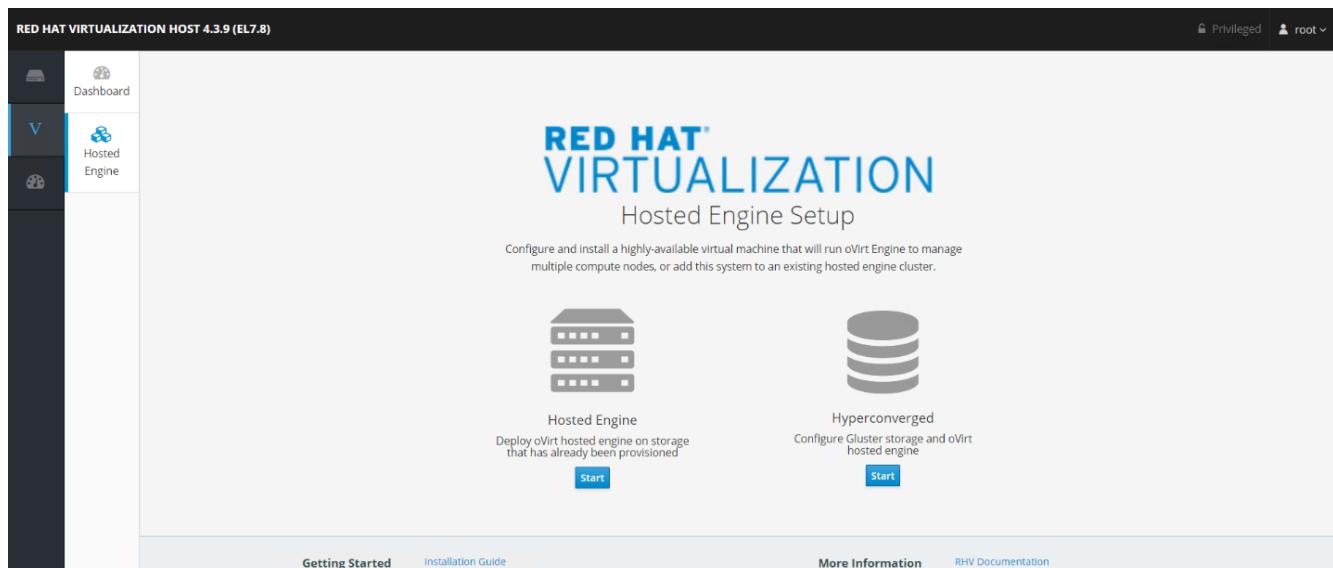
```

30. Install and prepare the other RHV host by repeating the steps 1 to 29.

## 5. Deploy the RHV Manager as a Self-Hosted Engine: NetApp HCI with RHV

This section describes the detailed steps for installing the Red Hat Virtualization Manager as a self-hosted engine. These steps begin after the RHV hosts are registered and the Cockpit GUI is accessible.

1. Log in to the Cockpit GUI of one of the RHV hosts at <https://<HostFQDN/IP>:9090> using the root credentials. Navigate to the Virtualization sub-tab and click Hosted Engine. Then click the Start button below the Hosted Engine content to initiate the engine deployment.



2. In the first screen of engine deployment, configure the RHV-M FQDN, network related configuration, root password, and resources for the engine VM (at least 4 CPUs and 16GB memory). Confirm the other configuration settings as required and click Next.

Hosted Engine Deployment

VM      Engine      Prepare VM      Storage      Finish

1      2      3      4      5

VM Settings

Engine VM FQDN	rhv-m.cie.netapp.com	✓
MAC Address	00:16:3e:4e:6b:05	
Network Configuration	Static	
VM IP Address	10.63.172.150	/ 24
Gateway Address	10.63.172.1	
DNS Servers	10.61.184.251	-
	10.61.184.252	- +
Bridge Interface	bond0.1172	
Root Password	.....	👁
Root SSH Access	Yes	
Number of Virtual CPUs	4	
Memory Size (MiB)	16384	511,548MB available

> Advanced

Cancel   < Back   **Next >**



Make sure that the engine VM FQDN is resolvable by the specified DNS servers.

3. In the next screen, enter the admin portal password. Optionally, enter the notification settings for alerts to be sent by email. Then click Next.

## Hosted Engine Deployment

X



### Engine Credentials

Admin Portal Password

### Notification Settings

Server Name

Server Port Number

Sender E-Mail Address

Recipient E-Mail Addresses

[Cancel](#)

[< Back](#)

[Next >](#)

4. In the next screen, review the configuration for the engine VM. If any changes are desired, go back at this point and make them. If the information is correct, click Prepare the VM.



Please review the configuration. Once you click the 'Prepare VM' button, a local virtual machine will be started and used to prepare the management services and their data. This operation may take some time depending on your hardware.

▼ VM

**Engine FQDN:** rhv-m.cie.netapp.com  
**MAC Address:** 00:16:3e:4e:6b:05  
**Network Configuration:** Static  
**VM IP Address:** 10.63.172.150/24  
**Gateway Address:** 10.63.172.1  
**DNS Servers:** 10.61.184.251,10.61.184.252  
**Root User SSH Access:** yes  
**Number of Virtual CPUs:** 4  
**Memory Size (MiB):** 16384  
**Root User SSH Public Key:** (None)  
**Add Lines to /etc/hosts:** yes  
**Bridge Name:** ovirtmgmt  
**Apply OpenSCAP profile:** no

▼ Engine

**SMTP Server Name:** localhost  
**SMTP Server Port Number:** 25  
**Sender E-Mail Address:** root@localhost  
**Recipient E-Mail Addresses:** root@localhost

[Cancel](#)

[◀ Back](#)

[Prepare VM](#)

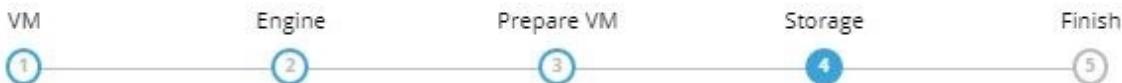
- 
5. The VM installation begins and can take some time to complete as it downloads a machine image and stages the VM locally. After it has completed, it displays the Execution Completed Successfully message. Click Next.



Execution completed successfully. Please proceed to the next step.

[Cancel](#)[< Back](#)[Next >](#)

6. After RHV-M is installed, enter the details of the hosted engine storage domain where it copies the VM from local storage to the shared storage domain to facilitate a high availability engine quorum.
7. Enter the Storage Type as iSCSI, provide the iSCSI portal details, click Retrieve Target List, which fetches the iSCSI target list corresponding to the portal, and select the volume and LUN to be mapped to the hosted engine storage domain. Click Next.



Please configure the storage domain that will be used to host the disk for the management VM. Please note that the management VM needs to be responsive and reliable enough to be able to manage all resources of your deployment, so highly available storage is preferred.

#### Storage Settings

Storage Type	iSCSI
Portal IP Address	172.21.87.140
Portal Port	3260
Portal Username	admin
Portal Password	.....

**Retrieve Target List**

The following targets have been found:

- ④ iqn.2010-01.com.solidfire:nh35.rhv-hostedengine.1, TPGT: 1  
172.21.87.140:3260

The following luns have been found on the requested target:

- ④ ID: 36f47acc1000000006e68333500000003  
Size (GiB): 186.00  
Description: SolidFire SSD SAN  
Status: free  
Number of Paths: 1

› Advanced



If the Hosted Engine setup is unable to discover the storage, open an interactive SSH session to the node and verify that you can reach the SVIP IP address through your node's storage interface. If the network is reachable, you might need to manually discover or log in to the iSCSI LUN intended for the Hosted Engine install.

8. On the next screen, review the storage configuration and, if any changes are desired, go back and make them. If the information is correct, click Finish Deployment. It takes some time as the VM is copied to the storage domain. After deployment is complete, click Close.

## Hosted Engine Deployment

X



Hosted engine deployment complete!

**Close**

9. The next step is to register and enable the Red Hat Virtualization Manager repositories. Log in to the RHV-M VM with SSH to register it with Subscription Manager.

```
# subscription-manager register
Registering to: subscription.rhsm.redhat.com:443/subscription
Username: redhat_user
Password: redhat_password
The system has been registered with ID: 99d06fcb-a3fd74-41230f-bad583-0ae61264f9a3
The registered system name is: rhv-m.cie.netapp.com
```

10. After registration, list the available subscriptions and record the pool ID for RHV-M.

```
# subscription-manager list --available
<snip>
Subscription Name: Red Hat Virtualization Manager
Provides: Red Hat Beta
Red Hat Enterprise Linux Server
Red Hat CodeReady Linux Builder for x86_64
Red Hat Enterprise Linux for x86_64
Red Hat Virtualization Manager
Red Hat OpenShift Container Platform
Red Hat Ansible Engine
Red Hat Enterprise Linux Fast Datapath
Red Hat JBoss Core Services
JBoss Enterprise Application Platform
SKU: RV00045
Contract:
Pool ID: 8a85f9937a1a2a57c0171a366b5682540112a313 ⚡ Pool ID
Provides Management: No
Available: 6
Suggested: 0
Service Type: L1-L3
Roles:
Service Level: Layered
Usage:
Add-ons:
Subscription Type: Stackable
Starts: 04/22/2020
Ends: 04/21/2021
Entitlement Type: Physical
<snip>
```

11. Attach the RHV-M subscription using the recorded pool ID.

```
# subscription-manager attach --pool=8a85f9937a1a2a57c0171a366b5682540112a313
Successfully attached a subscription for: Red Hat Virtualization Manager
```

12. Enable the required RHV-M repositories.

```
# subscription-manager repos \
--disable='*' \
--enable=rhel-7-server-rpms \
--enable=rhel-7-server-supplementary-rpms \
--enable=rhel-7-server-rhv-4.3-manager-rpms \
--enable=rhel-7-server-rhv-4-manager-tools-rpms \
--enable=rhel-7-server-ansible-2-rpms \
--enable=jb-eap-7.2-for-rhel-7-server-rpms
Repository 'rhel-7-server-ansible-2-rpms' is enabled for this system.
Repository 'rhel-7-server-rhv-4-manager-tools-rpms' is enabled for this system.
Repository 'rhel-7-server-rhv-4.3-manager-rpms' is enabled for this system.
Repository 'rhel-7-server-rpms' is enabled for this system.
Repository 'jb-eap-7.2-for-rhel-7-server-rpms' is enabled for this system.
Repository 'rhel-7-server-supplementary-rpms' is enabled for this system.
```

13. Next, create a storage domain to hold the VM disks or OVF files for all VMs in the same datacenter as that of the hosts.
14. To log into the RHV-M Administrative portal using a browser, log into <https://<ManagerFQDN>/ovirt-engine>, select Administrative Portal, and log in as the **admin @ internal** user.
15. Navigate to Storage > Storage Domains and click New Domain.
16. From the dropdown menu, select Data for the Domain Function, select iSCSI for the Storage Type, select the host to map the volume, enter a name of your choice, confirm that the data center is correct, and then expand the data domain iSCSI target and add the LUN. Click OK to create the domain.

New Domain

Data Center	Default (V5)	Name	data_domain
Domain Function	Data	Description	Data Domain for VMs
Storage Type	iSCSI	Comment	
Host	rhv-h01.cie.netapp.com		

- Discover Targets Login All

Target Name	Address	Port	
iqn.2010-01.com.solidfire:nh35.rhv-hostedengine-1.3	172.21.87.140	3260	<span style="color: green;">→</span>
iqn.2010-01.com.solidfire:nh35.rhv-hostedengine.1	172.21.87.140	3260	<span style="color: green;">→</span>
iqn.2010-01.com.solidfire:nh35.data-domain.5	172.21.87.140	3260	<span style="color: green;">→</span>

LUNs > Targets

LUNs > LUNS

Advanced Parameters

OK Cancel



If the Hosted Engine setup is unable to discover the storage, you might need to manually discover or log in to the iSCSI LUN intended for the data domain.

17. Add the second host to the hosted engine quorum. Navigate to Compute > Hosts and click New. In the New Host pane, select the appropriate cluster, provide the details of the second host, and check the Activate Host After Install checkbox.

New Host X

- General** >
- Power Management
- SPM
- Console and GPU
- Kernel
- Hosted Engine
- Affinity

**Host Cluster** Default ▼  
*Data Center: Default*

Use Foreman/Satellite

---

**Name**

**Comment**

**Hostname/IP** i

**SSH Port**

Activate host after install

**Authentication**

**User Name**

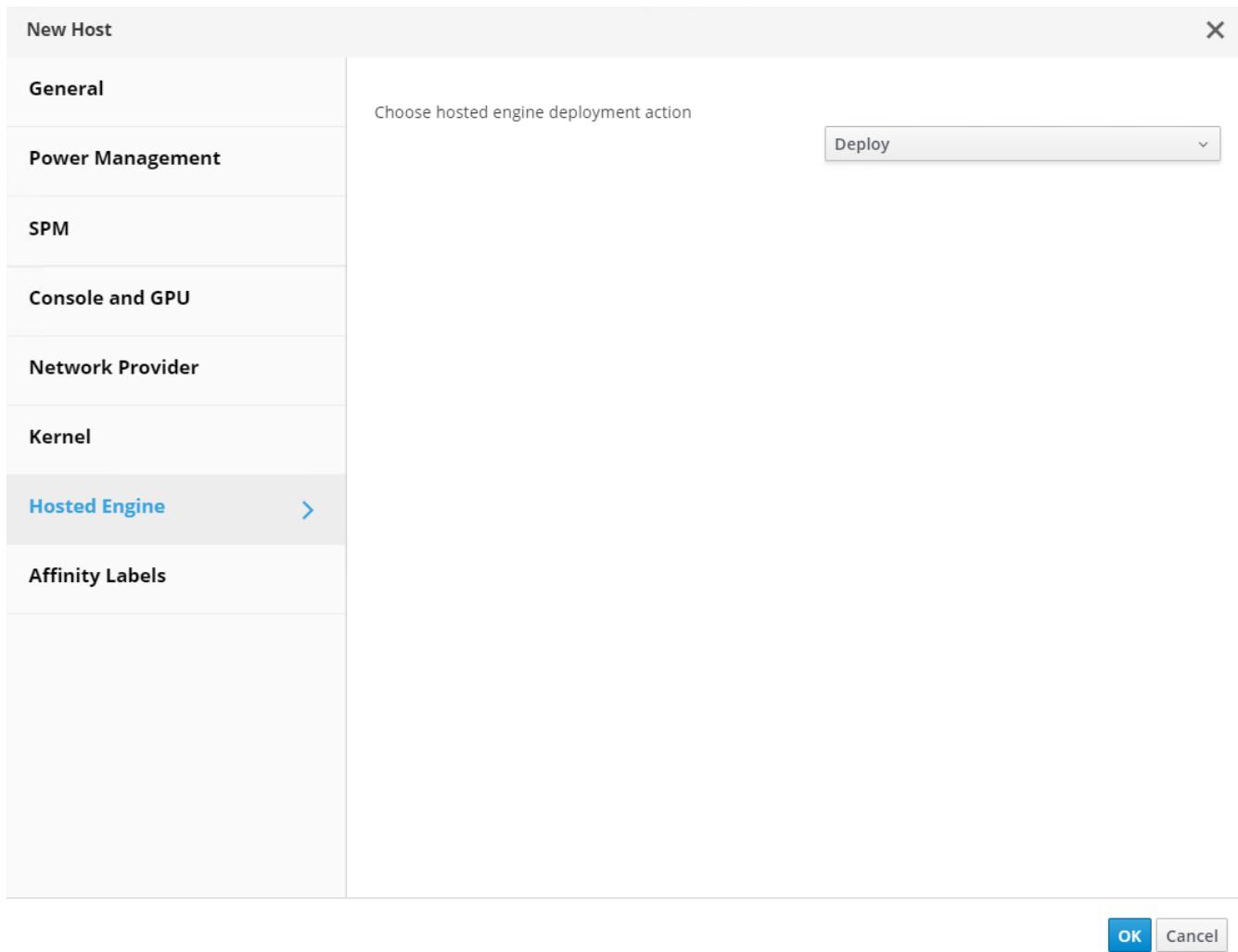
Password

SSH Public Key

Advanced Parameters

OK Cancel

18. Click the Hosted Engine sub-tab in the New Host pane dropdown and select Deploy from the hosted engine deployment action. Click OK to add the host to the quorum. This begins the installation of the necessary packages to support the hosted engine and activate the host. This process might take a while.



19. Next, create a storage virtual network for hosts. Navigate to Network > Networks and click New. Enter the name of your choice, enable VLAN tagging, and enter the VLAN ID for the Storage network. Confirm that the VM Network checkbox is checked and that the MTU is set to 9000. Go to the Cluster sub-tab and make sure that Attach and Require are checked. Then click OK to create the storage network.

New Logical Network

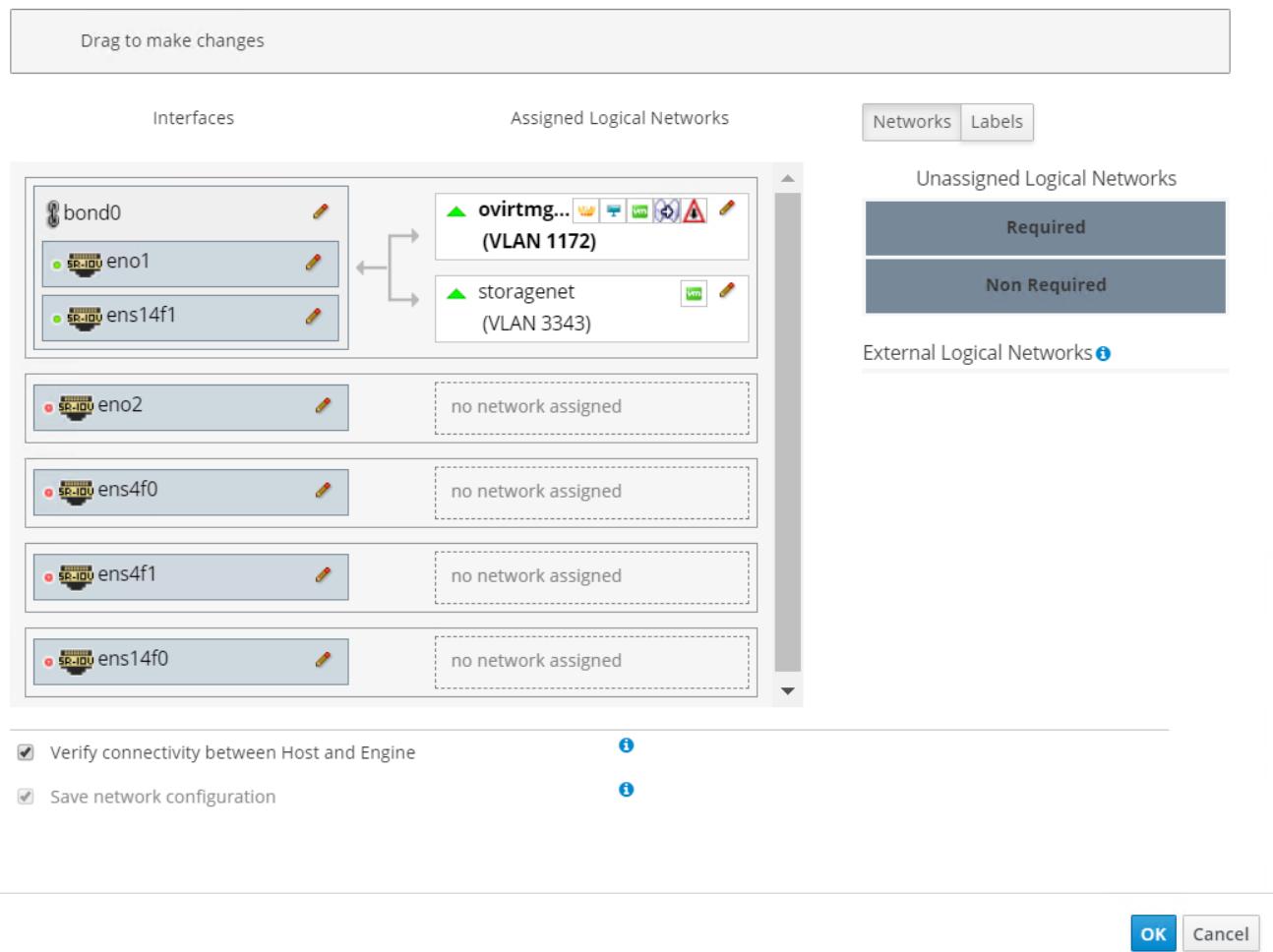
<b>General</b>	Data Center	Default
<b>Cluster</b>	Name <small>i</small>	storagenet
<b>vNIC Profiles</b>	Description	
	Comment	
<b>Network Parameters</b>		
	Network Label	
	<input checked="" type="checkbox"/> Enable VLAN tagging	3343
	<input checked="" type="checkbox"/> VM network <small>vm</small>	
	MTU	<input type="radio"/> Default (1500) <input checked="" type="radio"/> Custom 9000
	Host Network QoS	[Unlimited]

**OK** **Cancel**

20. Assign the storage logical network to the second host in the cluster or to whichever host is not currently hosting the hosted engine VM.
21. Navigate to Compute > Hosts, and click the host that has silver crown in the second column. Then navigate to the Network Interfaces sub-tab, click Setup Host Networks, and drag and drop the storage logical network into the Assigned Logical Networks column to the right of bond0.

## Setup Host rhv-h02.cie.netapp.com Networks

X



22. Click the pen symbol on the storage network interface under bond0. Configure the IP address and the netmask, and then click OK. Click OK again in the Setup Host Networks pane.

Edit Network storagenet

**IPv4**

Sync network i

**IPv6**

**QoS**

**Custom Properties**

**DNS Configuration**

Boot Protocol  
 None  
 DHCP  
 Static

IP: 172.21.87.33

Netmask / Routing Prefix: 24

Gateway:

**OK** **Cancel**

23. Migrate the hosted engine VM to the host that was just configured so that the storage logical network can be configured on the second host. Navigate to Compute > Virtual Machines, click HostedEngine and then click Migrate. Select the second host from the dropdown menu Destination Host and click Migrate.

Migrate VM(s)

Select a host to migrate 1 virtual machine(s) to:

**Destination Host** i: rhv-h02.cie.netapp.com

**Migrate VMs in Affinity** i:  Migrate all VMs in positive enforcing affinity with selected VMs.

**Virtual Machines**: HostedEngine

**Cancel** **Migrate**

After the migration is successful and the hosted engine VM is migrated to the second host, repeat steps 21 and 22 for the host that currently possesses the silver crown.

24. After you have completed this process, you should see that both the hosts are up. One of the hosts has a golden crown, indicating that it is hosting the hosted engine VM, and the other host has a silver crown indicating that it is capable of hosting the hosted engine VM.

## 6. Configure RHV-M Infrastructure: NetApp HCI with RHV

To configure the RHV-M infrastructure, complete the following steps:

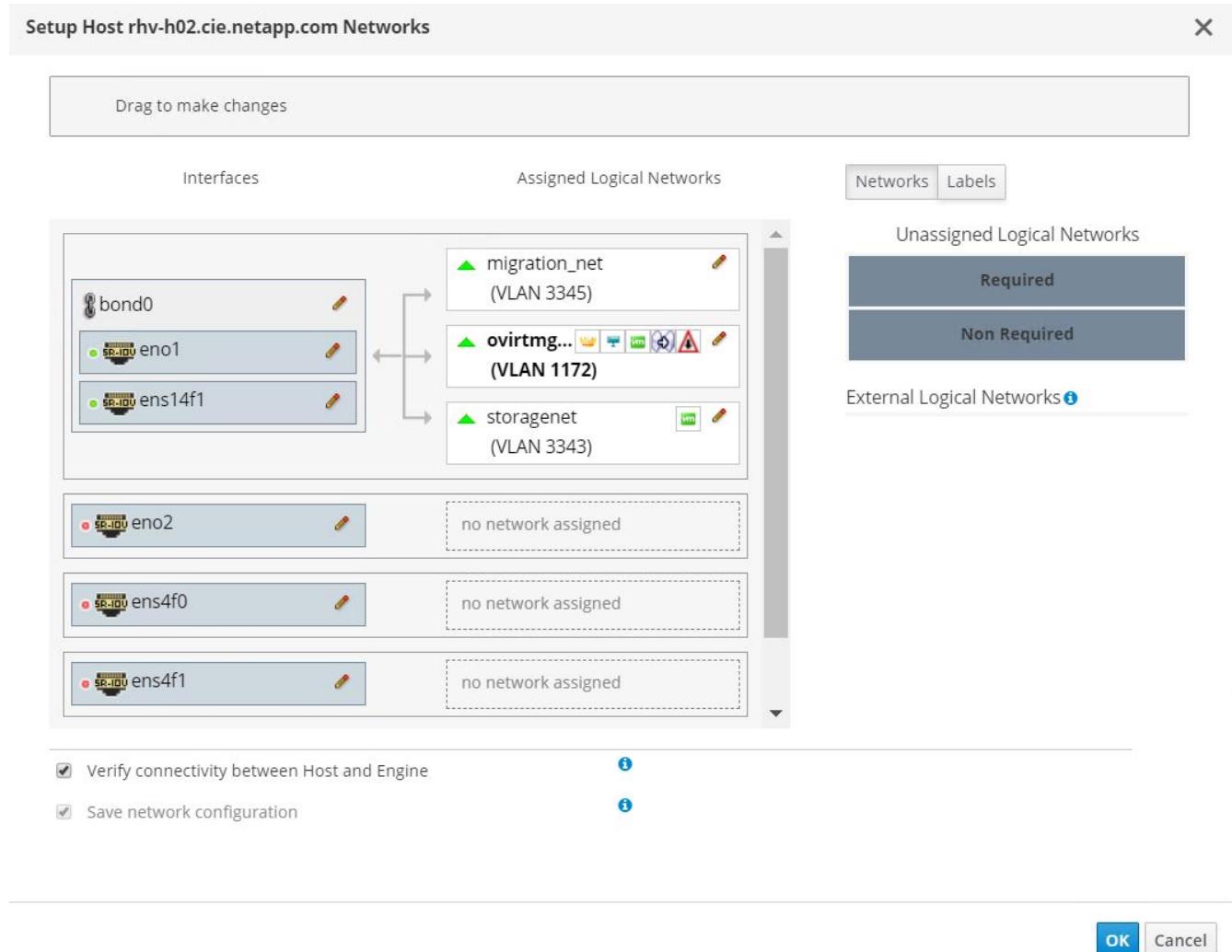
1. By default, the ovirtmgmt network is used for all purposes, including the migration of VMs and virtual guest data.
2. It is a best practice to specify different networks for these purposes. To configure the migration network, navigate to Network > Networks and click New. Enter the name of your choice, enable VLAN tagging, and enter the VLAN ID for the migration network.
3. Make sure that the VM Network checkbox is unchecked. Go to the Cluster sub-tab and make sure that Attach and Require are checked. Then click OK to create the network.

<b>Name</b>	migration_net
<b>Network Label</b>	
<b>Enable VLAN tagging</b>	3345
<b>VM network</b>	<input type="checkbox"/>
<b>MTU</b>	<input type="radio"/> Default (1500) <input type="radio"/> Custom
<b>Host Network QoS</b>	[Unlimited]

4. To assign the migration logical network to both the hosts, navigate to Compute > Hosts, click the

hosts, and navigate to the Network Interfaces sub-tab.

5. Then click Setup Host Networks and drag and drop the migration logical network into the Assigned Logical Networks column to the right of bond0.



6. Click the pen symbol on the migration network interface under bond0. Configure the IP address details and click OK. Then click OK again in the Setup Host Networks pane.

Edit Network migration\_net

**IPv4**

Sync network i

**IPv6**

**QoS**

**Custom Properties**

**DNS Configuration**

Boot Protocol  
 None  
 DHCP  
 Static

IP: 172.21.89.10

Netmask / Routing Prefix: 24

Gateway:

**OK** **Cancel**

7. Repeat steps 4 through 6 for the other host as well.
8. The newly created network must be assigned the role of the migration network. Navigate to Compute > Clusters and click the cluster that the RHV hosts belong to, click the Logical Networks sub-tab, and click Manage Networks. For the migration network, enable the checkbox under Migration Network column. Click OK.

**Manage Networks**

Name	<input checked="" type="checkbox"/> Assign All	<input checked="" type="checkbox"/> Require All	VM Network	Management	Display Network	Migration Network
ovirtmgmt	<input checked="" type="checkbox"/> Assign	<input checked="" type="checkbox"/> Require		<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
migration_net	<input checked="" type="checkbox"/> Assign	<input checked="" type="checkbox"/> Require		<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
storagenet	<input checked="" type="checkbox"/> Assign	<input checked="" type="checkbox"/> Require		<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

9. Next, as a best practice, create a separate VM network rather than using the ovirtmgmt network for VMs.
10. Navigate to Network > Networks and click New. Enter the name of your choice, enable VLAN tagging, and enter the VLAN ID for the VM guest network. Make sure that the checkbox VM Network is checked. Go to the Cluster's sub-tab and make sure that Attach and Require are checked. Then click OK to create the VM guest network.

New Logical Network

<b>General</b>	<b>Data Center</b>	Default
<b>Cluster</b>	<b>Name</b>	vGuest
<b>vNIC Profiles</b>	<b>Description</b>	
	<b>Comment</b>	
	<b>Network Parameters</b>	
	<b>Network Label</b>	
	<input checked="" type="checkbox"/> Enable VLAN tagging	3346
	<input checked="" type="checkbox"/> VM network	
	<b>MTU</b>	<input checked="" type="radio"/> Default (1500) <input type="radio"/> Custom
	<b>Host Network QoS</b>	[Unlimited]

- Assign the VM guest logical network to both the hosts. Navigate to Compute > Hosts, click the host names and navigate to the Network Interfaces sub-tab. Then click Setup Host Networks and drag and drop the VM guest logical network into the Assigned Logical Networks column to the right of bond0. There is no need to assign an IP to this logical network, because it provides passthrough networking for the VMs.

The VM guest network should be able to reach the internet to allow guests to register with Red Hat Subscription Manager.

## 7. Deploy the NetApp mNode: NetApp HCI with RHV

The management node (mNode) is a VM that runs in parallel with one or more Element software-based storage clusters. It is used for the following purposes:

- Providing system services including monitoring and telemetry
- Managing cluster assets and settings
- Running system diagnostic tests and utilities
- Enabling callhome for NetApp ActiveIQ for additional support

To install the NetApp mNode on Red Hat Virtualization, complete the following steps:

- Upload the mNode ISO as a disk to the storage domain. Navigate to Storage > Disks > Upload and

click Start. Then click Upload Image and select the downloaded mNode ISO image. Verify the storage domain, the host to perform the upload, and additional details. Then click OK to upload the image to the domain. A progress bar indicates when the upload is complete and the ISO is usable.

2. Create a VM disk by navigating to Storage > Disks and click New. The mNode disk must be at least 400 GB in size but can be thin-provisioned. In the wizard, enter the name of your choice, select the proper data center, make sure that the proper storage domain is selected, select Thin Provisioning for the allocation policy, and check the Wipe After Delete checkbox. Click OK.

**New Virtual Disk**

Image	Direct LUN	Cinder	Managed Block
Size (GiB)	400		
Alias	mNode_disk		
Description			
Data Center	Default		
Storage Domain	data_domain (1784 GiB free of 1907 GiB)		
Allocation Policy	Thin Provision		
Disk Profile	data_domain		
<input checked="" type="checkbox"/> Wipe After Delete <input type="checkbox"/> Shareable			

3. Next, navigate to Compute > Virtual Machines and click New. In the General sub-tab, select the appropriate cluster, enter the name of your choice, click attach, and select the disk created in the previous step. Check the box below OS to emphasize that it is a bootable drive. Click OK.

**Attach Virtual Disks**

Image	Direct LUN	Cinder	Managed Block
<input checked="" type="radio"/>	mNode_disk		
ID	0438434a-9...	Virtual Size	Actual Size
	400 GiB	1 GiB	
Storage Domain		Interface	R/O
data_domain	VirtIO	<input type="checkbox"/>	<input checked="" type="checkbox"/>

4. Select ovirtmgmt from the dropdown for nic1. Click the (+) sign and select the storage network interface from the dropdown list for nic2.

New Virtual Machine X

General >	
Cluster	<input type="text" value="Default"/> <span style="float: right;">▼</span> <small>Data Center: Default</small>
Template	<input type="text" value="Blank   (0)"/> <span style="float: right;">▼</span>
Operating System	<input type="text" value="Other OS"/> <span style="float: right;">▼</span>
Instance Type	<input type="text" value="Custom"/> <span style="float: right;">▼</span>
Optimized for	<input type="text" value="Server"/> <span style="float: right;">▼</span>
Name	<input type="text" value="NetApp mNode"/>
Description	<input type="text"/>
Comment	<input type="text"/>
VM ID	<input type="text"/>
<input type="checkbox"/> Stateless <input type="checkbox"/> Start in Pause Mode <input type="checkbox"/> Delete Protection	
<b>Custom Properties</b>	
Icon	<input type="text"/> <span style="float: right;">Edit <span style="border: 1px solid #ccc; padding: 0 5px;">+</span> <span style="border: 1px solid #ccc; padding: 0 5px;">-</span></span>
<b>Foreman/Satellite</b>	
Instantiate VM network interfaces by picking a vNIC profile.	
nic1	<input type="text" value="ovirtmgmt/ovirtmgmt"/> <span style="float: right;">▼</span>
nic2	<input type="text" value="storagenet/storagenet"/> <span style="float: right;">▼</span> <span style="float: right;">+ <span style="border: 1px solid #ccc; padding: 0 5px;">-</span></span>

Hide Advanced Options **OK** Cancel

- Click the System sub-tab and make sure that it has at least 12GB of memory and 6 virtual CPUs as recommended.

New Virtual Machine X

<b>General</b>	Cluster <span style="float: right;">Default <span style="font-size: small;">▼</span></span> <i>Data Center: Default</i>
<b>System</b> >	Template <span style="float: right;">Blank   (0) <span style="font-size: small;">▼</span></span> Operating System <span style="float: right;">Other OS <span style="font-size: small;">▼</span></span> Instance Type <span style="float: right;">Custom <span style="font-size: small;">▼</span></span> Optimized for <span style="float: right;">Server <span style="font-size: small;">▼</span></span>
<b>Initial Run</b>	
<b>Console</b>	
<b>Host</b>	
<b>High Availability</b>	
<b>Resource Allocation</b>	Memory Size <span style="float: right;">12288 MB</span> Maximum memory <span style="float: right;">49152 MB</span> Physical Memory Guaranteed <span style="float: right;">12288 MB</span> Total Virtual CPUs <span style="float: right;">6</span>
<b>Boot Options</b>	<span style="color: blue;">(+) Advanced Parameters</span>
<b>Random Generator</b>	
<b>Custom Properties</b>	General Hardware Clock Time Offset <span style="float: right;">default: (GMT+00:00) GMT Standard Time <span style="font-size: small;">▼</span></span>
<b>Icon</b>	<input type="checkbox"/> Provide custom serial number policy <span style="float: right;">(i)</span>
<b>Foreman/Satellite</b>	
<b>Affinity Labels</b>	

Hide Advanced Options OK Cancel

6. Click the Boot Options sub-tab, select CD-ROM as the first device in the boot sequence, select Hard Drive as the second device. Enable Attach CD and attach the mNode ISO. Then click OK.

New Virtual Machine X

<b>General</b>	Cluster <input type="text" value="Default"/> <small>Data Center: Default</small>	
<b>System</b>	Template <input type="text" value="Blank   (0)"/>	
<b>Initial Run</b>	Operating System <input type="text" value="Other OS"/>	
<b>Console</b>	Instance Type <input type="text" value="Custom"/>	
<b>Host</b>	Optimized for <input type="text" value="Server"/>	
<b>High Availability</b>	Boot Sequence:	
	First Device <input type="text" value="CD-ROM"/>	
	Second Device <input type="text" value="Hard Disk"/>	
<b>Resource Allocation</b>	<input checked="" type="checkbox"/> Attach CD <input type="text" value="solidfire-fdva-sodium-patch5-11.5.0."/> <small>?</small>	
<b>Boot Options</b> >	<input type="checkbox"/> Enable menu to select boot device	
<b>Random Generator</b>		
<b>Custom Properties</b>		
<b>Icon</b>		
<b>Foreman/Satellite</b>		
<b>Affinity Labels</b>		

Hide Advanced Options **OK** **Cancel**

The VM is created.

- After the VM becomes available, power it on, and open a console to it. It begins to load the NetApp Solidfire mNode installer. When the installer is loaded, you are prompted to start the RTFI magnesium installation; type **yes** and press Enter. The installation process begins, and after it is complete, it automatically powers off the VM.



.....

Starting SolidFire RTFI magnesium

Proceed (Yes,No)

yes

8. Next, click the mNode VM and click Edit. In the Boot Options sub-tab, uncheck the Attach CD checkbox and click the OK button.

Edit Virtual Machine X

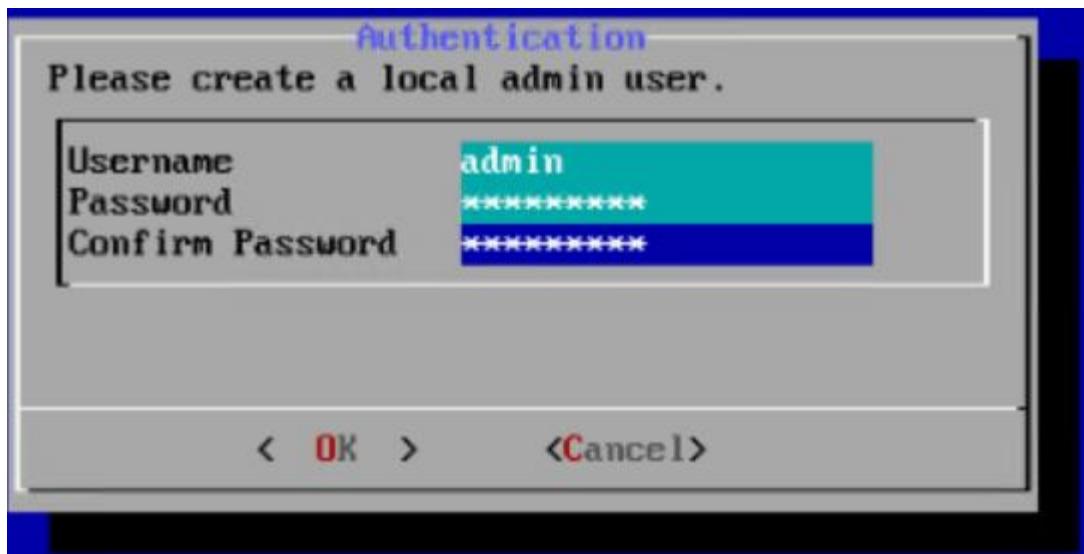
<b>General</b>	Cluster <input type="text" value="Default"/> <small>Data Center: Default</small>	
<b>System</b>	Template <input type="text" value="Blank   (0)"/>	
<b>Initial Run</b>	Operating System <input type="text" value="Other OS"/>	
<b>Console</b>	Instance Type <input type="text" value="Custom"/>	
<b>Host</b>	Optimized for <input type="text" value="Server"/>	
<b>High Availability</b>	<b>Boot Sequence:</b>	
	First Device <input type="text" value="CD-ROM"/>	
	Second Device <input type="text" value="Hard Disk"/>	
	<input type="checkbox"/> Attach CD <input type="text" value="solidfire-fdva-magnesium-12.0.0.333"/> <span style="float: right;"></span>	
	<input type="checkbox"/> Enable menu to select boot device	
<b>Boot Options</b> >		
<b>Random Generator</b>		
<b>Custom Properties</b>		
<b>Icon</b>		
<b>Foreman/Satellite</b>		
<b>Affinity Labels</b>		

Hide Advanced Options OK Cancel

- Power on the mNode VM. Using the terminal user interface (TUI), create a management node admin user.



To move through the menu options, press the Up or Down arrow keys. To move through the buttons, press Tab. To move from the buttons to the fields, press Tab. To navigate between fields, press the Up or Down arrow keys.



10. After the user is created, you are returned to a login screen. Log in with the credentials that were just created.
11. To configure the network interfaces starting with the management interface, navigate to Network > Network Config > eth0 and enter the IP address, netmask, gateway, DNS servers, and search domain for your environment. Click OK.

NetApp Management Node -> Network -> Network Config -> eth0

Hit 'tab' to navigate between the form and buttons. Use ↑/↓ to navigate between fields. Start typing or hit ←/→ to enter the field to make changes. Press 'enter' with a field selected, or hit 'tab' then 'enter' to submit all pending changes.

\* denotes required fields.

Method:	static
Link speed:	0
*IPv4 Address:	10.63.172.141
*IPv4 Subnet_Mask:	255.255.255.0
IPv4 Gateway:	10.63.172.1
Mtu:	1500
Dns:	10.61.184.251, 10.61.184.252
Domains:	cie.netapp.com
IPv6 Address:	
IPv6 Gateway:	
*Status:	UpAndRunning
Ulan:	0

< OK >

<Cancel>

< Help >

12. Next, configure eth1 to access the storage network. Navigate to Network > Network Config > eth1 and enter the IP address and netmask. Verify that the MTU is 9000. Then click OK.

NetApp Management Node -> Network -> Network Config -> eth1

Hit 'tab' to navigate between the form and buttons. Use ↑↓ to navigate between fields. Start typing or hit ←→ to enter the field to make changes. Press 'enter' with a field selected, or hit 'tab' then 'enter' to submit all pending changes.  
\* denotes required fields.

Method:	dhcp
Link speed:	0
IPv4 Address:	172.21.87.141
IPv4 Subnet_Mask:	255.255.255.0
IPv4 Gateway:	
Mtu:	9000
*Status:	UpAndRunning
Vlan:	0

< OK >      <Cancel>      < Help >

You can now close the TUI interface.

13. SSH into the management node using the management IP, escalate to root and register the mNode with the HCI storage cluster.

```
admin@SF-3D1C ~ $ sudo su
```

```
SF-3D1C /home/admin # /sf/packages/mnode/setup-mnode --mnode_admin_user admin  
--storage_mvip 10.63.172.140 --storage_username admin --telemetry_active true
```

Enter the password for storage user admin:

Enter password for mNode user admin:

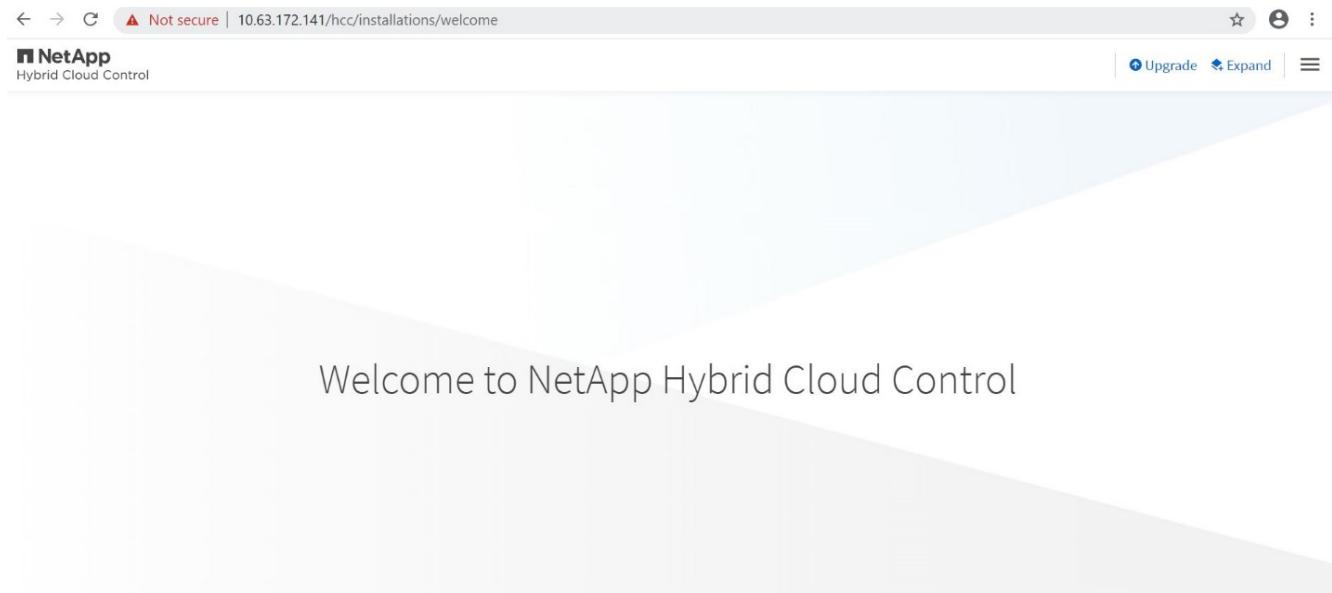
```
[2020-05-21T17:19:53.281657Z]:[setup_mnode:296] INFO:Starting mNode deployment  
[2020-05-21T17:19:53.286153Z]:[config_util:1313] INFO>No previously running mNode.  
Continuing with deployment.
```

```
[2020-05-21T17:19:53.286687Z]:[config_util:1320] INFO:Validating credentials for mNode  
host.  
[2020-05-21T17:19:53.316270Z]:[config_util:1232] INFO:Checking Cluster information.  
[2020-05-21T17:19:53.380168Z]:[config_util:112] INFO:Cluster credentials verification  
successful.
```

```
[2020-05-21T17:19:53.380665Z]:[config_util:1252] INFO:Cluster version check  
successful.  
[2020-05-21T17:19:53.458271Z]:[config_util:112] INFO:Successfully queried system  
configuration  
[2020-05-21T17:19:53.463611Z]:[config_util:497] INFO:CIDR range 172.16.0.0/22 open.  
Using for docker ingress.  
[2020-05-21T17:19:53.464179Z]:[mnodcfg:141] INFO:Configuring mNode  
[2020-05-21T17:19:53.464687Z]:[config_util:194] INFO:Wait for ping of 127.0.0.1 to  
succeed  
[2020-05-21T17:19:53.475619Z]:[mnodcfg:145] INFO:Validating the supplied MNode  
network configuration  
[2020-05-21T17:19:53.476119Z]:[mnodcfg:155] INFO:Testing the MNode network  
configuration  
[2020-05-21T17:19:53.476687Z]:[config_util:353] INFO:Testing network connection to  
storage MVIP: 10.63.172.140  
[2020-05-21T17:19:53.477165Z]:[config_util:194] INFO:Wait for ping of 10.63.172.140 to  
succeed  
[2020-05-21T17:19:53.488045Z]:[config_util:356] INFO:Successfully reached storage  
MVIP: 10.63.172.140  
[2020-05-21T17:19:53.488569Z]:[mnodcfg:158] INFO:Configuring MNode storage (this can  
take several minutes)  
[2020-05-21T17:19:57.057435Z]:[config_util:536] INFO:Configuring MNode storage  
succeeded.  
[2020-05-21T17:19:57.057938Z]:[config_util:445] INFO:Replacing default ingress  
network.  
[2020-05-21T17:19:57.078685Z]:[mnodcfg:163] INFO:Extracting services tar (this can  
take several minutes)  
[2020-05-21T17:20:36.066185Z]:[config_util:1282] INFO:Extracting services tar  
succeeded  
[2020-05-21T17:20:36.066808Z]:[mnodcfg:166] INFO:Configuring MNode authentication  
[2020-05-21T17:20:36.067950Z]:[config_util:1485] INFO:Updating element-auth  
configuration  
[2020-05-21T17:20:41.581716Z]:[mnodcfg:169] INFO:Deploying MNode services (this can  
take several minutes)  
[2020-05-21T17:20:41.810264Z]:[config_util:557] INFO:Deploying MNode services  
succeeded  
[2020-05-21T17:20:41.810768Z]:[mnodcfg:172] INFO:Deploying MNode Assets  
[2020-05-21T17:20:42.162081Z]:[config_util:122] INFO:Retrying 1/45 time...  
[2020-05-21T17:20:42.162640Z]:[config_util:125] INFO:Waiting 10 seconds before next  
attempt.  
[2020-05-21T17:20:52.199224Z]:[config_util:112] INFO:Mnode is up!  
[2020-05-21T17:20:52.280329Z]:[config_util:112] INFO:Root asset created.  
[2020-05-21T17:20:52.280859Z]:[config_util:122] INFO:Retrying 1/5 time...  
[2020-05-21T17:20:52.281280Z]:[config_util:125] INFO:Waiting 10 seconds before next  
attempt.  
[2020-05-21T17:21:02.299565Z]:[config_util:112] INFO:Successfully queried storage  
assets  
[2020-05-21T17:21:02.696930Z]:[config_util:112] INFO:Storage asset created.
```

