# FlashStack Datacenter with Oracle 21c RAC Databases on Cisco UCS X-Series, Pure Storage with NVMe/FC

## Design and Deployment Guide

Published: July 2024


In partnership with:

## About the Cisco Validated Design Program

The Cisco Validated Design (CVD) program consists of systems and solutions designed, tested, and documented to facilitate faster, more reliable, and more predictable customer deployments. For more information, go to: http://www.cisco.com/go/designzone.

## Executive Summary

The IT industry has been transforming rapidly to converged infrastructure, which enables faster provisioning, scalability, lower data center costs, simpler management infrastructure with technology advancement. There is a current industry trend for pre-engineered solutions which standardize the data center infrastructure and offers operational efficiencies and agility to address enterprise applications and IT services. This standardized data center needs to be seamless instead of siloed when spanning multiple sites, delivering a uniform network and storage experience to the compute systems and end users accessing these data centers.

The FlashStack solution provides best of breed technology from Cisco Unified Computing System (Cisco UCS) and Pure Storage to gain the benefits that converged infrastructure brings to the table. FlashStack solution provides the advantage of having the compute, storage, and network stack integrated with the programmability of the the Cisco Unified Computing System. Cisco Validated Designs (CVDs) consist of systems and solutions that are designed, tested, and documented to facilitate and improve customer deployments.

This Cisco Validated Design (CVD) describes a FlashStack reference architecture for deploying an end-to-end 100Gbps network for deploying highly available Oracle Multitenant RAC 21c Databases environment using NVMe/FC on Pure Storage FlashArray//XL170 using Cisco Unified Computing System X-Series, Cisco Fabric Interconnect Switches, Cisco Nexus Switches, Cisco MDS Switches and Red Hat Enterprise Linux into a Datacenter platform with the ability to monitor and manage components from the cloud using Cisco Intersight.

By moving the management from the fabric interconnects into the cloud, the solution can respond to the speed and scale of your deployments with a constant stream of new capabilities delivered from Cisco Intersight software-as-a-service model at cloud-scale. For those that require management within a secure datacenter, Cisco Intersight is also offered as an on-site appliance with both connected and internet disconnected options.

# Solution Overview

This chapter contains the following:



## Introduction

The Cisco Unified Computing System X-Series (Cisco UCSX) with Intersight Managed Mode (IMM) is a modular compute system, configured and managed from the cloud. It is designed to meet the needs of modern applications and to improve operational efficiency, agility, and scale through an adaptable, future-ready, modular design. The Cisco Intersight platform is a Software-as-a-Service (SaaS) infrastructure lifecycle management platform that delivers simplified configuration, deployment, maintenance, and support.

Powered by the Cisco Intersight cloud-operations platform, the Cisco UCS X-Series enables the next-generation cloud-operated FlashStack infrastructure that not only simplifies data-center management but also allows the infra-structure to adapt to the unpredictable needs of modern applications as well as traditional workloads.

This CVD describes how the Cisco UCS X-Series can be used in conjunction with Pure Storage FlashArray//XL170 systems to implement a mission-critical application such as an Oracle 21c RAC databases solution using modern SANs on NVMe over Fabrics (NVMe over Fibre-Channel or NVMe/FC).

## Audience

The intended audience for this document includes, but is not limited to customers, field consultants, database administrators, IT architects, Oracle database architects, and sales engineers who want to deploy Oracle RAC 21c database solution on FlashStack Converged Infrastructure with Pure Storage and the Cisco UCS X-Series platform using Intersight Managed Mode (IMM) to deliver IT efficiency and enable IT innovation. A working knowledge of Oracle RAC Database, Linux, Storage technology, and Network is assumed but is not a prerequisite to read this document.

## Purpose of this Document

The purpose of this document is to provide step-by-step configuration and implementation guide for the FlashStack Datacenter with Cisco UCS X-Series Compute Servers, Cisco Fabric Interconnect Switches, Cisco MDS Switches, Cisco Nexus Switches and Pure Storage to deploy an Oracle RAC Database solution. Furthermore, it provides references for incorporating Cisco Intersight–managed Cisco UCS X-Series platform with end-to-end

100Gbps within a Datacenter infrastructure. This document introduces various design elements and explains various considerations and best practices for a successful deployment.

The document also highlights the design and product requirements for integrating compute, network, and storage systems to Cisco Intersight to deliver a true cloud-based integrated approach to infrastructure management. The goal of this document is to build, validate and evaluate the performance of this FlashStack reference architecture while running various types of Oracle OLTP and DSS database workloads using various benchmarking exercises and showcase Oracle database server read latency, peak sustained throughput and IOPS under various stress tests.

## What's New in this Release?

The following design elements distinguish this version of FlashStack from previous models:

- Deploying and managing Cisco UCS X9508 chassis equipped with Cisco UCS X210c M7 compute nodes from the cloud using Cisco Intersight
- Integration of low latency and high performance NVMe Pure Storage FlashArray//XL170
- Support for the NVMe/FC on Cisco UCS and Pure Storage
- Integration of the 5th Generation Cisco UCS 15000 Series VICs into FlashStack Datacenter
- Integration of the Cisco UCSX-I-9108-100G Intelligent Fabric Module into the Cisco X-Series 9508 Chassis
- Implementation of end-to-end 100G network to optimize the I/O path between Oracle databases and the RAC Servers
- Implementation of FC and NVMe/FC on the same architecture
- Validation of Oracle 21c Grid Infrastructure and 21c Databases

## FlashStack System Overview

The FlashStack platform, developed by Cisco and Pure Storage, is a flexible, integrated infrastructure solution that delivers pre-validated storage, networking, and server technologies. Composed of defined set of hardware and software, this FlashStack solution is designed to increase IT responsiveness to organizational needs and reduce the cost of computing with maximum uptime and minimal risk.

Cisco and Pure Storage have carefully validated and verified the FlashStack solution architecture and its many use cases while creating a portfolio of detailed documentation, information, and references to assist customers in transforming their data centers to this shared infrastructure model.

FlashStack provides the following differentiators:

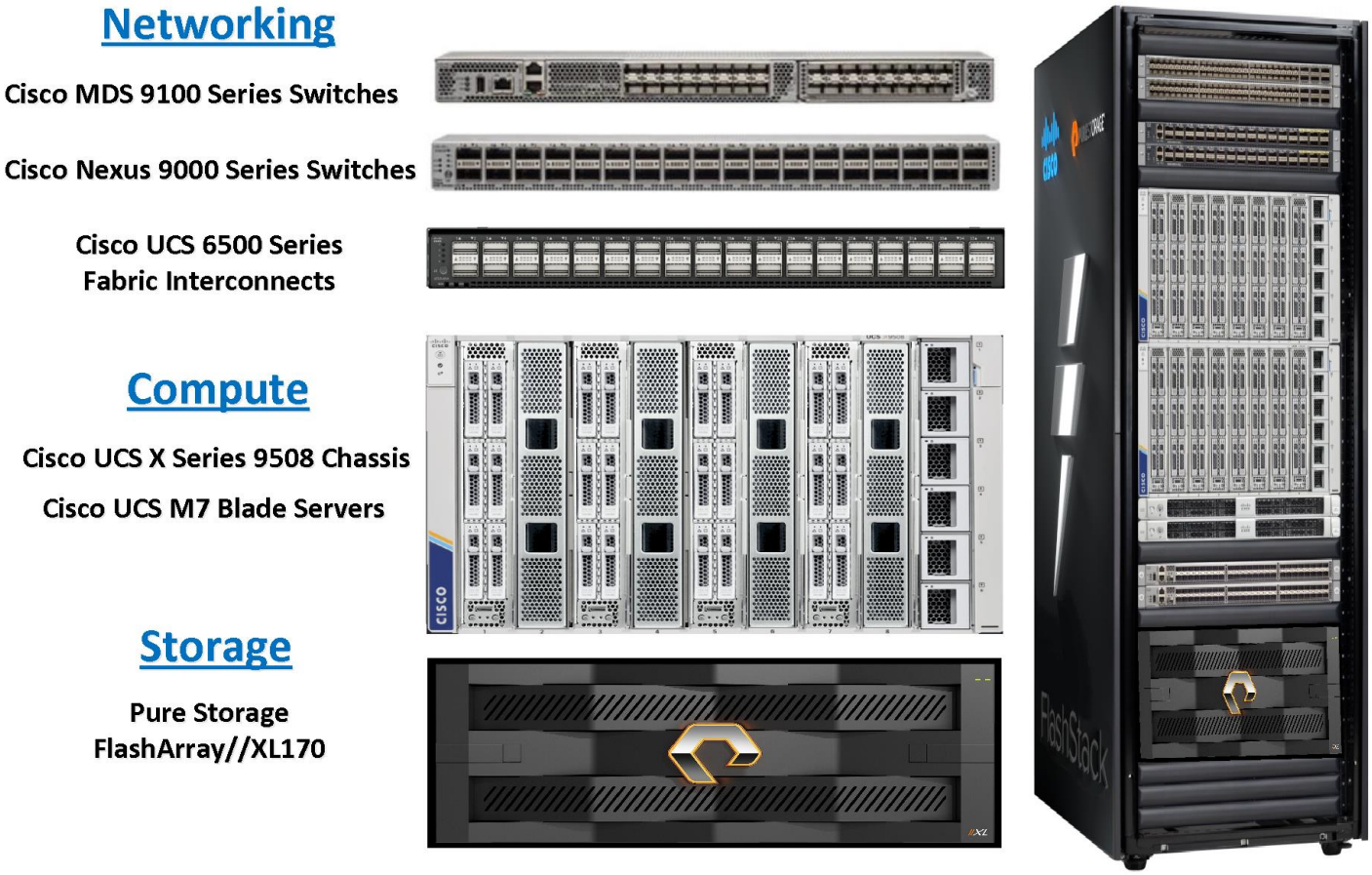
- A cohesive, integrated system that is managed, serviced, and tested as a whole
- Reduces Operational Risk – Highly available architecture with no single point of failure, non-disruptive operations, and no downtime
- Guarantee customer success with prebuilt, pre-tested drivers and Oracle database software.
- Cisco Validated Designs (CVDs) explaining a variety of reference architectures and use cases

## Key Elements of a Datacenter FlashStack Solution

Cisco and Pure storage have carefully validated and verified the FlashStack solution architecture and its many use cases while creating a portfolio of detailed documentation, information, and references to assist customers in transforming their data centers to this shared infrastructure model.

This reference FlashStack Datacenter architecture is built using the following infrastructure components for compute, network, and storage:

- Compute – Cisco UCS X-Series Chassis with Cisco UCS X210c M7 Blade Servers
- Network – Cisco UCS Fabric Interconnects, Cisco Nexus switches and Cisco MDS switches
- Storage – Pure Storage FlashArray//XL170




All FlashStack components have been integrated so you can deploy the solution quickly and economically while eliminating many of the risks associated with researching, designing, building, and deploying similar solutions from the foundation.

Each of the component families (Cisco UCS, Cisco FI, Cisco Nexus, Cisco MDS and Pure Storage) shown in the figure above offers platform and resource options to scale up or scale out the infrastructure while supporting the same features. The design is flexible enough that the networking, computing, and storage can fit in one data center rack or be deployed according to a customer's data center design. The reference architecture reinforces

the "wire-once" strategy, because as additional storage is added to the architecture, no re-cabling is required from the hosts to the Cisco UCS fabric interconnect.

This FlashStack Datacenter solution for deploying Oracle RAC 21c Databases is built using the following hardware components:

- Fifth-generation Cisco UCS 6536 Fabric Interconnects to support 10/25/40/100GbE and Cisco Intersight platform to deploy, maintain and support UCS and FlashStack components.
- Two Cisco UCS X9508 Chassis with each chassis having two Cisco UCSX-I-9108-100G Intelligent Fabric Modules to deploy end to end 100GE connectivity.
- Total of eight Cisco UCS X210c M7 Compute Nodes (4 Nodes per Chassis) with each node having one Cisco Virtual Interface Cards (VICs) 15231.
- High-speed Cisco NX-OS-based Cisco Nexus C9336C-FX2 switching design to support up to 100GE connectivity and Cisco MDS 9132T Fibre Channel Switches for Storage Networking
- NVMe Pure Storage FlashArray//XL170 with 100GE/32GFC connectivity.

There are two modes to configure Cisco UCS, one is UCSM (UCS Managed), and the other is IMM (Intersight Managed Mode). This reference solution was deployed using Intersight Managed Mode (IMM). The best practices and setup recommendations are described later in this document.

**Note:** In this validated and deployed solution, the Cisco UCS X-Series is only supported in IMM mode.

## Solution Summary

This solution provides an end-to-end 100Gbps Ethernet/FCoE-capable architecture to demonstrate the benefits for running Oracle RAC Database 21c environment with superior performance, scalability and high availability using NVMe over Fibre Channel (NVMe/FC).

NVMe-oF extends the high-performance and low-latency benefits of NVMe across network fabrics that connect servers and storage. NVMe-oF takes the lightweight and streamlined NVMe command set, and the more efficient queueing model, and replaces the PCIe transport with alternate transports, like Fibre Channel, RDMA over Converged Ethernet (RoCE v2), TCP. NVMe over Fibre Channel (NVMe/FC) is implemented through the Fibre Channel NVMe (FC-NVMe) standard which is designed to enable NVMe based message commands to transfer data and status information between a host computer and a target storage subsystem over a Fibre Channel network fabric.

Most high-performance latency sensitive applications and workloads are running on FCP today. Since the NVMe/FC and Fibre Channel networks use the same underlying transport protocol (FCP), they can use common hardware components. It's even possible to use the same switches, cables, and storage to communicate with both protocols at the same time. The ability to use either protocol by itself or both at the same time on the same hardware makes transitioning from FCP to NVMe/FC both simple and seamless.

Large-scale block flash-based storage environments that use Fibre Channel are the most likely to adopt NVMe over FC. FC-NVMe offers the same structure, predictability, and reliability characteristics for NVMe-oF that Fibre Channel does for SCSI. Plus, NVMe-oF traffic and traditional SCSI-based traffic can run simultaneously on the same FC fabric.

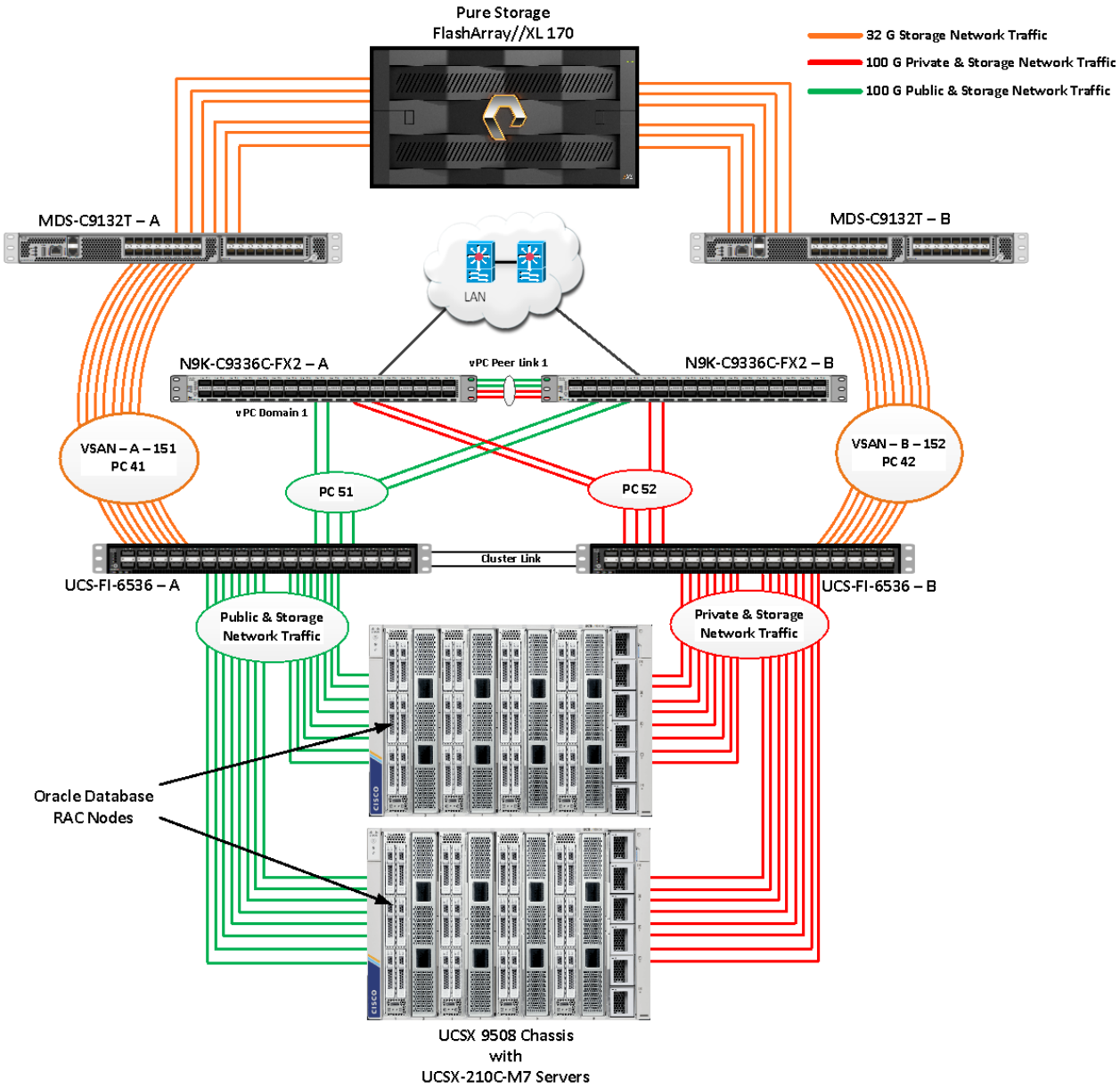
This FlashStack solution showcases the Cisco UCS System with Pure Storage FlashArray//XL170 running on NVMe over FibreChannel (NVMe/FC) which can provide efficiency and performance of NVMe, and the benefits of

all-flash robust scale out storage system that combines low-latency performance with comprehensive data management, built-in efficiencies, integrated data protection, multiprotocol support, and nondisruptive operations.

## Physical Topology

Figure 1 shows the architecture diagram of the FlashStack components to deploy an eight node Oracle RAC 21c Database solution on NVMe/FC. This reference design is a typical network configuration that can be deployed in a customer's environment.

**Figure 1. FlashStack components architecture**




As shown in Figure 1, a pair of Cisco UCS 6536 Fabric Interconnects (FI) carries both storage and network traffic from the Cisco UCS X210c M7 server with the help of Cisco Nexus 9336C-FX2 switches and Cisco MDS 9132T

switches. The Fabric Interconnects and the Cisco Nexus Switches are clustered with the peer link between them to provide high availability.

As illustrated in Figure 1, 16 (8 x 100G link per chassis) links from the blade server chassis go to Fabric Interconnect – A. Similarly, 16 (8 x 100G link per chassis) links from the blade server chassis go to Fabric Interconnect – B. Fabric Interconnect – A links are used for Oracle Public Network Traffic (VLAN-135) and Storage Network Traffic (VSAN 151) shown as green lines while Fabric Interconnect – B links are used for Oracle Private Interconnect Traffic (VLAN 10) and Storage Network Traffic (VSAN 152) shown as red lines. Two virtual Port-Channels (vPCs) are configured to provide public network and private network traffic paths for the server blades to northbound Nexus switches.

FC and NVMe/FC Storage access from both Fabric Interconnects to MDS Switches and Pure Storage Array are shown as orange lines. Eight 32Gb links are connected from FI – A to MDS – A Switch. Similarly, eight 32Gb links are connected from FI – B to MDS – B Switch. The Pure Storage FlashArray//XL170 has twelve active FC connections that go to the Cisco MDS Switches. Six FC ports are connected to MDS-A, and the other six FC ports are connected to MDS-B Switch.

The Pure Storage FlashArray//XL170 SAN ports CT1 and CT2 SAN ports FC4, FC6 and FC32 are connected to MDS – A Switch while the Controller CT1 and Controller CT2 SAN ports FC5, FC7 and FC33 are connected to MDS – B Switch. Also, two FC Port-Channels (PC) are configured (vPC 41 & vPC 42) to provide storage network paths from the server blades to storage array. Each port-channel has VSANs (VSAN 151 & VSAN 152) created for application and storage network data access.

**Note:** For the Oracle RAC configuration on Cisco Unified Computing System, we recommend keeping all private interconnect network traffic local on a single Fabric interconnect. In this case, the private traffic will stay local to that fabric interconnect and will not be routed through the northbound network switch. This way all the inter server blade (or RAC node private) communications will be resolved locally at the fabric interconnects and this significantly reduces latency for Oracle Cache Fusion traffic.

Additional 1Gb management connections are needed for an out-of-band network switch that is apart from this FlashStack infrastructure. Each Cisco UCS FI, Cisco MDS and Cisco Nexus switch is connected to the out-of-band network switch, and each Pure Storage FA controller also has two connections to the out-of-band network switch.

Although this is the base design, each of the components can be scaled easily to support specific business requirements. For example, more servers or even blade chassis can be deployed to increase compute capacity, additional storage disk shelves can be deployed to improve I/O capability and throughput, and special hardware or software features can be added to introduce new features. This document guides you through the detailed steps for deploying the base architecture, as shown in Figure 1. These procedures cover everything from physical cabling to network, compute, and storage device configurations.

## Design Topology

This section describes the hardware and software components used to deploy an eight node Oracle RAC 21c Database Solution on this architecture.

The inventory of the components used in this solution architecture is listed in Table 1.

**Table 1. Table for Hardware Inventory and Bill of Material**

| Name | Model/Product ID | Description | Quantity |
|---|---|---|---|
| Cisco UCS X Blade Server Chassis | UCSX-9508 | Cisco UCS X Series Blade Server Chassis, 7RU which can house a combination of compute nodes and a pool of future I/O resources that may include GPU accelerators, disk storage, and nonvolatile memory. | 2 |
| Cisco UCS 9108 100G IFM (Intelligent Fabric Module) | UCSX-I-9108-100G | Cisco UCS 9108 100G IFM connects the I/O fabric between the Cisco UCS X9508 Chassis and 6536 Fabric Interconnects<br>800 Gb/s (8x100Gb/s) Port IO Module for compute nodes | 4 |
| Cisco UCS X210c M7 Compute Server | UCSX-210c-M7 | Cisco UCS X210c M7 2 Socket Blade Server (2x 4th Gen Intel Xeon Scalable Processors) | 8 |
| Cisco UCS VIC 15231 | UCSX-ML-V5D200G | Cisco UCS VIC 15231 2x100/200G mLOM for X Compute Node | 8 |
| Cisco UCS 6536 Fabric Interconnect | UCS-FI-6536 | Cisco UCS 6536 Fabric Interconnect providing both network connectivity and management capabilities for the system | 2 |
| Cisco MDS Switch | DS-C9132T-8PMESK9 | Cisco MDS 9132T 32-Gbps 32-Port Fibre Channel Switch | 2 |
| Cisco Nexus Switch | N9K-9336C-FX2 | Cisco Nexus 9336C-FX2 Switch | 2 |
| Pure Storage FlashArray | FlashArray//XL170 | Pure Storage All Flash NVMe Array | 1 |

**Note:** In this solution design, we used 8 identical Cisco UCS X210c M7 Blade Servers to configure the Red Hat Linux 8.9 Operating system and then deploy an 8 node Oracle RAC Databases. The Cisco UCS X210c M7 Server configuration is listed in Table 2.

**Table 2. Cisco UCS X210c M7 Compute Server Configuration**

| Cisco UCS X210c M7 Server Configuration | | |
|---|---|---|
| Processor | 2 x Intel(R) Xeon(R) Gold 6448H CPU @ 2.4 GHz 250W 32C 60MB Cache (2 x 32 CPU Cores = 64 Core Total) | PID - UCSX-CPU-I6448H |
| Memory | 16 x Samsung 32GB DDR5-4800-MHz (512 GB) | PID - UCSX-MRX32G1RE1 |
| VIC 15231 | Cisco UCS VIC 15231 Blade Server MLOM (200G for compute node) | PID - UCSX-ML-V5D200G |

| | | |
|---|---|---|
| | (2x100G through each fabric) | |

**Table 3. vNIC and vHBA Configured on each Linux Host**

| vNIC Details | |
|---|---|
| vNIC 0 (eth0) | Management and Public Network Traffic Interface for Oracle RAC. MTU = 1500 |
| vNIC 1 (eth1) | Private Server-to-Server Network (Cache Fusion) Traffic Interface for Oracle RAC. MTU = 9000 |
| vHBA0 | FC Network Traffic & Boot from SAN through MDS-A Switch |
| vHBA1 | FC Network Traffic & Boot from SAN through MDS-B Switch |
| vHBA2 | NVMe/FC Network Traffic (Oracle RAC Storage Traffic) through MDS-A Switch |
| vHBA3 | NVMe/FC Network Traffic (Oracle RAC Storage Traffic) through MDS-B Switch |
| vHBA4 | NVMe/FC Network Traffic (Oracle RAC Storage Traffic) through MDS-A Switch |
| vHBA5 | NVMe/FC Network Traffic (Oracle RAC Storage Traffic) through MDS-B Switch |
| vHBA6 | NVMe/FC Network Traffic (Oracle RAC Storage Traffic) through MDS-A Switch |
| vHBA7 | NVMe/FC Network Traffic (Oracle RAC Storage Traffic) through MDS-B Switch |
| vHBA8 | NVMe/FC Network Traffic (Oracle RAC Storage Traffic) through MDS-A Switch |
| vHBA9 | NVMe/FC Network Traffic (Oracle RAC Storage Traffic) through MDS-B Switch |

**Note:** For this solution, we configured 2 VLANs to carry public and private network traffic as well as two VSANs to carry FC and NVMe/FC storage traffic as listed in Table 4.

**Table 4. VLAN and VSAN Configuration**

| VLAN Configuration | | |
|---|---|---|
| **VLAN** | | |
| Name | ID | Description |
| Default VLAN | 1 | Native VLAN |
| Public VLAN | 135 | VLAN for Public Network Traffic |
| Private VLAN | 10 | VLAN for Private Network Traffic |

| VLAN Configuration | | |
|---|---|---|
| **VSAN** | | |
| Name | ID | Description |
| VSAN-A | 151 | FC and NVMe/FC Network Traffic through for Fabric Interconnect A |
| VSAN-B | 152 | FC and NVMe/FC Network Traffic through for Fabric Interconnect B |

This FlashStack solution consists of Pure Storage FlashArray//XL170 as listed in Table 5.

**Table 5. Pure Storage FA//XL170 Storage Configuration**

| Storage Components | Description |
|---|---|
| Pure Storage FA//XL170 | Pure Storage FlashArray//XL170 (30 x 3.9 TB NVMe SSD Drives) |
| Capacity | 116.9 TB |
| Connectivity | 12 x 32 Gb/s redundant FC, NVMe/FC<br>1 Gb/s redundant Ethernet (Management port) |
| Physical | 4 Rack Units |

**Table 6. Software and Firmware Revisions**

| Software and Firmware | Version |
|---|---|
| Cisco UCS FI 6536 | Bundle Version 4.3(4.240066) or NX-OS Version – 9.3(5)I43(4a)<br>Image Name – intersight-ucs-infra-5gfi.4.2.3e.bin |
| Cisco UCS X210c M7 Server | 5.2(0.230041)<br>Image Name – intersight-ucs-infra-5gfi.4.3.4.240066.bin |
| Cisco UCS Adapter VIC 15231 | 5.3(3.85) |
| Cisco eNIC (Cisco VIC Ethernet NIC Driver)<br>(modinfo enic) | 4.6.0.0-977.3<br>(kmod-enic-4.6.0.0-977.3.rhel8u9_4.18.0_513.5.1.x86_64) |
| Cisco fNIC (Cisco VIC FC HBA Driver)<br>(modinfo fnic) | 2.0.0.96-324.0<br>(kmod-fnic-2.0.0.96-324.0.rhel8u9.x86_64) |
| Red Hat Enterprise Linux Server | Red Hat Enterprise Linux release 8.9 |

| Software and Firmware | Version |
| --- | --- |
| | (Kernel – 4.18.0-513.5.1.el8_9.x86_64) |
| Oracle Database 21c Grid Infrastructure for Linux x86-64 | 21.3.0.0.0 |
| Oracle Database 21c Enterprise Edition for Linux x86-64 | 21.3.0.0.0 |
| Cisco Nexus 9336C-FX2 NXOS | NXOS System Version – 9.3(7) & BIOS Version – 05.45 |
| Cisco MDS 9132T Software | System Version – 9.3(2) & BIOS Version – 1.43.0 |
| Pure Storage FA//XL170 | Purity//FA 6.5.2 |
| FIO | fio-3.19-4.el8.x86_64 |
| Oracle Swingbench | 2.7 |
| SLOB | 2.5.4.0 |

# Solution Configuration

This chapter contains the following:

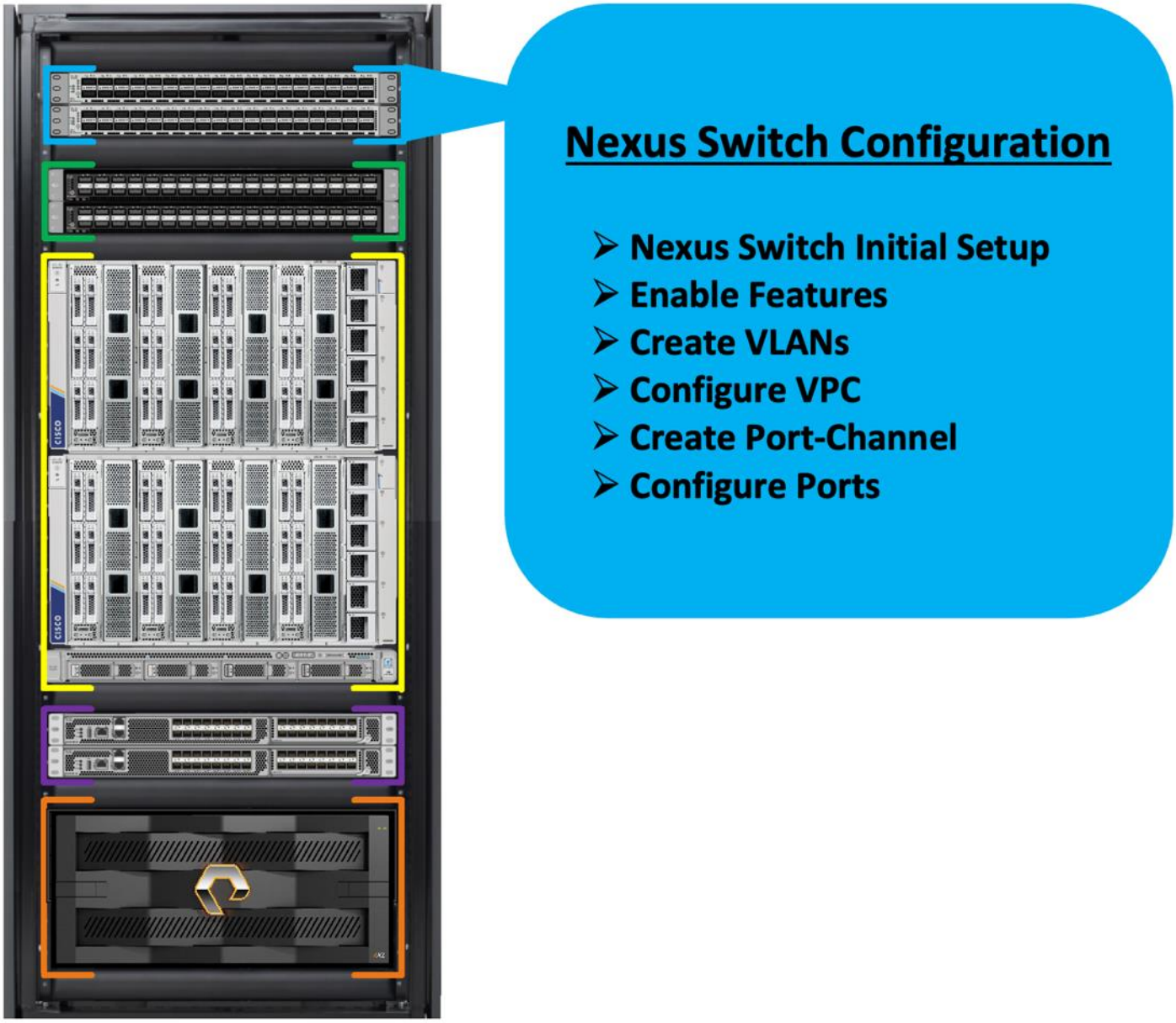


## Cisco Nexus Switch Configuration

This section details the high-level steps to configure Cisco Nexus Switches.

Figure 2 illustrates the high-level overview and steps to configure various components to deploy and test the Oracle RAC Database 21c for this FlashStack reference architecture.

Figure 2. Cisco Nexus Switch configuration architecture




The following procedures describe how to configure the Cisco Nexus switches to use in a base FlashStack environment. This procedure assumes you're using Cisco Nexus 9336C-FX2 switches deployed with the 100Gb end-to-end topology.

**Note:** On initial boot and connection to the serial or console port of the switch, the NX-OS setup should automatically start and attempt to enter Power on Auto Provisioning.

### Cisco Nexus A Switch

**Procedure 1.** Initial Setup for the Cisco Nexus A Switch

**Step 1.** To set up the initial configuration for the Cisco Nexus A Switch on <nexus-A-hostname>, run the following:

```
Abort Power on Auto Provisioning and continue with normal setup? (yes/no) [n]: yes
Do you want to enforce secure password standard (yes/no) [y]: Enter
Enter the password for "admin": <password>
Confirm the password for "admin": <password>
Would you like to enter the basic configuration dialog (yes/no): yes
Create another login account (yes/no) [n]: Enter
Configure read-only SNMP community string (yes/no) [n]: Enter
Configure read-write SNMP community string (yes/no) [n]: Enter
Enter the switch name: <nexus-A-hostname>
Continue with Out-of-band (mgmt0) management configuration? (yes/no) [y]: Enter
Mgmt0 IPv4 address: <nexus-A-mgmt0-ip>
Mgmt0 IPv4 netmask: <nexus-A-mgmt0-netmask>
Configure the default gateway? (yes/no) [y]: Enter
IPv4 address of the default gateway: <nexus-A-mgmt0-gw>
Configure advanced IP options? (yes/no) [n]: Enter
Enable the telnet service? (yes/no) [n]: Enter
Enable the ssh service? (yes/no) [y]: Enter
Type of ssh key you would like to generate (dsa/rsa) [rsa]: Enter
Number of rsa key bits <1024-2048> [1024]: Enter
Configure the ntp server? (yes/no) [n]: y
NTP server IPv4 address: <global-ntp-server-ip>
Configure default interface layer (L3/L2) [L3]: L2
Configure default switchport interface state (shut/noshut) [noshut]: Enter
Configure CoPP system profile (strict/moderate/lenient/dense/skip) [strict]: Enter
Would you like to edit the configuration? (yes/no) [n]: Enter
```

**Cisco Nexus B Switch**

Similarly, follow the steps in the procedure Initial Setup for the Cisco Nexus A Switch to setup the initial configuration for the Cisco Nexus B Switch and change the relevant switch hostname and management IP address according to your environment.

## **Procedure 1.** Configure Global Settings

Configure the global setting on both Cisco Nexus Switches.

**Step 1.** Login as admin user into the Cisco Nexus Switch A and run the following commands to set the global configurations on switch A:

```
configure terminal
feature interface-vlan
feature hsrp
```

```
feature lacp
feature vpc
feature lldp
spanning-tree port type network default
spanning-tree port type edge bpduguard default

port-channel load-balance src-dst l4port

policy-map type network-qos jumbo
  class type network-qos class-default
    mtu 9216

system qos
  service-policy type network-qos jumbo

vrf context management
  ip route 0.0.0.0/0 10.29.135.1
copy run start
```

**Step 2.** Login as admin user into the Nexus Switch B and run the same above commands to set global configurations on Nexus Switch B.

**Note:** Make sure to run copy run start to save the configuration on each switch after the configuration is completed.

## Procedure 2. VLANs Configuration

Create the necessary virtual local area networks (VLANs) on both Cisco Nexus switches.

**Step 1.** Login as admin user into the Cisco Nexus Switch A.

**Step 2.** Create VLAN 135 for Public Network Traffic and VLAN 10 for Private Network Traffic.

```
configure terminal

vlan 135
name Oracle_RAC_Public_Traffic
no shutdown

vlan 10
name Oracle_RAC_Private_Traffic
no shutdown

interface Ethernet 1/29
```

```
description To-Management-Uplink-Switch
switchport access vlan 135
speed 1000

copy run start
```

**Step 3.** Login as admin user into the Nexus Switch B and similar way, create all the VLANs 135 for Oracle RAC Public Network Traffic and VLAN 10 for Oracle RAC Private Network Traffic.

**Note:** Make sure to run copy run start to save the configuration on each switch after the configuration is completed.

## Virtual Port Channel (vPC) Summary for Network Traffic

A port channel bundles individual links into a channel group to create a single logical link that provides the aggregate bandwidth of up to eight physical links. If a member port within a port channel fails, traffic previously carried over the failed link switches to the remaining member ports within the port channel. Port channeling also load balances traffic across these physical interfaces. The port channel stays operational as long as at least one physical interface within the port channel is operational. Using port channels, Cisco NX-OS provides wider bandwidth, redundancy, and load balancing across the channels.

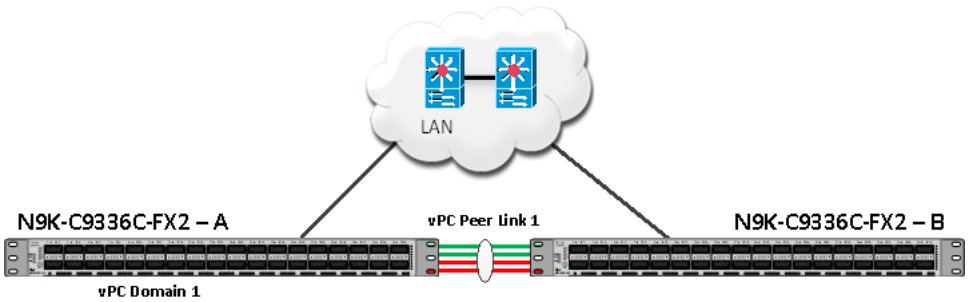
In the Cisco Nexus Switch topology, a single vPC feature is enabled to provide HA, faster convergence in the event of a failure, and greater throughput. The Cisco Nexus vPC configurations with the vPC domains and corresponding vPC names and IDs for Oracle Database Servers are listed in Table 7.

**Table 7. vPC Summary**

| vPC Domain | vPC Name | vPC ID |
|---|---|---|
| 1 | Peer-Link | 1 |
| 51 | vPC FI-A | 51 |
| 52 | vPC FI-B | 52 |

As listed in Table 7, a single vPC domain with Domain ID 1 is created across two Nexus switches to define vPC members to carry specific VLAN network traffic. In this topology, we defined a total number of 3 vPCs.

vPC ID 1 is defined as Peer link communication between the two Cisco Nexus switches. vPC IDs 51 and 52 are configured for both Cisco UCS Fabric Interconnects.



**Note:** A port channel bundles up to eight individual interfaces into a group to provide increased bandwidth and redundancy.

### Procedure 3. Create vPC Peer-Link

**Note:** For vPC 1 as Peer-link, we used interfaces 1 to 4 for Peer-Link. You may choose an appropriate number of ports based on your needs.

Create the necessary port channels between devices on both Cisco Nexus Switches.

**Step 1.** Login as admin user into the Cisco Nexus Switch A:

```
configure terminal

vpc domain 1
  peer-keepalive destination 10.29.135.56 source 10.29.135.55
  auto-recovery

interface port-channel 1
  description vPC peer-link
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  spanning-tree port type network
  vpc peer-link
  no shut

interface Ethernet 1/1
  description Peer link connected to FS-ORA-N9K-B-Eth-1/1
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  channel-group 1 mode active
  no shut

interface Ethernet 1/2
  description Peer link connected to FS-ORA-N9K-B-Eth-1/2
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  channel-group 1 mode active
  no shut

interface Ethernet 1/3
  description Peer link connected to FS-ORA-N9K-B-Eth-1/3
```

```
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  channel-group 1 mode active
  no shut

interface Ethernet 1/4
  description Peer link connected to FS-ORA-N9K-B-Eth-1/4
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  channel-group 1 mode active
  no shut

exit
copy run start
```

**Step 2.** Login as admin user into the Cisco Nexus Switch B and repeat step 1 to configure the second Cisco Nexus Switch.

**Note:** Make sure to change the description of the interfaces and peer-keepalive destination and source IP addresses.

**Step 3.** Configure the vPC on the other Cisco Nexus switch. Login as admin for the Cisco Nexus Switch B:

```
configure terminal

vpc domain 1
  peer-keepalive destination 10.29.135.55 source 10.29.135.56
  auto-recovery

interface port-channel 1
  description vPC peer-link
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  spanning-tree port type network
  vpc peer-link
  no shut

interface Ethernet 1/1
  description Peer link connected to FS-ORA-N9K-A-Eth-1/1
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  channel-group 1 mode active
  no shut
```

```
interface Ethernet 1/2
  description Peer link connected to FS-ORA-N9K-A-Eth-1/2
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  channel-group 1 mode active
  no shut

interface Ethernet 1/3
  description Peer link connected to FS-ORA-N9K-A-Eth-1/3
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  channel-group 1 mode active
  no shut

interface Ethernet 1/4
  description Peer link connected to FS-ORA-N9K-A-Eth-1/4
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  channel-group 1 mode active
  no shut

exit
copy run start
```

### Create vPC Configuration between Cisco Nexus and Fabric Interconnect Switches

This section describes how to create and configure port channel 51 and 52 for network traffic between the Cisco Nexus and Fabric Interconnect Switches.

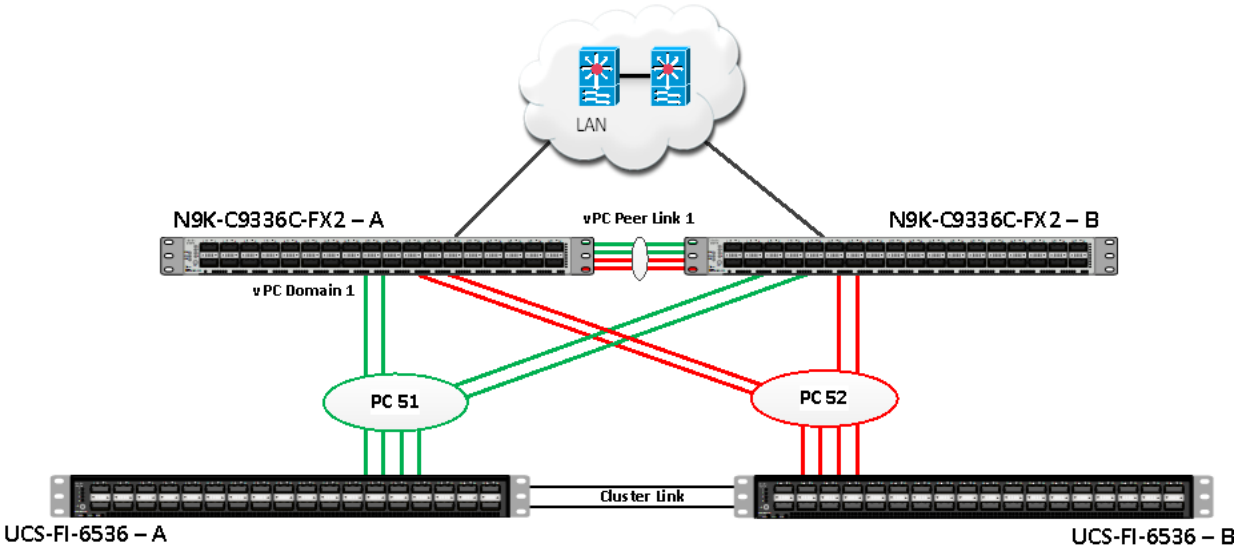



Table 8 lists the vPC IDs, allowed VLAN IDs, and ethernet uplink ports.

**Table 8. vPC IDs and VLAN IDs**

| vPC Description | vPC ID | Fabric Interconnects Ports | Cisco Nexus Switch Ports | Allowed VLANs |
|---|---|---|---|---|
| Port Channel FI-A | 51 | FI-A Port 1/27 | N9K-A Port 1/9 | 10,135<br>Note: VLAN 10 is needed for failover. |
| | | FI-A Port 1/28 | N9K-A Port 1/10 | |
| | | FI-A Port 1/29 | N9K-B Port 1/9 | |
| | | FI-A Port 1/30 | N9K-B Port 1/10 | |
| Port Channel FI-B | 52 | FI-B Port 1/27 | N9K-A Port 1/11 | 10,135<br>Note: VLAN 135 is needed for failover. |
| | | FI-B Port 1/28 | N9K-A Port 1/12 | |
| | | FI-B Port 1/29 | N9K-B Port 1/11 | |
| | | FI-B Port 1/30 | N9K-B Port 1/12 | |

## Verify the Port Connectivity on both Cisco Nexus Switches

**Figure 3. Cisco Nexus A Connectivity**

```
FS-ORA-N9K-A# show lldp neighbors
Capability codes:
  (R) Router, (B) Bridge, (T) Telephone, (C) DOCSIS Cable Device
  (W) WLAN Access Point, (P) Repeater, (S) Station, (O) Other
Device ID            Local Intf      Hold-time  Capability  Port ID
FS-ORA-N9K-B         Eth1/1          120        BR          Ethernet1/1
FS-ORA-N9K-B         Eth1/2          120        BR          Ethernet1/2
FS-ORA-N9K-B         Eth1/3          120        BR          Ethernet1/3
FS-ORA-N9K-B         Eth1/4          120        BR          Ethernet1/4
FS-ORA-FI-A          Eth1/9          120        BR          Eth1/27
FS-ORA-FI-A          Eth1/10         120        BR          Eth1/28
FS-ORA-FI-B          Eth1/11         120        BR          Eth1/27
FS-ORA-FI-B          Eth1/12         120        BR          Eth1/28
Total entries displayed: 8
```

**Figure 4. Cisco Nexus B Connectivity**

```
FS-ORA-N9K-B# show lldp neighbors
Capability codes:
  (R) Router, (B) Bridge, (T) Telephone, (C) DOCSIS Cable Device
  (W) WLAN Access Point, (P) Repeater, (S) Station, (O) Other
Device ID            Local Intf      Hold-time  Capability  Port ID
FS-ORA-N9K-A         Eth1/1          120        BR          Ethernet1/1
FS-ORA-N9K-A         Eth1/2          120        BR          Ethernet1/2
FS-ORA-N9K-A         Eth1/3          120        BR          Ethernet1/3
FS-ORA-N9K-A         Eth1/4          120        BR          Ethernet1/4
FS-ORA-FI-A          Eth1/9          120        BR          Eth1/29
FS-ORA-FI-A          Eth1/10         120        BR          Eth1/30
FS-ORA-FI-B          Eth1/11         120        BR          Eth1/29
FS-ORA-FI-B          Eth1/12         120        BR          Eth1/30
Total entries displayed: 8
```

**Procedure 1.** Configure the port channels on the Cisco Nexus Switches

**Step 1.** Login as admin user into Cisco Nexus Switch A and run the following commands:

```
configure terminal

interface port-channel 51
  description connect to FS-ORA-FI-A
  switchport mode trunk
```

```
  switchport trunk allowed vlan 1,10,135
  spanning-tree port type edge trunk
  mtu 9216
  vpc 51
  no shutdown

interface port-channel 52
  description connect to FS-ORA-FI-B
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  spanning-tree port type edge trunk
  mtu 9216
  vpc 52
  no shutdown

interface Ethernet 1/9
  description Fabric-Interconnect-A-27
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  spanning-tree port type edge trunk
  mtu 9216
  channel-group 51 mode active
  no shutdown

interface Ethernet 1/10
  description Fabric-Interconnect-A-28
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  spanning-tree port type edge trunk
  mtu 9216
  channel-group 51 mode active
  no shutdown

interface Ethernet1/11
  description Fabric-Interconnect-B-27
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  spanning-tree port type edge trunk
  mtu 9216
```

```
  channel-group 52 mode active
  no shutdown

interface Ethernet 1/12
  description Fabric-Interconnect-B-28
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  spanning-tree port type edge trunk
  mtu 9216
  channel-group 52 mode active
  no shutdown

copy run start
```

**Step 2.** Login as admin user into Cisco Nexus Switch B and run the following commands to configure the second Cisco Nexus Switch:

```
configure terminal

interface port-channel 51
  description connect to FS-ORA-FI-A
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  spanning-tree port type edge trunk
  mtu 9216
  vpc 51
  no shutdown

interface port-channel 52
  description connect to FS-ORA-FI-B
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  spanning-tree port type edge trunk
  mtu 9216
  vpc 52
  no shutdown

interface Ethernet 1/9
  description Fabric-Interconnect-A-29
  switchport mode trunk
```

```
  switchport trunk allowed vlan 1,10,135
  spanning-tree port type edge trunk
  mtu 9216
  channel-group 51 mode active
  no shutdown

interface Ethernet 1/10
  description Fabric-Interconnect-A-30
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  spanning-tree port type edge trunk
  mtu 9216
  channel-group 51 mode active
  no shutdown

interface Ethernet 1/11
  description Fabric-Interconnect-B-29
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  spanning-tree port type edge trunk
  mtu 9216
  channel-group 52 mode active
  no shutdown

interface Ethernet 1/12
  description Fabric-Interconnect-B-30
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  spanning-tree port type edge trunk
  mtu 9216
  channel-group 52 mode active
  no shutdown

copy run start
```

## Verify All vPC Status

**Procedure 1.** Verify the status of all port-channels using Cisco Nexus Switches

**Step 1.** Cisco Nexus Switch A Port-Channel Summary:

```
FS-ORA-N9K-A# show port-channel summary
Flags:  D - Down        P - Up in port-channel (members)
        I - Individual  H - Hot-standby (LACP only)
        s - Suspended   r - Module-removed
        b - BFD Session Wait
        S - Switched    R - Routed
        U - Up (port-channel)
        p - Up in delay-lacp mode (member)
        M - Not in use. Min-links not met
--------------------------------------------------------------------------------
Group Port-       Type     Protocol  Member Ports
      Channel
--------------------------------------------------------------------------------
1     Po1(SU)     Eth      LACP      Eth1/1(P)    Eth1/2(P)    Eth1/3(P)
                                     Eth1/4(P)
51    Po51(SU)    Eth      LACP      Eth1/9(P)    Eth1/10(P)
52    Po52(SU)    Eth      LACP      Eth1/11(P)   Eth1/12(P)
```

**Step 2.** Cisco Nexus Switch B Port-Channel Summary:

```
FS-ORA-N9K-B# show port-channel summary
Flags:  D - Down        P - Up in port-channel (members)
        I - Individual  H - Hot-standby (LACP only)
        s - Suspended   r - Module-removed
        b - BFD Session Wait
        S - Switched    R - Routed
        U - Up (port-channel)
        p - Up in delay-lacp mode (member)
        M - Not in use. Min-links not met
--------------------------------------------------------------------------------
Group Port-       Type     Protocol  Member Ports
      Channel
--------------------------------------------------------------------------------
1     Po1(SU)     Eth      LACP      Eth1/1(P)    Eth1/2(P)    Eth1/3(P)
                                     Eth1/4(P)
51    Po51(SU)    Eth      LACP      Eth1/9(P)    Eth1/10(P)
52    Po52(SU)    Eth      LACP      Eth1/11(P)   Eth1/12(P)
```

**Step 3.** Cisco Nexus Switch A vPC Status:

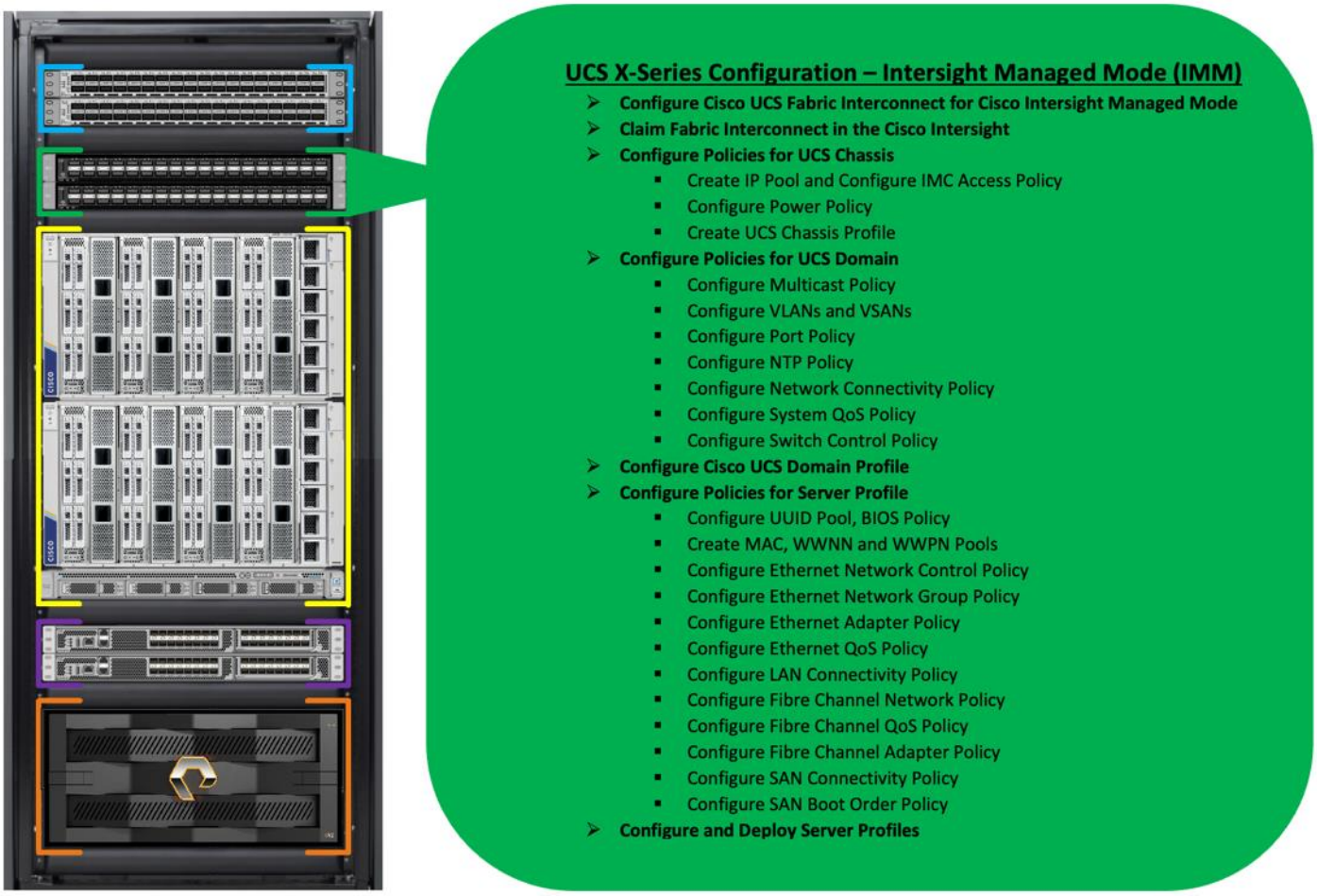
```
FS-ORA-N9K-A# show vpc brief
Legend:
                (*) - local vPC is down, forwarding via vPC peer-link

vPC domain id                     : 1
Peer status                       : peer adjacency formed ok
vPC keep-alive status             : peer is alive
Configuration consistency status  : success
Per-vlan consistency status       : success
Type-2 consistency status         : success
vPC role                          : secondary
Number of vPCs configured         : 2
Peer Gateway                      : Disabled
Dual-active excluded VLANs        : -
Graceful Consistency Check        : Enabled
Auto-recovery status              : Enabled, timer is off.(timeout = 240s)
Delay-restore status              : Timer is off.(timeout = 30s)
Delay-restore SVI status          : Timer is off.(timeout = 10s)
Operational Layer3 Peer-router    : Disabled
Virtual-peerlink mode             : Disabled

vPC Peer-link status
--------------------------------------------------------------------
id    Port   Status Active vlans
--    ----   ------ --------------------------------------------------
1     Po1    up     1,10,135

vPC status
----------------------------------------------------------------------------
Id    Port          Status Consistency Reason                Active vlans
--    ------------  ------ ----------- ------                ---------------
51    Po51          up     success     success               1,10,135

52    Po52          up     success     success               1,10,135


Please check "show vpc consistency-parameters vpc <vpc-num>" for the
consistency reason of down vpc and for type-2 consistency reasons for
any vpc.
```

**Step 4.** Cisco Nexus Switch B vPC Status:

```
FS-ORA-N9K-B# show vpc brief
Legend:
                (*) - local vPC is down, forwarding via vPC peer-link

vPC domain id                     : 1
Peer status                       : peer adjacency formed ok
vPC keep-alive status             : peer is alive
Configuration consistency status  : success
Per-vlan consistency status       : success
Type-2 consistency status         : success
vPC role                          : primary
Number of vPCs configured         : 2
Peer Gateway                      : Disabled
Dual-active excluded VLANs        : -
Graceful Consistency Check        : Enabled
Auto-recovery status              : Enabled, timer is off.(timeout = 240s)
Delay-restore status              : Timer is off.(timeout = 30s)
Delay-restore SVI status          : Timer is off.(timeout = 10s)
Operational Layer3 Peer-router    : Disabled
Virtual-peerlink mode             : Disabled

vPC Peer-link status
---------------------------------------------------------------------
id    Port   Status Active vlans
--    ----   ------ --------------------------------------------------
1     Po1    up     1,10,135

vPC status
----------------------------------------------------------------------------
Id    Port          Status Consistency Reason                Active vlans
--    ------------  ------ ----------- ------                ---------------
51    Po51          up     success     success               1,10,135

52    Po52          up     success     success               1,10,135


Please check "show vpc consistency-parameters vpc <vpc-num>" for the
consistency reason of down vpc and for type-2 consistency reasons for
any vpc.
```

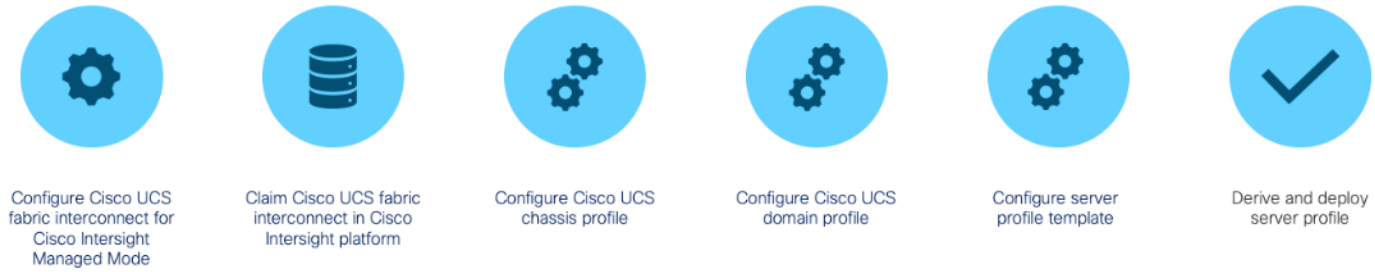
## Cisco UCS X-Series Configuration – Intersight Managed Mode (IMM)

This section details the high-level steps for the Cisco UCS X-Series Configuration in Intersight Managed Mode.




Cisco Intersight Managed Mode standardizes policy and operation management for Cisco UCS X-Series. The compute nodes in Cisco UCS X-Series are configured using server profiles defined in Cisco Intersight. These server profiles derive all the server characteristics from various policies and templates. At a high level, configuring Cisco UCS using Intersight Managed Mode consists of the steps shown in Figure 5.

**Figure 5. Configuration Steps for Cisco Intersight Managed Mode**




## Procedure 1. Configure Cisco UCS Fabric Interconnect for Cisco Intersight Managed Mode

During the initial configuration, for the management mode, the configuration wizard enables you to choose whether to manage the fabric interconnect through Cisco UCS Manager or the Cisco Intersight platform. You can switch the management mode for the fabric interconnects between Cisco Intersight and Cisco UCS Manager at

any time; however, Cisco UCS FIs must be set up in Intersight Managed Mode (IMM) for configuring the Cisco UCS X-Series system.

**Step 1.** Verify the following physical connections on the fabric interconnect:

- The management Ethernet port (mgmt0) is connected to an external hub, switch, or router.
- The L1 ports on both fabric interconnects are directly connected to each other.
- The L2 ports on both fabric interconnects are directly connected to each other.

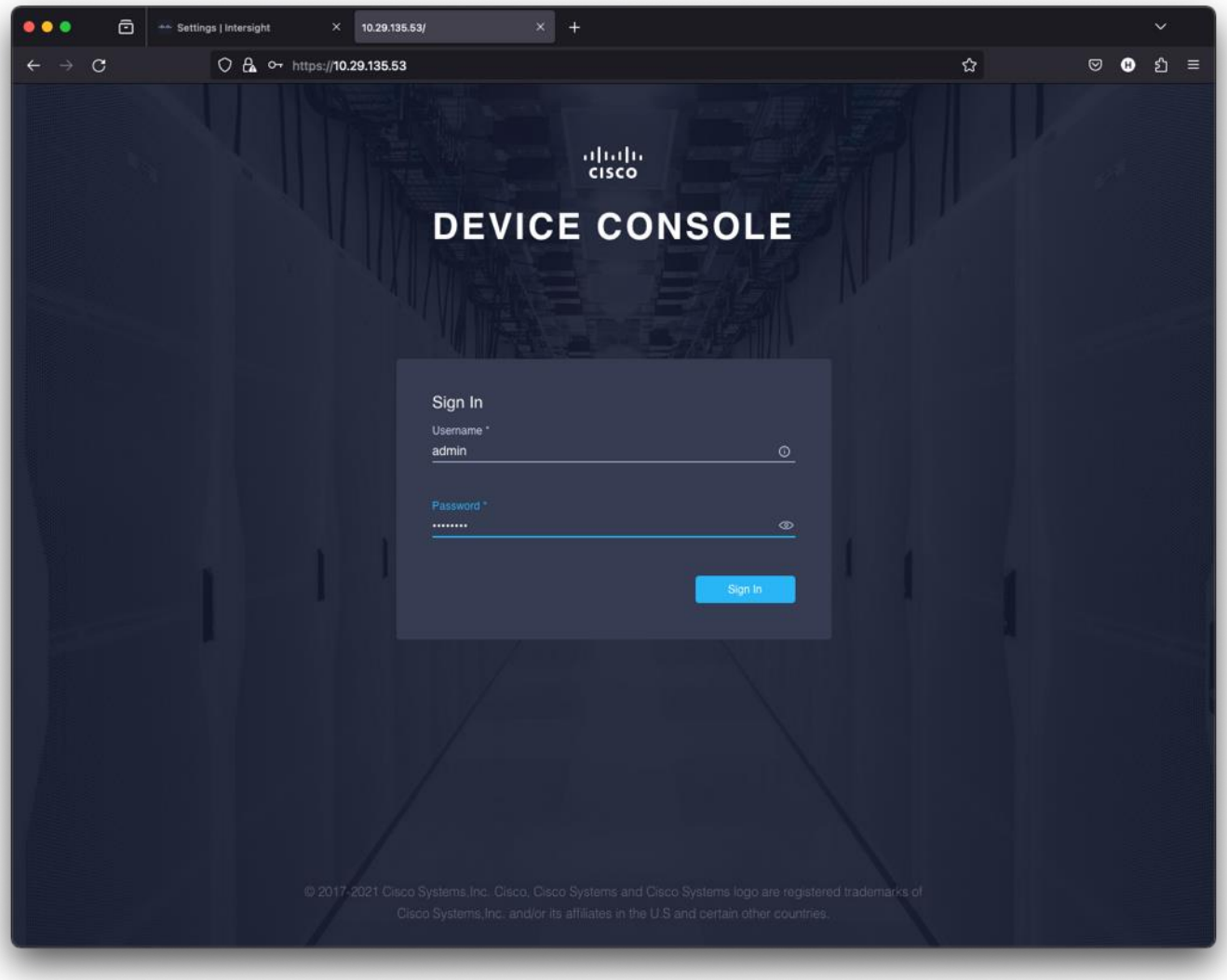
**Step 2.** Connect to the console port on the first fabric interconnect and configure the first FI as shown below:

```
  Enter the configuration method. (console/gui) ?
  Enter the configuration method. (console/gui) ?
  Enter the configuration method. (console/gui) ? console

  Enter the management mode. (ucsm/intersight)? intersight

  The Fabric interconnect will be configured in the intersight managed mode. Choose (y/n) to proceed: y

  Enforce strong password? (y/n) [y]: n

  Enter the password for "admin":
  Confirm the password for "admin":

  Enter the switch fabric (A/B) []: A

  Enter the system name:  FS-ORA-FI

  Physical Switch Mgmt0 IP address : 10.29.135.53

  Physical Switch Mgmt0 IPv4 netmask : 255.255.255.0

  IPv4 address of the default gateway : 10.29.135.1

    DNS IP address : 171.70.168.183

  Configure the default domain name? (yes/no) [n]:

  Following configurations will be applied:

    Management Mode=intersight
    Switch Fabric=A
    System Name=FS-ORA-FI
    Enforced Strong Password=no
    Physical Switch Mgmt0 IP Address=10.29.135.53
    Physical Switch Mgmt0 IP Netmask=255.255.255.0
    Default Gateway=10.29.135.1
    DNS Server=171.70.168.183

  Apply and save the configuration (select 'no' if you want to re-enter)? (yes/no): yes
  Applying configuration. Please wait.

  Configuration file - Ok
XML interface to system may become unavailable since ssh is disabled

 Completing basic configuration setup
2024 Jan  9 20:26:36 FS-ORA-FI-A %$ VDC-1 %$ %SECURITYD-2-FEATURE_ENABLE_DISABLE: User  has enabled the feature bash-shell
2024 Jan  9 20:26:41 FS-ORA-FI-A %$ VDC-1 %$ %SECURITYD-2-FEATURE_NXAPI_ENABLE: Feature nxapi is being enabled on HTTPS.


Cisco UCS 6400 Series Fabric Interconnect
FS-ORA-FI-A login: 2024 Jan  9 20:27:14 FS-ORA-FI-A %$ VDC-1 %$ %USER-2-SYSTEM_MSG: Starting Intersight managed UCS Processes.. - pmon

Cisco UCS 6400 Series Fabric Interconnect
FS-ORA-FI-A login:
```

**Step 3.** Connect the console port on the second fabric interconnect B and configure it as shown below:

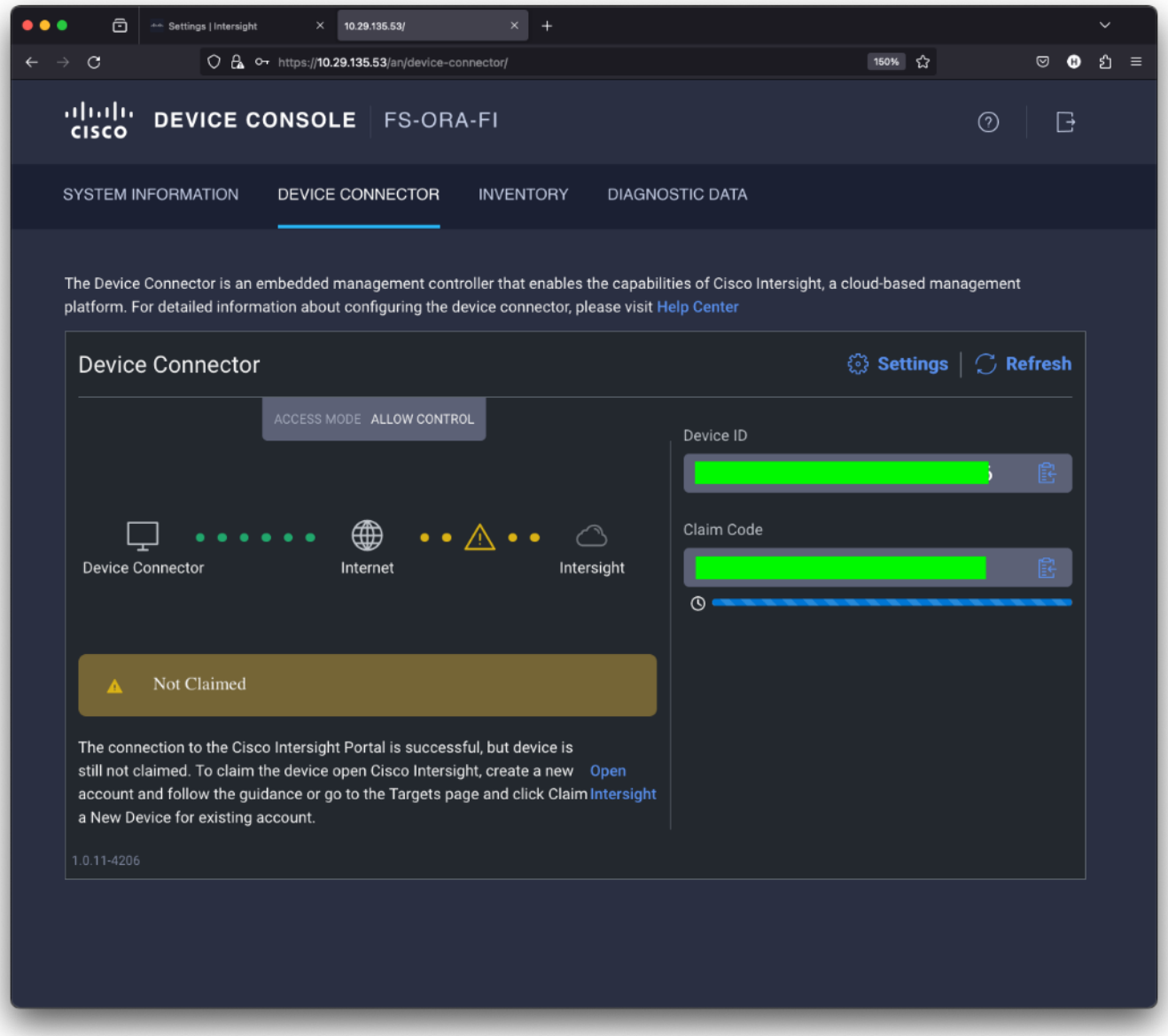
```
  Enter the configuration method. (console/gui) ? console

  Installer has detected the presence of a peer Fabric interconnect. This Fabric interconnect will be added to the cluster. Continue (y/n) ? y

  Enter the admin password of the peer Fabric interconnect:
    Connecting to peer Fabric interconnect... done
    Retrieving config from peer Fabric interconnect... done
    Peer Fabric interconnect management mode   : intersight
    Peer Fabric interconnect Mgmt0 IPv4 Address: 10.29.135.53
    Peer Fabric interconnect Mgmt0 IPv4 Netmask: 255.255.255.0

    Peer FI is IPv4 Cluster enabled. Please Provide Local Fabric Interconnect Mgmt0 IPv4 Address

  Physical Switch Mgmt0 IP address : 10.29.135.54


  Apply and save the configuration (select 'no' if you want to re-enter)? (yes/no): yes
  Applying configuration. Please wait.

  Configuration file - Ok
XML interface to system may become unavailable since ssh is disabled

 Completing basic configuration setup
2024 Jan  9 20:29:46 FS-ORA-FI-B %$ VDC-1 %$ %SECURITYD-2-FEATURE_ENABLE_DISABLE: User  has enabled the feature bash-shell
2024 Jan  9 20:29:53 FS-ORA-FI-B %$ VDC-1 %$ %SECURITYD-2-FEATURE_NXAPI_ENABLE: Feature nxapi is being enabled on HTTPS.


Cisco UCS 6400 Series Fabric Interconnect
FS-ORA-FI-B login:
```

**Step 4.** After configuring both the FI management address, open a web browser and navigate to the Cisco UCS fabric interconnect management address as configured. If prompted to accept security certificates, accept, as necessary.



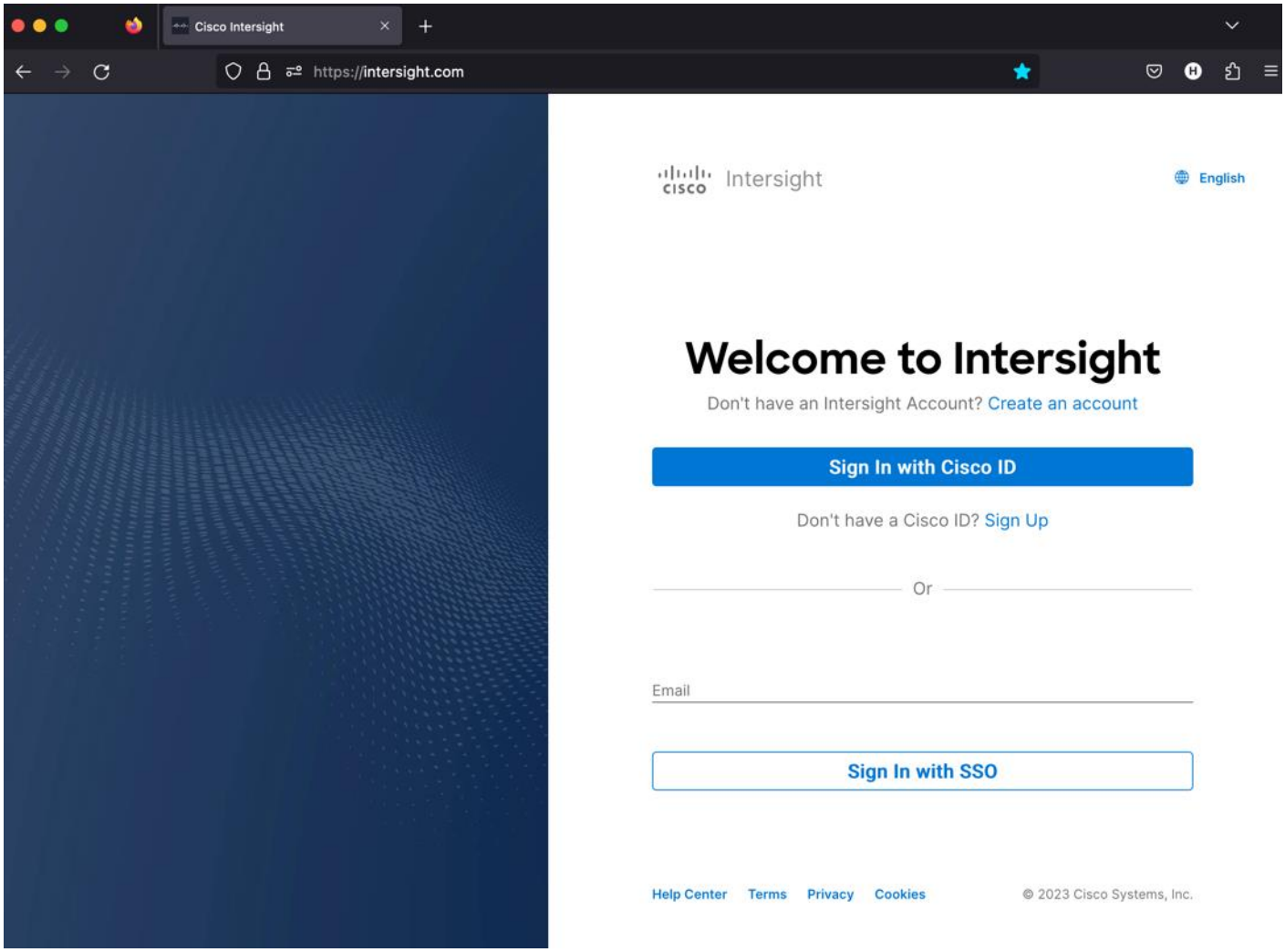

**Step 5.** Log into the device console for FI-A by entering your username and password.

**Step 6.** Go to the Device Connector tab and get the DEVICE ID and CLAIM Code as shown below:



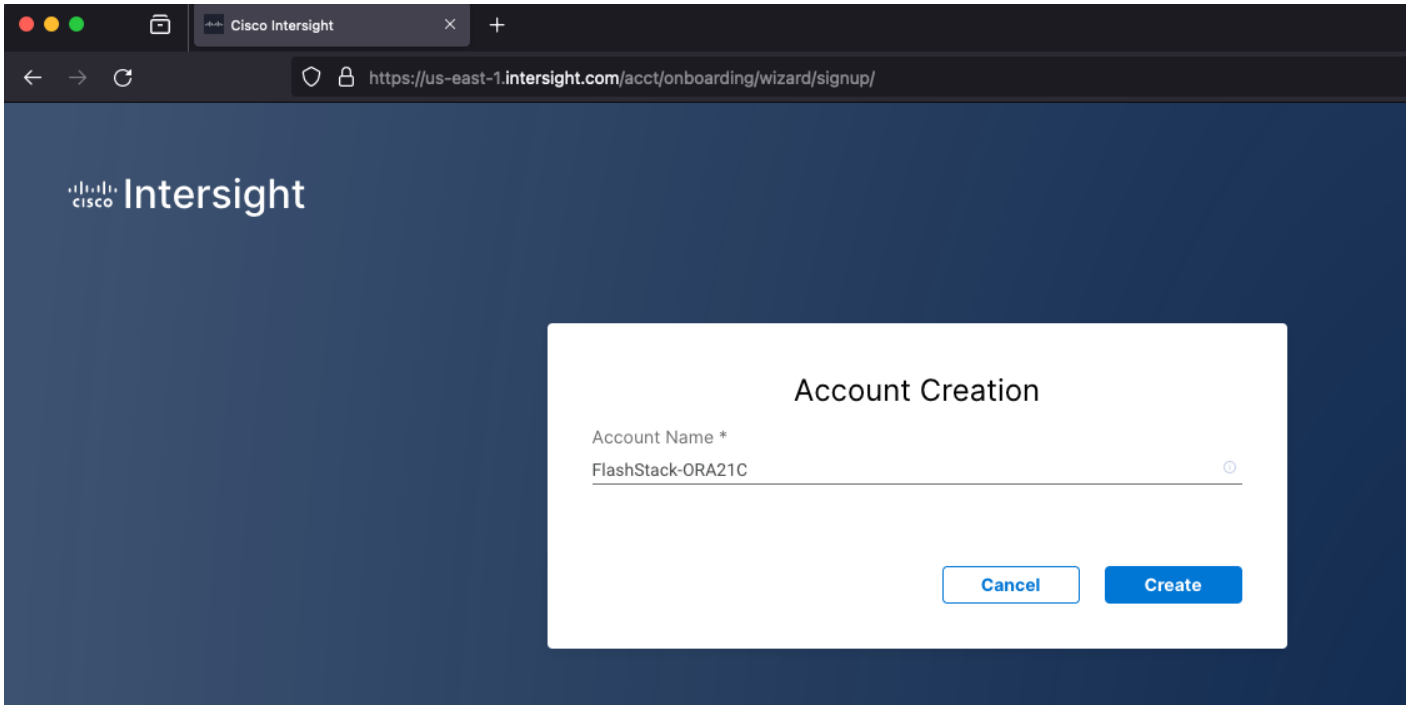

## Procedure 2. Claim Fabric Interconnect in Cisco Intersight Platform

After setting up the Cisco UCS fabric interconnect for Cisco Intersight Managed Mode, FIs can be claimed to a new or an existing Cisco Intersight account. When a Cisco UCS Fabric Interconnect is successfully added to the Cisco Intersight platform, all future configuration steps are completed in the Cisco Intersight portal. After getting the device id and claim code of FI, go to https://intersight.com/.

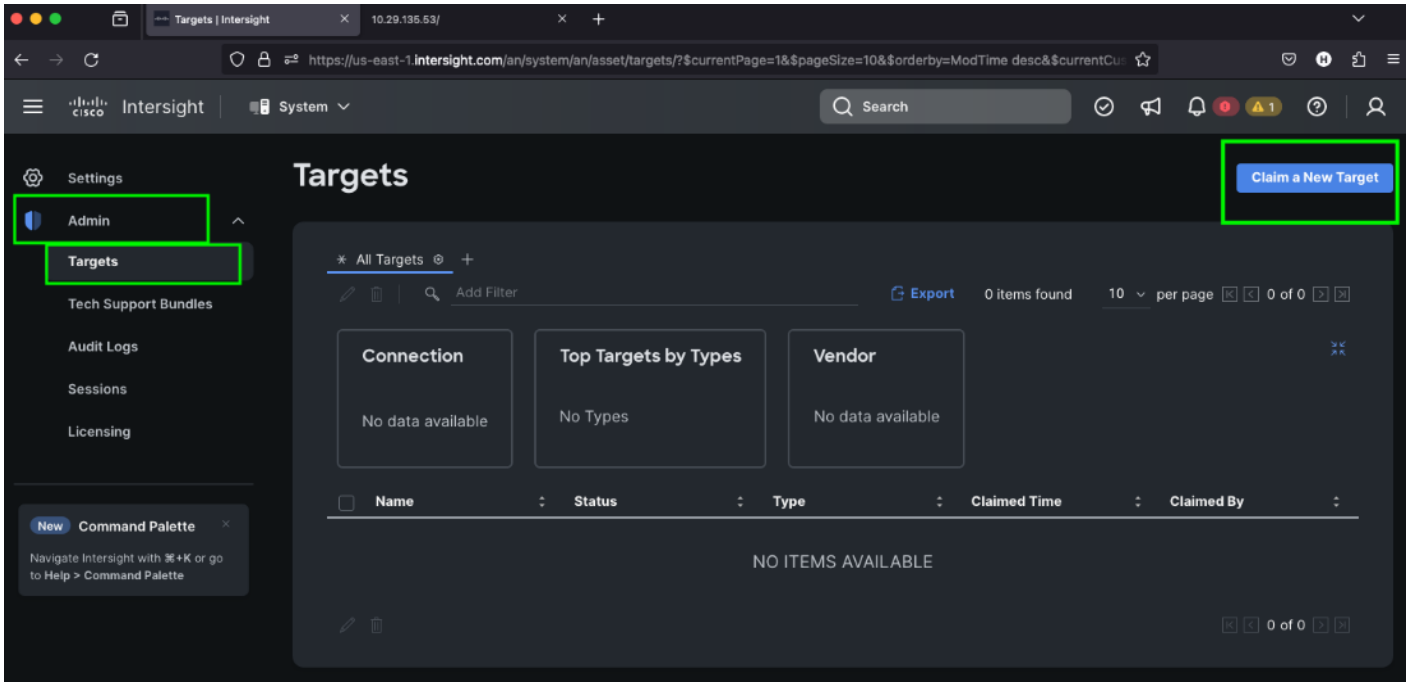



**Step 7.** Sign in with your Cisco ID or if you don’t have one, click Sing Up and setup your account.

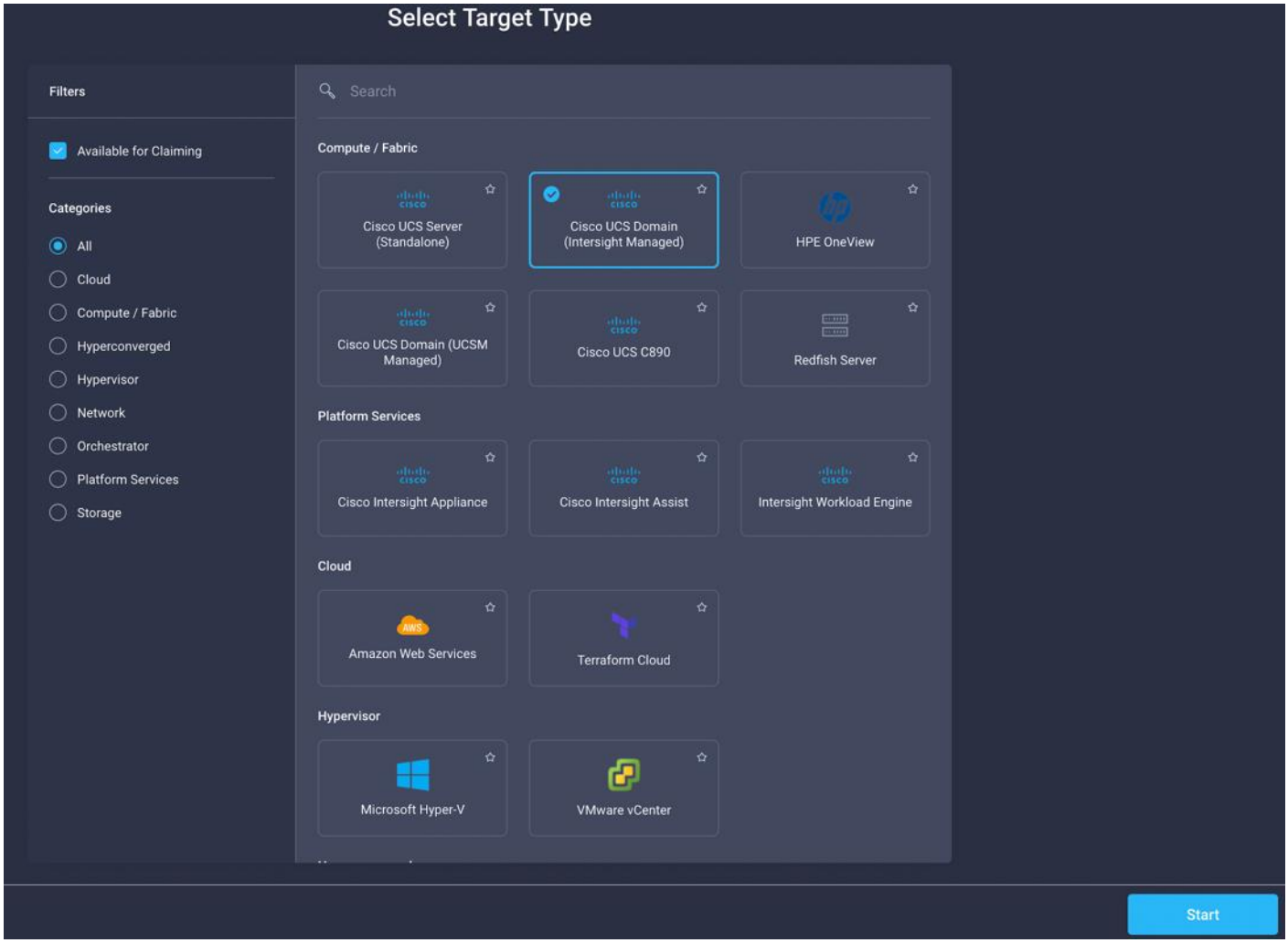
**Note:** We created the “FlashStack-ORA21C” account for this solution.



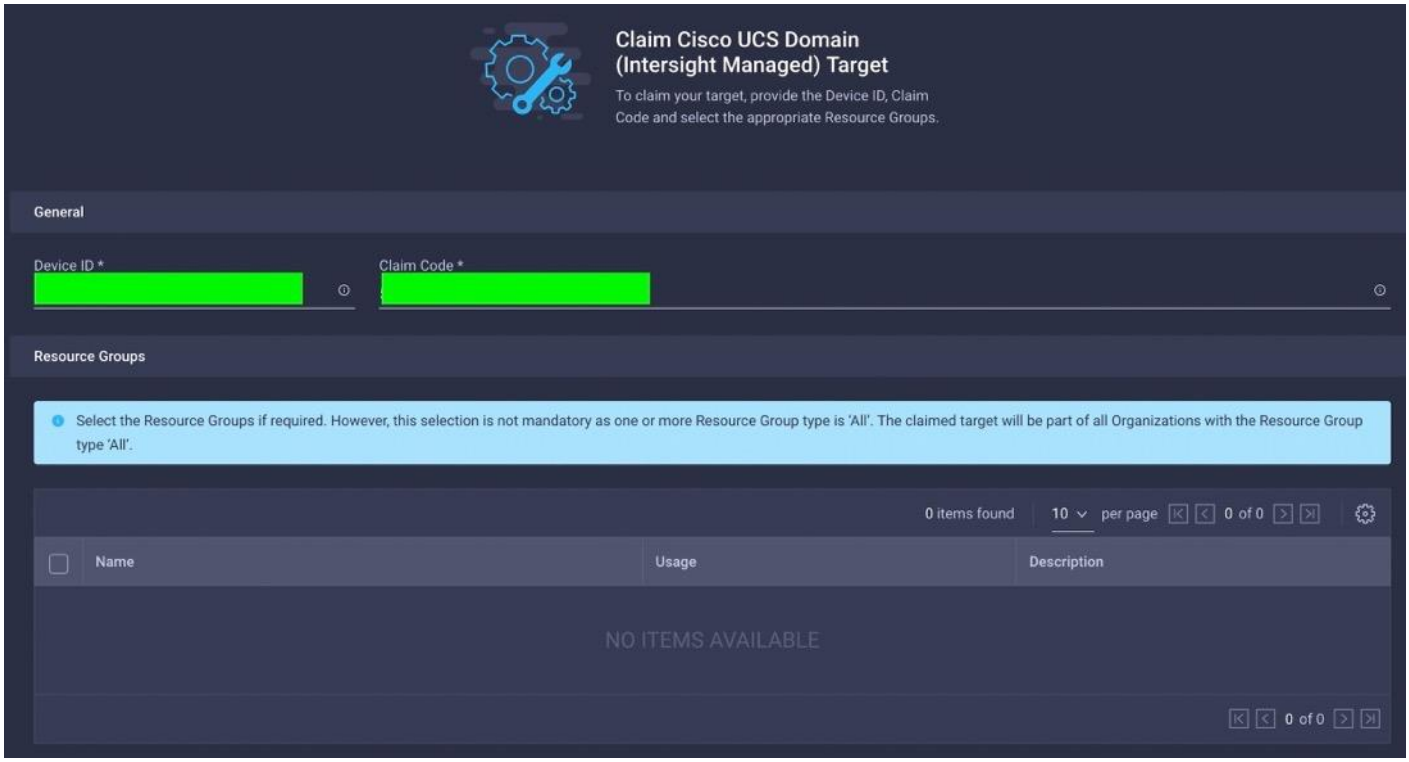

**Step 8.** After logging into your Cisco Intersight account, go to > ADMIN > Targets > Claim a New Target.



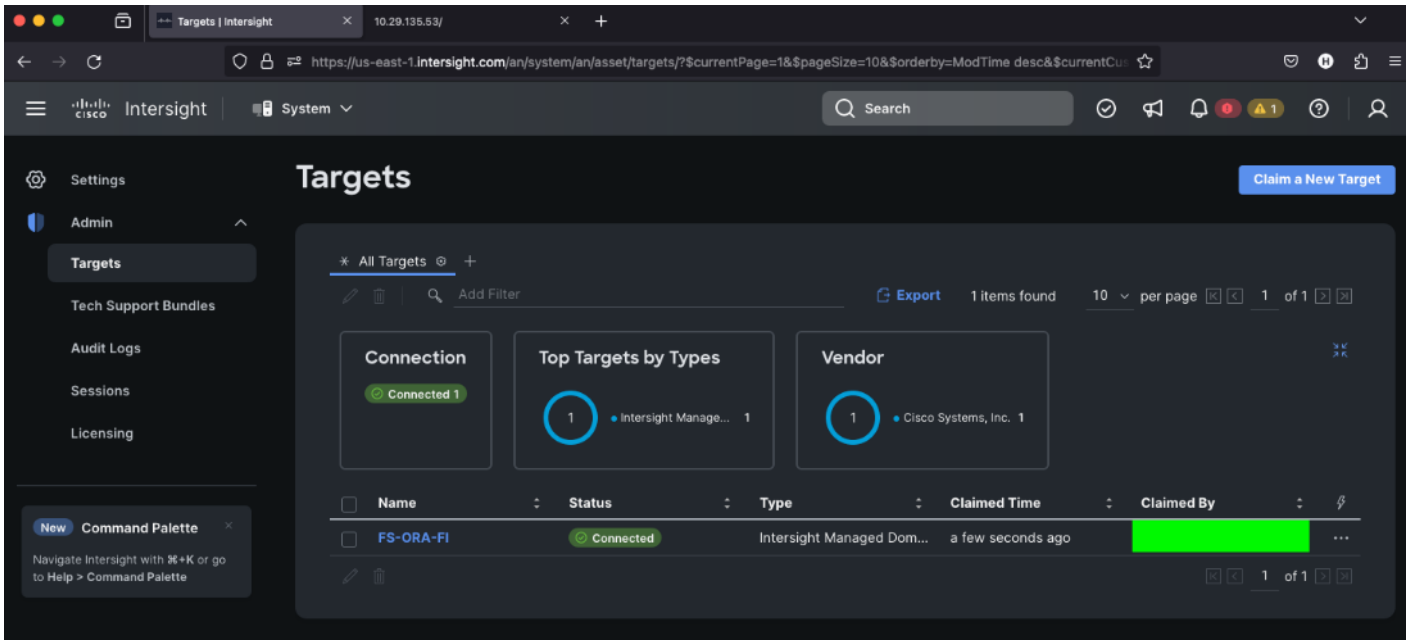

**Step 9.** For the Select Target Type, select “Cisco UCS Domain (Intersight Managed)” and click Start.

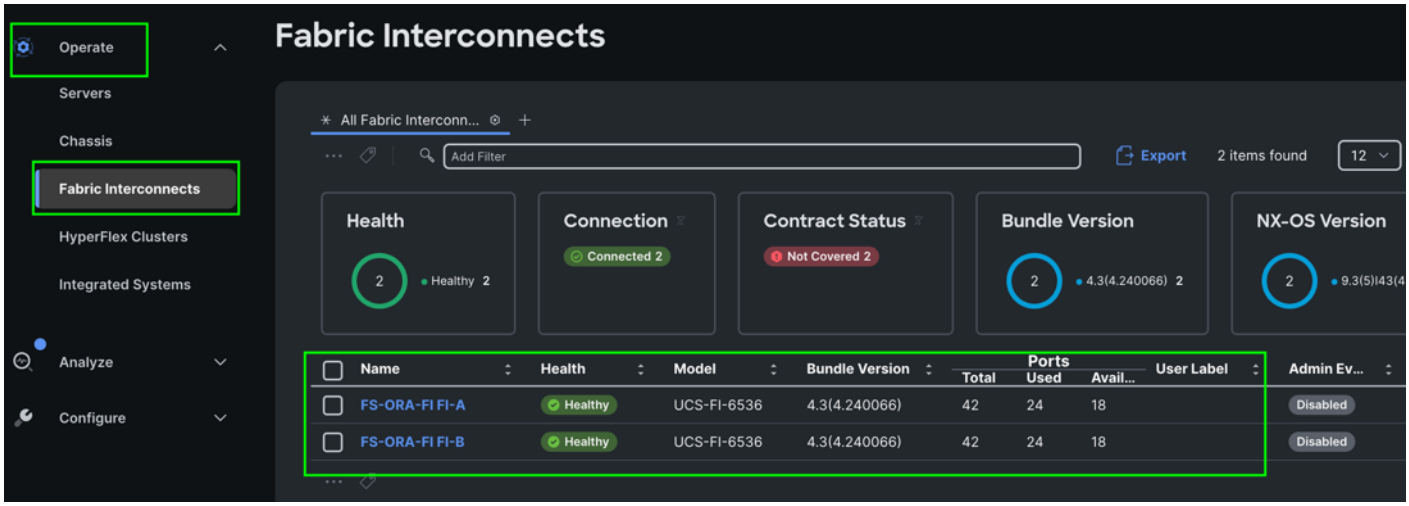



**Step 10.** Enter the Device ID and Claim Code which was previously captured. Click Claim to claim this domain in Cisco Intersight.




