Dell PowerStore: Oracle Database Best Practices

October 2022

H18200.5

White Paper

Abstract

This document provides best practices for deploying Oracle databases with Dell PowerStore.

Dell Technologies

Copyright

The information in this publication is provided as is. Dell Inc. makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose.

Use, copying, and distribution of any software described in this publication requires an applicable software license.

Dell Inc. believes the information in this document is accurate as of its publication date. The information is subject to change without notice.

Contents

Executive summary

Introduction

This paper delivers guidance for using Dell PowerStore T model arrays in an Oracle 19c Oracle Standalone environment. This document also includes guidance for Oracle Real Application Clusters (RAC) database environments on Linux operation systems.

This paper was developed using the PowerStore 5200T model array, but the information is also applicable to PowerStore 500T, PowerStore x000T and PowerStore x200T models. The Linux operating system that was used in this paper was Oracle Linux (OL) 8.5 with the Unbreakable Enterprise Kernel (UEK). The general operating system content is applicable to OL7 UEK and Red Hat Enterprise Linux 7 versions, but commands and specific processes might be different.

Some recommendations in the paper might not apply to all environments. For questions about the applicability of these guidelines in your environment, contact your Dell Technologies representative.

Audience

This document is intended for IT administrators, storage architects, partners, and Dell Technologies employees. This audience also includes individuals who evaluate, acquire, manage, operate, or design a Dell networked storage environment using PowerStore systems.

Revisions

Table 1. Revisions

Date	Description
April 2020	Initial release: PowerStoreOS 1.0
July 2020	Minor edits
April 2021	Updates for PowerStoreOS 2.0
November 2021	Template update
June 2022	Updates for PowerStoreOS 3.0
October 2022	Added PowerStore File and Oracle dNFS content

Note: This document might contain language from third-party content that is not under Dell Technologies' control and is not consistent with current guidelines for Dell Technologies' own content. When such third-party content is updated by the relevant third parties, this document will be revised accordingly.

Terminology

The following table provides definitions for some terms that are used in this document:

Table 2. Terminology

Term	Definition
2-port card	PowerStore 1000 and higher models require two 100 GbE optical QSFP port card to support NVMe Expansion Shelves.

Term	Definition
4-port card	Card for each node that provides four ports. Options include 25 GbE optical and 10 GbE Base-T. PowerStore 500 requires the 25 GbE card to support NVMe Expansion Shelf. The 4-port mezzanine card supports federate storage (that is, cluster multiple PowerStore appliances) and nonblock storage. PowerStore 3.0 supports a new 100 GbE 4-port card in I/O module for slot 0 in PowerStore x000 and x200 models.
Appliance	The appliance is the base enclosure and any PowerStore system.
Base enclosure	With PowerStore 3.0, the base enclosure contains 25 NVMe drive slots. PowerStore 3.0 can also scale outside the base enclosure with NVMe expansion shelves. For more information, see *Expansion enclosure* in this table. With PowerStore 2.0, NVMe drives cannot scale outside the base enclosure. If PowerStore 2.0 needs to be scaled out, one or more SAS expansion shelves must be used.
Cluster	A cluster is a group of one to four PowerStore appliances. PowerStore T model clusters are expandable by adding more appliances (up to four total).
Expansion enclosure	Enclosures that can be attached to a base enclosure to provide additional storage in the form of either NVMe or SAS drives. A maximum of three 25-drive expansion shelves are supported. Depending on the PowerStore model, the expansion enclosure must either be a NVMe expansion enclosure, or a SAS expansion enclosure. Mixing expansion enclosure types is not supported. PowerStore 3.0 models x200 support only NVMe expansion shelves. PowerStore 2.0 models x000 support only SAS expansion shelves.
Embedded module	With PowerStore 3.0 models x200, a new and improved embedded module v2 is embedded on each node providing: • Management and service ports • Two NVMe expansion ports (100 GbE, QSFP BE-ports) • Four ports for front-end connectivity. PowerStore 3.0 model 500 does not have an embedded module. It uses a 4-port 25 GbE optical mezzanine card. Ports 0 and 1 are used for the system bond. The bonded ports are used for federation and NAS. Ports 2 and 3 of the four mezzanine ports are required and reserved for back-end connectivity to support NVMe expansion shelves. With PowerStore 2.0 models x000, the embedded module v1 is needed for SAS expansion (SAS BE-ports) and only available and required on PowerStore x000 modules. For PowerStore 2.0 model 500, the 4-port 25 GbE optical mezzanine card on embedded module v1 is not required. It is only required if the system will be configured for: • Unified storage • Federated storage—that is, clustering of multiple appliances
Node	The component within the base enclosure that contains processors and memory. Each appliance consists of two nodes.
PowerStore Manager	PowerStore Manager is a web-based user interface (UI) for storage management.

Term	Definition
NVMe over Fibre Channel (NVMe/FC)	NVMe/FC allow hosts to access storage systems across a network fabric with the NVMe protocol using Fibre Channel as the underlying transport. To use NVMe/FC, the host operating system must support NVMe protocols. OL 8 supports the NVMe protocol and was used for this paper.
NVMe over TCP (NVMe/TCP)	NVMe/TCP allow hosts to access storage systems across a TCP fabric with the NVMe protocol using TCP protocol as the underlying transport. NVMe/TCP connectivity can be enabled on the same ports as iSCSI, or on different Ethernet ports. To use NVMe/TCP, the host operating system must support NVMe protocols.

We value your feedback

Dell Technologies and the authors of this document welcome your feedback on this document. Contact the Dell Technologies team by email.

Author: Mark Tomczik

Note: For links to other documentation for this topic, see the PowerStore Info Hub.

Introduction

PowerStore overview

PowerStore achieves new levels of operational simplicity and agility. It uses a container-based microservices architecture, advanced storage technologies, and integrated machine learning to unlock the power of your data. PowerStore is a versatile platform with a performance-centric design that delivers multidimensional scale, always-on data reduction, and support for next-generation media.

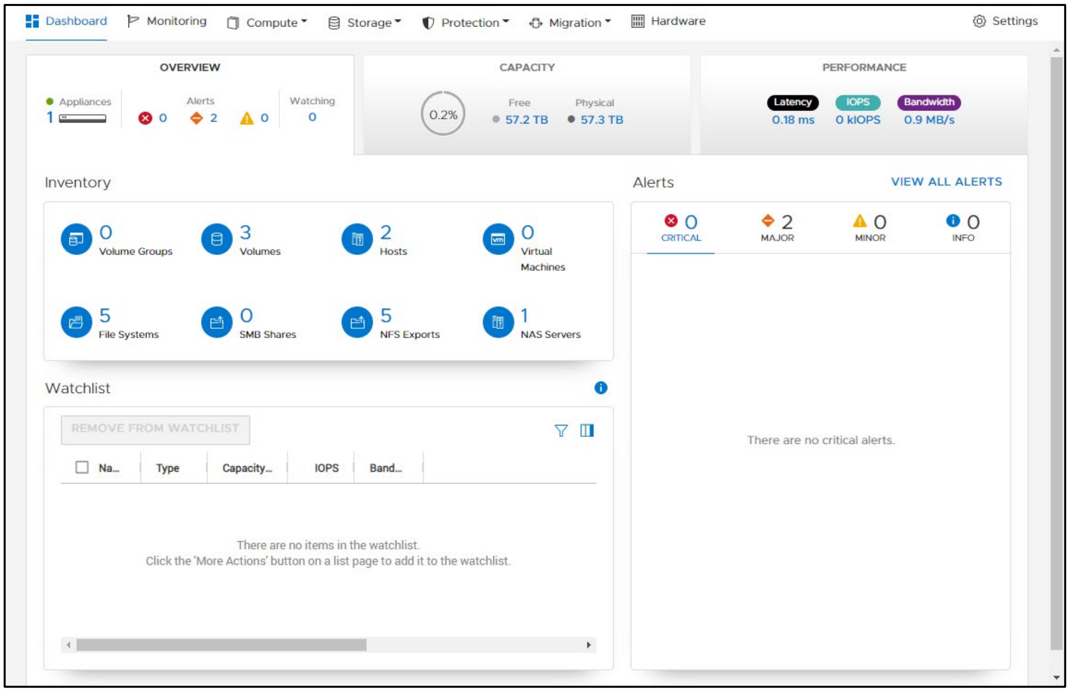
PowerStore brings the simplicity of public cloud to on-premises infrastructure, streamlining operations with an integrated machine-learning engine and seamless automation. It also offers predictive analytics to easily monitor, analyze, and troubleshoot the environment. PowerStore is highly adaptable, providing the flexibility to host specialized workloads directly on the appliance and modernize infrastructure without disruption. It also offers investment protection through flexible payment solutions and data-in-place upgrades.

With all the capabilities of PowerStore mentioned above, PowerStore offers a robust and flexible enterprise storage option for Oracle solutions.

Management tools

The PowerStore Manager UI is the primary management tool for PowerStore configuration and administration (see the following figure). Some benefits of PowerStore Manager are:

- No client installation is required.
- It is HTML5-based so there is no Java requirement).
- Support for multiple web browsers. Mozilla Firefox was used for this paper.



Appliances 1; Alerts 0, 2, 0; Watching 0

0.2%; Free 57.2 TB; Physical 57.3 TB

Latency 0.18 ms; IOPS 0 kIOPS; Bandwidth 0.9 MB/s

0 Volume Groups; 3 Volumes; 2 Hosts; 0 Virtual Machines; 5 File Systems; 0 SMB Shares; 5 NFS Exports; 1 NAS Servers

0 CRITICAL; 2 MAJOR; 0 MINOR; 0 INFO

There are no critical alerts.

There are no items in the watchlist.
Click the 'More Actions' button on a list page to add it to the watchlist.

Figure 1. PowerStore Manager

The PowerStore platform also has a REST API available for administrators to automate management tasks.

Models

All PowerStore models are 2U rack-mountable enclosures, and are configured at the factory in one of two base-model configurations:

- **PowerStore T models**: Unified (file and block) or block-optimized (block only)
- **PowerStore X models**: VMware ESXi based hypervisor for storage and guest workloads

With the initial PowerStore release, the following models were available:

- PowerStore 1000, 8 core @ 1.8 GHz with 192 GB memory
- PowerStore 3000, 12 core @ 2.1 GHz with 384 GB memory
- PowerStore 5000, 16 core @ 2.1 GHz with 576 GB memory
- PowerStore 7000, 20 core @ 2.4 GHz with 768 GB memory
- PowerStore 9000, 28 core @ 2.1 GHz with 1280 GB memory

Four additional models with dual CPU sockets are now available:

- PowerStore 1200, 10 core @ 2.4 GHz with 192 GB memory
- PowerStore 3200, 16 core @ 2.1 GHz with 384 GB memory
- PowerStore 5200, 24 core @ 2.2 GHz with 576 GB memory
- PowerStore 9200, 28 core @ 2.2 GHz with 1280 GB memory

PowerStore also offers the PowerStore 500, a single CPU socket, 12 core @ 2.2 GHz with 96 GB memory.

All PowerStore 3.0 models use UEFI partitioning and provide secure boot.

This paper discusses PowerStore T models with Oracle Standalone single instance and Oracle RAC databases.

PowerStore cluster

A PowerStore cluster consists of one to four appliances.

A PowerStore 5200T model single-appliance cluster (see the following figure) was used for this paper.

NVMe SSD Drives

NVMe drives provides lower latency and higher bandwidth in throughput when compared to SAS drives.

With PowerStore 2.0, NVMe drives were only available in the base enclosure. When scaling-out PowerStore was a requirement, SAS expansion shelves were needed, with a maximum of three 25-drive SAS expansion shelves.

Starting with PowerStore 3.0, only NVMe expansion shelves are available for scaling-out PowerStore. The NVMe expansion shelves eliminate the SAS-to-NVMe performance bottle neck when SAS expansion shelves were used in scaling PowerStore 2.0. NVMe expansion shelves provide a complete end-to-end NVMe model for improved performance.

Expansion shelves in a PowerStore appliance must either be all NVMe or SAS. Mixing NVMe and SAS expansion shelves in a PowerStore appliance is not supported. NVMe SCM SSDs are not supported in NVMe expansion shelves.

PowerStore 3.0 prioritizes the NVMe drives in the base enclosure as the first choice for meta data, providing the best possible performance for internal meta data. NVMe drives can also be:

- Added to NVMe expansion shelves one drive at a time.
- Moved between base enclosure and NVMe expansion shelf, and
- Moved between different NVMe expansion shelves

For the PowerStore 500, a 4-port 25 GbE optical mezzanine card is required to support NVMe expansion shelves using reserved ports 3 and 4.

For the new PowerStore 3.0 models, the embedded Module v2 comes with the system. A 2-port 100 GbE card on the embedded Module v2 is optional and is only required if NVMe expansion shelves will be used.

Embedded Module v1 only ships with PowerStore x000 models and is required for SAS expansion shelves.

When deploying PowerStore for large, critical Oracle Enterprise solutions demanding greater performance requirements, Dell Technologies recommends using PowerStore models x200 as they support NVMe expansion shelves.

Documentation and support

Online support, context-sensitive help, and general support are provided through PowerStore Manager which is the primary reference material for optimal configuration of PowerStore. This white paper provides additional guidance. For supplemental information, see References.

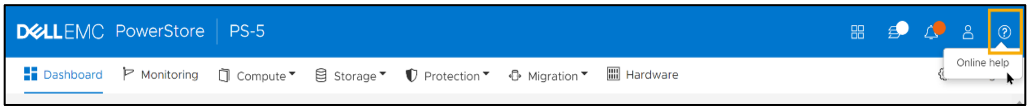


Figure 2. PowerStore support and documentation

Oracle product overview

According to DB-Engines, Oracle was the most popular relational database system (RDBMS) in April 2021. Oracle provides a way to write and read data in tabular form though a structured query language (SQL). It has a scalable relational database architecture and is the main application for data access and storage in many data centers.

Oracle databases are typically installed as either a single-instance Standalone database, or Oracle RAC database. For this paper, a RAC one-node environment was used. However, the content in this paper is also applicable to a single-instance Standalone database.

With single instance, Oracle databases run on a single host and are simple to install. With Oracle RAC, an Oracle database runs on one or more hosts. Some benefits of Oracle RAC deployments include:

- Expanded scalability for performance by adding hosts
- High availability with host isolation so host failure does not impact the application residing in the database

Prerequisites

This document is intended for readers having prior experience with or training in the following areas:

- Dell PowerStore Host Configuration Guide
- Linux operating environment
- Native Linux multipath software
- Oracle Automatic Storage Management (ASM)
- Oracle Standalone environments
- Oracle RAC environments
- Fibre Channel (FC) and Ethernet network administration

For PowerStore documentation, see www.dell.com/powerstoredocs.

PowerStore sizing

Overview

Regardless of the underlying storage array, understand the design of the entire application stack and requirements before deployment. This knowledge helps ensure that the PowerStore array is properly sized to deliver the expected performance and capacity. In addition to PowerStore sizing, changes to the infrastructure components such as the storage fabric might also be required. If the application is new, determine application design factors and expected performance metrics before sizing the array and supporting infrastructure.

- **Latency**: The amount of time that an I/O operation takes to complete. High latencies typically indicate an I/O bottleneck.
- **IOPS**: The number of reads and writes occurring each second. IOPS is key for determining the number of required disks in an array while maintaining accepted response times. If the array uses SSDs, the array typically provides enough IOPS once capacity, and throughput are met. IOPS is a key metric used for designing OLTP databases.
- **Throughput**: The amount of data in bytes per second transferred between the server and storage array. Throughput is primarily used to define the path between the server and array and the number of required drives. A few SSDs can often meet IOPS requirements but might not meet throughput requirements. Throughput can be calculated as follows using IOPS and the average I/O size: Throughput MBs = IOPS x I/O size.

Configure I/O for bandwidth and not capacity

PowerStore configurations for a database should be chosen based on I/O bandwidth or throughput, and not necessarily storage capacity.

Stripe far and wide

The guiding principle in configuring an I/O system for a database is to maximize I/O bandwidth by having multiple disks and channels accessing the data. To maximize I/O bandwidth, stripe the database files. The goal is to ensure that each Oracle tablespace is striped across many disks so data can be accessed with the highest possible I/O bandwidth. When using PowerStore arrays, striping is accomplished automatically at the storage level. Oracle ASM also provides stripping at the application level. Oracle ASM is Oracle's recommended storage solution.

OLTP and OLAP/DSS workloads

OLTP systems usually support predefined operations on specific data, and their workloads generally have small, random I/Os for rapidly changing data. As such, PowerStore arrays should be primarily sized based on the number of IOPS for OLTP systems.

Data warehouses are designed to accommodate queries, OLAP, DSS, and ETL processing. Their workloads generally have large sequential reads. Storage solutions servicing workloads of this type are predominantly sized based on I/O bandwidth or throughput and not capacity or IOPS. When sizing for throughput, the expected throughput of each component in the I/O path (CPU cores, HBAs, FC connections, FC switches and ports, disk controllers, and disks) must be known. The entire I/O path needs to be sized appropriately to guarantee balanced system resources. Appropriately sized I/O paths maximize I/O throughput and allow the system to grow without compromising the I/O bandwidth. Sometimes, throughput can easily be exceeded when using SSDs. Understand the characteristics of SSDs and the expected I/O pattern of Oracle.

General sizing recommendations

When sizing an array:

- Assume that all I/O will be random. Assuming random I/O yields best results.
- Before releasing any storage system to production, use Dell Live Optics (formally DPACK) on a simulated production system during at least a 24-hour period that includes the peak workload. The simulation helps define the I/O requirements. It might also be possible to use Iometer to simulate the production system. After production begins, repeat the analysis on the production system.
- Understand what level of ASM disk group redundancy (external, normal, high) is being considered. Using ASM external redundancy with PowerStore is recommended unless Extended Distance Oracle RAC Clusters are used. For Oracle RAC Extended Distance Clusters, use either normal or high ASM redundancy. To avoid the complexity of configuring Oracle RAC Extended distance clusters, consider using of PowerStore metro node Metro instead. For more information, see References.
- Have a good understanding of the application workloads (OLTP, OLAP, or hybrid).
- NVMe NVRAM drives are reserved for system write cache in PowerStore 1200T through PowerStore 9200T models. User and system metadata is written to the other drive types (NVMe SCM, NVMe SSD, SAS SSD). NVMe drives provide lower latency and higher bandwidth than SAS SSD, so are a great choice for an Oracle solution.
- Understand the required performance metrics of the servers connected to the PowerStore array. The IOPS and throughput help determine the number of disks required in the array, and throughput helps define the paths between the PowerStore array and server.

Test the I/O system before implementation

Test I/O bandwidth and IOPS on dedicated components of the I/O path to ensure that expected performance is achieved before creating a database. On most operating systems, testing I/O bandwidth and IOPS can be done using one of the following methods:

- Test acceptable response times with large amounts of data being loaded within a window of time.
- Test acceptable response times with large amounts of data with queries during peak production times.
- Use throughput numbers and experience from an existing identical-configured environment.

Testing could be performed with simple scripts to measure the performance of reading and writing large test files that perform large block sequential I/Os. The tests could be performed using Linux command dd or Oracle ORION. Two large test files should be used with each volume owned by a different PowerStore node. The test verifies that all I/O paths are fully functional. If the resulting throughput matches the expected throughput for the components in the I/O path, the paths are set up correctly.

Note: Exercise caution if the test is run on a live system because the test could cause significant performance issues.

To help define I/O requirements, Dell Technologies recommends using Dell Live Optics on a simulated production system during at least a 24-hour period that includes the peak workload. If it is not possible to simulate the production system, using Iometer to simulate the production system might be possible. For other available testing tools, see the following table:

Table 3. Performance analysis tools

Category	Tool	Vendor
I/O subsystem	Dell Live Optics (formally DPACK)	Dell Technologies
	Oracle ORION calibration Tool	Oracle
	Iometer	Iometer Org
	fio	Freecode and SourceForge
	ioping	Free Software Foundation
	dd	Linux operating system
RDBMS	SLOB	Kevin Closson
	Oracle PL/SQL package DBMS_RESOURCE_MANAGER.CALIBRATE_IO	Oracle
Transactional	Benchmark Factory	Quest
	HammerDB	Open Source
	SwingBench	Dominicgiles
	Oracle Real Application testing	Oracle

The performance analysis tools can:

- Configure block size
- Specify number of outstanding requests
- Configure test file size
- Configure number of threads
- Support multiple test files
- Write blocks of nonzeros

If it is not possible to run Dell Live Optics or another testing tool, test the paths between the server and PowerStore. The tests should perform large block sequential reads using small files (one file per volume, per PowerStore node). Tests should use multiple threads and use 512 KB blocks and a queue depth of 32 to saturate all paths. A successful test verifies that all paths are functioning and will yield the I/O potential of the array. If the throughput matches the expected throughput for the number of server HBA ports, the I/O paths between the PowerStore array and the server are configured correctly. If the test is run on a PowerStore array not dedicated to this test, it could cause significant performance issues. If smaller blocks are used for the test, I/O saturation rates might not be achievable. Small block tests might not verify that all paths are functioning and will yield the I/O potential of the array.

Repeat this test and validate the process on the production server after go-live to validate and establish a benchmark of initial performance metrics.

Once a design can deliver the expected throughput requirement, more disks can be added to the storage solution to meet capacity requirements. However, the converse is not necessarily true. If a design meets the expected capacity requirements, adding disks to the storage solution might not make the design meet the required throughput requirements. For example, because disk drive capacity is growing faster than disk I/O throughput rates, a situation can occur where fewer disks can store a large volume of data; however, the small number of disks cannot provide the same I/O throughput as a larger number of smaller disks.

After validating throughput of I/O paths between the PowerStore array and the server, and meeting capacity requirements, test the disk I/O capabilities for the designed workload of the PowerStore array. Successful tests validate that the storage design provides the required IOPS and throughput with acceptable latencies. The test must not exceed the designed capacity of the array, otherwise the test results will be misleading. If the test does exceed the designed capacity, reduce the number of test threads, outstanding I/Os, or both. Testing random I/O should be done with I/O sizes of 8 KB and 16 KB as a starting point and adjust from there. When testing sequential I/O, I/O sizes should be 8 KB, 16 KB, 64 KB, or larger.

This testing methodology assumes the guidelines mentioned in previous sections have been followed and modified according to business requirements.

The principle of stripe-far-and-wide needs to be used in tandem with data warehouses to increase the throughput potential of the I/O paths.

Plan for growth

A plan should exist for future growth of a database. There are many ways to handle growth, and the key consideration is to be able to grow the I/O system without compromising on the I/O bandwidth.

PowerStore storage pool

The PowerStore storage pool can be configured with a minimum of six drives in the base enclosure. Storage pools on PowerStore 1200T through 9200T models can be expanded up to 100 drives through expansion enclosures. The total disk capacity does not automatically guarantee disk performance. There must be enough drives to meet I/O and capacity demands. Base enclosure front-end ports can also be expanded.

If horizontal scaling is a design requirement, PowerStore models support a scale-out configuration with one to four appliances in a single cluster. Contact your Dell Technologies representatives for assistance with sizing a PowerStore solution.

Dynamic Resiliency Engine (DRE) and storage containers

PowerStore automatically manages the underlying storage for maximum performance and capacity, eliminating the need for administrators to configure the storage pool. Manually setting or configuring these options is unnecessary in PowerStore. The underlying drive configuration and management are automatic and require no management.

PowerStore installation best practices

Introduction

PowerStore T models support a unified (block and file) or block-optimized (block only) configuration. Either configuration can support an Oracle environment. For a multiappliance PowerStore T model cluster, only the first appliance in the cluster (the primary appliance) can be configured to support unified storage. Extra appliances added to create a multiappliance cluster are configured automatically as block-optimized during the initial configuration.

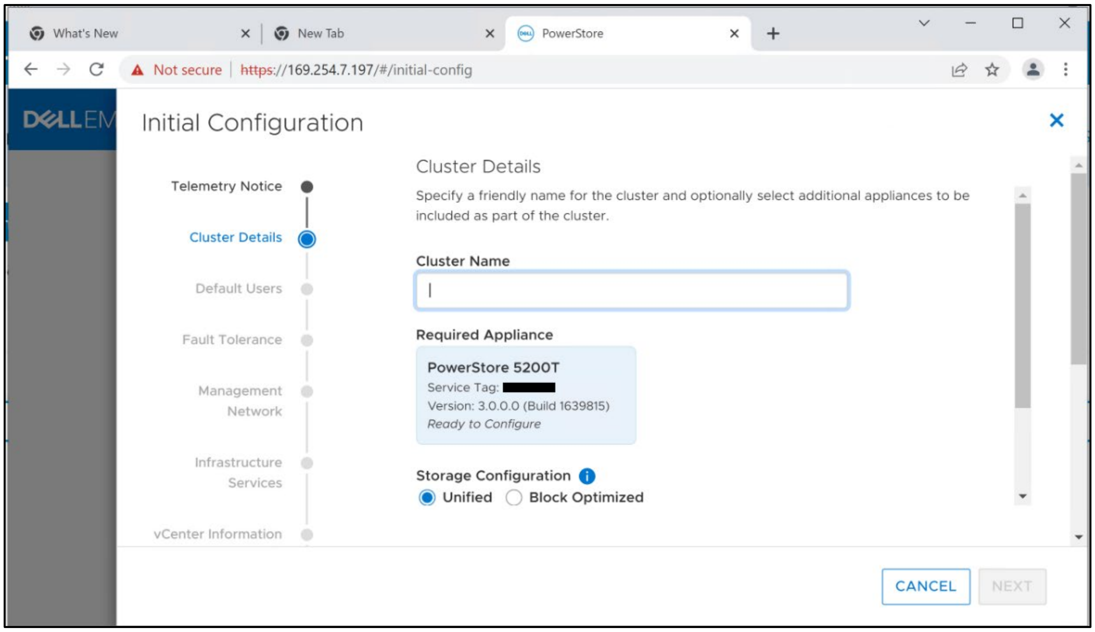
Unified or block-optimized

If file services (NAS) are needed or might be needed later, select **Unified** during initial configuration. With unified storage configurations, some PowerStore compute and storage resources on the appliances are reserved for NAS.

If file services will not be required, choose block-optimized during the initial configuration of the PowerStore system.

Once PowerStore T is configured to support a particular storage configuration (**Unified** or **Block Optimized**), a reinitialization is required to change the configuration to the other storage configuration.

When defining a cluster, specify the name and select the required storage configuration.



Initial Configuration

Cluster Details

Specify a friendly name for the cluster and optionally select additional appliances to be included as part of the cluster.

Cluster Name

Required Appliance

PowerStore 5200T
Service Tag: [redacted]
Version: 3.0.0.0 (Build 1639815)
Ready to Configure

Storage Configuration

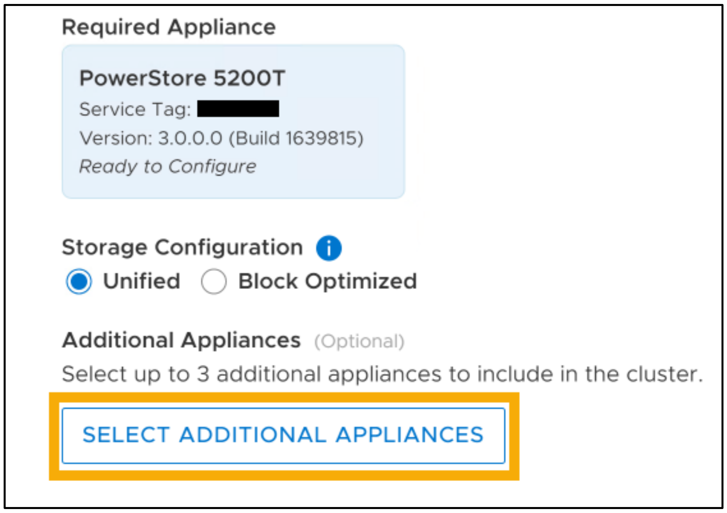
● Unified ○ Block Optimized

Figure 3. PowerStore Unified (block and file) configuration

Note: A PowerStore 5200T model appliance was deployed with unified configuration for this paper (see the preceding figure).

Single or multiple-appliance cluster

If horizontal scaling is required within the PowerStore T model cluster, during the initial configuration, edit the appliance section, as shown in the following figure. In a multiple-appliance cluster, only the first appliance (primary) can support unified storage. Extra appliances are automatically configured as block-optimized.



Required Appliance

PowerStore 5200T
Service Tag: [redacted]
Version: 3.0.0.0 (Build 1639815)
Ready to Configure

Storage Configuration
● Unified ○ Block Optimized

Additional Appliances (Optional)
Select up to 3 additional appliances to include in the cluster.

Figure 4. Selecting appliances for a cluster

PowerStore host objects

When configuring PowerStore for and Oracle environment, the physical database server must be configured as a virtual host object in PowerStore.

The process of creating a virtual host object in PowerStore is shown in PowerStore storage provisioning and configuration best practices.

Host and Linux configuration

Introduction

Oracle Standalone and RAC databases are commonly deployed on Linux operating systems. While most settings in Linux can remain at the defaults, some changes are recommended for stability and efficiency with PowerStore arrays. The following sections describe best practices when working with Linux operating systems on PowerStore storage systems hosting either Oracle Standalone or Oracle RAC databases. For more information about specific changes, see the *Dell PowerStore Host Configuration Guide* on www.dell.com/powerstoredocs.

For this paper, the Unbreakable Enterprise Kernel for Oracle Linux 8.5 was used.

If Oracle Direct NFS will be used, when selecting which slots should be used for PCIe interfaces, consider which CPU is controlling which slot. Dell Technologies recommends using a balanced number of slots controlled by both CPUs so that CPUs are balanced concerning PCIe slots. For more information, see *Dell PowerEdge R730 and R730xd Technical Guide*.

Server FC HBA driver settings: timeouts and queue depth

Queue depth is the number of disk transactions that are in flight between an initiator (HBA port on a Linux server) and a target (port on PowerStore appliance). The initiators are one or more FC or iSCSI ports on the host server which are paired with corresponding target ports of the same protocol type on PowerStore. Any given target port can be paired with multiple initiator ports. To address this issue, the initiator queue depth throttles the number of transactions that any given initiator can send to a target port from a host. This throttling helps to prevent the target ports from becoming flooded. When flooding happens, the transactions are queued, which causes higher latency and degraded performance for the affected workloads.

The default queue depth value of 32 might be adequate for Oracle applications, but other values might work too. These values should be determined as directed in Test the I/O system before implementation.

Often, changing the default queue depth is not necessary. However, changing the queue depth might improve performance in specific use cases. These changes should only be made, tested, and evaluated in a nonproduction environment before being moved to production.

For example, if a storage array is connected to a few Linux servers with large-block, sequential-read application workloads, increasing the queue depth might be beneficial. However, if the storage has many hosts competing for a few target ports, increasing the queue depth on a few hosts might overdrive the target ports. This result might negatively impact the performance of all connected hosts.

Increasing the queue depth can sometimes increase performance for specific workloads. If the queue depth is set too high, there is an increased risk of overdriving the target ports on PowerStore. Generally, if transactions are queued and performance is impacted, try increasing the queue depth. If this change results in saturation of the target ports, increase the number of front-end target ports on PowerStore. This action to spread out I/O can be an effective remediation.

Scanning for non-NVMe LUNs

After installing sg3_utils and lsscsi, scan for the PowerStore non-NVMe volumes with the following command:

```
/usr/bin/rescan-scsi-bus.sh -a
```

Querying WWNs using scsi_id command

To query the WWN on a Linux operating system, run the following commands against the device file.

Oracle Linux or Red Hat Enterprise Linux 6.x:

```
# /sbin/scsi_id --page=0x83 --whitelisted --device=<device>
```

Oracle Linux or Red Hat Enterprise Linux 7.x and 8.x:

```
# /usr/lib/udev/scsi_id --page=0x83 --whitelisted --
device=<device>
```

Querying WWNs using multipath command

If the system has Linux device-mapper-multipath software enabled, the multipath command displays the multipath device properties including the WWN. For example:

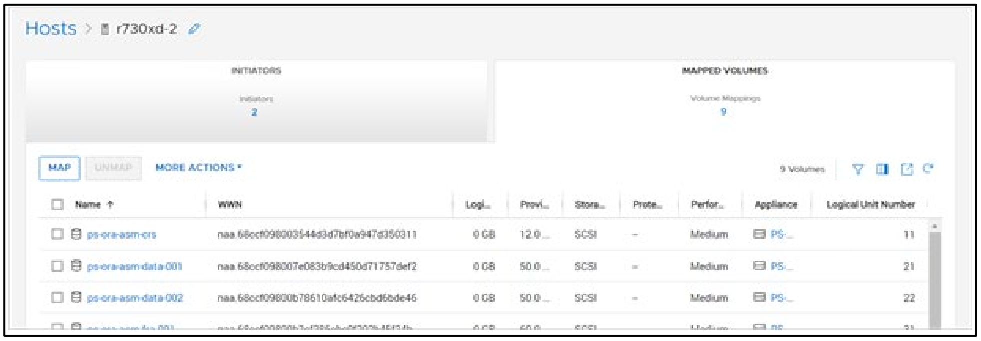
```
# multipath -ll
mpatha (36006016010e0420093a88859586140a5) dm-0 DellEMC
,PowerStore
```

Identifying volume WWNs and LUN IDs in PowerStore

To view the WWN and LUN ID information in PowerStore Manager, follow these steps:

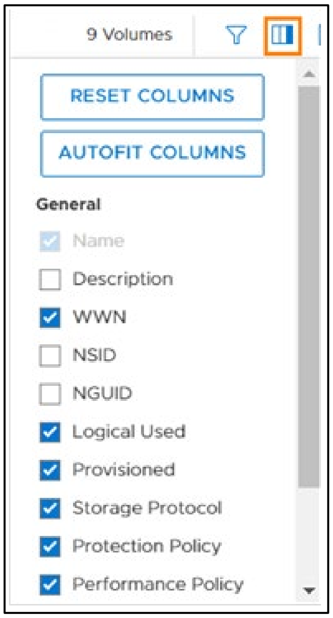
1. In PowerStore Manager, click **Compute** > **Hosts Information**.
2. Select the link of the host.
3. Select **MAPPED VOLUMES**.

   The WWN and Logical Unit Number (LUN ID) appear in the list of volumes that PowerStore Manager returns.



☐	Name ↑	WWN	Logi...	Provi...	Stora...	Prote...	Perfor...	Appliance	Logical Unit Number
☐	ps-ora-asm-crs	naa.68ccf0980035444d3d7bf0a947d350311	0 GB	12.0 ...	SCSI	–	Medium	PS-...	11
☐	ps-ora-asm-data-001	naa.68ccf098007e083b9cd450d71757def2	0 GB	50.0 ...	SCSI	–	Medium	PS-...	21
☐	ps-ora-asm-data-002	naa.68ccf09800b78610afc6426cbd6bde46	0 GB	50.0 ...	SCSI	–	Medium	PS-...	22

   If columns **WWN** and **Logical Unit Number** are not displayed, use the field selector to add them to the displayed data.



General

- ☑ Name
- ☐ Description
- ☑ WWN
- ☐ NSID
- ☐ NGUID
- ☑ Logical Used
- ☑ Provisioned
- ☑ Storage Protocol
- ☑ Protection Policy
- ☑ Performance Policy

Figure 5. Select WWN and Logical Unit Number columns

Discovery and connection of NVMe end-points

With OL 8 or Red Hat Enterprise Linux 8 and PowerStore 3.0, the discovery and connection of NVMe/FC end-points is performed automatically during the creation of the PowerStore host object. After the end-point connections have been made, any volume created in PowerStore and exposed to the host object will immediately be seen on the physical server. For more information, see PowerStore storage provisioning and configuration best practices.

Multipathing

Multipathing is a software solution that is implemented at the host-operating-system level. While multipathing is optional, it provides path redundancy, failover, and performance-enhancing capabilities. It is highly recommended to deploy the solution in any environment where availability and performance are critical.

Note: For this paper, native multipathing was used for NVMe/FC volumes and Udev was used to control device persistence.

The main benefits of using an MPIO solution are:

- Increased database availability through automatic path failover and failback
- Enhanced database I/O performance through automatic load balancing and capabilities for multiple parallel I/O paths
- Easier administration with the use of persistent, intuitive names for the storage devices across cluster nodes

Multipath software solutions

The native Linux multipath solution is supported and bundled with most popular Linux distributions in use today. Because the software is widely and readily available at no additional cost, many administrators prefer it to other third-party solutions.

Only one multipath software solution should be enabled on the host.

Connectivity guidelines

Array-to-host connectivity best practices include:

- Have at least two FC/iSCSI HBAs or ports to provide path redundancy.
- Connect the same port on both PowerStore nodes to the same switch. PowerStore matches the physical port assignment on both nodes.
- Use multiple switches to provide switch redundancy.

For details, see the *Dell Host Connectivity Guide for Linux*.

Configuration file

To simplify deployment, the native Linux multipath software comes with default settings for an extensive list of storage models from different vendors including PowerStore. The default settings allow the software to work with PowerStore immediately. However, these settings might not be optimal for all situations and should be reviewed and modified if necessary.

The multipath daemon configuration file must be created on newly installed systems. A basic template can be copied from **/usr/share/doc/device-mapper-multipath-<version>/multipath.conf** to **/etc/multipath.conf** as a starting point. Any settings that are not defined explicitly in /etc/multipath.conf would assume the default values. The full list of settings (explicitly set and default values) can be obtained running the following command. The default settings generally work without any issues.

```
# multipathd -k"show config"
```

For more information, see the *Dell Host Connectivity Guide for Linux*.

Creating device-mapper aliases

For ease of management, assign meaningful names (aliases) to the multipath devices. For example, create aliases that are based on the application type and environment that the device is in. The following snippet assigns an alias of `ora-asm-data-001` to the PowerStore LUN with WWN `368ccf098007e083b9cd450d71757def2`.

```
multipaths {
        multipath {
           wwid "368ccf098007e083b9cd450d71757def2"
           alias ora-asm-data-001
        }
}
```

For more information, see the *Dell Host Connectivity Guide for Linux*.

Asymmetric Logic Unit Access

PowerStore supports Asymmetric Logic Unit Access (ALUA) for host access. This feature allows the host operating system to differentiate between optimized paths and nonoptimized paths.

LUN partitioning

A PowerStore volume intended for ASM can be configured as a raw LUN or as a single partitioned LUN. If ASMLib is used to manage the LUN, the LUN must be configured as a single partitioned LUN. If ASMLib is not used, the LUN can either be configured as raw or as a single partitioned LUN by ASM. Choosing the type of LUN to use depends on the environment, infrastructure design, and daily operations.

Note: For this paper, raw unpartitioned LUNs were used for ASM.

If ASMLib or ASMFD are intended to manage ASM disks created from NVMe volumes, review the NVMe information available on Oracle Support.

Partition alignment

When partitioning a LUN, aligning the partition on the 1M boundary is recommended. Use either `fdisk` or `parted` to create the partition. However, only `parted` can create partitions larger than 2 TB.

Creating partition using parted

Before creating the partition, label the device as `GPT`. Then, specify the partition offset at 2048 sector (1M). The following command creates a single partition that takes up the entire LUN. Once the partition is created, use the partition file /dev/mapper/ora-asm-data-001p1 to create the ASMLib volume.

```
# parted /dev/mapper/ora-asm-data-001
GNU Parted 3.1
Using /dev/mapper/ora-asm-data-001
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) mklabel gpt
(parted) quit
Information: You may need to update /etc/fstab.

# parted /dev/mapper/ora-asm-data-001 mkpart primary 2048s 100%
```

Partitioned devices and file systems

When creating a file system, create the file system on a properly aligned partitioned device.

I/O scheduler for Oracle ASM devices

Oracle recommends using the deadline I/O scheduler for the best performance of Oracle ASM. For Oracle Linux, the deadline I/O scheduler is enabled by default in Oracle Unbreakable Enterprise Kernel. For other Linux operating systems, verify the I/O scheduler and make necessary updates if necessary.

