



# Provision NetApp storage for OpenNebula

## NetApp virtualization solutions

NetApp  
February 13, 2026

# Table of Contents

- Provision NetApp storage for OpenNebula ..... 1
  - Learn about ONTAP storage architecture for OpenNebula ..... 1
    - Solution architecture ..... 1
    - ONTAP features for OpenNebula ..... 2
  - Supported storage types for OpenNebula ..... 3
    - NAS protocol support ..... 3
    - SAN protocol support ..... 4
    - NetApp ONTAP API Driver ..... 4
    - Storage type compatibility matrix ..... 4
    - OpenNebula cluster storage types supported with ONTAP ..... 4
- Configure storage protocols with NetApp for OpenNebula ..... 5
  - Learn about storage protocols for OpenNebula with NetApp ONTAP ..... 5
  - Configure SMB/CIFS Datastore storage for OpenNebula ..... 6
  - Configure NFS storage for OpenNebula using ONTAP ..... 14
  - Configure NetApp Datastore with iSCSI for OpenNebula ..... 21
  - Configure LVM Thin with ONTAP FC for OpenNebula ..... 25
  - Configure LVM Thin with ONTAP iSCSI for OpenNebula ..... 30
  - Configure LVM Thin with ONTAP NVMe/FC for OpenNebula ..... 35
  - Configure LVM Thin with ONTAP NVMe/TCP for OpenNebula ..... 39

# Provision NetApp storage for OpenNebula

## Learn about ONTAP storage architecture for OpenNebula

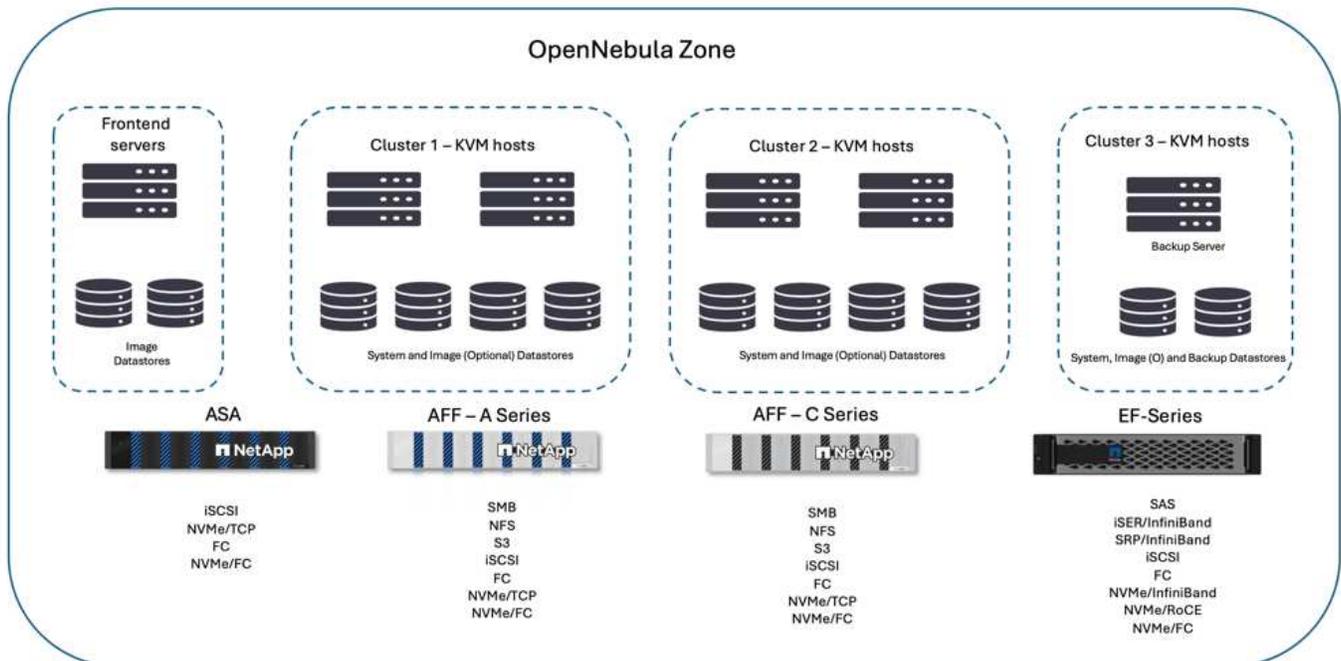
NetApp products integrates with OpenNebula to provide enterprise-grade storage capabilities through NAS and SAN protocols. ONTAP delivers advanced data management features including snapshots, cloning, replication, and ransomware protection for virtualized workloads running on OpenNebula clusters.

### Solution architecture

The solution architecture includes the following key components:

- **OpenNebula Frontend servers:** A set of OpenNebula frontend servers that provide cloud management features to manage virtual machines (VMs) and Kubernetes Clusters.
- **OpenNebula Compute nodes:** A set of compute nodes that host the virtual machines and Kubernetes clusters managed by OpenNebula.
- **NetApp ONTAP storage:** A high-performance, scalable storage system that provides shared storage for the OpenNebula cluster.
- **Network infrastructure:** A robust network setup that ensures low-latency and high-throughput connectivity between OpenNebula nodes and ONTAP storage.
- **NetApp Console:** A centralized management interface for managing multiple NetApp storage systems and data services.

The following diagram shows the high-level architecture of OpenNebula integrated with NetApp ONTAP storage:



## ONTAP features for OpenNebula

ONTAP provides a comprehensive set of enterprise storage features that enhance OpenNebula deployments. These features span data management, protection, efficiency, and protocol support across both NAS and SAN storage architectures.

### Core data management features

- Scale-out cluster architecture
- Secure authentication and RBAC support
- Zero trust multi-admin support
- Secure multitenancy
- Data replication with SnapMirror
- Point-in-time copies with snapshots
- Space-efficient clones
- Storage efficiency features including deduplication and compression
- Trident CSI support for Kubernetes
- SnapLock for compliance
- Tamperproof snapshot copy locking
- Ransomware protection with autonomous threat detection
- Data-at-rest and data-in-flight encryption
- FabricPool to tier cold data to object storage
- NetApp Console and Data Infrastructure Insights integration
- Microsoft Offloaded Data Transfer (ODX)

### NAS protocol features

- FlexGroup volumes provide scale-out NAS containers with high performance, load distribution, and scalability
- FlexCache distributes data globally while providing local read and write access
- Multiprotocol support enables the same data to be accessible via both SMB and NFS
- NFS nConnect allows multiple TCP sessions per connection to increase network throughput and utilize high-speed NICs
- NFS session trunking provides increased data transfer speeds, high availability, and fault tolerance
- SMB multichannel provides increased data transfer speeds, high availability, and fault tolerance
- Integration with Active Directory and LDAP for file permissions
- Secure connections with NFS over TLS
- NFS Kerberos authentication support
- NFS over RDMA for low-latency access
- Name mapping between Windows and Unix identities
- Autonomous ransomware protection with built-in threat detection
- File System Analytics for capacity and usage insights

## SAN protocol features

- Stretch clusters across fault domains with SnapMirror active sync (always check the [Interoperability Matrix Tool](#) for supported configurations)
- ASA models provide active-active multipathing and fast path failover
- Protocol support for FC, iSCSI, and NVMe-oF
- iSCSI CHAP mutual authentication
- Selective LUN mapping and portsets for enhanced security

## Supported storage types for OpenNebula

OpenNebula supports multiple storage protocols with NetApp ONTAP, including NFS and SMB for NAS and FC, iSCSI, and NVMe-oF for SAN. Based on existing skillset and requirements, users can choose the appropriate storage protocol. If not planning to use any data services that ONTAP provides, consider SANtricity systems which can provide block storage using FC, iSCSI, Infiniband, and NVMe-oF protocols.

OpenNebula uses Datasets which are typically mounted at `/var/lib/one/datasets` folder or as defined by `DATASET_LOCATION` attribute in `/etc/one/oned.conf` file. The storage can be mounted using the `/etc/fstab` file, dynamically mounted using Automounter or other procedures that your environment supports. Most of the folder permissions are set to owned by `oneadmin` user and group. Ensure that the hypervisor hosts have access to the storage system using the required protocol.

When using automounter, use direct mounts to avoid issues with automounter taking control of parent folder. To create direct mounts, create a file under `/etc/auto.master.d/`. For example, create a file named `one.autofs` with the following command:

```
echo "/- /etc/auto.one --timeout=60 --ghost" >
/etc/auto.master.d/one.autofs
```

Frontend servers need to have access to image datasets. Mounting the image datasets on hypervisor hosts is optional but recommended for better performance. System datasets need to be mounted on hypervisor hosts as they host the virtual machine disks. Kernel and File datasets are used for VM kernels, ramdisks and other files that are required to the VM through contextualization process. It can be mounted on both frontend servers and hypervisor hosts. Backup datasets are used for VM backups and can be mounted on the all hosts on the OpenNebula cluster. VMs on other clusters can also use the same backup dataset if KVM hosts can access backup dataset hosts with SSH for `rsync`, SFTP for `restic`. If using Veeam, `oVirtAPI` should be available.

## NAS protocol support

NAS protocols (NFS and SMB) support shared filesystem across the frontend and hypervisor hosts. ONTAP snapshots can be made visible to clients for accessing point-in-time copies of data. ONTAP FlexCache can be used for Image Datasets within zones that are geographically distributed. ONTAP NFS supports `nConnect` for better performance by using multiple connections per session. While using `FlexGroup` for large datasets (> 100TB), `pNFS` is recommended to distribute the load across multiple nodes. Remember to configure at least one data lif per controller in the ONTAP cluster and hypervisor hosts need to have connectivity.

All Dataset types (Image, System, Kernel and File, and Backup) support NAS protocols.

## SAN protocol support

Enterprise edition includes NetApp driver which works with iSCSI protocol on AFF and FAS systems. Other SAN protocols (FC, InfiniBand and NVMe-oF) are typically configured for LVM driver. LVM thin pool is created per virtual machine for thin provisioning and snapshot support. Hypervisor hosts need to have connectivity to the storage system. Cluster LVM support is not required to use LVM storage type.