When you claim this domain, you can see both FIs under this domain and verify it's under Intersight Managed Mode:






## Procedure 3. Configure Policies for Cisco UCS Chassis

**Note:** For this solution, we configured Organization as “ORA21.” You will configure all the profile, pools, and policies under this common organization to better consolidate resources.

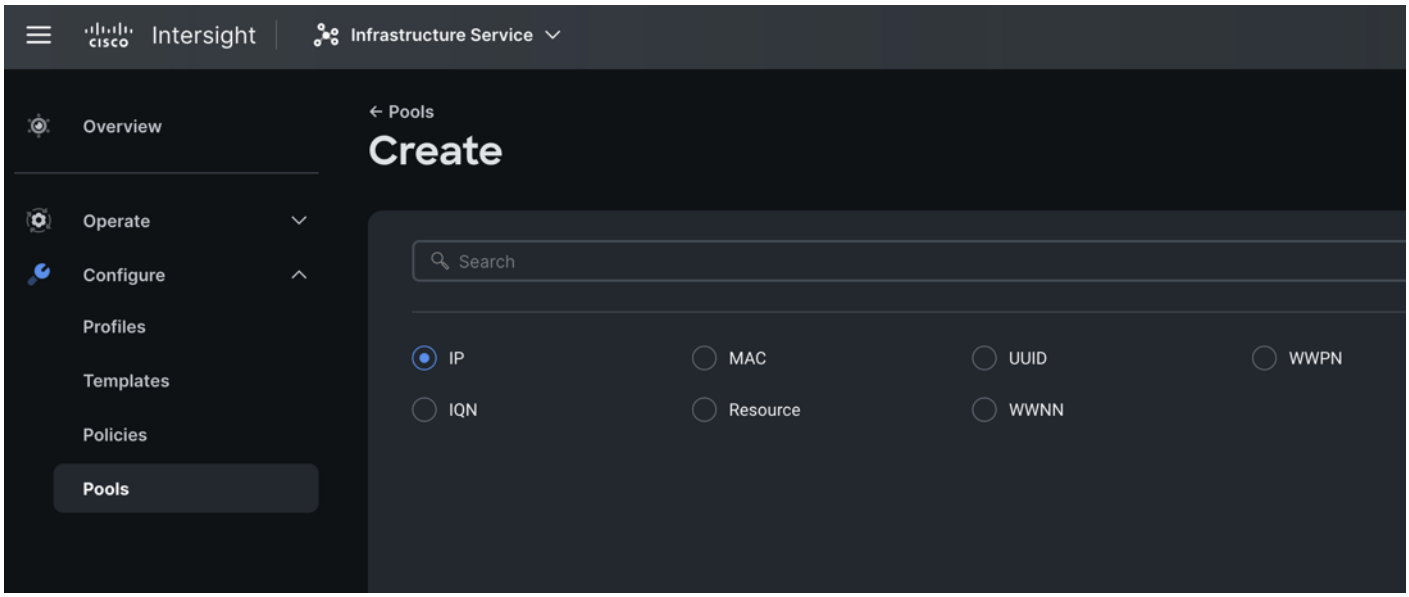
**Step 1.** To create Organization, go to Cisco Intersight > Settings > Organization and create depending upon your environment.

**Note:** We configured the IP Pool, IMC Access Policy, and Power Policy for the Cisco UCS Chassis profile as explained below.

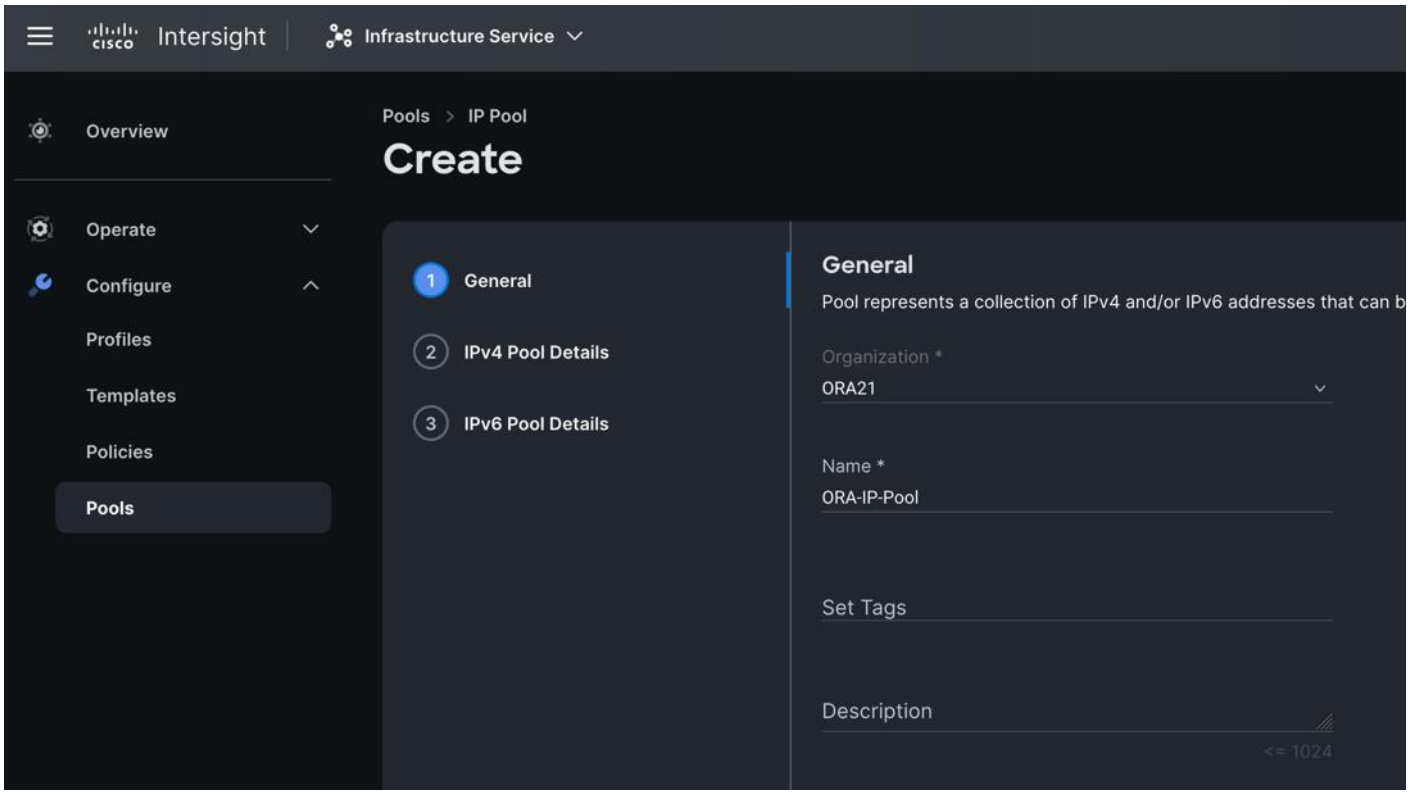
## Procedure 4. Create IP Pool

**Step 1.** To configure the IP Pool for the Cisco UCS Chassis profile, go to > Infrastructure Service > Configure > Pools > and then select “Create Pool” on the top right corner.

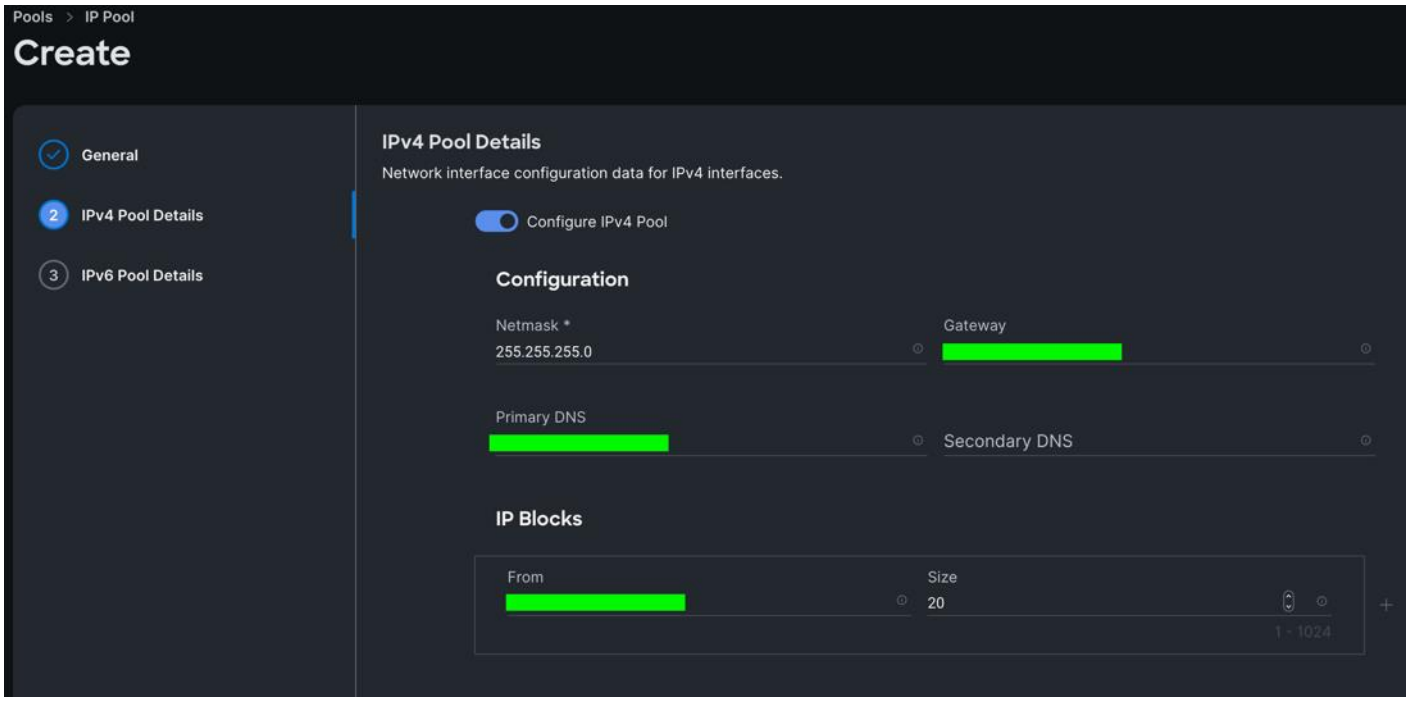
**Step 2.** Select option “IP” as shown below to create the IP Pool.



**Step 3.** In the IP Pool Create section, for Organization select “ORA21” and enter the Policy name “ORA-IP-Pool” and click Next.




**Step 4.** Enter Netmask, Gateway, Primary DNS, IP Blocks and Size according to your environment and click Next.

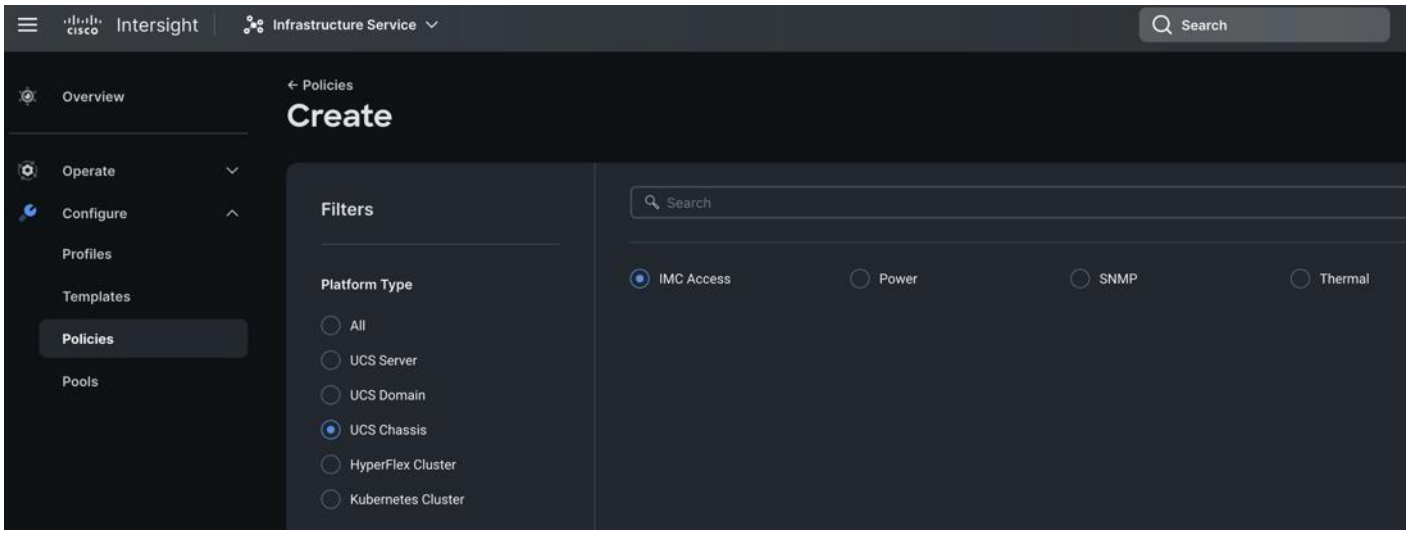


**Note:** For this solution, we did not configure the IPv6 Pool. Keep the Configure IPv6 Pool option disabled and click Create to create the IP Pool.

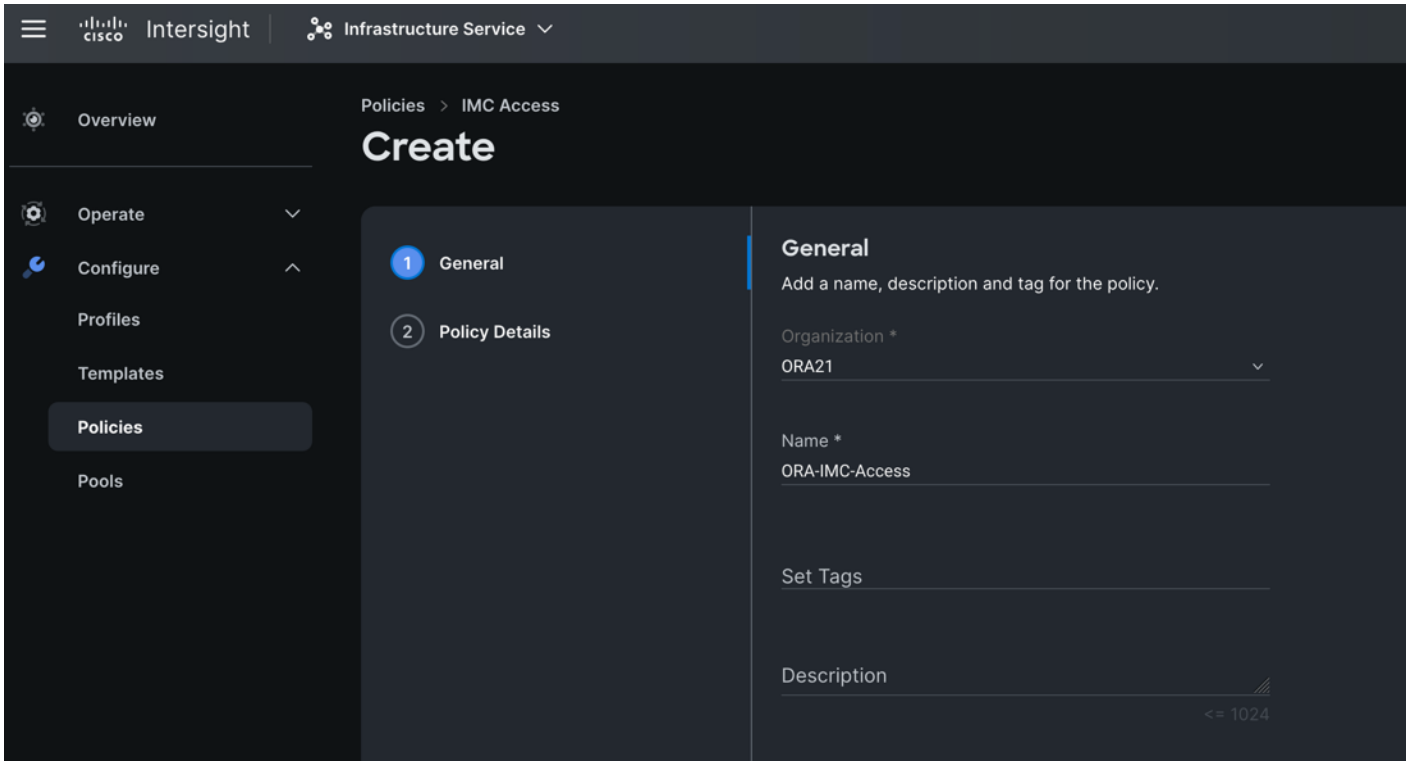
## Procedure 5. Configure IMC Access Policy

**Step 1.** To configure the IMC Access Policy for the Cisco UCS Chassis profile, go to > Infrastructure Service > Configure > Polices > and click Create Policy.

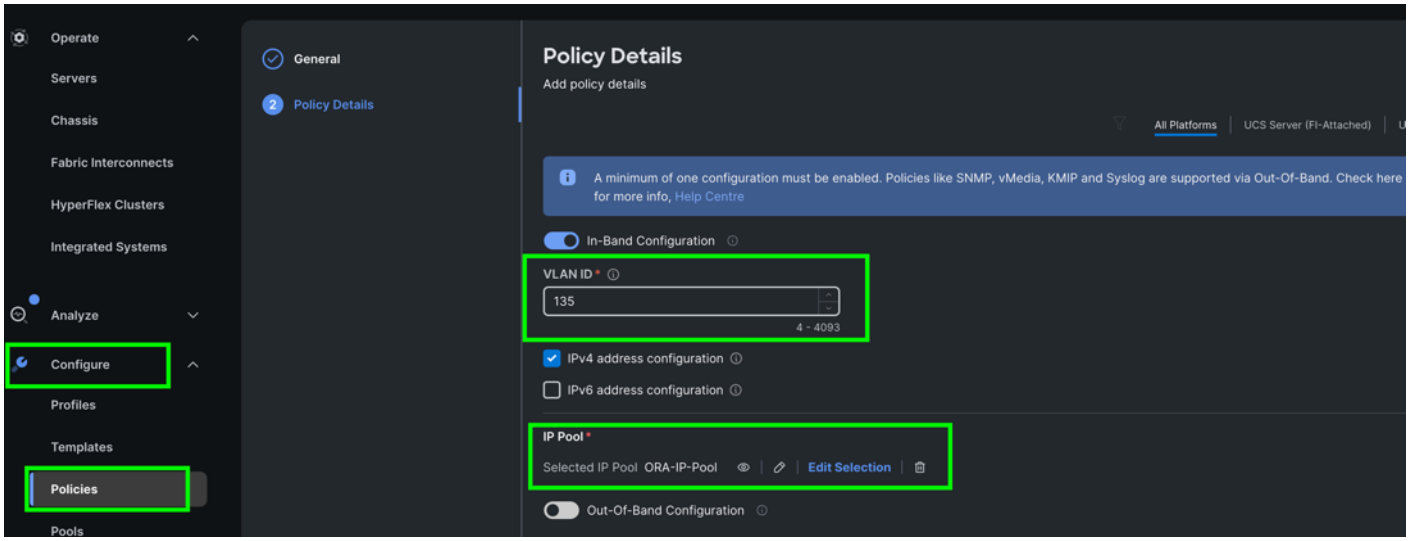
**Step 2.** Select the platform type “UCS Chassis” and select “IMC Access” policy.




**Step 3.** In the IMC Access Create section, for Organization select “ORA21” and enter the Policy name “ORA-IMC-Access” and click Next.




**Step 4.** In the Policy Details section, enter the VLAN ID as 135 and select the IP Pool “ORA-IP-Pool.”

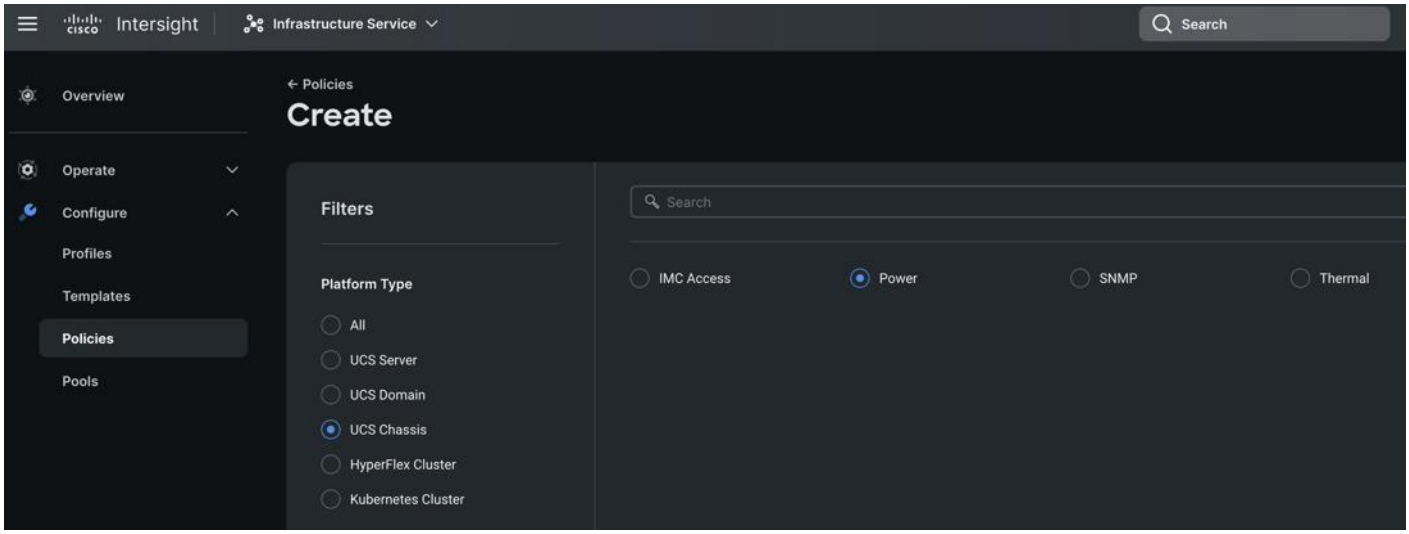



**Step 5.** Click Create to create this policy.

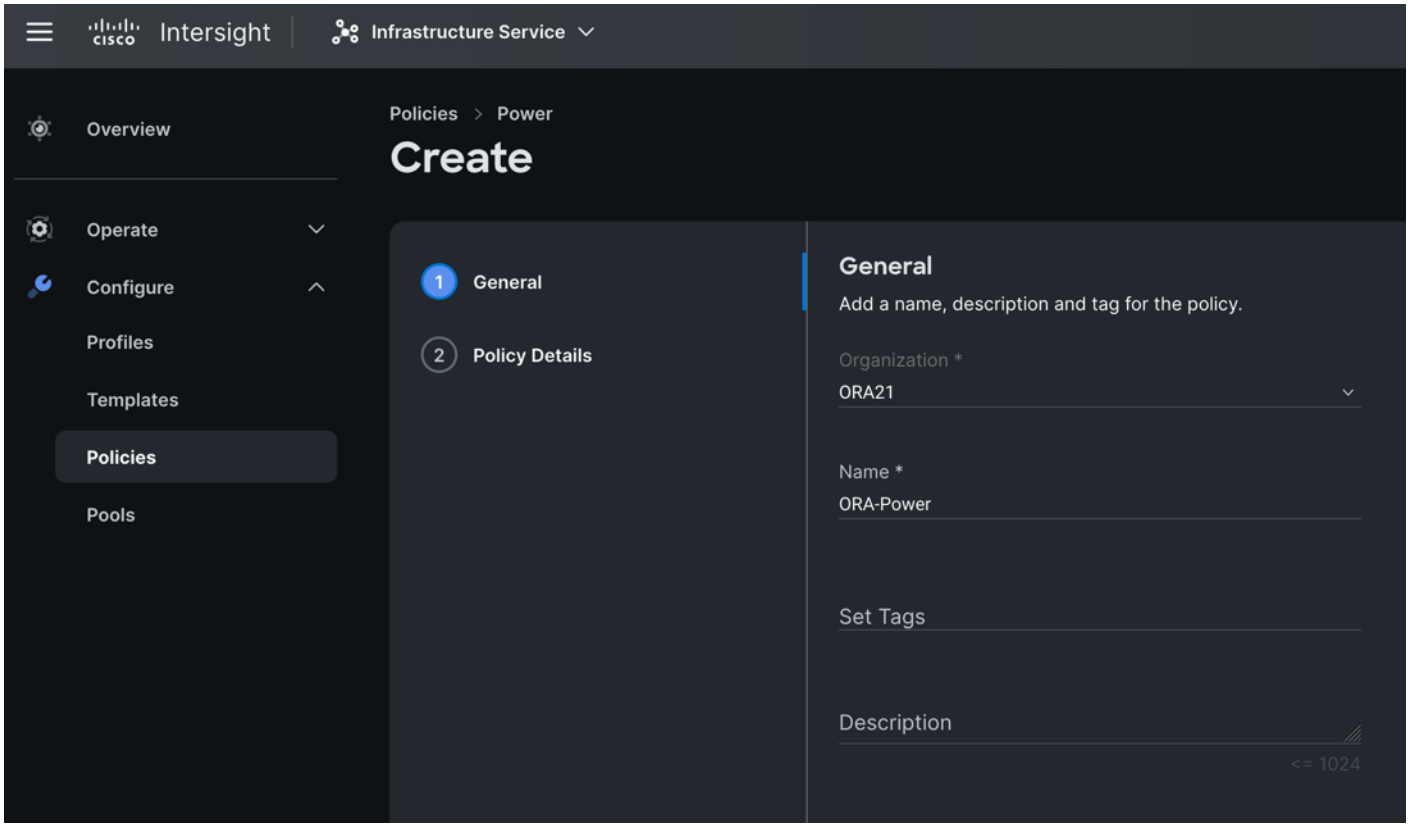
## Procedure 6. Configure Power Policy

**Step 1.** To configure the Power Policy for the Cisco UCS Chassis profile, go to > Infrastructure Service > Configure > Polices > and click Create Policy.

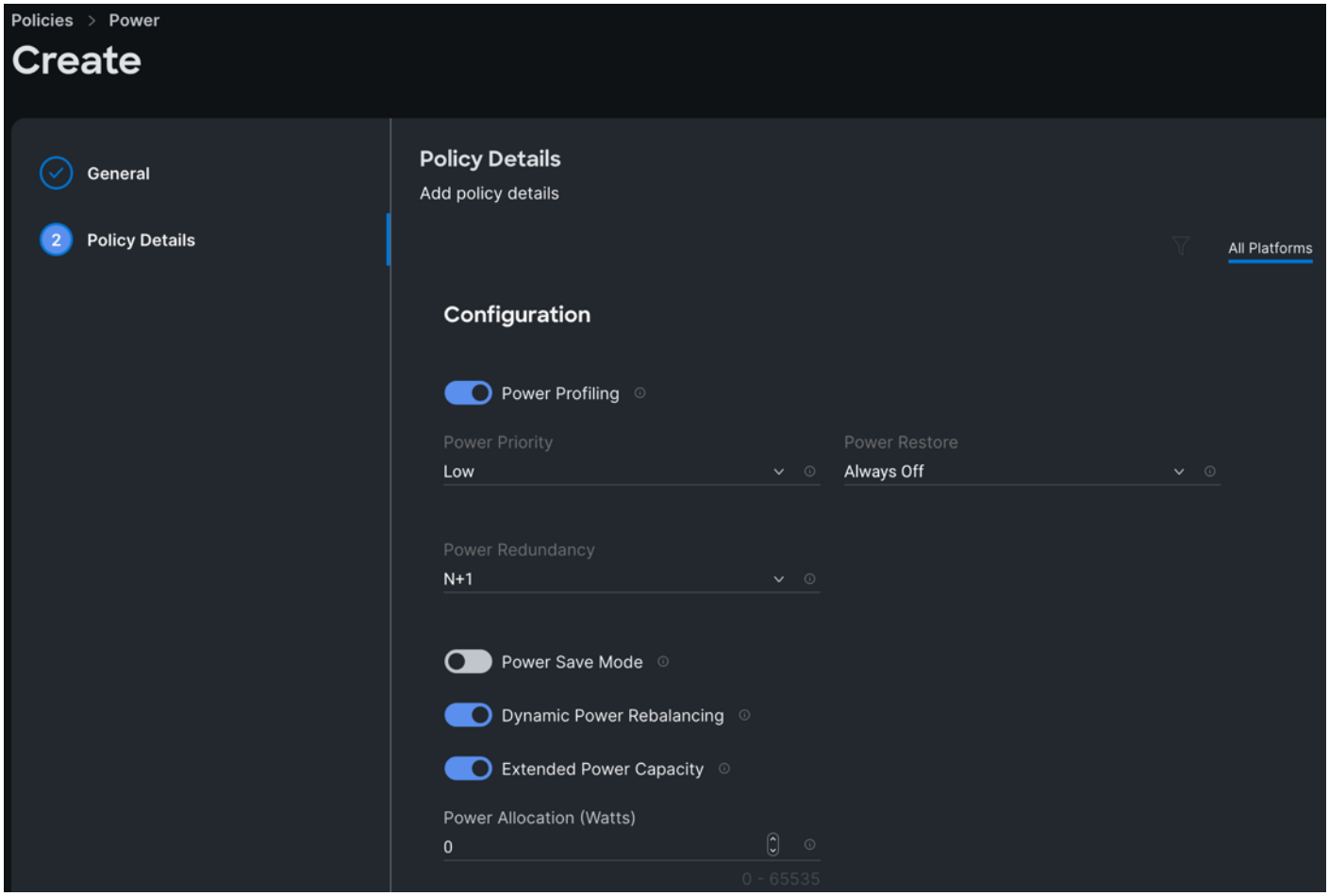
**Step 2.** Select the platform type “UCS Chassis” and select “Power.”




**Step 3.** In the Power Policy Create section, for Organization select “ORA21” and enter the Policy name “ORA-Power” and click Next.




**Step 4.** In the Policy Details section, for Power Redundancy select N+1 and turn off Power Save Mode.



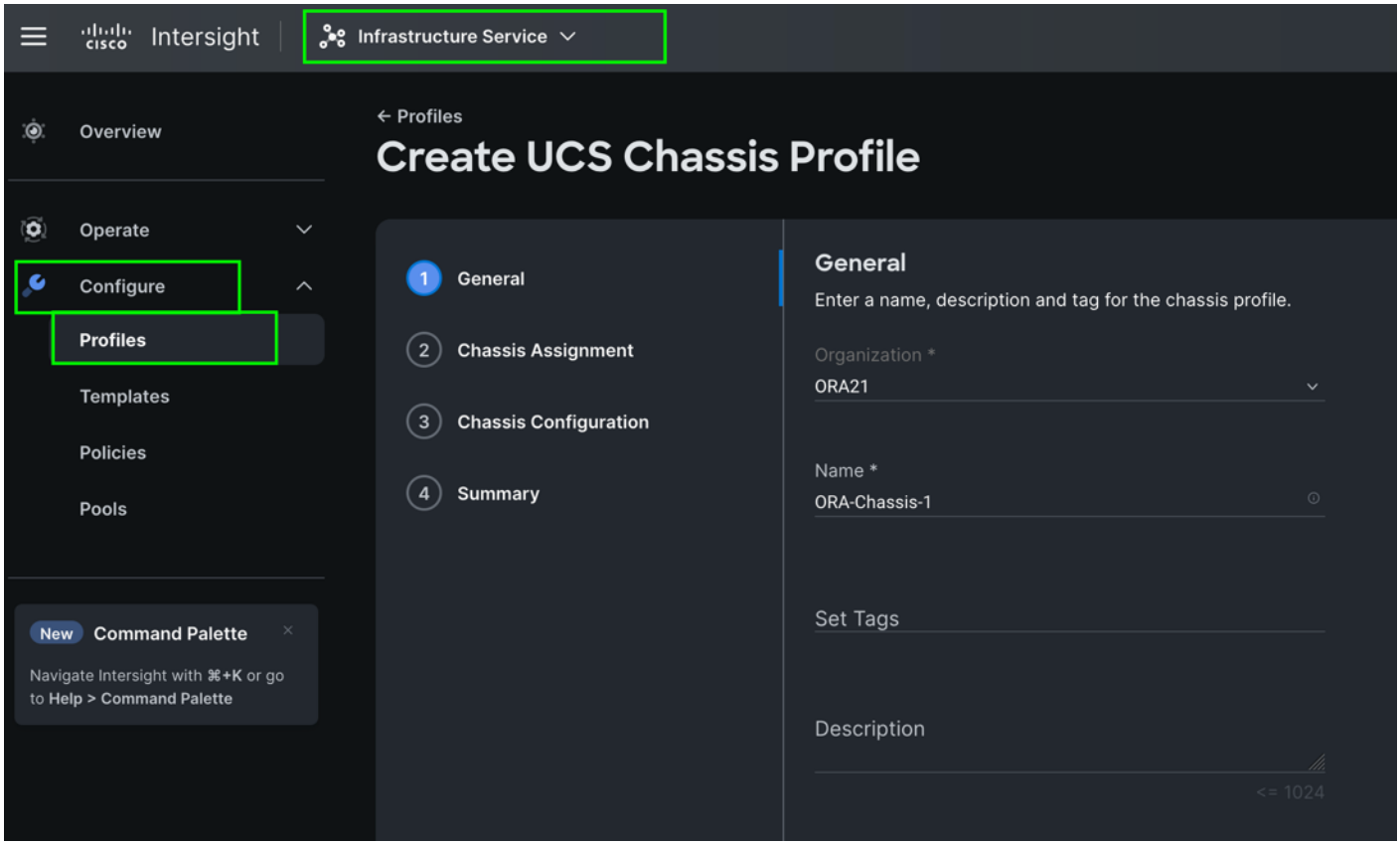

**Step 5.** Click Create to create this policy.

## Procedure 7. Create Cisco UCS Chassis Profile

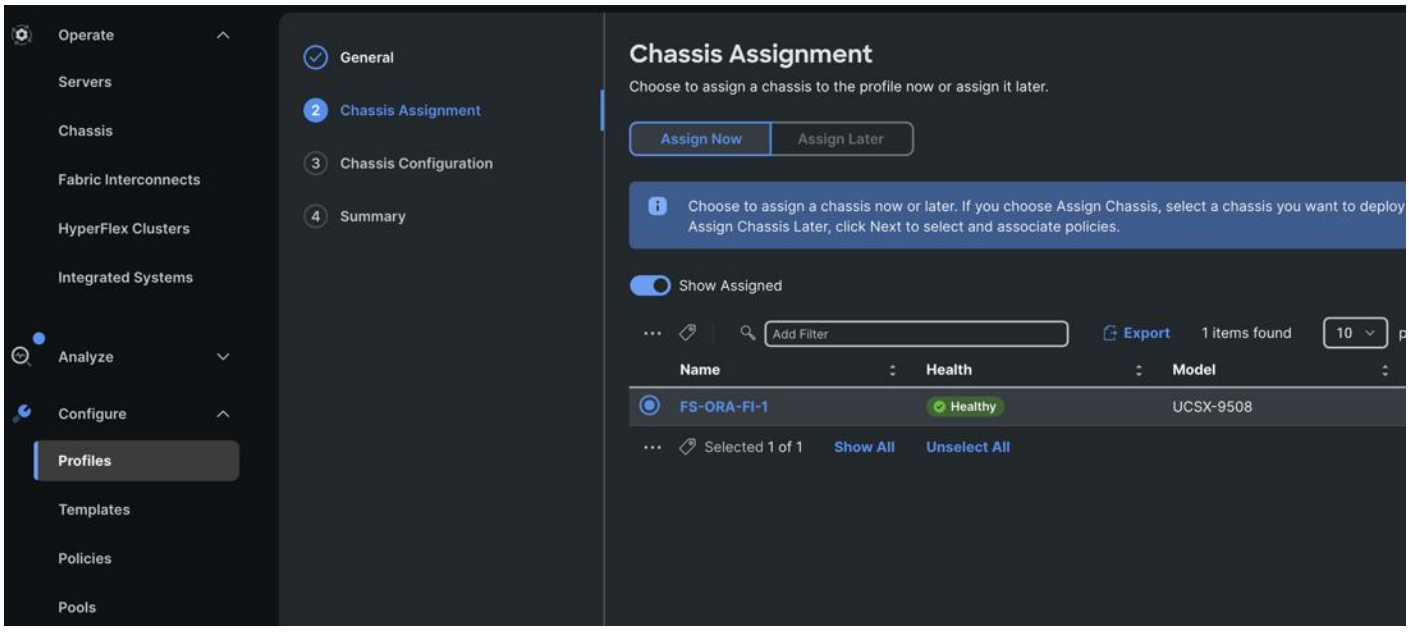
A Cisco UCS Chassis profile enables you to create and associate chassis policies to an Intersight Managed Mode (IMM) claimed chassis. When a chassis profile is associated with a chassis, Cisco Intersight automatically configures the chassis to match the configurations specified in the policies of the chassis profile. The chassis-related policies can be attached to the profile either at the time of creation or later. For more information, go to: https://intersight.com/help/saas/features/chassis/configure#chassis_profiles.

The chassis profile in a FlashStack is used to set the power policy for the chassis. By default, Cisco UCSX power supplies are configured in GRID mode, but the power policy can be utilized to set the power supplies in non-redundant or N+1/N+2 redundant modes

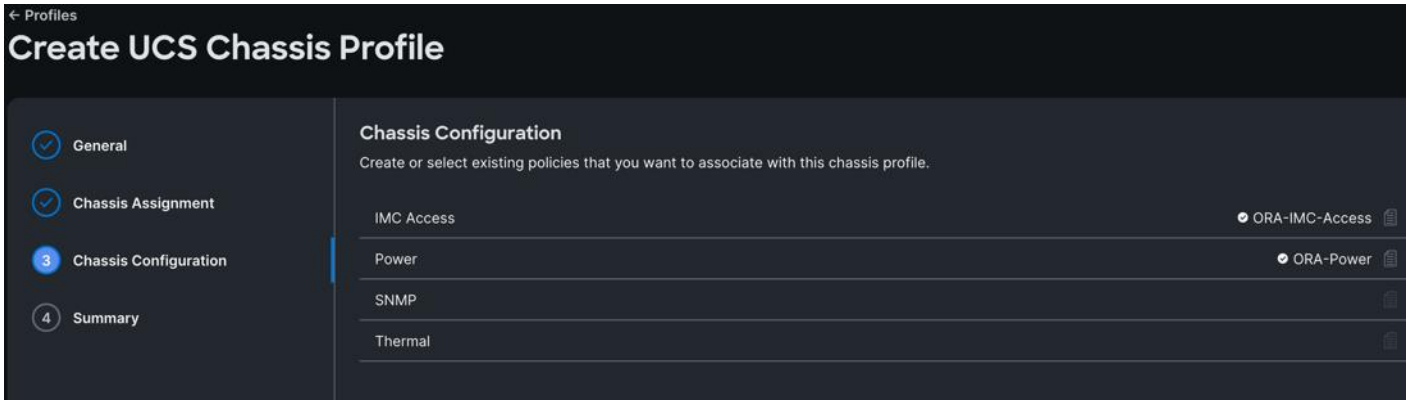
**Step 1.** To create a Cisco UCS Chassis Profile, go to Infrastructure Service > Configure > Profiles > UCS Chassis Domain Profiles tab > and click Create UCS Chassis Profile.




**Step 2.** In the Chassis Assignment menu, for the first chassis, click “FS-ORA-FI-1” and click Next.

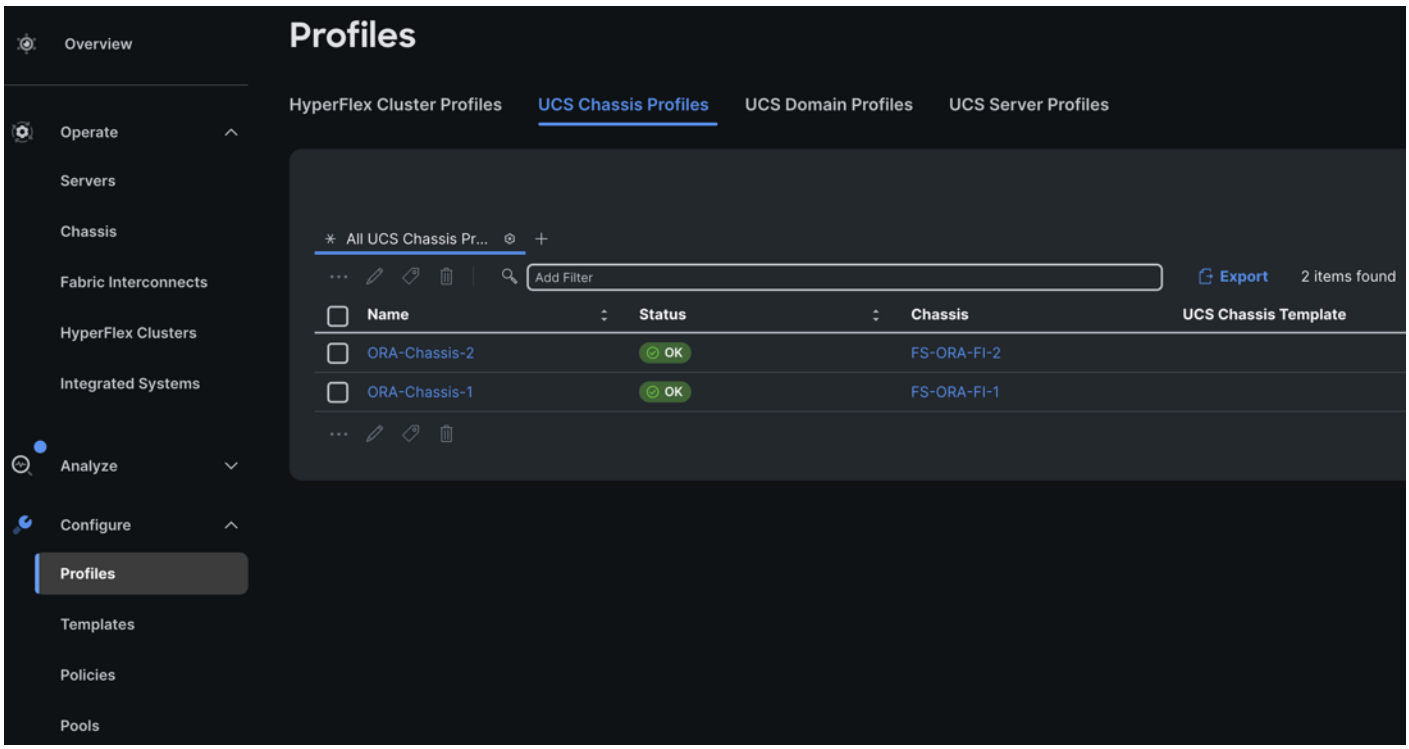



**Step 3.** In the Chassis configuration section, for the policy for IMC Access select “ORA-IMC-Access” and for the Power policy select “ORA-Power.”




**Step 4.** Review the configuration settings summary for the Chassis Profile and click Deploy to create the Cisco UCS Chassis Profile for the first chassis.

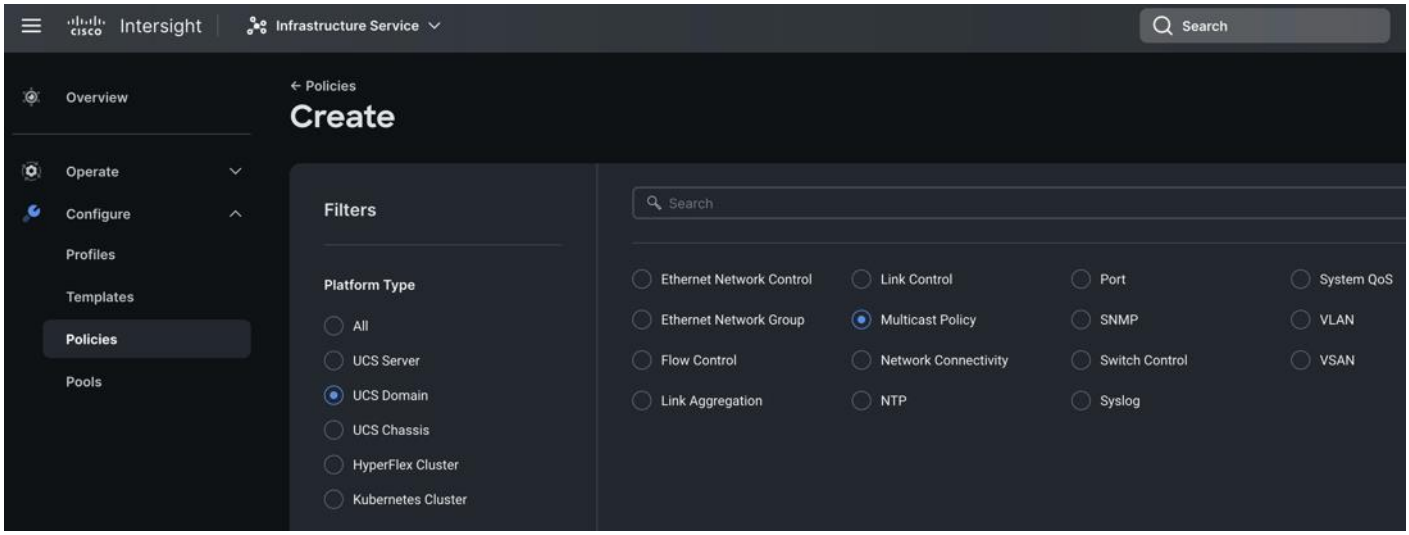
**Note:** For this solution, we created two Chassis Profile (ORA-Chassis-1 and ORA-Chassis-2) and assigned to both the chassis as shown below:




## Configure Policies for Cisco UCS Domain

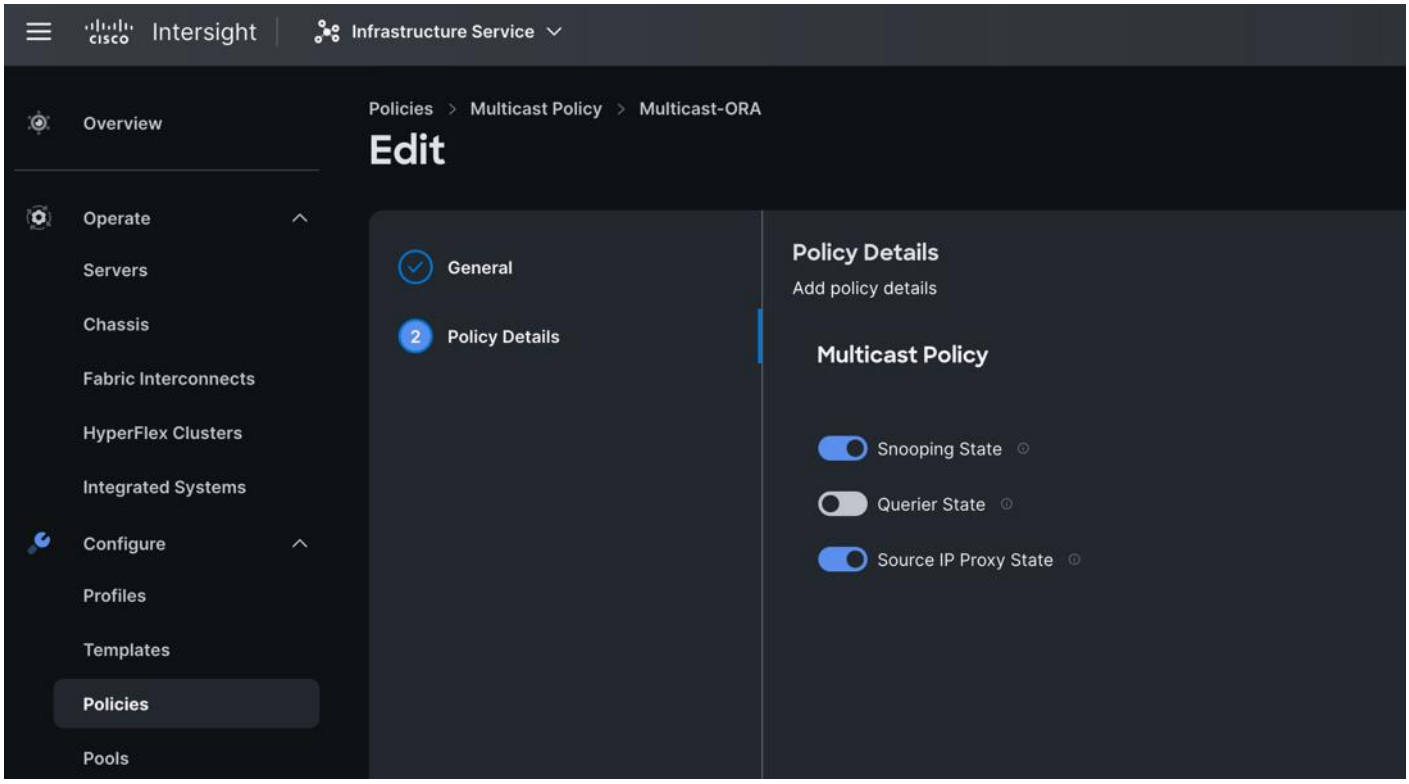
### Procedure 1. Configure Multicast Policy

**Step 1.** To configure Multicast Policy for a Cisco UCS Domain profile, go to > Infrastructure Service > Configure > Polices > and click Create Policy. For the platform type select “UCS Domain” and for Policy, select “Multicast Policy.”




**Step 2.** In the Multicast Policy Create section, for the Organization select “ORA21” and for the Policy name “Multicast-ORA.” Click Next.

**Step 3.** In the Policy Details section, select Snooping State and Source IP Proxy State.

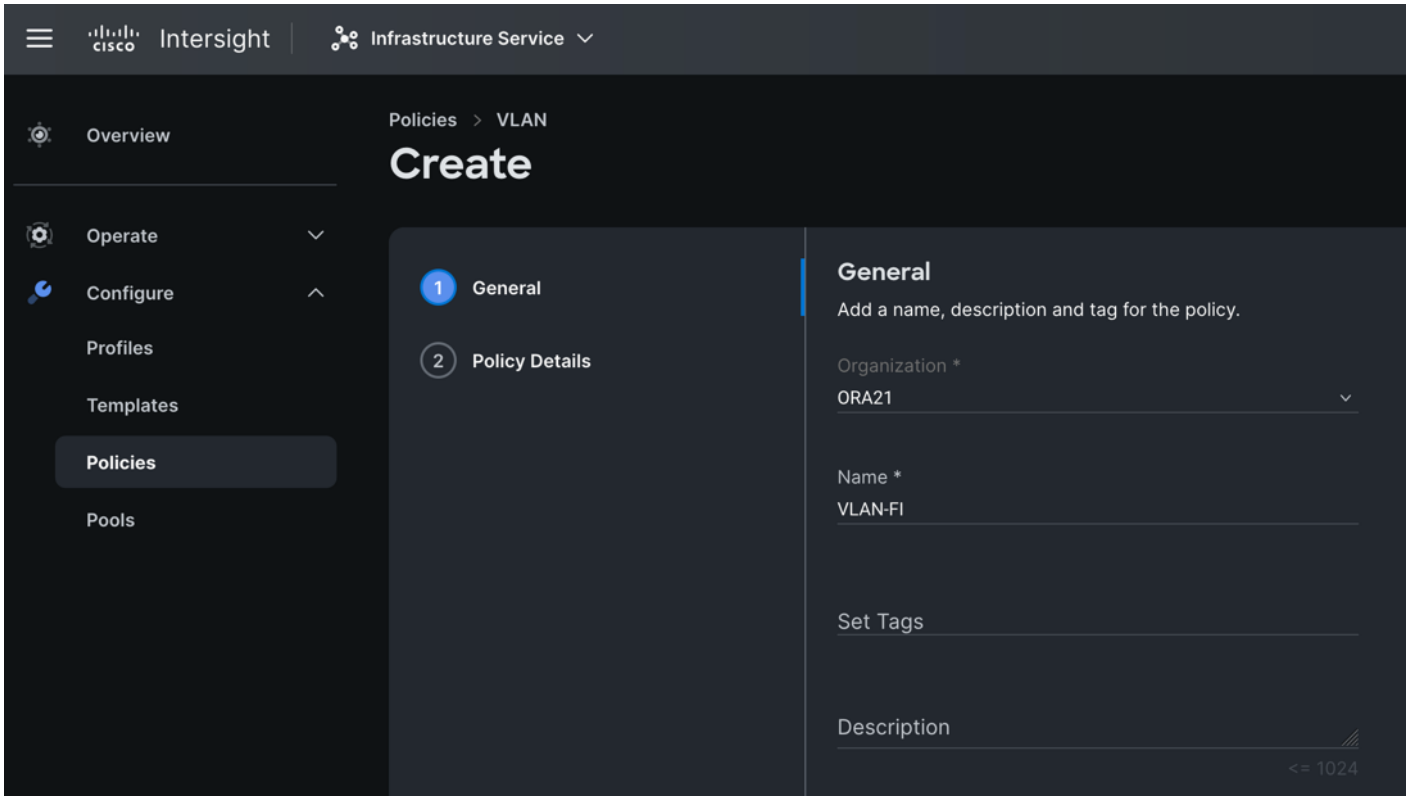



**Step 4.** Click Create to create this policy.

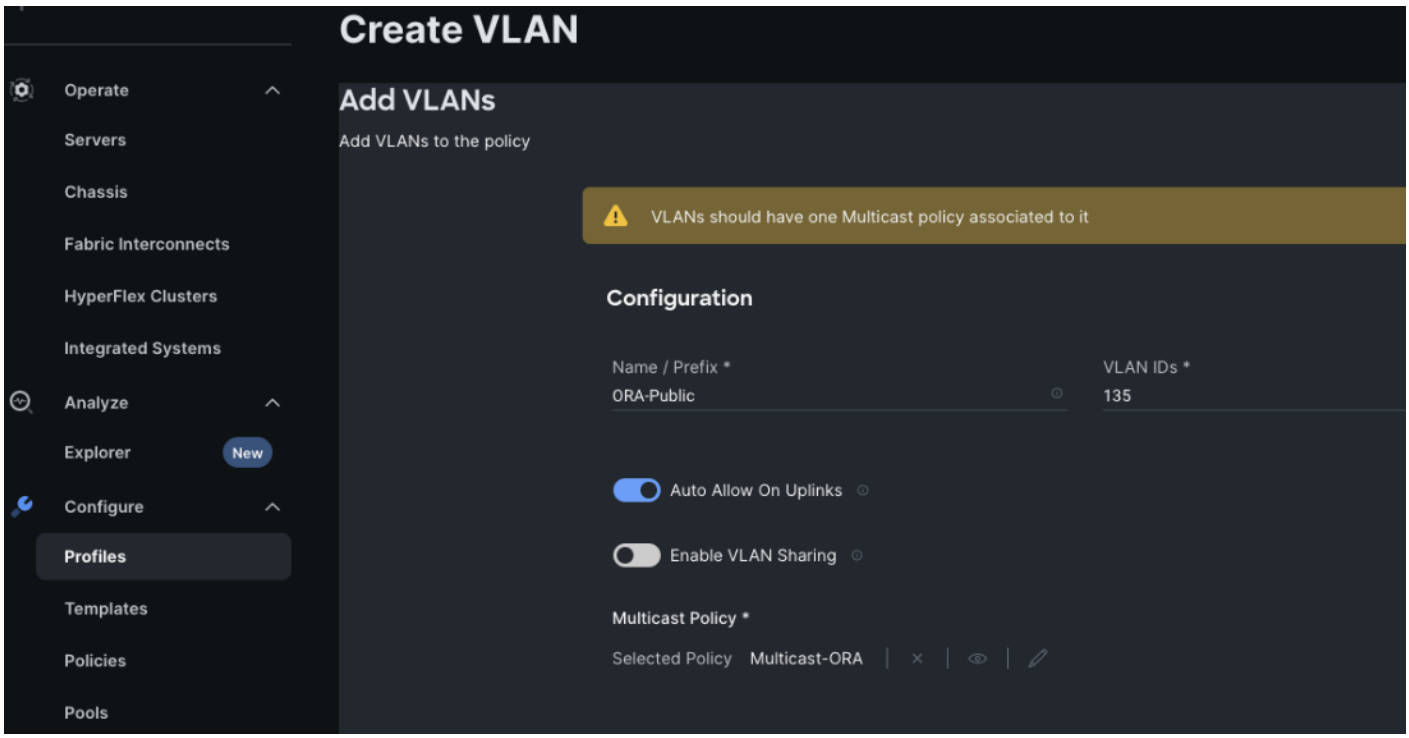
## Procedure 2. Configure VLANs

**Step 1.** To configure the VLAN Policy for the Cisco UCS Domain profile, go to > Infrastructure Service > Configure > Polices > and click Create Policy. For the platform type select “UCS Domain” and for the Policy select “VLAN.”

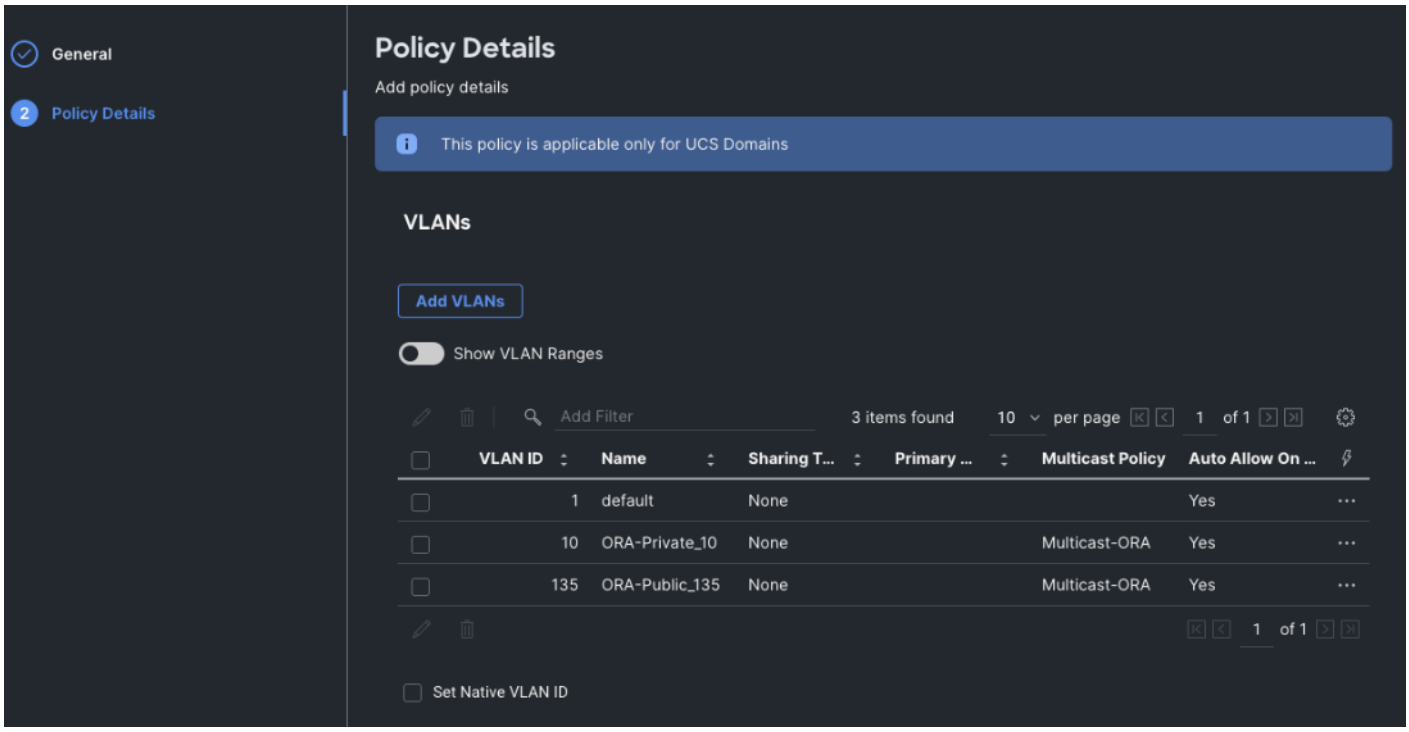
**Step 2.** In the VLAN Policy Create section, for the Organization select “ORA21” and for the Policy name select “VLAN-FI.” Click Next.




**Step 3.** In the Policy Details section, to configure the individual VLANs, select "Add VLANs." Provide a name, VLAN ID for the VLAN and select the Multicast Policy as shown below:




**Step 4.** Click Add to add this VLAN to the policy. Add another VLAN 10 and provide the names to various network traffic of this solution.

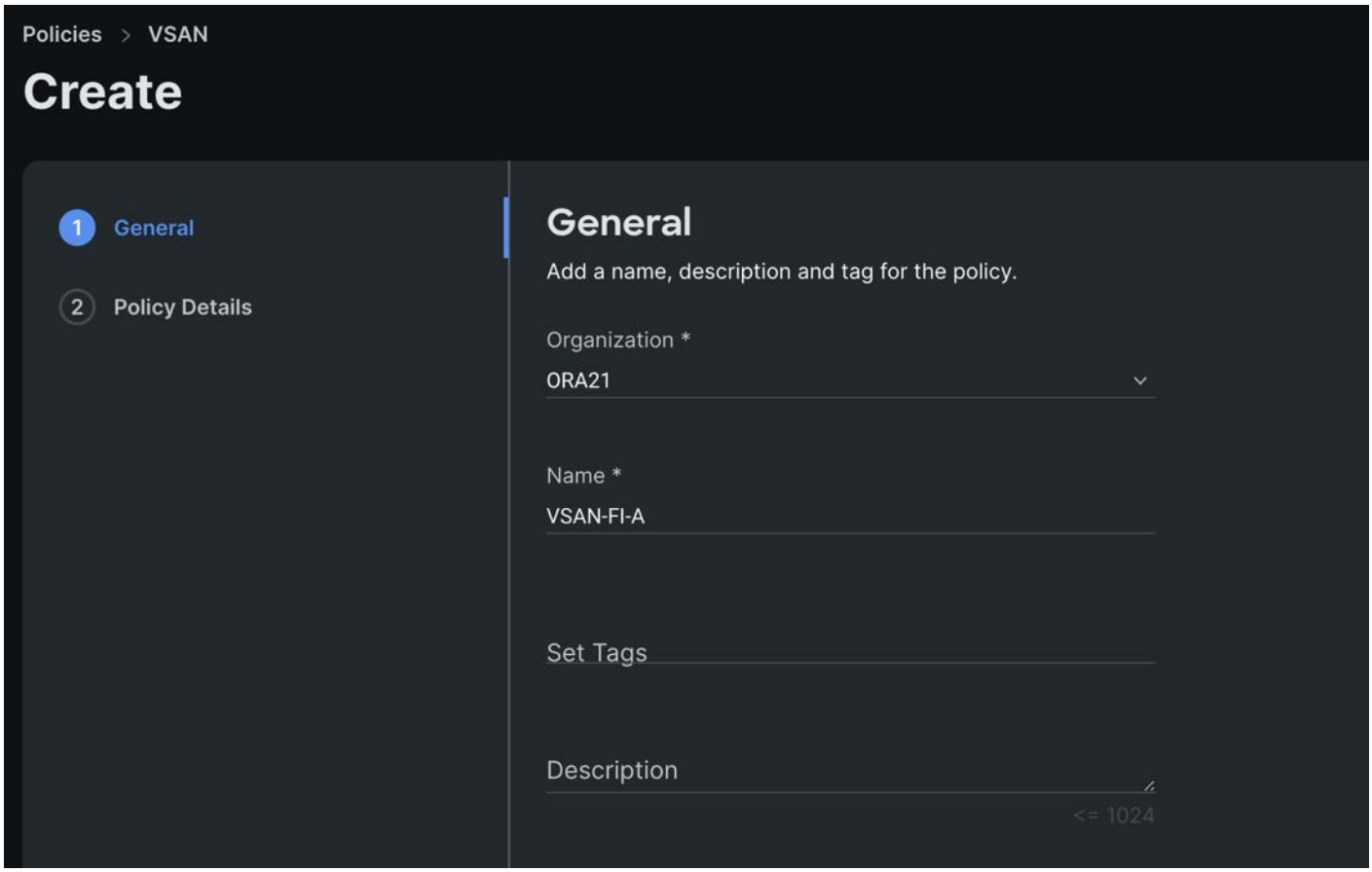



**Step 5.** Click Create to create this policy.

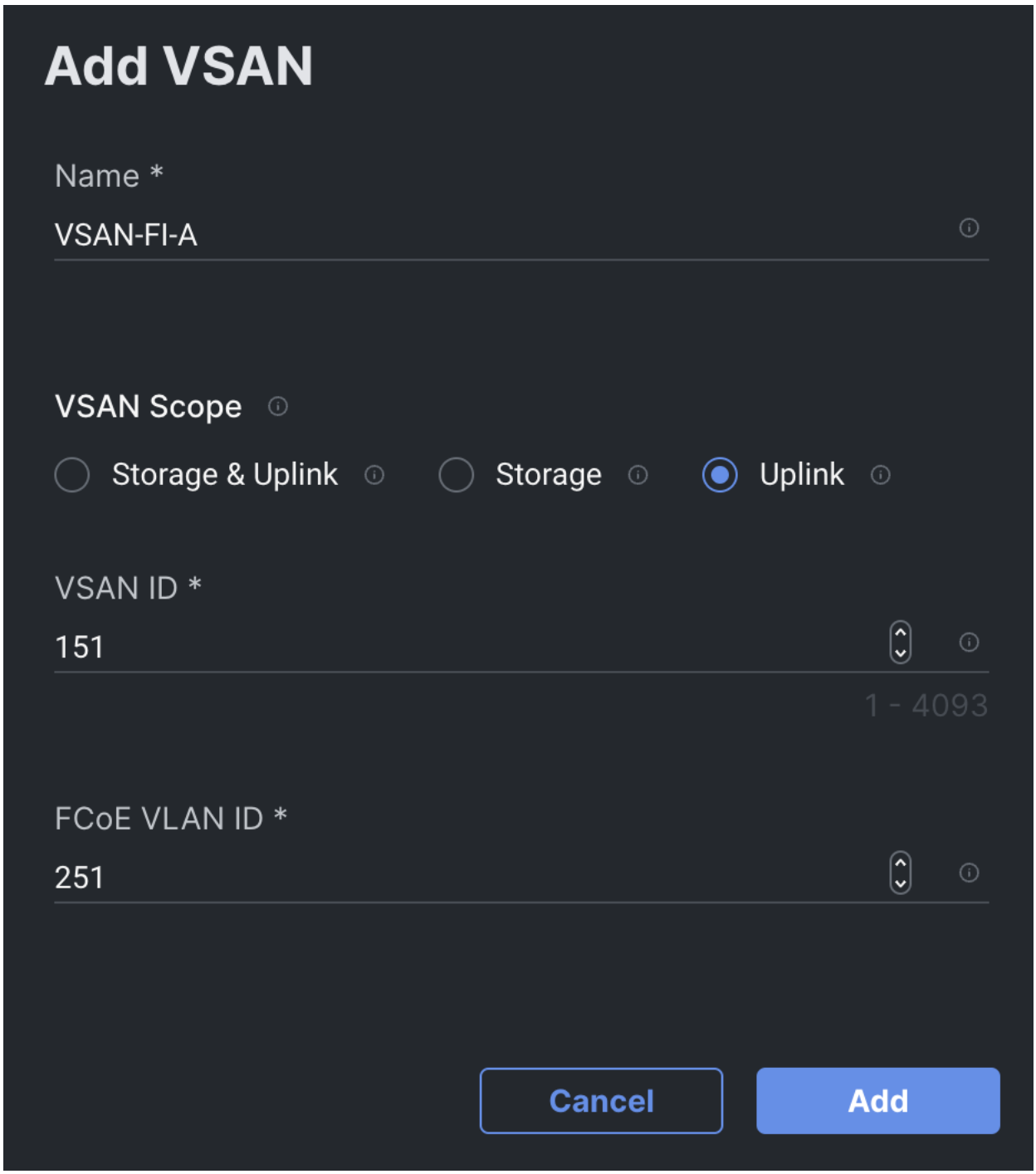
**Procedure 3.** Configure VSANs

**Step 1.** To configure the VSAN Policy for the Cisco UCS Domain profile, go to > Infrastructure Service > Configure > Polices > and click Create Policy. For the platform type select “UCS Domain” and for the Policy select “VSAN.”

**Step 2.** In the VSAN Policy Create section, for the Organization select “ORA21” and for the Policy name select “VLAN-FI-A.” Click Next.



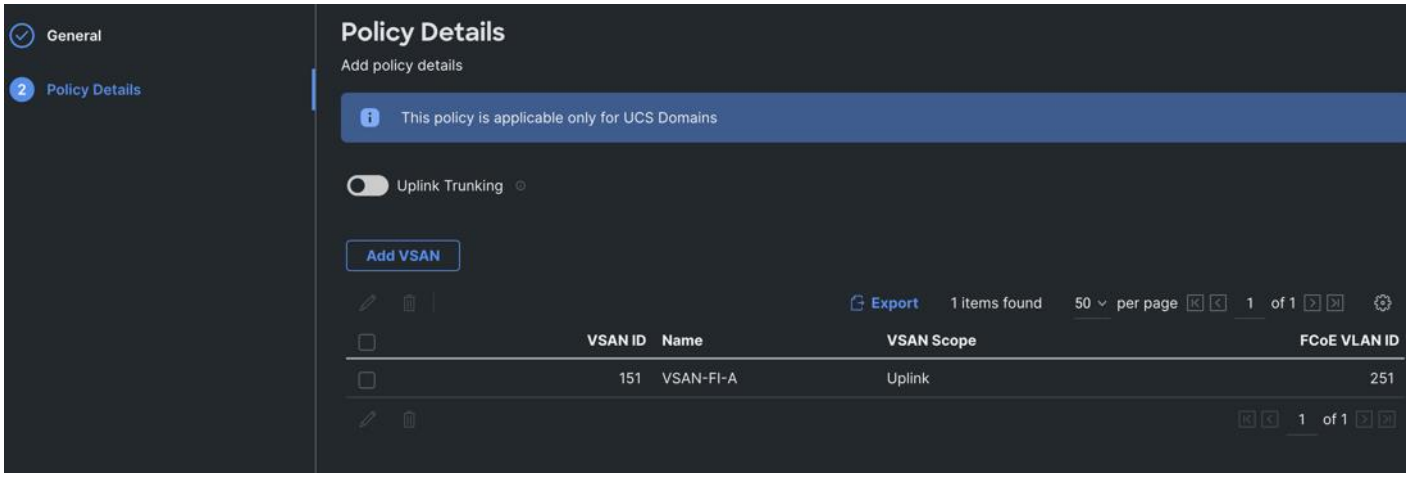

**Step 3.** In the Policy Details section, to configure the individual VSAN, select " Add VSAN." Provide a name, VSAN ID, FCoE VLAN ID and VSAN Scope for the VSAN on FI-A side as shown below:




**Note:** Storage & Uplink VSAN scope allows you to provision SAN and Direct Attached Storage, using the fabric interconnect running in FC Switching mode. You must externally provision the zones for the VSAN on upstream FC/FCoE switches. Storage VSAN scope allows you to connect and configure Direct Attached Storage, using the fabric interconnect running in FC Switching mode. You can configure local zones on this VSAN using FC Zone policies. All unmanaged zones in the fabric interconnect are cleared when this VSAN is configured for the first time. Do NOT configure this VSAN on upstream FC/FCoE switches.

**Note:** Uplink scope VSAN allows you to provision SAN connectivity using the Fabric Interconnect.

**Step 4.** Click Add to add this VSAN to the policy.



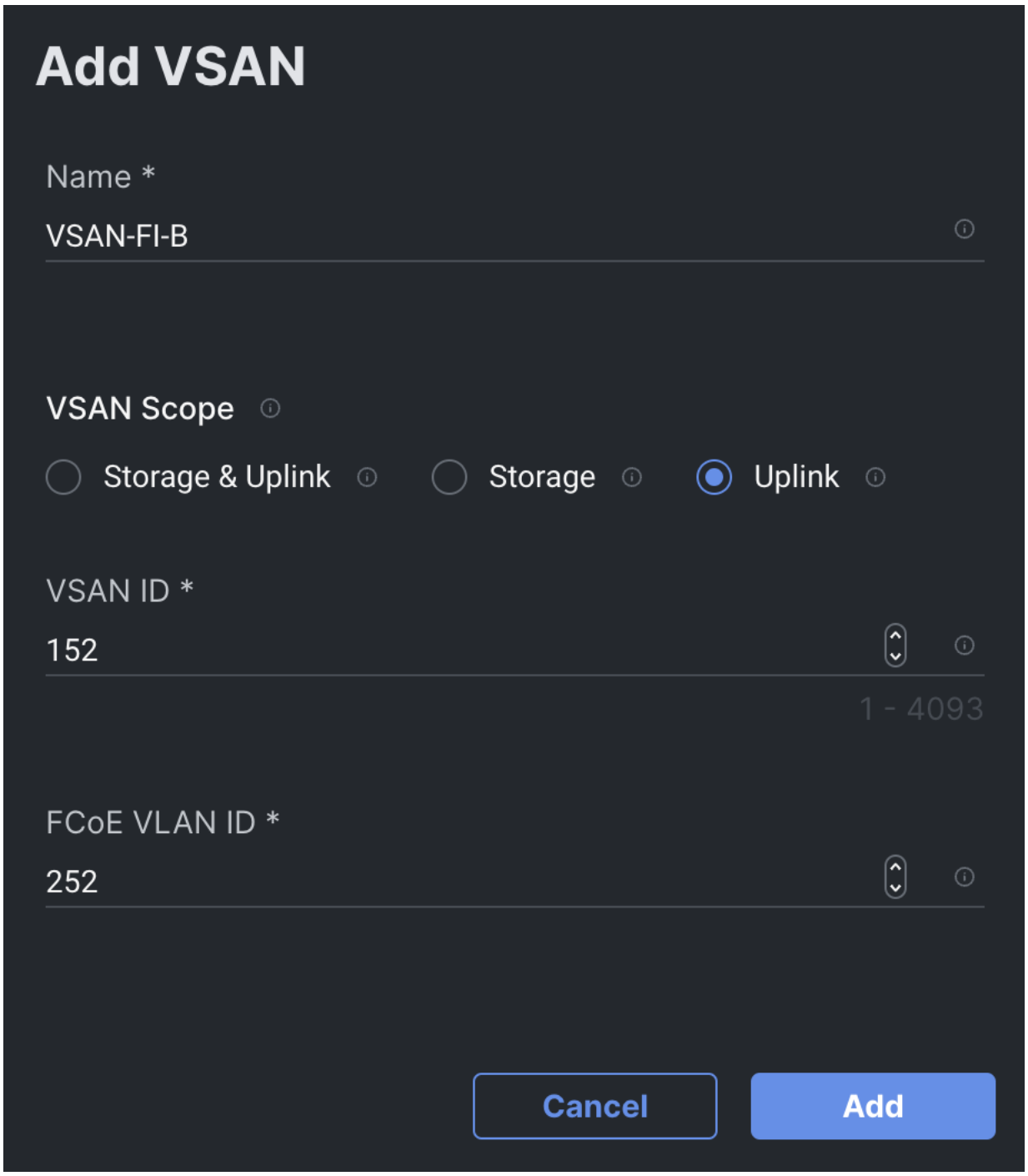

**Step 5.** Click Create to create this VSAN policy for FI-A.

**Step 6.** Configure VSAN policy for FI-B:

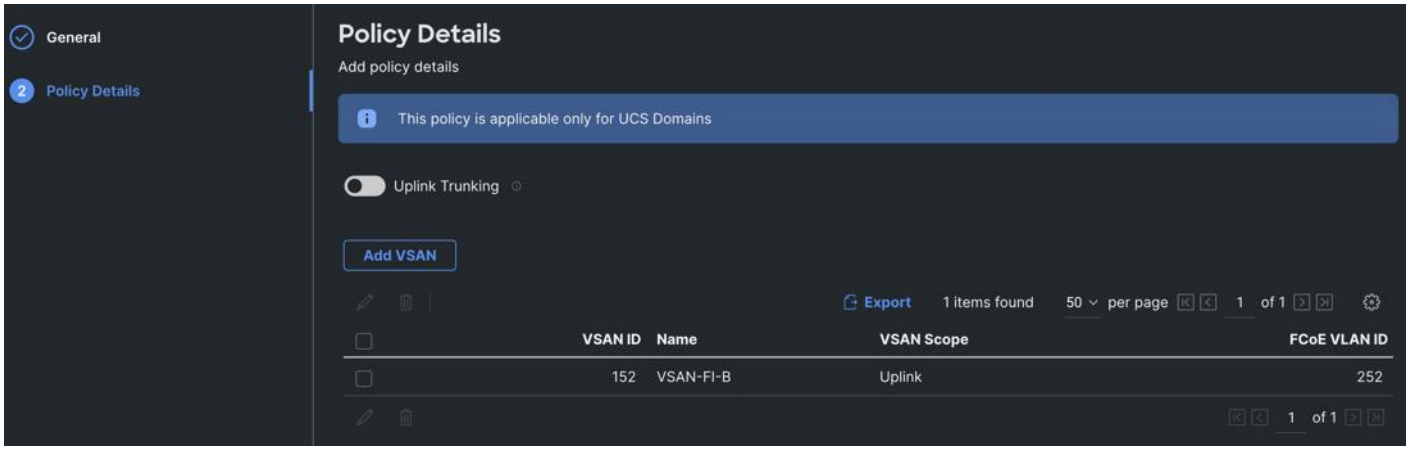
a. To configure the VSAN Policy for the Cisco UCS Domain profile, go to > Infrastructure Service > Configure > Polices > and click Create Policy. For the platform type select “UCS Domain” and for the Policy select “VSAN.”

b. In the VSAN Policy Create section, for the Organization select “ORA21” and for the Policy name select “VLAN-FI-B.” Click Next.

c. In the Policy Details section, to configure the individual VSAN, select " Add VSAN." Provide a name, VSAN ID, FCoE VLAN ID and VSAN Scope for the VSAN on FI-B side as shown below:




**Step 7.** Click Add to add this VSAN to the policy.

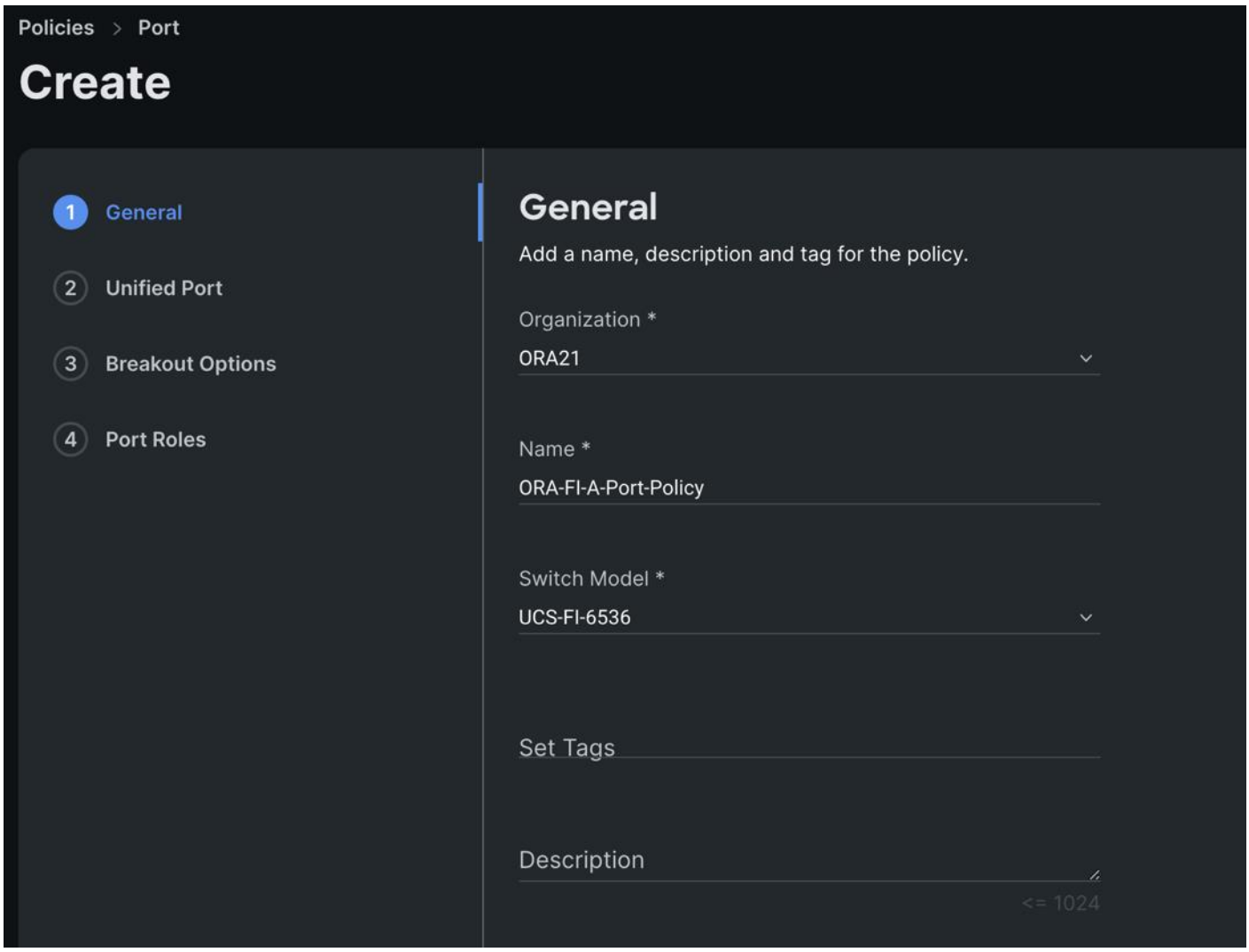



**Step 8.** Click Create to create this VSAN policy for FI-B.

## Procedure 4. Configure Port Policy

**Step 1.** To configure the Port Policy for the Cisco UCS Domain profile, go to > Infrastructure Service > Configure > Polices > and click Create Policy. For the platform type select “UCS Domain” and for the policy, select “Port.”

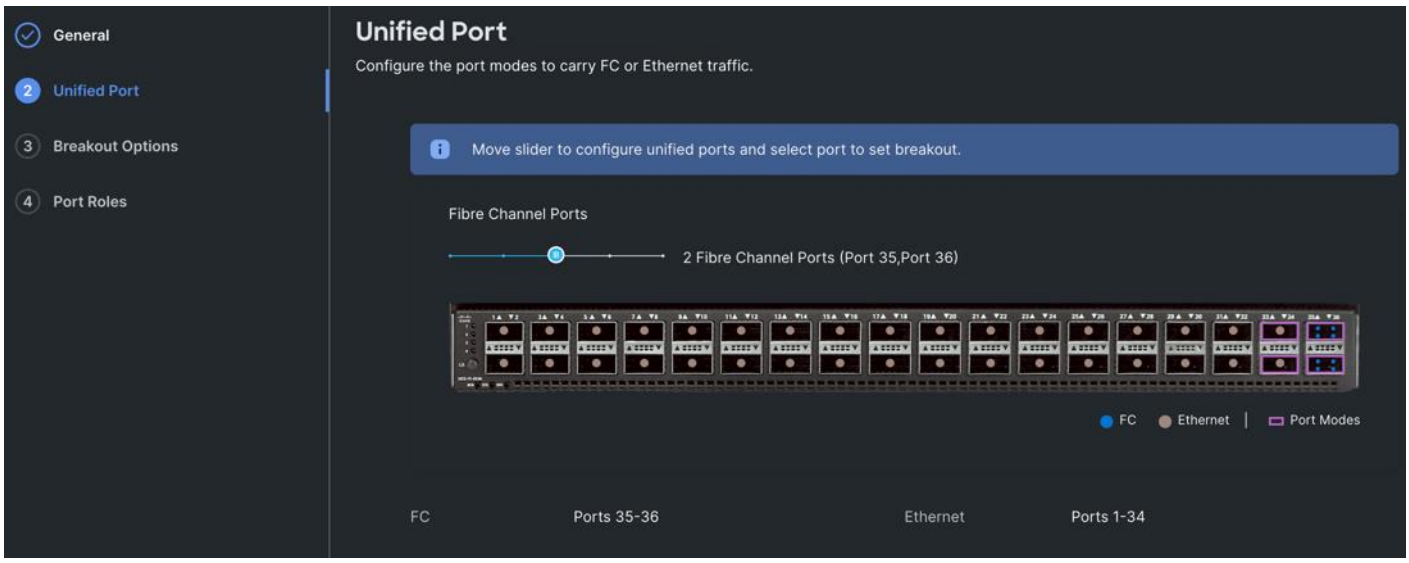
**Step 2.** In the Port Policy Create section, for the Organization, select “ORA21”, for the policy name select “ORA-FI-A-Port-Policy” and for the Switch Model select " UCS-FI-6536.” Click Next.



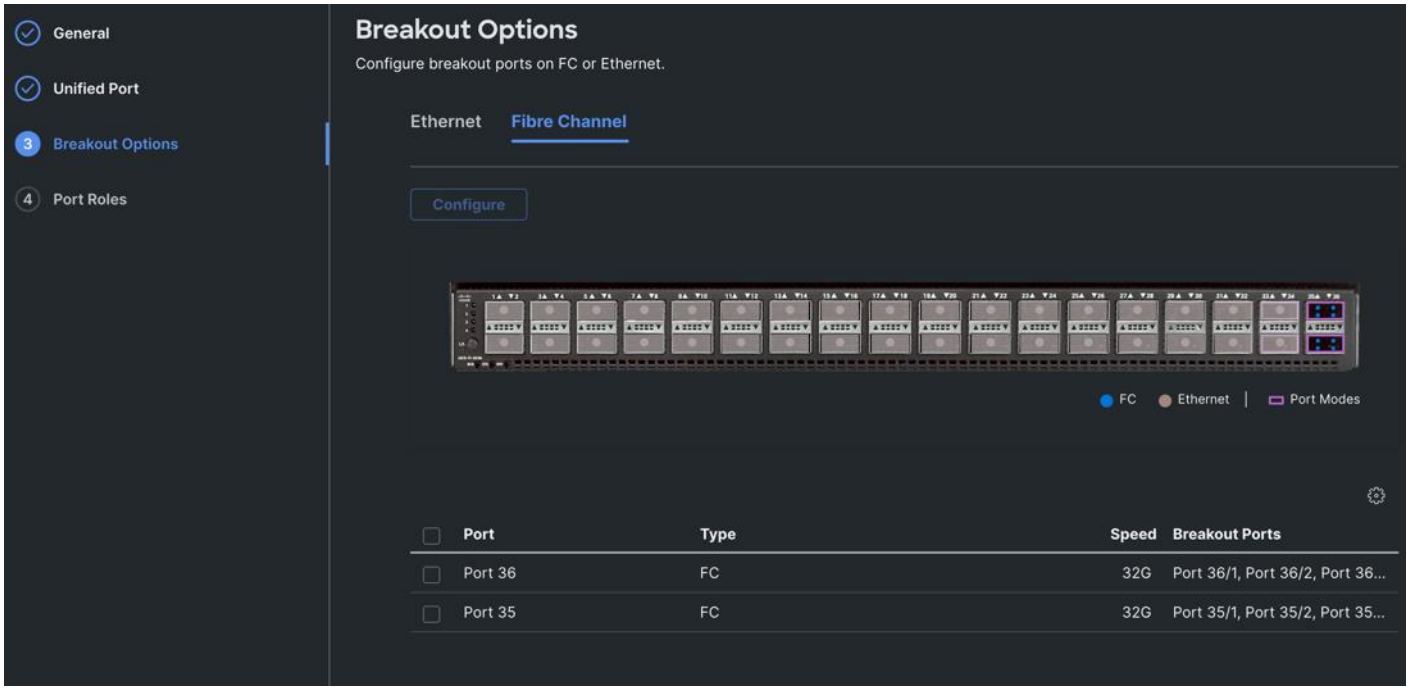

**Note:** We did not configure the Fibre Channel Ports for this solution. In the Unified Port section, leave it as default and click Next.