```
[2020-05-21T17:21:03.238455Z]:[config_util:112] INFO:Storage asset registered.
[2020-05-21T17:21:03.241966Z]:[mnnodecfg:175] INFO:Attempting to set up VCP-SIOC
credentials
[2020-05-21T17:21:03.242659Z]:[config_util:953] INFO:No VCP-SIOC credential given from
NDE. Using default credentials for VCP-SIOC service.
[2020-05-21T17:21:03.243117Z]:[mnnodecfg:185] INFO:Configuration Successfully Completed
```

14. Using a browser, log into the management node GUI using <https://<mNodeIP>>. mNode or Hybrid Cloud Control facilitates expansion, monitoring, and upgrading the Element cluster.



15. Click the three parallel lines on the top right and click View Active IQ. Search for the HCI storage cluster by filtering the cluster name and make sure that it is logging the most recent updates.

 A screenshot of the Active IQ dashboard. The top navigation bar includes 'Active IQ', 'All Clusters View', 'Select a Cluster', and 'Admin'. The right side shows the user 'Network Appliance, Inc kulkarni'. The left sidebar has sections for 'Dashboard', 'Alerts', and 'Capacity Licensing'. The main content area is titled 'Overview' and shows a table of storage clusters. The columns are: Company, Cluster, Cluster ID, Version, Nodes, Volumes, Efficiency, Used Block Capacity %, Faults, SVIP, MVIP, and Last Update. One row is visible: NetApp Inc., RHV-Store, 1913154, 12.0.0.333, 4, 2, 149.4x, 0.2%, 0, 172.21.87.140, 10.63.172.140, 2020-05-21 10:28:56.
 

Company	Cluster	Cluster ID	Version	Nodes	Volumes	Efficiency	Used Block Capacity %	Faults	SVIP	MVIP	Last Update
NetApp Inc.	RHV-Store	1913154	12.0.0.333	4	2	149.4x	0.2%	0	172.21.87.140	10.63.172.140	2020-05-21 10:28:56

## Best Practices for Production Deployments

### Updating RHV Manager and RHV-H Hosts: NetApp HCI with RHV

It is a recommended best practice to make sure that both the RHV Manager and the RHV-H hosts have the latest security and stability updates applied to make sure that the environment is protected and continues to run as expected. To apply the updates to the hosts in the deployment, they must first be subscribed to either the

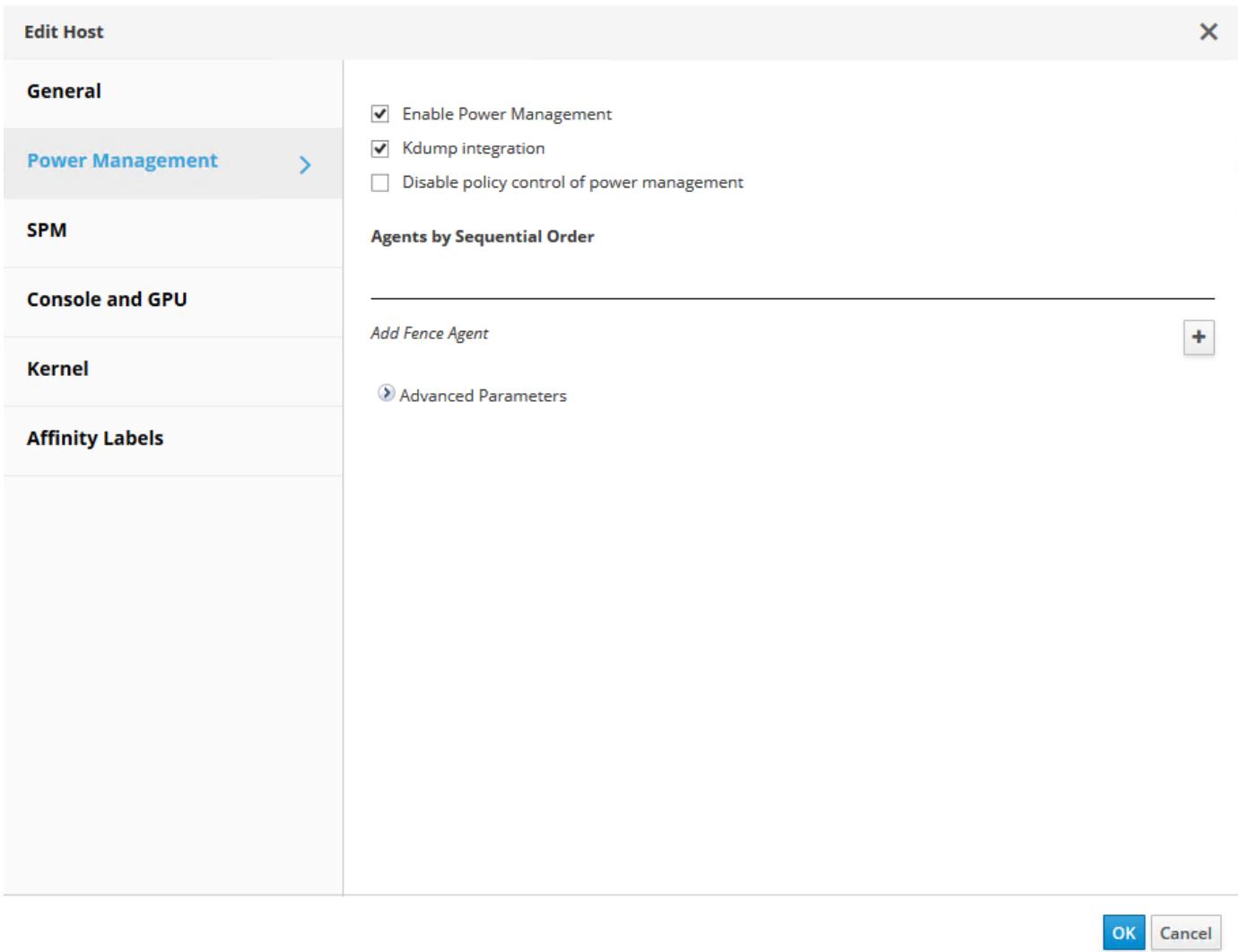
Red Hat Content Delivery Network or a local Red Hat Satellite repository. The tasks involved in updating the platform include updating the manager VM and afterward updating each physical host non-disruptively after ensuring virtual guests are migrated to another node in the cluster.

Official documentation to support the upgrade of RHV 4.3 between minor releases can be found [here](#).

#### **Enabling Fencing for RHV-H Hosts: NetApp HCI with RHV**

Fencing is a process by which the RHV Manager can provide high availability of the VMs in the environment by automatically shutting down a non-responsive hypervisor host. It does this by sending commands to a fencing agent, which in the case of NetApp HCI is available through the IPMI out-of-band management interface on the compute nodes and rebooting the host. This action releases the locks that the non-responsive hypervisor node has on VM disks and allows for those virtual guests to be restarted on another node in the cluster without risking data corruption. After the host completes its boot process, it automatically attempts to rejoin the cluster it was a part of prior to the shutdown. If it is successful, it is once again allowed to host VMs.

To enable fencing, each host must have power management enabled; this can be found by highlighting the host and clicking the Edit button in the upper right-hand corner or by right-clicking on the host and selecting Edit.



After power management is enabled, the next step involves configuring a fencing agent. Click on the plus sign (+) near the Add Fence Agent, and a new window pops up that must be filled out with the information for the IPMI connection on the NetApp HCI compute nodes. The type of connection is IPMILAN, and the agent needs the IP address, username, and password for the console login. After you have provided this information, you can click test to validate the configuration. If properly configured, it should report the current power status of the node.

Edit fence agent

X

Address	<input type="text" value="172.16.14.31"/>
User Name	<input type="text" value="ADMIN"/>
Password	<input type="password" value="*****"/>
Type	<input style="width: 100px;" type="text" value="ipmilan"/> ▼
Options	<input type="text"/>

Please use a comma-separated list of 'key=value'

Test successful: power on

---

With fencing enabled, the RHV environment is configured to support a highly available deployment should one of the hypervisor nodes become nonresponsive.

### Optimizing Memory for Red Hat Virtualization: NetApp HCI with RHV

One of the primary benefits for deploying a virtual infrastructure is to enable the more efficient use of physical resources in the environment. In a case in which the guest VMs underutilize the memory allotted, you can use memory overcommitment to optimize memory usage. With this feature, the sum of the memory allocated to guest VMs on a host is allowed to exceed the amount of physical memory on that host.

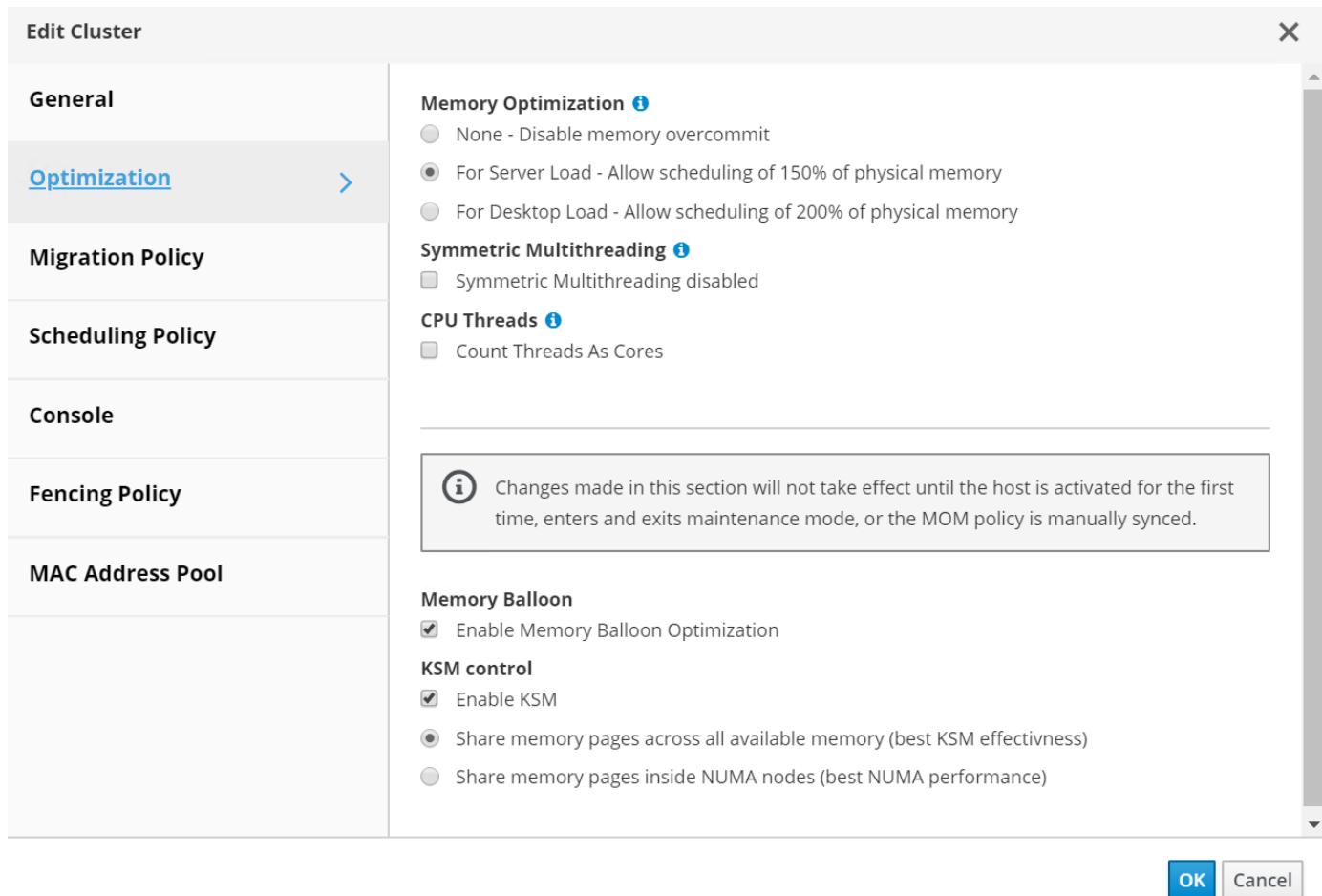
The concept behind memory overcommitment is similar to thin provisioning of storage resources. At any given moment, every VM on the host does not use the total amount of memory allocated to it. When one VM has excess memory, its unused memory is available for other VMs to use. Therefore, an end user can deploy more VMs than the physical infrastructure would normally allow. Memory overcommitment on the hosts in the cluster is handled by Memory Overcommit Manager (MoM). Techniques like memory ballooning and Kernel Same-page Merging (KSM) can improve memory overcommitment depending on the kind of workload.

Memory ballooning is a memory management technique which allows a host to artificially expand its memory by reclaiming unused memory that was previously allocated to various VMs, with a limitation of the guaranteed memory size of every VM. For memory ballooning to work, each VM by default has a balloon device with the necessary drivers. Ballooning essentially is a cooperative operation between the VM driver and the host. Depending on the memory needs of the host, it instructs the guest OS to inflate (provide memory to host) or deflate (regain the memory) the balloon which is controlled by the balloon device.

Kernel Same-page Merging (KSM) allows the host kernel to examine two or more running VMs and compare their image and memory. If any memory regions or pages are identical, KSM reduces multiple identical memory pages to a single page. This page is then marked ‘copy on write’ and a new page is created for that guest VM if the contents of the page are modified by a guest VM.

Both features can be enabled at a cluster level to apply to all hosts in that cluster. To enable these features, navigate to Compute > Clusters, select the desired cluster and click Edit. Then click the Optimization sub-tab and perform the following steps based on your requirements:

1. Depending on the use-case and workload, enable Memory Optimization to allow overcommitment of memory to either 150% or 200% of the available physical memory.
2. To enable memory ballooning, check the Enable Memory Balloon Optimization checkbox.
3. To enable KSM, check the Enable KSM checkbox.
4. Click Ok to confirm the changes.



Be aware that after these changes have been applied, they do not take effect until you manually sync the MoM policy. To sync the MoM policy, navigate to Compute > Clusters and click the cluster for which you made the optimization changes. Navigate to the Hosts sub-tab, select all the hosts, and then click Sync MoM Policy.

Compute » Clusters » Default =

Edit Remove Upgrade ⋮

General	Logical Networks	Hosts	Virtual Machines	Affinity Groups	Affinity Labels	CPU Profiles	Permissions
Red Hat Documentation							
<span style="border: 1px solid #ccc; padding: 2px 10px;">Sync MoM Policy</span>							
1 - 2 <span style="border: 1px solid #ccc; padding: 0 5px;">&lt;</span> <span style="border: 1px solid #ccc; padding: 0 5px;">&gt;</span>							
Name	Hostname/IP	Status	Load	Display Address Overridden			
▲ rhv-h01.cie.netapp.com	rhev-h01.cie.netapp.com	Up	3 VMs	No			
▲ rhv-h02.cie.netapp.com	rhev-h02.cie.netapp.com	Up	5 VMs	No			

KSM and ballooning can free up some memory on the host and facilitate overcommitment, but, if the amount of shareable memory decreases and the use of physical memory increases, it might cause an out-of-memory condition. Therefore, the administrator should be sure to reserve enough memory to

avoid out-of-memory conditions if the shareable memory decreases.

In some scenarios, memory ballooning may collide with KSM. In such situations, MoM tries to adjust the balloon size to minimize collisions. Also, there can be scenarios for which ballooning might cause sub-optimal performance. Therefore, depending on the workload requirements, you can consider enabling either or both the techniques.

## Where to Find Additional Information: NetApp HCI with RHV

To learn more about the information described in this document, review the following documents and/or websites:

- NetApp HCI Documentation <https://www.netapp.com/us/documentation/hci.aspx>
- Red Hat Virtualization Documentation [https://access.redhat.com/documentation/en-us/red\\_hat\\_virtualization/4.3/](https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/)

## TR-4857: NetApp HCI with Cisco ACI

Abhinav Singh, Nikhil M Kulkarni, NetApp

Cisco Application Centric Infrastructure (Cisco ACI) is an industry-leading, secure, open, and comprehensive Software-Defined Networking (SDN) solution. Cisco ACI radically simplifies, optimizes, and accelerates infrastructure deployment and governance, and it expedites the application deployment lifecycle. Cisco ACI deployed in data centers is proven to work with NetApp HCI with full interoperability. You can manage Ethernet networks for compute, storage, and access with Cisco ACI. You can establish and manage secure network segments for server-to-server and virtual machine (VM)-to-VM communications as well as secure storage-network access through iSCSI from server-to-NetApp HCI storage. This level of endpoint-to-endpoint network security allows customers to architect and operate NetApp HCI in a more secure fashion.

[Next: Use Cases](#)

## Use Cases

The NetApp HCI with Cisco ACI solution delivers exceptional value for customers with the following use cases:

- On-premises software-defined compute, storage, and networking infrastructure
- Large enterprise and service-provider environments
- Private cloud (VMware and Red Hat)
- End User Computing and Virtual Desktop Infrastructure
- Mixed-workload and mixed-storage environments

[Next: Architecture](#)

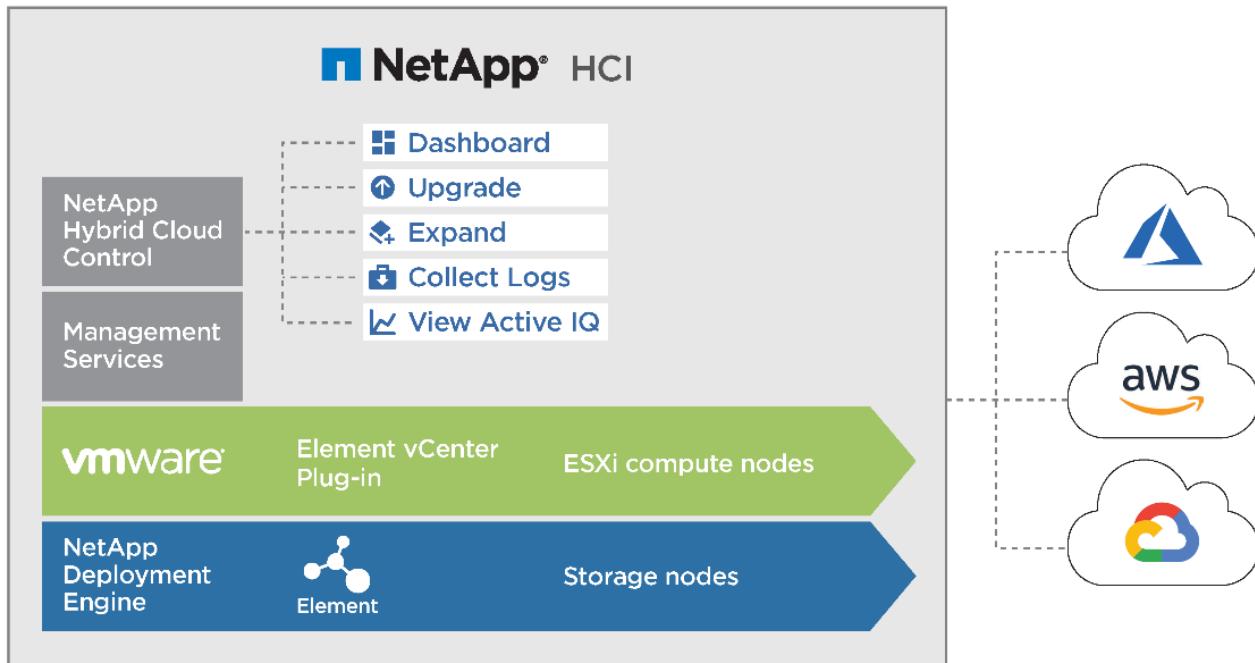
## Architecture

### Solution Technology

This document outlines the best practices to follow for a fully featured on-premises data center or private cloud while interoperating NetApp HCI with Cisco ACI. To demonstrate workload independence, networking best practices are extended to virtualization solutions, including VMware vSphere and Red Hat Virtualization when deployed over NetApp HCI, and to other storage solutions like NetApp ONTAP and StorageGRID. It also emphasizes the interoperability of Cisco ACI switches with different virtual switches, for example, VMware Distributed Switch (VDS), Cisco ACI Virtual Edge (AVE), Linux Bridge, or Open vSwitch.

### NetApp HCI

NetApp HCI is an enterprise-scale, hyper-converged infrastructure solution that delivers compute and storage resources in an agile, scalable, easy-to-manage architecture. Running multiple enterprise-grade workloads can result in resource contention, where one workload interferes with the performance of another. NetApp HCI alleviates this concern with storage quality-of-service (QoS) limits that are available natively within NetApp Element software. Element enables the granular control of every application and volume, helps to eliminate noisy neighbors, and satisfies enterprise performance SLAs. NetApp HCI multitenancy capabilities can help eliminate many traditional performance related problems. See the following graphic for an overview of NetApp HCI.



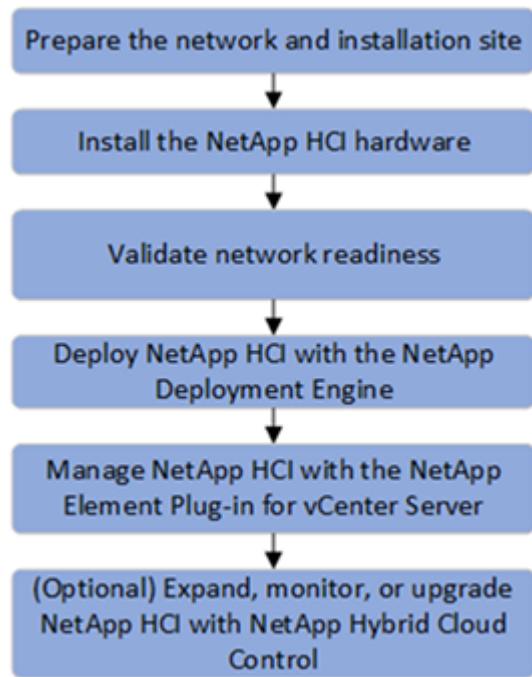
NetApp HCI streamlines installation through the NetApp Deployment Engine (NDE), an intuitive

deployment engine that automates more than 400 inputs to fewer than 30 to get your setup running in about 45 minutes. In addition, a robust suite of APIs enables seamless integration into higher-level management, orchestration, backup, and disaster recovery tools. With the NetApp Hybrid Cloud Control management suite, you can manage, monitor, and upgrade your entire infrastructure throughout its lifecycle through a single pane of glass.

### Software-Defined Architecture

NetApp HCI provides a software-defined approach for deploying and managing data and storage resources. NetApp HCI uses NetApp Element software to provide an easy-to-use GUI-based portal and REST-based API for storage automation, configuration, and management. NetApp Element software provides modular and scalable performance, with each storage node delivering guaranteed capacity and throughput to the environment.

NetApp HCI uses the NetApp Deployment Engine (NDE) to automate the configuration and deployment of physical infrastructure, including the installation and configuration of the VMware vSphere environment and the integration of the NetApp Element Plug-in for vCenter Server. The following figure depicts an overview of the process for deploying NetApp HCI.



### Performance Guarantee

A common challenge is delivering predictable performance when multiple applications are sharing the same infrastructure. An application interfering with other applications creates performance degradation. Mainstream applications have unique I/O patterns that can affect each other's performance when deployed in a shared environment. To address these issues, the NetApp HCI Quality of Service (QoS) feature allows fine-grained control of performance for every application, thereby eliminating noisy neighbors and satisfying performance SLAs. In NetApp HCI, each volume is configured with minimum, maximum, and burst IOPS values. The minimum IOPS setting guarantees

performance, independent of what other applications on the system are doing. The maximum and burst values control allocation, enabling the system to deliver consistent performance to all workloads.

NetApp Element software uses the iSCSI storage protocol, a standard way to encapsulate SCSI commands on a traditional TCP/IP network. Element uses a technique called iSCSI login redirection for better performance. iSCSI login redirection is a key part of the NetApp Element software cluster. When a host login request is received, the node decides which member of the cluster should handle the traffic based on IOPS and the capacity requirements for the volume. Volumes are distributed across the NetApp Element software cluster and are redistributed if a single node is handling too much traffic for its volumes or if a new node is added. Multiple copies of a given volume are allocated across the array. In this manner, if a node failure is followed by volume redistribution, there is no effect on host connectivity beyond a logout and login with redirection to the new location. With iSCSI login redirection, a NetApp Element software cluster is a self-healing, scale-out architecture that is capable of nondisruptive upgrades and operations.

## Interoperability

Previous generations of hyperconverged infrastructure typically required fixed resource ratios, limiting deployments to four-node and eight-node configurations. NetApp HCI is a disaggregated hyper-converged infrastructure that can scale compute and storage resources independently. Independent scaling prevents costly and inefficient overprovisioning and simplifies capacity and performance planning.

The architectural design choices offered enables you to confidently scale on your terms, making HCI viable for core Tier-1 data center applications and platforms. It is architected in building blocks at either the chassis or the node level. Each chassis can hold four nodes in a mixed configuration of storage or compute nodes. NetApp HCI is available in mix-and-match, small, medium, and large storage and compute configurations.

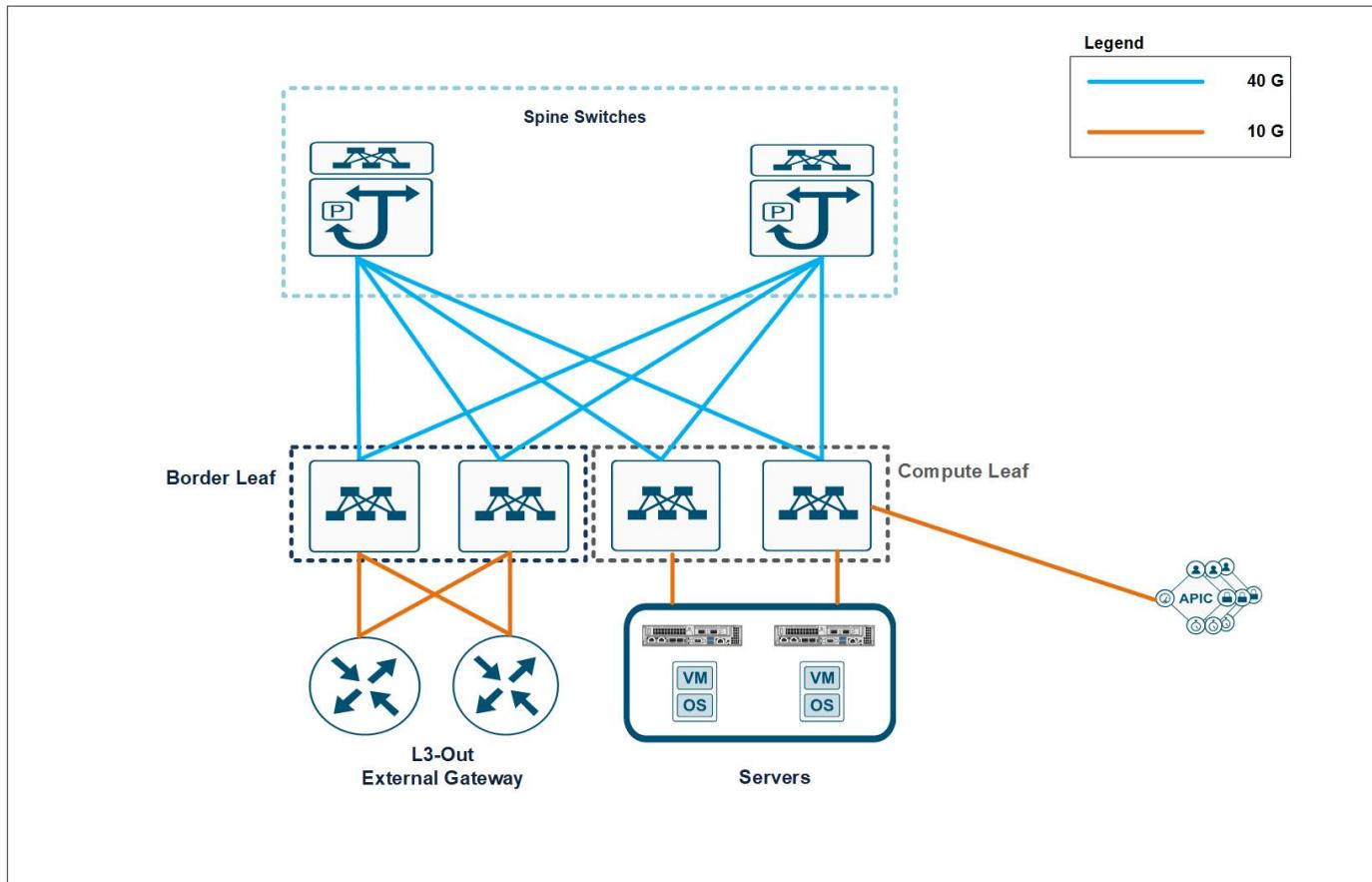
NetApp HCI provides proven multiprotocol and hybrid- cloud support with enterprise grade features. It also offers easy interoperability with multiple different host virtualization technologies and storage solutions. Deploying ONTAP Select and StorageGRID as appliances expands NetApp HCI storage capabilities to include file, block, and object storage services. NetApp HCI provides an agile infrastructure platform for virtual data centers of different flavors. VMware vSphere, Red Hat Virtualization, KVM, Citrix Hypervisor, and so on are supported platforms that can use the NetApp HCI infrastructure to provide a scalable, enterprise-grade on-premises virtual environment.

For more details, see the [NetApp HCI documentation](#).

## Cisco ACI

Cisco ACI is an industry leading software-defined networking solution that facilitates application agility and data center automation. Cisco ACI has a holistic architecture with a centralized policy-driven management. It implements a programmable data center Virtual Extensible LAN (VXLAN) fabric that delivers distributed networking and security for any workload, regardless of its nature (virtual, physical, container, and so on).

Cisco pioneered the introduction of intent-based networking with Cisco ACI in the data center. It combines the high-performance hardware and robust software integrated with two important SDN features—overlays and centralized control. The ACI fabric consists of Cisco Nexus 9000 series switches running in ACI mode and a cluster of at least three centrally managed Application Policy Infrastructure Controllers (APIC) servers. The following figure provides an overview of Cisco ACI.

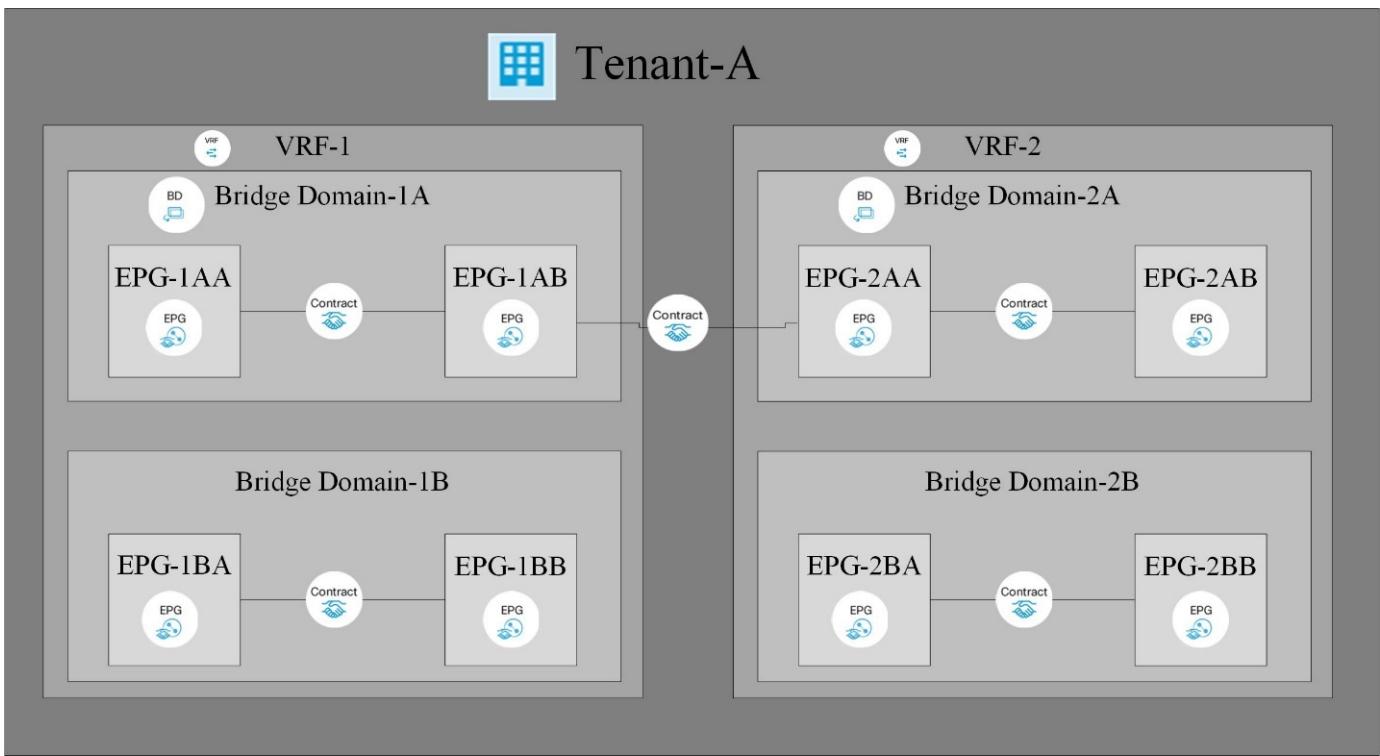


### Policy-Driven Networking

Cisco ACI, with its policy driven model, makes network hardware stateless. The Application Policy Infrastructure Controller (APIC) acts as the central controller managing and configuring all the switches in the ACI fabric. The Cisco ACI fabric consists of Cisco Nexus 9000 series switches which are centrally configured and managed by the cluster of APICs using the declarative policy model.

Cisco ACI uses logical constructs to form a layered policy architecture to define and manage the different functions of the entire fabric, including infrastructure, authentication, security, services, applications, and diagnostics.

The following figure depicts the categorization and relation between different logical constructs in Cisco ACI.



Tenants are logical containers with administrative boundaries that exercise domain-based access control. It is a logical policy isolation and does not equate to a real network construct.

Within the tenant, a context is a unique layer-3 forwarding policy domain. A context can be directly mapped to the Virtual Routing and Forwarding (VRF) concept of traditional networks. In fact, a context is also called VRF. Because each context is a separate layer- 3 domain, two different contexts can have overlapping IP spaces.

Within a context, a bridge domain (BD) represents a unique layer-2 forwarding construct. The bridge domain defines the unique layer-2 MAC address space and can be equated to a layer-2 flood domain or to a layer-3 gateway. A bridge domain can have zero subnets, but it must have at least one subnet if it is to perform routing for the hosts residing in the BD.

In ACI, an endpoint is anything that communicates on the network, be it a compute host, a storage device, a network entity that is not part of the ACI fabric, a VM, and so on. A group of endpoints that have the same policy requirements are categorized into an Endpoint Group (EPG). An EPG is used to configure and manage multiple endpoints together. An EPG is a member of a bridge domain. One EPG cannot be a member of multiple bridge domains, but multiple EPGs can be members of a single bridge domain.

All the endpoints that belong to the same EPG can communicate with each other. However, endpoints in different EPGs cannot communicate by default, but they can communicate if a contract exists between the two EPGs allowing that communication. Contracts can be equated to ACLs in traditional networking. However, it differs from an ACL in the way that it doesn't involve specifying specific IP addresses as source and destination and that contracts are applied to an EPG as a whole.

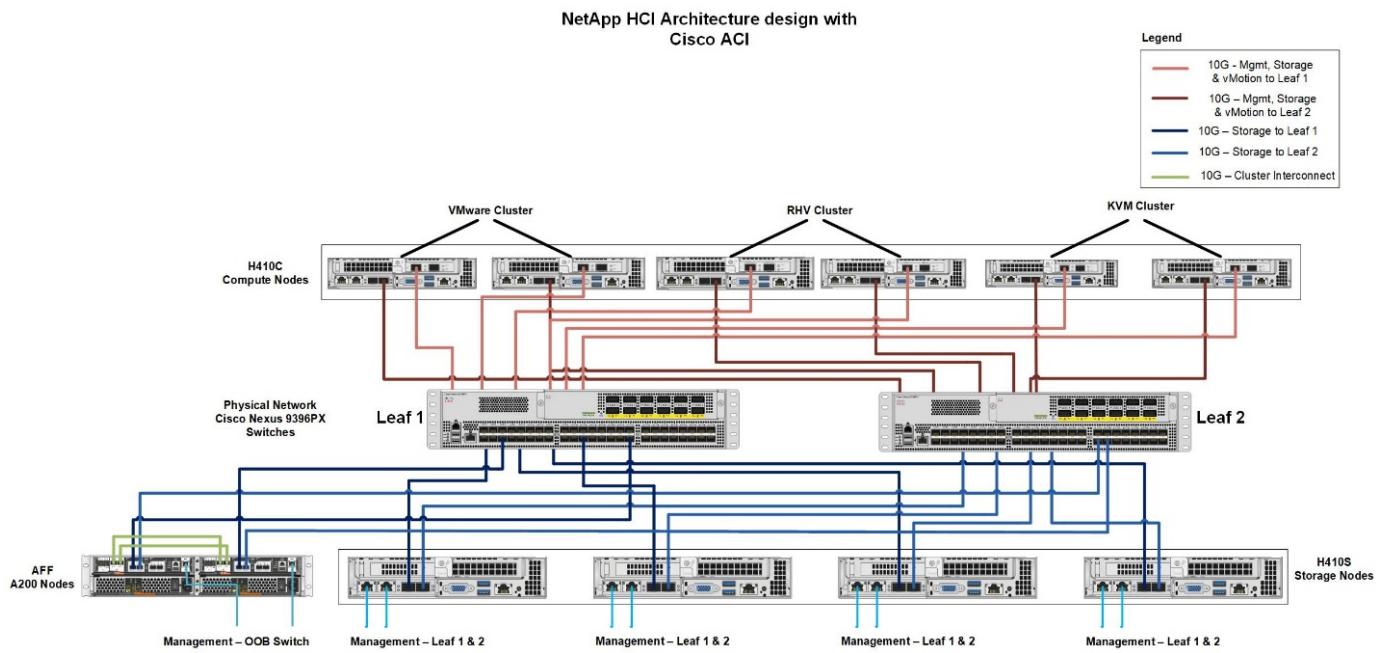
See the [Cisco ACI documentation](#) for more information.

## Networking Advantages

Cisco ACI provides many advantages over traditional networking. Programmability and automation are critical features of a scalable data center virtualization infrastructure and the policy driven mechanism of Cisco ACI opens a lot of opportunities for providing optimal physical and virtual networking.

- **Virtual Machine Manager (VMM) Integration.** With the Cisco ACI open REST API features, integration with virtualized environments is easy. Cisco ACI supports VMM integration with multiple hypervisors and provides automated access and control over the hypervisor virtual switches to the networking constructs in ACI. VMM integration in ACI seamlessly extends the ACI policy framework to virtual workloads. In other words, VMM integration allows Cisco ACI to control the virtual switches running on virtualization hosts and to extend the ACI fabric access policies to virtual workloads. The integration also automates the hypervisor's virtual switch deployment and configuration tasks. Cisco ACI VMM integration provides the following benefits:
  - Single point of policy management for physical and virtual environments through APIC
  - Faster application deployment, with transparent instantiation of applications in virtual environments
  - Full integrated visibility into the health of the application through holistic aggregation of information across physical and virtual environments
  - Simplified networking configuration for virtual workloads because the port-group or VM NIC profiles required to attach to the VMs are created automatically. For more information on Cisco ACI VMM integration, see the [Cisco documentation](#). In addition, see the [Cisco ACI virtualization compatibility matrix](#) for version compatibility details.
- **Micro-segmentation.** Micro-segmentation in Cisco ACI allows you to classify the endpoints in existing application EPGs into microsegment (uSeg) EPGs using network-based or VM-based attributes. This helps for filtering the endpoints more granularly and apply specific dynamic policies on those endpoints. Micro-segmentation can be applied to any endpoints within the tenant. Cisco supports micro-segmentation on a variety of virtual switches - Cisco ACI Virtual Edge, VMware VDS and Microsoft vSwitch. uSeg EPGs can be configured with multiple attributes but an endpoint can be assigned to only one EPG. For more details, see the [Cisco ACI Virtualization guide](#) for the specific version.
- **Intra-EPG Isolation.** By default, all endpoints belonging to the same EPG can communicate with each other. Intra-EPG Isolation in Cisco ACI is a feature to prevent endpoints in the same EPG communicate with each other. It achieves isolation by using different VLANs for traffic from ACI leaf to hypervisor hosts and from hypervisor hosts to ACI leaf. Intra-EPG isolation can be enforced on both application EPGs and microsegment EPGs. See the specific version of the [Cisco ACI virtualization guide](#) for more information.

## Architectural Diagram



This diagram represents the physical architecture of NetApp HCI with Cisco ACI that was designed for this solution. Two leaf switches connected via spines and managed by a cluster of three APICs forms the ACI fabric. The leaf switches are connected to upstream routers for external connectivity. Three pairs of NetApp HCI compute nodes (each pair dedicated for a hypervisor) are configured with a two-cable option. Four storage nodes were configured with four-cable option to form the Element cluster. A pair of AFF A200 nodes are used to provide the ONTAP capabilities to the system.

## Hardware and Software Requirements

### Compute

The following tables list the hardware and software compute resources utilized in the solution. The components that are used in any implementation of the solution might vary based on customer requirements.

Hardware	Model	Quantity
NetApp HCI compute nodes	NetApp H410C	6

Software	Purpose	Version
VMware ESXi	Virtualization	6.7
VMware vCenter Server Appliance	Virtualization management	6.7
Red Hat Enterprise Linux	Operating system	7.7
KVM	Virtualization	1.5.3-167
Red Hat Virtualization	Virtualization	4.3.9

## Storage

The following tables list the hardware and software storage resources used in this solution. The components that are used in any particular implementation of the solution might vary based on customer requirements.

Hardware	Model	Quantity
NetApp HCI storage nodes	NetApp H410S	4
AFF	A200	2

Software	Purpose	Version
NetApp HCI	Infrastructure	1.8
NetApp Element	Storage	12.0
ONTAP	Storage	9.7P6
ONTAP Select	Storage	9.7
Storage Grid	Storage	11.3

## Networking

The following tables list the hardware and software network resources used in this solution. The components that are used in any particular implementation of the solution might vary based on customer requirements.

Hardware	Model	Quantity
Cisco UCS server	UCS C-220 M3	3
Cisco Nexus	N9K-C9336-PQ	2
Cisco Nexus	N9K-C9396-PX	2

Software	Purpose	Version
Cisco APIC	Network Management	3.2(9h)
Cisco Nexus ACI-mode Switch	Network	13.2(9h)
Cisco AVE	Network	1.2.9
Open vSwitch (OVS)	Network	2.9.2
VMware Virtual Distributed Switch	Network	6.6

[Next: Design Considerations](#)

# Design Considerations

## Network Design

The minimum configuration of a Cisco ACI fabric consists of two leaf switches and two spine switches with a cluster at least three APICs managing and controlling the whole fabric. All the workloads connect to leaf switches. Spine switches are the backbone of the network and are responsible for interconnecting all leaf switches. No two leaf switches can be interconnected. Each leaf switch is connected to each of the spine switches in a full-mesh topology.

With this two-tier spine-and-leaf architecture, no matter which leaf switch the server is connected to, it's traffic always crosses the same number of devices to get to another server attached to the fabric (unless the other server is located on the same leaf). This approach keeps latency at a predictable level.

## Compute Design

The minimum number of compute nodes required for a highly available infrastructure using NetApp HCI is two. NetApp HCI provides two options for cabling: two-cable and six-cable. NetApp HCI H410C compute nodes are available with two 1GbE ports (ports A and B) and four 10/25GbE ports (ports C, D, E, and F) on board. For a two-cable option, ports D and E are used for connectivity to uplink switches, and, for a six-cable option, all ports from A to F are used. Each node also has an additional out-of-band management port that supports Intelligent Platform Management Interface (IPMI) functionality. This solution utilizes the two-cable option for compute nodes.

For VMware deployments, NetApp HCI comes with an automated deployment tool called the NetApp Deployment Engine (NDE). For non-VMware deployments, manual installation of hypervisors or operating systems is required on the compute nodes.

## Storage Design

NetApp HCI uses four-cable option for storage nodes. NetApp HCI H410S storage nodes are available with two 1GbE ports (ports A and B) and two 10/25GbE ports (ports C and D) on board. The two 1GbE ports are bundled as Bond1G (active/passive mode) used for management traffic and the two 10/25GbE ports are bundled as Bond10G (LACP active mode) used for storage data traffic.

For non-VMware deployments, the minimum configuration of NetApp HCI storage cluster is four nodes. For NetApp HCI versions earlier than 1.8 with VMware deployments, the minimum configuration is four storage nodes. However, for HCI version 1.8 with VMware deployments, the minimum configuration for NetApp HCI storage cluster is two nodes. For more information on NetApp HCI two-node storage cluster, see the documentation [here](#).

Next: [VMware vSphere: NetApp HCI with Cisco ACI](#)

## Deploying NetApp HCI with Cisco ACI

## VMware vSphere: NetApp HCI with Cisco ACI

VMware vSphere is an industry-leading virtualization platform that provides a way to build a resilient and reliable virtual infrastructure. vSphere contains virtualization, management, and interface layers. The two core components of VMware vSphere are ESXi server and the vCenter Server. VMware ESXi is hypervisor software installed on a physical machine that facilitates hosting of VMs and virtual appliances. vCenter Server is the service through which you manage multiple ESXi hosts connected in a network and pool host resources. For more information on VMware vSphere, see the documentation [here](#).

### Workflow

The following workflow was used to set up the virtual environment. Each of these steps might involve several individual tasks.

1. Install and configure Nexus 9000 switches in ACI mode and APIC software on the UCS C-series server. See the [Install and Upgrade documentation](#) for detailed steps.
2. Configure and setup ACI fabric by referring to the [documentation](#).
3. Configure the tenants, application profiles, bridge domains, and EPGs required for NetApp HCI nodes. NetApp recommends using one BD to one EPG framework, except for iSCSI. See the documentation [here](#) for more details. The minimum set of EPGs required are in-band management, iSCSI, iSCSI-A, iSCSI-B, VM motion, VM-data network, and native.



iSCSI multipathing requires two iSCSI EPGs: iSCSI-A and iSCSI-B, each with one active uplink.



NetApp mNode requires an iSCSI EPG with both uplinks active.

4. Create the VLAN pool, physical domain, and AEP based on the requirements. Create the switch and interface profiles for individual ports. Then attach the physical domain and configure the static paths to the EPGs. See the [configuration guide](#) for more details.

## VLAN Pool - HCI-Internal-Phys-Dom-VLAN (Static Allocation)



Policy   Operational   Faults   History



↑ ↓ ⌂ ⌂

### Properties

Name:	HCI-Internal-Phys-Dom-VLAN										
Description:	optional										
Alias:											
Allocation Mode:	Static Allocation										
Encap Blocks:	<table border="1"> <thead> <tr> <th>VLAN Range</th> <th>Allocation Mode</th> <th>Role</th> </tr> </thead> <tbody> <tr> <td>[2]</td> <td>Inherit allocMode from parent</td> <td>External or On the wire encapsulations</td> </tr> <tr> <td>[3201-3250]</td> <td>Inherit allocMode from parent</td> <td>External or On the wire encapsulations</td> </tr> </tbody> </table>		VLAN Range	Allocation Mode	Role	[2]	Inherit allocMode from parent	External or On the wire encapsulations	[3201-3250]	Inherit allocMode from parent	External or On the wire encapsulations
VLAN Range	Allocation Mode	Role									
[2]	Inherit allocMode from parent	External or On the wire encapsulations									
[3201-3250]	Inherit allocMode from parent	External or On the wire encapsulations									
Domains:	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> </tr> </thead> <tbody> <tr> <td>HCI-Internal-Phys-Dom</td> <td>Physical Domain</td> </tr> </tbody> </table>		Name	Type	HCI-Internal-Phys-Dom	Physical Domain					
Name	Type										
HCI-Internal-Phys-Dom	Physical Domain										

## Leaf Access Port Policy Group - HCI-Compute-ESX



### Properties

Name:	HCI-Compute-ESX	
Description:	optional	
Alias:		
Link Level Policy:	10G-Auto	<input type="button" value=""/>
CDP Policy:	CDP-Disabled	<input type="button" value=""/>
MCP Policy:	select a value	<input type="button" value=""/>
CoPP Policy:	select a value	<input type="button" value=""/>
LLDP Policy:	LLDP-Enabled	<input type="button" value=""/>



Use an access port policy group for interfaces connecting to NetApp HCI compute nodes, and use vPC policy group for interfaces to NetApp HCI storage nodes.

5. Create and assign contracts for tightly-controlled access between workloads. For more information

on configuring the contracts, see the guide [here](#).

6. Install and configure NetApp HCI using NDE. NDE configures all the required parameters, including VDS port groups for networking, and also installs the mNode VM. See the [deployment guide](#) for more information.
7. Though VMM integration of Cisco ACI with VMware VDS is optional, using the VMM integration feature is a best practice. When not using VMM integration, an NDE-installed VDS can be used for networking with physical domain attachment on Cisco ACI.
8. If you are using VMM integration, NDE-installed VDS cannot be fully managed by ACI and can be added as read-only VMM domain. To avoid that scenario and make efficient use of Cisco ACI's VMM networking feature, create a new VMware VMM domain in ACI with an explicit dynamic VLAN pool. The VMM domain created can integrate with any supported virtual switch.
  - a. **Integrate with VDS.** If you wish to integrate ACI with VDS, select the virtual switch type to be VMware Distributed Switch. Consider the configuration best practices noted in the following table. See the [configuration guide](#) for more details.

#### Properties

Name: hci-aci-vds-02

Virtual Switch: Distributed Switch

Associated Attachable Entity ▲ Name

Profiles:

HCI-Internal

---

Encapsulation: vlan

Delimiter:

Enable Tag Collection:

Enable VM Folder Data Retrieval:

Access Mode: Read Only Mode Read Write Mode

Endpoint Retention Time (seconds): 0 ^ v

VLAN Pool: hci-aci-vmware(dynamic) ▼ ✚

- b. **Integrate with Cisco AVE.** If you are integrating Cisco AVE with Cisco ACI, select the virtual switch type to be Cisco AVE. Cisco AVE requires a unique VLAN pool of type Internal for communicating between internal and external port groups. Follow the configuration best practices noted in this table. See the [installation guide](#) to install and configure Cisco AVE.

## Properties

Name: hci-vmware-ave

Virtual Switch: Cisco AVE

AVE Time-out Time (seconds):  ^ ▼

Host Availability Assurance:

Associated Attachable Entity ▲ Name  
Profiles: **HCI-Internal**

---

Switching Preference:  No Local Switching  Local Switching

Enhanced Lag Policy:  ▼

Encapsulation: vxlan

Default Encap Mode:  Unspecified  VLAN  VXLAN

---

Enable Tag Collection:

Enable VM Folder Data Retrieval:

Endpoint Retention Time (seconds):  ^ ▼

VLAN Pool: hci-aci-vmware(dynamic) ▼ ✚

AVE Fabric-Wide Multicast   
Address: Must Use a Multicast Address different  
from the Multicast Address Ranges.

Pool of Multicast Addresses (one  
per-EPG):  ▼ ✚

9. Attach the VMM domain to the EPGs using Pre-Provision Resolution Immediacy. Then migrate all the VMNICs, VMkernel ports, and VNICs from the NDE-created VDS to ACI-created VDS or AVE and so on. Configure the uplink failover and teaming policy for iSCSI-A and iSCSI-B to have one active uplink each. VMs can now attach their VMNICs to ACI-created port groups to access network resources. The port groups on VDS that are managed by Cisco ACI are in the format of <tenant-name>|<application-profile-name>|<epg-name>.



Pre-Provision Resolution Immediacy is required to ensure the port policies are downloaded to the leaf switch even before the VMM controller is attached to the virtual switch.

The screenshot shows the NetApp HCI Datacenter interface. On the left, a navigation tree displays a hierarchy: 'NetApp-HCI-Datacenter-01' under 'NetApp-HCI-Datacenter-01', which contains 'hci-aci-vds-02'. This switch has several port profiles listed under it, including 'HCI-Infra|AFF-A200|AFF-NFS', 'HCI-Infra|HCl|HCl-iSCSI', and 'HCI-Infra|HCl|HCl-Select-Internal'. The right panel is titled 'hci-aci-vds-02' and shows the 'Summary' tab selected. It displays basic information: Manufacturer: VMware, Inc., Version: 6.6.0, and a link to 'Upgrades available'. Below this is a section titled 'Switch Details' and a 'Notes' section containing the text 'APIC Virtual Switch' with a link to 'Edit Notes...'.

## VMkernel adapters

VMkernel adapters						
Device	Network Label	Switch	IP Address	TCP/IP Stack	vMotion	Provisioning
vmk0	HCI-Infra HCl H...	hci-vmware-ave	172.22.9.60	Default	Disabled	Disabled
vmk1	HCI-Infra HCl S...	hci-vmware-ave	172.22.10.60	Default	Disabled	Disabled
vmk2	HCI-Infra HCl S...	hci-vmware-ave	172.22.10.58	Default	Disabled	Disabled
vmk3	HCI-Infra HCl H...	hci-vmware-ave	172.22.13.60	Default	Enabled	Disabled
vmk4	HCI-Infra AFF-A...	hci-vmware-ave	172.22.15.60	Default	Disabled	Disabled

10. If you intend to use micro-segmentation, then create micro-segment (uSeg) EPGs attaching to the right BD. Create attributes in VMware vSphere and attach them to the required VMs. Ensure the VMM domain has Enable Tag Collection enabled. Configure the uSeg EPGs with the corresponding attribute and attach the VMM domain to it. This provides more granular control of communication on the endpoint VMs.

The screenshot shows the Cisco Application Virtualization Engine (AVE) interface. On the left, a navigation tree shows 'useg-ubuntu-prod' under 'Domains (VMs and Bare-Me...)', which includes 'Static Leafs', 'Contracts', 'Static Endpoint', and 'uSeg Attributes'. The right panel is titled 'uSeg Attributes' and shows a search interface with a dropdown menu 'Match Any' and a search bar containing 'VM - Tag' and 'ubuntu'. There are also buttons for 'Contains' and 'ubuntu-prod'.

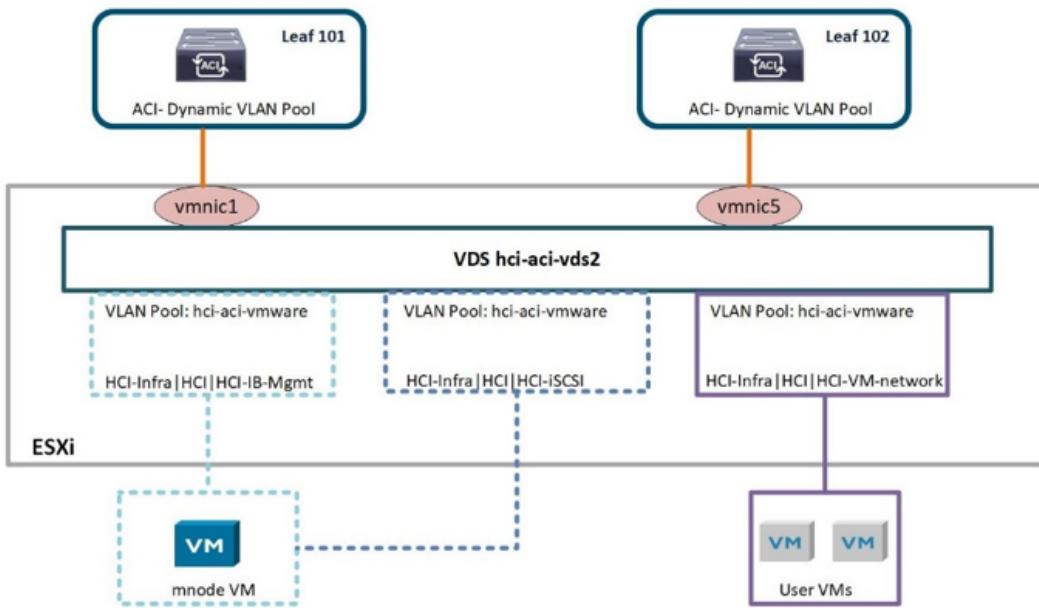
The networking functionality for VMware vSphere on NetApp HCI in this solution is provided either using VMware VDS or Cisco AVE.

## VMware VDS

VMware vSphere Distributed Switch (VDS) is a virtual switch that connects to multiple ESXi hosts in the cluster or set of clusters allowing virtual machines to maintain consistent network configuration as they migrate across multiple hosts. VDS also provides for centralized management of network configurations in a vSphere environment. For more details, see the [VDS documentation](#).

### Legends

	EPG:HCI-IB-Mgmt (VMKernel)
	EPG:HCI-iSCSI (VMKernel)
	EPG:HCI-VM-network



The following table outlines the necessary parameters and best practices for configuring and integrating Cisco ACI with VMware VDS.

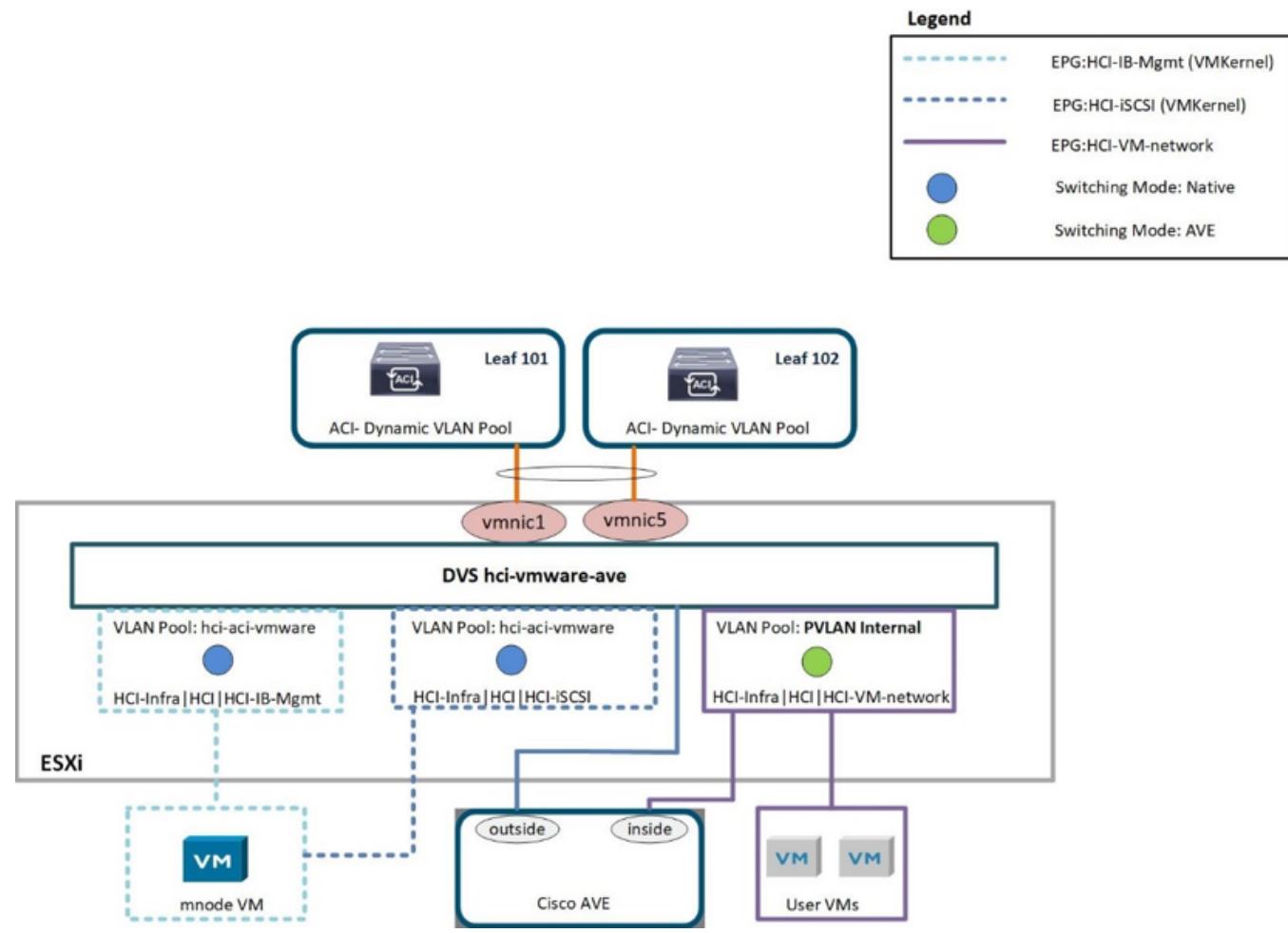
Resource	Configuration Considerations	Best Practices
Endpoint groups	<ul style="list-style-type: none"> <li>• Separate EPG for native VLANs</li> <li>• Static binding of interfaces to HCI storage and compute nodes in native VLAN EPG uses 802.1P mode. This is required for node discovery to run NDE.</li> <li>• Separate EPGs for iSCSI, iSCSI-A, and iSCSI-B with a common BD</li> <li>• iSCSI-A and iSCSI-B are for iSCSI multipathing and are used for VMkernel ports on ESXi hosts</li> <li>• Physical domain to be attached to iSCSI EPG before running NDE</li> <li>• VMM domain to be attached to iSCSI, iSCSI-A, and iSCSI-B EPGs</li> </ul>	<ul style="list-style-type: none"> <li>• Contracts between EPGs to be well defined. Allow only required ports for communication.</li> <li>• Use unique native VLAN for NDE node discovery</li> <li>• For EPGs corresponding to port-groups being attached to VMkernel ports, VMM domain to be attached with Pre-Provision for Resolution Immediacy</li> </ul>
Interface policy	<ul style="list-style-type: none"> <li>• A common leaf access port policy group for all ESXi hosts</li> <li>• One vPC policy group per NetApp HCI storage node</li> <li>• LLDP enabled, CDP disabled</li> </ul>	<ul style="list-style-type: none"> <li>• Separate VLAN pool for VMM domain with dynamic allocation turned on</li> <li>• Recommended to use vPC with LACP Active port-channel policy for interfaces towards NetApp HCI Storage Nodes</li> <li>• Recommended to use individual interfaces for Compute Nodes, No LACP.</li> </ul>

Resource	Configuration Considerations	Best Practices
VMM Integration	<ul style="list-style-type: none"> <li>Local switching preference</li> <li>Access mode is Read Write.</li> </ul>	<ul style="list-style-type: none"> <li>MAC-Pinning-Physical-NIC-Load for vSwitch policy</li> <li>LLDP for discovery policy</li> <li>Enable Tag collection if micro-segmentation is used</li> </ul>
VDS	<ul style="list-style-type: none"> <li>Both uplinks active for iSCSI port-group</li> <li>One uplink each for iSCSI-A and iSCSI-B</li> </ul>	<ul style="list-style-type: none"> <li>Load balancing method for all port-groups to be ‘Route based on physical NIC load’</li> <li>iSCSI VMkernel port migration to be done one at a time from NDE deployed VDS to ACI integrated VDS</li> </ul>

For traffic load-balancing, port channels with vPCs can be used on Cisco ACI along with LAGs on VDS with LACP in active mode. However, using LACP can affect storage performance when compared to iSCSI multipathing.

### Cisco AVE

Cisco ACI Virtual Edge (AVE) is a virtual switch offering by Cisco that extends the Cisco ACI policy model to virtual infrastructure. It is a hypervisor-independent distributed network service that sits on top of the native virtual switch of the hypervisor. It leverages the underlying virtual switch using a VM-based solution to provide network visibility into the virtual environments. For more details on Cisco AVE, see the [documentation](#). The following figure depicts the internal networking of Cisco AVE on an ESXi host (as tested).



The following table lists the necessary parameters and best practices for configuring and integrating Cisco ACI with Cisco AVE on VMware ESXi. Cisco AVE is currently only supported with VMware vSphere.

Resource	Configuration Considerations	Best Practices
Endpoint Groups	<p>Separate EPG for native VLANs</p> <p>Static binding of interfaces towards HCI storage and compute nodes in native VLAN</p> <p>EPG uses 802.1P mode. This is required for node discovery to run NDE.</p> <p>Separate EPGs for iSCSI, iSCSI-A and iSCSI-B with a common BD</p> <p>iSCSI-A and iSCSI-B are for iSCSI multipathing and are used for VMkernel ports on ESXi hosts</p> <p>Physical domain to be attached to iSCSI EPG before running NDE</p> <p>VMM domain is attached to iSCSI, iSCSI-A, and iSCSI-B EPGs</p>	<p>Separate VLAN pool for VMM domain with dynamic allocation turned on</p> <p>Contracts between EPGs to be well defined. Allow only required ports for communication.</p> <p>Use unique native VLAN for NDE node discovery</p> <p>Use native switching mode in VMM domain for EPGs that correspond to port groups being attached to host's VMkernel adapters</p> <p>Use AVE switching mode in VMM domain for EPGs corresponding to port groups carrying user VM traffic</p> <p>For EPGs corresponding to port-groups being attached to VMkernel ports, VMM domain is attached with Pre-Provision for Resolution Immediacy</p>
Interface Policy	<ul style="list-style-type: none"> <li>One vPC policy group per ESXi host</li> <li>One vPC policy group per NetApp HCI storage node</li> <li>LLDP enabled, CDP disabled</li> </ul>	<ul style="list-style-type: none"> <li>NetApp recommends using vPCs to ESXi hosts</li> <li>Use static mode on port-channel policy for vPCs to ESXi</li> <li>Use Layer-4 SRC port load balancing hashing method for port-channel policy</li> <li>NetApp recommends using vPC with LACP active port-channel policy for interfaces to NetApp HCI storage nodes</li> </ul>

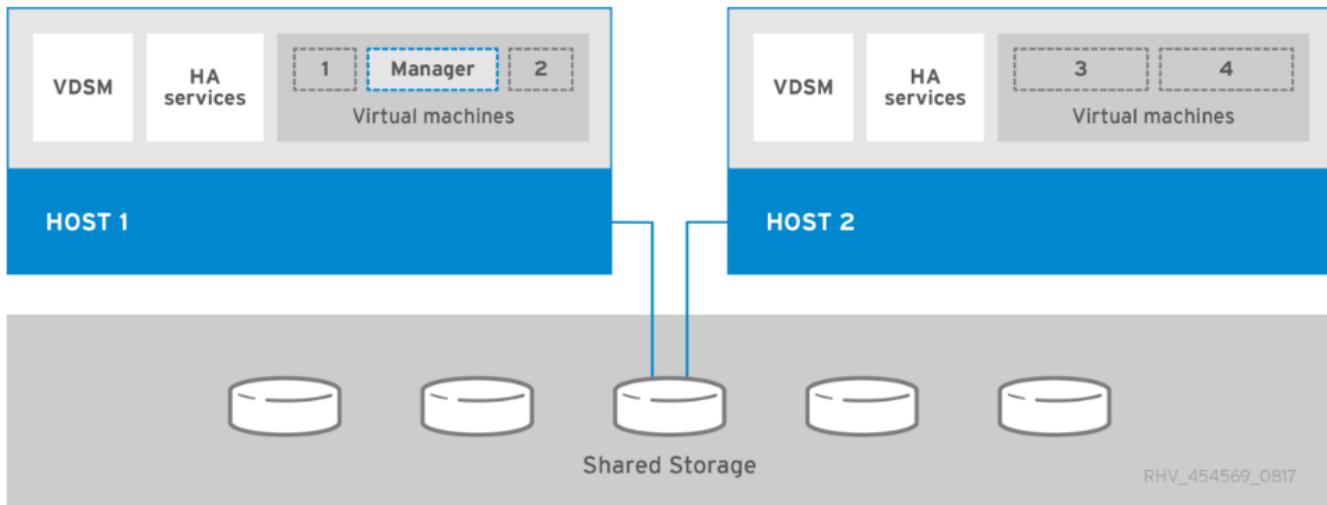
Resource	Configuration Considerations	Best Practices
VMM Integration	<ul style="list-style-type: none"> <li>Create a new VLAN range [or Encap Block] with role Internal and Dynamic allocation' attached to the VLAN pool intended for VMM domain</li> <li>Create a pool of multicast addresses (one address per EPG)</li> <li>Reserve another multicast address different from the pool of multicast addresses intended for AVE fabric-wide multicast address</li> <li>Local switching preference</li> <li>Access mode to be Read Write mode</li> </ul>	<ul style="list-style-type: none"> <li>Static mode on for vSwitch policy</li> <li>Ensure that vSwitch port-channel policy and interface policy group's port-channel policy are using the same mode</li> <li>LLDP for discovery policy</li> <li>Enable Tag collection if using micro-segmentation</li> <li>Recommended option for Default Encap mode is VXLAN</li> </ul>
VDS	<ul style="list-style-type: none"> <li>- Both uplinks active for iSCSI port-group</li> <li>- One uplink each for iSCSI-A and iSCSI-B</li> </ul>	<ul style="list-style-type: none"> <li>iSCSI VMkernel port migration is done one at a time from NDE deployed VDS to ACI integrated VDS</li> <li>Load balancing method for all port-groups to be Route based on IP hash</li> </ul>



For traffic load balancing, port channel with vPCs can be used on Cisco ACI along with LAGs on ESXi hosts with LACP in active mode. However, using LACP can affect storage performance when compared to iSCSI multipathing.

#### Red Hat Virtualization: NetApp HCI with Cisco ACI

Red Hat Virtualization (RHV) is an enterprise virtual data center platform that runs on Red Hat Enterprise Linux using the KVM hypervisor. The key components of RHV include Red Hat Virtualization Hosts (RHV- H) and the Red Hat Virtualization Manager (RHV- M). RHV-M provides centralized, enterprise-grade management for the physical and logical resources within the virtualized RHV environment. RHV-H is a minimal, light-weight operating system based on Red Hat Enterprise Linux that is optimized for the ease of setting up physical servers as RHV hypervisors. For more information on RHV, see the documentation [here](#). The following figure provides an overview of RHV.



Starting with Cisco APIC release 3.1, Cisco ACI supports VMM integration with Red Hat Virtualization environments. The RHV VMM domain in Cisco APIC is connected to RHV-M and directly associated with a data center object. All the RHV-H clusters under this data center are considered part of the VMM domain. Cisco ACI automatically creates logical networks in RHV- M when the EPGs are attached to the RHV VMM domain in ACI. RHV hosts that are part of a Red Hat VMM domain can use Linux bridge or Open vSwitch as its virtual switch. This integration simplifies and automates networking configuration on RHV-M, saving a lot of manual work for system and network administrators.

## Workflow

The following workflow is used to set up the virtual environment. Each of these steps might involve several individual tasks.

1. Install and configure Nexus 9000 switches in ACI mode and APIC software on the UCS C-series server. Refer to the [Install and Upgrade documentation](#) for detailed steps.
2. Configure and setup the ACI fabric by referring to the [documentation](#).
3. Configure tenants, application profiles, bridge domains, and EPGs required for NetApp HCI nodes. NetApp recommends using one BD to one EPG framework, except for iSCSI. See the documentation [here](#) for more details. The minimum set of EPGs required are in-band management, iSCSI, VM motion, VM-data network, and native.
4. Create the VLAN pool, physical domain, and AEP based on the requirements. Create the switch and interface profiles and policies for vPCs and individual ports. Then attach the physical domain and configure the static paths to the EPGs. see the [configuration guide](#) for more details. This table lists best practices for integrating ACI with Linux bridge on RHV.

# PC/VPC Interface Policy Group - HCI-RHVH01

Properties

Name: HCI-RHVH01

Description: optional

Link Aggregation Type: Port Channel **VPC**

Link Level Policy: 10G-Auto 

CDP Policy: CDP-Disabled 

MCP Policy: select a value 

CoPP Policy: select a value 

LLDP Policy: LLDP-Enabled 

STP Interface Policy: select a value 

Egress Data Plane Policing Policy: select a value 

Ingress Data Plane Policing Policy: select a value 

Priority Flow Control Policy: select a value 

Fibre Channel Interface Policy: select a value 

Slow Drain Policy: select a value 

Port Channel Policy: LACP-Active 



Use a vPC policy group for interfaces connecting to NetApp HCI storage and compute nodes.

5. Create and assign contracts for tightly controlled access between workloads. For more information on configuring the contracts, see the guide [here](#).
6. Install and configure the NetApp HCI Element cluster. Do not use NDE for this install; rather, install a standalone Element cluster on the HCI storage nodes. Then configure the required volumes for installation of RHV. Install RHV on NetApp HCI. Refer to [RHV on NetApp HCI NVA](#) for more details.
7. RHV installation creates a default management network called ovirtmgmt. Though VMM integration of Cisco ACI with RHV is optional, leveraging VMM integration is preferred. Do not create other logical networks manually. To use Cisco ACI VMM integration, create a Red Hat VMM

domain and attach the VMM domain to all the required EPGs, using Pre- Provision Resolution Immediacy. This process automatically creates corresponding logical networks and vNIC profiles. The vNIC profiles can be directly used to attach to hosts and VMs for their communication. The networks that are managed by Cisco ACI are in the format <tenant-name>|<application-profile-name>|<epg-name> tagged with a label of format aci\_<rhv-vmm-domain-name>. See [Cisco's whitepaper](#) for creating and configuring a VMM domain for RHV. Also, see this table for best practices when integrating RHV on NetApp HCI with Cisco ACI.



Except for ovirtmgmt, all other logical networks can be managed by Cisco ACI.

Network > Networks

Network:		x	☆	Q	New	Import	Edit	Remove	
Name	Comment	Data Center	Description	Role	VLAN Tag	QoS	Label	Provider	MTU
HCI-Infra AFF-A200 AFT-NFS	Default			■	1569	-	aci_hci-aci-rhv		9000
HCI-Infra HCI HCI-IB-Mgmt	Default			■	1567	-	aci_hci-aci-rhv		Default (1500)
HCI-Infra HCI HCI-SCSI	Default			■	1568	-	aci_hci-aci-rhv		9000
HCI-Infra HCI HCI-VM-motion	Default				1634	-	aci_hci-aci-rhv		Default (1500)
HCI-Infra HCI HCI-VM-network	Default			■	1570	-	aci_hci-aci-rhv		Default (1500)
ovirtmgmt	Default		Management Network	■	3201	-	-		Default (1500)
quarantine	Default			■	666	-	aci_hci-aci-rhv		Default (1500)
uplinkNetwork	Default		uplinkNetwork	■	-	-	-		Default (1500)

## Setup Host hci-aci-rtp-rvhv01.cie.netapp.com Networks

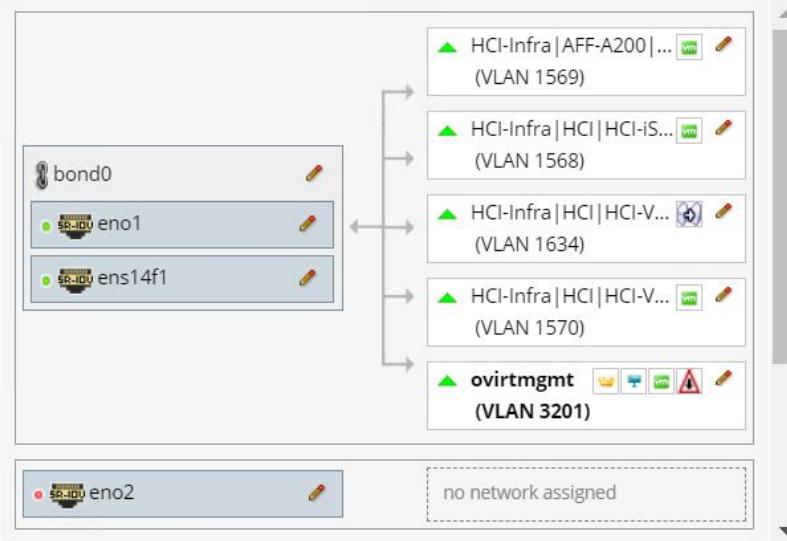
X

Drag to make changes

### Interfaces

### Assigned Logical Networks

Networks Labels



- [New Label]
- aci\_hci-aci-rhv
  - HCI-Infra|AFF-A200... (VLAN 1569)
  - HCI-Infra|HCI|HCI-IS... (VLAN 1568)
  - HCI-Infra|HCI|HCI-V... (VLAN 1634)
  - HCI-Infra|HCI|HCI-V... (VLAN 1570)
  - ovirtmgmt (VLAN 3201)
- HCI-Infra|HCI|HCI-IB-Mg... (VLAN 1567)
- HCI-Infra|HCI|HCI-i... (VLAN 1568)
- HCI-Infra|HCI|HCI-V... (VLAN 1634)
- HCI-Infra|HCI|HCI-V... (VLAN 1569)

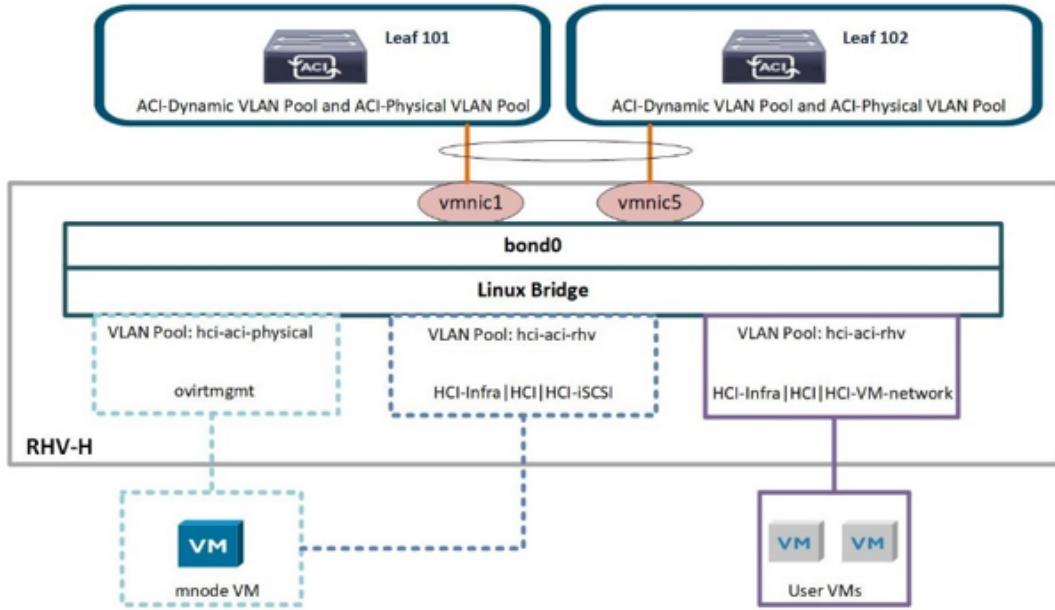
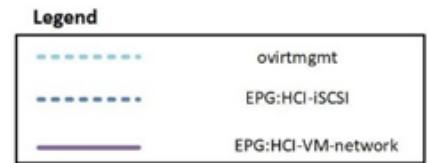
- Verify connectivity between Host and Engine ⓘ
- Save network configuration ⓘ

OK Cancel

The networking functionality for RHVH hosts in this solution is provided by Linux bridge.

## Linux Bridge

Linux Bridge is a default virtual switch on all Linux distributions that is usually used with KVM/QEMU-based hypervisors. It is designated to forward traffic between networks based on MAC addresses and thus is regarded as a layer-2 virtual switch. For more information, see the documentation [here](#). The following figure depicts the internal networking of Linux Bridge on RHV-H (as tested).



The following table outlines the necessary parameters and best practices for configuring and integrating Cisco ACI with Linux Bridge on RHV hosts.

Resource	Configuration considerations	Best Practices
Endpoint groups	<ul style="list-style-type: none"> <li>Separate EPG for native VLAN</li> <li>Static binding of interfaces towards HCI storage and compute nodes in native VLAN EPG to be on 802.1P mode</li> <li>Static binding of vPCs required on In-band management EPG and iSCSI EPG before RHV installation</li> </ul>	<ul style="list-style-type: none"> <li>Separate VLAN pool for VMM domain with dynamic allocation turned on</li> <li>Contracts between EPGs to be well defined. Allow only required ports for communication.</li> <li>Use unique native VLAN for discovery during Element cluster formation</li> <li>For EPGs corresponding to port-groups being attached to VMkernel ports, VMM domain to be attached with ‘Pre-Provision’ for Resolution Immediacy</li> </ul>
Interface policy	<ul style="list-style-type: none"> <li>One vPC policy group per RHV-H host</li> <li>One vPC policy group per NetApp HCI storage node</li> <li>LLDP enabled, CDP disabled</li> </ul>	<ul style="list-style-type: none"> <li>Recommended to use vPC towards RHV-H hosts</li> <li>Use ‘LACP Active’ for the port-channel policy</li> <li>Use only ‘Graceful Convergence’ and ‘Symmetric Hashing’ control bits for port-channel policy Use ‘Layer4 Src-port’ load balancing hashing method for port-channel policy Recommended to use vPC with LACP Active port-channel policy for interfaces towards NetApp HCI storage nodes</li> </ul>
VMM Integration	Do not migrate host management logical interfaces from ovirtmgmt to any other logical network	iSCSI host logical interface to be migrated to iSCSI logical network managed by ACI VMM integration

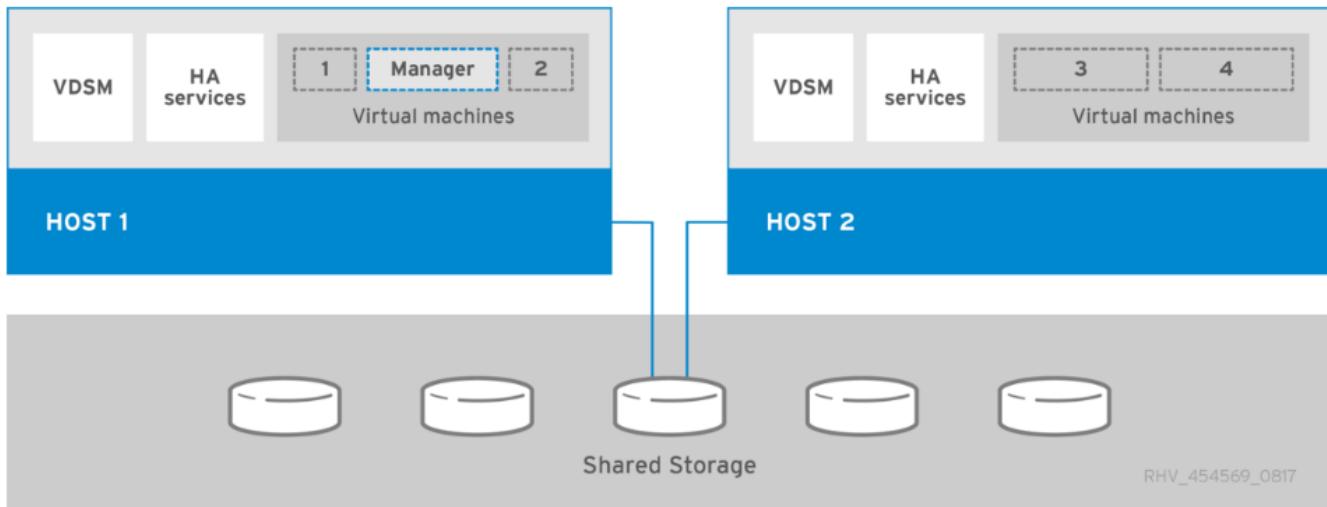


Except for the ovirtmgmt logical network, it is possible to create all other infrastructure logical networks on Cisco APIC and map them to the VMM domain. ‘ovirtmgmt’ logical network uses the static path binding on the In-band management EPG attached with the physical domain.

Next: KVM on RHEL: NetApp HCI with Cisco ACI

## Red Hat Virtualization: NetApp HCI with Cisco ACI

Red Hat Virtualization (RHV) is an enterprise virtual data center platform that runs on Red Hat Enterprise Linux using the KVM hypervisor. The key components of RHV include Red Hat Virtualization Hosts (RHV-H) and the Red Hat Virtualization Manager (RHV-M). RHV-M provides centralized, enterprise-grade management for the physical and logical resources within the virtualized RHV environment. RHV-H is a minimal, light-weight operating system based on Red Hat Enterprise Linux that is optimized for the ease of setting up physical servers as RHV hypervisors. For more information on RHV, see the documentation [here](#). The following figure provides an overview of RHV.



Starting with Cisco APIC release 3.1, Cisco ACI supports VMM integration with Red Hat Virtualization environments. The RHV VMM domain in Cisco APIC is connected to RHV-M and directly associated with a data center object. All the RHV-H clusters under this data center are considered part of the VMM domain. Cisco ACI automatically creates logical networks in RHV-M when the EPGs are attached to the RHV VMM domain in ACI. RHV hosts that are part of a Red Hat VMM domain can use Linux bridge or Open vSwitch as its virtual switch. This integration simplifies and automates networking configuration on RHV-M, saving a lot of manual work for system and network administrators.