Verify the I/O schedule by running the following commands:

```
# egrep "*" /sys/block/sd*/queue/scheduler
/sys/block/sdaa/queue/scheduler:noop [deadline] cfq
/sys/block/sdab/queue/scheduler:noop [deadline] cfq
/sys/block/sdac/queue/scheduler:noop [deadline] cfq
/sys/block/sdad/queue/scheduler:noop [deadline] cfq
/sys/block/sdae/queue/scheduler:noop [deadline] cfq
/sys/block/sdaf/queue/scheduler:noop [deadline] cfq
```

To set the I/O schedule persistently, create a Udev rule that updates the devices.

The following example shows setting the deadline I/O scheduler on all /dev/sd* devices. The rule is appended to the 99-oracle-asmdevices.rule file.

```
# cat /etc/udev/rules.d/99-oracle-asmdevices.rules
ACTION=="add|change", KERNEL=="sd*", RUN+="/bin/sh -c '/bin/echo
deadline > /sys$env{DEVPATH}/queue/scheduler'"

# udevadm control --reload-rules
# udevadm trigger
```

UDEV rules for LUNs

If ASMLib or ASMFD are not used to manage ASM disk persistence, Udev must be used.

In our lab, we exposed raw NVMe LUNS to the database server and used Udev rules to manage them.

```
# cat 99-nvme.rules
```

```
KERNEL=="nvme*[0-9]n*[0-9]", SUBSYSTEM=="block",
ENV{DEVTYPE}=="disk",
ENV{ID_WWN}=="eui.119aee66b19fd8478ccf096800e757cf",
SYMLINK+="oracleasm/ORA-ASM-CRS-01",  OWNER="grid",
GROUP="oinstall", MODE="0660"
KERNEL=="nvme*[0-9]n*[0-9]", SUBSYSTEM=="block",
ENV{DEVTYPE}=="disk",
ENV{ID_WWN}=="eui.5c26bf084208e48f8ccf096800a1797e",
SYMLINK+="oracleasm/ORA-ASM-CRS-02",  OWNER="grid",
GROUP="oinstall", MODE="0660"
KERNEL=="nvme*[0-9]n*[0-9]", SUBSYSTEM=="block",
ENV{DEVTYPE}=="disk",
ENV{ID_WWN}=="eui.5f7f4857f9fd372f8ccf096800c83504",
SYMLINK+="oracleasm/ORA-ASM-CRS-03",  OWNER="grid",
GROUP="oinstall", MODE="0660"
KERNEL=="nvme*[0-9]n*[0-9]", SUBSYSTEM=="block",
ENV{DEVTYPE}=="disk",
ENV{ID_WWN}=="eui.67a575ce1ad9c64c8ccf09680041d5c8",
SYMLINK+="oracleasm/ORA-ASM-DATA-01", OWNER="grid",
GROUP="oinstall", MODE="0660"
KERNEL=="nvme*[0-9]n*[0-9]", SUBSYSTEM=="block",
ENV{DEVTYPE}=="disk",
ENV{ID_WWN}=="eui.9a1fa26a642f902f8ccf096800a468c1",
SYMLINK+="oracleasm/ORA-ASM-FRA-01",  OWNER="grid",
GROUP="oinstall", MODE="0660"
#
```

Note: The ID_WWN values were derived from PowerStore Manager.

After the Udev rules are defined, Udev must be restarted. Then, the rules must be triggered to be used.

```
# udevadm control --reload-rules
# udevadm trigger
```

After the Udev rules have been triggered and NVMe volumes created in PowerStore and presented to the database server, the physical server can see the LUNs.

```
# ls -ltr /dev/oracleasm
total 0
lrwxrwxrwx 1 root root 10 Apr 20 11:26 ORA-ASM-CRS-03 ->
../nvme0n2
lrwxrwxrwx 1 root root 10 Apr 20 11:26 ORA-ASM-CRS-01 ->
../nvme0n3
lrwxrwxrwx 1 root root 10 Apr 20 11:40 ORA-ASM-CRS-02 ->
../nvme0n4
lrwxrwxrwx 1 root root 10 Apr 20 11:40 ORA-ASM-DATA-01 ->
../nvme0n5
lrwxrwxrwx 1 root root 10 Apr 20 11:40 ORA-ASM-FRA-01 ->
../nvme0n6
# ls -ltr /dev | grep nvme0n[23456]
```

```
brw-rw----   1 grid oinstall 259,   7 Apr 20 11:41 nvme0n4
brw-rw----   1 grid oinstall 259,   3 Apr 20 11:41 nvme0n2
brw-rw----   1 grid oinstall 259,   5 Apr 20 11:45 nvme0n3
brw-rw----   1 grid oinstall 259,  13 Apr 20 11:45 nvme0n6
brw-rw----   1 grid oinstall 259,   9 Apr 20 11:45 nvme0n5
[root@r730xd-1 rules.d]#
```

Deploying Oracle on PowerStore T models

Introduction

PowerStore T models offer a storage solution for Oracle workloads regardless of the application characteristics. This section discusses best practices for the architecture and configuration of PowerStore storage for Oracle databases to create an optimal and manageable environment.

Oracle database design considerations

The storage system is a critical component of any Oracle database environment. Sizing and configuring a storage system without understanding the requirements can have adverse consequences. This section discusses the types of database workloads and some common tools available to help define the requirements.

OLTP workloads

An online transaction processing (OLTP) workload typically consists of small random reads and writes. The I/O sizes are generally equivalent to the database block size. The primary goal of designing a storage system for this type of workload is to maximize the number of IOPS while minimizing the latency.

OLAP or DSS workloads

Unlike an OLTP workload, an online analytic processing (OLAP) or decision support system (DSS) workload has a relatively low volume of transactions. Most of the activities involve complex queries and aggregate a large dataset. The volume of data tends to grow steadily over time and is kept available for a longer time. OLAP workloads generally have large sequential reads or writes.

The primary goal of designing a storage system that services this type of workload is to optimize the I/O throughput. The design must consider all components in the entire I/O path between the hosts and the drives in PowerStore. Meeting high-throughput requirements might require having multiple HBAs on the server and adding front-end ports on PowerStore.

Mixed workloads

Oracle I/O patterns do not always follow a strict OLTP or OLAP pattern because Oracle databases can be designed to service both OLTP and OLAP applications. In cases like this, gather performance metrics and choose a design that provides the best sizing result for mixed workloads.

Oracle Automatic Storage Management

Dell Technologies and Oracle recommend using Oracle Automatic Storage Management (ASM) to manage database LUNs. This section reviews general guidelines and additional considerations for an Oracle database.

Preparing storage for Oracle ASM

Ensure that proper Linux user ownership, group ownership, and permissions are set correctly on ASM disks. The operating-system user that owns the ASM instance must own the LUNs and have read/write privilege to them. For example, if user `grid` with primary group `oinstall` is the owner of the ASM instance, `grid:oinstall` should be assigned to the LUNs. These settings must be persistent across host reboots, and across all nodes in a RAC cluster.

Persistent device ownership and permissions

Persistent device ownership and permission can be managed through various software. The following section describes how to use ASMLib to accomplish this management task. ASMFD can also be used to manage device persistence, ownership, and permissions.

Note: If ASMLib or ASMFD are intended to manage ASM disks created from NVMe volumes, review the NVMe information available on Oracle Support.

Oracle ASMLib

Oracle ASMLib simplifies storage management and reduces kernel resource usage. It provides persistency for the device file name, ownership, permission, and reduces the number of open file handles that the database processes require. Using Udev rules is not required when ASMLib is used.

When LUNs are initialized with ASMLib, special device files are created in the /dev/oracleasm/disks folder with proper ownership and permission automatically applied. When the system reboots, the ASMLib driver restarts and re-creates the device files. ASMLib consists of three packages:

- oracleasm-support-*version.arch*.rpm
- oracleasm-kernel-*version.arch*.rpm
- oracleasmlib-*version.arch*.rpm

Each Linux vendor maintains their own kernel driver RPM (oracleasm-kernel-version.arch.rpm). With Oracle Linux, the kernel driver is already included with Oracle Linux Unbreakable Enterprise Kernel, so do not install it. For more information about ASMLib and to download the software, see Oracle ASMLib.

The ownership of the ASMLib devices is defined in the `/etc/sysconfig/oracleasm` configuration file which is generated by running `oracleasm configure -i`. Update the configuration file if necessary to reflect the proper ownership, the disk scanning order, and the disk scanning exclude list.

```
# cat /etc/sysconfig/oracleasm
# ORACLEASM_ENABLED: 'true' means to load the driver on boot.
ORACLEASM_ENABLED=true

# ORACLEASM_UID: Default user owning the /dev/oracleasm mount
point.
ORACLEASM_UID=grid
```

```
# ORACLEASM_GID: Default group owning the /dev/oracleasm mount
point.
ORACLEASM_GID=oinstall

# ORACLEASM_SCANBOOT: 'true' means scan for ASM disks on boot.
ORACLEASM_SCANBOOT=true

# ORACLEASM_SCANORDER: Matching patterns to order disk scanning
ORACLEASM_SCANORDER="dm-"

# ORACLEASM_SCANEXCLUDE: Matching patterns to exclude disks from
scan
ORACLEASM_SCANEXCLUDE="sd"

# ORACLEASM_SCAN_DIRECTORIES: Scan disks under these directories
ORACLEASM_SCAN_DIRECTORIES=""

# ORACLEASM_USE_LOGICAL_BLOCK_SIZE: 'true' means use the logical
block size
# reported by the underlying disk instead of the physical. The
default
# is 'false'
ORACLEASM_USE_LOGICAL_BLOCK_SIZE=false
```

This configuration file indicates `grid:oinstall` for the ownership and it searches for multipath devices (dm) and excludes any single path devices (sd).

Note: The asterisk (*) cannot be used in the value for `ORACLEASM_SCANORDER` and `ORACLEASM_SCANEXCLUDE`.

Oracle requires partitioning the LUNs for ASMLib use. First, create a partition with `parted`, and use `oracleasm` to label the partition. ASMLib does not provide multipath capability and relies on native or third-party multipath software to provide the function. The following example shows creating an ASMLib device on a partition of a Linux Multipath device. The oracleasm command writes the ASMLib header to `/dev/mapper/mpathap1` and generates the ASMLib device file in `/dev/oracleasm/disks` with ownership as indicated in the `/etc/sysconfig/oracleasm` file.

```
# oracleasm createdisk DATA01 /dev/mapper/ora-asm-data-001
```

Setting the asm_diskstring ASM instance parameter

The `asm_diskstring` ASM instance parameter tells ASM the location of the ASM devices. During the Grid Infrastructure installation, the parameter defaults to null and it should be updated to reflect the correct location of the device files.

The following table provides an example of settings.

Table 4. Example of asm_diskstring settings

Device files	asm_diskstring setting
Linux native multipath	asm_diskstring='/dev/mapper/ORA*'
Oracle ASMLib	asm_diskstring='ORCL:*'

Oracle ASM guidelines

Dell Technologies and Oracle recommend using Oracle ASM as the preferred storage management solution for either a single-instance database or RAC database. ASM takes the place of the traditional Linux volume manager and file system. It takes over the management of disks and disk groups where database data reside. Disks are either raw LUNs or single partitioned LUNs. Both types of LUNs are stamped with a disk header that identifies them as ASM disks. Logical collections of ASM disks are known as ASM disk groups.

Benefits of using Oracle ASM

ASM offers many advantages over the traditional Linux storage management solution such as Logical Volume Manager (LVM). The main benefits include:

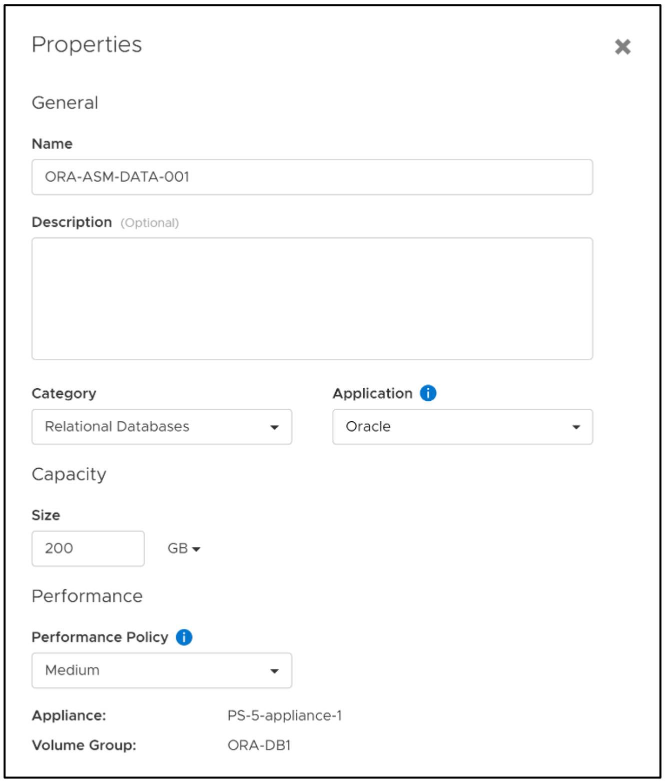
- Automatic file management
- Online datafile rebalancing across ASM disks
- Online addition and removal of ASM disks without downtime
- Single solution for both volume and file management that is integrated with Oracle software
- Improved I/O performance because ASM stripes all files across all disks in a disk group
- Transparent integration with PowerStore features such as snapshots, thin-provisioning, thin clones, compression, and Data at Rest Encryption (D@RE)

ASM disk and disk group guidelines

When creating an Oracle ASM disk group, consider the following guidelines:

- For flexibility and configuration consistency, consider creating separate ASM disk groups for each of the following items:
  - Oracle Cluster Registry (OCR/CRS disk group) and voting files
  - Grid Infrastructure Management Repository (GIMR disk group)
  - Database datafile for each database (one or more ASM disk groups)
  - Fast recovery area for each database (FRA disk group)
  - Offline redo logs (ARCH disk group)
  - Temporary files (TEMP disk group)
  - Online redo logs to provide multiplexing of redo logs and Oracle control files (REDO disk group)

- If Dell AppSync will be used to manage database snapshots, restores, and recoveries, ensure that disk groups are created using the recommendations in the AppSync documentation.
- Configure a database which can span across multiple ASM disk groups but with each ASM disk group mounted and used by one database exclusively. This configuration allows for independently optimizing the storage and snapshot configuration for each individual database.
- LUNs within an ASM disk group should be created with the same capacity and volume performance policy and belong to the same volume group (see the following figure). For information about volume groups, see Volume groups.



Properties

General

Name: ORA-ASM-DATA-001

Description (Optional)

Category: Relational Databases

Application: Oracle

Capacity

Size: 200 GB

Performance

Performance Policy: Medium

Appliance: PS-5-appliance-1

Volume Group: ORA-DB1

Figure 6. Volume Properties

- Use fewer but larger LUNs to reduce the number of managed objects.
- To take an array-based snapshot on an Oracle Standalone or RAC database, ensure that all LUNs belonging to the same database are snapped together. Perform this task by grouping the LUNs in a volume group. Also ensure that the write-order attribute on the volume group is selected. For more information, see Volume groups.
- While ASM can provide software-level mirroring, it is not necessary because of the integrated Dynamic Resiliency Engine (DRE) in PowerStore. Using external redundancy for ASM disk groups is recommended. External redundancy yields substantial storage savings, reduces overall IOPS from ASM, and improves I/O performance.
- For optimal storage efficiency, create thin-provisioned LUNs on PowerStore for ASM use. When administrators create the tablespaces and datafile on the ASM disk

groups, they can set an initial size of each datafile. They can also specify the `autoextend` clause to include an extent size for growth. PowerStore allocates storage for the initial datafile size. As the data are written to the datafile, additional space is allocated in increments of autoextend size. Here is an example of the `create tablespace` statement:

```
SQL> create tablespace DATATS datafile '+DATADG' size 10G
autoextend on next 1024M maxsize unlimited;
```

- Space reclamation: When ASM rebalances the disk group, there is a compact phase at the end of the rebalance. One of the actions taken during the compact phase is space reclamation providing the ASM stack supports it. If ASMlib is used, space is not reclaimed as ASMLib does not support SCSI unmap and SCSI trim. However, ASMFD does support SCSI unmap and SCSI trim and performs space reclamation during the compact phase. It is possible to disable the compact phase on individual ASM disk groups by setting the hidden disk group attribute `_rebalance_compact` to `'FALSE'`:

  ```
  SQL > ALTER DISKGROUP DATADG SET ATTRIBUTE
  '_rebalance_compact'='FALSE';
  ```

  For Oracle pre-12c releases, this phase can only be disabled on the ASM-instance level which affects all ASM disk groups. Because PowerStore 3.0 uses all NVMe SSDs in the base enclosure, turning off the compact phase should not adversely impact performance. However, if PowerStore 3.0 uses expansion shelves of SAS drives, turning off the compact phase might adversely impact performance. Because all applications are different and have their own data usage patterns, test this feature disabled before implementing the change in production.

  For more information about ASM compact-rebalancing, see Oracle KB Doc ID 1902001.1 on Oracle Support.

The following table shows an example of how ASM disk groups can be organized:

Table 5. Example of simple ASM disk group configuration

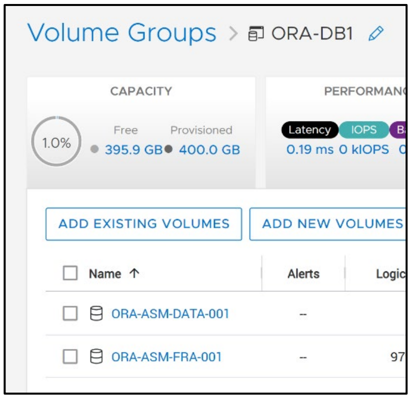
Purpose	ASM disk group	LUN size	PowerStore volume group	Description
OCR	GIDATA	10 GB	N/A	Clusterware-related information such as the OCR and voting disks. Also used for Oracle Restart.
Grid Infrastructure Management Repository	MGMT	50 GB	mgmt_cg	Starting with 12cR2, a separate disk group is created for the GI Management Repository data.
Test database (testdb)	DATADG	200 GB	testdb_cg	Disk group that holds the database files, temporary table space, and online redo logs; contains system-related table spaces such as SYSTEM and UNDO. Contains only testdb data.
	FRADG	100 GB		Disk group for database archive logs and backup data. Contains only testdb logs.

Purpose	ASM disk group	LUN size	PowerStore volume group	Description
Development database (devdb)	DATA2DG	200 GB	devdb_cg	Disk group for database files, temporary table space, online redo logs; contains system-related table spaces such as SYSTEM and UNDO. Contains only devdb data.
	FRA2DG	100 GB		Disk group for database archive logs and backup data. Contains only devdb logs.

ASM disk groups and PowerStore volume groups

For performance reasons, it is common for a database to span across multiple LUNs to increase I/O parallelism to the storage devices. Group the LUNs of a database (Standalone or RAC) into a PowerStore volume group and enable the **Apply write-order consistency to protect all volume group members** option in PowerStore Manager. This grouping ensures data consistency when taking storage snapshots. The PowerStore snapshot feature is a quick and space-efficient way to create a point-in-time snapshot of the entire database. For information about using PowerStore snapshots and thin clones to reduce database recovery time and create space-efficient copies of the database, see Snapshots and recoveries with Oracle and Thin clones.

In the following figure, all volumes belonging to db1 are members of Volume Group ORA-DB1. Within ASM, those volumes belong to two ASM disk groups: **+DATA**, and **+FRA**, respectively.



Free 395.9 GB ● Provisioned 400.0 GB

Latency 0.19 ms IOPS 0 kIOPS

ORA-ASM-DATA-001 --

ORA-ASM-FRA-001 -- 97

Figure 7. Volume members of volume group ORA-DB1

Volume groups defined with the write-order attribute selected, allow consistent snapshots to be taken of a 12c database spanning multiple LUNs within the same appliance. With Oracle 12c and later versions, if PowerStore is used, there is no need to issue BEGIN and END BACKUP command options in Oracle. For more information about:

- Supported backup, restore, and recovery operations using third-party snapshots—See Oracle KB Doc ID 604683.1 on Oracle Support.
- Preparing the database for snapshots and ensuring consistent recovery—See Oracle KB Doc ID 221779.1 on Oracle Support.

Note: Because a PowerStore volume group cannot span multiple appliances, do not use storage from multiple appliances for the same Standalone or RAC database. Storage snapshots that are taken on a multiple-LUN database without a volume group might not be usable to refresh or restore a database successfully.

Expanding Oracle ASM storage

As the storage consumption grows over time, increasing and growing the existing storage capacity both in PowerStore and in the database might be necessary. PowerStore is ideal because it supports adding capacity online with minimal business interruptions. PowerStore has the flexibility to expand the current storage system with no interruption to the application. The following nondisruptive operations can be performed online in PowerStore Manager:

- Add flash devices (drives) to available bays
- Create and add new LUNs to existing hosts

The following sections discuss the different ways to increase ASM storage capacity. Each method has advantages and disadvantages.

Increasing Oracle ASM storage by adding new LUNs

Storage capacity can be added to an ASM disk group by adding new LUNs to the disk group. The advantage of this method is that the process is relatively simple and safe because no changes are made to the existing LUNs.

The general process is as follows:

1. Create a volume in PowerStore Manager.
2. Ensure that the size of the new volume and other volume attributes matches the attributes of existing volumes of the same ASM disk group.
3. Add the volume to the appropriate volume group.
4. Map the new volume to the host system.
5. Perform a SCSI scan on the host systems.
6. Configure multipath for the new device.
7. Prepare the LUN for ASM, and then create the ASM disk.
8. Add the ASM disk to the ASM disk group.

   Because ASM automatically rebalances the data after a LUN is added, adding multiple LUNs in a single operation is recommended. This approach minimizes the amount of rebalancing work required. The following example shows the `ALTER DISKGROUP ADD DISK` Oracle SQL statement to add multiple devices to a disk group:

   ```
   ALTER DISKGROUP DATADG ADD DISK 'AFD:DATADG_VOL1',
   'AFD:DATADG_VOL2' REBALANCE POWER 10 NOWAIT;
   ```

9. In Linux, verify the status and capacity of the disk group.

   ```
   # asmcmd lsdsk -gk -G datadg
   # asmcmd lsdg -g datadg
   ```

Space reclamation

PowerStore supports SCSI unmap. This feature allows operating systems to inform which data blocks are no longer in use and can be released for other uses. For space reclamation to work, the LUNs must be thin-provisioned in PowerStore, and both the Linux kernel and Oracle ASM must also support the feature.

With trim/unmap enabled on a Linux mount point, significant wait time occurs when creating a file system that is more than a few TB in size on the PowerStore volume. The larger the volume is, the longer the format wait time is. This scenario is a common with external storage.

File systems

A local file system is preferred to store Oracle software and diagnostic logs. Datafiles can reside in a local file system, but using Oracle ASM on block devices or Oracle Direct NFS with PowerStore NFS services is recommended.

PowerStore storage provisioning and configuration best practices

Introduction

To provision storage in PowerStore, host objects and volumes must be created in PowerStore T models.

Setting up host objects in PowerStore T models

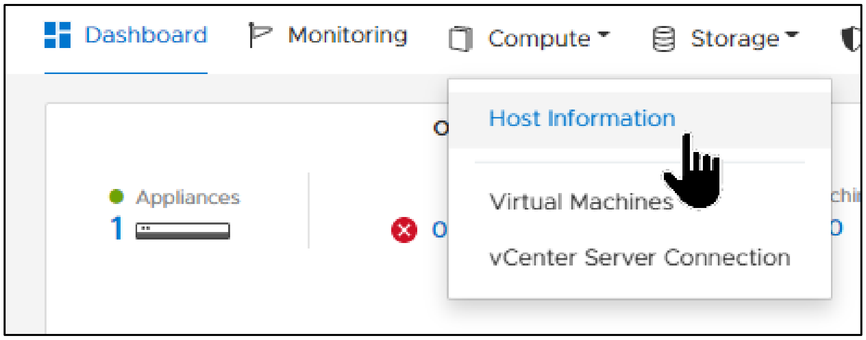
A PowerStore host object must be created for each Linux server that has its storage serviced by PowerStore. To simplify defining the logical server objects in PowerStore, create them after completing the following steps:

1. Configure the appropriate hardware for the physical server, and rack and cable the server.
2. Zone the server into the fabric.
3. Power on the database server.
4. On the database server, enable the HBAs in QLogic BIOS.
5. Allow PowerStore to see the enabled HBAs in the server as it interrogates the fabric.

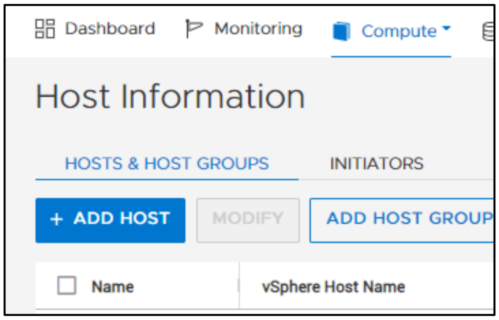
Installing at least two 32 Gb dual-port HBAs in the physical server is recommended. Two dual-port HBAs provide redundancy of initiator ports and HBAs. More HBAs or HBA ports might be necessary to provide the bandwidth necessary to support the expected database I/O performance requirements.

Perform the following steps to define a host object in PowerStore T models:

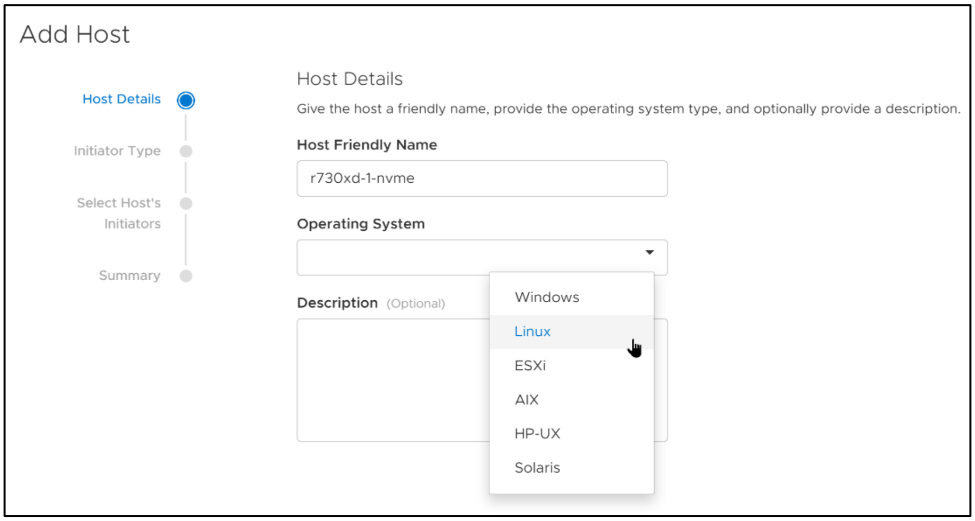
1. In PowerStore Manager, select **Compute** > **Hosts Information**.



2. Click **ADD HOST**.



3. In the **Add Host** wizard, enter the **Host Friendly Name** and **Operating System.** Then, click **NEXT**.



Host Details

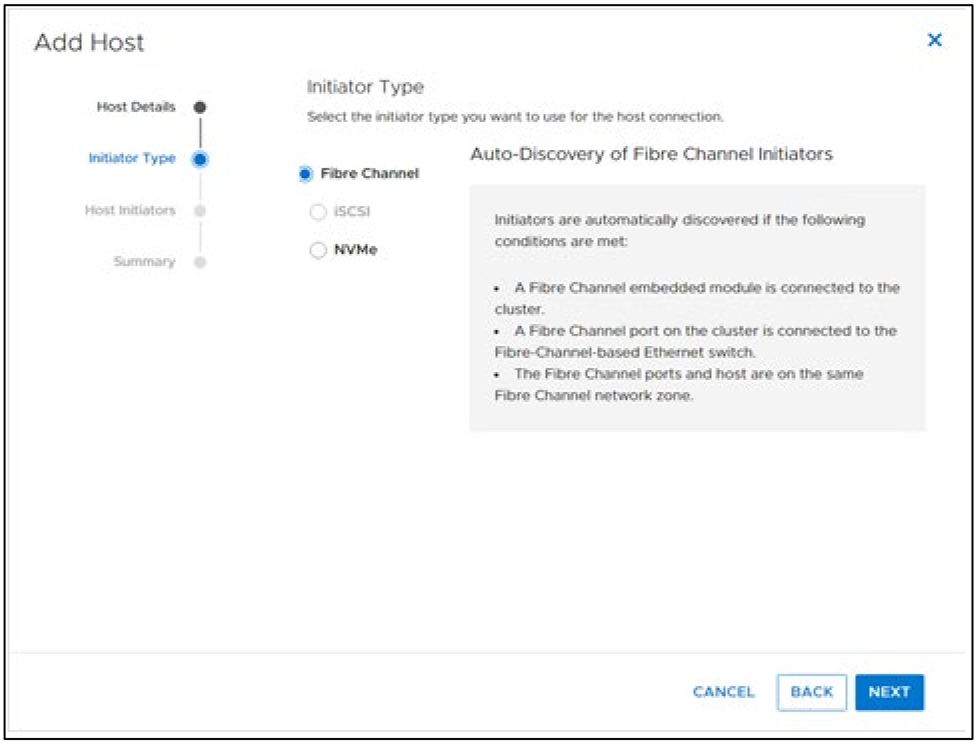
Give the host a friendly name, provide the operating system type, and optionally provide a description.

Host Friendly Name: r730xd-1-nvme

Operating System:

Description (Optional):

4. Select the **Initiator Type** that should be used for this host. Then, click **NEXT**.



Initiator Type

Select the initiator type you want to use for the host connection.

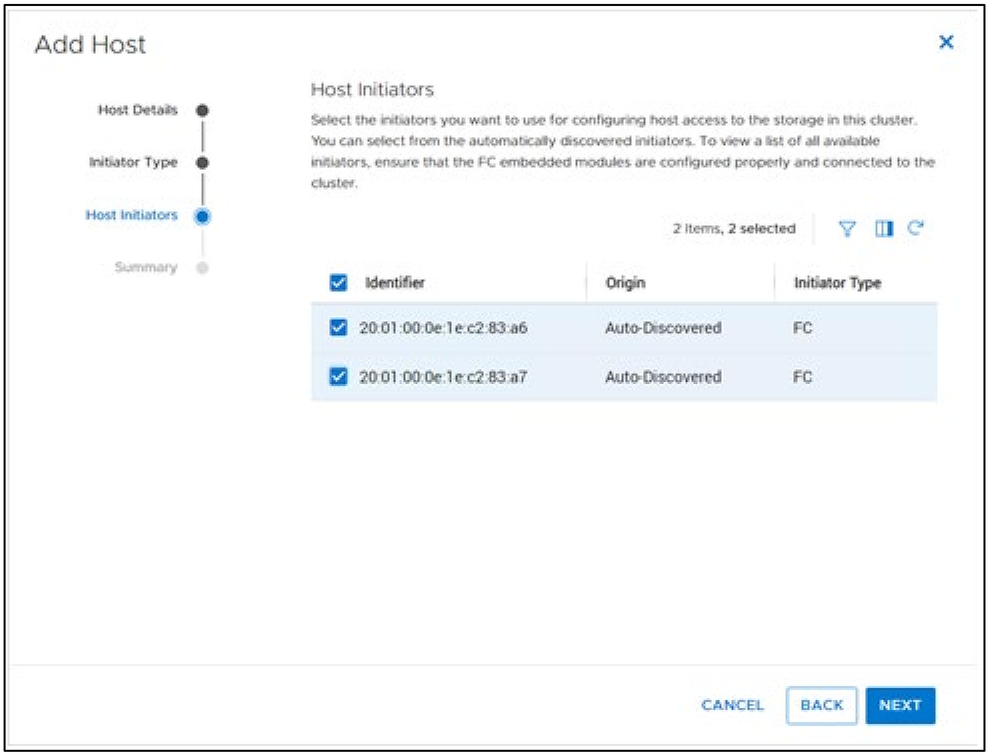
● Fibre Channel
○ iSCSI
○ NVMe

Auto-Discovery of Fibre Channel Initiators

Initiators are automatically discovered if the following conditions are met:

- A Fibre Channel embedded module is connected to the cluster.
- A Fibre Channel port on the cluster is connected to the Fibre-Channel-based Ethernet switch.
- The Fibre Channel ports and host are on the same Fibre Channel network zone.

5. If NVMe was selected in the previous step, go to step 8. Otherwise, if Fibre Channel was selected, select the host initiators belonging to the physical server and click **NEXT**.



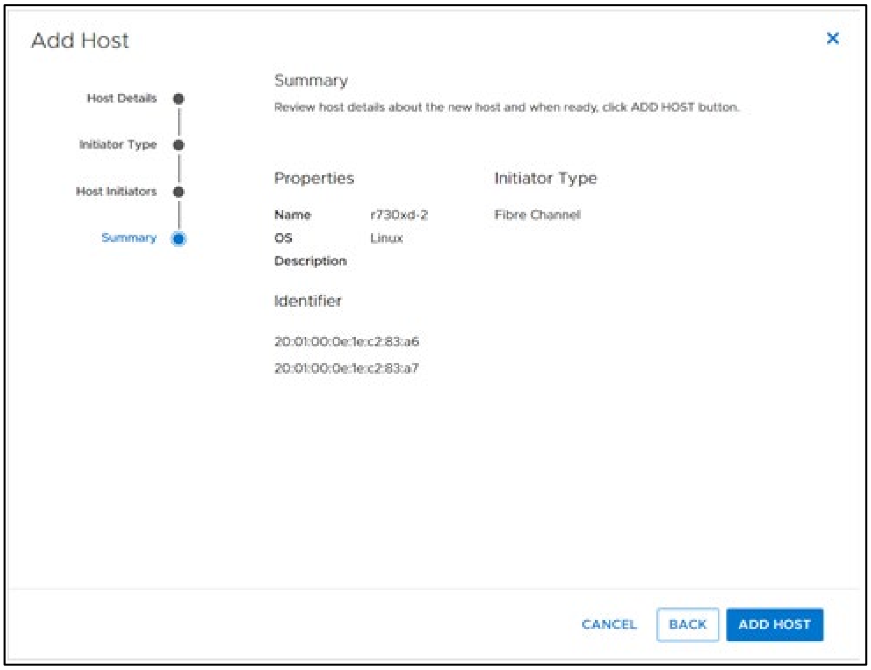
Host Initiators

Select the initiators you want to use for configuring host access to the storage in this cluster. You can select from the automatically discovered initiators. To view a list of all available initiators, ensure that the FC embedded modules are configured properly and connected to the cluster.

2 Items, 2 selected

☑	Identifier	Origin	Initiator Type
☑	20:01:00:0e:1e:c2:83:a6	Auto-Discovered	FC
☑	20:01:00:0e:1e:c2:83:a7	Auto-Discovered	FC

6. Review the host properties and click **ADD HOST**.



Add Host

Summary

Review host details about the new host and when ready, click ADD HOST button.

Properties

Name r730xd-2
OS Linux
Description

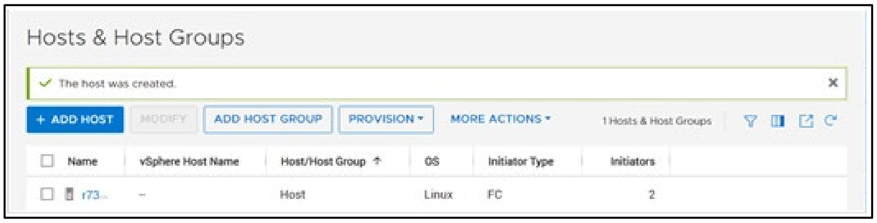
Initiator Type

Fibre Channel

Identifier

20:01:00:0e:1e:c2:83:a6
20:01:00:0e:1e:c2:83:a7

7. PowerStore Manager displays the newly added host object.

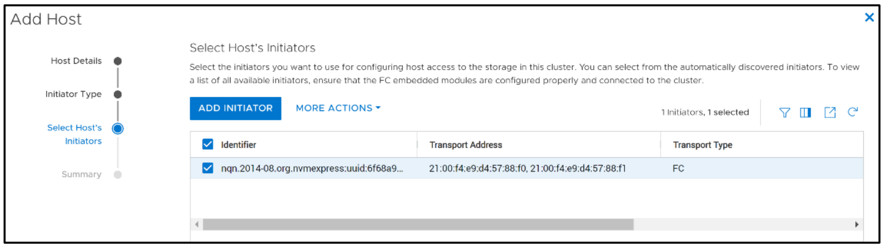


Hosts & Host Groups

✓ The host was created.

□	Name	vSphere Host Name	Host/Host Group ↑	OS	Initiator Type	Initiators
□	r73...	--	Host	Linux	FC	2

8. If the initiator type was set to NVMe, select the initiator that has an **Identifier** value matching the database server's NQN. Then, click **NEXT**.



Add Host

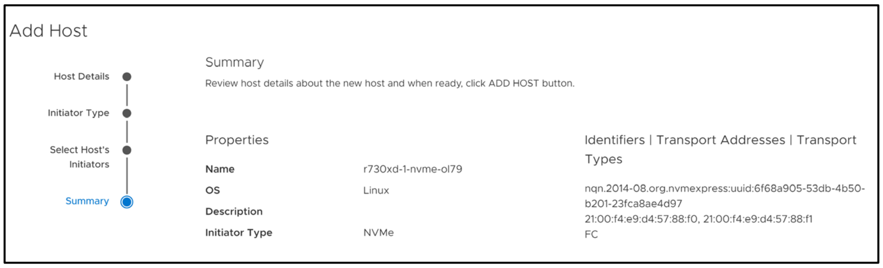
Select Host's Initiators

Select the initiators you want to use for configuring host access to the storage in this cluster. You can select from the automatically discovered initiators. To view a list of all available initiators, ensure that the FC embedded modules are configured properly and connected to the cluster.

1 Initiators, 1 selected

☑	Identifier	Transport Address	Transport Type
☑	nqn.2014-08.org.nvmexpress:uuid:6f68a9...	21:00:f4:e9:d4:57:88:f0, 21:00:f4:e9:d4:57:88:f1	FC

9. Review the request to create the host object. If the request is correct, click **ADD HOST**.



Add Host

Summary

Review host details about the new host and when ready, click ADD HOST button.

Properties

Name	r730xd-1-nvme-ol79
OS	Linux
Description	
Initiator Type	NVMe

Identifiers | Transport Addresses | Transport Types

nqn.2014-08.org.nvmexpress:uuid:6f68a905-53db-4b50-b201-23fca8ae4d97
21:00:f4:e9:d4:57:88:f0, 21:00:f4:e9:d4:57:88:f1
FC

If Oracle RAC is used, repeat this process for each host that belongs to the Oracle cluster.

Provisioning volumes

For Oracle, volumes must reside in a PowerStore volume group if snapshots will be created while the database is open and used for database clones, restores, or recoveries. There are several best practices for provisioning storage from PowerStore T models for an Oracle database. This section illustrates one way to create volume groups and volumes, and how to assign volumes to a volume group.

Volume groups

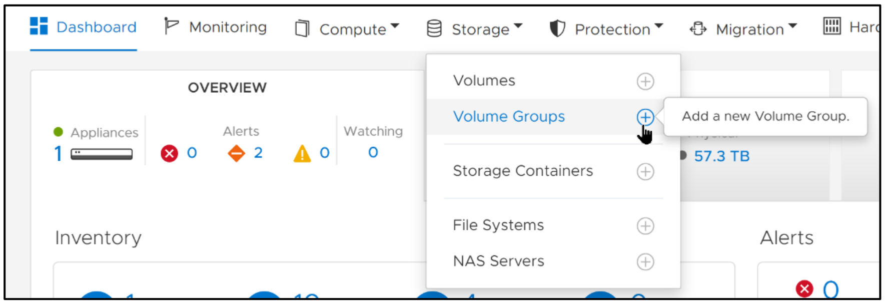
A volume group is a collection of volumes or thin-clone volumes that reside on the same PowerStore appliance. The volume group provides a single point of management for the volume members. Volumes from multiple appliances are not allowed within the same volume group.