**Note:** We did not configure the Breakout options for this solution. Leave it as default and click Next.

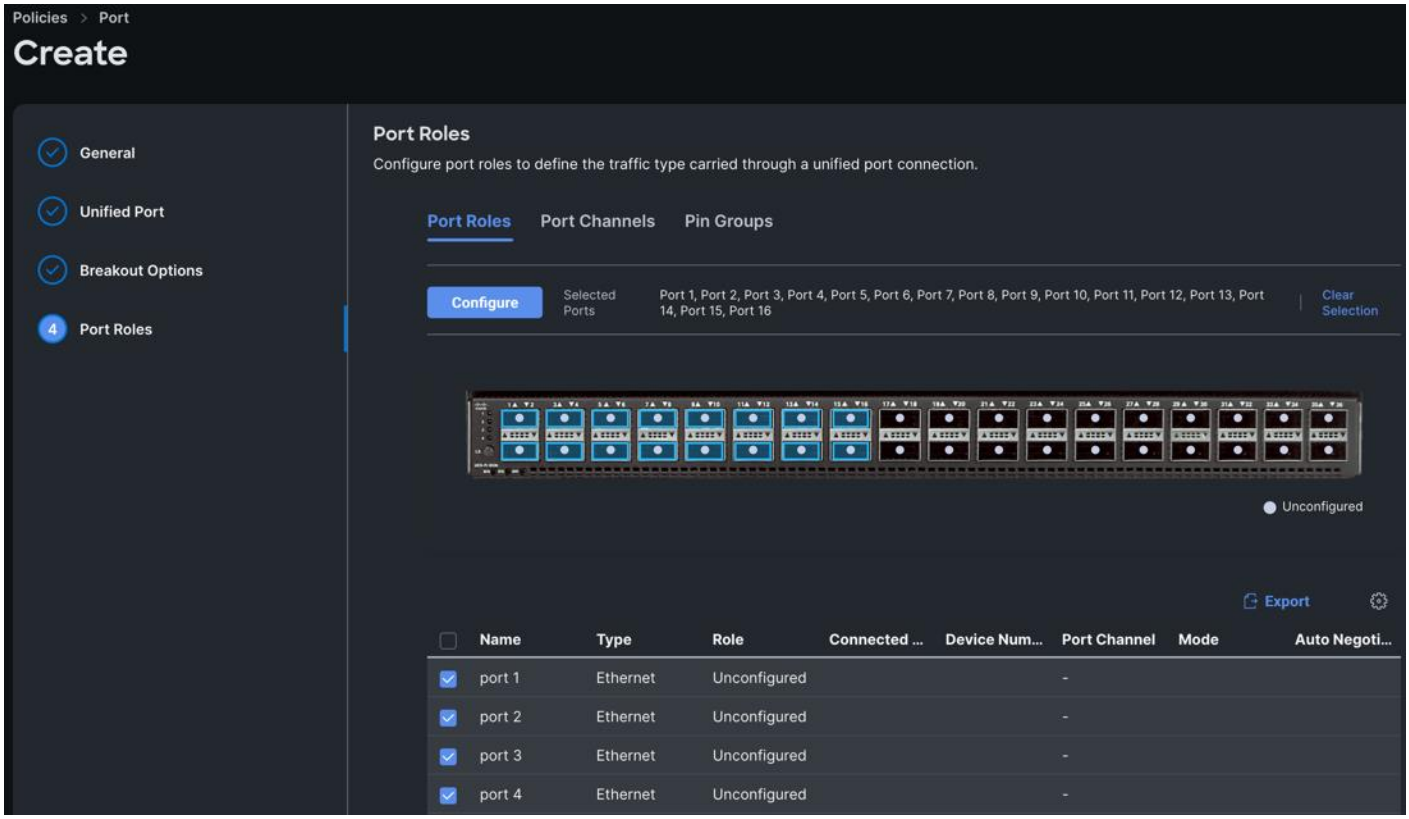
**Step 3.** In the Unified Port section, move the slider to right side as shown below. This changes Port 35 and Port 36 to FC port.



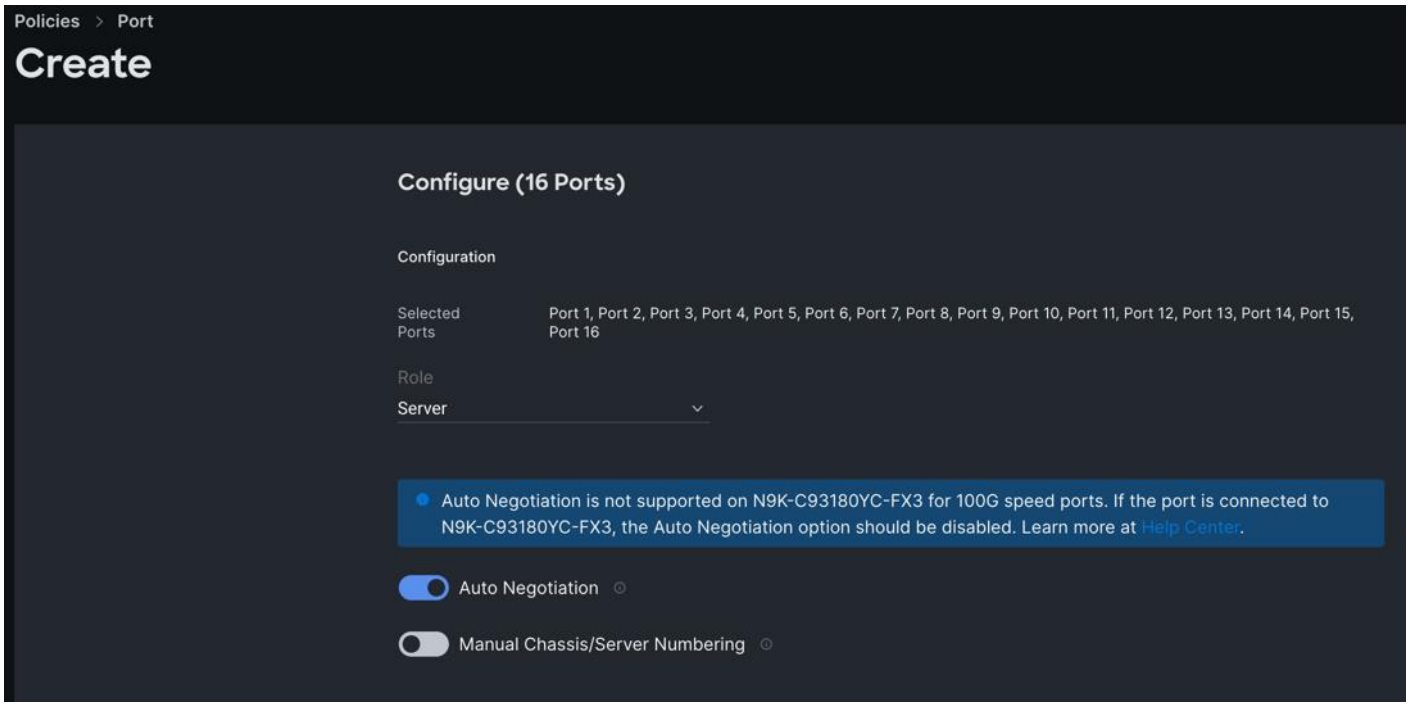

**Step 4.** In the Breakout Options section, go to Fibre Channel tab and select Port 35 and 36 and click Configure. Set Port 35 and 36 to “4x32G” and click Next.




**Step 5.** In the Port Role section, select port 1 to 16 and click Configure.

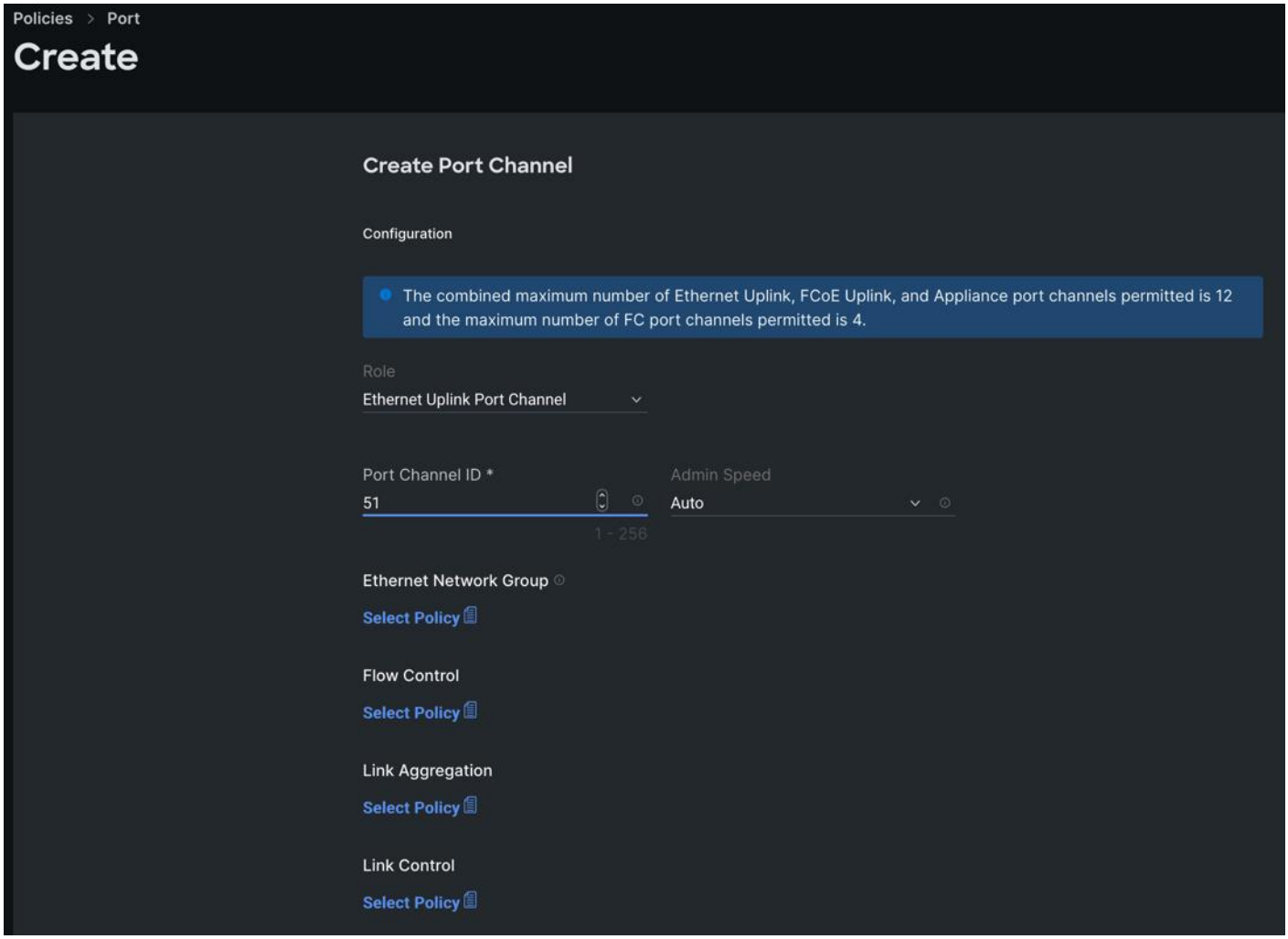



**Step 6.** In the Configure section, for Role select Server and keep the Auto Negotiation ON.

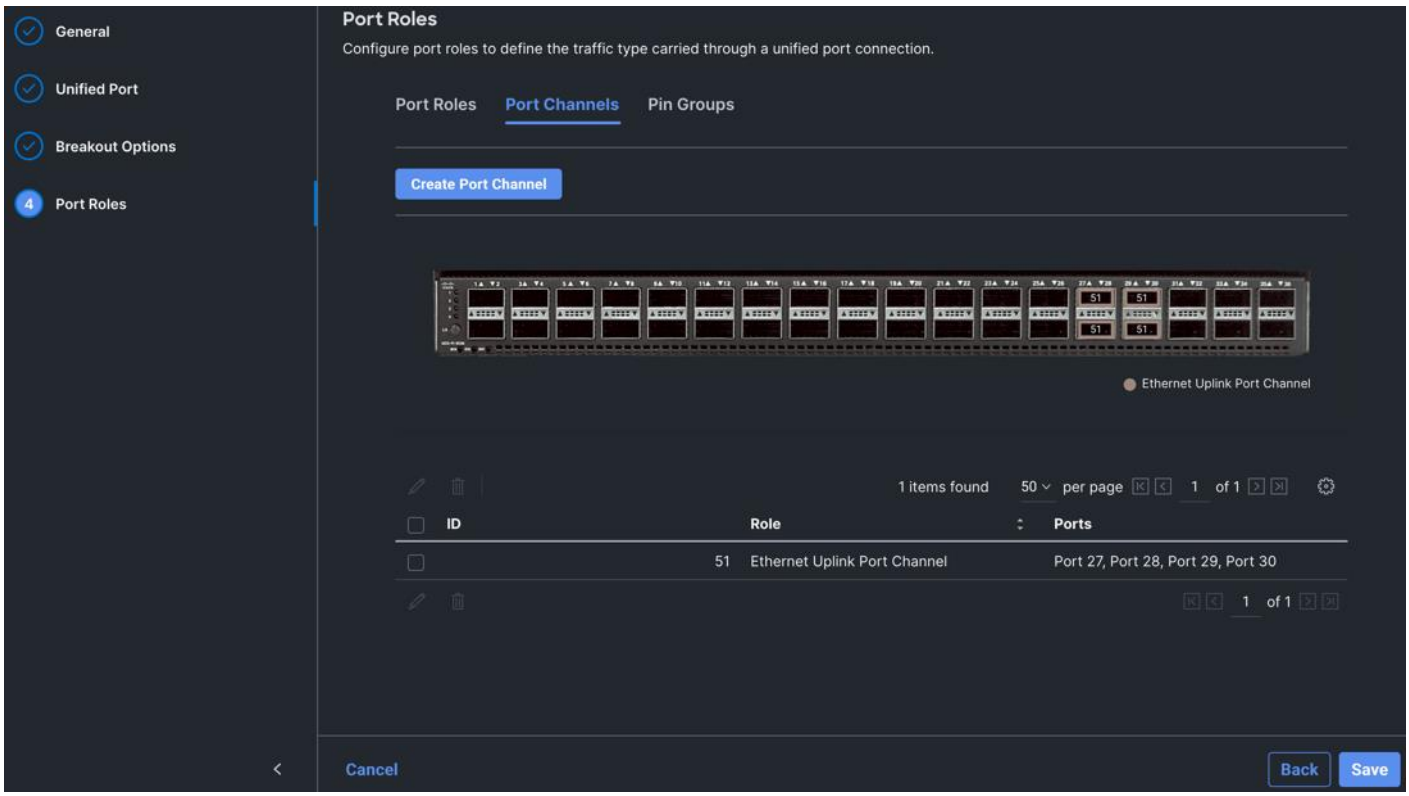



**Step 7.** Click SAVE to add this configuration for port roles.

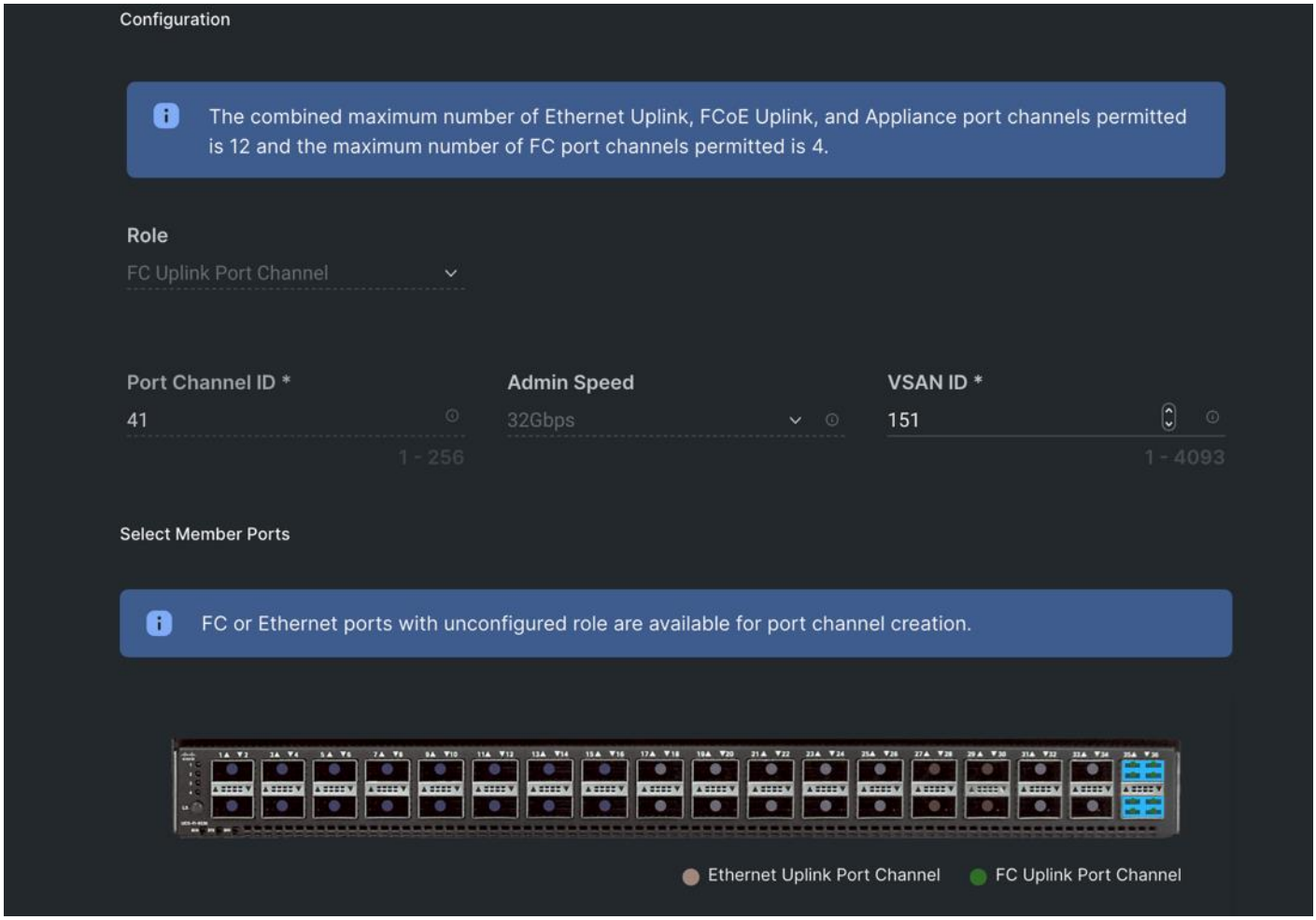
**Step 8.** Go to the Port Channels tab and select Port 27 to 30 and click Create Port Channel between FI-A and both Cisco Nexus Switches. In the Create Port Channel section, for Role select Ethernet Uplinks Port Channel, and for the Port Channel ID select 51 and select Auto for the Admin Speed.




**Step 9.** Click SAVE to add this configuration for uplink port roles.

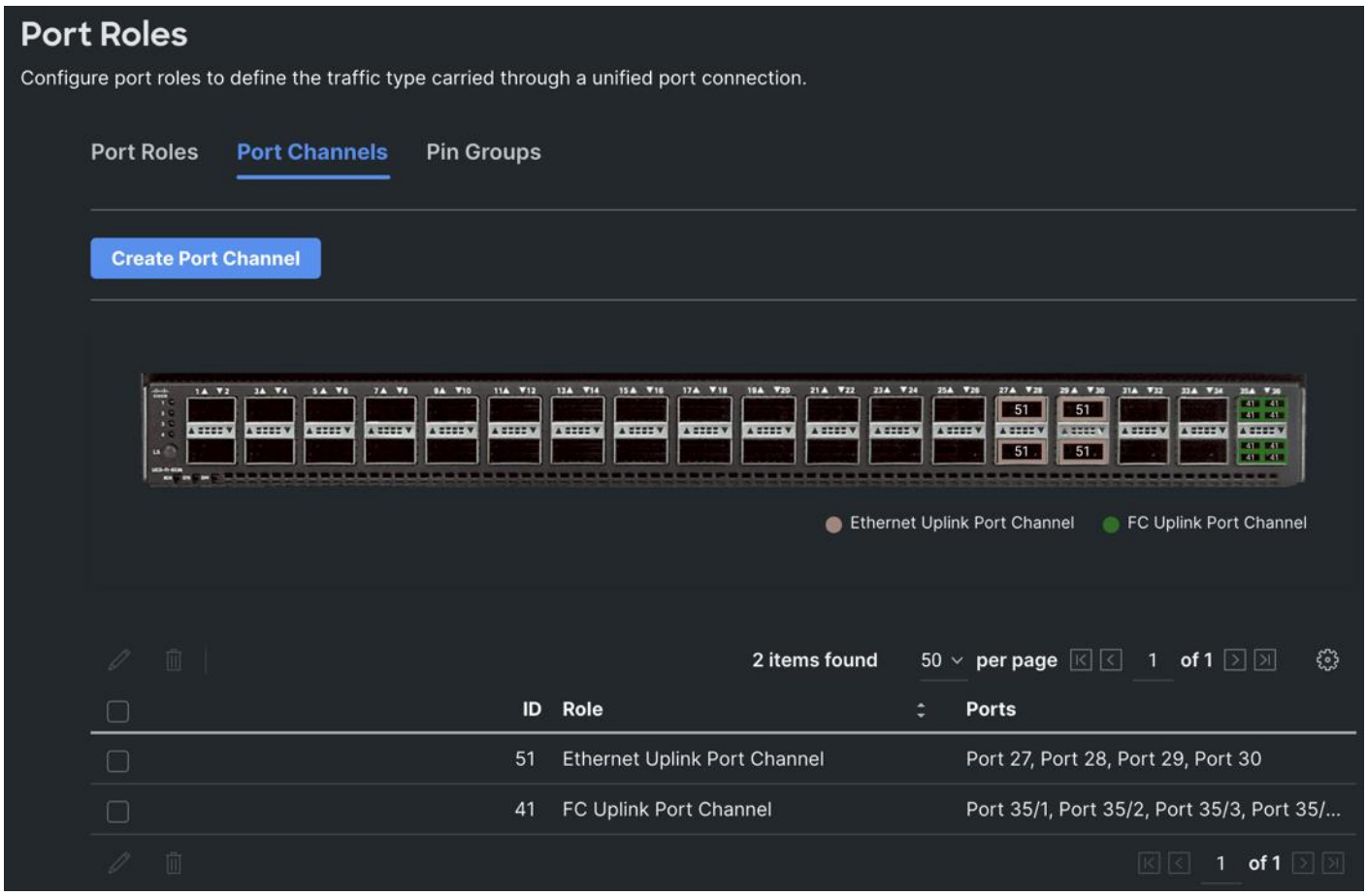



**Step 10.** Go to the Port Channels tab and now select Port 35/1 to 35/4 and 36/1 to 36/4. Click Create Port Channel between FI-A and Cisco MDS A Switch. In the Create Port Channel section, for Role select FC Uplink Port Channel, and for the Port Channel ID select 41 and enter 151 as VSAN ID.




**Step 11.** Click SAVE to add this configuration for storage uplink port roles.

**Step 12.** Verify both the port channel as shown below:



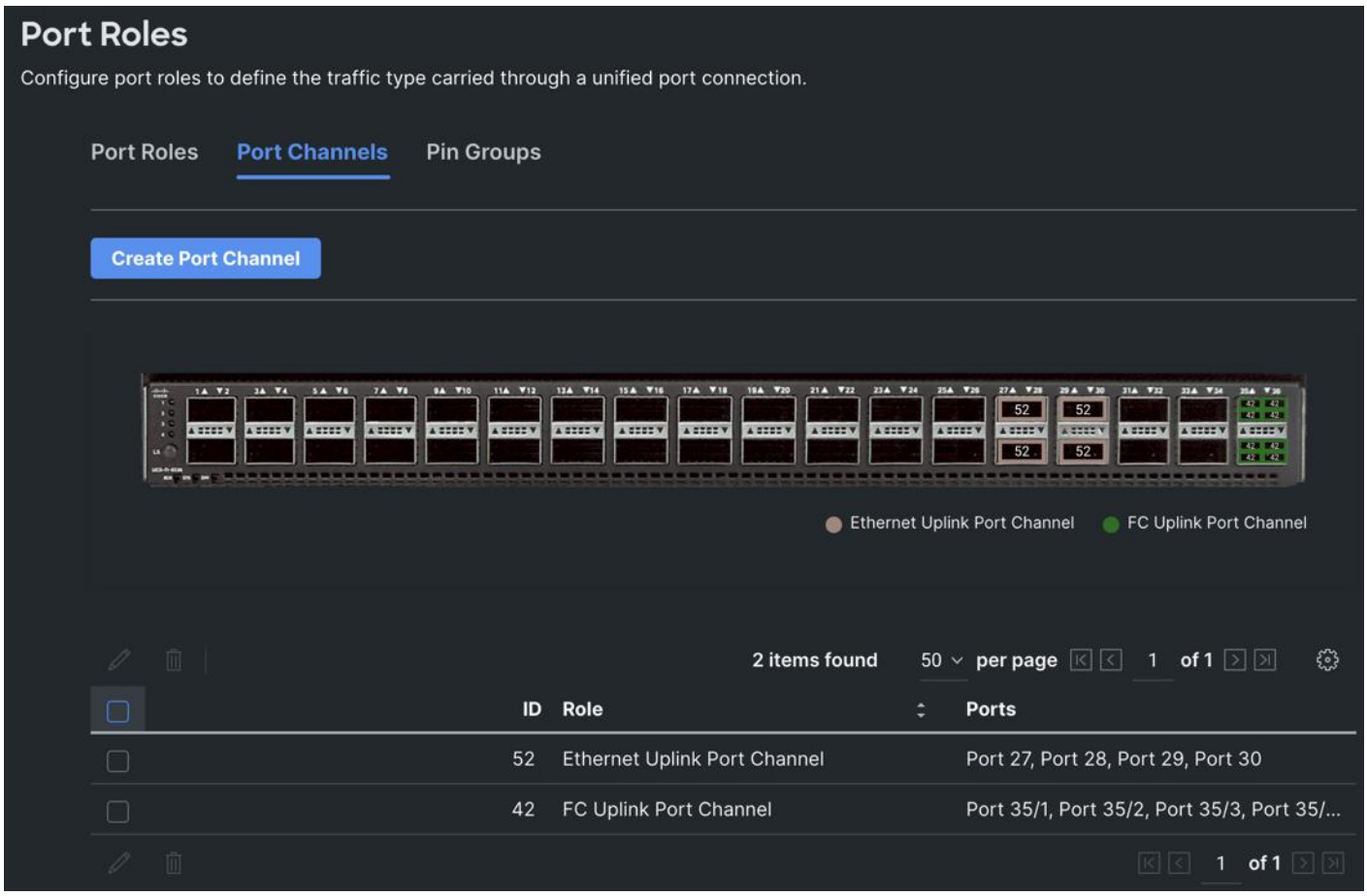

**Step 13.** Click SAVE to complete this configuration for all the server ports and uplink port roles.

**Note:** We configured the FI-B ports and created a Port Policy for FI-B, “ORA-FI-B-Port-Policy.”

**Note:** In the FI-B port policy, we also configured unified ports as well as breakout options for 4x32G on port 35 and 36 for FC Traffic.

**Note:** As configured for FI-A, we configured the port policy for FI-B. For FI-B, configured port 1 to 16 for server ports, port 27 to 30 as the ethernet uplink port-channel ports and 35/1-35/4 to 36/1-36/4 ports as FC uplink Port channel ports.

**Note:** For FI-B, we configured Port-Channel ID as 52 for Ethernet Uplink Port Channel and Port-Channel ID as 42 for FC Uplink Port Channel as shown below:



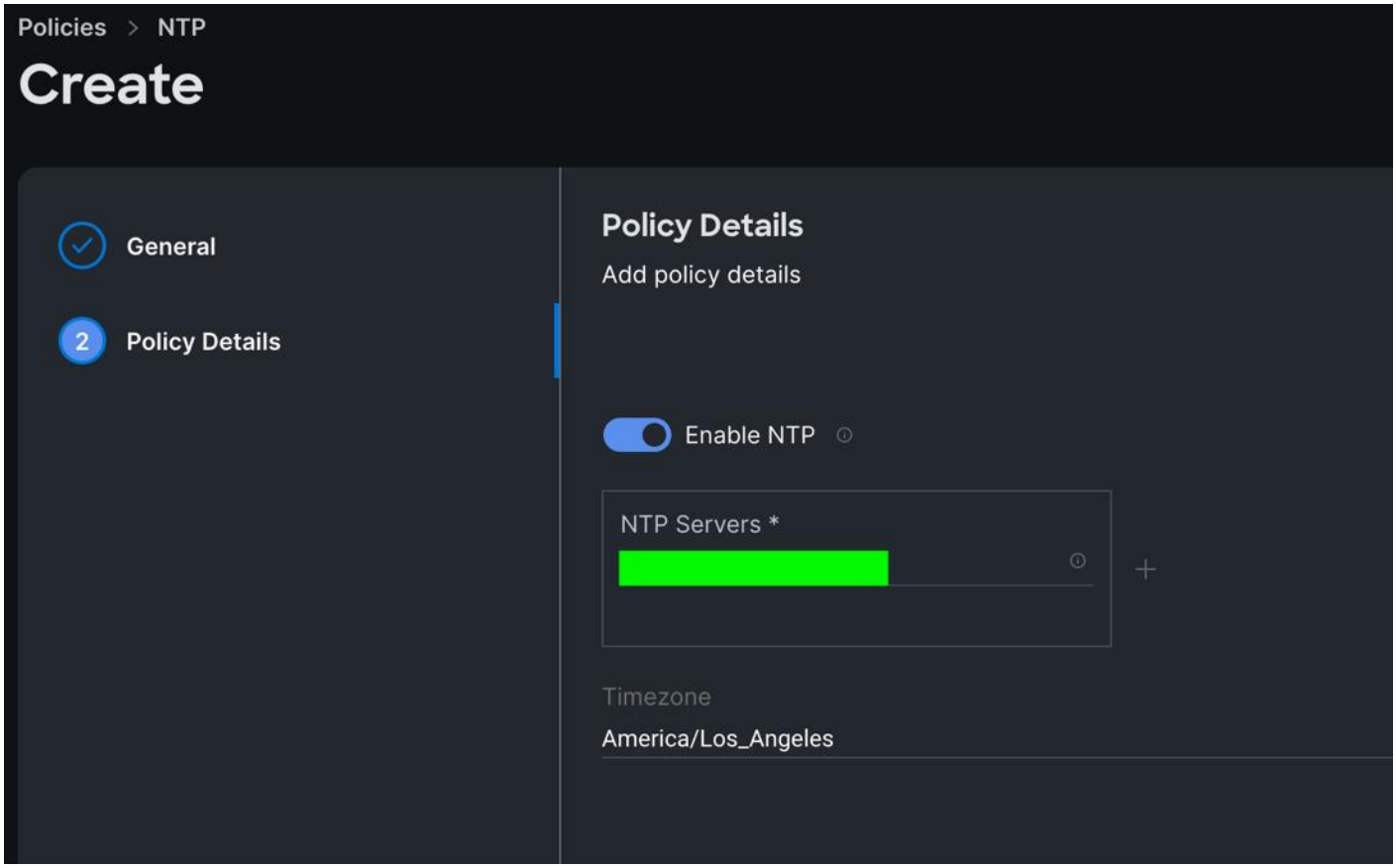

This completes the Port Policy for FI-A and FI-B for Cisco UCS Domain profile.

## Procedure 5. Configure NTP Policy

**Step 1.** To configure the NTP Policy for the Cisco UCS Domain profile, go to > Infrastructure Service > Configure > Polices > and click Create Policy. For the platform type select “UCS Domain” and for the policy select “NTP.”

**Step 2.** In the NTP Policy Create section, for the Organization select “ORA21” and for the policy name select “NTP-Policy.” Click Next.

**Step 3.** In the Policy Details section, select the option to enable the NTP Server and enter your NTP Server details as shown below.



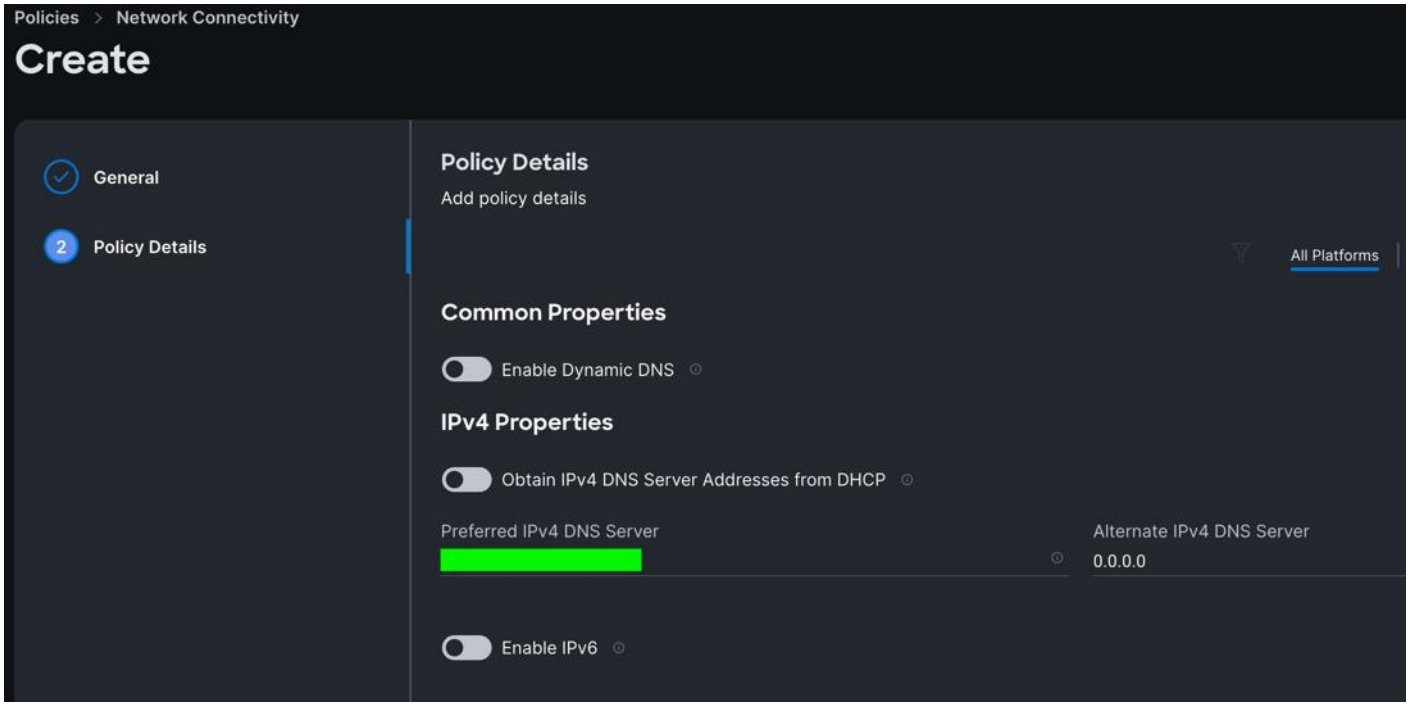

**Step 4.** Click Create.

**Procedure 6.** Configure Network Connectivity Policy

**Step 1.** To configure to Network Connectivity Policy for the Cisco UCS Domain profile, go to > Infrastructure Service > Configure > Polices > and click Create Policy. For the platform type select “UCS Domain” and for the policy select “Network Connectivity.”

**Step 2.** In the Network Connectivity Policy Create section, for the Organization select “ORA21” and for the policy name select “Network-Connectivity-Policy.” Click Next.

**Step 3.** In the Policy Details section, enter the IPv4 DNS Server information according to your environment details as shown below:



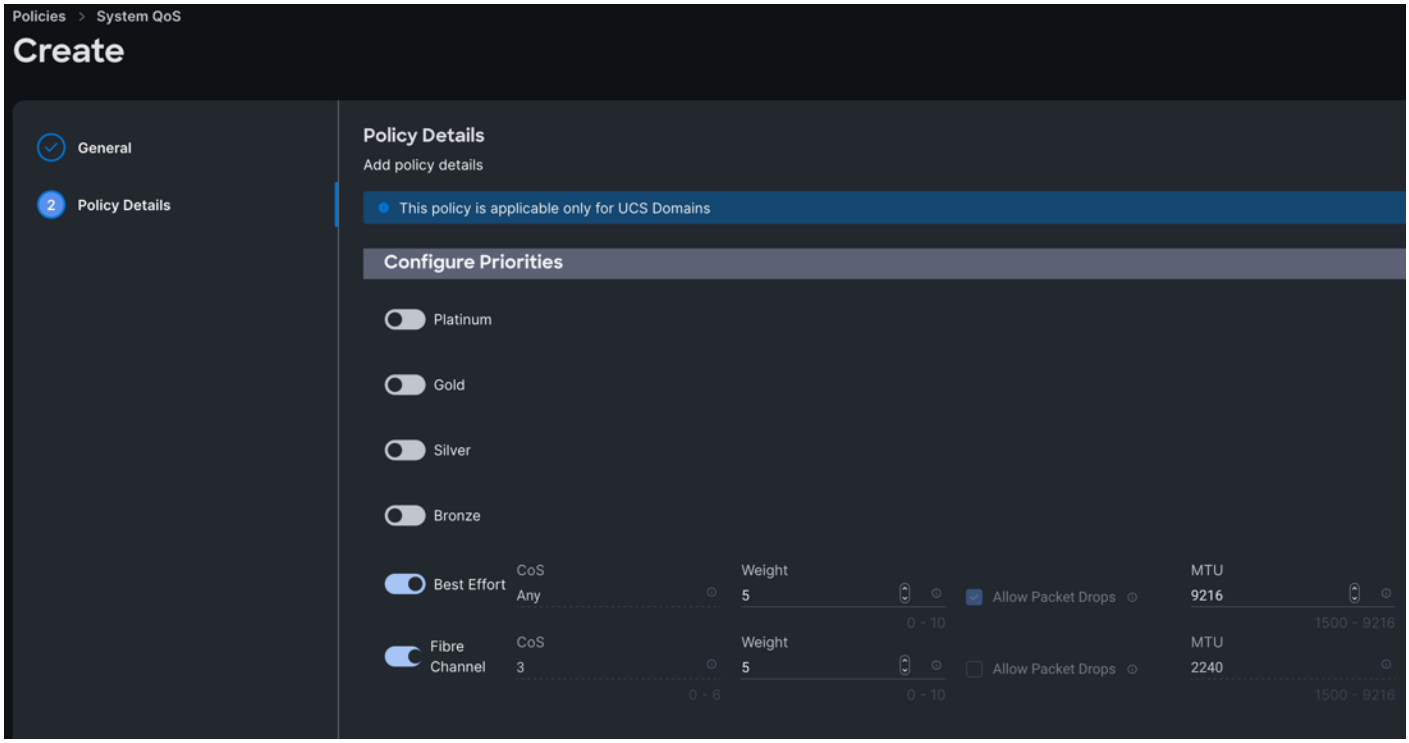

**Step 4.** Click Create.

## Procedure 7. Configure System QoS Policy

**Step 1.** To configure the System QoS Policy for the Cisco UCS Domain profile, go to > Infrastructure Service > Configure > Polices > and click Create Policy. For the platform type select “UCS Domain” and for the policy select “System QoS.”

**Step 2.** In the System QoS Policy Create section, for the Organization select “ORA21” and for the policy name select “ORA-QoS.” Click Next.

**Step 3.** In the Policy Details section under Configure Priorities, select Best Effort and set the MTU size to 9216.



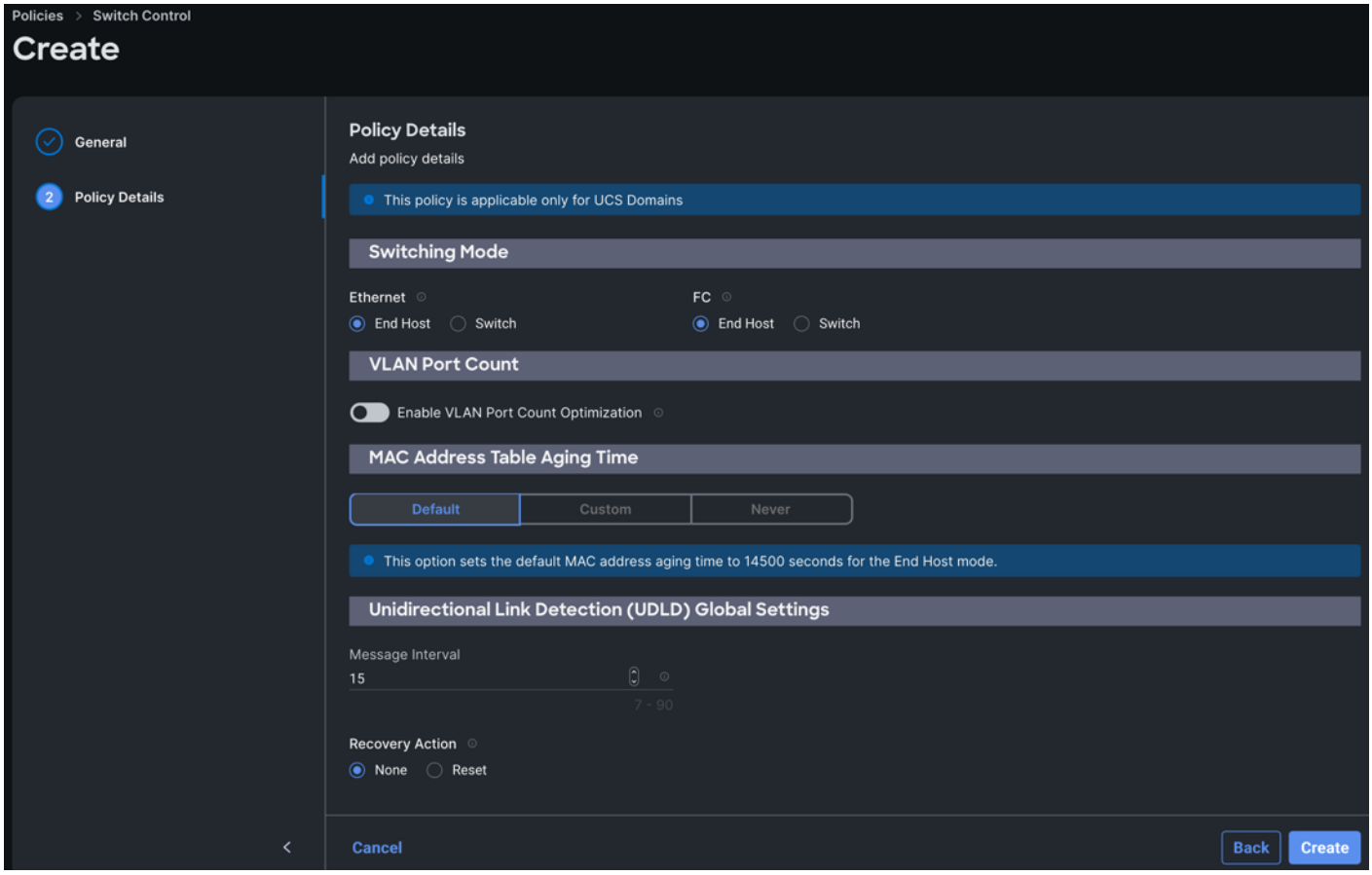

**Step 4.** Click Create.

## Procedure 8. Configure Switch Control Policy

**Step 1.** To configure the Switch Control Policy for the UCS Domain profile, go to > Infrastructure Service > Configure > Polices > and click Create Policy. For the platform type select “UCS Domain” and for the policy select “Switch Control.”

**Step 2.** In the Switch Control Policy Create section, for the Organization select “ORA21” and for the policy name select “ORA-Switch-Control.” Click Next.

**Step 3.** In the Policy Details section, for the Switching Mode for Ethernet as well as FC, select and keep " End Host" Mode.




**Step 4.** Click Create to create this policy.

## Configure Cisco UCS Domain Profile

With Cisco Intersight, a domain profile configures a fabric interconnect pair through reusable policies, allows for configuration of the ports and port channels, and configures the VLANs and VSANs in the network. It defines the characteristics of and configures ports on fabric interconnects. You can create a domain profile and associate it with a fabric interconnect domain. The domain-related policies can be attached to the profile either at the time of creation or later. One UCS Domain profile can be assigned to one fabric interconnect domain. For more information, go to: https://intersight.com/help/saas/features/fabric_interconnects/configure#domain_profile

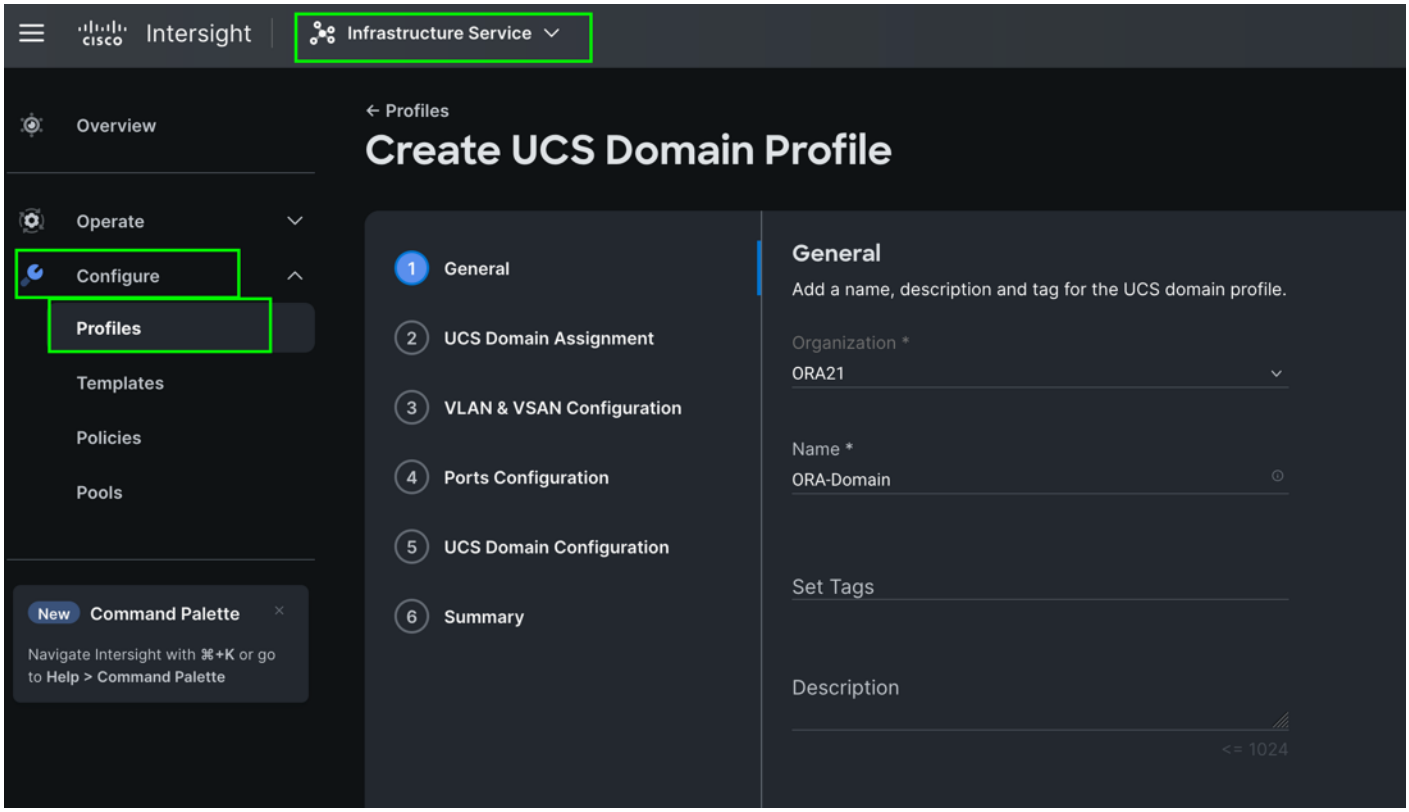
Some of the characteristics of the Cisco UCS domain profile in the FlashStack environment are:

- A single domain profile (ORA-Domain) is created for the pair of Cisco UCS fabric interconnects.
- Unique port policies are defined for the two fabric interconnects.
- The VLAN configuration policy is common to the fabric interconnect pair because both fabric interconnects are configured for the same set of VLANs.
- The VSAN configuration policy is different to each of the fabric interconnects because both fabric interconnects are configured to carry separate storage traffic through separate VSANs.

- The Network Time Protocol (NTP), network connectivity, and system Quality-of-Service (QoS) policies are common to the fabric interconnect pair.

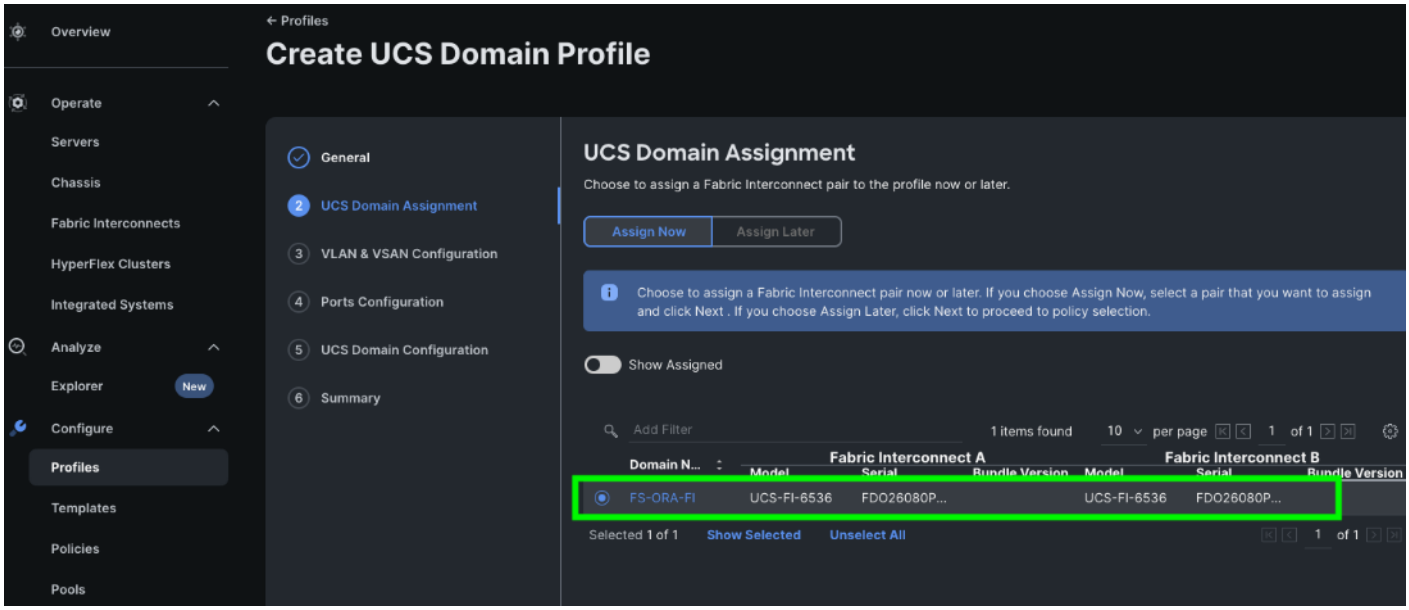
**Procedure 1.** Create a domain profile

**Step 1.** To create a domain profile, go to Infrastructure Service > Configure > Profiles > then go to the UCS Domain Profiles tab and click Create UCS Domain Profile.

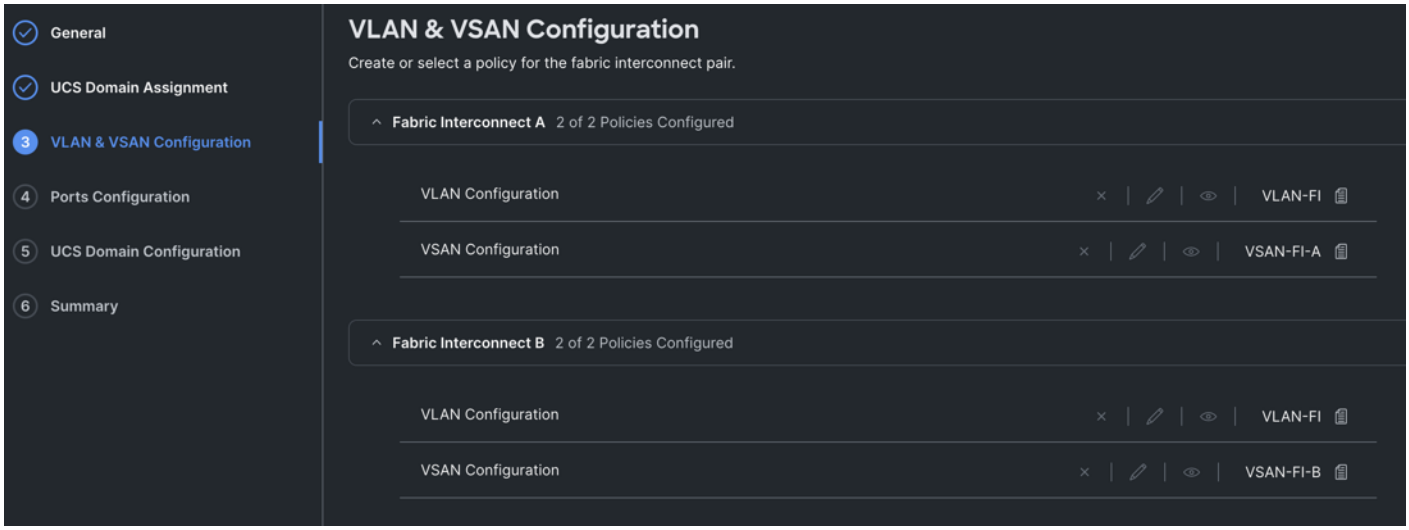



**Step 2.** For the domain profile name, enter “ORA-Domain” and for the Organization select what was previously configured. Click Next.

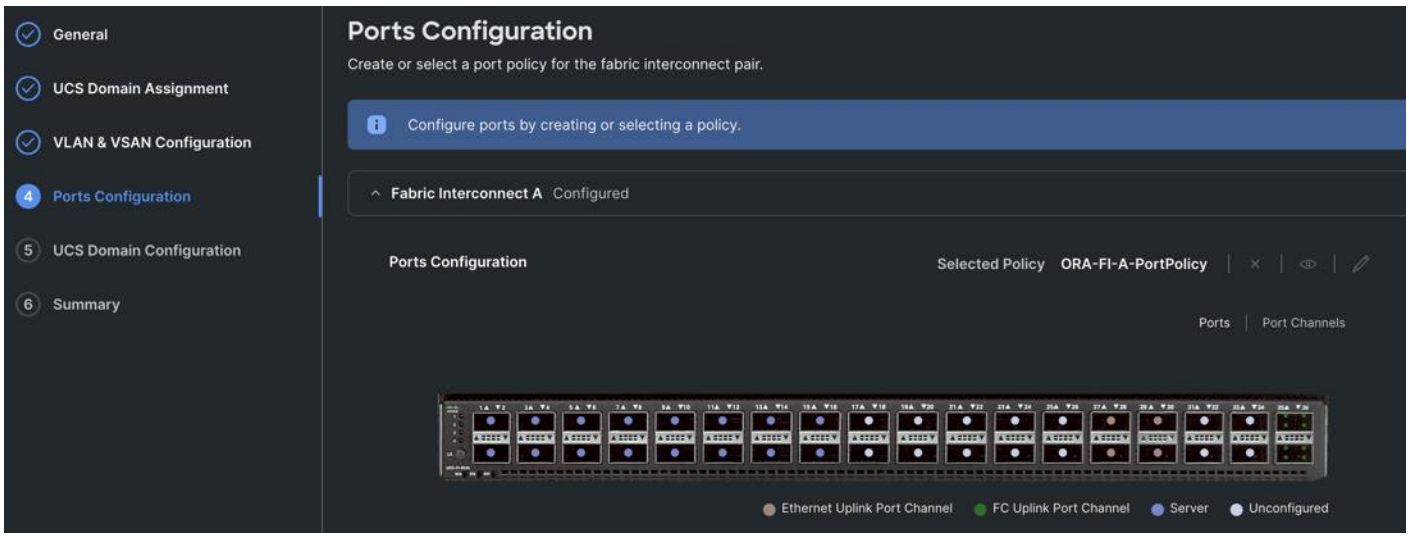
**Step 3.** In the UCS Domain Assignment menu, for the Domain Name select “ORA21C-FI” which was added previously into this domain and click Next.



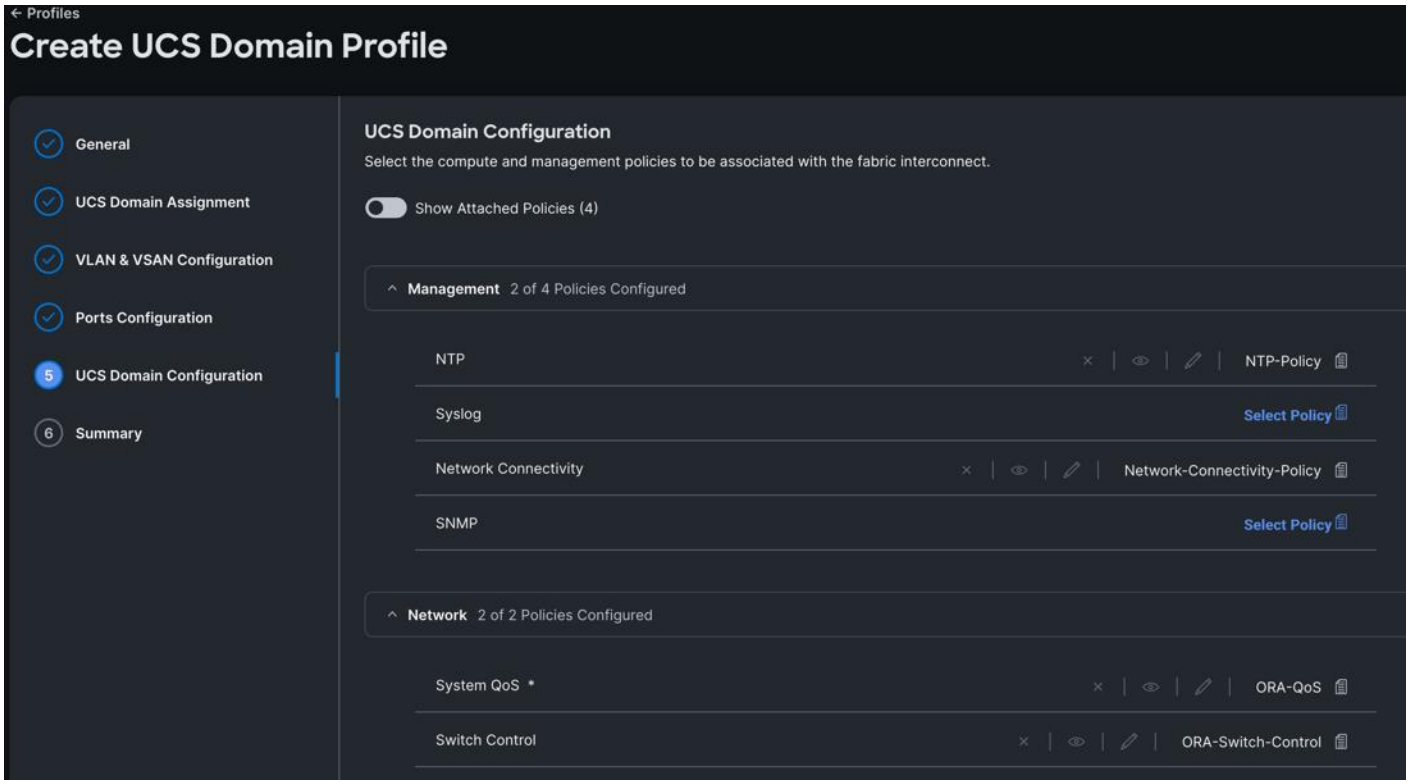

**Step 4.** In the VLAN & VSAN Configuration screen, for the VLAN Configuration for both FIs, select VLAN-FI. For the VSAN configuration for FI-A, select VSAN-FI-A and for FI-B select VSAN-FI-B that were configured in the previous section. Click Next.




**Step 5.** In the Port Configuration section, for the Port Configuration Policy for FI-A select ORA-FI-A-PortPolicy. For the port configuration policy for FI-B select ORA-FI-B-PortPolicy.



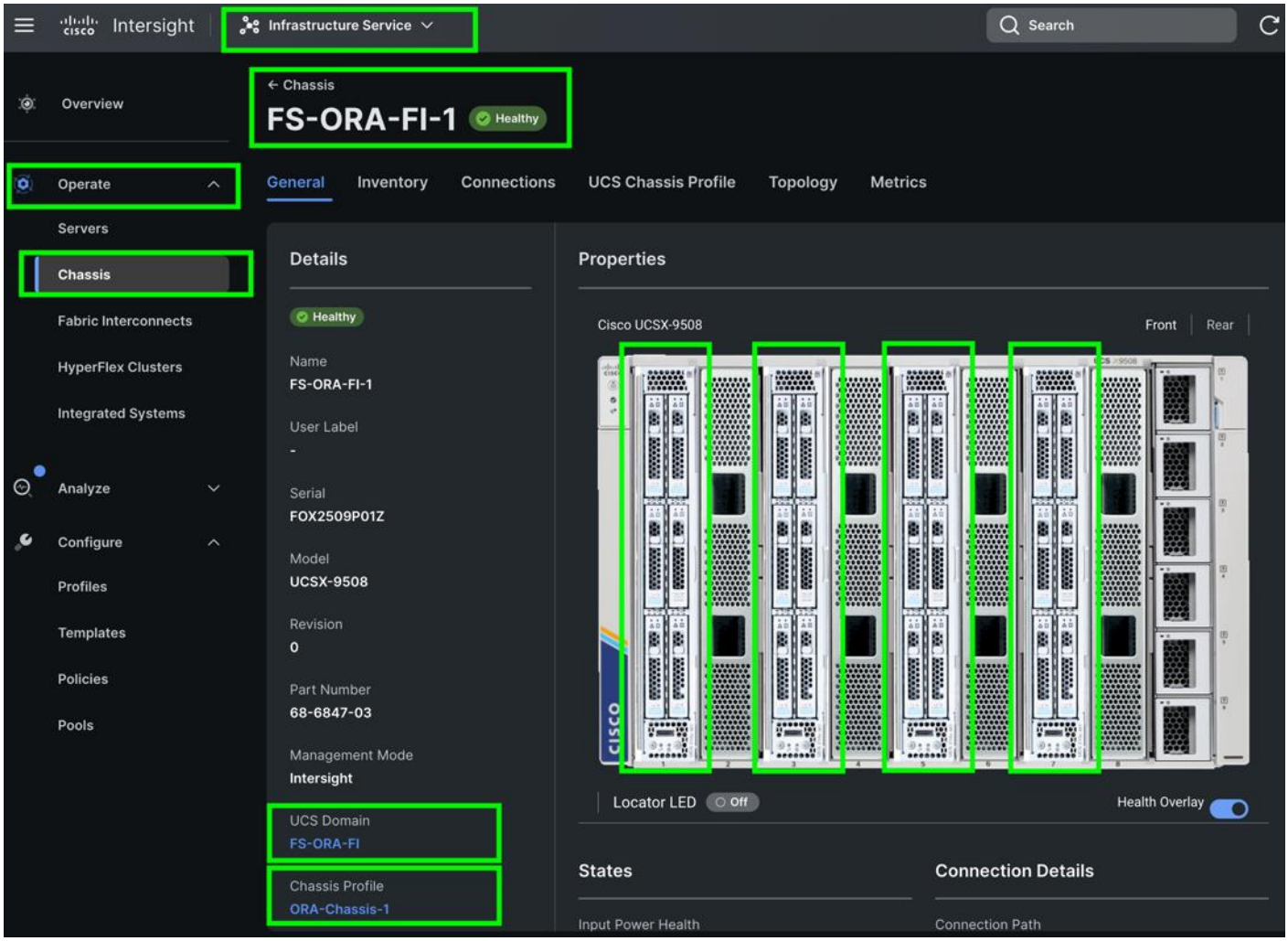

**Step 6.** In the UCS Domain Configuration section, select the policy for NTP, Network Connectivity, System QoS and Switch Control as shown below:

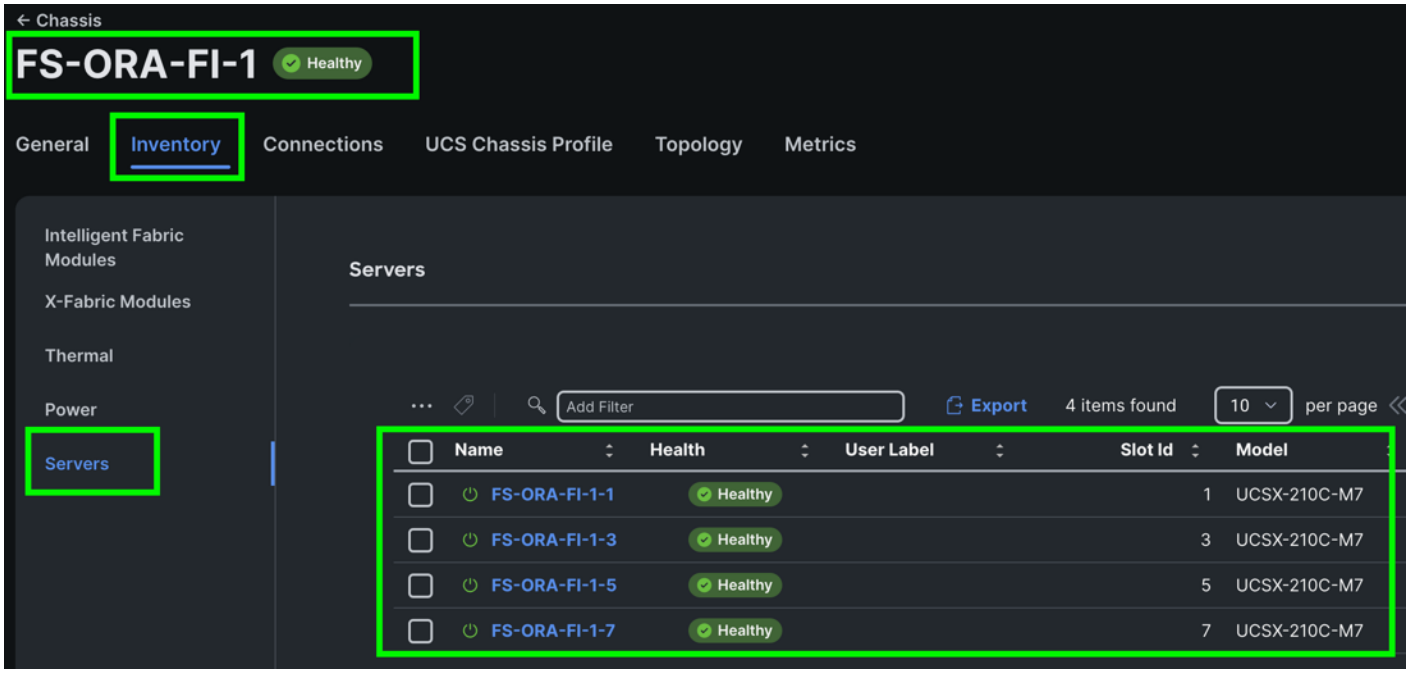


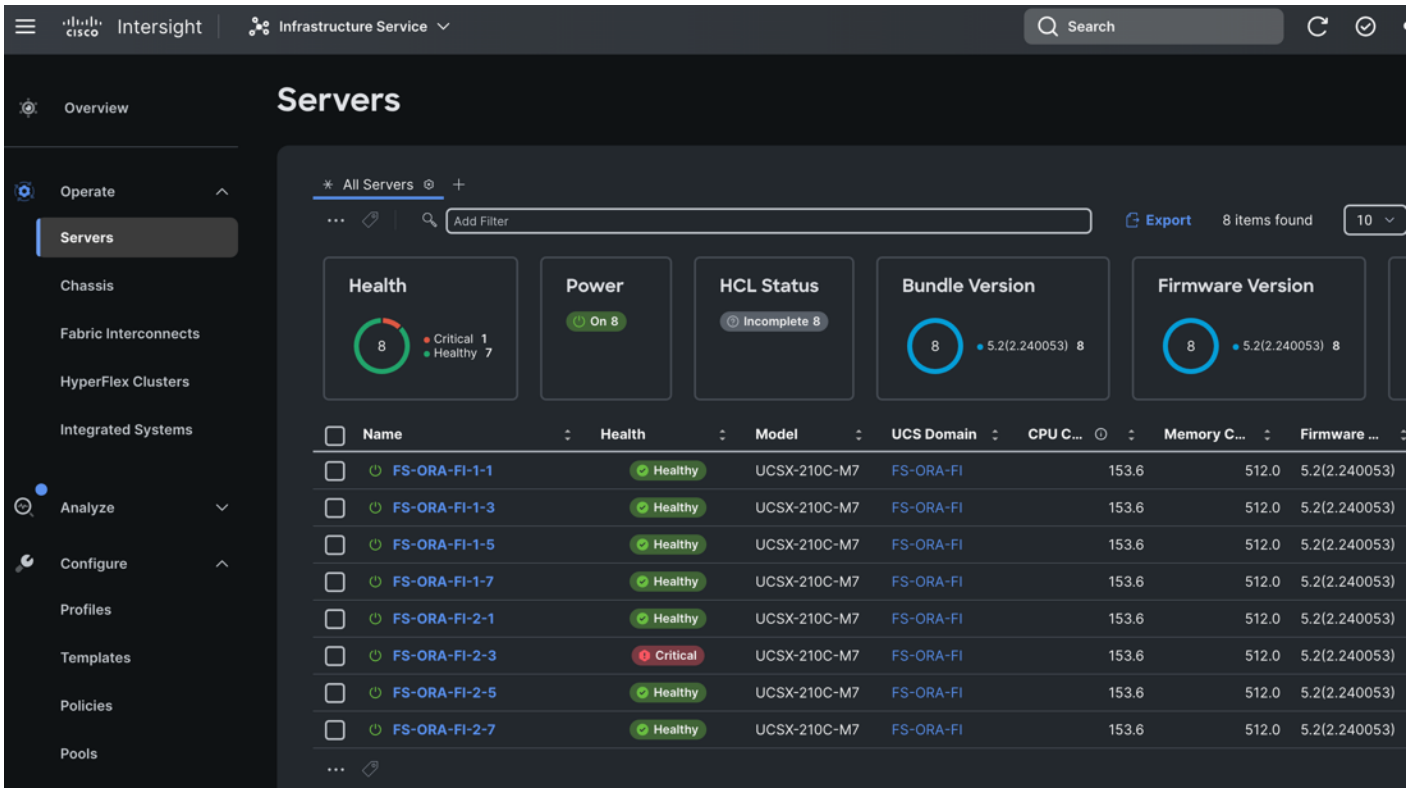

**Step 7.** In the Summary window, review the policies and click Deploy to create Domain Profile.

After the Cisco UCS domain profile has been successfully created and deployed, the policies including the port policies are pushed to the Cisco UCS fabric interconnects. The Cisco UCS domain profile can easily be cloned to install additional Cisco UCS systems. When cloning the Cisco UCS domain profile, the new Cisco UCS domains utilize the existing policies for the consistent deployment of additional Cisco UCS systems at scale.

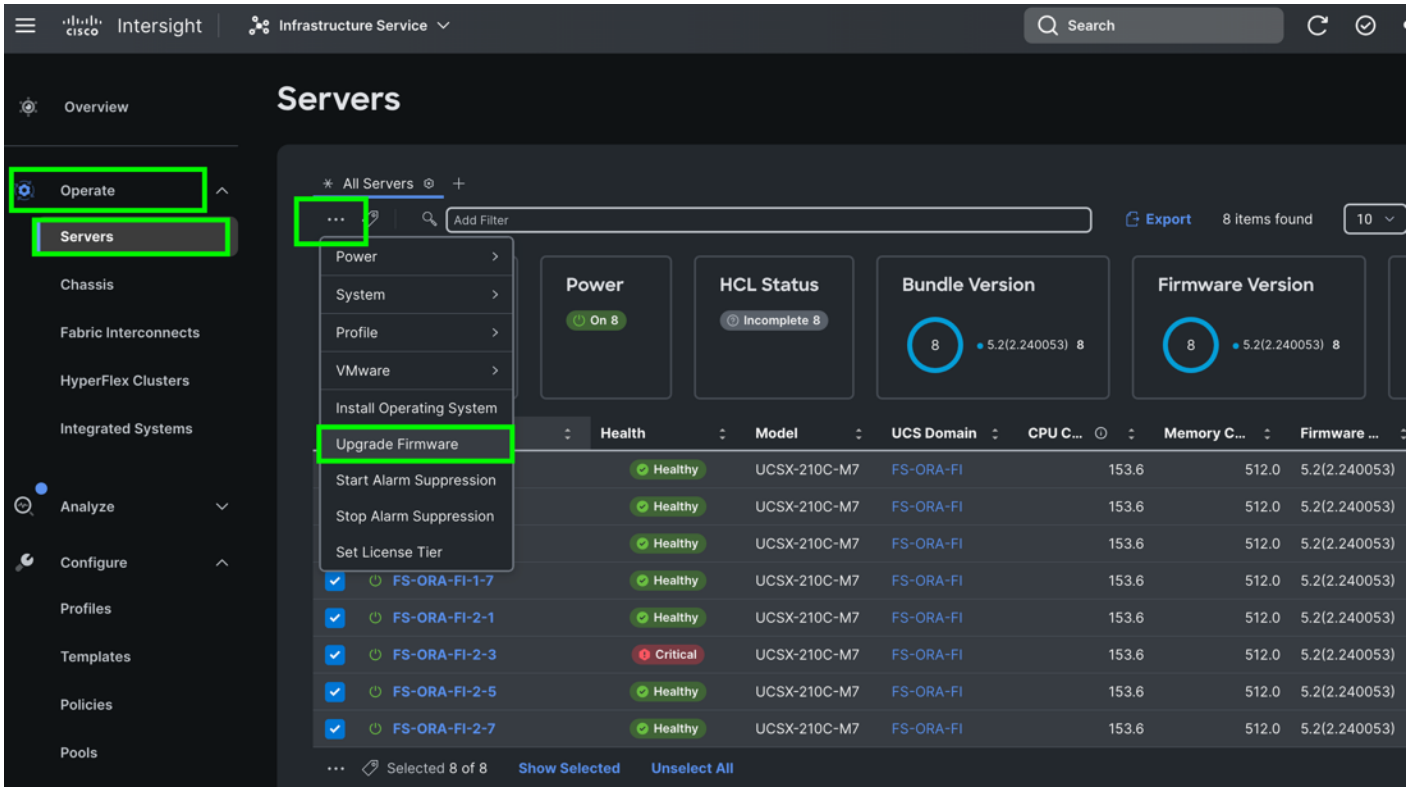
The Cisco UCS X9508 Chassis and Cisco UCS X210c M7 Compute Nodes are automatically discovered when the ports are successfully configured using the domain profile as shown below:



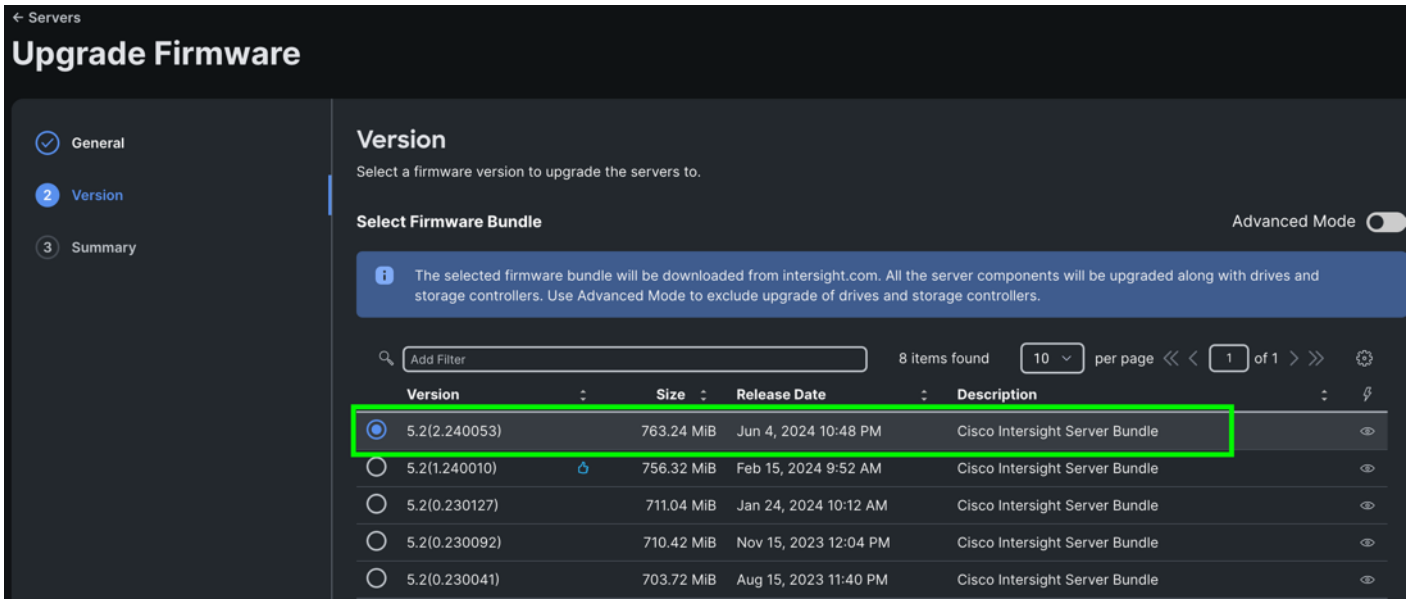



**Step 8.** After discovering the servers successfully, upgrade all server firmware through IMM to the supported release. To do this, check the box for All Servers and then click the ellipses and from the drop-down list, select Upgrade Firmware.




**Step 9.** In the Upgrade Firmware section, select all servers and click Next. In the Version section, for the supported firmware version release select “5.2(2.240053)” and click Next, then click Upgrade to upgrade the firmware on all servers simultaneously.




After the successful firmware upgrade, you can create a server profile template and a server profile for IMM configuration.

## Configure Policies for Server Profile

A server profile enables resource management by simplifying policy alignment and server configuration. The server profile wizard groups the server policies into the following categories to provide a quick summary view of the policies that are attached to a profile:

- Compute Configuration: BIOS, Boot Order, and Virtual Media policies.
- Management Configuration: Certificate Management, IMC Access, IPMI (Intelligent Platform Management Interface) Over LAN, Local User, Serial Over LAN, SNMP (Simple Network Management Protocol), Syslog and Virtual KVM (Keyboard, Video, and Mouse).
- Storage Configuration: SD Card, Storage.
- Network Configuration: LAN connectivity and SAN connectivity policies.

Some of the characteristics of the server profile template for FlashStack are as follows:

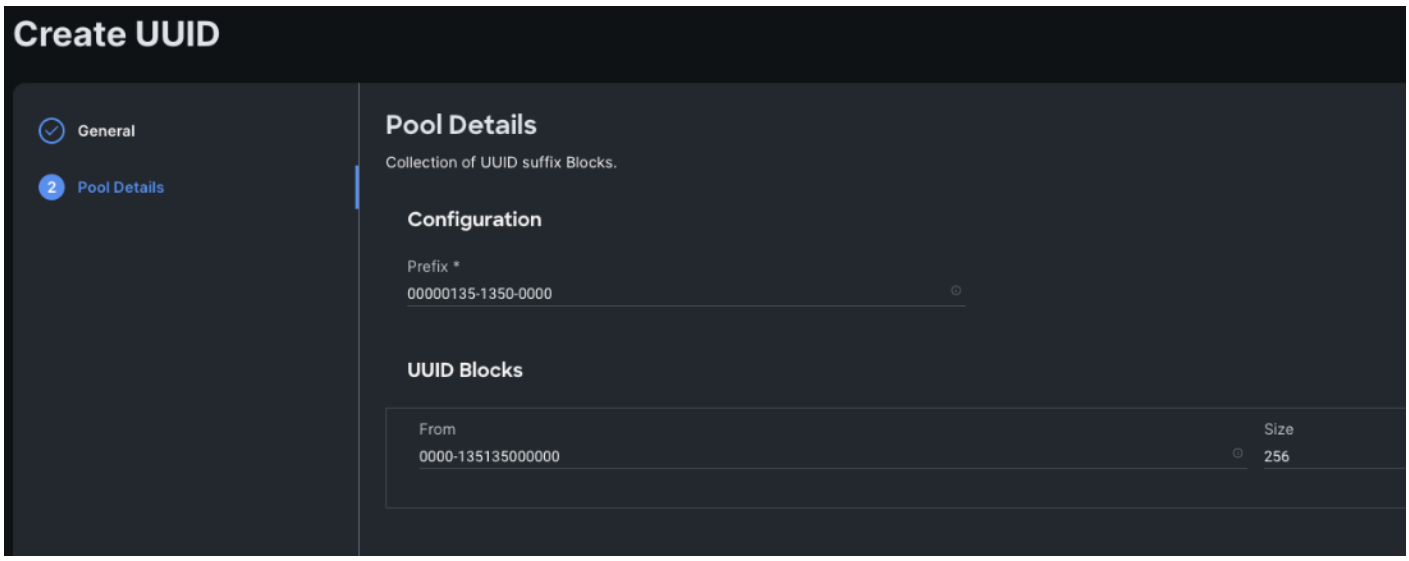
- BIOS policy is created to specify various server parameters in accordance with FlashStack best practices.
- Boot order policy defines virtual media (KVM mapper DVD) and SAN boot through Pure Storage.
- IMC access policy defines the management IP address pool for KVM access.
- LAN connectivity policy is used to create two virtual network interface cards (vNICs) – One vNIC for Server Node Management and Public Network Traffic, second vNIC for Private Server-to-Server Network (Cache Fusion) Traffic Interface for Oracle RAC.
- SAN connectivity policy is used to create total 10 vHBA (2 vHBA for FC SAN Boot and 8 vHBA for NVMe FC Database traffic) per server to boot through FC SAN as well as run NVMe FC traffics on the same server node.

### Procedure 1. Configure UUID Pool

**Step 1.** To create UUID Pool for a Cisco UCS, go to > Infrastructure Service > Configure > Pools > and click Create Pool. Select option UUID.

**Step 2.** In the UUID Pool Create section, for the Organization, select ORA21 and for the Policy name ORA-UUID. Click Next.

**Step 3.** Select Prefix, UUID block and size according to your environment and click Create.




**Procedure 2.** Configure BIOS Policy

**Note:** For more information, see "Performance Tuning Best Practices Guide for Cisco UCS M7 Platforms"

**Note:** For this specific database solution, we created a BIOS policy and used all "Platform Default" values.

**Step 1.** To create BIOS Policy, go to > Infrastructure Service > Configure > Policies > and select Platform type as UCS Server and select on BIOS and click on start.

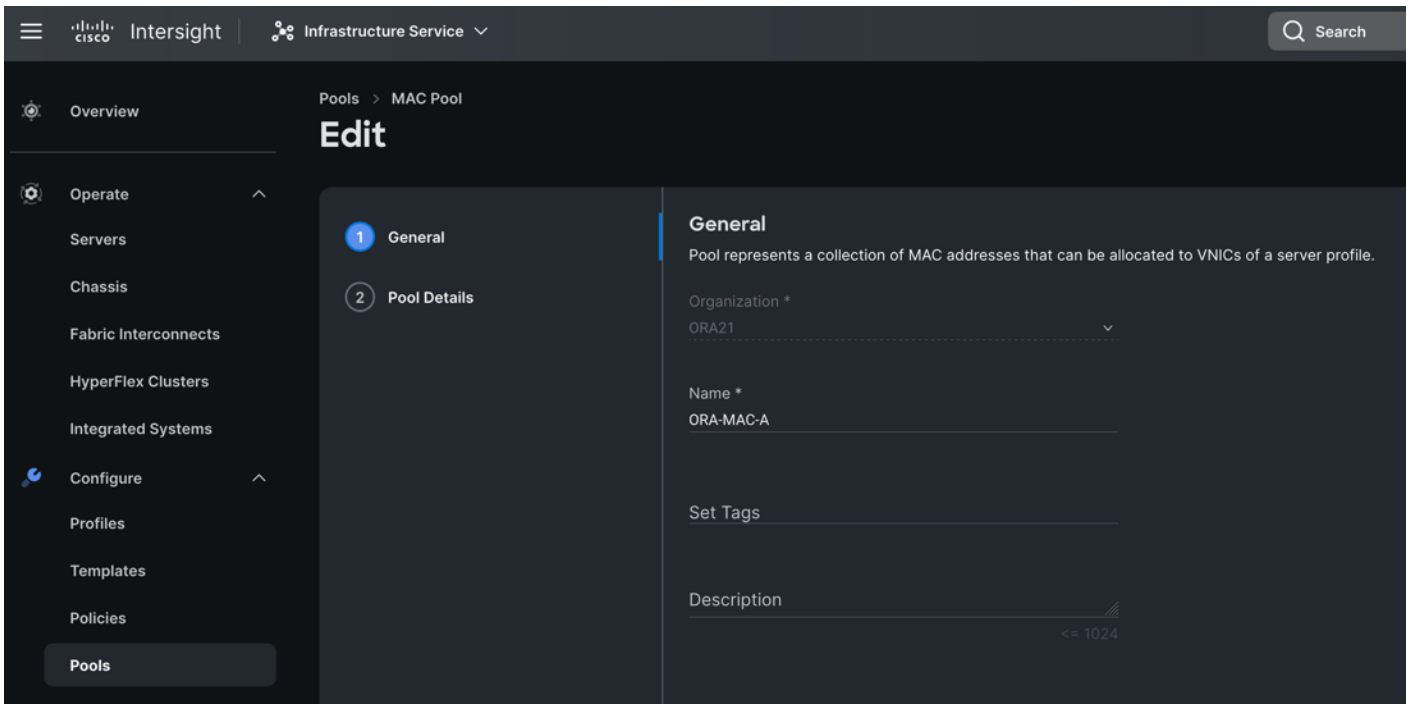
**Step 2.** In the BIOS create general menu, for the Organization, select ORA21 and for the Policy name ORA-BIOS. Click Next

**Step 3.** Click Create to create the platform default BIOS policy.

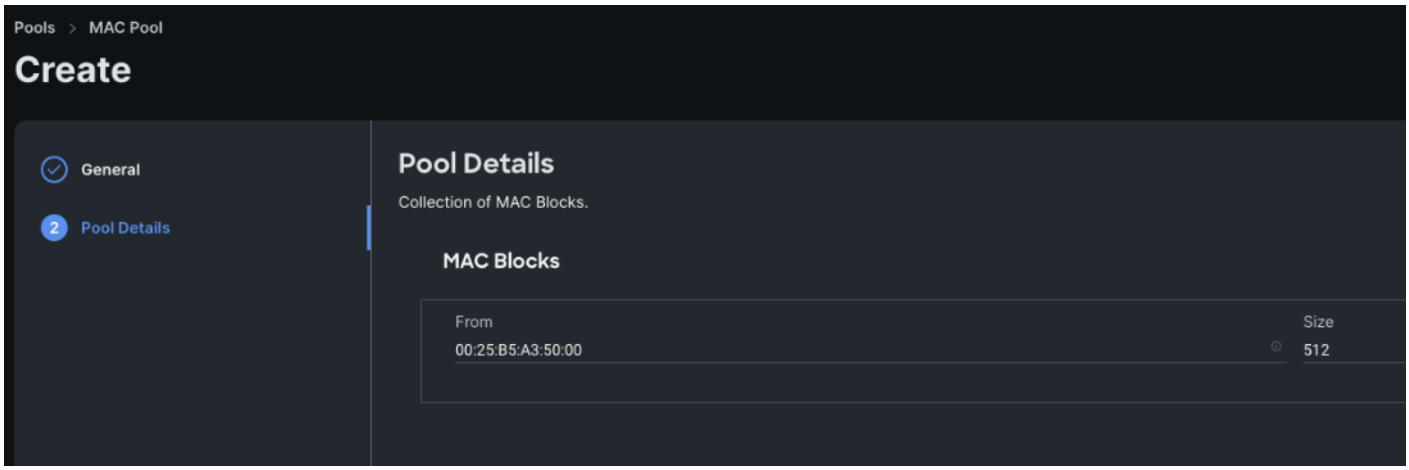
**Procedure 3.** Create MAC Pool

**Step 1.** To configure a MAC Pool for a Cisco UCS Domain profile, go to > Infrastructure Service > Configure > Pools > and click Create Pool. Select option MAC to create MAC Pool.

**Step 2.** In the MAC Pool Create section, for the Organization, select ORA21 and for the Policy name ORA-MAC-A. Click Next.




**Step 3.** Enter the MAC Blocks from and Size of the pool according to your environment and click Create.



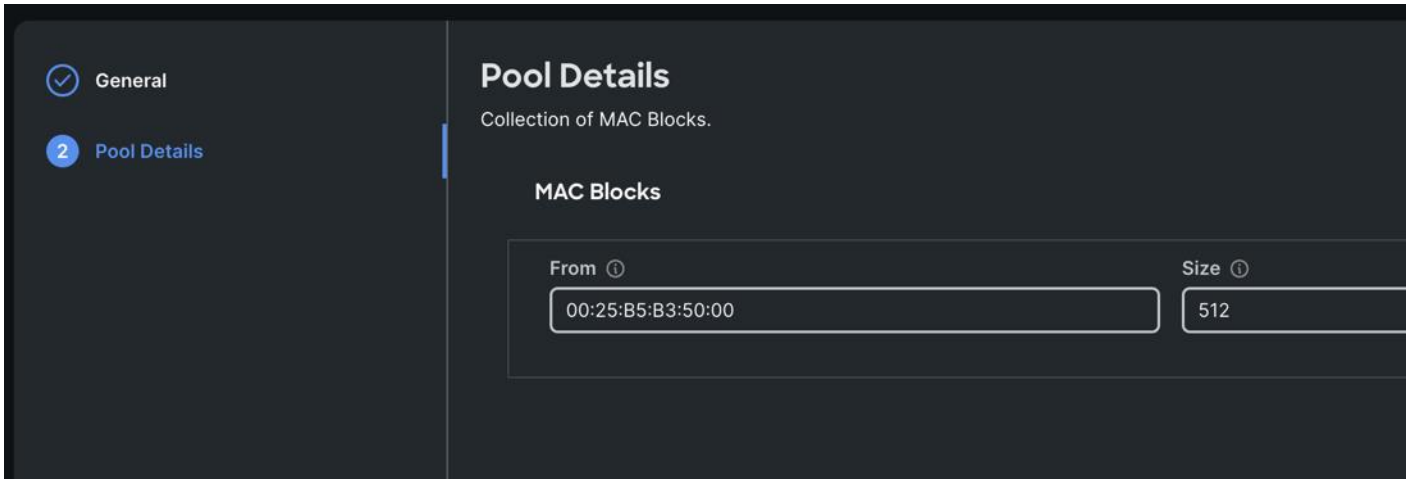

**Note:** For this solution, we configured two MAC Pools. ORA-MAC-A for vNICs MAC Address VLAN 135 (public network traffic) on all servers through FI-A Side. ORA-MAC-B for vNICs MAC Address of VLAN 10 (private network traffic) on all servers through FI-B Side.

**Step 4.** Create a second MAC Pool to provide MAC addresses to all vNICs running on VLAN 10.

**Step 5.** Go to > Infrastructure Service > Configure > Pools > and click Create Pool. Select option MAC to create MAC Pool.

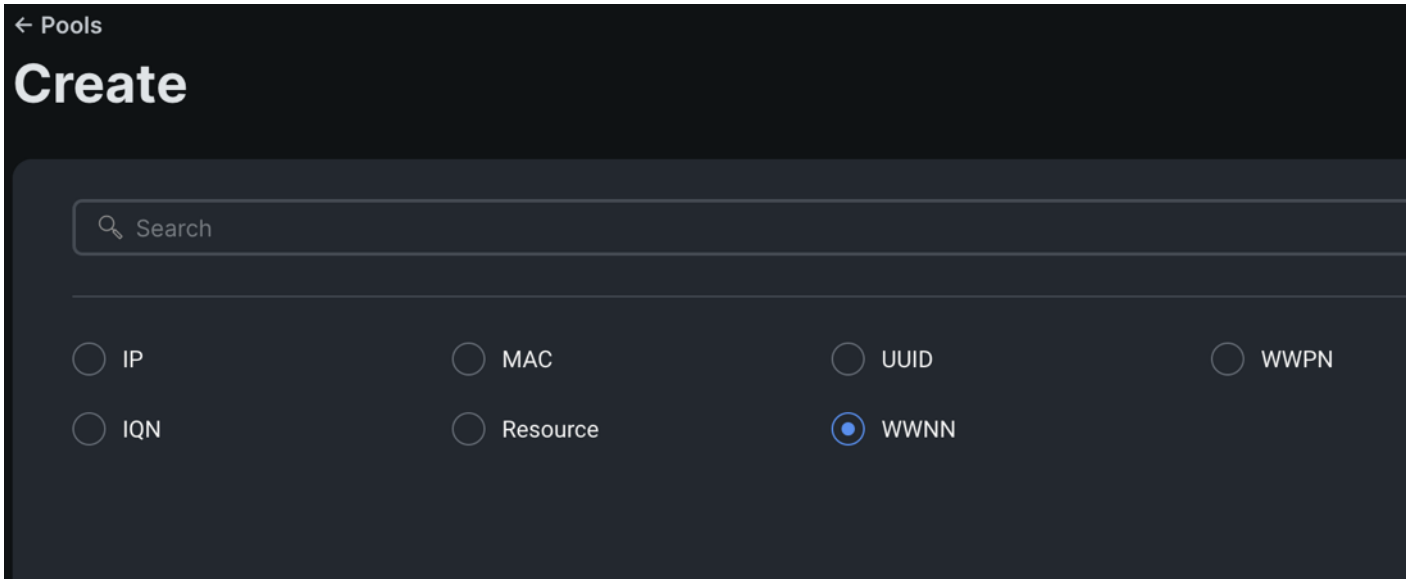
**Step 6.** In the MAC Pool Create section, for the Organization, select ORA21 and for the Policy name “ORA-MAC-B.” Click Next.