### Workflow

The following workflow is used to set up the virtual environment. Each of these steps might involve several individual tasks.

1. Install and configure Nexus 9000 switches in ACI mode and APIC software on the UCS C-series

server. Refer to the Install and Upgrade [documentation](#) for detailed steps.

2. Configure and setup the ACI fabric by referring to the [documentation](#).
3. Configure tenants, application profiles, bridge domains, and EPGs required for NetApp HCI nodes. NetApp recommends using one BD to one EPG framework, except for iSCSI. See the documentation [here](#) for more details. The minimum set of EPGs required are in-band management, iSCSI, VM motion, VM-data network, and native.
4. Create the VLAN pool, physical domain, and AEP based on the requirements. Create the switch and interface profiles and policies for vPCs and individual ports. Then attach the physical domain and configure the static paths to the EPGs. see the [configuration guide](#) for more details. This table lists best practices for integrating ACI with Linux bridge on RHV.

## PC/VPC Interface Policy Group - HCI-RHVH01

Properties

Name: HCI-RHVH01

Description: optional

Link Aggregation Type: Port Channel **VPC**

Link Level Policy: 10G-Auto

CDP Policy: CDP-Disabled

MCP Policy: select a value

CoPP Policy: select a value

LLDP Policy: LLDP-Enabled

STP Interface Policy: select a value

Egress Data Plane Policing Policy: select a value

Ingress Data Plane Policing Policy: select a value

Priority Flow Control Policy: select a value

Fibre Channel Interface Policy: select a value

Slow Drain Policy: select a value

Port Channel Policy: LACP-Active



Use a vPC policy group for interfaces connecting to NetApp HCI storage and compute nodes.

5. Create and assign contracts for tightly controlled access between workloads. For more information on configuring the contracts, see the guide [here](#).
6. Install and configure the NetApp HCI Element cluster. Do not use NDE for this install; rather, install a standalone Element cluster on the HCI storage nodes. Then configure the required volumes for installation of RHV. Install RHV on NetApp HCI. Refer to [RHV on NetApp HCI NVA](#) for more details.
7. RHV installation creates a default management network called ovirtmgmt. Though VMM integration of Cisco ACI with RHV is optional, leveraging VMM integration is preferred. Do not create other logical networks manually. To use Cisco ACI VMM integration, create a Red Hat VMM domain and attach the VMM domain to all the required EPGs, using Pre- Provision Resolution Immediacy. This process automatically creates corresponding logical networks and vNIC profiles. The vNIC profiles can be directly used to attach to hosts and VMs for their communication. The networks that are managed by Cisco ACI are in the format <tenant-name>|<application-profile-name>|<epg-name> tagged with a label of format aci\_<rhv-vmm-domain-name>. See [Cisco's whitepaper](#) for creating and configuring a VMM domain for RHV. Also, see this table for best practices when integrating RHV on NetApp HCI with Cisco ACI.



Except for ovirtmgmt, all other logical networks can be managed by Cisco ACI.

Network > Networks

Name	Comment	Data Center	Description	Role	VLAN	Tag	QoS	Nam	Label	Provider	MTU
HCI-Infra AFF-A200 AFF-NFS		Default			1569	-		ad_hci-aci-rhv			9000
HCI-Infra HCI HCI-IB-Mgmt		Default			1567	-		aci_hci-aci-rhv			Default (1500)
HCI-Infra HCI HCI-SCSI		Default			1568	-		aci_hci-aci-rhv			9000
HCI-Infra HCI HCI-VM-motion		Default			1634	-		aci_hci-aci-rhv			Default (1500)
HCI-Infra HCI HCI-VM-network		Default			1570	-		aci_hci-aci-rhv			Default (1500)
ovirtmgmt		Default	Management Network		3201	-		-			Default (1500)
quarantine		Default			666	-		aci_hci-aci-rhv			Default (1500)
uplinkNetwork		Default	uplinkNetwork		-	-	-	-			Default (1500)

## Setup Host hci-aci-rtp-rvhv01.cie.netapp.com Networks

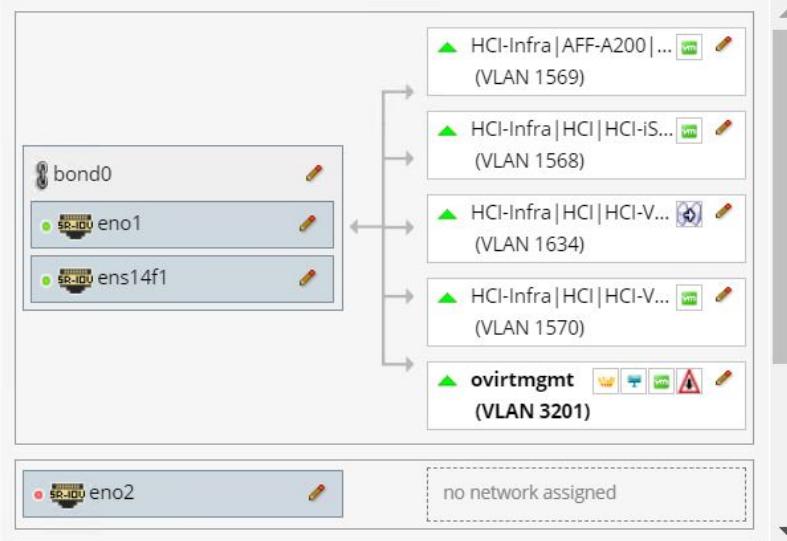
X

Drag to make changes

### Interfaces

### Assigned Logical Networks

Networks Labels



[New Label]

aci\_hci-aci-rhv

HCI-Infra|AFF-A200|... (VLAN 1569)

HCI-Infra|HCI|HCI-IB-Mg... (VLAN 1567)

HCI-Infra|HCI|HCI-i... (VLAN 1568)

HCI-Infra|HCI|HCI-V... (VLAN 1634)

HCI-Infra|HCI|HCI-V... (VLAN 1569)

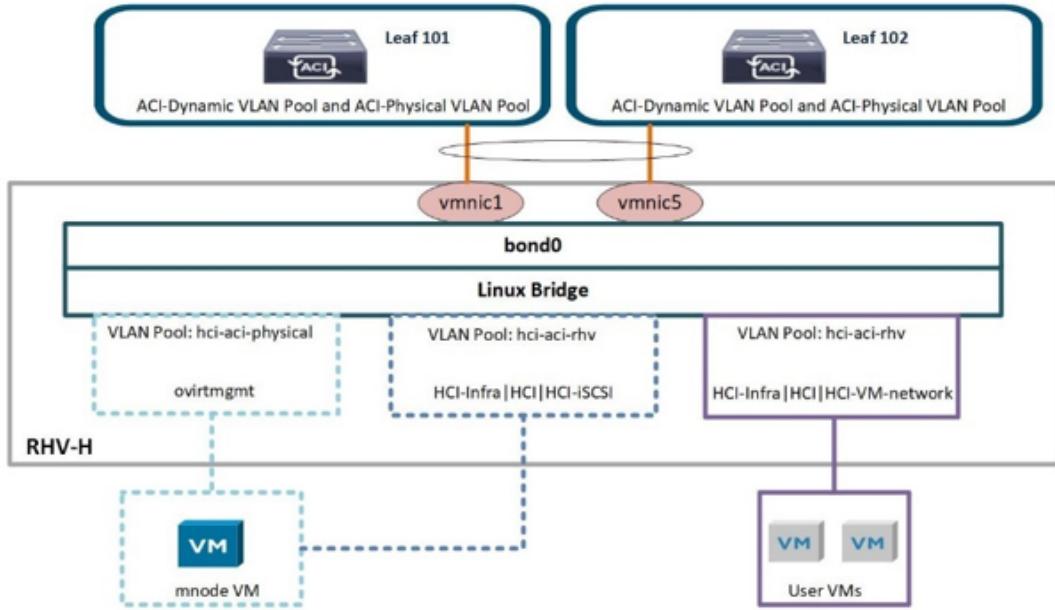
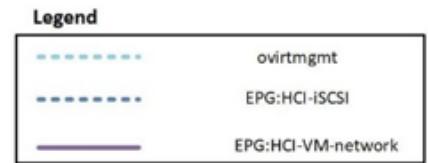
- Verify connectivity between Host and Engine
- Save network configuration

OK Cancel

The networking functionality for RHVH hosts in this solution is provided by Linux bridge.

### Linux Bridge

Linux Bridge is a default virtual switch on all Linux distributions that is usually used with KVM/QEMU-based hypervisors. It is designated to forward traffic between networks based on MAC addresses and thus is regarded as a layer-2 virtual switch. For more information, see the documentation [here](#). The following figure depicts the internal networking of Linux Bridge on RHV-H (as tested).



The following table outlines the necessary parameters and best practices for configuring and integrating Cisco ACI with Linux Bridge on RHV hosts.

Resource	Configuration considerations	Best Practices
Endpoint groups	<ul style="list-style-type: none"> <li>Separate EPG for native VLAN</li> <li>Static binding of interfaces towards HCI storage and compute nodes in native VLAN EPG to be on 802.1P mode</li> <li>Static binding of vPCs required on In-band management EPG and iSCSI EPG before RHV installation</li> </ul>	<ul style="list-style-type: none"> <li>Separate VLAN pool for VMM domain with dynamic allocation turned on</li> <li>Contracts between EPGs to be well defined. Allow only required ports for communication.</li> <li>Use unique native VLAN for discovery during Element cluster formation</li> <li>For EPGs corresponding to port-groups being attached to VMkernel ports, VMM domain to be attached with 'Pre-Provision' for Resolution Immediacy</li> </ul>
Interface policy	<ul style="list-style-type: none"> <li>One vPC policy group per RHV-H host</li> <li>One vPC policy group per NetApp HCI storage node</li> <li>LLDP enabled, CDP disabled</li> </ul>	<ul style="list-style-type: none"> <li>Recommended to use vPC towards RHV-H hosts</li> <li>Use 'LACP Active' for the port-channel policy</li> <li>Use only 'Graceful Convergence' and 'Symmetric Hashing' control bits for port-channel policy Use 'Layer4 Src-port' load balancing hashing method for port-channel policy Recommended to use vPC with LACP Active port-channel policy for interfaces towards NetApp HCI storage nodes</li> </ul>
VMM Integration	Do not migrate host management logical interfaces from ovirtmgmt to any other logical network	iSCSI host logical interface to be migrated to iSCSI logical network managed by ACI VMM integration



Except for the ovirtmgmt logical network, it is possible to create all other infrastructure logical networks on Cisco APIC and map them to the VMM domain. ‘ovirtmgmt’ logical network uses the static path binding on the In-band management EPG attached with the physical domain.

Next: [KVM on RHEL: NetApp HCI with Cisco ACI](#)

## KVM on RHEL: NetApp HCI with Cisco ACI

KVM (for Kernel-based Virtual Machine) is an open-source full virtualization solution for Linux on x86 hardware such as Intel VT or AMD-V. In other words, KVM lets you turn a Linux machine into a hypervisor that allows the host to run multiple, isolated VMs.

KVM converts any Linux machine into a type-1 (bare-metal) hypervisor. KVM can be implemented on any Linux distribution, but implementing KVM on a supported Linux distribution—like Red Hat Enterprise Linux—expands KVM’s capabilities. You can swap resources among guests, share common libraries, and optimize system performance.

### Workflow

The following high-level workflow was used to set up the virtual environment. Each of these steps might involve several individual tasks.

1. Install and configure Nexus 9000 switches in ACI mode, and install and configure APIC software on a UCS C-series server. See the [Install and Upgrade documentation](#) for detailed steps.
2. Configure and set up the ACI fabric by referring to the [documentation](#).
3. Configure the tenants, application profiles, bridge domains, and EPGs required for NetApp HCI nodes. NetApp recommends using a one-BD-to-one-EPG framework except for iSCSI. See the documentation [here](#) for more details. The minimum set of EPGs required are in-band management, iSCSI, VM Motion, VM-data network, and native.
4. Create the VLAN pool, physical domain, and AEP based on the requirements. Create the switch and interface profiles and policies for vPCs and individual ports. Then attach the physical domain and configure the static paths to the EPGs. See the [configuration guide](#) for more details. Also see this table <link> for best practices for integrating ACI with Open vSwitch on the RHEL–KVM hypervisor.



Use a vPC policy group for interfaces connecting to NetApp HCI storage and compute nodes.

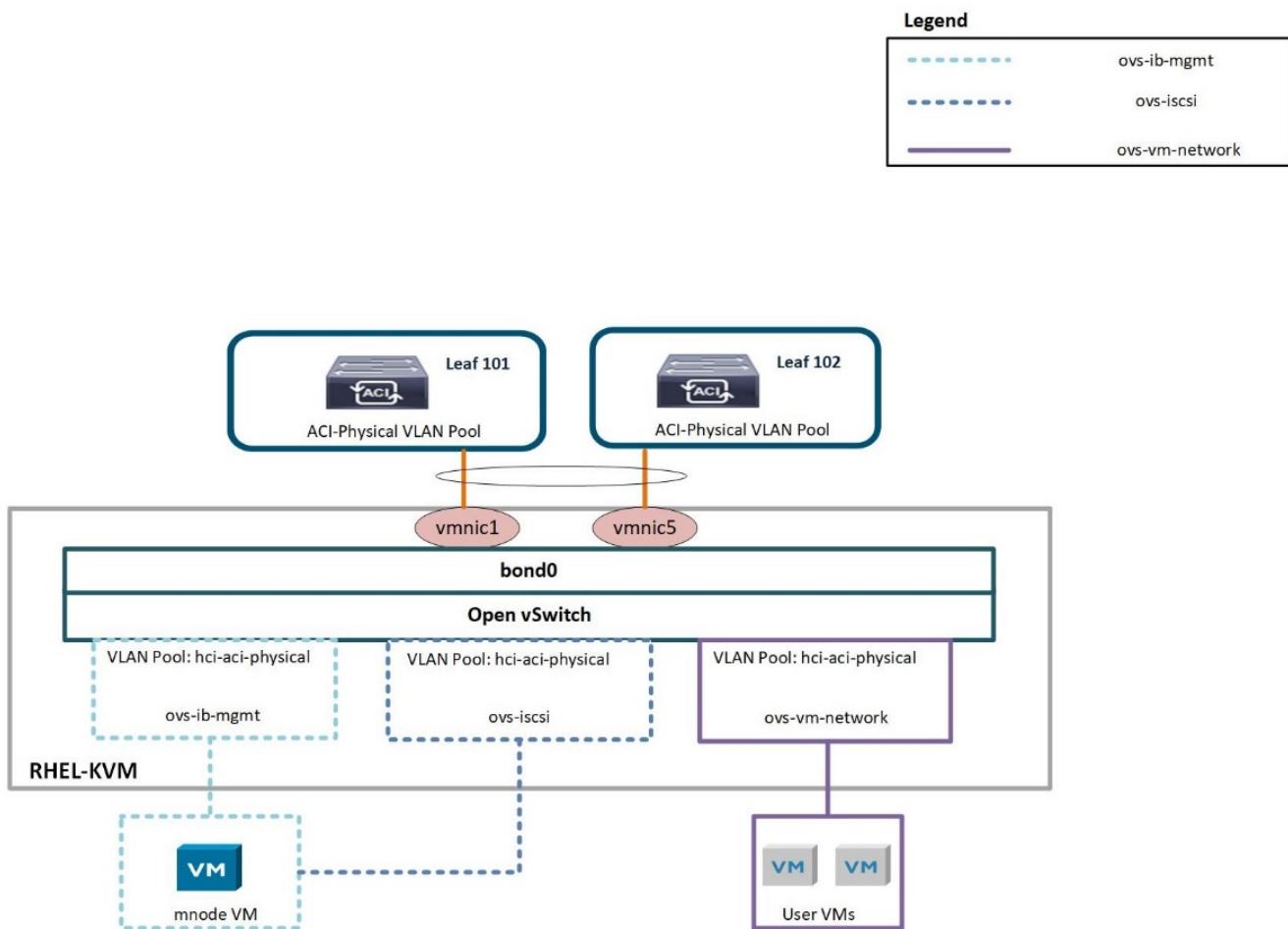
5. Create and assign contracts for tightly-controlled access between workloads. For more details on configuring the contracts, see the guide [here](#).
6. Install and configure a NetApp HCI Element cluster. Do not use NDE for this installation; rather, install a standalone Element cluster on HCI storage nodes. Then configure the required volumes for

the installation of RHEL. Install RHEL, KVM, and Open vSwitch on the NetApp HCI compute nodes. Configure storage pools on the hypervisor using Element volumes for a shared storage service for hosts and VMs. For more details on installation and configuration of KVM on RHEL, see the [Red Hat documentation](#). See the [OVS documentation](#) for details on configuring Open vSwitch.

7. RHEL KVM hypervisor's Open vSwitch cannot be VMM integrated with Cisco ACI. Physical domain and static paths must be configured on all required EPGs to allow the required VLANs on the interfaces connecting the ACI leaf switches and RHEL hosts. Also configure the corresponding OVS bridges on RHEL hosts and configure VMs to use those bridges. The networking functionality for the RHEL KVM hosts in this solution is achieved using Open vSwitch virtual switch.

### Open vSwitch

Open vSwitch is an open-source, enterprise-grade virtual switch platform. It uses virtual network bridges and flow rules to forward packets between hosts. Programming flow rules work differently in OVS than in the standard Linux Bridge. The OVS plugin does not use VLANs to tag traffic. Instead, it programs flow rules on the virtual switches that dictate how traffic should be manipulated before forwarded to the exit interface. Flow rules determine how inbound and outbound traffic should be treated. The following figure depicts the internal networking of Open vSwitch on an RHEL-based KVM host.



The following table outlines the necessary parameters and best practices for configuring Cisco ACI and Open vSwitch on RHEL based KVM hosts.

Resource	Configuration Considerations	Best Practices
Endpoint groups	<ul style="list-style-type: none"> <li>• Separate EPG for native VLAN</li> <li>• Static binding of interfaces towards HCI storage and compute nodes in native VLAN EPG to be on 802.1P mode</li> <li>• Static binding of vPCs required on in-band management EPG and iSCSI EPG before KVM installation</li> </ul>	<ul style="list-style-type: none"> <li>• Separate VLAN pool for physical domain with static allocation turned on</li> <li>• Contracts between EPGs to be well defined. Allow only required ports for communication.</li> <li>• Use unique native VLAN for discovery during Element cluster formation</li> </ul>
Interface Policy	<ul style="list-style-type: none"> <li>- One vPC policy group per RHEL host</li> <li>- One vPC policy group per NetApp HCI storage node</li> <li>- LLDP enabled, CDP disabled</li> </ul>	<ul style="list-style-type: none"> <li>• NetApp recommends using vPC towards RHV-H hosts</li> <li>• Use LACP Active for the port-channel policy</li> <li>• Use only Graceful Convergence and Symmetric Hashing control bits for port-channel policy</li> <li>• Use Layer4 Src-Port load-balancing hashing method for port-channel policy</li> <li>• NetApp recommends using vPC with LACP Active port-channel policy for interfaces towards NetApp HCI storage nodes</li> </ul>

[Next: ONTAP on AFF: NetApp HCI and Cisco ACI](#)

### ONTAP on AFF: NetApp HCI and Cisco ACI

NetApp AFF is a robust storage platform that provides low-latency performance, integrated data protection, multiprotocol support, and nondisruptive operations. Powered by NetApp ONTAP data management software, NetApp AFF ensures

nondisruptive operations, from maintenance to upgrades to complete replacement of your storage system.

NetApp ONTAP is a powerful storage operating system with capabilities like inline compression, nondisruptive hardware upgrades, and cross-storage import. A NetApp ONTAP cluster provides a unified storage system with simultaneous data access and management of Network File System (NFS), Common Internet File System (CIFS), iSCSI, Fibre Channel (FC), Fibre Channel over Ethernet (FCoE), and NVMe/FC protocols. ONTAP provides robust data protection capabilities, such as NetApp MetroCluster, SnapLock, Snapshot copies, SnapVault, SnapMirror, SyncMirror technologies and more. For more information, see the [ONTAP documentation](#).

To extend the capabilities of storage to file services and add many more data protection abilities, ONTAP can be used in conjunction with NetApp HCI. If NetApp ONTAP already exists in your environment, you can easily integrate it with NetApp HCI and Cisco ACI.

## Workflow

The following high-level workflow was used to set up the environment. Each of these steps might involve several individual tasks.

1. Create a separate bridge domain and EPG on ACI for NFS and/or other protocols with the corresponding subnets. You can use the same HCI-related iSCSI EPGs.
2. Make sure you have proper contracts in place to allow inter-EPG communication for only the required ports.
3. Configure the interface policy group and selector for interfaces towards AFF controllers. Create a vPC policy group with the LACP Active mode for port-channel policy.

## PC/VPC Interface Policy Group - Storage-AFF-01

Properties

Name: Storage-AFF-01

Description: optional

Link Aggregation Type: Port Channel **VPC**

Link Level Policy: 10G-Auto

CDP Policy: CDP-Enabled

MCP Policy: select a value

CoPP Policy: select a value

LLDP Policy: LLDP-Enabled

STP Interface Policy: select a value

Egress Data Plane Policing Policy: select a value

Ingress Data Plane Policing Policy: select a value

Priority Flow Control Policy: select a value

Fibre Channel Interface Policy: select a value

Slow Drain Policy: select a value

Port Channel Policy: LACP-Active

4. Attach both a physical and VMM domain to the EPGs created. Attach the vPC policy as static paths and, in the case of theCisco AVE virtual switch, use Native switching mode when you attach the VMM domain.

VMware/hci-vmware-ave VMM Domain  On Demand Immediate formed  e.g., vlan-1  e.g., vlan-1 native VLAN

**Update** **Cancel**

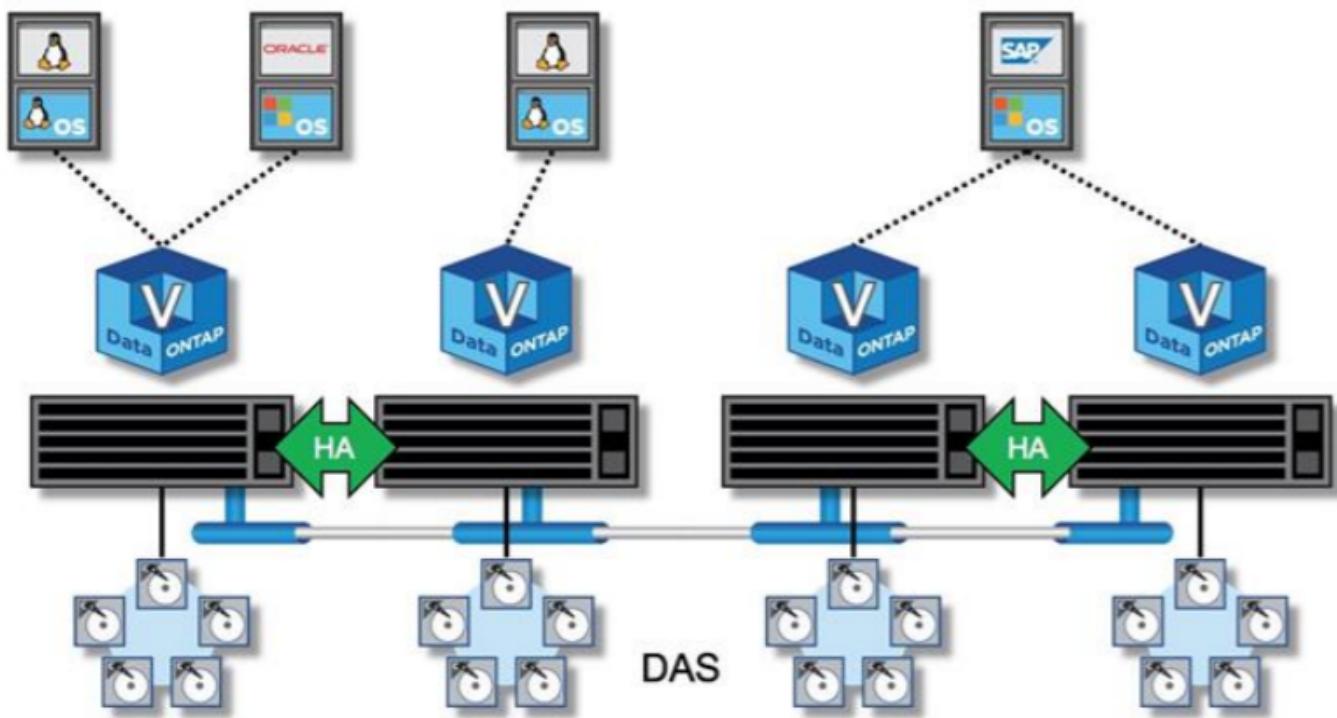
5. Install and configure an ONTAP cluster on the AFF controllers. Then create and configure NFS and/or iSCSI volumes/LUNs. See the [AFF and ONTAP documentation](#) for more information.
6. Create a VMkernel adapter (in the case of VMware ESXi) or a logical interface (in the case of RHV-H and RHEL-KVM hosts) attaching the NFS (or other protocols) port group or logical network.
7. Create additional datastores, storage domains, or storage pools on hypervisors (VMware, RHV, or KVM) using AFF storage.

[Next: ONTAP Select with VMware vSphere: NetApp HCI and Cisco ACI](#)

## ONTAP Select with VMware vSphere: NetApp HCI and Cisco ACI

NetApp ONTAP Select is the NetApp solution for software-defined storage (SDS), bringing enterprise-class storage management features to the software-defined data center. ONTAP Select extends ONTAP functionality to extreme edge use cases including IoT and tactical servers as a software-defined storage appliance that acts as a full storage system. It can run as a simple VM on top of a virtual environment to provide a flexible and scalable storage solution.

Running ONTAP as software on top of another software application allows you to leverage much of the qualification work done by the hypervisor. This capability is critical for helping us to rapidly expand our list of supported platforms. Also, positioning ONTAP as a virtual machine (VM) allows customers to plug into existing management and orchestration frameworks, which allows rapid provisioning and end-to-end automation from deployment to sunsetting. The following figure provides an overview of a four-node ONTAP Select instance.



Deploying ONTAP Select in the environment to use the storage offered by NetApp HCI extends the capabilities of NetApp Element.

### Workflow

The following workflow was used to set up the environment. In this solution, we deployed a two-node ONTAP Select cluster. Each of these steps might involve several individual tasks.

1. Create an L2 BD and EPG for the OTS cluster's internal communication and attach the VMM domain to the EPG in the Native switching mode (in case of a Cisco AVE virtual switch) with Pre-Provision Resolution Immediacy.

## EPG - HCI-Select-Internal

The screenshot shows the 'Properties' dialog for an EPG named 'HCI-Select-Internal'. The dialog includes the following fields:

- Contract Exception Tag: (empty input field)
- QoS class: Unspecified (dropdown menu)
- Custom QoS: select a value (dropdown menu)
- Data-Plane Policer: select a value (dropdown menu)
- Intra EPG Isolation: Enforced (button), Unenforced (button, highlighted in blue)
- Preferred Group Member: Exclude (button), Include (button)
- Flood on Encapsulation: Disabled (button), Enabled (button)
- Configuration Status: applied
- Configuration Issues:
  - Label Match Criteria: AtleastOne (dropdown menu)
  - Bridge Domain: SELECT-Internal (dropdown menu, highlighted in blue)
- Resolved Bridge Domain: HCI-Infra/SELECT-Internal
- Monitoring Policy: select a value (dropdown menu)
- FHS Trust Control Policy: select a value (dropdown menu)

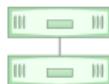
2. Verify that you have a VMware vSphere license.
3. Create a datastore that hosts OTS.
4. Deploy and configure ONTAP Select according to the [ONTAP Select documentation](#).

## **i Cluster Details**

Name	hci-aci-ontap-select	Cluster Size	2 node cluster (1 HA Pairs)
ONTAP Image Version	9.7	Licensing	evaluation
IPv4 Address	172.22.9.81	Cluster MTU	9000
Netmask	255.255.255.0	Domain Names	cie.netapp.com
Gateway	172.22.9.1	Server IP Addresses	10.61.184.251, 10.61.184.252
Mediator Status	HA Active	NTP Server	10.61.184.48
Last Refresh	-		

## **i Node Details**

### > HA Pair 1



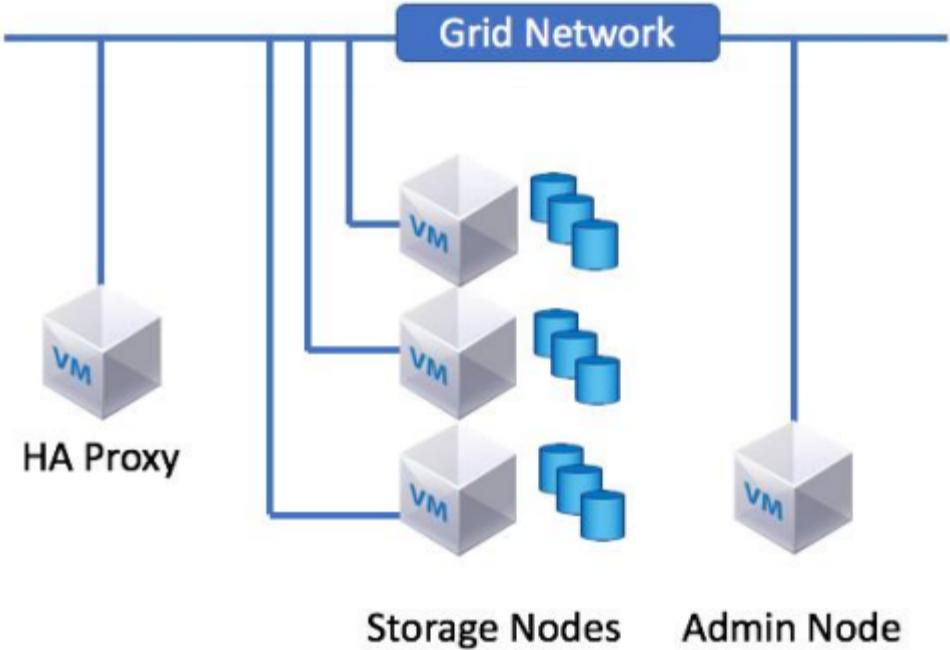
Node 1	hci-aci-ontap-select... — 2 TB +	Host 1	172.22.9.61 — (Small (4 CPU, 16 GB Memory))
Node 2	hci-aci-ontap-select... — 2 TB +	Host 2	172.22.9.60 — (Small (4 CPU, 16 GB Memory))

5. Create additional datastores using ONTAP Select to make use of additional capabilities.

Next: [StorageGRID with VMware vSphere: NetApp HCI and Cisco ACI](#)

## **StorageGRID with VMware vSphere: NetApp HCI and Cisco ACI**

StorageGRID is a robust software-defined, object-based storage platform that stores and manages unstructured data with a tiered approach along with intelligent policy-driven management. It allows you to manage data while optimizing durability, protection, and performance. StorageGRID can also be deployed as hardware or as an appliance on top of a virtual environment that decouples storage management software from the underlying hardware. StorageGRID opens a new realm of supported storage platforms, increasing flexibility and scalability. StorageGRID platform services are also the foundation for realizing the promise of the hybrid cloud, letting you tier and replicate data to public or other S3-compatible clouds. See the [StorageGRID](#) documentation for more details. The following figure provides an overview of StorageGRID nodes.



### Workflow

The following workflow was used to set up the environment. Each of these steps might involve several individual tasks.

1. Create an L2 BD and EPG for the grid network used for internal communication between the nodes in the StorageGRID system. However, if your network design for StorageGRID consists of multiple grid networks, then create an L3 BD instead of an L2 BD. Attach the VMM domain to the EPG with the Native switching mode (in the case of a Cisco AVE virtual switch) and with Pre-Provision Resolution Immediacy. The corresponding port group is used for the grid network on StorageGRID nodes.

# EPG - GridNetwork

The screenshot shows the configuration interface for an EPG named 'GridNetwork'. At the top, there are four small circular icons: a red one with an 'X', an orange one with a checkmark, a yellow one with an exclamation mark, and a green one with a question mark.

**Properties**

QoS class: Unspecified

Custom QoS: select a value

Data-Plane Policer: select a value

Intra EPG Isolation: Enforced (highlighted in blue)

Preferred Group Member: Exclude (highlighted in blue)

Flood on Encapsulation: Disabled (highlighted in blue)

Configuration Status: applied

Configuration Issues:

Label Match Criteria: AtleastOne

Bridge Domain: GridNetwork-BD (highlighted in blue)

Resolved Bridge Domain: HCI-Infra/GridNetwork-BD

Monitoring Policy: select a value

FHS Trust Control Policy: select a value

EPG Contract Master: (empty field)

2. Create a datastore to host the StorageGRID nodes.
3. Deploy and configure StorageGRID. For more details on installation and configuration, see the [StorageGRID documentation](#). If the environment already has ONTAP or ONTAP Select, then you can use the NetApp Fabric Pool feature. Fabric Pool is an automated storage tiering feature in which active data resides on local high-performance solid-state drives (SSDs) and inactive data is tiered to low-cost object storage. It was first made available in NetApp ONTAP 9.2. For more information on Fabric Pool, see the documentation [here](#).

Next: Validation Results

## Validation Results

We used the iPerf tool for testing network throughput, and the baseline expectation was that the test systems should achieve throughput within 10% of the maximum line rate. Test results for different virtual switches is indicated in the following table.

For storage IOPS subsystem measurement, we used the IOmeter tool. The baseline expectation was that the test systems should achieve read/write throughput within 10% of the maximum. Test results for different hypervisors is indicated in the following table.

We considered the following scenarios for the network line rate and storage IOPS testing:

### VMware

- VMs on a NetApp HCI datastore (with and without micro-segmentation)
- VMs on a NetApp ONTAP datastore
- VMs on a NetApp ONTAP Select datastore

### Red Hat Virtualization

- VMs on a NetApp HCI datastore
- VMs on a NetApp ONTAP datastore

### KVM (RHEL)

- VMs on a NetApp HCI datastore

### Miscellaneous

- One VM on RHV with a NetApp HCI datastore and one VM on VMware vSphere with a NetApp ONTAP datastore.

Hypervisor	Virtual Switch	iPerf	IOmeter	Micro-segmentation
VMware	VDS	Pass	Pass	Pass
RHV	Linux Bridge	Pass	Pass	N/A
RHEL-KVM	Open vSwitch	Pass	Pass	N/A

[Next: Where to Find Additional Information](#)

## Where to Find Additional Information

To learn more about the information that is described in this document, review the following documents and/or websites:

- NetApp HCI Documentation

<https://www.netapp.com/us/documentation/hci.aspx>

- Cisco ACI Documentation

<https://www.cisco.com/c/en/us/solutions/data-center-virtualization/application-centric-infrastructure/index.html>

- Cisco Nexus 9000 Series Switches

<http://www.cisco.com/c/en/us/products/switches/nexus-9000-series-switches/index.html>

- NetApp AFF A-series

<http://www.netapp.com/us/products/storage-systems/all-flash-array/aff-a-series.aspx>

- ONTAP Documentation

<https://docs.netapp.com/ontap-9/index.jsp>

- ONTAP Select Documentation

<https://docs.netapp.com/us-en/ontap-select/>

- StorageGRID Documentation

<https://docs.netapp.com/sgws-113/index.jsp>

- Red Hat Virtualization

[https://access.redhat.com/documentation/en-us/red\\_hat\\_virtualization/4.3/](https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/)

- VMware vSphere

<https://docs.vmware.com/en/VMware-vSphere/index.html>

- VMware vCenter Server

<http://www.vmware.com/products/vcenter-server/overview.html>

- NetApp Interoperability Matrix Tool

<http://now.netapp.com/matrix>

- Cisco ACI Virtualization Compatibility Matrix

<https://www.cisco.com/c/dam/en/us/td/docs/Website/datacenter/aci/virtualization/matrix/virtmatrix.html>

- VMware Compatibility Guide

<http://www.vmware.com/resources/compatibility>

# Containers and DevOps

## NVA-1141: NetApp HCI with Anthos, Design and Deployment

Alan Cowles

The program solutions described in this document are designed and thoroughly tested to minimize deployment risks and accelerate time to market.

This document is for NetApp and partner solutions engineers and customer strategic decision makers. It describes the architecture design considerations that were used to determine the specific equipment, cabling, and configurations required to support the validated workload.

NetApp HCI with Anthos is a verified, best-practice hybrid cloud architecture for the deployment of an on-premises Google Kubernetes Engine (GKE) environment in a reliable and dependable manner. This NetApp Verified Architecture reference document serves as both a design guide and a deployment validation of the Anthos solution on NetApp HCI. The architecture described in this document has been validated by subject matter experts at NetApp and Google to provide the advantage of running Anthos on NetApp HCI within your own enterprise data-center environment.

NetApp HCI, is the industry's first and leading disaggregated hybrid cloud infrastructure, providing the widely recognized benefits of hyperconverged solutions. Benefits include lower TCO and ease of acquisition, deployment, and management for virtualized workloads, while also allowing enterprise customers to independently scale compute and storage resources as needed. NetApp HCI with Anthos provides an on-premises, cloud-like experience for the deployment of containerized workloads managed by Anthos GKE on-premises. This solution provides simplified management, detailed metrics, and a range of additional functionalities that enable the easy movement of workloads deployed both on-site and in the cloud.

## Features

With NetApp HCI for Anthos, you can deploy a fully integrated, production-grade Anthos GKE environment in your on-premises data center, which allows you to take advantage of the following features:

- NetApp HCI compute and storage nodes
  - Enterprise-grade hyperconverged infrastructure designed for hybrid cloud workloads
  - NetApp Element storage software
  - Intel-based server compute nodes, including options for Nvidia GPUs
- VMware vSphere 6.5
  - Enterprise hypervisor solution for deployment and management of virtual infrastructures

- Anthos GKE in Google Cloud and On-Prem
  - Deploy Anthos GKE instances in Google Cloud or on NetApp HCI

The NetApp Verified Architecture program gives customers reference configurations and sizing guidance for specific workloads and use cases.

[Next: Solution Components](#)

## Solution Components

The solution described in this document builds on the solid foundation of NetApp HCI, VMware vSphere, and the Anthos hybrid-cloud Kubernetes data center solution.

### NetApp HCI

By providing an agile turnkey infrastructure platform, NetApp HCI enables you to run enterprise-class virtualized and containerized workloads in an accelerated manner. At its core, NetApp HCI is designed to provide predictable performance, linear scalability of both compute and storage resources, and a simple deployment and management experience.

- **Predictable.** One of the biggest challenges in a multitenant environment is delivering consistent, predictable performance for all your workloads. Running multiple enterprise-grade workloads can result in resource contention, in which one workload might interfere with the performance of another. NetApp HCI alleviates this concern with storage quality-of-service (QoS) limits that are available natively with NetApp Element® software. Element enables the granular control of every application and volume, helps to eliminate noisy neighbors, and satisfies enterprise performance SLAs. NetApp HCI multitenancy capabilities can help eliminate many traditional performance-related problems.
- **Flexible.** Previous generations of hyperconverged infrastructures often required fixed resource ratios, limiting deployments to four-node and eight-node configurations. NetApp HCI is a disaggregated hyper-converged infrastructure that can scale compute and storage resources independently. Independent scaling prevents costly and inefficient overprovisioning, eliminates the 10% to 30% HCI tax from controller VM overhead, and simplifies capacity and performance planning. NetApp HCI is available in mix-and-match small, medium, and large storage and compute configurations.

The architectural design choices offered enable you to confidently scale on your terms, making HCI viable for core Tier 1 data center applications and platforms. NetApp HCI is architected in building blocks at either the chassis or the node level. Each chassis can hold four nodes in a mixed configuration of storage or compute nodes.

- **Simple.** A driving imperative within the IT community is to simplify deployment and automate routine tasks, eliminating the risk of user error while freeing up resources to focus on more interesting, higher-value projects. NetApp HCI can help your IT department become more agile and responsive by both simplifying deployment and ongoing management. The NetApp Deployment

Engine (NDE) tool eases the configuration and deployment of physical infrastructure, including the installation of the VMware vSphere environment and the integration of the NetApp Element Plug-in for vCenter Server. With NDE, future scaling operations can be performed without difficulty.

### NetApp HCI Configuration

NetApp HCI is an enterprise-scale disaggregated hybrid cloud infrastructure (HCI) solution that delivers compute and storage resources in an agile, scalable, and easy-to-manage two-rack unit (2RU) four-node building block. It can also be configured with 1RU compute and server nodes. The minimum deployment consists of four NetApp HCI storage nodes and two NetApp HCI compute nodes. The compute nodes are installed as VMware ESXi hypervisors in an HA cluster without the enforcement of VMware DRS anti-affinity rules. This minimum deployment can be easily scaled to fit customer enterprise workload demands by adding additional NetApp HCI storage or compute nodes to expand available storage.

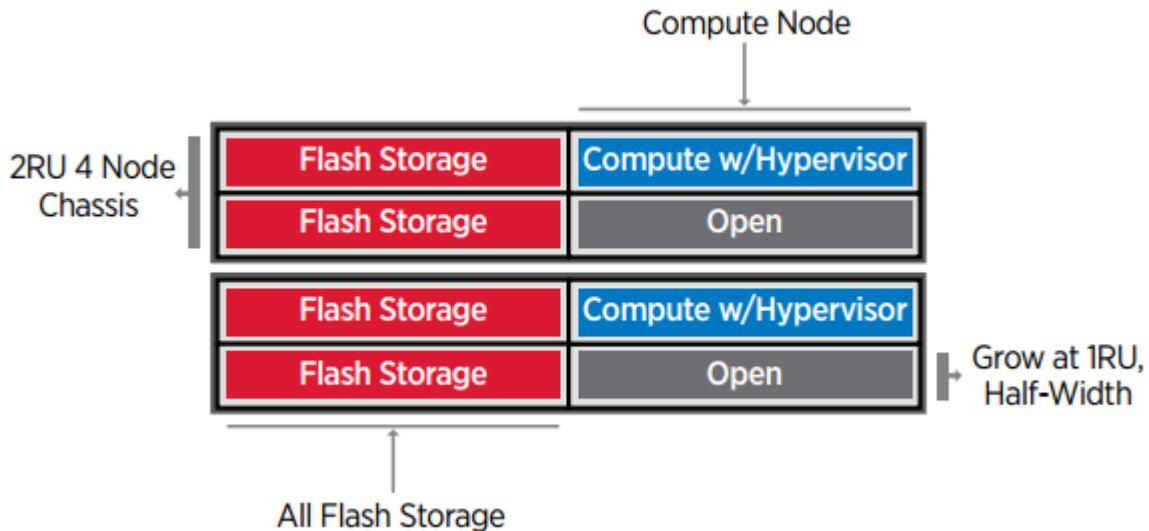


Figure 1. NetApp HCI minimum configuration.

The design for NetApp HCI for Anthos consists of the following components in a minimum starting configuration:

- NetApp H-Series all-flash storage nodes running NetApp Element software
- NetApp H-Series compute nodes running VMware vSphere 6.5U2

For more information about compute and storage nodes in NetApp HCI, see the [NetApp HCI Datasheet](#).

### NetApp Element Software

NetApp Element software provides modular, scalable performance, with each storage node delivering guaranteed capacity and throughput to the environment. You can also specify per-volume storage QoS policies to support dedicated performance levels for even the most demanding workloads.

## iSCSI Login Redirection and Self-Healing Capabilities

NetApp Element software uses the iSCSI storage protocol, a standard way to encapsulate SCSI commands on a traditional TCP/IP network. When SCSI standards change or when Ethernet network performance improves, the iSCSI storage protocol benefits without the need for any changes.

Although all storage nodes have a management IP and a storage IP, NetApp Element software advertises a single storage virtual IP address (SVIP address) for all storage traffic in the cluster. As a part of the iSCSI login process, storage can respond that the target volume has been moved to a different address, and therefore it cannot proceed with the negotiation process. The host then reissues the login request to the new address in a process that requires no host-side reconfiguration. This process is known as iSCSI login redirection.

iSCSI login redirection is a key part of the NetApp Element software cluster. When a host login request is received, the node decides which member of the cluster should handle the traffic based on IOPS and the capacity requirements for the volume. Volumes are distributed across the NetApp Element software cluster and are redistributed if a single node is handling too much traffic for its volumes or if a new node is added. Multiple copies of a given volume are allocated across the array. In this manner, if a node failure is followed by volume redistribution, there is no effect on host connectivity beyond a logout and login with redirection to the new location. With iSCSI login redirection, a NetApp Element software cluster is a self-healing, scale-out architecture that is capable of nondisruptive upgrades and operations.

## NetApp Element Software Cluster QoS

A NetApp Element software cluster allows QoS to be dynamically configured on a per-volume basis. You can use per-volume QoS settings to control storage performance based on SLAs that you define. The following three configurable parameters define the QoS:

- **Minimum IOPS.** The minimum number of sustained IOPS that the NetApp Element software cluster provides to a volume. The minimum IOPS configured for a volume is the guaranteed level of performance for a volume. Per-volume performance does not drop below this level.
- **Maximum IOPS.** The maximum number of sustained IOPS that the NetApp Element software cluster provides to a specific volume.
- **Burst IOPS.** The maximum number of IOPS allowed in a short burst scenario. The burst duration setting is configurable, with a default of 1 minute. If a volume has been running below the maximum IOPS level, burst credits are accumulated. When performance levels become very high and are pushed, short bursts of IOPS beyond the maximum IOPS are allowed on the volume.

## Multitenancy

Secure multitenancy is achieved with the following features:

- **Secure authentication.** The Challenge-Handshake Authentication Protocol (CHAP) is used for secure volume access. The Lightweight Directory Access Protocol (LDAP) is used for secure access to the cluster for management and reporting.

- **Volume access groups (VAGs).** Optionally, VAGs can be used in lieu of authentication, mapping any number of iSCSI initiator-specific iSCSI Qualified Names (IQNs) to one or more volumes. To access a volume in a VAG, the initiator's IQN must be in the allowed IQN list for the group of volumes.
- **Tenant virtual LANs (VLANs).** At the network level, end-to-end network security between iSCSI initiators and the NetApp Element software cluster is facilitated by using VLANs. For any VLAN that is created to isolate a workload or a tenant, NetApp Element Software creates a separate iSCSI target SVIP address that is accessible only through the specific VLAN.
- **VPN routing/forwarding (VRF)-enabled VLANs.** To further support security and scalability in the data center, NetApp Element software allows you to enable any tenant VLAN for VRF-like functionality. This feature adds these two key capabilities:
  - **L3 routing to a tenant SVIP address.** This feature allows you to situate iSCSI initiators on a separate network or VLAN from that of the NetApp Element software cluster.
  - **Overlapping or duplicate IP subnets.** This feature enables you to add a template to tenant environments, allowing each respective tenant VLAN to be assigned IP addresses from the same IP subnet. This capability can be useful for service provider environments where scale and preservation of IP-space are important.

## Enterprise Storage Efficiencies

The NetApp Element software cluster increases overall storage efficiency and performance. The following features are performed inline, are always on, and require no manual configuration by the user:

- **Deduplication.** The system only stores unique 4K blocks. Any duplicate 4K blocks are automatically associated to an already stored version of the data. Data is on block drives and is mirrored by using Element Helix data protection. This system significantly reduces capacity consumption and write operations within the system.
- **Compression.** Compression is performed inline before data is written to NVRAM. Data is compressed, stored in 4K blocks, and remains compressed in the system. This compression significantly reduces capacity consumption, write operations, and bandwidth consumption across the cluster.
- **Thin provisioning.** This capability provides the right amount of storage at the time that you need it, eliminating capacity consumption that caused by overprovisioned volumes or underutilized volumes.
- **Helix.** The metadata for an individual volume is stored on a metadata drive and is replicated to a secondary metadata drive for redundancy.

**Note:** Element was designed for automation. All the storage features mentioned above can be managed with APIs. These APIs are the only method that the UI uses to control the system whether actions are performed directly through Element or through the vSphere plug-in for Element.

## VMware vSphere

VMware vSphere is the industry leading virtualization solution built on VMware ESXi hypervisors and managed by vCenter Server, which provides advanced functionality often required for enterprise datacenters. When using the NDE with NetApp HCI, a VMware vSphere environment is configured and installed. The following features are available after the environment is deployed:

- **Centralized Management.** Through vSphere, individual hypervisors can be grouped into data centers and combined into clusters, allowing for advanced organization to ease the overall management of resources.
- **VMware HA.** This feature allows virtual guests to restart automatically if their host becomes unavailable. By enabling this feature, virtual guests become fault tolerant, and virtual infrastructures experience minimal disruption when there are physical failures in the environment.
- **VMware Distributed Resource Scheduler (DRS).** VMware vMotion allows for the movement of guests between hosts nondisruptively when certain user-defined thresholds are met. This capability makes the virtual guests in an environment highly available.
- **vSphere Distributed Switch (vDS).** A virtual switch is controlled by the vCenter server, enabling centralized configuration and management of connectivity for each host by creating port groups that map to the physical interfaces on each host.

## Anthos

Anthos is a hybrid-cloud Kubernetes data center solution that enables organizations to construct and manage modern hybrid-cloud infrastructures, while adopting agile workflows focused on application development. Anthos on VMware, a solution built on open-source technologies, runs on-premises in a VMware vSphere-based infrastructure, which can connect and interoperate with Anthos GKE in Google Cloud.

Adopting containers, service mesh, and other transformational technologies enables organizations to experience consistent application development cycles and production-ready workloads in local and cloud-based environments. The following figure depicts the Anthos solution and how a deployment in an on-premises data center interconnects with infrastructure in the cloud.

For more information about Anthos, see the Anthos website located [here](#).

Anthos provides the following features:

- **Anthos configuration management.** Automates the policy and security of hybrid Kubernetes deployments.
- **Anthos Service Mesh.** Enhances application observability, security, and control with an Istio-powered service mesh.
- **Google Cloud Marketplace for Kubernetes Applications.** A catalog of curated container applications available for easy deployment.
- **Migrate for Anthos.** Automatic migration of physical services and VMs from on-premises to the cloud.

- **Stackdriver.** Management service offered by Google for logging and monitoring cloud instances.

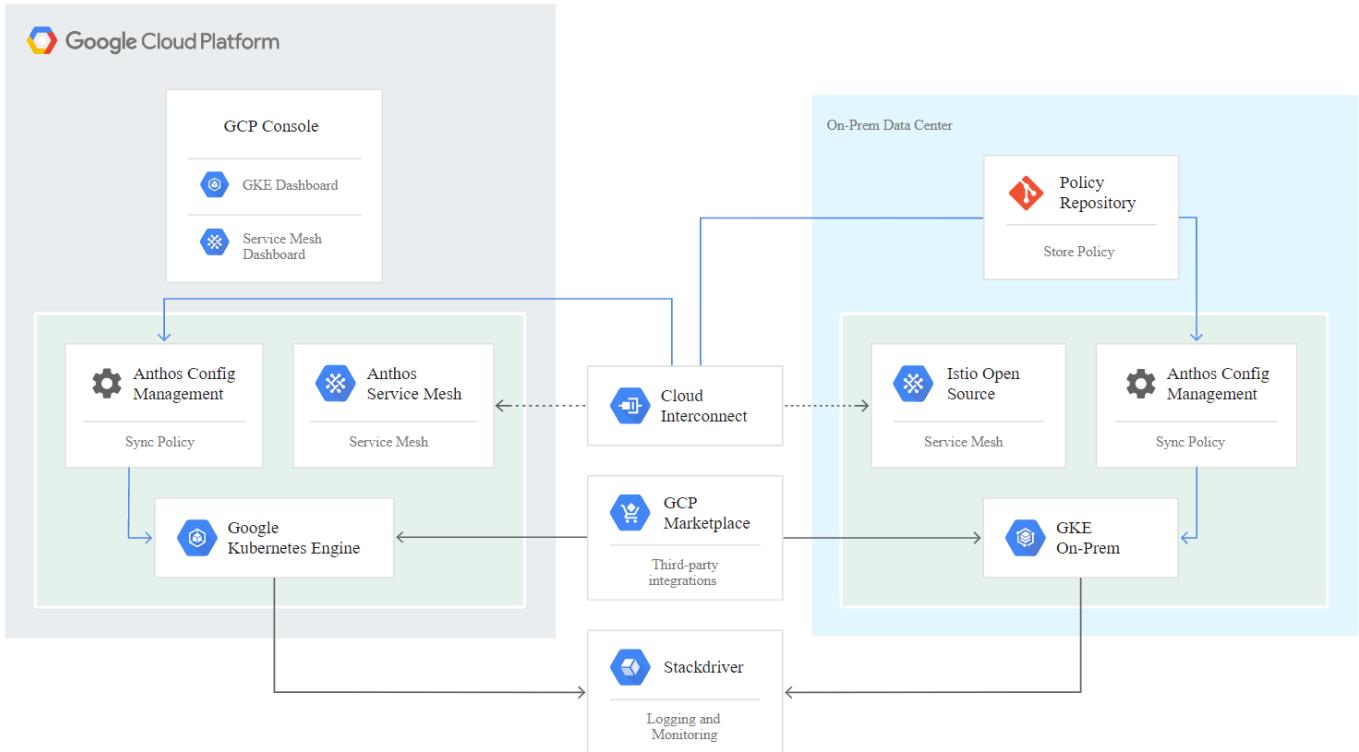


Figure 2. Anthos architecture.

### Containers and Kubernetes Orchestration

Container technology has been available to developers for a long time. However, it has only recently become a core concept in data center architecture and design as more enterprises have adopted application-specific workload requirements.

A traditional development environment requires a dedicated development host deployed on either a bare-metal or virtual server. Such environments require each application to have its own dedicated machine, complete with operating system (OS) and networking connectivity. These machines often must be managed by the enterprise system administration team, who must account for the application versions installed as well as host OS patches. In contrast, containers by design require less overhead to deploy. All that is needed is the packaging of application code and supporting libraries together, because all other services depend on the host OS. Rather than managing a complete virtual machine (VM) environment, developers can instead focus on the application development process.

As container technology began to find appeal in the enterprise landscape, many enterprise features, such as fault tolerance and application scaling, were both requested and expected. In response, Google partnered with the Linux Foundation to form the Cloud Native Computing Foundation (CNCF). Together, they introduced Kubernetes (K8s), an open-source platform for orchestrating and managing containers. Kubernetes was designed by Google to be a successor to both the Omega and Borg container management platforms that had been used in their data centers in the previous decade.

## Anthos GKE

Anthos GKE is a certified distribution of Kubernetes in the Google Cloud. It allows end users to easily deploy managed, production-ready Kubernetes clusters, enabling developers to focus primarily on application development rather than on the management of their environment. Deploying Kubernetes clusters in Anthos GKE offers the following benefits:

- **Simplifying Deployment of Applications.** Anthos GKE allows for rapid development, deployment, and updates of applications and services. By providing simple descriptions of the expected system resources (compute, memory, and storage) required by the application containers, the Kubernetes Engine automatically provisions and manages the lifecycle of the cluster environment.
- **Ensuring Availability of Clusters.** The environment is made extremely accessible and easy to manage by using the dashboard built into the Google Cloud console. Anthos GKE clusters are continually monitored by Google Site Reliability Engineers (SREs) to make sure that clusters behave as expected by collecting regular metrics and observing the use of assigned system resources. A user can also leverage available health checks to make sure that their deployed applications are highly available and that they can recover easily should something go awry.
- **Securing Clusters in Google Cloud.** An end user can ensure that clusters are secure and accessible by customizing network policies available from Google Cloud's Global Virtual Private Cloud. Public services can be placed behind a single global IP address for load balancing purposes. A single IP can help provide high availability for applications and protect against Distributed Denial of Service (DDOS) and other forms of attacks that might hinder service performance.
- **Easily Scaling to Meet Requirements.** An end user can enable auto-scaling on their cluster to easily counter both planned and unexpected increases in application demands. Auto-scaling helps make sure that system resources are always available by increasing capacity during high-demand windows. It also allows the cluster to return to its previous state and size after peak demand wanes.

## Anthos on VMware

Anthos on VMware is an extension of Google Kubernetes Engine that is deployed in an end user's private data center. An organization can deploy the same applications designed to run in containers in Google Cloud in Kubernetes clusters on premises. Anthos on VMware offers the following benefits:

- **Cost Savings.** End users can realize significant cost savings by utilizing their own physical resources for their application deployments instead of provisioning resources in their Google Cloud environment.
- **Develop Then Publish.** On-premises deployments can be used while applications are in development, which allows for testing of applications in the privacy of a local data center before being made publicly available in the cloud.
- **Security Requirements.** Customers with increased security concerns or sensitive data sets that cannot be stored in the public cloud are able to run their applications from the security of their own data centers, thereby meeting organizational requirements.

[Next: Design Considerations](#)

## Design Considerations

This section describes the design considerations necessary for the successful deployment of the NetApp HCI Anthos solution.

### Port Identification

NetApp HCI consists of NetApp H-Series nodes dedicated to either compute or storage. Both node configurations are available with two 1GbE ports (ports A and B) and two 10/25 GbE ports (ports C and D) on board. The compute nodes have additional 10/25GbE ports (ports E and F) available in the first mezzanine slot. Each node also has an additional out-of-band management port that supports Intelligent Platform Management Interface (IPMI) functionality. The following figure identifies each of these ports on the rear of an H410C node.

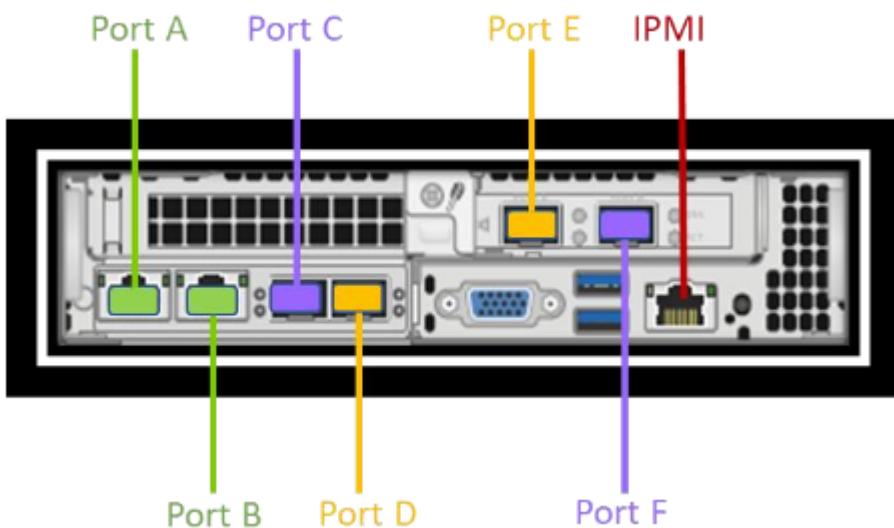


Figure 3. NetApp HCI network ports (compute node).

### Network Design

The NetApp HCI with Anthos solution uses two data switches to provide primary data connectivity at 25Gbps. It also uses two additional management switches that provide connectivity at 1Gbps for in-band management for the storage nodes and out-of-band management for IPMI functionality.

### Cabling Storage Nodes

The management ports A and B must be active on each storage node to run NDE, configure the NetApp HCI cluster, and provide management accessibility to Element after the solution is deployed. The two 25Gbps ports (C and D) should be connected, one to each data switch, to provide physical fault tolerance. The switch ports should be configured for multi-chassis link aggregation (MLAG) and the data ports on the node should be configured for LACP with jumbo-frames support enabled. The IPMI ports on each node can be used to remotely manage the node after it is installed in a data center. With

IPMI, the node can be accessed with a web-browser-based console to run the initial installation, run diagnostics, and reboot or shut down the node if necessary.

### Cabling Compute Nodes

The 25Gbps ports on the compute nodes are cabled with one onboard port © cabled to one data switch, and an additional port from the PCI slot (E) cabled to the second switch to provide physical fault tolerance. These ports should be configured to support jumbo frames. Connectivity for the node is managed by the vDS after VMware vSphere is deployed in the environment. The IPMI ports can also be used to remotely manage the node after it is installed in a data center. With IPMI, the node can be accessed via a web-browser-based console to run diagnostics and to be rebooted or shut down if necessary.

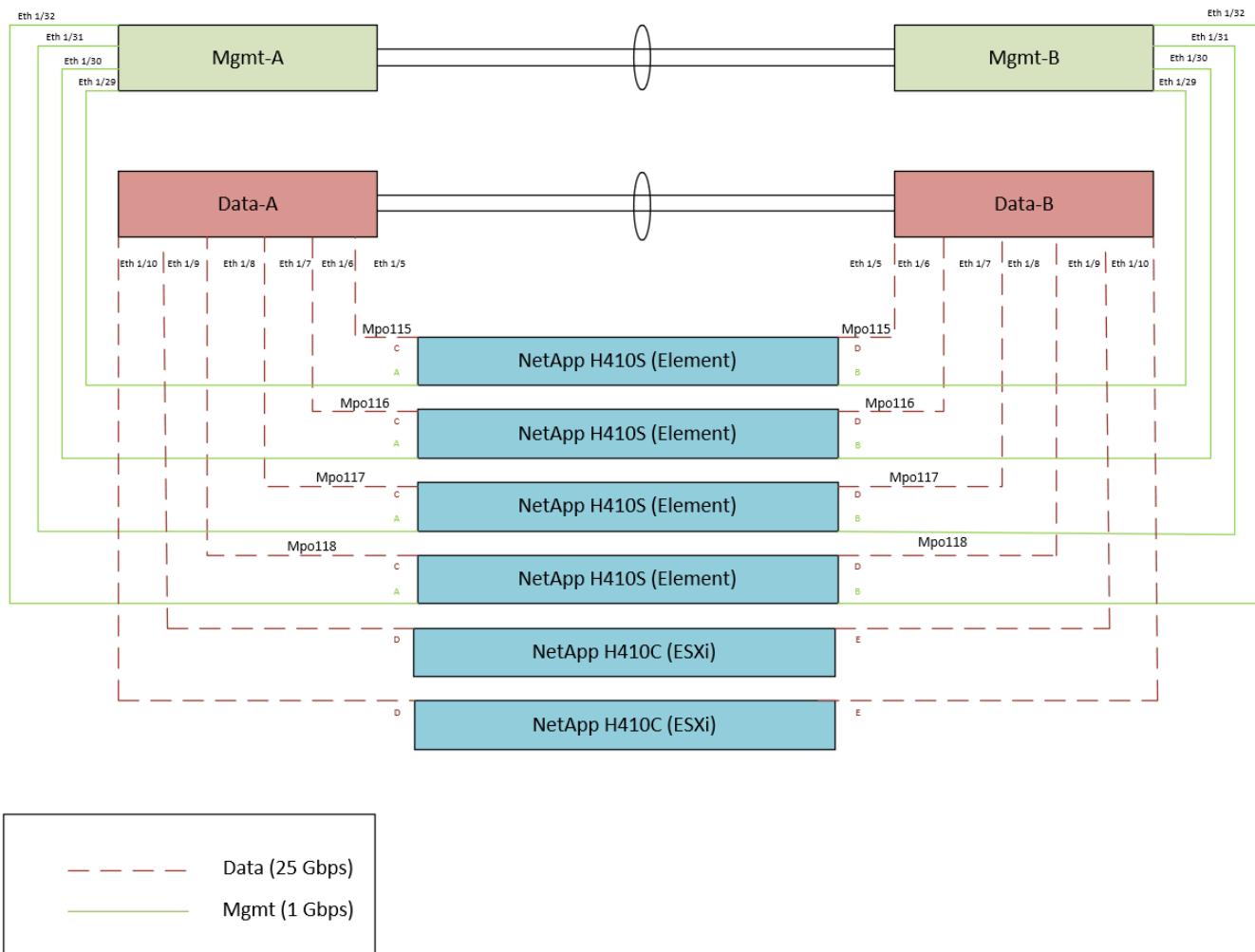


Figure 4. Network cabling reference diagram.

### VLAN Requirements

The solution is designed to logically separate network traffic for different purposes by using Virtual Local Area Networks (VLANs). NetApp HCI requires a minimum of three network segments. However, this configuration can be scaled to meet customer demands or to provide further isolation for specific network services. The following table lists the VLANs that are required to implement the solution, as

well as the specific VLAN IDs that are used later in the validated architecture deployment.

VLANs	Purpose	VLAN Used
Out-of-band management	Management for HCI nodes	16
In-band management	Management for HCI nodes and infrastructure virtual guests	3480
Storage Network	Storage network for NetApp Element	3481
vMotion network	Network for VMware vMotion	3482
VM network	Network for virtual guests	1172

## Network Infrastructure Support Resources

The following infrastructure should be in place prior to the deployment of the Anthos on NetApp HCI solution:

- A DHCP server providing addresses for both the in-band management network and the VM network. The DHCP pool must be large enough to support at least 10 VMs for an initial deployment and should be scaled as necessary.
- At least one DNS server providing full host-name resolution that is accessible from the in-band management network and the VM network.
- At least one NTP server that is accessible from the in-band management network and the VM network.
- Outbound internet connectivity for both the in-band management network and the VM network.

## Best Practices

### Install a Second F5 Big-IP Virtual Edition Appliance

In a production environment, it is a best practice to avoid single points of failure in your environment. For this validation, a single F5 BIG-IP Virtual Edition Load Balancer appliance was used to validate connectivity to the control plane and the ingress VIP addresses for the Anthos on VMware clusters. While this works fine for a simple validation, loss of communication with the control plane VIP for a cluster can make a cluster inaccessible or unable to be managed from the admin workstation or the Google Cloud console. F5 BIG-IP Virtual Edition supports application-based HA to make sure disruptions do not happen. Although this issue is mentioned briefly, setup procedures for this functionality are not described in detail in this document. However, NetApp recommends investigating this feature further before deploying the NetApp HCI for Anthos solution into production.

### Enable VMware vSphere DRS and Configure Anti-Affinity Rules

VMware vSphere provides a feature that makes sure that no single node in the cluster runs low on physical resources available to virtual guests. The Distributed Resource Scheduler (DRS) can be

configured on vSphere clusters consisting of at least three ESXi nodes. The NetApp HCI minimum configuration described in this deployment guide consists of two compute nodes and is unable to make use of this feature. As a result of this limitation, we were also forced to disable anti-affinity rules for the Anthos on VMware clusters that we deployed.

Anti-affinity rules ensure all masters or all workers for a specific user cluster run on different nodes, so that a single node failure cannot disable an entire user cluster or the pods that it is hosting. As the NetApp HCI system is both easily and rapidly scalable, and considering the minimum deployment described in this validation has two open chassis slots for immediate expansion of HCI 410C nodes, NetApp suggests adding additional compute nodes into the empty chassis slots prior to deploying the solution into production, and enabling DRS with Anti-Affinity rules.

#### **Leverage SnapMirror to Copy Data Remotely for Disaster Recovery**

NetApp Element storage systems can use NetApp SnapMirror technology to replicate storage volumes to systems running the NetApp ONTAP system, including AFF, FAS, and Cloud Volumes ONTAP. You can set up regularly scheduled SnapMirror operations to back up the VMware datastores and restore from a remote site in the event of a disaster. It is also possible to use SnapMirror to back up or migrate the persistent volumes provisioned by Trident and reattach them to Kubernetes clusters deployed in other environments and in the cloud.

[Next: Hardware and Software Requirements](#)

## **Hardware and Software Requirements**

This section describes the hardware and software requirements for the NetApp HCI and Anthos solution.

### **Hardware Requirements**

The following table lists the minimum number of hardware components that are required to implement the solution. The hardware components that are used in specific implementations of the solution might vary based on customer requirements.

<b>Hardware</b>	<b>Model</b>	<b>Quantity</b>
NetApp HCI compute nodes	NetApp H410C	2
NetApp HCI storage nodes	NetApp H410S	4
Data switches	Cisco Nexus 3048	2
Management switches	Mellanox NS2010	2

### **Software Requirements**

The following table lists the software components that are required to implement the solution. The software components that are used in any implementation of the solution might vary based on

customer requirements.

Software	Purpose	Version
NetApp HCI	Infrastructure (compute/storage)	1.6P1
VMware vSphere	Virtualization	6.5U2
Anthos on VMware	Container orchestration	1.1
F5 Big-IP Virtual Edition	Load balancing	15.0.1
HashiCorp Terraform	Automation and provisioning	0.12.12
NetApp Trident	Storage management	19.10

[Next: Workflow Summary](#)

## Deployment Steps

This section provides detailed protocols for implementing the NetApp HCI solution for Anthos.

This deployment is divided into the following high-level tasks:

1. Configure management switches
2. Configure data switches
3. Deploy NetApp HCI with the NetApp Deployment Engine
4. Configure the vCenter Server
5. Deploy and configure the F5 Big-IP Virtual Edition appliance
6. Complete Anthos prerequisites
7. Deploy the Anthos admin workstation
8. Deploy the admin and the first user cluster
9. Deploy additional user clusters
10. Enable access to the cluster with the GKE console
11. Install and configure NetApp Trident storage provisioner

[Next: Configure Management Switches](#)

### 1. Configure Management Switches

Cisco Nexus 3048 switches are used in this deployment procedure to provide 1Gbps connectivity for in and out-of-band management of the compute and storage nodes. These steps begin after the switches have been racked, powered, and put through the initial setup process. To configure the switches to provide management connectivity to the infrastructure, complete the following steps:

## **Enable Advanced Features for Cisco Nexus**

Run the following commands on each Cisco Nexus 3048 switch to configure advanced features:

1. Enter configuration mode.