To use with NetApp driver, iSCSI sessions and multipath need to be configured on hypervisor hosts. For LVM driver, except for system datastore, the logical volume needs to have filesystem created and mounted. For system datastore, the volume group needs to be named as "vg-one-`<datastore_id>`" where `<datastore_id>` is the numeric identifier of the datastore in OpenNebula. All Datastore types (Image, System, Kernel and File, and Backup) support SAN protocols.

## NetApp ONTAP API Driver

OpenNebula's native NetApp integration uses ONTAP's API to automatically create and manage volumes, LUNs, snapshots, and mappings. This method offers the best level of automation and avoids manual iSCSI and LVM setup. Refer [OpenNebula documentation](#) for more information.

## Storage type compatibility matrix

Datastore Type	NFS	SMB/CIFS	FC	iSCSI	NVMe-oF
Image	Yes	Yes	Yes	Yes <sup>1</sup>	Yes
System	Yes	Yes	Yes	Yes <sup>1</sup>	Yes
Kernel and File	Yes	Yes	Yes	Yes	Yes
Backup	Yes	Yes	Yes	Yes	Yes

### Notes:

1. NetApp driver is available in Enterprise edition for iSCSI protocol to utilize native ONTAP features.

## OpenNebula cluster storage types supported with ONTAP

Here is a comparison of features supported by different storage types in OpenNebula when using NetApp ONTAP as the backend.

Feature	NetApp ONTAP API	LVM-thin	NFS/SMB
VM disks	Yes	Yes	Yes
Image storage <sup>1</sup>	Yes	Yes	Yes
Live snapshots	Yes	Yes	Yes
Clone VM or image	Yes	Yes	Yes
Incremental backup <sup>2</sup>	Yes	Yes	Yes

### Notes:

1. Image storage refers to using the backend for OpenNebula image datastores. LVM-thin and ONTAP API

methods involve copying or creating block devices from the image source.

2. Incremental backups work with `qcow2` disks (on NFS/SMB) or with block devices that support tracking changes. The OpenNebula NetApp ONTAP driver uses rolling snapshots to make incremental backups.



incremental backups require the `nbd` kernel module to be loaded.

## Configure storage protocols with NetApp for OpenNebula

### Learn about storage protocols for OpenNebula with NetApp ONTAP

Provision ONTAP storage for OpenNebula using NAS protocols (NFS, SMB) and SAN protocols (FC, iSCSI, NVMe). Select the appropriate protocol-specific procedure to configure shared storage for your OpenNebula environment.

Ensure OpenNebula frontend and hypervisor hosts have FC, Ethernet, or other supported interfaces cabled to switches with communication to ONTAP logical interfaces. Always check the [Interoperability Matrix Tool](#) for supported configurations. Example scenarios are created with assumption that two high speed network interface cards are available on each OpenNebula host which are connected together to create bonded interfaces for fault tolerance and performance. Same uplink connections are used for all network traffic including host management, VM/container traffic, and storage access. When more network interfaces are available, consider separating storage traffic from other types of traffic.

For information about ONTAP storage architecture and supported storage types, see [Learn about NetApp storage architecture for OpenNebula](#) and [Learn about supported storage types for OpenNebula](#).



When using LVM with SAN protocols (FC, iSCSI, NVMe-oF), the volume group can contain multiple LUNs or NVMe namespaces. In that case, all the LUNs or namespaces must be part of same consistency group to ensure data integrity. We don't support volume group that spans multiple ONTAP SVMs. Each volume group must be created from LUNs or namespaces from the same SVM.

### Choose a storage protocol

Select the protocol that matches your environment and requirements:

- [Configure NetApp driver with iSCSI](#) - Configure OpenNebula NetApp driver with iSCSI for block storage access over standard Ethernet networks with multipath support. This is Enterprise Edition only feature. It utilizes ONTAP native clones for efficient VM provisioning.
- [Configure SMB/CIFS storage](#) - Configure SMB/CIFS file shares for OpenNebula with multichannel support for fault tolerance and enhanced performance over multiple network connections.
- [Configure NFS storage](#) - Configure NFS storage for OpenNebula with nConnect or session trunking for fault tolerance and performance enhancements using multiple network connections.
- [Configure LVM Thin with FC](#) - Configure Logical Volume Manager (LVM) with Fibre Channel for high-performance, low-latency block storage access across OpenNebula hosts.
- [Configure LVM Thin with iSCSI](#) - Configure Logical Volume Manager (LVM) with iSCSI for block storage access over standard Ethernet networks with multipath support.
- [Configure LVM Thin with NVMe/FC](#) - Configure Logical Volume Manager (LVM) with NVMe over Fibre Channel for high-performance block storage using the modern NVMe protocol.

- [Configure LVM Thin with NVMe/TCP](#) - Configure Logical Volume Manager (LVM) with NVMe over TCP for high-performance block storage over standard Ethernet networks using the modern NVMe protocol.



If need assistance with E-Series or EF-Series storage protocols, check the link [NetApp E-Series and EF-Series documentation for setting up LVM on Linux environments](#) along with one of the LVM Thin documentation for reference.

## Configure SMB/CIFS Datastore storage for OpenNebula

Configure SMB/CIFS Datastore storage for OpenNebula using NetApp ONTAP. SMB multichannel provides fault tolerance and boosts performance with multiple network connections to the storage system.

SMB/CIFS file shares require configuration tasks by both storage and virtualization administrators. For more details, refer to [TR4740 - SMB 3.0 Multichannel](#).



Passwords are saved in clear text files and are accessible only to the root user. Ensure proper security measures are in place to protect sensitive information.

### Storage administrator tasks

If you are new to ONTAP, use System Manager Interface to complete these tasks.

1. Enable the SVM for SMB. Follow [ONTAP 9 documentation](#) for more information.
2. Create at least two LIFs per controller. Follow the steps from the documentation. For reference, here is a screenshot of LIFs used in this solution.

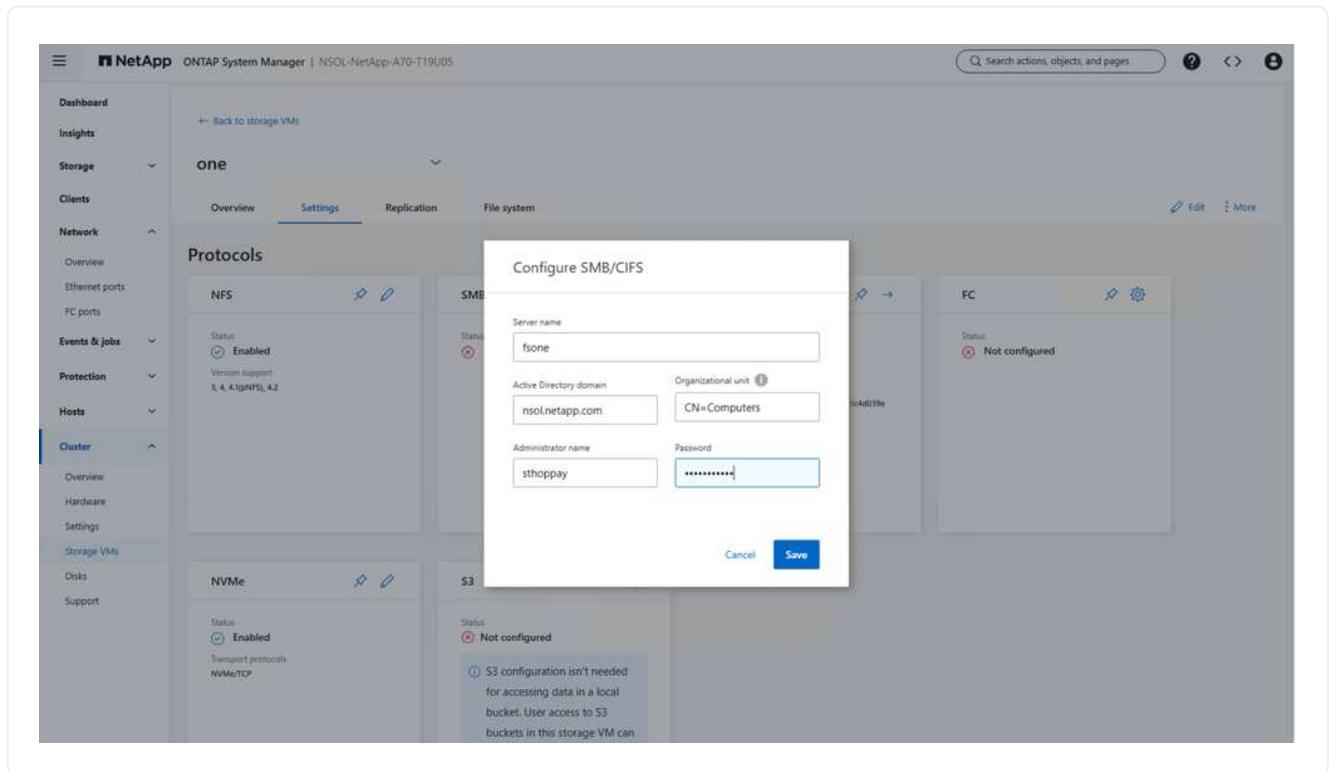
#### Show example

The screenshot shows the NetApp ONTAP System Manager interface. The main content area is titled "Network overview" and displays a table of "Network interfaces". The table has columns for Name, Status, Storage VM, IPspace, Address, Current node, Current port, Portset, Protocols, Type, and Throughput. There are four rows of data representing different LIFs.

Name	Status	Storage VM	IPspace	Address	Current node	Current port	Portset	Protocols	Type	Throughput
lif_one_a1	🟢	one	Default	192.168.121.245	NSOL-NetApp-A70-T19U05a	a0a-121		SMB/CIFS, ...	Data	0
lif_one_a2	🟢	one	Default	192.168.121.246	NSOL-NetApp-A70-T19U05a	a0a-121		SMB/CIFS, ...	Data	0
lif_one_b1	🟢	one	Default	192.168.122.245	NSOL-NetApp-A70-T19U05b	a0a-122		SMB/CIFS, ...	Data	0
lif_one_b2	🟢	one	Default	192.168.122.246	NSOL-NetApp-A70-T19U05b	a0a-122		SMB/CIFS, ...	Data	0

3. Configure Active Directory or workgroup-based authentication. Follow the steps from the documentation.

## Show example



4. Create a volume. Check the option to distribute data across the cluster to use FlexGroup. Ensure Anti-Ransomware protection is enabled on the volume.