The write-order attribute is an optional volume group attribute. This attribute ensures that the order of writes entering the system is maintained for all volume group members. When a snapshot is taken of the volume group using this attribute, a crash-consistent point-in-time snapshot is taken across all volume-group members simultaneously. A crash-consistent snapshot that uses the write-order attribute does not guarantee application consistency. The application must be able to provide that capability.

If snapshots will be used for refreshes and restores, all LUNs that contain the database must reside in the same volume group.

Perform the following steps to create a volume group:

1. In PowerStore Manager, select **Storage** > **Volume Groups** (plus symbol).



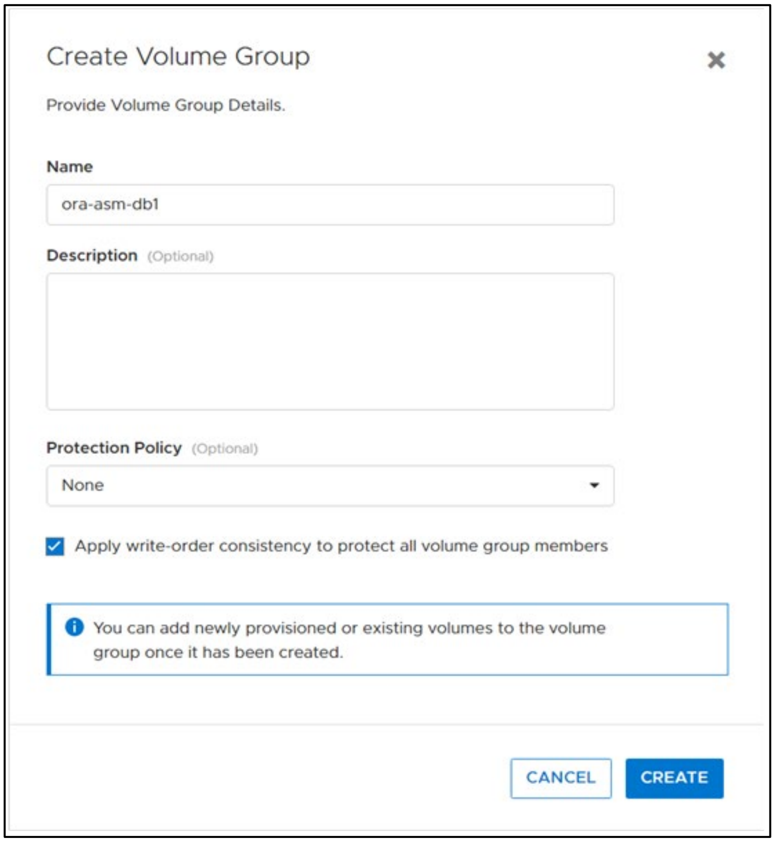
2. Complete the **Create Volume Group** wizard.
   a. Use a meaningful name for the volume group.

b. If snapshots will be used for database restore and refresh operations, keep the default selection of **Apply write-order consistency to protect all volume group members.**

c. Set the Protection Policy.

   This example uses **None** to allow for manual scheduled snapshots.

d. Click **CREATE**.



Create Volume Group

Provide Volume Group Details.

Name: ora-asm-db1

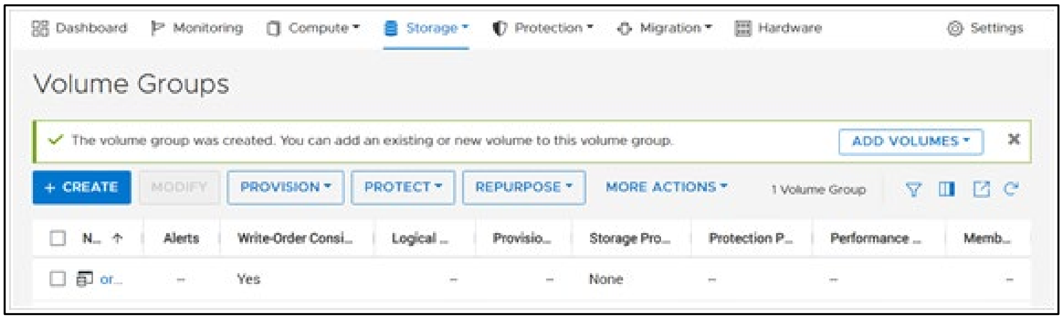
Description (Optional)

Protection Policy (Optional): None

☑ Apply write-order consistency to protect all volume group members

You can add newly provisioned or existing volumes to the volume group once it has been created.

PowerStore Manager displays a message indicating that the volume group was created.



Volume Groups

✓ The volume group was created. You can add an existing or new volume to this volume group.

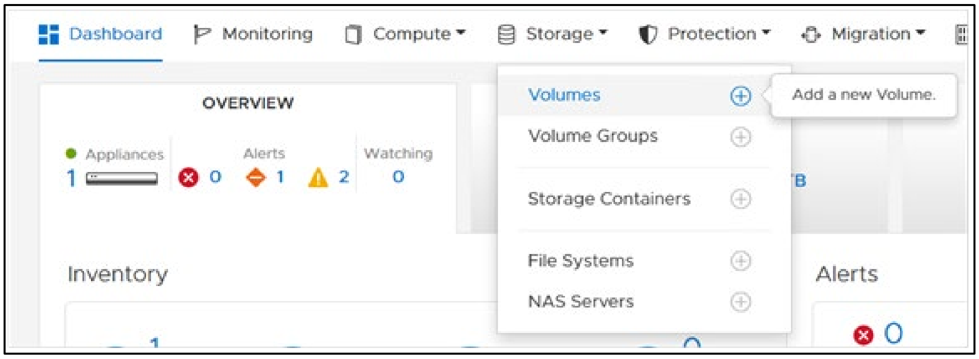
1 Volume Group

□	N... ↑	Alerts	Write-Order Consi...	Logical ...	Provisio...	Storage Pro...	Protection P...	Performance ...	Memb...
□	or...	–	Yes	–	–	None	–	–	–

Creating volumes

Perform the following steps to create a volume and assign it to a volume group:

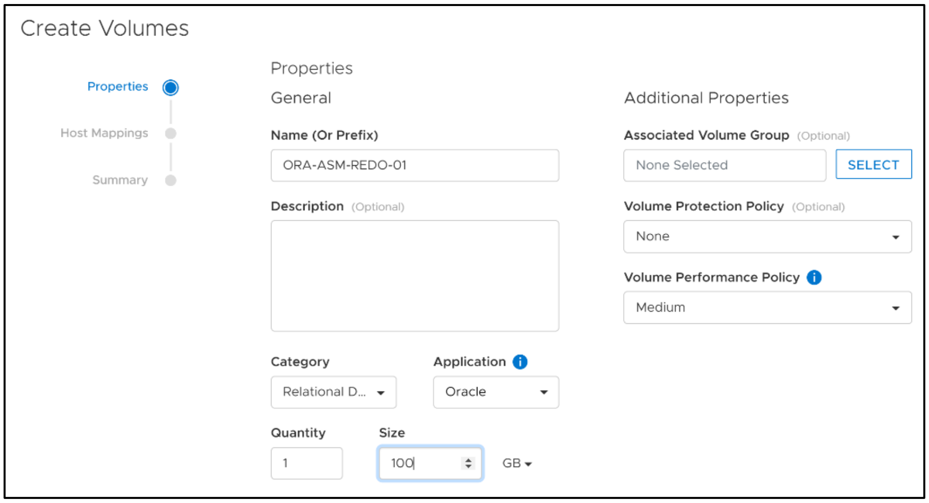
1. In PowerStore Manager, select **Storage** > **Volumes** (plus symbol). Then click **+ CREATE.**



PowerStore Manager displays the Create Volumes wizard.

2. Complete the form, and then click **NEXT**.

Note: Performance policy specifies I/O performance requirements for storage. PowerStore uses share-based QoS. When there is contention at the system level, a performance policy of **High** handles more IOPS than other performance policies. For critical Oracle applications, consider selecting the **High** performance policy so it does not compete for I/O against less-critical applications.



Properties

General

Name (Or Prefix): ORA-ASM-REDO-01

Description (Optional):

Category: Relational D...

Application: Oracle

Quantity: 1

Size: 100 GB

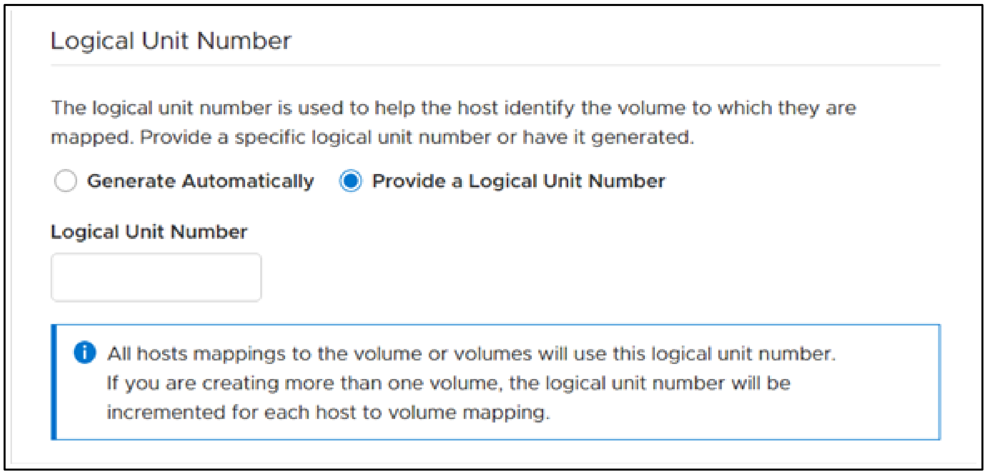
Additional Properties

Associated Volume Group (Optional): None Selected

Volume Protection Policy (Optional): None

Volume Performance Policy: Medium

3. Select the host to present the volume to. If the volume is used in a RAC database, select all nodes that are part of the RAC cluster.
4. Set the **Logical Unit Number**.
   a. Click **Provide a Logical Unit Number**.
   b. Enter the LUN ID in field **Logical Unit Number**. For Oracle RAC environments, ensure that the specified LUN ID is available on all nodes of the RAC cluster.
   c. Click **NEXT**.



Logical Unit Number

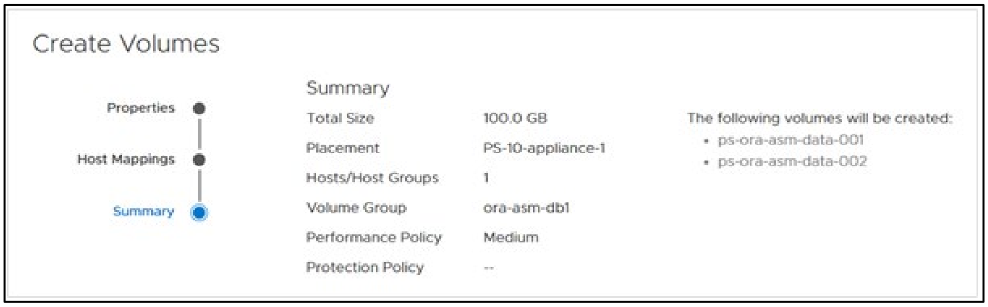
The logical unit number is used to help the host identify the volume to which they are mapped. Provide a specific logical unit number or have it generated.

○ Generate Automatically ● Provide a Logical Unit Number

Logical Unit Number

All hosts mappings to the volume or volumes will use this logical unit number. If you are creating more than one volume, the logical unit number will be incremented for each host to volume mapping.

5. Review the summary information and click **CREATE**.



Create Volumes

Summary

Total Size	100.0 GB
Placement	PS-10-appliance-1
Hosts/Host Groups	1
Volume Group	ora-asm-db1
Performance Policy	Medium
Protection Policy	--

The following volumes will be created:

- ps-ora-asm-data-001
- ps-ora-asm-data-002

PowerStore data reduction and Oracle

PowerStore inline data reduction (compression and deduplication) includes the core features of zero detection, compression, and deduplication. While the amount of reduction varies depending on the type of data, the system automatically selects the best option for reducing the data footprint. Data reduction works seamlessly in the background with PowerStore, is always enabled, and cannot be disabled.

Because data reduction is always active in PowerStore, enabling Oracle database compression might not provide additional savings. Testing database compression in a nonproduction environment before enabling it in a production environment is recommended.

Data encryption

Data at Rest Encryption (D@RE) is enabled by default on PowerStore. No configuration steps are necessary to protect the drives. A top concern for many businesses is application data security, which includes drive encryption requirements, specifically on data at rest. Lost or stolen data can critically damage a business to the point where it might not survive. Dell Technologies engineered PowerStore with D@RE by using self-encrypting drives and array-based, self-managed keys. When activated, data is written to disk using the 256-bit Advanced Encryption Standard (AES). This security feature is provided without adding overhead to administrative tasks or to the application. It also avoids potential performance impact to the application and has no performance impact on the array.

Scripting and automation

To access the PowerStore REST API interface, open a supported web browser and go to the management IP address of the PowerStore cluster. Then, add `/swaggerui` to the end of the address. For example: `https://<mgmt_IP_address>/swaggerui`.

PowerStore file storage

Introduction

PowerStore storage can serve file data through virtual file servers (NAS servers) while providing many of the advanced capabilities of PowerStore systems. Some of these capabilities are shown in the following list, while others are mentioned in the remainder of this section:

- Advanced static routing
- Packet reflect
- IP multitenancy
- NAS server mobility
- Configurable PowerStore system parameters

PowerStore front-end Ethernet connectivity for file storage

PowerStore provides multiple options for Ethernet front-end connectivity, through ports directly on the onboard card, and optional I/O modules. In general, front-end ports need to be connected and configured symmetrically across both nodes to facilitate high availability and continued connectivity if a PowerStore node fails.

With PowerStore file storage, all networks used in the virtual NAS server must be bonded interfaces. Single interfaces are not supported. Each PowerStore NAS server must have at least one bonded network. More bonded networks can be added to the PowerStore NAS server as needed.

For best performance, use all front-end ports that are installed in the system so that workload is spread across all front-end ports for Oracle Direct NFS (dNFS) data traffic. Dell Technologies recommends investigating and testing the need for additional networks in the NAS server.

The following PowerStore has two optional I/O modules (Ethernet and Fibre Channel) per node. With the onboard cards and optional Ethernet I/O modules, each node has eight Ethernet ports, providing four bonded networks with dual, unshared interfaces per bond per node per NAS server.

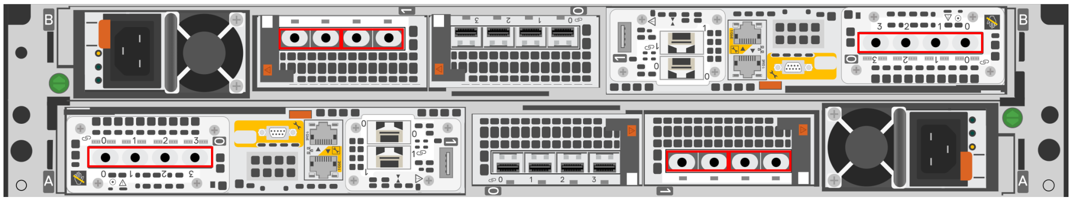


Figure 8. PowerStore onboard and I/O modules—Ethernet interfaces

Note: Dell Technologies recommends configuring the bond using ports from different I/O modules or the onboard card. However, some restrictions apply. For more information, see *Dell PowerStore—Networking Guide for PowerStore T Models*.

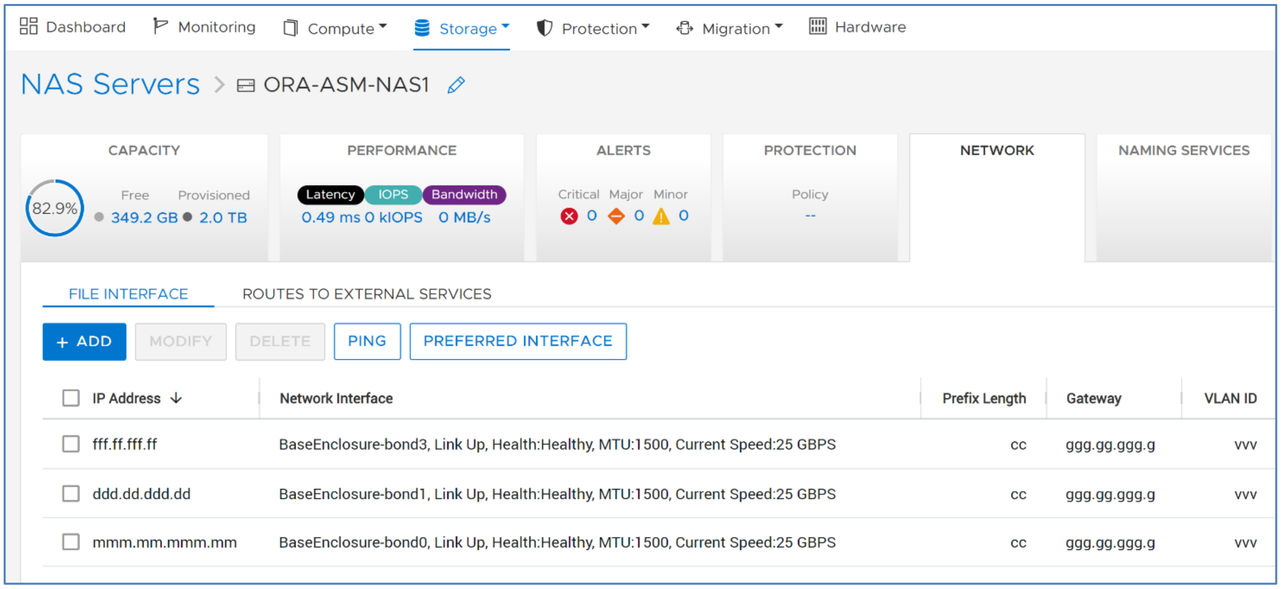
Obfuscated IP addresses and network information

For security reasons, in this paper character strings replace all network configuration information (IPv6 and IPv4 IP addresses, subnet, CIDR, gateway, and VLAN, and MAC addresses). With other redaction methods, the ability to visually map interface mappings and network routing between interfaces is lost. A given character string represents the same IP address throughout the paper. When applying the content of this paper, replace the given character string with the appropriate network information for the target environment.

Table 6. Obfuscated IP addresses and network information used in this paper

NAS server bonded interface and port members	NAS server network IP address	Host interface	Host interface IP address	Usage
		eno3	www.ww.www.ww	Public Ethernet
BaseEnclousre-bond0 (4PortCard-FEPort0 4PortCard-FEPort1)	mmm.mm.mmm.mm	bond1 (enp132s0f0, enp132s0f1)	xxx.xx.xxx.xx	dNFS control traffic
BaseEnclosure-bond1 (IoModule1-FEPort0 IoModule1-FEPort1)	ddd.dd.ddd.dd	enp131s0f0	yyy.yy.yyy.yy	dNFS channel—application data traffic
BaseEnclosure-bond3 (IoModule1-FEPort2 IoModule1-FEPort3)	fff.ff.fff.ff	enp3s0f0	ttt.tt.ttt.tt	dNFS channel—application data traffic

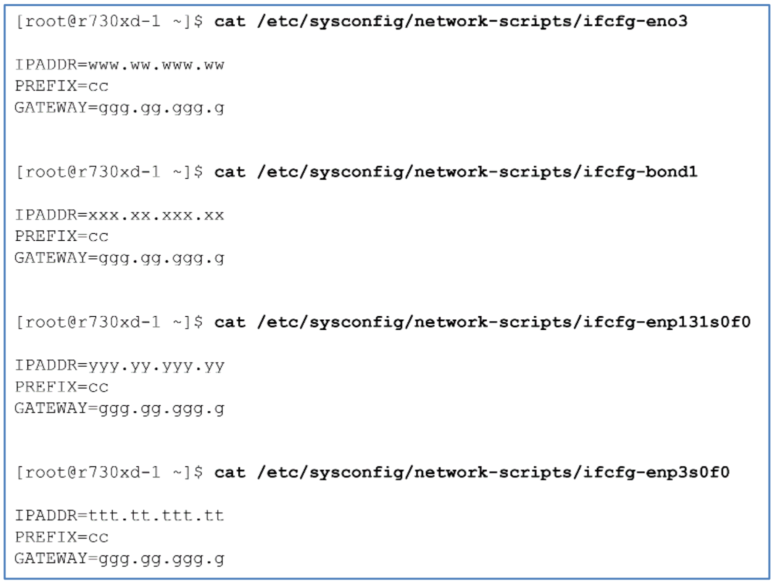
Network component	Obfuscated string
Broadcast	bbb.bbb.bbb.bbb
Subnet (network ID)	nnn.nnn.nnn.nn
Subnet mask	sss.sss.sss.sss
CIDR	cc
Gateway	ggg.gg.ggg.g
VLAN	vvv



NAS Servers > ORA-ASM-NAS1

□	IP Address	Network Interface	Prefix Length	Gateway	VLAN ID
□	fff.ff.fff.ff	BaseEnclosure-bond3, Link Up, Health:Healthy, MTU:1500, Current Speed:25 GBPS	cc	ggg.gg.ggg.g	vvv
□	ddd.dd.ddd.dd	BaseEnclosure-bond1, Link Up, Health:Healthy, MTU:1500, Current Speed:25 GBPS	cc	ggg.gg.ggg.g	vvv
□	mmm.mm.mmm.mm	BaseEnclosure-bond0, Link Up, Health:Healthy, MTU:1500, Current Speed:25 GBPS	cc	ggg.gg.ggg.g	vvv

Figure 9. PowerStore bonded network interfaces



```
[root@r730xd-1 ~]$ cat /etc/sysconfig/network-scripts/ifcfg-eno3

IPADDR=www.ww.www.ww
PREFIX=cc
GATEWAY=ggg.gg.ggg.g


[root@r730xd-1 ~]$ cat /etc/sysconfig/network-scripts/ifcfg-bond1

IPADDR=xxx.xx.xxx.xx
PREFIX=cc
GATEWAY=ggg.gg.ggg.g


[root@r730xd-1 ~]$ cat /etc/sysconfig/network-scripts/ifcfg-enp131s0f0

IPADDR=yyy.yy.yyy.yy
PREFIX=cc
GATEWAY=ggg.gg.ggg.g


[root@r730xd-1 ~]$ cat /etc/sysconfig/network-scripts/ifcfg-enp3s0f0

IPADDR=ttt.tt.ttt.tt
PREFIX=cc
GATEWAY=ggg.gg.ggg.g
```

Figure 10. Snippets of interface configuration files from the NFS client

Dell PowerStore NAS servers

PowerStore virtual NAS servers are assigned to a single PowerStore node. All file systems serviced by a NAS server have their I/O processed by the node on which the NAS server is resident or current. If multiple NAS servers are required, load-balance the NAS servers so that front-end NFS I/O is roughly distributed evenly between the nodes. Do not overprovision either of the nodes so that the peer node does not become overloaded if a failover occurs.

Because each NAS server is logically separate, NFS clients of one NAS server cannot access data on another NAS server. Logically separate NAS servers provide database isolation and protection across multiple NFS clients (database servers).

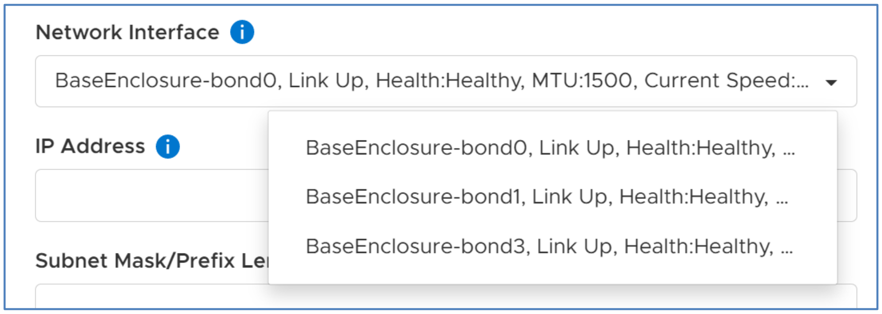
Creating a NAS server and adding a link aggregation network interface

To create a NAS server, perform the following steps in PowerStore Manager:

1. Select **Storage** > **NAS Servers.**
2. Click **+ CREATE.**
3. Provide all the information requested by the Create NAS Server wizard. When prompted for the **Network Interface**, select the appropriate link aggregate.

   For information about creating link aggregations in PowerStore, see PowerStore link aggregation configuration. The link aggregate that is chosen when the NAS server is created becomes the preferred link aggregate for the NAS server. The preferred link aggregate is the network used by PowerStore for controlling and managing the NAS server. PowerStore also uses the preferred link for noncontrolling and nonmanagement type NFS data. If more link aggregates are added to the NAS server, they are added as nonpreferred.

   Note: Only bonded network interfaces in PowerStore can be used for PowerStore NAS networks.

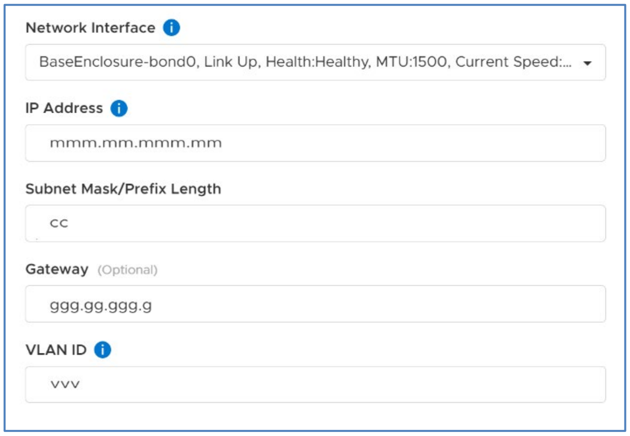
   Network Interface ⓘ
   BaseEnclosure-bond0, Link Up, Health:Healthy, MTU:1500, Current Speed:...

   BaseEnclosure-bond0, Link Up, Health:Healthy, ...
   BaseEnclosure-bond1, Link Up, Health:Healthy, ...
   BaseEnclosure-bond3, Link Up, Health:Healthy, ...

   IP Address ⓘ

   Subnet Mask/Prefix Le

4. Provide the network information for the bonded interface:

   Network Interface ⓘ
   BaseEnclosure-bond0, Link Up, Health:Healthy, MTU:1500, Current Speed:...

   IP Address ⓘ
   mmm.mm.mmm.mm

   Subnet Mask/Prefix Length
   cc

   Gateway (Optional)
   ggg.gg.ggg.g

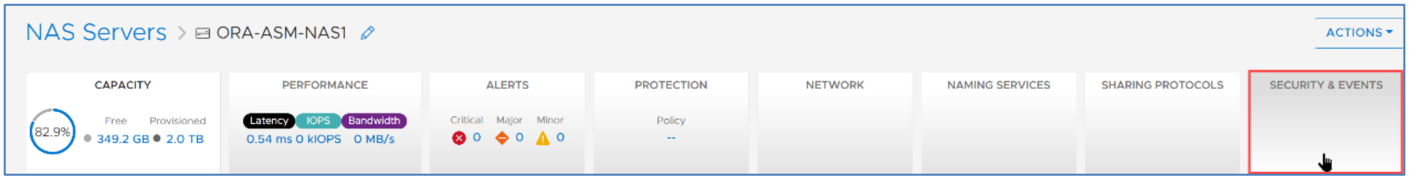
   VLAN ID ⓘ
   vvv

5. Click **NEXT.**
6. Select the protocol type.
7. Click **NEXT.**
8. Do not select anything for **Unix Directory Services.**
9. Click **NEXT.**

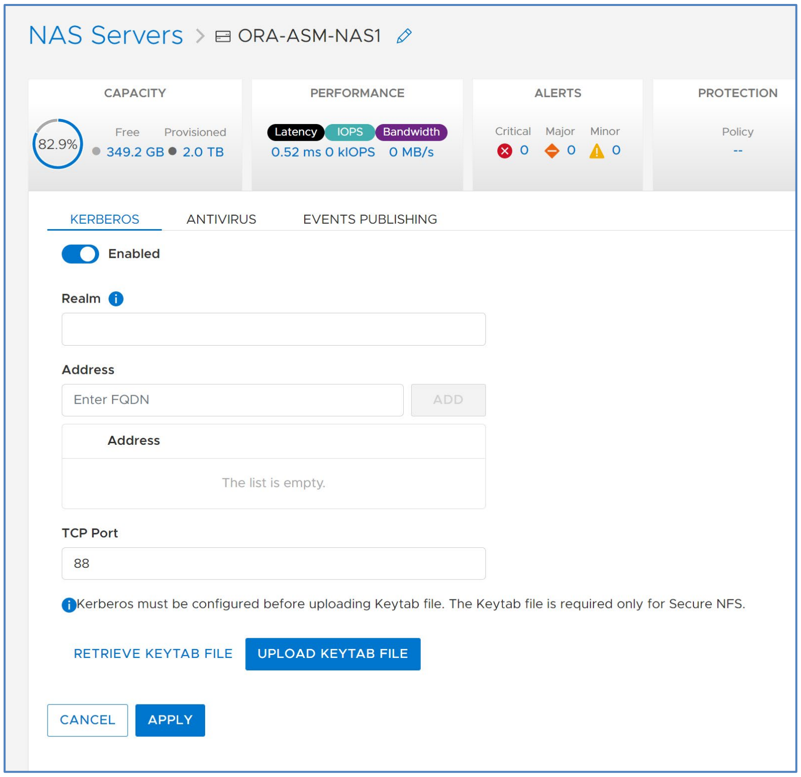
10. Do not select anything for **DNS.**
11. Click **NEXT.**
12. Specify the protection policy if needed.
13. Click **NEXT.**
14. Review the summary information.
15. If the summary information is correct, click **CREATE NAS SERVER.**

If Kerberos is needed for authentication, continue with the following steps:

1. In PowerStore, select **Storage** > **NAS Servers.**
2. Click the name of the NAS server from the NAS Servers view.
3. Click **SECURITY & EVENTS**.



4. Click **Disabled** to switch KERBEROS to **Enabled**.
5. Complete the form:



Enabled

Realm

Address

Enter FQDN ADD

Address

The list is empty.

TCP Port

88

Kerberos must be configured before uploading Keytab file. The Keytab file is required only for Secure NFS.

Note: If Kerberos is used, make note of the fully qualified name that is entered into the **Address** field. The value must be entered in file `oranfstab` on the NFS client (database server) when dNFS is configured.

6. Click **APPLY.**

If throughput is restricted using only one bonded link aggregate Ethernet interface, consider configuring multiple bonded Ethernet interfaces and adding them to the NAS server. For information about adding more bonded network interfaces to a PowerStore NAS server, see the following section.

Adding link aggregate network interfaces to a PowerStore NAS server

To add link aggregates to the PowerStore NAS server, perform the following steps in PowerStore:

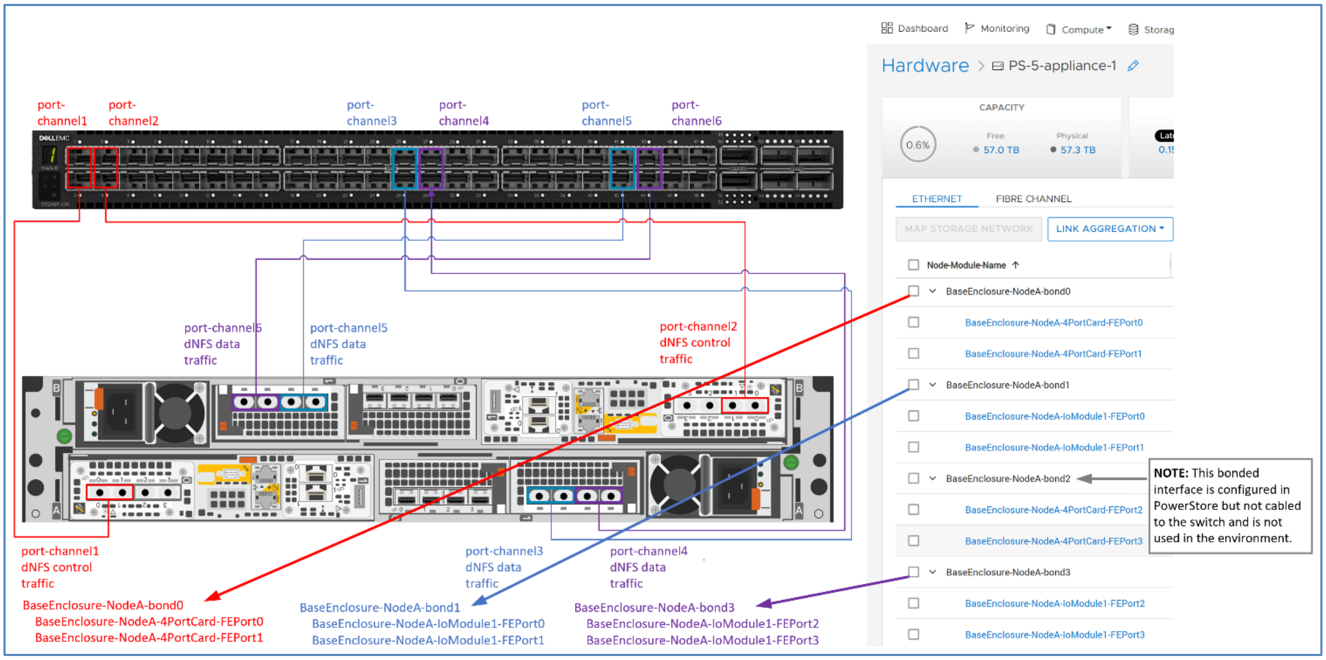
1. Select **Storage** > **NAS Servers.**
2. Under the **Name** column, click the NAS server name.
3. Click **NETWORK**.
4. Click **+ ADD.**
5. Set **Role** to **Production**.

Note: The Production setting allows CIFS, NFS, and FTP access. **Backup** allows NFS access only.

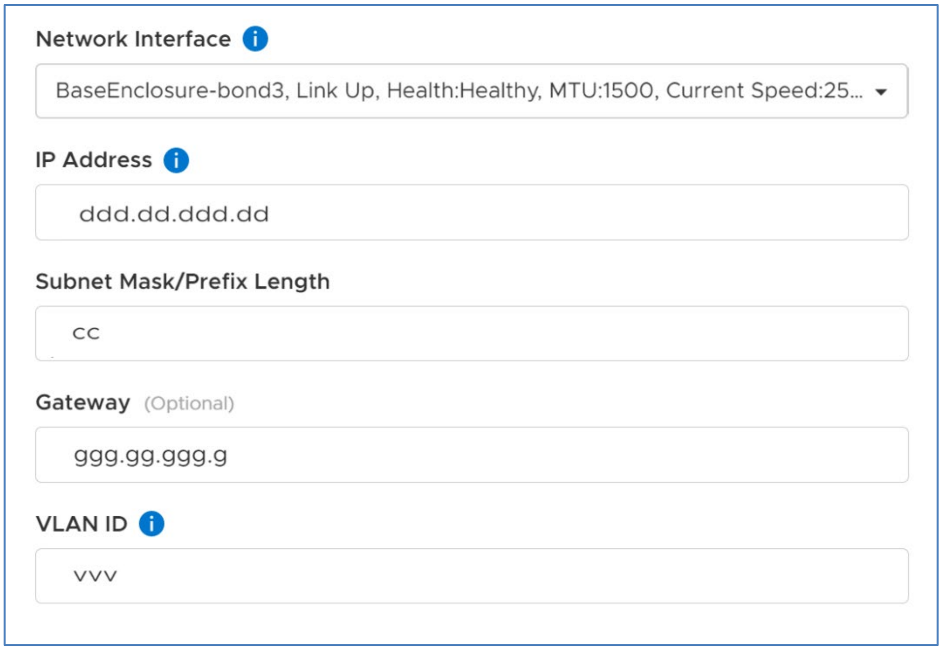
6. Set **Network Interface** to the wanted **Link Aggregate**.
7. Set the network configuration.

   Note: Dell Technologies recommends using two TOR switches for redundancy in production environments. The lab that we used for this paper is configured with a single TOR switch.

   When selecting the **Network Interface** for the NAS server, select the value of the PowerStore object **Node-Module-Name** that corresponds to the appropriate configured interface ports on PowerStore. PowerStore automatically selects and adds the corresponding bond on IoModule1 from node B.


Dashboard
Monitoring
Compute
Storag
Hardware
PS-5-appliance-1
CAPACITY
0.6%
Free
57.0 TB
Physical
57.3 TB
ETHERNET
FIBRE CHANNEL
MAP STORAGE NETWORK
LINK AGGREGATION
Node-Module-Name
BaseEnclosure-NodeA-bond0
BaseEnclosure-NodeA-4PortCard-FEPort0
BaseEnclosure-NodeA-4PortCard-FEPort1
BaseEnclosure-NodeA-bond1
BaseEnclosure-NodeA-IoModule1-FEPort0
BaseEnclosure-NodeA-IoModule1-FEPort1
BaseEnclosure-NodeA-bond2
BaseEnclosure-NodeA-4PortCard-FEPort2
BaseEnclosure-NodeA-4PortCard-FEPort3
BaseEnclosure-NodeA-bond3
BaseEnclosure-NodeA-IoModule1-FEPort2
BaseEnclosure-NodeA-IoModule1-FEPort3
port-channel1
port-channel2
port-channel3
port-channel4
port-channel5
port-channel6
port-channel6 dNFS data traffic
port-channel5 dNFS data traffic
port-channel2 dNFS control traffic
port-channel1 dNFS control traffic
port-channel3 dNFS data traffic
port-channel4 dNFS data traffic
BaseEnclosure-NodeA-bond0
BaseEnclosure-NodeA-4PortCard-FEPort0
BaseEnclosure-NodeA-4PortCard-FEPort1
BaseEnclosure-NodeA-bond1
BaseEnclosure-NodeA-IoModule1-FEPort0
BaseEnclosure-NodeA-IoModule1-FEPort1
BaseEnclosure-NodeA-bond3
BaseEnclosure-NodeA-IoModule1-FEPort2
BaseEnclosure-NodeA-IoModule1-FEPort3

NOTE: This bonded interface is configured in PowerStore but not cabled to the switch and is not used in the environment.



Network Interface

BaseEnclosure-bond3, Link Up, Health:Healthy, MTU:1500, Current Speed:25...