```
Switch-01# configure terminal
```

2. Enable VLAN functionality.

```
Switch-01(config)# feature interface-vlan
```

3. Enable LACP.

```
Switch-01(config)# feature lacp
```

4. Enable virtual port channels (vPCs).

```
Switch-01(config)# feature vpc
```

5. Set the global port-channel load-balancing configuration.

```
Switch-01(config)# port-channel load-balance src-dst ip-l4port
```

6. Perform the global spanning-tree configuration.

```
Switch-01(config)# spanning-tree port type network default
Switch-01(config)# spanning-tree port type edge bpduguard default
```

## **Configure Ports on the Switch for In-Band Management**

1. Run the following commands to create VLANs for management purposes.

```
Switch-01(config)# vlan 2
Switch-01(config-vlan)# Name Native_VLAN
Switch-01(config-vlan)# vlan 16
Switch-01(config-vlan)# Name OOB_Network
Switch-01(config-vlan)# vlan 3480
Switch-01(config-vlan)# Name MGMT_Network
Switch-01(config-vlan)# exit
```

2. Configure the ports ETH1/29-32 as VLAN trunk ports that connect to management interfaces on each HCI storage node.

```
Switch-01(config)# int eth 1/29
Switch-01(config-if)# description HCI-STG-01 PortA
Switch-01(config-if)# switchport mode trunk
Switch-01(config-if)# switchport trunk native vlan 2
Switch-01(config-if)# switchport trunk allowed vlan 3480
Switch-01(config-if)# spanning tree port type edge trunk
Switch-01(config-if)# int eth 1/30
Switch-01(config-if)# description HCI-STG-02 PortA
Switch-01(config-if)# switchport mode trunk
Switch-01(config-if)# switchport trunk native vlan 2
Switch-01(config-if)# switchport trunk allowed vlan 3480
Switch-01(config-if)# spanning tree port type edge trunk
Switch-01(config-if)# int eth 1/31
Switch-01(config-if)# description HCI-STG-03 PortA
Switch-01(config-if)# switchport mode trunk
Switch-01(config-if)# switchport trunk native vlan 2
Switch-01(config-if)# switchport trunk allowed vlan 3480
Switch-01(config-if)# spanning tree port type edge trunk
Switch-01(config-if)# int eth 1/32
Switch-01(config-if)# description HCI-STG-04 PortA
Switch-01(config-if)# switchport mode trunk
Switch-01(config-if)# switchport trunk native vlan 2
Switch-01(config-if)# switchport trunk allowed vlan 3480
Switch-01(config-if)# spanning tree port type edge trunk
Switch-01(config-if)# exit
```

#### Configure Ports on the Switch for Out-of-Band Management

1. Run the following commands to configure the ports for cabling the IPMI interfaces on each HCI node.

```
Switch-01(config)# int eth 1/13
Switch-01(config-if)# description HCI-CMP-01 IPMI
Switch-01(config-if)# switchport mode access
Switch-01(config-if)# switchport access vlan 16
Switch-01(config-if)# spanning-tree port type edge
Switch-01(config-if)# int eth 1/14
Switch-01(config-if)# description HCI-STG-01 IPMI
Switch-01(config-if)# switchport mode access
Switch-01(config-if)# switchport access vlan 16
Switch-01(config-if)# spanning-tree port type edge
Switch-01(config-if)# int eth 1/15
Switch-01(config-if)# description HCI-STG-03 IPMI
Switch-01(config-if)# switchport mode access
Switch-01(config-if)# switchport access vlan 16
Switch-01(config-if)# spanning-tree port type edge
Switch-01(config-if)# exit
```



In the validated configuration, we cabled odd-node IPMI interfaces to Switch-01, and even-node IPMI interfaces to Switch-02.

#### Create a vPC Domain to Ensure Fault Tolerance

1. Activate the ports used for the vPC peer-link between the two switches.

```
Switch-01(config)# int eth 1/1
Switch-01(config-if)# description vPC peer-link Switch-02 1/1
Switch-01(config-if)# int eth 1/2
Switch-01(config-if)# description vPC peer-link Switch-02 1/2
Switch-01(config-if)# exit
```

2. Perform the vPC global configuration.

```
Switch-01(config)# vpc domain 1
Switch-01(config-vpc-domain)# role priority 10
Switch-01(config-vpc-domain)# peer-keepalive destination <switch-02_mgmt_address>
source <switch-01_mgmt_address> vrf managment
Switch-01(config-vpc-domain)# peer-gateway
Switch-01(config-vpc-domain)# auto recovery
Switch-01(config-vpc-domain)# ip arp synchronize
Switch-01(config-vpc-domain)# int eth 1/1-2
Switch-01(config-vpc-domain)# channel-group 10 mode active
Switch-01(config-vpc-domain)# int Po10
Switch-01(config-if)# description vPC peer-link
Switch-01(config-if)# switchport mode trunk
Switch-01(config-if)# switchport trunk native vlan 2
Switch-01(config-if)# switchport trunk allowed vlan 16,3480
Switch-01(config-if)# spanning-tree port type network
Switch-01(config-if)# vpc peer-link
Switch-01(config-if)# exit
```

[Next: Configure Data Switches](#)

## 2. Configure Data Switches

Mellanox SN2010 switches provide 25Gbps connectivity for the data plane of the compute and storage nodes.

To configure the switches to provide data connectivity to the infrastructure, complete the following steps:

### Create MLAG Cluster to Ensure Fault Tolerance

1. Run the following commands on each Mellanox SN210 switch for general configuration:

1. Enter configuration mode.

```
Switch-01 enable
Switch-01 configure terminal
```

2. Enable the LACP required for the Inter-Peer Link (IPL).

```
Switch-01 (config) # lacp
```

3. Enable the Link Layer Discovery Protocol (LLDP).

```
Switch-01 (config) # lldp
```

4. Enable IP routing.

```
Switch-01 (config) # ip routing
```

5. Enable the MLAG protocol

```
Switch-01 (config) # protocol mlag
```

6. Enable global QoS.

```
Switch-01 (config) # dcb priority-flow-control enable force
```

2. For MLAG to function, the switches must be made peers to each other through an IPL. This should consist of two or more physical links for redundancy. The MTU for the IPL is set for jumbo frames (9216), and all VLANs are enabled by default. Run the following commands on each switch in the domain:

1. Create port channel 10 for the IPL.

```
Switch-01 (config) # interface port-channel 10
Switch-01 (config interface port-channel 10) # description IPL
Switch-01 (config interface port-channel 10) # exit
```

2. Add interfaces ETH 1/20 and 1/22 to the port channel.

```
Switch-01 (config) # interface ethernet 1/20 channel-group 10 mode active
Switch-01 (config) # interface ethernet 1/20 description ISL-SWB_01
Switch-01 (config) # interface ethernet 1/22 channel-group 10 mode active
Switch-01 (config) # interface ethernet 1/22 description ISL-SWB_02
```

3. Create a VLAN outside of the standard range dedicated to IPL traffic.

```
Switch-01 (config) # vlan 4000
Switch-01 (config vlan 4000) # name IPL VLAN
Switch-01 (config vlan 4000) # exit
```

4. Define the port channel as the IPL.

```
Switch-01 (config) # interface port-channel 10 ipl 1
Switch-01 (config) # interface port-channel 10 dcb priority-flow-control mode on
force
```

5. Set an IP for each IPL member (non-routable; it is not advertised outside of the switch).

```
Switch-01 (config) # interface vlan 4000
Switch-01 (config vlan 4000) # ip address 10.0.0.1 255.255.255.0
Switch-01 (config vlan 4000) # ipl 1 peer-address 10.0.0.2
Switch-01 (config vlan 4000) # exit
```

3. Create a unique MLAG domain name for the two switches and assign a MLAG virtual IP (VIP). This IP is used for keep-alive heartbeat messages between the two switches. Run these commands on each switch in the domain:

1. Create the MLAG domain and set the IP address and subnet.

```
Switch-01 (config) # mlag-vip MLAG-VIP-DOM ip a.b.c.d /24 force
```

2. Create a virtual MAC address for the system MLAG.

```
Switch-01 (config) # mlag system-mac AA:BB:CC:DD:EE:FF
```

3. Configure the MLAG domain so that it is active globally.

```
Switch-01 (config) # no mlag shutdown
```



The IP used for the MLAG VIP must be in the same subnet as the switch management network (mgmt0).



The MAC address used can be any unicast MAC address and must be set to the same value on both switches in the MLAG domain.

#### Configure Ports to Connect to Storage and Compute Hosts

1. Create each of the VLANs needed to support the services for NetApp HCI. Run these commands on each switch in the domain:

1. Create VLANs.

```
Switch-01 (config) # vlan 1172
Switch-01 (config vlan 1172) exit
Switch-01 (config) # vlan 3480-3482
Switch-01 (config vlan 3480-3482) exit
```

2. Create names for each VLAN for easier accounting.

```
Switch-01 (config) # vlan 1172 name "VM_Network"
Switch-01 (config) # vlan 3480 name "MGMT_Network"
Switch-01 (config) # vlan 3481 name "Storage_Network"
Switch-01 (config) # vlan 3482 name "vMotion_Network"
+
```

2. Create Hybrid VLAN ports on ports ETH1/9-10 so that you can tag the appropriate VLANs for the NetApp HCI compute nodes.

1. Select the ports you want to work with.

```
Switch-01 (config) # interface ethernet 1/9-1/10
```

2. Set the MTU for each port.

```
Switch-01 (config interface ethernet 1/9-1/10) # mtu 9216 force
```

3. Modify spanning-tree settings for each port.

```
Switch-01 (config interface ethernet 1/9-1/10) # spanning-tree bpdufilter enable
Switch-01 (config interface ethernet 1/9-1/10) # spanning-tree port type edge
Switch-01 (config interface ethernet 1/9-1/10) # spanning-tree bpduguard enable
```

4. Set the switchport mode to hybrid.

```
Switch-01 (config interface ethernet 1/9-1/10 ) # switchport mode hybrid
Switch-01 (config interface ethernet 1/9-1/10 ) # exit
```

5. Create descriptions for each port being modified.

```
Switch-01 (config) # interface ethernet 1/9 description HCI-CMP-01 PortD
Switch-01 (config) # interface ethernet 1/10 description HCI-CMP-02 PortD
```

6. Tag the appropriate VLANs for the NetApp HCI environment.

```
Switch-01 (config) # interface ethernet 1/9 switchport hybrid allowed-vlan add 1172
Switch-01 (config) # interface ethernet 1/9 switchport hybrid allowed-vlan add
3480-3482
Switch-01 (config) # interface ethernet 1/10 switchport hybrid allowed-vlan add
1172
Switch-01 (config) # interface ethernet 1/10 switchport hybrid allowed-vlan add
3480-3482
```

3. Create MLAG interfaces and hybrid VLAN ports on ports ETH1/5-8 so that you can distribute connectivity between the switches and tag the appropriate VLANs for the NetApp HCI storage nodes.

1. Select the ports that you want to work with.

```
Switch-01 (config) # interface ethernet 1/5-1/8
```

2. Set the MTU for each port.

```
Switch-01 (config interface ethernet 1/5-1/8) # mtu 9216 force
```

3. Modify spanning tree settings for each port.

```
Switch-01 (config interface ethernet 1/5-1/8) # spanning-tree bpdufilter enable
Switch-01 (config interface ethernet 1/5-1/8) # spanning-tree port type edge
Switch-01 (config interface ethernet 1/5-1/8) # spanning-tree bpduguard enable
```

4. Set the switchport mode to hybrid.

```
Switch-01 (config interface ethernet 1/5-1/8 ) # switchport mode hybrid
Switch-01 (config interface ethernet 1/5-1/8 ) # exit
```

5. Create descriptions for each port being modified.

```
Switch-01 (config) # interface ethernet 1/5 description HCI-STG-01 PortD
Switch-01 (config) # interface ethernet 1/6 description HCI-STG-02 PortD
Switch-01 (config) # interface ethernet 1/7 description HCI-STG-03 PortD
Switch-01 (config) # interface ethernet 1/8 description HCI-STG-04 PortD
```

6. Create and configure the MLAG port channels.

```
Switch-01 (config) # interface mlag-port-channel 115-118
Switch-01 (config interface mlag-port-channel 115-118) # exit
Switch-01 (config) # interface mlag-port-channel 115-118 no shutdown
Switch-01 (config) # interface mlag-port-channel 115-118 mtu 9216 force
Switch-01 (config) # interface mlag-port-channel 115-118 lacp-individual enable
force
Switch-01 (config) # interface ethernet 1/5-1/8 lacp port-priority 10
Switch-01 (config) # interface ethernet 1/5-1/8 lacp rate fast
Switch-01 (config) # interface ethernet 1/5 mlag-channel-group 115 mode active
Switch-01 (config) # interface ethernet 1/6 mlag-channel-group 116 mode active
Switch-01 (config) # interface ethernet 1/7 mlag-channel-group 117 mode active
Switch-01 (config) # interface ethernet 1/8 mlag-channel-group 118 mode active
```

7. Tag the appropriate VLANs for the storage environment.

```
Switch-01 (config) # interface mlag-port-channel 115-118 switchport mode hybrid
Switch-01 (config) # interface mlag-port-channel 115 switchport hybrid allowed-vlan
add 1172 Switch-01 (config) # interface mlag-port-channel 116 switchport hybrid
allowed-vlan add 1172
Switch-01 (config) # interface mlag-port-channel 117 switchport hybrid allowed-vlan
add 1172
Switch-01 (config) # interface mlag-port-channel 118 switchport hybrid allowed-vlan
add 1172
Switch-01 (config) # interface mlag-port-channel 115 switchport hybrid allowed-vlan
add 3481
Switch-01 (config) # interface mlag-port-channel 116 switchport hybrid allowed-vlan
add 3481
Switch-01 (config) # interface mlag-port-channel 117 switchport hybrid allowed-vlan
add 3481
Switch-01 (config) # interface mlag-port-channel 118 switchport hybrid allowed-vlan
add 3481
```



The configurations in this section must also be run on the second switch in the MLAG domain. NetApp recommends that the descriptions for each port are updated to reflect the device ports that are being cabled and configured on the other switch.

#### Create Uplink Ports for the Switches

1. Create an MLAG interface to provide uplinks to both Mellanox SN2010 switches from the core network.

```
Switch-01 (config) # interface mlag port-channel 101
Switch-01 (config interface mlag port-channel) # description Uplink CORE-SWITCH port
PORT
Switch-01 (config interface mlag port-channel) # exit
```

2. Configure the MLAG members.

```
Switch-01 (config) # interface ethernet 1/18 description Uplink to CORE-SWITCH port
PORT
Switch-01 (config) # interface ethernet 1/18 speed 10000 force
Switch-01 (config) # interface mlag-port-channel 101 mtu 9216 force
Switch-01 (config) # interface ethernet 1/18 mlag-channel-group 101 mode active
```

3. Set the switchport mode to hybrid and allow all VLANs from the core uplink switches.

```
Switch-01 (config) # interface mlag-port-channel switchport mode hybrid
Switch-01 (config) # interface mlag-port-channel switchport hybrid allowed-vlan all
```

4. Verify that the MLAG interface is up.

```
Switch-01 (config) # interface mlag-port-channel 101 no shutdown
Switch-01 (config) # exit
```

[Next: Deploy NetApp HCI with the NetApp Deployment Engine](#)

### 3. Deploy NetApp HCI with the NetApp Deployment Engine

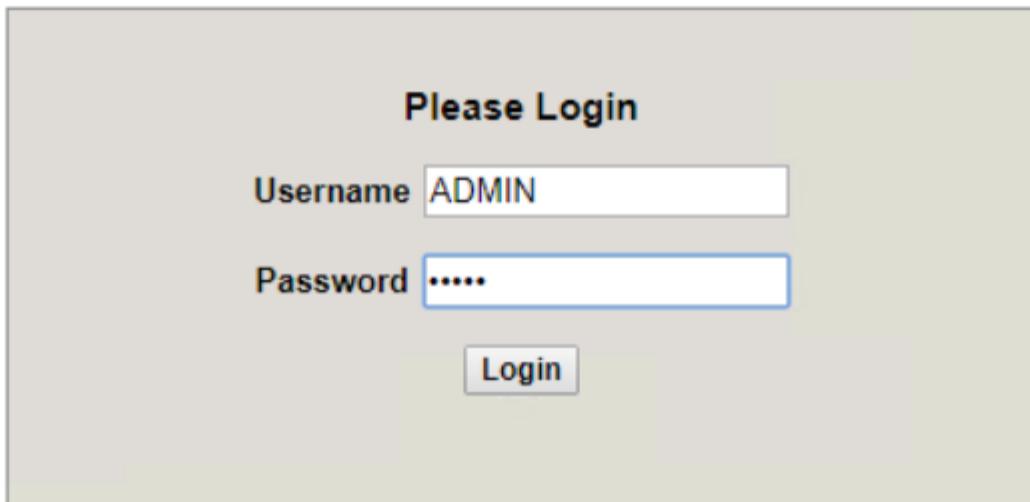
NDE delivers a simple and streamlined deployment experience for the NetApp HCI solution.

These steps begin after the nodes have been racked, and cabled, and the IPMI port has been configured on each node using the console.

A detailed guide to using NDE 1.6 to deploy your NetApp HCI system can be found [here](#).

To Deploy the NetApp HCI solution using NDE, complete the following steps:

1. Access the out-of-band management console for one of the storage nodes in the cluster and log in with the default credentials ADMIN/ADMIN.



The image shows a login interface for a NetApp system. At the top center, it says "Please Login". Below that, there is a "Username" field containing "ADMIN". Below the username field is a "Password" field containing "\*\*\*\*\*". At the bottom center is a "Login" button.

2. Click the Remote Console Preview image in the center of the screen to download a JNLP file launched by Java Web Start, which launches an interactive console to the system.
3. With the virtual console launched, a user can log in to the HCI Storage node, using the ADMIN/ADMIN username and password combination.
4. The Bond1G interface must have an IP, a netmask, and a gateway set statically; its VLAN set to 3480; and DNS servers defined for the environment.

```
Bond10G
      Method          : static

      Link Speed     : 50000

      IPv4 Address   :

      IPv4 Subnet Mask  :
      --->

      IPv4 Gateway Address :
      --->

      MTU            : 9000
      --->

      Bond Mode       : LACP  [ActivePassive, ALB, LACP]
      --->

      LACP Rate       : Fast   [Fast, Slow]
      --->

      Status          : UpAndRunning  [Down, Up, UpAndRunning]
      --->

      Virtual Network Tag :
      --->

      Routes          : Number of routes: 0.
      --->
```



Select an IP that is within the subnet you intend to use for in-band management, but not an IP you would like to use in production. NDE reconfigures the node with a production IP after initial access.



This task must only be performed on the first storage node. Afterward, the other nodes in the infrastructure are discovered by the Automatic Private IP Address (APIPA) addresses assigned to each storage interface when left unconfigured.

5. The Bond 10G interface must have its MTU setting changed to enable jumbo frames and its bond mode changed to LACP.

```

Bond10G
      Method          : static

      Link Speed     : 50000

      IPv4 Address    :

      IPv4 Subnet Mask   :
      --->

      IPv4 Gateway Address  :
      --->

      MTU             : 9000
      --->

      Bond Mode       : LACP  [ActivePassive, ALB, LACP]
      --->

      LACP Rate        : Fast   [Fast, Slow]
      --->

      Status           : UpAndRunning  [Down, Up, UpAndRunning]
      --->

      Virtual Network Tag  :
      --->

      Routes          : Number of routes: 0.
      --->

```



Configure each of the four storage nodes in the NetApp HCI solution this way. The NDE process is then able to discover all the nodes in the solution and configure them. You do not need to modify the Bond10g interfaces on the two compute nodes.

6. After completion, open a web browser and visit the IP address you configured for the management port to start the NetApp HCI configuration with NDE.
7. On the Welcome to NetApp HCI page, click the Get Started button.
8. Check each associated box on the Prerequisites page and click Continue.
9. The next page presents End User Licenses for NetApp HCI and VMware vSphere. If you accept the terms, click I Accept at the end of each agreement and then click Continue.
10. Click Configure a New vSphere Deployment, select vSphere 6.5U2, and enter the Fully Qualified Domain Name (FQDN) of your vCenter Server. Then click Continue.

# vSphere Configuration

You may elect to configure a new vSphere deployment or to join an existing vSphere deployment.

- Configure a new vSphere deployment
- Configure Using vSphere Version 6.7 Update 1
- Configure Using vSphere Version 6.5 Update 2
- Join and extend an existing vSphere deployment

If you have set up a DNS record for your new vCenter server, then configure your server using its fully qualified domain name and DNS server IP address:

- Configure Using a Fully Qualified Domain Name Best Practice!

**vCenter Server Fully Qualified Domain Name**

anthos-vc.cie.netapp.com



**Note:** The domain name must resolve to an unused IP address.

**DNS Server IP Address**

10.61.184.251



If you have not set up a DNS record for your new vCenter server, you may configure using an IP address that we define:

- Configure Using an IP Address ?

**Note:** Once defined, the IP address cannot be changed.

**Back**

**Continue**

11. NDE asks for the credentials to be used in the environment. This is used for VMware vSphere, the NetApp Element storage cluster, and the NetApp Mnode, which provides management functionality for the cluster. When you are finished, click Continue.

# Credentials

Define the user name and password that will be used for the storage cluster, vCenter, and the management node.

User Name

admin



Password

.....



**Password must contain:**

- ✓ At least 8 characters
- ✓ No more than 20 characters
- ✓ 1 uppercase letter that is not the first character
- ✓ 1 lowercase letter
- ✓ 1 of the following special characters: !@#\$
- ✓ Allowed characters: A-Z a-z 0-9 !@#\$
- ✓ 1 number that is not the last character

Re-enter Password

.....



[Back](#)

[Continue](#)

12. NDE then prompts for the network topology used to cable the NetApp HCI environment. The validated solution in this document has been deployed using the 2 Cable Option for the compute nodes, and the 4 Cable Option for the storage nodes. Click Continue.

# Network Topology

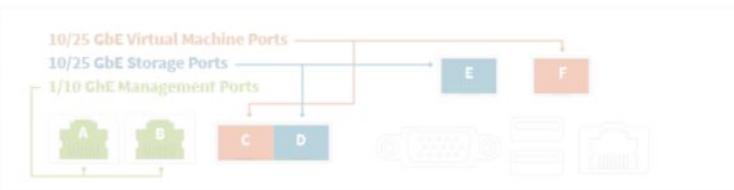
Select a compute node topology and a storage node topology appropriate for your hardware installation.

## Compute Node Topology

### 6 Cable Option

The 6 cable option provides dedicated ports for management (2 x 1/10 GbE), virtual machines (2 x 10/25 GbE) and storage (2 x 10/25 GbE).

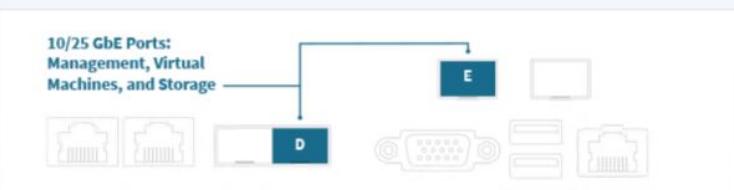
Use vSphere Distributed Switch? [?](#)



(H300E,H410C,H500E,H700E)

### 2 Cable Option

The 2 cable option provides shared management with ports for virtual machines and storage (2 x 10/25 GbE). The 2 cable option uses vSphere Distributed Switch. [?](#)

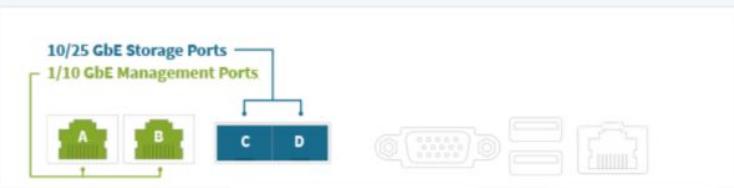


(H300E,H410C,H500E,H700E)

## Storage Node Topology

### 4 Cable Option

The 4 cable option provides dedicated ports for management (2 x 1/10 GbE) and storage (2 x 10/25 GbE).



(H300S,H410S,H500S,H700S)

[Back](#)

[Continue](#)

13. The next page presented by NDE is the inventory of the environment as discovered by the APIPA addressed on the storage network. The storage node that is currently running NDE is already selected with a green check mark. Select the corresponding boxes to add additional nodes to the NetApp HCI environment. Click Continue.

## Inventory

Verify the available nodes and select **at least 2 compute nodes and 4 storage nodes** to include in your installation.

[Refresh Inventory](#)

### Compute Nodes

	Serial Number	Chassis Serial Number / Slot	Node Type	Software Version	Physical CPU Cores	Memory	1 GbE Ports	10 GbE Ports
<input checked="" type="checkbox"/>	HM17CS002729	002170990158 / B	H410C	1.6	8	384 GB	0 of 2 detected	2 of 4 detected
<input checked="" type="checkbox"/>	HM181S002024	002170990158 / A	H410C	1.6	8	384 GB	0 of 2 detected	2 of 4 detected

1 - 2 of 2 results

◀ ▶ 1 ▶ ▷

20 ▾

2 compute nodes selected

### Storage Nodes

	Serial Number	Chassis Serial Number / Slot	Node Type	Raw Capacity	Element Version	Drive Count	1 GbE Ports	10 GbE Ports
<input checked="" type="checkbox"/>	221814003506	221814003436 / C	H500S	5.76 TB	11.3.1.5	6 of 6 detected	2 of 2 detected	2 of 2 detected
<input checked="" type="checkbox"/>	221818004613	221814003436 / D	H500S	5.76 TB	11.3.1.5	6 of 6 detected	2 of 2 detected	2 of 2 detected
<input checked="" type="checkbox"/> ?	221826005865	002170990158 / C	H500S	5.76 TB	11.3.1.5	6 of 6 detected	2 of 2 detected	2 of 2 detected
<input checked="" type="checkbox"/>	221826005866	002170990158 / D	H500S	5.76 TB	11.3.1.5	6 of 6 detected	2 of 2 detected	2 of 2 detected

1 - 4 of 4 results

◀ ▶ 1 ▶ ▷

20 ▾

4 storage nodes selected

[Back](#)

[Continue](#)



If there are any nodes missing from the inventory screen, wait a few minutes and click Refresh Inventory. If the node still fails to appear, additional investigation of environment networking may be required.

14. You must next configure the permanent network settings for the NetApp HCI deployment. The first page configures infrastructure services (DNS and NTP), vCenter networking, and Mnode networking.

# Network Settings

Provide the network settings that will be used for your installation.

Live network validation is: **On**

## Infrastructure Services

DNS Server IP Address 1	DNS Server IP Address 2 (Optional)
<input type="text" value="10.61.184.251"/>	<input type="text" value="10.61.184.252"/>
NTP Server Address 1	NTP Server Address 2 (Optional)
<input type="text" value="10.61.184.251"/>	<input type="text" value="10.61.184.252"/>

To save time, launch the easy form to enter fewer network settings.

## vCenter Networking

VLAN ID	Subnet	Default Gateway	FQDN	IP Address
<input type="text" value="3480"/>	<input type="text" value="172.21.224.0/24"/>	<input type="text" value="172.21.224.1"/>	anthos-vc.cie.netapp.com	<input type="text" value="172.21.224.10"/>

## Management Node Networking

Management Network		iSCSI Network
VLAN ID	Subnet	VLAN ID
<input type="text" value="3480"/>	<input type="text" value="172.21.224.0/24"/>	<input type="text" value="3481"/>
Subnet	Default Gateway	Subnet
<input type="text" value="172.21.224.0/24"/>	<input type="text" value="172.21.224.1"/>	<input type="text" value="172.21.225.0/24"/>
Default Gateway	Management IP Address	Storage (iSCSI) IP Address
<input type="text" value="anthos-mnode"/>	<input type="text" value="172.21.224.50"/>	<input type="text" value="172.21.225.50"/>

15. The next page allows you to configure each node in the environment. For the compute nodes, it allows you to configure the host name, management network, vMotion network, and storage network. For the storage nodes, name the storage cluster and configure the management and storage networks being used for each node. Click Continue.

## Compute Node Networking

Management Network		vMotion Network		iSCSI A Network		iSCSI B Network	
VLAN ID	3480	VLAN ID	3482	VLAN ID	3481	VLAN ID	3481
Serial Number	Hostname	Management IP Address	vMotion IP Address	iSCSI A - IP Address	iSCSI B - IP Address		
HM17CS002729	Anthos-ESXi-01	172.21.224.11	172.21.226.11	172.21.225.11	172.21.225.111	172.21.225.111	172.21.225.111
HM181S002024	Anthos-ESXi-02	172.21.224.12	172.21.226.12	172.21.225.12	172.21.225.112	172.21.225.112	172.21.225.112

## Storage Node Networking

### Storage Cluster Name

Note: The storage cluster name cannot be changed after deployment.

Management Network		iSCSI Network	
VLAN ID	3480	VLAN ID	3481
Serial Number	Hostname	Management IP Address	Storage (iSCSI) IP Address
221814003506	Anthos-Store-01	172.21.224.21	172.21.225.21
221818004613	Anthos-Store-02	172.21.224.22	172.21.225.22
221826005865	Anthos-Store-03	172.21.224.23	172.21.225.23
221826005866	Anthos-Store-04	172.21.224.24	172.21.225.24

[Back](#)

Live network validation is:  On [?](#)

[Continue](#)

- On the next page, review all the settings that have been defined for the environment by expanding each section, and, if necessary, click Edit to make corrections. There is also a check box on this page that enables or disables the Mnode from sending real-time health and diagnostics information to NetApp Active IQ®. If all the information is correct, click Start Deployment.



If you want to enable Active IQ, verify that your management network can reach the internet. If NDE is unable to reach Active IQ, the deployment can fail.

- A summary page appears along with a progress bar for each component of the NetApp HCI solution, as well as the overall solution. When complete, you are presented with an option to launch the vSphere client and begin working with your environment.

Your setup is complete.

[Launch vSphere Client](#)

Configure Network	Complete	✓
Set up NetApp Cluster	Complete	✓
Set up ESXi	Complete	✓
Set up vCenter	Complete	✓
Configure Management Node	Complete	✓
Finalize Configuration	Complete	✓

Overall Progress

100%

 [Export all setup information to CSV file](#)

[Next: Configure the vCenter Server](#)

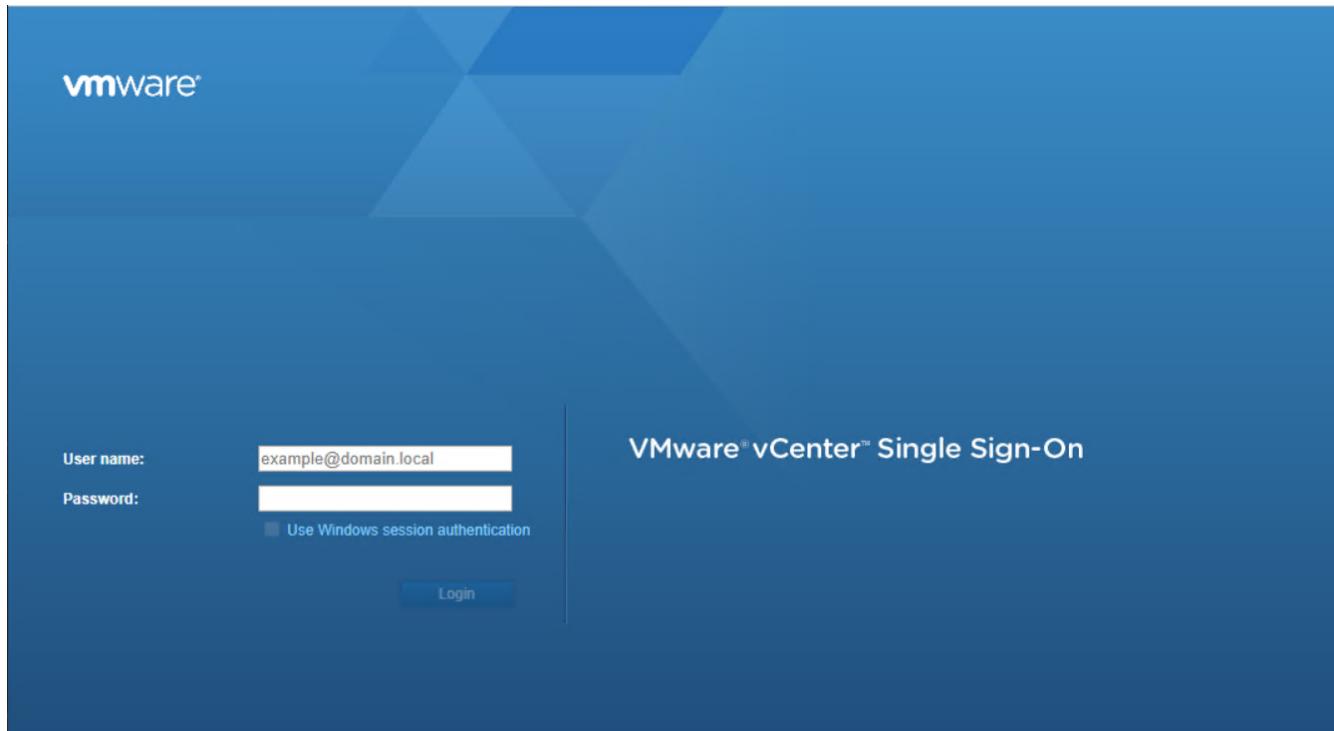
#### 4. Configure the vCenter Server

NDE deploys the solution with vCenter server and integrates the solution with the Element cluster by provisioning the Mnode VM and installing the NetApp Element Plug-in for vCenter.

After deployment, you must make a few modifications to the environment, including the creation of additional vDS portgroups, datastores, and resource groups for the deployment of the Anthos on VMware solution.

Complete the following steps to configure your vCenter Server for use:

1. Log into the VMware vCenter server using the [Administrator@vsphere.local](#) account and the password chosen for the admin user during NDE configuration.



2. Right-click **NetApp-HCI-Cluster-01** created by NDE and select the option to create a new resource pool. Name this pool **Infrastructure-Resource-Pool** and accept the defaults by clicking OK. This resource pool is used in a later configuration step.

# New Resource Pool

NetApp-HCI-Cluster-01

X

Name	Infrastructure Resource		
<b>CPU</b>			
Shares	Normal	4000	
Reservation	0	MHz	▼
Max reservation: 54,128 MHz			
Reservation Type	<input checked="" type="checkbox"/> Expandable		
Limit	Unlimited	MHz	▼
Max limit: 58,128 MHz			
<b>Memory</b>			
Shares	Normal	163840	
Reservation	0	MB	▼
Max reservation: 751,064 MB			
Reservation Type	<input checked="" type="checkbox"/> Expandable		
Limit	Unlimited	MB	▼
Max limit: 756,820 MB			

CANCEL

OK



The reservations in this resource pool can be modified based on the resources available in the environment. NetApp HCI is deployed as an all-in-one solution. Therefore, NetApp recommends reserving the resources necessary to provide availability for the infrastructure services by placing them into this resource pool and adjusting the resources appropriately. Infrastructure services include vCenter Server, NetApp Mnode, and F5 Big-IP Load Balancer.

3. Repeat this step to create another resource pool for VMs deployed by Anthos. Name this pool Anthos-Resource-Pool and click the OK button to accept the default values. Adjust the resource availability based on the specific environment in which you are deploying the solution. This resource pool is used in a later deployment step.
4. To configure Element volumes to be used as vSphere datastores, click the dropdown menu and select NetApp Element Management from the list.
5. A Getting Started screen appears with details about your Element cluster.



6. Click Management, and the vSphere client presents a list of datastores. Click Create Datastore to create one datastore to host VMs and another to host ISOs for future guest installs.
7. Next click the Network menu item in the left panel. This displays a screen with information about the vDS deployed by NDE.
8. Several virtual port groups are defined by the initial configuration. NetApp recommends leaving these alone to support the infrastructure, and additional port groups should be created for user-deployed virtual guests. Right-click the NetApp HCI VDS 01 vDS in the left panel, and then select Distributed Port Group followed by the New Distributed Port Group option from the expanded menu.
9. Create a new distributed port group called **Management\_Network**. Then click Next.
10. On the next screen, select the VLAN type as VLAN, and set the VLAN ID to 3480 for management purposes. Click Next, and, after reviewing the options on the summary page, click Next again to complete the creation of the distributed port group.
11. Repeat these steps to create distributed port groups for the **VM\_Network** (VLAN 1172), as well as any other networks that might be used in the NetApp HCI environment.

 Additional networks can be defined to segment any additional deployed VMs. Examples of this use could be for a dedicated HA network for additional F5 Big-IP appliances if provisioned. Such configurations are in addition to the environment deployed in this validated solution and are considered out of scope for this NVA document.

[Next: Deploy and Configure the F5 Big-IP Virtual Edition Appliance](#)

## 5. Deploy and Configure the F5 Big-IP Virtual Edition Appliance

Anthos enables native integration with F5 Big-IP load balancers to expose services from each pod to the world.

This solution makes use of the virtual appliance deployed in VMware vSphere as deployed by NDE. Networking for the F5 Big-IP virtual appliance can be configured in a two-armed or three-armed configuration based on your network environment. The deployment in this document is based on the two-armed configuration. Additional details on configuring the virtual appliance for use with Anthos can be found [here](#).

To deploy the F5 Big-IP Virtual Edition appliance, complete the following steps:

1. Download the virtual application Open Virtual Appliance (OVA) file from F5 [here](#).



To download the appliance, a user must register with F5. They provide a 30-day demo license for the Big-IP Virtual Edition Load Balancer. NetApp recommends a permanent 10Gbps license for the production deployment of an appliance.

2. Right-click the Infrastructure Resource Pool and select Deploy OVF Template. A wizard launches that allows you to select the OVA file that you just downloaded in Step 1. Click Next.

## Deploy OVF Template

### 1 Select an OVF template

- 2 Select a name and folder
- 3 Select a compute resource
- 4 Review details
- 5 Select storage
- 6 Ready to complete

#### Select an OVF template

Select an OVF template from remote URL or local file system

Enter a URL to download and install the OVF package from the Internet, or browse to a location accessible from your computer, such as a local hard drive, a network share, or a CD/DVD drive.

URL

http | https://remoteserver-address/filetodeploy.ovf | .ova

Local file

BIGIP-15.0.1-0....ALL-vmware.ova

3. Click Next to continue through each step and accept the default values for each screen presented until you reach the storage selection screen. Select the VM\_Datastore that was created earlier, and then click Next.
4. The next screen presented by the wizard allows you to customize the virtual networks for use in the environment. Select VM\_Network for the External field and select Management\_Network for the Management field. Internal and HA are used for advanced configurations for the F5 Big-IP appliance and are not configured. These parameters can be left alone, or they can be configured to connect to non-infrastructure, distributed port groups. Click Next.
5. Review the summary screen for the appliance, and, if all the information is correct, click Finish to start the deployment.
6. After the virtual appliance is deployed, right-click it and power it up. It should receive a DHCP address on the management network. The appliance is Linux-based, and it has VMware Tools deployed, so you can view the DHCP address it receives in the vSphere client.
7. Open a web browser and connect to the appliance at the IP address from the previous step. The

default login is admin/admin, and, after the first login, the appliance immediately prompts you to change the admin password. It then returns you to a screen where you must log in with the new credentials.



8. The first screen prompts the user to complete the Setup Utility. Begin the utility by clicking Next.
  9. The next screen prompts for activation of the license for the appliance. Click Activate to begin. When prompted on the next page, paste either the 30-day evaluation license key you received when you registered for the download or the permanent license you acquired when you purchased the appliance. Click Next.
- i

For the device to perform activation, the network defined on the management interface must be able to reach the internet.
10. On the next screen, the End User License Agreement (EULA) is presented. If the terms in the license are acceptable, click Accept.
  11. The next screen counts the elapsed time as it verifies the configuration changes that have been made so far. Click Continue to resume with the initial configuration.
  12. The Configuration Change window closes, and the Setup Utility displays the Resource Provisioning

menu. This window lists the features that are currently licensed and the current resource allocations for the virtual appliance and each running service.

13. Clicking the Platform menu option on the left enables additional modification of the platform. Modifications include setting the management IP address configured with DHCP, setting the host name and the time zone the appliance is installed in, and securing the appliance from SSH accessibility.
14. Next click the Network menu, which enables you to configure standard networking features. Click Next to begin the Standard Network Configuration wizard.
15. The first page of the wizard configures redundancy; leave the defaults and click Next. The next page enables you to configure an internal interface on the load balancer. Interface 1.1 maps to the vmnic labeled Internal in the OVF deployment wizard.

[Big-IP Configuration] | *big-IP\_config\_8.png*



The spaces in this page for Self IP Address, Netmask, and Floating IP address can be filled with a non-routable IP for use as a placeholder. They can also be filled with an internal network that has been configured as a distributed port group for virtual guests if you are deploying the three-armed configuration. They must be completed to continue with the wizard.

16. The next page enables you to configure an external network that is used to map services to the pods deployed in Kubernetes. Select a static IP from the VM\_Network range, the appropriate subnet mask, and a floating IP from that same range. Interface 1.2 maps to the vmnic labeled External in the OVF deployment wizard.

[Big-IP Configuration] | *big-IP\_config\_9.png*

17. On the next page, you can configure an internal-HA network if you are deploying multiple virtual appliances in the environment. To proceed, you must fill the Self-IP Address and the Netmask fields, and you must select interface 1.3 as the VLAN Interface, which maps to the HA network defined by the OVF template wizard.
18. The next page enables you to configure the NTP servers. Then click Next to continue to the DNS setup. The DNS servers and domain search list should already be populated by the DHCP server. Click Next to accept the defaults and continue.
19. For the remainder of the wizard, click Next to continue through the advanced peering setup, the configuration of which is beyond the scope of this document. Then click Finish to exit the wizard.
20. Create individual partitions for the Anthos admin cluster and each user cluster deployed in the environment. Click System in the menu on the left, navigate to Users, and click Partition List.

21. The displayed screen only shows the current common partition. Click Create on the right to create the first additional partition, and name it **Anthos-Admin**. Then click Repeat, and name the partition **Anthos-Cluster1**, and click the Repeat button again to name the next partition **Anthos-Cluster2**. Finally click Finished to complete the wizard. The Partition list screen returns with all the partitions now listed.

[Next: Complete Anthos Prerequisites](#)

## Complete Anthos Prerequisites

Now that the physical environment is set up, you can begin Anthos deployment. This starts with several prerequisites that you must meet to deploy the solution and access it afterward. Each of these steps are discussed in depth in the [Anthos GKE On-Prem Guide](#).

To prepare your environment for the deployment of Anthos on VMware, complete the following steps:

1. Create a Google Cloud project following the instructions available [here](#).



Your organization might already have a project in place intended for this purpose. Check with your cloud administration team to see if a project exists and is already configured for access to Anthos on VMware. All projects intended for use with Anthos must be whitelisted by Google. This includes the primary user account, additional team members, and the access service account created in a later step.

2. Create a deployment workstation from which to manage the installation of Anthos on VMware. The deployment workstation can be Linux, MacOS, or Windows. For the purposes of this validated deployment, Red Hat Enterprise Linux 7 was used.



This workstation can be hosted either internal or external to the NetApp HCI deployment. The only requirement is that it must be able to successfully communicate with the deployed VMware vCenter Server and the internet to function correctly.

3. Install [Google Cloud SDK](#) for interactions with Google Cloud. It can be downloaded as an archive of binaries for manual install or installed by either the apt-get (Ubuntu/Debian) or yum (RHEL) package managers.

```
[user@rhel7 ~]$ sudo yum install google-cloud-sdk
Failed to set locale, defaulting to C
Loaded plugins: langpacks, product-id, search-disabled-repos, subscription-manager
Resolving Dependencies
--> Running transaction check
--> Package google-cloud-sdk.noarch 0:270.0.0-1 will be installed
--> Finished Dependency Resolution
```

Dependencies Resolved

```
=====
=====
Package           Arch      Version       Repository
Size
=====
=====
Installing:
google-cloud-sdk      noarch    270.0.0-1   google-cloud-sdk
36 M
```

Transaction Summary

```
=====
=====
Install 1 Package
```

Total download size: 36 M

Installed size: 174 M

Is this ok [y/d/N]: y

Downloading packages:

```
6d81c821884ae40244c746f6044fc1bcd801143a0d9c8da06767036b8d090a24-google-cloud-sdk-
270.0.0-1.noar | 36 MB  00:00:00
```

Running transaction check

Running transaction test

Transaction test succeeded

Running transaction

```
  Installing : google-cloud-sdk-270.0.0-1.noarch
1/1
```

```
  Verifying  : google-cloud-sdk-270.0.0-1.noarch
1/1
```

Installed:

```
  google-cloud-sdk.noarch 0:270.0.0-1
```

Complete!



The gcloud binary must be at least version 265.0.0. You can update a manual install with a gcloud components update. However, if SDK was installed by a package manager, future updates must also be performed using that same package manager.

4. Install [govc](#), the CLI for VMware vSphere. Installing govc allows you to interact directly with the management of VMware vCenter. Govc is available as a pre-packaged binary available in a gzip format for download. For installation, a user must download the gzip archive, unzip, and copy the resulting binary to a local path directory such as /usr/local/bin.

```
[user@rhel7 ~]$ wget
https://github.com/vmware/govmomi/releases/download/v0.20.0/govc_linux_amd64.gz
--2019-11-06 09:06:10--
https://github.com/vmware/govmomi/releases/download/v0.20.0/govc_linux_amd64.gz
Resolving github.com (github.com)... 13.250.177.223
Connecting to github.com (github.com)|13.250.177.223|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 7650188 (7.3M) [application/octet-stream]
Saving to: 'govc_linux_amd64.gz.1'

100%[=====] 7,650,188  --.-K/s   in 0.1s

2019-11-06 09:06:12 (76.4 MB/s) - 'govc_linux_amd64.gz' saved [7650188/7650188]

[user@rhel7 ~]$ gunzip govc_linux_amd64.gz
[user@rhel7 ~]$ sudo cp govc_linux_amd64 /usr/local/bin/govc
[user@rhel7 ~]$ which govc
/usr/local/bin/govc
```

5. Install [Hashicorp Terraform](#). Terraform enables the automated deployment of VMs in a VMware vSphere environment and is used to deploy the Anthos admin workstation in a later step. Like the govc install, Terraform can be downloaded as a zip archive and unzipped, and the resulting binary can be copied to a directory in the local user's path.

```
[user@rhel7 ~]$ wget https://releases.hashicorp.com/terraform/0.12.13/terraform_0.12.13_linux_amd64.zip --2019-11-06 09:13:40-- https://releases.hashicorp.com/terraform/0.12.13/terraform_0.12.13_linux_amd64.zip Resolving releases.hashicorp.com (releases.hashicorp.com)... 151.101.193.183, 151.101.1.183, 151.101.65.183, ... Connecting to releases.hashicorp.com (releases.hashicorp.com)|151.101.193.183|:443... connected. HTTP request sent, awaiting response... 200 OK Length: 16341595 (16M) [application/zip] Saving to: 'terraform_0.12.13_linux_amd64.zip.1'

100%[=====>] 16,341,595  --.-K/s  in 0.1s

2019-11-06 09:13:40 (141 MB/s) - 'terraform_0.12.13_linux_amd64.zip.1' saved [16341595/16341595]

[user@rhel7 ~]$ unzip terraform_0.12.13_linux_amd64.zip
Archive:  terraform_0.12.13_linux_amd64.zip
  inflating: terraform
[user@rhel7 ~]$ cp terraform /usr/local/bin
[user@rhel7 ~]$ which terraform
/usr/local/bin/terraform
```

6. With the workstation configured, log in to Google Cloud with your credentials. To do so, enter the login command from the deployment workstation and retrieve a link that can be copied and pasted into a browser to allow interactive sign-in to Google services. After you have logged in, the web page presents a code that you can copy and paste back into the deployment workstation when prompted.

```
[user@rhel7 ~]$ gcloud auth login  
Go to the following link in your browser:
```

```
https://accounts.google.com/o/oauth2/auth?code_challenge=-  
7oPNSySHr_Sd2ZZ4K83koIeGTLVcdbjc8omr6zCbAI&prompt=select_account&code_challenge_method  
=S256&access_type=offline&redirect_uri=urn%3Aietf%3Awg%3Aoauth%3A2.0%3Aoob&response_ty  
pe=code&client_id=32655940559.apps.googleusercontent.com&scope=https%3A%3F%2Fwww.googl  
eapis.com%2Fauth%2Fuserinfo.email+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fcloud-  
platform+https%3A%6F%2Fwww.googleapis.com%2Fauth%2Fappengine.admin+https%3A%2F%2Fwww.g  
oogleapis.com%2Fauth%2Fcompute+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Faccounts.rau  
th
```

Enter verification code: 6/swGAh52VVgB-TRS5LVrSvP79ZdD1b9V60byUGqoY67a3zp9NPciIKsM

You are now logged in as [user@netapp.com].

Your current project is [anthos-dev]. You can change this setting by running:

```
$ gcloud config set project PROJECT_ID
```

7. Before you can install Anthos on VMware, you must create four service accounts, each with a specific purpose in interacting with Google Cloud. The following table lists the accounts and their purposes.

*Table 1. Google Cloud Service Accounts*

Account Name	Purpose
access-service-account	Used to download the Anthos binaries from Cloud Storage.
register-service-account	Used to register Anthos clusters to the Google Cloud console.
connect-service-account	Used to maintain the connection between user clusters and the Google Cloud.
stackdriver-service-account	Used to write logging and monitoring data to Stackdriver.



Each account is assigned an email address that references your approved Google Cloud project name. The examples below all list the project Anthos-Dev which was used during the NetApp validation. Make sure to substitute your appropriate project name in syntax examples where necessary.

```
[user@rhel7 ~]$ gcloud iam service-accounts create access-service-account
[user@rhel7 ~]$ gcloud iam service-accounts create register-service-account
[user@rhel7 ~]$ gcloud iam service-accounts create connect-service-account
[user@rhel7 ~]$ gcloud iam service-accounts create stackdriver-service-account
[user@rhel7 ~]$ gcloud iam service-accounts list
NAME          EMAIL
DISABLED
False          stackdriver-service-account@anthos-dev.iam.gserviceaccount.com
False          register-service-account@anthos-dev.iam.gserviceaccount.com
False          access-service-account@anthos-dev.iam.gserviceaccount.com
False          connect-service-account@anthos-dev.iam.gserviceaccount.com
False
```

8. Enable several APIs so that your environment can communicate with Google Cloud. The pods deployed in your clusters must be able to access <https://www.googleapis.com> and <https://gkeconnect.googleapis.com> to function as expected. Therefore, the VM\_Network that the worker nodes are attached to must have internet access. To enable the necessary APIs, run the following command from the deployment workstation:

```
[user@rhel7 ~]$ gcloud services enable \
cloudresourcemanager.googleapis.com \
container.googleapis.com \
gkeconnect.googleapis.com \
gkehub.googleapis.com \
serviceusage.googleapis.com \
stackdriver.googleapis.com \
monitoring.googleapis.com \
logging.googleapis.com
```

9. The final step needed to prepare your environment to deploy Anthos is to limit certain privileges to your service accounts. You need the associated email address for each service account listed in Step 7.
  1. Using the register service account, assign the roles for `gkehub.admin` and `serviceusage.serviceUsageViewer`.

```
[user@rhel7 ~]$ gcloud projects add-iam-policy-binding anthos-dev \
--member "serviceAccount: register-service-account@anthos-
dev.iam.gserviceaccount.com"\ \
--role "roles/gkehub.admin"

[user@rhel7 ~]$ gcloud projects add-iam-policy-binding anthos-dev \
--member "serviceAccount: register-service-account@anthos-
dev.iam.gserviceaccount.com"\ \
--role "roles/serviceusage.serviceUsageViewer"
```

2. Using the connect service account, assign the roles for `gkehub.connect`.

```
[user@rhel7 ~]$ gcloud projects add-iam-policy-binding anthos-dev \
--member "serviceAccount: connect-service-account@anthos-
dev.iam.gserviceaccount.com"\ \
--role "roles/gkehub.connect"
```

3. With the stackdriver service account, assign the roles for `stackdriver.resourceMetadata.writer`, `logging.logWriter`, and `monitoring.metricWriter`.

```
[user@rhel7 ~]$ gcloud projects add-iam-policy-binding anthos-dev \
--member "serviceAccount: stackdriver-service-account@anthos-
dev.iam.gserviceaccount.com"\ \
--role "roles/stackdriver.resourceMetadata.writer"

[user@rhel7 ~]$ gcloud projects add-iam-policy-binding anthos-dev \
--member "serviceAccount: stackdriver-service-account@anthos-
dev.iam.gserviceaccount.com"\ \
--role "roles/logging.logWriter"

[user@rhel7 ~]$ gcloud projects add-iam-policy-binding anthos-dev \
--member "serviceAccount: stackdriver-service-account@anthos-
dev.iam.gserviceaccount.com"\ \
--role "roles/monitoring.metricWriter"
```

Next: [Deploy the Anthos Admin Workstation](#)

## 7. Deploy the Anthos Admin Workstation

The admin workstation is a vSphere VM deployed within your NetApp HCI environment that is preinstalled with all the tools necessary to administer the Anthos on VMware solution. Follow the instructions in this section to download,

## deploy, and configure the Anthos admin workstation.



The admin workstation image is packaged as an OVA file and is only available for download to those users with whitelisted access service accounts. If you are unable to download the OVA from cloud storage, contact your project administrator or open a support ticket with Google Cloud support.

To deploy the Anthos admin workstation, complete the following steps:

1. Download the latest version of the virtual appliance here. You must first set your account to the whitelisted access service account that has permission to download the OVA.

```
[user@rhel7 ~]$ gcloud config set account 'access-service-account@anthos-dev.iam.gserviceaccount.com'  
Updated property [core/account].  
[user@rhel7 ~]$ gsutil cp gs://gke-on-prem-release/admin-appliance/1.1.1-gke.2/gke-on-prem-admin-appliance-vsphere-1.1.1-gke.2.{ova,ova.sig} ~/  
Copying gs://gke-on-prem-release/admin-appliance/1.1.1-gke.2/gke-on-prem-admin-appliance-vsphere-1.1.1-gke.2.ova...  
==> NOTE: You are downloading one or more large file(s), which would run significantly faster if you enabled sliced object downloads. This feature is enabled by default but requires that compiled crcmod be installed (see "gsutil help crcmod").  
- [0 files][ 40.2 MiB/ 5.1 GiB]
```

2. When the OVA file is downloaded, use govc to load it into your vCenter. You need to set several environment variables for your current user that provide specific information about your VMware vSphere environment deployed on NetApp HCI.

```
[user@rhel7 ~]$ export GOVC_URL=https://anthos-vc.cie.netapp.com/sdk  
[user@rhel7 ~]$ export GOVC_USERNAME=administrator@vsphere.local  
[user@rhel7 ~]$ export GOVC_PASSWORD=vcpassword  
[user@rhel7 ~]$ export GOVC_DATASTORE=VM_Datastore  
[user@rhel7 ~]$ export GOVC_DATACENTER=NetApp-HCI-Datacenter-01  
[user@rhel7 ~]$ export GOVC_RESOURCE_POOL= Anthos-Resource-Pool  
[user@rhel7 ~]$ export GOVC_INSECURE=true
```

3. Because the NetApp HCI deployment uses a vDS, you need to create a custom options file that you pass as an additional argument to govc. Create a file called **options\_specs**, and commit the following information to the file:

```

cat > options_spec <<EOF
{
  "DiskProvisioning": "thin",
  "MarkAsTemplate": true,
  "NetworkMapping": [
    {
      "Name": "VM Network",
      "Network": "VM_Network"
    }
  ]
}
EOF

```

- To import the OVA file as template in VMware vSphere, enter the following command referencing the downloaded virtual appliance file, as well as the options file that you just created. The console reports upload progress, and you can also verify the upload by browsing to the Recent Tasks window in VMware vSphere.

```

[user@rhel7 ~]$ govc import.ova -options options_spec .\gke-on-prem-admin-appliance-
vsphere-1.1.1-gke.2.ova
[06-11-19 10:56:39] Uploading gke-on-prem-admin-appliance-vsphere-1.1.1-gke.2.vmdk...
(23%, 26.8MiB/s)

```

Recent Tasks	Alarms		
Task Name	Target	Status	Initiator
Deploy OVF template	gke-on-prem-admin-a...	51%	VSPHERE.LOCAL\Adminis...
Check new notifications	anthos-vc.cle.netapp....	✓ Completed	VMware vSphere Update ...
<hr/>			
All			

- With the template uploaded, we can now use Terraform to deploy the Anthos admin workstation VM. Google provides a Terraform template and variable files for environments using DHCP as well those using static IP addresses. Create a directory for these Terraform files, and copy and paste the text for both the TF and TFPARS files available here into files created with a text editor.

```

[user@rhel7 ~]$ mkdir terraform
[user@rhel7 ~]$ cd terraform
[user@rhel7 ~]$ vi terraform.tfvars

```

- See the following **terraform.tfvars** file contents.

```
# vCenter Server username
vcenter_user = "administrator@vsphere.local"

# vCenter Server password
vcenter_password = "vcpassword"

# vCenter Server IP or hostname
vcenter_server = "https://anthos-vc.cie.netapp.com"

# Path in which the admin workstation's VM's public key should be saved
ssh_public_key_path = "~/.ssh/vsphere_workstation.pub"

# Hostname for the VM
vm_name = "admin-workstation"

# vSphere datastore to use for storage
datastore = "VM_Datastore"

# vSphere datacenter in which to create the VM
datacenter = "NetApp-HCI-Datacenter-01"

# vSphere cluster in which to create the VM
cluster = "NetApp-HCI-Cluster-01"

# vSphere resource pool in which to create VM, if you are using a non-default resource
pool
# If you are using the default resource pool, provide a value like "CLUSTER-
NAME/Resources"
resource_pool = "Anthos-Resource-Pool"

# vSphere network to use for the VM
network = "VM_Network"

# Number of CPUs for this VM. Recommended minimum 4.
num_cpus = 4

# Memory in MB for this VM. Recommended minimum 8192.
memory = 8192

# The VM template (OVA) to clone. Change the version if you imported a different
version of the OVA.
vm_template = "gke-on-prem-admin-appliance-vsphere-1.1.1-gke.2"
```

```
[user@rhel7 ~]$ vi terraform.tf
```

7. See the following `terraform.tf` file contents.