Show example



## Add volume

×

Name

smb01

Storage VM

one

Add as a cache for a remote volume (FlexCache)

Simplifies file distribution, reduces WAN latency, and lowers WAN bandwidth costs.

## Storage and optimization

Capacity

150

TiB

Performance service level

Extreme

Not sure? [Get help selecting type](#)

Optimization options

Distribute volume data across the cluster (FlexGroup) ⓘ

Advanced capacity balancing

ONTAP distributes file data to maintain balance as files grow.

## Access permissions

Export via NFS

Share via SMB/CIFS

Name

one\_sys\_01

Grant access to user(s)

Everyone

Permission

Full control

## SnapLock

[SnapLock considerations](#)

Enable SnapLock

With SnapLock, files can be stored and committed to a non-erasable, non-rewritable state either forever or for a designated retention period.

## Protection

Enable snapshots (local)

Snapshot policy

default

Schedule na...	Maximum snapshots	Schedule	SnapMirror label	SnapLock retention period
hourly	6	At 5 minutes past the hour, every hour	-	0 second
daily	2	At 12:10 AM, every day	daily	0 second
weekly	2	At 12:15 AM, only on Sunday	weekly	0 second

Enable snapshot locking ⓘ

Enables the ability to lock snapshots that were created either manually or by snapshot policies. The snapshots are locked only when a retention period is specified.

Enable SnapMirror (local or remote)

Save

Cancel

[Save to Ansible playbook](#)

5. Create an SMB share and adjust permissions. Follow [ONTAP 9 documentation](#) for more information.

### Show example

## Edit Share ×

Share Name  
**smb01**

Path  
**/smb01**

Description

Access permission

User/group	User type	Access permission	
Everyone	Windows	Full control	

[+ Add](#)

Symbolic links

Symlinks

Symlinks and widelinks

Disable

Share properties

Enable continuous availability  
Enable this function to have uninterrupted access to shares that contain Hyper-V and SQL Server over SMB.

Allow clients to access the snapshots directory  
Client systems will be able to access the snapshots directory.

Encrypt data while accessing this share  
Encrypts data using SMB 3.0 to prevent unauthorized file access on this share.

Enable oplocks  
Allows clients to lock files and cache content locally, which can increase the performance for file operations.

Enable change notify  
Allows SMB clients to request for change notifications for directories on this share.

Enable access-based enumeration (ABE)  
Displays folders or other shared resources based on the access permissions of the user.

6. Provide the SMB server, share name, and credentials to the virtualization administrator.

### **Virtualization administrator tasks**

Complete these tasks to add the SMB share as Datastore in OpenNebula and enable multichannel for improved performance and fault tolerance.

1. Collect the SMB server, share name, and credentials for share authentication.
2. Ensure following packages are installed on Fedora `sssd realmd adcli oddjob oddjob-mkhomedir samba-common-tools krb5-workstation cifs-utils` for Active Directory integration and SMB mount support. Debian packages are `realmd sssd sssd-tools libnss-sss libpam-sss adcli samba-common-bin packagekit krb5-user cifs-utils`.
3. Ensure at least two interfaces are configured in different VLANs for fault tolerance. Verify the NIC supports RSS.
4. SSH to one of the frontend server and create a configuration file based on desired Datastore type. Sample files are shown below:

## Backup

### 1. For Restic,

```
$cat smb-restic.conf
NAME = "Backup-Restic-SMB"
TYPE = "BACKUP_DS"

DS_MAD = "restic"
TM_MAD = "-"

RESTIC_PASSWORD = "<restic_password>"
RESTIC_SFTP_SERVER = "<backup server>"
```

### 1. For Rsync,

```
$cat smb-rsync.conf
NAME = "Backup-Rsync-SMB"
TYPE = "BACKUP_DS"

DS_MAD = "rsync"
TM_MAD = "-"

RSYNC_USER = "<rsync_user>"
RSYNC_HOST = "<backup server>"
```

## File

```
$cat smb-kernel.conf
NAME = "File-Kernel-SMB"
TYPE = "FILE_DS"
DS_MAD = "fs"
TM_MAD = "local"
SAFE_DIRS = "/var/tmp/files"
```

## Image

```
$cat smb-image.conf
NAME = "Image-SMB"
TYPE = "IMAGE_DS"
DS_MAD = "fs"
TM_MAD = "shared"
```

## System

```
$cat smb-system.conf
NAME = "System-SMB"
TYPE = "SYSTEM_DS"
TM_MAD = "shared"
```

5. Execute `onedatastore create <configuration file>`. Note the datastore ID returned after creation.

```
onedatastore create smb-system.conf
ID: 100
```

6. Create a smb credential file in `/etc/`. This step is not required if using kerberos authentication (KVM host joined to `<domain>`).

```
$cat /etc/smb-credentials-<datastore_id>.cfg
username=<smb_username>
password=<smb_password>
domain=<smb_domain>
```

7. Set proper permissions (640) on the credential file. Change ownership to `oneadmin` user and group if required.
8. Gather the uid and gid of `oneadmin` user using `id oneadmin` command.
9. Update `/etc/fstab` or `automount` configuration to enable multichannel. Assuming the default datastore location as `/var/lib/one/datastores`. If not check the `DATASTORE_LOCATION` parameter in `/etc/one/oned.conf`. Ensure the `<datastore_id>` folder exists under the `datastores` location. Sample entries are shown below:

### Using /etc/fstab

```
# credentials mapping to file option is not required when using
kerberos authentication
//<smb_server>/<smb_share> /var/lib/one/datastores/<datastore_id>
cifs credentials=/etc/smb-credentials-
<datastore_id>.cfg,_netdev,noauto,x-systemd.automount,vers
=3.0,multichannel,max_channels=16,nofail,uid=<oneadmin uid>,gid
=<oneadmin gid> 0 0
```

### Using automount

```
# credentials mapping to file option is not required when using
kerberos authentication
/var/lib/one/datastores/<datastore_id> -fstype=cifs,credentials
=/etc/smb-credentials-<datastore_id>.cfg,vers
=3.0,multichannel,max_channels=16,uid=<oneadmin uid>,gid=<oneadmin
gid> ://<smb_server>/<smb_share>
```

10. Mount the datastore using `mount -a` or `systemctl reload autofs` command.
11. Verify the datastore is mounted with `mount` command and verify the datastore capacity with `onedatastore show <datastore_id>` command.
12. Ensure oneadmin user and group own the datastore folder. Adjust permissions using `chown -R oneadmin:oneadmin /var/lib/one/datastores/<datastore_id>` command.

## Configure NFS storage for OpenNebula using ONTAP

Configure NFS storage for OpenNebula using NetApp ONTAP. Use nConnect or session trunking with pNFS (v4.1 or later )while using FlexGroup volumes for efficient resource management, fault tolerance, and performance enhancements. A single NFS export can be used for both Image and System datastores for a OpenNebula cluster. When planning to use FlexCache, dedicate the nfs export for Image datastores only.

Consider MetroCluster configuration for high availability and disaster recovery scenarios.

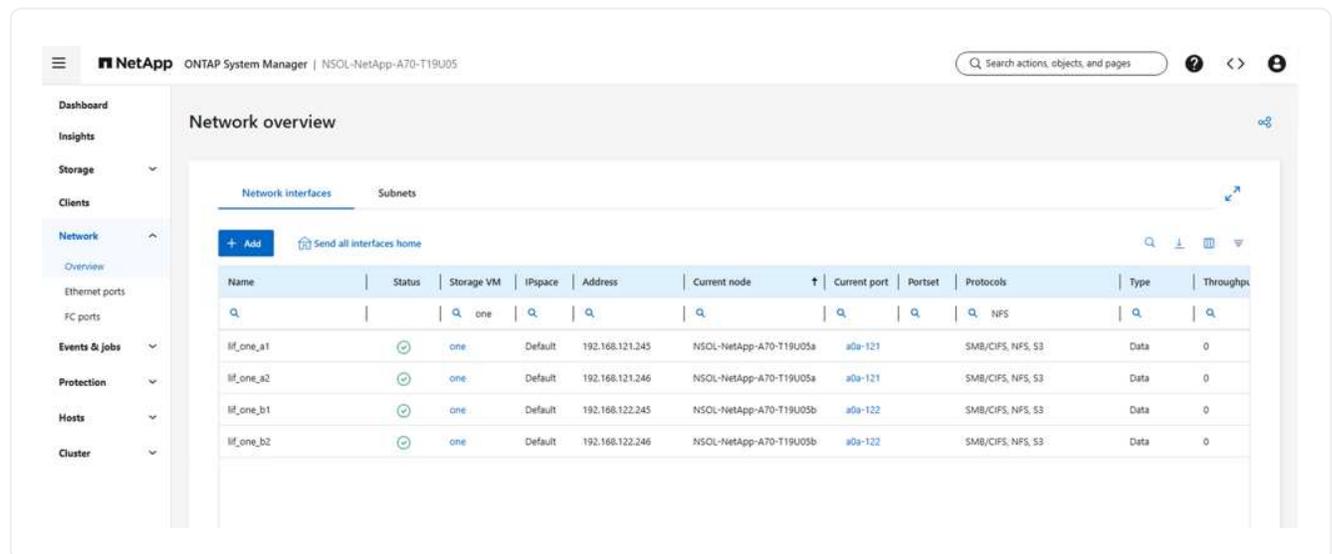
If you are new to ONTAP, use System Manager Interface to complete these tasks.

### Storage administrator tasks

Complete these tasks to provision NFS storage on ONTAP for use with OpenNebula.

1. Enable the SVM for NFS. Refer to [ONTAP 9 documentation](#).
2. Create at least two LIFs per controller. Follow the steps from the documentation. For reference, here is a screenshot of LIFs used in the lab.