IP Address

ddd.dd.ddd.dd

Subnet Mask/Prefix Length

cc

Gateway (Optional)

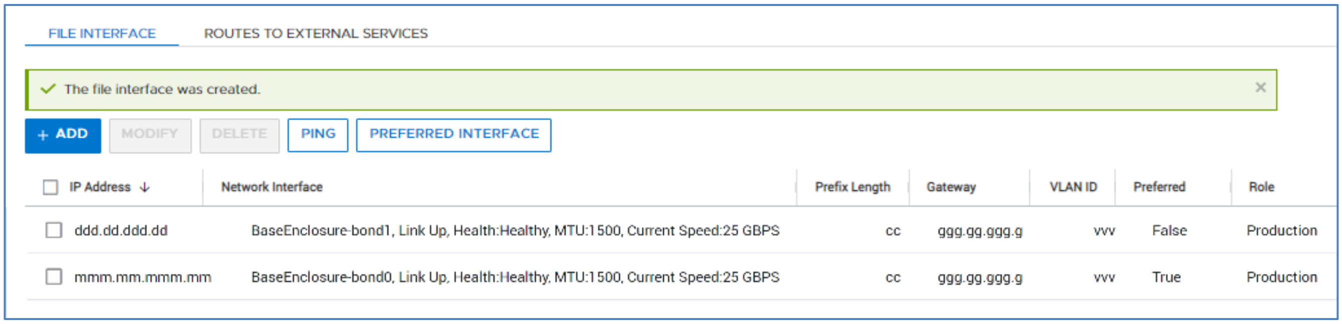
ggg.gg.ggg.g

VLAN ID

vvv

8. Click **ADD**.

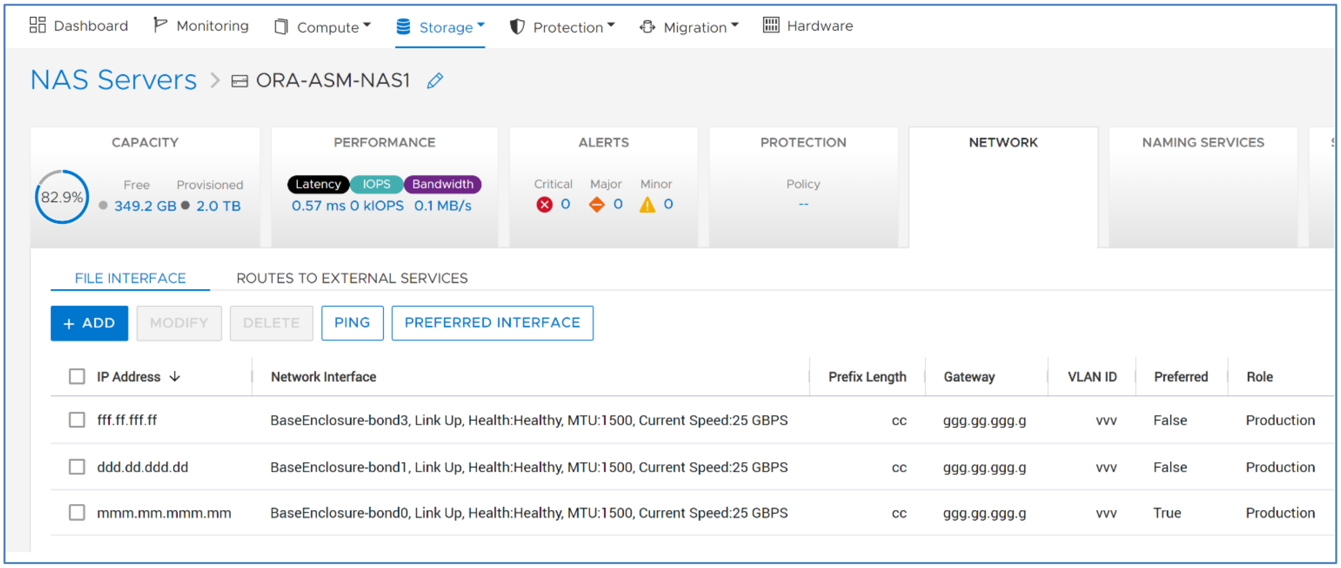
   PowerStore adds the link aggregate as a nonpreferred network interface.



✓ The file interface was created.

□	IP Address ↓	Network Interface	Prefix Length	Gateway	VLAN ID	Preferred	Role
□	ddd.dd.ddd.dd	BaseEnclosure-bond1, Link Up, Health:Healthy, MTU:1500, Current Speed:25 GBPS	cc	ggg.gg.ggg.g	vvv	False	Production
□	mmm.mm.mmm.mm	BaseEnclosure-bond0, Link Up, Health:Healthy, MTU:1500, Current Speed:25 GBPS	cc	ggg.gg.ggg.g	vvv	True	Production

The following PowerStore NAS networks were created for this paper.



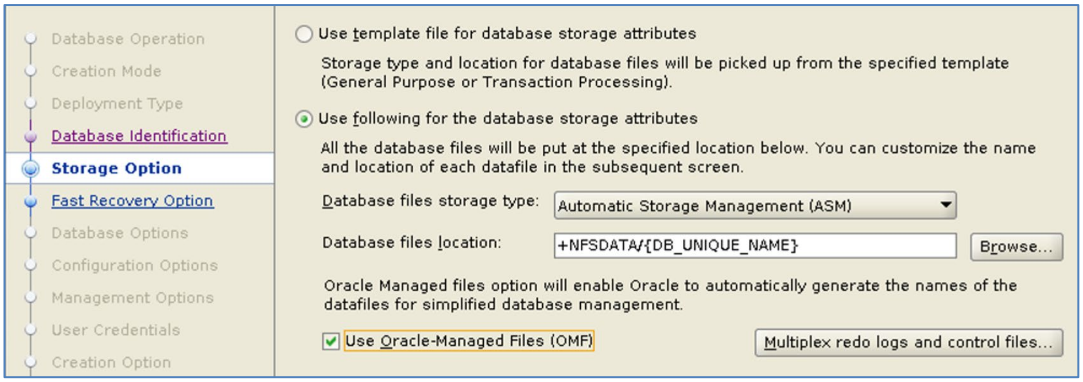
□	IP Address ↓	Network Interface	Prefix Length	Gateway	VLAN ID	Preferred	Role
□	fff.ff.fff.ff	BaseEnclosure-bond3, Link Up, Health:Healthy, MTU:1500, Current Speed:25 GBPS	cc	ggg.gg.ggg.g	vvv	False	Production
□	ddd.dd.ddd.dd	BaseEnclosure-bond1, Link Up, Health:Healthy, MTU:1500, Current Speed:25 GBPS	cc	ggg.gg.ggg.g	vvv	False	Production
□	mmm.mm.mmm.mm	BaseEnclosure-bond0, Link Up, Health:Healthy, MTU:1500, Current Speed:25 GBPS	cc	ggg.gg.ggg.g	vvv	True	Production

Figure 11. PowerStore bonded NAS network interfaces

Dell PowerStore NFS file system and NFS export

PowerStore file systems are well suited for Oracle. They provide scalability, maximum system size, flexible file system, storage efficiency, security, isolation, availability, recoverability, virtualization, and performance.

PowerStore NFS file systems can host Oracle datafiles that exist on ASM or file systems, or both.



○ Use template file for database storage attributes

Storage type and location for database files will be picked up from the specified template (General Purpose or Transaction Processing).

● Use following for the database storage attributes

All the database files will be put at the specified location below. You can customize the name and location of each datafile in the subsequent screen.

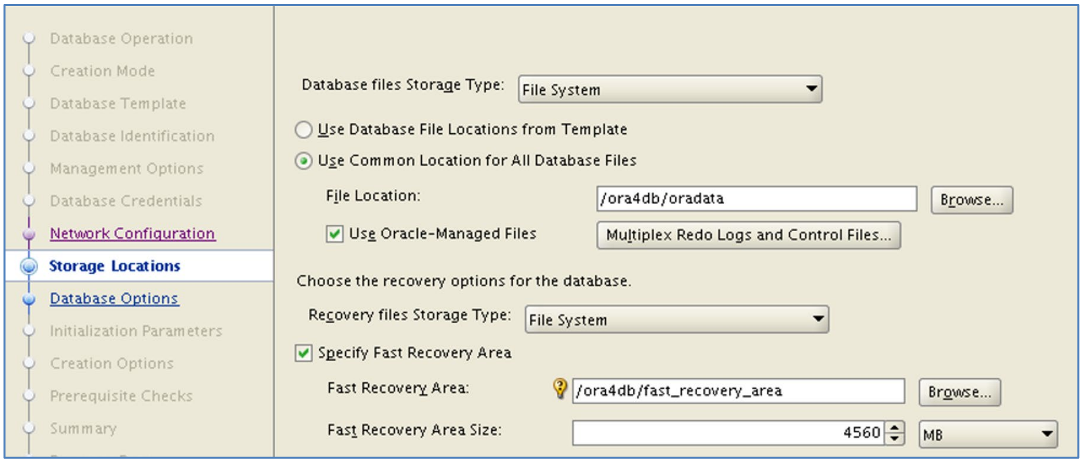
Database files storage type: Automatic Storage Management (ASM)

Database files location: +NFSDATA/{DB_UNIQUE_NAME}

Oracle Managed files option will enable Oracle to automatically generate the names of the datafiles for simplified database management.

☑ Use Oracle-Managed Files (OMF)

Figure 12. Using ASM disk group on NFS file system for Oracle datafiles



Database files Storage Type: File System

○ Use Database File Locations from Template

● Use Common Location for All Database Files

File Location: /ora4db/oradata Browse...

☑ Use Oracle-Managed Files Multiplex Redo Logs and Control Files...

Choose the recovery options for the database.

Recovery files Storage Type: File System

☑ Specify Fast Recovery Area

Fast Recovery Area: /ora4db/fast_recovery_area Browse...

Fast Recovery Area Size: 4560 MB

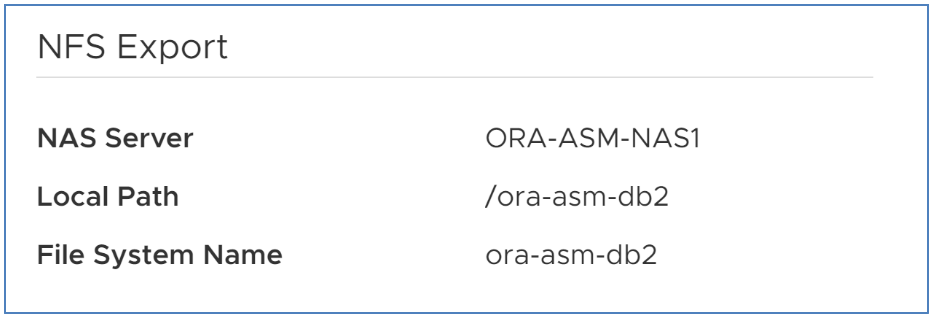
Figure 13. Using file system for Oracle datafiles

To create a PowerStore NFS file system, perform the following steps in PowerStore:

1. Select **Storage** > **File Systems**.
2. Click **+ CREATE**.
3. For **Select Type**, click **General.**
4. Select the checkbox that corresponds to the NAS server that will service the file system.
5. Click **NEXT**.
6. Provide the name, size, and optional description for the file system.
7. Click **NEXT**.
8. Ensure that **File-level Retention** is set to **Off**.

Note: NAS file systems used by Oracle databases should not use file-level retention (FLR). FLR prevents the modification or deletion of locked files until a specified retention date is reached. Once the retention date is reached, the files on the file system can either be deleted or relocked. The files cannot be modified or changed to append only. This type of FLR is known as Write Once, Read Many (WORM).

9. Click **NEXT**.
10. Provide a name for the file system and an optional description.



NFS Export

NAS Server	ORA-ASM-NAS1
Local Path	/ora-asm-db2
File System Name	ora-asm-db2

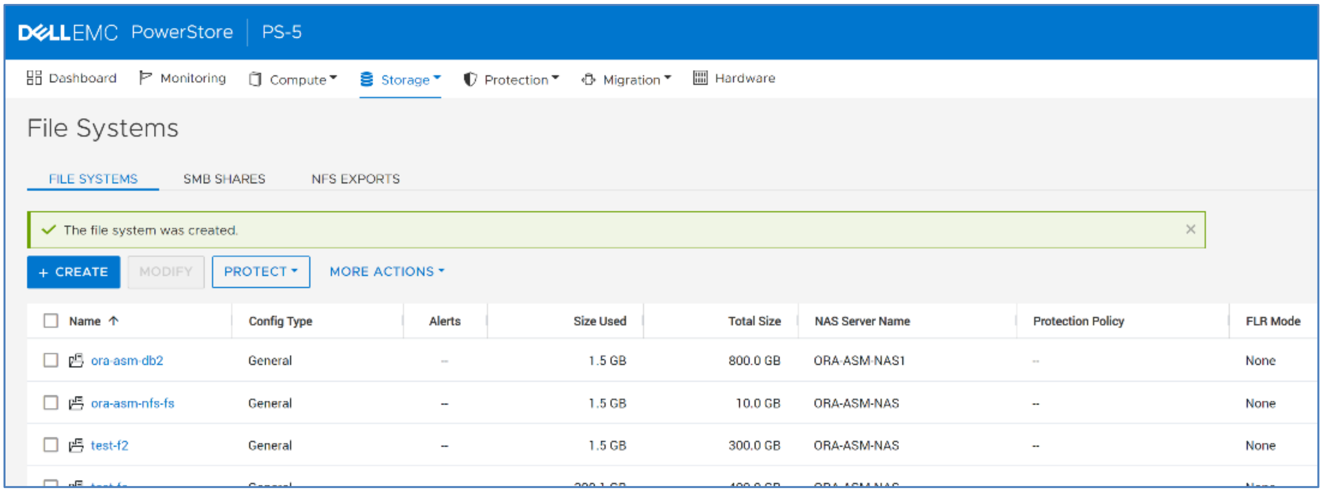
11. Click **NEXT.**

12. When prompted to configure access, click **NEXT**.

    By default, the file system will be exported such that all NFS clients can see it.

13. If the NFS export needs to be restricted to a specific set of NFS clients, click **+ ADD HOST**, and then add the list of NFS clients where the NFS file system should be exported.
14. Set **Default Access** to **Read/Write, allow Root**, and ensure that **Minimum Security** is set to **Sys**.
15. Click **NEXT.**
16. If applicable, apply a protection policy.
17. Click **NEXT.**
18. Review the request to create the file system.
19. If the request is correct, click **CREATE FILE SYSTEM**.

    PowerStore Manager adds the new file system.



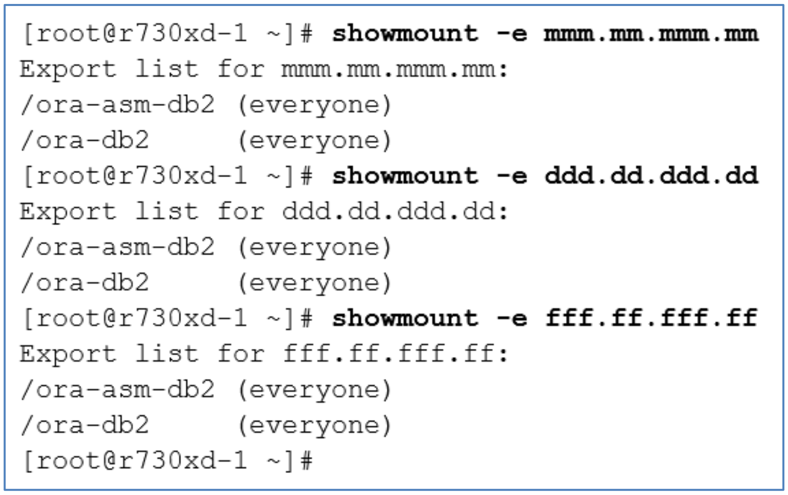
✓ The file system was created.

□	Name ↑	Config Type	Alerts	Size Used	Total Size	NAS Server Name	Protection Policy	FLR Mode
□	ora-asm-db2	General	--	1.5 GB	800.0 GB	ORA-ASM-NAS1	--	None
□	ora-asm-nfs-fs	General	--	1.5 GB	10.0 GB	ORA-ASM-NAS	--	None
□	test-f2	General	--	1.5 GB	300.0 GB	ORA-ASM-NAS	--	None

Verify access to the PowerStore NFS export

After PowerStore file storage has been configured for the NFS client, verify that the database server has access to the export through all the IP addresses (see Figure 11) defined for the NFS share. To verify access, use the `showmount` command in Linux on all the IP addresses shown in the list of Exported Paths. If any of the IP addresses do not have access to the NFS export, resolve the issue before configuring the NFS client, including configuring Oracle dNFS.

Log in as root to the NFS client and perform `showmount` using all the IP addresses to verify that the client can see the NFS export and file system:



```
[root@r730xd-1 ~]# showmount -e mmm.mm.mmm.mm
Export list for mmm.mm.mmm.mm:
/ora-asm-db2 (everyone)
/ora-db2     (everyone)
[root@r730xd-1 ~]# showmount -e ddd.dd.ddd.dd
Export list for ddd.dd.ddd.dd:
/ora-asm-db2 (everyone)
/ora-db2     (everyone)
[root@r730xd-1 ~]# showmount -e fff.ff.fff.ff
Export list for fff.ff.fff.ff:
/ora-asm-db2 (everyone)
/ora-db2     (everyone)
[root@r730xd-1 ~]#
```

Figure 14. Verifying NAS exports are accessible

NFS protocol

PowerStore storage supports NFSv3 through NFSv4.1, including secure NFS.

NFSv4 is a version of the NFS protocol that is considerably different from previous implementations. NFSv4 is a stateful protocol, meaning that it maintains a session state and does not treat each request as an independent transaction without the need for additional preexisting information.

While PowerStore storage fully supports most of the NFSv4 and NVSv4.1 capabilities, directory delegation and parallel NFS (known as pNFS) are not supported. Therefore, do not configure Oracle to use pNFS. For increased performance, consider using NFSv4 and dNFS with multiple network interfaces for load-balancing purposes.

Note: pNFS and dNFS are different. pNFS refers to a protocol to access a NFS server, while dNFS refers to a driver built into the Oracle Disk Manager (ODM).

PowerStore file system and Oracle ASM disks

If you plan to place storage on NFS protocol devices, Oracle recommends using Oracle Direct NFS (dNFS) rather than native kNFS.

Note: According to Oracle, caution should be exercised when considering using NFS storage in an Oracle database that uses ASM. In general, such configurations should be avoided or at least carefully considered. The issue is that to prevent data corruption, NFS file systems must be hard-mounted. Hard mounting means that the database or an ASM instance must wait indefinitely when an NFS server, or the communications link it operates over, becomes unavailable. NFS on ASM might be reasonable in some situations. See Oracle MOS note doc ID 1620238.1, "Creating File Devices On NAS/NFS FileSystems for ASM Diskgroups."

To create ASM disks from PowerStore file systems, follow these steps:

1. In PowerStore Manager, create the PowerStore NAS export.
2. Log in to the NFS client (database server) as root.
3. Create a mount point for the NAS export. Change the owner, group, and privileges of the mount point to the Linux user owning the Oracle Grid software install.

```
mkdir /ora-asm-db2
chown grid:oinstall /ora-asm-db2
chmod 660 /ora-asm-db2
```

4. Mount the PowerStore export using the IP address in PowerStore Manager with a **Preferred** status of **True**. See Figure 11.

```
mount -t nfs -o rw,bg,hard,nointr,tcp,vers=3,\
timeo=600,rsize=32768,wsize=32768,\
actimeo=0 mmm.mm.mmm.mm:/ora-asm-db2 /ora-asm-db2
```

5. The first time the NFS share is mounted, change the owner, group, and privileges of the root directory on the NFS export to the Linux user owning the Oracle Grid software install:

```
chown grid:oinstall /ora-asm-db2
chmod 660 /ora-asm-db2
```

6. Create the raw files to be used as ASM disks and set their permissions and ownership to the operating system user owning the Oracle Grid software:

```
dd if=/dev/zero of=/ora-asm-db2/asm-data1 bs=1024k count=100000 oflag=direct
dd if=/dev/zero of=/ora-asm-db2/asm-fra1 bs=1024k count=200000 oflag=direct
dd if=/dev/zero of=/ora-asm-db2/asm-redo1 bs=1024k count=50000 oflag=direct
chown grid:oinstall /ora-asm-db2/asm*
chmod 660 /ora-asm-db2/asm*
```

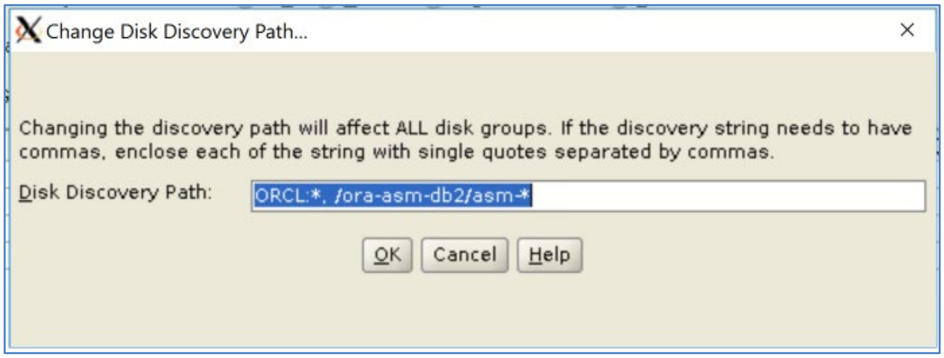
Creating ASM disk groups using NFS

ASM disk groups can be created using NFS files from multiple NAS servers. Spanning across multiple NAS servers might provide improved load balancing and flexible capacity planning. If snapshots are needed, use the same NAS server for all the disk groups containing the database to be snapped. Snapshots are not supported across multiple NAS servers.

To create ASM disk groups from ASM disks residing on PowerStore NFS exports, follow these steps.

1. Add the location and name of the ASM disks created in the previous procedure to the current value of ASM's disk string.

   The location used in this paper is: `/ora-asm-db2/asm-*`. This example uses `asmca` to change the disk string:

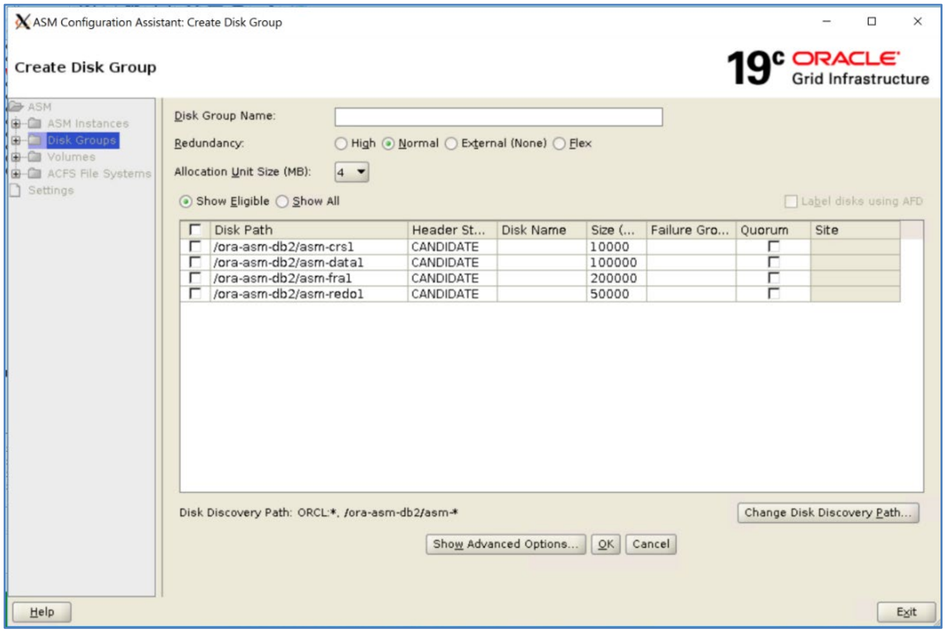


Change Disk Discovery Path...

Changing the discovery path will affect ALL disk groups. If the discovery string needs to have commas, enclose each of the string with single quotes separated by commas.

Disk Discovery Path: ORCL:*, /ora-asm-db2/asm-*

After changing the disk string value, `asmca` displays the available disks created in a previous step



Disk Group Name:

Redundancy: ○ High ● Normal ○ External (None) ○ Flex

Allocation Unit Size (MB): 4

● Show Eligible ○ Show All   □ Label disks using AFD

□ Disk Path	Header St...	Disk Name	Size (...	Failure Gro...	Quorum	Site
□ /ora-asm-db2/asm-crs1	CANDIDATE		10000		□	
□ /ora-asm-db2/asm-data1	CANDIDATE		100000		□	
□ /ora-asm-db2/asm-fra1	CANDIDATE		200000		□	
□ /ora-asm-db2/asm-redo1	CANDIDATE		50000		□	

Disk Discovery Path: ORCL:*, /ora-asm-db2/asm-*

2. Create the ASM disk groups.

   `asmca` can be used to create the ASM disk groups, or the groups can be created using SQL*Plus.

```
SQL> create diskgroup data external redundancy disk
  2    '/ora-asm-db2/asm-data1';

SQL> create diskgroup fra external redundancy disk
  2    '/ora-asm-db2/asm-fra1';

SQL> create diskgroup redo external redundancy disk
  2    '/ora-asm-db2/asm-redo1';
```

Oracle Disk Manager

Introduction

The Oracle Disk Manager (ODM) library `$ORACLE_HOME/lib/libodm19.so` manages all Oracle I/O activity and file management. ODM can also use NFS devices for database I/O without using the native Linux NFS kernel (kNFS). To have ODM use NFS devices, the embedded Oracle NFS client (`$ORACLE_HOME/rdbms/lib/odm/libnfsodm19.so`) must be enabled.

Note: The version number of Oracle is part of the ODM library and Oracle NFS client file names. For example, in Oracle 19c, the respective ODM library and Oracle NFS client names are:

```
$ORACLE_HOME/lib/libodm19.a
$ORACLE_HOME/lib/libodm19.so
$ORACLE_HOME/lib/libodmd19.so
$ORACLE_HOME/rdbms/lib/odm/libnfsodm19.so
```

NFS traffic

Generally, NFS traffic can be classified as either control/management traffic or application data I/O traffic. Concerning the operating system, regardless of whether the Oracle ODM NFS client library is enabled, the native NFS kernel client (kNFS) driver manages all NFS control/management traffic of NFS devices. When the ODM library containing the embedded Oracle NFS client is enabled, the Oracle environment is said to be using dNFS for all database I/O. The database I/O is considered NFS data traffic and flows through the dNFS driver. When the ODM library containing the embedded Oracle NFS client is disabled, all database I/O flows through the kNFS client driver.

Some examples of NFS control and management activity involve the following operations on the NFS share:

- get attribute
- set attribute
- access
- create
- mkdir
- rmdir
- mount
- umount

NFS supported storage option

NFS is a supported storage option for Oracle and can be used for:

- Oracle clusterware binaries
- OCR and voting files
- Oracle Standalone and RAC database binaries
- Oracle Standalone and RAC database datafiles
- Oracle database recovery files

Note: Oracle Direct NFS (dNFS) does not support Oracle clusterware files (voting disks and OCR). Also, because dNFS does not support soft mounts, dNFS does not support quorum disks (quorum failure groups). For quorum disks, use an NFS soft mount. The quorum failure group feature can be used in an Oracle ASM disk group without requiring a hard mount for NFS storage. With hard mounts, ASM or the database might stop responding if the NFS server becomes unavailable.

Oracle ADVM, Oracle ACFS, and ASMFD are not supported with NFS. However, NFS disks can be used with Oracle ACFS NAS Maximum Availability eXtension.

Oracle Direct NFS

Overview

Oracle Direct NFS (dNFS) is an optimized NFS client from Oracle for database I/O and resides in the ODM library as a part of the Oracle database kernel. dNFS improves the stability and reliability of NFS storage devices over TCP/IP, more so than the native Linux NFS driver (kNFS). dNFS also improves performance to NFS storage devices by bypassing the kNFS I/O stack. When mounting NFS database datafile, Oracle first loads dNFS functionality if the Direct NFS client ODM library is enabled. If dNFS cannot access an NFS storage device, dNFS silently reverts to using the kNFS client. However, to ensure that this reversion occurs, kNFS client mount options `rsize` and `wsize` must be used.

Benefits of dNFS

The advantage of using Oracle dNFS stems from the fact that it is part of the Oracle database kernel. When dNFS is enabled, all application data I/O to the NFS storage devices is through dNFS rather than kNFS.

dNFS:

- Provides the ability to manage the best possible configuration
- Automatically tunes NFS data traffic for performance optimizations
- Takes advantage of the Oracle buffer cache
- Appropriately uses available resources for optimal NFS data traffic, without the overhead of the client operating system kernel software

Oracle recommends using dNFS if NFS storage devices are used so these benefits can be exploited.

Creating NFS client mount points

An Oracle installation prompts for directory locations for storing the software and components. Usually, these directories can reside on NFS shares. For more information, see Oracle documentation.

The following table shows examples of different Oracle directories that could reside on NFS shares:

Table 7. Example of directories for use with NFS

Oracle directory	Environment variables and typical values	Description
Oracle base	`$ORACLE_BASE=/u01/app/oracle/`	The top-level directory for installations. Subsequent installations can either use the same Oracle base or a different one.

Oracle inventory	`/u01/app/oraInventory/`
	or
	`$ORACLE_BASE/<srv>/oraInventory/`
		All Oracle installations use the same Oracle inventory directory for the installation repository metadata. If possible, Oracle recommends that the inventory directory resides on a local file system.
		`/u01/app/oraInventory`
		If a NAS device must be used for the inventory, to prevent multiple systems from writing to the same inventory, create a unique directory for each database server.
		`$ORACLE_BASE/<srv>/oraInventory`

Oracle home	`$ORACLE_HOME=$ORACLE_BASE/product/<versions>/dbhome_1/`	This directory contains the binaries, library, configuration files, and other files from a single release of one product and cannot be shared with other releases or other Oracle products.

Database file directory	`$ORACLE_BASE/oradata/`
	or
	`/<PowerStore-NFS-shrnm>/`
		Location to hold the database. Using a different NFS mount point for database files is recommended to:
		• Isolate specific mount options to the database.
		• Distribute database I/O without impact from other uses of the mount point.
		For better performance, Oracle recommends that Oracle software does not reside on the same disk as the database.

Oracle recovery directory	`$ORACLE_BASE/fast_recovery_area/`	Oracle recommends that recovery files and database files do not exist on the same file system.

Oracle product directory	`$ORACLE_BASE/product/`	This mount point can be used to install software from different releases, for example:
		`/u01/app/oracle/product/18c/dbhome_1/`
		`/u01/app/oracle/product/19c/dbhome_1/`

Oracle release directory	`$ORACLE_BASE/product/<version>/`	This mount point can be used to install different Oracle products from the same version. For example:
		`$ORACLE_BASE/product/19c/dbhome_1`
		`$ORACLE_BASE/product/19c/client_1`
		Although an option, it is not recommended to install both the RDBMS and client on the database server. If the client is required, defining a separate NFS share and using a nondatabase server to host the client installation is recommended.

After determining what NFS shares Oracle will use, create the NFS shares in PowerStore. Once the NFS shares are created in PowerStore, create the mount points in Linux for the NFS shares. Set the privileges, owner, and group of the Linux mount points for the NFS shares per Oracle requirements. For information about mounting the PowerStore NFS export for Oracle, see PowerStore file system and Oracle ASM disks.

The environment used for this paper used only one NFS share (ora-asm-db2) to host a single Standalone Oracle database.

Mount options for an NFS share

Before the dNFS driver is configured or used on an NFS share, the NFS share must first be mounted using the kNFS driver. Specific mounting options are required when mounting an NFS share for Oracle usage. If the NFS export will be used for Oracle services that need to be automatically restarted when the server restarts, add the NFS export and mount options to `/etc/fstab`. If the NFS share and mount options are not specified in `/etc/fstab`, Oracle will experience issues. In an Oracle RAC cluster, ensure that all nodes in the cluster use the same mount options and `/etc/fstab` entry for each identical NFS share mount point.

After the NFS share is mounted using kNFS, dNFS mounts and unmounts the NFS share logically as needed. Because dNFS uses a logical mount, after Oracle unmounts the NFS share, the NFS share is still physically mounted to the server and can be accessed through kNFS.

If NFS is used for database files, the NFS buffer size for reads (`rsize`) and writes (`wsize`) must be set to at least 16,384. Oracle recommends a value of 32,768. These values are set in `/etc/fstab` or when explicitly mounting an NFS share. Because a dNFS write size (`v$dnfs_servers.wtmax`) of 32,768 or larger is supported in PowerStore, dNFS does not fall back to the traditional kNFS kernel path. dNFS clients issue writes with (`v$dnfs_servers.wtmax`) granularity to the PowerStore NAS server.

The required mount options for NFS share mount points used by Oracle are:

- Mount options for binaries (ORACLE_HOME, CRS_HOME) and database files[1,2]:

```
rw,bg,hard,nointr,rsize=32768,wsize=32768,tcp,nfsvers={3|4},
timeo=600,actimeo=0
```

[1] The mount options are applicable only if ORACLE_HOME is shared. Oracle also recommends that the Oracle inventory directory is kept on a local file system. If it must be placed on a NAS device, create a specific directory for each system to prevent multiple systems from writing to the same inventory directory. Oracle clusterware is not certified on dNFS.

[2] Do not replace `tcp` with `udp`. UDP should never be used. dNFS cannot serve an NFS server with write size less than `32768`. PowerStore supports NFS versions 3 and 4. Therefore, `nfsvers={3|4}` indicates that nfsvers should be set to only one value (nfsvers=3 or nfsvers=4) when mounting the NFS share. Mount option `vers` is identical to mount option `nfsvers` and is provided in Oracle Linux for compatibility reasons. Set option `vers` to either 3 or 4 and ensure that the NFS sharing protocol on PowerStore NAS server is set accordingly. Starting with Oracle 12cR2, both OCR and voting disks must reside in ASM. For more information, see Oracle MOS note 2201844.1. dNFS is RAC aware. dNFS automatically recognizes RAC instances and takes appropriate action for datafiles without additional user configuration. This eliminates the need to specify `noac` when mounting NFS file systems for Oracle datafiles or binaries. This exception does not pertain to CRS voting disks or OCR files on NFS. NFS file systems hosting CRS voting disks and OCR files must be mounted with `noac`. Do not use option `noac` for RMAN backup set, image copies, and data pump dump files. RMAN and data pump do not check for `noac`, and specifying it can adversely affect performance.

- Mount options for CRS voting disk and OCR[2]:

  ```
  rw,bg,hard,nointr,rsize=32768,wsize=32768,tcp,nfsvers={3|4},
  timeo=600,actimeo=0,noac
  ```

Note: Mount option `actimeo=0` is required only for Oracle RAC and certain Oracle components (for example: GoldenGate binaries and datafiles). For more information, see Oracle documentation or Oracle Support MOS notes.

When configuring an Oracle RAC environment that uses NFS, ensure that the entry in `/etc/fstab` is the same on each node. The following snippet from `/etc/fstab` mounts an NFS mount point for ORACLE_HOME binaries (`/u01`), and a RAC database that will use ASM. These `/etc/fstab` entries were not used for the creation of this document.

```
zzz.zz.zzz.zz:/ora-bin        /u01           nfs
rw,bg,hard,nointr,rsize=32768,wsize=32768,tcp,vers=3,timeo=600,act
imeo=0,defaults 0 0

uuu.uu.uuu.uu:/ORA-ASM-NFS    /oraasmnas     nfs
rw,bg,hard,nointr,rsize=32768,wsize=32768,tcp,vers=3,timeo=600,act
imeo=0,defaults 0 0
```

When adding multiple mount options for a specific mount point in `/etc/fstab`, do not insert spaces after options because the operating system might not properly parse the options.

Mount options `timeo`, `hard`, `soft`, and `intr` control the NFS client behavior if the NFS server becomes temporarily unreachable. Whenever the NFS client sends a request to the NFS server, it expects the operation to have finished after a given interval (specified in the `timeout` option). If no confirmation is received within this time, a minor timeout occurs, and the operation is retried with the timeout interval doubled. After a maximum timeout of 60 s is reached, a major timeout occurs. By default, a major timeout causes the NFS client to print a message to the console and start over with an initial timeout interval twice that of the previous cascade. Potentially, this action could be repeated indefinitely. Volumes that retry an operation until the server becomes available again are called hard-mounted.

For additional mount options and information about using NFS shares intended for Oracle, see the Oracle MOS notes listed in References.

Ethernet networks and dNFS

Performance of Oracle on NAS devices depends in part on network performance between the NFS client and server. Therefore, consider whether to use a dedicated interface for NFS control and NFS data traffic.

If NFS and network redundancy is a concern, consider creating a bonded interface for the kNFS control traffic on the NFS client. All PowerStore NAS network interfaces are created as bonded interfaces. On the database server, this bonded interface could be a bonded public network or even the bonded interface for the RAC interconnect in a RAC environment.

Using bonded interfaces or directing NFS control and data traffic to different NICs might not always be possible because of a limited number of NICs or infrastructure limitations. In such cases, using a single unbonded interface for both NFS control and data traffic is possible. However, that configuration might cause network performance issues under heavy loads because the server will not perform network load balancing and there will be no redundancy for NFS control traffic.

When using dNFS, Oracle supports one to five network paths for NFS traffic between a NAS server and NFS client—one path for NFS control traffic and up to four paths for NFS data traffic. When employing dNFS, using multiple network paths is recommended. Ensure that each NFS network path belongs to a subnet that is not being used for any other NIC interface on the NFS client (database server). This includes not using the subnet of the public network for NFS. Using unique subnets simplifies configuration of dNFS.

Sometimes, fewer available subnets exist than are needed for dNFS paths. If so, dNFS paths on the NFS client can be set up to use existing subnets already in use on the NFS client. Using shared subnets requires additional configuration in the operating system network layer and in Oracle. For the operating system, ingress filtering must be relaxed for multihomed networks, and static routes must be defined. For Oracle, file `oranfstab` must be created if dNFS data traffic will flow across paired endpoint IP addresses in the same shared subnet. This extra configuration prevents the operating system from using the default route and allows multiple NIC interfaces in the same server to use the same subnet.

If the operating system chooses the dynamic route, it will invariably use the first best-matched route possible from the routing table for all paths defined. Usually, that route will be incorrectand result in no dNFS load balancing, no NFS traffic scalability, and dNFS path failover not working as expected. Therefore~~,~~ create file `oranfstab` and configure static operating system routing and for each dNFS network path to ensure that load balancing, scalability, and path failover work as expected.

If different subnets are used for NFS traffic, routing is taken care of automatically by the native network driver and the default route entries in the routing table. Static routes are not necessary when different subnets are used for dNFS traffic.

All paired end-point IP addresses must be defined in the Oracle file `oranfstab` when dNFS data traffic is being isolated to one or more dedicated paths. This requirement applies to all environments, whether the same or different subnets are used to define multiple paths for dNFS data traffic.

Jumbo frames

Using jumbo frames end-to-end (database server, Ethernet switches, and PowerStore) in the infrastructure is supported and recommended by Oracle when dNFS is used. Using jumbo frames allows the network stack to bundle transfers into larger frames and reduce the TCP overhead. The value used for any frame can vary depending on the immediate needs of the network session established between the NFS client and NFS server. Raising the limit from end point to end point in the network path allows the session to take advantage of a wider range of frame sizes. MTU set to `9000` is a good starting value, but it should be evaluated.

Single network path for dNFS

The following figure shows a single network path for dNFS between the PowerStore NAS server and NFS client (database server). NFS control/management and data traffic will use this path.

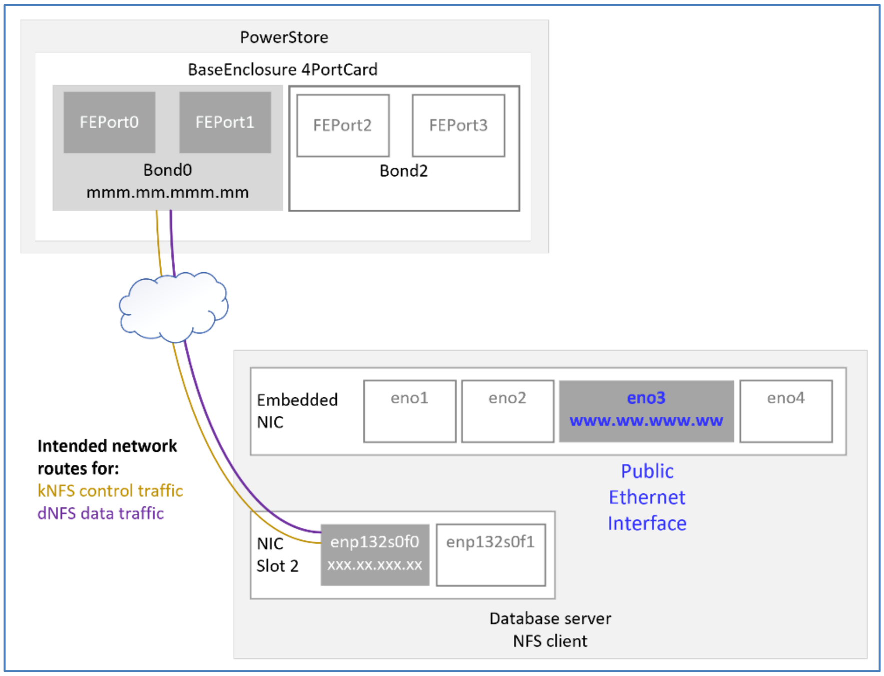
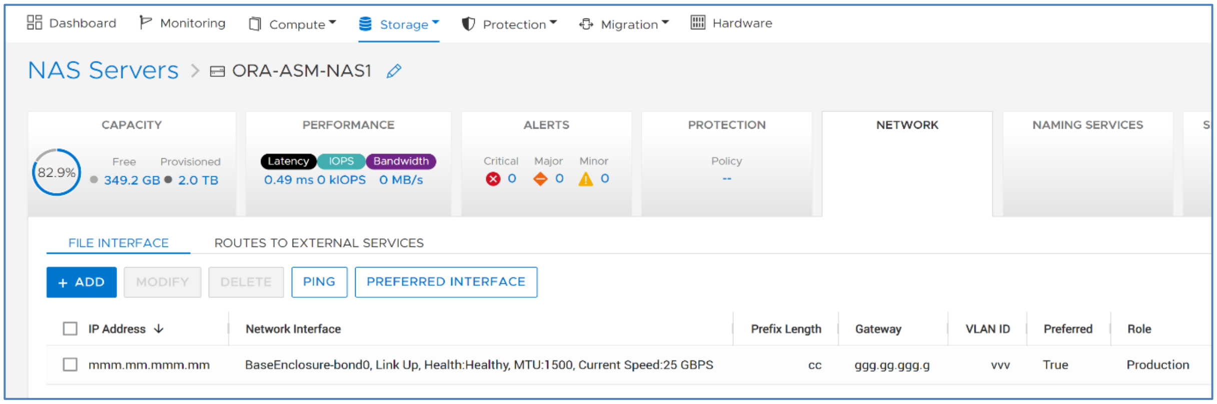


Figure 15. Network route for dNFS traffic between the NFS client and PowerStore NAS server

Note: Figure 15 is not a cabling diagram. It represents a network route between a PowerStore bonded interface and an interface on the database server that is intended to be used by NFS control and data traffic. For more information about creating network routes and using specific network routes for certain network traffic, see the following sections: Shared subnets, Static routing, dNFS without oranfstab, and dNFS with oranfstab.