```
#####
##### VARIABLES #####
#####

# The following variables are declared in the accompanying TFVARS file

# vCenter Server username
variable "vcenter_user" { }

# vCenter Server password
variable "vcenter_password" { }

# vCenter Server address
variable "vcenter_server" { }

# Path in which the VM's public key should be saved
variable "ssh_public_key_path" { default = "~/.ssh/vsphere_workstation.pub" }

# vSphere network to use for the VM
variable "network" { default = "VM Network" }

# Hostname for the VM
variable "vm_name" { default = "vsphere-workstation" }

# vSphere datacenter in which to create the admin workstation VM
variable "datacenter" { }

# vSphere datastore to use for storage
variable "datastore" { }

# vSphere cluster in which to create the VM
variable "cluster" { }

# vSphere resource pool in which to create the VM
variable "resource_pool" { }

# Number of CPUs for this VM. Recommended minimum 4.
variable "num_cpus" { default = 4 }

# Memory in MB for this VM. Recommended minimum 8192.
variable "memory" { default = 8192 }

# The VM template (OVA) to clone
variable "vm_template" { }
```

```

#####
#####

provider "vsphere" {
  version      = "~> 1.5"
  user         = "${var.vcenter_user}"
  password     = "${var.vcenter_password}"
  vcenter_server = "${var.vcenter_server}"

  # if you have a self-signed cert
  allow_unverified_ssl = true
}

### vSphere Data ###

data "vsphere_datastore" "datastore" {
  name          = "${var.datastore}"
  datacenter_id = "${data.vsphere_datacenter.dc.id}"
}

data "vsphere_datacenter" "dc" {
  name = "${var.datacenter}"
}

data "vsphere_compute_cluster" "cluster" {
  name          = "${var.cluster}"
  datacenter_id = "${data.vsphere_datacenter.dc.id}"
}

data "vsphere_resource_pool" "pool" {
  name          = "${var.resource_pool}"
  datacenter_id = "${data.vsphere_datacenter.dc.id}"
}

data "vsphere_network" "network" {
  name          = "${var.network}"
  datacenter_id = "${data.vsphere_datacenter.dc.id}"
}

data "vsphere_virtual_machine" "template_from_ovf" {
  name          = "${var.vm_template}"
  datacenter_id = "${data.vsphere_datacenter.dc.id}"
}

data "template_file" "dhcp_ip_config" {
  template = <<EOF
network:

```

```

version: 2
ethernets:
  ens192:
    dhcp4: true
EOF
}

data "template_file" "user_data" {
  template = <<EOF
#cloud-config
apt:
  primary:
    - arches: [default]
      uri: http://us-west1.gce.archive.ubuntu.com/ubuntu/
write_files:
  - path: /etc/netplan/99-dhcp.yaml
    permissions: '0644'
    encoding: base64
    content: |
      $$\{dhcp_ip_config\}
runcmd:
  - netplan apply
  - /var/lib/gke/guest-startup.sh
EOF
vars = {
  dhcp_ip_config = "${base64encode(data.template_file.dhcp_ip_config.rendered)}"
}
}

### vSphere Resources ###

resource "vsphere_virtual_machine" "vm" {
  name          = "${var.vm_name}"
  resource_pool_id = "${data.vsphere_resource_pool.pool.id}"
  datastore_id    = "${data.vsphere_datastore.datastore.id}"
  num_cpus        = "${var.num_cpus}"
  memory          = "${var.memory}"
  guest_id         = "${data.vsphere_virtual_machine.template_from_ovf.guest_id}"
  enable_disk_uuid = "true"
  scsi_type       = "${data.vsphere_virtual_machine.template_from_ovf.scsi_type}"
  network_interface {
    network_id   = "${data.vsphere_network.network.id}"
    adapter_type =
      "${data.vsphere_virtual_machine.template_from_ovf.network_interface_types[0]}"
  }
  wait_for_guest_net_timeout = 15
}

```

```

nested_hv_enabled = false
cpu_performance_counters_enabled = false

disk {
    label          = "disk0"
    size           = "${max(50,
data.vsphere_virtual_machine.template_from_ovf.disks.0.size)}"
    eagerly_scrub   =
"${data.vsphere_virtual_machine.template_from_ovf.disks.0.eagerly_scrub}"
    thin_provisioned =
"${data.vsphere_virtual_machine.template_from_ovf.disks.0.thin_provisioned}"
}

cdrom {
    client_device = true
}

vapp {
    properties = {
        hostname      = "${var.vm_name}"
        public-keys   = "${file(var.ssh_public_key_path)}"
        user-data     = "${base64encode(data.template_file.user_data.rendered)}"
    }
}

clone {
    template_uuid = "${data.vsphere_virtual_machine.template_from_ovf.id}"
}
}

output "ip_address" {
    value = "${vsphere_virtual_machine.vm.default_ip_address}"
}

```



Values specific to the deployed environment have been added to the `terraform.tfvars` file. However, you should not modify the `terraform.tf` file in any manner.

8. Create an SSH public/private keypair used to log in to the admin workstation after it is deployed. Name the public key so that it matches the variable that was assigned in the `terraform.tfvars` file.

```
[user@rhel7 ~]$ ssh-keygen -t rsa -f ~/.ssh/vsphere_workstation -N ""
Generating public/private rsa key pair.
Your identification has been saved in /home/user/.ssh/vsphere_workstation2.
Your public key has been saved in /home/user/.ssh/vsphere_workstation2.pub.
The key fingerprint is:
SHA256:qEk8G13LhwIKqf85ekHHZkIZduX2MkZUxGNEHvFT2vw user@rhel7
The key's randomart image is:
+---[RSA 2048]---+
|   oo.0+*B. . |
|   .o. o .+o = |
|   ... B.o.+ o |
| o ooo=B *   . |
|o ..==+ S .     E|
|. ...*       . |
|.   +.       . |
|.   o.       . |
|   o+o.       . |
+---[SHA256]---+
```

9. Navigate to the directory created to host the TF and TFPARS files. Within this directory, initialize Terraform and use it to launch the deployment of the admin workstation VM.

```
[user@rhel7 ~]$ cd terraform
[user@rhel7 terraform]$ ls
terraform.tf  terraform.tfvars
[user@rhel7 terraform]$ terraform init && terraform apply -auto-approve -input=false
Initializing the backend...
Initializing provider plugins...
The following providers do not have any version constraints in configuration,
so the latest version was installed.
To prevent automatic upgrades to new major versions that may contain breaking
changes, it is recommended to add version = "..." constraints to the
corresponding provider blocks in configuration, with the constraint strings
suggested below.
* provider.template: version = "~> 2.1"
Terraform has been successfully initialized!

data.template_file.dhcp_ip_config: Refreshing state...
data.template_file.user_data: Refreshing state...
data.vsphere_datacenter.dc: Refreshing state...
data.vsphere_resource_pool.pool: Refreshing state...
data.vsphere_datastore.datastore: Refreshing state...
data.vsphere_virtual_machine.template_from_ovf: Refreshing state...
data.vsphere_network.network: Refreshing state...
data.vsphere_compute_cluster.cluster: Refreshing state...
vsphere_virtual_machine.vm: Creating...
vsphere_virtual_machine.vm: Still creating... [10s elapsed]
vsphere_virtual_machine.vm: Still creating... [20s elapsed]
vsphere_virtual_machine.vm: Still creating... [30s elapsed]
vsphere_virtual_machine.vm: Still creating... [40s elapsed]
vsphere_virtual_machine.vm: Creation complete after 49s [id=42118cfa-d464-b815-f7b1-37cd85b2943a]
```

Warning: "vcenter\_server": [DEPRECATED] This field has been renamed to vsphere\_server.

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

Outputs:

```
ip_address = 10.63.172.21
```

[Next: Deploy the Admin and the First User Cluster](#)

## 8. Deploy the Admin and the First User Cluster

All Kubernetes clusters deployed as a part of the Anthos solution are deployed from the Anthos admin workstation that you just created. A user logs into the

admin workstation using SSH, the public key created in a previous step, and the IP address provided at the end of the VM deployment. They can then begin creating their first clusters.



There are specific procedures for deploying clusters that use static IP addresses [here](#), and procedures for environments with DHCP can be found [here](#). In this guide, we use the second set of instructions.

To deploy the initial admin and first user cluster, complete the following steps:

1. Login to the admin workstation using the SSH public key and the IP address output at the end of Terraform deployment.

```
[user@rhel7 ~]$ ssh -i ~/.ssh/vsphere_workstation ubuntu@10.63.172.21
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-62-generic x86_64)
```

```
* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:       https://ubuntu.com/advantage
```

This system has been minimized by removing packages and content that are not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.

```
36 packages can be updated.
23 updates are security updates.
```

```
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
```

```
ubuntu@admin-workstation:~$
```

2. Log in with the `gcloud auth` command as you did before from your deployment workstation by copying the URL into a web browser, signing into your Google account, and pasting the verification code back into the workstation.

```
ubuntu@admin-workstation:~$ gcloud auth login  
Go to the following link in your browser:
```

```
https://accounts.google.com/o/oauth2/auth?code_challenge=Q1F7H-  
CMUMuArasQD6AzHA0avKoLGfDqUyUgjFxf9ZI&prompt=select_account&code_challenge_method=S256  
&access_type=offline&redirect_uri=urn%3Aietf%3Awg%3Aoauth%3A2.0%3Aoob&response_type=co  
de&client_id=32555940559.apps.googleusercontent.com&scope=https%3A%2F%2Fwww.googleapis  
.com%2Fauth%2Fuserinfo.email+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fcloud-  
platform+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fappengine.admin+https%3A%2F%2Fwww.g  
oogleapis.com%2Fauth%2Fcompute+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Faccounts.rau  
th
```

```
Enter verification code: 6/swFG_ZKZyd0eb1fuXsgtnBgYrIt1XwFQeB5ASrEqaaDfa09aio0bnNg
```

```
WARNING: `gcloud auth login` no longer writes application default credentials.
```

```
If you need to use ADC, see:
```

```
gcloud auth application-default --help
```

```
You are now logged in as [user@netapp.com].
```

```
Your current project is [None]. You can change this setting by running:
```

```
$ gcloud config set project PROJECT_ID
```

3. Set the project that you intend to deploy your clusters in for Anthos on VMWare, because the project is not set by default when you log in.

```
ubuntu@admin-workstation:~$ gcloud config set project anthos-dev  
Updated property [core/project].
```

4. Register gcloud as a Docker credential helper, which enables it to manage the credentials for Docker registries used for deployment. This way, the default credential store is not used for operations involving the credentials of the specified registries.

```
ubuntu@admin-workstation:~$ gcloud auth configure-docker
The following settings will be added to your Docker config file
located at [/home/ubuntu/.docker/config.json]:
{
  "credHelpers": {
    "gcr.io": "gcloud",
    "us.gcr.io": "gcloud",
    "eu.gcr.io": "gcloud",
    "asia.gcr.io": "gcloud",
    "staging-k8s.gcr.io": "gcloud",
    "marketplace.gcr.io": "gcloud"
  }
}
```

Do you want to continue (Y/n)? y

Docker configuration file updated.



By default, Anthos on VMware uses a pre-existing, Google-owned container image registry that requires no additional setup. If you choose to use a private Docker registry for deployment, then you must configure that registry separately based on instructions found [here](#). This step is beyond the scope of this deployment guide.

5. In the next step to deploy an admin cluster, create a private key file in the JSON format for each of the service accounts created in the prerequisites section.

```

ubuntu@admin-workstation:~$ gcloud iam service-accounts list
NAME          EMAIL
DISABLED
stackdriver-service-account@anthos-dev.iam.gserviceaccount.com
False
register-service-account@anthos-dev.iam.gserviceaccount.com
False
access-service-account@anthos-dev.iam.gserviceaccount.com
False
connect-service-account@anthos-dev.iam.gserviceaccount.com
False

ubuntu@admin-workstation:~$ gcloud iam service-accounts keys create access-key.json
--iam-account access-service-account@anthos-dev.iam.gserviceaccount.com
created key [8d5f8ce039dd98766e18a3c5ee6794912fb8d095] of type [json] as [access-
key.json] for [access-service-account@anthos-dev.iam.gserviceaccount.com]
ubuntu@admin-workstation:~$ gcloud iam service-accounts keys create register-key.json
--iam-account register-service-account@anthos-dev.iam.gserviceaccount.com
created key [f08b494c665321f83bcb8c8526ba21185b456a11] of type [json] as [register-
key.json] for [register-service-account@anthos-dev.iam.gserviceaccount.com]
ubuntu@admin-workstation:~$ gcloud iam service-accounts keys create connect-key.json
--iam-account connect-service-account@anthos-dev.iam.gserviceaccount.com
created key [c9640021ff6157d3df2a15db49f5c85b7b1495c2] of type [json] as [connect-
key.json] for [connect-service-account@anthos-dev.iam.gserviceaccount.com]
ubuntu@admin-workstation:~$ gcloud iam service-accounts keys create stackdriver-
key.json --iam-account stackdriver-service-account@anthos-dev.iam.gserviceaccount.com
created key [3c9427dfdef161d139ff998be896565c1df0b122] of type [json] as [stackdriver-
key.json] for [stackdriver-service-account@anthos-dev.iam.gserviceaccount.com]

```

- For the next step, you must use the access service key created in the previous step to activate the associated service account.

```

ubuntu@admin-workstation:~$ gcloud auth activate-service-account --key-file=access
-key.json
Activated service account credentials for: [access-service-account@anthos-
dev.iam.gserviceaccount.com]

```

- The deployment of the first clusters is performed using inputs from a config file generated by GKE. A generic config file can be created with no additional input, or an existing config file can be referenced to create additional clusters.

```
ubuntu@admin-workstation:~$ gkectl create-config  
ubuntu@admin-workstation:~$ ls  
access-key.json config.yaml connect-key.json register-key.json stackdriver-  
key.json
```

8. The **config.yaml** file created by running the previous command has several variables that must be customized for the current environment.

1. First, you must determine the full path location and the name of the current GKE bundle that is deployed into the environment. The file exists in the **/var/lib/gke/bundles** directory on the admin workstation.

```
ubuntu@admin-workstation:~$ ls /var/lib/gke/bundles  
gke-onprem-vsphere-1.1.1-gke.2-full.tgz gke-onprem-vsphere-1.1.1-gke.2.tgz
```

2. Next, you must get the fully recognized host name or IP address of our vCenter Server as displayed in its default SSL certificate. Connect to vSphere and dump the certificate contents into a file called **vcenter.pem**. Examining this file gives you the information that you need for the value of **Subject: CN** (common name).

```
ubuntu@admin-workstation:~$ true | openssl s_client -connect anthos-  
vc.cie.netapp.com:443 -showcerts 2>/dev/null | sed -ne '/-BEGIN/,/-END/p' >  
vcenter.pem  
ubuntu@admin-workstation:~$ openssl x509 -in vcenter.pem -text -noout | grep  
Subject:\ CN  
Subject: CN = anthos-vc.cie.netapp.com, C = US
```



If the value added to the **config.yaml** file does not match that of the CN found in the certificate, communication with the vCenter server fails.

9. With the information from those two commands and the generated **vcenter.pem** file, we can now edit the **config.yaml** file to prepare for deployment. Editing this file is very similar to the edits that you performed to the **terraform.tfvars** file to provide specifics regarding the VMware vCenter instance deployed in NetApp HCI.



When deploying the cluster, determine which IP addresses to use for the control plane and ingress VIPs for both the admin and user cluster. Also determine the compute and memory resources that must be reserved for each node deployed, because it is not possible to edit a cluster after it has been deployed.

```
# Absolute path to a GKE bundle on disk  
bundlepath: "/var/lib/gke/bundles/gke-onprem-vsphere-1.1.1-gke.2-full.tgz"
```

```

# Specify which vCenter resources to use for deployment
vcenter:
  # The credentials and address GKE should use to connect to vCenter
  credentials:
    address: "anthos-vc.cie.netapp.com"
    username: "administrator@vsphere.local"
    password: "vcpass"
  datacenter: "NetApp-HCI-Datacenter-01"
  datastore: "VM_Datastore"
  cluster: "NetApp-HCI-Cluster-01"
  network: "VM_Network"
  resourcepool: "Anthos Resource Pool"
  # Provide the name for the persistent disk to be used by the deployment (ending
  # in .vmdk). Any directory in the supplied path must be created before deployment.
  # Not required when adding additional user clusters
  datadisk: "anthos-admin-data-disk.vmdk"
  # Provide the path to vCenter CA certificate pub key for SSL verification
  cacertpath: "/home/ubuntu/vcenter.pem"

# Specify the proxy configuration.
proxy:
  # The URL of the proxy
  url: ""
  # The domains and IP addresses excluded from proxying
  noproxy: ""

# Specify admin cluster settings for a fresh GKE On-Prem deployment. Omit this section
# and use the --adminconfig flag when adding a new user cluster to an existing
# deployment
admincluster:
  # In-Cluster vCenter configuration
  vcenter:
    # If specified it overwrites the network field in global vcenter configuration
    network: ""
    # # The absolute or relative path to the yaml file to use for static IP allocation.
    # # Do not include if using DHCP
    # ipblockfilepath: ""
    # # Specify pre-defined nodeports if using "manual" load balancer mode
    # manuallbspec:
    #   ingresshttpnodeport: 32527
    #   ingresshttpsnodeport: 30139
    #   controlplanenodeport: 30968
    #   addonsnodeport: 31405
    # Specify the already-existing partition and credentials to use with F5
    bigip:
      # To re-use credentials across clusters we recommend using YAML node anchors.
      # See https://yaml.org/spec/1.2/spec.html#id2785586
      credentials:
        address: "172.21.224.22"
        username: "admin"

```

```

    password: "lbpass"
    partition: "Anthos-Admin-Part"
    # # Optionally specify a pool name if using SNAT
    # snatpoolname: ""
# The VIPs to use for load balancing
vips:
    # Used to connect to the Kubernetes API
    controlplanevip: "10.63.172.98"
    # Shared by all services for ingress traffic
    ingressvip: "10.63.172.99"
    # # Used for admin cluster addons (needed for multi cluster features). Must be the
same
    # # across clusters
    # addonsvip: ""
# The Kubernetes service CIDR range for the cluster. Must not overlap with the pod
# CIDR range
serviceiprange: 10.96.232.0/24
# The Kubernetes pod CIDR range for the cluster. Must not overlap with the service
# CIDR range
podiprange: 192.168.0.0/16
# Specify settings when deploying a new user cluster. Used both with a fresh
deployment
# or when adding a new cluster to an existing deployment.
usercluster:
    antiaffinitygroups:
        enabled: false
    # In-Cluster vCenter configuration
    vcenter:
        # If specified it overwrites the network field in global vcenter configuration
        network: ""
        # # The absolute or relative path to the yaml file to use for static IP allocation.
        # # Do not include if using DHCP
        # ipblockfilepath: ""
        # # Specify pre-defined nodeports if using "manual" load balancer mode
        # manuallbspec:
        #     ingresshttpnodeport: 30243
        #     ingresshttpsnodeport: 30879
        #     controlplanenodeport: 30562
        #     addonsnodeport: 0
        # Specify the already-existing partition and credentials to use with F5
        bigip:
            # To re-use credentials across clusters we recommend using YAML node anchors.
            # See https://yaml.org/spec/1.2/spec.html#id2785586
            credentials:
                address: "172.21.224.22"
                username: "admin"
                password: "lbpass"
            partition: "Anthos-Cluster01-Part"

```

```
# # Optionally specify a pool name if using SNAT
# snatpoolname: ""

# The VIPs to use for load balancing
vips:
    # Used to connect to the Kubernetes API
    controlplanevip: "10.63.172.105"
    # Shared by all services for ingress traffic
    ingressvip: "10.63.172.106"
    # # Used for admin cluster addons (needed for multi cluster features). Must be the
    same
        # # across clusters
        # addonsvip: ""

    # A unique name for this cluster
    clustername: "anthos-cluster01"

    # User cluster master nodes must have either 1 or 3 replicas
    masternode:
        cpus: 4
        memorymb: 8192
        # How many machines of this type to deploy
        replicas: 1

    # The number of worker nodes to deploy and their size. Min. 2 replicas
    workernode:
        cpus: 4
        memorymb: 8192
        # How many machines of this type to deploy
        replicas: 3

    # The Kubernetes service CIDR range for the cluster
    serviceiprange: 10.96.0.0/12
    # The Kubernetes pod CIDR range for the cluster
    podiprange: 192.168.0.0/16

    # # Uncomment this section to use OIDC authentication
    # oidc:
    #     issuerurl: ""
    #     kubectlredirecturl: ""
    #     clientid: ""
    #     clientsecret: ""
    #     username: ""
    #     usernameprefix: ""
    #     group: ""
    #     groupprefix: ""
    #     scopes: ""
    #     extraparams: ""
    #     # Set value to string "true" or "false"
    #     usehttpproxy: ""
    #     # # The absolute or relative path to the CA file (optional)
    #     # capath: ""
    #     # # Optionally provide an additional serving certificate for the API server
    #     sni:
```

```

#   certpath: ""
#   keypath: ""
# Which load balancer mode to use "Manual" or "Integrated"
lbmode: Integrated
# Specify which GCP project to connect your GKE clusters to
gkeconnect:
  projectid: "anthos-dev"
  # The absolute or relative path to the key file for a GCP service account used to
  # register the cluster
  registerserviceaccountkeypath: "/home/ubuntu/register-key.json"
  # The absolute or relative path to the key file for a GCP service account used by
  # the GKE connect agent
  agentserviceaccountkeypath: "/home/ubuntu/connect-key.json"
# Specify which GCP project to connect your logs and metrics to
stackdriver:
  projectid: "anthos-dev"
  # A GCP region where you would like to store logs and metrics for this cluster.
  clusterlocation: "us-east1"
  enablevpc: false
  # The absolute or relative path to the key file for a GCP service account used to
  # send logs and metrics from the cluster
  serviceaccountkeypath: "/home/ubuntu/stackdriver-key.json"
# # Optionally use a private Docker registry to host GKE images
# privaterRegistryconfig:
#   # Do not include the scheme with your registry address
#   credentials:
#     address: ""
#     username: ""
#     password: ""
#   # The absolute or relative path to the CA certificate for this registry
#   cacertpath: ""
# The absolute or relative path to the GCP service account key that will be used to
# pull GKE images
gcrkeypath: "/home/ubuntu/access-key.json"
# Configure kubernetes apiserver audit logging
cloudauditlogging:
  projectid: ""
  # A GCP region where you would like to store audit logs for this cluster.
  clusterlocation: ""
  # The absolute or relative path to the key file for a GCP service account used to
  # send audit logs from the cluster
  serviceaccountkeypath: ""

```

- Because spacing in YAML files can be very important, you can check the syntax of the config file by running the following command. If the command outputs any failures, be sure to examine the file and make any needed corrections.

```
ubuntu@admin-workstation:~$ gkectl check-config --config config.yaml
```

- Validation Category: Config Check
  - [SUCCESS] Config
- Validation Category: Docker Registry
  - [SUCCESS] gcr.io/gke-on-prem-release access
- Validation Category: vCenter
  - [SUCCESS] Credentials
  - [SUCCESS] Datacenter
  - [SUCCESS] Datastore
  - [SUCCESS] Data Disk
  - [SUCCESS] Resource Pool
  - [SUCCESS] Network
- Validation Category: F5 BIG-IP
  - [SUCCESS] Credentials
  - [SUCCESS] Partition
- Validation Category: Network Configuration
  - [SUCCESS] CIDR, VIP and static IP (availability and overlapping)
- Validation Category: VIPs
  - [SUCCESS] ping (availability)
- Validation Category: Node IPs
  - [SKIPPED] ping (availability): All specified clusters use DHCP.

Some validations FAILED or SKIPPED. Check report above.



Using DHCP skips the step to validate node IP availability. This is an expected behavior and deployment can continue.

11. Preparing the cluster for deployment and deploying the cluster are performed with two commands:
  1. The `gkectl prepare` command initializes the vSphere environment by uploading the node OS image, marking it as a template, and validating the build attestations for all container images.

```
ubuntu@admin-workstation:~$ gkectl prepare --config config.yaml
- Validation Category: Config Check
  - [SUCCESS] Config

- Validation Category: Docker Registry
  - [SUCCESS] gcr.io/gke-on-prem-release access

- Validation Category: vCenter
  - [SUCCESS] Credentials
  - [SUCCESS] Datacenter
  - [SUCCESS] Datastore
  - [SUCCESS] Data Disk
  - [SUCCESS] Resource Pool
  - [SUCCESS] Network
```

All validations SUCCEEDED.

```
Downloading OS image gke-on-prem-osimage-1.13.7-gke.20-20190816-8138298d96.ova...
DONE
```

Setting up OS image as a VM template in vSphere... DONE

2. The `gkectl create cluster` command deploys the cluster as depicted in the `config.yaml` file.

```
ubuntu@admin-workstation:~$ gkectl create cluster --config config.yaml
```

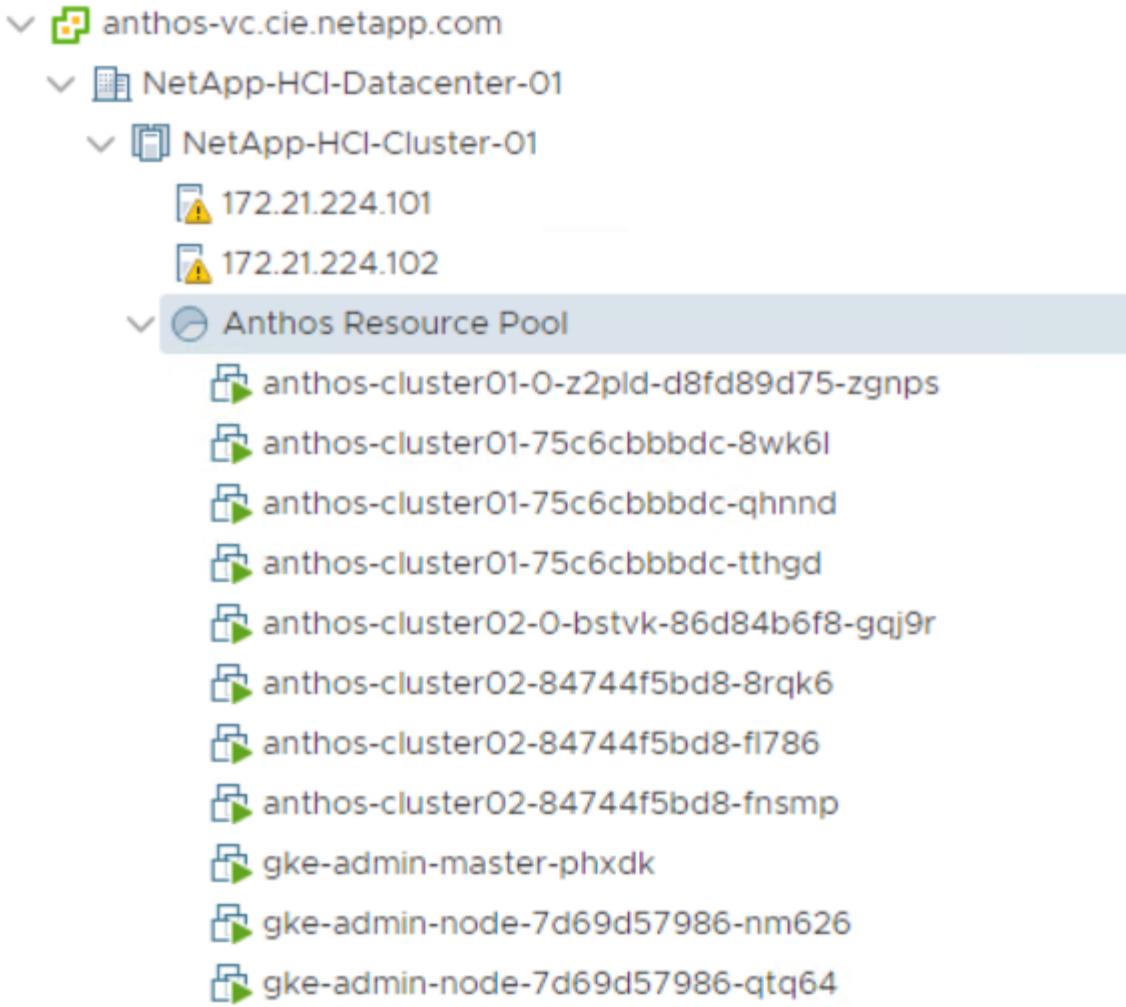
3. The process runs for several minutes and can be monitored on screen and in vCenter by watching the resource pool as the VMs populate. When complete, you should be able to see the `gke-admin` cluster (three nodes) and the first user cluster (four nodes).



During the deployment process, the standard out might display several messages about the current node not being available or not being ready. This is normal and happens when the control plane checks for machines that have not yet completed deployment or received DHCP addresses.



When using DHCP, if a deployment fails because nodes cannot be reached, there might not be enough available addresses in the pool. Leases for previously failed deployments might need to be cleared manually to allow for additional deployment attempts.



4. You can access and execute commands against the user cluster that has been deployed using the `kubectl` command line tool and the `kubeconfig` file generated by the process (stored in the working directory).

```
ubuntu@Anthos-Admin-Workstation:~$ kubectl get nodes --kubeconfig anthos-cluster01-kubeconfig
NAME                  STATUS  ROLES   AGE      VERSION
anthos-cluster01-75c6cbbbdc-8wk6l  Ready   <none>  149m    v1.13.7-gke.20
anthos-cluster01-75c6cbbbdc-qhnnd  Ready   <none>  149m    v1.13.7-gke.20
anthos-cluster01-75c6cbbbdc-tthgd  Ready   <none>  149m    v1.13.7-gke.20
```

## 9. Deploy Additional User Clusters: NetApp HCI with Anthos

With Anthos, organizations can scale their environments to incorporate multiple user clusters and segregate workloads between teams. A single admin cluster can support up to five user clusters, and each user cluster can support up to twenty-five nodes.

To add additional user clusters to your deployment, complete the following steps:

1. Copy the `config.yaml` file to a new file named `anthos-cluster02-config.yaml`.

```
ubuntu@Anthos-Admin-Workstation:~$ cp config.yaml anthos-cluster02-config.yaml
```

2. Make the following edits to the newly created file:

1. Comment out the sections that refer to the existing admin cluster with (#).
2. When you get to the `usercluster` section, update the following fields:
  1. Update the partition name under the `bigip` section.
  2. Update the `controlplanvip` and `ingressvip` values under the `vip` section.
  3. Update the `clusternamespace` value.

```
usercluster:  
  # In-Cluster vCenter configuration  
  vcenter:  
    # If specified it overwrites the network field in global vcenter  
    configuration  
    network: ""  
    # # The absolute or relative path to the yaml file to use for static IP  
    allocation.  
    # # Do not include if using DHCP  
    # ipblockfilepath: ""  
    # # Specify pre-defined nodeports if using "manual" load balancer mode  
    # manuallbspec:  
    #   ingresshttpnodeport: 30243  
    #   ingresshttpsnodeport: 30879  
    #   controlplanenodeport: 30562  
    #   addonsnodeport: 0  
    # Specify the already-existing partition and credentials to use with F5  
    bigip:  
      # To re-use credentials across clusters we recommend using YAML node  
      anchors.  
      # See https://yaml.org/spec/1.2/spec.html#id2785586  
      credentials:  
        address: "172.21.224.22"  
        username: "admin"  
        password: "NetApp!23"  
      partition: "Anthos-Cluster02-Part"  
      # # Optionally specify a pool name if using SNAT  
      # snatpoolname: ""  
      # The VIPs to use for load balancing  
      vips:  
        # Used to connect to the Kubernetes API
```

```
controlplanevip: "10.63.172.108"
# Shared by all services for ingress traffic
ingressvip: "10.63.172.109"
# # Used for admin cluster addons (needed for multi cluster features). Must
be the same
# # across clusters
# addonsvip: ""
# A unique name for this cluster
clustername: "anthos-cluster02"
# User cluster master nodes must have either 1 or 3 replicas
masternode:
  cpus: 4
  memorymb: 8192
  # How many machines of this type to deploy
  replicas: 1
# The number of worker nodes to deploy and their size. Min. 2 replicas
workernode:
  cpus: 4
  memorymb: 8192
  # How many machines of this type to deploy
  replicas: 3
# The Kubernetes service CIDR range for the cluster
serviceiprange: 10.96.0.0/12
# The Kubernetes pod CIDR range for the cluster
podiprange: 192.168.0.0/16
```

3. Run the following command to check the config file again to verify that there are no syntax errors. Because you have removed the admin section, you must reference the **kubeconfig** file for the admin cluster named **kubeconfig** (found in the working directory).

```
ubuntu@Anthos-Admin-Workstation:~$ gkectl check-config --config anthos-cluster02-
config.yaml --kubeconfig kubeconfig
- Validation Category: Config Check
  - [SUCCESS] Config

- Validation Category: Docker Registry
  - [SUCCESS] gcr.io/gke-on-prem-release access

- Validation Category: vCenter
  - [SUCCESS] Credentials
  - [SUCCESS] Datacenter
  - [SUCCESS] Datastore
  - [FAILURE] Data Disk: vCenter data disk already exists
  - [SUCCESS] Resource Pool
  - [SUCCESS] Network

- Validation Category: F5 BIG-IP
  - [SUCCESS] Credentials
  - [SUCCESS] Partition

- Validation Category: Network Configuration
  - [SUCCESS] CIDR, VIP and static IP (availability and overlapping)

- Validation Category: VIPs
  - [SUCCESS] ping (availability)

- Validation Category: Node IPs
  - [SUCCESS] ping (availability)
```

Some validations FAILED or SKIPPED. Check report above.

4. If all the checks succeed as expected, you can deploy this new user cluster in a manner very similar to the first cluster creation, referencing the **kubeconfig** file from the admin cluster.

```
ubuntu@Anthos-Admin-Workstation:~$ gkectl create cluster --config anthos-cluster02-
config.yaml --kubeconfig kubeconfig
```

5. As with the previous deployment, the process runs for several minutes and can be monitored on screen and in vCenter by watching the resource pool as the VMs populate. When complete, you should be able to see the new user cluster (four nodes).

The screenshot shows the Google Cloud Anthos Resource Pool interface. At the top level, there's a node named 'anthos-vc.cie.netapp.com'. Under it, a 'NetApp-HCI-Datacenter-01' project is expanded, showing a 'NetApp-HCI-Cluster-01' sub-project. This cluster contains two nodes: '172.21.224.101' and '172.21.224.102', both of which have warning icons. Below the cluster, an 'Anthos Resource Pool' section is expanded, listing several deployed clusters: 'anthos-cluster01-0-z2pld-d8fd89d75-zgnps', 'anthos-cluster01-75c6cbbbdc-8wk6l', 'anthos-cluster01-75c6cbbbdc-qhnnd', 'anthos-cluster01-75c6cbbbdc-tthgd', 'gke-admin-master-phxdk', 'gke-admin-node-7d69d57986-nm626', and 'gke-admin-node-7d69d57986-qtq64'.

6. You can access and execute commands against the deployed user cluster using the `kubectl` command line tool and the `kubeconfig` file generated by the process (stored in the working directory).

```
ubuntu@Anthos-Admin-Workstation:~$ kubectl get nodes --kubeconfig anthos-cluster02-kubeconfig
NAME                  STATUS   ROLES      AGE      VERSION
anthos-cluster02-84744f5bd8-8rqk6  Ready    <none>    9m16s   v1.13.7-gke.20
anthos-cluster02-84744f5bd8-f1786  Ready    <none>    9m28s   v1.13.7-gke.20
anthos-cluster02-84744f5bd8-fnsmp  Ready    <none>    9m21s   v1.13.7-gke.20
```

## 10. Enable Access to the Cluster with the GKE Console: NetApp HCI with Anthos

After clusters are deployed and registered with Google Cloud, they must be logged into with the Google Cloud console to be managed and to receive additional cluster details. The official procedure to gain access to Anthos user clusters after they are deployed is detailed [here](#).

 The project and the specific user must be whitelisted to access on-premises clusters in the Google Cloud console and use Anthos on VMware services. If you are unable to see the clusters after they are deployed, you might need to open a support ticket with Google.

*Figure 5. Non-whitelisted view.*

*Figure 6. View of clusters.*

To enable access to your user clusters using the GKE console, complete the following steps:

1. Create a `node-reader.yaml` file that gives you the ability to access the cluster.

```
kind: clusterrole
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  name: node-reader
rules:
- apiGroups: []
  resources: ["nodes"]
  verbs: ["get", "list", "watch"]
```

2. Apply this file to the cluster that you want to log into with the `kubectl` command.

```
ubuntu@Anthos-Admin-Workstation:~$ kubectl apply -f node-reader.yaml --kubeconfig
anthos-cluster01-kubeconfig
clusterrole.rbac.authorization.k8s.io/node-reader created
```

3. Create a Kubernetes service account (KSA) that you can use to login. Name this account after the user that uses this account to log into the cluster.

```
ubuntu@Anthos-Admin-Workstation:~$ kubectl create serviceaccount netapp-user
--kubeconfig anthos-cluster01-kubeconfig
serviceaccount/netapp-user created
```

4. Create cluster role-binding resources to bind both the view and newly created node-reader roles to the newly created KSA.

```
ubuntu@Anthos-Admin-Workstation:~$ kubectl create clusterrolebinding netapp-user-view
--clusterrole view --serviceaccount default:netapp-user --kubeconfig anthos-cluster01-
kubeconfig
clusterrolebinding.rbac.authorization.k8s.io/netapp-user-view created
ubuntu@Anthos-Admin-Workstation:~$ kubectl create clusterrolebinding netapp-user-node-
reader --clusterrole node-reader -
--serviceaccount default:netapp-user --kubeconfig anthos-cluster01-kubeconfig
clusterrolebinding.rbac.authorization.k8s.io/netapp-user-node-reader created
```

5. If you need to extend permissions further, you can grant the KSA user a role with cluster admin

permissions in a similar manner.

```
ubuntu@Anthos-Admin-Workstation:~$ kubectl create clusterrolebinding netapp-user-admin --clusterrole cluster-admin --serviceaccount default:netapp-user --kubeconfig anthos-cluster01-kubeconfig  
clusterrolebinding.rbac.authorization.k8s.io/netapp-user-admin created
```

- With the KSA account created and assigned with correct permissions, you can create a bearer token to allow access with the GKE Console. To do so, set a system variable for the secret name, and pass that variable through a `kubectl` command to generate the token.

```
ubuntu@Anthos-Admin-Workstation:~$ SECRET_NAME=$(kubectl get serviceaccount netapp-user --kubeconfig anthos-cluster01-kubeconfig -o jsonpath='{$.secrets[0].name}')  
ubuntu@Anthos-Admin-Workstation:~$ kubectl get secret ${SECRET_NAME} --kubeconfig anthos-cluster01-kubeconfig -o jsonpath='{$.data.token}' | base64 -d  
eyJhbGciOiJSUzI1NiIsImtpZCI6IiJ9.eyJpc3MiOiJrdWJlcmt5ldGVzL3NlcnZpY2VhY2NvdW50Iiwia3ViZXJuZXRlc...  
XJuZXRlc...  
yZ...  
2aWN1YWNjb3VudC9zZWNyZXQubmFtZSI6Im5ldGFwcC11c2VyLXRva2VuLWJxd3piIwia3ViZXJuZXRlc...  
y9zZXJ2aWN1YWNjb3VudC9zZXJ2aWN1LWFjY291bnQubmFtZSI6Im5ldGFwcC11c2VyIwia3ViZXJuZXRlc...  
pby9zZXJ2aWN1YWNjb3VudC9zZXJ2aWN1LWFjY291bnQudWlkIjo...  
mFiZmR1YjYwND...  
YrHn4kYlb3gwxVKCLyo7p6J1f7mwwIgZqNw9eTvIk...  
dcwtFXHoEfMcOa6SIn40MVw1d5BGloaESn8150VCK3xES2DHA...  
aE-  
DHD59P1bIgPdioiKREgbOddKdMn6XTVsui...  
9nMd63JGdHL4hfXu6PPDxc9By6LgOW0nyaH4...  
_gexy4uIa61fNLKV2SKe4_gAN41ffOCKe4Tq8sa6zMo-8g
```

- With this token, you can visit the [Google Cloud Console](#) and log in to the cluster by clicking the login button and pasting in the token.

## Log in to cluster

Choose the method you want to use for authentication to the cluster

Token

Basic authentication

Authenticate with Identity Provider configured for the cluster

CLOSE    LOGIN

- After login is complete, you see a green check mark next to the cluster name, and information is

displayed about the physical environment. Clicking the cluster name displays more verbose information.

## 11. Install and Configure NetApp Trident Storage Provisioner: NetApp HCI with Anthos

Trident is a storage orchestrator for containers. With Trident, microservices and containerized applications can take advantage of enterprise-class storage services provided by the full NetApp portfolio of storage systems for persistent storage mounts. Depending on an application's requirements, Trident dynamically provisions storage for ONTAP-based products such as NetApp AFF and FAS systems and Element storage systems like NetApp SolidFire® and NetApp HCI.

To install Trident on the deployed user cluster and provision a persistent volume, complete the following steps:

1. Download the installation archive to the admin workstation and extract the contents. The current version of Trident is 19.10, which can be downloaded [here](#).

```
ubuntu@Anthos-Admin-Workstation:~$ wget
https://github.com/NetApp/trident/releases/download/v19.10.0/trident-installer-
19.10.0.tar.gz
--2019-11-07 16:45:33--
https://github.com/NetApp/trident/releases/download/v19.10.0/trident-installer-
19.10.0.tar.gz
Resolving github.com (github.com)... 140.82.118.4
Connecting to github.com (github.com)|140.82.118.4|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://github-production-release-asset-
2e65be.s3.amazonaws.com/77179634/4d3b5900-
Resolving github-production-release-asset-2e65be.s3.amazonaws.com (github-production-
release-asset-2e65be.s3.amazonaws.com)... 52.216.81.8
Connecting to github-production-release-asset-2e65be.s3.amazonaws.com (github-
production-release-asset-2e65be.s3.amazonaws.com)|52.216.81.8|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 68903585 (66M) [application/octet-stream]
Saving to: 'trident-installer-19.10.0.tar.gz.1'

trident-installer-19.10.0.tar 100%[=====] 65.71M 53.8MB/s   in 1.2s

2019-11-07 16:45:35 (53.8 MB/s) - 'trident-installer-19.10.0.tar.gz.1' saved
[68903585/68903585]

ubuntu@Anthos-Admin-Workstation:~$ tar -xf trident-installer-19.10.0.tar.gz
```

2. First set the location of the user cluster's **kubeconfig** file as an environment variable so that you don't have to reference it, because Trident has no option to pass this file.

```
ubuntu@Anthos-Admin-Workstation:~$ export KUBECONFIG=~/anthos-cluster01-kubeconfig
```

3. Navigate to the Trident directory and execute the tridentctl tool to install trident to your cluster. NetApp recommends installing Trident into its own namespace within the cluster. You can then verify that the install finished correctly.

```
ubuntu@Anthos-Admin-Workstation:~$ cd trident-installer --csi
ubuntu@Anthos-Admin-Workstation:~/trident-installer$ ./tridentctl install -n trident
INFO Starting Trident installation.           namespace=trident
INFO Created namespace.                      namespace=trident
INFO Created service account.
INFO Created cluster role.
INFO Created cluster role binding.
INFO Created custom resource definitions.    namespace=trident
INFO Added finalizers to custom resource definitions.
INFO Created Trident deployment.
INFO Waiting for Trident pod to start.
INFO Trident pod started.                   namespace=trident pod=trident-
79c76ff764-77sgl
INFO Waiting for Trident REST interface.
INFO Trident REST interface is up.          version=19.10.0
INFO Trident installation succeeded.
ubuntu@Anthos-Admin-Workstation:~/trident-installer$ ./tridentctl version -n trident
+-----+-----+
| SERVER VERSION | CLIENT VERSION |
+-----+-----+
| 19.10.0        | 19.10.0       |
+-----+-----+
```

4. The next step in enabling Trident integration with the NetApp HCI solution and Anthos is to create a backend that enables communication with the storage system. There are sample backend files available in the downloaded installation archive in the **sample-input** folder. Copy the **backend-solidfire.json** to your working directory and edit it to provide information detailing the storage system environment.

```
ubuntu@Anthos-Admin-Workstation:~/trident-installer$ cp sample-input/backend-
solidfire.json ./
ubuntu@Anthos-Admin-Workstation:~/trident-installer$ vi backend-solidfire.json
```

5. Edit the user, password, and MVIP value on the EndPoint line.

6. Edit the SVIP value.

```
{  
    "version": 1,  
    "storageDriverName": "solidfire-san",  
    "Endpoint": "https://trident:password@172.21.224.150/json-rpc/8.0",  
    "SVIP": "10.63.172.100:3260",  
    "TenantName": "trident",  
    "Types": [{"Type": "Bronze", "Qos": {"minIOPS": 1000, "maxIOPS": 2000, "burstIOPS": 4000}, {"Type": "Silver", "Qos": {"minIOPS": 4000, "maxIOPS": 6000, "burstIOPS": 8000}}, {"Type": "Gold", "Qos": {"minIOPS": 6000, "maxIOPS": 8000, "burstIOPS": 10000}}]  
}
```

7. With this back-end file in place, run the following command to create your first backend.

```
ubuntu@Anthos-Admin-Workstation:~/trident-installer$ ./tridentctl -n trident create  
backend -f backend.json  
+-----+-----+-----+  
+-----+  
|   NAME          | STORAGE DRIVER |           UUID           | STATE  |  
VOLUMES |  
+-----+-----+-----+  
+-----+  
| solidfire-backend | solidfire-san | a5f9e159-c8f4-4340-a13a-c615fef0f433 | online |  
0 |  
+-----+-----+-----+  
+-----+
```

8. With the backend created, you must next create a storage class. Just as with the backend, there is a sample storage class file that can be edited for the environment available in the sample-inputs folder. Copy it to the working directory and make necessary edits to reflect the backend created.

```
ubuntu@Anthos-Admin-Workstation:~/trident-installer$ cp sample-input/storage-class-  
csi.yaml.templ ./storage-class-basic.yaml  
ubuntu@Anthos-Admin-Workstation:~/trident-installer$ vi storage-class-basic.yaml
```

9. The only edit that must be made to this file is to define the `backendType` value to the name of the storage driver from the newly created backend. Also note the `name-field` value, which must be referenced in a later step.

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: basic-csi
  provisioner: csi.trident.netapp.io
parameters:
  backendType: "solidfire-san"
```

- Run the `kubectl` command to create the storage class.

```
ubuntu@Anthos-Admin-Workstation:~/trident-installer$ kubectl create -f sample-input/storage-class-basic.yaml
```

- With the storage class created, you must then create the first persistent volume claim (PVC). There is a sample `pvc-basic.yaml` file that can be used to perform this action located in `sample-inputs` as well. The only edit that must be made to this file is ensuring that the `storageClassName` field matches the one just created.

```
ubuntu@Anthos-Admin-Workstation:~/trident-installer$ vi sample-input/pvc-basic.yaml
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: basic
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 1Gi
  storageClassName: basic-csi
```

- Create the PVC by issuing the `kubectl` command, Creation can take some time depending on the size of the backing volume being created, so you can watch the process as it completes.

```
ubuntu@Anthos-Admin-Workstation:~/trident-installer$ kubectl create -f sample-input/pvc-basic.yaml
```

```
ubuntu@Anthos-Admin-Workstation:~/trident-installer$ kubectl get pvc --watch
NAME      STATUS    VOLUME                                     CAPACITY   ACCESS MODES
STORAGECLASS AGE
basic      Pending
basic      1s
basic      Pending   pvc-2azg0d2c-b13e-12e6-8d5f-5342040d22bf   0
basic      5s
basic      Bound    pvc-2azg0d2c-b13e-12e6-8d5f-5342040d22bf   1Gi        RWO
basic      7s
```

## Video Demos: NetApp HCI with Anthos

The following videos demonstrate some of the capabilities documented in this NVA.

- Deploying an application from the Google Cloud Application Marketplace to Anthos:  
► <https://docs.netapp.com/us-en/hci-solutions/media/Anthos-Deploy-App-Demo.mp4> (video)
- Dynamic scaling of Kubernetes clusters deployed on Anthos on VMware:  
► <https://docs.netapp.com/us-en/hci-solutions/media/Anthos-Scaling-Demo.mp4> (video)
- Using NetApp Trident to provision and attach a persistent volume to a Kubernetes pod on Anthos:  
► <https://docs.netapp.com/us-en/hci-solutions/media/Anthos-Trident-Demo.mp4> (video)

## Where to Find Additional Information: NetApp HCI with Anthos

To learn more about the information described in this document, review the following documents and/or websites:

- [Anthos Documentation](#)
- [NetApp HCI Documentation](#)
- [NetApp NDE 1.6 Deployment Guide](#)
- [NetApp Trident Documentation](#)
- [VMware vSphere 6.5 Documentation](#)
- [F5 Big-IP Documentation](#)

# NVA-1149: NetApp HCI for Red Hat OpenShift on Red Hat Virtualization

Alan Cowles and Nikhil M Kulkarni, NetApp

NetApp HCI for Red Hat OpenShift on Red Hat Virtualization (RHV) is a best-practice deployment guide for the fully automated install of Red Hat OpenShift through the Installer Provisioned Infrastructure (IPI) method onto the verified enterprise architecture of [NVA-1148: NetApp HCI with Red Hat Virtualization](#). The purpose of this NetApp Verified Architecture deployment guide is to provide a concise set of verified instructions to be followed for the deployment of the solution. The architecture and deployment methods described in this document have been validated jointly by subject matter experts at NetApp and Red Hat to provide a best-practice implementation of the solution.

## Use Cases

The NetApp HCI for Red Hat OpenShift on RHV solution is architected to deliver exceptional value for customers with the following use cases:

- Infrastructure to scale on demand with NetApp HCI
- Enterprise virtualized workloads in RHV
- Enterprise containerized workloads in Red Hat OpenShift

## Business Value

Enterprises are increasingly adopting DevOps practices to create new products, shorten release cycles, and rapidly add new features. Because of their innate agile nature, containers and microservices play a crucial role in supporting DevOps practices. However, practicing DevOps at a production scale in an enterprise environment presents its own challenges and imposes certain requirements on the underlying infrastructure, such as the following:

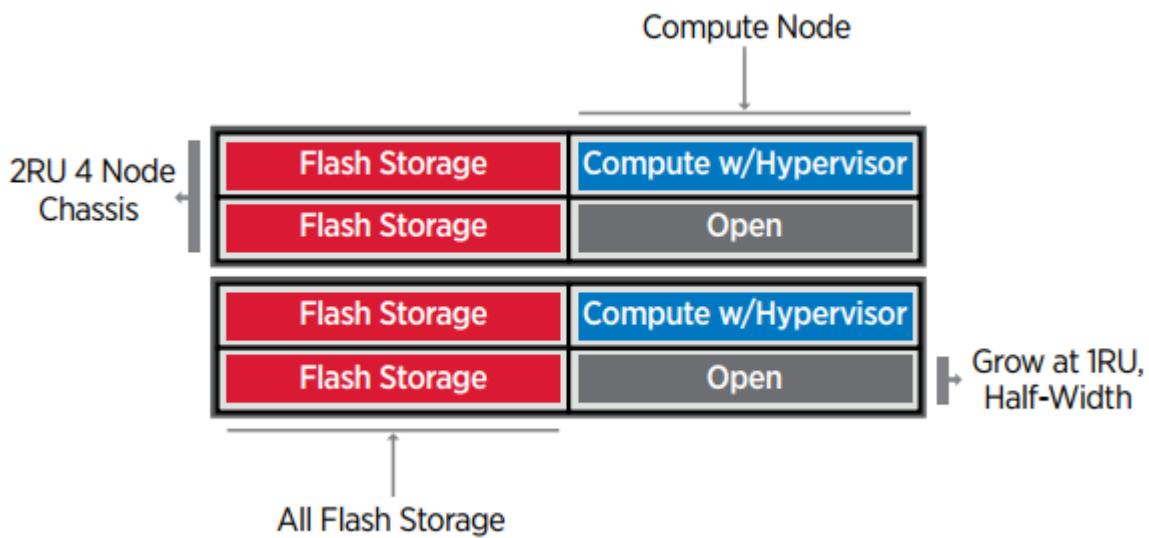
- High availability at all layers in the stack
- Ease of deployment procedures
- Nondisruptive operations and upgrades
- API-driven and programmable infrastructure to keep up with microservices agility
- Multitenancy with performance guarantees
- Ability to run virtualized and containerized workloads simultaneously
- Ability to scale infrastructure independently based on workload demands

NetApp HCI for Red Hat OpenShift on RHV acknowledges these challenges and presents a solution that helps address each concern by implementing the fully automated deployment of Red Hat OpenShift IPI on the RHV enterprise hypervisor. The remainder of this document details the components used in this verified architecture.

## Technology Overview

### NetApp HCI

NetApp HCI is an enterprise-scale, disaggregated hybrid cloud infrastructure (HCI) solution that delivers compute and storage resources in an agile, scalable, and easy-to-manage two-rack unit (2RU), four-node building block. It can also be configured with 1RU compute and server nodes. The minimum deployment depicted in the figure below consists of four NetApp HCI storage nodes and two NetApp HCI compute nodes. The compute nodes are installed as Red Hat Virtualization Hosts (RHV-H) hypervisors in a high-availability (HA) cluster. This minimum deployment can be easily scaled to fit customer enterprise workload demands by adding additional NetApp HCI storage or compute nodes to expand available resources.



The design for NetApp HCI for Red Hat Virtualization consists of the following components in a minimum starting configuration:

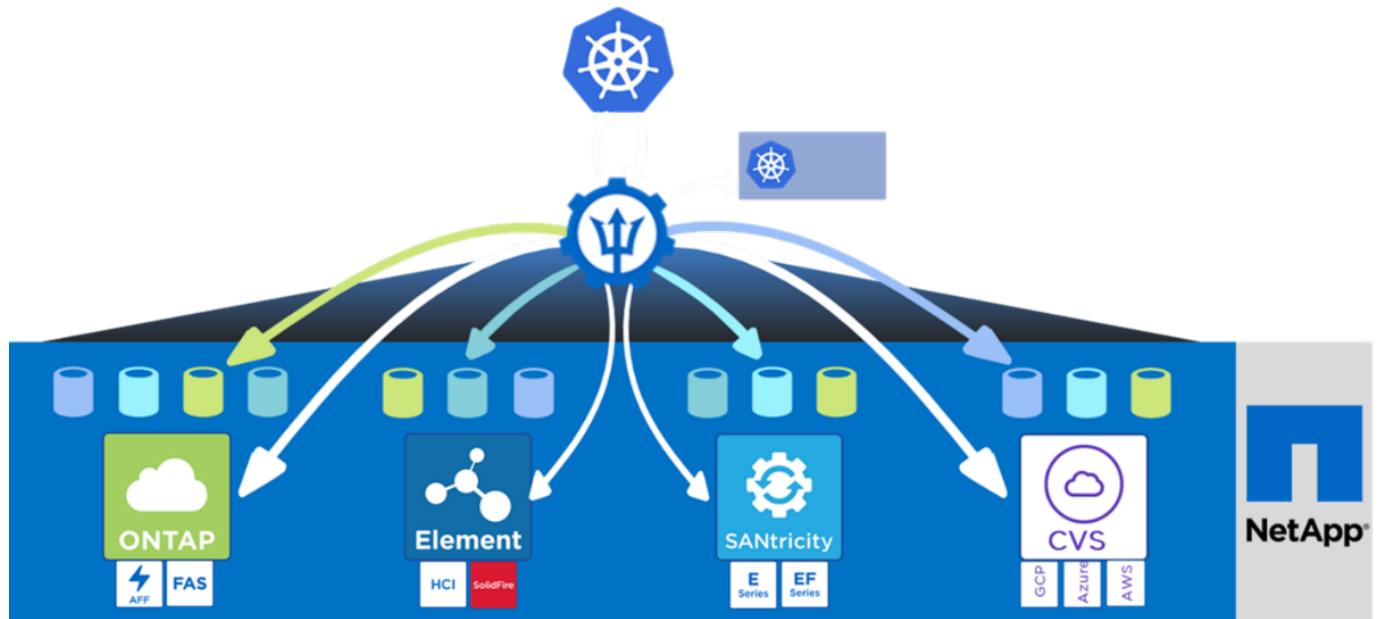
- NetApp H-Series all-flash storage nodes running NetApp Element software
- NetApp H-Series compute nodes running the Red Hat Virtualization RHV-H hypervisor

For more information about compute and storage nodes in NetApp HCI, see [NetApp HCI Datasheet](#).

### NetApp Trident

Trident is a NetApp open-source and fully supported storage orchestrator for containers and Kubernetes distributions, including Red Hat OpenShift. It works with the entire NetApp storage portfolio, including the NetApp Element storage system that is deployed as a part of the NetApp HCI solution. Trident provides the ability to accelerate the DevOps workflow by allowing end users to provision and manage storage from their NetApp storage systems, without requiring intervention from a storage administrator. An administrator can configure a number of storage backends based on project needs, and storage system models that allow for any number of advanced storage features, such as: compression, specific disk types, or QoS levels that guarantee a certain performance. After

they are defined, these backends can be leveraged by developers as part of their projects to create persistent volume claims (PVCs) and attach persistent storage to their containers on demand.



### Red Hat Virtualization

RHV is an enterprise virtual data center platform that runs on Red Hat Enterprise Linux (RHEL) and uses the KVM hypervisor.

For more information about RHV, see the [Red Hat Virtualization website](#).

RHV provides the following features:

- **Centralized management of VMs and hosts.** The RHV manager runs as a physical or virtual machine (VM) in the deployment and provides a web-based GUI for the management of the solution from a central interface.
- **Self-hosted engine.** To minimize the hardware requirements, RHV allows RHV Manager (RHV-M) to be deployed as a VM on the same hosts that run guest VMs.
- **High availability.** In event of host failures, to avoid disruption, RHV allows VMs to be configured for high availability. The highly available VMs are controlled at the cluster level using resiliency policies.
- **High scalability.** A single RHV cluster can have up to 200 hypervisor hosts enabling it to support requirements of massive VMs to hold resource-greedy, enterprise-class workloads.
- **Enhanced security.** Inherited from RHV, Secure Virtualization (sVirt) and Security Enhanced Linux (SELinux) technologies are employed by RHV for the purposes of elevated security and hardening for the hosts and VMs. The key advantage from these features is logical isolation of a VM and its associated resources.

## **Red Hat Virtualization Manager**

RHV-M provides centralized enterprise-grade management for the physical and logical resources within the RHV virtualized environment. A web-based GUI with different role-based portals are provided to access RHV-M features.

RHV-M exposes configuration and management of RHV resources via open-source, community-driven RESTful API. It also supports full-fledged integration with Red Hat CloudForms and Red Hat Ansible for automation and orchestration.

## **Red Hat Virtualization Hosts**

Hosts (also called hypervisors) are the physical servers that provide hardware resources for the VMs to run on. Kernel-based Virtual Machine (KVM) provides full virtualization support, and Virtual Desktop Server Manager (VDSM) is the host agent that is responsible for communication of the hosts with the RHV-M.

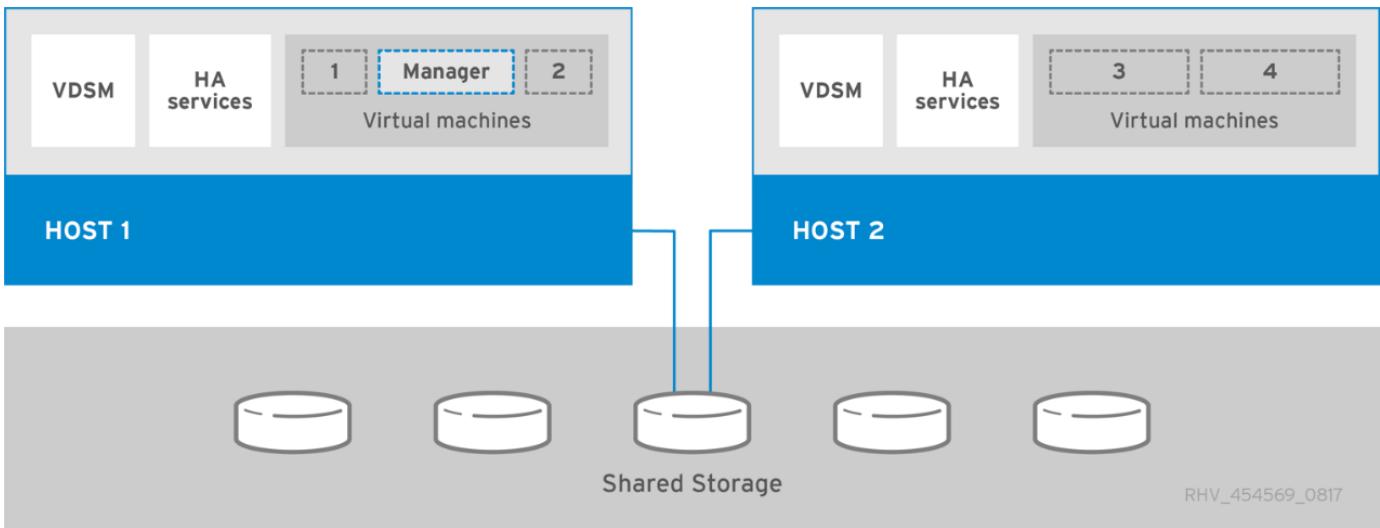
Two types of hosts are supported in RHV are RHV-H and RHEL hosts:

- RHV-H is a light-weight minimal operating system based on RHEL, optimized for ease of setting up physical servers as RHV hypervisors.
- RHEL hosts are servers that run the standard RHEL operating system and are later configured with the required subscriptions to install the packages required to permit the physical servers to be used as RHV hosts.

## **Red Hat Virtualization Architecture**

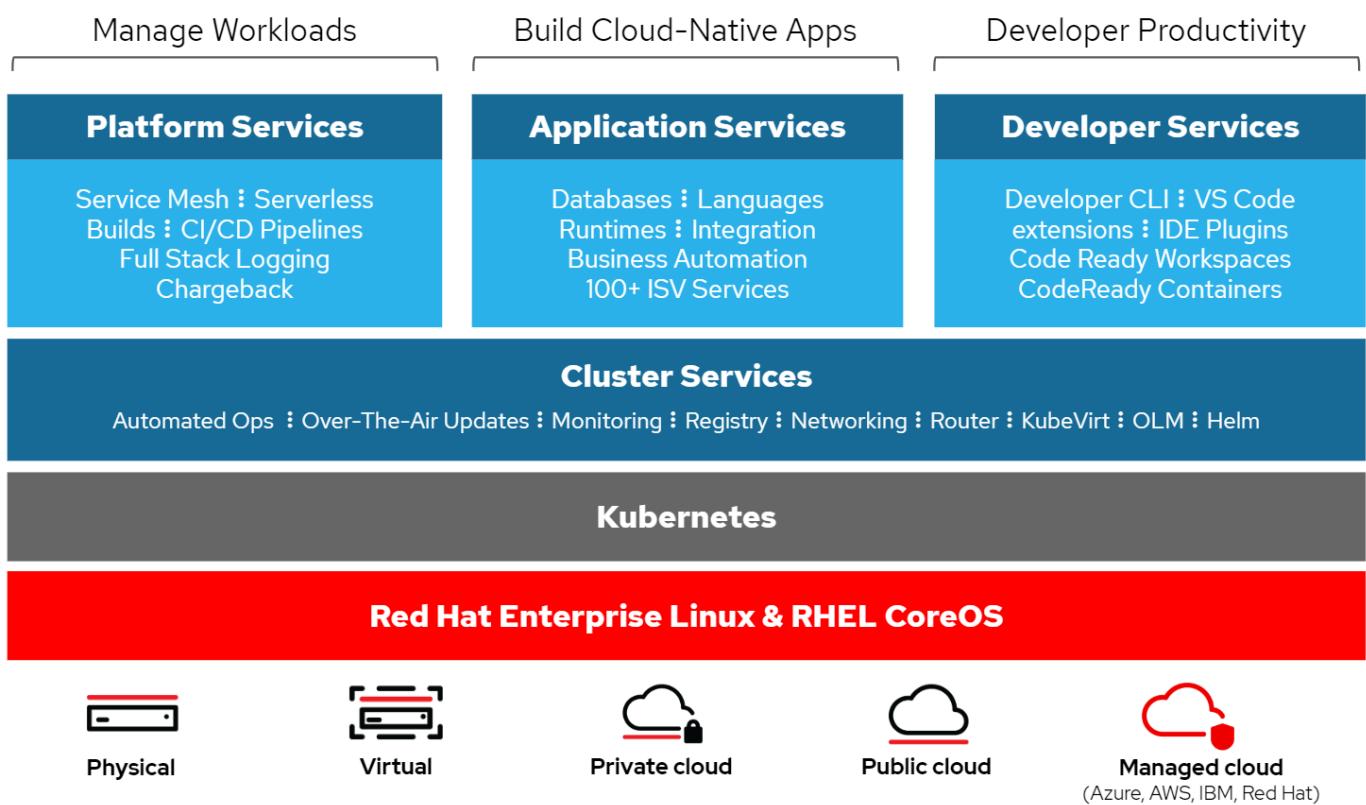
RHV can be deployed in two different architectures: with the RHV-M as a physical server in the infrastructure or with the RHV-M configured as a self-hosted engine. The self-hosted engine deployment, where the RHV-M is a VM hosted in the same environment as other VMs, is recommended and used specifically in this deployment guide.

A minimum of two self-hosted nodes are required for high availability of guest VMs and RHV-M as depicted in the figure below. For ensuring the high availability of the manager VM, HA services are enabled and run on all the self-hosted engine nodes.



### Red Hat OpenShift Container Platform

Red Hat OpenShift Container Platform is a fully supported enterprise Kubernetes platform. Red Hat makes several enhancements to open-source Kubernetes to deliver an application platform with all the components fully integrated to build, deploy, and manage containerized applications. With Red Hat OpenShift 4.4, the installation and management processes have been streamlined through the IPI method which has been deployed in this solution. By leveraging this deployment method, a fully functional OpenShift cluster providing metering and monitoring at both the cluster and application level can be fully configured and deployed on top of Red Hat Virtualization in less than an hour. OpenShift nodes are based upon RHEL CoreOS, an immutable system image designed to run containers, based on RHEL, which can be upgraded or scaled easily on demand as the needs of the end user require, helping to deliver the benefits of the public cloud to the local data center.



## Architectural Overview: NetApp HCI for Red Hat OpenShift on RHV

### Hardware Requirements

The following table lists the minimum number of hardware components that are required to implement the solution. The hardware components that are used in specific implementations of the solution might vary based on customer requirements.

Hardware	Model	Quantity
NetApp HCI compute nodes	NetApp H410C	2
NetApp HCI storage nodes	NetApp H410S	4
Data switches	Mellanox SN2010	2
Management switches	Cisco Nexus 3048	2

### Software Requirements

The following table lists the software components that are required to implement the solution. The software components that are used in any implementation of the solution might vary based on customer requirements.

Software	Purpose	Version
NetApp HCI	Infrastructure (compute/storage)	1.8

Software	Purpose	Version
NetApp Element	Storage	12.0
NetApp Trident	Storage orchestration	20.04
RHV	Virtualization	4.3.9
Red Hat OpenShift	Container orchestration	4.4.6

## Design Considerations: NetApp HCI for Red Hat OpenShift on RHV

### Network Design

The Red Hat OpenShift on RHV on HCI solution uses two data switches to provide primary data connectivity at 25Gbps. It also uses two additional management switches that provide connectivity at 1Gbps for in-band management for the storage nodes and out-of-band management for IPMI functionality. OCP uses the logical network on the RHV for the cluster management. This section describes the arrangement and purpose of each virtual network segment used in the solution and outlines the pre-requisites for deployment of the solution.