**Step 7.** Enter the MAC Blocks from and Size of the pool according to your environment and click Create.




**Procedure 4.** Create WWNN and WWPN Pools

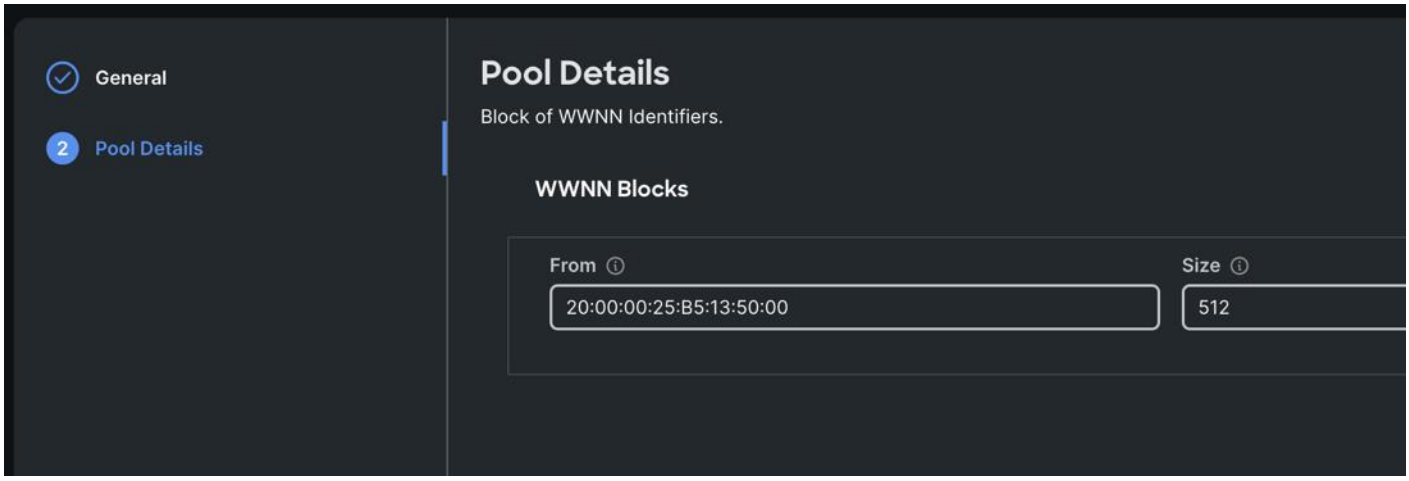
**Step 1.** To create WWNN Pool, go to > Infrastructure Service > Configure > Pools > and click Create Pool. Select option WWNN.




**Step 2.** In the WWNN Pool Create section, for the Organization select ORA21 and name it “WWNN-Pool.” Click Next.

**Step 3.** Add WWNN Block and Size of the pool according to your environment and click Create.

**Step 4.** Click Create to create this policy.

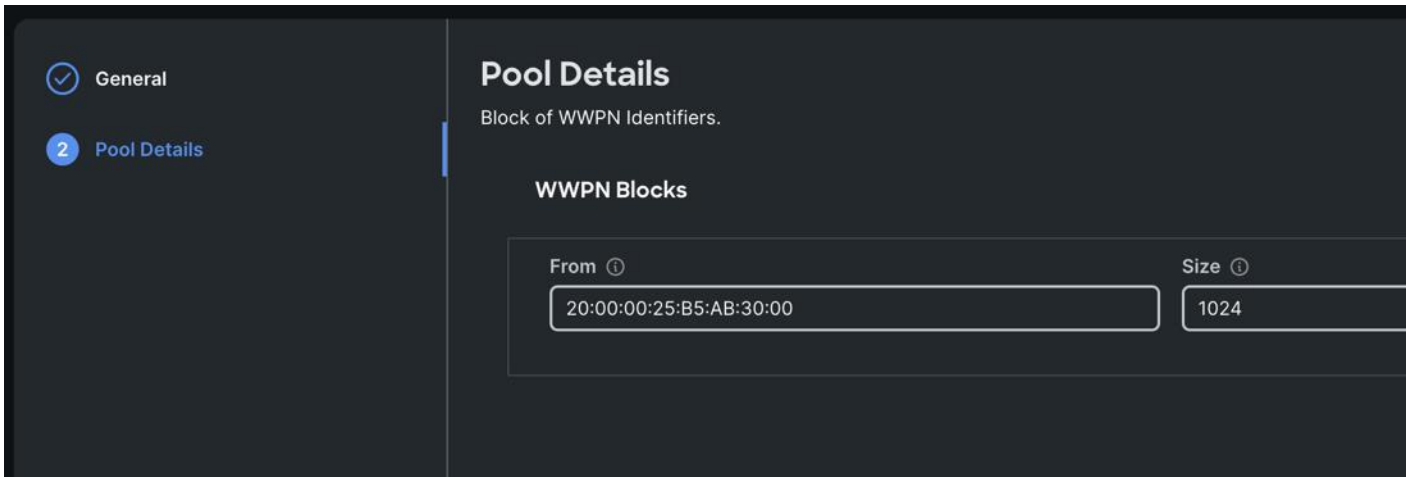



**Step 5.** Create WWPN Pool, go to > Infrastructure Service > Configure > Pools > and click Create Pool. Select option WWPN.

**Step 6.** In the WWPN Pool Create section, for the Organization select ORA21 and name it “WWPN-Pool.” Click Next.

**Step 7.** Add WWPN Block and Size of the pool according to your environment and click Create.

**Step 8.** Click Create to create this policy.



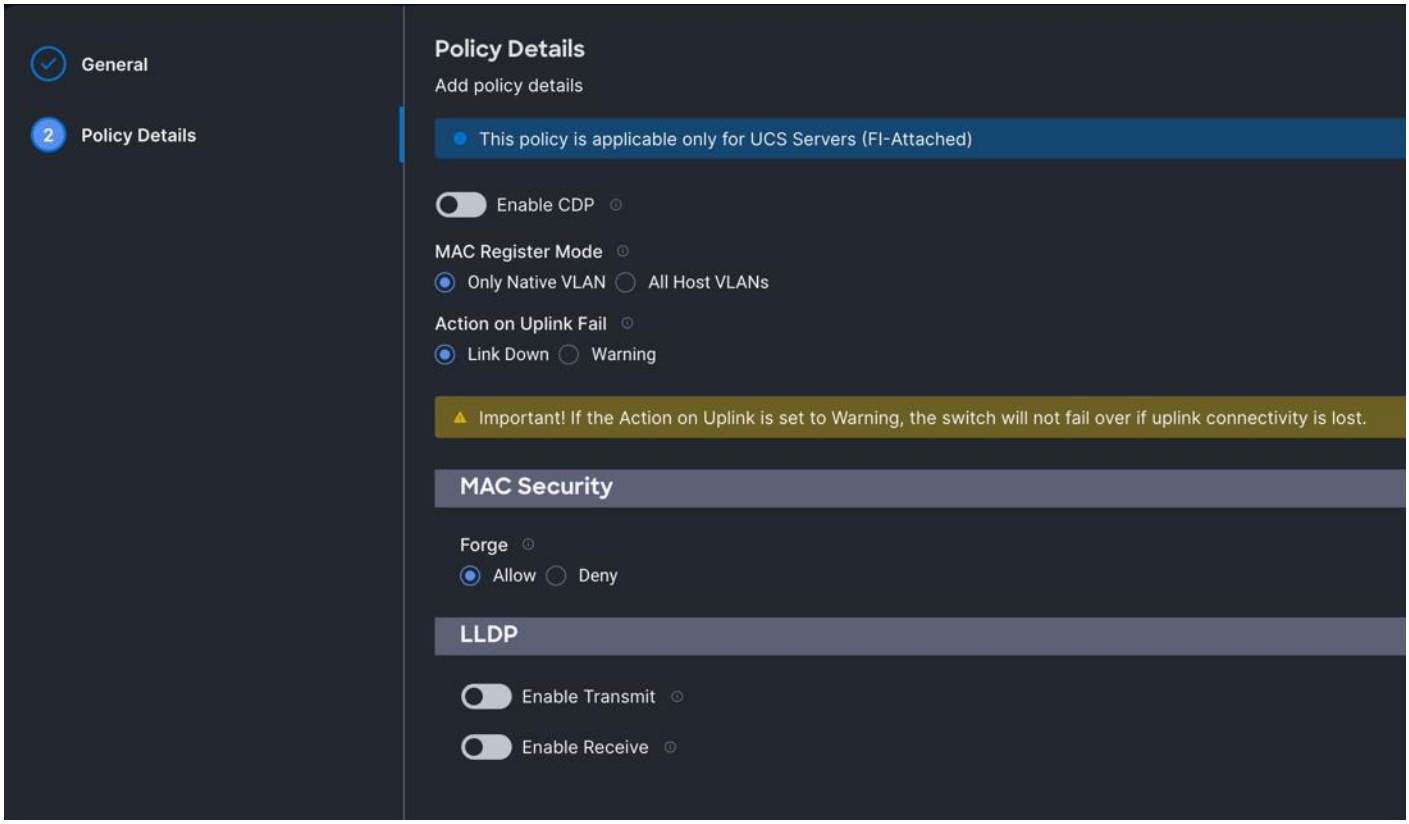

## Procedure 5. Configure Ethernet Network Control Policy

**Step 1.** To configure the Ethernet Network Control Policy for the UCS server profile, go to > Infrastructure Service > Configure > Polices > and click Create Policy.

**Step 2.** For the platform type select UCS Server and for the policy select Ethernet Network Control.

**Step 3.** In the Switch Control Policy Create section, for the Organization select ORA21 and for the policy name enter “ORA-Eth-Network-Control.” Click Next.

**Step 4.** In the Policy Details section, keep the parameter as shown below:




**Step 5.** Click Create to create this policy.

## Procedure 6. Configure Ethernet Network Group Policy

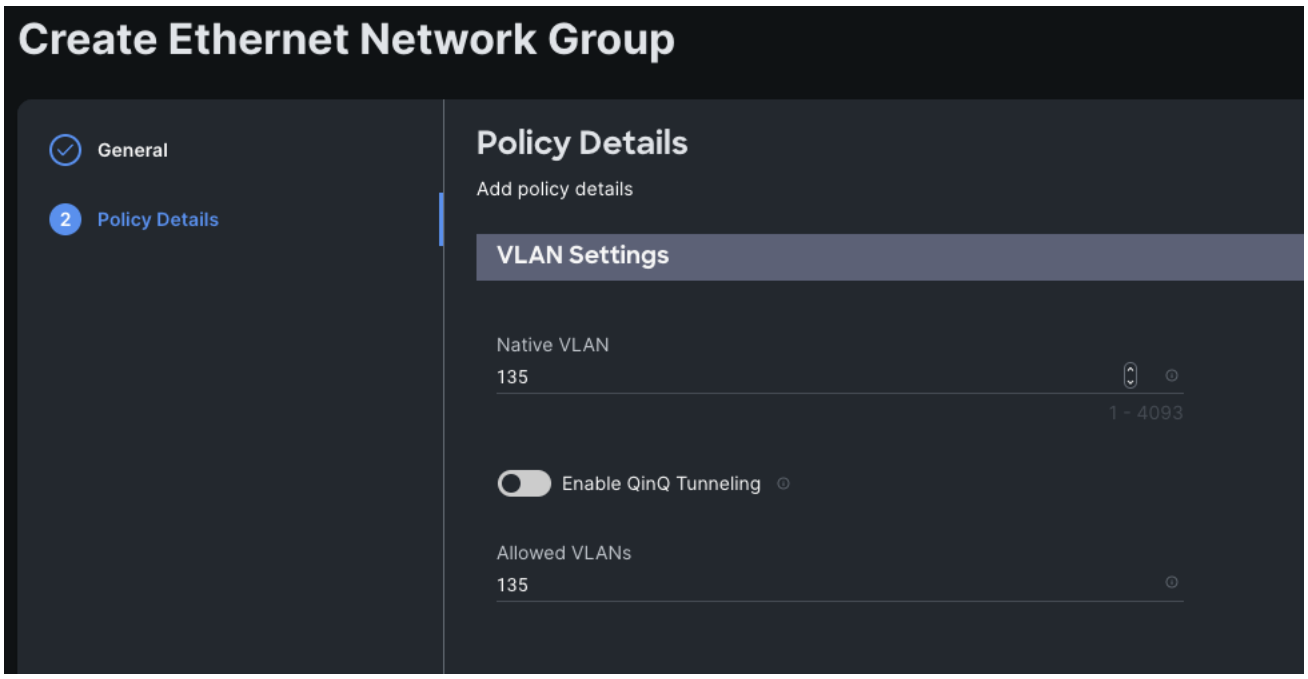
**Note:** We configured two Ethernet Network Groups to allow two different VLAN traffic for this solution.

**Step 1.** To configure the Ethernet Network Group Policy for the UCS server profile, go to > Infrastructure Service > Configure > Polices > and click Create Policy.

**Step 2.** For the platform type select UCS Server and for the policy select Ethernet Network Group.

**Step 3.** In the Switch Control Policy Create section, for the Organization select ORA21 and for the policy name enter “Eth-Network-135.” Click Next.

**Step 4.** In the Policy Details section, for the Allowed VLANs and Native VLAN enter 135 as shown below:




**Step 5.** Click Create to create this policy for VLAN 135.

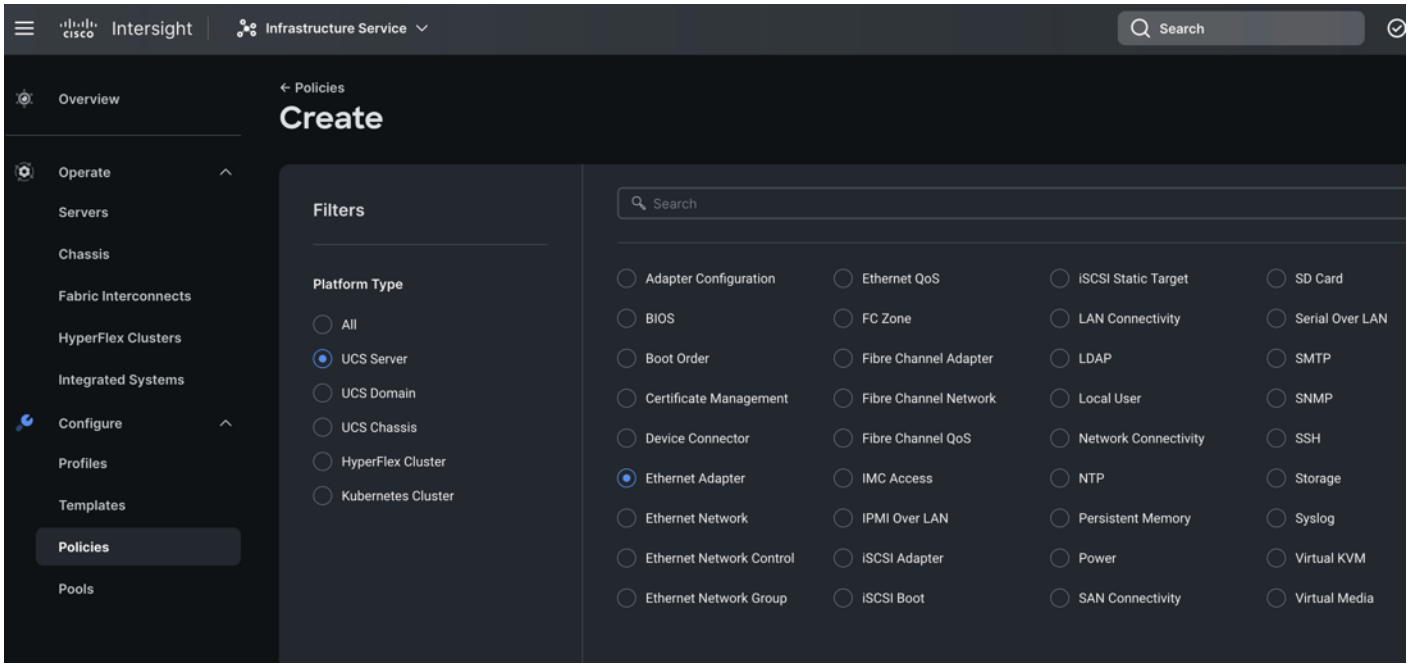
**Step 6.** Create “Eth-Network-10” and add VLAN 10 for the Allowed VLANs and Native VLAN.

**Note:** For this solution, we used these two Ethernet Network Group policies and applied them on different vNICs to carry individual both the VLAN traffic.

## Procedure 7. Configure Ethernet Adapter Policy

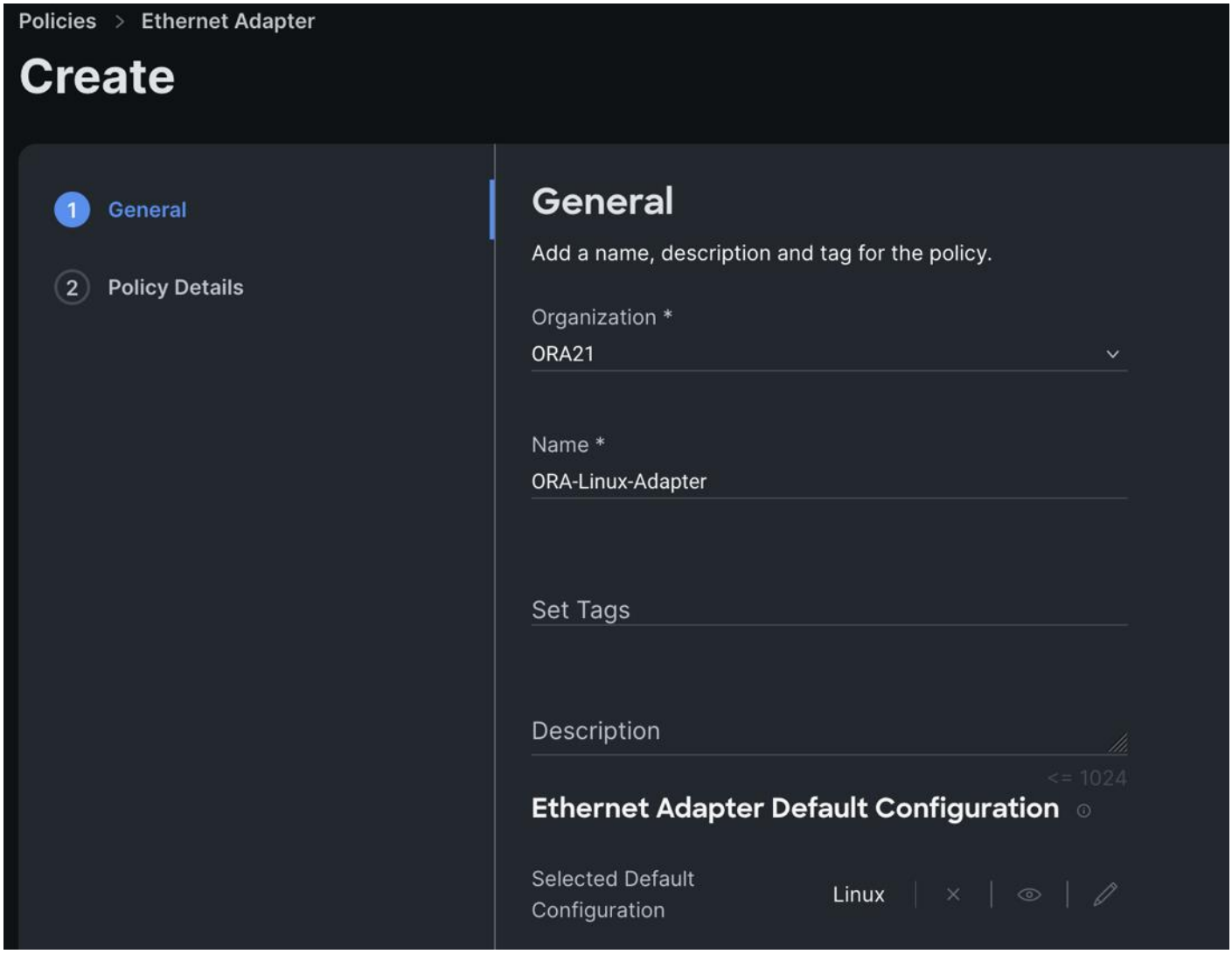
**Step 1.** To configure the Ethernet Adapter Policy for the UCS Server profile, go to > Infrastructure Service > Configure > Polices > and click Create Policy.

**Step 2.** For the platform type select UCS Server and for the policy select Ethernet Adapter.



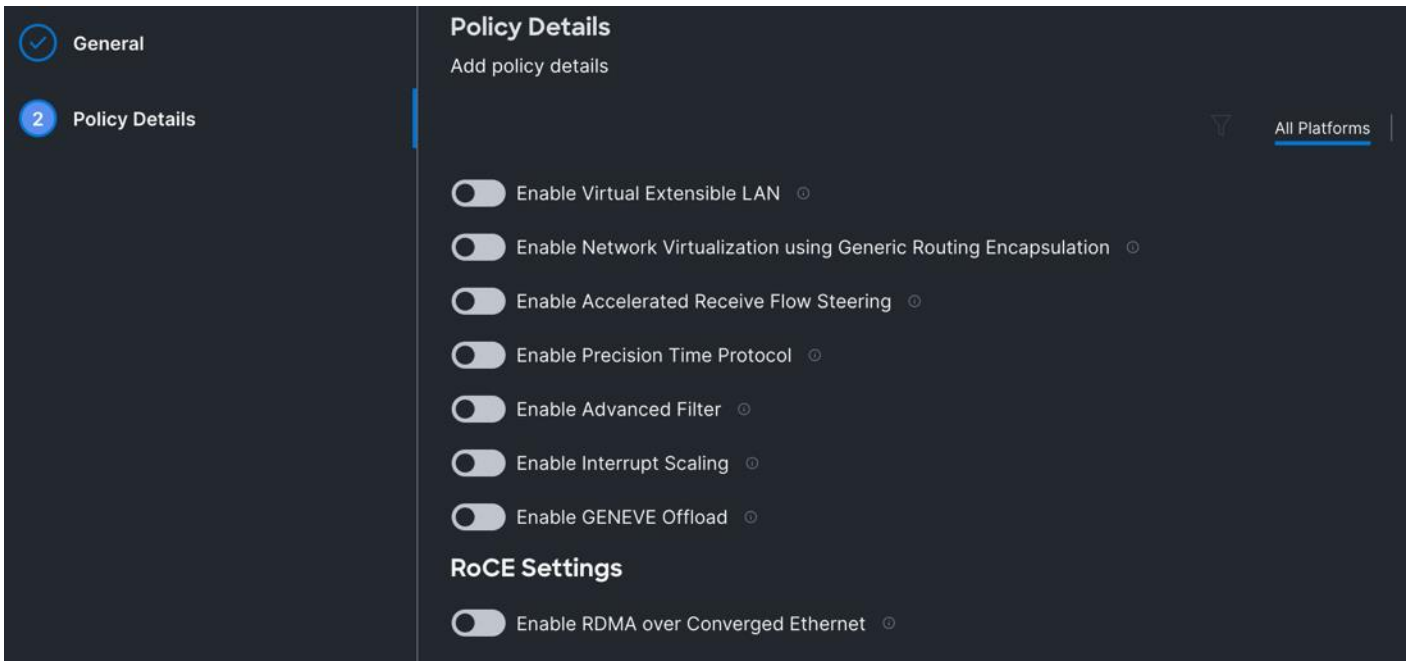

**Step 3.** In the Ethernet Adapter Configuration section, for the Organization select ORA21 and for the policy name enter ORA-Linux-Adapter.

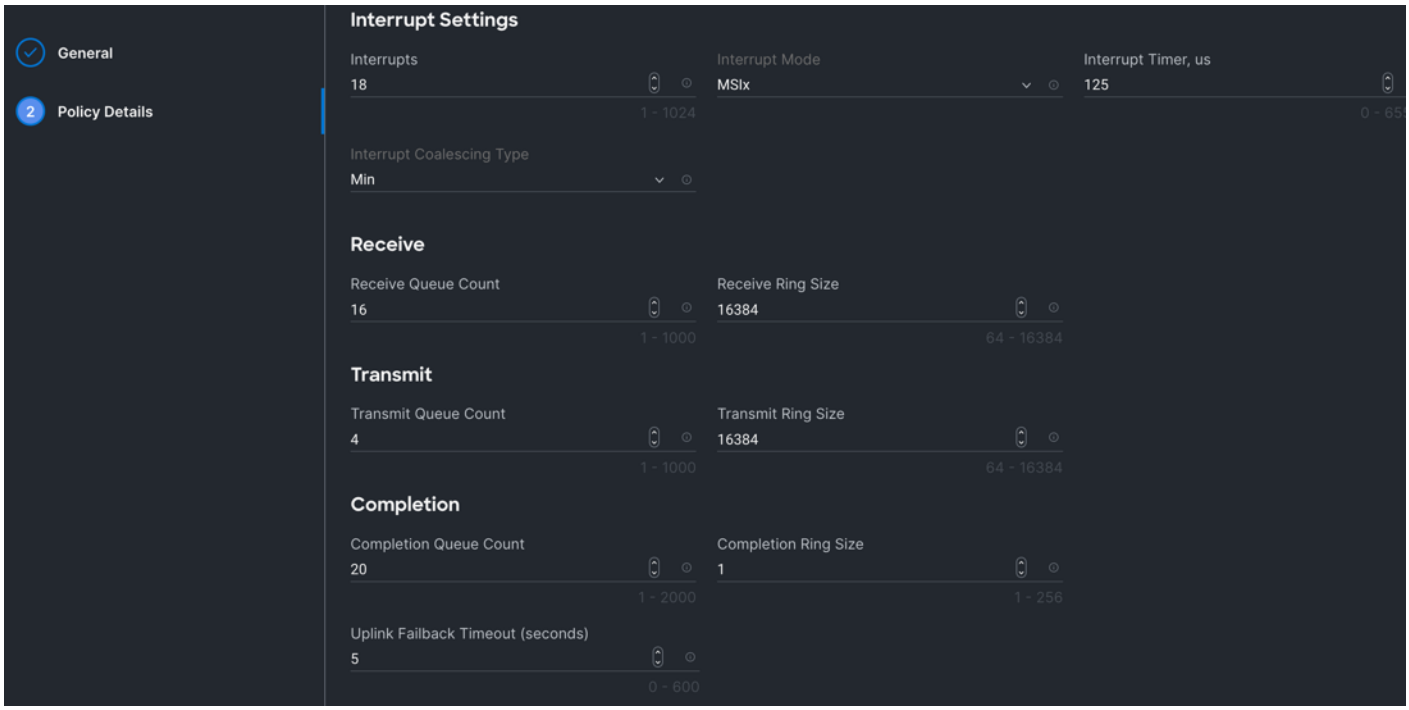
**Step 4.** Select the Default Ethernet Adapter Configuration option and select Linux from the popup menu. Click Next.

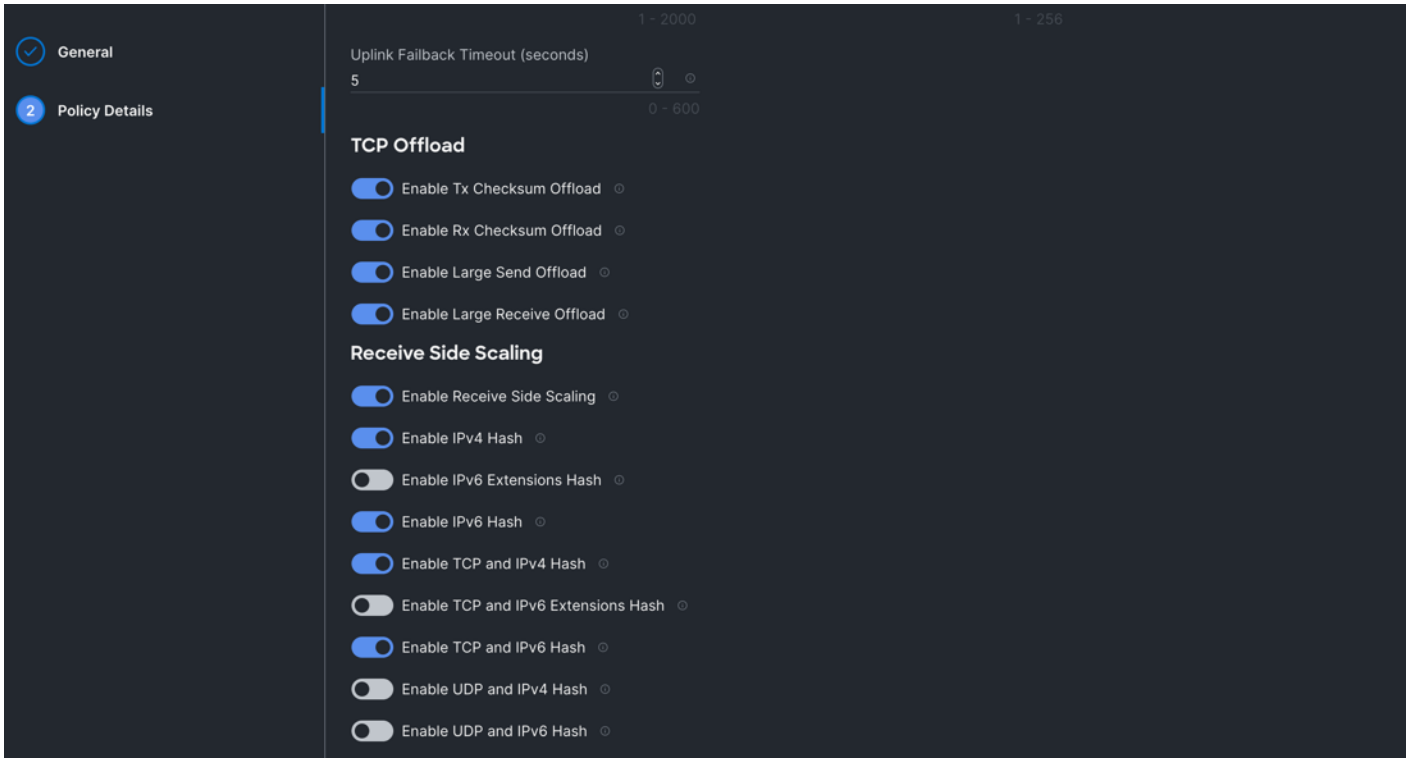



**Step 5.** In the Ethernet Adapter Configuration section, for the Organization select ORA21 and for the policy name enter “ORA-Linux-Adapter.” Select the Default Ethernet Adapter Configuration option and select Linux from the popup menu. Click Next.

**Step 6.** In the Policy Details section, for the recommended performance on the ethernet adapter, keep the “Interrupt Settings” parameter.






**Step 7.** Click Create to create this policy.

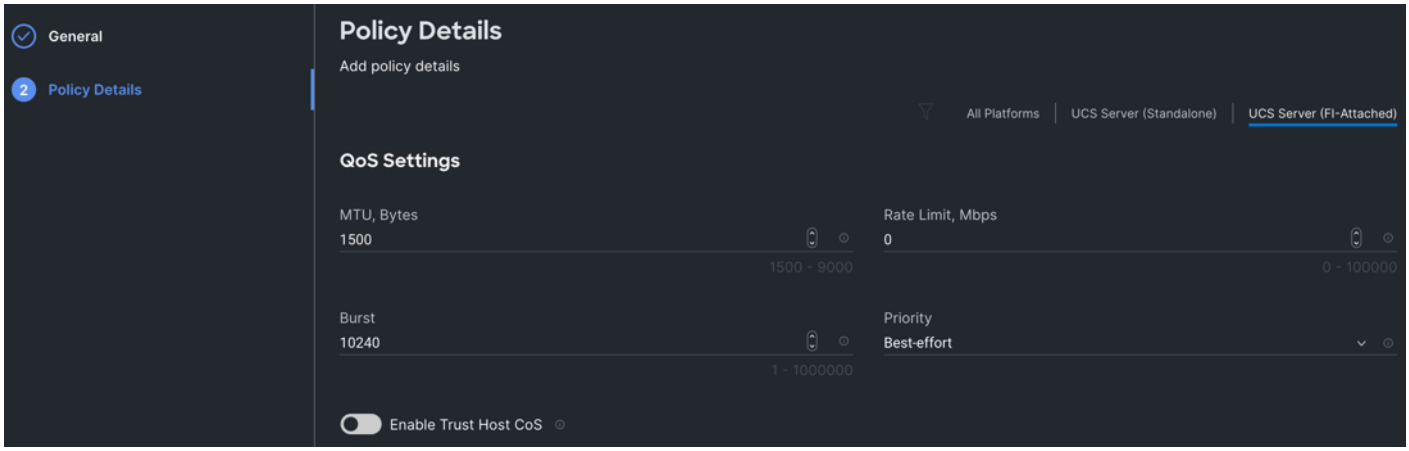
## Procedure 8. Create Ethernet QoS Policy

**Step 1.** To configure the Ethernet QoS Policy for the UCS Server profile, go to > Infrastructure Service > Configure > Polices > and click Create Policy.

**Step 2.** For the platform type select UCS Server and for the policy select Ethernet QoS.

**Step 3.** In the Create Ethernet QoS Configuration section, for the Organization select ORA21 and for the policy name enter “ORA-Eth-QoS-1500” click Next.

**Step 4.** Enter QoS Settings as shown below to configure 1500 MTU for management vNIC.

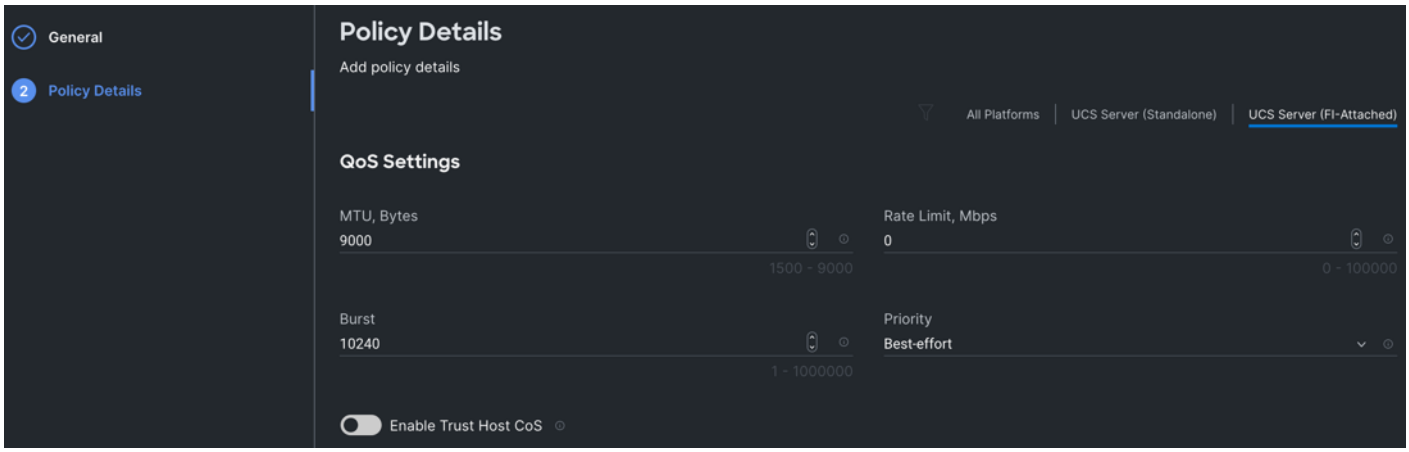



**Step 5.** Click Create to create this policy for vNIC0.

**Step 6.** Create another QoS policy for second vNIC running oracle private network and interconnect traffic.

**Step 7.** In the Create Ethernet QoS Configuration section, for the Organization select ORA21 and for the policy name enter “ORA-Eth-QoS-9000.” Click Next.

**Step 8.** Enter QoS Settings as shown below to configure 9000 MTU for oracle database private interconnect vNIC traffic.




**Step 9.** Click Create to create this policy for vNIC1.

## Procedure 9. Configure LAN Connectivity Policy

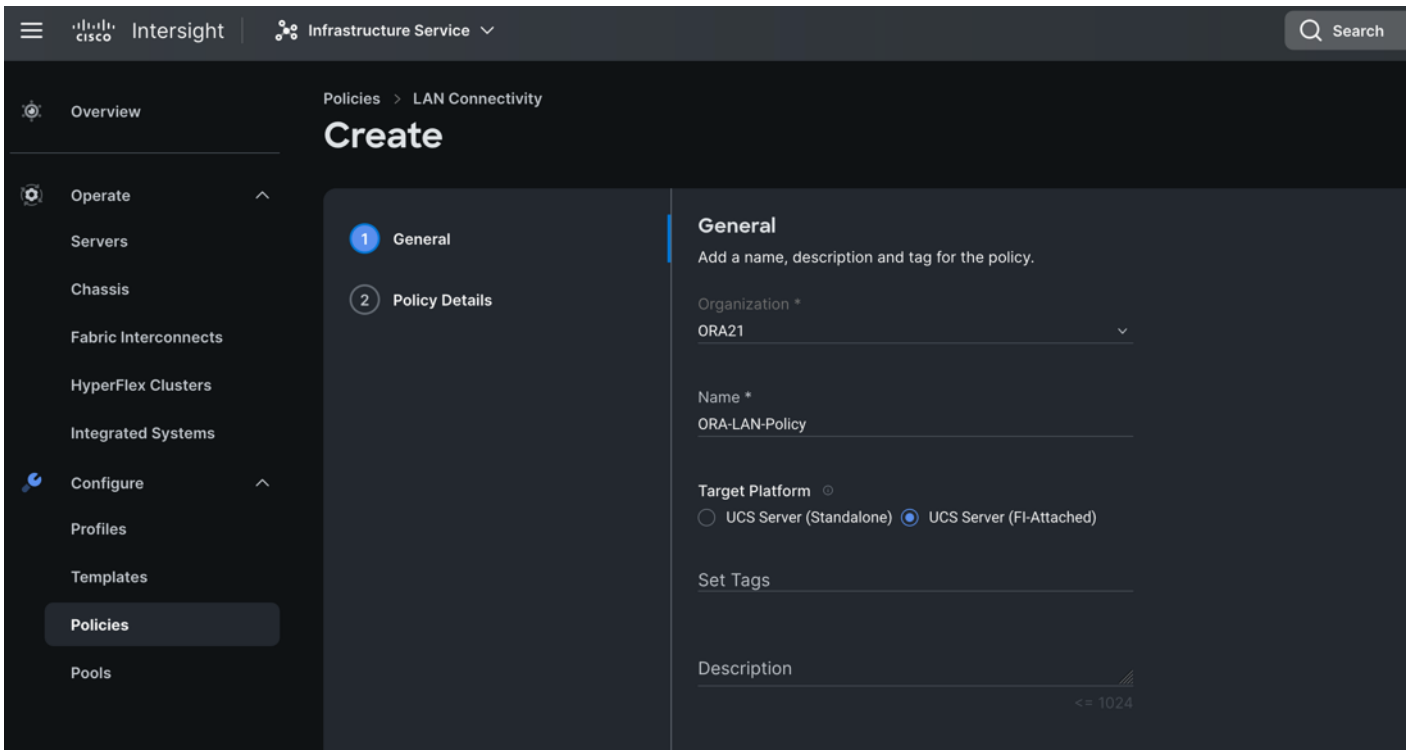
Two vNICs were configured per server as shown in Table 9.

**Table 9. Configured VNICs**

| Name | Switch ID | PCI-Order | MAC Pool | Fail-Over |
|---|---|---|---|---|
| vNIC0 | FI – A | 0 | ORA-MAC-A | Enabled |
| vNIC1 | FI – B | 1 | ORA-MAC-B | Enabled |

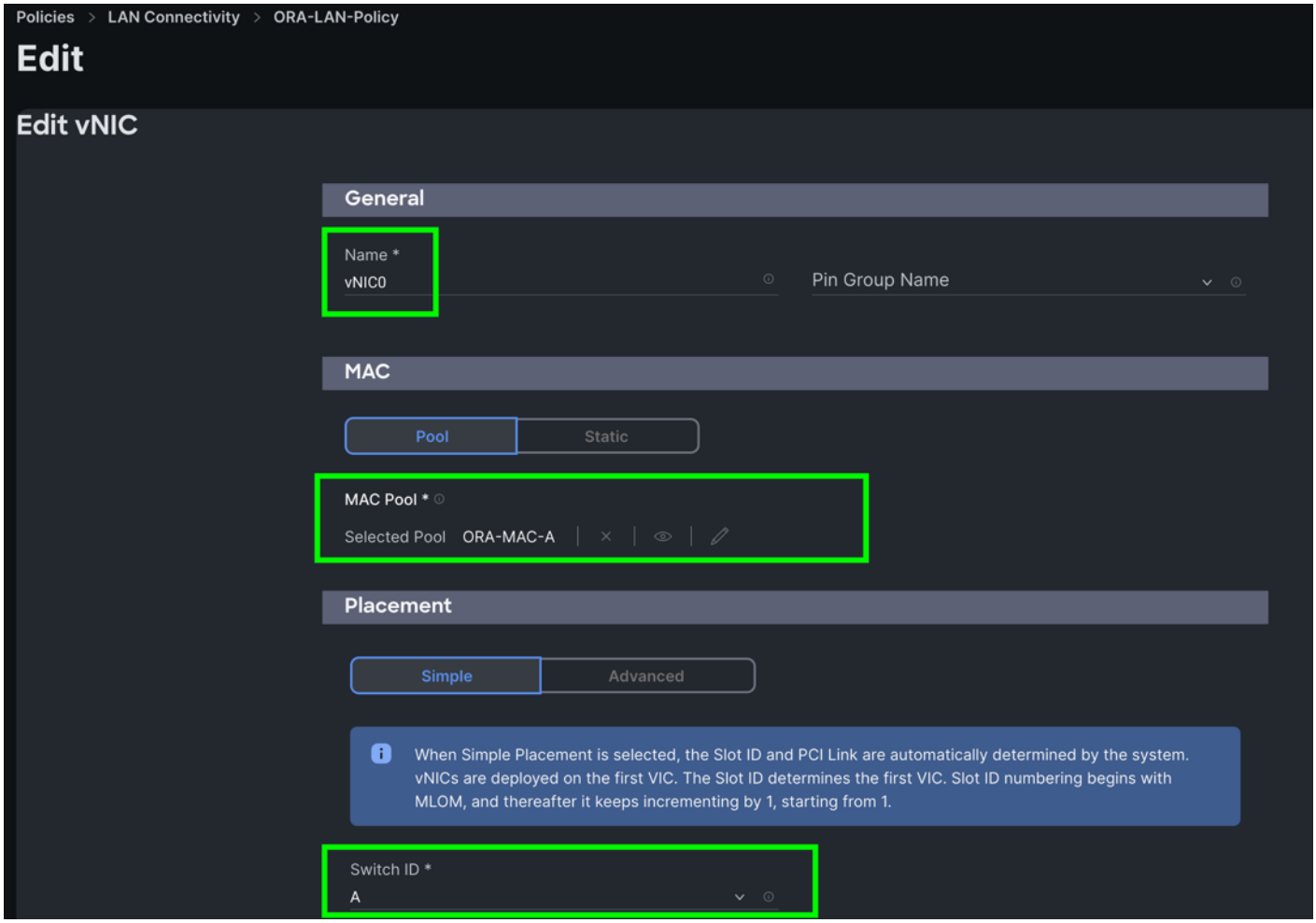
**Step 1.** Go to > Infrastructure Service > Configure > Polices > and click Create Policy. For the platform type select “UCS Server” and for the policy select “LAN Connectivity.”

**Step 2.** In the LAN Connectivity Policy Create section, for the Organization select ORA21,for the policy name enter “ORA-LAN-Policy” and for the Target Platform select UCS Server (FI-Attached). Click Next.

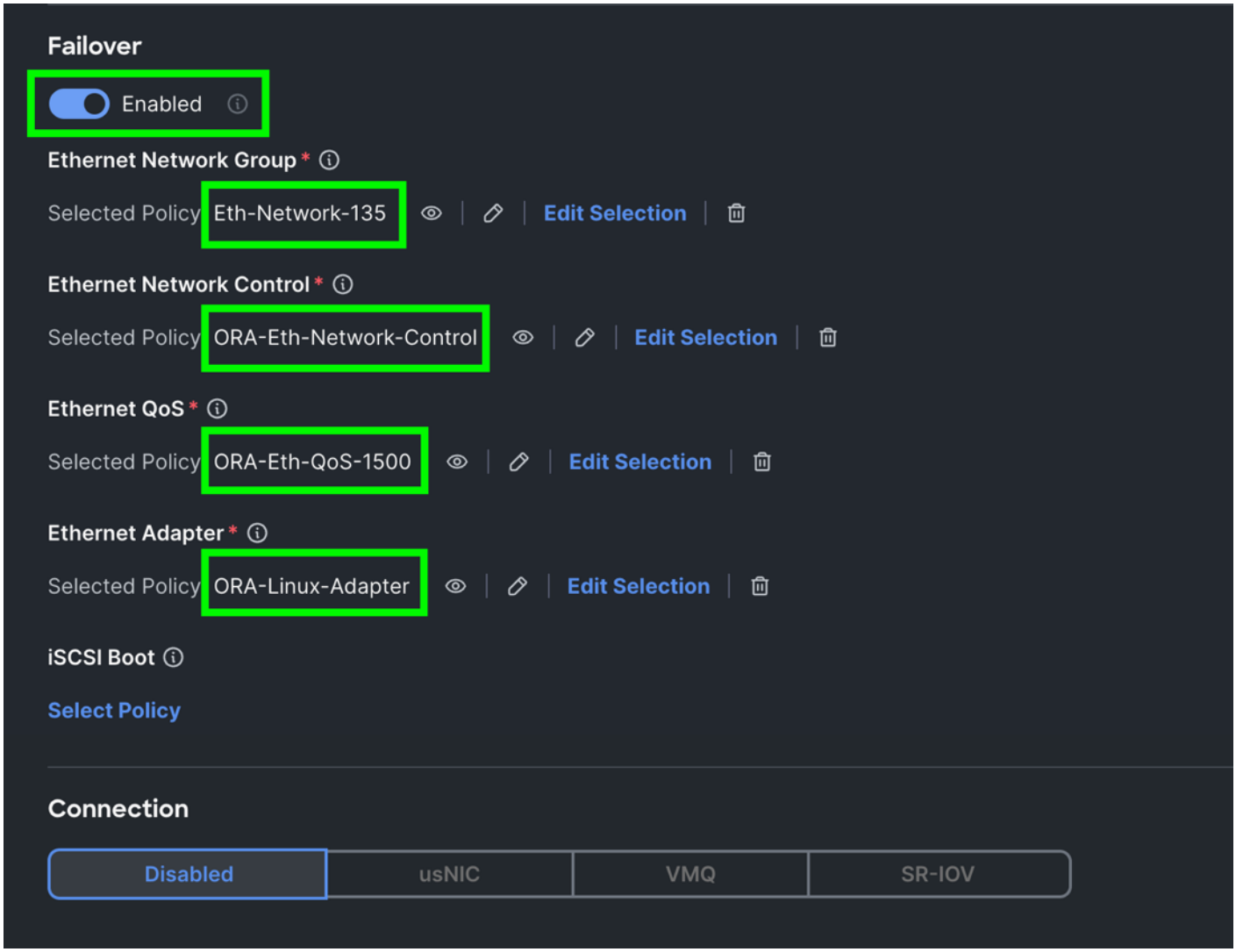



**Step 3.** In the Policy Details section, click Add vNIC. In the Add vNIC section, for the first vNIC enter vNIC0. In the Edit vNIC section, for the vNIC name enter " vNIC0" and for the MAC Pool select ORA-MAC-A.

**Step 4.** In the Placement option, select Simple and for the Switch ID select A as shown below:



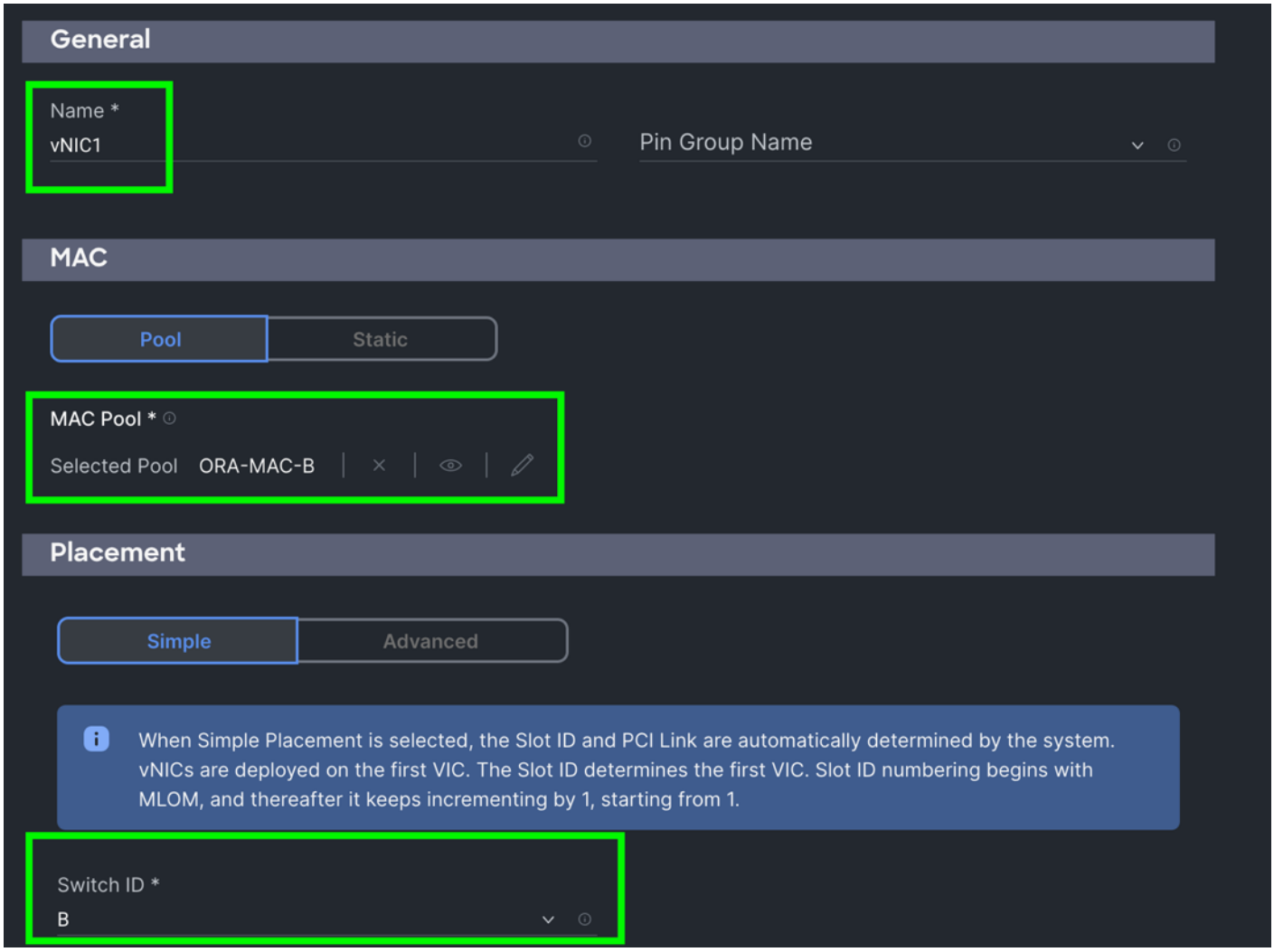

**Step 5.** For Failover select Enable for this vNIC configuration. This enables the vNIC to failover to another FI.




**Step 6.** For the Ethernet Network Group Policy, select Eth-Network-135. For the Ethernet Network Control Policy select ORA-Eth-Network-Control. For Ethernet QoS, select ORA-Eth-QoS-1500, and for the Ethernet Adapter, select ORA-Linux-Adapter. Click Add to add vNIC0 to this policy.

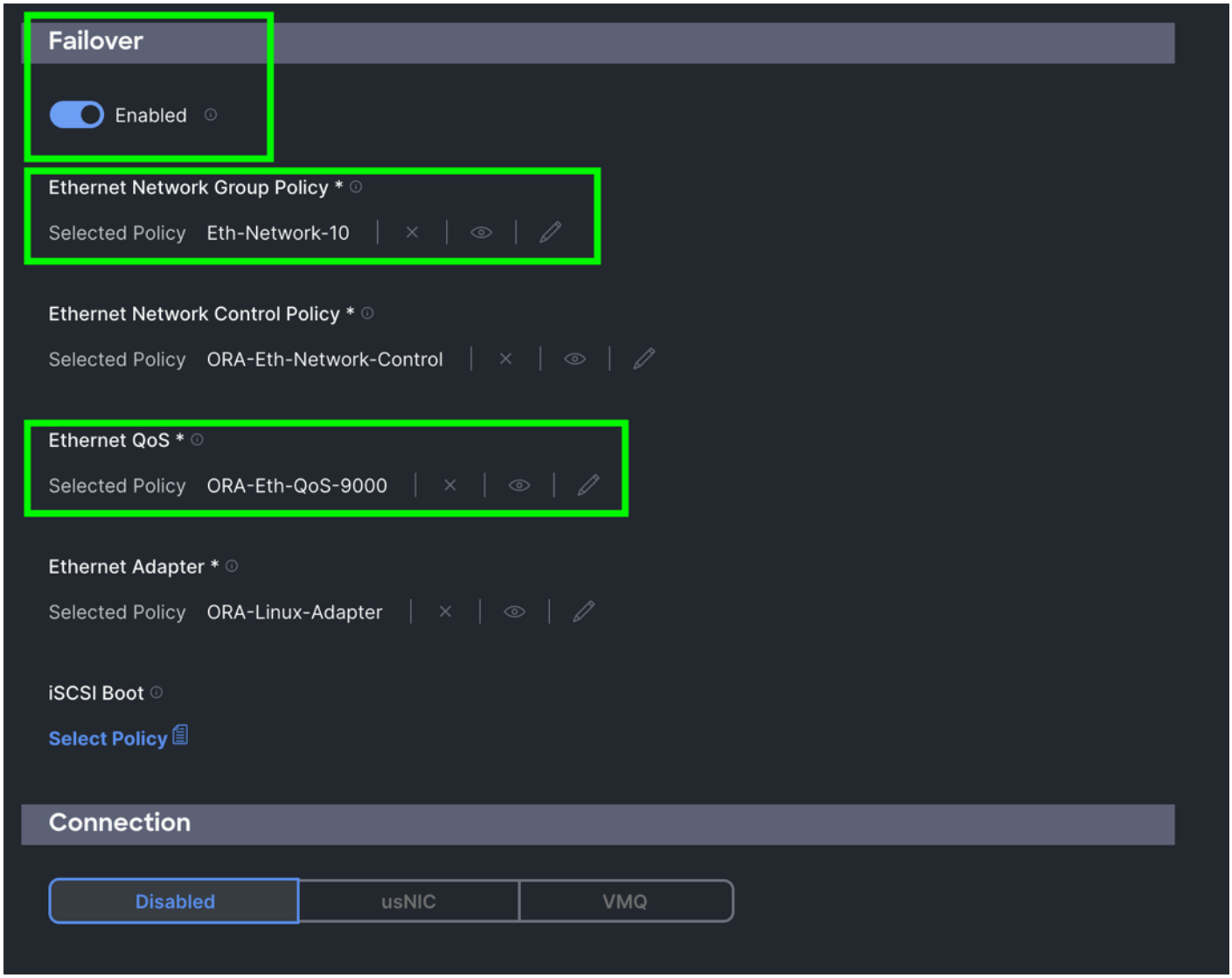
**Step 7.** Add a second vNIC. For the name enter "vNIC1" and for the MAC Pool select ORA-MAC-B.

**Step 8.** In the Placement option, select Simple and for the Switch ID select B as shown below:



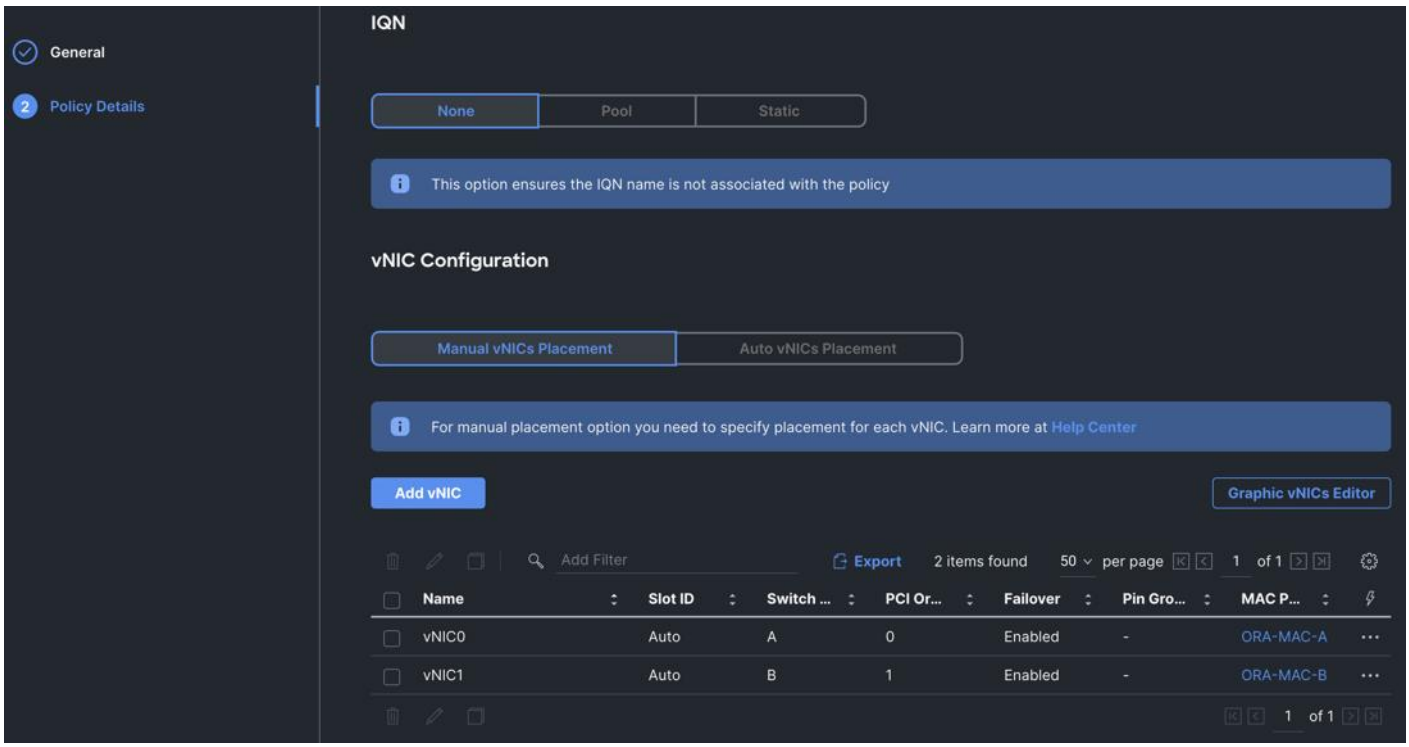

**Step 9.** For Failover select Enable for this vNIC configuration. This enables the vNIC to failover to another FI.

**Step 10.** For the Ethernet Network Group Policy, select Eth-Network-10. For the Ethernet Network Control Policy, select ORA-Eth-Network-Control. For the Ethernet QoS, select ORA-Eth-QoS-9000, and for the Ethernet Adapter, select ORA-Linux-Adapter. Click Add to add vNIC0 to this policy.




**Step 11.** Click Add to add vNIC1 into this policy.

**Step 12.** After adding these two vNICs, review and make sure the Switch ID, PCI Order, Failover Enabled, and MAC Pool are as shown below:




**Step 13.** Click Create to create this policy.

## **Procedure 10.** Create Fibre Channel Network Policy

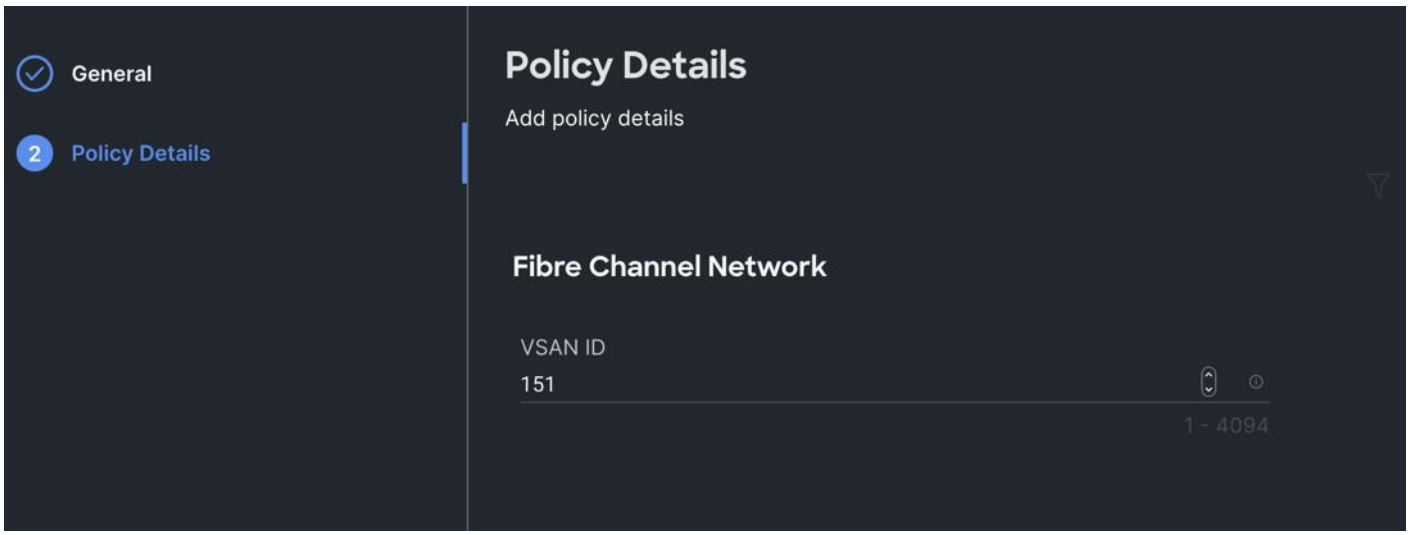
**Step 1.** To configure the Fibre Channel Network Policy for the UCS Server profile, go to > Infrastructure Service > Configure > Polices > and click Create Policy.

**Step 2.** For the platform type select UCS Server and for the policy select Fibre Channel Network.

**Note:** For this solution, we configured two Fibre Channel network policy as “ORA-FC-Network-151” and “ORA-FC-Network-152” to carry two VSAN traffic 151 and 152 on each of the Fabric Interconnect.

**Step 3.** In the Create Fibre Channel Network Configuration section, for the Organization select ORA21 and for the policy name enter “ORA-FC-Network-151.” Click Next.

**Step 4.** For the VSAN ID enter 151 as shown below:



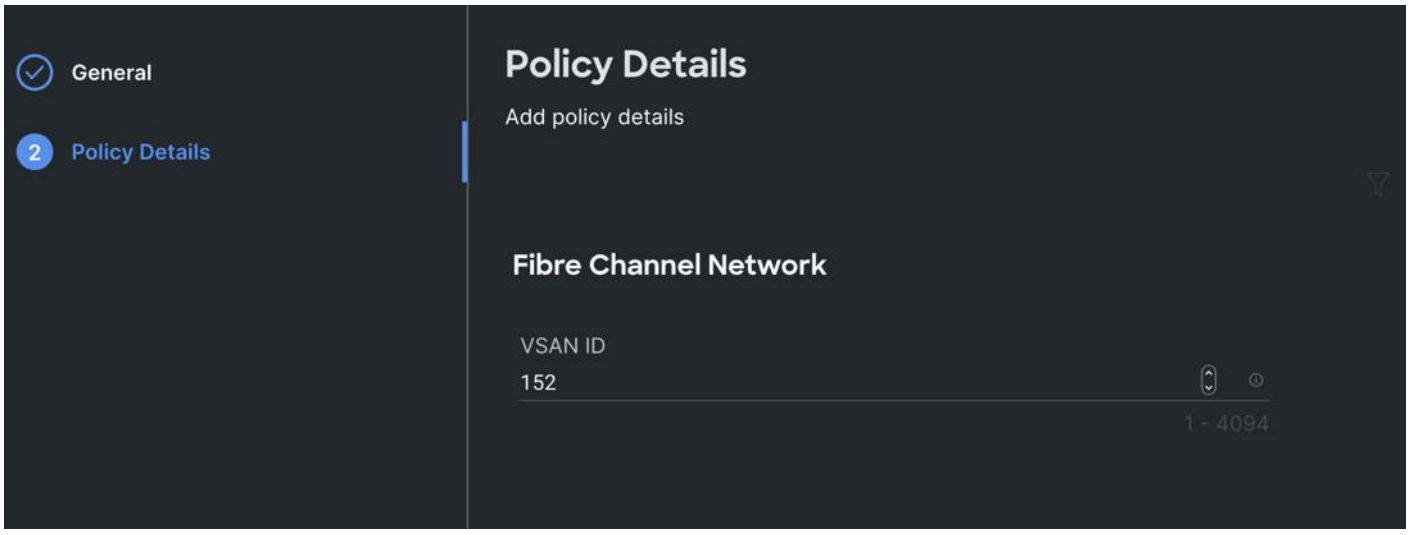

**Step 5.** Click Create to create this policy for VSAN 151.

**Step 6.** Create another Fibre Channel Network Policy for the UCS Server profile, go to > Infrastructure Service > Configure > Polices > and click Create Policy.

**Step 7.** For the platform type select UCS Server and for the policy select Fibre Channel Network.

**Step 8.** In the Create Fibre Channel Network Configuration section, for the Organization select ORA21 and for the policy name enter “ORA-FC-Network-152.” Click Next.

**Step 9.** For the VSAN ID enter 152 as shown below:




**Step 10.** Click Create to create this policy for VSAN 152.

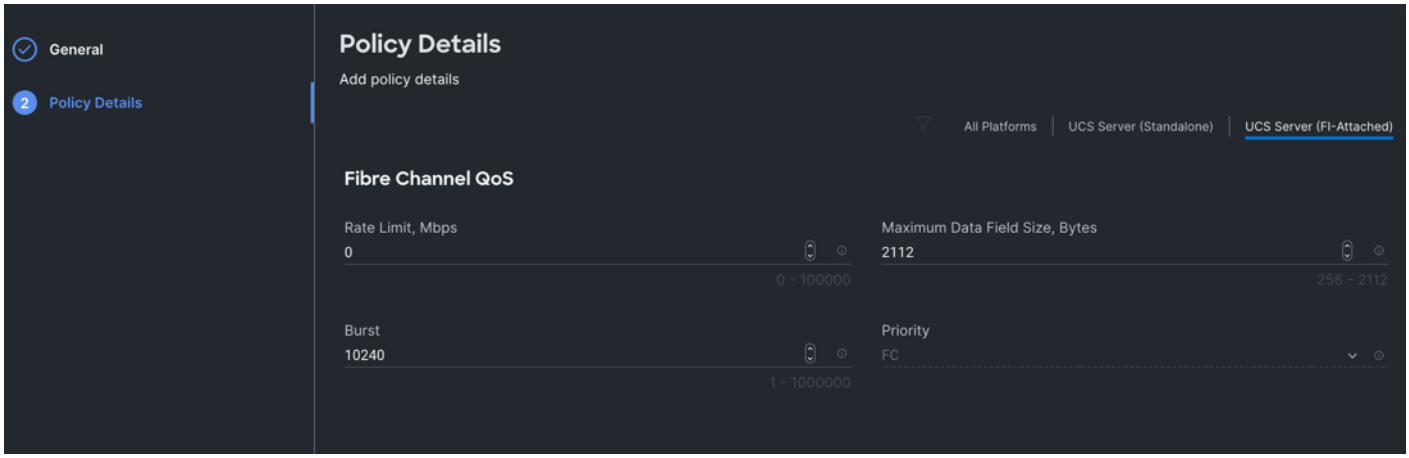
**Procedure 11.** Create Fibre Chanel QoS Policy

**Step 1.** To configure the Fibre Channel QoS Policy for the UCS Server profile, go to > Infrastructure Service > Configure > Polices > and click Create Policy.

**Step 2.** For the platform type select UCS Server and for the policy select Fibre Channel QoS.

**Step 3.** In the Create Fibre Channel QoS Configuration section, for the Organization select ORA21 and for the policy name enter ORA-FC-QoS click Next.

**Step 4.** Enter QoS Settings as shown below to configure QoS for Fibre Channel for vHBA0:




**Step 5.** Click Create to create this policy for Fibre Channel QoS.

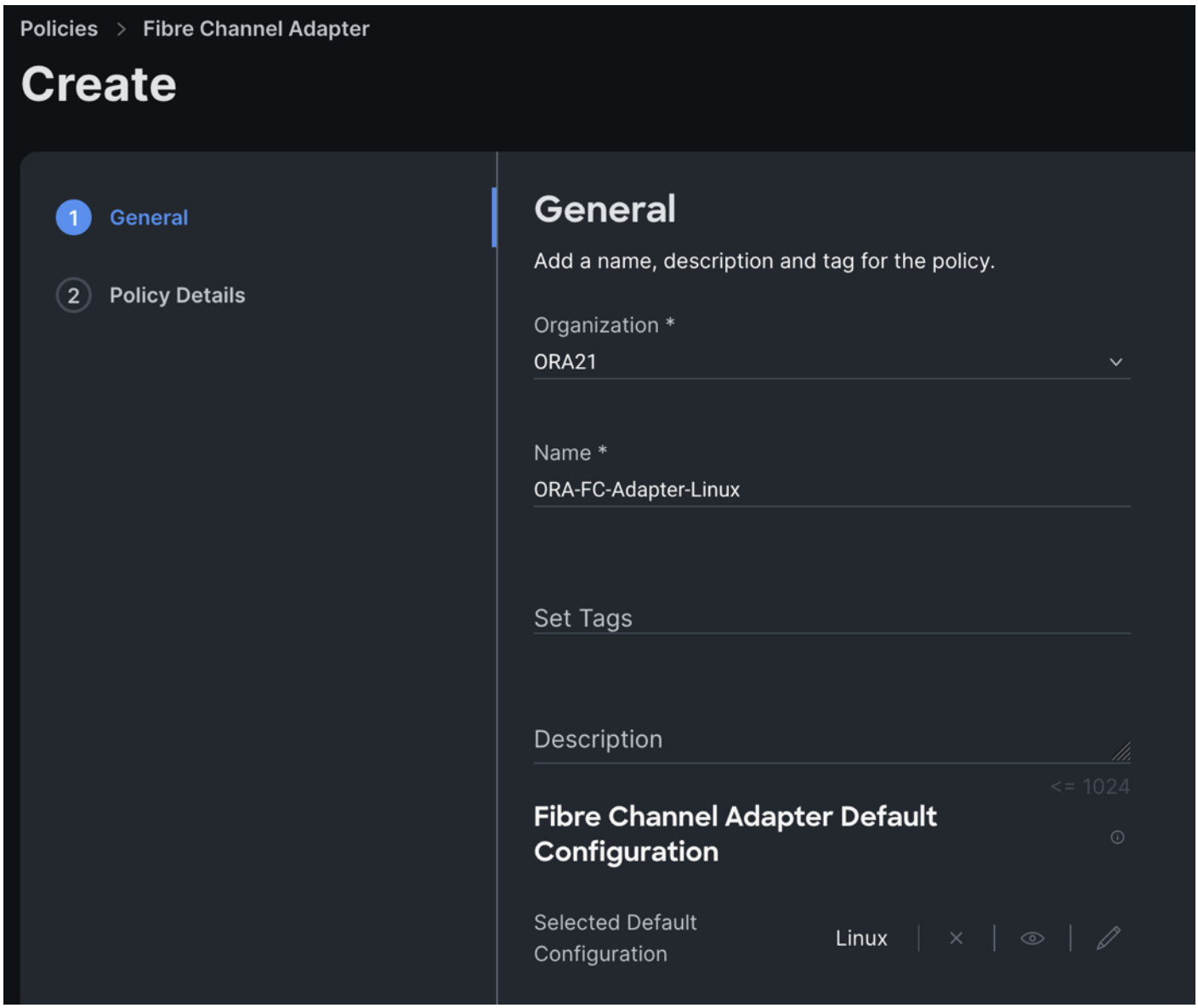
## **Procedure 12.** Create Fibre Channel Adapter Policy

Two vHBA (HBA0 and HBA1) were configured for Boot from SAN and eight vHBAs (HBA2 to HBA9) were configured to carry the NVMe/FC network traffic for the databases. We have created two different Fibre Channel Adapter Policy for both FC and NVMe/FC as explained below.

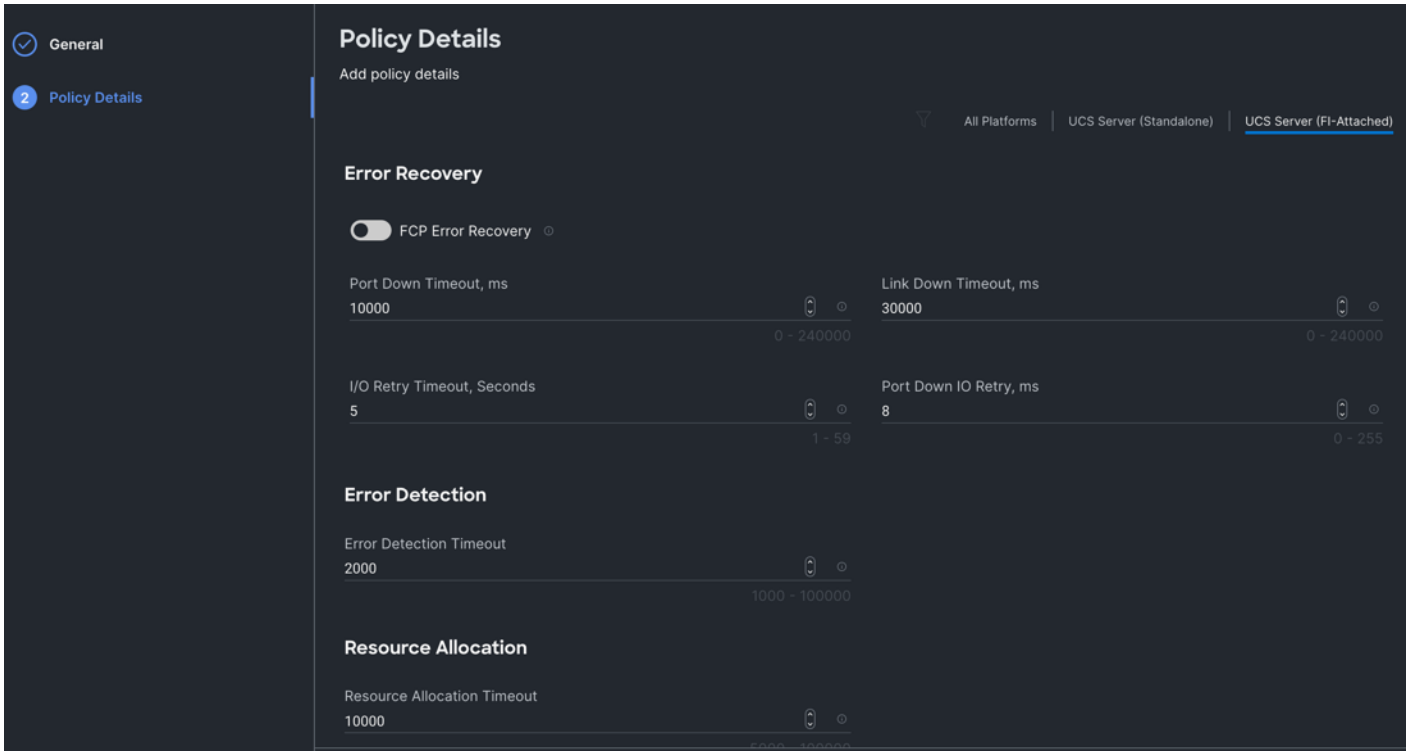
**Step 1.** To configure the Fibre Channel Adapter Policy for the UCS Server profile, go to > Infrastructure Service > Configure > Polices > and click Create Policy.

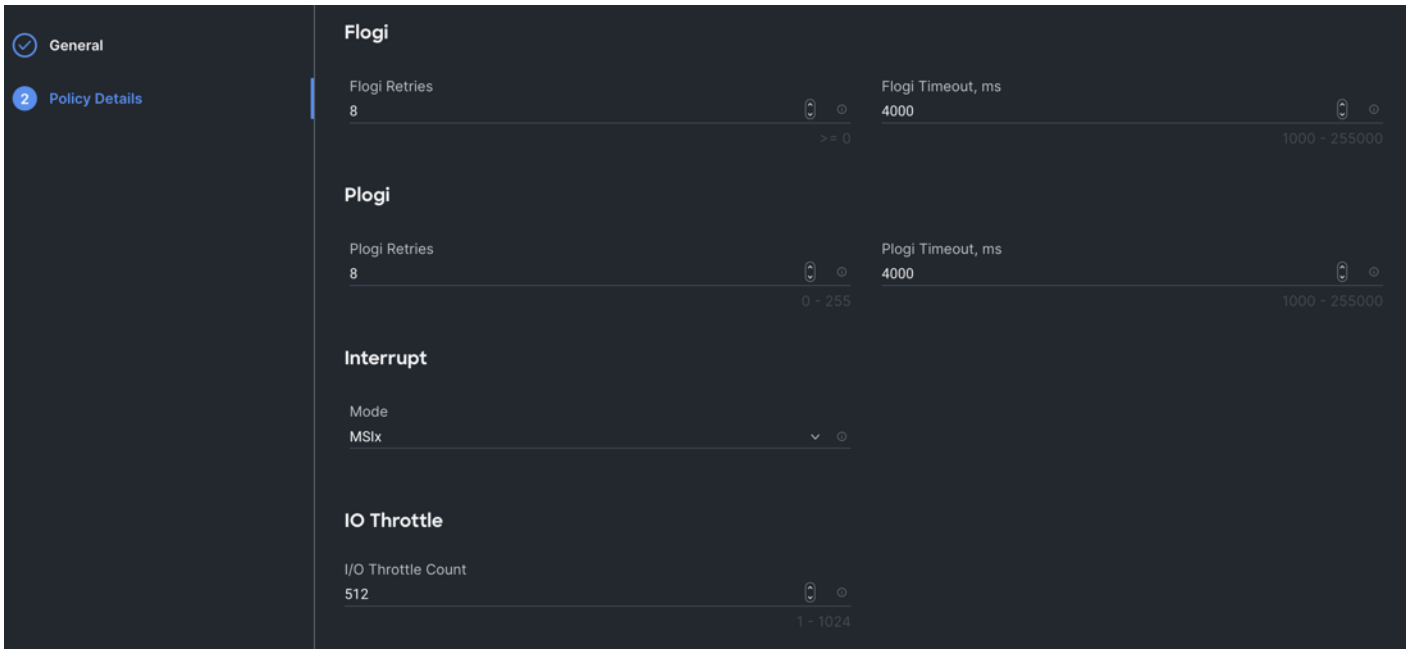
**Step 2.** For the platform type select UCS Server and for the policy select Fibre Channel Adapter.

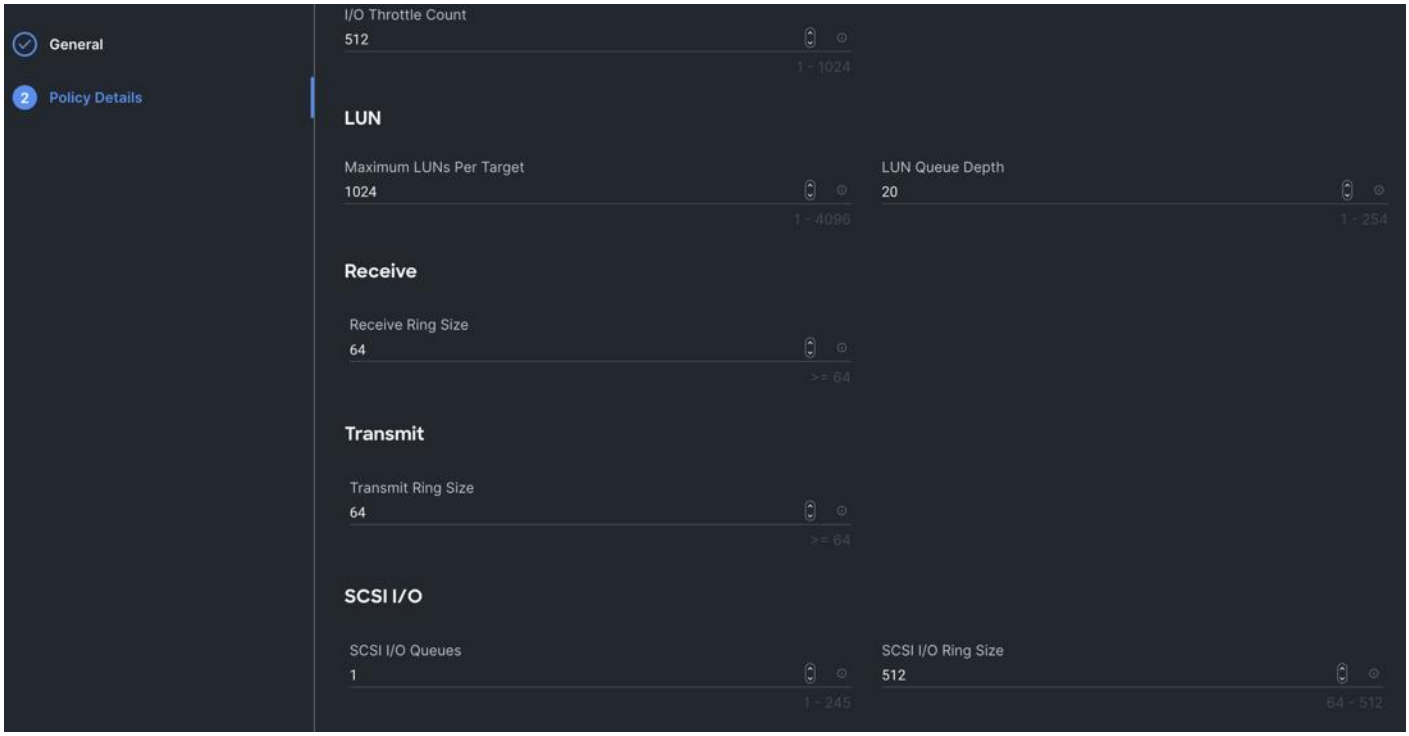
**Step 3.** In the Create Fibre Channel Adapter Configuration section, for the Organization select ORA21 and for the policy name enter “ORA-FC-Adapter-Linux”. For the Fibre Channel Adapter Default Configuration, select Linux and click Next.




**Note:** For this solution, we used the default linux adapter settings to configure the FC HBA's while we used FCNVMeInitiator configuration for NVMe/FC HBA's.



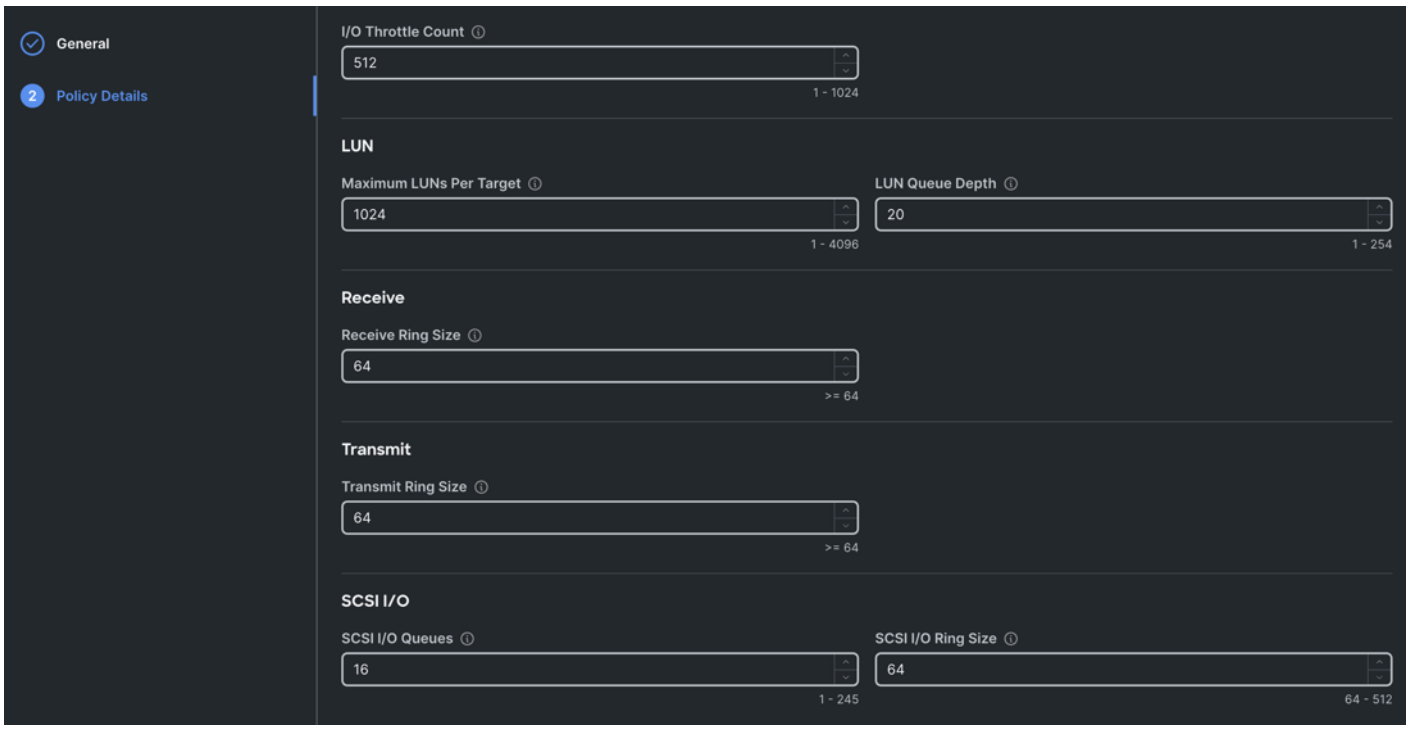



**Step 4.** Click Create to create this policy for vHBA for FC HBA's.

**Step 5.** Now similarly, to configure another Fibre Channel Adapter Policy for the NVMe/FC HBA's, go to > Infrastructure Service > Configure > Polices > and click Create Policy. For the platform type select UCS Server and for the policy select Fibre Channel Adapter.

**Step 6.** In the Create Fibre Channel Adapter Configuration section, for the Organization select ORA21 and for the policy name enter "ORA-NVMe-Adapter-Linux." For the Fibre Channel Adapter Default Configuration, select "FCNVMeInitiator" and click Next.

**Note:** We kept all the parameter to default settings and set "SCSI I/O Queues" to 16 as shown below:




**Step 7.** Click on Create to create this policy for vHBA for NVMe HBA's.

## Procedure 13. Configure SAN Connectivity Policy

As mentioned previously, two vHBA (HBA0 and HBA1) were configured for Boot from SAN on two VSANs. HBA0 was configured to carry the FC Network Traffic on VSAN 151 and boot from SAN through the MDS-A Switch while HBA1 was configured to carry the FC Network Traffic on VSAN 152 and boot from SAN through the MDS-B Switch.

A total of eight vHBAs were configured to carry the NVMe/FC network traffic for the database on two VSANs. Four vHBAs (HBA2, HBA4, HBA6 and HBA8) were configured to carry the NVMe/FC network traffic on VSAN 151 for Oracle RAC database storage traffic through MDS-A Switch. Four vHBA (HBA3, HBA5, HBA7 and HBA9) were configured to carry the NVMe/FC network traffic on VSAN 152 for Oracle RAC database storage traffic through the MDS-B Switch.

For each Server node, a total of 10 vHBAs were configured as listed in Table 10.

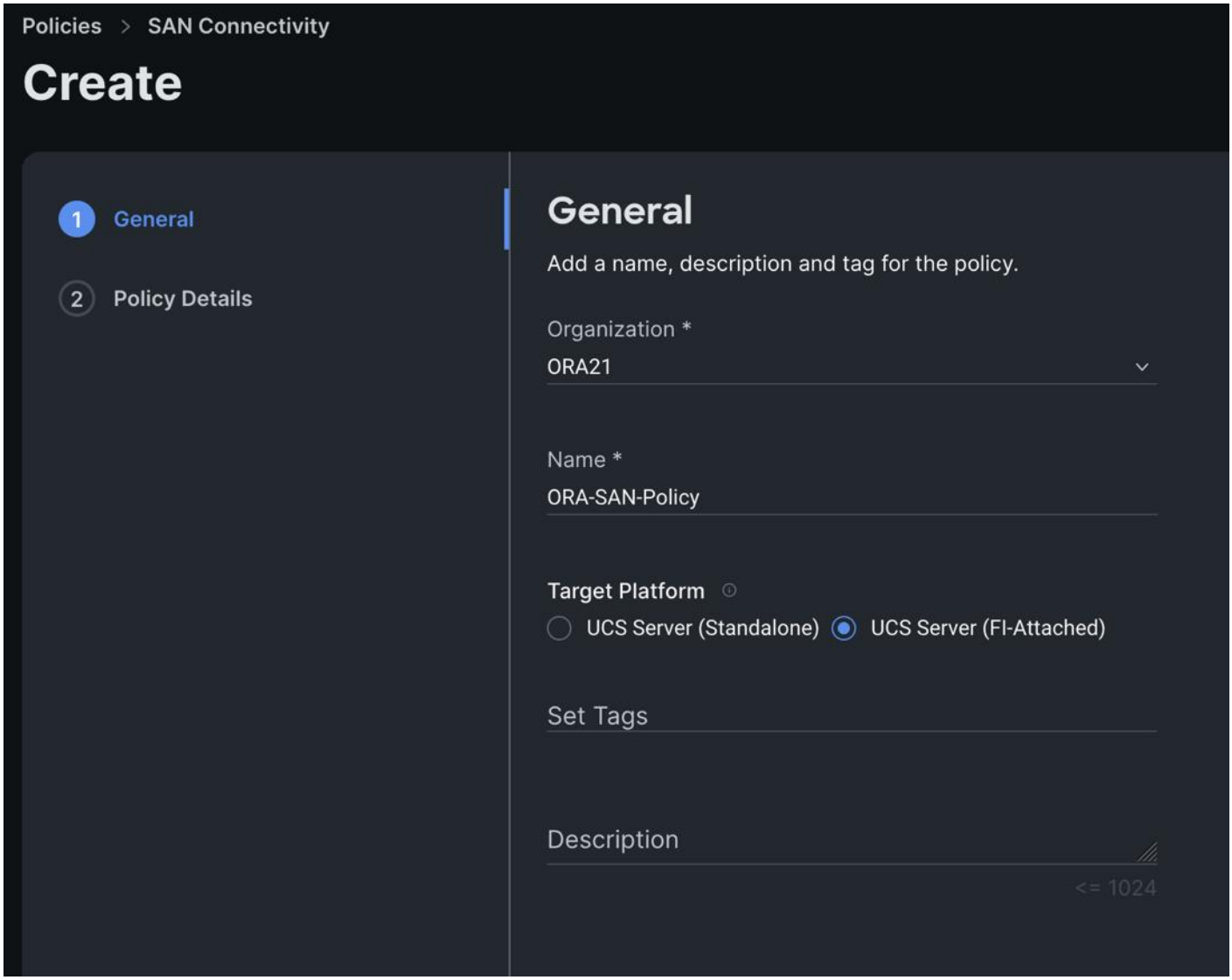
**Table 10. Configured vHBAs**

| Name | vHBA Type | Switch ID | PCI-Order | Fibre Channel Network | Fibre Channel Adapter | Fibre Channel QoS |
|---|---|---|---|---|---|---|
| HBA0 | fc-initiator | FI – A | 2 | ORA-FC-Network-151 | ORA-FC-Adapter-Linux | ORA-FC-QoS |
| HBA1 | fc-initiator | FI – B | 3 | ORA-FC-Network-152 | ORA-FC-Adapter-Linux | ORA-FC-QoS |
| HBA2 | fc-nvme-initiator | FI – A | 4 | ORA-FC-Network-151 | ORA-NVMe-Adapter-Linux | ORA-FC-QoS |

| Name | vHBA Type | Switch ID | PCI-Order | Fibre Channel Network | Fibre Channel Adapter | Fibre Channel QoS |
|---|---|---|---|---|---|---|
| HBA3 | fc-nvme-initiator | FI – B | 5 | ORA-FC-Network-152 | ORA-NVMe-Adapter-Linux | ORA-FC-QoS |
| HBA4 | fc-nvme-initiator | FI – A | 6 | ORA-FC-Network-151 | ORA-NVMe-Adapter-Linux | ORA-FC-QoS |
| HBA5 | fc-nvme-initiator | FI – B | 7 | ORA-FC-Network-152 | ORA-NVMe-Adapter-Linux | ORA-FC-QoS |
| HBA6 | fc-nvme-initiator | FI – A | 8 | ORA-FC-Network-151 | ORA-NVMe-Adapter-Linux | ORA-FC-QoS |
| HBA7 | fc-nvme-initiator | FI – B | 9 | ORA-FC-Network-152 | ORA-NVMe-Adapter-Linux | ORA-FC-QoS |
| HBA8 | fc-nvme-initiator | FI – A | 10 | ORA-FC-Network-151 | ORA-NVMe-Adapter-Linux | ORA-FC-QoS |
| HBA9 | fc-nvme-initiator | FI – B | 11 | ORA-FC-Network-152 | ORA-NVMe-Adapter-Linux | ORA-FC-QoS |

**Step 1.** Go to > Infrastructure Service > Configure > Polices > and click Create Policy. For the platform type select UCS Server and for the policy select SAN Connectivity.