## Show example

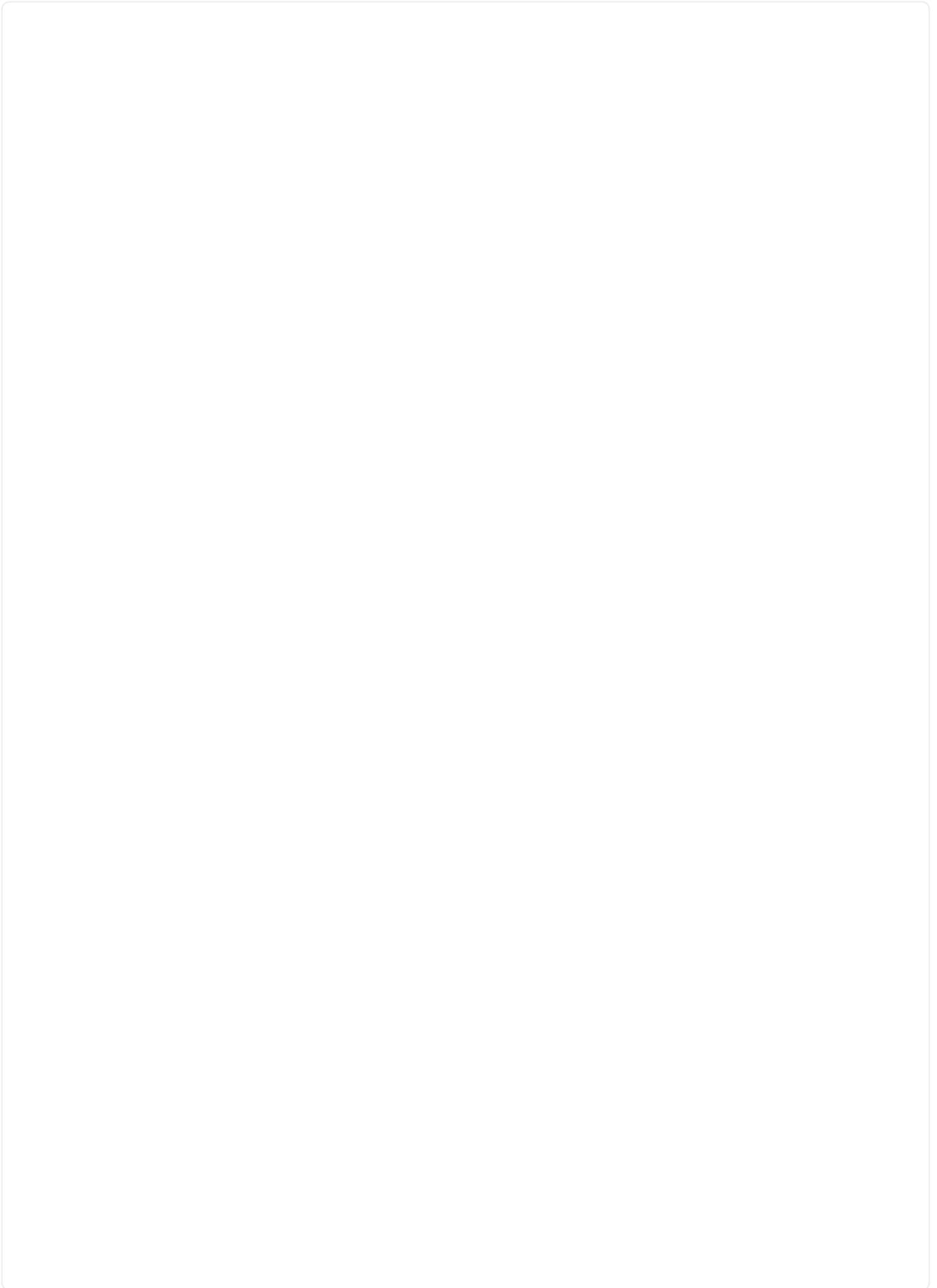


The screenshot displays the NetApp ONTAP System Manager interface. The left sidebar shows a navigation menu with categories like Dashboard, Insights, Storage, Clients, Network, Events & Jobs, Protection, Hosts, and Cluster. The main content area is titled "Network overview" and has two tabs: "Network interfaces" (selected) and "Subnets". Below the tabs, there is a table of network interfaces. The table has columns for Name, Status, Storage VM, IPspace, Address, Current node, Current port, Portset, Protocols, Type, and Throughput. The table contains four rows of data, each representing a network interface.

Name	Status	Storage VM	IPspace	Address	Current node	Current port	Portset	Protocols	Type	Throughput
Mf_one_a1	🟢	one	Default	192.168.121.245	NSQL-NetApp-A70-T19U05a	a0a-121	a0a-121	SMB/CIFS, NFS, S3	Data	0
Mf_one_a2	🟢	one	Default	192.168.121.246	NSQL-NetApp-A70-T19U05a	a0a-121	a0a-121	SMB/CIFS, NFS, S3	Data	0
Mf_one_b1	🟢	one	Default	192.168.122.245	NSQL-NetApp-A70-T19U05b	a0a-122	a0a-122	SMB/CIFS, NFS, S3	Data	0
Mf_one_b2	🟢	one	Default	192.168.122.246	NSQL-NetApp-A70-T19U05b	a0a-122	a0a-122	SMB/CIFS, NFS, S3	Data	0

3. Create or update an NFS export policy to provide access to OpenNebula host IP addresses or subnets. Refer to [Export policy creation](#) and [Add rule to an export policy](#).
4. [Create a volume](#). For large capacity needs (>100TB), check the option to distribute data across the cluster to use FlexGroup. If using FlexGroup, consider enabling pNFS on the SVM for better performance by following [Enable pNFS on SVM](#). When using pNFS, ensure the OpenNebula hosts has data access to all controllers (data LIFs). Ensure Anti-Ransomware protection is enabled on the volume.

Show example



## Add volume

×

Name

NFS01

Storage VM

one

Add as a cache for a remote volume (FlexCache)

Simplifies file distribution, reduces WAN latency, and lowers WAN bandwidth costs.

## Storage and optimization

Capacity

120

TiB

Performance service level

Extreme

Not sure? [Get help selecting type](#)

Optimization options

Distribute volume data across the cluster (FlexGroup) ⓘ

Advanced capacity balancing

ONTAP distributes file data to maintain balance as files grow.

⚠ You can't revert to an ONTAP version earlier than 9.16.1 after enabling. This option can't be undone. [Learn more](#)

## Access permissions

Export via NFS

Grant access to host

default

Create a new export policy, or select an existing export policy.

Rule index	Clients	Access protocols	Read-only rule	Read/write rule
1	192.168.121.0/24	Any	Any	Any
2	192.168.122.0/24	Any	Any	Any

Share via SMB/CIFS

## SnapLock

[SnapLock considerations](#)

Enable SnapLock

With SnapLock, files can be stored and committed to a non-erasable, non-rewritable state either forever or for a designated retention period.

## Protection

Enable snapshots (local)

Snapshot policy

default

Schedule na...	Maximum snapshots	Schedule	SnapMirror label	SnapLock retention period
hourly	6	At 5 minutes past the hour, every hour	-	0 second
daily	2	At 12:10 AM, every day	daily	0 second
weekly	2	At 12:15 AM, only on Sunday	weekly	0 second

Enable snapshot locking ⓘ

Enables the ability to lock snapshots that were created either manually or by snapshot policies. The snapshots are locked only when a retention period is specified.

Enable SnapMirror (local or remote)

Save

Cancel

Save to Ansible playbook

5. Notify the virtualization administrator that the NFS volume is ready and provide NFS export path details.

### **Virtualization administrator tasks**

Complete these tasks to add the NFS volume as Datastore in OpenNebula and configure nConnect or session trunking for improved performance.

1. Ensure at least two interfaces are configured in different VLANs for fault tolerance. Use NIC bonding.
2. SSH to one of the frontend server and create a configuration file based on desired Datastore type. Sample files are shown below:

## Backup

### 1. For Restic,

```
$cat nfs-restic.conf
NAME = "Backup-Restic-NFS"
TYPE = "BACKUP_DS"

DS_MAD = "restic"
TM_MAD = "-"

RESTIC_PASSWORD = "<restic_password>"
RESTIC_SFTP_SERVER = "<backup server>"
```

### 1. For Rsync,

```
$cat nfs-rsync.conf
NAME = "Backup-Rsync-NFS"
TYPE = "BACKUP_DS"

DS_MAD = "rsync"
TM_MAD = "-"

RSYNC_USER = "<rsync_user>"
RSYNC_HOST = "<backup server>"
```

## File

```
$cat nfs-kernel.conf
NAME = "File-Kernel-NFS"
TYPE = "FILE_DS"
DS_MAD = "fs"
TM_MAD = "local"
SAFE_DIRS = "/var/tmp/files"
```

## Image

```
$cat nfs-image.conf
NAME = "Image-NFS"
TYPE = "IMAGE_DS"
DS_MAD = "fs"
TM_MAD = "shared"
```

## System

```
$cat nfs-system.conf
NAME = "System-NFS"
TYPE = "SYSTEM_DS"
TM_MAD = "shared"
```

3. Execute `onedatastore create <configuration file>`. Note the datastore ID returned after creation.

```
onedatastore create nfs-system.conf
ID: 101
```

4. Gather the uid and gid of `oneadmin` user using `id oneadmin` command.
5. Update `/etc/fstab` or `automount` configuration to mount the datastore with desired mount options. Assuming the default datastore location as `/var/lib/one/datastores`. Can be validated with `onedatastore show <datastore_id>`. If not check the `DATASTORE_LOCATION` parameter in `/etc/one/oned.conf`. Ensure the `<datastore_id>` folder exists under the `datastores` location. Sample entries are shown below:

### Using /etc/fstab

```
# To use session trunking, use the option trunkdiscovery
//<nfs_server>/<nfs_share> /var/lib/one/datastores/<datastore_id>
nfs nconnect=8,max_channels=16,_netdev,noauto,x-
systemd.automount,nofail,uid=<oneadmin uid>,gid=<oneadmin gid> 0 0
```

### Using automount

```
# To use session trunking, use the option trunkdiscovery
/var/lib/one/datastores/<datastore_id> -fstype=nfs,nconnect
=8,max_channels=16,_netdev,noauto,x-systemd.automount,nofail,uid
=<oneadmin uid>,gid=<oneadmin gid> <nfs_server>:/<nfs_share>
```

6. Mount the datastore using `mount -a` or `systemctl reload autofs` command.
7. Verify the datastore is mounted with `mount` command and verify the datastore capacity with `onedatastore show <datastore_id>` command.
8. Ensure `oneadmin` user and group own the datastore folder. Adjust permissions using `chown -R oneadmin:oneadmin /var/lib/one/datastores/<datastore_id>` command.
9. To verify the `nConnect` option is set, run `ss -an | grep :2049` on any OpenNebula host and check for multiple connections to the NFS server IP. To verify pNFS is enabled, run `nfsstat -c` and check the layout-related metrics. Based on data traffic, multiple connections to data LIFs should be visible.