### VLAN Requirements

The NetApp HCI for Red Hat OpenShift on RHV solution is designed to logically separate network traffic for different purposes by using virtual local area networks (VLANs). NetApp HCI requires a minimum of three network segments. However, this configuration can be scaled to meet customer demands or to provide further isolation for specific network services. The following table lists the VLANs that are required to implement the solution, as well as the specific VLAN IDs that are used later in the verified architecture deployment.

VLANs	Purpose	VLAN ID
Out-of-band management network	Management for HCI nodes and IPMI	16
In-band management network	Management for HCI nodes, ovirtmgmt, and VMs	1172
Storage network	Storage network for NetApp Element	3343
Migration network	Network for virtual guest migration	3345

### Network Infrastructure Support Resources

The following infrastructure should be in place prior to the deployment of the OpenShift Container Platform (OCP) on Red Hat Virtualization on NetApp HCI solution:

- At least one DNS server which provides a full host-name resolution that is accessible from the in-

band management network and the VM network.

- At least one NTP server that is accessible from the in-band management network and the VM network.
- (Optional) Outbound internet connectivity for both the in-band management network and the VM network.
- RHV cluster should have at least 28x vCPUs, 112GB RAM, and 840GB of available storage (depending on the production workload requirements).

## Deploying NetApp HCI for Red Hat OpenShift on RHV

### Deployment Summary: NetApp HCI for Red Hat OpenShift on RHV

The detailed steps provided in this section provide a validation for the minimum hardware and software configuration required to deploy and validate the NetApp HCI for Red Hat OpenShift on RHV solution.

Deploying Red Hat OpenShift Container Platform through IPI on Red Hat Virtualization consists of the following steps:

1. Create storage network VLAN
2. Download OpenShift installation files
3. Download CA cert from RHV
4. Register API/Apps in DNS
5. Generate and add SSH private key
6. Install OpenShift Container Platform
7. Access console/web console
8. Configure worker nodes to run storage services
9. Download and install Trident through Operator

### 1. Create Storage Network VLAN: NetApp HCI for Red Hat OpenShift on RHV

To create a storage network VLAN, complete the following steps:

To support Element storage access for NetApp Trident to attach persistent volumes to pods deployed in OpenShift, the machine network being used for each worker in the OCP deployment must be able to reach the storage resources. If the machine network cannot access the Element storage network by default, an additional network/VLAN can be created in the Element cluster to allow access:

1. Using any browser, log in to the Element Cluster at the cluster's MVIP.
2. Navigate to Cluster > Network and click Create VLAN.

3. Before you provide the details, reserve at least five IP addresses from the network that is reachable from the OCP network (one for the virtual network storage VIP and one for virtual network IP on each storage node).

Enter a VLAN name of your choice, enter the VLAN ID, SVIP, and netmask, select the Enable VRF option, and enter the gateway IP for the network. In the IP address blocks, enter the starting IP of the other addresses reserved for the storage nodes. In this example, the size is four because there are four storage nodes in this cluster. Click Create VLAN.

## Create a New VLAN

X

VLAN Name

VLAN Tag

SVIP

Netmask

Enable VRF

Gateway

Description

IP Address Blocks

Starting IP

Size

Add A Block

**Create VLAN**

**Cancel**

## 2. Download OpenShift Installation Files: NetApp HCI for Red Hat OpenShift on RHV

To download the OpenShift installation files, complete the following steps:

1. Go to the [Red Hat login page](#) and log in with your Red Hat credentials.
2. On the Clusters page, click Create Cluster.

The screenshot shows the Red Hat OpenShift Cluster Manager web interface. On the left, there's a sidebar with a navigation menu. The 'Clusters' option is highlighted with a blue underline. Other options include 'Red Hat OpenShift Cluster Manager', 'Subscriptions', 'Documentation', 'Support Cases', 'Cluster Manager Feedback', and 'Red Hat Marketplace'. The main content area has a large red circular icon with a white arrow. Below it, the text 'No OpenShift clusters to display' is centered. A descriptive paragraph explains the purpose of the Cluster Manager. At the bottom of the main area are three buttons: 'Create cluster' (in a blue box), 'Register cluster', and 'View archived clusters'.

### 3. Select OpenShift Container Platform.

[Clusters](#) > [Create](#)

#### Create a Cluster to Get Started

This screenshot shows the 'Create a Cluster to Get Started' page. It features two main options side-by-side. The left option is for 'Red Hat OpenShift Container Platform', represented by a red hat icon and the text 'Red Hat OpenShift Container Platform'. Below this, a description reads: 'Create an OCP cluster using the command-line installer. Your cluster will automatically register to the Cluster Manager after installation completes.' The right option is for 'Red Hat OpenShift Dedicated', represented by a black hat icon and the text 'Red Hat OpenShift Dedicated'. Below this, a description reads: 'Create a Red Hat-managed cluster (OSD), provisioned on Amazon Web Services or Google Cloud Platform.'

### 4. Select Run on Red Hat Virtualization.

## Install OpenShift Container Platform 4

Select an infrastructure provider

Run on Amazon Web Services	Run on Microsoft Azure	Run on Google Cloud Platform	Run on VMware vSphere
Red Hat OpenStack Platform Run on Red Hat OpenStack	Red Hat Virtualization Run on Red Hat Virtualization	Run on Bare Metal	IBM Z IBM LinuxONE Run on IBM Z
Power Systems Run on Power	Run on Laptop Powered by Red Hat CodeReady Containers		

- The next page allows you to download the OpenShift installer (available for Linux and MacOS), a unique pull secret that is required to create the `install-config` file and the `oc` command-line tools (available for Linux, Windows, and MacOS).

Download the files, transfer them to a RHEL administrative workstation from where you can run the OpenShift installation, or download these files directly using wget or curl on a RHEL administrative workstation.

### Downloads

**OpenShift installer**

Download and extract the install program for your operating system and place the file in the directory where you will store the installation configuration files. Note: The OpenShift install program is only available for Linux and macOS at this time.

Linux ▾ [Download installer](#)

**Pull secret**

Download or copy your pull secret. The install program will prompt you for your pull secret during installation.

[Download pull secret](#)  [Copy pull secret](#)

**Command-line interface**

Download the OpenShift command-line tools and add them to your PATH.

Linux ▾ [Download command-line tools](#)

When the installer is complete you will see the console URL and credentials for accessing your new cluster. A `kubeconfig` file will also be generated for you to use with the `oc` CLI tools you downloaded.

### 3. Download CA Certificate from RHV: NetApp HCI for Red Hat OpenShift on RHV

To download the CA certificate from RHV, complete the following steps:

1. In order to access the RHV manager from the RHEL machine during the deployment process, the CA certificate trust must be updated on the machine to trust connections to RHV-M. To download the RHV Manager's CA certificate, run the following commands:

```
sudo curl -k 'https://<engine-fqdn>/ovirt-engine/services/pki-resource?resource=ca-certificate&format=X509-PEM-CA' -o /tmp/ca.pem  
[user@rhel7 ~]$ sudo curl -k 'https://rhv-m.cie.netapp.com/ovirt-engine/services/pki-resource?resource=ca-certificate&format=X509-PEM-CA' -o /tmp/ca.pem  
% Total    % Received % Xferd  Average Speed   Time     Time      Time  Current  
          Dload  Upload Total Spent   Left Speed  
100  1376  100  1376    0     0  9685      0 --:--:-- --:--:-- --:--:--  9690
```

2. Copy the CA certificate to the directory for server certificates and update the CA trust.

```
[user@rhel7 ~]$ sudo cp /tmp/ca.pem /etc/pki/ca-trust/source/anchors/ca.pem  
[user@rhel7 ~]$ sudo update-ca-trust
```

#### 4. Register API/Apps in DNS: NetApp HCI for Red Hat OpenShift on RHV

To register API/Apps in DNS, complete the following steps:

1. Reserve three static IP addresses from the network being used for OCP: the first IP address for OpenShift Container Platform REST API, the second IP address for pointing to the wildcard application ingress, and the third IP address for the internal DNS service. The first two IPs require an entry in the DNS server.



The default value of the `machineNetwork` subnet as created by IPI during OpenShift install is `10.0.0.0/16`. If the IPs you intend to use for your cluster's management network fall outside of this range, you might need to customize your deployment and edit these values before deploying the cluster. For more information, see the section [Use a Custom Install File for OpenShift Deployment](#).

2. Configure the API domain name by using the format `api.<openshift-cluster-name>.<base-domain>` pointing to the reserved IP.

## New Host

X

Name (uses parent domain name if blank):

Fully qualified domain name (FQDN):

IP address:

Create associated pointer (PTR) record

Allow any authenticated user to update DNS records with the same owner name

Add Host

Cancel

3. Configure the wildcard application ingress domain name by using the format \*.apps.<openshift-cluster-name>.<base-domain> pointing to the reserved IP.

New Host X

Name (uses parent domain name if blank):  
\*.apps.rhv-ocp-cluster

Fully qualified domain name (FQDN):  
\*.apps.rhv-ocp-cluster.cie.netapp.com.

IP address:  
10.63.172.152

Create associated pointer (PTR) record  
 Allow any authenticated user to update DNS records with the same owner name

Add Host Cancel

## 5. Generate and Add SSH Private Key: NetApp HCI for Red Hat OpenShift on RHV

To generate and add an SSH private key, complete the following steps:

1. For the installation debugging or disaster recovery on the OpenShift cluster, you must provide an SSH key to both the **ssh-agent** and the installation program. Create an SSH key if one does not already exist for password-less authentication on the RHEL machine.

```
[user@rhel7 ~]$ ssh-keygen -t rsa -b 4096 -N '' -f ~/.ssh/id_rsa
```

2. Start the **ssh-agent** process and configure it as a background running task.

```
[user@rhel7 ~]$ eval "$(ssh-agent -s)"  
Agent pid 31874
```

3. Add the SSH private key that you created in step 2 to the **ssh-agent**, which enables you to SSH

directly to the nodes without having to interactively pass the key.

```
[user@rhel7 ~]$ ssh-add ~/.ssh/id_rsa
```

## 6. Install OpenShift Container Platform: NetApp HCI for Red Hat OpenShift on RHV

To install OpenShift Container Platform, compete the following steps:

1. Create a directory for OpenShift installation and transfer the downloaded files to it. Extract the OpenShift installer files from the tar archive.

```
[user@rhel7 ~]$ mkdir openshift-deploy
[user@rhel7 ~]$ cd openshift-deploy
[user@rhel7 openshift-deploy]$ tar xvf openshift-install-linux.tar.gz
README.md
openshift-install
[user@rhel7 openshift-deploy]$ ls -la
total 453260
drwxr-xr-x. 2 user user      146 May 26 16:01 .
dr-xr-x---. 16 user user     4096 May 26 15:58 ..
-rw-r--r--. 1 user user  25249648 May 26 15:59 openshift-client-linux.tar.gz
-rwxr-xr-x. 1 user user 354664448 Apr 27 01:37 openshift-install
-rw-r--r--. 1 user user  84207215 May 26 16:00 openshift-install-linux.tar.gz
-rw-r--r--. 1 user user    2736 May 26 15:59 pull-secret.txt
-rw-r--r--. 1 user user     706 Apr 27 01:37 README.md
```



The installation program creates several files in the directory used for installation of the cluster. Both the installation program and the files created by the installation program must be kept even after the cluster is up.



The binary files that you previously downloaded, such as `openshift-install` or `oc`, can be copied to a directory that is in the user's path (for example, `/usr/local/bin`) to make them easier to run.

2. Create the cluster by running the `openshift-install create cluster` command and respond to the installation program prompts. Pass the SSH public key, select ovirt from the platform, provide the RHV infrastructure details, provide the three reserved IP addresses and the downloaded pull secret to the installation program prompts. After all the inputs are provided, the installation program creates and configures a bootstrap machine with a temporary Kubernetes control plane which then creates and configures the master VMs with the production Kubernetes control plane. The control plane on the master nodes creates and configures the worker VMs.

It can take approximately 30–45 minutes to get the complete cluster up and running.

```
[user@rhel7 openshift-deploy]$ ./openshift-install create cluster
--dir=/home/user/openshift-deploy --log-level=info
? SSH Public Key
/home/user/.ssh/id_rsa.pub
? Platform ovirt
? oVirt cluster Default
? oVirt storage domain data_domain
? oVirt network ovirtmgmt
? Internal API virtual IP 10.63. 172.151
? Internal DNS virtual IP 10.63. 172.153
? Ingress virtual IP 10.63. 172.152
? Base Domain cie.netapp.com
? Cluster Name rhv-ocp-cluster
? Pull Secret [? for help]
*****
*****
*****
*****
*****
```

INFO Obtaining RHCOS image file from 'https://releases-art-rhcos.svc.ci.openshift.org/art/storage/releases/rhcos-4.4/44.81.202004250133-0/x86\_64/rhcos-44.81.202004250133-0-openstack.x86\_64.qcow2.gz?sha256=f8a44e0ea8cc45882dc22eb632a63afb90b414839b8aa92f3836ede001dfe9cf'

INFO The file was found in cache: /home/user/.cache/openshift-installer/image\_cache/e263efbc53c0caf612bcfaad10e3dff0. Reusing...

INFO Creating infrastructure resources...

INFO Waiting up to 20m0s for the Kubernetes API at https://api.rhv-ocp-cluster.cie.netapp.com:6443...

INFO API v1.17.1 up

INFO Waiting up to 40m0s for bootstrapping to complete...

INFO Destroying the bootstrap resources...

INFO Waiting up to 30m0s for the cluster at https://api.rhv-ocp-cluster.cie.netapp.com:6443 to initialize...

INFO Waiting up to 10m0s for the openshift-console route to be created...

INFO Install complete!

INFO To access the cluster as the system:admin user when using 'oc', run 'export KUBECONFIG=/home/user/openshift-deploy/auth/kubeconfig'

INFO Access the OpenShift web-console here: https://console-openshift-console.apps.rhv-ocp-cluster.cie.netapp.com

INFO Login to the console with user: kubeadmin, password: NtsqU-p3qUb-8Hscu-JfAq7

3. When the cluster deployment is complete, the directions for accessing the OpenShift cluster, including a link to its web console and credentials for the kubeadmin user, are displayed. Make sure to take a note of these details.
4. Log in to the RHV Manager and observe that the VMs relating to the OCP cluster are up and running.

Name	Comment	Host	IP Addresses	FQDN	Cluster	Data Center	Memory	CPU	Network	Graphics	Status	Uptime
HostedEngine		rhv-h02.cie.netapp.o	10.63.172.150 fe8...	rhv-m.cie.netapp....	Default	Default	30%	15%	0%	SPICE + ...	Up	5 day
NetApp-mNode		rhv-h02.cie.netapp.o			Default	Default	24%	2%	0%	SPICE + ...	Up	25 mi
rhv-ocp-cluster-hdr7k-master-0		rhv-h01.cie.netapp.o			Default	Default	69%	53%	0%	SPICE + ...	Up	1 h
rhv-ocp-cluster-hdr7k-master-1		rhv-h02.cie.netapp.o			Default	Default	50%	35%	0%	SPICE + ...	Up	1 h
rhv-ocp-cluster-hdr7k-master-2		rhv-h01.cie.netapp.o			Default	Default	59%	51%	0%	SPICE + ...	Up	1 h
rhv-ocp-cluster-hdr7k-worker-0-ghskz		rhv-h02.cie.netapp.o			Default	Default	16%	16%	0%	SPICE + ...	Up	1 h
rhv-ocp-cluster-hdr7k-worker-0-xd199		rhv-h01.cie.netapp.o			Default	Default	14%	12%	0%	SPICE + ...	Up	1 h
rhv-ocp-cluster-hdr7k-worker-0-zkmt		rhv-h02.cie.netapp.o			Default	Default	15%	14%	0%	SPICE + ...	Up	1 h
tmpvm-for-rhv-ocp-cluster-hdr7k-rhcos					Default	Default	--	--	--	None	Down	

## 7. Access Console/Web Console: NetApp HCI for Red Hat OpenShift on RHV

To access the console or web console, complete the following steps:

1. To access the OCP cluster through the CLI, extract the `oc` command-line tools tar file and place its content in a directory that is in the user's path.

```
[user@rhel7 openshift-deploy]$ tar xvf openshift-client-linux.tar.gz
README.md
oc
kubectl
[user@rhel7 openshift-deploy]$ echo $PATH
/usr/local/bin: /usr/local/sbin:/sbin:/bin:/usr/sbin:/usr/bin

[user@rhel7 openshift-deploy]$ cp oc /usr/local/bin
```

2. To interact with the cluster through the CLI, you can use the `kubeconfig` file provided by the IPI process located in the `/auth` directory inside the folder from where you launched the installation program. To easily interact with the cluster, export the file that is created in the directory. After a successful cluster deployment, the file location and the following command are displayed.

```
[user@rhel7 openshift-deploy]$ export KUBECONFIG=/home/user/openshift-
deploy/auth/kubeconfig
```

3. Verify whether you have access to the cluster and whether the nodes are in the Ready state.

```
[user@rhel7 openshift-deploy]$ oc get nodes
NAME                               STATUS  ROLES   AGE    VERSION
rhv-ocp-cluster-hdr7k-master-0     Ready   master  93m   v1.17.1
rhv-ocp-cluster-hdr7k-master-1     Ready   master  93m   v1.17.1
rhv-ocp-cluster-hdr7k-master-2     Ready   master  93m   v1.17.1
rhv-ocp-cluster-hdr7k-worker-0-ghskz Ready   worker  83m   v1.17.1
rhv-ocp-cluster-hdr7k-worker-0-xdl99 Ready   worker  86m   v1.17.1
rhv-ocp-cluster-hdr7k-worker-0-zkxmt Ready   worker  85m   v1.17.1
```

4. Log in to the web console URL by using the credentials, both of which were provided after the successful deployment of the cluster, and then verify GUI access to the cluster.

The screenshot shows the Red Hat OpenShift Container Platform web interface. The top navigation bar includes the Red Hat logo, 'OpenShift Container Platform', and a user dropdown set to 'kube:admin'. The left sidebar has a 'Administrator' section with links for Home, Operators, Workloads, Networking, and Storage. The main content area is titled 'Overview' under the 'Cluster' section. It displays a 'Status' summary with three items: 'Cluster' (green checkmark), 'Control Plane' (green checkmark), and 'Operators' (green checkmark). A blue banner at the top of the content area states: 'You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.'

## 8. Configure Worker Nodes to Run Storage Services: NetApp HCI for Red Hat OpenShift on RHV

To configure the worker nodes to run storage services, complete the following steps:

1. To access storage from the Element system, each of the worker nodes must have iSCSI available and running as a service. To create a machine configuration that can enable and start the `iscisd` service, log in to the OCP web console and navigate to Compute > Machine Configs and click Create Machine Config. Paste the YAML file and click Create.

## Create Machine Config

Create by manually entering YAML or JSON definitions, or by dragging and dropping a file into the editor.

[View shortcuts](#)

```
1  apiVersion: machineconfiguration.openshift.io/v1
2  kind: MachineConfig
3  metadata:
4    labels:
5      | machineconfiguration.openshift.io/role: worker
6      name: worker-iscsi-configuration
7  spec:
8    config:
9      ignition:
10     version: 2.2.0
11     systemd:
12       units:
13         - name: iscsid.service
14           enabled: true
15           state: started
16   osImageURL: ""
```

[Create](#)

[Cancel](#)

[Download](#)

2. After the configuration is created, it will take approximately 20–30 minutes to apply the configuration to the worker nodes and reload them. Verify whether the machine config is applied by using `oc get mcp` and make sure that the machine config pool for workers is updated. You can also log in to the worker nodes to confirm that the iscsid service is running.

```
[user@rhel7 openshift-deploy]$ oc get mcp
NAME      CONFIG                                     UPDATED   UPDATING   DEGRADED
master    rendered-master-a520ae930e1d135e0dee7168  True     False      False
worker    rendered-worker-de321b36eeba62df41feb7bc  True     False      False
[user@rhel7 openshift-deploy]$ ssh core@10.63.172.22 sudo systemctl status iscsid
● iscsid.service - Open-iSCSI
   Loaded: loaded (/usr/lib/systemd/system/iscsid.service; enabled; vendor preset: disabled)
     Active: active (running) since Tue 2020-05-26 13:36:22 UTC; 3 min ago
       Docs: man:iscsid(8)
              man:iscsiadm(8)
   Main PID: 1242 (iscsid)
     Status: "Ready to process requests"
      Tasks: 1
     Memory: 4.9M
        CPU: 9ms
      CGroup: /system.slice/iscsid.service
              └─1242 /usr/sbin/iscsid -f
```



It is also possible to confirm that the MachineConfig has been successfully applied and services have been started as expected by running the `oc debug` command with the appropriate flags.

## 9. Download and Install NetApp Trident: NetApp HCI for Red Hat OpenShift on RHV

To download and install NetApp Trident, complete the following steps:

1. Make sure that the user that is logged in to the OCP cluster has sufficient privileges for installing Trident.

```
[user@rhel7 openshift-deploy]$ oc auth can-i '*' '*' --all-namespaces
yes
```

2. Verify that you can download an image from the registry and access the MVIP of the NetApp Element cluster.

```
[user@rhel7 openshift-deploy]$ oc run -i --tty ping --image=busybox --restart=Never  
--rm -- ping 10.63.172.140  
If you don't see a command prompt, try pressing enter.  
64 bytes from 10.63.172.140: seq=1 ttl=63 time=0.312 ms  
64 bytes from 10.63.172.140: seq=2 ttl=63 time=0.271 ms  
64 bytes from 10.63.172.140: seq=3 ttl=63 time=0.254 ms  
64 bytes from 10.63.172.140: seq=4 ttl=63 time=0.309 ms  
64 bytes from 10.63.172.140: seq=5 ttl=63 time=0.319 ms  
64 bytes from 10.63.172.140: seq=6 ttl=63 time=0.303 ms  
^C  
--- 10.63.172.140 ping statistics ---  
7 packets transmitted, 7 packets received, 0% packet loss  
round-trip min/avg/max = 0.254/0.387/0.946 ms  
pod "ping" deleted
```

3. Download the Trident installer bundle using the following commands and extract it to a directory.

```
[user@rhel7 ~]$ wget  
[user@rhel7 ~]$ tar -xf trident-installer-20.04.0.tar.gz  
[user@rhel7 ~]$ cd trident-installer
```

4. The Trident installer contains manifests for defining all the required resources. Using the appropriate manifests, create the TridentProvisioner custom resource definition.

```
[user@rhel7 trident-installer]$ oc create -f  
deploy/crds/trident.netapp.io_tridentprovisioners_crd_post1.16.yaml  
  
customresourcedefinition.apiextensions.k8s.io/tridentprovisioners.trident.netapp.io  
created
```

5. Create a Trident namespace, which is required for the Trident operator.

```
[user@rhel7 trident-installer]$ oc create namespace trident  
namespace/trident created
```

6. Create the resources required for the Trident operator deployment, such as a ServiceAccount for the operator, a ClusterRole and ClusterRoleBinding to the ServiceAccount, a dedicated PodSecurityPolicy, or the operator itself.

```
[user@rhel7 trident-installer]$ oc kustomize deploy/ > deploy/bundle.yaml
[user@rhel7 trident-installer]$ oc create -f deploy/bundle.yaml
serviceaccount/trident-operator created
clusterrole.rbac.authorization.k8s.io/trident-operator created
clusterrolebinding.rbac.authorization.k8s.io/trident-operator created
deployment.apps/trident-operator created
podsecuritypolicy.policy/tridentoperatorpods created
```

7. Verify that the Trident operator is deployed.

```
[user@rhel7 trident-installer]$ oc get deployment -n trident
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
trident-operator   1/1     1           1          56s
[user@rhel7 trident-installer]$ oc get pods -n trident
NAME                           READY   STATUS    RESTARTS   AGE
trident-operator-564d7d66f-qrz7v   1/1     Running   0          71s
```

8. After the Trident operator is installed, install Trident using this operator. In this example, TridentProvisioner custom resource (CR) was created. The Trident installer comes with definitions for creating a TridentProvisioner CR. These can be modified based on the requirements.

```
[user@rhel7 trident-installer]$ oc create -f deploy/crds/tridentprovisioner_cr.yaml
tridentprovisioner.trident.netapp.io/trident created
```

9. Approve the Trident serving CSR certificates by using `oc get csr -o name | xargs oc adm certificate approve`.

```
[user@rhel7 trident-installer]$ oc get csr -o name | xargs oc adm certificate approve
certificatesigningrequest.certificates.k8s.io/csr-4b7zh approved
certificatesigningrequest.certificates.k8s.io/csr-4hkwc approved
certificatesigningrequest.certificates.k8s.io/csr-5bgh5 approved
certificatesigningrequest.certificates.k8s.io/csr-5g4d6 approved
certificatesigningrequest.certificates.k8s.io/csr-5j9hz approved
certificatesigningrequest.certificates.k8s.io/csr-5m8qb approved
certificatesigningrequest.certificates.k8s.io/csr-66hv2 approved
certificatesigningrequest.certificates.k8s.io/csr-6rdgg approved
certificatesigningrequest.certificates.k8s.io/csr-6t24f approved
certificatesigningrequest.certificates.k8s.io/csr-76wgv approved
certificatesigningrequest.certificates.k8s.io/csr-78qsq approved
certificatesigningrequest.certificates.k8s.io/csr-7r58n approved
certificatesigningrequest.certificates.k8s.io/csr-8ghmk approved
certificatesigningrequest.certificates.k8s.io/csr-8sn5q approved
```

10. Verify that Trident 20.04 is installed by using the TridentProvisioner CR, and verify that the pods related to Trident are.

```
[user@rhel7 trident-installer]$ oc get tprov -n trident
NAME      AGE
trident   9m49s

[user@rhel7 trident-installer]$ oc describe tprov trident -n trident
Name:          trident
Namespace:     trident
Labels:        <none>
Annotations:   <none>
API Version:  trident.netapp.io/v1
Kind:          TridentProvisioner
Metadata:
  Creation Timestamp: 2020-05-26T18:49:19Z
  Generation:        1
  Resource Version:  640347
  Self Link:
    /apis/trident.netapp.io/v1/namespaces/trident/tridentprovisioners/trident
  UID:               52656806-0414-4ed8-b355-fc123fafbf4e
Spec:
  Debug:  true
Status:
  Message: Trident installed
  Status:  Installed
  Version: v20.04
Events:
  Type  Reason  Age           From           Message
  ----  -----  --  -----
  Normal  Installing  9m32s       trident-operator.netapp.io  Installing
Trident
  Normal  Installed  3m47s (x5 over 8m56s)  trident-operator.netapp.io  Trident
installed

[user@rhel7 trident-installer]$ oc get pods -n trident
NAME                           READY  STATUS    RESTARTS  AGE
trident-csi-7f769c7875-s6fmt  5/5    Running   0          10m
trident-csi-cp7wg              2/2    Running   0          10m
trident-csi-hhx94              2/2    Running   0          10m
trident-csi-l72bt              2/2    Running   0          10m
trident-csi-xfl9d              2/2    Running   0          10m
trident-csi-xrhqx              2/2    Running   0          10m
trident-csi-zb7ws              2/2    Running   0          10m
trident-operator-564d7d66f-qrz7v 1/1    Running   0          27m

[user@rhel7 trident-installer]$ ./tridentctl -n trident version
+-----+
```

SERVER VERSION	CLIENT VERSION
20.04.0	20.04.0

11. Create a storage backend that will be used by Trident to provision volumes. The storage backend specifies the Element cluster in NetApp HCI. You also can specify sample bronze, silver, and gold types with corresponding QoS specs.

```
[user@rhel7 trident-installer]$ vi backend.json
{
    "version": 1,
    "storageDriverName": "solidfire-san",
    "Endpoint": "https://admin: admin- password@10.63.172.140/json-rpc/8.0",
    "SVIP": "10.61.185.205:3260",
    "TenantName": "trident",
    "Types": [{"Type": "Bronze", "Qos": {"minIOPS": 1000, "maxIOPS": 2000,
"burstIOPS": 4000}, {"Type": "Silver", "Qos": {"minIOPS": 4000, "maxIOPS": 6000,
"burstIOPS": 8000}}, {"Type": "Gold", "Qos": {"minIOPS": 6000, "maxIOPS": 8000, "burstIOPS": 10000}}]
}
[user@rhel7 trident-installer]$ ./tridentctl -n trident create backend -f backend.json
+-----+
+-----+
|           NAME          | STORAGE DRIVER |          UUID          |
| STATE | VOLUMES | +-----+ +-----+ +-----+
+-----+-----+
| solidfire_10.61.185.205 | solidfire-san | 40f48d99-5d2e-4f6c-89ab-8aee2be71255 |
| online |      0 | +-----+ +-----+ +-----+
+-----+-----+
```

Modify the `backend.json` to accommodate the details or requirements of your environment for the following values:

- Endpoint corresponds to the credentials and the MVIP of the NetApp HCI Element cluster.
- SVIP corresponds to the SVIP configured over the VM network in the section titled [Create Storage Network VLAN](#).
- Types corresponds to different QoS bands. New persistent volumes can be created with specific QoS settings by specifying the exact storage pool.

12. Create a StorageClass that specifies Trident as the provisioner and the storage backend as

`solidfire-san`.

```
[user@rhel7 trident-installer]$ vi storage-class-basic.yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: basic-csi
  annotations:
    storageclass.kubernetes.io/is-default-class: "true"
provisioner: csi.trident.netapp.io
parameters:
  backendType: "solidfire-san"
  provisioningType: "thin"
```

```
[user@rhel7 trident-installer]$ oc create -f storage-class-basic.yaml
storageclass.storage.k8s.io/basic created
```



In this example, the StorageClass created is set as a default, however an OpenShift administrator can define multiple storage classes corresponding to different QoS requirements and other factors based upon their applications. Trident selects a storage backend that can satisfy all the criteria specified in the parameters section in the storage class definition. End users can then provision storage as needed, without administrative intervention.

## Validation Results: NetApp HCI for Red Hat OpenShift on RHV

This section provides the steps to deploy a continuous integration/continuous delivery or deployment (CI/CD) pipeline with Jenkins in order to validate the operation of the solution.

### Create the Resources Required for Jenkins Deployment

To create the resources required for deploying the Jenkins application, complete the following steps:

1. Create a new project named Jenkins.

# Create Project

Name \*

Display Name

Description

Cancel

Create

2. In this example, we deployed Jenkins with persistent storage. To support the Jenkins build, create the PVC. Navigate to Storage > Persistent Volume Claims and click Create Persistent Volume Claim. Select the storage class that was created, make sure that the Persistent Volume Claim Name is jenkins, select the appropriate size and access mode, and then click Create.

## Create Persistent Volume Claim

[Edit YAML](#)**Storage Class** SC basic

Storage class for the new claim.

**Persistent Volume Claim Name \*** jenkins

A unique name for the storage claim within the project.

**Access Mode \*** Single User (RWO)  Shared Access (RWX)  Read Only (ROX)

Permissions to the mounted drive.

**Size \*** 100

GiB



Desired storage capacity.

 Use label selectors to request storage

Use label selectors to define how storage is created.

**Create****Cancel**

### Deploy Jenkins with Persistent Storage

To deploy Jenkins with persistent storage, complete the following steps:

1. In the upper left corner, change the role from Administrator to Developer. Click +Add and select From Catalog. In the Filter by Keyword bar, search jenkins. Select Jenkins Service, with Persistent Storage.

## Developer Catalog

Add shared apps, services, or source-to-image builders to your project from the Developer Catalog. Cluster admins can install additional apps which will show up here automatically.

The screenshot shows the 'Developer Catalog' interface with a search bar containing 'jenkins'. Below the search bar, there are two dropdown menus: 'Group By: None' and 'All Items'. On the left, a sidebar lists categories: Languages, Databases, Middleware, CI/CD, and Other. Under 'Type', several options are listed with checkboxes: Operator Backed (0), Helm Charts (0), Builder Image (0), Template (4), and Service Class (0). The main area displays four template cards for Jenkins:

- Jenkins** (Template): Jenkins provided by Red Hat, Inc. Jenkins service, with persistent storage. NOTE: You must have persistent volumes available in...
- Jenkins** (Template): Jenkins provided by Red Hat, Inc. Jenkins service, with persistent storage. NOTE: You must have persistent volumes available in...
- Jenkins (Ephemeral)** (Template): Jenkins (Ephemeral) provided by Red Hat, Inc. Jenkins service, without persistent storage. WARNING: Any data stored will be lost upon...
- Jenkins (Ephemeral)** (Template): Jenkins (Ephemeral) provided by Red Hat, Inc. Jenkins service, without persistent storage. WARNING: Any data stored will be lost upon...

- Click Instantiate Template.

The screenshot shows the details page for the Jenkins template. At the top, there's a header with the Jenkins logo and the text 'Jenkins' and 'Provided by Red Hat, Inc.' A close button (X) is in the top right corner. Below the header is a large blue button labeled 'Instantiate Template'.

Provider	Description
Red Hat, Inc.	Jenkins service, with persistent storage.
Support	NOTE: You must have persistent volumes available in your cluster to use this template.
Get support ↗	
Created At	Documentation
May 26, 3:58 am	<a href="https://docs.okd.io/latest/using_images/other_images/jenkins.html">https://docs.okd.io/latest/using_images/other_images/jenkins.html</a> ↗

- By default, the details for the Jenkins application are populated. Based on your requirements, modify the parameters, and click Create. This process creates all the required resources for

supporting Jenkins on OpenShift.

## Instantiate Template

Namespace \*

PR Jenkins

Jenkins Service Name

jenkins

The name of the OpenShift Service exposed for the Jenkins container.

Jenkins JNLP Service Name

jenkins-jnlp

The name of the service used for master/slave communication.

Enable OAuth in Jenkins

true

Whether to enable OAuth OpenShift integration. If false, the static account 'admin' will be initialized with the password 'password'.

Memory Limit

1Gi

Maximum amount of memory the container can use.

Volume Capacity \*

50Gi

Volume space available for data, e.g. 512Mi, 2Gi.

Jenkins ImageStream Namespace

openshift

The OpenShift Namespace where the Jenkins ImageStream resides.

Disable memory intensive administrative monitors

false

Whether to perform memory intensive, possibly slow, synchronization with the Jenkins Update Center on start. If true, the Jenkins core update monitor and site warnings monitor are disabled.

Jenkins ImageStreamTag

jenkins:2

Name of the ImageStreamTag to be used for the Jenkins image.

Fatal Error Log File

false

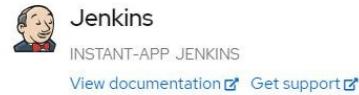
When a fatal error occurs, an error log is created with information and the state obtained at the time of the fatal error.

Allows use of Jenkins Update Center repository with invalid SSL certificate

false

Whether to allow use of a Jenkins Update Center that uses invalid certificate (self-signed, unknown CA). If any value other than 'false', certificate check is bypassed. By default, certificate check is enforced.

**Create** **Cancel**



Jenkins service, with persistent storage.

NOTE: You must have persistent volumes available in your cluster to use this template.

The following resources will be created:

- DeploymentConfig
- PersistentVolumeClaim
- RoleBinding
- Route
- Service
- ServiceAccount

4. The Jenkins pods take approximately 10–12 minutes to enter the Ready state.

Project: jenkins ▾

## Pods

Create Pod

Filter by name...

1	Running	0	Pending	0	Terminating	0	CrashLoopBackOff	1	Completed	0	Failed	0	Unknown
Select all filters													

1 of 2 Items

Name	Namespace	Status	Ready	Owner	Memory	CPU	⋮
jenkins-1-c77n9	jenkins	Running	1/1	jenkins-1	-	0.004 cores	⋮

5. After the pods are instantiated, navigate to Networking > Routes. To open the Jenkins webpage, click the URL provided for the jenkins route.

Project: jenkins ▾

## Routes

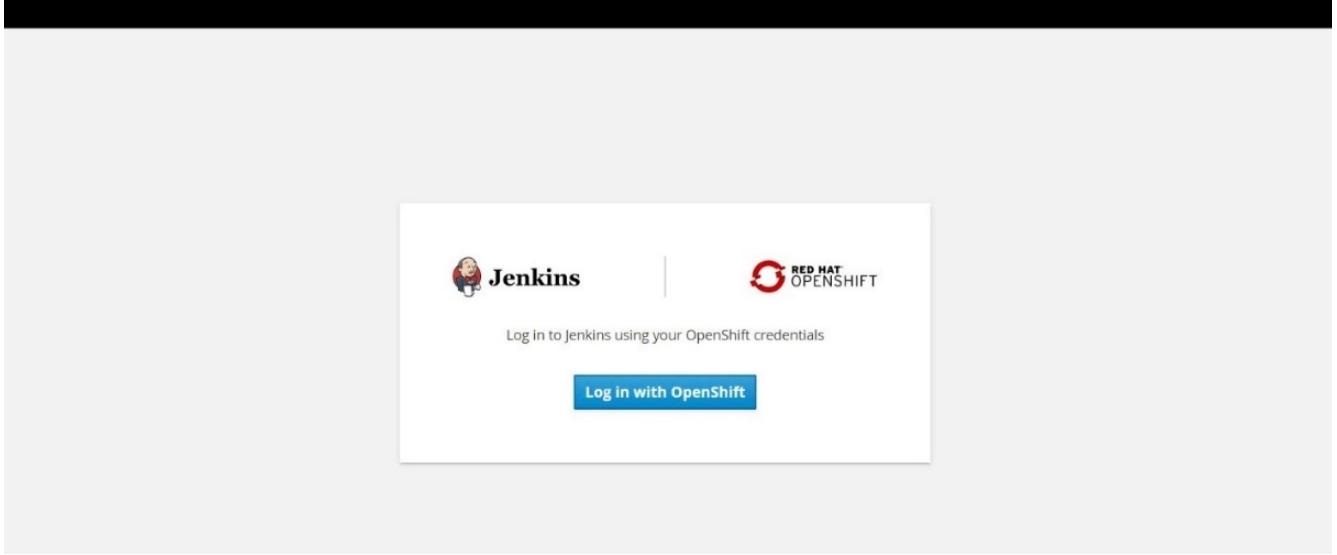
Create Route

Filter by name...

1	Accepted	0	Rejected	0	Pending	Select all filters	1 Item
---	----------	---	----------	---	---------	--------------------	--------

Name	Namespace	Status	Location	Service	⋮
jenkins	jenkins	Accepted	<a href="https://jenkins-jenkins.apps.rhv-ocp-cluster.cie.netapp.com">https://jenkins-jenkins.apps.rhv-ocp-cluster.cie.netapp.com</a>	jenkins	⋮

6. Because the OpenShift OAuth was used while creating the Jenkins app, click Log in with OpenShift.



7. Authorize jenkins service-account to access the OpenShift users.

## Authorize Access

Service account jenkins in project jenkins is requesting permission to access your account (kube:admin)

Requested permissions

**user:info**

Read-only access to your user information (including username, identities, and group membership)

**user:check-access**

Read-only access to view your privileges (for example, "can I create builds?")

You will be redirected to <https://jenkins-jenkins.apps.rhv-ocp-cluster.cie.netapp.com/securityRealm/finishLogin>

[Allow selected permissions](#)

[Deny](#)

8. The Jenkins welcome page is displayed. Because we are using a Maven build, complete the Maven installation first. Navigate to Manage Jenkins > Global Tool Configuration, then in the Maven subhead, click Add Maven. Enter the name of your choice and make sure that the Install Automatically option is selected. Click Save.

Maven

Maven installations

Add Maven

Maven

Name M3

Install automatically

Install from Apache

Version 3.6.3

Add Installer

Delete installer

Add Maven

List of Maven installations on this system

9. You can now create a pipeline to demonstrate the CI/CD workflow. On the home page, click Create New Jobs or New Item from the left-hand menu.

Jenkins

3

search

kube:admin | log out

ENABLE AUTO REFRESH

New Item

People

Build History

Manage Jenkins

My Views

Open Blue Ocean

Lockable Resources

Credentials

New View

Welcome to Jenkins!

Please [create new jobs](#) to get started.

Build Queue

No builds in the queue.

Build Executor Status

1 Idle

2 Idle

10. On the Create Item page, enter the name of your choice, select Pipeline, and click Ok.

## Enter an item name

sample-demo

» Required field



### Freestyle project

This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.



### Pipeline

Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.



### Multi-configuration project

Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.



### Bitbucket Team/Project

Scans a Bitbucket Cloud Team (or Bitbucket Server Project) for all repositories matching some defined markers.



### Folder

Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.



### GitHub Organization

Scans a GitHub organization (or user account) for all repositories matching some defined markers.

OK

### Multibranch Pipeline

Creates a set of Pipeline projects according to detected branches in one SCM repository

11. Select the Pipeline tab. From the Try Sample Pipeline drop- down menu, select Github + Maven. The code is automatically populated. Click Save.

General Build Triggers Advanced Project Options Pipeline

Advanced...

## Pipeline

Definition Pipeline script

Script

```
1 node {
2     def mvnHome
3     stage('Preparation') { // for display purposes
4         // Get some code from a GitHub repository
5         git 'https://github.com/jglick/simple-maven-project-with-tests.git'
6         // Get the Maven tool.
7         // ** NOTE: This 'M3' Maven tool must be configured
8         // ** in the global configuration.
9         mvnHome = tool 'M3'
10    }
11    stage('Build') {
12        // Run the maven build
13        withEnv(["MVN_HOME=$mvnHome"]) {
14            if (isUnix()) {
15                sh '$MVN_HOME/bin/mvn' -Dmaven.test.failure.ignore clean package'
16            } else {
17                bat("%MVN_HOME%\bin\mvn" -Dmaven.test.failure.ignore clean package)
18            }
19        }
20    }
21}
```

GitHub + Maven

Use Groovy Sandbox

[Pipeline Syntax](#)

[Save](#) [Apply](#)

12. Click Build Now to trigger the development through the preparation, build, and testing phase. It can take several minutes to complete the whole build process and display the results of the build.

 Jenkins

Jenkins > sample-demo >

[Back to Dashboard](#)

[Status](#)

[Changes](#)

[Build Now](#)

[Delete Pipeline](#)

[Configure](#)

[Full Stage View](#)

[Open Blue Ocean](#)

[Rename](#)

[Pipeline Syntax](#)

**Build History** [trend](#)

find X

#1 May 27, 2020 3:53 PM

[Atom feed for all](#) [Atom feed for failures](#)

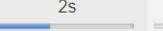
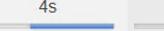
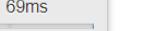
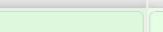
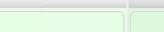
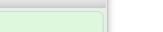
**Pipeline sample-demo**

Last Successful Artifacts  
 [simple-maven-project-with-tests-1.0-SNAPSHOT.jar](#) 1.71 KB [view](#)

Recent Changes  


**Stage View**

Average stage times:  
(Average full run time: ~7s)

Preparation	Build	Results
2s	4s	69ms
 2s	 4s	 69ms
 2s	 4s	 69ms

 [Latest Test Result \(no failures\)](#)

#### Permalinks

- [Last build \(#1\), 1 min 23 sec ago](#)
- [Last stable build \(#1\), 1 min 23 sec ago](#)
- [Last successful build \(#1\), 1 min 23 sec ago](#)
- [Last completed build \(#1\), 1 min 23 sec ago](#)

13. Whenever there are any code changes, the pipeline can be rebuilt to patch the new version of software enabling continuous integration and continuous delivery. Click Recent Changes to track the changes from the previous version.

## Best Practices for Production Deployments: NetApp HCI for Red Hat OpenShift on RHV

This section lists several best practices that an organization should take into consideration before deploying this solution into production.

### Deploy OpenShift to an RHV Cluster of at Least Three Nodes

The verified architecture described in this document presents the minimum hardware deployment suitable for HA operations by deploying two RHV-H hypervisor nodes and ensuring a fault tolerant configuration where both hosts can manage the hosted-engine and deployed VMs can migrate between the two hypervisors. Because Red Hat OpenShift initially deploys with three master nodes, it is ensured in a two-node configuration that at least two masters will occupy the same node, which can lead to a possible outage for OpenShift if that specific node becomes unavailable. Therefore, it is a Red Hat best practice that at least three RHV-H hypervisor nodes be deployed as part of the solution so that the OpenShift masters can be distributed evenly, and the solution receives an added degree of fault tolerance.

## Configure Virtual Machine/Host Affinity

Ensuring the distribution of the OpenShift masters across multiple hypervisor nodes can be achieved by enabling VM/host affinity. Affinity is a way to define rules for a set of VMs and/or hosts that determine whether the VMs run together on the same host or hosts in the group or on different hosts. It is applied to VMs by creating affinity groups that consist of VMs and/or hosts with a set of identical parameters and conditions. Depending on whether the VMs in an affinity group run on the same host or hosts in the group or separately on different hosts, the parameters of the affinity group can define either positive affinity or negative affinity. The conditions defined for the parameters can be either hard enforcement or soft enforcement. Hard enforcement ensures that the VMs in an affinity group always follows the positive/negative affinity strictly without any regards to external conditions. Soft enforcement, on the other hand, ensures that a higher preference is set out for the VMs in an affinity group to follow the positive/negative affinity whenever feasible. In a two or three hypervisor configuration as described in this document soft affinity is the recommended setting, in larger clusters hard affinity can be relied on to ensure OpenShift nodes are distributed. To configure affinity groups, see the [Red Hat 6.11. Affinity Groups documentation](#).

## Use a Custom Install File for OpenShift Deployment

IPI makes the deployment of OpenShift clusters extremely easy through the interactive wizard discussed earlier in this document. However, it is possible that there are some default values that might need to be changed as a part of a cluster deployment. In these instances, the wizard can be run and tasked without immediately deploying a cluster, but instead outputting a configuration file from which the cluster can be deployed later. This is very useful if any IPI defaults need to be changed, or if a user wants to deploy multiple identical clusters in their environment for other uses such as multitenancy. For more information about creating a customized install configuration for OpenShift, see [Red Hat OpenShift Installing a Cluster on RHV with Customizations](#).

## Videos and Demos: NetApp HCI for Red Hat OpenShift on RHV

The following video demonstrates some of the capabilities documented in this document:

 | *NetApp HCI for Red Hat OpenShift on Red Hat Virtualization*

## Additional Information: NetApp HCI for Red Hat OpenShift on RHV

To learn more about the information described in this document, review the following websites:

- NetApp HCI Documentation <https://www.netapp.com/us/documentation/hci.aspx>
- NetApp Trident Documentation <https://netapp-trident.readthedocs.io/en/stable-v20.04/>
- Red Hat Virtualization Documentation [https://access.redhat.com/documentation/en-us/red\\_hat\\_virtualization/4.3/](https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/)

- Red Hat OpenShift Documentation [https://access.redhat.com/documentation/en-us/openshift\\_container\\_platform/4.4/](https://access.redhat.com/documentation/en-us/openshift_container_platform/4.4/)

# Database

# Data Fabric

# Data Protection

## TR-4830: NetApp HCI Disaster Recovery with Cleondris

Michael White, NetApp

### Overview of Business Continuity and Disaster Recovery

The business continuity and disaster recovery (BCDR) model is about getting people back to work. Disaster recovery focuses on bringing technology, such as an email server, back to life. Business continuity makes it possible for people to access that email server. Disaster recovery alone would mean that the technology is working, but nobody might be using it; BCDR means that people have started using the recovered technology.

### Business Impact Assessment

It is hard to know what is required to make a tier 1 application work. It is usually obvious that authentication servers and DNS are important. But is there a database server somewhere too?

This information is critical because you need to package tier 1 applications so that they work in both a test failover and a real failover. An accounting firm can perform a business impact assessment (BIA) to provide you with all the necessary information to successfully protect your applications: for example, determining the required components, the application owner, and the best support person for the application.

### Application Catalog

If you do not have a BIA, you can do a version of it yourself: an application catalog. It is often done in a spreadsheet with the following fields: application name, components, requirements, owner, support, support phone number, and sponsor or business application owner. Such a catalog is important and useful in protecting your applications. The help desk can sometimes help with an application catalog; they often have already started one.

### What Not to Protect

There are applications that should not be protected. For example, you can easily and cheaply have a domain controller running as a virtual machine (VM) at your disaster recovery site, so there is no need to protect one. In fact, recovering a domain controller can cause issues during recovery. Monitoring software that is used in the production site does not necessarily work in the disaster recovery site if it is recovered there.

It is usually unnecessary to protect applications that can be protected with high availability. High availability is the best possible protection; its failover times are often less than a second. Therefore, disaster recovery orchestration tools should not protect these applications, but high availability can.

An example is the software in banks that support ATMs.

You can tell that you need to look at high-availability solutions for an application when an application owner has a 20-second recovery time objective (RTO). That RTO is beyond replication solutions.

## Product Overview

The Cleondris HCI Control Center (HCC) adds disaster recovery capabilities to new and existing NetApp HCI deployments. It is fully integrated with the NetApp SolidFire storage engine and can protect any kind of data and applications. When a customer site fails, HCC can be used to recover all data at a secondary NetApp HCI site, including policy-based VM startup orchestration.

Setting up replication for multiple volumes can be time consuming and error prone when performed manually. HCC can help with its Replication Wizard. The wizard helps set up the replication correctly so that the servers can access the volumes if a disaster occurs. With HCC, the VMware environment can be started on the secondary system in a sandbox without affecting production. The VMs are started in an isolated network and a functional test is possible.

## Installing Cleondris: NetApp HCI DR with Cleondris

### Prerequisites

There are several things to have ready before you start with the installation.

This technical report assumes that you have your NetApp HCI infrastructure working at both your production site and your disaster recovery site.

- **DNS.** You should have DNS prepared for your HCC disaster recovery tool when you install it.
- **FQDN.** A fully qualified domain name for the disaster recovery tool should be prepared before installation.
- **IP address.** The IP will be part of the FQDN before it is put into DNS.
- **NTP.** You need a Network Time Protocol (NTP) server address. It can be either your own internal or external address, but it needs to be accessible.
- **Storage location.** When you install HCC, you must know which datastore it should be installed to.
- **vCenter Server service account.** You will need to have a service account created in vCenter Server on both the disaster recovery and production side for HCC to use. It does not require administrator-level permissions at the root level. If you like, you can find exactly what is required in the HCC user guide.
- **NetApp HCI service account.** You need a service account in your NetApp HCI storage for both the disaster recovery and production side for HCC to use. Full access is required.
- **Test network.** This network should be connected to all your hosts in the disaster recovery site, and it should be isolated and nonrouting. This network is used to make sure applications work during a test failover. The built-in test network that is temporary only is a one-host network. Therefore, if

your test failover has VMs scattered on multiple hosts, they will not be able to communicate. I recommend that you create a distributed port group in the disaster recovery site that spans all hosts but is isolated and nonrouting. Testing is important to success.

- **RTOs.** You should have RTOs approved by management for your application groups. Often it is 1 or 2 hours for tier 1 applications; for tier 4 applications, it can be as long as 12 hours. These decisions must be approved by management because they will determine how quickly things work after a critical outage. These times will determine replication schedules.
- **Application information.** You should know which application you need to protect first, and what it needs to work. For example, Microsoft Exchange needs a domain controller that has a role of Global Catalog to start. In my own experience, a customer said that they had one email server to protect. It did not test well, and when I investigated, I discovered the customer had 24 VMs that were part of the email application.

## Download Information

You can download HCC from the [Cleondris site](#). When you buy it, you receive an email with a download link as well.

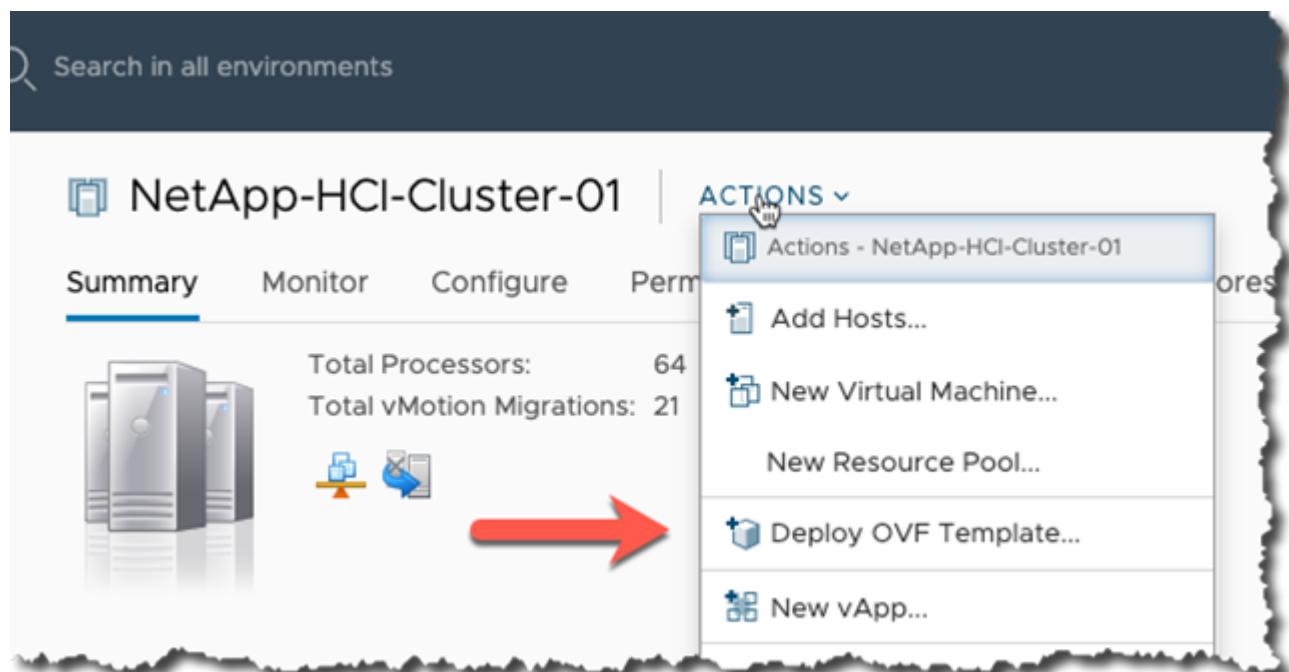
## License

Your license will arrive in an email when you purchase or if you get a not-for-resale (NFR) version. You can get a trial license through the [Cleondris Support Portal](#).

## Deployment

You download an OVF file, so it is deployed like many other things.

1. Start by using the Actions menu available at the cluster level.



2. Select the file.

## Deploy OVF Template

### 1 Select an OVF template

2 Select a name and folder

3 Select a compute resource

4 Review details

5 Select storage

6 Ready to complete

### Select an OVF template

Select an OVF template from remote URL or local file system

Enter a URL to download and install the OVF package from the Internet, or browse to a location accessible from your computer, such as a local hard drive, a network share, or a CD/DVD drive.

URL

http | https://remoteserver-address/filetodeploy.ovf | .ova

Local file

Choose Files cleondris-appliance-1705.ova

3. Name the appliance and select the location for it in the vCenter infrastructure.

## Deploy OVF Template

### ✓ 1 Select an OVF template

### 2 Select a name and folder

3 Select a compute resource

4 Review details

5 Select storage

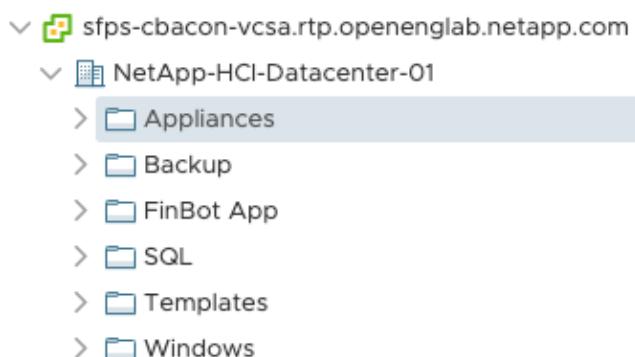
6 Ready to complete

### Select a name and folder

Specify a unique name and target location

Virtual machine name: dr-hcc

Select a location for the virtual machine.

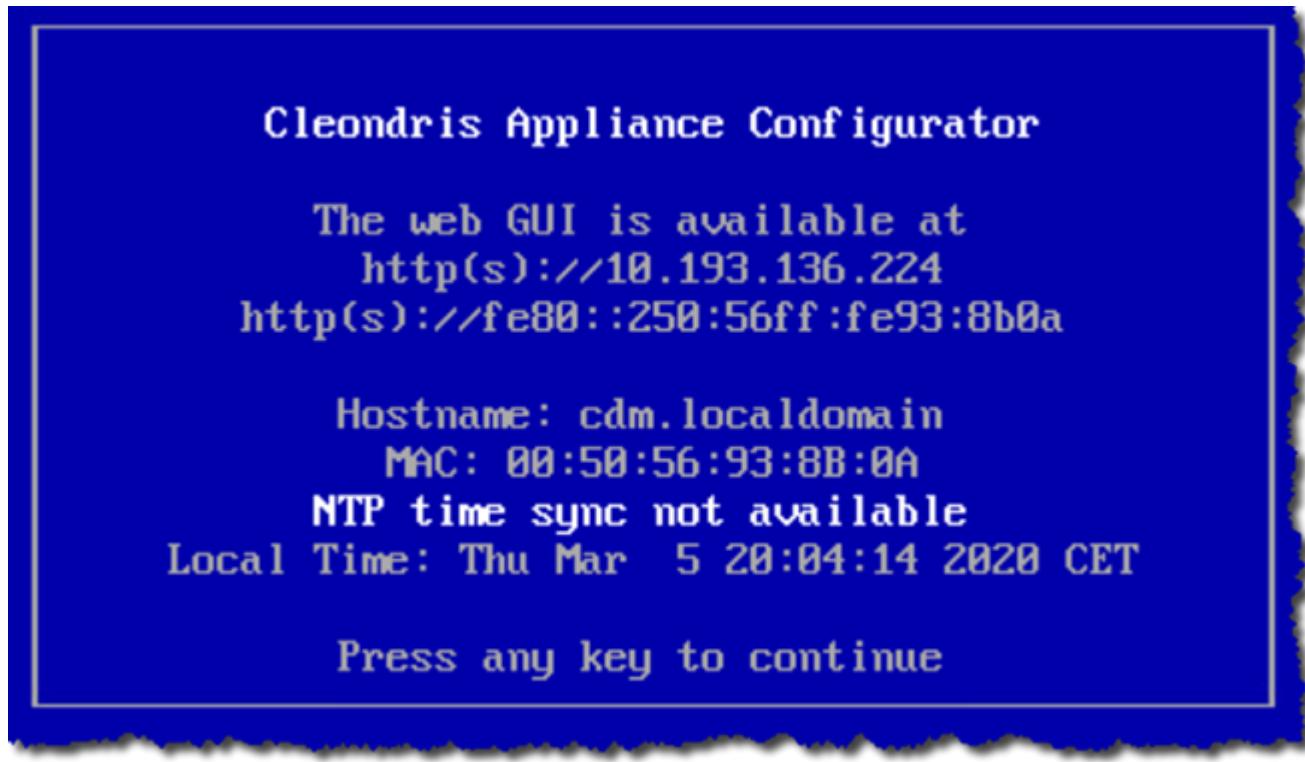


4. Select the Compute location.
5. Confirm the details.
6. Accept the license details.
7. Select the appropriate storage location.
8. Select the network that our appliance will work on.
9. Review the details again and click Finish.
10. Now wait for the appliance to be deployed, and then power it up. As it powers up, you might see a message saying that VMware tools are not installed. You can ignore this message; it will go away automatically.

## Initial Configuration

To start the initial configuration, complete the following steps:

1. This phase involves doing the configuration in the Appliance Configurator, which is the VM console. So, after the appliance powers up, change to work in the console by using the VMware Remote Console (VMRC) or the HTML5 VMRC version. Look for a blue Cleondris screen.



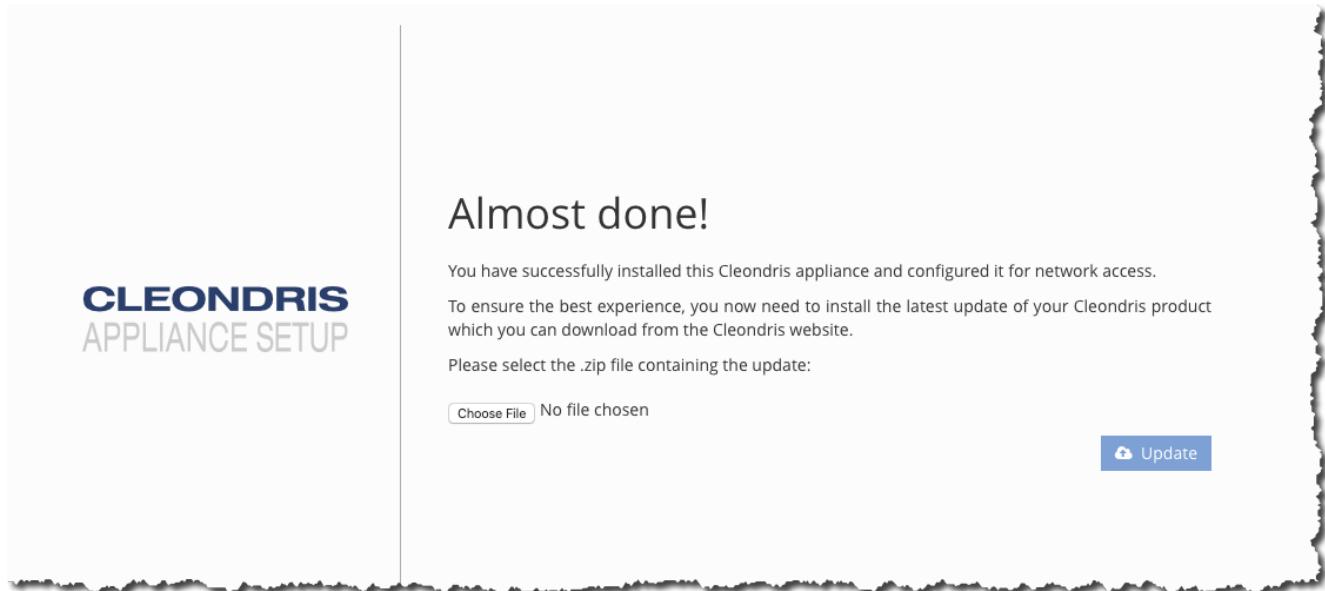
2. Press any key to proceed, and configure the following:
  - The web administrator password
  - The network configuration: IP, DNS, and so on
  - The time zone

- NTP
3. Select the Reboot and Activate Network/NTP Settings. You will see the appliance reboot. Afterward, do a ping test to confirm the FQDN and IP.

## Patching Cleondris

To update your Cleondris product, complete the following steps:

1. When you first log in to the appliance, you see a screen like the following:



2. Click Choose File to select the update you downloaded from the Cleondris website.

## Almost done!

You have successfully installed this Cleondris appliance and configured it for network access.

To ensure the best experience, you now need to install the latest update of your Cleondris product which you can download from the Cleondris website.

Please select the .zip file containing the update:



3. Upload the patch. After the appliance reboots, the following login screen is displayed:



4. You can now see the new version and build information; confirming that the update was successful. Now you can continue with the configuration.

## Software Used

This technical report uses the following software versions:

- vSphere 6.5 on production
- vSphere 6.7 U3 on DR
- NetApp Element 11.5 on production
- NetApp Element 12.0 on DR
- Cleondris HCC 8.0.2007 Build 20200707-1555 and 8.0.2007X2 build 20200709-1936.

## Configuring Cleondris: NetApp HCI DR with Cleondris

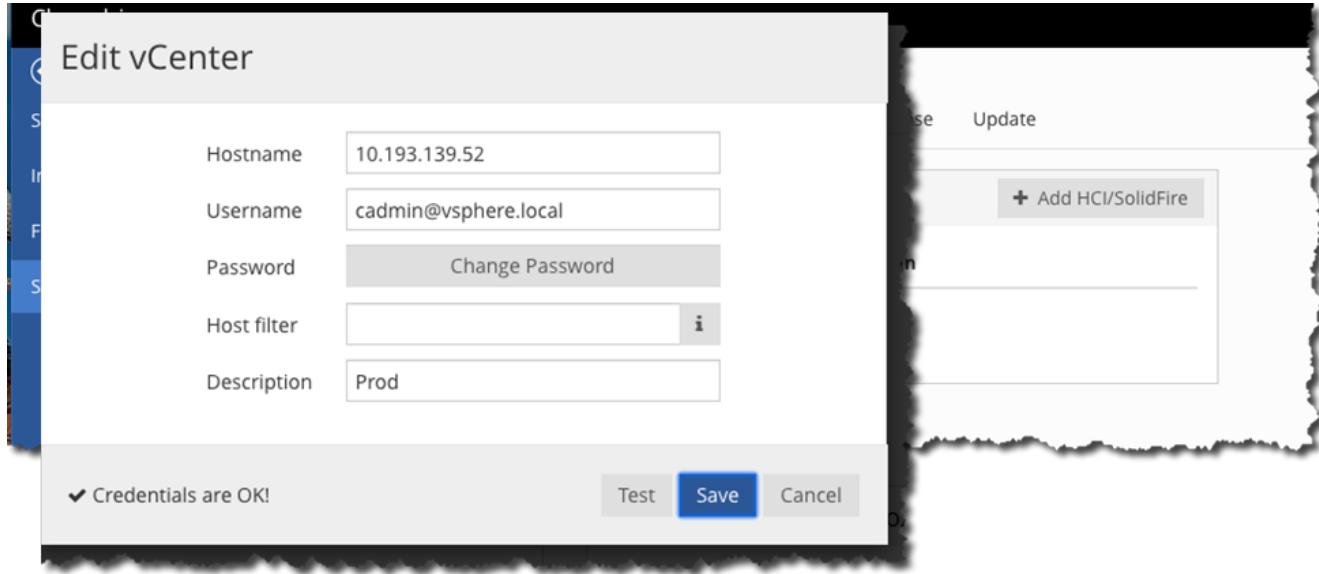
You now configure Cleondris to communicate with your vCenter Servers and storage. If you have logged out, returned, and log in again to start here, you are prompted for the following information:

1. Accept the EULA.
2. Copy and paste the license.
3. You are prompted to perform configuration, but skip this step for now. It is better to perform this configuration as detailed later in this paper.
4. When you log back in and see the green boxes, you must change to the Setup area.