The interfaces shown in Figure 15 were cabled to switches, as shown by the diagram in step 7 of Adding link aggregate network interfaces to a PowerStore NAS server and in Figure 15.

The following figures show the corresponding NAS server, file system, NFS export, and NFS client interface configuration using this path:



NAS Servers > ORA-ASM-NAS1

CAPACITY: 82.9%, Free 349.2 GB, Provisioned 2.0 TB

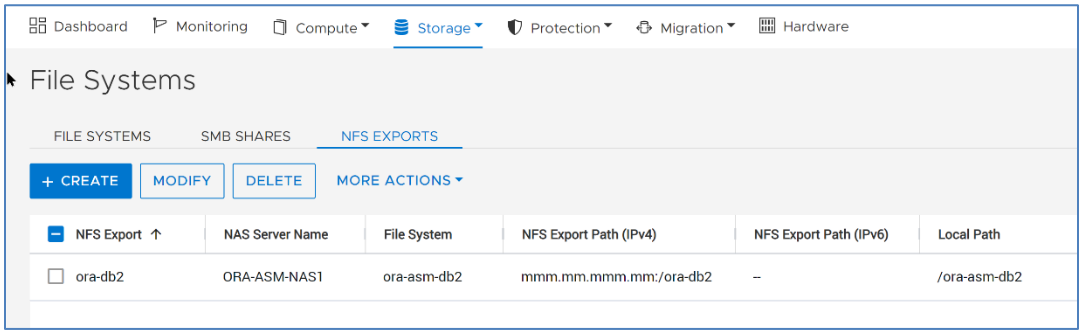
PERFORMANCE: Latency 0.49 ms, IOPS 0 kIOPS, Bandwidth 0 MB/s

ALERTS: Critical 0, Major 0, Minor 0

PROTECTION: Policy --

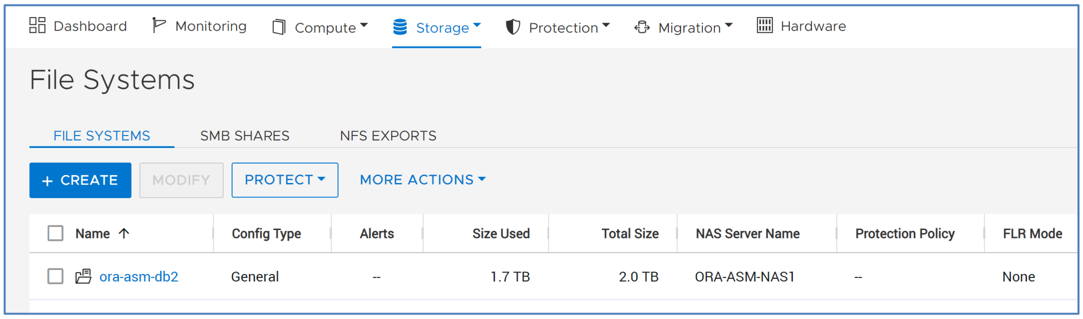
☐	IP Address ↓	Network Interface	Prefix Length	Gateway	VLAN ID	Preferred	Role
☐	mmm.mm.mmm.mm	BaseEnclosure-bond0, Link Up, Health:Healthy, MTU:1500, Current Speed:25 GBPS	cc	ggg.gg.ggg.g	vvv	True	Production

Figure 16. PowerStore NAS server with a single bonded Ethernet network for NFS traffic



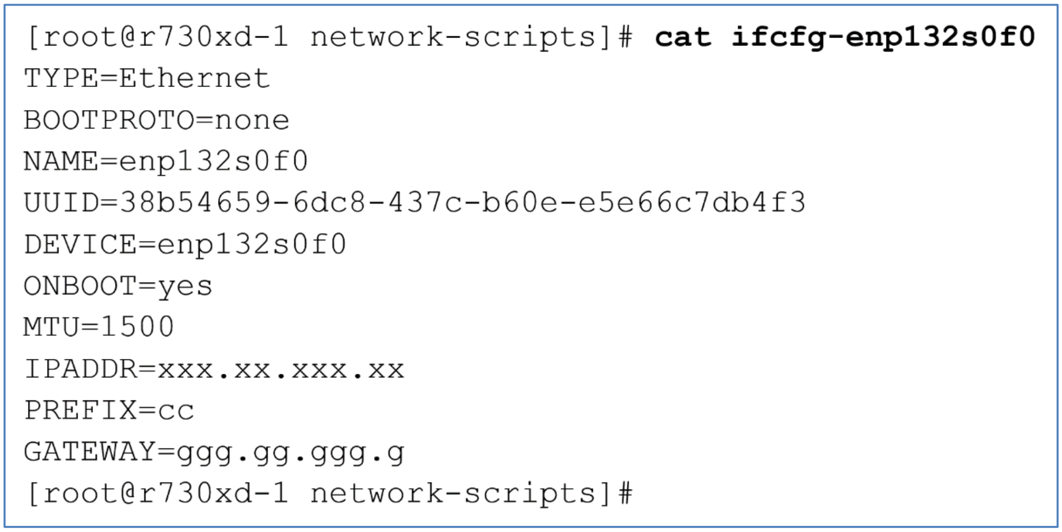
NFS Export ↑	NAS Server Name	File System	NFS Export Path (IPv4)	NFS Export Path (IPv6)	Local Path
ora-db2	ORA-ASM-NAS1	ora-asm-db2	mmm.mm.mmm.mm:/ora-db2	--	/ora-asm-db2

Figure 17. PowerStore NFS export



Name ↑	Config Type	Alerts	Size Used	Total Size	NAS Server Name	Protection Policy	FLR Mode
ora-asm-db2	General	--	1.7 TB	2.0 TB	ORA-ASM-NAS1	--	None

Figure 18. PowerStore file system



```
[root@r730xd-1 network-scripts]# cat ifcfg-enp132s0f0
TYPE=Ethernet
BOOTPROTO=none
NAME=enp132s0f0
UUID=38b54659-6dc8-437c-b60e-e5e66c7db4f3
DEVICE=enp132s0f0
ONBOOT=yes
MTU=1500
IPADDR=xxx.xx.xxx.xx
PREFIX=cc
GATEWAY=ggg.gg.ggg.g
[root@r730xd-1 network-scripts]#
```

Figure 19. NFS client Ethernet interface configuration

dNFS uses two kinds of NFS mounts: the native operating system mount of NFS (known as a kernel kNFS mount) and the Oracle database NFS mount (dNFS mount). When a single network path for dNFS is used, file `oranfstab` is not necessary because Oracle dNFS gleans the required information for the matching mounted NFS share in file `/etc/mtab`. If dNFS is unable to find the necessary information in `/etc/mtab`, control is handed back to the database, and file access is attempted through kNFS.

```
[oracle@r730xd-1 ~]$ grep ora-asm-db2 /etc/mtab
mmm.mm.mmm.mm:/ora-asm-db2 /ora-asm-db2 nfs
rw,relatime,vers=3,rsize=32768,wsize=32768,namlen=255,acregmin=0,a
cregmax=0,acdirmin=0,acdirmax=0,hard,proto=tcp,timeo=600,retrans=2
,sec=sys,mountaddr=mmm.mm.mmm.mm,mountvers=3,mountport=1234,mountp
roto=tcp,local_lock=none,addr= mmm.mm.mmm.mm 0 0
[oracle@r730xd-1 ~]$
```

Figure 20. mtab entry for the NFS export

Multiple network paths for dNFS

If multiple network paths are intended for NFS traffic, consider using one path for NFS control/management traffic and the remaining NFS paths for NFS data traffic. This ensures that the NFS data path is only used for NFS data traffic.

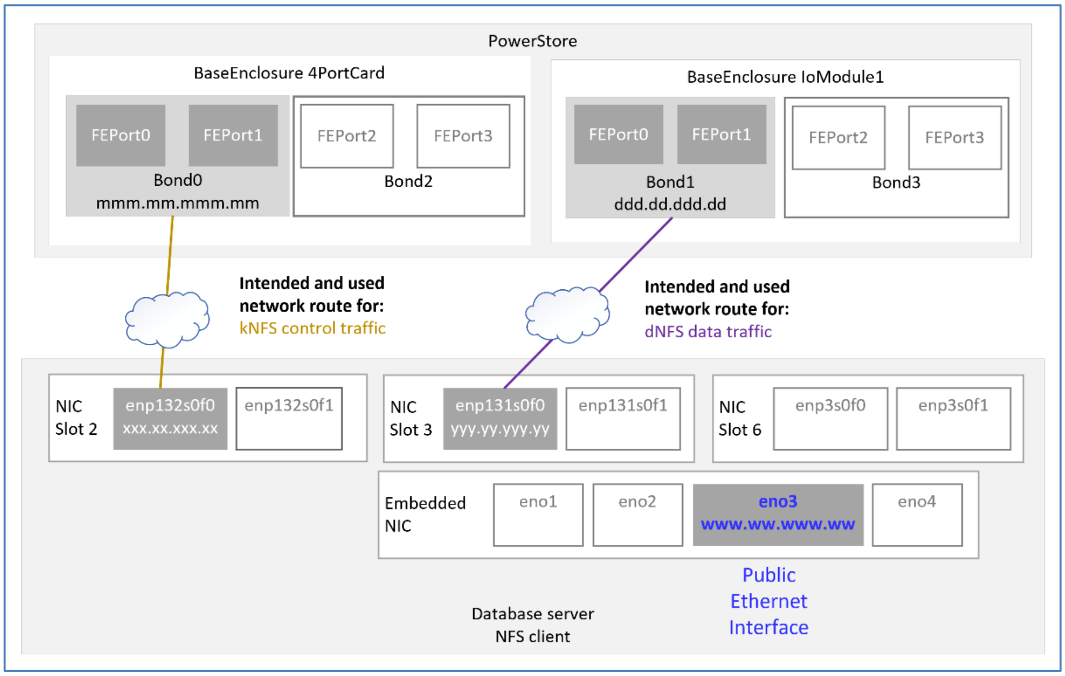
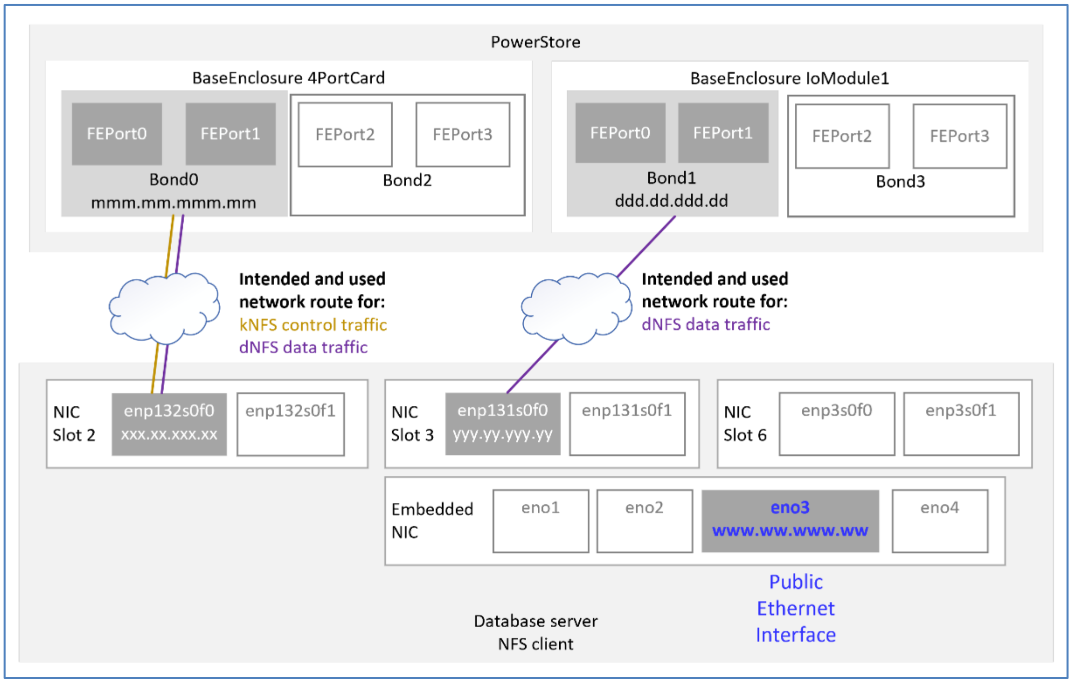


Figure 21. Network routes using dedicated interfaces for NFS control and data traffic

Note: The preceding figure is not a cabling diagram. It represents network routes between PowerStore bonded interfaces and unbonded interfaces on the database server that are intended to be used by NFS control and data traffic. For more information about creating network routes and using specific network routes for certain network traffic, see the following sections: Shared subnets, Static routing, dNFS without oranfstab, and dNFS with oranfstab.

In the figure, the interfaces (both bonded and unbonded) are cabled to switches, as shown in the network configuration step in Adding link aggregate network interfaces to a PowerStore NAS server and in Figure 15.



Network routes using shared and dedicated interfaces for dNFS data traffic

Note: The preceding figure is not a cabling diagram. It represents network routes between PowerStore bonded interfaces and unbonded interfaces on the database server that are intended to be used by NFS control and NFS data traffic. For more information about creating network routes and using specific network routes for certain network traffic, see the following sections: Shared subnets, Static routing, dNFS without oranfstab, and dNFS with oranfstab.

In the figure, the interfaces (both bonded and unbonded) are cabled to switches, as shown in the network configuration step in Adding link aggregate network interfaces to a PowerStore NAS server and in Figure 15.

If multiple dNFS paths are defined for data traffic, when a dNFS data path fails, dNFS reissues requests over any of the remaining dNFS data paths, improving database availability. Multiple data paths also provide Oracle the ability to automatically tune the paths to the NFS storage devices, and manually tuning NFS is unnecessary. Because dNFS implements multipath I/O internally, configuring LACP for channel-bonding interfaces for dNFS data traffic through active-backup or link aggregation is unnecessary. If the LACP protocol is configured on the NIC interfaces intended for dNFS data traffic, remove the channel-bond on those interfaces so that the interfaces operate as independent ports.

If a single interface is used for the kNFS mount, NFS control traffic can be blocked if the interface is down or the network cable is unplugged. This blocked NFS traffic causes the database to appear unavailable. To mitigate this single point of failure in the network, configure LACP on multiple interfaces to create a channel-bonded interface for NFS control/management traffic. This configuration is recommended for NFS control traffic. It provides increased database availability and additional network bandwidth.

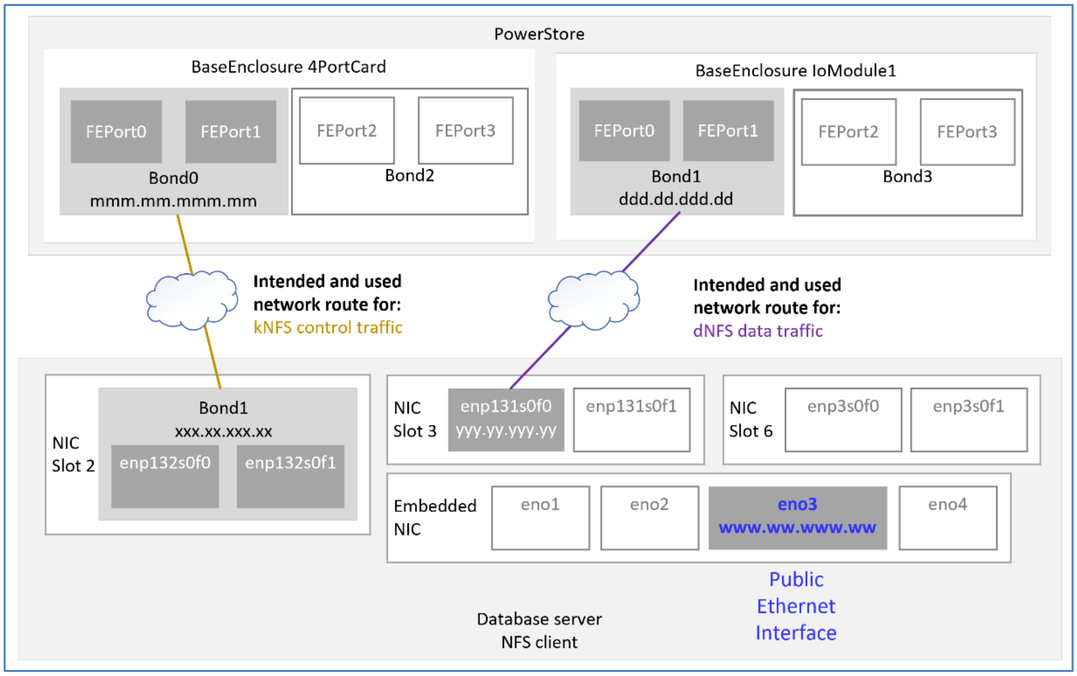


Figure 22. Network route using bonded client interface for NFS control traffic

Note: The preceding figure is not a cabling diagram. It represents network routes between PowerStore bonded interfaces and unbonded and bonded interfaces on the database server that are intended to be used by NFS control and NFS data traffic. For more information about creating network routes and using specific network routes for certain network traffic, see the following sections: Shared subnets, Static routing, dNFS without oranfstab, and dNFS with oranfstab.

In the figure, the interfaces (both bonded and unbonded) are cabled to switches, as shown in the network configuration step in Adding link aggregate network interfaces to a PowerStore NAS server and in Figure 15.

When configuring dNFS with multiple network paths, use a unique network for each of the paths if possible. When multiple unique networks are not available, or not wanted, a different IP address from the same subnet can be used for each of the network paths.

Shared subnets

If multiple network paths are needed for dNFS/kNFS control traffic and dNFS data traffic, Oracle recommends that each path uses a separate subnet. Using separate subnets, however, might not always be feasible.

This section discusses the use of subnet sharing on more than one network path for Oracle dNFS.

When configuring dNFS to use a subnet that is shared with another network interface, additional configuration, including IPv4 network routing filter and static routing, is required.

According to Oracle, if any IP address used by dNFS is from a shared subnet on another NIC interface, one of the following additional configuration changes must be made:

- Use `dontroute` in `oranfstab`.

- Configure operating system static routing for each of the network paths used by dNFS.

Note: For more information about dontroute and oranfstab, see *Oracle Database—Database Installation Guide*.

For more information about shared subnets, see Static routing.

IPv4 network routing filters

Linux 8 follows the recommendations of ingress filtering for multihomed networks (http://tools.ietf.org/html/rfc3704). These routing filters must be relaxed for multiple NIC interfaces in the same server to use the same subnet. Before configuring network interfaces to use the same subnet, ensure that you relax routing filters.

If the Oracle 19c preinstall rpm is used to configure the operating system before installing Oracle, the routing filters will be relaxed appropriately.

With OL 8u5, rp_filter is set to 2 for all network interfaces including the default interface. A value of 2 (loose filtering) is acceptable. Should no filtering (rp_filter = 0) or strict filtering (rp_filter = 1) be needed, see *Oracle Grid Infrastructure—Grid Infrastructure Installation and Upgrade Guide*, and My Oracle Support Doc ID 1286796.1.

If dNFS multipathing is needed and there is only one subnet, loose filtering (rp_filter = 2) needs to be set.

Setting rp_filter to an incorrect value can cause interconnect packets to be blocked or discarded in a RAC configuration.

Table 8. Reverse path filtering (rp_filter)

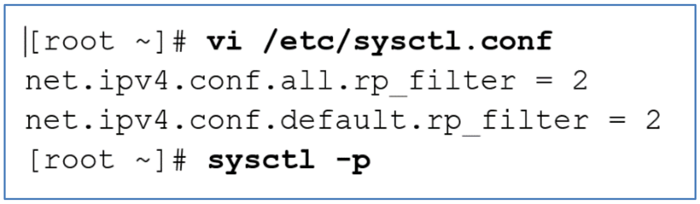
rp_filter value	Description
0	No filtering
1	Strict filtering
2	Loose filtering

To verify that IPv4 routing filters have been relaxed, look at the rp_filter settings in /etc/sysctl.conf. The values should be 2.

```
[oracle@r730xd-1 ~]$ grep rp_filter /etc/sysctl.conf | grep 2
# oracle-database-preinstall-19c setting for net.ipv4.conf.all.rp_filter is 2
net.ipv4.conf.all.rp_filter = 2
# oracle-database-preinstall-19c setting for net.ipv4.conf.default.rp_filter is 2
net.ipv4.conf.default.rp_filter = 2
[oracle@r730xd-1 ~]$
```

Figure 23. Loose IPv4 rp_filter

If IPv4 reverse path filtering is not set, add or change the IPv4 rp_filters in /etc/sysctl.conf and make them active.



```
|[root ~]# vi /etc/sysctl.conf
net.ipv4.conf.all.rp_filter = 2
net.ipv4.conf.default.rp_filter = 2
[root ~]# sysctl -p
```

Figure 24. Loose IPv4 rp_filter

For more information about channel bonded interfaces for NFS control traffic, see Oracle MOS notes:

- How to Setup Direct NFS Client Multipaths in Same Subnet (Doc ID 822481.1)
- How to configure DNFS to Use Multiple IPs (Doc ID 1552831.1)

Starting with OL kernel 2.6.31, Oracle RAC systems that use multiple NICs for the private interconnect must have parameter `rp_filter` set appropriately.

Static routing

When dNFS traffic on the NFS client must route traffic to an IP address from a subnet already in use on the NFS client, static routing must be configured. A static route must be defined for each of the network addresses from a shared subnet and used by dNFS as a route. This requirement includes defining a static route for dNFS data traffic should it use the same subnet as dNFS control traffic. If static routes are not defined, automatic load balancing and performance tuning of dNFS will not operate as expected per the dNFS path definitions in file `oranfstab`.

The remainder of this section provides two examples of static routing. The first example considers two interfaces sharing a subnet, and the second example considers four network interfaces sharing a subnet.

Shared subnet on multiple interfaces with single dedicated dNFS data path

The following figure illustrates a scenario where an NFS client has two interfaces. One interface, enp132s0f0, is intended to be dedicated to dNFS traffic, and the other, eno3, for general public Ethernet traffic. Both interfaces use a network IP address from the same subnet, and default routing is defined. The figure shows dNFS traffic flowing through interface eno3 rather than the intended interface enp132s0f0.

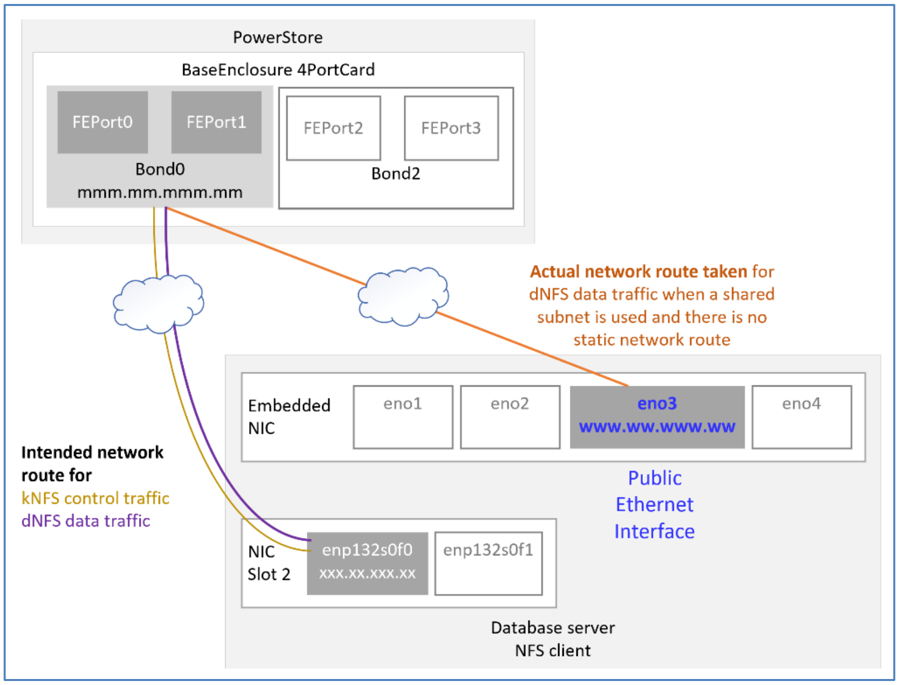
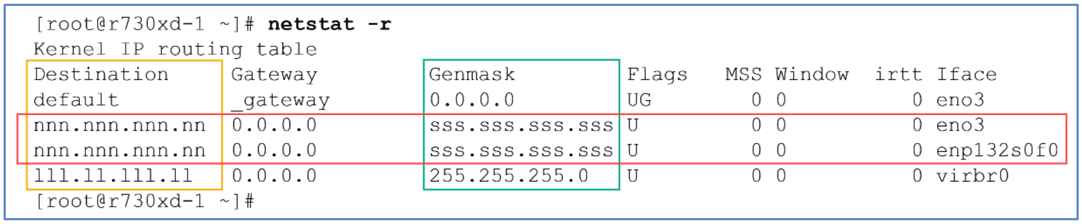


Figure 25. Incorrect network route taken by dNFS data traffic on shared subnet with default routing

Note: The preceding figure is not a cabling diagram. It represents network routes between a PowerStore bonded interface and interfaces on the database server. One network route is intended to be used for NFS control and NFS data traffic. The other network route is the network route that the operating system uses for NFS control and NFS data traffic. Because the operating system is using default network routing and shared subnets are used on multiple database server interfaces, the intended network route is not used. For more information about creating network routes and using specific network routes for certain network traffic, see the following sections: Shared subnets, Static routing, dNFS without oranfstab, and dNFS with oranfstab.

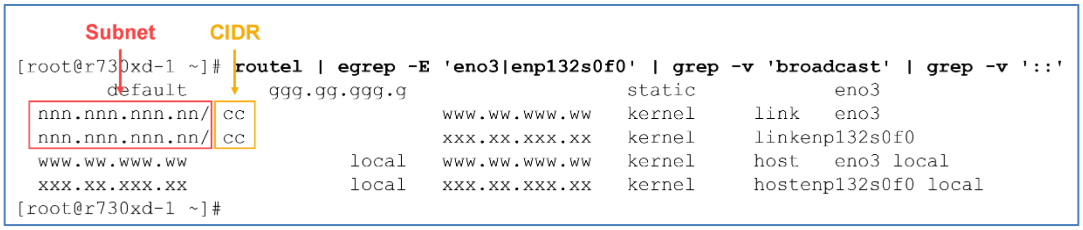
In the figure, the interfaces (both bonded and unbonded) are cabled to switches, as shown in the network configuration step in Adding link aggregate network interfaces to a PowerStore NAS server and in Figure 15.

The following two figures show the corresponding routing table for Figure 25. Each of the following figures was produced with different Linux commands: `netstat` and `routel`:



```
[root@r730xd-1 ~]# netstat -r
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
default         _gateway        0.0.0.0         UG        0 0          0 eno3
nnn.nnn.nnn.nn  0.0.0.0         sss.sss.sss.sss U         0 0          0 eno3
nnn.nnn.nnn.nn  0.0.0.0         sss.sss.sss.sss U         0 0          0 enp132s0f0
lll.ll.lll.ll   0.0.0.0         255.255.255.0   U         0 0          0 virbr0
[root@r730xd-1 ~]#
```

Figure 26. Default routing—netstat -r: single interface for dNFS on shared subnet not used



```
[root@r730xd-1 ~]# routel | egrep -E 'eno3|enp132s0f0' | grep -v 'broadcast' | grep -v '::'
       default       ggg.gg.ggg.g                     static          eno3
  nnn.nnn.nnn.nn/ cc                     www.ww.www.ww   kernel     link   eno3
  nnn.nnn.nnn.nn/ cc                     xxx.xx.xxx.xx   kernel     linkenp132s0f0
  www.ww.www.ww               local      www.ww.www.ww   kernel     host   eno3 local
  xxx.xx.xxx.xx               local      xxx.xx.xxx.xx   kernel     hostenp132s0f0 local
[root@r730xd-1 ~]#
```

Figure 27. Default routing—routel: single interface for dNFS on shared subnet not used

If default routing and IP addresses from the same subnet are used, the operating system searches the routing table for the route that best matches the destination address and mask. The operating system will use the best-matched route found. Because interfaces eno3 and enp132s0f0 share the same subnet (nnn.nn.nnn.nn) and CIDR (cc) [or Destination nnn.nn.nnn.nn and subnet mask (Genmask sss.ss.sss.ss)], the operating system considers both entries as best-matched. When multiple best-matched entries exist, the operating system selects the top-most best-matched entry for the route. Because interface eno3 is the top-most best-matched entry, the operating system will use it to route dNFS data traffic to the target. However, as shown in Figure 25, interface enp132s0f0 should be the interface used for that traffic.

To mitigate traffic misdirection, a static route must be added to the operating system routing table. The static route forces the operating system to send dNFS data traffic between the IP addresses specified in the static route. In this case, the static route is:

- IP address (xxx.xx.xxx.xx) of interface enp132s0f0

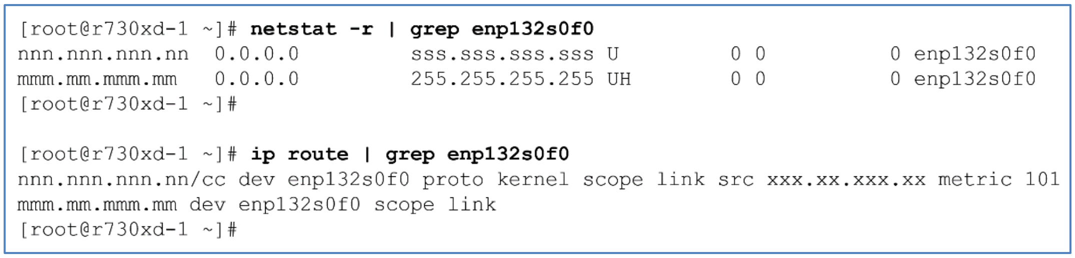
  and

- IP address (mmm.mm.mmm.mm) of bond0 of the PowerStore base enclosure 4-port card

To add a static route between interface enp132s0f0 and the IP of PowerStore BaseEnclosure-bond0, run the `ip route add` command. Specify the interface name `enp132s0f0` and the IP address of PowerStore BaseEnclosure-bond0:

```
ip route add mmm.mm.mmm.mm dev enp132s0f0
```

To verify that the static route appears in the routing table, run either a `netstat -r` or `ip route` command and filter for `enp132s0f0`:



```
[root@r730xd-1 ~]# netstat -r | grep enp132s0f0
nnn.nnn.nnn.nn  0.0.0.0        sss.sss.sss.sss U        0 0          0 enp132s0f0
mmm.mm.mmm.mm   0.0.0.0        255.255.255.255 UH       0 0          0 enp132s0f0
[root@r730xd-1 ~]#

[root@r730xd-1 ~]# ip route | grep enp132s0f0
nnn.nnn.nnn.nn/cc dev enp132s0f0 proto kernel scope link src xxx.xx.xxx.xx metric 101
mmm.mm.mmm.mm dev enp132s0f0 scope link
[root@r730xd-1 ~]#
```

Figure 28. Static route between enp132s0f0 and PowerStore bond1

With the static route in place, when the operating system searches the route table it finds the specific IP address from the static route and the subnetwork IP address. Because the static route defines a specific destination IP address rather than the subnetwork IP address, the operating system considers the specific destination IP address as best-matched and uses it. The static route forces Ethernet traffic from interface enp132s0f0 to PowerStore, as shown in the following figure:

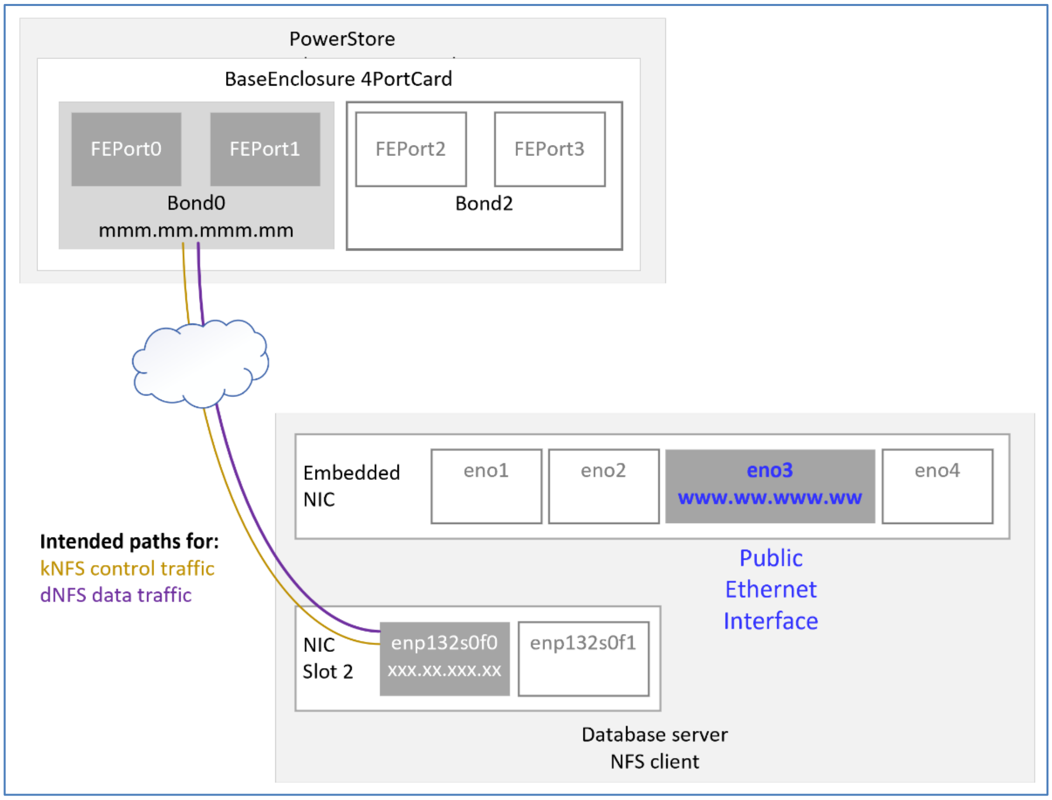


Figure 29. Static route—correct path taken for dNFS data traffic

Note: The preceding figure is not a cabling diagram. It represents a network route between a PowerStore bonded interface and an interface on the database server that is used for both NFS control and NFS data traffic. For more information about creating network routes and using specific network routes for certain network traffic, see the following sections: Shared subnets, Static routing, dNFS without oranfstab, and dNFS with oranfstab.

In the figure, the interfaces (both bonded and unbonded) are cabled to switches, as shown in the network configuration step in Adding link aggregate network interfaces to a PowerStore NAS server and in Figure 15.

Manual changes to the routing table are not persistent across server reboots. If static routes should be persistent, consider creating routing configuration files. For more information, see Persistent static routes.

Shared subnet on multiple interfaces with multiple dedicated dNFS data paths

This example illustrates how static routing changes the paths taken on multiple interfaces configured with IP addresses from the same subnet.

Figure 30 illustrates a scenario where an NFS client has four configured interfaces.

- Interface Bond1 is intended to be dedicated to kNFS/dNFS control traffic.
- Interfaces enp131s0f0 and enp3s0f0 are intended to be dedicatd to dNFS data traffic.

- Interface eno3 is intended for general public Ethernet traffic.

All four interfaces use a different network IP address from the same subnet, and default routing is defined. The figure shows that dNFS data traffic would flow through interface eno3 rather than intended interfaces enp131s0f0 and enp3s0f0.

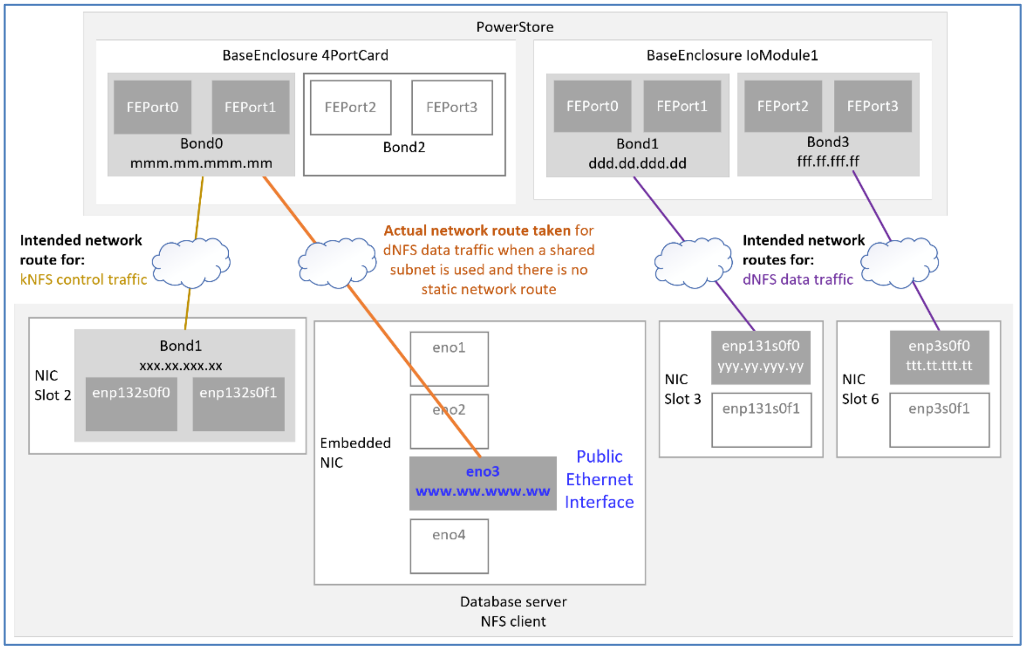
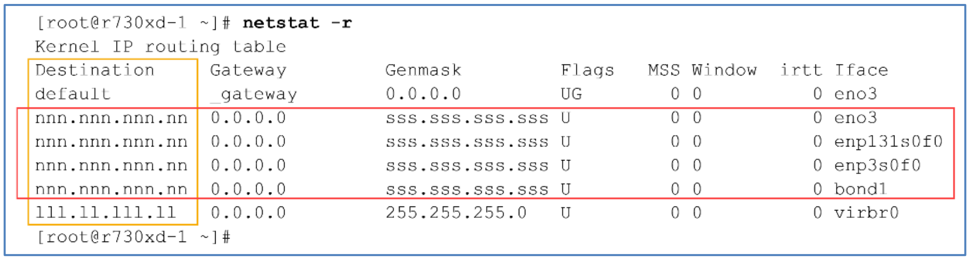


Figure 30. Incorrect network route taken by dNFS data traffic on shared subnet with default routing

Note: The preceding figure is not a cabling diagram. It represents network routes between PowerStore bonded interfaces and interfaces on the database server. Because the operating system is using default network routing and shared subnets are used on multiple database server interfaces, the intended network routes for dNFS data traffic are not used. For more information about creating network routes and using specific network routes for certain network traffic, see the following sections: Shared subnets, Static routing, dNFS without oranfstab, and dNFS with oranfstab.