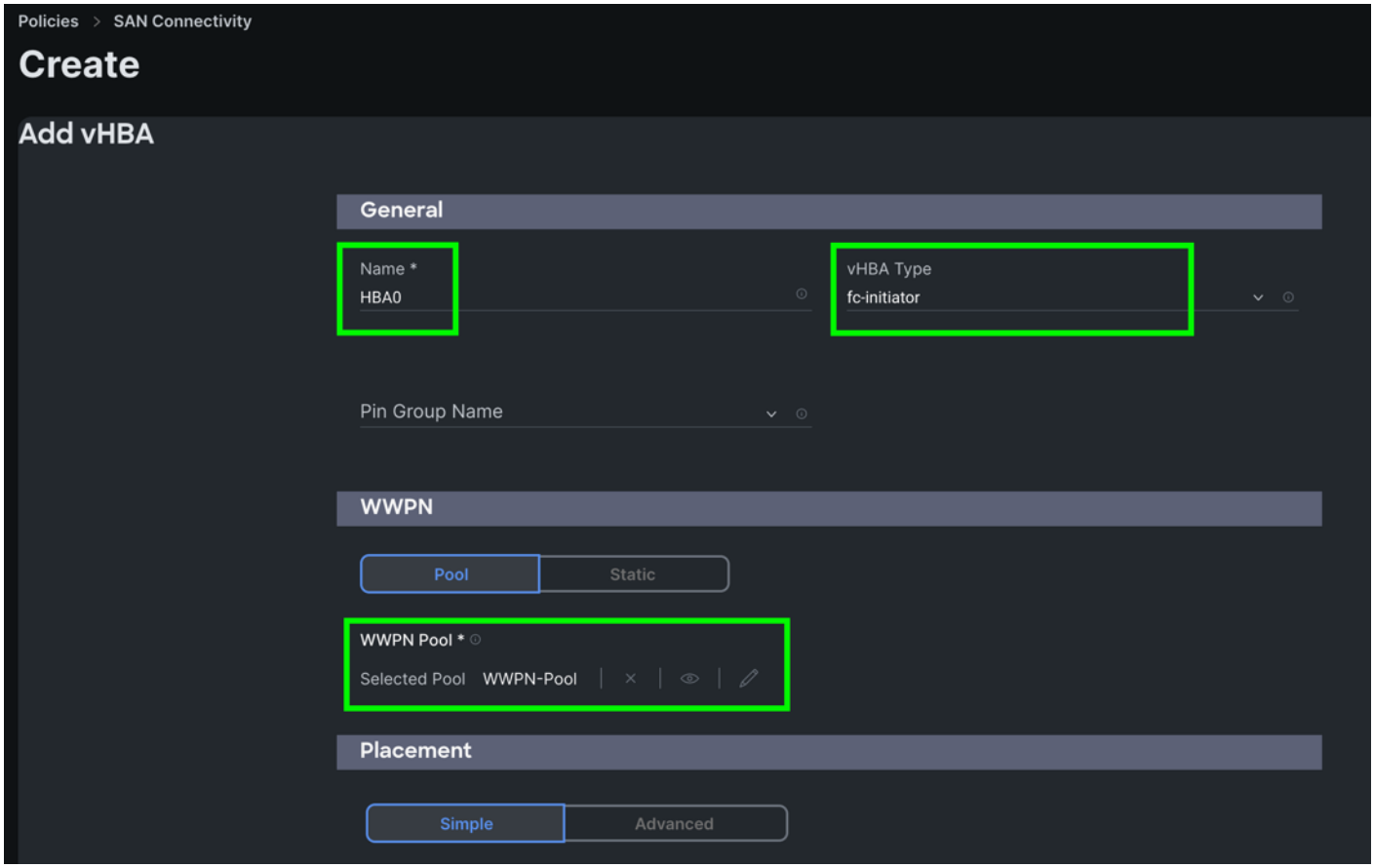
**Step 2.** In the SAN Connectivity Policy Create section, for the Organization select ORA21,for the policy name enter ORA-SAN-Policy and for the Target Platform select UCS Server (FI-Attached). Click Next.




**Step 3.** In the Policy Details section, select WWNN Pool and then select WWNN-Pool that you previously created. Click Add vHBA.

**Step 4.** In the Add vHBA section, for the Name enter “HBA0” and for the vHBA Type enter “fc-initiator.”

**Step 5.** For the WWPN Pool, select the WWPN-Pool that you previously created, as shown below:

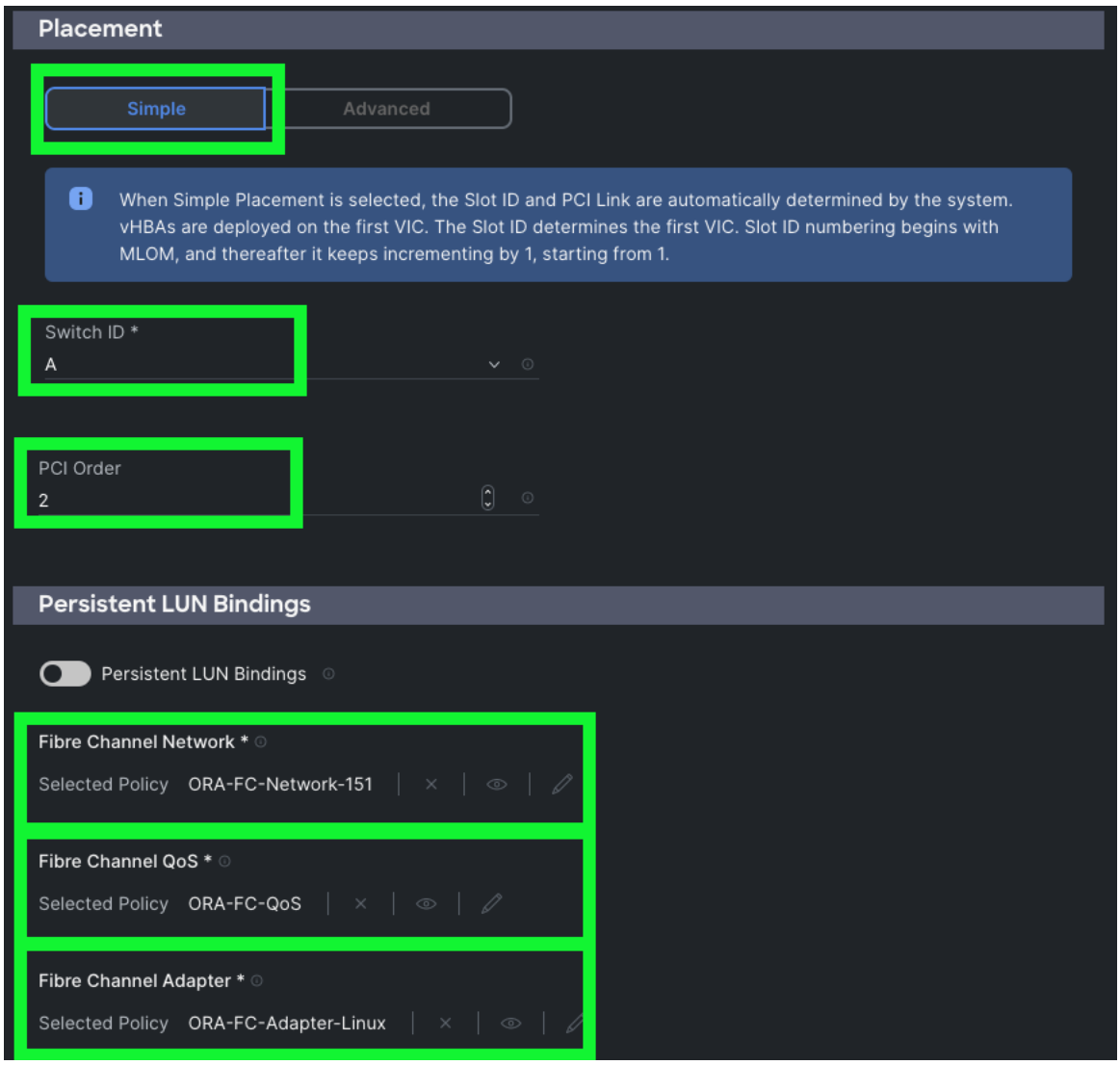



**Step 6.** For the Placement, keep the option Simple and for the Switch ID select A and for the PCI Order select 2.

**Step 7.** For the Fibre Channel Network select ORA-FC-Network-151.

**Step 8.** For the Fibre Channel QoS select ORA-FC-QoS.

**Step 9.** For the Fibre Channel Adapter select ORA-FC-Adapter-Linux.

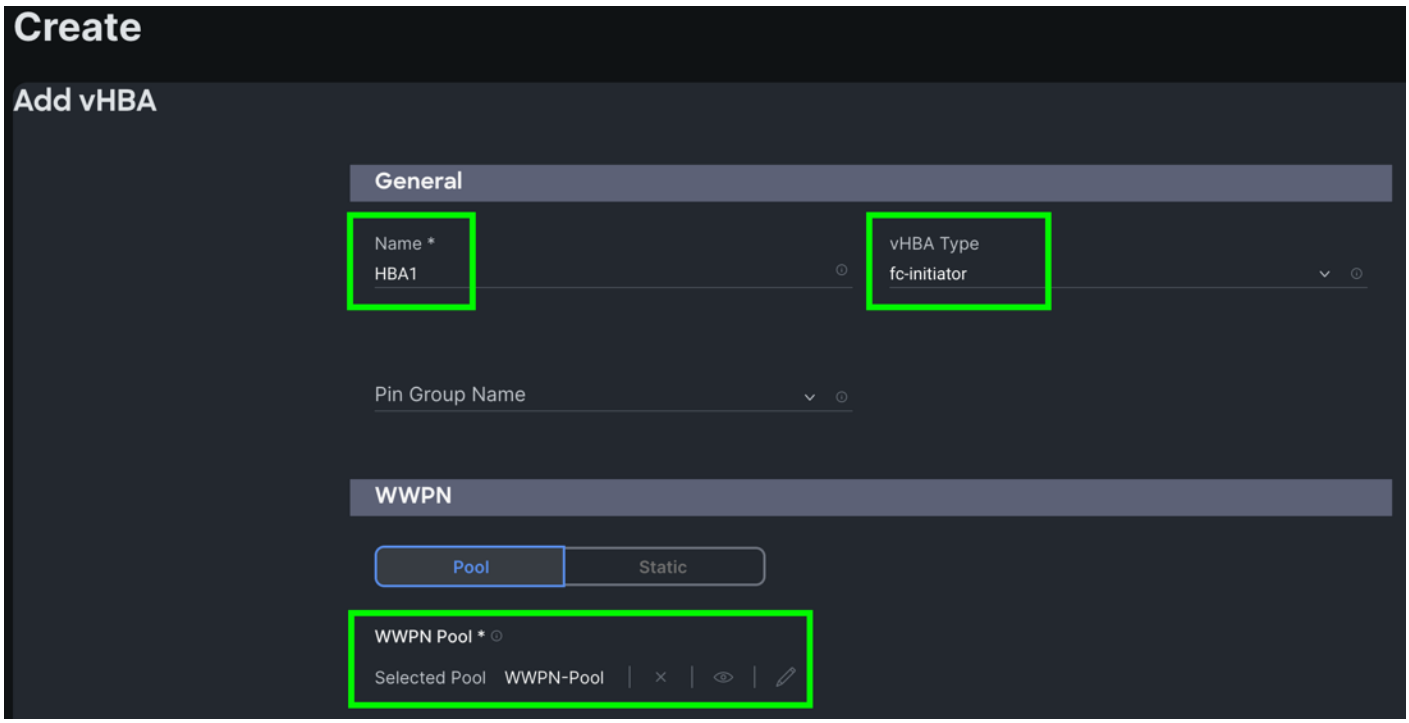



**Step 10.** Click Add to add this first HBA0.

**Step 11.** Click Add vHBA to add a second HBA.

**Step 12.** In the Add vHBA section, for the Name enter “HBA1” and for the vHBA Type select fc-initiator.

**Step 13.** For the WWPN Pool select WWPN-Pool that was previously create, as shown below:

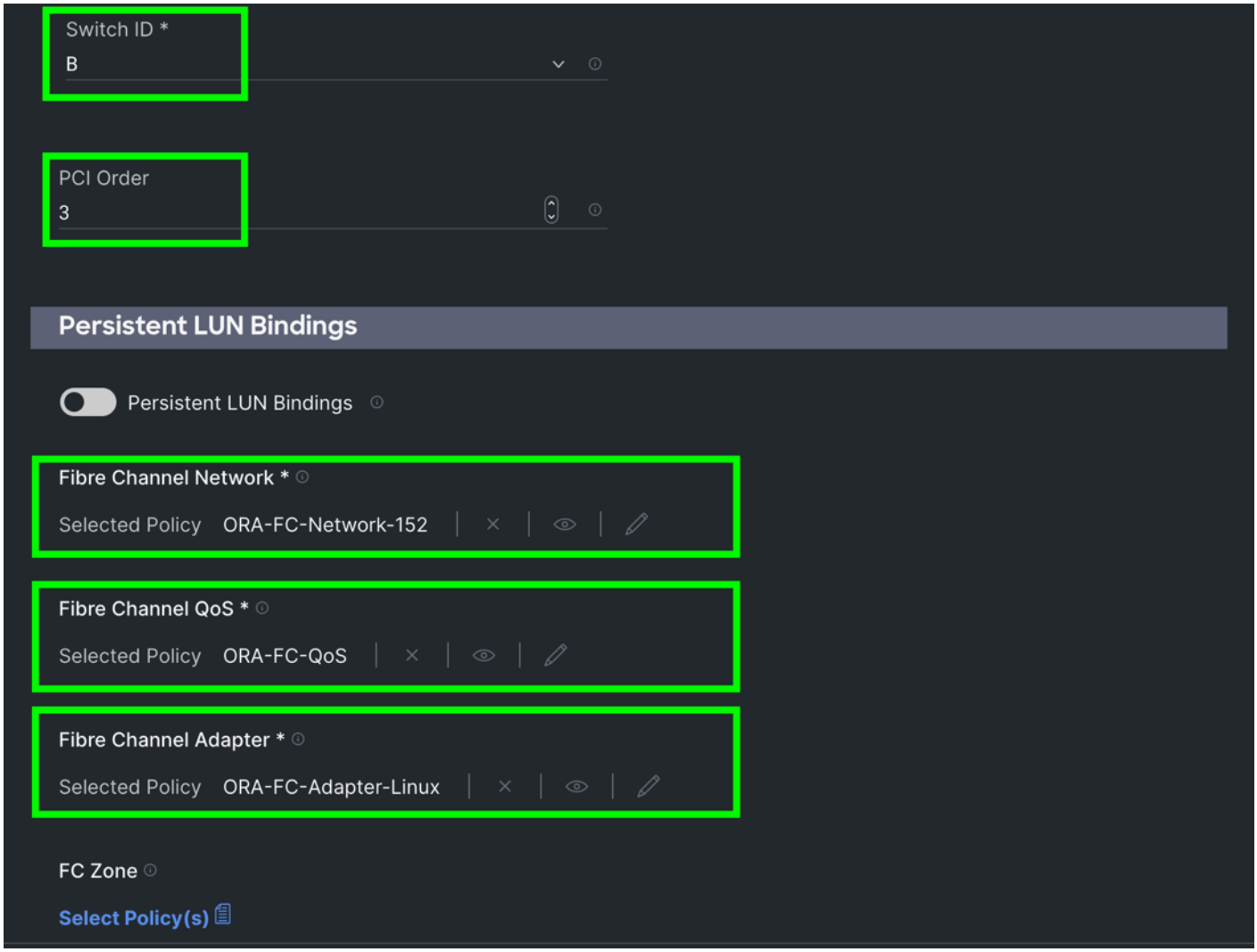



**Step 14.** For the Placement, keep the option Simple and for Switch ID select B and for the PCI Order select 3.

**Step 15.** For the Fibre Channel Network select ORA-FC-Network-152.

**Step 16.** For the Fibre Channel QoS select ORA-FC-QoS.

**Step 17.** For the Fibre Channel Adapter select ORA-FC-Adapter-Linux.



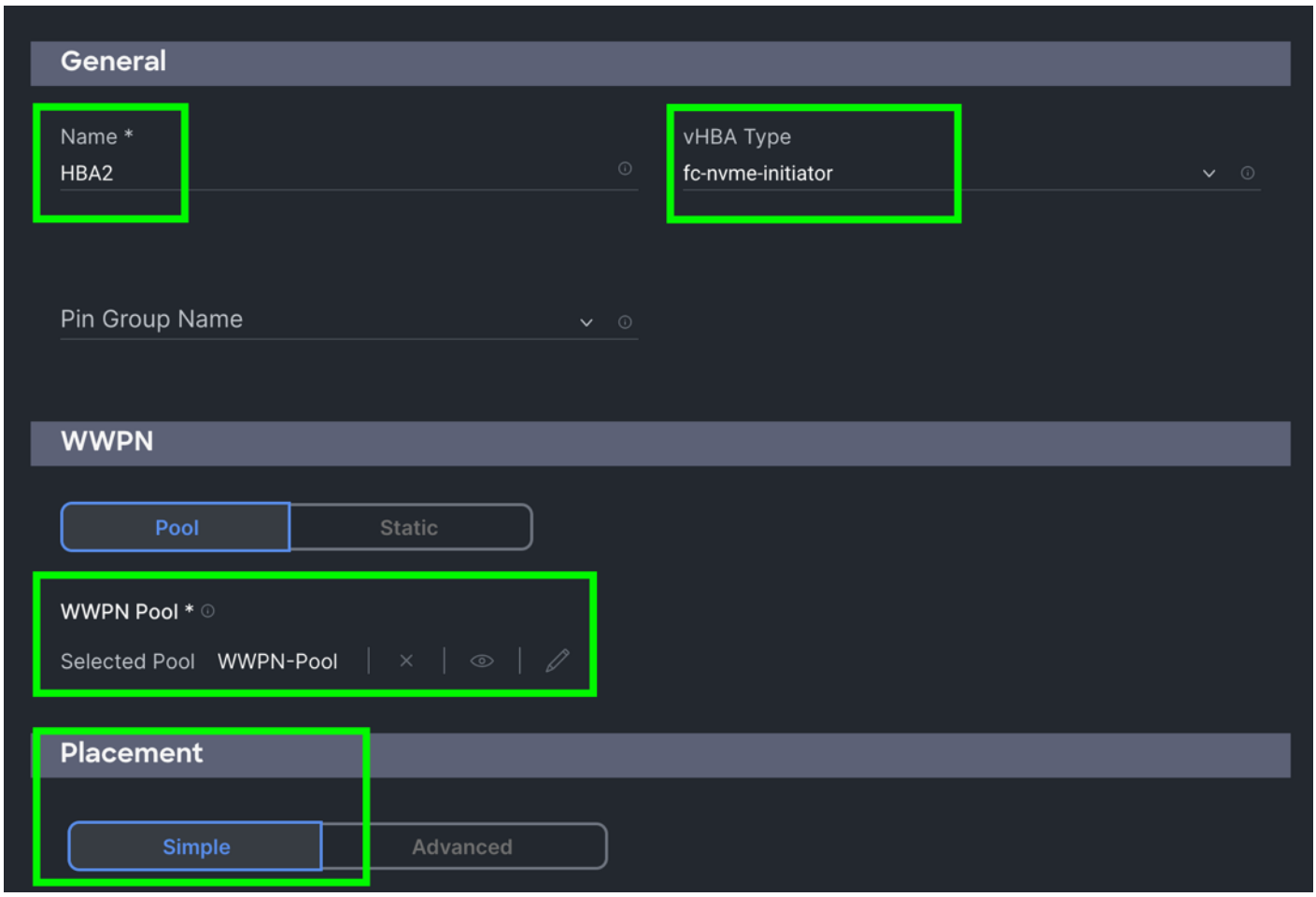

**Step 18.** Click Add to add this second HBA1.

**Note:** For this solution, we added another eight HBA for NVME/FC.

**Step 19.** Click Add vHBA.

**Step 20.** In the Add vHBA section, for the Name enter “HBA2” and for the vHBA Type select fc-nvme-initiator.

**Step 21.** For the WWPN Pool select WWPN-Pool, which was previously created, as shown below:




**Step 22.** For the Placement, keep the option Simple and for the Switch ID select A and for the PCI Order select 4.

**Step 23.** For the Fibre Channel Network select ORA-FC-Network-151.

**Step 24.** For the Fibre Channel QoS select ORA-FC-QoS.

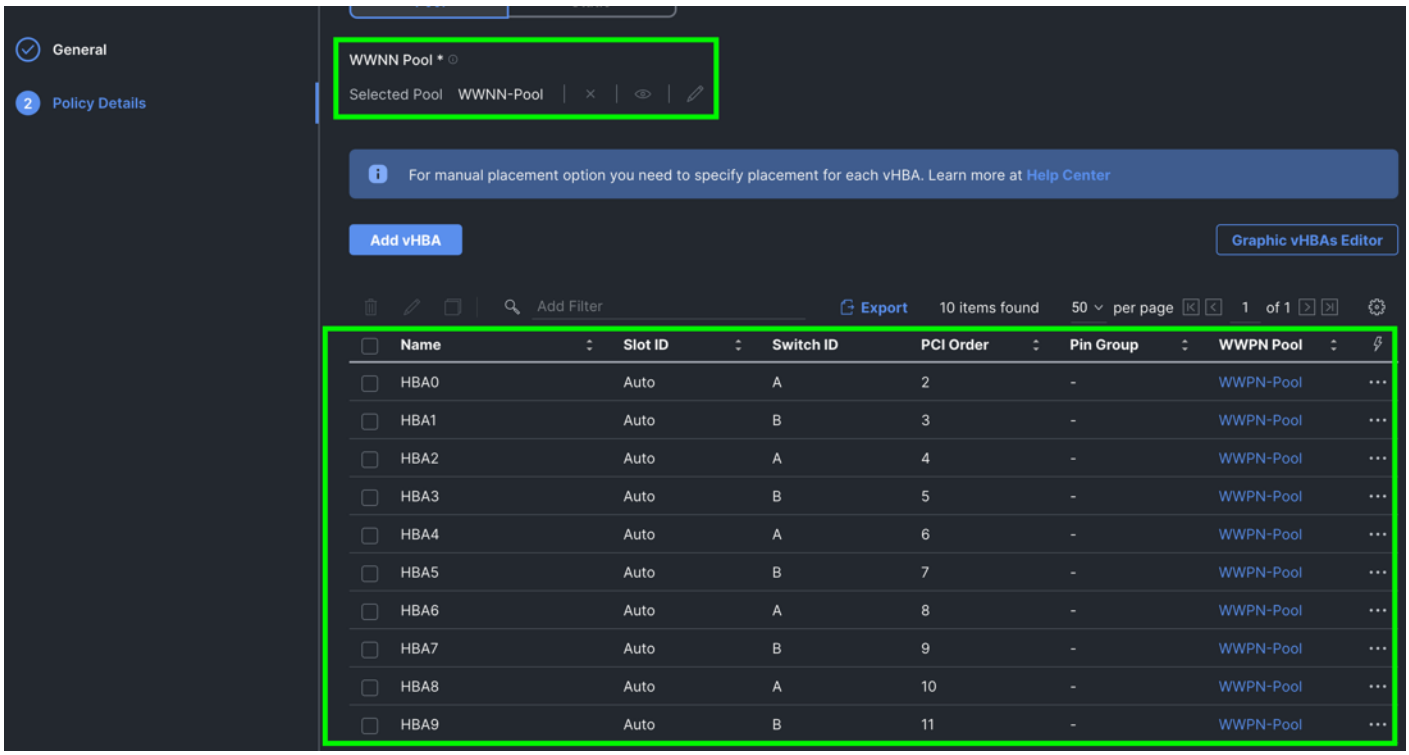
**Step 25.** For the Fibre Channel Adapter select ORA-NVMe-Adapter-Linux.

**Step 26.** Click Add to add this HBA2.

**Note:** For this solution, we added another seven HBA for NVME/FC.

**Step 27.** Click Add vHBA and select the appropriate vHBA Type, WWPN Pool, Simple Placement, Switch ID, PCI Order, Fibre Channel Network, Fibre Channel QoS, and Fibre Channel Adapter for all rest of the HBAs listed in Table 10.

**Step 28.** After adding the ten vHBAs, review and make sure the Switch ID, PCI Order, and HBA Type are as shown below:



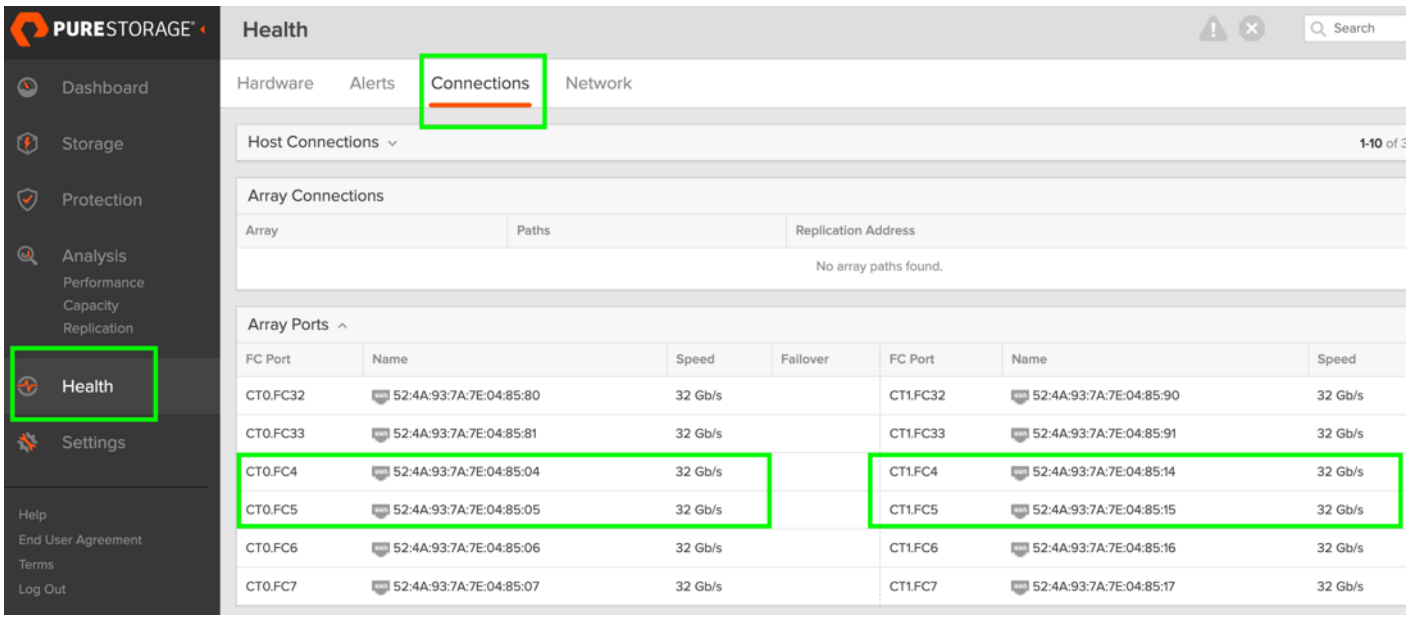

**Step 29.** Click Create to create this policy.

## **Procedure 14.** Configure Boot Order Policy

All Oracle server nodes are set to boot from SAN for this Cisco Validated Design, as part of the Service Profile. The benefits of booting from SAN are numerous; disaster recovery, lower cooling, and power requirements for each server since a local drive is not required, and better performance, and so on. We strongly recommend using “Boot from SAN” to realize the full benefits of Cisco UCS stateless computing features, such as service profile mobility.

**Note:** For this solution, we used SAN Boot and configured the SAN Boot order policy as detailed in this procedure.

To create SAN Boot Order Policy, you need to enter the WWPN of Pure Storage Target ports. The screenshot below shows both the Pure Storage Controller FC Ports and related WWPN:

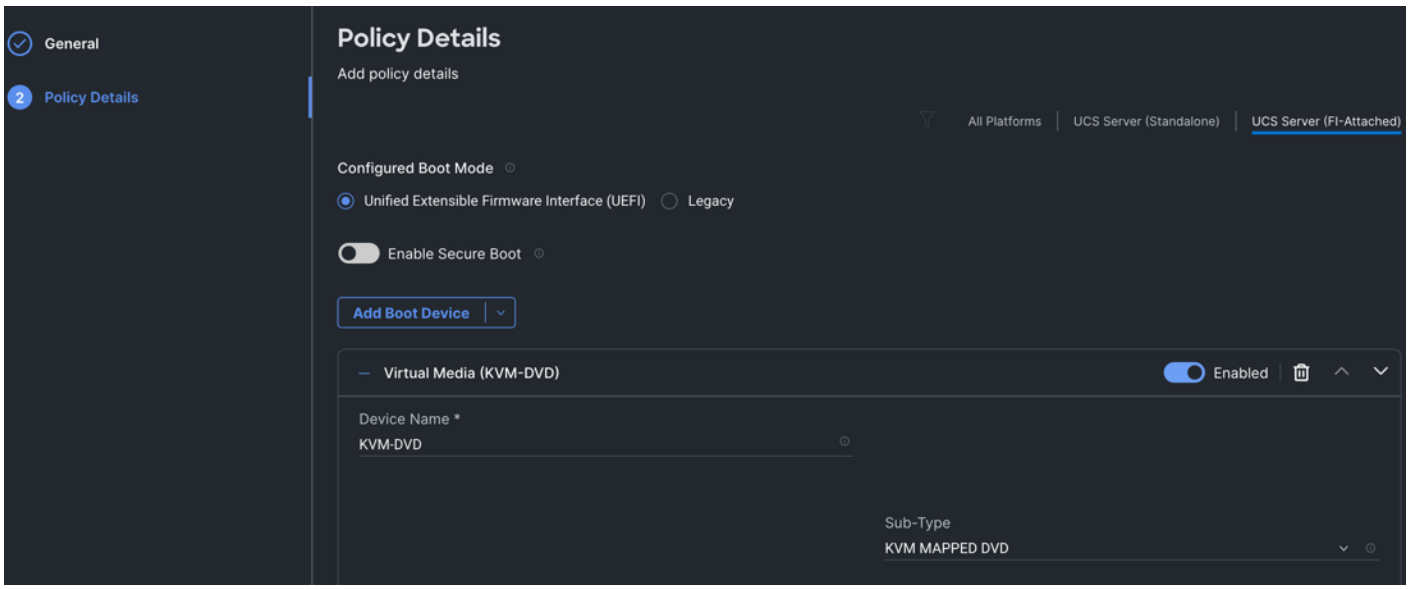



**Step 1.** To configure Boot Order Policy for UCS Server profile, go to > Infrastructure Service > Configure > Policies > and click Create Policy.

**Step 2.** For the platform type select UCS Server and for the policy select Boot Order.

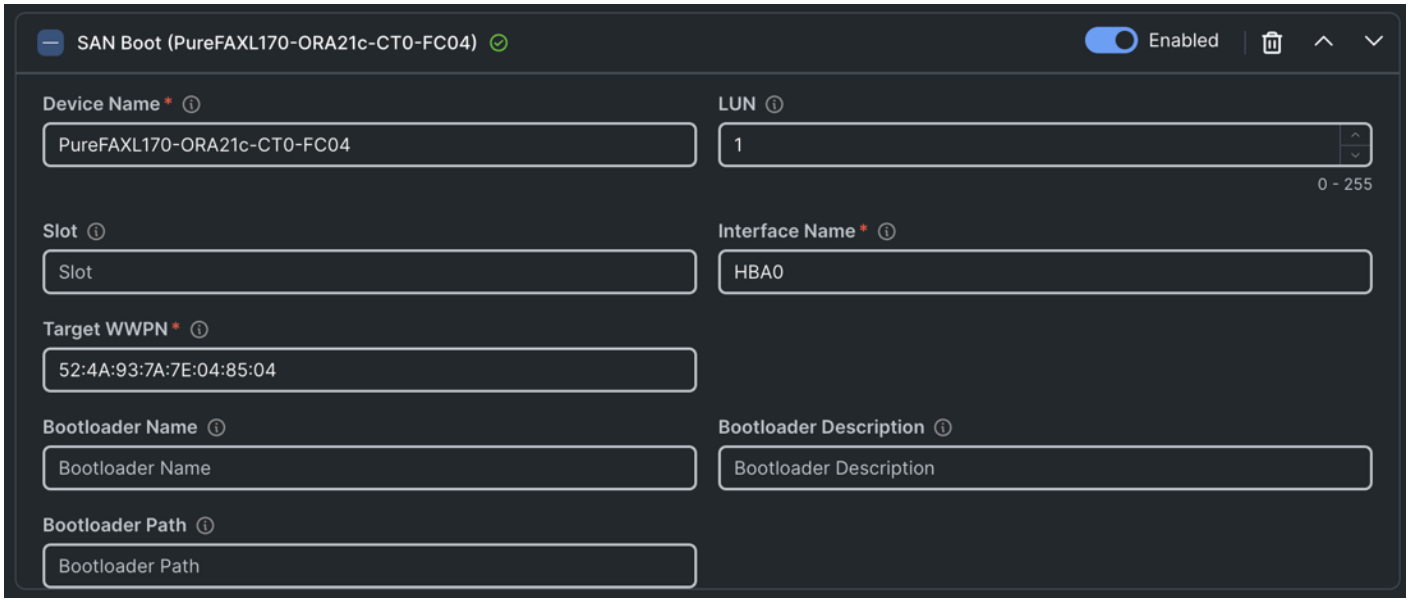
**Step 3.** In the Boot Order Policy Create section, for the Organization select ORA21 and for the name of the Policy select SAN-Boot. Click Next.

**Step 4.** In the Policy Details section, click Add Boot Device and select Virtual Media for the first boot order. Name the device “KVM-DVD” and for the Sub-type select KVM MAPPED DVD as shown below:



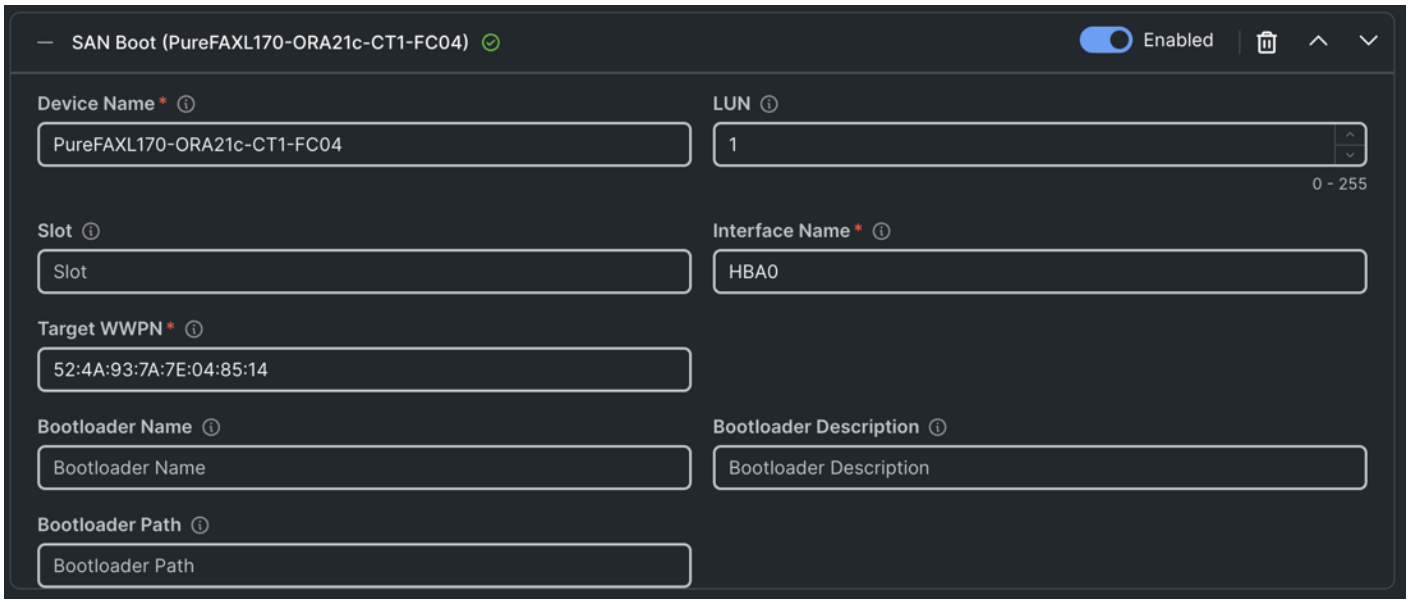

**Step 5.** Add the second boot order: Click Add Boot Device and for the second boot order for HBA0, select SAN Boot as the primary path through the Pure Storage Controller port CT0-FC04.

**Step 6.** Enter the Device Name, Interface Name, and Target WWPN according to storage target.



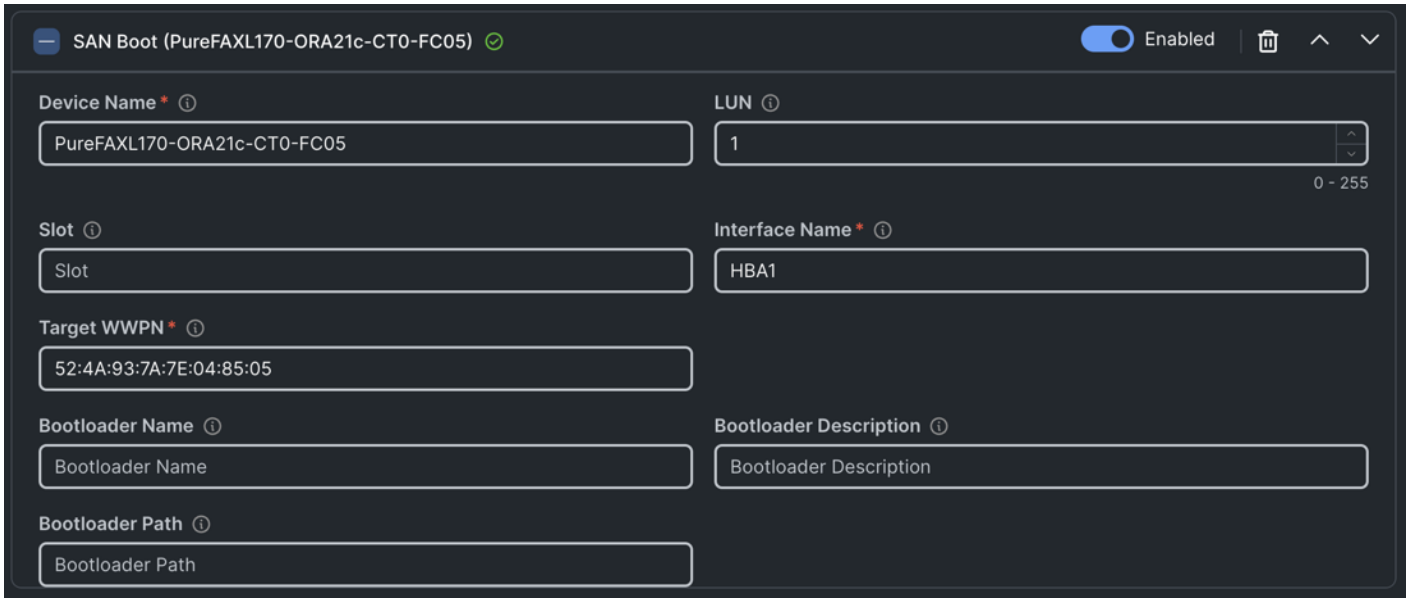

**Note:** We added a third boot order and the appropriate target for HBA0 as the secondary path through Pure Storage Controller port CT1-FC04 as shown in the screenshot below.

**Step 7.** Enter the Device Name, Interface Name, and Target WWPN according to storage target.



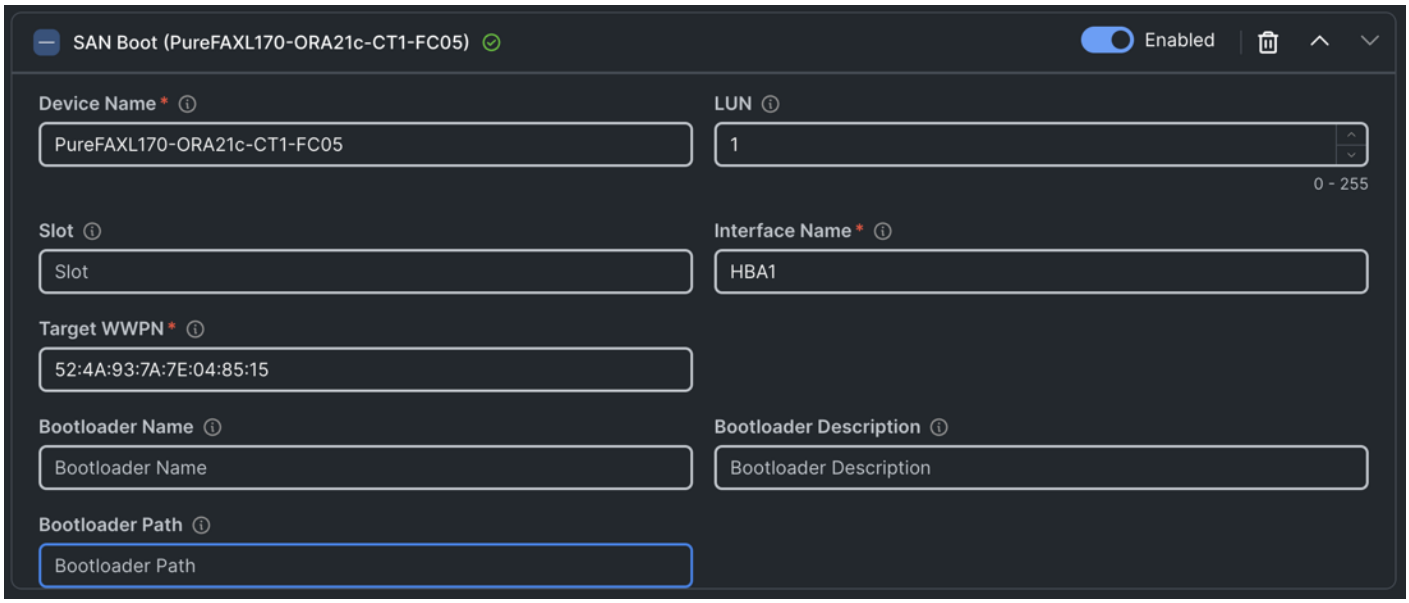

**Note:** We added a fourth boot order for now HBA1 as the primary path through Pure Storage Controller port CT0-FC05.

**Step 8.** Enter the Device Name, Interface Name and Target WWPN according to storage target.



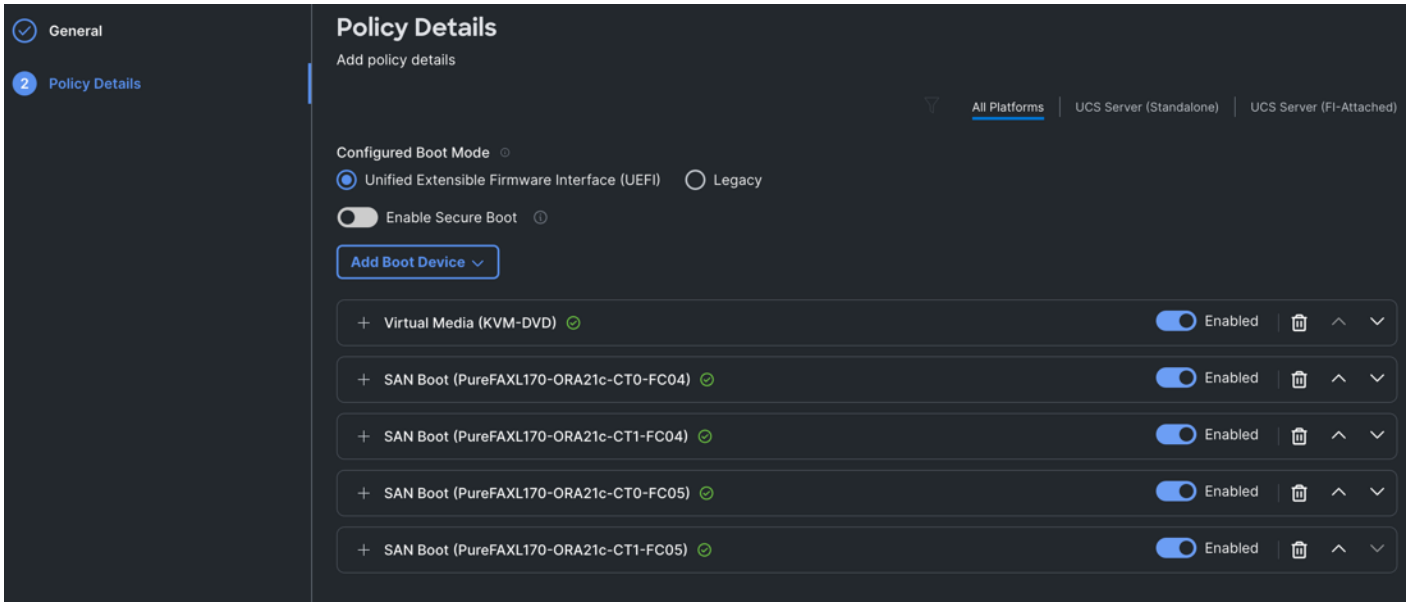

**Note:** We added a fifth boot order for HBA1 as the secondary path through Pure Storage Controller port CT1-FC05.

**Step 9.** Enter the Device Name, Interface Name and Target WWPN according to storage target.




**Step 10.** By configuring both FC Boot HBAs (HBA0 and HBA1) with the Primary and Secondary path, you have configured high availability for SAN boot, as well as a fourth path for the OS Boot LUNs.

**Step 11.** Review the Policy details and verify that all four SAN boot paths are configured to provide high availability as shown below:




**Step 12.** Click Create to create this SAN boot order policy.

## **Procedure 15.** Configure and Deploy Server Profiles

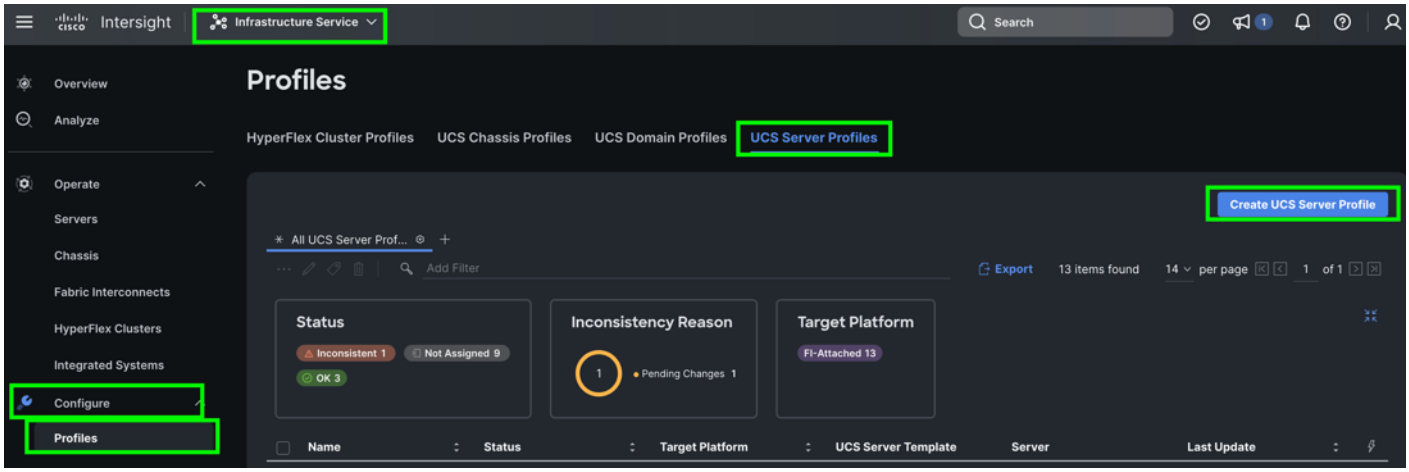
The Cisco Intersight server profile allows server configurations to be deployed directly on the compute nodes based on polices defined in the profile. After a server profile has been successfully created, server profiles can be attached with the Cisco UCS X210c M7 Compute Nodes.

**Note:** For this solution, we configured eight server profiles; ORARAC1 to ORARAC8. We assigned the server profile ORARAC1 to Chassis 1 Server 1, server profile ORARAC2 to Chassis 1 Server 3, server profile ORARAC3 to Chassis 1 Server 5 and server profile ORARAC4 to Chassis 1 Server 7. We assigned the server profile ORARAC5 to Chassis 2 Server 1, server profile ORARAC6 to Chassis 2 Server 3, server profile ORARAC7 to Chassis 2 Server 5 and server profile ORARAC8 to Chassis 2 Server 7.

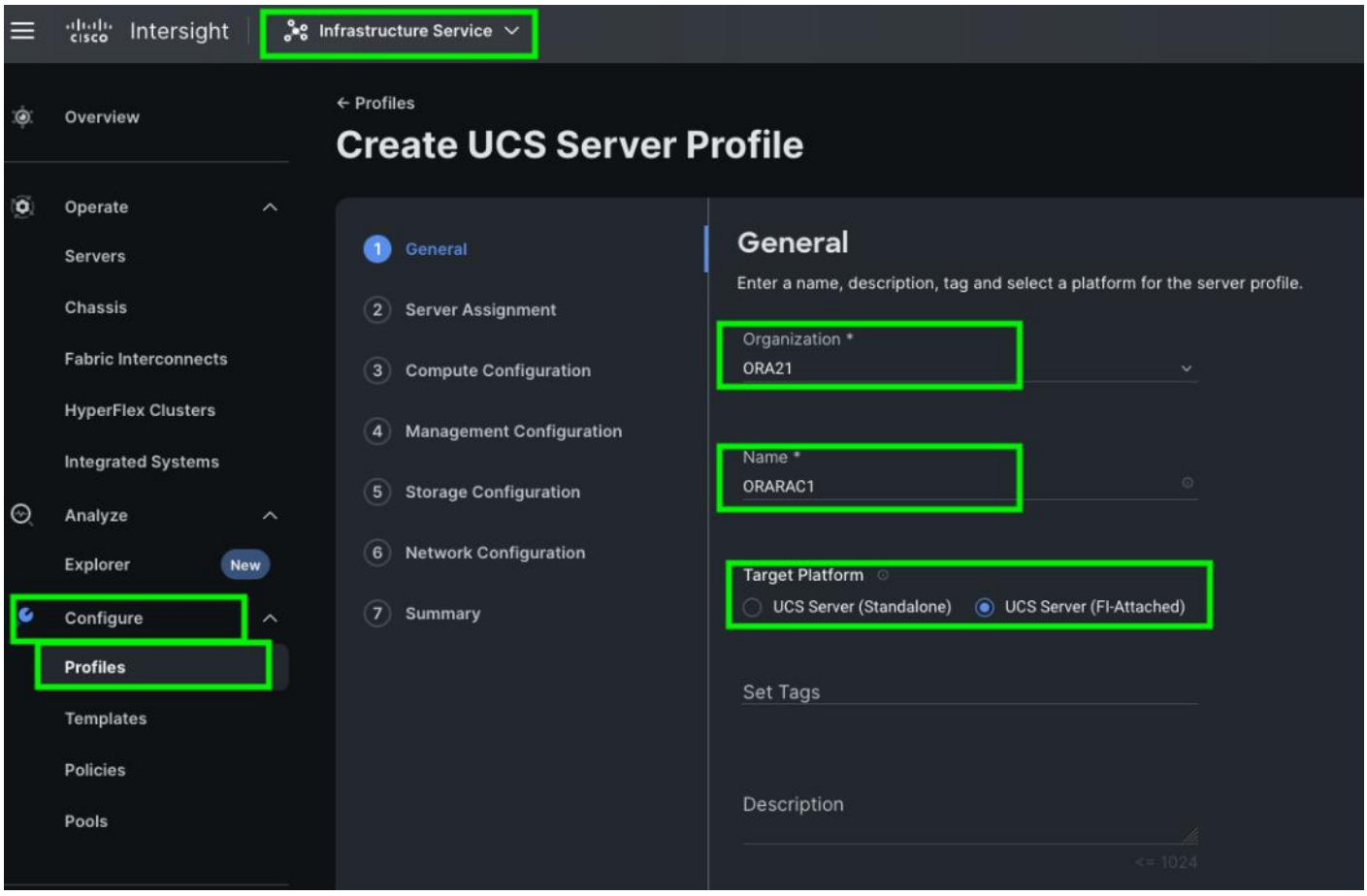
**Note:** All eight x210c M7 servers will be used to create Oracle RAC database nodes later in the database creation section.

**Note:** For this solution, we configured one server profile “ORARAC1” and attached all policies for the server profile which were configured in the previous section. We cloned the first server profile and created seven more server profiles; “ORARAC2”, “ORARAC3”, “ORARAC4”, “ORARAC5”, “ORARAC6”, “ORARAC7”and “ORARAC8”. Alternatively, you can create a server profile template with all server profile policies and the derive server profile from the standard template.

**Step 1.** To create a server profile, go to > Infrastructure Service > Configure > Profile > and then select the tab for UCS Server Profile. Click Create UCS Server Profile.

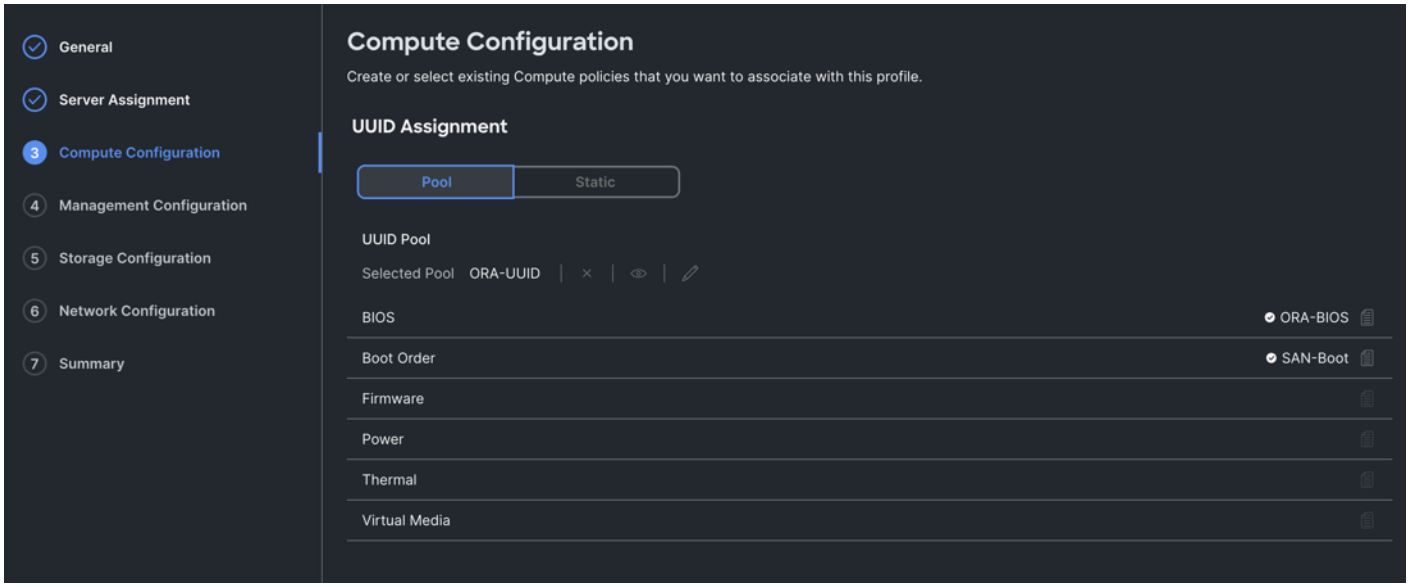



**Step 2.** In Create Server Profile, for the Organization select ORA21 and for the Name for the Server Profile enter "ORARAC1." For the Target Platform type select UCS Server (FI-Attached).

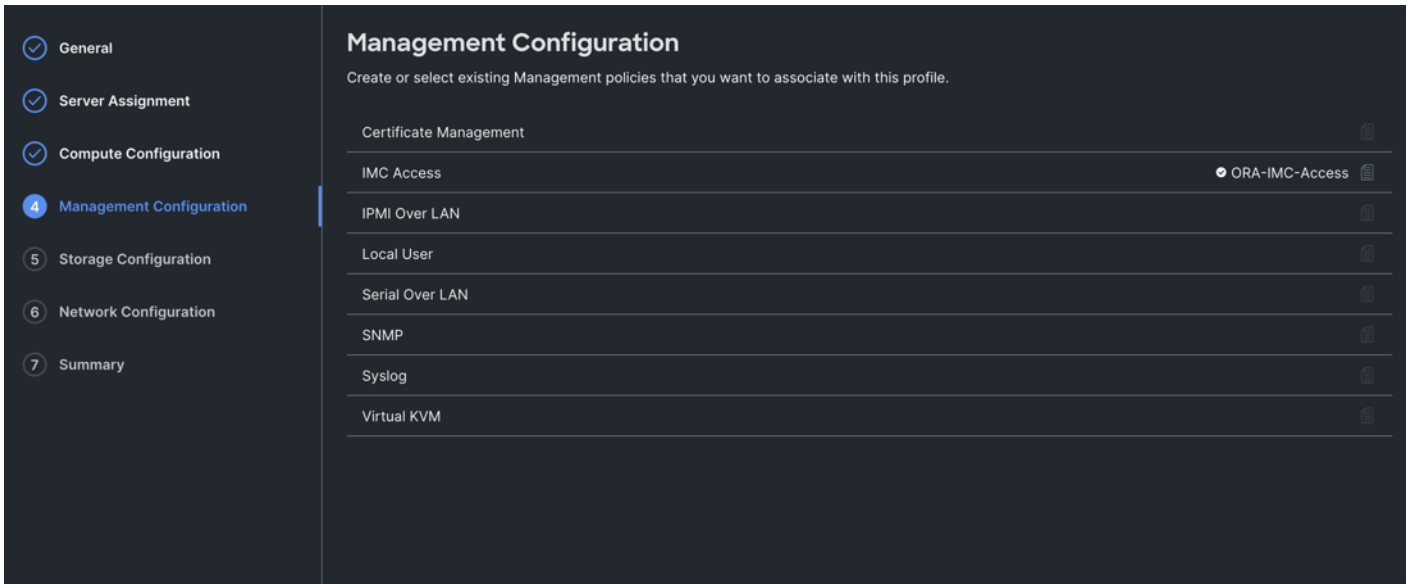



**Step 3.** In the Server Assignment menu, select Chassis 1 Server 1 to assign this server profile and click Next.

**Step 4.** In the Compute Configuration menu, select UUID Pool and select the ORA-UUID option that you previously created. For the BIOS select ORA-BIOS and for the Boot Order select SAN-Boot that you previously created. Click Next.



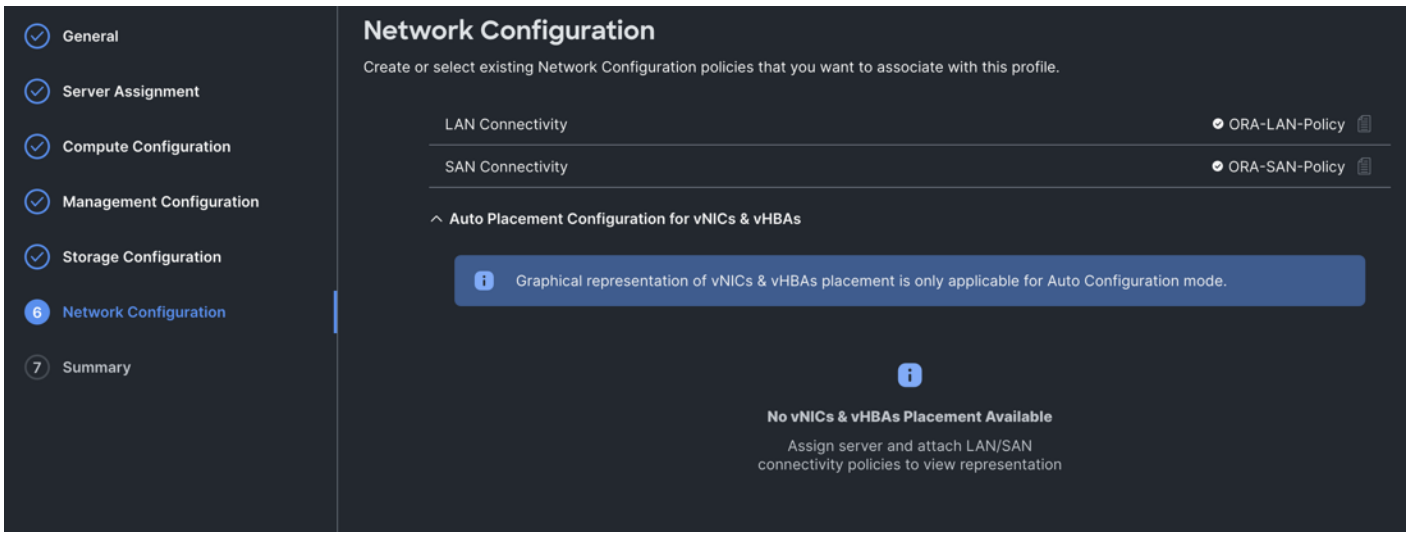

**Step 5.** In the Management Configuration menu, for the IMC Access select ORA-IMC-Access to configure the Server KVM access and then click Next.




**Note:** We didn't configure any local storage or any storage policies for this solution.

**Step 6.** Click Next to go to Network configuration.

**Step 7.** For the Network Configuration section, for the LAN connectivity select ORA-LAN-Policy and for the SAN connectivity select ORA-SAN-Policy that you previously created.

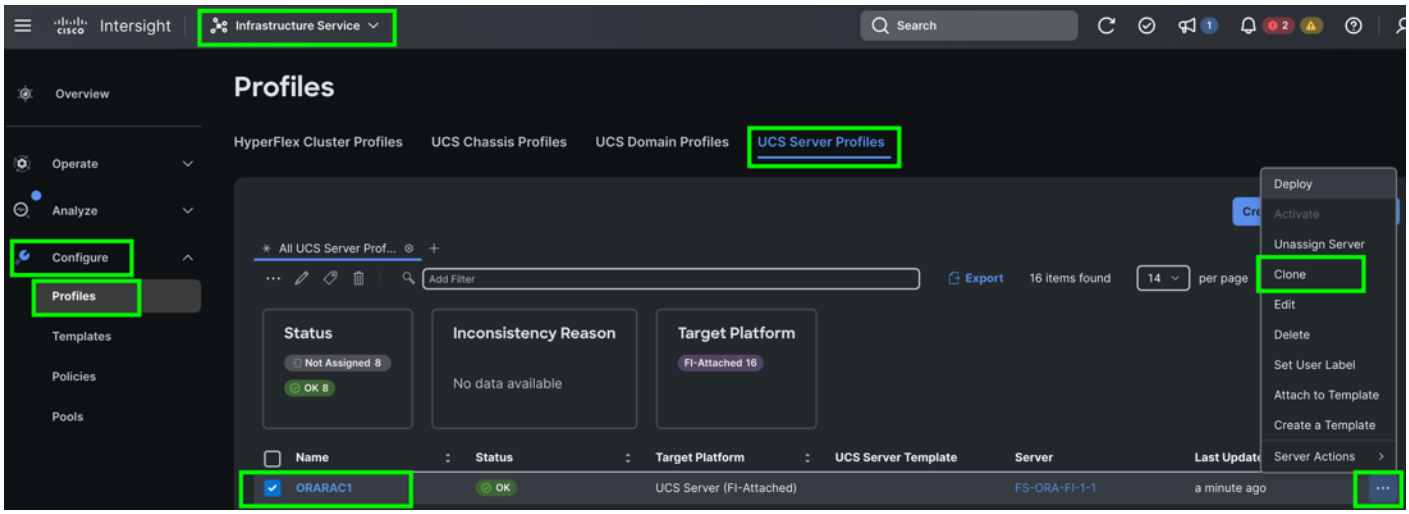



**Note:** By assigning these LAN and SAN connectivity in the server profile, the server profile will create and configure two vNIC and ten vHBA on the server for management, private interconnect, and storage network traffic.

**Step 8.** Click Next and review the summary for the server profile and click Deploy to assign this server profile to the first server.

**Note:** After this server profile "ORARAC1" deploys successfully on chassis 1 server 1, you can clone this server profile to create another seven identical server profile for the rest of the seven remaining server nodes.

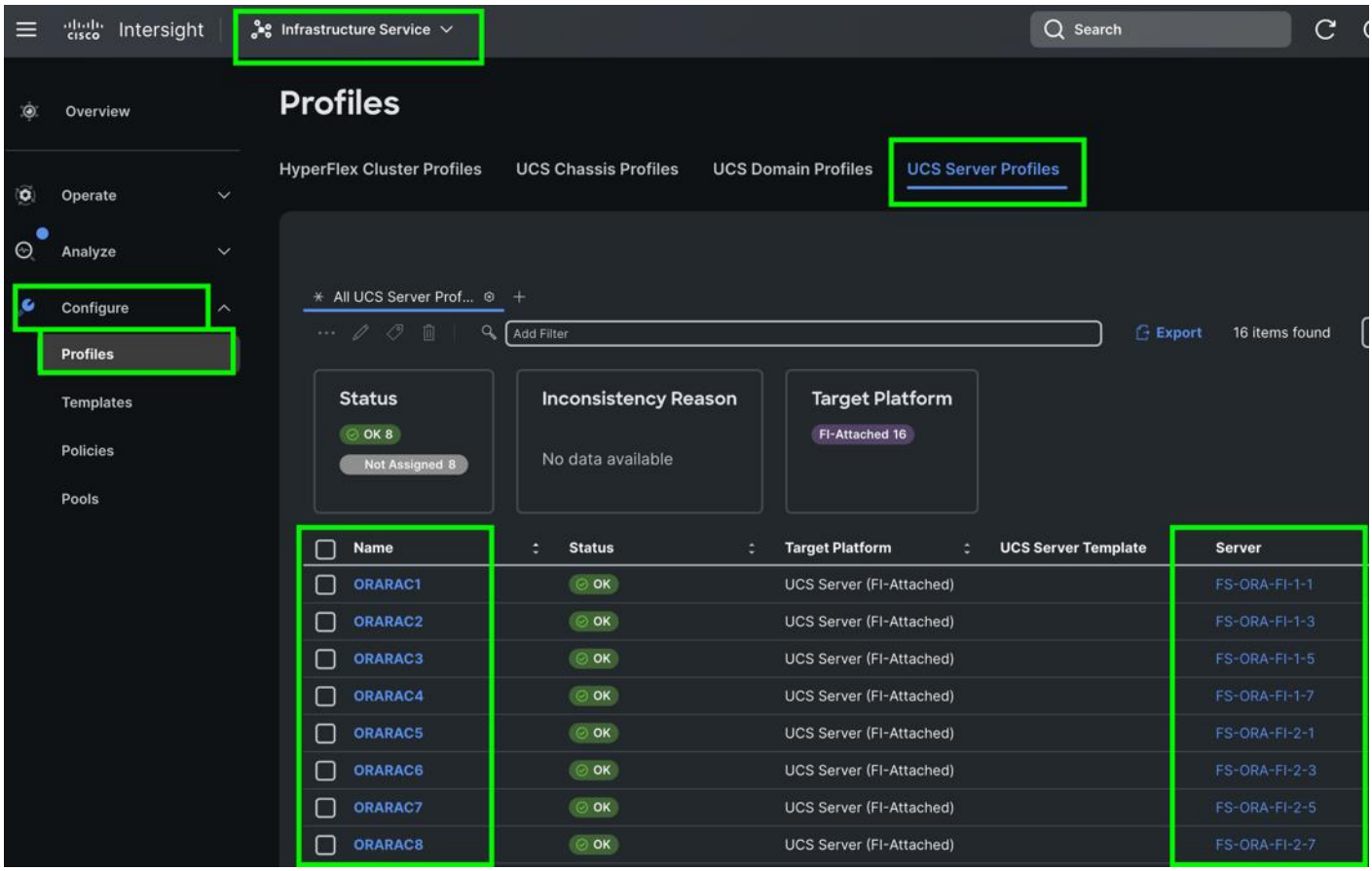
**Step 9.** To clone and create another server profile, go to Infrastructure Service > Configure > Profiles > UCS Server Profiles and Select server Profile ORARAC1 and click the radio button "---" and select option Clone as shown below:




**Step 10.** From the Clone configuration menu, select Chassis 1 Server 3 and click Next. For the Server Profile Clone Name enter "ORARAC2" and for the Organization select ORA21 to create a second server profile for the second Cisco UCS x210c M7 server on chassis 1 server 3.

**Note:** We created seven more server profile clones; ORARAC2 to ORARAC8 and assigned these cloned server profiles to all the remaining seven servers.

The following screenshot shows the server profiles with the Cisco UCS domain and assigned servers from both chassis:

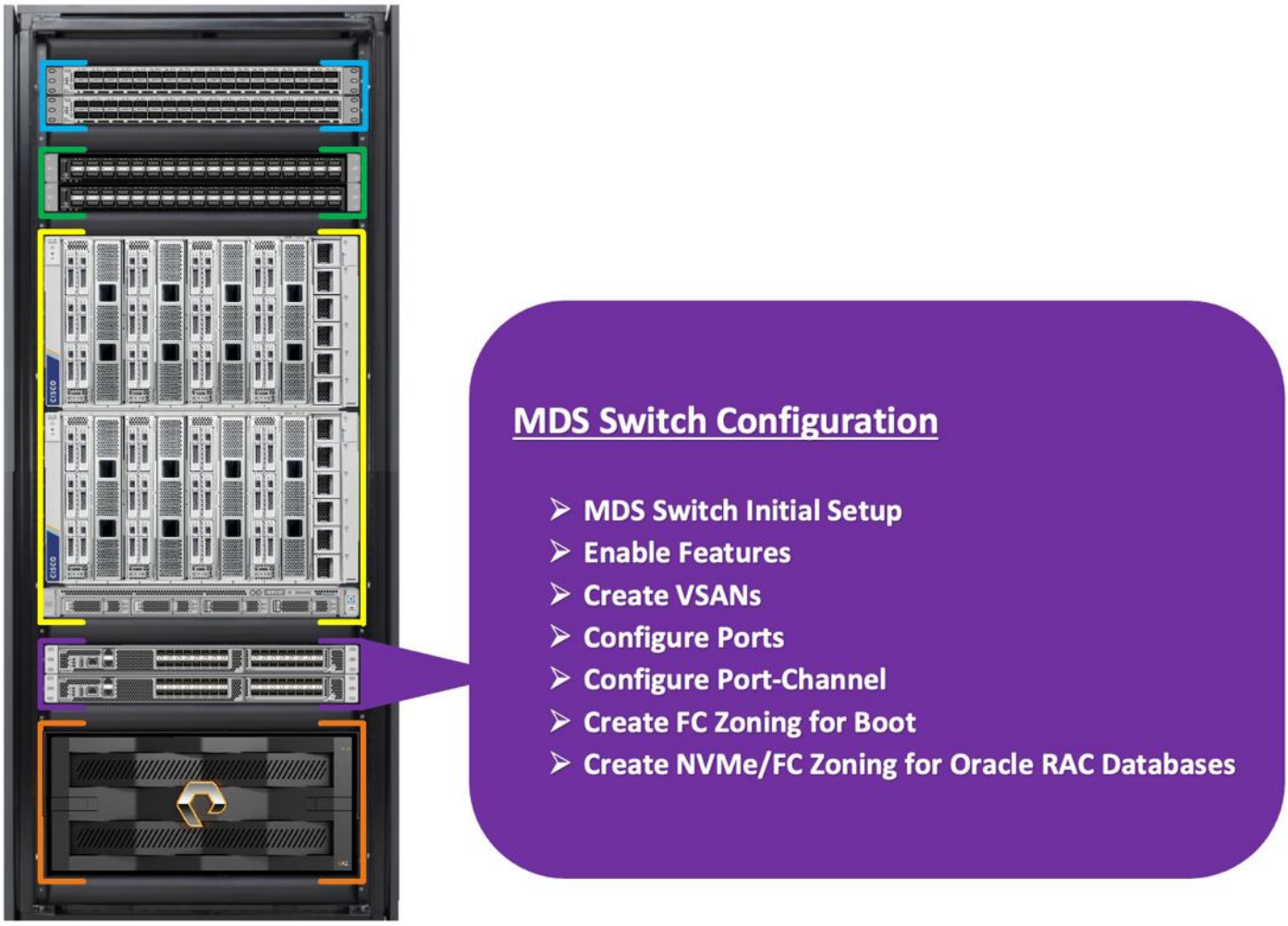



After the successful deployment of the server profile, the Cisco UCS X210c M7 Compute Nodes are configured with the parameters defined in the server profile. This completed Cisco UCS X-Series and Intersight Managed Mode (IMM) configuration can boot each server node from SAN LUN.

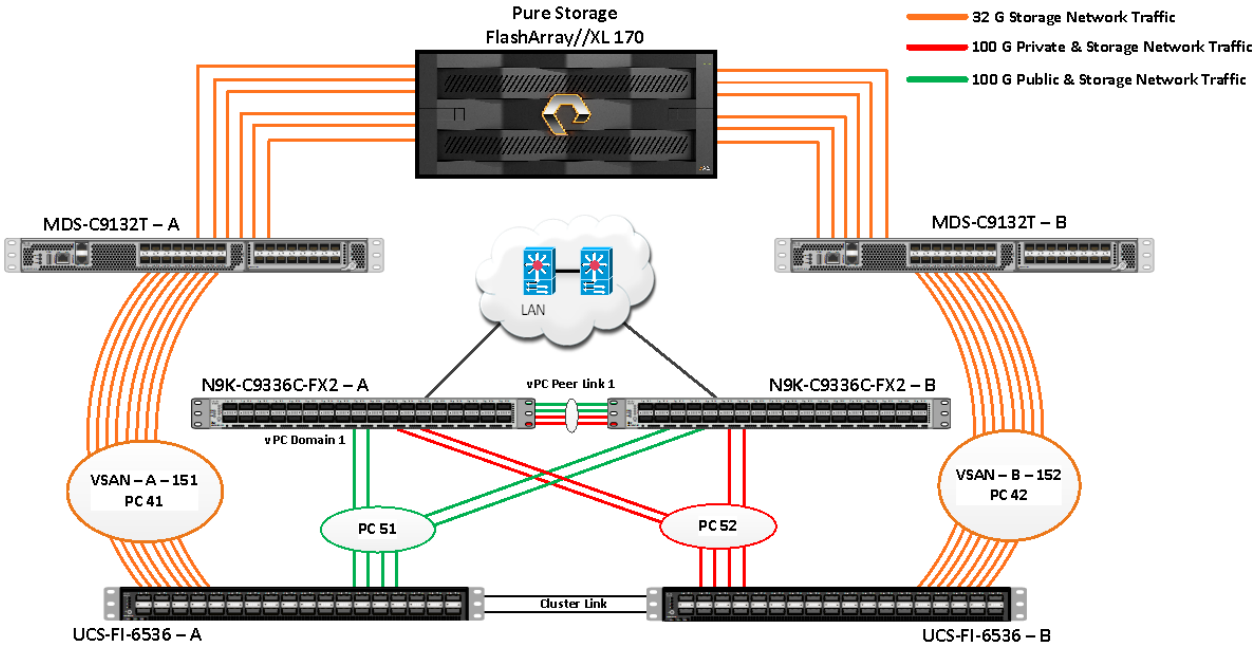
## Cisco MDS Switch Configuration

This section provides a detailed procedure for configuring the Cisco MDS 9132T Switches.

**IMPORTANT!** Follow these steps precisely because failure to do so could result in an improper configuration.




The Cisco MDS Switches are connected to the Fabric Interconnects and the Pure Storage FlashArray//XL170 System as shown below:




For this solution, eight ports (ports 1 to 8) of the MDS Switch A were connected to the Fabric Interconnect A (ports 1/35/1-4 and 1/36/1-4). The port-channel (PC 41) was configured on these ports between MDS-A to FI-A. Eight ports (ports 1 to 8) of the MDS Switch B were connected to the Fabric Interconnect B (ports 1/35/1-4 and ports 1/36/1-4). Another port-channel (PC 42) was created and on these ports were MDS-B to FI-B. All of the ports carry 32 Gb/s FC Traffic. Table 11 lists the port connectivity of Cisco MDS Switches to the Fabric Interconnects.

**Table 11. Cisco MDS Switch Port connectivity to Fabric Interconnects**

| vPC Description | vPC ID | Fabric Interconnects Ports | Cisco MDS Switch Ports | Allowed VSANs |
|---|---|---|---|---|
| Port Channel between MDS-A and FI-A | 41 | FI-A Port 1/35/1 | MDS-A-1/1 | 151 |
| | | FI-A Port 1/35/2 | MDS-A-1/2 | |
| | | FI-A Port 1/35/3 | MDS-A-1/3 | |
| | | FI-A Port 1/35/4 | MDS-A-1/4 | |
| | | FI-A Port 1/36/1 | MDS-A-1/5 | |
| | | FI-A Port 1/36/2 | MDS-A-1/6 | |
| | | FI-A Port 1/36/3 | MDS-A-1/7 | |
| | | FI-A Port 1/36/4 | MDS-A-1/8 | |
| Port Channel | 42 | FI-B Port 1/35/1 | MDS-B-1/1 | 152 |

| vPC Description | vPC ID | Fabric Interconnects Ports | Cisco MDS Switch Ports | Allowed VSANs |
|---|---|---|---|---|
| between MDS-B and FI-B | | FI-B Port 1/35/2 | MDS-B-1/2 | |
| | | FI-B Port 1/35/3 | MDS-B-1/3 | |
| | | FI-B Port 1/35/4 | MDS-B-1/4 | |
| | | FI-B Port 1/36/1 | MDS-B-1/5 | |
| | | FI-B Port 1/36/2 | MDS-B-1/6 | |
| | | FI-B Port 1/36/3 | MDS-B-1/7 | |
| | | FI-B Port 1/36/4 | MDS-B-1/8 | |

For this solution, MDS Switch A and MDS Switch B both were connected to both the Pure Storage Controllers for high availability in case of MDS or Pure storage controller failures. Six ports (ports 17 to 22) from the MDS Switch A were connected to both the Pure Storage FA//XL170 controllers CT0 and CT1. Six ports (ports 17 to 22) from the MDS Switch B were also connected to both the Pure Storage FA//XL170 controller CT0 and CT1. All ports carry 32 Gb/s FC Traffic. Table 12 lists the port connectivity of Cisco MDS Switches to both the Pure Storage FA//XL170 CT0 and CT1.

**Table 12. Cico MDS Switches port connectivity to the Pure Storage FA//XL170 Controller**

| MDS Switch | MDS Switch Port | Pure Storage FA//XL170 Controller | Pure Storage Controller Ports | Descriptions |
|---|---|---|---|---|
| MDS Switch A | FC Port 1/17 | Storage FA//XL170 CT0 | CT0.FC4 | PureFAXL170-ORA21c-CT0.FC4 |
| | FC Port 1/18 | Storage FA//XL170 CT1 | CT1.FC4 | PureFAXL170-ORA21c-CT1.FC4 |
| | FC Port 1/19 | Storage FA//XL170 CT0 | CT0.FC6 | PureFAXL170-ORA21c-CT0.FC6 |
| | FC Port 1/20 | Storage FA//XL170 CT1 | CT1.FC6 | PureFAXL170-ORA21c-CT1.FC6 |
| | FC Port 1/21 | Storage FA//XL170 CT0 | CT0.FC32 | PureFAXL170-ORA21c-CT0.FC32 |
| | FC Port 1/22 | Storage FA//XL170 CT1 | CT1.FC32 | PureFAXL170-ORA21c-CT1.FC32 |
| MDS Switch B | FC Port 1/17 | Storage FA//XL170 | CT0.FC5 | PureFAXL170-ORA21c-CT0.FC5 |

| MDS Switch | MDS Switch Port | Pure Storage FA//XL170 Controller | Pure Storage Controller Ports | Descriptions |
|---|---|---|---|---|
| | | CT0 | | |
| | FC Port 1/18 | Storage FA//XL170 CT1 | CT1.FC5 | PureFAXL170-ORA21c-CT1.FC5 |
| | FC Port 1/19 | Storage FA//XL170 CT0 | CT0.FC7 | PureFAXL170-ORA21c-CT0.FC7 |
| | FC Port 1/20 | Storage FA//XL170 CT1 | CT1.FC7 | PureFAXL170-ORA21c-CT1.FC7 |
| | FC Port 1/21 | Storage FA//XL170 CT0 | CT0.FC33 | PureFAXL170-ORA21c-CT0.FC33 |
| | FC Port 1/22 | Storage FA//XL170 CT1 | CT1.FC33 | PureFAXL170-ORA21c-CT1.FC33 |

The following procedures describe how to configure the Cisco MDS switches for use in a base FlashStack environment. These procedures assume you're using Cisco MDS 9332T FC switches.

### Cisco Feature on Cisco MDS Switches

**Procedure 1.** Configure Features

**Step 1.** Login as admin user into MDS Switch A and MDS Switch B and run the following commands:

```
config terminal
feature npiv
feature fport-channel-trunk
copy running-config startup-config
```

**Procedure 2.** Configure VSANs and Ports

**Step 1.** Login as Admin User into MDS Switch A.

**Step 2.** Create VSAN 151 for Storage network traffic and configure the ports by running the following commands:

```
config terminal
vsan database
vsan 151
vsan 151 name "VSAN-FI-A"
vsan 151 interface fc 1/1-24

interface port-channel 41
  switchport trunk allowed vsan 151
```

```
  switchport description Port-Channel-FI-A-MDS-A
  switchport rate-mode dedicated
  switchport trunk mode off
  no shut
interface fc1/1
  switchport description FS-ORA-FI-A-1/35/1
  switchport trunk mode off
  port-license acquire
  channel-group 41 force
  no shutdown
interface fc1/2
  switchport description FS-ORA-FI-A-1/35/2
  switchport trunk mode off
  port-license acquire
  channel-group 41 force
  no shutdown
interface fc1/3
  switchport description FS-ORA-FI-A-1/35/3
  switchport trunk mode off
  port-license acquire
  channel-group 41 force
  no shutdown
interface fc1/4
  switchport description FS-ORA-FI-A-1/35/4
  switchport trunk mode off
  port-license acquire
  channel-group 41 force
  no shutdown
interface fc1/5
  switchport description FS-ORA-FI-A-1/36/1
  switchport trunk mode off
  port-license acquire
  channel-group 41 force
  no shutdown
interface fc1/6
  switchport description FS-ORA-FI-A-1/36/2
  switchport trunk mode off
  port-license acquire
  channel-group 41 force
```

```
  no shutdown
interface fc1/7
  switchport description FS-ORA-FI-A-1/36/3
  switchport trunk mode off
  port-license acquire
  channel-group 41 force
  no shutdown
interface fc1/8
  switchport description FS-ORA-FI-A-1/36/4
  switchport trunk mode off
  port-license acquire
  channel-group 41 force
  no shutdown

interface fc1/17
  switchport trunk allowed vsan 151
  switchport description PureFAXL170-ORA21c-CT0.FC4
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/18
  switchport trunk allowed vsan 151
  switchport description PureFAXL170-ORA21c-CT1.FC4
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/19
  switchport trunk allowed vsan 151
  switchport description PureFAXL170-ORA21c-CT0.FC6
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/20
  switchport trunk allowed vsan 151
  switchport description PureFAXL170-ORA21c-CT1.FC6
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/21
```

```
  switchport trunk allowed vsan 151
  switchport description PureFAXL170-ORA21c-CT0.FC32
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/22
  switchport trunk allowed vsan 151
  switchport description PureFAXL170-ORA21c-CT1.FC32
  switchport trunk mode off
  port-license acquire
  no shutdown

vsan database
  vsan 151 interface port-channel 41
  vsan 151 interface fc1/17
  vsan 151 interface fc1/18
  vsan 151 interface fc1/19
  vsan 151 interface fc1/20
  vsan 151 interface fc1/21
  vsan 151 interface fc1/22



copy running-config startup-config
```

**Step 3.** Login as Admin User into MDS Switch B

**Step 4.** Create VSAN 152 for Storage network traffic and configure the ports by running the following commands:

```
config terminal
vsan database
vsan 152
vsan 152 name "VSAN-FI-B"
vsan 152 interface fc 1/1-24

interface port-channel 42
  switchport trunk allowed vsan 152
  switchport description Port-Channel-FI-B-MDS-B
  switchport rate-mode dedicated
  switchport trunk mode off
  no shut
```

```
interface fc1/1
  switchport description FS-ORA-FI-B-1/35/1
  switchport trunk mode off
  port-license acquire
  channel-group 42 force
  no shutdown
interface fc1/2
  switchport description FS-ORA-FI-B-1/35/2
  switchport trunk mode off
  port-license acquire
  channel-group 42 force
  no shutdown
interface fc1/3
  switchport description FS-ORA-FI-B-1/35/3
  switchport trunk mode off
  port-license acquire
  channel-group 42 force
  no shutdown
interface fc1/4
  switchport description FS-ORA-FI-B-1/35/4
  switchport trunk mode off
  port-license acquire
  channel-group 42 force
  no shutdown
interface fc1/5
  switchport description FS-ORA-FI-B-1/36/1
  switchport trunk mode off
  port-license acquire
  channel-group 42 force
  no shutdown
interface fc1/6
  switchport description FS-ORA-FI-B-1/36/2
  switchport trunk mode off
  port-license acquire
  channel-group 42 force
  no shutdown
interface fc1/7
  switchport description FS-ORA-FI-B-1/36/3
  switchport trunk mode off
```

```
  port-license acquire
  channel-group 42 force
  no shutdown
interface fc1/8
  switchport description FS-ORA-FI-B-1/36/4
  switchport trunk mode off
  port-license acquire
  channel-group 42 force
  no shutdown

interface fc1/17
  switchport trunk allowed vsan 152
  switchport description PureFAXL170-ORA21c-CT0.FC5
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/18
  switchport trunk allowed vsan 152
  switchport description PureFAXL170-ORA21c-CT1.FC5
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/19
  switchport trunk allowed vsan 152
  switchport description PureFAXL170-ORA21c-CT0.FC7
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/20
  switchport trunk allowed vsan 152
  switchport description PureFAXL170-ORA21c-CT1.FC7
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/21
  switchport trunk allowed vsan 152
  switchport description PureFAXL170-ORA21c-CT0.FC33
  switchport trunk mode off
  port-license acquire
```

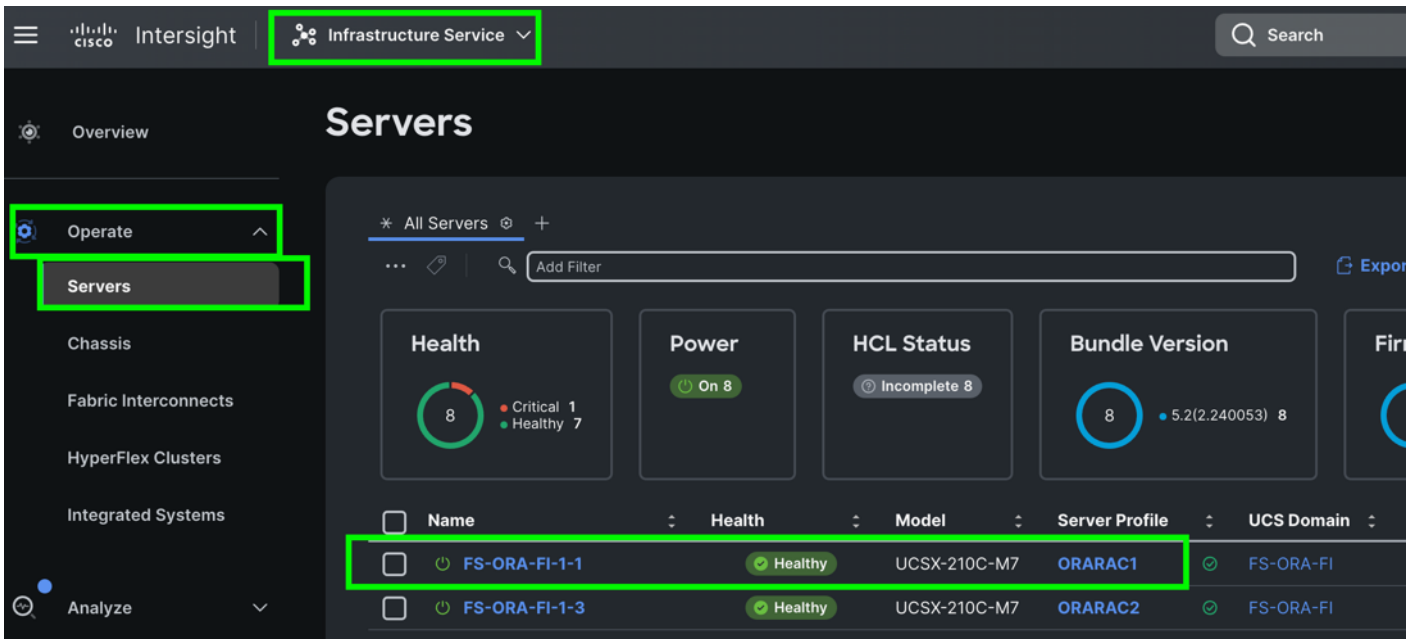
```
  no shutdown
interface fc1/22
  switchport trunk allowed vsan 152
  switchport description PureFAXL170-ORA21c-CT1.FC33
  switchport trunk mode off
  port-license acquire
  no shutdown

vsan database
  vsan 152 interface port-channel 42
  vsan 152 interface fc1/17
  vsan 152 interface fc1/18
  vsan 152 interface fc1/19
  vsan 152 interface fc1/20
  vsan 152 interface fc1/21
  vsan 152 interface fc1/22

copy running-config startup-config
```

**Procedure 3.** Create and configure Fibre Channel Zoning for FC Boot

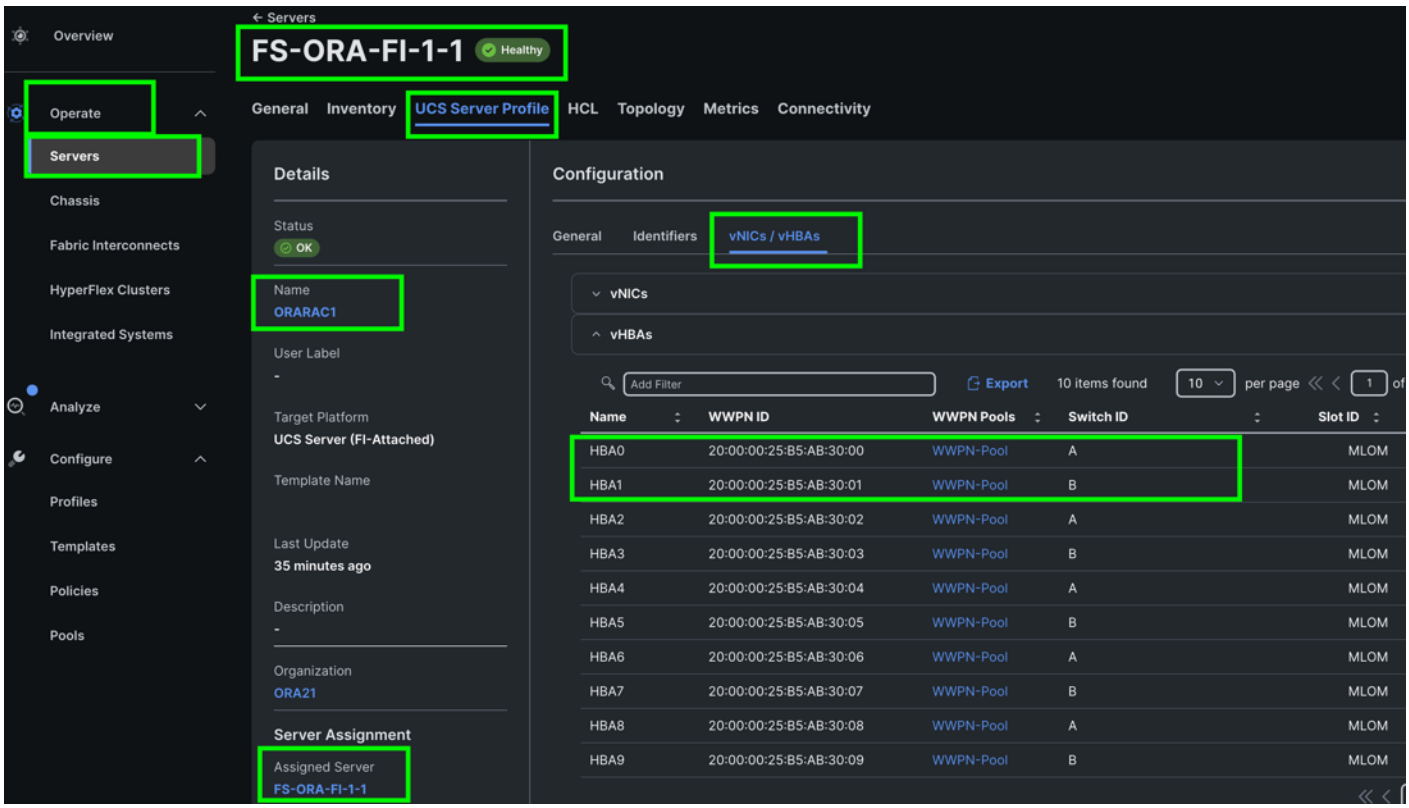
This procedure sets up the Fibre Channel connections between the Cisco MDS 9132T switches, the Cisco UCS Fabric Interconnects, and the Pure Storage systems. Before you configure the zoning details, decide how many paths are needed for each LUN and extract the WWPN numbers for each of the HBAs from each server.