In session trunking, the nconnect option is set on only one of the trunk interfaces. With pNFS, the nconnect option is set on metadata and data interfaces. For production environments, use either nConnect or session trunking, not both.

## Configure NetApp Datastore with iSCSI for OpenNebula

Configure OpenNebula Datastores using iSCSI protocol with NetApp ONTAP running on AFF or FAS systems. This configuration enables block-level storage access over standard Ethernet networks with multipath support. This datastore setup utilizes native ONTAP features, including snapshots and cloning, to enhance storage efficiency and data protection.

### Initial virtualization administrator tasks

Complete these initial tasks to prepare OpenNebula hosts for iSCSI connectivity and collect the necessary information for the storage administrator.

1. Verify two Linux VLAN interfaces are available.
2. Ensure multipath-tools and iSCSI initiator utilities are installed on all OpenNebula hosts and starts on boot.

#### Debian/Ubuntu

```
apt list | grep multipath-tools
# If need to install, execute the following line.
apt-get install multipath-tools open-iscsi
# If /etc/multipath.conf is not present, first make sure the
multipathd service is started.
systemctl enable --now multipathd
systemctl enable --now open-iscsi
```

#### RHEL/AlmaLinux

```
dnf list installed | grep device-mapper-multipath
# If need to install, execute the following line.
dnf install device-mapper-multipath iscsi-initiator-utils
# If /etc/multipath.conf is not present, first make sure the
multipathd service is started.
systemctl enable --now multipathd
systemctl enable --now iscsid
```

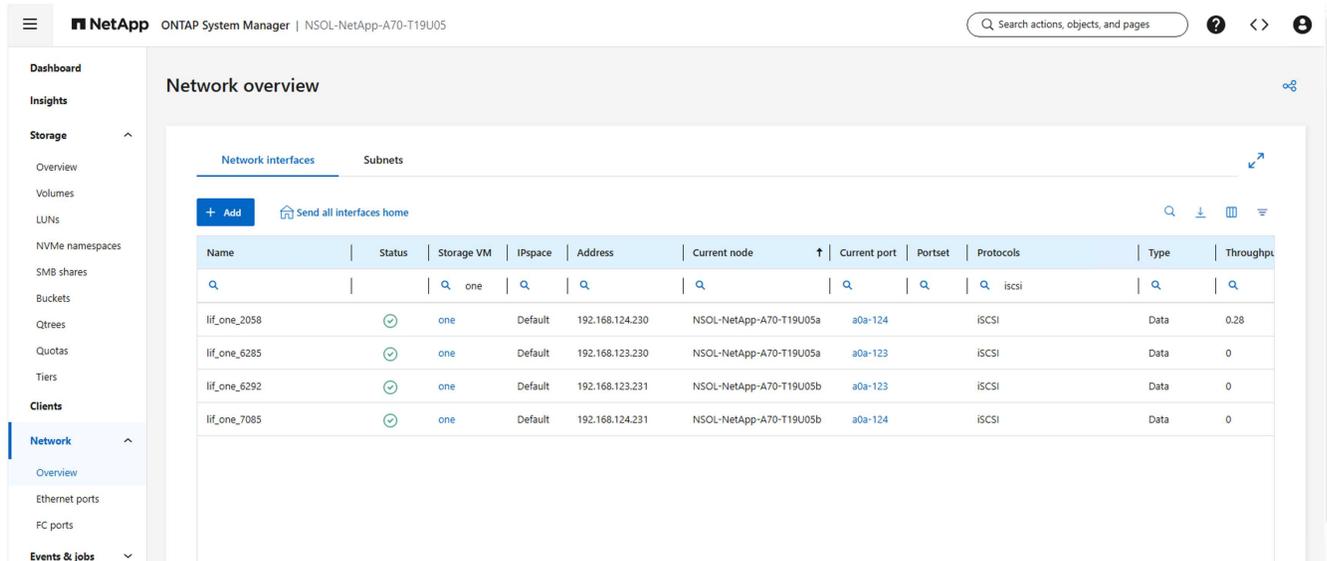
3. Collect the iSCSI host IQN for all OpenNebula hosts and provide it to the storage administrator.

```
cat /etc/iscsi/initiator.name
```

## Storage administrator tasks

If you are new to ONTAP, use System Manager for a better experience.

1. Ensure the SVM is available with iSCSI protocol enabled. Follow [ONTAP 9 documentation](#).
2. Create two LIFs per controller dedicated for iSCSI. Two LIFs per controller are recommended for redundancy and multipath performance. Ensure the LIFs are created on the VLAN interfaces configured on OpenNebula hosts. Jumbo frames (MTU 9000) are recommended for better performance.



The screenshot shows the NetApp System Manager interface for a specific SVM (NSOL-NetApp-A70-T19U05). The 'Network overview' page displays a table of network interfaces. The table has columns for Name, Status, Storage VM, IPspace, Address, Current node, Current port, Portset, Protocols, Type, and Throughput. Four interfaces are listed, all with a status of 'OK' and using the 'one' storage VM.

Name	Status	Storage VM	IPspace	Address	Current node	Current port	Portset	Protocols	Type	Throughput
lif_one_2058	OK	one	Default	192.168.124.230	NSOL-NetApp-A70-T19U05a	a0a-124		iSCSI	Data	0.28
lif_one_6285	OK	one	Default	192.168.123.230	NSOL-NetApp-A70-T19U05a	a0a-123		iSCSI	Data	0
lif_one_6292	OK	one	Default	192.168.123.231	NSOL-NetApp-A70-T19U05b	a0a-123		iSCSI	Data	0
lif_one_7085	OK	one	Default	192.168.124.231	NSOL-NetApp-A70-T19U05b	a0a-124		iSCSI	Data	0

3. Create an igroup and populate the host iSCSI initiators. Typically one igroup is created for one OpenNebula cluster. Include frontend servers and hypervisor hosts in the same igroup to support both Image and System datastores.
4. Create an ONTAP role and user account with ONTAP REST API access scoped to the target SVM. This user will be used by the NetApp driver in OpenNebula. See [Work with users and roles ONTAP documentation](#) for more information. Keep note of the Username and Password, to be used in the Virtualization Configuration Tasks.
5. Gather the SVM iSCSI Target IQN and UUIDs for the following resources for use in the Virtualization Configuration Tasks:
  - The SVM
  - The Aggregate(s) / Tier(s) to be used
  - The igroup with the OpenNebula hosts
  - The iSCSI Target IQN (typically same as SVM IQN). Virtualization admin can retrieve this information using `iscsiadm -m session` command after logging into one of the OpenNebula hosts and discovering the iSCSI target.

```
NETAPP_SVM="85c23687-d5d9-11f0-86c4-d039eac4d4b3"  
NETAPP_AGGREGATES="6e8f9995-42dd-400a-a440-646639dc5d0b"  
NETAPP_IGROUP="5ad9faf3-d62c-11f0-86c4-d039eac4d4b3"  
NETAPP_TARGET="iqn.1992-  
08.com.netapp:sn.85c23687d5d911f086c4d039eac4d4b3:vs.6"
```

TIP: System Manager displays the UUID in the URL when viewing the resource details.

## Final virtualization administrator tasks

Complete these tasks to configure the iSCSI Datastore on OpenNebula.

1. SSH to one of the frontend server and discover all iSCSI Lif portals by providing one of the iSCSI data lif addresses.

```
iscsiadm -m discovery -t sendtargets -p <iscsi data lif address>  
iscsiadm -m node  
iscsiadm -m node -l
```

2. Create a configuration file based on desired Datastore type. For complete attribute list, refer [OpenNebula NetApp SAN documentation](#). Sample files are shown below:

## Image

```
$cat netapp-image.conf
NAME = "Image-NetApp-iSCSI"
TYPE = "IMAGE_DS"
DS_MAD = "netapp"
TM_MAD = "netapp"
DISK_TYPE = "BLOCK"
NETAPP_HOST = "<ontap_cluster_ip>"
NETAPP_USER = "<ontap_api_user>"
NETAPP_PASS = "<ontap_api_password>"
NETAPP_SVM = "<ontap_svm_uuid>"
NETAPP_AGGREGATES = "<ontap_aggregate_uuid>"
NETAPP_IGROUP = "<ontap_igroup_uuid>"
NETAPP_TARGET = "<ontap_iscsi_target_iqn>"
# Optional suffix to share SVM across multiple tenants
NETAPP_SUFFIX = "t1"
```

## System

```
$cat netapp-system.conf
NAME = "System-NetApp-iSCSI"
TYPE = "SYSTEM_DS"
TM_MAD = "netapp"
DISK_TYPE = "BLOCK"
NETAPP_HOST = "<ontap_cluster_ip>"
NETAPP_USER = "<ontap_api_user>"
NETAPP_PASS = "<ontap_api_password>"
NETAPP_SVM = "<ontap_svm_uuid>"
NETAPP_AGGREGATES = "<ontap_aggregate_uuid>"
NETAPP_IGROUP = "<ontap_igroup_uuid>"
NETAPP_TARGET = "<ontap_iscsi_target_iqn>"
# Optional suffix to share SVM across multiple tenants
NETAPP_SUFFIX = "t1"
```

3. Execute `onedatastore create <configuration file>`. Note the datastore ID returned after creation.

```
onedatastore create netapp-system.conf
ID: 105
```

4. Verify the datastore is created successfully by executing `onedatastore show <datastore_id>`.
5. Download Apps on Image datastore and create VM using Templates to provision on System datastore.