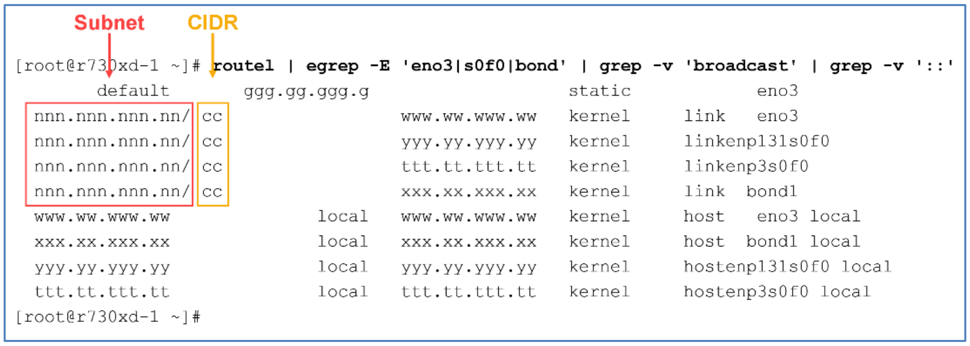
In the figure, the interfaces (both bonded and unbonded) are cabled to switches, as shown in the network configuration step in Adding link aggregate network interfaces to a PowerStore NAS server and in Figure 15.

The following two figures show the corresponding routing table for Figure 30. Each of the following figures was produced with different Linux commands: `netstat` and `routel`:



```
[root@r730xd-1 ~]# netstat -r
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
default         _gateway        0.0.0.0         UG        0 0          0 eno3
nnn.nnn.nnn.nn  0.0.0.0         sss.sss.sss.sss U         0 0          0 eno3
nnn.nnn.nnn.nn  0.0.0.0         sss.sss.sss.sss U         0 0          0 enp131s0f0
nnn.nnn.nnn.nn  0.0.0.0         sss.sss.sss.sss U         0 0          0 enp3s0f0
nnn.nnn.nnn.nn  0.0.0.0         sss.sss.sss.sss U         0 0          0 bond1
111.11.111.11   0.0.0.0         255.255.255.0   U         0 0          0 virbr0
[root@r730xd-1 ~]#
```

Figure 31. Default routing—netstat -r: traffic routed through interface eno3



```
[root@r730xd-1 ~]# routel | egrep -E 'eno3|s0f0|bond' | grep -v 'broadcast' | grep -v '::'
        default          ggg.gg.ggg.g                       static            eno3
  nnn.nnn.nnn.nn/ cc                      www.ww.www.ww   kernel      link    eno3
  nnn.nnn.nnn.nn/ cc                      yyy.yy.yyy.yy   kernel      linkenp131s0f0
  nnn.nnn.nnn.nn/ cc                      ttt.tt.ttt.tt   kernel      linkenp3s0f0
  nnn.nnn.nnn.nn/ cc                      xxx.xx.xxx.xx   kernel      link   bond1
  www.ww.www.ww                  local    www.ww.www.ww   kernel      host    eno3 local
  xxx.xx.xxx.xx                  local    xxx.xx.xxx.xx   kernel      host   bond1 local
  yyy.yy.yyy.yy                  local    yyy.yy.yyy.yy   kernel      hostenp131s0f0 local
  ttt.tt.ttt.tt                  local    ttt.tt.ttt.tt   kernel      hostenp3s0f0 local
[root@r730xd-1 ~]#
```

Figure 32. Default routing—routel: traffic routed through interface eno3

All dNFS traffic would again flow through interface eno3 because it is the top-most best-matched entry in the routing table that uses the same subnet and CIDR.

To mitigate this traffic misdirection, static routes must be added to the operating system routing table. The static route forces the operating system to send dNFS data traffic between the IP addresses specified in the static route. In this case, the static routes are:

- IP address (xxx.xx.xxx.xx) of interface bond1 of the database server, and IP address (mmm.mm.mmm.mm) of bond0 of the PowerStore base enclosure 4-port card
- IP address (yyy.yy.yyy.yy) of interface enp131s0f0 and IP address (ddd.dd.ddd.dd) of bond1 of the PowerStore IoModule1 4-port card
- IP address (ttt.tt.ttt.tt) of interface enp3s0f0 and IP address (fff.ff.fff.ff) of bond3 of the PowerStore IoModule1 4-port card

To direct Ethernet traffic to flow through all the intended interfaces (enp131s0f0, enp3s0f0, and bond1), the following static routes are needed:

```
ip route add mmm.mm.mmm.mm dev bond1
ip route add ddd.dd.ddd.dd dev enp131s0f0
ip route add fff.ff.fff.ff dev enp3s0f0
```

Once the static routes are defined, the operating system chooses static routes because they best fit the destination and subnet mask/CIDR of the destination interface (see Figure 33).

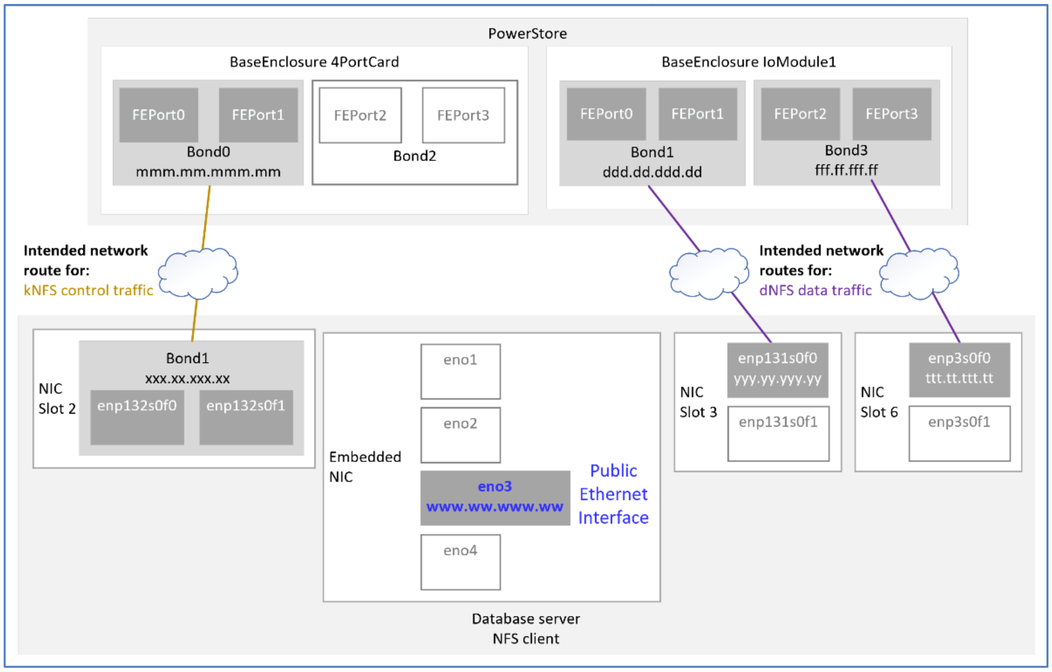


Figure 33. Static route—correct path taken for dNFS data traffic

Note: The preceding figure is not a cabling diagram. It represents network routes between PowerStore bonded interfaces and several interfaces on the database server that are used for both NFS control and NFS data traffic. For more information about creating network routes and using specific network routes for certain network traffic, see the following sections: Shared subnets, Static routing, dNFS without oranfstab, and dNFS with oranfstab.

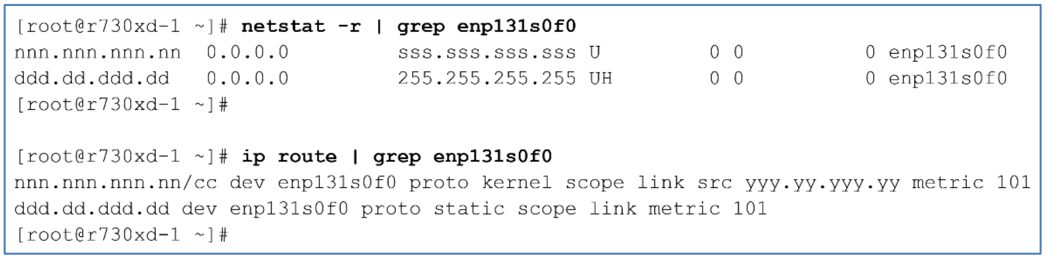
In the figure, the interfaces (both bonded and unbonded) are cabled to switches, as shown in the network configuration step in Adding link aggregate network interfaces to a PowerStore NAS server and in Figure 15.

Manual changes to the routing table are not persistent across server reboots. If static routes should be persistent, consider creating routing configuration files. For more information, see Persistent static routes.

To verify that the static route appears in the routing table, run either a `netstat -r` or `ip route` command and filter on the interface name:

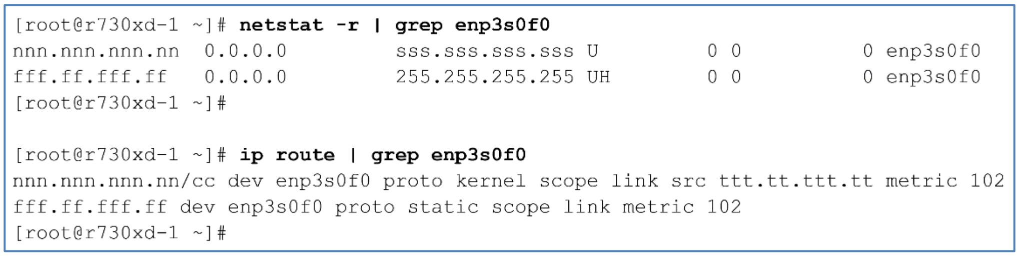
```
[root@r730xd-1 ~]# netstat -r | grep bond1
nnn.nnn.nnn.nn  0.0.0.0         sss.sss.sss.sss U         0 0          0 bond1
mmm.mm.mmm.mm   0.0.0.0         255.255.255.255 UH        0 0          0 bond1
[root@r730xd-1 ~]#

[root@r730xd-1 ~]# ip route | grep bond1
nnn.nnn.nnn.nn/cc dev bond1 proto kernel scope link src xxx.xx.xxx.xx metric 300
mmm.mm.mmm.mm dev bond1 proto static scope link metric 300
[root@r730xd-1 ~]#
```

Figure 34. Static route between database server interface bond1 and PowerStore bond0



```
[root@r730xd-1 ~]# netstat -r | grep enp131s0f0
nnn.nnn.nnn.nn  0.0.0.0         sss.sss.sss.sss U         0 0          0 enp131s0f0
ddd.dd.ddd.dd   0.0.0.0         255.255.255.255 UH        0 0          0 enp131s0f0
[root@r730xd-1 ~]#

[root@r730xd-1 ~]# ip route | grep enp131s0f0
nnn.nnn.nnn.nn/cc dev enp131s0f0 proto kernel scope link src yyy.yy.yyy.yy metric 101
ddd.dd.ddd.dd dev enp131s0f0 proto static scope link metric 101
[root@r730xd-1 ~]#
```

Figure 35. Static route between database server interface enp131s0f0 and PowerStore bond1



```
[root@r730xd-1 ~]# netstat -r | grep enp3s0f0
nnn.nnn.nnn.nn  0.0.0.0         sss.sss.sss.sss U         0 0          0 enp3s0f0
fff.ff.fff.ff   0.0.0.0         255.255.255.255 UH        0 0          0 enp3s0f0
[root@r730xd-1 ~]#

[root@r730xd-1 ~]# ip route | grep enp3s0f0
nnn.nnn.nnn.nn/cc dev enp3s0f0 proto kernel scope link src ttt.tt.ttt.tt metric 102
fff.ff.fff.ff dev enp3s0f0 proto static scope link metric 102
[root@r730xd-1 ~]#
```

Figure 36. Static route between database server enp3s0f0 and PowerStore bond3

Persistent static routes

To ensure that static routes persist across server reboots, create a static route configuration file in directory `/etc/sysconfig/network-scripts` for each of the static routes. Name the static route configuration file `route-<NAS client interface>`.

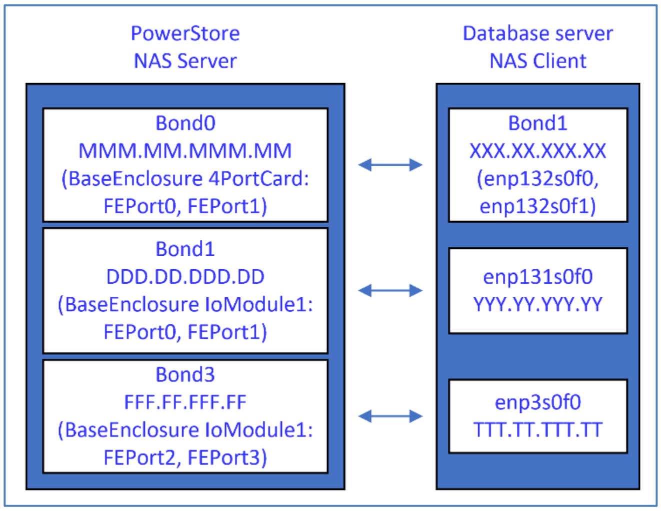


Figure 37. Static routes for dNFS data and kNFS control traffic

The following static route configuration files define the routes for the preceding figure:

```
[root@r730xd-1 network-scripts]# cat route-bond1
ADDRESS0=mmm.mm.mmm.mm
NETMASK0=255.255.255.255
GATEWAY0=ggg.gg.ggg.g
[root@r730xd-1 network-scripts]#

[root@r730xd-1 network-scripts]# cat route-enp131s0f0
ADDRESS0=ddd.dd.ddd.dd
```

```
NETMASK0=255.255.255.255
GATEWAY0=ggg.gg.ggg.g
[root@r730xd-1 network-scripts]#

[root@r730xd-1 network-scripts]# cat route-enp3s0f0
ADDRESS0=fff.ff.fff.ff
NETMASK0=255.255.255.255
GATEWAY0=ggg.gg.ggg.g
[root@r730xd-1 network-scripts]#
```

Configuring LACP—NFS client

On NFS clients requiring greater availability, a bonded NIC interface for NFS control traffic is recommended.

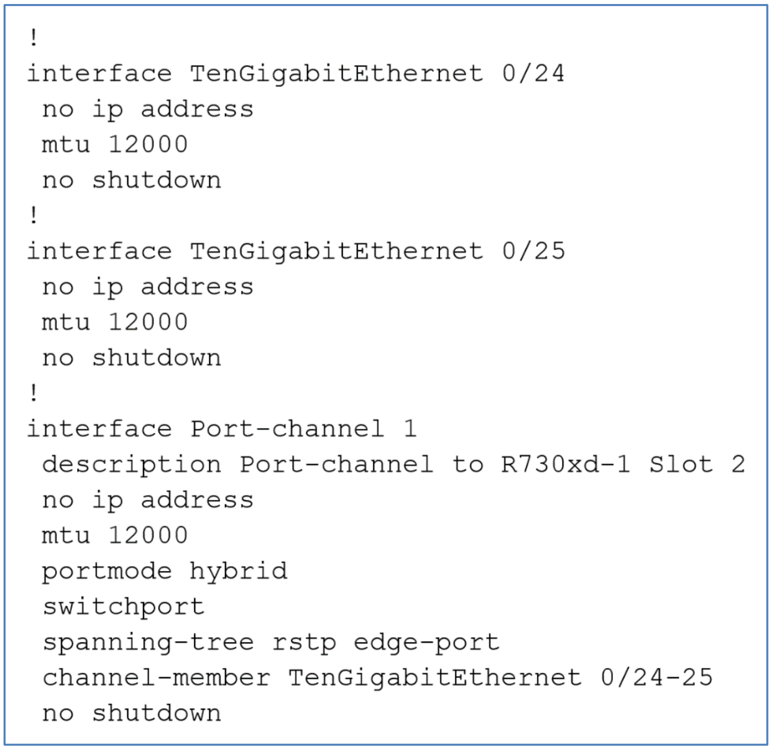
This section addresses:

- Switch port-channel configuration for NFS client (database server)
- NFS client (database server) and channel-bonding configuration
- Ethernet switch and port-channel configuration for PowerStore NAS network
- PowerStore link aggregation configuration

Switch port-channel configuration for NFS client (database server)

If high availability of NFS control traffic is needed, switch interfaces used for NFS control traffic must be configured in a port-channel. For reference, see Figure 42.

The following figure shows a snippet of the switch configuration for the port channel used for the bonded database server interface for NFS control traffic:



```
!
interface TenGigabitEthernet 0/24
 no ip address
 mtu 12000
 no shutdown
!
interface TenGigabitEthernet 0/25
 no ip address
 mtu 12000
 no shutdown
!
interface Port-channel 1
 description Port-channel to R730xd-1 Slot 2
 no ip address
 mtu 12000
 portmode hybrid
 switchport
 spanning-tree rstp edge-port
 channel-member TenGigabitEthernet 0/24-25
 no shutdown
```

Figure 38. Switch configuration for interfaces used by kNFS control traffic.

NFS client (database server) and channel-bonding configuration

As mentioned earlier, if an unbonded interface is used for NFS control traffic and that interface sustains an outage, the database can appear unavailable under certain operations. To mitigate this issue, configure LACP protocol on multiple interfaces to create a channel-bonded interface for NFS control/management traffic. Figure 40 and Figure 41 show two different possible bonded interfaces. Figure 40 shows a bonded interface created from interfaces residing on the same NIC. Figure 41 shows a bonded interface created from interfaces from multiple NICs. The bonded interface in Figure 41 provides redundancy with NIC card failure. This paper used the configuration in Figure 40.

NFS client (database server) interface configuration and channel-bonding

After determining the slot number to use, configure the interface configuration file or files.

For this paper, the database server used the following slots and interfaces:

- Bonded ports on the interfaces in slot 2
- A single interface (port 0) from both slots 3
- A single interface (port 0) from both slots 6

See Figure 39. In the figure, network paths are labeled with intended NFS traffic.

Note: Dell Technologies recommends using two TOR switches for redundancy. Using two TOR switches allows the port-channel members and dNFS data path interfaces to span switches and provide greater availability.

The lab used for this paper used only a single TOR switch. Using a single TOR switch is not recommended for environments requiring greater availability.

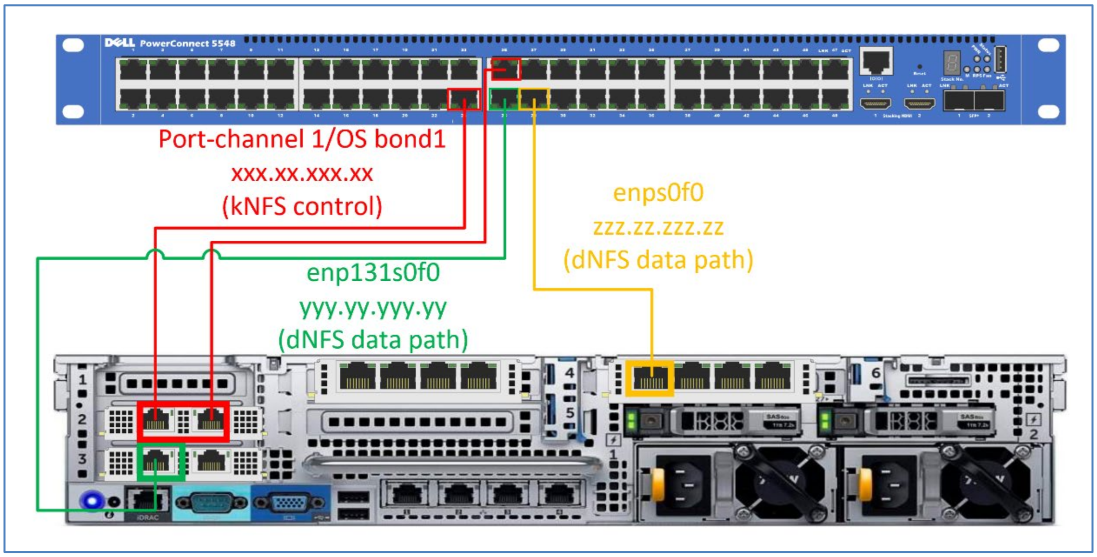


Figure 39. Database server interfaces and intended NFS traffic

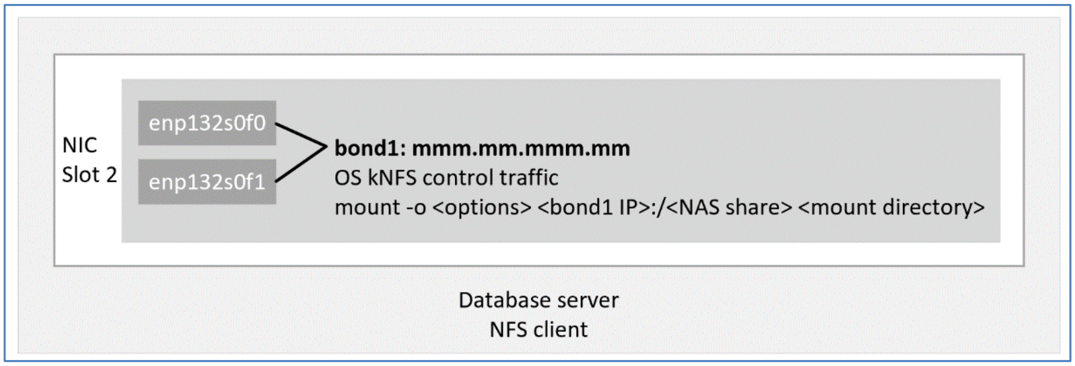


Figure 40. Bonded interface on a single NIC for NFS control traffic

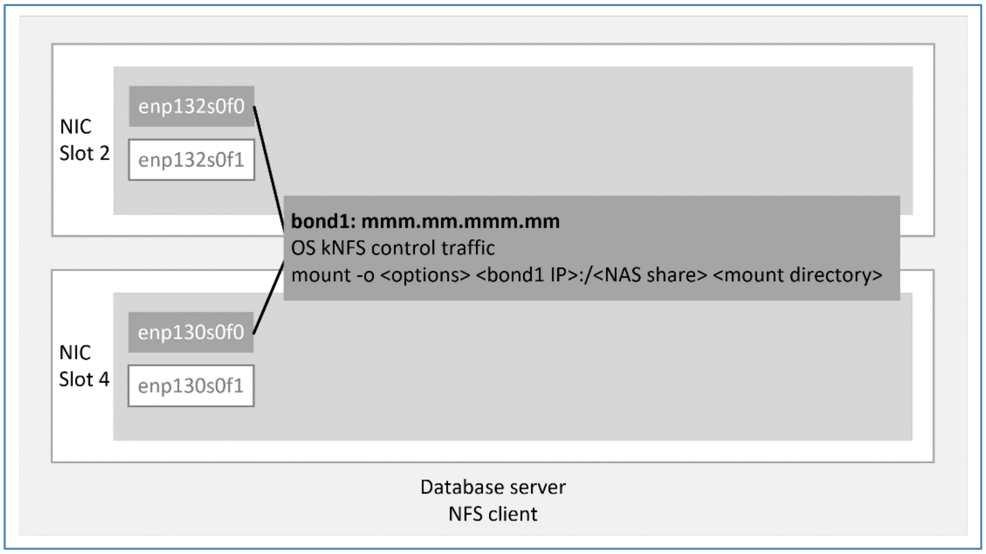


Figure 41. Bonded interface across multiple NICs for NFS control traffic

Should one of the interface members of the channel-bond suffer an outage, traffic can flow through the remaining healthy interface. This bonded interface could be shared with other traffic such as public network traffic or even the RAC interconnect in a RAC environment. A dedicated channel-bonded interface for NFS control traffic should not be necessary because NFS control or metadata traffic should be minimal.

If LACP is configured on the NFS client for NFS control traffic, a port channel must be configured in the Ethernet switch connected to the NFS client. The Ethernet switch port channel members must be cabled to the interfaces on the NFS client that are members of the bonded interface. Figure 42 shows the cabling for the bonded interface on the database server.

Note: Dell Technologies recommends using two TOR switches for redundancy. Using two TOR switches allows the port-channel members for dNFS control traffic and dNFS data path interfaces to span switches and provide greater availability. The lab used for this paper used only a single TOR switch. Using a single TOR switch is not recommended for environments requiring greater availability.

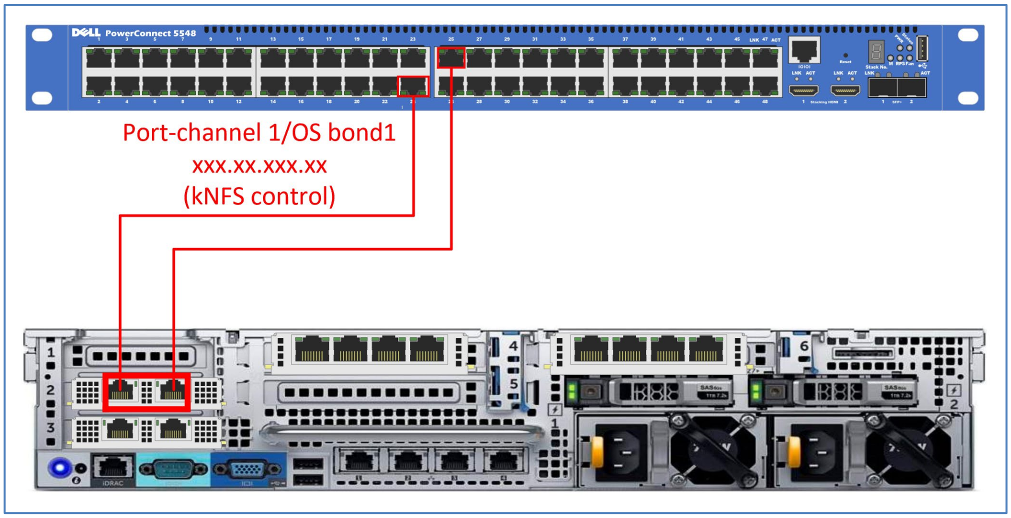


Figure 42. Cabling bonded interfaces between the server and switch

Interface configuration file for bond 1 (interfaces in slot 2)

```
[root@r730xd-1 network-scripts]# cat ifcfg-bond1
DEVICE=bond1
NAME=bond1
TYPE=Bond
BONDING_MASTER=yes
IPADDR=XXX.XX.XXX.XX
PREFIX=CC
GATEWAY=GGG.GG.GGG.G
ONBOOT=yes
BOOTPROTO=none
[root@r730xd-1 network-scripts]#
```

Interface configuration files bond 1 members [ifcfg-enp132s0f0 (slot2) and ifcfg-enp132s0f1 (slot2)]

```
[root@r730xd-1 network-scripts]# cat ifcfg-enp132s0f0
TYPE=Ethernet
BOOTPROTO=none
NAME=bond1-slave
MASTER=bond1
SLAVE=yes
UUID=38b54659-6dc8-437c-b60e-e5e66c7db4f3
DEVICE=enp132s0f0
ONBOOT=yes
MTU=9000
[root@r730xd-1 network-scripts]#

[root@r730xd-1 network-scripts]# cat ifcfg-enp132s0f1
TYPE=Ethernet
BOOTPROTO=none
NAME=bond1-slave
MASTER=bond1
SLAVE=yes
```

```
UUID=62d5bf50-ea4d-44a9-9849-fc76033cf0ad
DEVICE=enp132s0f1
ONBOOT=yes
MTU=9000
[root@r730xd-1 network-scripts]#
```

Interface configuration file ifcfg-enp131s0f0 (slot3)

```
[root@r730xd-1 network-scripts]# cat ifcfg-enp131s0f0
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=none
DEFROUTE=no
IPV4_FAILURE_FATAL=no
IPV6INIT=no
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
NAME=enp131s0f0
UUID=30e39629-19b3-415f-94db-d3843c01a582
DEVICE=enp131s0f0
ONBOOT=yes
MTU=9000
IPADDR=YYY.YY.YYY.YY
PREFIX=CC
GATEWAY=GGG.GG.GGG.G
[root@r730xd-1 network-scripts]#
```

Interface configuration file ifcfg-enp3s0f0 (slot6)

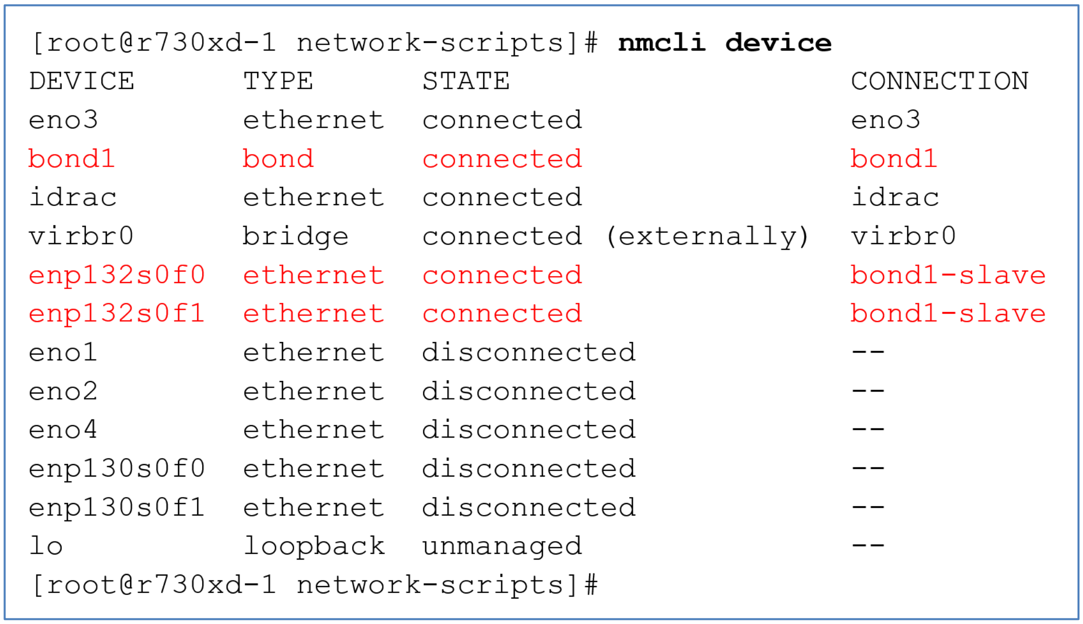
```
[root@r730xd-1 network-scripts]# cat ifcfg-enp3s0f0
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=none
DEFROUTE=no
IPV4_FAILURE_FATAL=no
IPV6INIT=no
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
NAME=enp3s0f0
UUID=22bdbed7-b10c-4e94-bf35-6b47bbe4c198
DEVICE=enp3s0f0
ONBOOT=yes
MTU=9000
IPADDR=TTT.TT.TTT.TT
PREFIX=CC
GATEWAY=GGG.GG.GGG.G
[root@r730xd-1 network-scripts]#
```

Starting NFS client (database server) interfaces and bonds

```
[root@r730xd-1 network-scripts]# ifup bond1
Connection successfully activated (master waiting for slaves) (D-
Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/5)
[root@r730xd-1 network-scripts]#

[root@r730xd-1 network-scripts]# ifup enp3s0f1
Connection successfully activated (D-Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/9)
[root@r730xd-1 network-scripts]#

[root@r730xd-1 network-scripts]# ifup enp131s0f1
Connection successfully activated (D-Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/9)
[root@r730xd-1 network-scripts]#
```

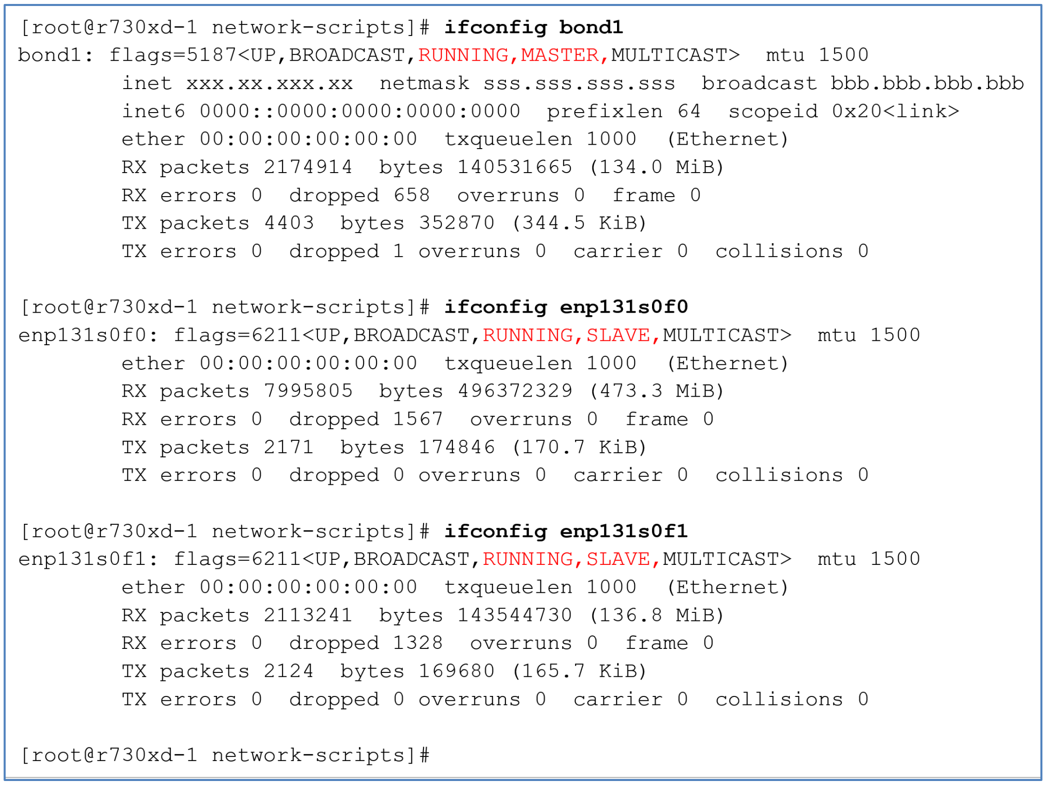
Verifying NFS client (database server) interfaces are running and connected

The following process only verifies that the bonded interface is connected. Repeat this process for all interfaces.



```
[root@r730xd-1 network-scripts]# nmcli device
DEVICE       TYPE      STATE                   CONNECTION
eno3         ethernet  connected               eno3
bond1        bond      connected               bond1
idrac        ethernet  connected               idrac
virbr0       bridge    connected (externally)  virbr0
enp132s0f0   ethernet  connected               bond1-slave
enp132s0f1   ethernet  connected               bond1-slave
eno1         ethernet  disconnected            --
eno2         ethernet  disconnected            --
eno4         ethernet  disconnected            --
enp130s0f0   ethernet  disconnected            --
enp130s0f1   ethernet  disconnected            --
lo           loopback  unmanaged               --
[root@r730xd-1 network-scripts]#
```

Figure 43. Display connected devices



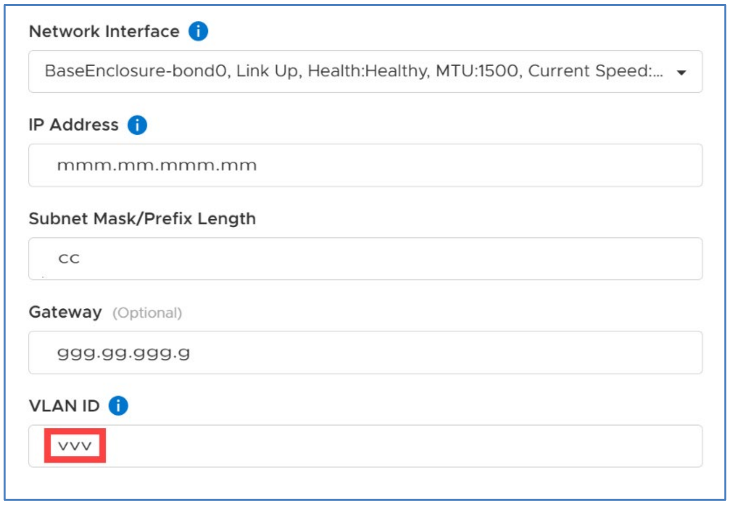
```
[root@r730xd-1 network-scripts]# ifconfig bond1
bond1: flags=5187<UP,BROADCAST,RUNNING,MASTER,MULTICAST>  mtu 1500
        inet xxx.xx.xxx.xx  netmask sss.sss.sss.sss  broadcast bbb.bbb.bbb.bbb
        inet6 0000::0000:0000:0000:0000  prefixlen 64  scopeid 0x20<link>
        ether 00:00:00:00:00:00  txqueuelen 1000  (Ethernet)
        RX packets 2174914  bytes 140531665 (134.0 MiB)
        RX errors 0  dropped 658  overruns 0  frame 0
        TX packets 4403  bytes 352870 (344.5 KiB)
        TX errors 0  dropped 1 overruns 0  carrier 0  collisions 0

[root@r730xd-1 network-scripts]# ifconfig enp131s0f0
enp131s0f0: flags=6211<UP,BROADCAST,RUNNING,SLAVE,MULTICAST>  mtu 1500
        ether 00:00:00:00:00:00  txqueuelen 1000  (Ethernet)
        RX packets 7995805  bytes 496372329 (473.3 MiB)
        RX errors 0  dropped 1567  overruns 0  frame 0
        TX packets 2171  bytes 174846 (170.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[root@r730xd-1 network-scripts]# ifconfig enp131s0f1
enp131s0f1: flags=6211<UP,BROADCAST,RUNNING,SLAVE,MULTICAST>  mtu 1500
        ether 00:00:00:00:00:00  txqueuelen 1000  (Ethernet)
        RX packets 2113241  bytes 143544730 (136.8 MiB)
        RX errors 0  dropped 1328  overruns 0  frame 0
        TX packets 2124  bytes 169680 (165.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[root@r730xd-1 network-scripts]#
```

Figure 44. Display the status of a running interface

Ethernet switch and port-channel configuration for PowerStore NAS network

The switch interfaces cabled to the PowerStore interfaces used in a PowerStore NAS server must be configured with LACP (see Figure 49).

If the candidate switch interfaces for the bonded interfaces are in a VLAN, remove them from the VLAN before configuring the port channel on the switch. When creating the port channel, add the port channel to the same VLAN that was or will be specified when defining the NAS network in PowerStore, as shown in the following figure:



Network Interface
BaseEnclosure-bond0, Link Up, Health:Healthy, MTU:1500, Current Speed:...

IP Address
mmm.mm.mmm.mm

Subnet Mask/Prefix Length
cc

Gateway (Optional)
ggg.gg.ggg.g

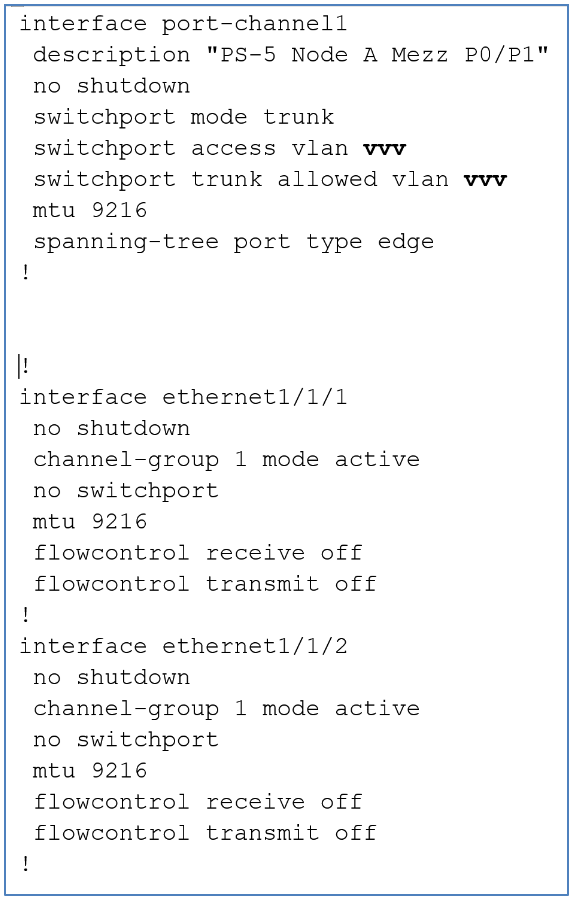
VLAN ID
vvv

Figure 45. PowerStore NAS network interface VLAN ID

Every NAS server must have at least one bonded interface. This bonded interface is specified when the NAS server is created and becomes the preferred network for the NAS server. NFS control traffic, export of the NFS share, and other control management functions will all be performed with the preferred network.