## Add vCenter Servers

To add the vCenter Servers, complete the following steps:

1. Change to the VMware tab and add your two vCenter Servers. When you are defining them, add a good description and use the Test button.



This example uses an IP address instead of an FQDN. (This FQDN didn't work at first; I later found out that I had not entered the proper DNS information. After correcting the DNS information, the FQDN worked fine.) Also notice the description, which is useful.

2. After both vCenter Servers are done, the screen displays them.

The screenshot shows the VMware tab of the interface. It lists two vCenter servers: 'sfps-megatron-vcsa.rtp.openenglab.netapp.com' and 'sfps-cbacon-vcsa.rtp.openenglab.netapp.com'. Each entry includes columns for Hostname, Username, and Description, along with edit and delete icons.

Hostname	Username	Description
sfps-megatron-vcsa.rtp.openenglab.netapp.com	cadmin@vsphere.local	Prod
sfps-cbacon-vcsa.rtp.openenglab.netapp.com	administrator@vsphere.local	DR

## Add NetApp HCI Clusters

To add the NetApp HCI clusters, complete the following steps:

1. Change to the NetApp tab and add your production and disaster recovery storage. Again, add a good description and use the Test button.

## Register HCI/SolidFire

Hostname	10.193.139.9
Username	admin
Password	.....
Description	DR

✓ Credentials are OK!

**Test** **Save** **Cancel**

- When you have added your storage and vCenter Servers, change to the Inventory view so that you can see the results of your configuration.

HCI/SolidFire (2)			
Hostname	Name	Vol	VM
10.193.139.9	sfps-cbacon-cluster	12	12
10.193.139.58	sfps-megatron-cluster	26	134

vCenter (2)		
vCenter	Hosts	VMs
sfps-cbacon-vcsa.rtp.openenglab.netapp.com	2	12
sfps-megatron-vcsa.rtp.openenglab.netapp.com	5	130

Here you can see the number of objects, which is a good way to confirm that things are working.

## Replication

You can use HCC to enable replication between your two sites. This allows us to stay in the HCC UI and decide what volumes to replicate.

**Important:** If a replicated volume contains VMs that are in two plans, only the first plan that fails over works because it will disable replication on that volume.

I recommend that each tier 1 application have its own volume. Tier 4 applications can all be on one volume, but there should be only one failover plan.

## Disaster Recovery Pairing: NetApp HCI DR with Cleondris

- Display the Failover page.

2. On the diagram of your vCenter Servers and storage, select the Protection tab.

The screenshot shows the Cleondris software interface. At the top, there is a header bar with the Cleondris logo. Below the header is a navigation menu on the left side with the following items: Status, Inventory, Backup, Restore, Failover (which is highlighted in blue), and Setup. To the right of the navigation menu is a main content area. The top part of the content area has tabs: Overview, Protection (which is underlined in blue), Plans, and Activity. Below these tabs is a section titled "VMware Virtual Machine Protection" which contains a "vCenter" section listing two hosts: sfps-megatron-vcsa.rtp.openenglab.netapp.com and sfps-primus-vcsa.rtp.openenglab.netapp.com. The bottom part of the content area is titled "HCI Storage Protection" and contains a "Cluster" section listing two clusters: sfps-megatron-cluster and sfps-primus-cluster.

The far side of the screen displays some useful information, such as how many protected VMs you have. (In this example, none right now.) You can also access the Replication Wizard here.

**Protected Datastores    Protected VMs**

0/24                      0/133

0/5                      0/6

 Replication Wizard

**Protected Datastores    Protected VMs**

0/17                      0/137

0/3                      0/6

2004P6 - API-20200410-2157 - Copyright © Cleondris GmbH 2010-2020

This wizard makes the replication setup easy.

## HCI Replication Wizard

Source Volumes      Destination      vCenter      Preview

Select the cluster you want to protect:

Cluster: sfps-megatron-cluster

	ID	Type	Name
<input type="checkbox"/>	1	Primary	<a href="#">NetApp-HCI-Datastore-01</a>
<input type="checkbox"/>	2	Primary	<a href="#">NetApp-HCI-Datastore-02</a>
<input type="checkbox"/>	3	Primary	<a href="#">NetApp-HCI-Select-Install</a>
<input type="checkbox"/>	4	Primary	<a href="#">NetApp-HCI-Select-Data-01</a>
<input type="checkbox"/>	5	Primary	<a href="#">NetApp-HCI-Select-Data-02</a>
<input type="checkbox"/>	6	Primary	<a href="#">NetApp-HCI-Select-Data-03</a>
<input type="checkbox"/>	7	Primary	<a href="#">NetApp-HCI-Select-Data-04</a>
<input type="checkbox"/>	8	Primary	<a href="#">INFRASTRUCTURE</a>
<input type="checkbox"/>	12	Primary	<a href="#">DESKTOP02</a>
<input checked="" type="checkbox"/>	15	Primary	<a href="#">DESKTOP03</a>
<input type="checkbox"/>	16	Primary	<a href="#">DESKTOP04</a>
<input type="checkbox"/>	569	Primary	<a href="#">workload-db-mongo-1</a>

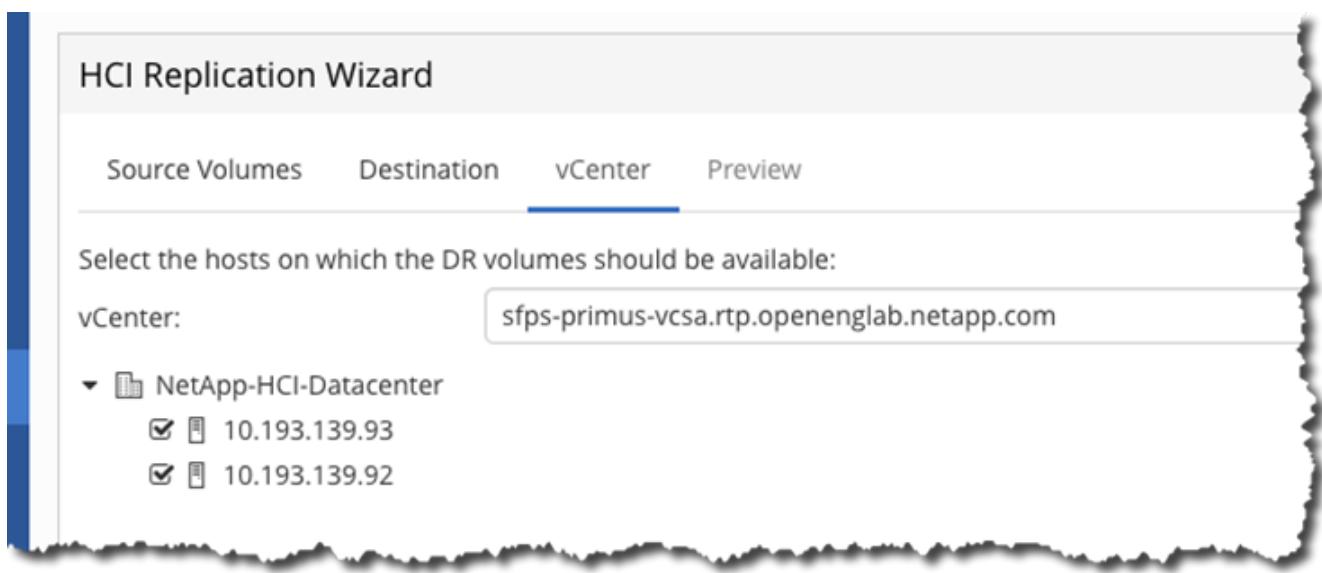
3. You can select the volumes that are important to you, but also make sure that you have the proper vCenter Server selected at the top in the cluster field.

At the far right, you see the pairing type, and only Sync is allowed or supported.

After you click Next, the destination area is displayed.



4. The default information is normally right, but it's still worth checking. Then click Next.



It is important to make sure that the disaster recovery site vCenter Server is displayed and that all hosts are selected. After that is complete, use the Preview button.

5. Next you see a summary. You can click Create DR to set the volume pairing and start replication.

Depending on your settings, replication might take a while. I suggest that you wait overnight.

## Failover: NetApp HCI DR with Cleondris

### Test Failover

A test failover is important, because it proves to you, your application owner, your manager, and the BCDR people that your disaster recovery plan works.

To test failover, complete the following steps:

1. From the Failover page, click Start Failover.

2. On the Failover page, you have some choices to make.

Carefully specify the plan, where the VMs came from, and where they are going to be recovered.

From: sfps-megatron-cluster To: sfps-primus-cluster							<span style="color: orange;">⚠</span> 3 VMs not included in this plan will lose protection
Plan	Priority	Name	Datastore	Source Volume	Destination Volume	Current vCenter	Destination vCenter
MultiTier	1	taxdb	DESKTOP03	DESKTOP03 ID: 15	DESKTOP03 ID: 138	sfps-megatron-vcsa.rtp.openenglab.netapp.com	sfps-megatron-vcsa.rtp.openeng
MultiTier	1	crmdb	DESKTOP03	DESKTOP03 ID: 15	DESKTOP03 ID: 138	sfps-megatron-vcsa.rtp.openenglab.netapp.com	sfps-megatron-vcsa.rtp.openeng
MultiTier	1	FinRptdb	DESKTOP03	DESKTOP03 ID: 15	DESKTOP03 ID: 138	sfps-megatron-vcsa.rtp.openenglab.netapp.com	sfps-megatron-vcsa.rtp.openeng
MultiTier	2	crmA	DESKTOP03	DESKTOP03 ID: 15	DESKTOP03 ID: 138	sfps-megatron-vcsa.rtp.openenglab.netapp.com	sfps-megatron-vcsa.rtp.openeng
MultiTier	2	FinRptA	DESKTOP03	DESKTOP03 ID: 15	DESKTOP03 ID: 138	sfps-megatron-vcsa.rtp.openenglab.netapp.com	sfps-megatron-vcsa.rtp.openeng
MultiTier	2	taxA	DESKTOP03	DESKTOP03 ID: 15	DESKTOP03 ID: 138	sfps-megatron-vcsa.rtp.openenglab.netapp.com	sfps-megatron-vcsa.rtp.openeng
MultiTier	3	taxW	DESKTOP03	DESKTOP03 ID: 15	DESKTOP03 ID: 138	sfps-megatron-vcsa.rtp.openenglab.netapp.com	sfps-megatron-vcsa.rtp.openeng
MultiTier	3	crmW	DESKTOP03	DESKTOP03 ID: 15	DESKTOP03 ID: 138	sfps-megatron-vcsa.rtp.openenglab.netapp.com	sfps-megatron-vcsa.rtp.openeng
MultiTier	3	FinRptW	DESKTOP03	DESKTOP03 ID: 15	DESKTOP03 ID: 138	sfps-megatron-vcsa.rtp.openenglab.netapp.com	sfps-megatron-vcsa.rtp.openeng

Failover to Sandbox		Start	Cancel
---------------------	--	-------	--------

The screen displays a list of the VMs that are in the plan. In this example, a warning at the top right says that three VMs are not included. That means there are three VMs we did not make part of the plan in the replicated volume.

If you see a red X in the first column on the left, you can click it and learn what the problem is.

- At the bottom right of the screen, you must choose whether to test the failover (Failover to Sandbox) or start a real failover. In this example, we select Failover to Sandbox.

Cleondris HCC

Inventory search

Settings Logout

Status

Inventory

Backup

Restore

**Failover**

Setup

Overview Protection Plans Activity

Failover Plan Execution

ID	Description	User	Plan	Date	Status
2	Sandbox failover using plan Mass	admin	Mass	2020-04-14 13:21	Running

8.0.2004P6 - API-20200410-2157 - Copyright © Cleondris GmbH 2010-2020

- A summary now lists plans in action. For more information, use the magnifying glass in the far left (described in “Monitoring,” later in this document).

## Running Failover

At first, the failover is the same as the test failover. But the procedure changes when you arrive at the point shown here:

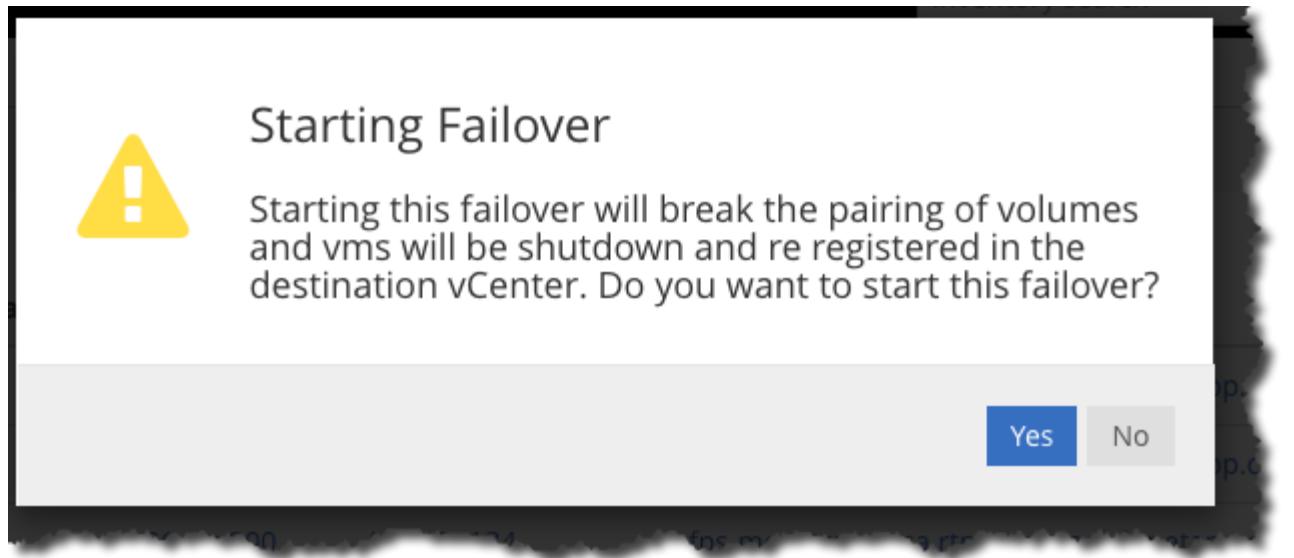
- Instead of selecting the Failover to Sandbox option, select Start.

Failover Preview

From:	sfps-megatron-cluster		To:	sfps-primus-cluster			
Plan	Priority	Name	Datastore	Source Volume	Destination Volume	Current vCenter	Destination vCenter
ABC	1	ABC01	ABC	ABC ID: 800	ABC ID: 134	sfps-megatron-vcsa.rtp.openenglab.netapp.com	sfps-primus-vcsa.rtp.openenglab.netapp.com
ABC	1	ABC03	ABC	ABC ID: 800	ABC ID: 134	sfps-megatron-vcsa.rtp.openenglab.netapp.com	sfps-primus-vcsa.rtp.openenglab.netapp.com
ABC	1	ABC02	ABC	ABC ID: 800	ABC ID: 134	sfps-megatron-vcsa.rtp.openenglab.netapp.com	sfps-primus-vcsa.rtp.openenglab.netapp.com

Failover to Sandbox Start Cancel

- Select Yes.



3. The screen shows that this is a failover, and it is running. For more information, use the magnifying glass (discussed in the “Monitoring” section).

A screenshot of a software interface showing the "Activity" tab. On the left, there is a sidebar with navigation links: Status, Inventory, Backup, Restore, **Failover**, and Setup. The "Failover" link is highlighted. The main area has tabs: Overview, Protection, Plans, and **Activity**. The "Activity" tab is selected. It displays a table titled "Failover Plan Execution".  
Table Data:  
| Id | Description | User | Plan | Date | Status |  
| 4 | Failover using plan ABC | admin | ABC | 2020-04-15 08:25 | Running |

## Monitoring During a Failover

1. When a failover or a test failover is running, you can monitor it by using the magnifying glass at the far right.

A screenshot of a software interface showing the "Activity" tab. On the left, there is a sidebar with navigation links: Status, Inventory, Backup, Restore, **Failover**, and Setup. The "Failover" link is highlighted. The main area has tabs: Overview, Protection, Plans, and **Activity**. The "Activity" tab is selected. It displays a table titled "Failover Plan Execution".  
Table Data:  
| Id | Description | User | Plan | Date | Status |  
| 2 | Sandbox failover using plan Mass | admin | Mass | 2020-04-14 13:21 | Running |  
A magnifying glass icon is positioned to the right of the status column header in the table.

2. Click the magnifying glass to see much more detail.

The screenshot shows the Cleondris HCC web interface. On the left is a vertical navigation bar with options: Status, Inventory, Backup, Restore, Failover (which is selected), and Setup. The main content area displays a failover log for a ticket. At the top, it shows the ticket details: Ticket: Sandbox failover using plan Mass, Started: 2020-04-14 12:30, and Status: Finished. Below this are two tabs: Log (selected) and Details. A table follows, showing log entries:

Date	Type	Message
2020-04-14 12:30	Log	Starting failover activity using plan 'Mass'
2020-04-14 12:30	Log	Sandbox mode = on
2020-04-14 12:30	Log	Creating clone of volume 137

A blue "Download Report" button is located in the top right corner of the content area.

3. As the failover or test failover progresses, a VM Screenshots option appears.

The screenshot shows the Cleondris HCC web interface. The navigation bar on the left is identical to the previous one. The main content area displays a failover log for a ticket. The ticket details are the same: Ticket: Sandbox failover using plan Mass, Started: 2020-04-14 13:29, and Status: WaitRelease. Below the ticket details are three tabs: Log (selected), Details, and VM Screenshots. A table follows, showing log entries:

Date	Type	Message
2020-04-14 13:29	Log	Starting failover activity using plan 'Mass'
2020-04-14 13:29	Log	Sandbox mode = on

Sometimes it is useful to see the screenshots to confirm that the VM is running. It is not logged in, so you cannot tell if the applications are running, but at least you know that the VM is.

### Looking at History When No Failover Is Running

To view past tests or failovers, click the Show Historical button on the Activity tab. Use the magnifying glass for more detail.

Cleondris HCC

Inventory search  Settings Logout

Status Overview Protection Plans **Activity**

**Failover Plan Execution** Show Historical

ID	Description	User	Plan	Date	Status	Actions
2	Sandbox failover using plan Mass	admin	Mass	2020-04-14 13:21	Running	

8.0.2004P6 - API-20200410-2157 - Copyright © Cleondris GmbH 2010-2020

Cleondris HCC

Inventory search  Settings Logout

Status Overview Protection Plans **Activity**

**Failover Plan Execution** Hide Historical

ID	Description	User	Plan	Date	Status	Actions
2	Sandbox failover using plan Mass	admin	Mass	2020-04-14 13:21	Running	
1	Sandbox failover using plan Mass	admin	Mass	2020-04-14 12:30	Finished	

8.0.2004P6 - API-20200410-2157 - Copyright © Cleondris GmbH 2010-2020

You can also download a report with the details.

Cleondris HCC

Inventory search  Settings Logout

Status Download Report

**Ticket:** Sandbox failover using plan Mass  
**Started:** 2020-04-14 12:30  
**Status:** Finished

Log Details

Date	Type	Message
2020-04-14 12:30	Log	Starting failover activity using plan 'Mass'
2020-04-14 12:30	Log	Sandbox mode = on
2020-04-14 12:30	Log	Creating clone of volume 137

These reports have various uses: for example, to prove to an application owner that you tested the failover of that application. Also, the report can provide details that might help you troubleshoot a failed failover.

You can add text to a report by adding the text to the plan in the comment field.

Failover Plan Editor

Plan Name:	ABC
Comment (Added to the report)	App expert is Joe Smith.
Create temporary network when running in sandbox mode:	<input type="checkbox"/>
Network to use for sandbox mode	TestNetwork

## Where to Find Additional Information: NetApp HCI DR with Cleondris

To learn more about the information that is described in this document, review the following websites:

- NetApp HCI Documentation Center  
<https://docs.netapp.com/hci/index.jsp>
- NetApp HCI Documentation Resources page  
<https://www.netapp.com/us/documentation/hci.aspx>
- NetApp Product Documentation  
<https://www.netapp.com/us/documentation/index.aspx>
- Cleondris HCC product page  
<https://www.cleondris.com/en/hci-control-center.xhtml>
- Cleondris Support Portal  
<https://support.cleondris.com/>

# Artificial Intelligence (AI)

## NVA-1144: NetApp HCI AI Inferencing at the Edge Data Center with H615c and NVIDIA T4

Arvind Ramakrishnan, NetApp

This document describes how NetApp HCI can be designed to host artificial intelligence (AI) inferencing workloads at edge data center locations. The design is based on NVIDIA T4 GPU-powered NetApp HCI compute nodes, an NVIDIA Triton Inference Server, and a Kubernetes infrastructure built using NVIDIA DeepOps. The design also establishes the data pipeline between the core and edge data centers and illustrates implementation to complete the data lifecycle path.

Modern applications that are driven by AI and machine learning (ML) have pushed the limits of the internet. End users and devices demand access to applications, data, and services at any place and any time, with minimal latency. To meet these demands, data centers are moving closer to their users to boost performance, reduce back-and-forth data transfer, and provide cost-effective ways to meet user requirements.

In the context of AI, the core data center is a platform that provides centralized services, such as machine learning and analytics, and the edge data centers are where the real-time production data is subject to inferencing. These edge data centers are usually connected to a core data center. They provide end-user services and serve as a staging layer for data generated by IoT devices that need additional processing and that is too time sensitive to be transmitted back to a centralized core.

This document describes a reference architecture for AI inferencing that uses NetApp HCI as the base platform.

### Customer Value

NetApp HCI offers differentiation in the hyperconverged market for this inferencing solution, including the following advantages:

- A disaggregated architecture allows independent scaling of compute and storage and lowers the virtualization licensing costs and performance tax on independent NetApp HCI storage nodes.
- NetApp Element storage provides quality of service (QoS) for each storage volume, which provides guaranteed storage performance for workloads on NetApp HCI. Therefore, adjacent workloads do not negatively affect inferencing performance.
- A data fabric powered by NetApp allows data to be replicated from core to edge to cloud data centers, which moves data closer to where application needs it.
- With a data fabric powered by NetApp and NetApp FlexCache software, AI deep learning models trained on NetApp ONTAP AI can be accessed from NetApp HCI without having to export the model.
- NetApp HCI can host inference servers on the same infrastructure concurrently with multiple

workloads, either virtual-machine (VM) or container-based, without performance degradation.

- NetApp HCI is certified as NVIDIA GPU Cloud (NGC) ready for NVIDIA AI containerized applications.
- NGC-ready means that the stack is validated by NVIDIA, is purpose built for AI, and enterprise support is available through NGC Support Services.
- With its extensive AI portfolio, NetApp can support the entire spectrum of AI use cases from edge to core to cloud, including ONTAP AI for training and inferencing, Cloud Volumes Service and Azure NetApp Files for training in the cloud, and inferencing on the edge with NetApp HCI.

[Next: Use Cases](#)

## Use Cases

Although all applications today are not AI driven, they are evolving capabilities that allow them to access the immense benefits of AI. To support the adoption of AI, applications need an infrastructure that provides them with the resources needed to function at an optimum level and support their continuing evolution.

For AI-driven applications, edge locations act as a major source of data. Available data can be used for training when collected from multiple edge locations over a period of time to form a training dataset. The trained model can then be deployed back to the edge locations where the data was collected, enabling faster inferencing without the need to repeatedly transfer production data to a dedicated inferencing platform.

The NetApp HCI AI inferencing solution, powered by NetApp H615c compute nodes with NVIDIA T4 GPUs and NetApp cloud-connected storage systems, was developed and verified by NetApp and NVIDIA. NetApp HCI simplifies the deployment of AI inferencing solutions at edge data centers by addressing areas of ambiguity, eliminating complexities in the design and ending guesswork.

This solution gives IT organizations a prescriptive architecture that:

- Enables AI inferencing at edge data centers
- Optimizes consumption of GPU resources
- Provides a Kubernetes-based inferencing platform for flexibility and scalability
- Eliminates design complexities

Edge data centers manage and process data at locations that are very near to the generation point. This proximity increases the efficiency and reduces the latency involved in handling data. Many vertical markets have realized the benefits of an edge data center and are heavily adopting this distributed approach to data processing.

The following table lists the edge verticals and applications.

<b>Vertical</b>	<b>Applications</b>
Medical	Computer-aided diagnostics assist medical staff in early disease detection
Oil and gas	Autonomous inspection of remote production facilities, video, and image analytics
Aviation	Air traffic control assistance and real-time video feed analytics
Media and entertainment	Audio/video content filtering to deliver family-friendly content
Business analytics	Brand recognition to analyze brand appearance in live-streamed televised events
E-Commerce	Smart bundling of supplier offers to find ideal merchant and warehouse combinations
Retail	Automated checkout to recognize items a customer placed in cart and facilitate digital payment
Smart city	Improve traffic flow, optimize parking, and enhance pedestrian and cyclist safety
Manufacturing	Quality control, assembly-line monitoring, and defect identification
Customer service	Customer service automation to analyze and triage inquiries (phone, email, and social media)
Agriculture	Intelligent farm operation and activity planning, to optimize fertilizer and herbicide application

## Target Audience

The target audience for the solution includes the following groups:

- Data scientists
- IT architects
- Field consultants
- Professional services
- IT managers
- Anyone else who needs an infrastructure that delivers IT innovation and robust data and application services at edge locations

[Next: Architecture](#)

# Architecture

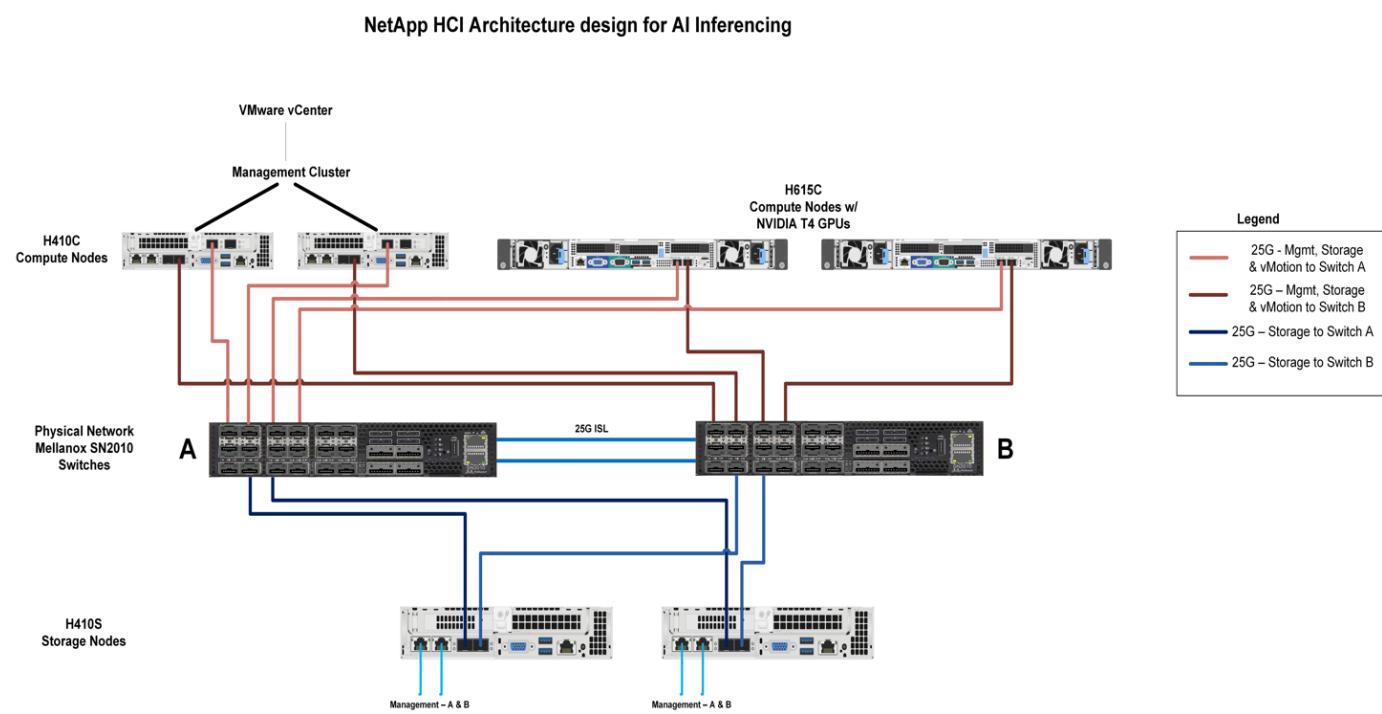
## Solution Technology

This solution is designed with a NetApp HCI system that contains the following components:

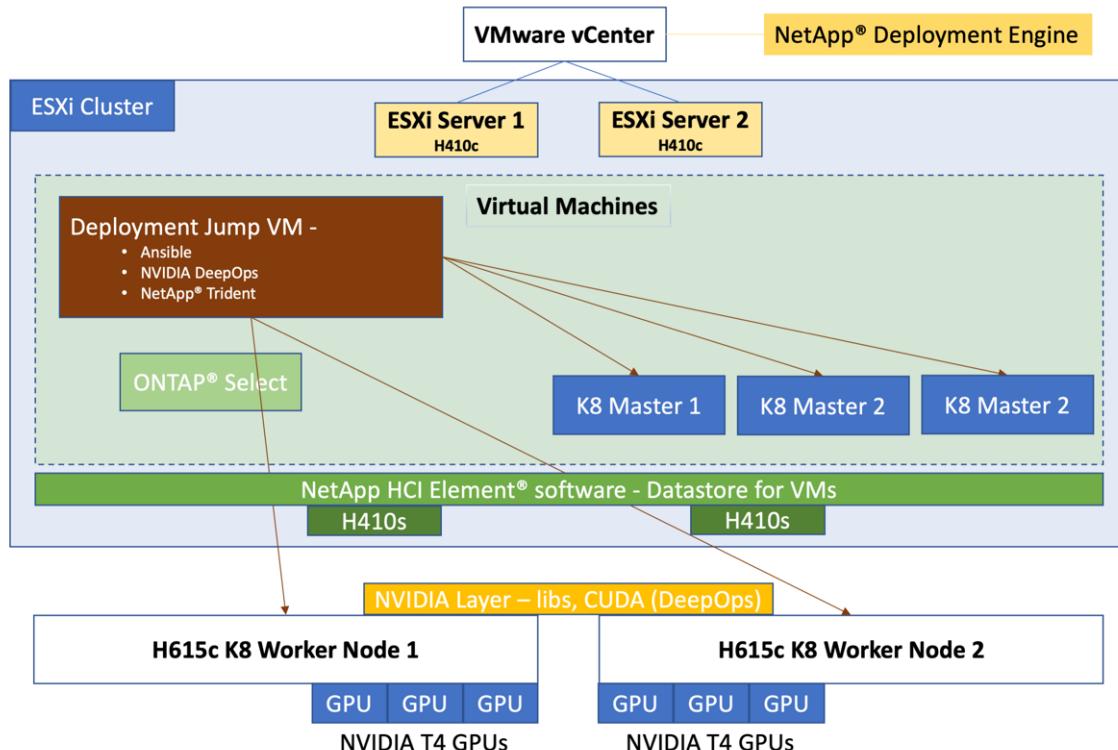
- Two H615c compute nodes with NVIDIA T4 GPUs
- Two H410c compute nodes
- Two H410s storage nodes
- Two Mellanox SN2010 10GbE/25GbE switches

## Architectural Diagram

The following diagram illustrates the solution architecture for the NetApp HCI AI inferencing solution.



The following diagram illustrates the virtual and physical elements of this solution.



A VMware infrastructure is used to host the management services required by this inferencing solution. These services do not need to be deployed on a dedicated infrastructure; they can coexist with any existing workloads. The NetApp Deployment Engine (NDE) uses the H410c and H410s nodes to deploy the VMware infrastructure.

After NDE has completed the configuration, the following components are deployed as VMs in the virtual infrastructure:

- **Deployment Jump VM.** Used to automate the deployment of NVIDIA DeepOps. See [NVIDIA DeepOps](#) and storage management using NetApp Trident.
- **ONTAP Select.** An instance of ONTAP Select is deployed to provide NFS file services and persistent storage to the AI workload running on Kubernetes.
- **Kubernetes Masters.** During deployment, three VMs are installed and configured with a supported Linux distribution and configured as Kubernetes master nodes. After the management services have been set up, two H615c compute nodes with NVIDIA T4 GPUs are installed with a supported Linux distribution. These two nodes function as the Kubernetes worker nodes and provide the infrastructure for the inferencing platform.

## Hardware Requirements

The following table lists the hardware components that are required to implement the solution. The hardware components that are used in any particular implementation of the solution might vary based on customer requirements.

<b>Layer</b>	<b>Product Family</b>	<b>Quantity</b>	<b>Details</b>
Compute	H615c	2	3 NVIDIA Tesla T4 GPUs per node
	H410c	2	Compute nodes for management infrastructure
Storage	H410s	2	Storage for OS and workload
Network	Mellanox SN2010	2	10G/25G switches

## Software Requirements

The following table lists the software components that are required to implement the solution. The software components that are used in any particular implementation of the solution might vary based on customer requirements.

<b>Layer</b>	<b>Software</b>	<b>Version</b>
Storage	NetApp Element software	12.0.0.333
	ONTAP Select	9.7
	NetApp Trident	20.07
NetApp HCI engine	NDE	1.8
Hypervisor	Hypervisor	VMware vSphere ESXi 6.7U1
	Hypervisor Management System	VMware vCenter Server 6.7U1
Inferencing Platform	NVIDIA DeepOps	20.08
	NVIDIA GPU Operator	1.1.7
	Ansible	2.9.5
	Kubernetes	1.17.9
	Docker	Docker CE 18.09.7
	CUDA Version	10.2
	GPU Device Plugin	0.6.0
	Helm	3.1.2
	NVIDIA Tesla Driver	440.64.00
	NVIDIA Triton Inference Server	2.1.0 – NGC Container v20.07

<b>Layer</b>	<b>Software</b>	<b>Version</b>
K8 Master VMs	Linux	<p>Any supported distribution across NetApp IMT, NVIDIA DeepOps, and GPUOperator</p> <p>Ubuntu 18.04.4 LTS was used in this solution</p> <p>Kernel version: 4.15</p>
Host OS/ K8 Worker Nodes	Linux	<p>Any supported distribution across NetApp IMT, NVIDIA DeepOps, and GPUOperator</p> <p>Ubuntu 18.04.4 LTS was used in this solution</p> <p>Kernel version: 4.15</p>

[Next: Design Considerations](#)

## Design Considerations

### Network Design

The switches used to handle the NetApp HCI traffic require a specific configuration for successful deployment.

Consult the NetApp HCI Network Setup Guide for the physical cabling and switch details. This solution uses a two-cable design for compute nodes. Optionally, compute nodes can be configured in a six-node cable design affording options for deployment of compute nodes.

The diagram under [Architecture](#) depicts the network topology of this NetApp HCI solution with a two-cable design for the compute nodes.

### Compute Design

The NetApp HCI compute nodes are available in two form factors, half-width and full-width, and in two rack unit sizes, 1 RU and 2 RU. The 410c nodes used in this solution are half-width and 1 RU and are housed in a chassis that can hold a maximum of four such nodes. The other compute node that is used in this solution is the H615c, which is a full-width node, 1 RU in size. The H410c nodes are based on Intel Skylake processors, and the H615c nodes are based on the second-generation Intel Cascade Lake processors. NVIDIA GPUs can be added to the H615c nodes, and each node can host a maximum of three NVIDIA Tesla T4 16GB GPUs.

The H615c nodes are the latest series of compute nodes for NetApp HCI and the second series that can support GPUs. The first model to support GPUs is the H610c node (full width, 2RU), which can support two NVIDIA Tesla M10 GPUs.

In this solution, H615c nodes are preferred over H610c nodes because of the following advantages:

- Reduced data center footprint, critical for edge deployments
- Support for a newer generation of GPUs designed for faster inferencing
- Reduced power consumption
- Reduced heat dissipation

## NVIDIA T4 GPUs

The resource requirements of inferencing are nowhere close to those of training workloads. In fact, most modern hand-held devices are capable of handling small amounts of inferencing without powerful resources like GPUs. However, for mission-critical applications and data centers that are dealing with a wide variety of applications that demand very low inferencing latencies while subject to extreme parallelization and massive input batch sizes, the GPUs play a key role in reducing inference time and help to boost application performance.

The NVIDIA Tesla T4 is an x16 PCIe Gen3 single-slot low-profile GPU based on the Turing architecture. The T4 GPUs deliver universal inference acceleration that spans applications such as image classification and tagging, video analytics, natural language processing, automatic speech recognition, and intelligent search. The breadth of the Tesla T4's inferencing capabilities enables it to be used in enterprise solutions and edge devices.

These GPUs are ideal for deployment in edge infrastructures due to their low power consumption and small PCIe form factor. The size of the T4 GPUs enables the installation of two T4 GPUs in the same space as a double-slot full-sized GPU. Although they are small, with 16GB memory, the T4s can support large ML models or run inference on multiple smaller models simultaneously.

The Turing- based T4 GPUs include an enhanced version of Tensor Cores and support a full range of precisions for inferencing FP32, FP16, INT8, and INT4. The GPU includes 2,560 CUDA cores and 320 Tensor Cores, delivering up to 130 tera operations per second (TOPS) of INT8 and up to 260 TOPS of INT4 inferencing performance. When compared to CPU-based inferencing, the Tesla T4, powered by the new Turing Tensor Cores, delivers up to 40 times higher inference performance.

The Turing Tensor Cores accelerate the matrix-matrix multiplication at the heart of neural network training and inferencing functions. They particularly excel at inference computations in which useful and relevant information can be inferred and delivered by a trained deep neural network based on a given input.

The Turing GPU architecture inherits the enhanced Multi-Process Service (MPS) feature that was introduced in the Volta architecture. Compared to Pascal-based Tesla GPUs, MPS on Tesla T4 improves inference performance for small batch sizes, reduces launch latency, improves QoS, and enables the servicing of higher numbers of concurrent client requests.

The NVIDIA T4 GPU is a part of the NVIDIA AI Inference Platform that supports all AI frameworks and provides comprehensive tooling and integrations to drastically simplify the development and

deployment of advanced AI.

## **Storage Design: Element Software**

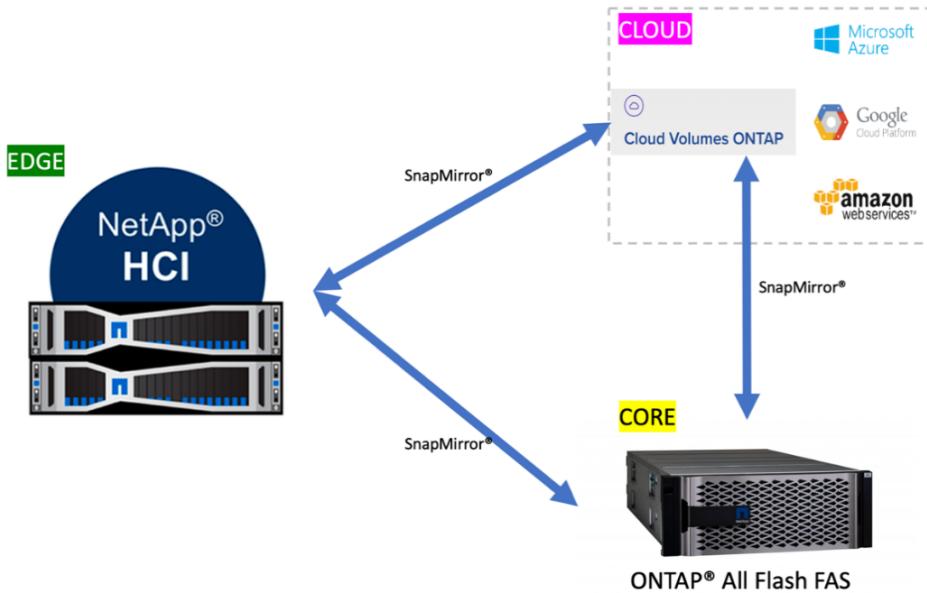
NetApp Element software powers the storage of the NetApp HCI systems. It delivers agile automation through scale-out flexibility and guaranteed application performance to accelerate new services.

Storage nodes can be added to the system non-disruptively in increments of one, and the storage resources are made available to the applications instantly. Every new node added to the system delivers a precise amount of additional performance and capacity to a usable pool. The data is automatically load balanced in the background across all nodes in the cluster, maintaining even utilization as the system grows.

Element software supports the NetApp HCI system to comfortably host multiple workloads by guaranteeing QoS to each workload. By providing fine-grained performance control with minimum, maximum, and burst settings for each workload, the software allows well-planned consolidations while protecting application performance. It decouples performance from capacity and allows each volume to be allocated with a specific amount of capacity and performance. These specifications can be modified dynamically without any interruption to data access.

As illustrated in the following figure, Element software integrates with NetApp ONTAP to enable data mobility between NetApp storage systems that are running different storage operating systems. Data can be moved from the Element software to ONTAP or vice versa by using NetApp SnapMirror technology. Element uses the same technology to provide cloud connectivity by integrating with NetApp Cloud Volumes ONTAP, which enables data mobility from the edge to the core and to multiple public cloud service providers.

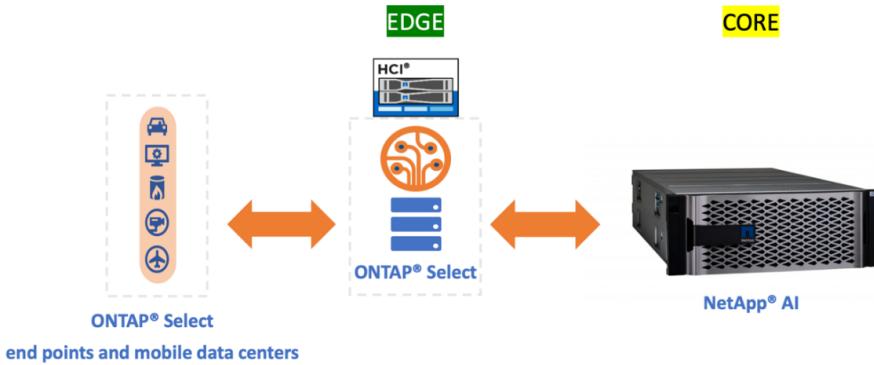
In this solution, the Element-backed storage provides the storage services that are required to run the workloads and applications on the NetApp HCI system.



## Storage Design: ONTAP Select

NetApp ONTAP Select introduces a software-defined data storage service model on top of NetApp HCI. It builds on NetApp HCI capabilities, adding a rich set of file and data services to the HCI platform while extending the data fabric.

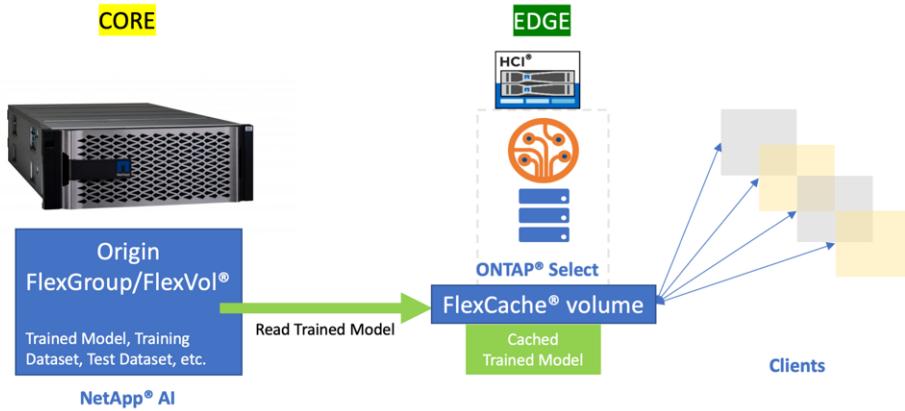
Although ONTAP Select is an optional component for implementing this solution, it does provide a host of benefits, including data gathering, protection, mobility, and so on, that are extremely useful in the context of the overall AI data lifecycle. It helps to simplify several day-to-day challenges for data handling, including ingestion, collection, training, deployment, and tiering.



ONTAP Select can run as a VM on VMware and still bring in most of the ONTAP capabilities that are available when it is running on a dedicated FAS platform, such as the following:

- Support for NFS and CIFS
- NetApp FlexClone technology
- NetApp FlexCache technology
- NetApp ONTAP FlexGroup volumes
- NetApp SnapMirror software

ONTAP Select can be used to leverage the FlexCache feature, which helps to reduce data-read latencies by caching frequently read data from a back-end origin volume, as is shown in the following figure. In the case of high-end inferencing applications with a lot of parallelization, multiple instances of the same model are deployed across the inferencing platform, leading to multiple reads of the same model. Newer versions of the trained model can be seamlessly introduced to the inferencing platform by verifying that the desired model is available in the origin or source volume.



## NetApp Trident

NetApp Trident is an open-source dynamic storage orchestrator that allows you to manage storage resources across all major NetApp storage platforms. It integrates with Kubernetes natively so that persistent volumes (PVs) can be provisioned on demand with native Kubernetes interfaces and constructs. Trident enables microservices and containerized applications to use enterprise-class storage services such as QoS, storage efficiencies, and cloning to meet the persistent storage demands of applications.

Containers are among the most popular methods of packaging and deploying applications, and Kubernetes is one of the most popular platforms for hosting containerized applications. In this solution, the inferencing platform is built on top of a Kubernetes infrastructure.

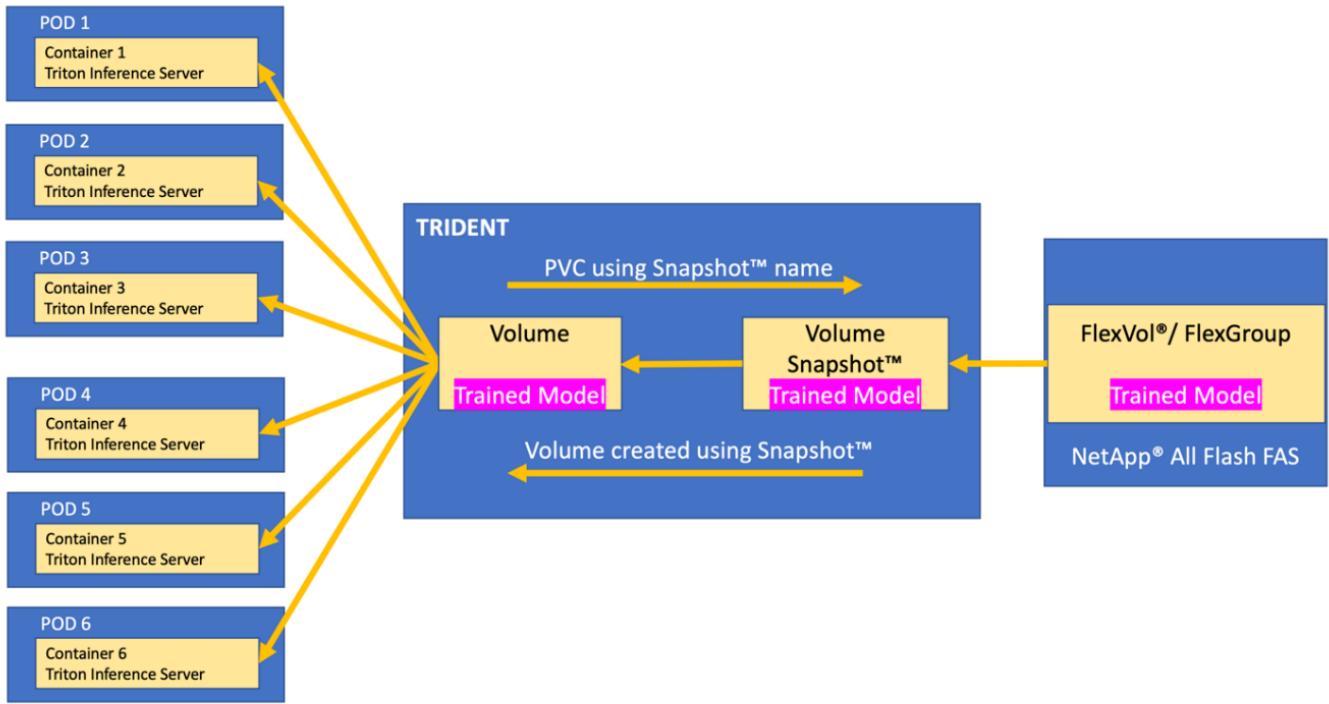
Trident currently supports storage orchestration across the following platforms:

- ONTAP: NetApp AFF, FAS, and Select
- Element software: NetApp HCI and NetApp SolidFire all-flash storage
- NetApp SANtricity software: E-Series and EF-series
- Cloud Volumes ONTAP
- Azure NetApp Files
- NetApp Cloud Volumes Service: AWS and Google Cloud

Trident is a simple but powerful tool to enable storage orchestration not just across multiple storage platforms, but also across the entire spectrum of the AI data lifecycle, ranging from the edge to the core

to the cloud.

Trident can be used to provision a PV from a NetApp Snapshot copy that makes up the trained model. The following figure illustrates the Trident workflow in which a persistent volume claim (PVC) is created by referring to an existing Snapshot copy. Following this, Trident creates a volume by using the Snapshot copy.



This method of introducing trained models from a Snapshot copy supports robust model versioning. It simplifies the process of introducing newer versions of models to applications and switching inferencing between different versions of the model.

## NVIDIA DeepOps

NVIDIA DeepOps is a modular collection of Ansible scripts that can be used to automate the deployment of a Kubernetes infrastructure. There are multiple deployment tools available that can automate the deployment of a Kubernetes cluster. In this solution, DeepOps is the preferred choice because it does not just deploy a Kubernetes infrastructure, it also installs the necessary GPU drivers, NVIDIA Container Runtime for Docker (nvidia-docker2), and various other dependencies for GPU-accelerated work. It encapsulates the best practices for NVIDIA GPUs and can be customized or run as individual components as needed.

DeepOps internally uses Kubespray to deploy Kubernetes, and it is included as a submodule in DeepOps. Therefore, common Kubernetes cluster management operations such as adding nodes, removing nodes, and cluster upgrades should be performed using Kubespray.

A software based L2 LoadBalancer using MetallB and an Ingress Controller based on NGINX are also

deployed as part of this solution by using the scripts that are available with DeepOps.

In this solution, three Kubernetes master nodes are deployed as VMs, and the two H615c compute nodes with NVIDIA Tesla T4 GPUs are set up as Kubernetes worker nodes.

## NVIDIA GPU Operator

The GPU operator deploys the NVIDIA k8s-device-plugin for GPU support and runs the NVIDIA drivers as containers. It is based on the Kubernetes operator framework, which helps to automate the management of all NVIDIA software components that are needed to provision GPUs. The components include NVIDIA drivers, Kubernetes device plug-in for GPUs, the NVIDIA container runtime, and automatic node labeling, which is used in tandem with Kubernetes Node Feature Discovery.

The GPU operator is an important component of the [NVIDIA EGX](#) software-defined platform that is designed to make large-scale hybrid-cloud and edge operations possible and efficient. It is specifically useful when the Kubernetes cluster needs to scale quickly—for example, when provisioning additional GPU-based worker nodes and managing the lifecycle of the underlying software components. Because the GPU operator runs everything as containers, including NVIDIA drivers, administrators can easily swap various components by simply starting or stopping containers.

## NVIDIA Triton Inference Server

NVIDIA Triton Inference Server (Triton Server) simplifies the deployment of AI inferencing solutions in production data centers. This microservice is specifically designed for inferencing in production data centers. It maximizes GPU utilization and integrates seamlessly into DevOps deployments with Docker and Kubernetes.

Triton Server provides a common solution for AI inferencing. Therefore, researchers can focus on creating high-quality trained models, DevOps engineers can focus on deployment, and developers can focus on applications without the need to redesign the platform for each AI-powered application.

Here are some of the key features of Triton Server:

- **Support for multiple frameworks.** Triton Server can handle a mix of models, and the number of models is limited only by system disk and memory resources. It can support the TensorRT, TensorFlow GraphDef, TensorFlow SavedModel, ONNX, PyTorch, and Caffe2 NetDef model formats.
- \*Concurrent model execution. \*Multiple models or multiple instances of the same model can be run simultaneously on a GPU.
- **Multi-GPU support.** Triton Server can maximize GPU utilization by enabling inference for multiple models on one or more GPUs.
- **Support for batching.** Triton Server can accept requests for a batch of inputs and respond with the corresponding batch of outputs. The inference server supports multiple scheduling and batching algorithms that combine individual inference requests together to improve inference throughput. Batching algorithms are available for both stateless and stateful applications and need to be used appropriately. These scheduling and batching decisions are transparent to the client that is

requesting inference.

- **Ensemble support.** An ensemble is a pipeline with multiple models with connections of input and output tensors between those models. An inference request can be made to an ensemble, which results in the execution of the complete pipeline.
- **Metrics.** Metrics are details about GPU utilization, server throughput, server latency, and health for auto scaling and load balancing.

NetApp HCI is a hybrid multi-cloud infrastructure that can host multiple workloads and applications, and the Triton Inference Server is well equipped to support the inferencing requirements of multiple applications.

In this solution, Triton Server is deployed on the Kubernetes cluster using a deployment file. With this method, the default configuration of Triton Server can be overridden and customized as required. Triton Server also provides an inference service using an HTTP or GRPC endpoint, allowing remote clients to request inferencing for any model that is being managed by the server.

A Persistent Volume is presented via NetApp Trident to the container that runs the Triton Inference Server and this persistent volume is configured as the model repository for the Inference server.

The Triton Inference Server is deployed with varying sets of resources using Kubernetes deployment files, and each server instance is presented with a LoadBalancer front end for seamless scalability. This approach also illustrates the flexibility and simplicity with which resources can be allocated to the inferencing workloads.

[Next: Deploying NetApp HCI – AI Inferencing at the Edge](#)

## Overview

This section describes the steps required to deploy the AI inferencing platform using NetApp HCI. The following list provides the high-level tasks involved in the setup:

1. [Configure network switches](#)
2. [Deploy the VMware virtual infrastructure on NetApp HCI using NDE](#)
3. [Configure the H615c compute nodes to be used as K8 worker nodes](#)
4. [Set up the deployment jump VM and K8 master VMs](#)
5. [Deploy a Kubernetes cluster with NVIDIA DeepOps](#)
6. [Deploy ONTAP Select within the virtual infrastructure](#)
7. [Deploy NetApp Trident](#)
8. [Deploy NVIDIA Triton inference Server](#)
9. [Deploy the client for the Triton inference server](#)

## 10. Collect inference metrics from the Triton inference server

Next: Configure Network Switches

### Configure Network Switches (Automated Deployment)

#### Prepare Required VLAN IDs

The following table lists the necessary VLANs for deployment, as outlined in this solution validation. You should configure these VLANs on the network switches prior to executing NDE.

Network Segment	Details	VLAN ID
Out-of-band management network	Network for HCI terminal user interface (TUI)	16
In-band management network	Network for accessing management interfaces of nodes, hosts, and guests	3488
VMware vMotion	Network for live migration of VMs	3489
iSCSI SAN storage	Network for iSCSI storage traffic	3490
Application	Network for Application traffic	3487
NFS	Network for NFS storage traffic	3491
IPL*	Interpeer link between Mellanox switches	4000
Native	Native VLAN	2

\*Only for Mellanox switches

#### Switch Configuration

This solution uses Mellanox SN2010 switches running Onyx. The Mellanox switches are configured using an Ansible playbook. Prior to running the Ansible playbook, you should perform the initial configuration of the switches manually:

1. Install and cable the switches to the uplink switch, compute, and storage nodes.
2. Power on the switches and configure them with the following details:
  - a. Host name
  - b. Management IP and gateway
  - c. NTP
3. Log into the Mellanox switches and run the following commands:

```
configuration write to pre-ansible  
configuration write to post-ansible
```

The **pre-ansible** configuration file created can be used to restore the switch's configuration to the state before the Ansible playbook execution.

The switch configuration for this solution is stored in the **post-ansible** configuration file.



The configuration playbook for Mellanox switches that follows best practices and requirements for NetApp HCI can be downloaded [here](#).

4. Fill out the credentials to access the switches and variables needed for the environment. The following text is a sample of the variable file for this solution.

```
# vars file for nar_hci_mellanox_deploy  
#These set of variables will setup the Mellanox switches for NetApp HCI that uses a 2-cable compute connectivity option.  
#Ansible connection variables for mellanox  
ansible_connection: network_cli  
ansible_network_os: onyx  
#-----  
# Primary Variables  
#-----  
#Necessary VLANs for Standard NetApp HCI Deployment [native, Management, iSCSI_Storage, vMotion, VM_Network, IPL]  
#Any additional VLANs can be added to this in the prescribed format below  
netapp_hci_vlans:  
- {vlan_id: 2 , vlan_name: "Native" }  
- {vlan_id: 3488 , vlan_name: "IB-Management" }  
- {vlan_id: 3490 , vlan_name: "iSCSI_Storage" }  
- {vlan_id: 3489 , vlan_name: "vMotion" }  
- {vlan_id: 3491 , vlan_name: "NFS " }  
- {vlan_id: 3487 , vlan_name: "App_Network" }  
- {vlan_id: 4000 , vlan_name: "IPL" }#Modify the VLAN IDs to suit your environment  
#Spanning-tree protocol type for uplink connections.  
#The valid options are 'network' and 'normal'; selection depends on the uplink switch model.  
uplink_stp_type: network  
#-----  
# IPL variables  
#-----  
#Inter-Peer Link Portchannel  
#ipl_portchannel to be defined in the format - Po100  
ipl_portchannel: Po100  
#Inter-Peer Link Addresses
```

```

#The IPL IP address should not be part of the management network. This is typically a
private network
ipl_ipaddr_a: 10.0.0.1
ipl_ipaddr_b: 10.0.0.2
#Define the subnet mask in CIDR number format. Eg: For subnet /22, use ipl_ip_subnet:
22
ipl_ip_subnet: 24
#Inter-Peer Link Interfaces
#members to be defined with Eth in the format. Eg: Eth1/1
peer_link_interfaces:
  members: ['Eth1/20', 'Eth1/22']
  description: "peer link interfaces"
#MLAG VIP IP address should be in the same subnet as that of the switches' mgmt0
interface subnet
#mlag_vip_ip to be defined in the format - <vip_ip>/<subnet_mask>. Eg: x.x.x.x/y
mlag_vip_ip: <><mlag_vip_ip>>
#MLAG VIP Domain Name
#The mlag domain must be unique name for each mlag domain.
#In case you have more than one pair of MLAG switches on the same network, each domain
(consist of two switches) should be configured with different name.
mlag_domain_name: MLAG-VIP-DOM
-----
# Interface Details
-----
#Storage Bond10G Interface details
#members to be defined with Eth in the format. Eg: Eth1/1
#Only numerical digits between 100 to 1000 allowed for mlag_id
#Operational link speed [variable 'speed' below] to be defined in terms of bytes.
#For 10 Gigabyte operational speed, define 10G. [Possible values - 10G and 25G]
#Interface descriptions append storage node data port numbers assuming all Storage
Nodes' Port C -> Mellanox Switch A and all Storage Nodes' Port D -> Mellanox Switch B
#List the storage Bond10G interfaces, their description, speed and MLAG IDs in list of
dictionaries format
storage_interfaces:
- {members: "Eth1/1", description: "HCI_Storage_Node_01", mlag_id: 101, speed: 25G}
- {members: "Eth1/2", description: "HCI_Storage_Node_02", mlag_id: 102, speed: 25G}
#In case of additional storage nodes, add them here
#Storage Bond1G Interface
#Mention whether or not these Mellanox switches will also be used for Storage Node
Mgmt connections
#Possible inputs for storage_mgmt are 'yes' and 'no'
storage_mgmt: <><yes or no>>
#Storage Bond1G (Mgmt) interface details. Only if 'storage_mgmt' is set to 'yes'
#Members to be defined with Eth in the format. Eg: Eth1/1
#Interface descriptions append storage node management port numbers assuming all
Storage Nodes' Port A -> Mellanox Switch A and all Storage Nodes' Port B -> Mellanox
Switch B
#List the storage Bond1G interfaces and their description in list of dictionaries

```

```
format
storage_mgmt_interfaces:
- {members: "Ethx/y", description: "HCI_Storage_Node_01"}
- {members: "Ethx/y", description: "HCI_Storage_Node_02"}
#In case of additional storage nodes, add them here
#LACP load balancing algorithm for IP hash method
#Possible options are: 'destination-mac', 'destination-ip', 'destination-port',
'source-mac', 'source-ip', 'source-port', 'source-destination-mac', 'source-
destination-ip', 'source-destination-port'
#This variable takes multiple options in a single go
#For eg: if you want to configure load to be distributed in the port-channel based on
the traffic source and destination IP address and port number, use 'source-
destination-ip source-destination-port'
#By default, Mellanox sets it to source-destination-mac. Enter the values below only
if you intend to configure any other load balancing algorithm
#Make sure the load balancing algorithm that is set here is also replicated on the
host side
#Recommended algorithm is source-destination-ip source-destination-port
#Fill the lacp_load_balance variable only if you are using configuring interfaces on
compute nodes in bond or LAG with LACP
lacp_load_balance: "source-destination-ip source-destination-port"
#Compute Interface details
#Members to be defined with Eth in the format. Eg: Eth1/1
#Fill the mlag_id field only if you intend to configure interfaces of compute nodes
into bond or LAG with LACP
#In case you do not intend to configure LACP on interfaces of compute nodes, either
leave the mlag_id field unfilled or comment it or enter NA in the mlag_id field
#In case you have a mixed architecture where some compute nodes require LACP and some
don't,
#1. Fill the mlag_id field with appropriate MLAG ID for interfaces that connect to
compute nodes requiring LACP
#2. Either fill NA or leave the mlag_id field blank or comment it for interfaces
connecting to compute nodes that do not require LACP
#Only numerical digits between 100 to 1000 allowed for mlag_id.
#Operational link speed [variable 'speed' below] to be defined in terms of bytes.
#For 10 Gigabyte operational speed, define 10G. [Possible values - 10G and 25G]
#Interface descriptions append compute node port numbers assuming all Compute Nodes'
Port D -> Mellanox Switch A and all Compute Nodes' Port E -> Mellanox Switch B
#List the compute interfaces, their speed, MLAG IDs and their description in list of
dictionaries format
compute_interfaces:
- members: "Eth1/7"#Compute Node for ESXi, setup by NDE
  description: "HCI_Compute_Node_01"
  mlag_id: #Fill the mlag_id only if you wish to use LACP on interfaces towards
compute nodes
  speed: 25G
- members: "Eth1/8"#Compute Node for ESXi, setup by NDE
  description: "HCI_Compute_Node_02"
```

```

    mlag_id: #Fill the mlag_id only if you wish to use LACP on interfaces towards
    compute nodes
    speed: 25G
#In case of additional compute nodes, add them here in the same format as above-
members: "Eth1/9">#Compute Node for Kubernetes Worker node
    description: "HCI_Compute_Node_01"
    mlag_id: 109 #Fill the mlag_id only if you wish to use LACP on interfaces towards
    compute nodes
    speed: 10G
- members: "Eth1/10"#Compute Node for Kubernetes Worker node
    description: "HCI_Compute_Node_02"
    mlag_id: 110 #Fill the mlag_id only if you wish to use LACP on interfaces towards
    compute nodes
    speed: 10G
#Uplink Switch LACP support
#Possible options are 'yes' and 'no' - Set to 'yes' only if your uplink switch
supports LACP
uplink_switch_lacp: <>yes or no>>
#Uplink Interface details
#Members to be defined with Eth in the format. Eg: Eth1/1
#Only numerical digits between 100 to 1000 allowed for mlag_id.
#Operational link speed [variable 'speed' below] to be defined in terms of bytes.
#For 10 Gigabyte operational speed, define 10G. [Possible values in Mellanox are 1G,
10G and 25G]
#List the uplink interfaces, their description, MLAG IDs and their speed in list of
dictionaries format
uplink_interfaces:
- members: "Eth1/18"
    description_switch_a: "SwitchA:Ethx/y -> Uplink_Switch:Ethx/y"
    description_switch_b: "SwitchB:Ethx/y -> Uplink_Switch:Ethx/y"
    mlag_id: 118 #Fill the mlag_id only if 'uplink_switch_lacp' is set to 'yes'
    speed: 10G
    mtu: 1500

```



The fingerprint for the switch's key must match with that present in the host machine from where the playbook is being executed. To ensure this, add the key to `/root/.ssh/known_host` or any other appropriate location.

## Rollback the Switch Configuration

1. In case of any timeout failures or partial configuration, run the following command to roll back the switch to the initial state.

```
configuration switch-to pre-ansible
```



This operation requires a reboot of the switch.

2. Switch the configuration to the state before running the Ansible playbook.

```
configuration delete post-ansible
```

3. Delete the post-ansible file that had the configuration from the Ansible playbook.

```
configuration write to post-ansible
```

4. Create a new file with the same name post-ansible, write the pre-ansible configuration to it, and switch to the new configuration to restart configuration.

#### IP Address Requirements

The deployment of the NetApp HCI inferencing platform with VMware and Kubernetes requires multiple IP addresses to be allocated. The following table lists the number of IP addresses required. Unless otherwise indicated, addresses are assigned automatically by NDE.

IP Address Quantity	Details	VLAN ID	IP Address
One per storage and compute node*	HCI terminal user interface (TUI) addresses	16	
One per vCenter Server (VM)	vCenter Server management address	3488	
One per management node (VM)	Management node IP address		
One per ESXi host	ESXi compute management addresses		
One per storage/witness node	NetApp HCI storage node management addresses		
One per storage cluster	Storage cluster management address		
One per ESXi host	VMware vMotion address	3489	
Two per ESXi host	ESXi host initiator address for iSCSI storage traffic	3490	

IP Address Quantity	Details	VLAN ID	IP Address
Two per storage node	Storage node target address for iSCSI storage traffic		
Two per storage cluster	Storage cluster target address for iSCSI storage traffic		
Two for mNode	mNode iSCSI storage access		

The following IPs are assigned manually when the respective components are configured.