For this solution, 10 vHBAs were configured on each server node. Two vHBA (HBA0 and HBA1) were created to carry the FC Network Traffic and Boot from SAN through MDS-A and MDS-B Switches. Another eight vHBAs (HBA2 to HBA9) were configured for the NVMe/FC Network Traffic (Oracle RAC Storage Traffic) through MDS-A and MDS-B Switch.

**Step 1.** Log in to Cisco Intersight and go to Infrastructure service > Operate > Servers > and click server 1 (server profile as ORARAC1).




**Step 2.** Go to the UCS Server Profile tab and select connectivity > vNICs/vHBAs to get the details of all of the HBAs and their respective WWPN ID as shown below:

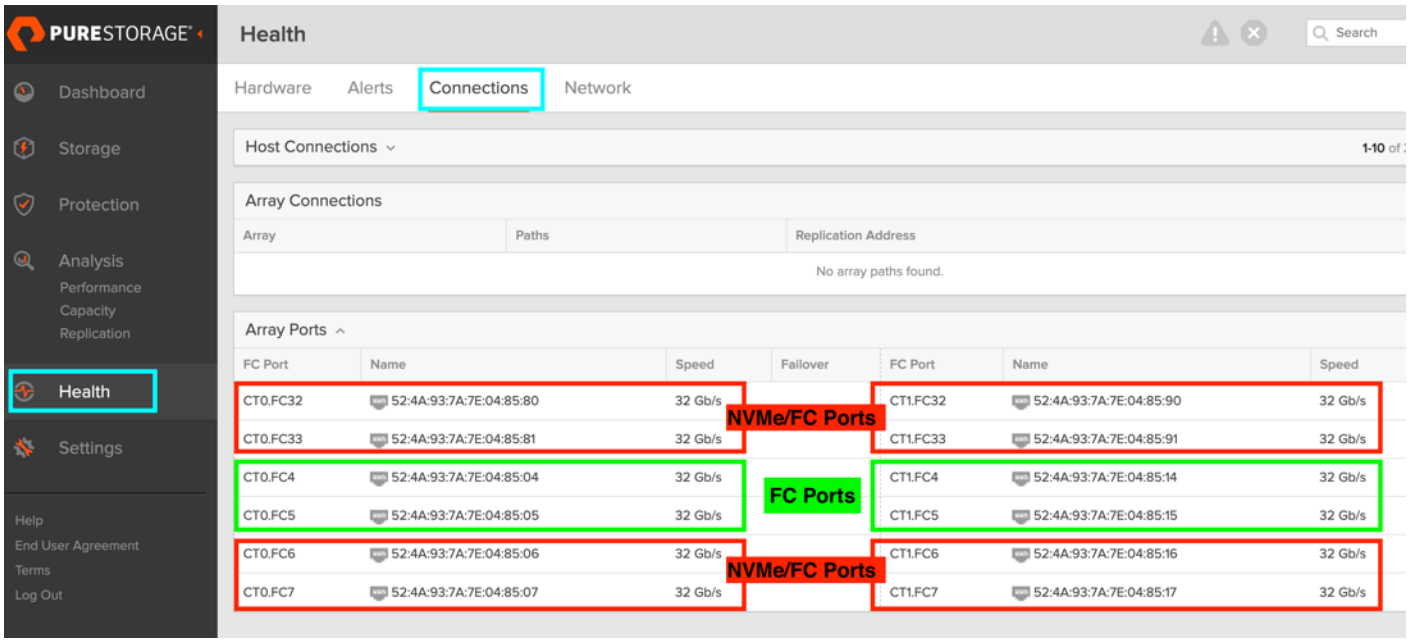



**Note:** For this solution, HBA0 (through FI-A) and HBA1 (Through FI-B) were configured for FC SAN Boot and one dedicated FC boot zone was created across both MDS switches.

**Note:** Four HBAs through FI-A (HBA2, HBA4, HBA6 and HBA8) and four HBAs through FI-B (HBA3, HBA5, HBA7 and HBA9) were configured for the NVMe FC database traffic and a dedicated NVMe FC zone was created across both MDS switches.

**Step 3.** Login into the Pure Storage controller and extract the WWPN of FC ports and verify that the port information is correct. This information can be found in the Pure Storage GUI under Health > Connections > Array Ports.

**Note:** For this solution, we used all the twelve FC ports from Pure Storage controllers. Four ports (two ports from each controller) were configured to carry "scsi-fc" services while remaining eight ports (four ports from each controller) were configured to carry "nvme-fc" services as shown in below green and red colors respectively.




For this solution, device aliases were created for zoning on MDS Switch A and Switch B as detailed below:

**Step 4.** To configure device aliases and zones for FC and NVMe/FC Network data paths on MDS switch A, complete the following steps

**Step 5.** Login as admin user and run the following commands into MDS Switch A:

```
config terminal

device-alias database
  device-alias name ORARAC-1-FC-HBA0 pwwn 20:00:00:25:b5:ab:30:00
  device-alias name ORARAC-2-FC-HBA0 pwwn 20:00:00:25:b5:ab:30:0a
  device-alias name ORARAC-3-FC-HBA0 pwwn 20:00:00:25:b5:ab:30:14
  device-alias name ORARAC-4-FC-HBA0 pwwn 20:00:00:25:b5:ab:30:1e
  device-alias name ORARAC-5-FC-HBA0 pwwn 20:00:00:25:b5:ab:30:28
  device-alias name ORARAC-6-FC-HBA0 pwwn 20:00:00:25:b5:ab:30:32
```

```
device-alias name ORARAC-7-FC-HBA0 pwwn 20:00:00:25:b5:ab:30:3c
device-alias name ORARAC-8-FC-HBA0 pwwn 20:00:00:25:b5:ab:30:46

device-alias name ORARAC1-NVMe-HBA2 pwwn 20:00:00:25:b5:ab:30:02
device-alias name ORARAC1-NVMe-HBA4 pwwn 20:00:00:25:b5:ab:30:04
device-alias name ORARAC1-NVMe-HBA6 pwwn 20:00:00:25:b5:ab:30:06
device-alias name ORARAC1-NVMe-HBA8 pwwn 20:00:00:25:b5:ab:30:08

device-alias name ORARAC2-NVMe-HBA2 pwwn 20:00:00:25:b5:ab:30:0c
device-alias name ORARAC2-NVMe-HBA4 pwwn 20:00:00:25:b5:ab:30:0e
device-alias name ORARAC2-NVMe-HBA6 pwwn 20:00:00:25:b5:ab:30:10
device-alias name ORARAC2-NVMe-HBA8 pwwn 20:00:00:25:b5:ab:30:12

device-alias name ORARAC3-NVMe-HBA2 pwwn 20:00:00:25:b5:ab:30:16
device-alias name ORARAC3-NVMe-HBA4 pwwn 20:00:00:25:b5:ab:30:18
device-alias name ORARAC3-NVMe-HBA6 pwwn 20:00:00:25:b5:ab:30:1a
device-alias name ORARAC3-NVMe-HBA8 pwwn 20:00:00:25:b5:ab:30:1c

device-alias name ORARAC4-NVMe-HBA2 pwwn 20:00:00:25:b5:ab:30:20
device-alias name ORARAC4-NVMe-HBA4 pwwn 20:00:00:25:b5:ab:30:22
device-alias name ORARAC4-NVMe-HBA6 pwwn 20:00:00:25:b5:ab:30:24
device-alias name ORARAC4-NVMe-HBA8 pwwn 20:00:00:25:b5:ab:30:26

device-alias name ORARAC5-NVMe-HBA2 pwwn 20:00:00:25:b5:ab:30:2a
device-alias name ORARAC5-NVMe-HBA4 pwwn 20:00:00:25:b5:ab:30:2c
device-alias name ORARAC5-NVMe-HBA6 pwwn 20:00:00:25:b5:ab:30:2e
device-alias name ORARAC5-NVMe-HBA8 pwwn 20:00:00:25:b5:ab:30:30

device-alias name ORARAC6-NVMe-HBA2 pwwn 20:00:00:25:b5:ab:30:34
device-alias name ORARAC6-NVMe-HBA4 pwwn 20:00:00:25:b5:ab:30:36
device-alias name ORARAC6-NVMe-HBA6 pwwn 20:00:00:25:b5:ab:30:38
device-alias name ORARAC6-NVMe-HBA8 pwwn 20:00:00:25:b5:ab:30:3a

device-alias name ORARAC7-NVMe-HBA2 pwwn 20:00:00:25:b5:ab:30:3e
device-alias name ORARAC7-NVMe-HBA4 pwwn 20:00:00:25:b5:ab:30:40
device-alias name ORARAC7-NVMe-HBA6 pwwn 20:00:00:25:b5:ab:30:42
device-alias name ORARAC7-NVMe-HBA8 pwwn 20:00:00:25:b5:ab:30:44

device-alias name ORARAC8-NVMe-HBA2 pwwn 20:00:00:25:b5:ab:30:48
```

```
  device-alias name ORARAC8-NVMe-HBA4 pwwn 20:00:00:25:b5:ab:30:4a
  device-alias name ORARAC8-NVMe-HBA6 pwwn 20:00:00:25:b5:ab:30:4c
  device-alias name ORARAC8-NVMe-HBA8 pwwn 20:00:00:25:b5:ab:30:4e

  device-alias name PureFAXL170-ORA21c-CT0-FC04 pwwn 52:4a:93:7a:7e:04:85:04
  device-alias name PureFAXL170-ORA21c-CT0-FC06 pwwn 52:4a:93:7a:7e:04:85:06
  device-alias name PureFAXL170-ORA21c-CT0-FC32 pwwn 52:4a:93:7a:7e:04:85:80
  device-alias name PureFAXL170-ORA21c-CT1-FC04 pwwn 52:4a:93:7a:7e:04:85:14
  device-alias name PureFAXL170-ORA21c-CT1-FC06 pwwn 52:4a:93:7a:7e:04:85:16
  device-alias name PureFAXL170-ORA21c-CT1-FC32 pwwn 52:4a:93:7a:7e:04:85:90

device-alias commit

copy run start
```

**Step 6.** Login as admin user and run the following commands into MDS Switch B:

```
config terminal

device-alias database

  device-alias name ORARAC-1-FC-HBA1 pwwn 20:00:00:25:b5:ab:30:01
  device-alias name ORARAC-2-FC-HBA1 pwwn 20:00:00:25:b5:ab:30:0b
  device-alias name ORARAC-3-FC-HBA1 pwwn 20:00:00:25:b5:ab:30:15
  device-alias name ORARAC-4-FC-HBA1 pwwn 20:00:00:25:b5:ab:30:1f
  device-alias name ORARAC-5-FC-HBA1 pwwn 20:00:00:25:b5:ab:30:29
  device-alias name ORARAC-6-FC-HBA1 pwwn 20:00:00:25:b5:ab:30:33
  device-alias name ORARAC-7-FC-HBA1 pwwn 20:00:00:25:b5:ab:30:3d
  device-alias name ORARAC-8-FC-HBA1 pwwn 20:00:00:25:b5:ab:30:47

  device-alias name ORARAC1-NVMe-HBA3 pwwn 20:00:00:25:b5:ab:30:03
  device-alias name ORARAC1-NVMe-HBA5 pwwn 20:00:00:25:b5:ab:30:05
  device-alias name ORARAC1-NVMe-HBA7 pwwn 20:00:00:25:b5:ab:30:07
  device-alias name ORARAC1-NVMe-HBA9 pwwn 20:00:00:25:b5:ab:30:09

  device-alias name ORARAC2-NVMe-HBA3 pwwn 20:00:00:25:b5:ab:30:0d
  device-alias name ORARAC2-NVMe-HBA5 pwwn 20:00:00:25:b5:ab:30:0f
  device-alias name ORARAC2-NVMe-HBA7 pwwn 20:00:00:25:b5:ab:30:11
  device-alias name ORARAC2-NVMe-HBA9 pwwn 20:00:00:25:b5:ab:30:13

  device-alias name ORARAC3-NVMe-HBA3 pwwn 20:00:00:25:b5:ab:30:17
```

```
  device-alias name ORARAC3-NVMe-HBA5 pwwn 20:00:00:25:b5:ab:30:19
  device-alias name ORARAC3-NVMe-HBA7 pwwn 20:00:00:25:b5:ab:30:1b
  device-alias name ORARAC3-NVMe-HBA9 pwwn 20:00:00:25:b5:ab:30:1d

  device-alias name ORARAC4-NVMe-HBA3 pwwn 20:00:00:25:b5:ab:30:21
  device-alias name ORARAC4-NVMe-HBA5 pwwn 20:00:00:25:b5:ab:30:23
  device-alias name ORARAC4-NVMe-HBA7 pwwn 20:00:00:25:b5:ab:30:25
  device-alias name ORARAC4-NVMe-HBA9 pwwn 20:00:00:25:b5:ab:30:27

  device-alias name ORARAC5-NVMe-HBA3 pwwn 20:00:00:25:b5:ab:30:2b
  device-alias name ORARAC5-NVMe-HBA5 pwwn 20:00:00:25:b5:ab:30:2d
  device-alias name ORARAC5-NVMe-HBA7 pwwn 20:00:00:25:b5:ab:30:2f
  device-alias name ORARAC5-NVMe-HBA9 pwwn 20:00:00:25:b5:ab:30:31

  device-alias name ORARAC6-NVMe-HBA3 pwwn 20:00:00:25:b5:ab:30:35
  device-alias name ORARAC6-NVMe-HBA5 pwwn 20:00:00:25:b5:ab:30:37
  device-alias name ORARAC6-NVMe-HBA7 pwwn 20:00:00:25:b5:ab:30:39
  device-alias name ORARAC6-NVMe-HBA9 pwwn 20:00:00:25:b5:ab:30:3b

  device-alias name ORARAC7-NVMe-HBA3 pwwn 20:00:00:25:b5:ab:30:3f
  device-alias name ORARAC7-NVMe-HBA5 pwwn 20:00:00:25:b5:ab:30:41
  device-alias name ORARAC7-NVMe-HBA7 pwwn 20:00:00:25:b5:ab:30:43
  device-alias name ORARAC7-NVMe-HBA9 pwwn 20:00:00:25:b5:ab:30:45

  device-alias name ORARAC8-NVMe-HBA3 pwwn 20:00:00:25:b5:ab:30:49
  device-alias name ORARAC8-NVMe-HBA5 pwwn 20:00:00:25:b5:ab:30:4b
  device-alias name ORARAC8-NVMe-HBA7 pwwn 20:00:00:25:b5:ab:30:4d
  device-alias name ORARAC8-NVMe-HBA9 pwwn 20:00:00:25:b5:ab:30:4f

  device-alias name PureFAXL170-ORA21c-CT0-FC05 pwwn 52:4a:93:7a:7e:04:85:05
  device-alias name PureFAXL170-ORA21c-CT0-FC07 pwwn 52:4a:93:7a:7e:04:85:07
  device-alias name PureFAXL170-ORA21c-CT0-FC33 pwwn 52:4a:93:7a:7e:04:85:81
  device-alias name PureFAXL170-ORA21c-CT1-FC05 pwwn 52:4a:93:7a:7e:04:85:15
  device-alias name PureFAXL170-ORA21c-CT1-FC07 pwwn 52:4a:93:7a:7e:04:85:17
  device-alias name PureFAXL170-ORA21c-CT1-FC33 pwwn 52:4a:93:7a:7e:04:85:91

device-alias commit

copy run start
```

For each of the traffic type (FC and NVMe/FC), you will create its individual zoning (FC Zoning for Boot and NVMe/FC Zoning for NVMe/FC network traffic) as explained in the following procedure.

**Procedure 4.** Create Zoning for FC SAN Boot on each node

**Step 1.** Login as admin user and run the following commands into MDS Switch A to create a zone:

```
config terminal

zone name ORARAC-1-Boot-A vsan 151
member device-alias ORARAC-1-FC-HBA0 init
member device-alias PureFAXL170-ORA21c-CT0-FC04 target
member device-alias PureFAXL170-ORA21c-CT1-FC04 target

zone name ORARAC-2-Boot-A vsan 151
member device-alias ORARAC-2-FC-HBA0 init
member device-alias PureFAXL170-ORA21c-CT0-FC04 target
member device-alias PureFAXL170-ORA21c-CT1-FC04 target

zone name ORARAC-3-Boot-A vsan 151
member device-alias ORARAC-3-FC-HBA0 init
member device-alias PureFAXL170-ORA21c-CT0-FC04 target
member device-alias PureFAXL170-ORA21c-CT1-FC04 target

zone name ORARAC-4-Boot-A vsan 151
member device-alias ORARAC-4-FC-HBA0 init
member device-alias PureFAXL170-ORA21c-CT0-FC04 target
member device-alias PureFAXL170-ORA21c-CT1-FC04 target

zone name ORARAC-5-Boot-A vsan 151
member device-alias ORARAC-5-FC-HBA0 init
member device-alias PureFAXL170-ORA21c-CT0-FC04 target
member device-alias PureFAXL170-ORA21c-CT1-FC04 target

zone name ORARAC-6-Boot-A vsan 151
member device-alias ORARAC-6-FC-HBA0 init
member device-alias PureFAXL170-ORA21c-CT0-FC04 target
member device-alias PureFAXL170-ORA21c-CT1-FC04 target

zone name ORARAC-7-Boot-A vsan 151
member device-alias ORARAC-7-FC-HBA0 init
member device-alias PureFAXL170-ORA21c-CT0-FC04 target
```

```
member device-alias PureFAXL170-ORA21c-CT1-FC04 target

zone name ORARAC-8-Boot-A vsan 151
member device-alias ORARAC-8-FC-HBA0 init
member device-alias PureFAXL170-ORA21c-CT0-FC04 target
member device-alias PureFAXL170-ORA21c-CT1-FC04 target
```

**Step 2.** Create zoneset and add all zone members:

```
config terminal
zoneset name ORARAC-A vsan 151
    member ORARAC-1-Boot-A
    member ORARAC-2-Boot-A
    member ORARAC-3-Boot-A
    member ORARAC-4-Boot-A
    member ORARAC-5-Boot-A
    member ORARAC-6-Boot-A
    member ORARAC-7-Boot-A
    member ORARAC-8-Boot-A
```

**Step 3.** Activate the zoneset and save the configuration:

```
zoneset activate name ORARAC-A vsan 151
copy run start
```

**Step 4.** Login as admin user and run the following commands into MDS Switch B to create a zone:

```
config terminal

zone name ORARAC-1-Boot-B vsan 152
member device-alias ORARAC-1-FC-HBA1 init
member device-alias PureFAXL170-ORA21c-CT0-FC05 target
member device-alias PureFAXL170-ORA21c-CT1-FC05 target

zone name ORARAC-2-Boot-B vsan 152
member device-alias ORARAC-2-FC-HBA1 init
member device-alias PureFAXL170-ORA21c-CT0-FC05 target
member device-alias PureFAXL170-ORA21c-CT1-FC05 target

zone name ORARAC-3-Boot-B vsan 152
member device-alias ORARAC-3-FC-HBA1 init
member device-alias PureFAXL170-ORA21c-CT0-FC05 target
member device-alias PureFAXL170-ORA21c-CT1-FC05 target

zone name ORARAC-4-Boot-B vsan 152
```

```
member device-alias ORARAC-4-FC-HBA1 init
member device-alias PureFAXL170-ORA21c-CT0-FC05 target
member device-alias PureFAXL170-ORA21c-CT1-FC05 target

zone name ORARAC-5-Boot-B vsan 152
member device-alias ORARAC-5-FC-HBA1 init
member device-alias PureFAXL170-ORA21c-CT0-FC05 target
member device-alias PureFAXL170-ORA21c-CT1-FC05 target

zone name ORARAC-6-Boot-B vsan 152
member device-alias ORARAC-6-FC-HBA1 init
member device-alias PureFAXL170-ORA21c-CT0-FC05 target
member device-alias PureFAXL170-ORA21c-CT1-FC05 target

zone name ORARAC-7-Boot-B vsan 152
member device-alias ORARAC-7-FC-HBA1 init
member device-alias PureFAXL170-ORA21c-CT0-FC05 target
member device-alias PureFAXL170-ORA21c-CT1-FC05 target

zone name ORARAC-8-Boot-B vsan 152
member device-alias ORARAC-8-FC-HBA1 init
member device-alias PureFAXL170-ORA21c-CT0-FC05 target
member device-alias PureFAXL170-ORA21c-CT1-FC05 target
```

**Step 5.** Create zoneset and add all zone members:

```
config terminal
zoneset name ORARAC-B vsan 152
    member ORARAC-1-Boot-B
    member ORARAC-2-Boot-B
    member ORARAC-3-Boot-B
    member ORARAC-4-Boot-B
    member ORARAC-5-Boot-B
    member ORARAC-6-Boot-B
    member ORARAC-7-Boot-B
    member ORARAC-8-Boot-B
```

**Step 6.** Activate the zoneset and save the configuration:

```
zoneset activate name ORARAC-B vsan 152
copy run start
```

**Procedure 5.** Create and Configure Zoning for NVMe FC on both Cisco MDS Switches

**Step 1.** Login as admin user and run the following commands on the MDS Switch A to create a zone:

```
config terminal


zone name ORARAC-1-NVMe-A1 vsan 151
member device-alias ORARAC1-NVMe-HBA2 init
member device-alias ORARAC1-NVMe-HBA4 init
member device-alias ORARAC1-NVMe-HBA6 init
member device-alias ORARAC1-NVMe-HBA8 init
member device-alias PureFAXL170-ORA21c-CT0-FC06 target
member device-alias PureFAXL170-ORA21c-CT1-FC06 target
member device-alias PureFAXL170-ORA21c-CT0-FC32 target
member device-alias PureFAXL170-ORA21c-CT1-FC32 target

zone name ORARAC-2-NVMe-A1 vsan 151
member device-alias ORARAC2-NVMe-HBA2 init
member device-alias ORARAC2-NVMe-HBA4 init
member device-alias ORARAC2-NVMe-HBA6 init
member device-alias ORARAC2-NVMe-HBA8 init
member device-alias PureFAXL170-ORA21c-CT0-FC06 target
member device-alias PureFAXL170-ORA21c-CT1-FC06 target
member device-alias PureFAXL170-ORA21c-CT0-FC32 target
member device-alias PureFAXL170-ORA21c-CT1-FC32 target

zone name ORARAC-3-NVMe-A1 vsan 151
member device-alias ORARAC3-NVMe-HBA2 init
member device-alias ORARAC3-NVMe-HBA4 init
member device-alias ORARAC3-NVMe-HBA6 init
member device-alias ORARAC3-NVMe-HBA8 init
member device-alias PureFAXL170-ORA21c-CT0-FC06 target
member device-alias PureFAXL170-ORA21c-CT1-FC06 target
member device-alias PureFAXL170-ORA21c-CT0-FC32 target
member device-alias PureFAXL170-ORA21c-CT1-FC32 target

zone name ORARAC-4-NVMe-A1 vsan 151
member device-alias ORARAC4-NVMe-HBA2 init
member device-alias ORARAC4-NVMe-HBA4 init
member device-alias ORARAC4-NVMe-HBA6 init
member device-alias ORARAC4-NVMe-HBA8 init
```

```
member device-alias PureFAXL170-ORA21c-CT0-FC06 target
member device-alias PureFAXL170-ORA21c-CT1-FC06 target
member device-alias PureFAXL170-ORA21c-CT0-FC32 target
member device-alias PureFAXL170-ORA21c-CT1-FC32 target

zone name ORARAC-5-NVMe-A1 vsan 151
member device-alias ORARAC5-NVMe-HBA2 init
member device-alias ORARAC5-NVMe-HBA4 init
member device-alias ORARAC5-NVMe-HBA6 init
member device-alias ORARAC5-NVMe-HBA8 init
member device-alias PureFAXL170-ORA21c-CT0-FC06 target
member device-alias PureFAXL170-ORA21c-CT1-FC06 target
member device-alias PureFAXL170-ORA21c-CT0-FC32 target
member device-alias PureFAXL170-ORA21c-CT1-FC32 target

zone name ORARAC-6-NVMe-A1 vsan 151
member device-alias ORARAC6-NVMe-HBA2 init
member device-alias ORARAC6-NVMe-HBA4 init
member device-alias ORARAC6-NVMe-HBA6 init
member device-alias ORARAC6-NVMe-HBA8 init
member device-alias PureFAXL170-ORA21c-CT0-FC06 target
member device-alias PureFAXL170-ORA21c-CT1-FC06 target
member device-alias PureFAXL170-ORA21c-CT0-FC32 target
member device-alias PureFAXL170-ORA21c-CT1-FC32 target

zone name ORARAC-7-NVMe-A1 vsan 151
member device-alias ORARAC7-NVMe-HBA2 init
member device-alias ORARAC7-NVMe-HBA4 init
member device-alias ORARAC7-NVMe-HBA6 init
member device-alias ORARAC7-NVMe-HBA8 init
member device-alias PureFAXL170-ORA21c-CT0-FC06 target
member device-alias PureFAXL170-ORA21c-CT1-FC06 target
member device-alias PureFAXL170-ORA21c-CT0-FC32 target
member device-alias PureFAXL170-ORA21c-CT1-FC32 target

zone name ORARAC-8-NVMe-A1 vsan 151
member device-alias ORARAC8-NVMe-HBA2 init
member device-alias ORARAC8-NVMe-HBA4 init
member device-alias ORARAC8-NVMe-HBA6 init
```

```
member device-alias ORARAC8-NVMe-HBA8 init
member device-alias PureFAXL170-ORA21c-CT0-FC06 target
member device-alias PureFAXL170-ORA21c-CT1-FC06 target
member device-alias PureFAXL170-ORA21c-CT0-FC32 target
member device-alias PureFAXL170-ORA21c-CT1-FC32 target
```

**Step 2.** Create a zoneset and add all zone members:

```
config terminal
zoneset name ORARAC-A vsan 151
    member ORARAC-1-NVMe-A1
    member ORARAC-2-NVMe-A1
    member ORARAC-3-NVMe-A1
    member ORARAC-4-NVMe-A1
    member ORARAC-5-NVMe-A1
    member ORARAC-6-NVMe-A1
    member ORARAC-7-NVMe-A1
    member ORARAC-8-NVMe-A1
```

**Step 3.** Activate the zoneset and save the configuration:

```
zoneset activate name ORARAC-A vsan 151
copy run start
```

**Step 4.** Login as admin user and run the following commands on the MDS Switch B to create a zone:

```
config terminal

zone name ORARAC-1-NVMe-B1 vsan 152
member device-alias ORARAC1-NVMe-HBA3 init
member device-alias ORARAC1-NVMe-HBA5 init
member device-alias ORARAC1-NVMe-HBA7 init
member device-alias ORARAC1-NVMe-HBA9 init
member device-alias PureFAXL170-ORA21c-CT0-FC07 target
member device-alias PureFAXL170-ORA21c-CT1-FC07 target
member device-alias PureFAXL170-ORA21c-CT0-FC33 target
member device-alias PureFAXL170-ORA21c-CT1-FC33 target

zone name ORARAC-2-NVMe-B1 vsan 152
member device-alias ORARAC2-NVMe-HBA3 init
member device-alias ORARAC2-NVMe-HBA5 init
member device-alias ORARAC2-NVMe-HBA7 init
member device-alias ORARAC2-NVMe-HBA9 init
member device-alias PureFAXL170-ORA21c-CT0-FC07 target
member device-alias PureFAXL170-ORA21c-CT1-FC07 target
```

```
member device-alias PureFAXL170-ORA21c-CT0-FC33 target
member device-alias PureFAXL170-ORA21c-CT1-FC33 target

zone name ORARAC-3-NVMe-B1 vsan 152
member device-alias ORARAC3-NVMe-HBA3 init
member device-alias ORARAC3-NVMe-HBA5 init
member device-alias ORARAC3-NVMe-HBA7 init
member device-alias ORARAC3-NVMe-HBA9 init
member device-alias PureFAXL170-ORA21c-CT0-FC07 target
member device-alias PureFAXL170-ORA21c-CT1-FC07 target
member device-alias PureFAXL170-ORA21c-CT0-FC33 target
member device-alias PureFAXL170-ORA21c-CT1-FC33 target

zone name ORARAC-4-NVMe-B1 vsan 152
member device-alias ORARAC4-NVMe-HBA3 init
member device-alias ORARAC4-NVMe-HBA5 init
member device-alias ORARAC4-NVMe-HBA7 init
member device-alias ORARAC4-NVMe-HBA9 init
member device-alias PureFAXL170-ORA21c-CT0-FC07 target
member device-alias PureFAXL170-ORA21c-CT1-FC07 target
member device-alias PureFAXL170-ORA21c-CT0-FC33 target
member device-alias PureFAXL170-ORA21c-CT1-FC33 target

zone name ORARAC-5-NVMe-B1 vsan 152
member device-alias ORARAC5-NVMe-HBA3 init
member device-alias ORARAC5-NVMe-HBA5 init
member device-alias ORARAC5-NVMe-HBA7 init
member device-alias ORARAC5-NVMe-HBA9 init
member device-alias PureFAXL170-ORA21c-CT0-FC07 target
member device-alias PureFAXL170-ORA21c-CT1-FC07 target
member device-alias PureFAXL170-ORA21c-CT0-FC33 target
member device-alias PureFAXL170-ORA21c-CT1-FC33 target

zone name ORARAC-6-NVMe-B1 vsan 152
member device-alias ORARAC6-NVMe-HBA3 init
member device-alias ORARAC6-NVMe-HBA5 init
member device-alias ORARAC6-NVMe-HBA7 init
member device-alias ORARAC6-NVMe-HBA9 init
member device-alias PureFAXL170-ORA21c-CT0-FC07 target
```

```
member device-alias PureFAXL170-ORA21c-CT1-FC07 target
member device-alias PureFAXL170-ORA21c-CT0-FC33 target
member device-alias PureFAXL170-ORA21c-CT1-FC33 target

zone name ORARAC-7-NVMe-B1 vsan 152
member device-alias ORARAC7-NVMe-HBA3 init
member device-alias ORARAC7-NVMe-HBA5 init
member device-alias ORARAC7-NVMe-HBA7 init
member device-alias ORARAC7-NVMe-HBA9 init
member device-alias PureFAXL170-ORA21c-CT0-FC07 target
member device-alias PureFAXL170-ORA21c-CT1-FC07 target
member device-alias PureFAXL170-ORA21c-CT0-FC33 target
member device-alias PureFAXL170-ORA21c-CT1-FC33 target

zone name ORARAC-8-NVMe-B1 vsan 152
member device-alias ORARAC8-NVMe-HBA3 init
member device-alias ORARAC8-NVMe-HBA5 init
member device-alias ORARAC8-NVMe-HBA7 init
member device-alias ORARAC8-NVMe-HBA9 init
member device-alias PureFAXL170-ORA21c-CT0-FC07 target
member device-alias PureFAXL170-ORA21c-CT1-FC07 target
member device-alias PureFAXL170-ORA21c-CT0-FC33 target
member device-alias PureFAXL170-ORA21c-CT1-FC33 target
```

**Step 5.** Create a zoneset and add all zone members:

```
config terminal
zoneset name ORARAC-B vsan 152
    member ORARAC-1-NVMe-B1
    member ORARAC-2-NVMe-B1
    member ORARAC-3-NVMe-B1
    member ORARAC-4-NVMe-B1
    member ORARAC-5-NVMe-B1
    member ORARAC-6-NVMe-B1
    member ORARAC-7-NVMe-B1
    member ORARAC-8-NVMe-B1
```

**Step 6.** Activate the zoneset and save the configuration:

```
zoneset activate name ORARAC-B vsan 152
copy run start
```

**Procedure 6.** Verify FC ports on MDS Switch A and MDS Switch B

**Step 1.** Login as admin user into MDS Switch A and verify all “flogi” by running “show flogi database vsan 151” as shown below:

```
FS-ORA-MDS-A# show flogi database vsan 151
--------------------------------------------------------------------------------
INTERFACE        VSAN    FCID           PORT NAME               NODE NAME
--------------------------------------------------------------------------------
fc1/17           151   0xd40081  52:4a:93:7a:7e:04:85:04 52:4a:93:7a:7e:04:85:04
                                   [PureFAXL170-ORA21c-CT0-FC04]
fc1/18           151   0xd40041  52:4a:93:7a:7e:04:85:14 52:4a:93:7a:7e:04:85:14
                                   [PureFAXL170-ORA21c-CT1-FC04]
fc1/19           151   0xd40042  52:4a:93:7a:7e:04:85:06 52:4a:93:7a:7e:04:85:06
                                   [PureFAXL170-ORA21c-CT0-FC06]
fc1/20           151   0xd40022  52:4a:93:7a:7e:04:85:16 52:4a:93:7a:7e:04:85:16
                                   [PureFAXL170-ORA21c-CT1-FC06]
fc1/21           151   0xd40061  52:4a:93:7a:7e:04:85:80 52:4a:93:7a:7e:04:85:80
                                   [PureFAXL170-ORA21c-CT0-FC32]
fc1/22           151   0xd40021  52:4a:93:7a:7e:04:85:90 52:4a:93:7a:7e:04:85:90
                                   [PureFAXL170-ORA21c-CT1-FC32]
port-channel41   151   0xd40000  24:29:00:08:31:07:e2:00 20:97:00:08:31:07:e2:01
port-channel41   151   0xd40001  20:00:00:25:b5:ab:30:00 20:00:00:25:b5:13:50:00
                                   [ORARAC-1-FC-HBA0]
port-channel41   151   0xd40002  20:00:00:25:b5:ab:30:02 20:00:00:25:b5:13:50:00
                                   [ORARAC1-NVMe-HBA2]
port-channel41   151   0xd40003  20:00:00:25:b5:ab:30:04 20:00:00:25:b5:13:50:00
                                   [ORARAC1-NVMe-HBA4]
port-channel41   151   0xd40004  20:00:00:25:b5:ab:30:06 20:00:00:25:b5:13:50:00
                                   [ORARAC1-NVMe-HBA6]
port-channel41   151   0xd40005  20:00:00:25:b5:ab:30:08 20:00:00:25:b5:13:50:00
                                   [ORARAC1-NVMe-HBA8]
port-channel41   151   0xd40006  20:00:00:25:b5:ab:30:1e 20:00:00:25:b5:13:50:03
                                   [ORARAC-4-FC-HBA0]
port-channel41   151   0xd40007  20:00:00:25:b5:ab:30:0a 20:00:00:25:b5:13:50:01
                                   [ORARAC-2-FC-HBA0]
port-channel41   151   0xd40008  20:00:00:25:b5:ab:30:14 20:00:00:25:b5:13:50:02
                                   [ORARAC-3-FC-HBA0]
port-channel41   151   0xd40009  20:00:00:25:b5:ab:30:0c 20:00:00:25:b5:13:50:01
                                   [ORARAC2-NVMe-HBA2]
port-channel41   151   0xd4000a  20:00:00:25:b5:ab:30:0e 20:00:00:25:b5:13:50:01
```

```
                                    [ORARAC2-NVMe-HBA4]
port-channel41   151   0xd4000b  20:00:00:25:b5:ab:30:10 20:00:00:25:b5:13:50:01
                                    [ORARAC2-NVMe-HBA6]
port-channel41   151   0xd4000c  20:00:00:25:b5:ab:30:12 20:00:00:25:b5:13:50:01
                                    [ORARAC2-NVMe-HBA8]
port-channel41   151   0xd4000d  20:00:00:25:b5:ab:30:16 20:00:00:25:b5:13:50:02
                                    [ORARAC3-NVMe-HBA2]
port-channel41   151   0xd4000e  20:00:00:25:b5:ab:30:18 20:00:00:25:b5:13:50:02
                                    [ORARAC3-NVMe-HBA4]
port-channel41   151   0xd4000f  20:00:00:25:b5:ab:30:1a 20:00:00:25:b5:13:50:02
                                    [ORARAC3-NVMe-HBA6]
port-channel41   151   0xd40010  20:00:00:25:b5:ab:30:1c 20:00:00:25:b5:13:50:02
                                    [ORARAC3-NVMe-HBA8]
port-channel41   151   0xd40011  20:00:00:25:b5:ab:30:20 20:00:00:25:b5:13:50:03
                                    [ORARAC4-NVMe-HBA2]
port-channel41   151   0xd40012  20:00:00:25:b5:ab:30:22 20:00:00:25:b5:13:50:03
                                    [ORARAC4-NVMe-HBA4]
port-channel41   151   0xd40013  20:00:00:25:b5:ab:30:24 20:00:00:25:b5:13:50:03
                                    [ORARAC4-NVMe-HBA6]
port-channel41   151   0xd40014  20:00:00:25:b5:ab:30:26 20:00:00:25:b5:13:50:03
                                    [ORARAC4-NVMe-HBA8]
port-channel41   151   0xd40015  20:00:00:25:b5:ab:30:28 20:00:00:25:b5:13:50:04
                                    [ORARAC-5-FC-HBA0]
port-channel41   151   0xd40016  20:00:00:25:b5:ab:30:32 20:00:00:25:b5:13:50:05
                                    [ORARAC-6-FC-HBA0]
port-channel41   151   0xd40017  20:00:00:25:b5:ab:30:46 20:00:00:25:b5:13:50:07
                                    [ORARAC-8-FC-HBA0]
port-channel41   151   0xd40018  20:00:00:25:b5:ab:30:3c 20:00:00:25:b5:13:50:06
                                    [ORARAC-7-FC-HBA0]
port-channel41   151   0xd40019  20:00:00:25:b5:ab:30:2a 20:00:00:25:b5:13:50:04
                                    [ORARAC5-NVMe-HBA2]
port-channel41   151   0xd4001a  20:00:00:25:b5:ab:30:2c 20:00:00:25:b5:13:50:04
                                    [ORARAC5-NVMe-HBA4]
port-channel41   151   0xd4001b  20:00:00:25:b5:ab:30:2e 20:00:00:25:b5:13:50:04
                                    [ORARAC5-NVMe-HBA6]
port-channel41   151   0xd4001c  20:00:00:25:b5:ab:30:30 20:00:00:25:b5:13:50:04
                                    [ORARAC5-NVMe-HBA8]
port-channel41   151   0xd4001d  20:00:00:25:b5:ab:30:34 20:00:00:25:b5:13:50:05
                                    [ORARAC6-NVMe-HBA2]
```

```
port-channel41   151   0xd4001e  20:00:00:25:b5:ab:30:36 20:00:00:25:b5:13:50:05
                                 [ORARAC6-NVMe-HBA4]
port-channel41   151   0xd4001f  20:00:00:25:b5:ab:30:38 20:00:00:25:b5:13:50:05
                                 [ORARAC6-NVMe-HBA6]
port-channel41   151   0xd400a0  20:00:00:25:b5:ab:30:3a 20:00:00:25:b5:13:50:05
                                 [ORARAC6-NVMe-HBA8]
port-channel41   151   0xd400a1  20:00:00:25:b5:ab:30:3e 20:00:00:25:b5:13:50:06
                                 [ORARAC7-NVMe-HBA2]
port-channel41   151   0xd400a2  20:00:00:25:b5:ab:30:40 20:00:00:25:b5:13:50:06
                                 [ORARAC7-NVMe-HBA4]
port-channel41   151   0xd400a3  20:00:00:25:b5:ab:30:42 20:00:00:25:b5:13:50:06
                                 [ORARAC7-NVMe-HBA6]
port-channel41   151   0xd400a4  20:00:00:25:b5:ab:30:44 20:00:00:25:b5:13:50:06
                                 [ORARAC7-NVMe-HBA8]
port-channel41   151   0xd400a5  20:00:00:25:b5:ab:30:48 20:00:00:25:b5:13:50:07
                                 [ORARAC8-NVMe-HBA2]
port-channel41   151   0xd400a6  20:00:00:25:b5:ab:30:4a 20:00:00:25:b5:13:50:07
                                 [ORARAC8-NVMe-HBA4]
port-channel41   151   0xd400a7  20:00:00:25:b5:ab:30:4c 20:00:00:25:b5:13:50:07
                                 [ORARAC8-NVMe-HBA6]
port-channel41   151   0xd400a8  20:00:00:25:b5:ab:30:4e 20:00:00:25:b5:13:50:07
                                 [ORARAC8-NVMe-HBA8]
Total number of flogi = 47.
```

**Step 2.** Login as admin user into MDS Switch B and verify all “flogi” by running “show flogi database vsan 152” as shown below:

```
FS-ORA-MDS-B# show flogi database vsan 152
--------------------------------------------------------------------------------
INTERFACE        VSAN    FCID           PORT NAME               NODE NAME
--------------------------------------------------------------------------------
fc1/17           152   0xc70042  52:4a:93:7a:7e:04:85:05 52:4a:93:7a:7e:04:85:05
                                 [PureFAXL170-ORA21c-CT0-FC05]
fc1/18           152   0xc70022  52:4a:93:7a:7e:04:85:15 52:4a:93:7a:7e:04:85:15
                                 [PureFAXL170-ORA21c-CT1-FC05]
fc1/19           152   0xc700a0  52:4a:93:7a:7e:04:85:07 52:4a:93:7a:7e:04:85:07
                                 [PureFAXL170-ORA21c-CT0-FC07]
fc1/20           152   0xc70062  52:4a:93:7a:7e:04:85:17 52:4a:93:7a:7e:04:85:17
                                 [PureFAXL170-ORA21c-CT1-FC07]
fc1/21           152   0xc70001  52:4a:93:7a:7e:04:85:81 52:4a:93:7a:7e:04:85:81
```

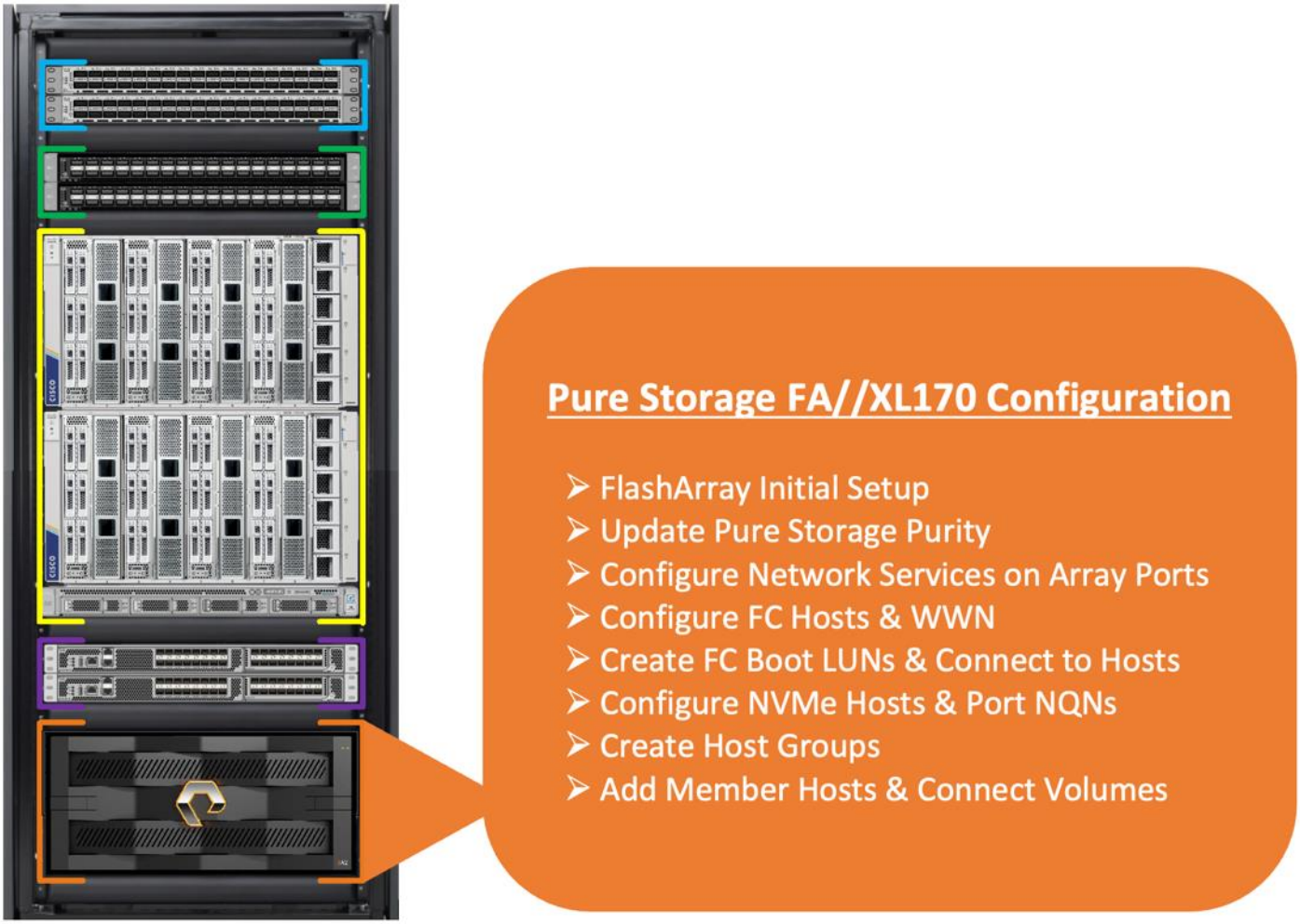
```
                                    [PureFAXL170-ORA21c-CT0-FC33]
fc1/22          152   0xc70061  52:4a:93:7a:7e:04:85:91 52:4a:93:7a:7e:04:85:91
                                    [PureFAXL170-ORA21c-CT1-FC33]
port-channel42  152   0xc70080  24:2a:00:08:31:0f:4d:64 20:98:00:08:31:0f:4d:65
port-channel42  152   0xc70081  20:00:00:25:b5:ab:30:01 20:00:00:25:b5:13:50:00
                                    [ORARAC-1-FC-HBA1]
port-channel42  152   0xc70082  20:00:00:25:b5:ab:30:03 20:00:00:25:b5:13:50:00
                                    [ORARAC1-NVMe-HBA3]
port-channel42  152   0xc70083  20:00:00:25:b5:ab:30:05 20:00:00:25:b5:13:50:00
                                    [ORARAC1-NVMe-HBA5]
port-channel42  152   0xc70084  20:00:00:25:b5:ab:30:07 20:00:00:25:b5:13:50:00
                                    [ORARAC1-NVMe-HBA7]
port-channel42  152   0xc70085  20:00:00:25:b5:ab:30:09 20:00:00:25:b5:13:50:00
                                    [ORARAC1-NVMe-HBA9]
port-channel42  152   0xc70086  20:00:00:25:b5:ab:30:1f 20:00:00:25:b5:13:50:03
                                    [ORARAC-4-FC-HBA1]
port-channel42  152   0xc70087  20:00:00:25:b5:ab:30:0b 20:00:00:25:b5:13:50:01
                                    [ORARAC-2-FC-HBA1]
port-channel42  152   0xc70088  20:00:00:25:b5:ab:30:15 20:00:00:25:b5:13:50:02
                                    [ORARAC-3-FC-HBA1]
port-channel42  152   0xc70089  20:00:00:25:b5:ab:30:0d 20:00:00:25:b5:13:50:01
                                    [ORARAC2-NVMe-HBA3]
port-channel42  152   0xc7008a  20:00:00:25:b5:ab:30:0f 20:00:00:25:b5:13:50:01
                                    [ORARAC2-NVMe-HBA5]
port-channel42  152   0xc7008b  20:00:00:25:b5:ab:30:11 20:00:00:25:b5:13:50:01
                                    [ORARAC2-NVMe-HBA7]
port-channel42  152   0xc7008c  20:00:00:25:b5:ab:30:13 20:00:00:25:b5:13:50:01
                                    [ORARAC2-NVMe-HBA9]
port-channel42  152   0xc7008d  20:00:00:25:b5:ab:30:17 20:00:00:25:b5:13:50:02
                                    [ORARAC3-NVMe-HBA3]
port-channel42  152   0xc7008e  20:00:00:25:b5:ab:30:19 20:00:00:25:b5:13:50:02
                                    [ORARAC3-NVMe-HBA5]
port-channel42  152   0xc7008f  20:00:00:25:b5:ab:30:1b 20:00:00:25:b5:13:50:02
                                    [ORARAC3-NVMe-HBA7]
port-channel42  152   0xc70090  20:00:00:25:b5:ab:30:1d 20:00:00:25:b5:13:50:02
                                    [ORARAC3-NVMe-HBA9]
port-channel42  152   0xc70091  20:00:00:25:b5:ab:30:21 20:00:00:25:b5:13:50:03
                                    [ORARAC4-NVMe-HBA3]
port-channel42  152   0xc70092  20:00:00:25:b5:ab:30:23 20:00:00:25:b5:13:50:03
```

```
                                    [ORARAC4-NVMe-HBA5]
port-channel42   152   0xc70093  20:00:00:25:b5:ab:30:25 20:00:00:25:b5:13:50:03
                                    [ORARAC4-NVMe-HBA7]
port-channel42   152   0xc70094  20:00:00:25:b5:ab:30:27 20:00:00:25:b5:13:50:03
                                    [ORARAC4-NVMe-HBA9]
port-channel42   152   0xc70095  20:00:00:25:b5:ab:30:29 20:00:00:25:b5:13:50:04
                                    [ORARAC-5-FC-HBA1]
port-channel42   152   0xc70096  20:00:00:25:b5:ab:30:33 20:00:00:25:b5:13:50:05
                                    [ORARAC-6-FC-HBA1]
port-channel42   152   0xc70097  20:00:00:25:b5:ab:30:47 20:00:00:25:b5:13:50:07
                                    [ORARAC-8-FC-HBA1]
port-channel42   152   0xc70098  20:00:00:25:b5:ab:30:3d 20:00:00:25:b5:13:50:06
                                    [ORARAC-7-FC-HBA1]
port-channel42   152   0xc70099  20:00:00:25:b5:ab:30:2b 20:00:00:25:b5:13:50:04
                                    [ORARAC5-NVMe-HBA3]
port-channel42   152   0xc7009a  20:00:00:25:b5:ab:30:2d 20:00:00:25:b5:13:50:04
                                    [ORARAC5-NVMe-HBA5]
port-channel42   152   0xc7009b  20:00:00:25:b5:ab:30:2f 20:00:00:25:b5:13:50:04
                                    [ORARAC5-NVMe-HBA7]
port-channel42   152   0xc7009c  20:00:00:25:b5:ab:30:31 20:00:00:25:b5:13:50:04
                                    [ORARAC5-NVMe-HBA9]
port-channel42   152   0xc7009d  20:00:00:25:b5:ab:30:35 20:00:00:25:b5:13:50:05
                                    [ORARAC6-NVMe-HBA3]
port-channel42   152   0xc7009e  20:00:00:25:b5:ab:30:37 20:00:00:25:b5:13:50:05
                                    [ORARAC6-NVMe-HBA5]
port-channel42   152   0xc7009f  20:00:00:25:b5:ab:30:39 20:00:00:25:b5:13:50:05
                                    [ORARAC6-NVMe-HBA7]
port-channel42   152   0xc700c0  20:00:00:25:b5:ab:30:3b 20:00:00:25:b5:13:50:05
                                    [ORARAC6-NVMe-HBA9]
port-channel42   152   0xc700c1  20:00:00:25:b5:ab:30:3f 20:00:00:25:b5:13:50:06
                                    [ORARAC7-NVMe-HBA3]
port-channel42   152   0xc700c2  20:00:00:25:b5:ab:30:41 20:00:00:25:b5:13:50:06
                                    [ORARAC7-NVMe-HBA5]
port-channel42   152   0xc700c3  20:00:00:25:b5:ab:30:43 20:00:00:25:b5:13:50:06
                                    [ORARAC7-NVMe-HBA7]
port-channel42   152   0xc700c4  20:00:00:25:b5:ab:30:45 20:00:00:25:b5:13:50:06
                                    [ORARAC7-NVMe-HBA9]
port-channel42   152   0xc700c5  20:00:00:25:b5:ab:30:49 20:00:00:25:b5:13:50:07
                                    [ORARAC8-NVMe-HBA3]
```

```
port-channel42    152    0xc700c6  20:00:00:25:b5:ab:30:4b 20:00:00:25:b5:13:50:07
                                   [ORARAC8-NVMe-HBA5]
port-channel42    152    0xc700c7  20:00:00:25:b5:ab:30:4d 20:00:00:25:b5:13:50:07
                                   [ORARAC8-NVMe-HBA7]
port-channel42    152    0xc700c8  20:00:00:25:b5:ab:30:4f 20:00:00:25:b5:13:50:07
                                   [ORARAC8-NVMe-HBA9]
Total number of flogi = 47.
```

## Pure Storage FlashArray//XL170 Configuration

This section details the high-level steps to configure the Pure Storage for this solution.




### Pure Storage Connectivity

It is beyond the scope of this document to explain the detailed information about the Pure storage installation. For detailed information, see: https://www.purestorage.com/content/dam/pdf/en/datasheets/ds-flasharray-xl.pdf and for information about the install upgrade, see:

https://support.purestorage.com/bundle/m_flasharrayx/page/FlashArray/FlashArray_Hardware/94_FlashArray_X/topics/concept/c_flasharrayx_install_and_upgrade_guides.html

**Note:** Currently, the initial deployment of a FlashArray™ requires a person to be physically present in the Data Center. This is inconvenient for some customers; particularly those that deploy many FlashArrays. To address this concern, Pure Storage® introduced a new DHCP boot feature, in which the management ports ct0.eth0 & ct1.eth0 on the FlashArray request IP addresses from a DHCP server when the array is first powered on. A REST API endpoint is also added on ct1.eth0, so that after the FlashArray powers on for the first time, users can connect to it remotely via the REST API and initialize the array. This process can be performed remotely and eliminates the need for a direct connection to the FlashArray via the console port for the initial setup.

**Note:** Both management interfaces must be configured on both controllers and both arrays with enabled and active links. The management interfaces are as follows:

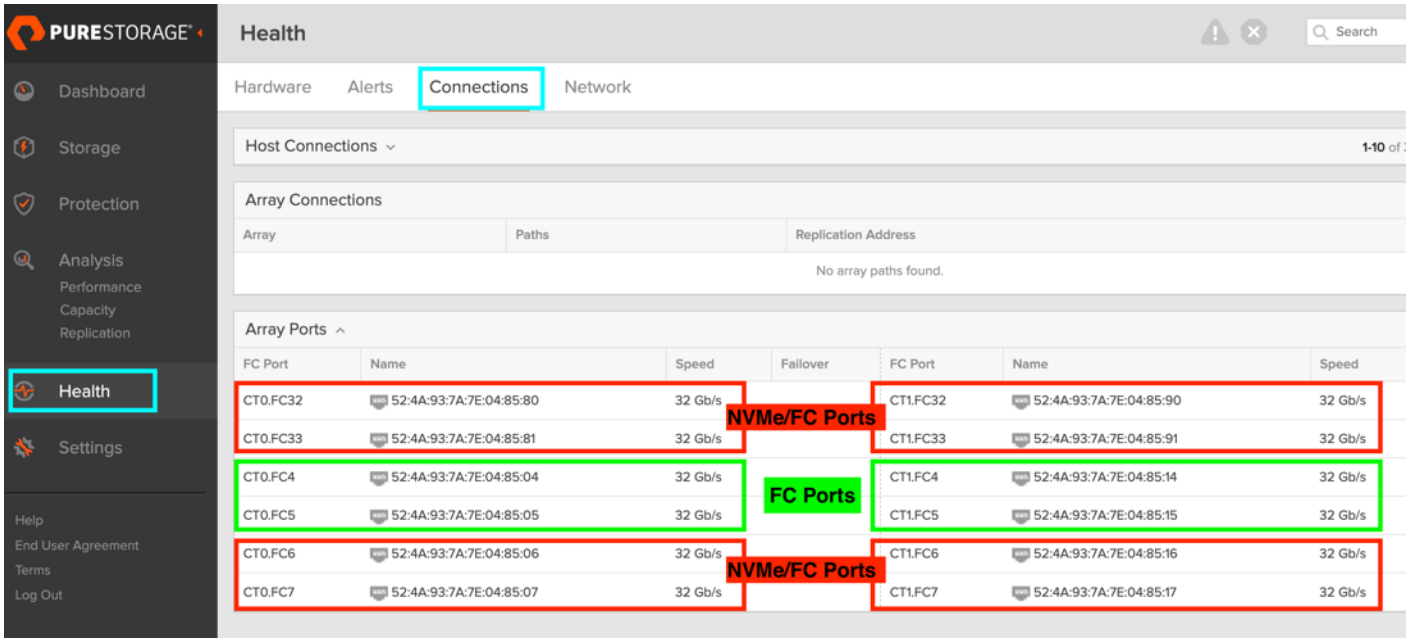
- FlashArray//XR4 - ct0.eth4, ct0.eth5, ct1.eth4, ct1.eth5
- All other FlashArray Models - ct0.eth0, ct0.eth1, ct1.eth0

**Note:** When the FlashArray first powers up in DHCP mode, no authentication is required to connect to the FlashArray via the REST API endpoint. Users can connect to the endpoint via the IP address assigned by the DHCP server. During the initialization process, the DHCP assigned IP addresses are replaced with static IP addresses. When the initialization process is complete, the FlashArray returns to its normal operating mode, the DHCP feature is disabled, and the FlashArray no longer allows remote REST API connections without authentication. Please contact Pure Storage support for setting up initial setup and configuration of the Storage array according to your environment.

As explained earlier in the MDS switch configuration section, both the MDS switches were connected with both the Pure Storage Controllers CT0 and CT1 and also zoning were configured on MDS switches to carry FC and NVMe/FC connectivity. Please refer to table 12 for more details on Pure storage connectivity in the earlier MDS configuration section.

For this solution, we used all the twelve FC ports from Pure Storage controllers. Four ports (two ports from each controller) were configured to carry “scsi-fc” services while remaining eight ports (four ports from each controller) were configured to carry “nvme-fc” services as shown in below green and red colors respectively.

As shown in the image below, both Pure Storage controller nodes, ports FC4 and FC5 were configured with “scsi-fc” services to carry FC network traffic. Also, from both storage controller nodes, ports FC6, FC7, FC32 and FC33 were configured with “nvme-fc” services to carry NVMe/FC database network traffic.




The overview of the network configuration and respective services are shown as below:

```
pureuser@PureFAXL170-ORA21c> purenetwork fc list
Name      Enabled  WWN                       Speed       Services
CT0.FC4   True     52:4A:93:7A:7E:04:85:04   32.00 Gb/s  scsi-fc
CT0.FC5   True     52:4A:93:7A:7E:04:85:05   32.00 Gb/s  scsi-fc
CT0.FC6   True     52:4A:93:7A:7E:04:85:06   32.00 Gb/s  nvme-fc
CT0.FC7   True     52:4A:93:7A:7E:04:85:07   32.00 Gb/s  nvme-fc
CT0.FC32  True     52:4A:93:7A:7E:04:85:80   32.00 Gb/s  nvme-fc
CT0.FC33  True     52:4A:93:7A:7E:04:85:81   32.00 Gb/s  nvme-fc
CT1.FC4   True     52:4A:93:7A:7E:04:85:14   32.00 Gb/s  scsi-fc
CT1.FC5   True     52:4A:93:7A:7E:04:85:15   32.00 Gb/s  scsi-fc
CT1.FC6   True     52:4A:93:7A:7E:04:85:16   32.00 Gb/s  nvme-fc
CT1.FC7   True     52:4A:93:7A:7E:04:85:17   32.00 Gb/s  nvme-fc
CT1.FC32  True     52:4A:93:7A:7E:04:85:90   32.00 Gb/s  nvme-fc
CT1.FC33  True     52:4A:93:7A:7E:04:85:91   32.00 Gb/s  nvme-fc
```

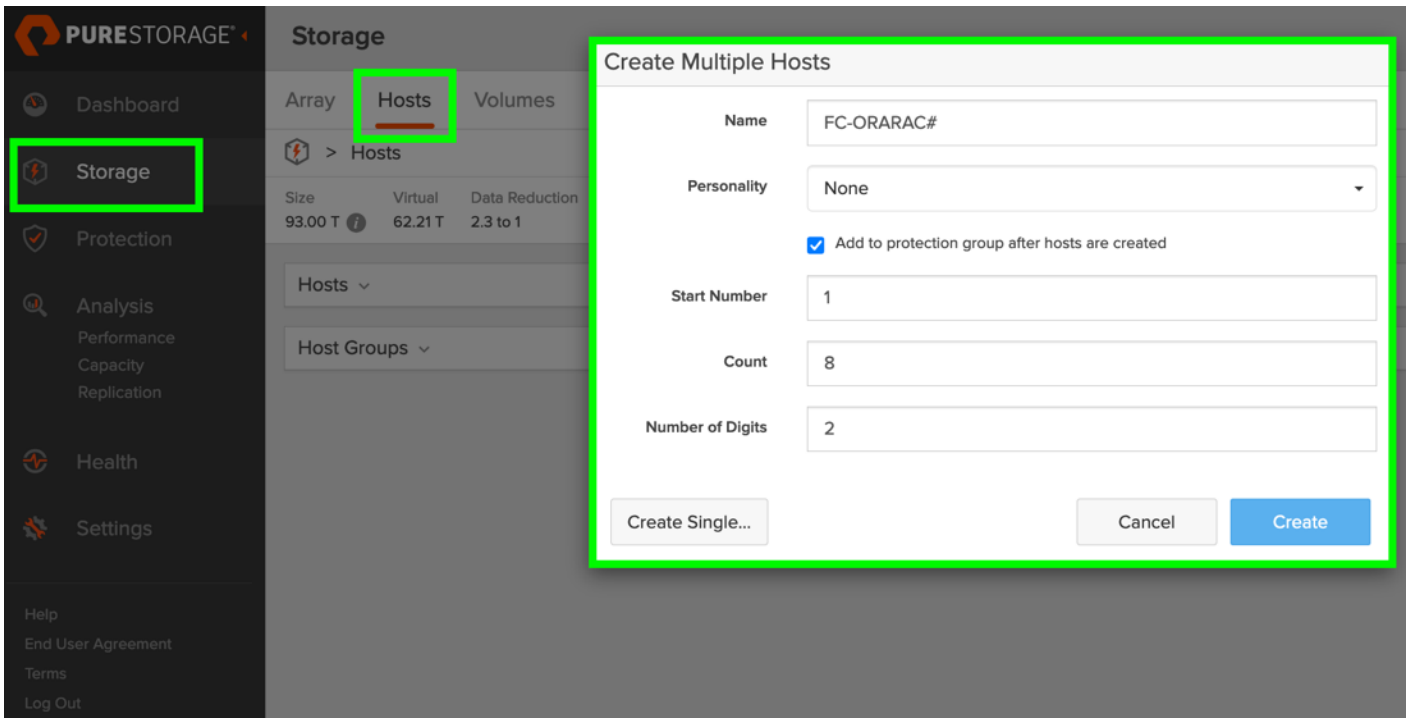
## Configure Host and attach LUN for SAN Boot

**Note:** Configure separate hosts to carry both FC and NVMe/FC network traffic in Pure Storage FlashArray GUI. You will use the eight FC hosts with each hosts having two WWNs for SAN Boot.

### Procedure 1. Configure the Pure Storage Host

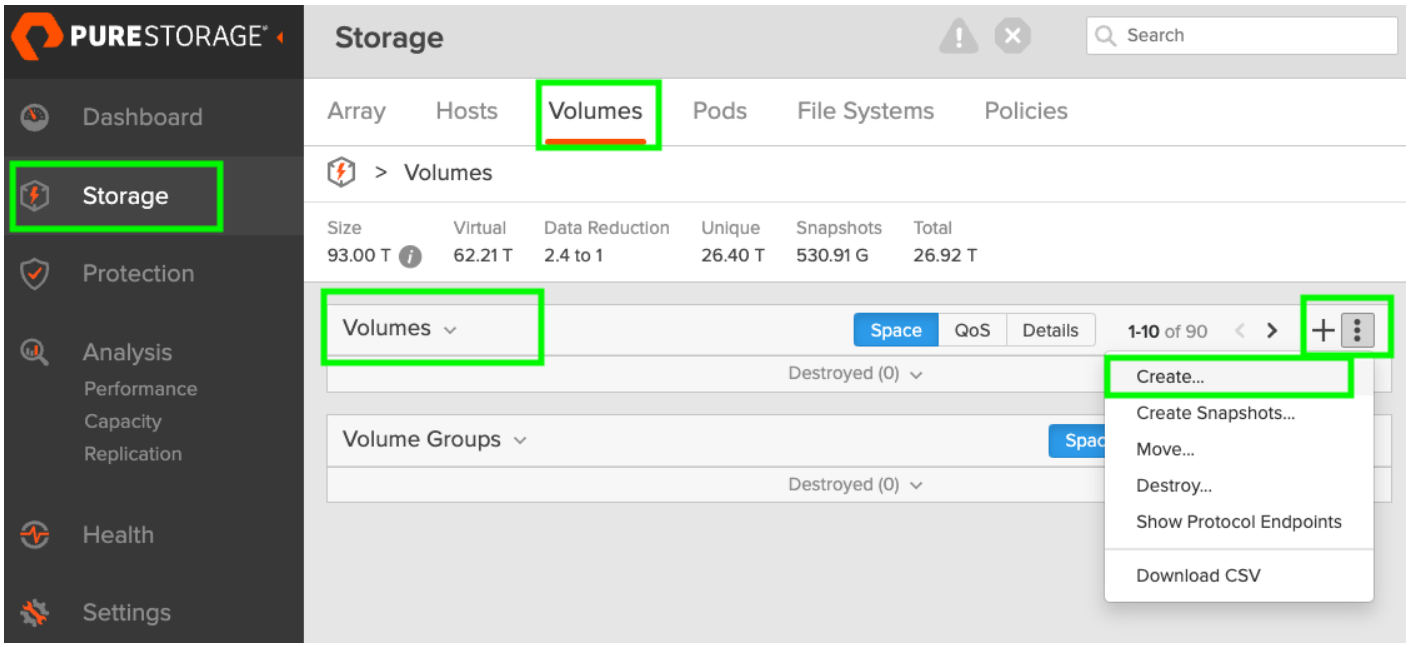
**Step 1.** Login into the Pure Storage array.

**Step 2.** To create a Host into Pure Storage GUI, go to Storage > Hosts > Hosts and under Hosts option in the right frame, click the + sign to create FC host as shown below:




**Step 3.** After creating all eight FC hosts for SAN boot, create 8 volumes and assign this each volume to individual host for FC SAN boot.

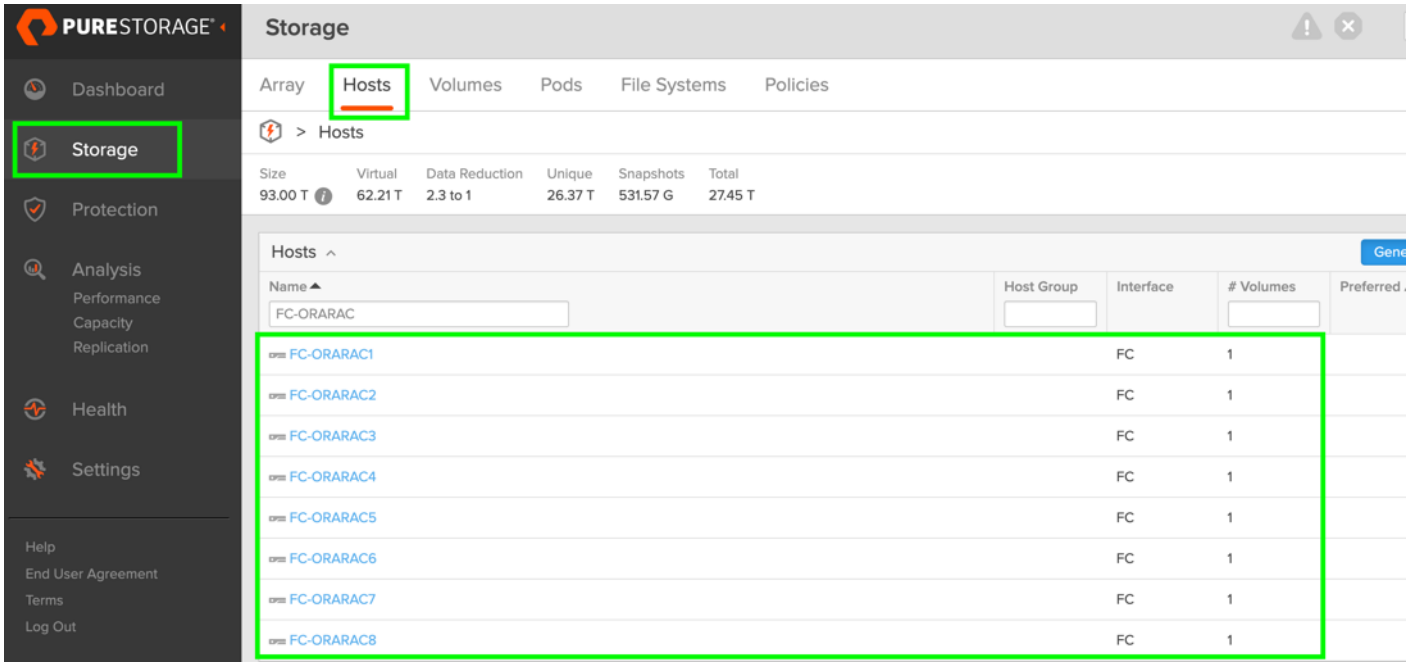
**Step 4.** Go to Storage > Volumes > Volumes > and click "+" on the right menu to create volumes as shown below:



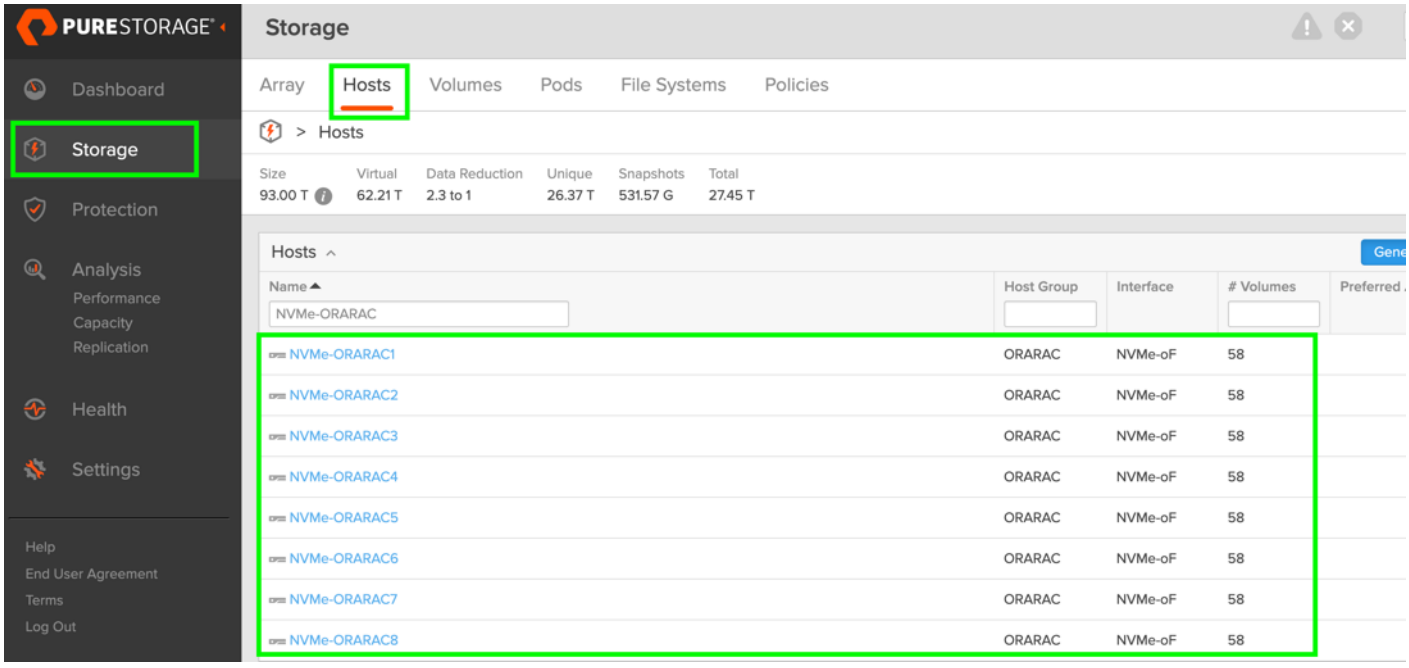

**Note:** We created one dedicated volume for each of the FC Hosts and installed RHEL OS on it.

**Note:** More volumes will be created and their respective NVMe/FC hosts in the database creation section.

Eight FC hosts were configured for FC SAN Boot with each host having two WWNs. After creating all eight FC hosts, eight volumes were created and each volume were mapped to an individual FC Hosts where the OS will be installed, as shown below:




Also, the same eight Hosts were configured to carry NVMe/FC database storage traffic with host's respective NQNs. Also, one host group was configured and all the NVMe hosts were added into that group so that we can share database volumes across all hosts.



Below is the screenshot of all the FC and NVMe/FC hosts configured for this solution as:

```
pureuser@PureFAXL170-ORA21c> purehost list
Name          WWN                     IQN  NQN                                                                Host Group
FC-ORARAC1    20:00:00:25:B5:AB:30:00 -    -                                                                  -
              20:00:00:25:B5:AB:30:01
FC-ORARAC2    20:00:00:25:B5:AB:30:0A -    -                                                                  -
              20:00:00:25:B5:AB:30:0B
FC-ORARAC3    20:00:00:25:B5:AB:30:14 -    -                                                                  -
              20:00:00:25:B5:AB:30:15
FC-ORARAC4    20:00:00:25:B5:AB:30:1E -    -                                                                  -
              20:00:00:25:B5:AB:30:1F
FC-ORARAC5    20:00:00:25:B5:AB:30:28 -    -                                                                  -
              20:00:00:25:B5:AB:30:29
FC-ORARAC6    20:00:00:25:B5:AB:30:32 -    -                                                                  -
              20:00:00:25:B5:AB:30:33
FC-ORARAC7    20:00:00:25:B5:AB:30:3C -    -                                                                  -
              20:00:00:25:B5:AB:30:3D
FC-ORARAC8    20:00:00:25:B5:AB:30:46 -    -                                                                  -
              20:00:00:25:B5:AB:30:47
NVMe-ORARAC1  -                       -    nqn.2014-08.org.nvmexpress:uuid:35010000-5013-0000-0000-135135000000  ORARAC
NVMe-ORARAC2  -                       -    nqn.2014-08.org.nvmexpress:uuid:35010000-5013-0000-0000-135135000001  ORARAC
NVMe-ORARAC3  -                       -    nqn.2014-08.org.nvmexpress:uuid:35010000-5013-0000-0000-135135000002  ORARAC
NVMe-ORARAC4  -                       -    nqn.2014-08.org.nvmexpress:uuid:35010000-5013-0000-0000-135135000003  ORARAC
NVMe-ORARAC5  -                       -    nqn.2014-08.org.nvmexpress:uuid:35010000-5013-0000-0000-135135000004  ORARAC
NVMe-ORARAC6  -                       -    nqn.2014-08.org.nvmexpress:uuid:35010000-5013-0000-0000-135135000005  ORARAC
NVMe-ORARAC7  -                       -    nqn.2014-08.org.nvmexpress:uuid:35010000-5013-0000-0000-135135000006  ORARAC
NVMe-ORARAC8  -                       -    nqn.2014-08.org.nvmexpress:uuid:35010000-5013-0000-0000-135135000007  ORARAC
```

After configuring FC and NVMe/FC hosts, we are now ready to install OS through SAN boot as described in the next section.

## Operating System and Database Deployment

This chapter contains the following:



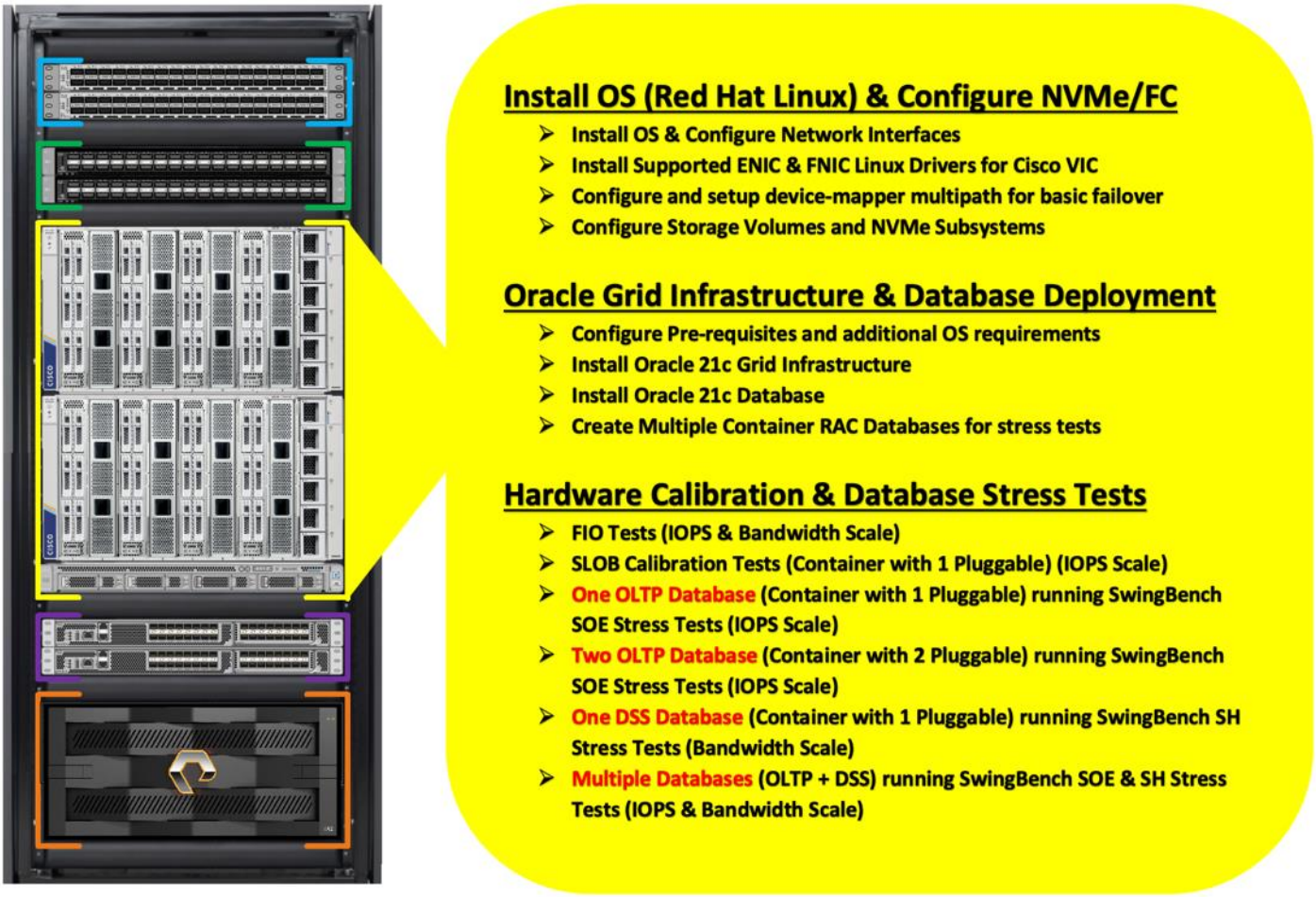
**Note:** Detailed steps to install the OS are not explained in this document, but the following section describes the high-level steps for an OS install.

The design goal of this reference architecture is to represent a real-world environment as closely as possible.

As explained in the previous section, the service profile was created using Cisco Intersight to rapidly deploy all stateless servers to deploy an eight node Oracle RAC. The SAN boot LUNs for these servers were hosted on the Pure Storage system to provision the OS. The zoning was performed on the Cisco MDS Switches to enable the initiators to discover the targets during the boot process.

Each server node has a dedicated single LUN to install the operating system. For this solution, the Red Hat Enterprise Linux Server 8.9 (4.18.0-513.5.1.el8_9.x86_64) was installed on these LUNs and the NVMe/FC connectivity was configured, all prerequisite packages were configured to install the Oracle Database 21c Grid Infrastructure, and the Oracle Database 21c software was used to create an eight node Oracle Multitenant RAC 21c database for this solution.

The following screenshot shows the high-level steps to configure the Linux Hosts and deploy the Oracle RAC Database solution:




This section describes the high-level steps to configure the Oracle Linux Hosts and deploy the Oracle RAC Database solution.

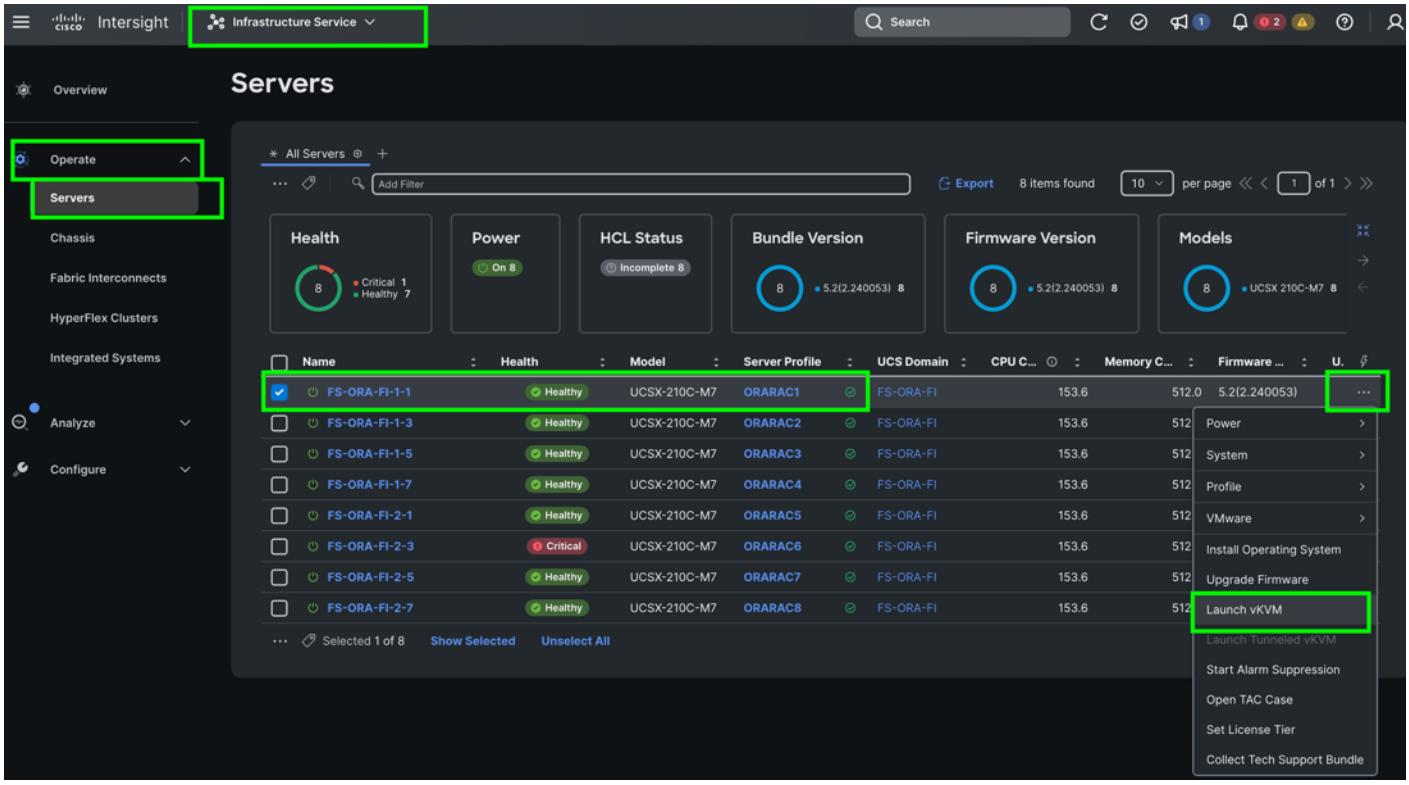
## Configure the Operating System

**Note:** The detailed installation process is not explained in this document, but the following procedure describes the high-level steps for the OS installation.

**Procedure 1.** Configure OS

**Step 1.** Download the Red Hat Enterprise Linux 8.9 OS image and save the IOS file to local disk.

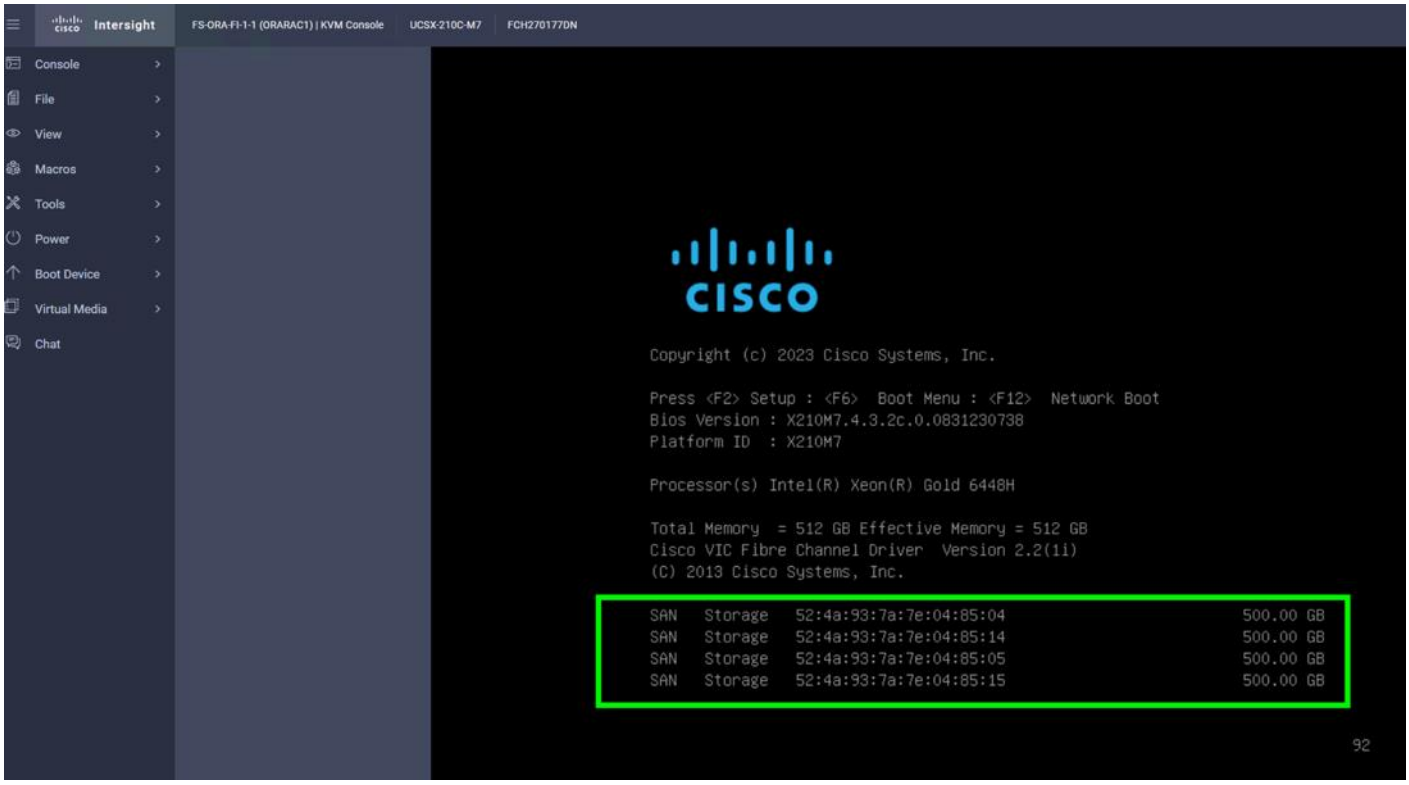
**Step 2.** Launch the vKVM console on your server by going to Cisco Intersight > Infrastructure Service > Operate > Servers > click Chassis 1 Server 1 > from the Actions drop-down list select Launch vKVM.




**Step 3.** Click Accept security and open KVM. Click Virtual Media > vKVM-Mapped vDVD. Click Browse and map the Oracle Linux ISO image, click Open and then click Map Drive. After mapping the iso file, click Power > Power Cycle System to reboot the server.

**Step 4.** When the Server boots, it will detect the boot order and start booting from the Virtual mapped DVD as previously configured.

**Step 5.** When the Server starts booting, it will detect the Pure Storage active FC paths. If you see those following storage targets in the KVM console while the server is rebooting along with the target WWPNs, it confirms the setup and zoning is done correctly and boot from SAN will be successful.




**Step 6.** During the server boot order, it detects the virtual media connected as RHEL OS ISO DVD media and it will launch the RHEL OS installer.

**Step 7.** Select language and for the Installation destination assign the local virtual drive. Apply the hostname and click Configure Network to configure any or all the network interfaces. Alternatively, you can configure only the “Public Network” in this step. You can configure additional interfaces as part of post OS install steps.

**Note:** For an additional RPM package, we recommend selecting the “Customize Now” option and the relevant packages according to your environment.

**Step 8.** After the OS installation finishes, reboot the server, and complete the appropriate registration steps.

**Step 9.** Repeat steps 1 – 4 on all server nodes and install RHEL 8.9 to create an eight node Linux system.

**Step 10.** Optionally, you can choose to synchronize the time with ntp server. Alternatively, you can choose to use the Oracle RAC cluster synchronization daemon (OCSSD). Both NTP and OCSSD are mutually exclusive and OCSSD will be setup during GRID install if NTP is not configured.

## ENIC and FNIC Drivers for Linux OS

For this solution, the following ENIC and FNIC versions were installed:

- ENIC: version: 4.6.0.0-977.3
- FNIC: version: 2.0.0.96-324.0

**Procedure 1.** Install the ENIC and FNIC drivers

**Step 1.** Download the supported UCS Linux Drivers from this link: https://software.cisco.com/download/home/286327804

**Step 2.** Mount the driver ISO to the Linux host KVM and install the relevant supported ENIC and FNIC drivers for the Linux OS. To configure the drivers, run the following commands:

- Check the current ENIC & FNIC version:

```
[root@orarac1 ~]# cat /sys/module/enic/version
[root@orarac1 ~]# cat /sys/module/fnic/version
[root@orarac1 ~]# rpm -qa | grep enic
[root@orarac1 ~]# rpm -qa | grep fnic
```

- Install the supported ENIC & FNIC driver from RPM:

```
[root@orarac1 software]# rpm -ivh kmod-enic-4.6.0.0-977.3.rhel8u9_4.18.0_513.5.1.x86_64
[root@orarac1 software]# rpm -ivh kmod-fnic-2.0.0.96-324.0.rhel8u9.x86_64
```

- Reboot the server and verify that the new driver is running as shown below:

```
[root@orarac1 ~]# rpm -qa | grep enic
kmod-enic-4.6.0.0-977.3.rhel8u9_4.18.0_513.5.1.x86_64

[root@orarac1 ~]# rpm -qa | grep fnic
kmod-fnic-2.0.0.96-324.0.rhel8u9.x86_64

[root@orarac1 ~]# modinfo enic | grep version
version:        4.6.0.0-977.3
rhelversion:    8.9
srcversion:     4248075B65C84CA281FE03E
vermagic:       4.18.0-513.5.1.el8_9.x86_64 SMP mod_unload modversions

[root@orarac1 ~]# modinfo fnic | grep version
version:        2.0.0.96-324.0
rhelversion:    8.9
srcversion:     0EA398F96B3E8444AF73198
vermagic:       4.18.0-513.5.1.el8_9.x86_64 SMP mod_unload modversions

[root@orarac1 ~]# cat /sys/module/enic/version
4.6.0.0-977.3

[root@orarac1 ~]# cat /sys/module/fnic/version
2.0.0.96-324.0
```

```
[root@orarac1 ~]# lsmod | grep fnic
fnic                  290816  8
nvme_fc                53248  3713 fnic
scsi_transport_fc      81920  1 fnic
[root@orarac1 ~]#
```

**Step 3.** Repeat steps 1 and 2 to configure the Cisco VIC linux drivers on all eight nodes.

**Note:** You should use a matching ENIC and FNIC pair.

**Note:** Check the Cisco UCS supported driver release for more information about the supported kernel version, here:
https://www.cisco.com/c/en/us/support/docs/servers-unified-computing/ucs-manager/116349-technote-product-00.html

## NVME CLI

The NVME hosts and targets are distinguished through their NQN. The FNIC NVME host reads its host nqn from the file /etc/nvme/hostnqn. With a successful installation of the nvme-cli package, the hostnqn file will be created automatically for some OS versions, such as RHEL.