6. Check the LUNs created on ONTAP for the image and virtual machine disks. The naming convention used is as follows:

- a. Image datastore: `one_<datastore_id>_<image_id>_<suffix>` (volume),  
`one_<datastore_id>_<image_id>_<suffix>_lun` (LUN)
- b. System datastore: `one_<vm_id>_disk_<disk_id>_<suffix>` (volume),  
`one_<datastore_id>_<vm_id>_disk_<disk_id>_<suffix>_lun` (LUN)

### Show example

Name	Storage VM	Volume	Size	IOPS	Latency (ms)	Throughput (MB/s)
one_106_47_11_lun	one	one_106_47_11	4 GiB	-	-	-
one_106_46_11_lun	one	one_106_46_11	2 GiB	-	-	-
one_106_45_11_lun	one	one_106_45_11	37 GiB	-	-	-
one_106_40_11_lun	one	one_106_40_11	120 GiB	-	-	-
one_106_39_11_lun	one	one_106_39_11	10 GiB	-	-	-
one_106_34_11_lun	one	one_106_34_11	10 GiB	0	0.02	0
one_106_33_11_lun	one	one_106_33_11	10 GiB	0	0.03	0
one_106_32_11_lun	one	one_106_32_11	512 MiB	0	0.02	0
one_39_disk_0_11_lun	one	one_39_disk_0_11	2 GiB	0	0.03	0
one_38_disk_0_11_lun	one	one_38_disk_0_11	4 GiB	0	0.03	0
one_35_disk_0_11_lun	one	one_35_disk_0_11	37 GiB	0	0.02	0
one_34_disk_0_11_lun	one	one_34_disk_0_11	37 GiB	0	0.02	0
one_33_disk_0_11_lun	one	one_33_disk_0_11	37 GiB	0	0.03	0
one_32_disk_0_11_lun	one	one_32_disk_0_11	10 GiB	0	0.02	0

## Configure LVM Thin with ONTAP FC for OpenNebula

Configure Logical Volume Manager (LVM) Datastore for shared storage across OpenNebula hosts using Fibre Channel protocol with NetApp ONTAP. This configuration enables block-level storage access with high performance and low latency.

### Initial virtualization administrator tasks

Complete these initial tasks to prepare OpenNebula hosts for FC connectivity and collect the necessary information for the storage administrator.

1. Verify two HBA interfaces are available.
2. Ensure multipath-tools is installed on all OpenNebula hosts and starts on boot.

## Debian/Ubuntu

```
apt list | grep multipath-tools
# If need to install, execute the following line.
apt-get install multipath-tools
# If /etc/multipath.conf is not present, first make sure the
multipathd service is started.
systemctl enable --now multipathd
```

## RHEL/AlmaLinux

```
dnf list installed | grep device-mapper-multipath
# If need to install, execute the following line.
dnf install device-mapper-multipath
# If /etc/multipath.conf is not present, first make sure the
multipathd service is started.
systemctl enable --now multipathd
```

3. Collect the WWPN for all OpenNebula hosts and provide it to the storage administrator and administrator who takes care of fabric zoning.

```
cat /sys/class/fc_host/host*/port_name
```

## Storage administrator tasks

If you are new to ONTAP, use System Manager for a better experience.

1. Ensure the SVM is available with FC protocol enabled. Follow [ONTAP 9 documentation](#).
2. Create two LIFs per controller dedicated for FC. Gather the WWPN addresses for the FC LIFs created and provide them to the administrator who takes care of fabric zoning.
3. Create an igroup and populate the host FC initiators. Typically one igroup is created for one OpenNebula cluster. Include frontend servers and hypervisor hosts in the same igroup to support both Image and System datastores.
4. Create the LUN with the desired size on the SVM and present it to the igroup created in the previous step. Ensure Anti-Ransomware protection is enabled on the security tab for ASA systems and on the volume security tab for AFF/FAS systems.
5. Notify the virtualization administrator that the LUN is created.

## Final virtualization administrator tasks

Complete these tasks to configure the FC LUN as shared LVM Datastore in OpenNebula.

1. SSH to all OpenNebula servers and complete the following steps on each host.
2. Execute `rescan-scsi-bus.sh` or `echo "- - -" > /sys/class/scsi_host/host*/scan` to

rescan the SCSI bus and detect new LUNs.

3. Verify the LUN is visible on all OpenNebula hosts using `lsblk -S` or `fdisk -l` command. Note the device name (e.g., `sde`, `sdf`) for the LUN created.
4. Add the device to multipath configuration by executing `multipath -a /dev/<device_name>`. Then, execute `multipath -r` to reload the multipath configuration. Verify the multipath configuration by executing `multipath -ll` command.
5. SSH to one of the frontend server and create a configuration file based on desired Datastore type. For complete attribute list, refer [OpenNebula LVM documentation](#). Sample files are shown below:

## Backup

### 1. For Restic,

```
$cat fc-restic.conf
NAME = "Backup-Restic-FC"
TYPE = "BACKUP_DS"

DS_MAD = "restic"
TM_MAD = "-"

RESTIC_PASSWORD = "<restic_password>"
RESTIC_SFTP_SERVER = "<backup server>"
```

### 1. For Rsync,

```
$cat fc-rsync.conf
NAME = "Backup-Rsync-FC"
TYPE = "BACKUP_DS"

DS_MAD = "rsync"
TM_MAD = "-"

RSYNC_USER = "<rsync_user>"
RSYNC_HOST = "<backup server>"
```

## File

```
$cat fc-kernel.conf
NAME = "File-Kernel-FC"
TYPE = "FILE_DS"
DS_MAD = "fs"
TM_MAD = "local"
SAFE_DIRS = "/var/tmp/files"
```

## Image

```
$cat fc-image.conf
NAME = "Image-FC01"
TYPE = "IMAGE_DS"
DS_MAD = "fs"
TM_MAD = "fs_lvm_ssh"
DISK_TYPE = "block"
LVM_THIN_ENABLE = "yes"
```

## System

```
$cat fc-system.conf
NAME = "System-FC02"
TYPE = "SYSTEM_DS"
TM_MAD = "fs_lvm_ssh"
DISK_TYPE = "block"
BRIDGE_LIST = "<space-separated list of OpenNebula hosts>" # If LUN
not presented to frontend hosts
LVM_THIN_ENABLE = "yes"
```

6. Execute `onedatastore create <configuration file>`. Note the datastore ID returned after creation.

```
onedatastore create fc-system.conf
ID: 107
```

7. Create volume group on the FC LUN using `vgcreate <vg_name> <multipath_device>` command. For Image datastores, the volume group name can be named anything desired. For System datastores, the volume group name must be of format `vg-one-<datastore id>`. This is required for OpenNebula to identify the correct volume group for system datastores. Proceed with following steps if you are creating Backup/File/Image datastore. For system datastores, stop here.
8. Create logical volume thin pool using `lvcreate -l 100%FREE -n <logical volume name> <volume group name>` command. For System datastores, OpenNebula automatically creates the LVM thin pool when required.
9. Create filesystem on the logical volume using `mkfs.ext4 /dev/<volume group>/<logical volume>` command. System datastores do not require filesystem creation.
10. Update `/etc/fstab` or automount configuration to mount the datastore with desired mount options. Assuming the default datastore location as `/var/lib/one/datastores`. Can be validated with `onedatastore show <datastore_id>`. If not check the `DATASTORE_LOCATION` parameter in `/etc/one/oned.conf`. Ensure the `<datastore_id>` folder exists under the datastores location. Sample entries are shown below:

### Using /etc/fstab

```
/dev/<vg name>/<logical volume>
/var/lib/one/datastores/<datastore_id> ext4 _netdev,noauto,x-
systemd.automount,nofail 0 2
```

### Using automount

```
/var/lib/one/datastores/<datastore_id> -fstype
=ext4,_netdev,noauto,x-systemd.automount,nofail,rw :/dev/<vg
name>/<logical volume>
```

11. Mount the datastore using `mount -a` or `systemctl reload autofs` command.
12. Verify the datastore is mounted with `mount` command and verify the datastore capacity with `onedatastore show <datastore_id>` command.
13. Ensure `oneadmin` user and group own the datastore folder. Adjust permissions using `chown -R oneadmin:oneadmin /var/lib/one/datastores/<datastore_id>` command.

## Configure LVM Thin with ONTAP iSCSI for OpenNebula

Configure Logical Volume Manager (LVM) datastore for shared storage across OpenNebula hosts using iSCSI protocol with NetApp ONTAP. This configuration enables block-level storage access over standard Ethernet networks with multipath support.

### Initial virtualization administrator tasks

Complete these initial tasks to prepare OpenNebula hosts for iSCSI connectivity and collect the necessary information for the storage administrator.

1. Verify two Linux VLAN interfaces are available.
2. Ensure `multipath-tools` and iSCSI initiator utilities are installed on all OpenNebula hosts and starts on boot.

#### Debian/Ubuntu

```
apt list | grep multipath-tools
# If need to install, execute the following line.
apt-get install multipath-tools open-iscsi
# If /etc/multipath.conf is not present, first make sure the
multipathd service is started.
systemctl enable --now multipathd
systemctl enable --now open-iscsi
```

#### RHEL/AlmaLinux

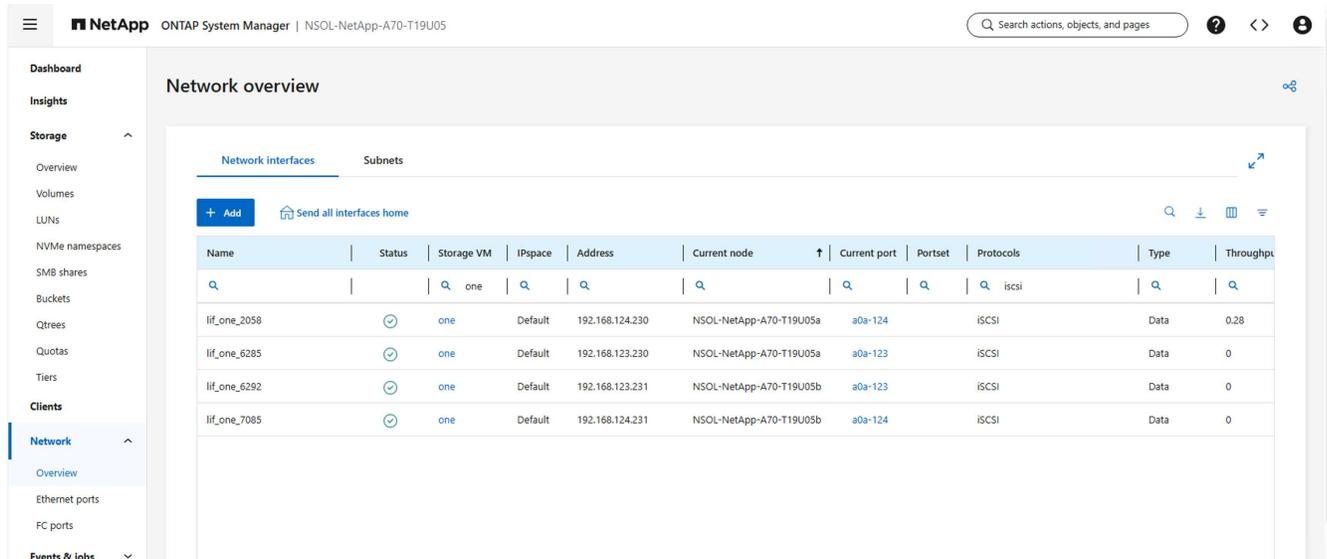
```
dnf list installed | grep device-mapper-multipath
# If need to install, execute the following line.
dnf install device-mapper-multipath iscsi-initiator-utils
# If /etc/multipath.conf is not present, first make sure the
multipathd service is started.
systemctl enable --now multipathd
systemctl enable --now iscsid
```

3. Collect the iSCSI host IQN for all OpenNebula hosts and provide it to the storage administrator.

```
cat /etc/iscsi/initiator.name
```

If you are new to ONTAP, use System Manager for a better experience.