Each PowerStore NAS network requires four switch interfaces. Two switch interfaces are used for the bonded interface from node A, and the other two switch interfaces are for the bonded interface from node B. The switch interfaces for PowerStore nodes A and B must be configured in their own switch port channel.

The following figures show a snippet of the switch configuration for the port channel used for the link aggregate for PowerStore node A and node B:



```
interface port-channel1
 description "PS-5 Node A Mezz P0/P1"
 no shutdown
 switchport mode trunk
 switchport access vlan vvv
 switchport trunk allowed vlan vvv
 mtu 9216
 spanning-tree port type edge
!


!
interface ethernet1/1/1
 no shutdown
 channel-group 1 mode active
 no switchport
 mtu 9216
 flowcontrol receive off
 flowcontrol transmit off
!
interface ethernet1/1/2
 no shutdown
 channel-group 1 mode active
 no switchport
 mtu 9216
 flowcontrol receive off
 flowcontrol transmit off
!
```

Figure 46. Switch configuration for link aggregate from PowerStore node A

```
interface port-channel2
 description "PS-5 Node B Mezz P0/P1"
 no shutdown
 switchport mode trunk
 switchport access vlan vvv
 switchport trunk allowed vlan vvv
 mtu 9216
 spanning-tree port type edge
!


!
interface ethernet1/1/3
 no shutdown
 channel-group 2 mode active
 no switchport
 mtu 9216
 flowcontrol receive off
 flowcontrol transmit off
!
interface ethernet1/1/4
 no shutdown
 channel-group 2 mode active
 no switchport
 mtu 9216
 flowcontrol receive off
 flowcontrol transmit off
!
```

Figure 47. Switch configuration for link aggregate from PowerStore node B

PowerStore link aggregation configuration

Only bonded network interfaces in PowerStore can be used for PowerStore NAS networks. The requirement for bonded network interfaces is to provide redundancy and additional throughput of the NAS network.

With the initial release of PowerStore and PowerStoreOS 2, the first two ports of the 4-port mezzanine card were the only interfaces that supported link aggregation for high availability purposes. PowerStore automatically creates this link aggregation. The link aggregation is known as the system bond and is not user configurable.

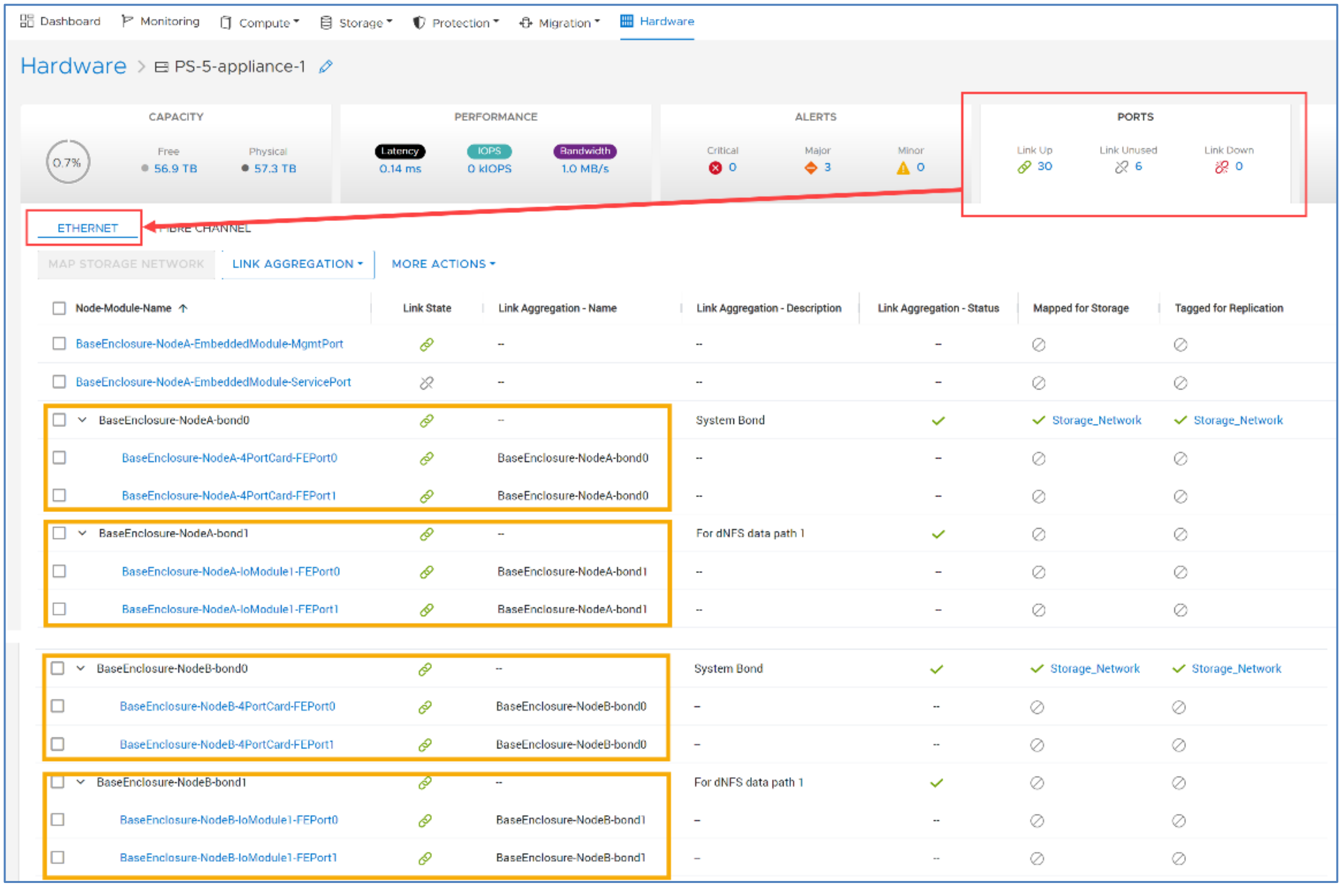
With PowerStoreOS 3, user-defined link aggregations are supported on the other network ports. LACP can be defined across two, three, or four ports, on the same card or I/O modules. Ports in that configuration must have the same speed, duplex, and MTU. User-created LACP is only supported for PowerStore NAS interfaces and is only available for PowerStore T models. For more information about what is user configurable, see the PowerStore documentation.

Note: Dell Technologies recommends using two TOR switches for redundancy. The lab used for this paper used only a single TOR switch.

Create bonded network interfaces before creating a PowerStore NAS server. If bonded network interfaces are not created first, the bonded network interface in PowerStore Manager will have a degraded status.

When a bonded interface is configured in PowerStore Manager, the candidate interfaces for the bonded interface must be cabled and configured in both node A and node B modules. They must also be cabled to the appropriate port channel on the switch.

The following figure shows two link aggregations (bond0 and bond1). Each of the link aggregations is mirrored on both node A and node B.



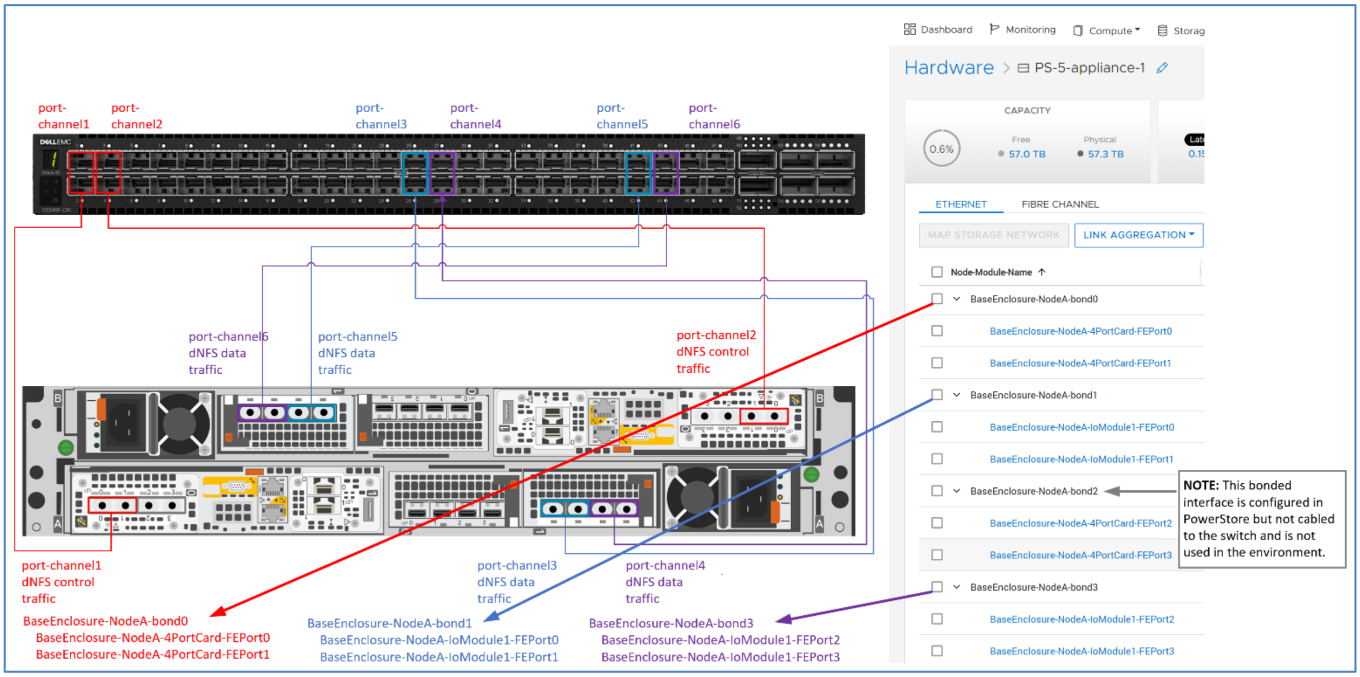
Node-Module-Name	Link State	Link Aggregation - Name	Link Aggregation - Description	Link Aggregation - Status	Mapped for Storage	Tagged for Replication
□ BaseEnclosure-NodeA-EmbeddedModule-MgmtPort	🔗	--	--	–	⊘	⊘
□ BaseEnclosure-NodeA-EmbeddedModule-ServicePort	⛓️‍💥	--	--	–	⊘	⊘
□ BaseEnclosure-NodeA-bond0	🔗	--	System Bond	✓	✓ Storage_Network	✓ Storage_Network
□ BaseEnclosure-NodeA-4PortCard-FEPort0	🔗	BaseEnclosure-NodeA-bond0	--	–	⊘	⊘
□ BaseEnclosure-NodeA-4PortCard-FEPort1	🔗	BaseEnclosure-NodeA-bond0	--	–	⊘	⊘
□ BaseEnclosure-NodeA-bond1	🔗	--	For dNFS data path 1	✓	⊘	⊘
□ BaseEnclosure-NodeA-IoModule1-FEPort0	🔗	BaseEnclosure-NodeA-bond1	--	–	⊘	⊘
□ BaseEnclosure-NodeA-IoModule1-FEPort1	🔗	BaseEnclosure-NodeA-bond1	--	–	⊘	⊘
□ BaseEnclosure-NodeB-bond0	🔗	--	System Bond	✓	✓ Storage_Network	✓ Storage_Network
□ BaseEnclosure-NodeB-4PortCard-FEPort0	🔗	BaseEnclosure-NodeB-bond0	–	--	⊘	⊘
□ BaseEnclosure-NodeB-4PortCard-FEPort1	🔗	BaseEnclosure-NodeB-bond0	–	--	⊘	⊘
□ BaseEnclosure-NodeB-bond1	🔗	--	For dNFS data path 1	✓	⊘	⊘
□ BaseEnclosure-NodeB-IoModule1-FEPort0	🔗	BaseEnclosure-NodeB-bond1	–	--	⊘	⊘
□ BaseEnclosure-NodeB-IoModule1-FEPort1	🔗	BaseEnclosure-NodeB-bond1	–	--	⊘	⊘

Figure 48. PowerStore bonded interfaces

Link aggregations must use the same ports from both nodes. This configuration is necessary because, in failover, the peer node uses the same ports. Link aggregations can be configured across ports from the same or different I/O modules within the same appliance, provided that the performance characteristics of the interfaces are the same.

The number of switch interfaces required for a PowerStore link aggregate is double the number of interfaces in the PowerStore link aggregate. If a PowerStore link aggregate contains two interfaces, four switch interfaces are required: two switch interfaces for the two interfaces in node A and two switch interfaces for the two interfaces in node B. The following figure shows the interface mappings between PowerStore bonded interfaces and the Ethernet switch port channels used in this paper.

Note: Dell Technologies recommends using two TOR switches for redundancy. The lab used for this paper used only a single TOR switch. Using one switch is not recommended.


Dashboard
Monitoring
Compute
Storag
Hardware > PS-5-appliance-1
CAPACITY
0.6%
Free 57.0 TB
Physical 57.3 TB
ETHERNET
FIBRE CHANNEL
MAP STORAGE NETWORK
LINK AGGREGATION
port-channel1
port-channel2
port-channel3
port-channel4
port-channel5
port-channel6
port-channel6 dNFS data traffic
port-channel5 dNFS data traffic
port-channel2 dNFS control traffic
port-channel1 dNFS control traffic
port-channel3 dNFS data traffic
port-channel4 dNFS data traffic

□ Node-Module-Name ↑
□ BaseEnclosure-NodeA-bond0
□ BaseEnclosure-NodeA-4PortCard-FEPort0
□ BaseEnclosure-NodeA-4PortCard-FEPort1
□ BaseEnclosure-NodeA-bond1
□ BaseEnclosure-NodeA-IoModule1-FEPort0
□ BaseEnclosure-NodeA-IoModule1-FEPort1
□ BaseEnclosure-NodeA-bond2
□ BaseEnclosure-NodeA-4PortCard-FEPort2
□ BaseEnclosure-NodeA-4PortCard-FEPort3
□ BaseEnclosure-NodeA-bond3
□ BaseEnclosure-NodeA-IoModule1-FEPort2
□ BaseEnclosure-NodeA-IoModule1-FEPort3

NOTE: This bonded interface is configured in PowerStore but not cabled to the switch and is not used in the environment.

BaseEnclosure-NodeA-bond0
BaseEnclosure-NodeA-4PortCard-FEPort0
BaseEnclosure-NodeA-4PortCard-FEPort1

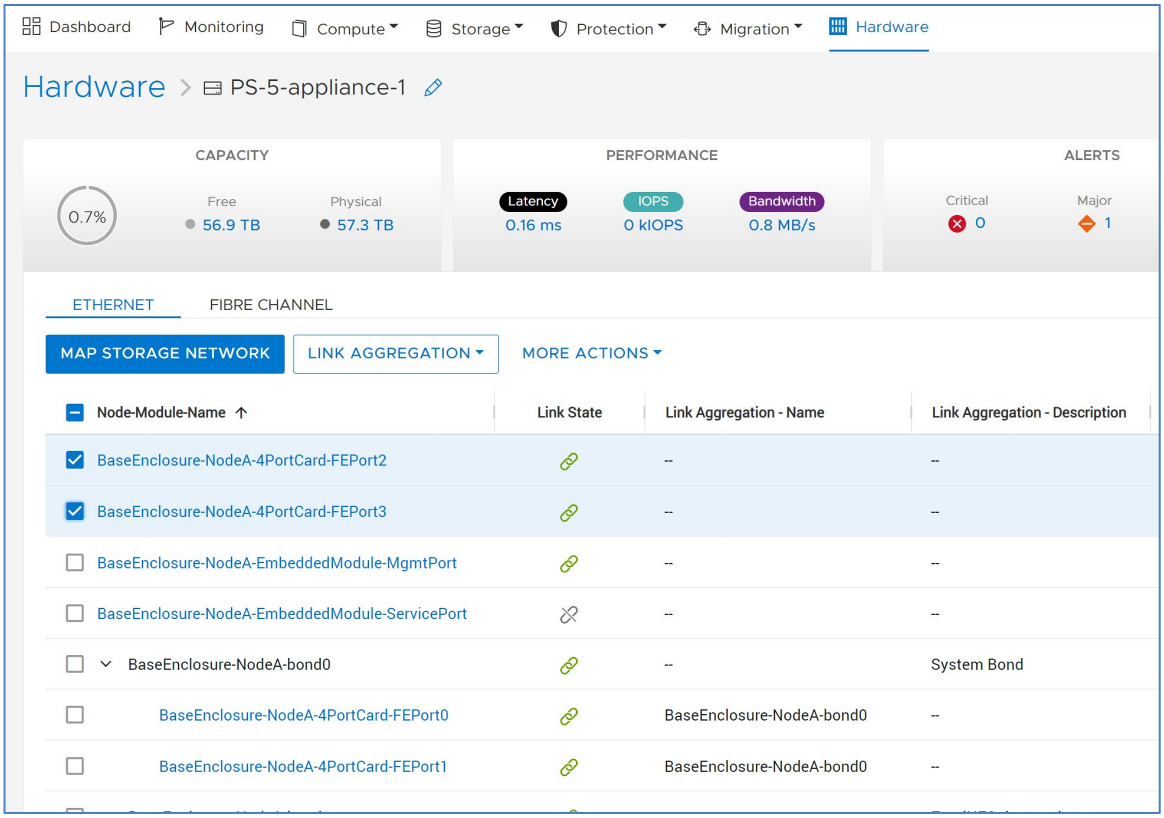
BaseEnclosure-NodeA-bond1
BaseEnclosure-NodeA-IoModule1-FEPort0
BaseEnclosure-NodeA-IoModule1-FEPort1

BaseEnclosure-NodeA-bond3
BaseEnclosure-NodeA-IoModule1-FEPort2
BaseEnclosure-NodeA-IoModule1-FEPort3

Figure 49. Interface mappings: PowerStore and Ethernet switch

To configure PowerStore link aggregations, perform the following steps in PowerStore Manager:

1. Log in to PowerStore Manager.
2. Click **Hardware**.
3. Click the **APPLIANCES** tab.
4. Click the appliance name where the link aggregation will be created.
5. Click **PORTS.**
6. Under the **Node-Module-Name** column, select the checkboxes corresponding to the interface members that will be members of the link aggregation:



Hardware > PS-5-appliance-1

CAPACITY: 0.7% — Free 56.9 TB — Physical 57.3 TB

PERFORMANCE: Latency 0.16 ms — IOPS 0 kIOPS — Bandwidth 0.8 MB/s

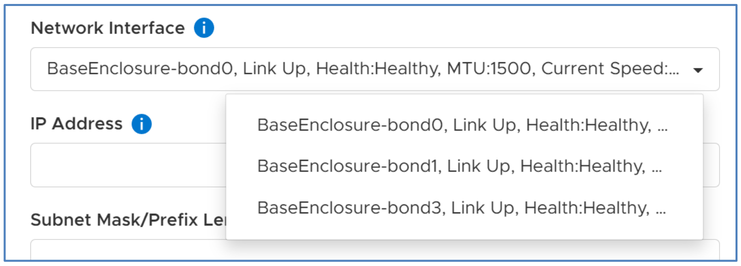
ALERTS: Critical 0 — Major 1

▬ Node-Module-Name ↑	Link State	Link Aggregation - Name	Link Aggregation - Description
☑ BaseEnclosure-NodeA-4PortCard-FEPort2	🔗	--	--
☑ BaseEnclosure-NodeA-4PortCard-FEPort3	🔗	--	--
□ BaseEnclosure-NodeA-EmbeddedModule-MgmtPort	🔗	--	--
□ BaseEnclosure-NodeA-EmbeddedModule-ServicePort	✗	--	--
□ ⌄ BaseEnclosure-NodeA-bond0	🔗	--	System Bond
□ BaseEnclosure-NodeA-4PortCard-FEPort0	🔗	BaseEnclosure-NodeA-bond0	--
□ BaseEnclosure-NodeA-4PortCard-FEPort1	🔗	BaseEnclosure-NodeA-bond0	--

7. Click **LINK AGGREGATION.**
8. Click **Aggregate Links.**
9. Provide an optional description.
10. Click **AGGREGATE**.

Repeat this process for each required PowerStore link aggregate.

Once a PowerStore link aggregate is created, it can be used as a NAS server network. The following figure shows all the link aggregations that were created before creating the NAS server used in this paper:



Network Interface

BaseEnclosure-bond0, Link Up, Health:Healthy, MTU:1500, Current Speed:...

- BaseEnclosure-bond0, Link Up, Health:Healthy, ...
- BaseEnclosure-bond1, Link Up, Health:Healthy, ...
- BaseEnclosure-bond3, Link Up, Health:Healthy, ...

Figure 50. Available PowerStore link aggregations

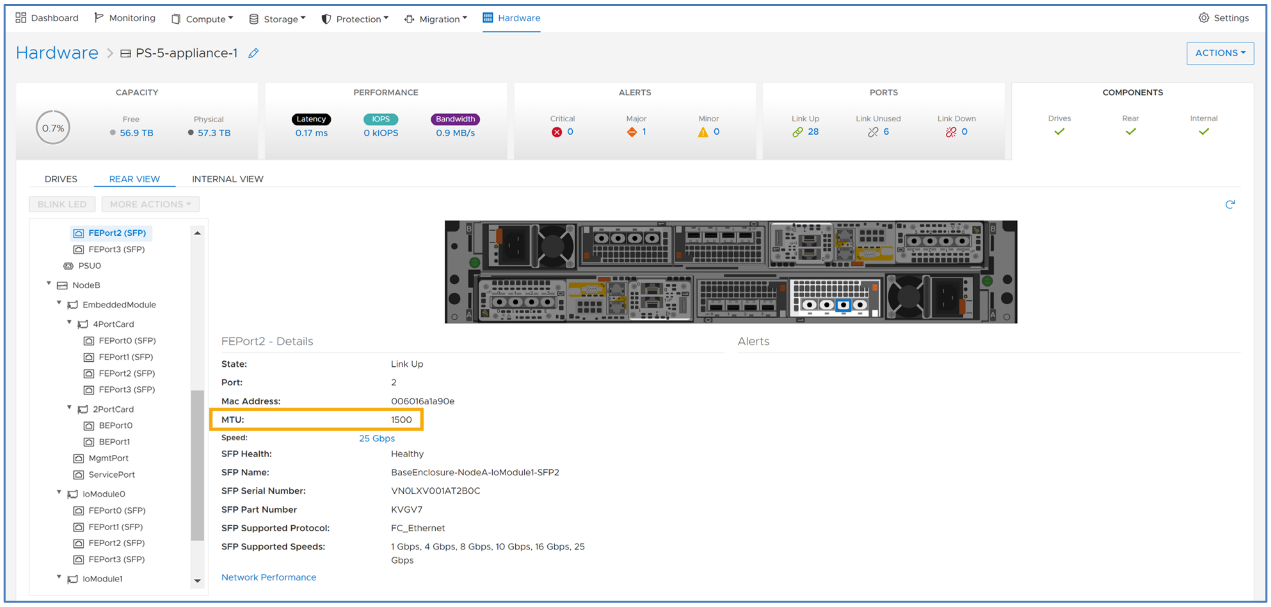
For information about creating a NAS server and assigning a bonded interface, see Creating a NAS server and adding a link aggregation network .

PowerStore jumbo frames (MTU)

If jumbo frames are needed on the PowerStore NAS network, see Jumbo frames. For more information, see the following documents:

- Dell PowerStore—Networking Guide for PowerStore T Models
- Dell PowerStore Host Configuration Guide
- Dell PowerStore—Configuring NFS

Note: Jumbo frames were not used in the configuration described in this paper. However, some of the components in the infrastructure are set for jumbo frames.



FEPort2 - Details

State: Link Up
Port: 2
Mac Address: 006016a1a90e
MTU: 1500
Speed: 25 Gbps
SFP Health: Healthy
SFP Name: BaseEnclosure-NodeA-IoModule1-SFP2
SFP Serial Number: VN0LXV001AT2B0C
SFP Part Number KVGV7
SFP Supported Protocol: FC_Ethernet
SFP Supported Speeds: 1 Gbps, 4 Gbps, 8 Gbps, 10 Gbps, 16 Gbps, 25 Gbps

Figure 51. PowerStore MTU

Oracle dNFS configuration file oranfstab

File `oranfstab` is optional. If it exists, it defines:

- Mount points that are available to dNFS
- Network paths that dNFS should use for dNFS data traffic

`oranfstab` can reside in either `/etc` or `$ORACLE_HOME/dbs`. If `oranfstab` resides in `/etc`, its contents will be global to all databases running on that server regardless of which `$ORACLE_HOME` the databases are running from. If `oranfstab` resides in `$ORACLE_HOME/dbs`, `oranfstab` is global to any database running from that `ORACLE_HOME`. If $`ORACLE_HOME` is shared between RAC nodes, all RAC databases running from the shared `$ORACLE_HOME` will use the same `$ORACLE_HOME/dbs/oranfstab`.

dNFS searches for mount entries in the following order and uses the first matching entry as the mount point:

- `$ORACLE_HOME/dbs/oranfstab`
- `/etc/oranfstab`
- `/etc/mtab`

If a database uses dNFS mount points configured in oranfstab, Oracle first verifies kNFS mount points by cross-checking entries in mtab and oranfstab. If a match does not exist, dNFS logs a message and fails to operate.

Benefits of using oranfstab

oranfstab provides dNFS with metadata to be able to perform multipathing. dNFS multipathing provides:

- Increased dNFS bandwidth
- Automatic dNFS data traffic load balancing
- Automatic dNFS channel failover

If any of these features are needed, file oranfstab is required.

When oranfstab is configured, dNFS automatically performs load balancing across all specified available channels in oranfstab. Also, if one channel fails, dNFS reissues I/O commands over any remaining available channel specified for that NAS server.

oranfstab directives

The following table shows the available parameters for oranfstab.

Table 9. oranfstab configuration parameters

Directive/attribute	Description
server	The name of the NAS server. If Kerberos authentication is used for the PowerStore NAS server, the value of server must be the fully qualified name of the NFS server. See Creating a NAS server and adding a link aggregation network . When KERBEROS is used, the value of server is used to create a service principal for Ticket Granting Service (TGS) requests from the Kerberos server. oranfstab may contain more than one server stanza. **Note:** When Kerberos is not used, Dell Technologies recommends setting this value to the name of the NAS server in PowerStore. However, the value can be any unique string of alphanumeric characters. For more information, see Oracle documentation.
local	A key:value entry. Key is local. Value is the IP address of an interface on the database server that should be used for dNFS data traffic. The operating system ifconfig command can be used to display the available interfaces and IP address configured on each interface. Up to four values of local can be supplied for each NAS server defined in oranfstab. A corresponding path value must exist for each local value.

Directive/attribute	Description
path	A key:value entry. Key is path. Value is the IP address of a bonded network interface on the PowerStore NAS server that will host the NFS shares serviced by dNFS on the database server. PowerStore Storage Manager can be used to display the available bonded network interface IP address. Up to four key:value entries of `path` can be supplied for each NAS server defined in `oranfstab`. A corresponding `local` value must exist for each `path` value.
export	NFS export path from PowerStore NAS server.
mount	The mount point of the NFS export on the database server.
mnt_timeout	Optional. Number of seconds the dNFS client should wait for a successful mount before timing out. Default is 600.
nfs_version	NFS protocol version used by the dNFS client. Valid values are: NFSv3, NFSv4, NFSv41, and pNFS. Default: NFSv3. PowerStore only supports NFSv3 and NFSv4, so do not configure Oracle dNFS to use parallel NFS (pNFS).
security_default	Optional. Specifies the default (sys) security mode applicable for all the exported NFS server paths for a NAS server entry in `oranfstab`. The value of `oranfstab` attribute `security` defines the security level.
security	Optional. The security level using Kerberos authentication protocol with the dNFS client. The value can be specified per export-mount pair.
dontroute	Specifies that outgoing messages should not be routed by the operating system but instead sent using the IP address to which they are bound. See the note that follows this table.
management	Directs the dNFS client to use the management interface for SNMP queries if SNMP is running on a separate management interface on the NAS server. Default: value of oranfstab server parameter.
community	The community string for use in SNMP queries. Default: public.

Note: `dontroute` is a POSIX option, which sometimes does not work on all Linux systems when multiple paths in the same subnet are used. Dell Technologies recommends investigating and testing the usage of `dontroute` and operating system static routes.

Figure 57 shows the oranfstab configuration used in this paper. For more `oranfstab` examples, see Oracle documentation.

Enabling and disabling Oracle dNFS

After Oracle 19c is installed, dNFS is enabled and disabled with the following commands from the Linux user owning the `ORACLE_HOME`.

Enabling dNFS

```
cd $ORACLE_HOME/rdbms/lib
make -f ins_rdbms.mk dnfs_on
```

Disabling dNFS

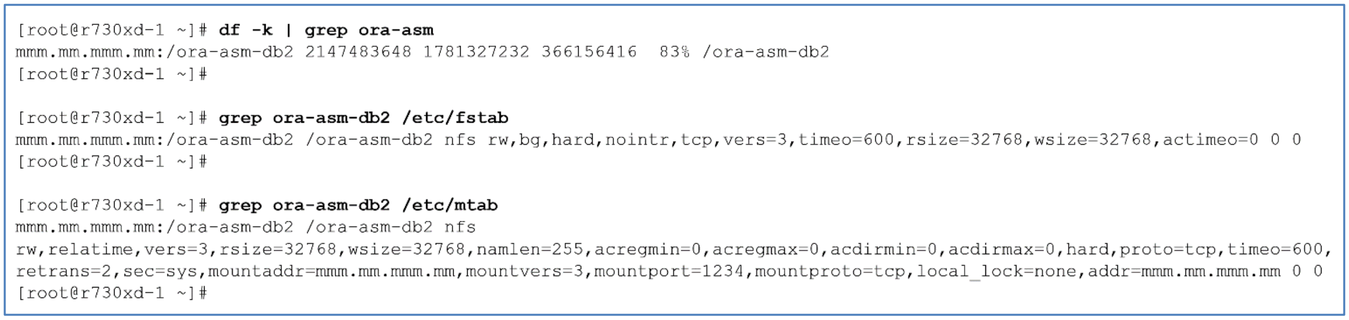
```
cd $ORACLE_HOME/rdbms/lib
make -f ins_rdbms.mk dnfs_off
```

dNFS without oranfstab

If `oranfstab` does not exist, dNFS creates a single dNFS channel for entries found in `/etc/mtab` that are required for the running database. This single dNFS channel is used for both dNFS control and dNFS data traffic. The local IP address used for the dNFS channel is the IP address used to mount the NFS share. In Oracle, the dNFS channel has a name equal to the local IP address of the mount entry in `/etc/mtab`.

Note: If `oranfstab` is not used, dNFS load balancing and scaling are not available. For more information, see Benefits of using oranfstab.

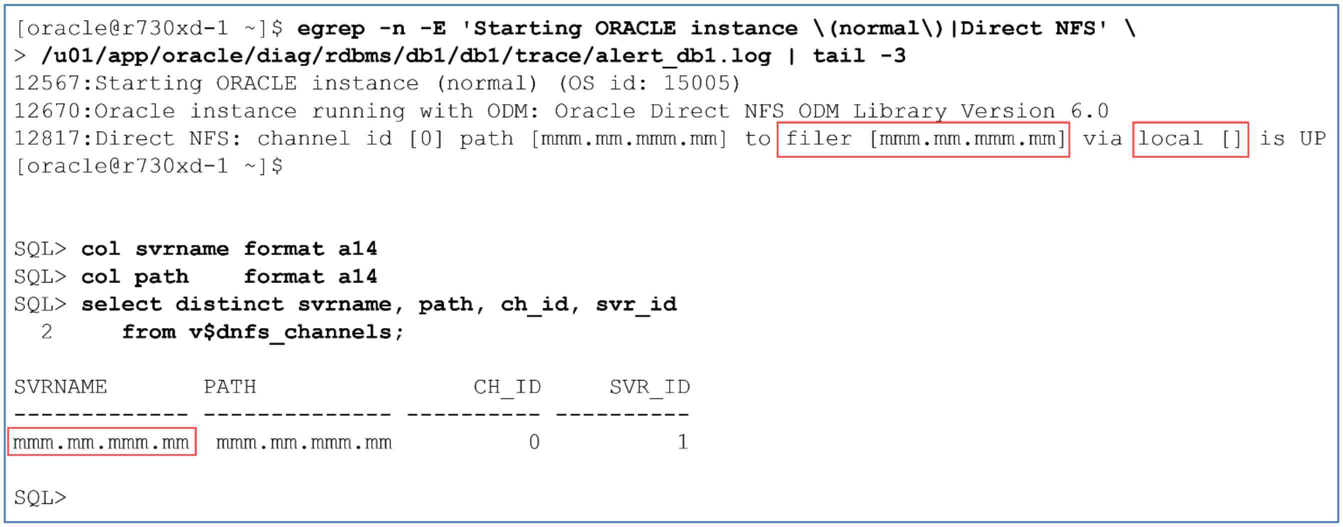
The following figure shows the mount point, `/etc/fstab`, and `/etc/mtab` entries for the single NFS share used in this paper:



```
[root@r730xd-1 ~]# df -k | grep ora-asm
mmm.mm.mmm.mm:/ora-asm-db2 2147483648 1781327232 366156416  83% /ora-asm-db2
[root@r730xd-1 ~]#

[root@r730xd-1 ~]# grep ora-asm-db2 /etc/fstab
mmm.mm.mmm.mm:/ora-asm-db2 /ora-asm-db2 nfs rw,bg,hard,nointr,tcp,vers=3,timeo=600,rsize=32768,wsize=32768,actimeo=0 0 0
[root@r730xd-1 ~]#

[root@r730xd-1 ~]# grep ora-asm-db2 /etc/mtab
mmm.mm.mmm.mm:/ora-asm-db2 /ora-asm-db2 nfs
rw,relatime,vers=3,rsize=32768,wsize=32768,namlen=255,acregmin=0,acregmax=0,acdirmin=0,acdirmax=0,hard,proto=tcp,timeo=600,
retrans=2,sec=sys,mountaddr=mmm.mm.mmm.mm,mountvers=3,mountport=1234,mountproto=tcp,local_lock=none,addr=mmm.mm.mmm.mm 0 0
[root@r730xd-1 ~]#
```

Figure 52. Mount point, /etc/fstab, /etc/mtab entries for PowerStore NFS share

The following figure shows Oracle referring to the name (filer) of the single channel as the IP address used to mount the PowerStore NFS export. Because a local IP address is not defined, dNFS reports the local IP in the alert log as a null value. In this case, dNFS uses the IP address used to mount the NFS share for dNFS data traffic.



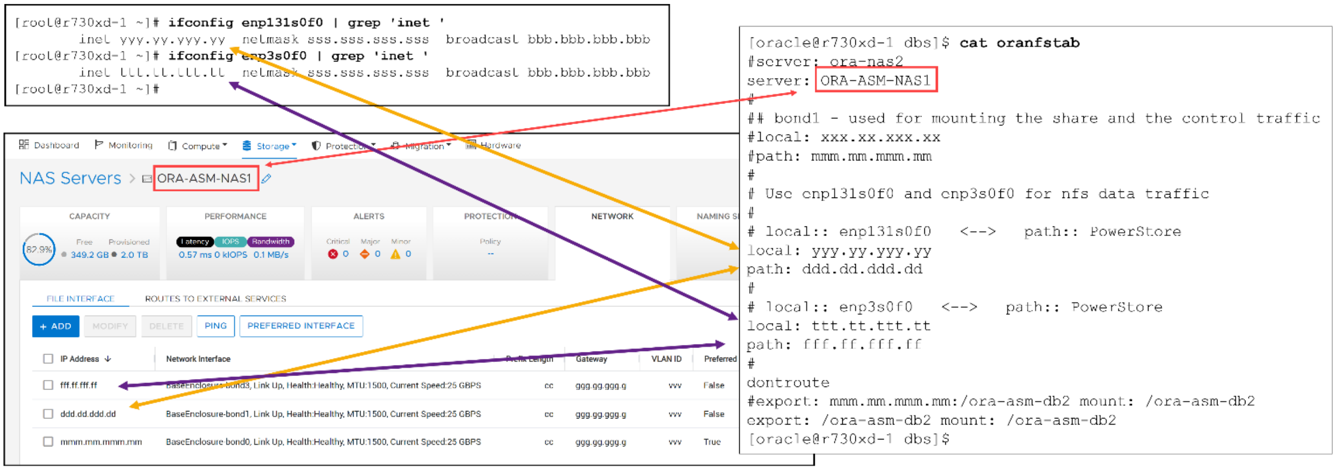
```
[oracle@r730xd-1 ~]$ egrep -n -E 'Starting ORACLE instance \(normal\)|Direct NFS' \
> /u01/app/oracle/diag/rdbms/db1/db1/trace/alert_db1.log | tail -3
12567:Starting ORACLE instance (normal) (OS id: 15005)
12670:Oracle instance running with ODM: Oracle Direct NFS ODM Library Version 6.0
12817:Direct NFS: channel id [0] path [mmm.mm.mmm.mm] to filer [mmm.mm.mmm.mm] via local [] is UP
[oracle@r730xd-1 ~]$


SQL> col svrname format a14
SQL> col path    format a14
SQL> select distinct svrname, path, ch_id, svr_id
  2      from v$dnfs_channels;

SVRNAME        PATH                CH_ID     SVR_ID
-------------- --------------- ---------- ----------
mmm.mm.mmm.mm   mmm.mm.mmm.mm           0          1

SQL>
```

Figure 53. Database using mount IP address for dNFS control and data traffic

If multiple channels to a NAS server are needed for increased dNFS bandwidth, automatic dNFS data traffic load balancing, or automatic dNFS channel failover, file `oranfstab` is required. dNFS automatically performs load balancing across all specified available channels; if one channel fails, dNFS reissues I/O commands over any remaining available channel for that NAS server.

dNFS with oranfstab

When `oranfstab` exists, dNFS control traffic uses the IP address used to mount the NFS share. dNFS data traffic is balanced across the local:path IP address mappings of the NAS server stanza specified in `oranfstab` to which the database belongs. `local` IP addresses are set to the IP address of the Ethernet interfaces on the database server that are to be used for the database. `path` IP addresses must be the IP addresses of the PowerStore NAS server bonded network interfaces.