**Note:** If the /etc/nvme/hostnqn file is not present after name-cli installed, then create the file manually.

**Procedure 1.** Install the NVME CLI

**Step 1.** Run the following commands to Install nvme-cli and get HostNQN information from the host:

```
[root@orarac1 ~]# rpm -q nvme-cli
nvme-cli-1.16-9.el8.x86_64

[root@orarac1 ~]# cat /etc/nvme/hostnqn
nqn.2014-08.org.nvmexpress:uuid:35010000-5013-0000-0000-135135000000
```

## Device-mapper Multipathing

For this solution, the DM-Multipath was configured for the FC Boot LUNs as well as NVMe/FC database workload. For more information, go to the following links:

- https://support.purestorage.com/bundle/m_linux/page/Production/Solutions/Oracle/Oracle_on_FlashArray/library/common_content/c_recommended_dmmultipath_settings.html
- https://support.purestorage.com/bundle/m_flasharrayx/page/FlashArray/FlashArray_Hardware/94_FlashArray_X/topics/concept/c_enabling_nvmefc.html

**Note:** For DM-Multipath Configuration and best practice, refer to Pure Storage Support article:
https://support.purestorage.com/bundle/m_linux/page/Solutions/Oracle/Oracle_on_FlashArray/library/common_content/c_recommended_dmmultipath_settings.html

**Note:** We made sure the multipathing packages were installed and enabled for an automatic restart across reboots.

**Procedure 1.** Configure device-mapper multipathing

**Step 1.** Enable and initialize the multipath configuration file:

```
[root@orarac1 ~]# mpathconf --enable

[root@orarac1 ~]# mpathconf
multipath is enabled
find_multipaths is no
user_friendly_names is disabled
default property blacklist is disabled
enable_foreign is not set (all foreign multipath devices will be shown)
dm_multipath module is loaded
multipathd is running

[root@orarac1 ~]# systemctl status multipathd.service
● multipathd.service - Device-Mapper Multipath Device Controller
   Loaded: loaded (/usr/lib/systemd/system/multipathd.service; enabled; vendor preset:
enabled)
   Active: active (running) since Mon 2024-06-17 12:09:07 PDT; 1 weeks 0 days ago
  Process: 3655 ExecStartPre=/sbin/multipath -A (code=exited, status=0/SUCCESS)
  Process: 3653 ExecStartPre=/sbin/modprobe -a scsi_dh_alua scsi_dh_emc scsi_dh_rdac
dm-multipath (code=exited, status=0/SUCCESS)
 Main PID: 3657 (multipathd)
   Status: "up"
    Tasks: 7
   Memory: 133.7M
   CGroup: /system.slice/multipathd.service
           └─3657 /sbin/multipathd -d -s
```

**Step 2.** Edit the “/etc/multipath.conf” file:

```
[root@orarac1 ~]# cat /etc/multipath.conf
defaults {
        polling_interval       10
}

devices {
    device {
        vendor                      "NVME"
```

```
        product                 "Pure Storage FlashArray"
        path_selector           "queue-length 0"
        path_grouping_policy    group_by_prio
        prio                    ana
        failback                immediate
        fast_io_fail_tmo        10
        user_friendly_names     no
        no_path_retry           0
        features                0
        dev_loss_tmo            60
    }
    device {
        vendor                  "PURE"
        product                 "FlashArray"
        path_selector           "service-time 0"
        hardware_handler        "1 alua"
        path_grouping_policy    group_by_prio
        prio                    alua
        failback                immediate
        path_checker            tur
        fast_io_fail_tmo        10
        user_friendly_names     no
        no_path_retry           0
        features                0
        dev_loss_tmo            600
    }
}

multipaths {
        multipath {
                wwid            3624a93704a5561942d7640ea00011436
                alias           ORARAC1-RHEL-OS
        }
}
```

**Note:** To ensure the best performance with the Pure Storage FlashArray, please use this guide for the configuration and implementation of Linux hosts in your environment: https://support.purestorage.com/bundle/m_troubleshooting_for_vmware_solutions/page/Solutions/VMware_Platform_Guide/Troubleshooting_for_VMware_Solutions/VMware-Related_KB_Articles/library/common_content/c_introduction_46.html. These recommendations apply to the versions of Linux that Pure storage have certified.

**Note:** Regarding path selectors as listed above, Pure Storage recommends using queue-length 0 with NVMe and service-time 0 with SCSI, which improve performance in situations where paths have differing latencies by biasing I/Os towards paths that are servicing I/O more quickly.

**Note:** To ensure the best performance with the Pure Storage FlashArray for Oracle deployment, please use this guide:
https://support.purestorage.com/bundle/m_oracle/page/Solutions/Oracle/Oracle_on_FlashArray/topics/concept/c_oracle_database_recommended_settings_for_flasharray.html

**Step 3.** Run "multipath -ll" command to view all the LUN id and enter that wwid information accordingly on each node:

```
[root@ororac1 ~]# multipath -ll
ORARAC1-RHEL-OS (3624a93704a5561942d7640ea00011436) dm-5 PURE,FlashArray
size=500G features='0' hwhandler='1 alua' wp=rw
`-+- policy='service-time 0' prio=50 status=active
  |- 3:0:1:1        sdh       8:112    active ready running
  |- 3:0:0:1        sdg       8:96     active ready running
  |- 4:0:0:1        sdi       8:128    active ready running
  `- 4:0:1:1        sdj       8:144    active ready running
```

## Public and Private Network Interfaces

If you have not configured network settings during OS installation, then configure it now. Each node must have at least two network interface cards (NICs), or network adapters. One adapter is for the public network interface and another adapter is for the private network interface (RAC interconnect).

**Procedure 1.** Configure Management Public and Private Network Interfaces

**Step 1.** Login as a root user into each Linux node and go to "/etc/sysconfig/network-scripts/"

**Step 2.** Configure the Public network and Private network IP addresses according to your environments.

**Note:** Configure the Private and Public network with the appropriate IP addresses on all eight Linux Oracle RAC nodes.

## Configure OS Prerequisites for Oracle Software

To successfully install the Oracle RAC Database 21c software, configure the operating system prerequisites on all eight Linux nodes.

**Note:** Follow the steps according to your environment and requirements. For more information, see the Install and Upgrade Guide for Linux for Oracle Database 21c:
https://docs.oracle.com/en/database/oracle/oracle-database/21/cwlin/index.html and
https://docs.oracle.com/en/database/oracle/oracle-database/21/ladbi/index.html

**Procedure 1.** Configure the OS prerequisites

**Step 1.** To configure the operating system prerequisites using RPM for Oracle 21c software on Linux node, install the “oracle-database-preinstall-21c (oracle-database-preinstall-21c-1.0-1.el8.x86_64.rpm)" rpm package on all eight nodes. You can also download the required packages from: https://public-yum.oracle.com/oracle-linux-8.html

**Step 2.** If you plan to use the “oracle-database-preinstall-21c" rpm package to perform all your prerequisites setup automatically, then login as root user and issue the following command on all each of the RAC nodes:

```
[root@orarac1 ~]# yum install oracle-database-preinstall-21c-1.0-1.el8.x86_64.rpm
```

**Note:** If you have not used the oracle-database-preinstall-21c package, then you will have to manually perform the prerequisites tasks on all the nodes.

## Configure Additional OS Prerequisites

After configuring the automatic or manual prerequisites steps, you have a few additional steps to complete the prerequisites to install the Oracle database software on all eight Linux nodes.

### Procedure 1. Disable SELinux

Since most organizations might already be running hardware-based firewalls to protect their corporate networks, you need to disabled Security Enhanced Linux (SELinux) and the firewalls at the server level for this reference architecture.

**Step 1.** Set the secure Linux to permissive by editing the "/etc/selinux/config" file, making sure the SELINUX flag is set as follows:

```
SELINUX=permissive
```

### Procedure 2. Disable Firewall

**Step 1.** Check the status of the firewall by running following commands. (The status displays as active (running) or inactive (dead)). If the firewall is active / running, run this command to stop it:

```
systemctl status firewalld.service
systemctl stop firewalld.service
```

**Step 2.** To completely disable the firewalld service so it does not reload when you restart the host machine, run the following command:

```
systemctl disable firewalld.service
```

### Procedure 3. Create Grid User

**Step 1.** Run this command to create a grid user:

```
useradd -u 54322 -g oinstall -G dba grid
```

### Procedure 4. Set the User Passwords

**Step 1.** Run these commands to change the password for Oracle and Grid Users:

```
passwd oracle
passwd grid
```

**Procedure 5.** Configure UDEV Rules for IO Policy

You need to configure the UDEV rules to assign the IO Policy in all Oracle RAC nodes to access the Pure Storage subsystems. To review the best practices for applying queue settings with UDEV rules, go to: https://support.purestorage.com/bundle/m_linux/page/Solutions/Oracle/Oracle_on_FlashArray/library/common_content/c_applying_queue_settings_with_udev.html

**Step 1.** Assign IO Policy by creating a new file named "99-pure-storage.rules" with the following entries on all the nodes:

```
[root@orarac1 ~]# cat /etc/udev/rules.d/99-pure-storage.rules

# Recommended settings for Pure Storage FlashArray.
# Use none scheduler for high-performance solid-state storage for SCSI devices
ACTION=="add|change", KERNEL=="sd*[!0-9]", SUBSYSTEM=="block", ENV{ID_VENDOR}=="PURE", ATTR{queue/scheduler}="none"
ACTION=="add|change", KERNEL=="dm-[0-9]*", SUBSYSTEM=="block", ENV{DM_NAME}=="3624a937*", ATTR{queue/scheduler}="none"

# Reduce CPU overhead due to entropy collection
ACTION=="add|change", KERNEL=="sd*[!0-9]", SUBSYSTEM=="block", ENV{ID_VENDOR}=="PURE", ATTR{queue/add_random}="0"
ACTION=="add|change", KERNEL=="dm-[0-9]*", SUBSYSTEM=="block", ENV{DM_NAME}=="3624a937*", ATTR{queue/add_random}="0"

# Spread CPU load by redirecting completions to originating CPU
ACTION=="add|change", KERNEL=="sd*[!0-9]", SUBSYSTEM=="block", ENV{ID_VENDOR}=="PURE", ATTR{queue/rq_affinity}="2"
ACTION=="add|change", KERNEL=="dm-[0-9]*", SUBSYSTEM=="block", ENV{DM_NAME}=="3624a937*", ATTR{queue/rq_affinity}="2"

# Set the HBA timeout to 60 seconds
ACTION=="add|change", KERNEL=="sd*[!0-9]", SUBSYSTEM=="block", ENV{ID_VENDOR}=="PURE", ATTR{device/timeout}="60"
```

**Procedure 6.** Configure "/etc/hosts"

**Step 1.** Login as a root user into the Linux node and edit the "/etc/hosts" file.

**Step 2.** Provide the details for Public IP Address, Private IP Address, SCAN IP Address, and Virtual IP Address for all the nodes. Configure these settings in each Oracle RAC Nodes as shown below:

```
[root@orarac1 ~]# cat /etc/hosts
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
```

```
###      Public IP
10.29.135.71      orarac1      orarac1.ciscoucs.com
10.29.135.72      orarac2      orarac2.ciscoucs.com
10.29.135.73      orarac3      orarac3.ciscoucs.com
10.29.135.74      orarac4      orarac4.ciscoucs.com
10.29.135.75      orarac5      orarac5.ciscoucs.com
10.29.135.76      orarac6      orarac6.ciscoucs.com
10.29.135.77      orarac7      orarac7.ciscoucs.com
10.29.135.78      orarac8      orarac8.ciscoucs.com

### Virtual IP
10.29.135.79      orarac1-vip  orarac1-vip.ciscoucs.com
10.29.135.80      orarac2-vip  orarac2-vip.ciscoucs.com
10.29.135.81      orarac3-vip  orarac3-vip.ciscoucs.com
10.29.135.82      orarac4-vip  orarac4-vip.ciscoucs.com
10.29.135.83      orarac5-vip  orarac5-vip.ciscoucs.com
10.29.135.84      orarac6-vip  orarac6-vip.ciscoucs.com
10.29.135.85      orarac7-vip  orarac7-vip.ciscoucs.com
10.29.135.86      orarac8-vip  orarac8-vip.ciscoucs.com

### Private IP
10.10.10.71 orarac1-priv  orarac1-priv.ciscoucs.com
10.10.10.72 orarac2-priv  orarac2-priv.ciscoucs.com
10.10.10.73 orarac3-priv  orarac3-priv.ciscoucs.com
10.10.10.74 orarac4-priv  orarac4-priv.ciscoucs.com
10.10.10.75 orarac5-priv  orarac5-priv.ciscoucs.com
10.10.10.76 orarac6-priv  orarac6-priv.ciscoucs.com
10.10.10.77 orarac7-priv  orarac7-priv.ciscoucs.com
10.10.10.78 orarac8-priv  orarac8-priv.ciscoucs.com

### SCAN IP
10.29.135.87      orarac-scan  orarac-scan.ciscoucs.com
10.29.135.88      orarac-scan  orarac-scan.ciscoucs.com
10.29.135.89      orarac-scan  orarac-scan.ciscoucs.com
```

**Step 3.** You must configure the following addresses manually in your corporate setup:

- A Public and Private IP Address for each Linux node
- A Virtual IP address for each Linux node
- Three single client access name (SCAN) address for the oracle database cluster

**Note:** These steps were performed on all eight Linux nodes. These steps complete the prerequisites for the Oracle Database 21c installation at OS level on the Oracle RAC Nodes.

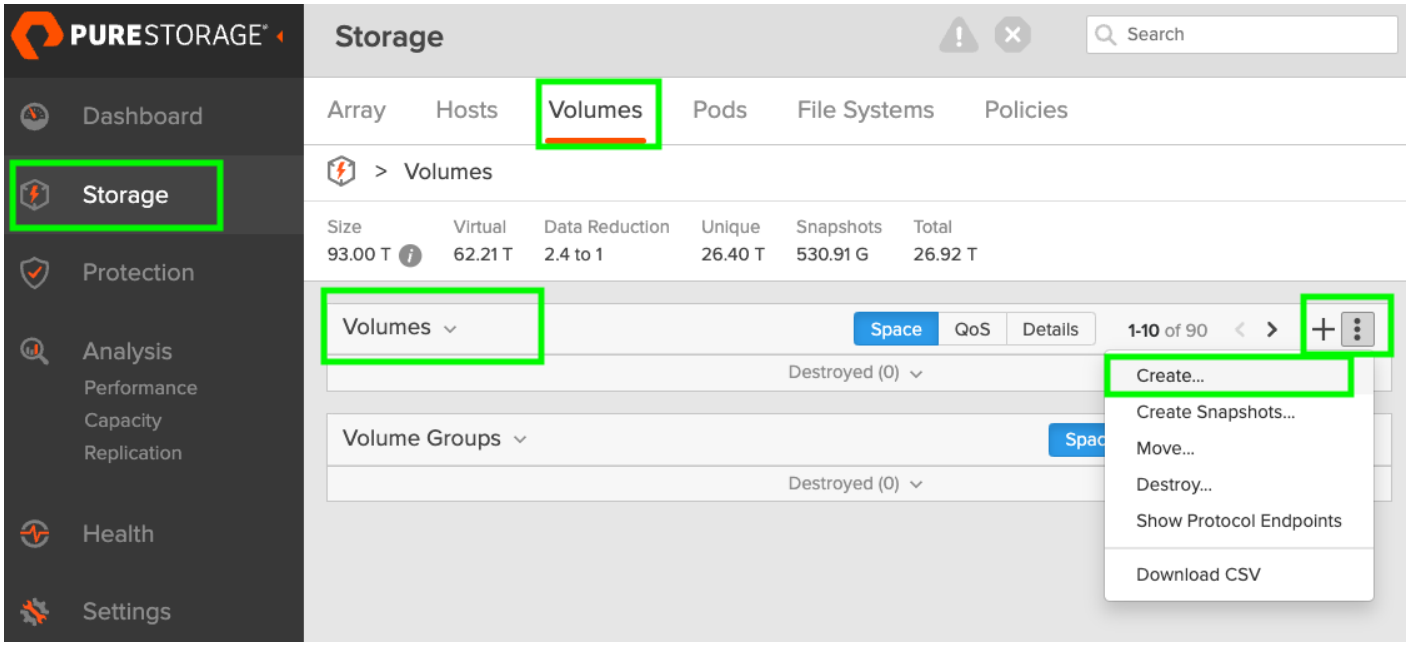
## Configure Volumes for OCR and Voting Disk

You will use the "OCRVOTE" file system on the storage array to store the OCR (Oracle Cluster Registry) files, Voting Disk files, and other Clusterware files. We have created two volumes and shared both these two volumes across all eight nodes so that database nodes can access these files.

**Procedure 1.** Configure the Pure Storage Host Group and Volumes for OCR and Voting Disk

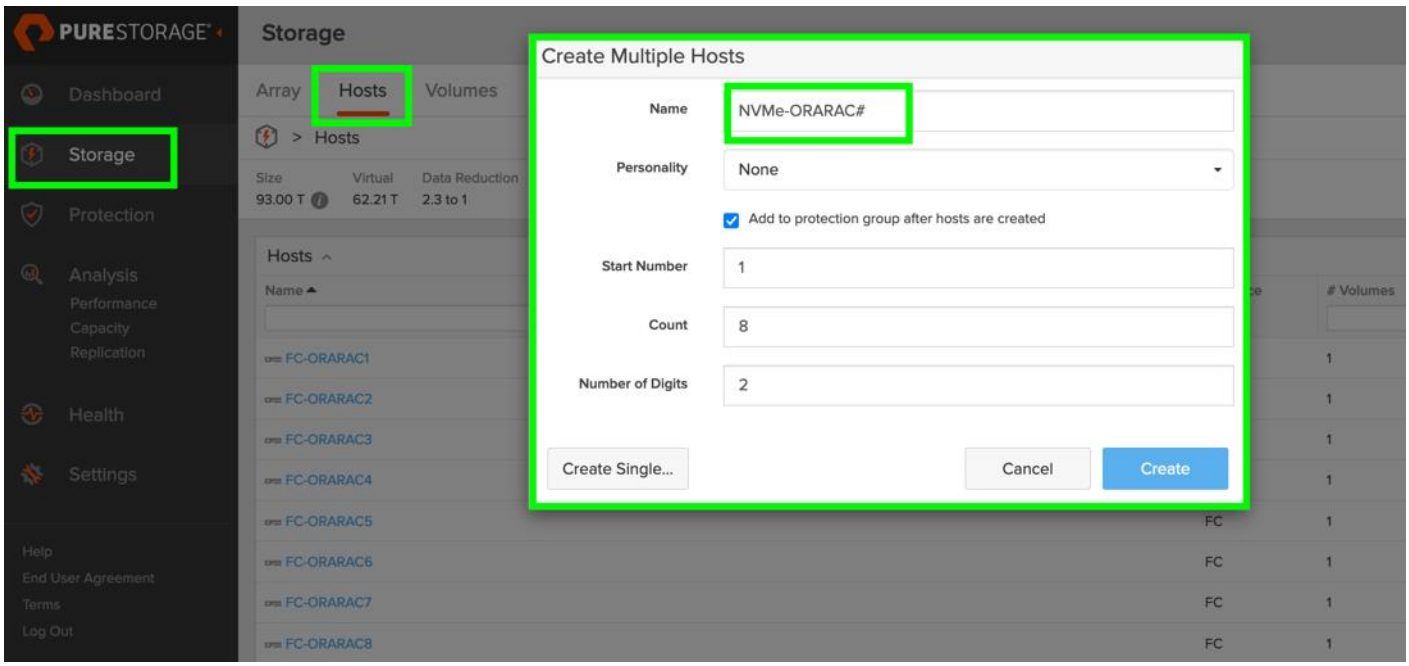
**Step 1.** Login into the Pure Storage array.

**Step 2.** Go to Storage > Volumes > Volumes > and click "+" on the right menu to create volumes as shown below:



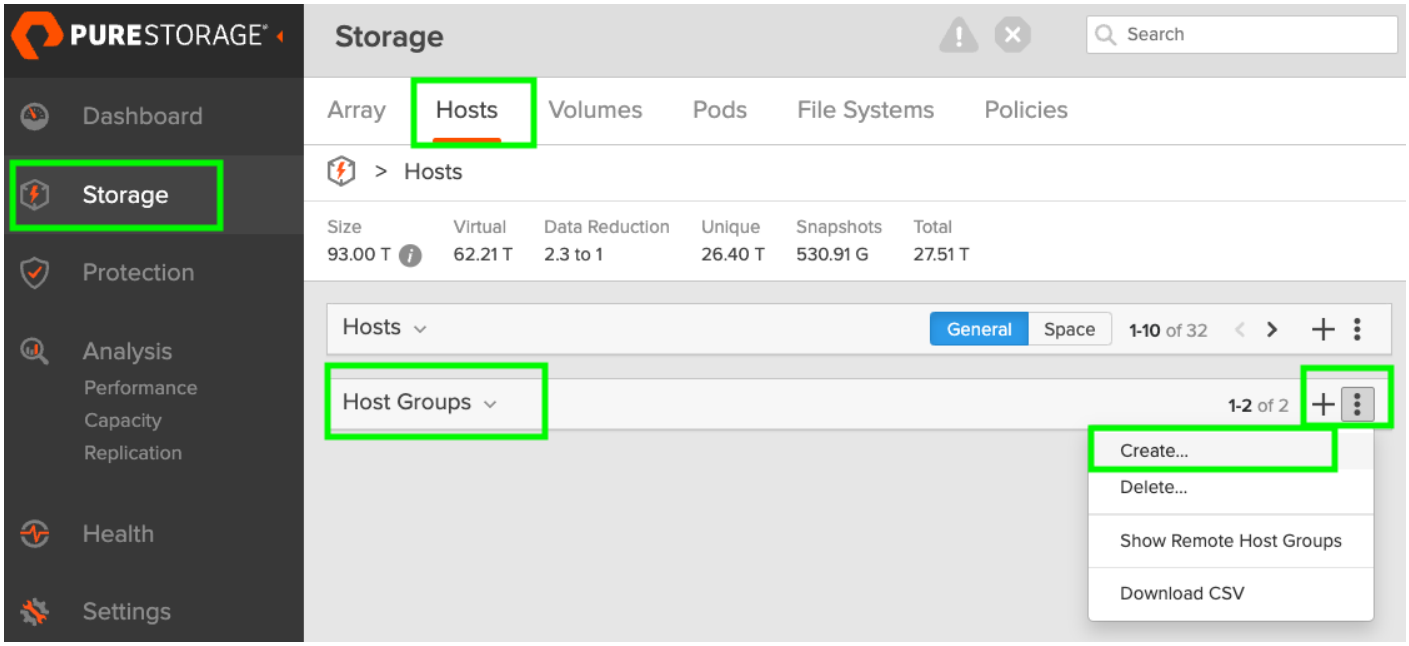

**Note:** We created two volumes for configuring these files as "ocrvote1" and "ocrvote2."

**Step 3.** To create Host a Host into Pure Storage GUI, go to Storage > Hosts > Hosts and under Hosts option in the right frame, click the + sign to create NVMe hosts.

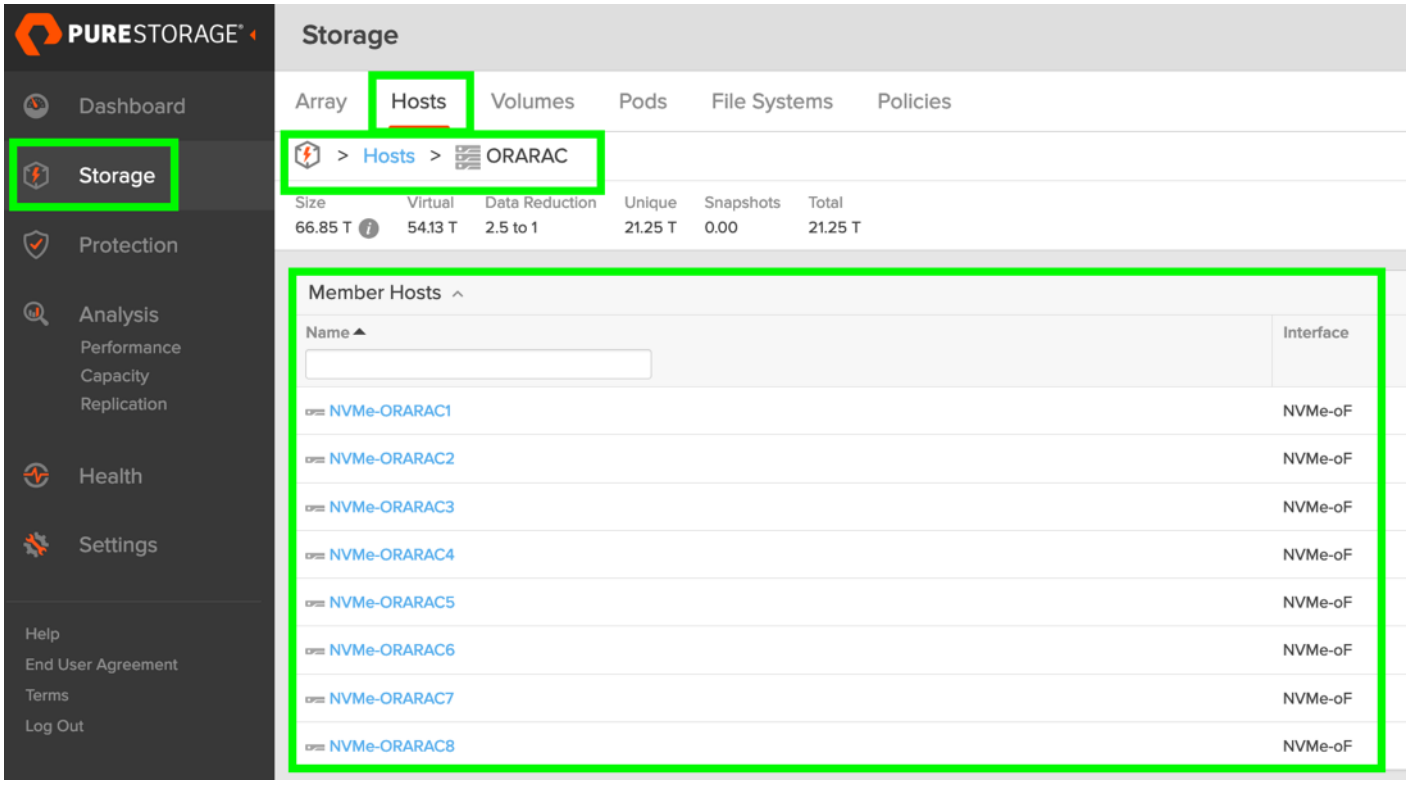



**Step 4.** For this solution, eight NVMe Hosts were configured to carry NVMe/FC database storage traffic with host's respective NQNs. Also, one host group was configured and all the NVMe hosts were added into that group so that we can share database volumes across all hosts

**Step 5.** Go to Storage > Hosts > Host Groups > and click "+" on the right menu to create host group as shown below:




**Note:** For this solution, we created one host group as "ORARAC" and added al eight hosts (orarac1 to orarac8) into this group.




**Step 6.** Into this “ORARAC” host group, connect two volumes “ocrvote1” and “ocrvote2” to share these two volumes across all eight nodes.

**Note:** You will create more volumes for storing database files later in the database creation.

**Step 7.** When the OS level prerequisites and file systems are configured, you are ready to install the Oracle Grid Infrastructure as grid user. Download the Oracle Database 21c (21.3.0.0.0) for Linux x86-64 and the Oracle Database 21c Grid Infrastructure (21.3.0.0.0) for Linux x86-64 software from Oracle Software site. Copy these software binaries to Oracle RAC Node 1 and unzip all files into appropriate directories.

**Note:** These steps complete the prerequisites for the Oracle Database 21c Installation at OS level on the Oracle RAC Nodes.

## Oracle Database 21c GRID Infrastructure Deployment

This section describes the high-level steps for the Oracle Database 21c RAC installation. This document provides a partial summary of details that might be relevant.

**Note:** It is not within the scope of this document to include the specifics of an Oracle RAC installation; you should refer to the Oracle installation documentation for specific installation instructions for your environment. For more information, click this link for the Oracle Database 21c install and upgrade guide: https://docs.oracle.com/en/database/oracle/oracle-database/21/cwlin/index.html

For this solution, two volumes of 50G each in size were created and shared across all eight Linux nodes for storing OCR and Voting Disk files for all RAC databases. Oracle 21c Release 21.3 Grid Infrastructure (GI) was installed on the first node as a grid user. The installation also configured and added the remaining three nodes as a part of the GI setup. We also configured Oracle Automatic Storage Management (ASM) in Flex mode.

Complete the following procedures to install the Oracle Grid Infrastructure software for the Oracle Standalone Cluster.

### Procedure 1. Create Directory Structure

**Step 1.** Download and copy the Oracle Grid Infrastructure image files to the first local node only. During installation, the software is copied and installed on all other nodes in the cluster.

**Step 2.** Create the directory structure according to your environment and run the following commands:

For example:

```
mkdir -p /u01/app/grid
mkdir -p /u01/app/21.3.0/grid
mkdir -p /u01/app/oraInventory
mkdir -p /u01/app/oracle/product/21.3.0/dbhome_1

chown -R grid:oinstall /u01/app/grid
chown -R grid:oinstall /u01/app/21.3.0/grid
chown -R grid:oinstall /u01/app/oraInventory
chown -R oracle:oinstall /u01/app/oracle
```

**Step 3.** As the grid user, download the Oracle Grid Infrastructure image files and extract the files into the Grid home:

```
cd /u01/app/21.3.0/grid
unzip -q <download_location>/LINUX.X64_213000_grid_home.zip
```

### Procedure 2. Configure UDEV Rules for ASM Disk Access

**Step 1.** Configure the UDEV rules to have read/write privileges on the storage volumes for grid user. This includes the device details and corresponding “uuid” of the storage volumes:

**Step 2.** Assign Owner & Permission on NVMe targets by creating a new file named “99-oracleasm.rules” with the following entries on all the nodes

```
[root@orarac1 ~]# cat /etc/udev/rules.d/99-oracleasm.rules
#All volumes which starts with ocrvote* #
ENV{DM_NAME}=="ocrvote*", OWNER:="grid", GROUP:="oinstall", MODE:="660"

#All volumes which starts with dg_oradata_* #
ENV{DM_NAME}=="*data*", OWNER:="oracle", GROUP:="oinstall", MODE:="660"

#All volumes which starts with dg_oraredo_* #
ENV{DM_NAME}=="*log*", OWNER:="oracle", GROUP:="oinstall", MODE:="660"
```

### HugePages

HugePages is a method to have a larger page size that is useful for working with very large memory. For Oracle Databases, using HugePages reduces the operating system maintenance of page states, and increases Translation Lookaside Buffer (TLB) hit ratio.

Advantage of HugePages:

- HugePages are not swappable so there is no page-in/page-out mechanism overhead.
- HugePages uses fewer pages to cover the physical address space, so the size of " bookkeeping" (mapping from the virtual to the physical address) decreases, so it requires fewer entries in the TLB and so TLB hit ratio improves.
- HugePages reduces page table overhead. Also, HugePages eliminates page table lookup overhead: Since the pages are not subject to replacement, page table lookups are not required.
- Faster overall memory performance: On virtual memory systems, each memory operation is two abstract memory operations. Since there are fewer pages to work on, the possible bottleneck on page table access is avoided.

**Note:** For this configuration, HugePages were used for all the OLTP and DSS workloads. Refer to the Oracle guidelines to configure HugePages:
https://docs.oracle.com/en/database/oracle/oracle-database/21/ladbi/disabling-transparent-hugepages.html

**Procedure 1.** Run Cluster Verification Utility

This procedure verifies that all the prerequisites are met to install the Oracle Grid Infrastructure software. Oracle Grid Infrastructure ships with the Cluster Verification Utility (CVU) that can run to validate the pre and post installation configurations.

**Step 1.** Login as Grid User in Oracle RAC Node 1 and go to the directory where the Oracle Grid software binaries are located. Run the script named “runcluvfy.sh” as follows:

```
./runcluvfy.sh stage -pre crsinst -n
orarac1,orarac2,orarac3,orarac4,orarac5,orarac6,orarac7,orarac8 –verbose
```

After the configuration, you are ready to install the Oracle Grid Infrastructure and Oracle Database 21c software.

**Note:** For this solution, Oracle home binaries were installed on the boot LUN of the nodes. The OCR, Data, and Redo Log files reside in the volumes configured on Pure Storage array

## Oracle Database Grid Infrastructure Software

**Note:** It is not within the scope of this document to include the specifics of an Oracle RAC installation. However, a partial summary of details is provided that might be relevant. Please refer to the Oracle installation documentation for specific installation instructions for your environment.

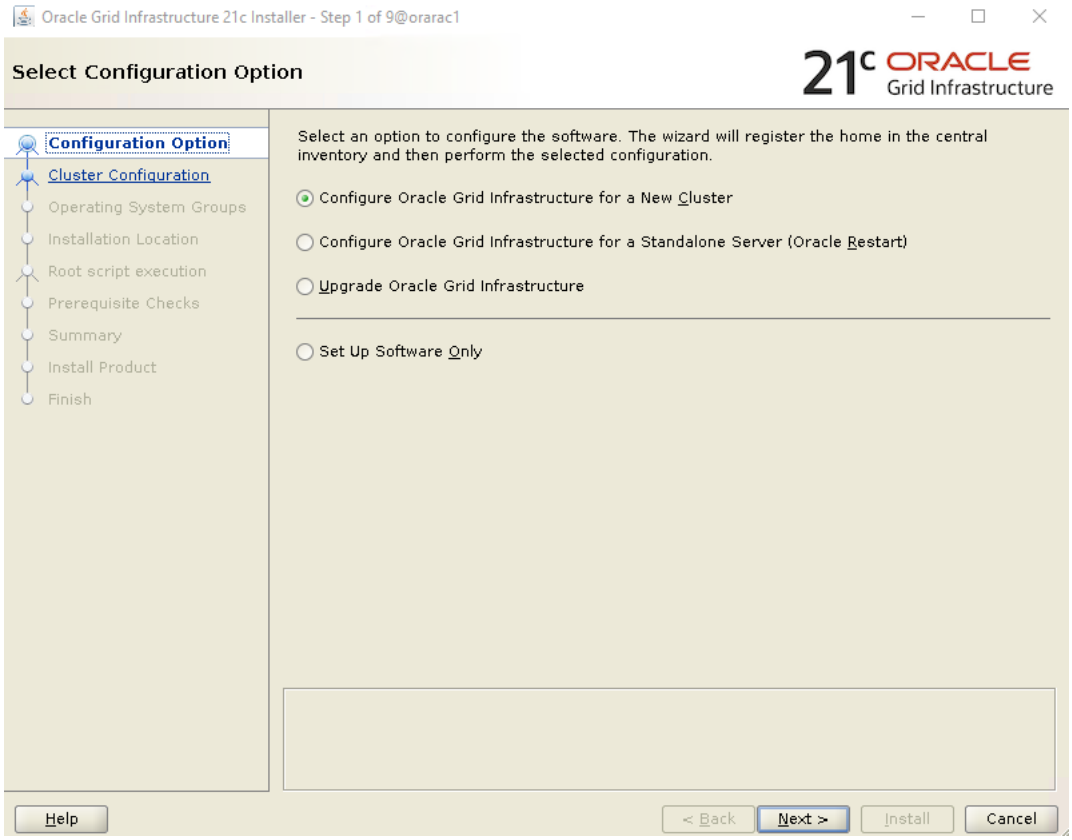
**Procedure 1.** Install and configure the Oracle Database Grid Infrastructure software

**Step 1.** Go to the Grid home where the Oracle 21c Grid Infrastructure software binaries are located and launch the installer as the " grid" user.

**Step 2.** Start the Oracle Grid Infrastructure installer by running the following command:

```
./gridSetup.sh
```

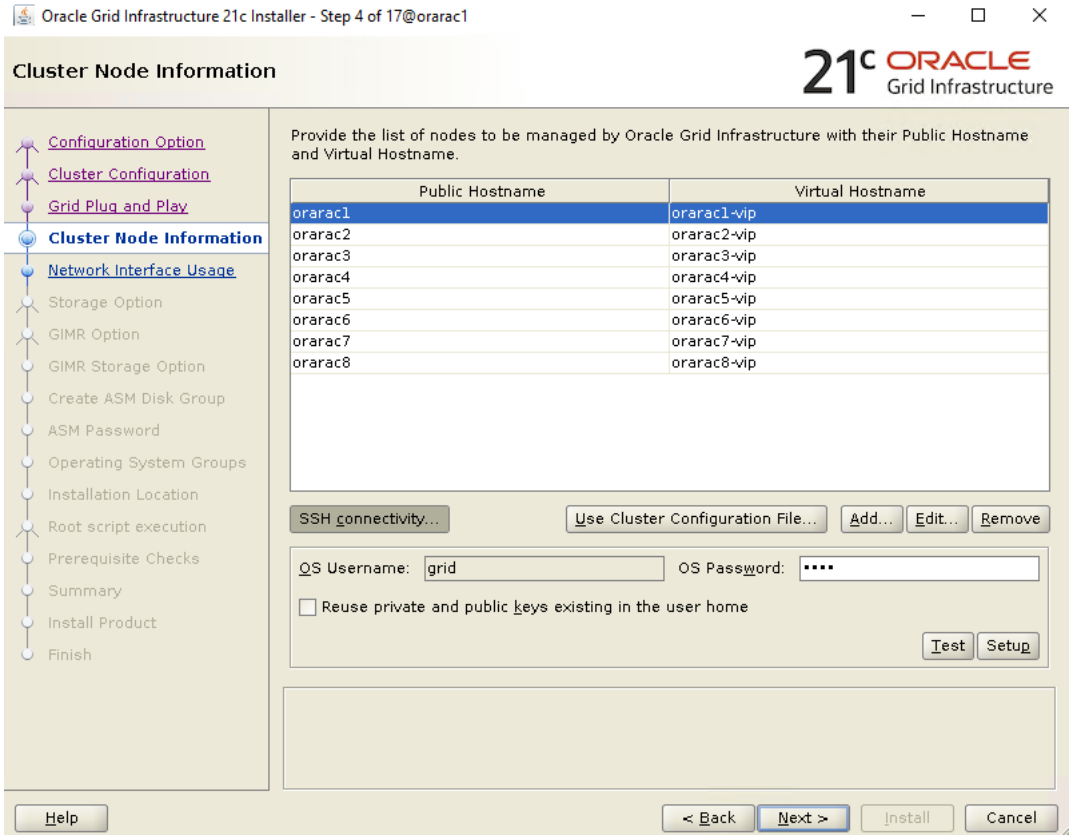
**Step 3.** Select the option Configure Oracle Grid Infrastructure for a New Cluster then click Next.




**Step 4.** For the Cluster Configuration select Configure an Oracle Standalone Cluster then clicks Next.

**Step 5.** In next window, enter the Cluster Name and SCAN Name fields. Enter the names for your cluster and cluster scan that are unique throughout your entire enterprise network. You can also select to Configure GNS if you have configured your domain name server (DNS) to send to the GNS virtual IP address name resolution requests.

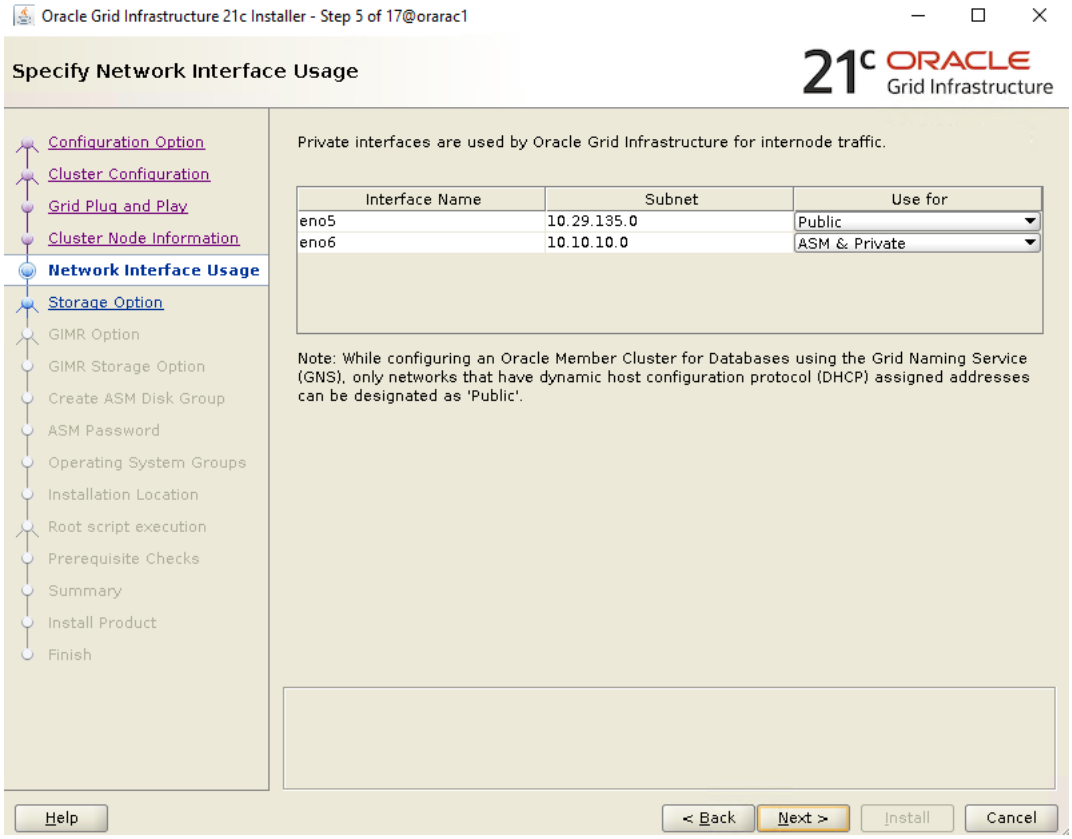
**Step 6.** In the Cluster node information window, click Add to add all eight nodes, Public Hostname and Virtual Host-name as shown below:




**Step 7.** You will see all nodes listed in the table of cluster nodes. Click the SSH Connectivity. Enter the operating system username and password for the Oracle software owner (grid). Click Setup.

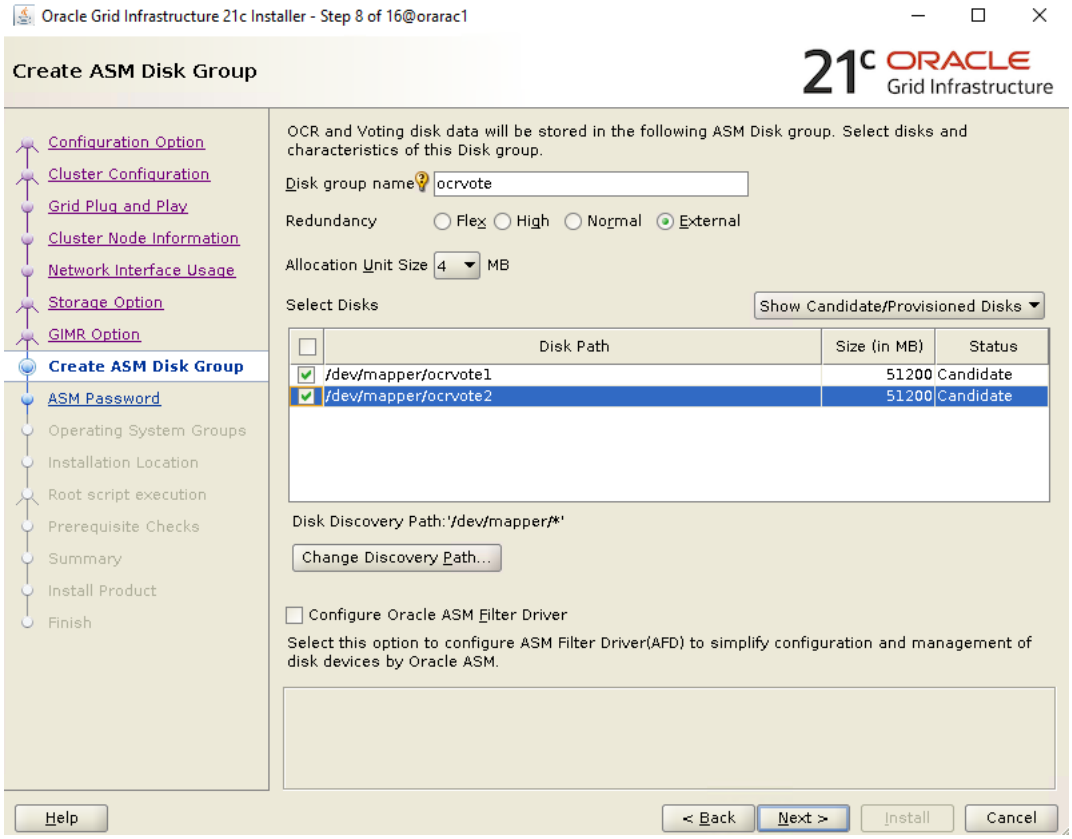
**Step 8.** A message window appears, indicating that it might take several minutes to configure SSH connectivity between the nodes. After some time, another message window appears indicating that password-less SSH connectivity has been established between the cluster nodes. Click OK to continue.

**Step 9.** In the Network Interface Usage screen, select the usage type for each network interface for Public and Private Network Traffic and click Next.




**Step 10.** In the storage option, select the option Use Oracle Flex ASM for storage then click Next. For this solution, the Do Not Use a GIMR database option was selected.

**Step 11.** In the Create ASM Disk Group window, select the “ocrvote1” & “ocrvote2” volumes which are configured into Pure Storage to store OCR and Voting disk files. Enter the name of disk group “OCRVOTE” and select appropriate external redundancy options as shown below:




**Note:** For this solution, we did not configure Oracle ASM Filter Driver.

**Step 12.** Select the password for the Oracle ASM account, then click Next:

**Step 13.** For this solution, “Do not use Intelligent Platform Management Interface (IPMI)” was selected. Click Next.

**Step 14.** You can configure to have this instance of the Oracle Grid Infrastructure and Oracle Automatic Storage Management to be managed by Enterprise Manager Cloud Control. For this solution, this option was not selected. You can choose to set it up according to your requirements.

**Step 15.** Select the appropriate operating system group names for Oracle ASM according to your environments.

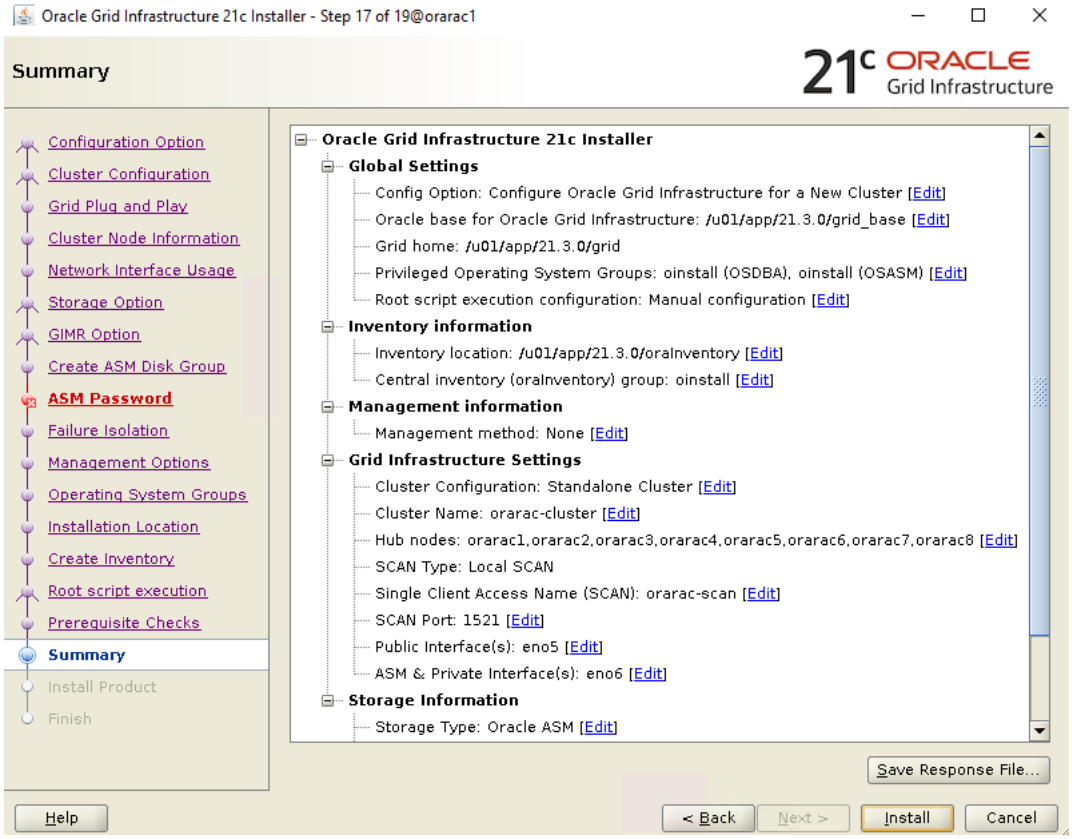
**Step 16.** Specify the Oracle base and inventory directory to use for the Oracle Grid Infrastructure installation and then click Next. The Oracle base directory must be different from the Oracle home directory. Click Next and select the Inventory Directory according to your setup.

**Step 17.** Click Automatically run configuration scripts to run scripts automatically and enter the relevant root user credentials. Click Next.

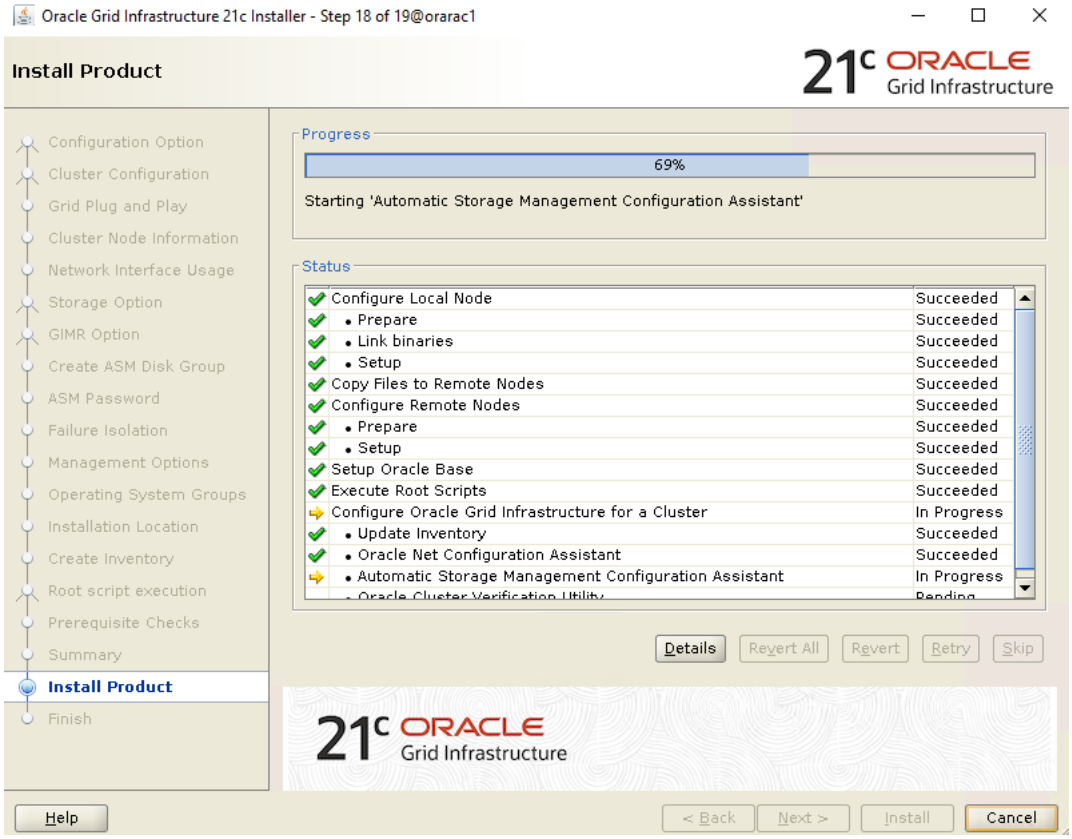
**Step 18.** Wait while the prerequisite checks complete.

**Step 19.** If you have any issues, click the " Fix & Check Again." If any of the checks have a status of Failed and are not fixable, then you must manually correct these issues. After you have fixed the issue, you can click Check Again to have the installer check the requirement and update the status. Repeat as needed until all the checks have a status of Succeeded. Click Next.

**Step 20.** Review the contents of the Summary window and then click Install. The installer displays a progress indicator enabling you to monitor the installation process.




**Step 21.** Wait for the grid installer configuration assistants to complete.



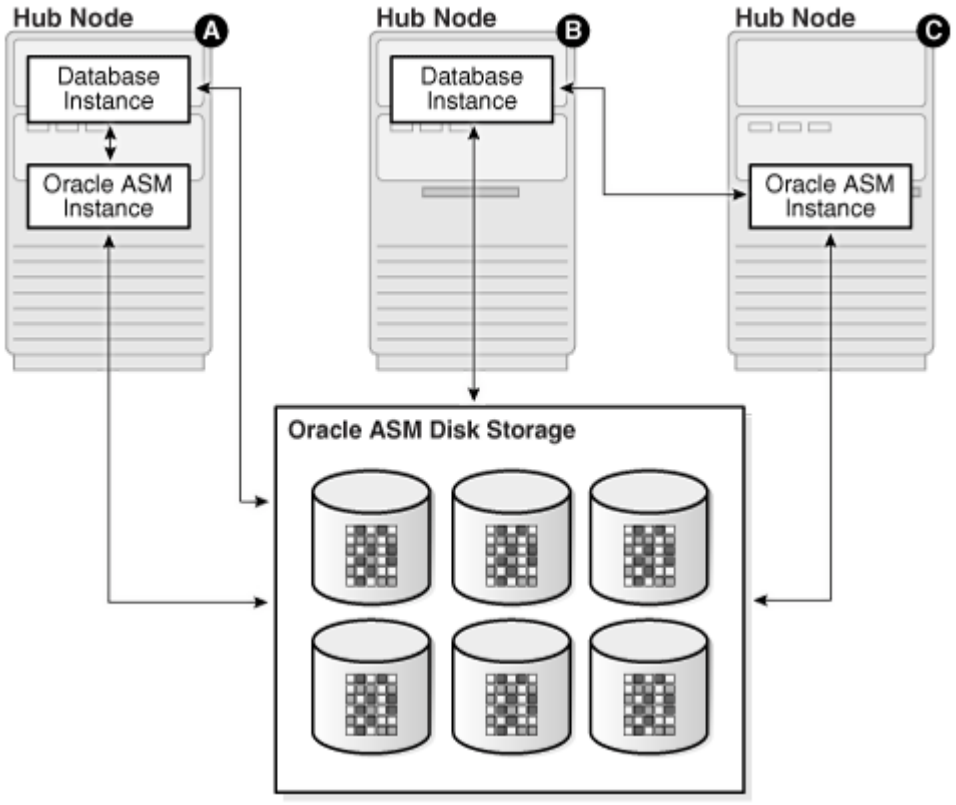

**Step 22.** When the configuration completes successfully, click Close to finish, and exit the grid installer.

**Step 23.** When the GRID installation is successful, login to each of the nodes and perform the minimum health checks to make sure that the Cluster state is healthy. After your Oracle Grid Infrastructure installation is complete, you can install Oracle Database on a cluster.

## Overview of Oracle Flex ASM

Oracle ASM is Oracle's recommended storage management solution that provides an alternative to conventional volume managers, file systems, and raw devices. Oracle ASM is a volume manager and a file system for Oracle Database files that reduces the administrative overhead for managing database storage by consolidating data storage into a small number of disk groups. The smaller number of disk groups consolidates the storage for multiple databases and provides for improved I/O performance.

Oracle Flex ASM enables an Oracle ASM instance to run on a separate physical server from the database servers. With this deployment, larger clusters of Oracle ASM instances can support more database clients while reducing the Oracle ASM footprint for the overall system.




When using Oracle Flex ASM, Oracle ASM clients are configured with direct access to storage. With Oracle Flex ASM, you can consolidate all the storage requirements into a single set of disk groups. All these disk groups are mounted by and managed by a small set of Oracle ASM instances running in a single cluster. You can specify the number of Oracle ASM instances with a cardinality setting. The default is three instances.

The following screenshot shows few more commands to check the cluster and FLEX ASM details:

```
[grid@orarac1 ~]$ crsctl check cluster
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
[grid@orarac1 ~]$
[grid@orarac1 ~]$ clear
[grid@orarac1 ~]$
[grid@orarac1 ~]$ crsctl check cluster
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
[grid@orarac1 ~]$ srvctl status asm -detail
ASM is running on orarac1,orarac2,orarac3
ASM is enabled.
ASM instance +ASM3 is running on node orarac3
Number of connected clients: 1
Client names: orarac3:_OCR:orarac-cluster
ASM instance +ASM1 is running on node orarac1
Number of connected clients: 1
Client names: orarac1:_OCR:orarac-cluster
ASM instance +ASM2 is running on node orarac2
Number of connected clients: 1
Client names: orarac2:_OCR:orarac-cluster
[grid@orarac1 ~]$ srvctl config asm -detail
ASM home: <CRS home>
Password file: +ocrvote/orapwASM
Backup of Password file: +ocrvote/orapwASM_backup
ASM listener: LISTENER
ASM is enabled.
ASM is individually enabled on nodes:
ASM is individually disabled on nodes:
ASM instance count: 3
Cluster ASM listener: ASMNET1LSNR_ASM
[grid@orarac1 ~]$ asmcmd
ASMCMD> showclustermode
ASM cluster : Flex mode enabled - Direct Storage Access
ASMCMD> showclusterstate
Normal
```

## Oracle Database Installation

After successfully installing the Oracle GRID, it's recommended to only install the Oracle Database 21c software. You can create databases using DBCA or database creation scripts at later stage.

**Note:** It is not within the scope of this document to include the specifics of an Oracle RAC database installation. However, a partial summary of details is provided that might be relevant. Please refer to the Oracle database installation documentation for specific installation instructions for your environment here: https://docs.oracle.com/en/database/oracle/oracle-database/21/ladbi/index.html

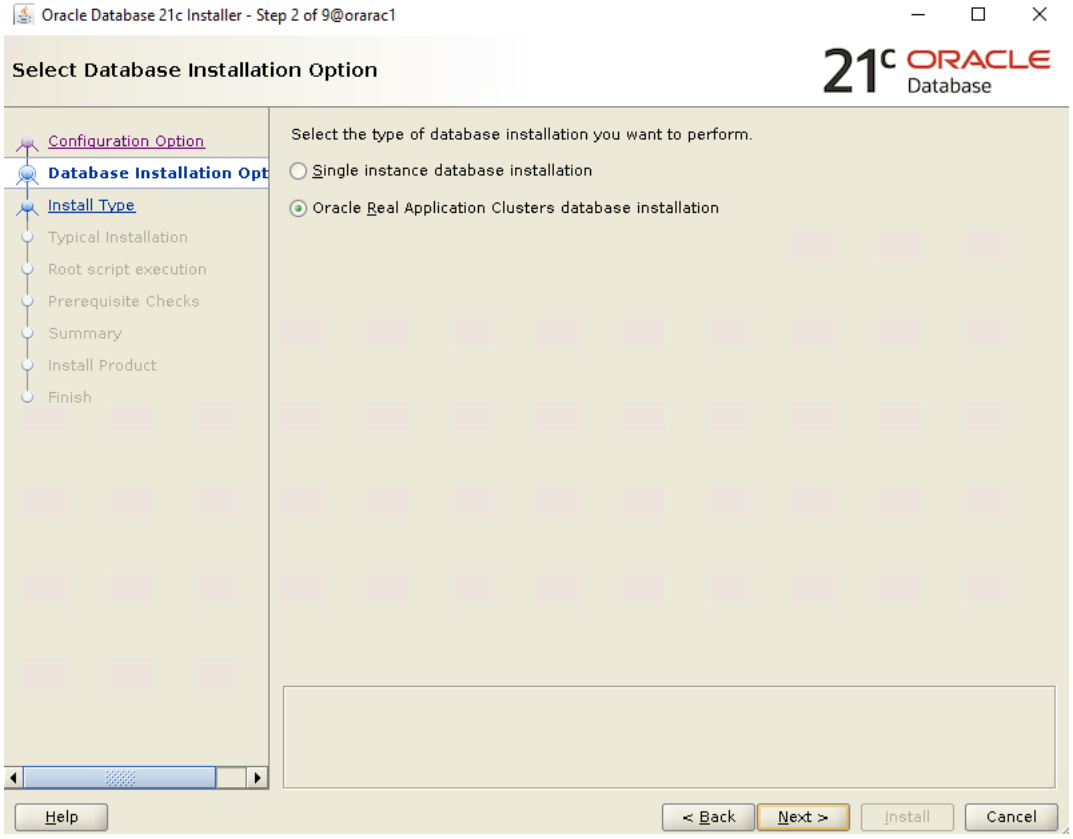
**Procedure 1.** Install Oracle database software

Complete the following steps as an "oracle" user.

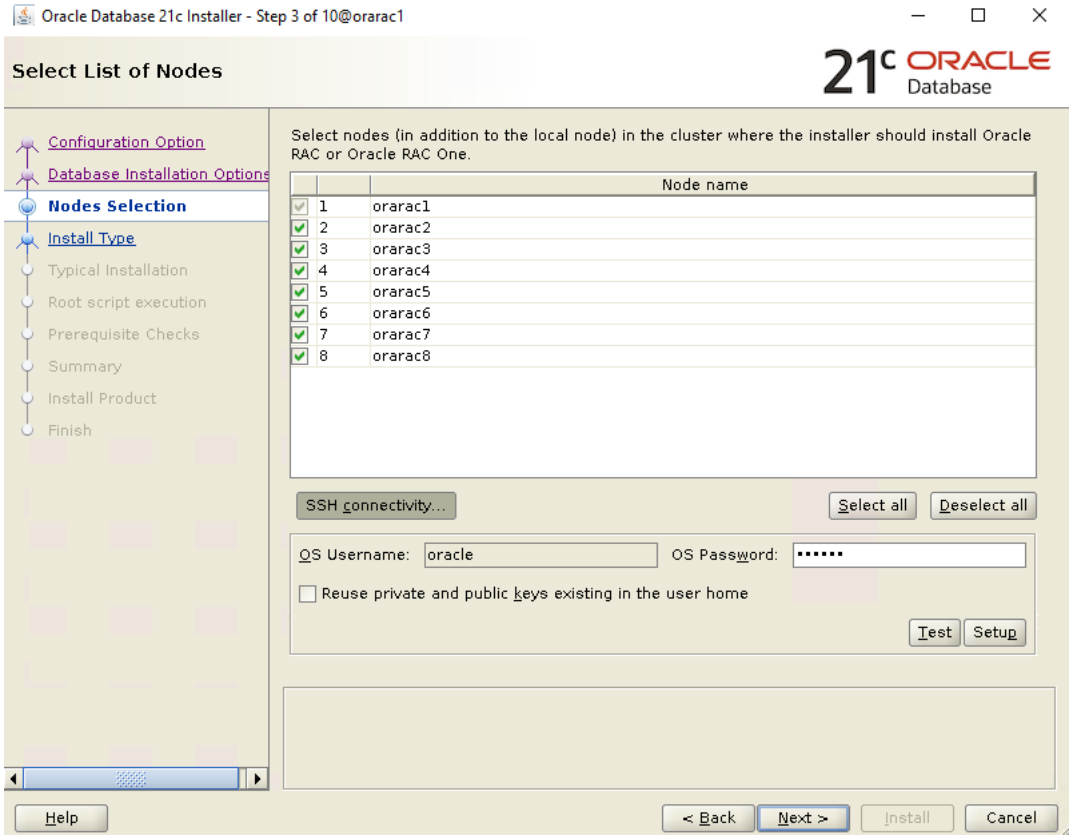
**Step 1.** Start the "./runInstaller" command from the Oracle Database 21c installation media where the Oracle database software is located.

**Step 2.** For Configuration Option, select the option Set Up Software Only.

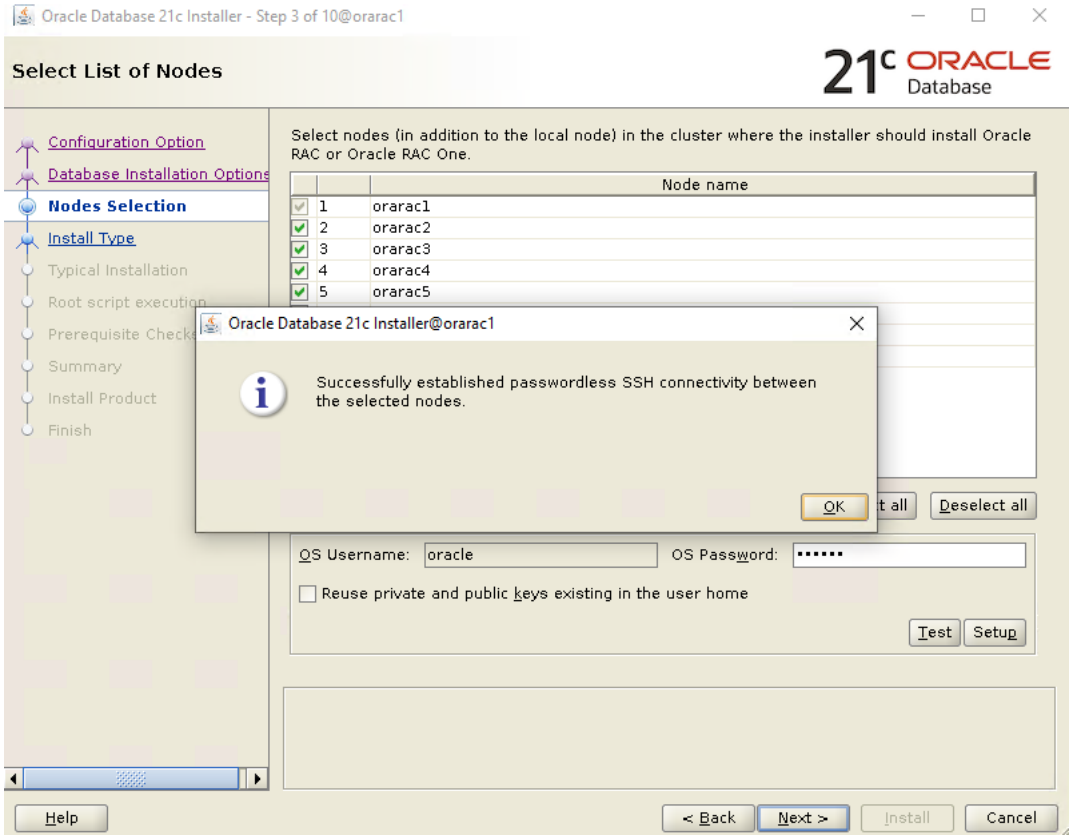
**Step 3.** Select the option " Oracle Real Application Clusters database installation" and click Next.




**Step 4.** Select all eight nodes in the cluster where the installer should install Oracle RAC. For this setup, install the software on all eight nodes as shown below:




**Step 5.** Click "SSH Connectivity..." and enter the password for the "oracle" user. Click Setup to configure passwordless SSH connectivity and click Test to test it when it is complete. When the test is complete, click Next.




**Step 6.** Select the Database Edition Options according to your environments and then click Next.

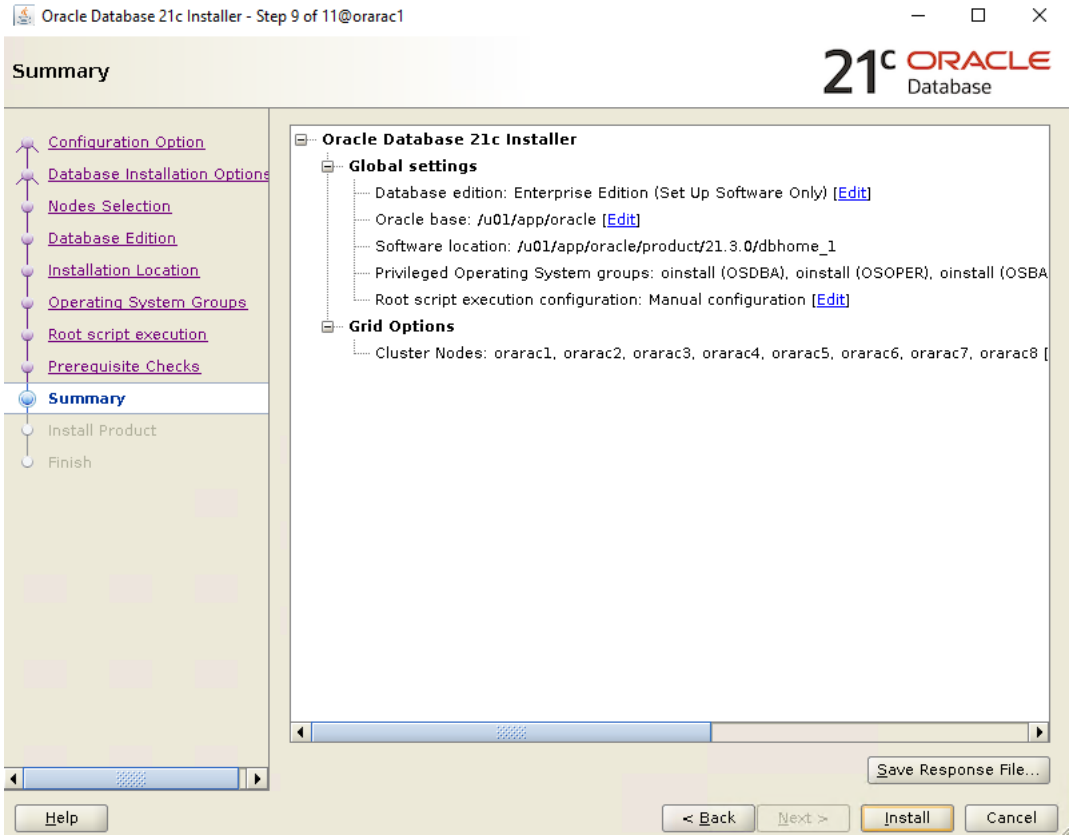
**Step 7.** Enter the appropriate Oracle Base, then click Next.

**Step 8.** Select the desired operating system groups and then click Next.

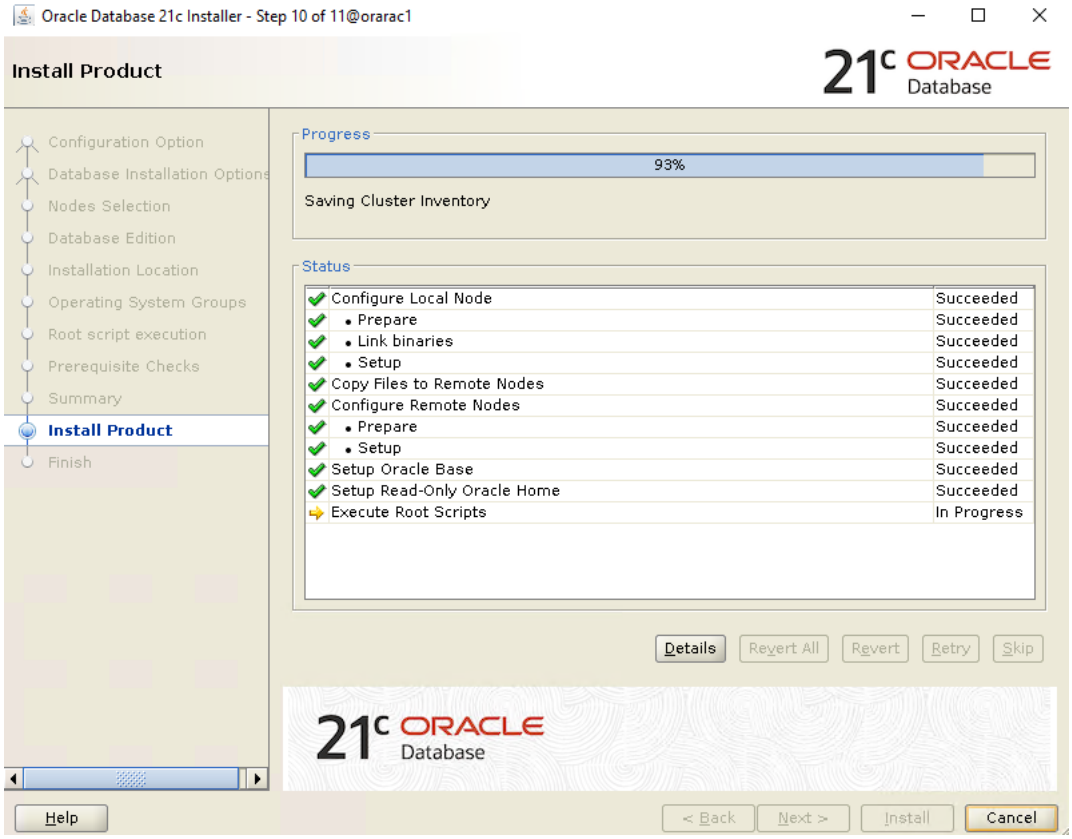
**Step 9.** Select the option Automatically run configuration script from the option Root script execution menu and click Next.

**Step 10.** Wait for the prerequisite check to complete. If there are any problems, click "Fix & Check Again" or try to fix those by checking and manually installing required packages. Click Next.

**Step 11.** Verify the Oracle Database summary information and then click Install.




**Step 12.** Wait for the installation of Oracle Database finish successfully, then click Close to exit of the installer.

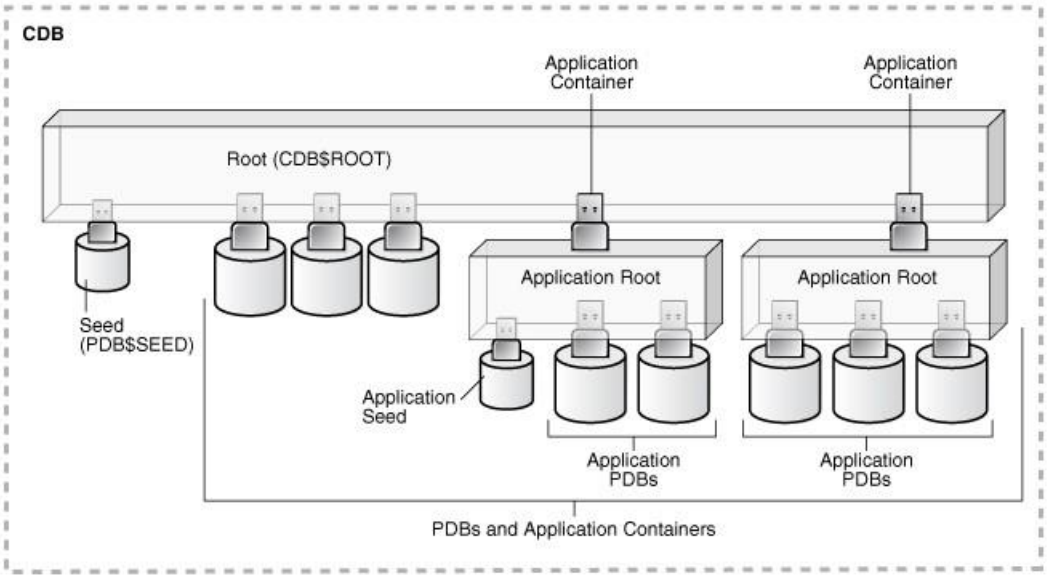



These steps complete the installation of the Oracle 21c Grid Infrastructure and Oracle 21c Database software.

## Oracle Database Multitenant Architecture

The multitenant architecture enables an Oracle database to function as a multitenant container database (CDB). A CDB includes zero, one, or many customer-created pluggable databases (PDBs). A PDB is a portable collection of schemas, schema objects, and non-schema objects that appears to an Oracle Net client as a non-CDB. All Oracle databases before Oracle Database 12c were non-CDBs.

A container is logical collection of data or metadata within the multitenant architecture. The following figure represents possible containers in a CDB:




The multitenant architecture solves several problems posed by the traditional non-CDB architecture. Large enterprises may use hundreds or thousands of databases. Often these databases run on different platforms on multiple physical servers. Because of improvements in hardware technology, especially the increase in the number of CPUs, servers can handle heavier workloads than before. A database may use only a fraction of the server hardware capacity. This approach wastes both hardware and human resources. Database consolidation is the process of consolidating data from multiple databases into one database on one computer. The Oracle Multitenant option enables you to consolidate data and code without altering existing schemas or applications.

For more information on Oracle Database Multitenant Architecture, go to:
https://docs.oracle.com/en/database/oracle/oracle-database/21/cncpt/CDBs-and-PDBs.html#GUID-5C339A60-2163-4ECE-B7A9-4D67D3D894FB

In this solution, multiple Container Databases were configured and validated system performance as explained in the next scalability test section.

Now you are ready to run synthetic IO tests against this infrastructure setup. “fio” was used as primary tools for IOPS tests.

## Scalability Test and Results

This chapter contains the following:



**Note:** Before creating databases for workload tests, it is extremely important to validate that this is indeed a balanced configuration that can deliver expected performance. In this solution, node and user scalability will be tested and validated on all eight node Oracle RAC Databases with various database benchmarking tools.

### Hardware Calibration Test using FIO

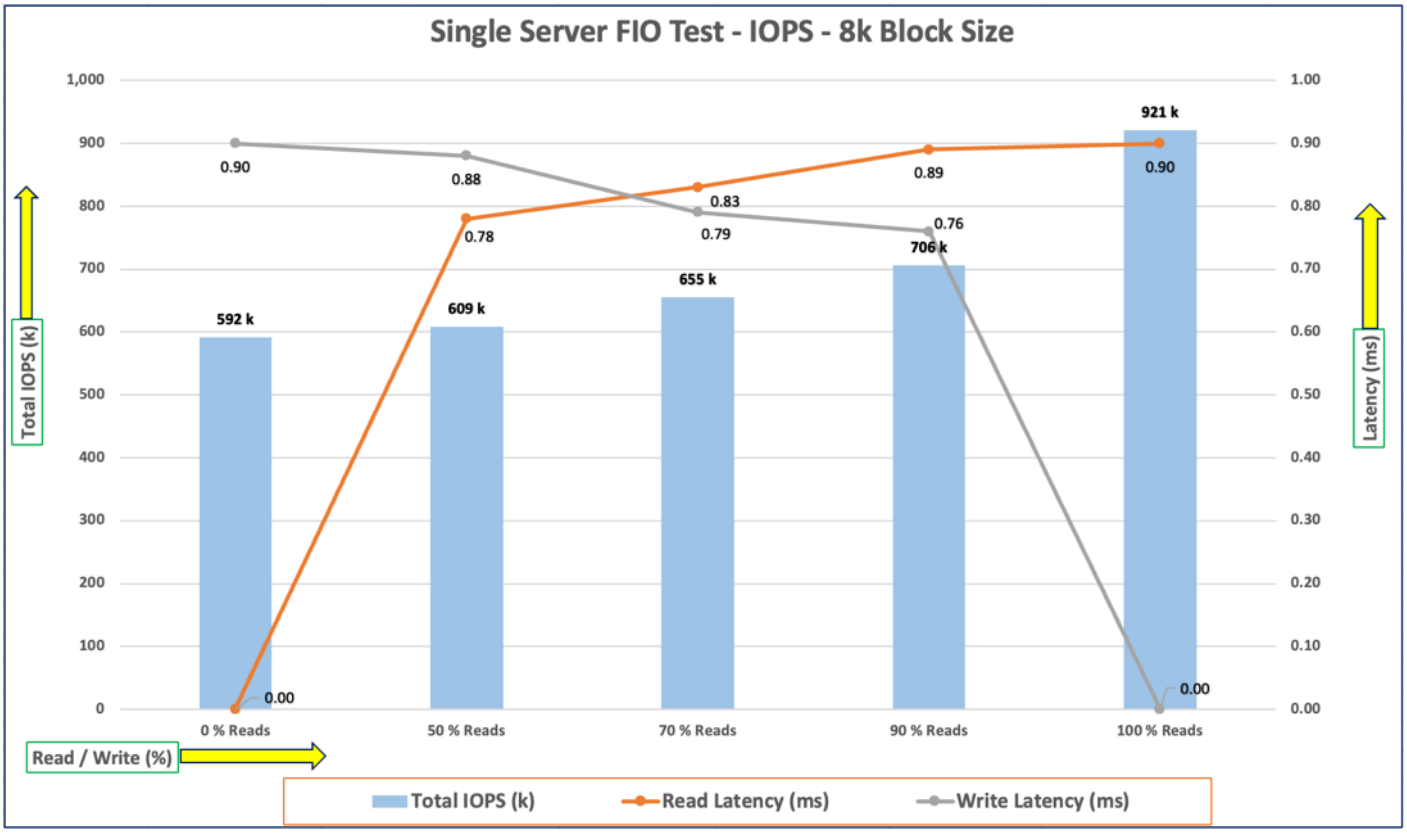
FIO is short for Flexible IO, a versatile IO workload generator. FIO is a tool that will spawn number of threads or processes doing a particular type of I/O action as specified by the user. For this solution, FIO is used to measure the performance of a Pure Storage FlashArray over a given period.

For the FIO Tests, we have created 8 volumes of 1 TB in size and were shared across all the eight nodes for read/write IO operations.

We run various FIO tests for measuring IOPS, Latency and Throughput performance of this solution by changing block size parameter into the FIO test. For each FIO test, we also changed read/write ratio as 0/100 % read/write, 50/50 % read/write, 70/30 % read/write, 90/10 % read/write and 100/0 % read/write to scale the performance of the storage system. We also ran each of the tests for at least 6 hours to help ensure that this configuration can sustain this type of load for longer period of time.

### IOPS Tests on Single x210c M7 Server

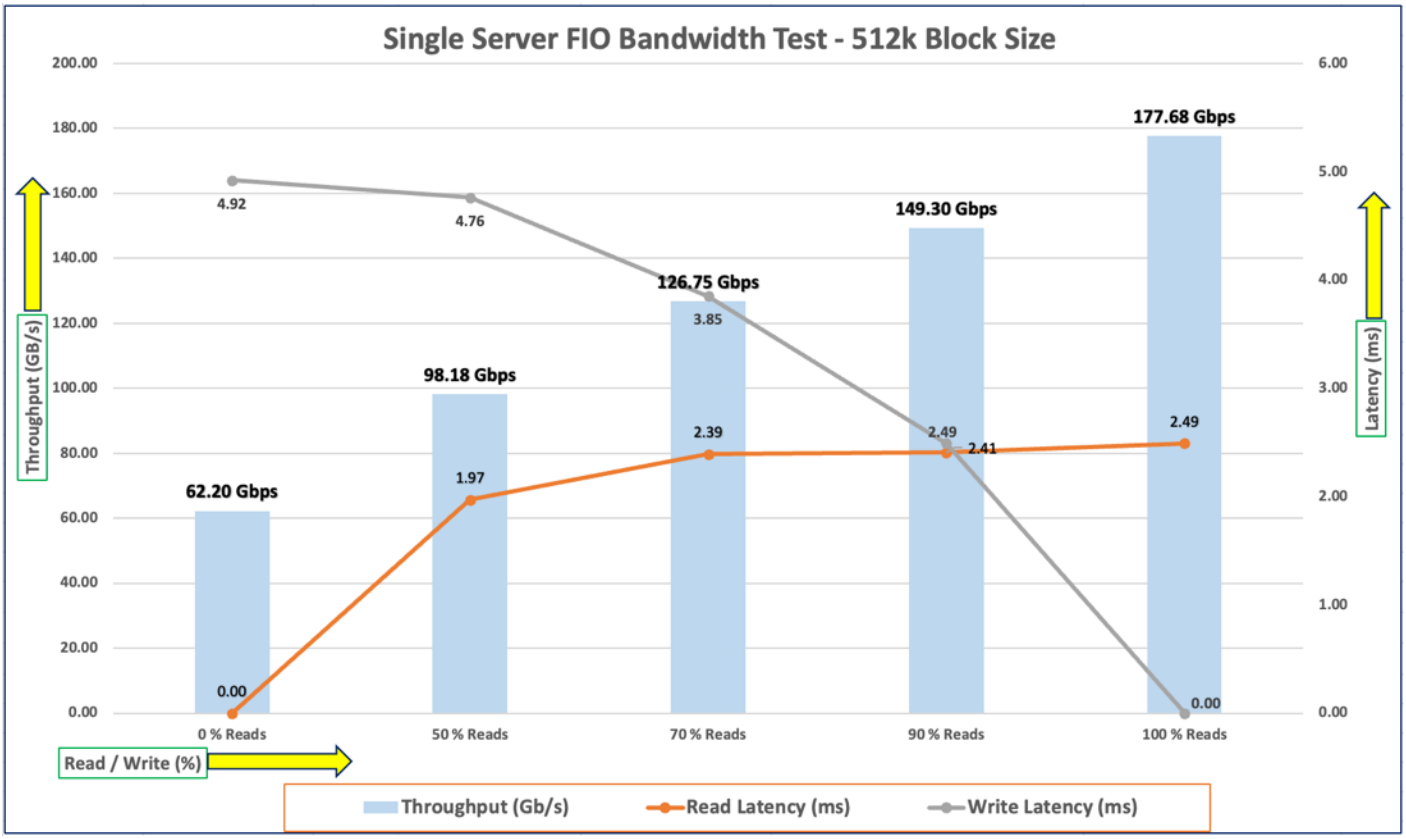
This type of FIO IOPS tests with 8k block size represents OLTP type of workloads. For this single server node IOPS scale, we used FIO with random read/write tests, changed read/write ratio and captured all the output as shown in the chart below:




For the single server node, we observed average 921k IOPS for 100/0 % read/write test with the read latency under 1 millisecond. Similarly, for the 90/10 % read/write test, we achieved around 706k IOPS and for the 70/30 % read/write test, we achieved around 655k IOPS with the read and write latency under 0.8 millisecond. For the 50/50 % read/write test, we achieved around 609k IOPS and for the 0/100 % read/write test, we achieved around 592k IOPS with the write and read latency under 1 millisecond.

## Bandwidth Tests

The bandwidth tests are carried out with sequential 512k IO Size and represents the DSS database type workloads. The chart below shows results for the various sequential read/write FIO test for the 512k block size. We ran bandwidth test on single x210c M7 server and captured the results as shown below:




For the 100/0 % read/write test, we achieved around 177 Gbps throughput with the read latency around 2.5 millisecond. Similarly, for the 90/10 % read/write test, we achieved around 149 Gbps throughput with the read and write latency under 2.5 milliseconds. For the 70/30 % read/write bandwidth test, we achieved around 126 Gbps throughput with the read latency around 2.3 milliseconds while the write latency around 3.8 milliseconds. For the 50/50 % read/write test, we achieved around 98 Gbps throughput with the read and write latency under 4 milliseconds And lastly, for the 0/100 % read/write test, we achieved around 62 Gbps throughput with the write latency around 5 milliseconds.

We did not see any performance dips or degradation over the period of run time. It is also important to note that this is not a benchmarking exercise, and these are practical and out of box test numbers that can be easily reproduced by anyone. At this time, we are ready to create OLTP database(s) and continue with database tests.

## Database Creation with DBCA

We used Oracle Database Configuration Assistant (DBCA) to create multiple OLTP and DSS databases for SLOB and SwingBench test calibration. For SLOB Tests, we configured one container database as “SLOBCDB” and under this container, we create one pluggable database as “SLOBPDB.” For SwingBench SOE (OLTP type) workload tests, we configured two container databases as “SOECDB” and “ENGCDB”. Under these two containers, we created one pluggable database on each container as “SOEPDB” and “ENGPDB” to demonstrate the system scalability running multiple OLTP container and pluggable databases for various SOE workloads. For SwingBench SH (DSS type) workload tests, we configured one container database as “SHCDB” and under this container, we created one pluggable database as “SHPDB.” Alternatively, you can use Database creation scripts to create the databases as well.

For each RAC database, we created a total number of 10 volumes. We connected all these volumes across eight nodes through single host group ORARAC as explained earlier. All database files were also spread evenly across the storage system so that each storage node served data for the databases. Table 13 lists the storage layout of all the volume configuration for all the databases. For each database, we created two disk groups to store the "data" and "log" files for storing the database files. We used 8 volumes to create Oracle ASM "Data" disk group and 2 volumes to create Oracle ASM "log" disk group for each database.

Table 13 lists the database volume configuration for this solution where we deployed all three databases to validate SLOB and SwingBench workloads.

**Table 13. Database volume configuration**

| Database Name | Volumes | Size (GB) | Notes |
|---|---|---|---|
| OCRVOTE | ocrvote1 | 50 | OCR & Voting Disk |
| | ocrvote2 | 50 | |
| SLOBCDB<br><br>(Container SLOBCDB with Pluggable Database as SLOBPDB) | slobdata01 | 800 | SLOB Database Data Files |
| | slobdata02 | 800 | |
| | slobdata03 | 800 | |
| | slobdata04 | 800 | |
| | slobdata05 | 800 | |
| | slobdata06 | 800 | |
| | slobdata07 | 800 | |
| | slobdata08 | 800 | |
| | sloblog01 | 100 | SLOB Database Redo Log Files |
| | sloblog02 | 100 | |
| SOECDB<br><br>(Container SOECDB with One Pluggable Database as SOEPDB) | soedata01 | 1500 | SOECDB Database Data Files |
| | soedata02 | 1500 | |
| | soedata03 | 1500 | |
| | soedata04 | 1500 | |
| | soedata05 | 1500 | |

| Database Name | Volumes | Size (GB) | Notes |
|---|---|---|---|
| | soedata06 | 1500 | |
| | soedata07 | 1500 | |
| | soedata08 | 1500 | |
| | soelog01 | 100 | SOECDB Database Redo Log Files |
| | soelog02 | 100 | |
| ENGCDB<br><br>(Container ENGCDB with One Pluggable Database as PDB) | engdata01 | 1200 | ENGCDB Database Data Files |
| | engdata02 | 1200 | |
| | engdata03 | 1200 | |
| | engdata04 | 1200 | |
| | engdata05 | 1200 | |
| | engdata06 | 1200 | |
| | engdata07 | 1200 | |
| | engdata08 | 1200 | |
| | englog01 | 100 | ENGCDB Database Redo Log Files |
| | englog02 | 100 | |
| SHCDB<br><br>(Container SHCDB with One Pluggable Database as SHPDB) | shdata01 | 2000 | SH Database Data Files |
| | shdata02 | 2000 | |
| | shdata03 | 2000 | |
| | shdata04 | 2000 | |
| | shdata05 | 2000 | |
| | shdata06 | 2000 | |
| | shdata07 | 2000 | |

| Database Name | Volumes | Size (GB) | Notes |
|---|---|---|---|
| | shdata08 | 2000 | |
| | shlog01 | 100 | SH Database Redo Log Files |
| | shlog02 | 100 | |

We used the widely adopted SLOB and Swingbench database performance test tools to test and validate throughput, IOPS, and latency for various test scenarios as explained in the following section.

## SLOB Test

The Silly Little Oracle Benchmark (SLOB) is a toolkit for generating and testing I/O through an Oracle database. SLOB is very effective in testing the I/O subsystem with genuine Oracle SGA-buffered physical I/O. SLOB supports testing physical random single-block reads (db file sequential read) and random single block writes (DBWR flushing capability). SLOB issues single block reads for the read workload that are generally 8K (as the database block size was 8K).