1. Ensure the SVM is available with iSCSI protocol enabled. Follow [ONTAP 9 documentation](#).
2. Create two LIFs per controller dedicated for iSCSI. Two LIFs per controller are recommended for redundancy and multipath performance. Ensure the LIFs are created on the VLAN interfaces configured on OpenNebula hosts. Jumbo frames (MTU 9000) are recommended for better performance.



The screenshot shows the NetApp ONTAP System Manager interface. The main content area is titled "Network overview" and displays a table of network interfaces. The table has columns for Name, Status, Storage VM, IPspace, Address, Current node, Current port, Portset, Protocols, Type, and Throughput. There are four rows of data, all with a status of "one" and protocols of "iscsi".

Name	Status	Storage VM	IPspace	Address	Current node	Current port	Portset	Protocols	Type	Throughput
lif_one_2058	one	one	Default	192.168.124.230	NSOL-NetApp-A70-T19U05a	a0a-124		iscsi	Data	0.28
lif_one_6285	one	one	Default	192.168.123.230	NSOL-NetApp-A70-T19U05a	a0a-123		iscsi	Data	0
lif_one_6292	one	one	Default	192.168.123.231	NSOL-NetApp-A70-T19U05b	a0a-123		iscsi	Data	0
lif_one_7085	one	one	Default	192.168.124.231	NSOL-NetApp-A70-T19U05b	a0a-124		iscsi	Data	0

3. Create LUNs and present to the host iSCSI initiators. Typically one igroup is created for one OpenNebula cluster. Include frontend servers and hypervisor hosts in the same igroup to support both Image and System datastores.
4. Notify the virtualization administrator that the LUN is created.

## Final virtualization administrator tasks

Complete these tasks to configure the iSCSI LUN as shared LVM Datastore in OpenNebula.

1. SSH to one of the frontend server and discover all iSCSI Lif portals by providing one of the iSCSI data lif addresses.

```
iscsiadm -m discovery -t sendtargets -p <iscsi data lif address>
iscsiadm -m node
iscsiadm -m node -l
iscsiadm -m session
```

2. Execute `rescan-scsi-bus.sh` or `echo "- - -" > /sys/class/scsi_host/host*/scan` to rescan the SCSI bus and detect new LUNs.
3. Verify the LUN is visible on all OpenNebula hosts using `lsblk -S` or `fdisk -l` command.
4. Execute `iscsiadm -m session -P 3` to retrieve LUN to device name mapping.
5. Add the device to multipath configuration by executing `multipath -a /dev/<device_name>`. Then, execute `multipath -r` to reload the multipath configuration. Verify the multipath configuration by executing `multipath -ll` command.

6. Create a configuration file based on desired Datastore type. For complete attribute list, refer [OpenNebula LVM documentation](#). Sample files are shown below:

## Backup

### 1. For Restic,

```
$cat iscsi-restic.conf
NAME = "Backup-Restic-iSCSI01"
TYPE = "BACKUP_DS"

DS_MAD = "restic"
TM_MAD = "-"

RESTIC_PASSWORD = "<restic_password>"
RESTIC_SFTP_SERVER = "<backup server>"
```

### 1. For Rsync,

```
$cat iscsi-rsync.conf
NAME = "Backup-Rsync-iSCSI02"
TYPE = "BACKUP_DS"

DS_MAD = "rsync"
TM_MAD = "-"

RSYNC_USER = "<rsync_user>"
RSYNC_HOST = "<backup server>"
```

## File

```
$cat iscsi-kernel.conf
NAME = "File-Kernel-iSCSI03"
TYPE = "FILE_DS"
DS_MAD = "fs"
TM_MAD = "local"
SAFE_DIRS = "/var/tmp/files"
```

## Image

```
$cat iscsi-image.conf
NAME = "Image-iSCSI04"
TYPE = "IMAGE_DS"
DS_MAD = "fs"
TM_MAD = "fs_lvm_ssh"
DISK_TYPE = "block"
LVM_THIN_ENABLE = "yes"
```

## System

```
$cat iscsi-system.conf
NAME = "System-iSCSI05"
TYPE = "SYSTEM_DS"
TM_MAD = "fs_lvm_ssh"
DISK_TYPE = "block"
BRIDGE_LIST = "<space-separated list of OpenNebula hosts>" # If LUN
not presented to frontend hosts
LVM_THIN_ENABLE = "yes"
```

7. Execute `onedatastore create <configuration file>`. Note the datastore ID returned after creation.

```
onedatastore create iscsi-system.conf
ID: 106
```

8. Create volume group on the iSCSI LUN using `vgcreate <vg_name> <multipath_device>` command. For Image datastores, the volume group name can be named anything desired. For System datastores, the volume group name must be of format `vg-one-<datastore id>`. This is required for OpenNebula to identify the correct volume group for system datastores. Proceed with following steps if you are creating Backup/File/Image datastore. For system datastores, stop here.
9. Create logical volume thin pool using `lvcreate -l 100%FREE -n <logical volume name> <volume group name>` command. For System datastores, OpenNebula automatically creates the LVM thin pool when required.
10. Create filesystem on the logical volume using `mkfs.ext4 /dev/<volume group>/<logical volume>` command. System datastores do not require filesystem creation.
11. Update `/etc/fstab` or automount configuration to mount the datastore with desired mount options. Assuming the default datastore location as `/var/lib/one/datastores`. Can be validated with `onedatastore show <datastore_id>`. If not check the `DATASTORE_LOCATION` parameter in `/etc/one/oned.conf`. Ensure the `<datastore_id>` folder exists under the datastores location. Sample entries are shown below:

### Using /etc/fstab

```
/dev/<vg name>/<logical volume>
/var/lib/one/datastores/<datastore_id> ext4 _netdev,noauto,x-
systemd.automount,nofail 0 2
```

### Using automount

```
/var/lib/one/datastores/<datastore_id> -fstype
=ext4,_netdev,noauto,x-systemd.automount,nofail,rw :/dev/<vg
name>/<logical volume>
```

12. Mount the datastore using `mount -a` or `systemctl reload autofs` command.
13. Verify the datastore is mounted with `mount` command and verify the datastore capacity with `onedatastore show <datastore_id>` command.
14. Ensure `oneadmin` user and group own the datastore folder. Adjust permissions using `chown -R oneadmin:oneadmin /var/lib/one/datastores/<datastore_id>` command.

## Configure LVM Thin with ONTAP NVMe/FC for OpenNebula

Configure Logical Volume Manager (LVM) for shared datastore across OpenNebula hosts using NVMe over Fibre Channel protocol with NetApp ONTAP. This configuration provides high-performance block-level storage access with low latency using the modern NVMe protocol.

### Initial virtualization administrator tasks

Complete these initial tasks to prepare OpenNebula hosts for NVMe/FC connectivity and collect the necessary information for the storage administrator.

1. Verify two HBA interfaces are available.
2. On every OpenNebula host in the cluster, run the following commands to collect the WWPN information and verify the `nvme-cli` package is installed.

#### Debian/Ubuntu

```
apt update
apt install nvme-cli
cat /sys/class/fc_host/host*/port_name
nvme show-hostnqn
```

#### RHEL/AlmaLinux

```
dnf update
dnf install nvme-cli
cat /sys/class/fc_host/host*/port_name
nvme show-hostnqn
```

3. Provide the collected host NQN and WWPN information to the storage administrator and request an NVMe namespace of the required size. WWPNs are needed for fabric zoning. Provide those info to the administrator who takes care of fabric zoning.

### Storage administrator tasks

If you are new to ONTAP, use System Manager for a better experience.

1. Ensure the SVM is available with NVMe protocol enabled. Refer to [NVMe tasks on ONTAP 9 documentation](#).

2. Ensure that two LIFs per controller are created and dedicated for NVMe/FC. Gather the WWPN addresses for the NVMe/FC LIFs created and provide them to the administrator who takes care of fabric zoning.
3. Create the NVMe namespace.
4. Create the subsystem and assign host NQNs.
5. Ensure Anti-Ransomware protection is enabled on the security tab.
6. Notify the virtualization administrator that the NVMe namespace is created.

### Final virtualization administrator tasks

Complete these tasks to configure the NVMe namespace as shared LVM storage in OpenNebula.