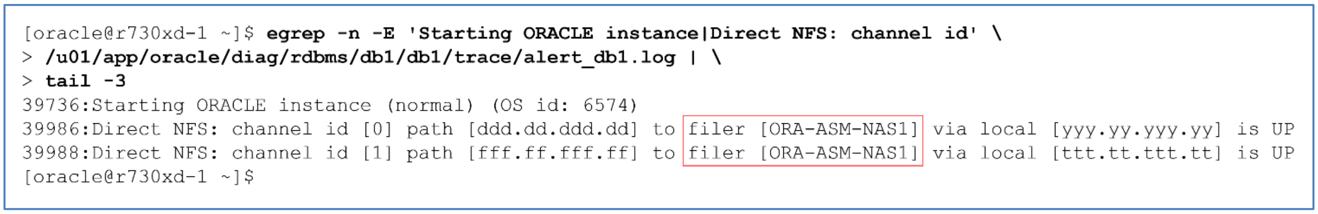
```
[root@r730xd-1 ~]# ifconfig enp131s0f0 | grep 'inet '
        inet yyy.yy.yyy.yy  netmask sss.sss.sss.sss  broadcast bbb.bbb.bbb.bbb
[root@r730xd-1 ~]# ifconfig enp3s0f0 | grep 'inet '
        inet ttt.tt.ttt.tt  netmask sss.sss.sss.sss  broadcast bbb.bbb.bbb.bbb
[root@r730xd-1 ~]#
```

IP Address	Network Interface	Prefix Length	Gateway	VLAN ID	Preferred
fff.ff.fff.ff	BaseEnclosure-bond3, Link Up, Health:Healthy, MTU:1500, Current Speed:25 GBPS	cc	ggg.gg.ggg.g	vvv	False
ddd.dd.ddd.dd	BaseEnclosure-bond1, Link Up, Health:Healthy, MTU:1500, Current Speed:25 GBPS	cc	ggg.gg.ggg.g	vvv	False
mmm.mm.mmm.mm	BaseEnclosure-bond0, Link Up, Health:Healthy, MTU:1500, Current Speed:25 GBPS	cc	ggg.gg.ggg.g	vvv	True

```
[oracle@r730xd-1 dbs]$ cat oranfstab
#server: ora-nas2
server: ORA-ASM-NAS1
#
## bond1 - used for mounting the share and the control traffic
#local: xxx.xx.xxx.xx
#path: mmm.mm.mmm.mm
#
# Use enp131s0f0 and enp3s0f0 for nfs data traffic
#
# local:: enp131s0f0   <-->   path:: PowerStore
local: yyy.yy.yyy.yy
path: ddd.dd.ddd.dd
#
# local:: enp3s0f0   <-->   path:: PowerStore
local: ttt.tt.ttt.tt
path: fff.ff.fff.ff
#
dontroute
#export: mmm.mm.mmm.mm:/ora-asm-db2 mount: /ora-asm-db2
export: /ora-asm-db2 mount: /ora-asm-db2
[oracle@r730xd-1 dbs]$
```

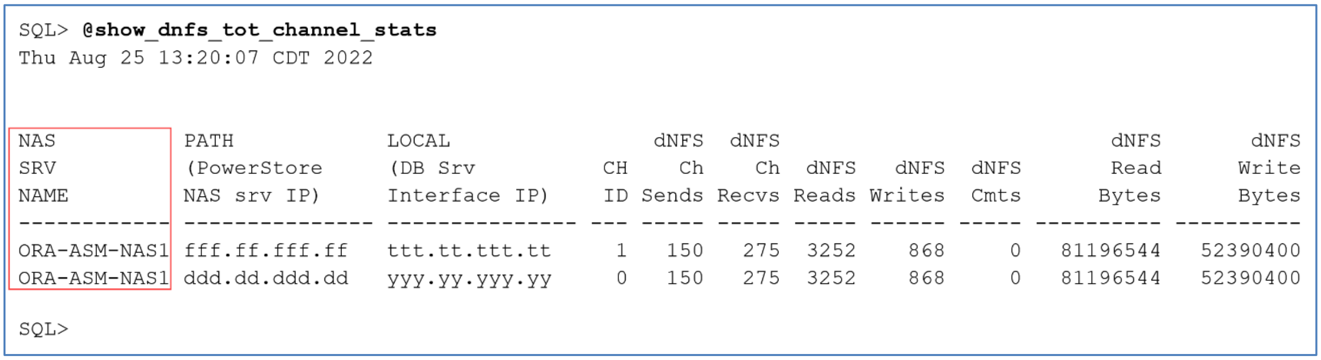
Figure 54. dNFS local:path IP address mappings

A dNFS channel for each of the local:path IP address mappings is created in the database instance. For each channel, the value of the NAS server name in `oranfstab` is recorded in the dNFS channel metadata.



```
[oracle@r730xd-1 ~]$ egrep -n -E 'Starting ORACLE instance|Direct NFS: channel id' \
> /u01/app/oracle/diag/rdbms/db1/db1/trace/alert_db1.log | \
> tail -3
39736:Starting ORACLE instance (normal) (OS id: 6574)
39986:Direct NFS: channel id [0] path [ddd.dd.ddd.dd] to filer [ORA-ASM-NAS1] via local [yyy.yy.yyy.yy] is UP
39988:Direct NFS: channel id [1] path [fff.ff.fff.ff] to filer [ORA-ASM-NAS1] via local [ttt.tt.ttt.tt] is UP
[oracle@r730xd-1 ~]$
```

Figure 55. dNFS filter name is value of oranfstab SERVER



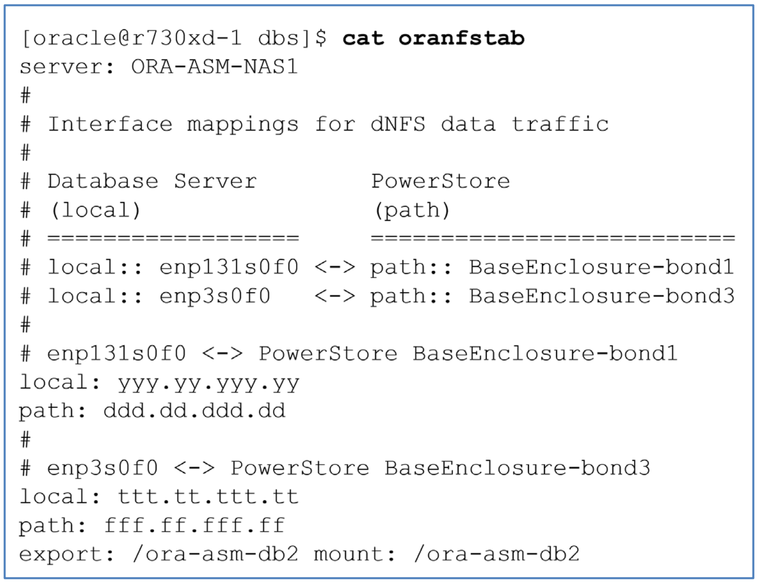
```
SQL> @show_dnfs_tot_channel_stats
Thu Aug 25 13:20:07 CDT 2022


NAS          PATH            LOCAL                dNFS  dNFS                                 dNFS       dNFS
SRV          (PowerStore     (DB Srv          CH    Ch    Ch  dNFS   dNFS  dNFS        Read      Write
NAME         NAS srv IP)     Interface IP)    ID Sends Recvs Reads Writes  Cmts       Bytes      Bytes
------------ --------------- --------------- --- ----- ----- ----- ------ ----- ---------- ----------
ORA-ASM-NAS1 fff.ff.fff.ff   ttt.tt.ttt.tt     1   150   275  3252    868     0   81196544   52390400
ORA-ASM-NAS1 ddd.dd.ddd.dd   yyy.yy.yyy.yy     0   150   275  3252    868     0   81196544   52390400

SQL>
```

Figure 56. dNFS NAS server name is value of oranfstab SERVER

`oranfstab` may contain multiple NAS server stanzas.

Single server stanza

The following `oranfstab` file defines one NAS server with two dNFS data paths:



```
[oracle@r730xd-1 dbs]$ cat oranfstab
server: ORA-ASM-NAS1
#
# Interface mappings for dNFS data traffic
#
# Database Server         PowerStore
# (local)                 (path)
# ==================      ==========================
# local:: enp131s0f0 <-> path:: BaseEnclosure-bond1
# local:: enp3s0f0   <-> path:: BaseEnclosure-bond3
#
# enp131s0f0 <-> PowerStore BaseEnclosure-bond1
local: yyy.yy.yyy.yy
path: ddd.dd.ddd.dd
#
# enp3s0f0 <-> PowerStore BaseEnclosure-bond3
local: ttt.tt.ttt.tt
path: fff.ff.fff.ff
export: /ora-asm-db2 mount: /ora-asm-db2
```

Figure 57. Single NAS server in oranfstab

Note: The format of specifying the dNFS data paths can vary within `oranfstab`. Also, if Kerberos is not used, the value of `server` can be any text. However, for ease of management, Dell Technologies recommends that the value of `server` be the name of the PowerStore NAS server for which the dNFS data paths defined in `oranfstab` apply.

Multiple server stanzas

The following `oranfstab` file contains two dNFS data paths to two NAS servers. Each NAS server services a different database.

```
server: ORA-NAS01
local: ddd.dd.ddd.dd path: jjj.jj.jjj.jj
local: eee.ee.eee.ee path: kkk.kk.kkk.kk
mnt_timeout: 60
export: /ORA-FS1 mount: /ora1db
#
server: ORA-ASM-NFS
local: hhh.hh.hhh.hh path: lll.ll.lll.ll
local: iii.ii.iii.ii path: nnn.nn.nnn.nn
mnt_timeout: 60
export: /ORA-ASM-NFS mount: /oraasmnas
```

Figure 58. Multiple NAS servers in oranfstab

When the Oracle instance is started, dNFS channels for the instance are created. The following example shows two channels for NAS server ORA-ASM-NFS. Channels for NAS server ORA-NAS01 are not shown because the current database relies only on ORA-ASM-NFS.

```
SQL> select distinct svrname
  2                , path
  3                , ch_id
  4                , svr_id
  5     from v$dnfs_channels
  6     order by ch_id;

SVRNAME         PATH                 CH_ID     SVR_ID
--------------- --------------- ---------- ----------
ORA-ASM-NFS     hhh.hh.hhh.hh            0          1
ORA-ASM-NFS     iii.ii.iii.ii            1          1
```

Figure 59. Displaying the dNFS channels

dNFS and network verification checks

This section discusses the following topics:

- Verifying the database is using dNFS
- Verifying all database files that rely on dNFS can be seen Oracle
- Verifying correct endpoints in dNFS paths
- Verifying correct operating system network routes are being used for dNFS data traffic

Verifying the database is using dNFS

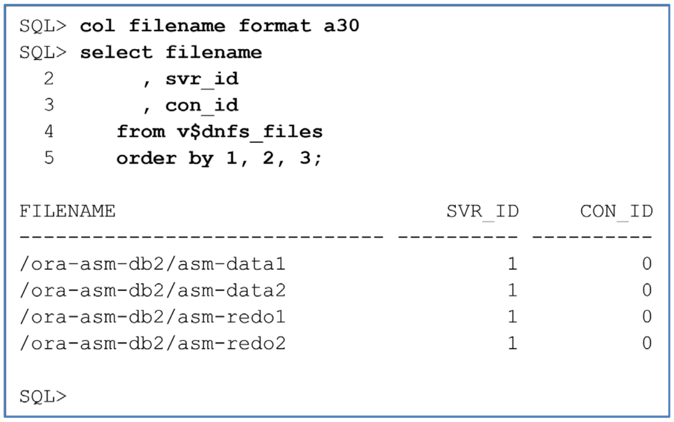
To verify that the database instance is using dNFS, check the database alert log for the `running with ODM` string. If the string is found, as shown in the following figure, the instance was started with the ODM library containing the direct NFS driver:

```
[oracle@r730xd-1 ~]$ egrep -n -E 'Starting ORACLE instance|running with ODM' \
> /u01/app/oracle/diag/rdbms/db1/db1/trace/alert_db1.log | tail -2
39736:Starting ORACLE instance (normal) (OS id: 6574)
39839:Oracle instance running with ODM: Oracle Direct NFS ODM Library Version 6.0
[oracle@r730xd-1 ~]$
```

Figure 60. Check if database instance was started with dNFS

Verifying all database files that rely on dNFS can be seen Oracle

The database compares the datafile names with the NFS mount to see if the datafiles can be used by dNFS. Any datafile that dNFS can work with resides in `v$dnfs_files`. To verify that the database can see all database files residing on the NFS share, run the following SQL*Plus commands:



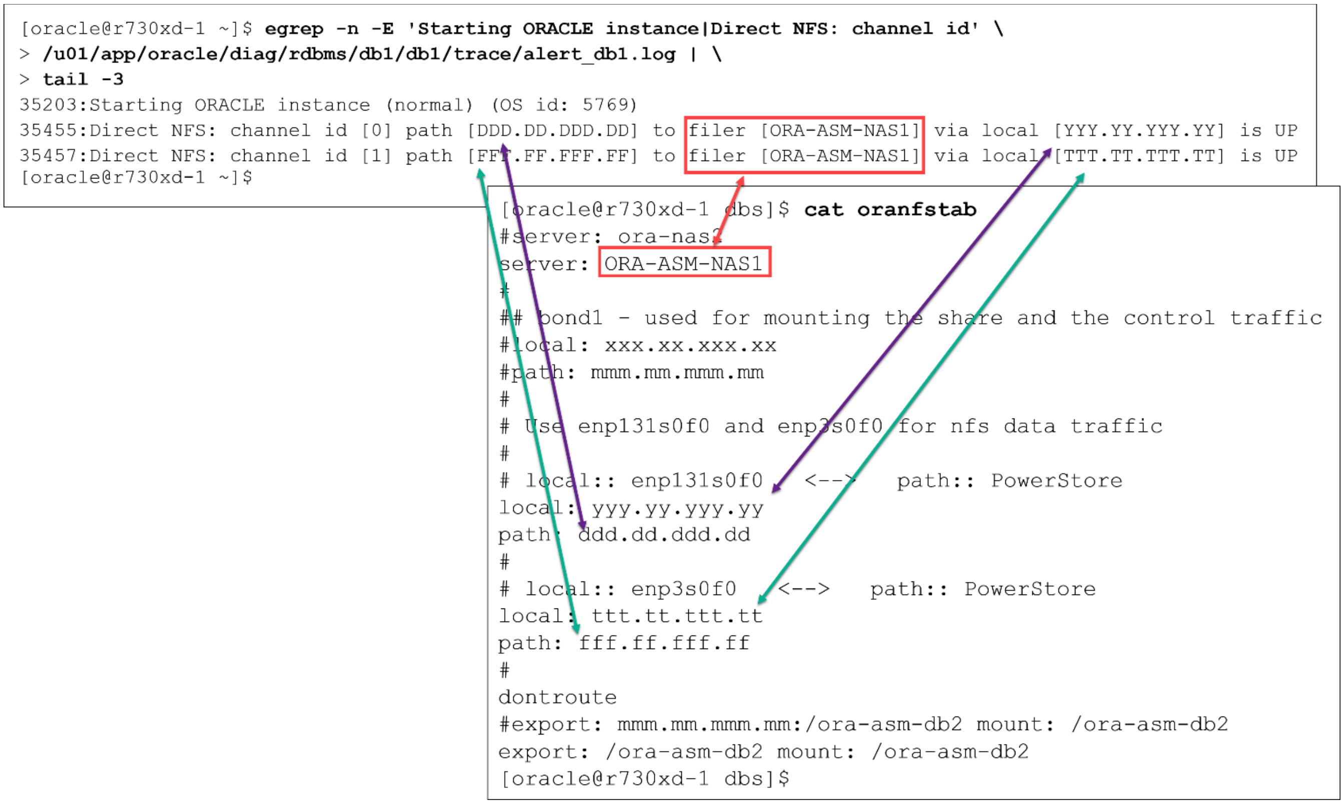
```
SQL> col filename format a30
SQL> select filename
  2       , svr_id
  3       , con_id
  4     from v$dnfs_files
  5     order by 1, 2, 3;

FILENAME                           SVR_ID     CON_ID
------------------------------ ---------- ----------
/ora-asm-db2/asm-data1                  1          0
/ora-asm-db2/asm-data2                  1          0
/ora-asm-db2/asm-redo1                  1          0
/ora-asm-db2/asm-redo2                  1          0

SQL>
```

Figure 61. Displaying database files using dNFS

Verifying correct endpoints in dNFS paths

To verify correct endpoints are being used with dNFS paths, look for the `Direct NFS: channel id` string in the database alert log. The `path` and `local` IP addresses shown in the alert log should match all the local:path mappings in `oranfstab` for the appropriate NAS server hosting the database. If Oracle automatically detects the local host interface because `oranfstab` is not defined, ensure that the chosen interface is intended to be used for the dNFS channel.

The following figure shows that the database instance created a dNFS channel for each path endpoint specified in `oranfstab`:



```
[oracle@r730xd-1 ~]$ egrep -n -E 'Starting ORACLE instance|Direct NFS: channel id' \
> /u01/app/oracle/diag/rdbms/db1/db1/trace/alert_db1.log | \
> tail -3
35203:Starting ORACLE instance (normal) (OS id: 5769)
35455:Direct NFS: channel id [0] path [DDD.DD.DDD.DD] to filer [ORA-ASM-NAS1] via local [YYY.YY.YYY.YY] is UP
35457:Direct NFS: channel id [1] path [FFF.FF.FFF.FF] to filer [ORA-ASM-NAS1] via local [TTT.TT.TTT.TT] is UP
[oracle@r730xd-1 ~]$
```

```
[oracle@r730xd-1 dbs]$ cat oranfstab
#server: ora-nas2
server: ORA-ASM-NAS1
#
## bond1 - used for mounting the share and the control traffic
#local: xxx.xx.xxx.xx
#path: mmm.mm.mmm.mm
#
# Use enp131s0f0 and enp3s0f0 for nfs data traffic
#
# local:: enp131s0f0   <-->   path:: PowerStore
local: yyy.yy.yyy.yy
path: ddd.dd.ddd.dd
#
# local:: enp3s0f0  <-->   path:: PowerStore
local: ttt.tt.ttt.tt
path: fff.ff.fff.ff
#
dontroute
#export: mmm.mm.mmm.mm:/ora-asm-db2 mount: /ora-asm-db2
export: /ora-asm-db2 mount: /ora-asm-db2
[oracle@r730xd-1 dbs]$
```

Figure 62. dNFS channels and corresponding oranfstab local:path mappings

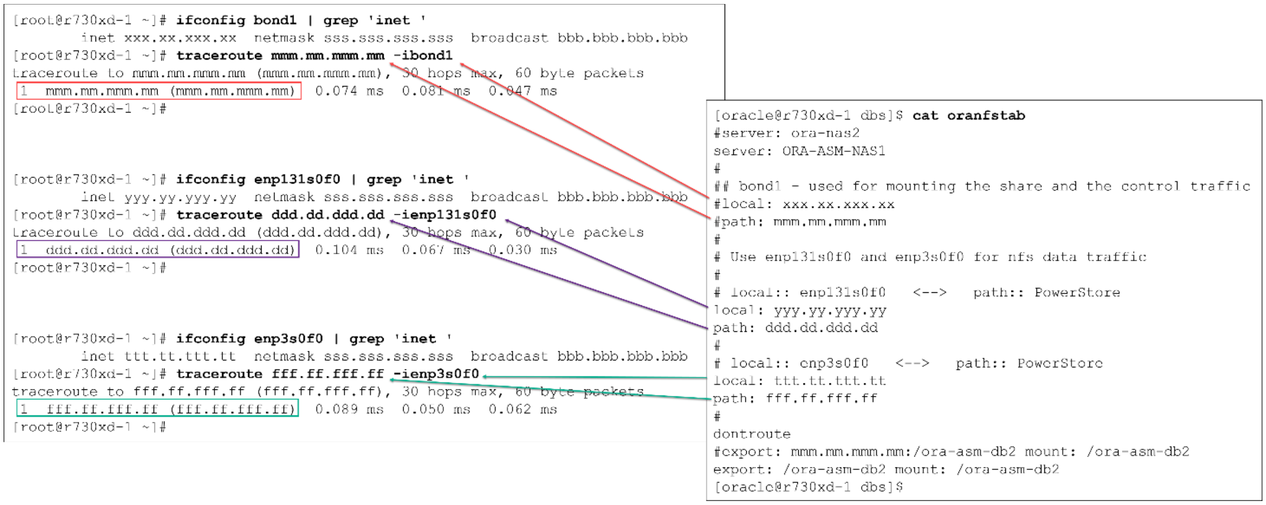
Verifying correct operating system network routes are being used for dNFS data traffic

If the operating system network routes are set up correctly for dNFS, a traceroute from the database server can reach the bonded NAS network interface in PowerStore. Also, netstat should report increasing TX/RX metrics on the interfaces intended for dNFS during database activity.

If dNFS is enabled but `oranfstab` is not configured, the Ethernet traffic for dNFS will reside on the interface used to mount the NAS share.

With or without `oranfstab`, dNFS activity will be displayed as changes to RX-OK and TX-OK values from `netstat`.

The following figure shows one hop between the endpoints in the routes:

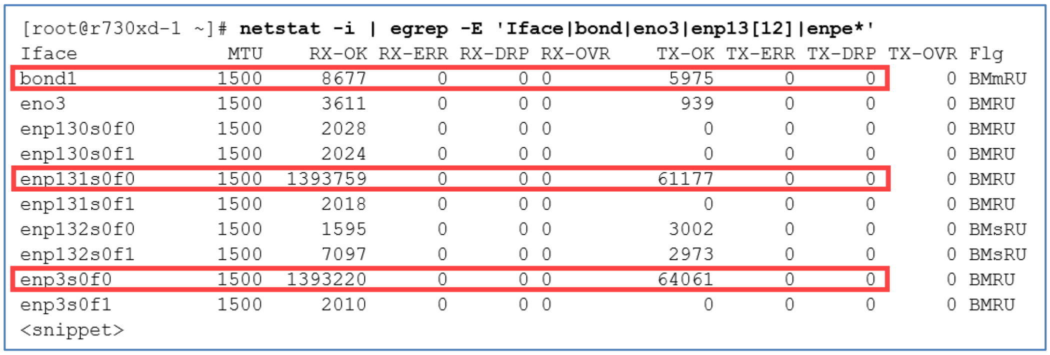


```
[root@r730xd-1 ~]# ifconfig bond1 | grep 'inet '
        inet xxx.xx.xxx.xx  netmask sss.sss.sss.sss  broadcast bbb.bbb.bbb.bbb
[root@r730xd-1 ~]# traceroute mmm.mm.mmm.mm -ibond1
traceroute to mmm.mm.mmm.mm (mmm.mm.mmm.mm), 30 hops max, 60 byte packets
 1  mmm.mm.mmm.mm (mmm.mm.mmm.mm)  0.074 ms  0.081 ms  0.047 ms
[root@r730xd-1 ~]#


[root@r730xd-1 ~]# ifconfig enp131s0f0 | grep 'inet '
        inet yyy.yy.yyy.yy  netmask sss.sss.sss.sss  broadcast bbb.bbb.bbb.bbb
[root@r730xd-1 ~]# traceroute ddd.dd.ddd.dd -ienp131s0f0
traceroute to ddd.dd.ddd.dd (ddd.dd.ddd.dd), 30 hops max, 60 byte packets
 1  ddd.dd.ddd.dd (ddd.dd.ddd.dd)  0.104 ms  0.067 ms  0.030 ms
[root@r730xd-1 ~]#


[root@r730xd-1 ~]# ifconfig enp3s0f0 | grep 'inet '
        inet ttt.tt.ttt.tt  netmask sss.sss.sss.sss  broadcast bbb.bbb.bbb.bbb
[root@r730xd-1 ~]# traceroute fff.ff.fff.ff -ienp3s0f0
traceroute to fff.ff.fff.ff (fff.ff.fff.ff), 30 hops max, 60 byte packets
 1  fff.ff.fff.ff (fff.ff.fff.ff)  0.089 ms  0.050 ms  0.062 ms
[root@r730xd-1 ~]#
```

```
[oracle@r730xd-1 dbs]$ cat oranfstab
#server: ora-nas2
server: ORA-ASM-NAS1
#
## bond1 - used for mounting the share and the control traffic
#local: xxx.xx.xxx.xx
#path: mmm.mm.mmm.mm
#
# Use enp131s0f0 and enp3s0f0 for nfs data traffic
#
# local:: enp131s0f0   <-->   path:: PowerStore
local: yyy.yy.yyy.yy
path: ddd.dd.ddd.dd
#
# local:: enp3s0f0   <-->   path:: PowerStore
local: ttt.tt.ttt.tt
path: fff.ff.fff.ff
#
dontroute
#export: mmm.mm.mmm.mm:/ora-asm-db2 mount: /ora-asm-db2
export: /ora-asm-db2 mount: /ora-asm-db2
[oracle@r730xd-1 dbs]$
```

Figure 63. Traceroutes between PS NAS bonded network interfaces and database server network interfaces

The following figure shows send/receive (TX/RX) statistics for the database server Ethernet interfaces. The `netstat` statistics can be monitored during database activity to ensure that the interfaces for dNFS data traffic have TX/RX activity. `eno3` is the public Ethernet interface. Low values of RX and TX for `eno3` are a good indication that the public Ethernet interface is not used by dNFS.

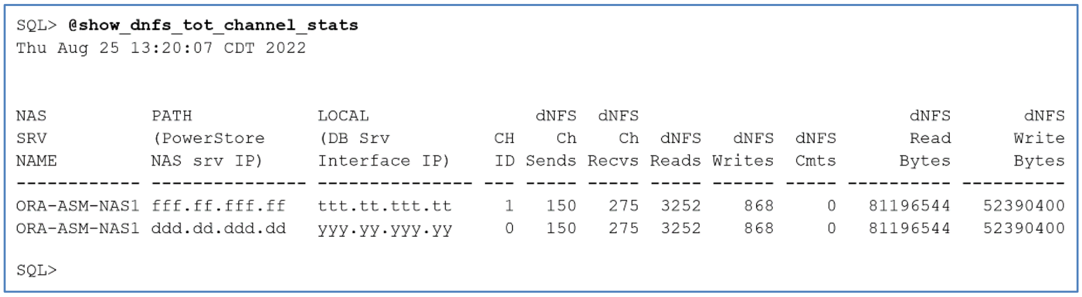


```
[root@r730xd-1 ~]# netstat -i | egrep -E 'Iface|bond|eno3|enp13[12]|enpe*'
Iface             MTU    RX-OK RX-ERR RX-DRP RX-OVR    TX-OK TX-ERR TX-DRP TX-OVR Flg
bond1            1500     8677      0      0 0          5975      0      0      0 BMmRU
eno3             1500     3611      0      0 0           939      0      0      0 BMRU
enp130s0f0       1500     2028      0      0 0             0      0      0      0 BMRU
enp130s0f1       1500     2024      0      0 0             0      0      0      0 BMRU
enp131s0f0       1500  1393759      0      0 0         61177      0      0      0 BMRU
enp131s0f1       1500     2018      0      0 0             0      0      0      0 BMRU
enp132s0f0       1500     1595      0      0 0          3002      0      0      0 BMsRU
enp132s0f1       1500     7097      0      0 0          2973      0      0      0 BMsRU
enp3s0f0         1500  1393220      0      0 0         64061      0      0      0 BMRU
enp3s0f1         1500     2010      0      0 0             0      0      0      0 BMRU
<snippet>
```

Figure 64. Netstat statistics of Ethernet interfaces used by xNFS and dNFS

Note: The `netstat` statistics are counters that are only reset with a system boot.

The following SQL*Plus output shows dNFS send/receive (TX/RX) statistics for each dNFS channel. If dNFS is correctly configured, all channels appear in the output and have read and write activity.



```
SQL> @show_dnfs_tot_channel_stats
Thu Aug 25 13:20:07 CDT 2022


NAS          PATH            LOCAL                 dNFS  dNFS                          dNFS       dNFS
SRV          (PowerStore     (DB Srv          CH    Ch    Ch  dNFS   dNFS  dNFS       Read      Write
NAME         NAS srv IP)     Interface IP)    ID Sends Recvs Reads Writes  Cmts      Bytes      Bytes
------------ --------------- --------------- --- ----- ----- ----- ------ ----- ---------- ----------
ORA-ASM-NAS1 fff.ff.fff.ff   ttt.tt.ttt.tt     1   150   275  3252    868     0   81196544   52390400
ORA-ASM-NAS1 ddd.dd.ddd.dd   yyy.yy.yyy.yy     0   150   275  3252    868     0   81196544   52390400

SQL>
```

Figure 65. Current dNFS statistics

Oracle dynamic dNFS views

Eight dNFS dynamic performance views are available in Oracle for monitoring ODM NFS storage devices. Four of the eight views are for Standalone deployments, and four are for RAC deployments. For a full description of the dynamic tables for Standalone deployments, see *Oracle Database—Database Reference*.

Table 10. Standalone deployment dNFS dynamic performance views

dNFS dynamic performance view	Description
v$dnfs_channels	Displays open network paths/channels to servers for which dNFS is providing files
v$dnfs_files	Displays database files opened by dNFS
v$dnfs_servers	Displays servers accessed using dNFS
v$dnfs_stats	Displays performance statistics for dNFS

Data protection and recovery

Introduction

PowerStore can create point-in-time snapshots of one or more volumes. Snapshots are space-efficient because they consist of pointers to frozen data blocks and use redirect-on-write technology. They consume no extra space unless, for example, a thin clone is created from a snapshot and mapped to a host, and new data is written to the thin clone.

Snapshots can be created manually or automatically using snapshot rules within protection policies. For snapshot cleanup, a service runs hourly in the background in PowerStore and removes any expired snapshots.

Hosts cannot access snapshots unless a thin clone is created from the snapshot and presented to the host.

Snapshots and recoveries with Oracle

Snapshots provide a fast and space-efficient way to protect Oracle databases. When using snapshots with Oracle databases, keep in mind these important considerations to ensure a successful database recovery:

- All Oracle database LUNs must be protected as a set. Enable the **Apply write-order consistency to protect all volume group members** attribute to ensure that the LUNs reside in the same volume group. This attribute ensures that the snapshot applies at a single point in time to the member volumes of a write-order consistent volume group.

- Snapshots do not replace Oracle RMAN for regular database backups. However, they offer additional protection to the database and allow offloading RMAN processing to an alternate host.
- Snapshots can be created or deleted manually or automatically based on the PowerStore protection policy.
- Special consideration of using BEGIN and END backup might be necessary before and after a snap is taken of pre-12c Oracle databases. For more information, see Oracle documentation.

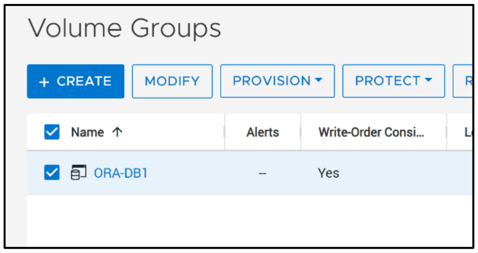
PowerStore has three data-recovery mechanisms that behave differently depending on the usage scenario.

- **Thin clone**: A thin clone takes an existing snapshot from a parent volume and creates a child volume from that point in time.
- **Refresh**: Using the refresh operation, snapshot data can replace existing data in the volume. The existing data is removed, and snapshot data from the new source is copied to it in place.
- **Restore**: The restore operation replaces the contents of a parent storage resource with data from an associated snapshot. The operation resets the data in the parent storage resource to the point in time that the snapshot was taken.

Creating a snapshot

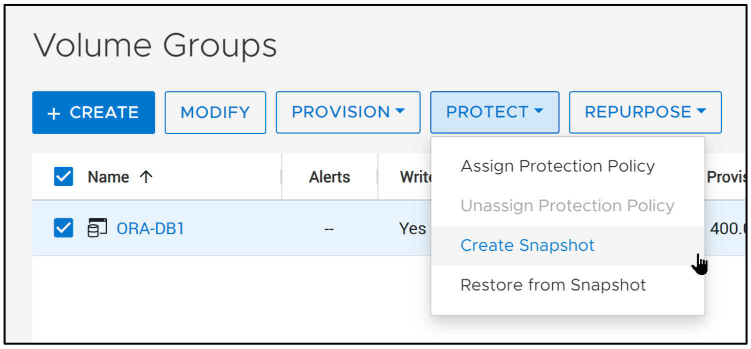
Perform the following steps to create a snapshot of a database:

1. In PowerStore Manager, select **Storage** > **Volume Groups**.
2. Click the checkbox associated with the volume group containing the database (for example, ORA-DB1).



☑ Name ↑	Alerts	Write-Order Consi...	L
☑ ORA-DB1	--	Yes	

3. Click **PROTECT** > **Create Snapshot**.



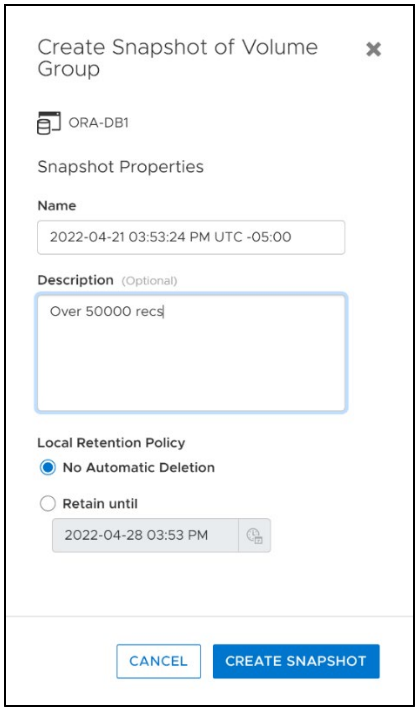
☑ Name ↑	Alerts	Writ	Provis
☑ ORA-DB1	--	Yes	400.(

4. In the **Create Snapshot of Volume Group** dialog box:

a. Consider changing the default-generated name.

b. Click **No Automatic Deletion** for manual deletion.

By default, the snapshot is created with a retention period. A use case for manual deletion is a snapshot intended for gold images. Dell Technologies recommends evaluating the needs of the business before selecting and using **No Automatic Deletion**. Snapshots created with **No Automatic Deletion** consume system resources until they are manually deleted.



Create Snapshot of Volume Group

ORA-DB1

Snapshot Properties

Name

2022-04-21 03:53:24 PM UTC -05:00

Description (Optional)

Over 50000 recs

Local Retention Policy

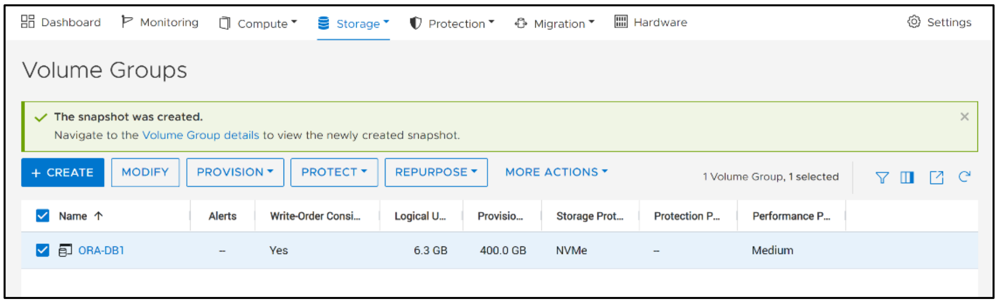
● No Automatic Deletion

○ Retain until

2022-04-28 03:53 PM

5. Click **CREATE SNAPSHOT**.

PowerStore displays a message indicating that the snapshot was created.



Volume Groups

✓ The snapshot was created.
Navigate to the Volume Group details to view the newly created snapshot.

1 Volume Group, 1 selected

☑	Name ↑	Alerts	Write-Order Consi...	Logical U...	Provisio...	Storage Prot...	Protection P...	Performance P...
☑	ORA-DB1	–	Yes	6.3 GB	400.0 GB	NVMe	–	Medium

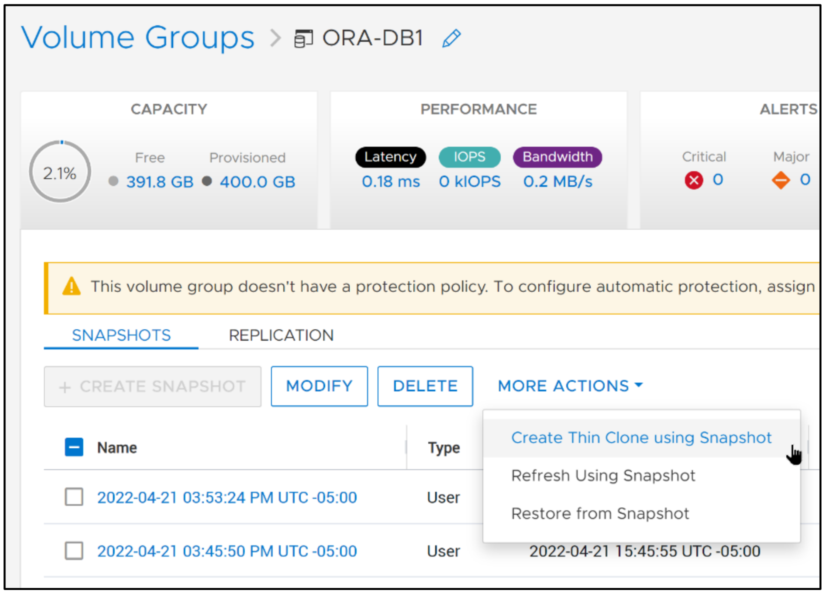
Thin clones

Thin clones are created from a snapshot. Because the thin clone volume shares data blocks with the parent, the thin clone uses no extra capacity. However, when the thin clone is mapped to a host that writes to it, only the deltas are written to the thin clone.

Perform the following steps to create a thin clone:

1. In PowerStore Manager, select **Storage** > **Volume Groups**.

2. Click the volume group.
3. Select the **PROTECTION** tab.
4. Select the checkbox of the snapshot.
5. Select **MORE ACTIONS** > **Create Thin Clone using Snapshot**.



This volume group doesn't have a protection policy. To configure automatic protection, assign

Name	Type	
□ 2022-04-21 03:53:24 PM UTC -05:00	User	
□ 2022-04-21 03:45:50 PM UTC -05:00	User	2022-04-21 15:45:55 UTC -05:00

6. Complete the **Create Thin Clone** wizard.
7. Click **CLONE**.

AppSync

Adding to the snapshot and replication abilities of PowerStore, Dell Technologies offers data-protection software that integrates with the PowerStore data protection features. Dell AppSync is optional software that can be used to enhance the overall application protection.

AppSync is software that enables integrated Copy Data Management (iCDM) with Dell primary storage systems, including PowerStore. It supports many applications, including Oracle, and storage replication technologies. For the latest support information, see the AppSync Support Matrix at the Dell E-lab Navigator.

AppSync simplifies and automates the process of creating and using snapshots of production data. By abstracting the underlying storage and replication technologies, and through application integration, AppSync allows application owners to manage data-copy needs themselves. The storage administrator only needs to be concerned with initial setup and policy management, resulting in a more agile environment.

References

Dell Technologies documentation

The following Dell Technologies documentation provides other information related to this document. Access to these documents depends on your login credentials. If you do not have access to a document, contact your Dell Technologies representative.

- PowerStore Info Hub
- Dell PowerEdge R730 and R730xd Technical Guide
- Configuring Dell EMC Networking S5248F-ON switches

Oracle documentation

See the following referenced or recommended Oracle resources at the Oracle Online Documentation Portal:

- Oracle Automatic Storage Management—Administrator's Guide 19c, F34287-06 July 2022
- Oracle Database—Database Administrator's Guide, 19c, E96348-15 August 2022
- Oracle Database—Database Performance Tuning Guide 19c, E96347-06 September 2022
- Oracle Grid Infrastructure—Grid Infrastructure Installation and Upgrade Guide, E96272-20 August 2022
- Oracle Database—Database Reference, 19c, E96196-23 September 2022

Oracle Support MOS notes

See the following referenced resources at My Oracle Support (login required):

- Doc ID 1286796.1: rp_filter for multiple private interconnects and Linux Kernel 2.6.32+.Doc ID 359515.1: Mount Options for Oracle files for RAC databases and Clusterware when used with NFS on NAS devices
- Doc ID 1620238.1: Creating File Devices On NAS/NFS FileSystems For ASM Diskgroups
- Doc ID 1003375.1: NFS-Filesystems: Hard vs Soft mounts Explained
- Doc ID 762374.1: Step by Step - Configure Direct NFS Client (DNFS) on Linux
- Doc ID 1601897.1: ASM Certification On Thin-Provisioned NFS Files
- Doc ID 731775.1: How To Create ASM Diskgroups using NFS/NAS Files?
- Doc ID 1164673.1: NFS Performance Decline Introduced by Mount Option "actimeo=0
- Doc ID 397194.1: How to Optimize NFS Performance with NFS options
- Doc ID 1552831.1: How to Configure DNFS to Use Multiple IPs
- Doc ID 1902001.1: What is ASM rebalance compact Phase and how it can be disabled
- Doc ID 604683.1: Supported Backup, Restore and Recovery Operations using Thrid Pary Snapshot Technologies
- Doc ID 221779.1: How to prepare the Oracle database for Snapshot Technologies and ensure a consistent recovery
- Doc ID 822481.1: How to Setup Direct NFS Client Multipaths in Same Subnet

Other documentation

Other assets that are referenced in the paper include:

- DB-Engines Ranking of Relational DBMS