IP Address Quantity	Details	VLAN ID	IP Address
One for Deployment Jump Management network	Deployment Jump VM to execute Ansible playbooks and configure other parts of the system – management connectivity	3488	
One per Kubernetes master node – management network	Kubernetes master node VMs (three nodes)	3488	
One per Kubernetes worker node – management network	Kubernetes worker nodes (two nodes)	3488	
One per Kubernetes worker node – NFS network	Kubernetes worker nodes (two nodes)	3491	
One per Kubernetes worker node – application network	Kubernetes worker nodes (two nodes)	3487	
Three for ONTAP Select – management network	ONTAP Select VM	3488	
One for ONTAP Select – NFS network	ONTAP Select VM – NFS data traffic	3491	
At least two for Triton Inference Server Load Balancer – application network	Load balancer IP range for Kubernetes load balancer service	3487	

\*This validation requires the initial setup of the first storage node TUI address. NDE automatically assigns the TUI address for subsequent nodes.

### DNS and Timekeeping Requirement

Depending on your deployment, you might need to prepare DNS records for your NetApp HCI system. NetApp HCI requires a valid NTP server for timekeeping; you can use a publicly available time server if you do not have one in your environment.

This validation involves deploying NetApp HCI with a new VMware vCenter Server instance using a fully qualified domain name (FQDN). Before deployment, you must have one Pointer (PTR) record and one Address (A) record created on the DNS server.

[Next: Virtual Infrastructure with Automated Deployment](#)

## Deploy VMware Virtual Infrastructure on NetApp HCI with NDE (Automated Deployment)

### NDE Deployment Prerequisites

Consult the [NetApp HCI Prerequisites Checklist](#) to see the requirements and recommendations for NetApp HCI before you begin deployment.

1. Network and switch requirements and configuration
2. Prepare required VLAN IDs
3. Switch configuration
4. IP Address Requirements for NetApp HCI and VMware
5. DNS and time-keeping requirements
6. Final preparations

### NDE Execution

Before you execute the NDE, you must complete the rack and stack of all components, configuration of the network switches, and verification of all prerequisites. You can execute NDE by connecting to the management address of a single storage node if you plan to allow NDE to automatically configure all addresses.

NDE performs the following tasks to bring an HCI system online:

1. Installs the storage node (NetApp Element software) on a minimum of two storage nodes.
2. Installs the VMware hypervisor on a minimum of two compute nodes.
3. Installs VMware vCenter to manage the entire NetApp HCI stack.
4. Installs and configures the NetApp storage management node (mNode) and NetApp Monitoring Agent.



This validation uses NDE to automatically configure all addresses. You can also set up DHCP in your environment or manually assign IP addresses for each storage node and compute node. These steps are not covered in this guide.

As mentioned previously, this validation uses a two-cable configuration for compute nodes.

Detailed steps for the NDE are not covered in this document.

For step-by-step guidance on completing the deployment of the base NetApp HCI platform, see the [Deployment guide](#).

5. After NDE has finished, login to the vCenter and create a Distributed Port Group [NetApp HCI VDS 01-NFS\\_Network](#) for the NFS network to be used by ONTAP Select and the application.

[Next: Configure NetApp H615c \(Manual Deployment\)](#)

### Configure NetApp H615c (Manual Deployment)

In this solution, the NetApp H615c compute nodes are configured as Kubernetes worker nodes. The Inferencing workload is hosted on these nodes.

Deploying the compute nodes involves the following tasks:

- Install Ubuntu 18.04.4 LTS.
- Configure networking for data and management access.
- Prepare the Ubuntu instances for Kubernetes deployment.

#### Install Ubuntu 18.04.4 LTS

The following high-level steps are required to install the operating system on the H615c compute nodes:

1. Download Ubuntu 18.04.4 LTS from [Ubuntu releases](#).
2. Using a browser, connect to the IPMI of the H615c node and launch Remote Control.
3. Map the Ubuntu ISO using the Virtual Media Wizard and start the installation.
4. Select one of the two physical interfaces as the [Primary network interface](#) when prompted.

An IP from a DHCP source is allocated when available, or you can switch to a manual IP configuration later. The network configuration is modified to a bond-based setup after the OS has been installed.

5. Provide a hostname followed by a domain name.
6. Create a user and provide a password.
7. Partition the disks according to your requirements.

8. Under Software Selection, select **OpenSSH server** and click Continue.

9. Reboot the node.

### Configure Networking for Data and Management Access

The two physical network interfaces of the Kubernetes worker nodes are set up as a bond and VLAN interfaces for management and application, and NFS data traffic is created on top of it.



The inferencing applications and associated containers use the application network for connectivity.

1. Connect to the console of the Ubuntu instance as a user with root privileges and launch a terminal session.
2. Navigate to `/etc/netplan` and open the `01-netcfg.yaml` file.
3. Update the netplan file based on the network details for the management, application, and NFS traffic in your environment.

The following template of the netplan file was used in this solution:

```
# This file describes the network interfaces available on your system
# For more information, see netplan(5).
network:
  version: 2
  renderer: networkd
  ethernets:
    enp59s0f0: #Physical Interface 1
      match:
        macaddress: <<mac_address Physical Interface 1>>
      set-name: enp59s0f0
      mtu: 9000
    enp59s0f1: # Physical Interface 2
      match:
        macaddress: <<mac_address Physical Interface 2>>
      set-name: enp59s0f1
      mtu: 9000
  bonds:
    bond0:
      mtu: 9000
      dhcp4: false
      dhcp6: false
      interfaces: [ enp59s0f0, enp59s0f1 ]
      parameters:
        mode: 802.3ad
        mii-monitor-interval: 100
  vlans:
```

```
vlan.3488: #Management VLAN
  id: 3488
  xref: bond0
  dhcp4: false
  addresses: [ipv4_address/subnet]
  routes:
    - to: 0.0.0.0/0
      via: 172.21.232.111
      metric: 100
      table: 3488
    - to: x.x.x.x/x # Additional routes if any
      via: y.y.y.y
      metric: <>metric>>
      table: <>table #>>
  routing-policy:
    - from: 0.0.0.0/0
      priority: 32768#Higher Priority than table 3487
      table: 3488
  nameservers:
    addresses: [nameserver_ip]
    search: [ search_domain ]
  mtu: 1500
vlan.3487:
  id: 3487
  xref: bond0
  dhcp4: false
  addresses: [ipv4_address/subnet]
  routes:
    - to: 0.0.0.0/0
      via: 172.21.231.111
      metric: 101
      table: 3487
    - to: x.x.x.x/x
      via: y.y.y.y
      metric: <>metric>>
      table: <>table #>>
  routing-policy:
    - from: 0.0.0.0/0
      priority: 32769#Lower Priority
      table: 3487
  nameservers:
    addresses: [nameserver_ip]
    search: [ search_domain ]
  mtu: 1500    vlan.3491:
  id: 3491
  xref: bond0
  dhcp4: false
  addresses: [ipv4_address/subnet]
```

```
mtu: 9000
```

4. Confirm that the priorities for the routing policies are lower than the priorities for the main and default tables.
5. Apply the netplan.

```
sudo netplan --debug apply
```

6. Make sure that there are no errors.
7. If Network Manager is running, stop and disable it.

```
systemctl stop NetworkManager  
systemctl disable NetworkManager
```

8. Add a host record for the server in DNS.
9. Open a VI editor to [/etc/iproute2/rt\\_tables](#) and add the two entries.

```
#  
# reserved values  
#  
255    local  
254    main  
253    default  
0      unspec  
#  
# local  
#  
#1     inr.ruhep  
101   3488  
102   3487
```

10. Match the table number to what you used in the netplan.
11. Open a VI editor to [/etc/sysctl.conf](#) and set the value of the following parameters.

```
net.ipv4.conf.default.rp_filter=0  
net.ipv4.conf.all.rp_filter=0net.ipv4.ip_forward=1
```

12. Update the system.

```
sudo apt-get update && sudo apt-get upgrade
```

13. Reboot the system
14. Repeat steps 1 through 13 for the other Ubuntu instance.

[Next: Set Up the Deployment Jump and the Kubernetes Master Node VMs \(Manual Deployment\)](#)

### **Set Up the Deployment Jump VM and the Kubernetes Master Node VMs (Manual Deployment)**

A Deployment Jump VM running a Linux distribution is used for the following purposes:

- Deploying ONTAP Select using an Ansible playbook
- Deploying the Kubernetes infrastructure with NVIDIA DeepOps and GPU Operator
- Installing and configuring NetApp Trident

Three more VMs running Linux are set up; these VMs are configured as Kubernetes Master Nodes in this solution.

Ubuntu 18.04.4 LTS was used in this solution deployment.

1. Deploy the Ubuntu 18.04.4 LTS VM with VMware tools

You can refer to the high-level steps described in section [Install Ubuntu 18.04.4 LTS](#).

2. Configure the in-band management network for the VM. See the following sample netplan template:

```
# This file describes the network interfaces available on your system
# For more information, see netplan(5).
network:
  version: 2
  renderer: networkd
  ethernets:
    ens160:
      dhcp4: false
      addresses: [ipv4_address/subnet]
      routes:
        - to: 0.0.0.0/0
          via: 172.21.232.111
          metric: 100
          table: 3488
      routing-policy:
        - from: 0.0.0.0/0
          priority: 32768
          table: 3488
      nameservers:
        addresses: [nameserver_ip]
        search: [ search_domain ]
    mtu: 1500
```

This template is not the only way to setup the network. You can use any other approach that you prefer.

3. Apply the netplan.

```
sudo netplan --debug apply
```

4. Stop and disable Network Manager if it is running.

```
systemctl stop NetworkManager
systemctl disable NetworkManager
```

5. Open a VI editor to [/etc/iproute2/rt\\_tables](#) and add a table entry.

```
#  
# reserved values  
#  
255    local  
254    main  
253    default  
0      unspec  
#  
# local  
#  
#1      inr.ruhep  
101    3488
```

6. Add a host record for the VM in DNS.
7. Verify outbound internet access.
8. Update the system.

```
sudo apt-get update && sudo apt-get upgrade
```

9. Reboot the system.
10. Repeat steps 1 through 9 to set up the other three VMs.

[Next: Deploy a Kubernetes Cluster with NVIDIA DeepOps \(Automated Deployment\)](#)

### Deploy a Kubernetes Cluster with NVIDIA DeepOps Automated Deployment

To deploy and configure the Kubernetes Cluster with NVIDIA DeepOps, complete the following steps:

1. Make sure that the same user account is present on all the Kubernetes master and worker nodes.
2. Clone the DeepOps repository.

```
git clone https://github.com/NVIDIA/deepops.git
```

3. Check out a recent release tag.

```
cd deepops  
git checkout tags/20.08
```

If this step is skipped, the latest development code is used, not an official release.

4. Prepare the Deployment Jump by installing the necessary prerequisites.

```
./scripts/setup.sh
```

5. Create and edit the Ansible inventory by opening a VI editor to [deepops/config/inventory](#).

- a. List all the master and worker nodes under [all].
- b. List all the master nodes under [kube-master]
- c. List all the master nodes under [etcd]
- d. List all the worker nodes under [kube-node]

```
#####
# ALL NODES
# NOTE: Use existing hostnames here, DeepOps will config them
#####
[all]
hci-ai-k8-master-01      ansible_host=172.21.232.114
hci-ai-k8-master-02      ansible_host=172.21.232.115
hci-ai-k8-master-03      ansible_host=172.21.232.116
hci-ai-k8-worker-01      ansible_host=172.21.232.109
hci-ai-k8-worker-02      ansible_host=172.21.232.110

#####
# KUBERNETES
#####
[kube-master]
hci-ai-k8-master-01
hci-ai-k8-master-02
hci-ai-k8-master-03

# Odd number of nodes required
[etcd]
hci-ai-k8-master-01
hci-ai-k8-master-02
hci-ai-k8-master-03

# Also add mgmt/master nodes here if they will run non kube-node
[kube-node]
hci-ai-k8-worker-01
hci-ai-k8-worker-02

[k8s-cluster:children]
kube-master
kube-node
```

6. Enable GPUOperator by opening a VI editor to `deepops/config/group_vars/k8s-cluster.yml`.

```
# Provide option to use GPU Operator instead of setting up NVIDIA driver and
# Docker configuration.
deepops_gpu_operator_enabled: true
```

7. Set the value of `deepops_gpu_operator_enabled` to true.

8. Verify the permissions and network configuration.

```
ansible all -m raw -a "hostname" -k -K
```

- If SSH to the remote hosts requires a password, use -k.
- If sudo on the remote hosts requires a password, use -K.

9. If the previous step passed without any issues, proceed with the setup of Kubernetes.

```
ansible-playbook --limit k8s-cluster playbooks/k8s-cluster.yml -k -K
```

10. To verify the status of the Kubernetes nodes and the pods, run the following commands:

```
kubectl get nodes
```

```
rarvind@deployment-jump:~/deepops$ kubectl get nodes
NAME           STATUS  ROLES   AGE    VERSION
hci-ai-k8-master-01  Ready  master  2d19h  v1.17.6
hci-ai-k8-master-02  Ready  master  2d19h  v1.17.6
hci-ai-k8-master-03  Ready  master  2d19h  v1.17.6
hci-ai-k8-worker-01  Ready  <none>  2d19h  v1.17.6
hci-ai-k8-worker-02  Ready  <none>  2d19h  v1.17.6
```

```
kubectl get pods -A
```

It can take a few minutes for all the pods to run.

NAMESPACE	NAME	READY	STATUS
default	gpu-operator-74c97448d9-ppdlc	1/1	Running
default	nvidia-gpu-operator-node-feature-discovery-master-ffccb57dx9wtl	1/1	Running
default	nvidia-gpu-operator-node-feature-discovery-worker-2lr9t	1/1	Running
default	nvidia-gpu-operator-node-feature-discovery-worker-616x7	1/1	Running
default	nvidia-gpu-operator-node-feature-discovery-worker-jf696	1/1	Running
default	nvidia-gpu-operator-node-feature-discovery-worker-tmtwv	1/1	Running
default	nvidia-gpu-operator-node-feature-discovery-worker-z4nlh	1/1	Running
gpu-operator-resources	nvidia-container-toolkit-daemonset-7jbl4	1/1	Running
gpu-operator-resources	nvidia-container-toolkit-daemonset-x5ktb	1/1	Running
gpu-operator-resources	nvidia-dcgm-exporter-5x94p	1/1	Running
gpu-operator-resources	nvidia-dcgm-exporter-7cb1	1/1	Running
gpu-operator-resources	nvidia-device-plugin-daemonset-n8vrk	1/1	Running
gpu-operator-resources	nvidia-device-plugin-daemonset-z7j6s	1/1	Running
gpu-operator-resources	nvidia-device-plugin-validation	0/1	Completed
gpu-operator-resources	nvidia-driver-daemonset-7h752	1/1	Running
gpu-operator-resources	nvidia-driver-daemonset-v4rbj	1/1	Running
gpu-operator-resources	nvidia-driver-validation	0/1	Completed
kube-system	calico-kube-controllers-777478f4ff-jknxg	1/1	Running
kube-system	calico-node-2j9mr	1/1	Running
kube-system	calico-node-czk76	1/1	Running
kube-system	calico-node-jpdxn	1/1	Running
kube-system	calico-node-nwnvn	1/1	Running
kube-system	calico-node-ssjrx	1/1	Running
kube-system	coredns-76798d84dd-5pvgf	1/1	Running
kube-system	coredns-76798d84dd-w7l2j	1/1	Running
kube-system	dns-autoscaler-85f898cd5c-qqrbp	1/1	Running
kube-system	kube-apiserver-hci-ai-k8-master-01	1/1	Running
kube-system	kube-apiserver-hci-ai-k8-master-02	1/1	Running
kube-system	kube-apiserver-hci-ai-k8-master-03	1/1	Running
kube-system	kube-controller-manager-hci-ai-k8-master-01	1/1	Running
kube-system	kube-controller-manager-hci-ai-k8-master-02	1/1	Running
kube-system	kube-controller-manager-hci-ai-k8-master-03	1/1	Running
kube-system	kube-proxy-5znxk	1/1	Running
kube-system	kube-proxy-fk6h6	1/1	Running
kube-system	kube-proxy-hphfb	1/1	Running
kube-system	kube-proxy-qzxhr	1/1	Running
kube-system	kube-proxy-rkjds	1/1	Running
kube-system	kube-scheduler-hci-ai-k8-master-01	1/1	Running
kube-system	kube-scheduler-hci-ai-k8-master-02	1/1	Running
kube-system	kube-scheduler-hci-ai-k8-master-03	1/1	Running
kube-system	kubernetes-dashboard-5fcff756f-dmswt	1/1	Running
kube-system	kubernetes-metrics-scrapers-747b4fd5cd-4q4p2	1/1	Running
kube-system	nginx-proxy-hci-ai-k8-worker-01	1/1	Running
kube-system	nginx-proxy-hci-ai-k8-worker-02	1/1	Running
kube-system	nodelocaldns-2dmjr	1/1	Running
kube-system	nodelocaldns-b7xrw	1/1	Running
kube-system	nodelocaldns-jrhs2	1/1	Running
kube-system	nodelocaldns-jztzs	1/1	Running
kube-system	nodelocaldns-wgx84	1/1	Running

## 11. Verify that the Kubernetes setup can access and use the GPUs.

```
./scripts/k8s_verify_gpu.sh
```

Expected sample output:

```
rarvind@deployment-jump:~/deepops$ ./scripts/k8s_verify_gpu.sh
job_name=cluster-gpu-tests
Node found with 3 GPUs
Node found with 3 GPUs
total_gpus=6
Creating/Deleting sandbox Namespace
updating test yaml
downloading containers ...
```

job.batch/cluster-gpu-tests condition met

executing ...

Mon Aug 17 16:02:45 2020

```
+-----+  
| NVIDIA-SMI 440.64.00     Driver Version: 440.64.00     CUDA Version: 10.2 |  
+-----+  
| GPU  Name      Persistence-M| Bus-Id      Disp.A | Volatile Uncorr. ECC |  
| Fan  Temp  Perf  Pwr:Usage/Cap|      Memory-Usage | GPU-Util  Compute M. |  
|-----+-----+-----+-----+-----+-----+  
|   0  Tesla T4           On  | 00000000:18:00.0 Off |                0 |  
| N/A   38C    P8    10W /  70W |      0MiB / 15109MiB |      0%     Default |  
+-----+-----+-----+-----+-----+-----+
```

```
+-----+  
| Processes:                               GPU Memory |  
| GPU      PID  Type  Process name          Usage  |  
|-----+-----+-----+-----+-----+-----+  
| No running processes found               |  
+-----+-----+-----+-----+-----+-----+
```

Mon Aug 17 16:02:45 2020

```
+-----+  
| NVIDIA-SMI 440.64.00     Driver Version: 440.64.00     CUDA Version: 10.2 |  
+-----+  
| GPU  Name      Persistence-M| Bus-Id      Disp.A | Volatile Uncorr. ECC |  
| Fan  Temp  Perf  Pwr:Usage/Cap|      Memory-Usage | GPU-Util  Compute M. |  
|-----+-----+-----+-----+-----+-----+  
|   0  Tesla T4           On  | 00000000:18:00.0 Off |                0 |  
| N/A   38C    P8    10W /  70W |      0MiB / 15109MiB |      0%     Default |  
+-----+-----+-----+-----+-----+-----+
```

```
+-----+  
| Processes:                               GPU Memory |  
| GPU      PID  Type  Process name          Usage  |  
|-----+-----+-----+-----+-----+-----+  
| No running processes found               |  
+-----+-----+-----+-----+-----+-----+
```

Mon Aug 17 16:02:45 2020

```
+-----+  
| NVIDIA-SMI 440.64.00     Driver Version: 440.64.00     CUDA Version: 10.2 |  
+-----+  
| GPU  Name      Persistence-M| Bus-Id      Disp.A | Volatile Uncorr. ECC |  
| Fan  Temp  Perf  Pwr:Usage/Cap|      Memory-Usage | GPU-Util  Compute M. |  
|-----+-----+-----+-----+-----+-----+  
|   0  Tesla T4           On  | 00000000:18:00.0 Off |                0 |  
| N/A   38C    P8    10W /  70W |      0MiB / 15109MiB |      0%     Default |  
+-----+-----+-----+-----+-----+-----+
```

```
+-----+  
| Processes:                               GPU Memory |  
| GPU      PID  Type  Process name          Usage  |  
|-----+-----+-----+-----+-----+-----+  
| No running processes found               |  
+-----+-----+-----+-----+-----+-----+
```

```
| No running processes found
+-----+
Mon Aug 17 16:02:45 2020
+-----+
| NVIDIA-SMI 440.64.00     Driver Version: 440.64.00    CUDA Version: 10.2 |
|-----+-----+-----+-----+-----+-----+-----+-----+
| GPU Name      Persistence-M| Bus-Id      Disp.A | Volatile Uncorr. ECC |
| Fan Temp Perf Pwr:Usage/Cap| Memory-Usage | GPU-Util Compute M. |
|-----+-----+-----+-----+-----+-----+-----+-----+
| 0 Tesla T4          On | 00000000:18:00.0 Off |                0 |
| N/A 38C   P8    10W / 70W |      0MiB / 15109MiB |      0%    Default |
+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+
| Processes:                               GPU Memory |
| GPU       PID  Type  Process name        Usage   |
|-----+-----+-----+-----+-----+-----+
| No running processes found
+-----+
Mon Aug 17 16:02:45 2020
+-----+
| NVIDIA-SMI 440.64.00     Driver Version: 440.64.00    CUDA Version: 10.2 |
|-----+-----+-----+-----+-----+-----+-----+-----+
| GPU Name      Persistence-M| Bus-Id      Disp.A | Volatile Uncorr. ECC |
| Fan Temp Perf Pwr:Usage/Cap| Memory-Usage | GPU-Util Compute M. |
|-----+-----+-----+-----+-----+-----+-----+-----+
| 0 Tesla T4          On | 00000000:18:00.0 Off |                0 |
| N/A 38C   P8    10W / 70W |      0MiB / 15109MiB |      0%    Default |
+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+
| Processes:                               GPU Memory |
| GPU       PID  Type  Process name        Usage   |
|-----+-----+-----+-----+-----+-----+
| No running processes found
+-----+
Mon Aug 17 16:02:45 2020
+-----+
| NVIDIA-SMI 440.64.00     Driver Version: 440.64.00    CUDA Version: 10.2 |
|-----+-----+-----+-----+-----+-----+-----+-----+
| GPU Name      Persistence-M| Bus-Id      Disp.A | Volatile Uncorr. ECC |
| Fan Temp Perf Pwr:Usage/Cap| Memory-Usage | GPU-Util Compute M. |
|-----+-----+-----+-----+-----+-----+-----+-----+
| 0 Tesla T4          On | 00000000:18:00.0 Off |                0 |
| N/A 38C   P8    10W / 70W |      0MiB / 15109MiB |      0%    Default |
+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+
| Processes:                               GPU Memory |
| GPU       PID  Type  Process name        Usage   |
|-----+-----+-----+-----+-----+-----|

```

```
| No running processes found  
+-----+  
Number of Nodes: 2  
Number of GPUs: 6  
6 / 6 GPU Jobs COMPLETED  
job.batch "cluster-gpu-tests" deleted  
namespace "cluster-gpu-verify" deleted
```

12. Install Helm on the Deployment Jump.

```
./scripts/install_helm.sh
```

13. Remove the taints on the master nodes.

```
kubectl taint nodes --all node-role.kubernetes.io/master-
```

This step is required to run the LoadBalancer pods.

14. Deploy LoadBalancer.

15. Edit the `config/helm/metallb.yaml` file and provide a range of IP ddresses in the `Application Network` to be used as LoadBalancer.

```
---  
# Default address range matches private network for the virtual cluster  
# defined in virtual/.  
# You should set this address range based on your site's infrastructure.  
configInline:  
  address-pools:  
    - name: default  
      protocol: layer2  
      addresses:  
        - 172.21.231.130-172.21.231.140#Application Network  
controller:  
  nodeSelector:  
    node-role.kubernetes.io/master: ""
```

16. Run a script to deploy LoadBalancer.

```
./scripts/k8s_deploy_loadbalancer.sh
```

17. Deploy an Ingress Controller.

```
./scripts/k8s_deploy_ingress.sh
```

Next: Deploy and Configure ONTAP Select in the VMware Virtual Infrastructure (Automated Deployment)

### Deploy and Configure ONTAP Select in the VMware Virtual Infrastructure (Automated Deployment)

To deploy and configure an ONTAP Select instance within the VMware Virtual Infrastructure, complete the following steps:

1. From the Deployment Jump VM, login to the [NetApp Support Site](#) and download the ONTAP Select OVA for ESXi.
2. Create a directory OTS and obtain the Ansible roles for deploying ONTAP Select.

```
mkdir OTS
cd OTS
git clone https://github.com/NetApp/ansible.git
cd ansible
```

3. Install the prerequisite libraries.

```

pip install requests
pip install pyvmomi
Open a VI Editor and create a playbook "ots_setup.yaml" with the below content to
deploy the ONTAP Select OVA and initialize the ONTAP cluster.

---
- name: Create ONTAP Select Deploy VM from OVA (ESXi)
  hosts: localhost
  gather_facts: false
  connection: 'local'
  vars_files:
    - ots_deploy_vars.yaml
  roles:
    - na_ots_deploy
- name: Wait for 1 minute before starting cluster setup
  hosts: localhost
  gather_facts: false
  tasks:
    - pause:
        minutes: 1
- name: Create ONTAP Select cluster (ESXi)
  hosts: localhost
  gather_facts: false
  vars_files:
    - ots_cluster_vars.yaml
  roles:
    - na_ots_cluster

```

4. Open a VI editor, create a variable file [ots\\_deploy\\_vars.yaml](#), and fill in hte following parameters:

```

target_vcenter_or_esxi_host: "10.xxx.xx.xx"# vCenter IP
host_login: "yourlogin@yourlab.local" # vCenter Username
ovf_path: "/run/deploy/ovapath/ONTAPdeploy.ova"# Path to OVA on Deployment Jump VM
datacenter_name: "your-Lab"# Datacenter name in vCenter
esx_cluster_name: "your Cluster"# Cluster name in vCenter
datastore_name: "your-select-dt"# Datastore name in vCenter
mgt_network: "your-mgmt-network"# Management Network to be used by OVA
deploy_name: "test-deploy-vm"# Name of the ONTAP Select VM
deploy_ipAddress: "10.xxx.xx.xx"# Management IP Address of ONTAP Select VM
deploy_gateway: "10.xxx.xx.1"# Default Gateway
deploy_proxy_url: ""# Proxy URL (Optional and if used)
deploy_netMask: "255.255.255.0"# Netmask
deploy_product_company: "NetApp"# Name of Organization
deploy_primaryDNS: "10.xxx.xx.xx"# Primary DNS IP
deploy_secondaryDNS: ""# Secondary DNS (Optional)
deploy_searchDomains: "your.search.domain.com"# Search Domain Name

```

Update the variables to match your environment.

5. Open a VI editor, create a variable file `ots_cluster_vars.yaml`, and fill it out with the following parameters:

```
node_count: 1#Number of nodes in the ONTAP Cluster
monitor_job: true
monitor_deploy_job: true
deploy_api_url: #Use the IP of the ONTAP Select VM
deploy_login: "admin"
vcenter_login: "administrator@vsphere.local"
vcenter_name: "172.21.232.100"
esxi_hosts:
  - host_name: 172.21.232.102
  - host_name: 172.21.232.103
cluster_name: "hci-ai-ots"# Name of ONTAP Cluster
cluster_ip: "172.21.232.118"# Cluster Management IP
cluster_netmask: "255.255.255.0"
cluster_gateway: "172.21.232.1"
cluster_ontap_image: "9.7"
cluster_ntp:
  - "10.61.186.231"
cluster_dns_ips:
  - "10.61.186.231"
cluster_dns_domains:
  - "sddc.netapp.com"
mgt_network: "NetApp HCI VDS 01-Management_Network"# Name of VM Port Group for Mgmt Network
data_network: "NetApp HCI VDS 01-NFS_Network"# Name of VM Port Group for NFS Network
internal_network: ""# Not needed for Single Node Cluster
instance_type: "small"
cluster_nodes:
  - node_name: "{{ cluster_name }}-01"
    ipAddress: 172.21.232.119# Node Management IP
    storage_pool: NetApp-HCI-Datastore-02 # Name of Datastore in vCenter to use
    capacityTB: 1# Usable capacity will be ~700GB
    host_name: 172.21.232.102# IP Address of an ESXi host to deploy node
```

Update the variables to match your environment.

6. Start ONTAP Select setup.

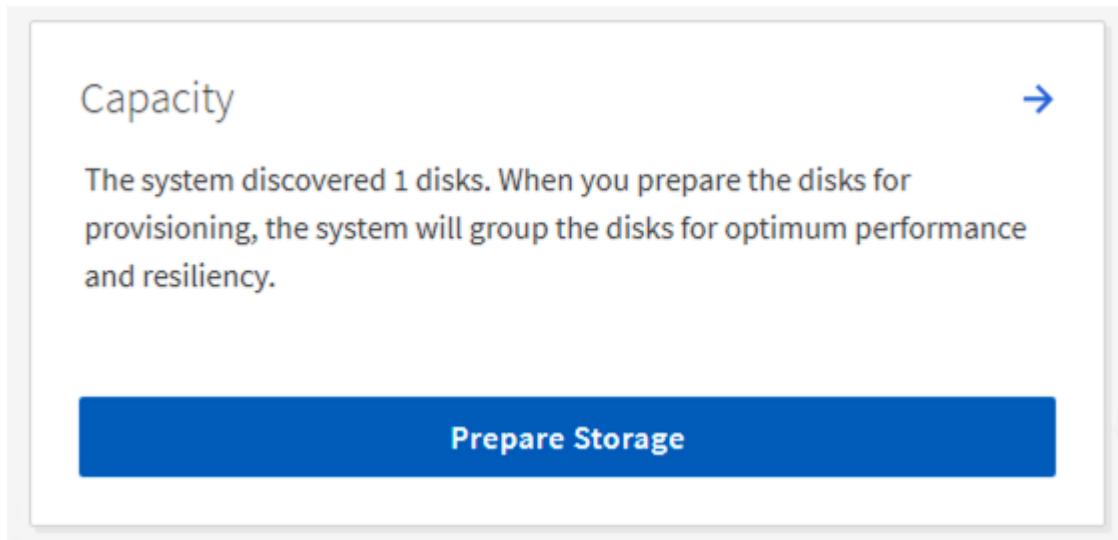
```
ansible-playbook ots_setup.yaml --extra-vars deploy_pwd=$'"P@ssw0rd"' --extra-vars
vcenter_password=$'"P@ssw0rd"' --extra-vars ontap_pwd=$'"P@ssw0rd"' --extra-vars
host_esx_password=$'"P@ssw0rd"' --extra-vars host_password=$'"P@ssw0rd"' --extra-vars
deploy_password=$'"P@ssw0rd"'
```

7. Update the command with `deploy_pwd` '(ONTAP Select VM instance), `\vcenter_password`(vCenter), `ontap_pwd` (ONTAP login password), `host_esx_password` (VMware ESXi), `host_password` (vCenter), and `deploy_password` (ONTAP Select VM instance).

#### Configure the ONTAP Select Cluster – Manual Deployment

To configure the ONTAP Select cluster, complete the following steps:

1. Open a browser and log into the ONTAP cluster's System Manager using its cluster management IP.
2. On the DASHBOARD page, click Prepare Storage under Capacity.



3. Select the radio button to continue without onboard key manager, and click Prepare Storage.
4. On the NETWORK page, click the + sign in the Broadcast Domains window.

Broadcast Domains		
Cluster	9000 MTU	IPspace: Cluster
Default	1500 MTU	IPspace: Default hci-ai-ots-01 e0b e0c
Mgmt	1500 MTU	IPspace: Default hci-ai-ots-01 e0a

5. Enter the Name as **NFS**, set the MTU to **9000**, and select the port **e0b**. Click Save.

# Add Broadcast Domain

Specify the following details to add a new broadcast domain.

NAME

NFS

MTU

9000

ASSIGN PORTS [?](#)

Port Name	hci-ai-ots-01
e0b	<input checked="" type="checkbox"/>
e0c	<input type="checkbox"/>

**Save**

[Cancel](#)

6. On the DASHBOARD page, click **Configure Protocols** under Network.

## Network

No protocols are enabled. To begin serving data to clients, enable the required protocols and assign the protocol addresses.

**Configure Protocols**

7. Enter a name for the SVM, select Enable NFS, provide an IP and subnet mask for the NFS LIF, set the Broadcast Domain to NFS, and click Save.

## Configure Protocols

X

ONTAP exposes protocol services through storage VMs. [More details](#)

STORAGE VM NAME

infra-NFS-hci-ai

### Access Protocol

SMB/CIFS and NFS

iSCSI

Enable SMB/CIFS

Enable NFS

DEFAULT LANGUAGE [?](#)

c.utf\_8



### NETWORK INTERFACE

One network interface per node is recommended.

hci-ai-ots-01

IP ADDRESS

172.21.235.119

SUBNET MASK

255.255.255.0

GATEWAY

[Add optional gateway](#)

BROADCAST DOMAIN

NFS



**Save**

[Cancel](#)

8. Click STORAGE in the left pane, and from the dropdown select Storage VMs

- a. Edit the SVM.

## Storage VMs

+ Add

Name	State
infra-NFS-hci-ai	running

⋮

[Edit](#)

[Delete](#)

[Stop](#)

- b. Select the checkbox under Resource Allocation, make sure that the local tier is listed, and click Save.

## Edit Storage VM

X

STORAGE VM NAME

infra-NFS-hci-ai

DEFAULT LANGUAGE

c.utf\_8



## Resource Allocation

Limit volume creation to preferred local tiers

LOCAL TIERS

hci\_ai\_ots\_01\_SSD\_1 X

Cancel

Save

9. Click the SVM name, and on the right panel scroll down to Policies.
10. Click the arrow within the Export Policies tile, and click the default policy.
11. If there is a rule already defined, you can edit it; if no rule exists, then create a new one.
  - a. Select NFS Network Clients as the Client Specification.
  - b. Select the Read-Only and Read/Write checkboxes.
  - c. Select the checkbox to Allow Superuser Access.

## New Rule

**CLIENT SPECIFICATION**

172.21.235.0/24

**ACCESS PROTOCOLS**

SMB/CIFS  
 FlexCache  
 NFS    NFSv3    NFSv4

**ACCESS DETAILS**

Type	<input checked="" type="checkbox"/> Read-Only	<input checked="" type="checkbox"/> Read/Write
UNIX	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kerberos 5	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kerberos 5i	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kerberos 5p	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
NTLM	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Allow Superuser Access

[Cancel](#)
[Save](#)

[Next: Deploy NetApp Trident \(Automated Deployment\)](#)

### Deploy NetApp Trident (Automated Deployment)

NetApp Trident is deployed by using an Ansible playbook that is available with NVIDIA DeepOps. Follow these steps to set up NetApp Trident:

1. From the Deployment Jump VM, navigate to the DeepOps directory and open a VI editor to `config/group_vars/netapp-trident.yml`. The file from DeepOps lists two backends and two storage classes. In this solution only one backend and storage class are used.

Use the following template to update the file and its parameters (highlighted in yellow) to match

your environment.

```
---
```

```
# vars file for netapp-trident playbook
# URL of the Trident installer package that you wish to download and use
trident_version: "20.07.0"># Version of Trident desired
trident_installer_url: "https://github.com/NetApp/trident/releases/download/v{{trident_version }}/trident-installer-{{ trident_version }}.tar.gz"
# Kubernetes version
# Note: Do not include patch version, e.g. provide value of 1.16, not 1.16.7.
# Note: Versions 1.14 and above are supported when deploying Trident with DeepOps.
# If you are using an earlier version, you must deploy Trident manually.
k8s_version: 1.17.9# Version of Kubernetes running
# Denotes whether or not to create new backends after deploying trident
# For more info, refer to: https://netapp-trident.readthedocs.io/en/stable-v20.04/kubernetes/operator-install.html#creating-a-trident-backend
create_backends: true
# List of backends to create
# For more info on parameter values, refer to: https://netapp-trident.readthedocs.io/en/stable-v20.04/kubernetes/operations/tasks/backends/ontap.html
# Note: Parameters other than those listed below are not available when creating a backend via DeepOps
# If you wish to use other parameter values, you must create your backend manually.
backends_to_create:
  - backendName: ontap-flexvol
    storageDriverName: ontap-nas # only 'ontap-nas' and 'ontap-nas-flexgroup' are supported when creating a backend via DeepOps
    managementLIF: 172.21.232.118# Cluster Management IP or SVM Mgmt LIF IP
    dataLIF: 172.21.235.119# NFS LIF IP
    svm: infra-NFS-hci-ai# Name of SVM
    username: admin# Username to connect to the ONTAP cluster
    password: P@ssw0rd# Password to login
    storagePrefix: trident
    limitAggregateUsage: ""
    limitVolumeSize: ""
    nfsMountOptions: ""
    defaults:
      spaceReserve: none
      snapshotPolicy: none
      snapshotReserve: 0
      splitOnClone: false
      encryption: false
      unixPermissions: 777
      snapshotDir: false
      exportPolicy: default
      securityStyle: unix
```

```

tieringPolicy: none
# Add additional backends as needed
# Denotes whether or not to create new StorageClasses for your NetApp storage
# For more info, refer to: https://netapp-trident.readthedocs.io/en/stable-
v20.04/kubernetes/operator-install.html#creating-a-storage-class
create_StorageClasses: true
# List of StorageClasses to create
# Note: Each item in the list should be an actual K8s StorageClass definition in yaml
format
# For more info on StorageClass definitions, refer to https://netapp-
trident.readthedocs.io/en/stable-v20.04/kubernetes/concepts/objects.html#kubernetes-
storageclass-objects.
storageClasses_to_create:
  - apiVersion: storage.k8s.io/v1
    kind: StorageClass
    metadata:
      name: ontap-flexvol
    annotations:
      storageclass.kubernetes.io/is-default-class: "true"
    provisioner: csi.trident.netapp.io
    parameters:
      backendType: "ontap-nas"
# Add additional StorageClasses as needed
# Denotes whether or not to copy tridentctl binary to localhost
copy_tridentctl_to_localhost: true
# Directory that tridentctl will be copied to on localhost
tridentctl_copy_to_directory: ... # will be copied to 'deepops/' directory

```

2. Setup NetApp Trident by using the Ansible playbook.

```
ansible-playbook -l k8s-cluster playbooks/netapp-trident.yml
```

3. Verify that Trident is running.

```
./tridentctl -n trident version
```

The expected output is as follows:

```
rarvind@deployment-jump:~/deepops$ ./tridentctl -n trident version
+-----+-----+
| SERVER VERSION | CLIENT VERSION |
+-----+-----+
| 20.07.0         | 20.07.0          |
+-----+-----+
```

[Next: Deploy NVIDIA Triton Inference Server \(Automated Deployment\)](#)

## Deploy NVIDIA Triton Inference Server (Automated Deployment)

To set up automated deployment for the Triton Inference Server, complete the following steps:

1. Open a VI editor and create a PVC yaml file `vi pvc-triton-model-repo.yaml`.

```
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: triton-pvc  namespace: triton
spec:
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 10Gi
  storageClassName: ontap-flexvol
```

2. Create the PVC.

```
kubectl create -f pvc-triton-model-repo.yaml
```

3. Open a VI editor, create a deployment for the Triton Inference Server, and call the file `triton_deployment.yaml`.

```
---
apiVersion: v1
kind: Service
metadata:
  labels:
    app: triton-3gpu
    name: triton-3gpu
    namespace: triton
  spec:
```

```
ports:
- name: grpc-trtis-serving
  port: 8001
  targetPort: 8001
- name: http-trtis-serving
  port: 8000
  targetPort: 8000
- name: prometheus-metrics
  port: 8002
  targetPort: 8002
selector:
  app: triton-3gpu
  type: LoadBalancer
---
apiVersion: v1
kind: Service
metadata:
  labels:
    app: triton-1gpu
    name: triton-1gpu
    namespace: triton
spec:
  ports:
    - name: grpc-trtis-serving
      port: 8001
      targetPort: 8001
    - name: http-trtis-serving
      port: 8000
      targetPort: 8000
    - name: prometheus-metrics
      port: 8002
      targetPort: 8002
  selector:
    app: triton-1gpu
    type: LoadBalancer
---
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app: triton-3gpu
    name: triton-3gpu
    namespace: triton
spec:
  replicas: 1
  selector:
    matchLabels:
      app: triton-3gpu      version: v1
```

```
template:
  metadata:
    labels:
      app: triton-3gpu
      version: v1
  spec:
    containers:
      - image: nvcr.io/nvidia/tritonserver:20.07-v1-py3
        command: ["/bin/sh", "-c"]
        args: ["trtserver --model-store=/mnt/model-repo"]
        imagePullPolicy: IfNotPresent
        name: triton-3gpu
        ports:
          - containerPort: 8000
          - containerPort: 8001
          - containerPort: 8002
        resources:
          limits:
            cpu: "2"
            memory: 4Gi
            nvidia.com/gpu: 3
          requests:
            cpu: "2"
            memory: 4Gi
            nvidia.com/gpu: 3
        volumeMounts:
          - name: triton-model-repo
            mountPath: /mnt/model-repo      nodeSelector:
              gpu-count: "3"
        volumes:
          - name: triton-model-repo
            persistentVolumeClaim:
              claimName: triton-pvc---
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app: triton-1gpu
    name: triton-1gpu
    namespace: triton
spec:
  replicas: 3
  selector:
    matchLabels:
      app: triton-1gpu
      version: v1
  template:
    metadata:
```

```

labels:
  app: triton-1gpu
  version: v1
spec:
  containers:
    - image: nvcr.io/nvidia/tritonserver:20.07-v1-py3
      command: ["/bin/sh", "-c", "sleep 1000"]
      args: ["trtserver --model-store=/mnt/model-repo"]
      imagePullPolicy: IfNotPresent
      name: triton-1gpu
      ports:
        - containerPort: 8000
        - containerPort: 8001
        - containerPort: 8002
      resources:
        limits:
          cpu: "2"
          memory: 4Gi
          nvidia.com/gpu: 1
        requests:
          cpu: "2"
          memory: 4Gi
          nvidia.com/gpu: 1
      volumeMounts:
        - name: triton-model-repo
          mountPath: /mnt/model-repo      nodeSelector:
            gpu-count: "1"
      volumes:
        - name: triton-model-repo
          persistentVolumeClaim:
            claimName: triton-pvc

```

Two deployments are created here as an example. The first deployment spins up a pod that uses three GPUs and has replicas set to 1. The other deployment spins up three pods each using one GPU while the replica is set to 3. Depending on your requirements, you can change the GPU allocation and replica counts.

Both of the deployments use the PVC created earlier and this persistent storage is provided to the Triton inference servers as the model repository.

For each deployment, a service of type LoadBalancer is created. The Triton Inference Server can be accessed by using the LoadBalancer IP which is in the application network.

A nodeSelector is used to ensure that both deployments get the required number of GPUs without any issues.

4. Label the K8 worker nodes.

```
kubectl label nodes hci-ai-k8-worker-01 gpu-count=3  
kubectl label nodes hci-ai-k8-worker-02 gpu-count=1
```

5. Create the deployment.

```
kubectl apply -f triton_deployment.yaml
```

6. Make a note of the LoadBalancer service external LPS.

```
kubectl get services -n triton
```

The expected sample output is as follows:

```
rarvind@deployment-jump:~/triton-inference-server$ kubectl get services -n triton  
NAME           TYPE      CLUSTER-IP   EXTERNAL-IP     PORT(S)          AGE  
triton-1gpu-v20-07-v1   LoadBalancer   10.233.21.185  172.21.231.133  8001:31238/TCP,8000:30171/TCP,8002:32348/TCP  10h  
triton-3gpu-v20-07-v1   LoadBalancer   10.233.13.17   172.21.231.132  8001:31549/TCP,8000:30220/TCP,8002:31517/TCP  10h
```

7. Connect to any one of the pods that were created from the deployment.

```
kubectl exec -n triton --stdin --tty triton-1gpu-86c4c8dd64-545lx -- /bin/bash
```

8. Set up the model repository by using the example model repository.

```
git clone  
cd triton-inference-server  
git checkout r20.07
```

9. Fetch any missing model definition files.

```
cd docs/examples  
.fetch_models.sh
```

10. Copy all the models to the model repository location or just a specific model that you wish to use.

```
cp -r model_repository/resnet50_netdef/ /mnt/model-repo/
```

In this solution, only the resnet50\_netdef model is copied over to the model repository as an example.

11. Check the status of the Triton Inference Server.

```
curl -v <<LoadBalancer_IP_recorded_earlier>>:8000/api/status
```

The expected sample output is as follows:

```
curl -v 172.21.231.132:8000/api/status
* Trying 172.21.231.132...
* TCP_NODELAY set
* Connected to 172.21.231.132 (172.21.231.132) port 8000 (#0)
> GET /api/status HTTP/1.1
> Host: 172.21.231.132:8000
> User-Agent: curl/7.58.0
> Accept: */*
>
< HTTP/1.1 200 OK
< NV-Status: code: SUCCESS server_id: "inference:0" request_id: 9
< Content-Length: 1124
< Content-Type: text/plain
<
id: "inference:0"
version: "1.15.0"
uptime_ns: 377890294368
model_status {
    key: "resnet50_netdef"
    value {
        config {
            name: "resnet50_netdef"
            platform: "caffe2_netdef"
            version_policy {
                latest {
                    num_versions: 1
                }
            }
        }
        max_batch_size: 128
        input {
            name: "gpu_0/data"
            data_type: TYPE_FP32
            format: FORMAT_NCHW
            dims: 3
            dims: 224
            dims: 224
        }
        output {
            name: "gpu_0/softmax"
            data_type: TYPE_FP32
            dims: 1000
            label_filename: "resnet50_labels.txt"
        }
    }
}
```

```
    }
    instance_group {
        name: "resnet50_netdef"
        count: 1
        gpus: 0
        gpus: 1
        gpus: 2
        kind: KIND_GPU
    }
    default_model_filename: "model.netdef"
    optimization {
        input_pinned_memory {
            enable: true
        }
        output_pinned_memory {
            enable: true
        }
    }
}
version_status {
    key: 1
    value {
        ready_state: MODEL_READY
        ready_state_reason {
        }
    }
}
}
ready_state: SERVER_READY
* Connection #0 to host 172.21.231.132 left intact
```

Next: [Deploy the Client for Triton Inference Server \(Automated Deployment\)](#)

## Deploy the Client for Triton Inference Server (Automated Deployment)

To deploy the client for the Triton Inference Server, complete the following steps:

1. Open a VI editor, create a deployment for the Triton client, and call the file `triton_client.yaml`.

```
---
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app: triton-client
    name: triton-client
    namespace: triton
spec:
  replicas: 1
  selector:
    matchLabels:
      app: triton-client
      version: v1
  template:
    metadata:
      labels:
        app: triton-client
        version: v1
    spec:
      containers:
        - image: nvcr.io/nvidia/tritonserver:20.07- v1- py3-clientsdk
          imagePullPolicy: IfNotPresent
          name: triton-client
          resources:
            limits:
              cpu: "2"
              memory: 4Gi
            requests:
              cpu: "2"
              memory: 4Gi
```

## 2. Deploy the client.

```
kubectl apply -f triton_client.yaml
```

Next: [Collect Inference Metrics from Triton Inference Server](#)

### Collect Inference Metrics from Triton Inference Server

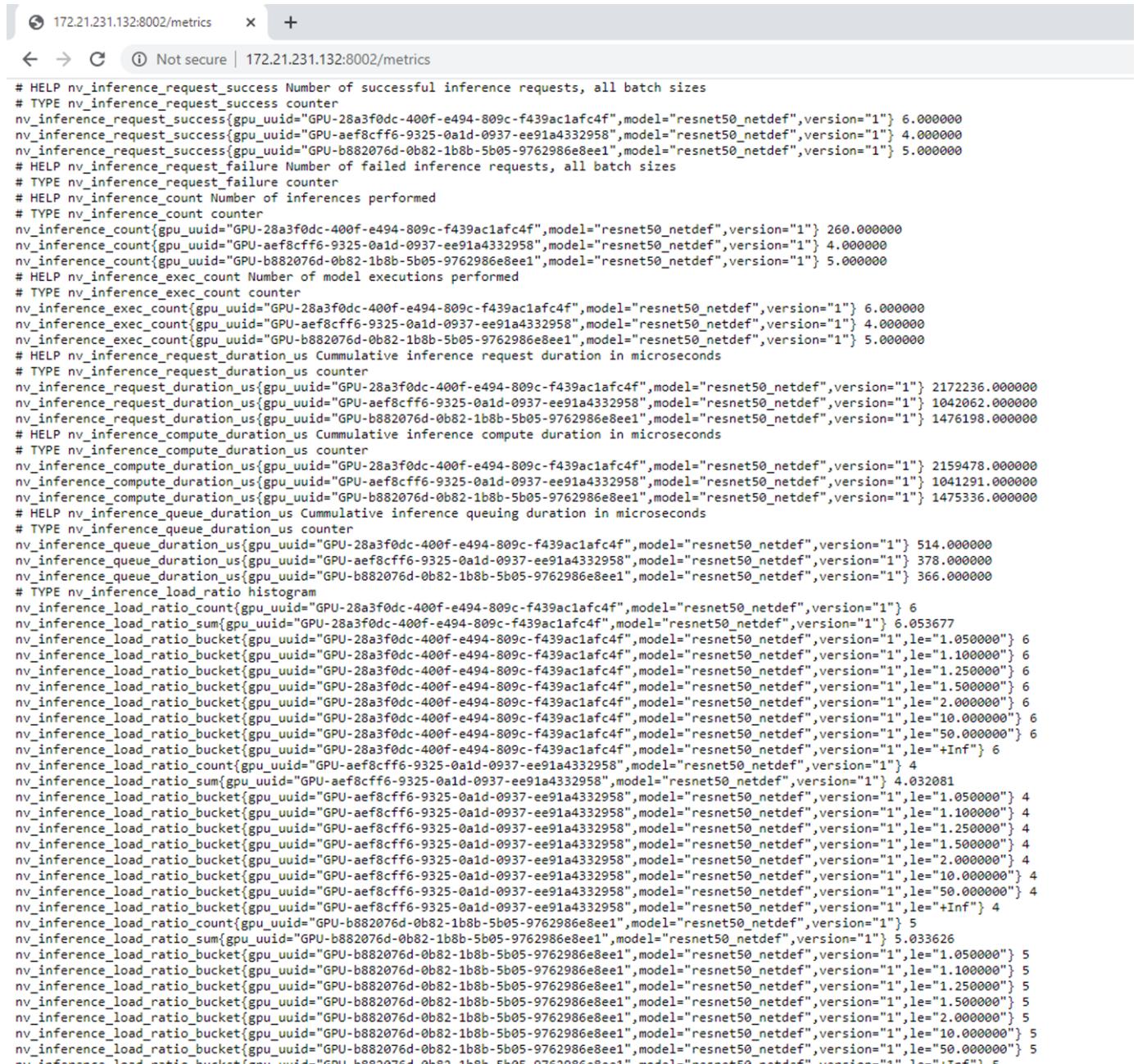
The Triton Inference Server provides Prometheus metrics indicating GPU and request statistics.

By default, these metrics are available at [http://<a href="#">localhost:8000/metrics](http://<a href=)

anchor="triton\_inference\_server\_IP">[triton\_inference\_server\_IP&#93;:>8002/metrics" class="bare">>http://<a anchor="triton\_inference\_server\_IP">[triton\_inference\_server\_IP&#93;:>8002/metrics</a>.

The Triton Inference Server IP is the LoadBalancer IP that was recorded earlier.

The metrics are only available by accessing the endpoint and are not pushed or published to any remote server.



```
# HELP nv_inference_request_success Number of successful inference requests, all batch sizes
# TYPE nv_inference_request_success counter
nv_inference_request_success{gpu_uuid="GPU-28a3f0dc-400f-e494-809c-f439ac1afc4f",model="resnet50_netdef",version="1"} 6.000000
nv_inference_request_success{gpu_uuid="GPU-aef8cff6-9325-0a1d-0937-ee91a4332958",model="resnet50_netdef",version="1"} 4.000000
nv_inference_request_success{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1",model="resnet50_netdef",version="1"} 5.000000
# HELP nv_inference_request_failure Number of failed inference requests, all batch sizes
# TYPE nv_inference_request_failure counter
# HELP nv_inference_count Number of inferences performed
# TYPE nv_inference_count counter
nv_inference_count{gpu_uuid="GPU-28a3f0dc-400f-e494-809c-f439ac1afc4f",model="resnet50_netdef",version="1"} 260.000000
nv_inference_count{gpu_uuid="GPU-aef8cff6-9325-0a1d-0937-ee91a4332958",model="resnet50_netdef",version="1"} 4.000000
nv_inference_count{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1",model="resnet50_netdef",version="1"} 5.000000
# HELP nv_inference_exec_count Number of model executions performed
# TYPE nv_inference_exec_count counter
nv_inference_exec_count{gpu_uuid="GPU-28a3f0dc-400f-e494-809c-f439ac1afc4f",model="resnet50_netdef",version="1"} 6.000000
nv_inference_exec_count{gpu_uuid="GPU-aef8cff6-9325-0a1d-0937-ee91a4332958",model="resnet50_netdef",version="1"} 4.000000
nv_inference_exec_count{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1",model="resnet50_netdef",version="1"} 5.000000
# HELP nv_inference_request_duration_us Cumulative inference request duration in microseconds
# TYPE nv_inference_request_duration_us counter
nv_inference_request_duration_us{gpu_uuid="GPU-28a3f0dc-400f-e494-809c-f439ac1afc4f",model="resnet50_netdef",version="1"} 2172236.000000
nv_inference_request_duration_us{gpu_uuid="GPU-aef8cff6-9325-0a1d-0937-ee91a4332958",model="resnet50_netdef",version="1"} 1042062.000000
nv_inference_request_duration_us{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1",model="resnet50_netdef",version="1"} 1476198.000000
# HELP nv_inference_compute_duration_us Cumulative inference compute duration in microseconds
# TYPE nv_inference_compute_duration_us counter
nv_inference_compute_duration_us{gpu_uuid="GPU-28a3f0dc-400f-e494-809c-f439ac1afc4f",model="resnet50_netdef",version="1"} 2159478.000000
nv_inference_compute_duration_us{gpu_uuid="GPU-aef8cff6-9325-0a1d-0937-ee91a4332958",model="resnet50_netdef",version="1"} 1041291.000000
nv_inference_compute_duration_us{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1",model="resnet50_netdef",version="1"} 1475336.000000
# HELP nv_inference_queue_duration_us Cumulative inference queuing duration in microseconds
# TYPE nv_inference_queue_duration_us counter
nv_inference_queue_duration_us{gpu_uuid="GPU-28a3f0dc-400f-e494-809c-f439ac1afc4f",model="resnet50_netdef",version="1"} 514.000000
nv_inference_queue_duration_us{gpu_uuid="GPU-aef8cff6-9325-0a1d-0937-ee91a4332958",model="resnet50_netdef",version="1"} 378.000000
nv_inference_queue_duration_us{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1",model="resnet50_netdef",version="1"} 366.000000
# TYPE nv_inference_load_ratio histogram
nv_inference_load_ratio_count{gpu_uuid="GPU-28a3f0dc-400f-e494-809c-f439ac1afc4f",model="resnet50_netdef",version="1"} 6
nv_inference_load_ratio_sum{gpu_uuid="GPU-28a3f0dc-400f-e494-809c-f439ac1afc4f",model="resnet50_netdef",version="1"} 6.053677
nv_inference_load_ratio_bucket{gpu_uuid="GPU-28a3f0dc-400f-e494-809c-f439ac1afc4f",model="resnet50_netdef",version="1",le="1.050000"} 6
nv_inference_load_ratio_bucket{gpu_uuid="GPU-28a3f0dc-400f-e494-809c-f439ac1afc4f",model="resnet50_netdef",version="1",le="1.100000"} 6
nv_inference_load_ratio_bucket{gpu_uuid="GPU-28a3f0dc-400f-e494-809c-f439ac1afc4f",model="resnet50_netdef",version="1",le="1.250000"} 6
nv_inference_load_ratio_bucket{gpu_uuid="GPU-28a3f0dc-400f-e494-809c-f439ac1afc4f",model="resnet50_netdef",version="1",le="1.500000"} 6
nv_inference_load_ratio_bucket{gpu_uuid="GPU-28a3f0dc-400f-e494-809c-f439ac1afc4f",model="resnet50_netdef",version="1",le="2.000000"} 6
nv_inference_load_ratio_bucket{gpu_uuid="GPU-28a3f0dc-400f-e494-809c-f439ac1afc4f",model="resnet50_netdef",version="1",le="10.000000"} 6
nv_inference_load_ratio_bucket{gpu_uuid="GPU-28a3f0dc-400f-e494-809c-f439ac1afc4f",model="resnet50_netdef",version="1",le="50.000000"} 6
nv_inference_load_ratio_bucket{gpu_uuid="GPU-28a3f0dc-400f-e494-809c-f439ac1afc4f",model="resnet50_netdef",version="1",le="+Inf"} 6
nv_inference_load_ratio_count{gpu_uuid="GPU-aef8cff6-9325-0a1d-0937-ee91a4332958",model="resnet50_netdef",version="1"} 4
nv_inference_load_ratio_sum{gpu_uuid="GPU-aef8cff6-9325-0a1d-0937-ee91a4332958",model="resnet50_netdef",version="1"} 4.032081
nv_inference_load_ratio_bucket{gpu_uuid="GPU-aef8cff6-9325-0a1d-0937-ee91a4332958",model="resnet50_netdef",version="1",le="1.050000"} 4
nv_inference_load_ratio_bucket{gpu_uuid="GPU-aef8cff6-9325-0a1d-0937-ee91a4332958",model="resnet50_netdef",version="1",le="1.100000"} 4
nv_inference_load_ratio_bucket{gpu_uuid="GPU-aef8cff6-9325-0a1d-0937-ee91a4332958",model="resnet50_netdef",version="1",le="1.250000"} 4
nv_inference_load_ratio_bucket{gpu_uuid="GPU-aef8cff6-9325-0a1d-0937-ee91a4332958",model="resnet50_netdef",version="1",le="1.500000"} 4
nv_inference_load_ratio_bucket{gpu_uuid="GPU-aef8cff6-9325-0a1d-0937-ee91a4332958",model="resnet50_netdef",version="1",le="2.000000"} 4
nv_inference_load_ratio_bucket{gpu_uuid="GPU-aef8cff6-9325-0a1d-0937-ee91a4332958",model="resnet50_netdef",version="1",le="10.000000"} 4
nv_inference_load_ratio_bucket{gpu_uuid="GPU-aef8cff6-9325-0a1d-0937-ee91a4332958",model="resnet50_netdef",version="1",le="50.000000"} 4
nv_inference_load_ratio_bucket{gpu_uuid="GPU-aef8cff6-9325-0a1d-0937-ee91a4332958",model="resnet50_netdef",version="1",le="+Inf"} 4
nv_inference_load_ratio_count{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1",model="resnet50_netdef",version="1"} 5
nv_inference_load_ratio_sum{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1",model="resnet50_netdef",version="1"} 5.033626
nv_inference_load_ratio_bucket{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1",model="resnet50_netdef",version="1",le="1.050000"} 5
nv_inference_load_ratio_bucket{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1",model="resnet50_netdef",version="1",le="1.100000"} 5
nv_inference_load_ratio_bucket{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1",model="resnet50_netdef",version="1",le="1.250000"} 5
nv_inference_load_ratio_bucket{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1",model="resnet50_netdef",version="1",le="1.500000"} 5
nv_inference_load_ratio_bucket{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1",model="resnet50_netdef",version="1",le="2.000000"} 5
nv_inference_load_ratio_bucket{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1",model="resnet50_netdef",version="1",le="10.000000"} 5
nv_inference_load_ratio_bucket{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1",model="resnet50_netdef",version="1",le="50.000000"} 5
nv_inference_load_ratio_bucket{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1",model="resnet50_netdef",version="1",le="+Inf"} 5
```

```

nv_inference_load_ratio_bucket{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1",model="resnet50_netdef",version="1",le="+Inf"} 5
# HELP nv_gpu_utilization GPU utilization rate [0.0 - 1.0]
# TYPE nv_gpu_utilization gauge
nv_gpu_utilization{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1"} 0.000000
nv_gpu_utilization{gpu_uuid="GPU-28a3f0dc-400f-e494-809c-f439ac1afc4f"} 0.000000
nv_gpu_utilization{gpu_uuid="GPU-aef8cff6-9325-0a1d-0937-ee91a4332958"} 0.000000
# HELP nv_gpu_memory_total_bytes GPU total memory, in bytes
# TYPE nv_gpu_memory_total_bytes gauge
nv_gpu_memory_total_bytes{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1"} 15843721216.000000
nv_gpu_memory_total_bytes{gpu_uuid="GPU-28a3f0dc-400f-e494-809c-f439ac1afc4f"} 15843721216.000000
nv_gpu_memory_total_bytes{gpu_uuid="GPU-aef8cff6-9325-0a1d-0937-ee91a4332958"} 15843721216.000000
# HELP nv_gpu_memory_used_bytes GPU used memory, in bytes
# TYPE nv_gpu_memory_used_bytes gauge
nv_gpu_memory_used_bytes{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1"} 1466236928.000000
nv_gpu_memory_used_bytes{gpu_uuid="GPU-28a3f0dc-400f-e494-809c-f439ac1afc4f"} 13004767232.000000
nv_gpu_memory_used_bytes{gpu_uuid="GPU-aef8cff6-9325-0a1d-0937-ee91a4332958"} 1466236928.000000
# HELP nv_gpu_power_usage GPU power usage in watts
# TYPE nv_gpu_power_usage gauge
nv_gpu_power_usage{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1"} 27.999000
nv_gpu_power_usage{gpu_uuid="GPU-28a3f0dc-400f-e494-809c-f439ac1afc4f"} 28.428000
nv_gpu_power_usage{gpu_uuid="GPU-aef8cff6-9325-0a1d-0937-ee91a4332958"} 27.632000
# HELP nv_gpu_power_limit GPU power management limit in watts
# TYPE nv_gpu_power_limit gauge
nv_gpu_power_limit{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1"} 70.000000
nv_gpu_power_limit{gpu_uuid="GPU-28a3f0dc-400f-e494-809c-f439ac1afc4f"} 70.000000
nv_gpu_power_limit{gpu_uuid="GPU-aef8cff6-9325-0a1d-0937-ee91a4332958"} 70.000000
# HELP nv_energy_consumption GPU energy consumption in joules since the Triton Server started
# TYPE nv_energy_consumption counter
nv_energy_consumption{gpu_uuid="GPU-b882076d-0b82-1b8b-5b05-9762986e8ee1"} 9796.449000
nv_energy_consumption{gpu_uuid="GPU-28a3f0dc-400f-e494-809c-f439ac1afc4f"} 9997.538000
nv_energy_consumption{gpu_uuid="GPU-aef8cff6-9325-0a1d-0937-ee91a4332958"} 9669.536000

```

---

## Next: Validation Results

## Validation Results

To run a sample inference request, complete the following steps:

1. Get a shell to the client container/pod.

```
kubectl exec --stdin --tty <<client_pod_name>> -- /bin/bash
```

2. Run a sample inference request.

```
image_client -m resnet50_netdef -s INCEPTION -u <<LoadBalancer_IP_recorded
earlier>>:8000 -c 3 images/mug.jpg
```

```
root@triton-client-v20-07-v1-5566895bc-zqz6w:/workspace# image_client -m resnet50_netdef -s INCEPTION -u 172.21.231.133:8000 -c 3 images/mug.jpg
Request 0, batch size 1
Image 'images/mug.jpg':
  504 (COFFEE MUG) = 0.723991
  968 (CUP) = 0.270953
  967 (ESPRESSO) = 0.00115996
```

This inferencing request calls the `resnet50_netdef` model that is used for image recognition. Other clients can also send inferencing requests concurrently by following a similar approach and calling out the appropriate model.

## Next: Where to Find Additional Information

=  
:hardbreaks:  
:nofooter:

```
:icons: font
:linkattrs:
:imagesdir: ./media/
```

## Additional Information

To learn more about the information that is described in this document, review the following documents and/or websites:

- NetApp HCI Theory of Operations

<https://www.netapp.com/us/media/wp-7261.pdf>

- NetApp Product Documentation

[docs.netapp.com](http://docs.netapp.com)

- NetApp HCI Solution Catalog Documentation

<https://docs.netapp.com/us-en/hci/solutions/index.html>

- HCI Resources page

<https://mysupport.netapp.com/info/web/ECMLP2831412.html>

- ONTAP Select

<https://www.netapp.com/us/products/data-management-software/ontap-select-sds.aspx>

- NetApp Trident

<https://netapp-trident.readthedocs.io/en/stable-v20.01/>

- NVIDIA DeepOps

<https://github.com/NVIDIA/deepops>

- NVIDIA Triton Inference Server

<https://docs.nvidia.com/deeplearning/sdk/triton-inference-server-master-branch-guide/docs/index.html>

# **Security**

## **Copyright Information**

Copyright © 2020 NetApp, Inc. All rights reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means-graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system-without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.277-7103 (October 1988) and FAR 52-227-19 (June 1987).

## **Trademark Information**

NETAPP, the NETAPP logo, and the marks listed at <http://www.netapp.com/TM> are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.