1. Navigate to a shell on each OpenNebula host in the cluster and verify the new namespace is visible.
2. Check namespace details.

```
nvme list
```

3. Inspect and collect device details.

```
nvme list  
nvme netapp ontapdevices  
nvme list-subsys  
lsblk -N
```

4. SSH to one of the frontend server and create a configuration file based on desired Datastore type. For complete attribute list, refer [OpenNebula LVM documentation](#). Sample files are shown below:

## Backup

### 1. For Restic,

```
$cat nvmeffc-restic.conf
NAME = "Backup-Restic-NVMEFC"
TYPE = "BACKUP_DS"

DS_MAD = "restic"
TM_MAD = "-"

RESTIC_PASSWORD = "<restic_password>"
RESTIC_SFTP_SERVER = "<backup server>"
```

### 1. For Rsync,

```
$cat nvmeffc-rsync.conf
NAME = "Backup-Rsync-NVMEFC"
TYPE = "BACKUP_DS"

DS_MAD = "rsync"
TM_MAD = "-"

RSYNC_USER = "<rsync_user>"
RSYNC_HOST = "<backup server>"
```

## File

```
$cat nvmeffc-kernel.conf
NAME = "File-Kernel-NVMEFC"
TYPE = "FILE_DS"
DS_MAD = "fs"
TM_MAD = "local"
SAFE_DIRS = "/var/tmp/files"
```

## Image

```
$cat nvmeffc-image.conf
NAME = "Image-NVMEFC01"
TYPE = "IMAGE_DS"
DS_MAD = "fs"
TM_MAD = "fs_lvm_ssh"
DISK_TYPE = "block"
LVM_THIN_ENABLE = "yes"
```

## System

```
$cat nvme-fc-system.conf
NAME = "System-NVMEFC02"
TYPE = "SYSTEM_DS"
TM_MAD = "fs_lvm_ssh"
DISK_TYPE = "block"
BRIDGE_LIST = "<space-separated list of OpenNebula hosts>" # If NVMe
namespace not presented to frontend hosts
LVM_THIN_ENABLE = "yes"
```

5. Execute `onedatastore create <configuration file>`. Note the datastore ID returned after creation.

```
onedatastore create nvme-fc-system.conf
ID: 108
```

6. Create volume group on the NVMe namespace using `vgcreate <vg_name> <nvme_device>` command. For Image datastores, the volume group name can be named anything desired. For System datastores, the volume group name must be of format `vg-one-<datastore id>`. This is required for OpenNebula to identify the correct volume group for system datastores. Proceed with following steps if you are creating Backup/File/Image datastore. For system datastores, stop here.
7. Create logical volume thin pool using `lvcreate -l 100%FREE -n <logical volume name> <volume group name>` command. For System datastores, OpenNebula automatically creates the LVM thin pool when required.
8. Create filesystem on the logical volume using `mkfs.ext4 /dev/<volume group>/<logical volume>` command. System datastores do not require filesystem creation.
9. Update `/etc/fstab` or automount configuration to mount the datastore with desired mount options. Assuming the default datastore location as `/var/lib/one/datastores`. Can be validated with `onedatastore show <datastore id>`. If not check the `DATASTORE_LOCATION` parameter in `/etc/one/oned.conf`. Ensure the `<datastore id>` folder exists under the datastores location. Sample entries are shown below:

### Using /etc/fstab

```
/dev/<vg name>/<logical volume>
/var/lib/one/datastores/<datastore_id> ext4 _netdev,noauto,x-
systemd.automount,nofail 0 2
```

### Using automount

```
/var/lib/one/datastores/<datastore_id> -fstype
=ext4,_netdev,noauto,x-systemd.automount,nofail,rw :/dev/<vg
name>/<logical volume>
```

10. Mount the datastore using `mount -a` or `systemctl reload autofs` command.
11. Verify the datastore is mounted with `mount` command and verify the datastore capacity with `onedatastore show <datastore_id>` command.
12. Ensure `oneadmin` user and group own the datastore folder. Adjust permissions using `chown -R oneadmin:oneadmin /var/lib/one/datastores/<datastore_id>` command.

## Configure LVM Thin with ONTAP NVMe/TCP for OpenNebula

Configure Logical Volume Manager (LVM) datastore for shared storage across OpenNebula hosts using NVMe over TCP protocol with NetApp ONTAP. This configuration provides high-performance block-level storage access over standard Ethernet networks using the modern NVMe protocol.

### Initial virtualization administrator tasks

Complete these initial tasks to prepare OpenNebula hosts for NVMe/TCP connectivity and collect the necessary information for the storage administrator.

1. Verify two Linux VLAN interfaces are available.
2. On every OpenNebula host, run the following command to collect the host initiator information.

```
nvme show-hostnqn
```

3. Provide the collected host NQN information along with hostname to the storage administrator and request an NVMe namespace of the required size.

### Storage administrator tasks

If you are new to ONTAP, use System Manager for a better experience.

1. Ensure the SVM is available with NVMe protocol enabled. Refer to [NVMe tasks on ONTAP 9 documentation](#).
2. Create the NVMe namespace.
3. Create the subsystem and assign to host NQNs. Create one subsystem for all OpenNebula hosts in a cluster and also to Frontend servers. Frontend servers are optional in subsystem assignment but required for Image datastores.
4. Ensure Anti-Ransomware protection is enabled on the security tab.
5. Notify the virtualization administrator that the NVMe namespace is created.

### Final virtualization administrator tasks

Complete these tasks to configure the NVMe namespace as shared LVM datastore in OpenNebula.

1. Navigate to a shell on each OpenNebula host in the cluster and create the `/etc/nvme/discovery.conf` file. Update the content specific to your environment.

```
root@onehost01:~# cat /etc/nvme/discovery.conf
# Used for extracting default parameters for discovery
#
# Example:
# --transport=<trtype> --traddr=<traddr> --trsvcid=<trsvcid> --host
-traddr=<host-traddr> --host-iface=<host-iface>

-t tcp -l 1800 -a 172.21.118.153
-t tcp -l 1800 -a 172.21.118.154
-t tcp -l 1800 -a 172.21.119.153
-t tcp -l 1800 -a 172.21.119.154
```

2. Log in to the NVMe subsystem.

```
nvme connect-all
```

3. To persist the NVMe namespace across reboots, enable nvme-autoconnect service.

```
systemctl enable nvme-autoconnect
```

4. Inspect and collect device details.

```
nvme list
nvme netapp ontapdevices
nvme list-subsys
lsblk -N
```

5. SSH to one of the frontend server and create a configuration file based on desired Datastore type. For complete attribute list, refer [OpenNebula LVM documentation](#). Sample files are shown below:

## Backup

### 1. For Restic,

```
$cat nvmetcp-restic.conf
NAME = "Backup-Restic-NVME TCP"
TYPE = "BACKUP_DS"

DS_MAD = "restic"
TM_MAD = "-"

RESTIC_PASSWORD = "<restic_password>"
RESTIC_SFTP_SERVER = "<backup server>"
```

### 1. For Rsync,

```
$cat nvmetcp-rsync.conf
NAME = "Backup-Rsync-NVME TCP"
TYPE = "BACKUP_DS"

DS_MAD = "rsync"
TM_MAD = "-"

RSYNC_USER = "<rsync_user>"
RSYNC_HOST = "<backup server>"
```

## File

```
$cat nvmetcp-kernel.conf
NAME = "File-Kernel-NVME TCP"
TYPE = "FILE_DS"
DS_MAD = "fs"
TM_MAD = "local"
SAFE_DIRS = "/var/tmp/files"
```

## Image

```
$cat nvmetcp-image.conf
NAME = "Image-NVME TCP01"
TYPE = "IMAGE_DS"
DS_MAD = "fs"
TM_MAD = "fs_lvm_ssh"
DISK_TYPE = "block"
LVM_THIN_ENABLE = "yes"
```

## System

```
$cat nvmetcp-system.conf
NAME = "System-NVME TCP02"
TYPE = "SYSTEM_DS"
TM_MAD = "fs_lvm_ssh"
DISK_TYPE = "block"
BRIDGE_LIST = "<space-separated list of OpenNebula hosts>" # If NVMe
namespace not presented to frontend hosts
LVM_THIN_ENABLE = "yes"
```

6. Execute `onedatastore create <configuration file>`. Note the datastore ID returned after creation.

```
onedatastore create nvmetcp-system.conf
ID: 109
```

7. Create volume group on the NVMe namespace using `vgcreate <vg_name> <nvme_device>` command. For Image datastores, the volume group name can be named anything desired. For System datastores, the volume group name must be of format `vg-one-<datastore id>`. This is required for OpenNebula to identify the correct volume group for system datastores. Proceed with following steps if you are creating Backup/File/Image datastore. For system datastores, stop here.
8. Create logical volume thin pool using `lvcreate -l 100%FREE -n <logical volume name> <volume group name>` command. For System datastores, OpenNebula automatically creates the LVM thin pool when required.
9. Create filesystem on the logical volume using `mkfs.ext4 /dev/<volume group>/<logical volume>` command. System datastores do not require filesystem creation.
10. Update `/etc/fstab` or automount configuration to mount the datastore with desired mount options. Assuming the default datastore location as `/var/lib/one/datastores`. Can be validated with `onedatastore show <datastore_id>`. If not check the `DATASTORE_LOCATION` parameter in `/etc/one/oned.conf`. Ensure the `<datastore_id>` folder exists under the datastores location. Sample entries are shown below:

### Using `/etc/fstab`

```
/dev/<vg name>/<logical volume>
/var/lib/one/datastores/<datastore_id> ext4 _netdev,noauto,x-
systemd.automount,nofail 0 2
```

### Using automount

```
/var/lib/one/datastores/<datastore_id> -fstype
=ext4,_netdev,noauto,x-systemd.automount,nofail,rw :/dev/<vg
name>/<logical volume>
```

11. Mount the datastore using `mount -a` or `systemctl reload autofs` command.
12. Verify the datastore is mounted with `mount` command and verify the datastore capacity with `onedatastore show <datastore_id>` command.
13. Ensure `oneadmin` user and group own the datastore folder. Adjust permissions using `chown -R oneadmin:oneadmin /var/lib/one/datastores/<datastore_id>` command.

## Copyright information

Copyright © 2026 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.

LIMITED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (b)(3) of the Rights in Technical Data -Noncommercial Items at DFARS 252.227-7013 (FEB 2014) and FAR 52.227-19 (DEC 2007).

Data contained herein pertains to a commercial product and/or commercial service (as defined in FAR 2.101) and is proprietary to NetApp, Inc. All NetApp technical data and computer software provided under this Agreement is commercial in nature and developed solely at private expense. The U.S. Government has a non-exclusive, non-transferrable, nonsublicensable, worldwide, limited irrevocable license to use the Data only in connection with and in support of the U.S. Government contract under which the Data was delivered. Except as provided herein, the Data may not be used, disclosed, reproduced, modified, performed, or displayed without the prior written approval of NetApp, Inc. United States Government license rights for the Department of Defense are limited to those rights identified in DFARS clause 252.227-7015(b) (FEB 2014).

## 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.