For testing the SLOB workload, we have created one container database as SLOBCDB. For SLOB database, we have created total 10 volumes. On these 10 volumes, we have created two disk groups to store the “data” and “log” files for the SLOB database. First disk-group “SLOBDATA” was created with 8 volumes (800 GB each) while second disk-group “SLOBLOG” was created with 2 volumes (100 GB each).

Those ASM disk groups provided the storage required to create the tablespaces for the SLOB Database. We loaded SLOB schema on “DATASLOB” disk-group of up to 3 TB in size.

We used SLOB2 to generate our OLTP workload. Each database server applied the workload to Oracle database, log, and temp files. The following tests were performed and various metrics like IOPS and latency were captured along with Oracle AWR reports for each test scenario.

### User Scalability Test

SLOB2 was configured to run against all the eight Oracle RAC nodes and the concurrent users were equally spread across all the nodes. We tested the environment by increasing the number of Oracle users in database from a minimum of 128 users up to a maximum of 512 users across all the nodes. At each load point, we verified that the storage system and the server nodes could maintain steady-state behavior without any issues. We also made sure that there were no bottlenecks across servers or networking systems.

The User Scalability test was performed with 128, 256, 384 and 512 users on 8 Oracle RAC nodes by varying read/write ratio as follows:

- 100% read (0% update)
- 90% read (10% update)
- 70% read (30% update)
- 50% read (50% update)

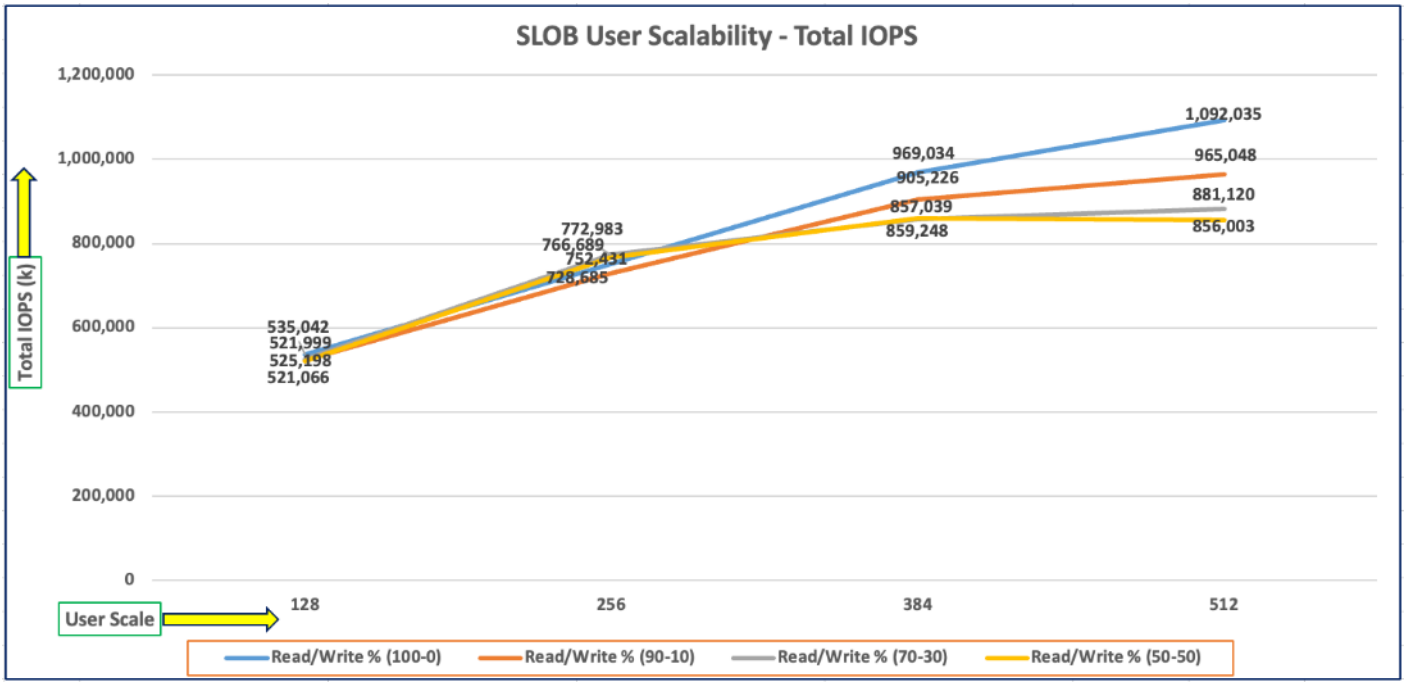
Table 14 lists the total number of IOPS (both read and write) available for user scalability test when run with 128, 256, 384 and 512 Users on the SLOB database.

**Table 14. Total number of IOPS**

| Users | Read/Write % (100-0) | Read/Write % (90-10) | Read/Write % (70-30) | Read/Write % (50-50) |
|---|---|---|---|---|
| 128 | 535,042 | 521,999 | 525,198 | 521,066 |
| 256 | 752,431 | 728,685 | 772,983 | 766,689 |
| 384 | 969,034 | 905,226 | 857,039 | 859,248 |
| 512 | 1,092,035 | 965,048 | 881,120 | 856,003 |

The following graphs demonstrate the total number of IOPS while running SLOB workload for various concurrent users for each test scenario.

The graph below shows the linear scalability with increased users and similar IOPS from 128 users to 512 users with 100% Read/Write, 90% Read/Write, 70% Read/Write and 50% Read/Write.




Due to variations in workload randomness, we conducted multiple runs to ensure consistency in behavior and test results. The AWR screenshot below was captured from one of the test run scenarios for 100% Read with 512 users running SLOB workload for 4 hours across all eight nodes.

Database Summary

| Database | | | | | | | Snapshot Ids | | Number of Instances | | Number of Hosts | | Report Total (minutes) | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Id | Name | Unique Name | Role | Edition | RAC | CDB | Block Size | Begin | End | In Report | Total | In Report | Total | DB time | Elapsed time |
| 2435367805 | SLOBCDB | slobcdb | PRIMARY | EE | YES | YES | 8192 | 115 | 120 | 8 | 8 | 8 | 8 | 123,019.14 | 240.28 |

Database Instances Included In Report
-> Listed in order of instance number, I#

| I# | Instance | Host | Startup | Begin Snap Time | End Snap Time | Release | Elapsed Time(min) | DB time(min) | Up Time(hrs) | Avg Active Sessions | Platform |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | slobcdb1 | orarac1 | 14-Mar-24 23:01 | 15-Mar-24 23:12 | 16-Mar-24 03:12 | 21.0.0.0.0 | 240.23 | 15,377.77 | 28.19 | 64.01 | Linux x86 64-bi |
| 2 | slobcdb2 | orarac2 | 14-Mar-24 23:01 | 15-Mar-24 23:12 | 16-Mar-24 03:12 | 21.0.0.0.0 | 240.23 | 15,377.14 | 28.19 | 64.01 | Linux x86 64-bi |
| 3 | slobcdb3 | orarac3 | 14-Mar-24 23:01 | 15-Mar-24 23:12 | 16-Mar-24 03:12 | 21.0.0.0.0 | 240.23 | 15,377.08 | 28.19 | 64.01 | Linux x86 64-bi |
| 4 | slobcdb4 | orarac4 | 14-Mar-24 23:01 | 15-Mar-24 23:12 | 16-Mar-24 03:12 | 21.0.0.0.0 | 240.23 | 15,376.92 | 28.19 | 64.01 | Linux x86 64-bi |
| 5 | slobcdb5 | orarac5 | 14-Mar-24 23:01 | 15-Mar-24 23:12 | 16-Mar-24 03:12 | 21.0.0.0.0 | 240.23 | 15,377.91 | 28.19 | 64.01 | Linux x86 64-bi |
| 6 | slobcdb6 | orarac6 | 14-Mar-24 23:01 | 15-Mar-24 23:12 | 16-Mar-24 03:12 | 21.0.0.0.0 | 240.23 | 15,377.63 | 28.19 | 64.01 | Linux x86 64-bi |
| 7 | slobcdb7 | orarac7 | 14-Mar-24 23:01 | 15-Mar-24 23:12 | 16-Mar-24 03:12 | 21.0.0.0.0 | 240.23 | 15,377.67 | 28.19 | 64.01 | Linux x86 64-bi |
| 8 | slobcdb8 | orarac8 | 14-Mar-24 23:01 | 15-Mar-24 23:12 | 16-Mar-24 03:12 | 21.0.0.0.0 | 240.22 | 15,377.01 | 28.19 | 64.01 | Linux x86 64-bi |

Open Pluggable Databases at Begin Snap: 3, End Snap: 3

The following screenshot shows a section from the Oracle AWR report that highlights Physical Reads/Sec and Physical Writes/Sec for each instance while running SLOB workload for the same above 100% read tests with 512 users running the workload. It highlights that IO load is distributed across all the cluster nodes performing workload operations.

System Statistics - Per Second  DB/Inst: SLOBCDB/slobcdb1  Snaps: 115-120

| I# | Logical Reads/s | Physical Reads/s | Physical Writes/s | Redo Size (k)/s | Block Changes/s | User Calls/s | Execs/s | Parses/s | Logons/s | Txns/s |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 164,493.85 | 149,030.0 | 1.5 | 4.1 | 11.5 | 43.2 | 2,708.9 | 4.1 | 10.40 | 0.1 |
| 2 | 164,216.05 | 149,032.4 | 1.9 | 4.2 | 11.3 | 42.8 | 2,550.8 | 4.6 | 10.34 | 0.1 |
| 3 | 164,186.77 | 148,966.7 | 1.8 | 8.6 | 43.5 | 43.0 | 2,555.8 | 20.8 | 10.39 | 0.1 |
| 4 | 164,142.35 | 149,000.3 | 1.4 | 4.0 | 9.7 | 42.8 | 2,533.6 | 4.0 | 10.36 | 0.1 |
| 5 | 164,426.44 | 149,034.8 | 1.2 | 3.9 | 9.3 | 42.6 | 2,670.3 | 3.8 | 10.30 | 0.1 |
| 6 | 164,151.75 | 148,967.6 | 1.6 | 4.0 | 9.8 | 42.4 | 2,539.7 | 4.0 | 10.25 | 0.1 |
| 7 | 164,185.13 | 148,976.8 | 1.4 | 4.0 | 10.4 | 42.2 | 2,548.6 | 4.9 | 10.21 | 0.1 |
| 8 | 164,153.44 | 149,014.5 | 1.3 | 3.4 | 8.9 | 42.0 | 2,539.5 | 4.4 | 10.16 | 0.1 |
| Sum | 1,313,955.78 | 1,192,023.1 | 12.2 | 36.0 | 114.4 | 341.0 | 20,647.2 | 50.6 | 82.42 | 0.7 |
| Avg | 164,244.47 | 149,002.9 | 1.5 | 4.5 | 14.3 | 42.6 | 2,580.9 | 6.3 | 10.30 | 0.1 |
| Std | 136.42 | 29.3 | 0.2 | 1.7 | 11.8 | 0.4 | 68.2 | 5.9 | 0.09 | 0.0 |

We also run one of the above SLOB test for sustained 24 hours with 512 Users with 70% Read (30% Update) SLOB workload and captured the results. The AWR screenshot below was captured from this sustained 24 hours test with 512 users running SLOB workload across all eight nodes.

Database Summary

| Database | | | | | | | Snapshot Ids | | Number of Instances | | Number of Hosts | | Report Total (minutes) | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Id | Name | Unique Name | Role | Edition | RAC | CDB | Block Size | Begin | End | In Report | Total | In Report | Total | DB time | Elapsed time |
| 2435367805 | SLOBCDB | slobcdb | PRIMARY | EE | YES | YES | 8192 | 161 | 185 | 8 | 8 | 8 | 8 | 737,584.91 | 1,440.37 |

Database Instances Included In Report
-> Listed in order of instance number, I#

| I# | Instance | Host | Startup | Begin Snap Time | End Snap Time | Release | Elapsed Time(min) | DB time(min) | Up Time(hrs) | Avg Active Sessions | Platform |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | slobcdb1 | orarac1 | 17-Mar-24 14:31 | 17-Mar-24 14:34 | 18-Mar-24 14:34 | 21.0.0.0.0 | 1,440.33 | 92,198.65 | 24.05 | 64.01 | Linux x86 64-bi |
| 2 | slobcdb2 | orarac2 | 17-Mar-24 14:31 | 17-Mar-24 14:34 | 18-Mar-24 14:34 | 21.0.0.0.0 | 1,440.33 | 92,198.06 | 24.05 | 64.01 | Linux x86 64-bi |
| 3 | slobcdb3 | orarac3 | 17-Mar-24 14:31 | 17-Mar-24 14:34 | 18-Mar-24 14:34 | 21.0.0.0.0 | 1,440.33 | 92,198.42 | 24.05 | 64.01 | Linux x86 64-bi |
| 4 | slobcdb4 | orarac4 | 17-Mar-24 14:31 | 17-Mar-24 14:34 | 18-Mar-24 14:34 | 21.0.0.0.0 | 1,440.35 | 92,199.65 | 24.05 | 64.01 | Linux x86 64-bi |
| 5 | slobcdb5 | orarac5 | 17-Mar-24 14:31 | 17-Mar-24 14:34 | 18-Mar-24 14:34 | 21.0.0.0.0 | 1,440.35 | 92,197.63 | 24.05 | 64.01 | Linux x86 64-bi |
| 6 | slobcdb6 | orarac6 | 17-Mar-24 14:31 | 17-Mar-24 14:34 | 18-Mar-24 14:34 | 21.0.0.0.0 | 1,440.35 | 92,198.58 | 24.05 | 64.01 | Linux x86 64-bi |
| 7 | slobcdb7 | orarac7 | 17-Mar-24 14:31 | 17-Mar-24 14:34 | 18-Mar-24 14:34 | 21.0.0.0.0 | 1,440.33 | 92,196.37 | 24.05 | 64.01 | Linux x86 64-bi |
| 8 | slobcdb8 | orarac8 | 17-Mar-24 14:31 | 17-Mar-24 14:34 | 18-Mar-24 14:34 | 21.0.0.0.0 | 1,440.33 | 92,197.55 | 24.05 | 64.01 | Linux x86 64-bi |

Open Pluggable Databases at Begin Snap: 3, End Snap: 3

The following screenshot shows a section from the Oracle AWR report that highlights Physical Reads/Sec and Physical Writes/Sec for each instance while running SLOB workload for sustained 24 hours. It highlights that IO load is distributed across all the cluster nodes performing workload operations.

System Statistics - Per Second DB/Inst: SLOBCDB/slobcdb1 Snaps: 161-185

| I# | Logical Reads/s | Physical Reads/s | Physical Writes/s | Redo Size (k)/s | Block Changes/s | User Calls/s | Execs/s | Parses/s | Logons/s | Txns/s |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 93,417.04 | 81,881.2 | 26,586.0 | 22,239.5 | 53,468.2 | 43.3 | 1,438.7 | 3.9 | 10.49 | 408.7 |
| 2 | 93,440.90 | 81,912.0 | 26,605.1 | 22,256.9 | 53,512.0 | 43.3 | 1,407.4 | 4.0 | 10.47 | 409.0 |
| 3 | 93,514.01 | 81,935.5 | 26,605.8 | 22,265.6 | 53,545.0 | 43.2 | 1,425.0 | 20.8 | 10.44 | 409.1 |
| 4 | 93,415.21 | 81,863.7 | 26,585.8 | 22,244.6 | 53,472.1 | 43.3 | 1,413.1 | 4.0 | 10.42 | 408.7 |
| 5 | 93,439.72 | 81,836.7 | 26,581.6 | 22,240.7 | 53,462.6 | 42.9 | 1,449.9 | 4.1 | 10.38 | 408.6 |
| 6 | 93,581.81 | 81,936.0 | 26,611.9 | 22,264.5 | 53,522.4 | 42.8 | 1,466.7 | 4.3 | 10.34 | 409.1 |
| 7 | 93,526.04 | 81,964.0 | 26,622.2 | 22,274.0 | 53,542.6 | 42.7 | 1,423.0 | 4.3 | 10.32 | 409.3 |
| 8 | 93,495.85 | 81,969.7 | 26,614.6 | 22,266.4 | 53,524.5 | 42.6 | 1,410.8 | 4.1 | 10.29 | 409.1 |
| Sum | 747,830.58 | 655,298.8 | 212,812.8 | 178,052.2 | 428,049.5 | 343.9 | 11,434.6 | 49.5 | 83.14 | 3,271.6 |
| Avg | 93,478.82 | 81,912.3 | 26,601.6 | 22,256.5 | 53,506.2 | 43.0 | 1,429.3 | 6.2 | 10.39 | 409.0 |
| Std | 59.99 | 48.0 | 15.2 | 13.3 | 33.7 | 0.3 | 20.9 | 5.9 | 0.07 | 0.2 |

The following screenshot shows “IO Profile” which was captured from the same 70% Read (30% update) Test scenario while running SLOB test with 512 users which shows 863k IOPS (655k Reads and 208k Writes) for this sustained 24 Hours test.

IO Profile (Global) DB/Inst: SLOBCDB/slobcdb1 Snaps: 161-185

| Statistic | Read+Write/s | Reads/s | Writes/s |
|---|---|---|---|
| Total Requests | 863,612.24 | 655,400.87 | 208,211.37 |
| Database Requests | 856,647.59 | 655,247.21 | 201,400.38 |
| Optimized Requests | 0.00 | 0.00 | 0.00 |
| Redo Requests | 2,967.88 | N/A | 2,967.88 |
| Total (MB) | 7,057.25 | 5,213.25 | 1,844.00 |
| Database (MB) | 6,781.98 | 5,119.42 | 1,662.56 |
| Optimized Total (MB) | 0.00 | 0.00 | 0.00 |
| Redo (MB) | 173.88 | N/A | 173.88 |
| Database (blocks) | 868,093.44 | 655,285.12 | 212,808.32 |
| Via Buffer Cache (blocks) | 868,082.87 | 655,278.66 | 212,804.21 |
| Direct (blocks) | 10.75 | 6.65 | 4.10 |

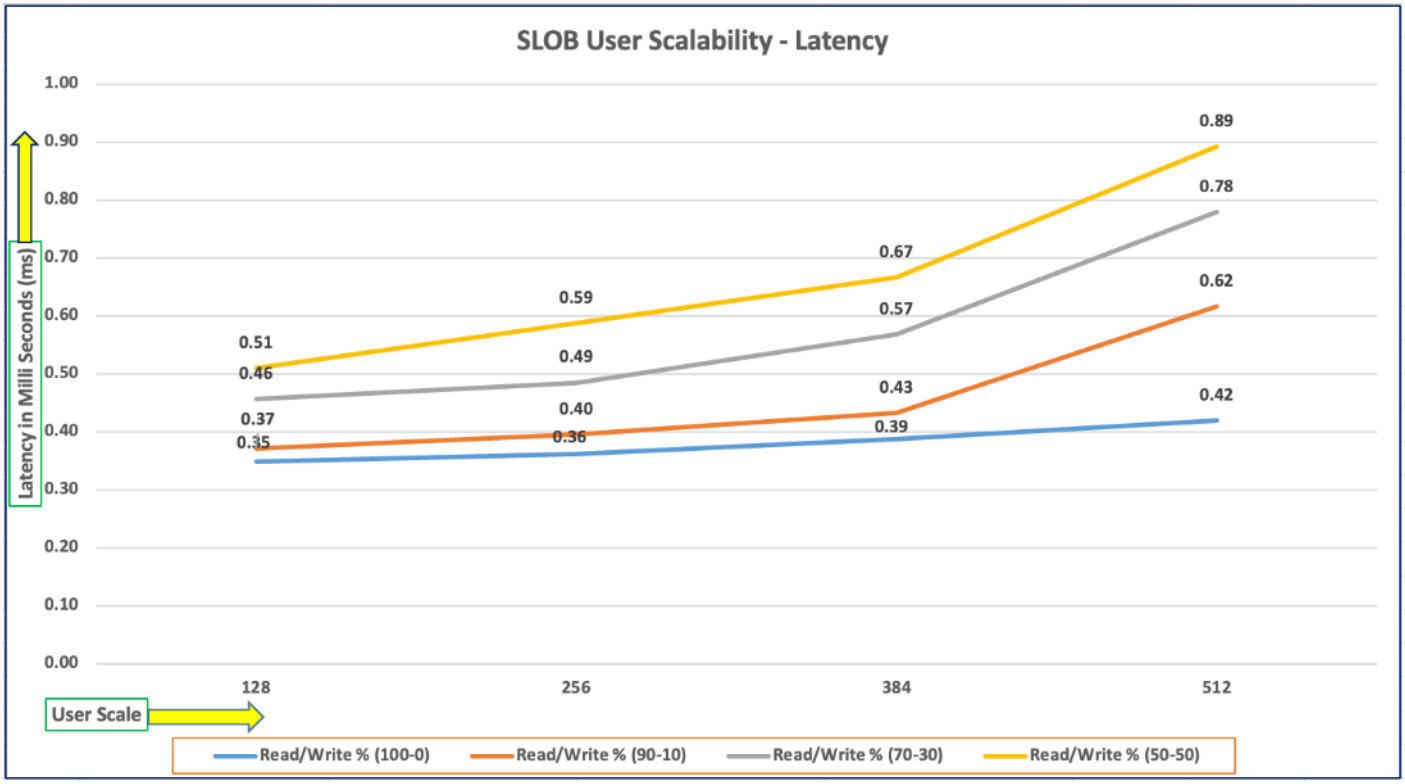
The following screenshot shows “Top Timed Events” and “Wait Time” during this 24-Hour SLOB test while running workload with 512 users.

Top Timed Events DB/Inst: SLOBCDB/slobcdb1 Snaps: 161-185
-> Instance '*' - cluster wide summary
-> '*' Waits, %Timeouts, Wait Time Total(s) : Cluster-wide total for the wait event
-> '*' 'Wait Time Avg' : Cluster-wide average computed as (Wait Time Total / Event Waits)
-> '*' Summary 'Avg Wait Time ' : Per-instance 'Wait Time Avg ' used to compute the following statistics
-> '*' [Avg/Min/Maximum/Standard Deviation of per-instance 'Wait Time Avg'
-> '*' Cnt : count of instances with wait times for the event

| | Wait | | Event | | Wait Time | | | Summary Avg Wait Time | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| I# | Class | Event | Waits | %Timeouts | Total(s) | Avg Wait | %DB time | Avg | Min | Max | Std Dev | Cnt |
| * | User I/O | db file sequential read | 5.654935E+10 | 0.0 | 4.3315019E+07 | 765.97us | 97.88 | 765.97us | 764.90us | 766.69us | 576.70ns | 8 |
| | | DB CPU | N/A | N/A | 1,100,549.33 | | 2.49 | | | | | 8 |
| | Other | GV$: slave acquisition retry wait time | 651,383 | 100.0 | 664,595.37 | 1020.28m | 1.50 | 1020.28ms | 1019.84ms | 1020.60ms | 236.84us | 8 |
| | System I/O | db file parallel write | 2.256983E+09 | 0.0 | 432,015.77 | 191.41us | 0.98 | 191.42us | 189.79us | 192.58us | 1.05us | 8 |
| | System I/O | log file parallel write | 513,778,417 | 0.0 | 395,137.34 | 769.08us | 0.89 | 769.14us | 751.69us | 783.42us | 11.48us | 8 |
| | System I/O | db file async I/O submit | 226,342,727 | 0.0 | 150,954.32 | 666.93us | 0.34 | 666.92us | 660.10us | 671.69us | 3.65us | 8 |
| | Cluster | gc cr grant 2-way | 1.218704E+09 | 0.0 | 101,534.11 | 83.31us | 0.23 | 83.35us | 82.14us | 85.08us | 1.01us | 8 |
| | Cluster | gc current grant 2-way | 296,481,530 | 0.0 | 31,475.30 | 106.16us | 0.07 | 106.43us | 100.89us | 113.03us | 3.80us | 8 |
| | Other | LGWR worker group ordering | 20,754,480 | 0.0 | 22,909.38 | 1.10ms | 0.05 | 1.10ms | 1.08ms | 1.13ms | 17.45us | 8 |
| | Other | LGWR any worker group | 22,092,081 | 0.0 | 20,207.75 | 914.71us | 0.05 | 914.92us | 904.72us | 921.11us | 5.91us | 8 |

The following screenshot was captured from Pure Storage GUI during this 24 Hour SLOB test while running workload.


The following graph illustrates the latency exhibited by the Pure Storage FA//XL170 across different workloads (100% Read/Write, 90% Read/Write, 70% Read/Write and 50% Read/Write). All the workloads experienced less than 1 millisecond latency and it varies based on the workloads. As expected, the 50% read (50% update) test exhibited higher latencies as the user counts increases.




## SwingBench Test

SwingBench is a simple to use, free, Java-based tool to generate various types of database workloads and perform stress testing using different benchmarks in Oracle database environments. SwingBench can be used to demonstrate and test technologies such as Real Application Clusters, Online table rebuilds, Standby databases, online backup, and recovery, and so on. In this solution, we used SwingBench tool for running various type of workload and check the overall performance of this reference architecture.

SwingBench provides four separate benchmarks, namely, Order Entry, Sales History, Calling Circle, and Stress Test. For the tests described in this solution, SwingBench Order Entry (SOE) benchmark was used for representing OLTP type of workload and the Sales History (SH) benchmark was used for representing DSS type of workload.

The Order Entry benchmark is based on SOE schema and is TPC-C like by types of transactions. The workload uses a very balanced read/write ratio around 60/40 and can be designed to run continuously and test the performance of a typical Order Entry workload against a small set of tables, producing contention for database resources.

The Sales History benchmark is based on the SH schema and is like TPC-H. The workload is query (read) centric and is designed to test the performance of queries against large tables.

The first step after the databases creation is calibration; about the number of concurrent users, nodes, throughput, IOPS and latency for database optimization. For this solution, we ran the SwingBench workloads on various combination of databases and captured the system performance as follows:

Typically encountered in the real-world deployments, we tested a combination of scalability and stress related scenarios that ran across all the 8-node Oracle RAC cluster, as follows:

- OLTP database user scalability workload representing small and random transactions.
- DSS database workload representing larger transactions.
- Mixed databases (OLTP and DSS) workloads running simultaneously.

For this SwingBench workload tests, we created three Container Database as SOECDB, ENGCDB and SHCDB. We configured the first container database as “SOECDB” and created one pluggable database as “SOEPDB” and second container database “ENGCDB” with one pluggable database as “ENGPDB” to run the SwingBench SOE workload representing OLTP type of workload characteristics. We configured the container databases “SHCDB” and created one pluggable database as “SHPDB” to run the SwingBench SH workload representing DSS type of workload characteristics.

For this solution, we deployed and validated multiple container databases as well as pluggable databases and run various SwingBench SOE and SH workloads to demonstrate the multitenancy capability, performance, and sustainability for this reference architecture.

For the OLTP databases, we created and configured SOE schema of 3.5 TB for the SOEPDB Database and 3 TB for the ENGPDB Database. For the DSS database, we created and configured SH schema of 4 TB for the SHPDB Database:

- One OLTP Database Performance
- Multiple (Two) OLTP Databases Performance
- One DSS Database Performance
- Multiple OLTP & DSS Databases Performance

## One OLTP Database Performance

For one OLTP database workload featuring Order Entry schema, we created one container database SOECDB and one pluggable database SOEPDB as explained earlier. We used 64 GB size of SGA for this database and, we ensured that the HugePages were in use. We ran the SwingBench SOE workload with varying the total number of users on this database from 200 Users to 800 Users. Each user scale iteration test was run for at least 3 hours and for each test scenario, we captured the Oracle AWR reports to check the overall system performance below:

### User Scalability

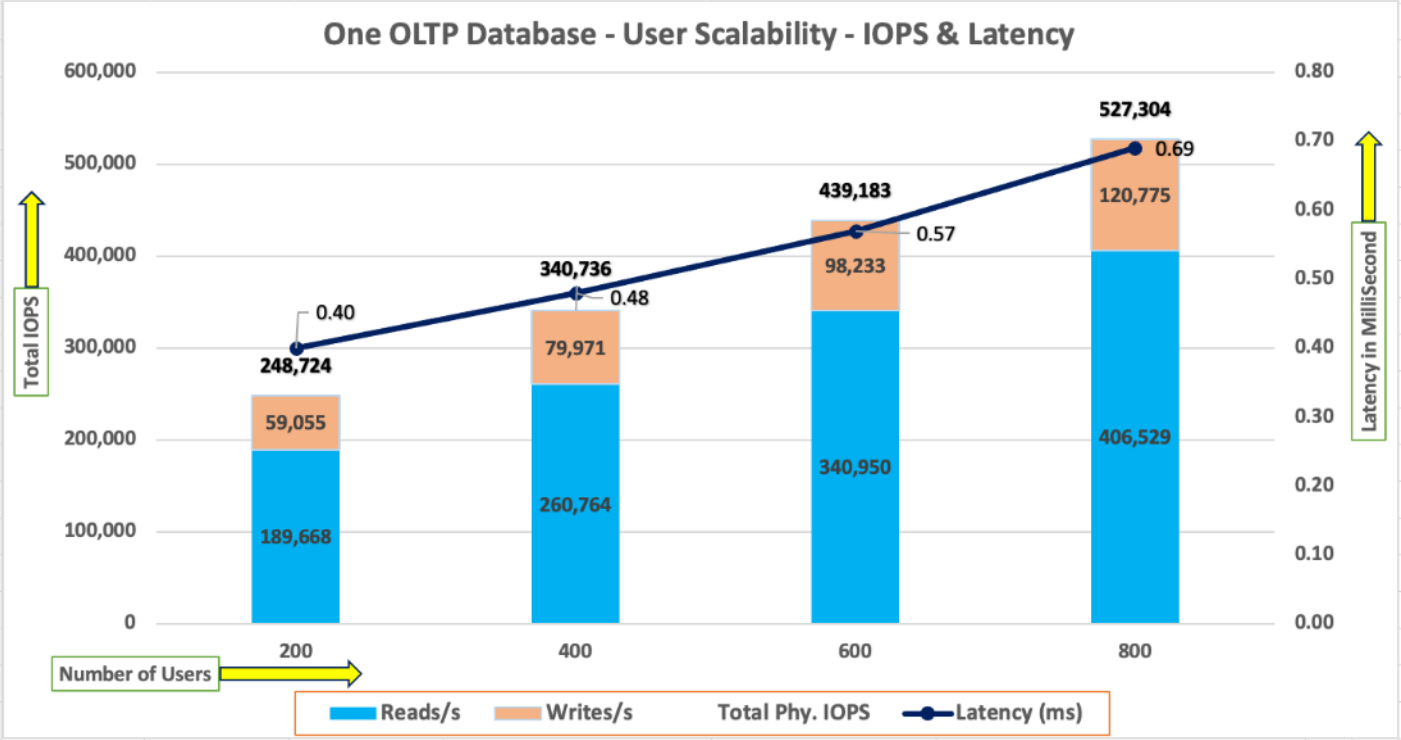
Table 15 lists the Transaction Per Minutes (TPM), IOPS, Latency and System Utilization for the SOECDB Database while running the workload from 200 users to 800 users across all the eight RAC nodes.

**Table 15. User Scale Test on One OLTP Database**

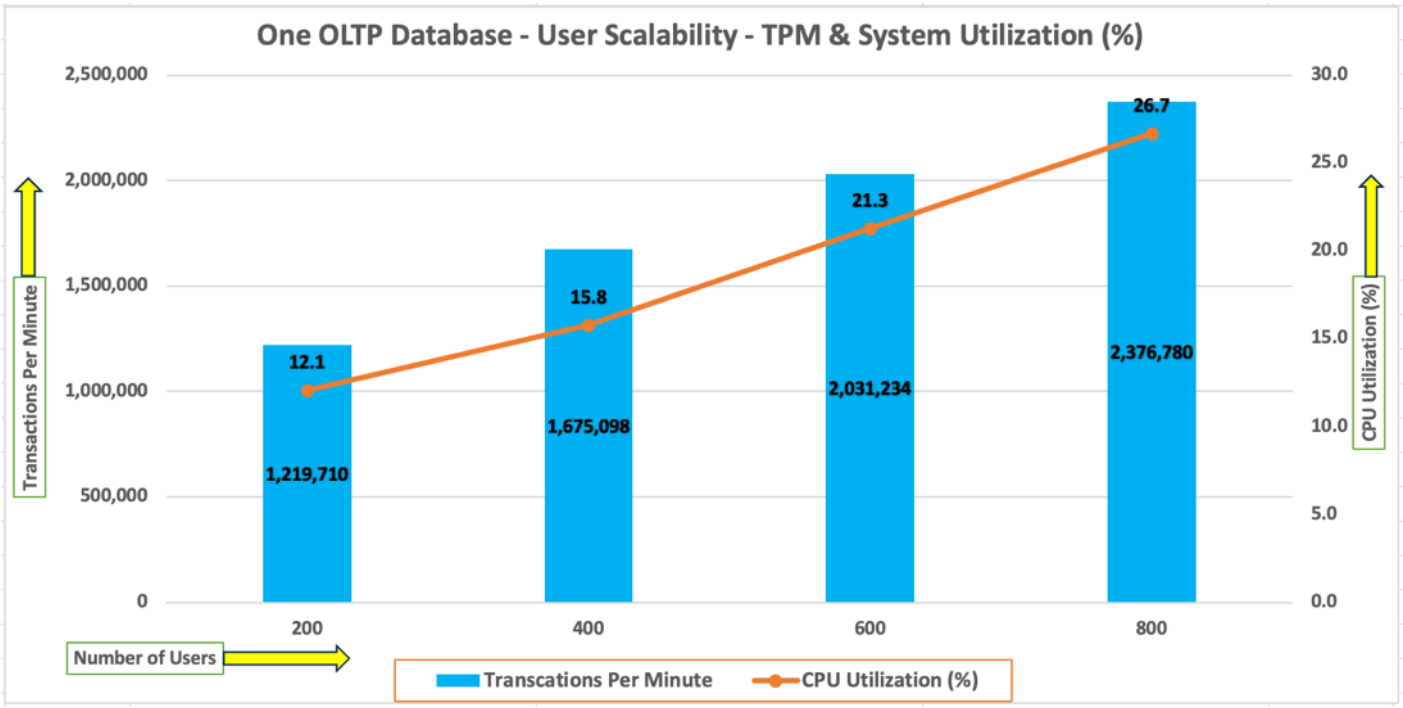
| Number of Users | Transactions | | Storage IOPS | | | Latency (milliseconds) | CPU Utilization (%) |
|---|---|---|---|---|---|---|---|
| | Per Seconds (TPS) | Per Minutes (TPM) | Reads/Sec | Writes/Sec | Total IOPS | | |
| 200 | 20,329 | 1,219,710 | 189,668 | 59,055 | 248,724 | 0.40 | 12.1 |
| 400 | 27,918 | 1,675,098 | 260,764 | 79,971 | 340,736 | 0.48 | 15.8 |

| Number of | Transactions | | Storage IOPS | | | Latency (milliseconds) | CPU Utilization |
|---|---|---|---|---|---|---|---|
| 600 | 33,854 | 2,031,234 | 340,950 | 98,233 | 439,183 | 0.57 | 21.3 |
| 800 | 39,613 | 2,376,780 | 406,529 | 120,775 | 527,304 | 0.69 | 26.7 |

The following chart shows the IOPS and Latency for the SOECDB Database while running the SwingBench Order Entry workload tests from 200 users to 800 users across all eight RAC nodes.




The chart below shows the Transaction Per Minutes (TPM) and System Utilization for the SOECDB Database while running the same SwingBench Order Entry workload tests from 200 users to 800 users:




The AWR screenshot below was captured from one of the above test scenarios with 800 users running SwingBench Order Entry workload for sustained 24 hours across all eight RAC nodes:

```
Database Summary
~~~~~~~~~~~~~~~~
                                  Database                                    Snapshot Ids     Number of Instances     Number of Hosts     Report Total (minutes)
---------------------------------------------------------------------------- ---------------- ------------------- ------------------- ------------------------
        Id Name      Unique Name Role            Edition RAC CDB Block Size      Begin      End   In Report    Total   In Report    Total       DB time Elapsed time
---------- --------- ----------- --------------- ------- --- --- ----------   -------- --------   --------- --------   --------- --------   ---------- ------------
1666996520 SOECDB    soecdb      PRIMARY         EE      YES YES       8192        342      366          8        8          8        8    566,828.31     1,440.25
Database Instances Included In Report
-> Listed in order of instance number, I#
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

                                                                                                                                                Avg Active
  I# Instance  Host       Startup         Begin Snap Time End Snap Time   Release         Elapsed Time(min) DB time(min) Up Time(hrs)   Sessions Platform
---- --------- ---------- --------------- --------------- --------------- --------------- ----------------- ------------ ------------ ---------- ---------------
   1 soecdb1   orarac1    10-Apr-24 14:55 13-Apr-24 15:42 14-Apr-24 15:42 21.0.0.0.0             1,440.15    69,571.25        96.78      48.31 Linux x86 64-bi
   2 soecdb2   orarac2    10-Apr-24 14:55 13-Apr-24 15:42 14-Apr-24 15:42 21.0.0.0.0             1,440.17    71,520.58        96.78      49.66 Linux x86 64-bi
   3 soecdb3   orarac3    10-Apr-24 14:55 13-Apr-24 15:42 14-Apr-24 15:42 21.0.0.0.0             1,440.17    72,033.69        96.78      50.02 Linux x86 64-bi
   4 soecdb4   orarac4    10-Apr-24 14:55 13-Apr-24 15:42 14-Apr-24 15:42 21.0.0.0.0             1,440.17    70,629.84        96.78      49.04 Linux x86 64-bi
   5 soecdb5   orarac5    10-Apr-24 14:55 13-Apr-24 15:42 14-Apr-24 15:42 21.0.0.0.0             1,440.17    70,658.88        96.78      49.06 Linux x86 64-bi
   6 soecdb6   orarac6    10-Apr-24 14:55 13-Apr-24 15:42 14-Apr-24 15:42 21.0.0.0.0             1,440.15    70,646.05        96.78      49.05 Linux x86 64-bi
   7 soecdb7   orarac7    10-Apr-24 14:55 13-Apr-24 15:42 14-Apr-24 15:42 21.0.0.0.0             1,440.15    71,131.32        96.78      49.39 Linux x86 64-bi
   8 soecdb8   orarac8    10-Apr-24 14:55 13-Apr-24 15:42 14-Apr-24 15:42 21.0.0.0.0             1,440.17    70,636.70        96.78      49.05 Linux x86 64-bi
Open Pluggable Databases at Begin Snap: 3, End Snap: 3
~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

The following screenshot captured from the Oracle AWR report highlights the Physical Reads/Sec, Physical Writes/Sec and Transactions per Seconds for the Container SOECDB Database for the same above test. We captured about 528k IOPS (412k Reads/s and 116k Writes/s) with the 38k TPS (Transactions Per Seconds) while running this 24-hour sustained SwingBench Order Entry workload on one OLTP database with 800 users.

System Statistics - Per Second DB/Inst: SOECDB/soecdb1 Snaps: 342-366

| I# | Logical Reads/s | Physical Reads/s | Physical Writes/s | Redo Size (k)/s | Block Changes/s | User Calls/s | Execs/s | Parses/s | Logons/s | Txns/s |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1,013,583.95 | 48,580.4 | 13,917.2 | 13,261.0 | 78,787.4 | 21,135.2 | 80,106.6 | 7,723.5 | 0.67 | 4,618.7 |
| 2 | 1,080,533.56 | 54,013.3 | 14,696.1 | 13,840.2 | 82,434.1 | 22,089.3 | 83,711.6 | 8,071.6 | 0.67 | 4,826.4 |
| 3 | 996,194.68 | 52,567.9 | 15,188.2 | 14,333.3 | 85,341.0 | 22,871.8 | 86,681.3 | 8,357.8 | 0.67 | 4,998.4 |
| 4 | 1,066,366.10 | 51,231.5 | 14,808.1 | 13,987.4 | 83,230.2 | 22,290.1 | 84,473.8 | 8,145.0 | 0.67 | 4,871.2 |
| 5 | 1,147,468.12 | 50,244.7 | 14,284.1 | 13,707.0 | 81,263.4 | 21,854.9 | 82,834.2 | 7,986.3 | 0.67 | 4,776.4 |
| 6 | 1,097,500.31 | 50,669.8 | 14,603.8 | 13,832.0 | 82,255.5 | 22,041.6 | 83,532.1 | 8,054.4 | 0.67 | 4,817.3 |
| 7 | 1,276,983.25 | 53,586.4 | 14,372.9 | 13,408.5 | 79,793.4 | 21,354.5 | 80,967.6 | 7,807.2 | 0.67 | 4,665.7 |
| 8 | 1,007,551.49 | 50,988.2 | 14,729.0 | 13,919.7 | 82,823.7 | 22,186.7 | 84,079.3 | 8,107.5 | 0.67 | 4,848.7 |
| Sum | 8,686,181.48 | 411,882.2 | 116,599.5 | 110,289.1 | 655,928.6 | 175,824.2 | 666,386.6 | 64,253.2 | 5.36 | 38,422.9 |
| Avg | 1,085,772.68 | 51,485.3 | 14,574.9 | 13,786.1 | 81,991.1 | 21,978.0 | 83,298.3 | 8,031.7 | 0.67 | 4,802.9 |
| Std | 92,785.00 | 1,809.9 | 383.2 | 335.4 | 2,047.3 | 543.9 | 2,052.8 | 198.0 | 0.00 | 119.0 |

The following screenshot captured from the Oracle AWR report shows the SOECDB database "IO Profile" for the "Reads/s" and "Writes/s" requests for the entire 24-hour of the test. The Total Requests (Read and Write Per Second) were around "552k" with Total (MB) "Read+Write" Per Second was around "4316" MB/s for the SOECDB database while running the SwingBench Order Entry workload test on one OLTP database.

IO Profile (Global) DB/Inst: SOECDB/soecdb1 Snaps: 342-366

| Statistic | Read+Write/s | Reads/s | Writes/s |
|---|---|---|---|
| Total Requests | 552,808.24 | 407,363.61 | 145,444.64 |
| Database Requests | 509,901.26 | 407,234.47 | 102,666.79 |
| Optimized Requests | 0.00 | 0.00 | 0.00 |
| Redo Requests | 23,936.71 | N/A | 23,936.71 |
| Total (MB) | 4,316.48 | 3,283.95 | 1,032.54 |
| Database (MB) | 4,128.50 | 3,217.62 | 910.87 |
| Optimized Total (MB) | 0.00 | 0.00 | 0.00 |
| Redo (MB) | 107.70 | N/A | 107.70 |
| Database (blocks) | 528,447.49 | 411,855.52 | 116,591.97 |
| Via Buffer Cache (blocks) | 523,794.18 | 407,215.13 | 116,579.05 |
| Direct (blocks) | 4,653.27 | 4,640.35 | 12.92 |

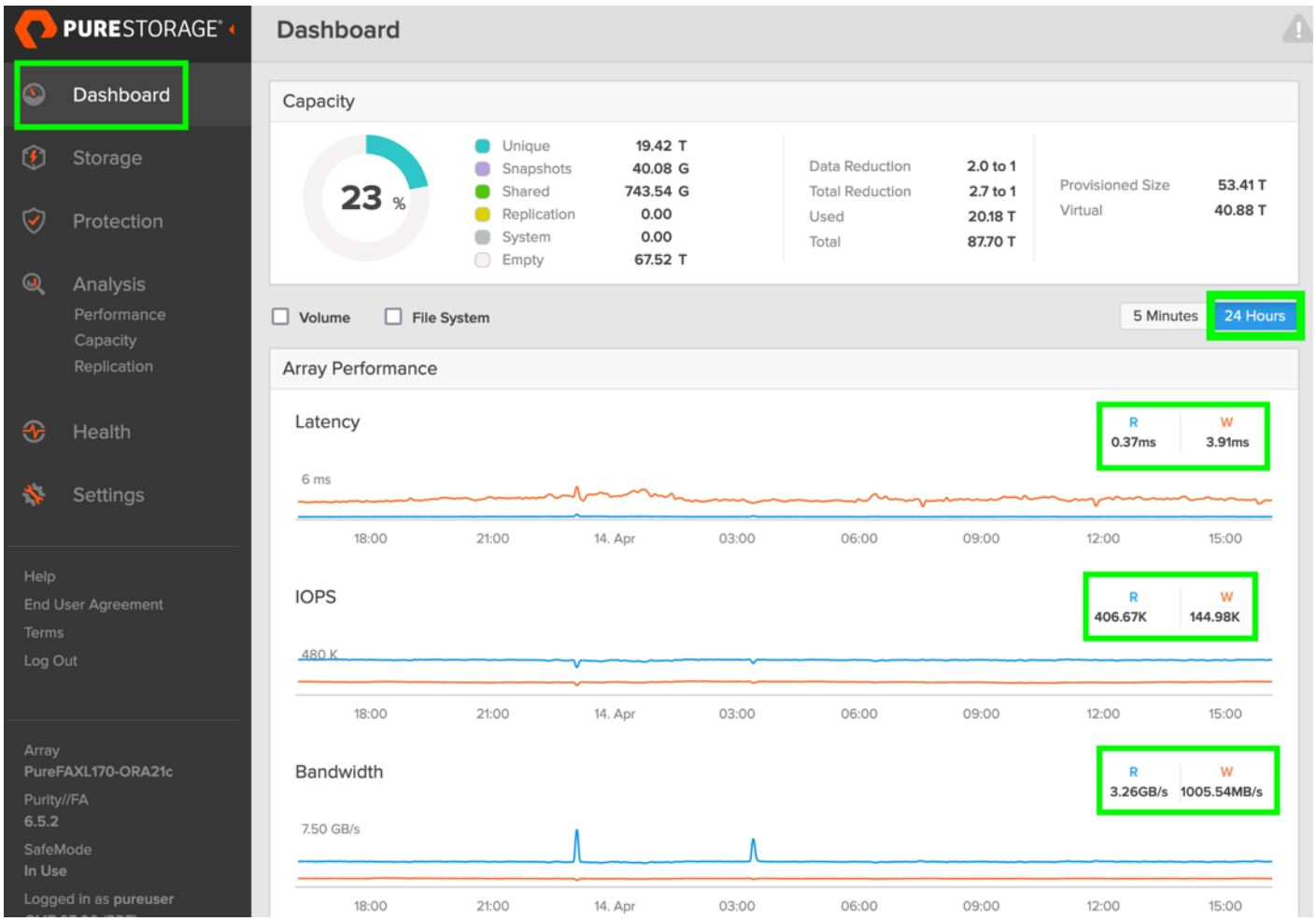
The following screenshot captured from the Oracle AWR report shows the "Top Timed Events" and average wait time for the SOECDB database for the entire duration of the 24-hour test running with 800 users.

Top Timed Events DB/Inst: SOECDB/soecdb1 Snaps: 342-366

-> Instance '*' - cluster wide summary
-> '*' Waits, %Timeouts, Wait Time Total(s) : Cluster-wide total for the wait event
-> '*' 'Wait Time Avg' : Cluster-wide average computed as (Wait Time Total / Event Waits)
-> '*' Summary 'Avg Wait Time ' : Per-instance 'Wait Time Avg ' used to compute the following statistics
-> '*' [Avg/Min/Max/Std Dev] : average/minimum/maximum/standard deviation of per-instance 'Wait Time Avg'
-> '*' Cnt : count of instances with wait times for the event

| | Wait | | Event | | Wait Time | | | Summary Avg Wait Time | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| I# | Class | Event | Waits | %Timeouts | Total(s) | Avg Wait | %DB time | Avg | Min | Max | Std Dev | Cnt |
| * | User I/O | db file sequential read | 2.821722E+10 | 0.0 | 1.4172606E+07 | 502.27us | 41.67 | 502.31us | 500.58us | 507.52us | 2.21us | 8 |
| | | DB CPU | N/A | N/A | 8,442,043.37 | | 24.82 | | | | | 8 |
| | Cluster | gc cr grant 2-way | 1.523692E+10 | 0.0 | 2,598,062.64 | 170.51us | 7.64 | 170.55us | 162.45us | 172.56us | 3.40us | 8 |
| | Commit | log file sync | 3.314268E+09 | 0.0 | 2,505,732.12 | 756.04us | 7.37 | 756.00us | 744.37us | 761.83us | 6.11us | 8 |
| | Cluster | gc current block 3-way | 7.664151E+09 | 0.0 | 1,928,758.97 | 251.66us | 5.67 | 251.59us | 243.27us | 255.81us | 3.79us | 8 |
| | Cluster | gc current grant 2-way | 9.342035E+09 | 0.0 | 1,494,311.17 | 159.96us | 4.39 | 160.00us | 157.47us | 170.89us | 4.43us | 8 |
| | Cluster | gc cr block 3-way | 4.339339E+09 | 0.0 | 1,111,474.72 | 256.14us | 3.27 | 256.05us | 247.48us | 260.51us | 3.93us | 8 |
| | Cluster | gc current block 2-way | 6.193674E+09 | 0.0 | 938,627.14 | 151.55us | 2.76 | 151.74us | 142.69us | 160.03us | 5.21us | 8 |
| | User I/O | db file parallel read | 3.095676E+09 | 0.0 | 851,807.37 | 275.16us | 2.50 | 275.17us | 274.67us | 276.88us | 712.13ns | 8 |
| | System I/O | log file parallel write | 4.138863E+09 | 0.0 | 721,970.19 | 174.44us | 2.12 | 174.44us | 173.56us | 174.90us | 474.38ns | 8 |

The following screenshot shows the Pure Storage array dashboard when one OLTP database was running the workload for the sustained 24-hour SwingBench SOE workload with 800 users. The screenshot shows the average IOPS “550k” with the average throughput of “4200 MB/s” with the average storage latency around “1.7 millisecond.”




The storage cluster utilization during the above test was average around 45% which was an indication that storage hasn’t reached the threshold and could take more load by adding multiple databases.

Also, for the entire 24-hour test, we observed the system performance (IOPS and Throughput) was consistent throughout and we did not observe any dips in performance while running one OLTP database stress test.

## Multiple (Two) OLTP Databases Performance

For running multiple OLTP database workload, we have created two container database SOECDB and ENGCDB. For each container database, one pluggable database was configured as SOEPDB and ENGPDB as explained earlier. We ran the SwingBench SOE workload on both the databases at the same time with varying the total number of users on both the databases from 200 Users to 1000 Users. Each user scale iteration test was run for at least 3 hours and for each test scenario, we captured the Oracle AWR reports to check the overall system performance.

Table 16 lists the IOPS and System Utilization for each of the pluggable databases while running the workload from total of 200 users to 1000 users across all the eight RAC nodes.

**Table 16. IOPS and System Utilization for Pluggable Databases**

| Users | IOPS for SOECDB | IOSP for ENGCDB | Total IOPS | System Utilization (%) |
|---|---|---|---|---|
| 200 | 168,946 | 153,727 | 322,673 | 19.7 |
| 400 | 231,312 | 211,475 | 442,786 | 24.6 |
| 600 | 281,617 | 258,175 | 539,792 | 29.1 |
| 800 | 306,521 | 279,627 | 586,149 | 33.4 |
| 1000 | 345,470 | 307,845 | 653,315 | 36.8 |

The following chart shows the IOPS and System Utilization for both the container databases while running the SwingBench SOE workload on them at the same time. We observed both databases were linearly scaling the IOPS after increasing and scaling more users. We observed average 653k IOPS with overall system utilization around 37% when scaling maximum number of users on multiple OLTP database workload test. After increasing users beyond certain level, we observed more GC cluster events and overall similar IOPS around 650k.

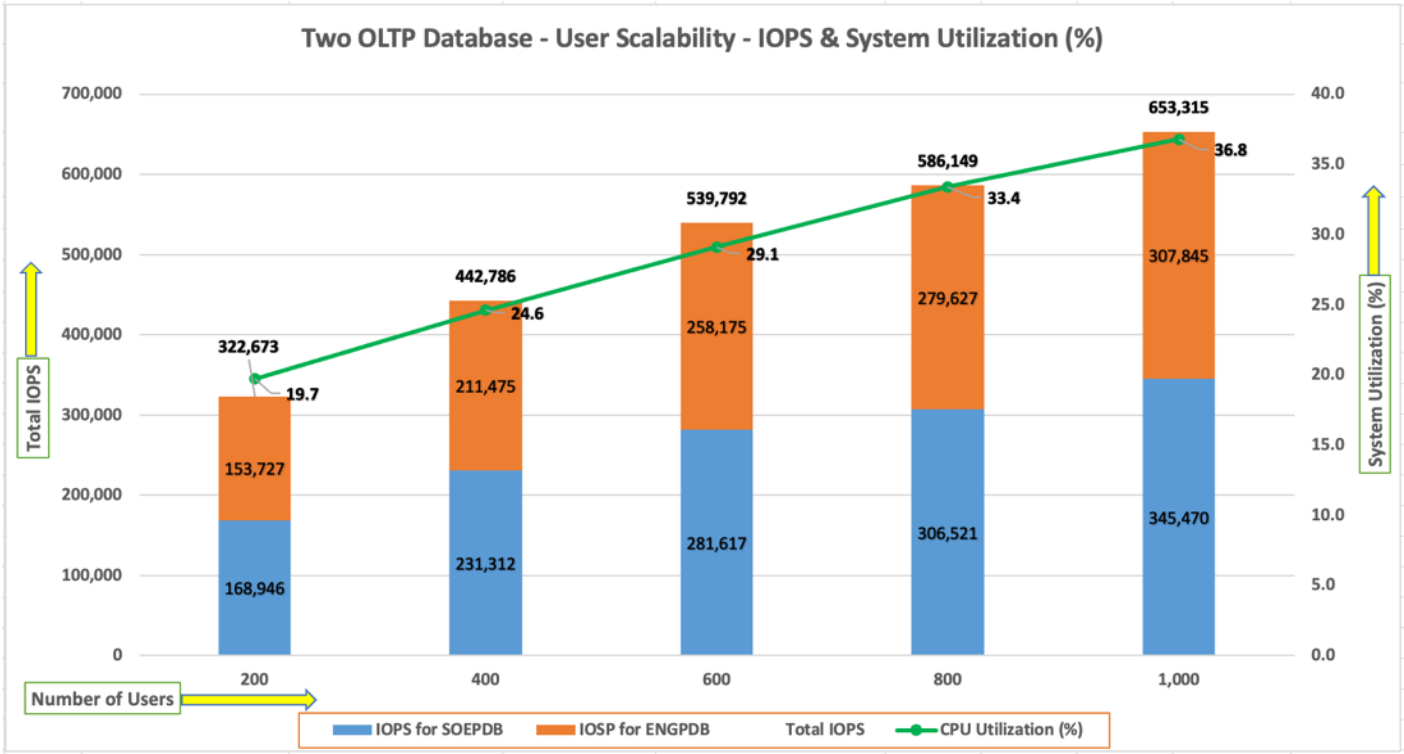
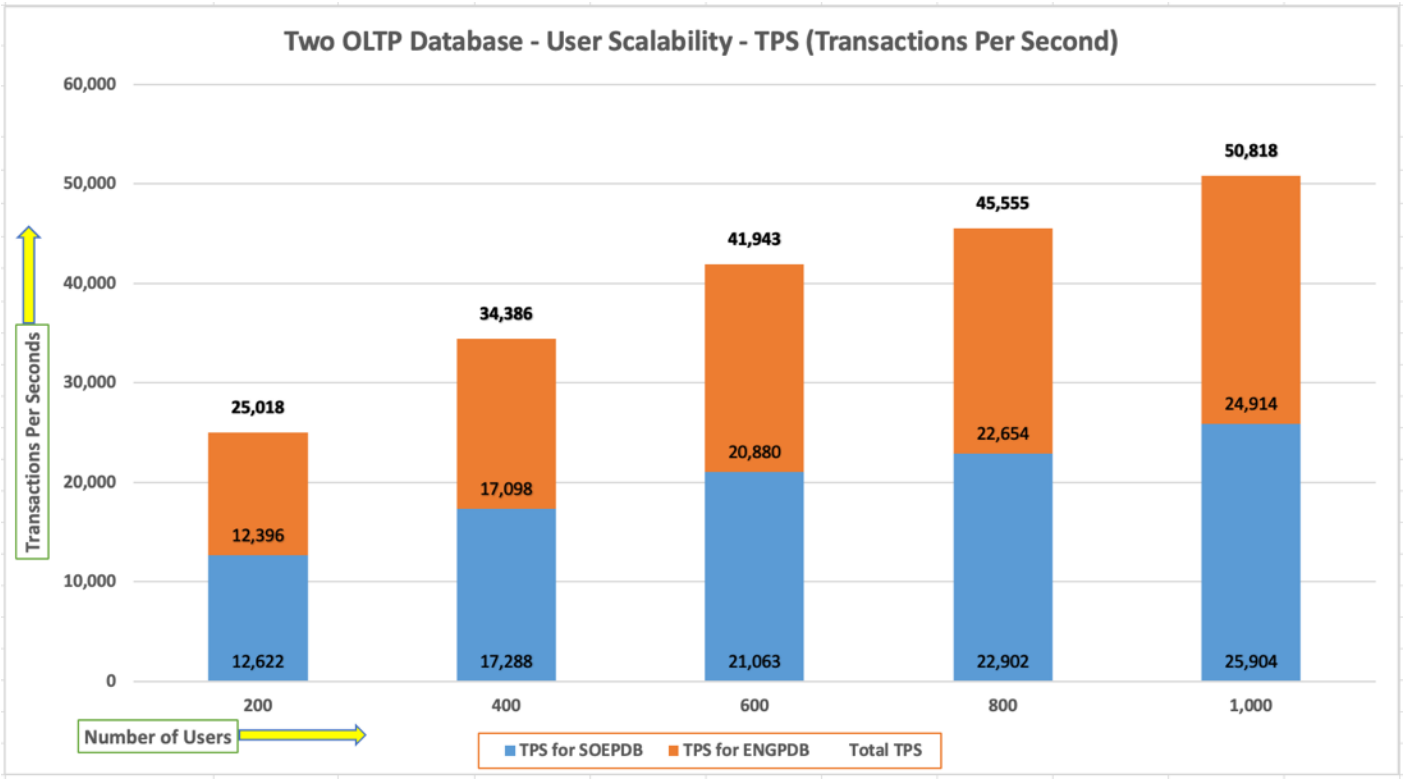



Table 17 lists the Transactions per Seconds (TPS) and Transactions per Minutes (TPM) for each of the pluggable databases while running the workload from total of 200 users to 1000 users across all the eight RAC nodes.

Table 17. Transactions per Seconds and Transactions per Minutes

| Users | TPS for SOECDB | TPS for ENGCDB | Total TPS | Total TPM |
|---|---|---|---|---|
| 200 | 12,622 | 12,396 | 25,018 | 1,501,080 |
| 400 | 17,288 | 17,098 | 34,386 | 2,063,178 |
| 600 | 21,063 | 20,880 | 41,943 | 2,516,580 |
| 800 | 22,902 | 22,654 | 45,555 | 2,733,318 |
| 1000 | 25,904 | 24,914 | 50,818 | 3,049,086 |

The following chart shows the Transactions per Seconds (TPS) for the same tests (above) on CDBDB Database for running the workload on both pluggable databases.




The following screenshot showcases the test start time for the first SOECDB database with 500 users running SwingBench Order Entry workload for sustained 24 hours across all eight nodes as:

Database Summary

| Database | | | | | | | Snapshot Ids | | Number of Instances | | Number of Hosts | | Report Total (minutes) | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Id | Name | Unique Name | Role | Edition | RAC | CDB | Block Size | Begin | End | In Report | Total | In Report | Total | DB time | Elapsed time |
| 1666996520 | SOECDB | soecdb | PRIMARY | EE | YES | YES | 8192 | 566 | 590 | 8 | 8 | 8 | 8 | 621,639.21 | 1,440.94 |

Database Instances Included In Report

-> Listed in order of instance number, I#

| I# | Instance | Host | Startup | Begin Snap Time | End Snap Time | Release | Elapsed Time(min) | DB time(min) | Up Time(hrs) | Avg Active Sessions | Platform |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | soecdb1 | orarac1 | 26-Apr-24 22:17 | 27-Apr-24 00:32 | 28-Apr-24 00:33 | 21.0.0.0.0 | 1,440.93 | 77,862.21 | 26.26 | 54.04 | Linux x86 64-bi |
| 2 | soecdb2 | orarac2 | 26-Apr-24 22:17 | 27-Apr-24 00:32 | 28-Apr-24 00:33 | 21.0.0.0.0 | 1,440.93 | 77,488.03 | 26.26 | 53.78 | Linux x86 64-bi |
| 3 | soecdb3 | orarac3 | 26-Apr-24 22:17 | 27-Apr-24 00:32 | 28-Apr-24 00:33 | 21.0.0.0.0 | 1,440.93 | 77,472.90 | 26.26 | 53.77 | Linux x86 64-bi |
| 4 | soecdb4 | orarac4 | 26-Apr-24 22:17 | 27-Apr-24 00:32 | 28-Apr-24 00:33 | 21.0.0.0.0 | 1,440.93 | 77,934.62 | 26.26 | 54.09 | Linux x86 64-bi |
| 5 | soecdb5 | orarac5 | 26-Apr-24 22:17 | 27-Apr-24 00:32 | 28-Apr-24 00:33 | 21.0.0.0.0 | 1,440.95 | 77,955.49 | 26.26 | 54.10 | Linux x86 64-bi |
| 6 | soecdb6 | orarac6 | 26-Apr-24 22:17 | 27-Apr-24 00:32 | 28-Apr-24 00:33 | 21.0.0.0.0 | 1,440.93 | 77,422.76 | 26.26 | 53.73 | Linux x86 64-bi |
| 7 | soecdb7 | orarac7 | 26-Apr-24 22:17 | 27-Apr-24 00:32 | 28-Apr-24 00:33 | 21.0.0.0.0 | 1,440.92 | 78,028.77 | 26.26 | 54.15 | Linux x86 64-bi |
| 8 | soecdb8 | orarac8 | 26-Apr-24 22:17 | 27-Apr-24 00:32 | 28-Apr-24 00:33 | 21.0.0.0.0 | 1,440.93 | 77,474.43 | 26.26 | 53.77 | Linux x86 64-bi |

Open Pluggable Databases at Begin Snap: 3, End Snap: 3

The following screenshot showcases the test start time for the second ENGCDB database with 500 users running SwingBench Order Entry workload for sustained 24 hours across all eight nodes at the same time as:

Database Summary

| Database | | | | | | | Snapshot Ids | | Number of Instances | | Number of Hosts | | Report Total (minutes) | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Id | Name | Unique Name | Role | Edition | RAC | CDB | Block Size | Begin | End | In Report | Total | In Report | Total | DB time | Elapsed time |
| 2518202654 | ENGCDB | engcdb | PRIMARY | EE | YES | YES | 8192 | 361 | 385 | 8 | 8 | 8 | 8 | 567,260.68 | 1,440.97 |

Database Instances Included In Report

-> Listed in order of instance number, I#

| I# | Instance | Host | Startup | Begin Snap Time | End Snap Time | Release | Elapsed Time(min) | DB time(min) | Up Time(hrs) | Avg Active Sessions | Platform |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | engcdb1 | orarac1 | 26-Apr-24 22:18 | 27-Apr-24 00:32 | 28-Apr-24 00:33 | 21.0.0.0.0 | 1,440.97 | 71,406.47 | 26.25 | 49.55 | Linux x86 64-bi |
| 2 | engcdb2 | orarac2 | 26-Apr-24 22:18 | 27-Apr-24 00:32 | 28-Apr-24 00:33 | 21.0.0.0.0 | 1,440.95 | 70,688.63 | 26.25 | 49.06 | Linux x86 64-bi |
| 3 | engcdb3 | orarac3 | 26-Apr-24 22:18 | 27-Apr-24 00:32 | 28-Apr-24 00:33 | 21.0.0.0.0 | 1,440.95 | 70,688.66 | 26.25 | 49.06 | Linux x86 64-bi |
| 4 | engcdb4 | orarac4 | 26-Apr-24 22:18 | 27-Apr-24 00:32 | 28-Apr-24 00:33 | 21.0.0.0.0 | 1,440.95 | 69,693.31 | 26.25 | 48.37 | Linux x86 64-bi |
| 5 | engcdb5 | orarac5 | 26-Apr-24 22:18 | 27-Apr-24 00:32 | 28-Apr-24 00:33 | 21.0.0.0.0 | 1,440.97 | 70,696.75 | 26.25 | 49.06 | Linux x86 64-bi |
| 6 | engcdb6 | orarac6 | 26-Apr-24 22:18 | 27-Apr-24 00:32 | 28-Apr-24 00:33 | 21.0.0.0.0 | 1,440.95 | 70,717.94 | 26.25 | 49.08 | Linux x86 64-bi |
| 7 | engcdb7 | orarac7 | 26-Apr-24 22:18 | 27-Apr-24 00:32 | 28-Apr-24 00:33 | 21.0.0.0.0 | 1,440.95 | 71,281.58 | 26.25 | 49.47 | Linux x86 64-bi |
| 8 | engcdb8 | orarac8 | 26-Apr-24 22:18 | 27-Apr-24 00:32 | 28-Apr-24 00:33 | 21.0.0.0.0 | 1,440.95 | 72,087.33 | 26.25 | 50.03 | Linux x86 64-bi |

Open Pluggable Databases at Begin Snap: 3, End Snap: 3

The following screenshot was captured from the Oracle AWR report, shows the “Physical Reads/Sec”, “Physical Writes/Sec” and “Transactions per Seconds” for the first Container Database SOECDB while running 500 users SOE workload for sustained 24-hour test. We captured about 312k IOPS (237k Reads/s and 75k Writes/s) with the 24k TPS (1,440,960 TPM) while running this workload test on two OLTP databases at the same time during this entire 24 hours sustained test.

System Statistics - Per Second DB/Inst: SOECDB/soecdb1 Snaps: 566-590

| I# | Logical Reads/s | Physical Reads/s | Physical Writes/s | Redo Size (k)/s | Block Changes/s | User Calls/s | Execs/s | Parses/s | Logons/s | Txns/s |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 725,743.05 | 28,160.3 | 8,678.3 | 8,715.8 | 54,162.2 | 14,580.9 | 55,441.8 | 5,294.5 | 0.70 | 2,791.7 |
| 2 | 810,068.86 | 30,255.9 | 9,420.8 | 9,341.8 | 58,130.4 | 15,656.6 | 59,529.3 | 5,684.1 | 0.69 | 2,998.0 |
| 3 | 879,336.88 | 29,421.5 | 9,526.5 | 9,312.1 | 57,968.5 | 15,609.0 | 59,358.9 | 5,668.9 | 0.69 | 2,988.3 |
| 4 | 818,098.58 | 29,483.3 | 9,573.8 | 9,528.9 | 59,316.8 | 15,976.1 | 60,753.5 | 5,773.7 | 0.69 | 3,058.9 |
| 5 | 885,919.32 | 31,850.4 | 9,392.6 | 9,401.6 | 58,513.5 | 15,759.6 | 59,926.5 | 5,722.8 | 0.70 | 3,017.9 |
| 6 | 783,996.17 | 28,518.2 | 9,518.9 | 9,534.1 | 59,359.5 | 15,987.4 | 60,784.6 | 5,804.4 | 0.69 | 3,061.4 |
| 7 | 810,401.99 | 30,430.0 | 9,780.7 | 9,623.3 | 59,923.7 | 16,139.4 | 61,368.8 | 5,860.2 | 0.69 | 3,090.2 |
| 8 | 849,192.66 | 29,637.1 | 9,351.4 | 9,382.3 | 58,361.3 | 15,720.3 | 59,770.5 | 5,707.4 | 0.70 | 3,009.9 |
| Sum | 6,562,757.51 | 237,756.7 | 75,243.0 | 74,840.0 | 465,735.9 | 125,429.3 | 476,933.9 | 45,516.0 | 5.54 | 24,016.3 |
| Avg | 820,344.69 | 29,719.6 | 9,405.4 | 9,355.0 | 58,217.0 | 15,678.7 | 59,616.7 | 5,689.5 | 0.69 | 3,002.0 |
| Std | 52,225.47 | 1,156.3 | 322.7 | 279.8 | 1,774.7 | 480.3 | 1,826.4 | 172.1 | 0.00 | 92.0 |

The following screenshot was captured from the second Container Database ENGCDB while running another 500 users on this second OLTP databases at the same time for sustained 24-hour test. We captured about 294k IOPS (225k Reads/s and 70k Writes/s) with the 21k TPS (1,307,700 TPM) while running the workload test on two databases at the same time during this 24 hour sustained test.

System Statistics - Per Second  DB/Inst: ENGCDB/engcdb1  Snaps: 361-385

| I# | Logical Reads/s | Physical Reads/s | Physical Writes/s | Redo Size (k)/s | Block Changes/s | User Calls/s | Execs/s | Parses/s | Logons/s | Txns/s |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 610,682.67 | 28,061.2 | 8,272.3 | 8,226.0 | 49,925.2 | 12,454.4 | 49,571.1 | 4,526.3 | 0.85 | 2,610.0 |
| 2 | 664,987.66 | 27,647.4 | 8,765.9 | 8,709.4 | 52,893.9 | 13,196.2 | 52,515.6 | 4,791.9 | 0.85 | 2,765.7 |
| 3 | 682,988.57 | 28,874.0 | 8,480.7 | 8,423.8 | 51,082.6 | 12,750.0 | 50,739.5 | 4,629.7 | 0.84 | 2,672.1 |
| 4 | 688,307.23 | 27,770.4 | 8,513.2 | 8,449.6 | 51,236.7 | 12,785.2 | 50,889.7 | 4,644.1 | 0.84 | 2,679.3 |
| 5 | 735,730.60 | 27,406.6 | 8,809.5 | 8,588.7 | 52,174.2 | 13,021.3 | 51,818.1 | 4,728.5 | 0.85 | 2,728.6 |
| 6 | 857,941.22 | 27,181.8 | 8,611.7 | 8,568.6 | 51,953.9 | 12,964.0 | 51,589.9 | 4,707.5 | 0.85 | 2,717.3 |
| 7 | 680,314.49 | 29,734.6 | 8,789.2 | 8,742.5 | 53,028.5 | 13,239.0 | 52,684.0 | 4,807.2 | 0.84 | 2,774.5 |
| 8 | 663,155.39 | 28,476.3 | 9,018.0 | 8,967.5 | 54,435.0 | 13,590.8 | 54,083.6 | 4,935.4 | 0.84 | 2,848.3 |
| Sum | 5,584,107.84 | 225,152.3 | 69,260.6 | 68,676.3 | 416,730.0 | 104,001.0 | 413,891.5 | 37,770.6 | 6.77 | 21,795.8 |
| Avg | 698,013.48 | 28,144.0 | 8,657.6 | 8,584.5 | 52,091.3 | 13,000.1 | 51,736.4 | 4,721.3 | 0.85 | 2,724.5 |
| Std | 73,256.05 | 848.2 | 234.4 | 226.8 | 1,387.8 | 349.0 | 1,385.7 | 125.8 | 0.00 | 73.2 |

The following screenshot shows the SOECDB database “IO Profile” for the “Reads/s” and “Writes/s” requests for this multiple OLTP test running workload together for sustained 24-hour test. The Total Requests (Read and Write Per Second) were around “318k” with Total (MB) “Read+Write” Per Second was around “2572” MB/s for the first SOECDB database during this 24-hour test.

IO Profile (Global)  DB/Inst: SOECDB/soecdb1  Snaps: 566-590

| Statistic | Read+Write/s | Reads/s | Writes/s |
|---|---|---|---|
| Total Requests | 318,123.05 | 232,337.57 | 85,785.48 |
| Database Requests | 296,842.37 | 232,225.91 | 64,616.46 |
| Optimized Requests | 0.00 | 0.00 | 0.00 |
| Redo Requests | 11,251.69 | N/A | 11,251.69 |
| Total (MB) | 2,572.70 | 1,904.10 | 668.60 |
| Database (MB) | 2,445.29 | 1,857.46 | 587.83 |
| Optimized Total (MB) | 0.00 | 0.00 | 0.00 |
| Redo (MB) | 73.09 | N/A | 73.09 |
| Database (blocks) | 312,997.43 | 237,754.96 | 75,242.47 |
| Via Buffer Cache (blocks) | 307,446.54 | 232,216.86 | 75,229.68 |
| Direct (blocks) | 5,550.93 | 5,538.13 | 12.79 |

The following screenshot shows the ENGCDB database “IO Profile” for the “Reads/s” and “Writes/s” requests for this multiple OLTP test running workload together for sustained 24-hour test. The Total Requests (Read and Write Per Second) were around “300k” with Total (MB) “Read+Write” Per Second was around “2418” MB/s for the second ENGCDB database while running this workload for 24-hour.

IO Profile (Global) DB/Inst: ENGCDB/engcdb1 Snaps: 361-385

| Statistic | Read+Write/s | Reads/s | Writes/s |
|---|---|---|---|
| Total Requests | 300,951.46 | 220,964.40 | 79,987.06 |
| Database Requests | 280,827.97 | 220,859.16 | 59,968.81 |
| Optimized Requests | 0.00 | 0.00 | 0.00 |
| Redo Requests | 11,032.67 | N/A | 11,032.67 |
| Total (MB) | 2,418.02 | 1,802.67 | 615.36 |
| Database (MB) | 2,300.08 | 1,758.99 | 541.09 |
| Optimized Total (MB) | 0.00 | 0.00 | 0.00 |
| Redo (MB) | 67.07 | N/A | 67.07 |
| Database (blocks) | 294,410.62 | 225,150.60 | 69,260.02 |
| Via Buffer Cache (blocks) | 290,126.64 | 220,881.29 | 69,245.35 |
| Direct (blocks) | 4,284.02 | 4,269.35 | 14.67 |

The following screenshot, “OS Statistics by Instance” while running the workload test for 24-hour on two OLTP databases at the same time. As shown below, the workload was equally spread across all the databases clusters while the average CPU utilization was around 35 % overall.

OS Statistics By Instance DB/Inst: SOECDB/soecdb1 Snaps: 566-590
-> Listed in order of instance number, I#
-> End values are diplayed only if different from begin values

| | CPU | | | Load | | % CPU | | | | | Time (s) | | | Memory | End Values | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| I# | #CPUs | #Core | #Sckt | Begin | End | % Busy | % Usr | % Sys | % WIO | % Idl | Busy | Idle | Total | MB | #CPU | #Cor | #Sck | Memory (M) |
| 1 | 128 | 64 | 2 | 101.2 | 91.8 | 36.4 | 25.1 | 7.4 | 21.4 | 63.6 | 3,961,137.1 | 6,932,947.9 | 10,894,085.0 | 515,140.6 | | | | |
| 2 | 128 | 64 | 2 | 86.5 | 85.4 | 28.3 | 20.1 | 5.3 | 25.5 | 71.7 | 3,082,962.5 | 7,829,047.9 | 10,912,010.4 | 515,140.6 | | | | |
| 3 | 128 | 64 | 2 | 87.9 | 81.6 | 27.5 | 19.4 | 5.2 | 25.7 | 72.5 | 2,997,067.1 | 7,917,152.6 | 10,914,219.7 | 515,139.3 | | | | |
| 4 | 128 | 64 | 2 | 87.3 | 85.4 | 30.3 | 21.6 | 5.6 | 24.8 | 69.7 | 3,306,828.4 | 7,604,960.6 | 10,911,789.0 | 515,140.6 | | | | |
| 5 | 128 | 64 | 2 | 87.1 | 82.3 | 30.5 | 21.8 | 5.6 | 24.6 | 69.5 | 3,333,392.4 | 7,581,843.7 | 10,915,236.0 | 515,140.6 | | | | |
| 6 | 128 | 64 | 2 | 82.5 | 80.7 | 28.0 | 20.0 | 5.1 | 24.9 | 72.0 | 3,055,782.8 | 7,864,220.9 | 10,920,003.7 | 450,628.7 | | | | |
| 7 | 128 | 64 | 2 | 85.2 | 81.9 | 29.5 | 21.0 | 5.5 | 25.2 | 70.5 | 3,222,748.0 | 7,690,916.1 | 10,913,664.1 | 515,140.6 | | | | |
| 8 | 128 | 64 | 2 | 84.4 | 84.0 | 27.2 | 19.3 | 5.1 | 25.6 | 72.8 | 2,969,284.9 | 7,944,814.2 | 10,914,099.1 | 515,127.9 | | | | |
| Sum | | | | | | | | | | | 25,929,203.1 | 61,365,903.8 | 87,295,106.9 | | | | | |

The following screenshot captured from the Oracle AWR report shows the “Top Timed Events” and average wait time for the first SOECDB database for the entire duration of the 24-hour workload test.

Top Timed Events DB/Inst: SOECDB/soecdb1 Snaps: 566-590
-> Instance '*' - cluster wide summary
-> '*' Waits, %Timeouts, Wait Time Total(s) : Cluster-wide total for the wait event
-> '*' 'Wait Time Avg' : Cluster-wide average computed as (Wait Time Total / Event Waits)
-> '*' Summary 'Avg Wait Time ' : Per-instance 'Wait Time Avg ' used to compute the following statistics
-> '*' [Avg/Minimum/Maximum/Std Dev] : average/minimum/maximum/standard deviation of per-instance 'Wait Time Avg'
-> '*' Cnt : count of instances with wait times for the event

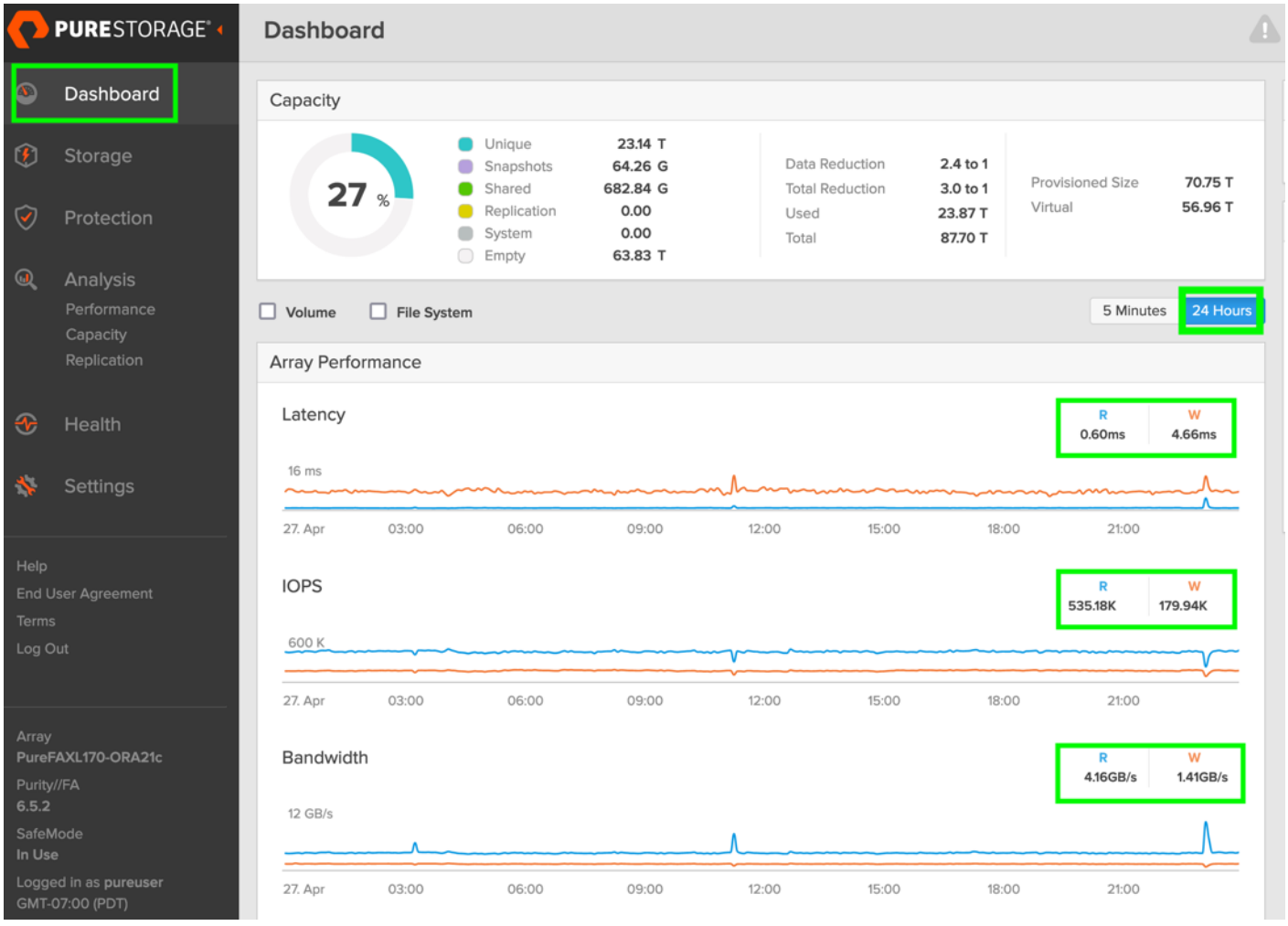
| | Wait | | Event | | Wait Time | | | Summary Avg Wait Time | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| I# | Class | Event | Waits | %Timeouts | Total(s) | Avg Wait | %DB time | Avg | Min | Max | Std Dev | Cnt |
| * | User I/O | db file sequential read | 1.741295E+10 | 0.0 | 1.5901031E+07 | 913.17us | 42.63 | 913.42us | 900.39us | .96ms | 18.90us | 8 |
| | | DB CPU | N/A | N/A | 7,134,351.92 | | 19.13 | | | | | 8 |
| | Commit | log file sync | 2.061055E+09 | 0.0 | 6,088,932.57 | 2.95ms | 16.32 | 2.95ms | 2.69ms | 3.19ms | 187.94us | 8 |
| | System I/O | db file parallel write | 1.751028E+09 | 0.0 | 1,976,960.23 | 1.13ms | 5.30 | 1.13ms | 1.07ms | 1.17ms | 42.79us | 8 |
| | Cluster | gc cr grant 2-way | 7.886311E+09 | 0.0 | 1,450,757.54 | 183.96us | 3.89 | 184.31us | 172.76us | 195.09us | 8.41us | 8 |
| | Cluster | gc current block 3-way | 5.194279E+09 | 0.0 | 1,230,961.05 | 236.98us | 3.30 | 237.12us | 230.63us | 243.04us | 4.95us | 8 |
| | Cluster | gc current grant 2-way | 6.040165E+09 | 0.0 | 1,115,555.59 | 184.69us | 2.99 | 184.90us | 178.58us | 199.67us | 6.93us | 8 |
| | Cluster | gc cr block busy | 108,875,532 | 0.0 | 892,799.75 | 8.20ms | 2.39 | 8.19ms | 7.08ms | 9.76ms | 1.07ms | 8 |
| | Cluster | gc cr block 3-way | 3.233596E+09 | 0.0 | 777,899.34 | 240.57us | 2.09 | 240.36us | 235.35us | 245.00us | 4.15us | 8 |
| | System I/O | log file parallel write | 1.946172E+09 | 0.0 | 749,207.38 | 384.96us | 2.01 | 385.13us | 370.71us | 398.03us | 10.24us | 8 |

The following screenshot captured from the Oracle AWR report shows the “Top Timed Events” and average wait time for the second ENGCDB database for the entire duration of the 24-hour sustained workload test.

Top Timed Events DB/Inst: ENGCDB/engcdb1 Snaps: 361-385
-> Instance '*' - cluster wide summary
-> '*' Waits, %Timeouts, Wait Time Total(s) : Cluster-wide total for the wait event
-> '*' 'Wait Time Avg' : Cluster-wide average computed as (Wait Time Total / Event Waits)
-> '*' Summary 'Avg Wait Time ' : Per-instance 'Wait Time Avg ' used to compute the following statistics
-> '*' [Avg/Min/Max/Std Dev] : average/minimum/maximum/standard deviation of per-instance 'Wait Time Avg'
-> '*' Cnt : count of instances with wait times for the event

| | Wait | | Event | | Wait Time | | | Summary Avg Wait Time | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| I# | Class | Event | Waits | %Timeouts | Total(s) | Avg Wait | %DB time | Avg | Min | Max | Std Dev | Cnt |
| * | User I/O | db file sequential read | 1.676326E+10 | 0.0 | 1.5065772E+07 | 898.74us | 44.26 | 899.16us | 885.62us | 947.36us | 20.26us | 8 |
| | | DB CPU | N/A | N/A | 6,392,254.62 | | 18.78 | | | | | 8 |
| | Commit | log file sync | 1.871619E+09 | 0.0 | 5,365,067.43 | 2.87ms | 15.76 | 2.87ms | 2.64ms | 3.13ms | 185.44us | 8 |
| | System I/O | db file parallel write | 1.643327E+09 | 0.0 | 1,900,735.66 | 1.16ms | 5.58 | 1.16ms | 1.10ms | 1.22ms | 45.47us | 8 |
| | Cluster | gc cr grant 2-way | 7.945959E+09 | 0.0 | 1,366,480.27 | 171.97us | 4.01 | 172.15us | 160.86us | 176.81us | 4.74us | 8 |
| | Cluster | gc current block 3-way | 5.078796E+09 | 0.0 | 1,246,443.05 | 245.42us | 3.66 | 245.47us | 238.90us | 254.45us | 4.94us | 8 |
| | Cluster | gc current grant 2-way | 5.514754E+09 | 0.0 | 907,432.76 | 164.55us | 2.67 | 164.59us | 160.55us | 180.66us | 6.73us | 8 |
| | Cluster | gc cr block busy | 78,084,021 | 0.0 | 785,625.38 | 10.06ms | 2.31 | 10.05ms | 7.95ms | 12.60ms | 1.73ms | 8 |
| | User I/O | db file parallel read | 1.378192E+09 | 0.0 | 735,332.68 | 533.55us | 2.16 | 533.80us | 526.83us | 565.43us | 12.91us | 8 |
| | Cluster | gc cr block 3-way | 2.902745E+09 | 0.0 | 732,464.97 | 252.34us | 2.15 | 252.11us | 245.85us | 256.96us | 3.98us | 8 |

The following screenshot shows the Pure Storage FlashArray dashboard when multiple OLTP database was running the workload at the same time. The screenshot shows the average IOPS “680k” with the average throughput of “5300 MB/s” with the average latency around “3 millisecond”.



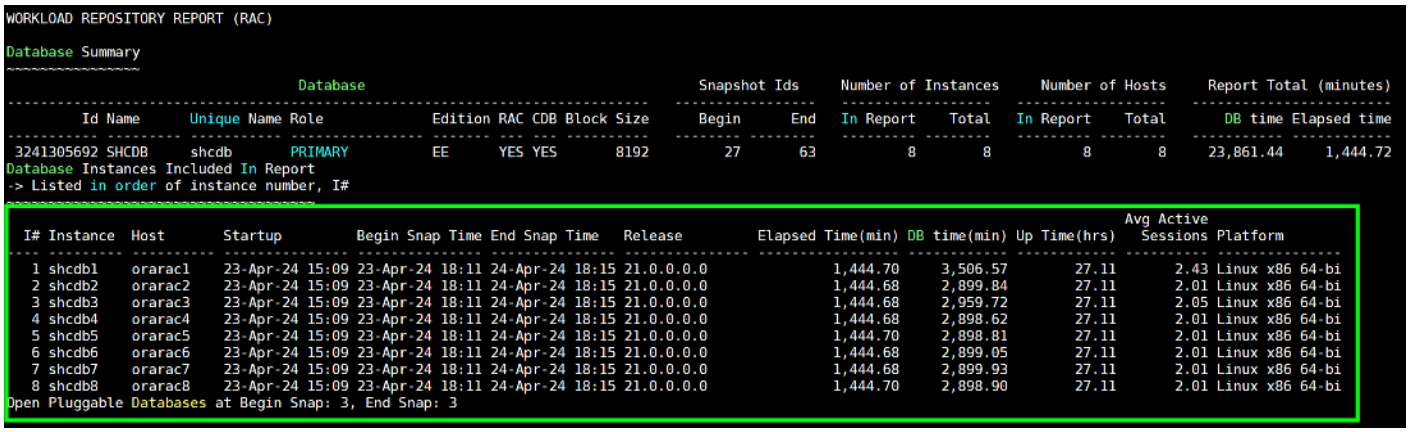

For the entire duration of the 24-hour test, we observed the system performance (IOPS, Latency and Throughput) was consistent throughout and we did not observe any dips in performance while running multiple OLTP database stress test.

## One DSS Database Performance

DSS database workloads are generally sequential in nature, read intensive and exercise large IO size. DSS database workload runs a small number of users that typically exercise extremely complex queries that run for hours. For running oracle database multitenancy architecture, we configured one container database as SHCDB and into that container, we created one pluggable database as SHPDB as explained earlier.

**Note:** We configured 4 TB of SHPDB pluggable database by loading Swingbench "SH" schema into Datafile Tablespace.

The following screenshot shows the database summary for the "SHCDB" database running for 24-hour duration. The container database "SHCDB" was also running with one pluggable databases "SHPDB" and the pluggable database was running the Swingbench SH workload for the entire 24-hour duration of the test across all eight RAC nodes.



WORKLOAD REPOSITORY REPORT (RAC)

Database Summary

| Database | | | | | | | Snapshot Ids | | Number of Instances | | Number of Hosts | | Report Total (minutes) | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Id | Name | Unique Name | Role | Edition | RAC | CDB | Block Size | Begin | End | In Report | Total | In Report | Total | DB time | Elapsed time |
| 3241305692 | SHCDB | shcdb | PRIMARY | EE | YES | YES | 8192 | 27 | 63 | 8 | 8 | 8 | 8 | 23,861.44 | 1,444.72 |

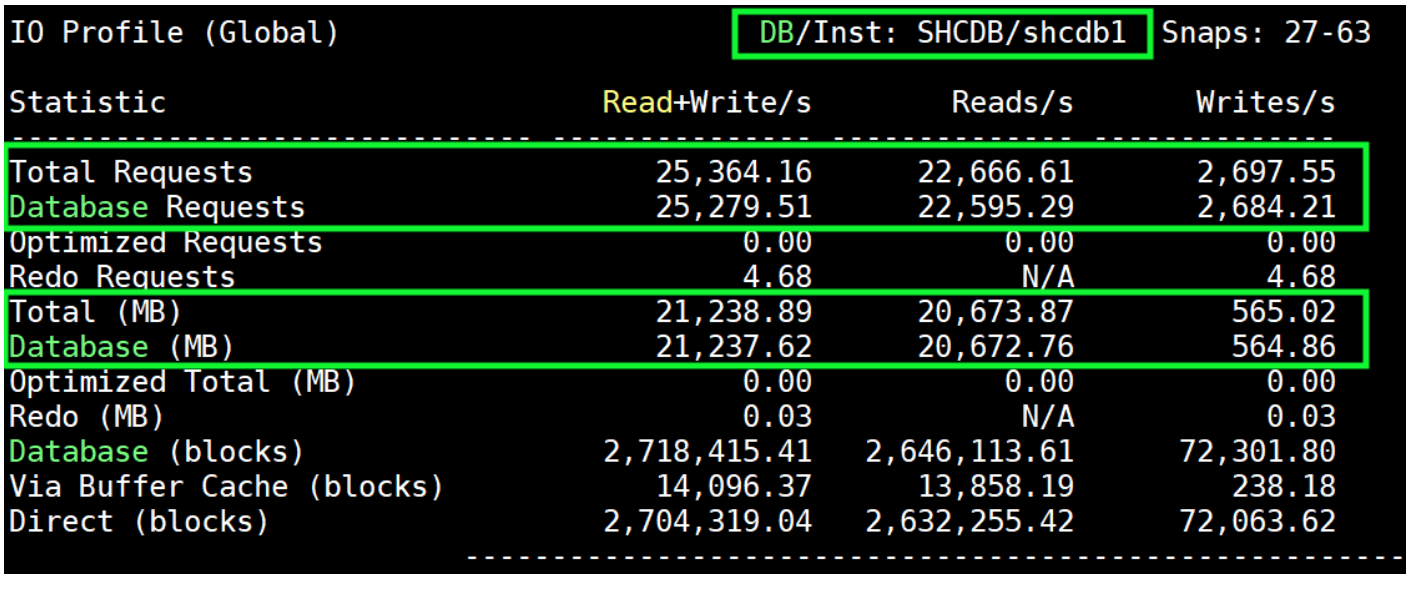
Database Instances Included In Report

-> Listed in order of instance number, I#

| I# | Instance | Host | Startup | Begin Snap Time | End Snap Time | Release | Elapsed Time(min) | DB time(min) | Up Time(hrs) | Avg Active Sessions | Platform |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | shcdb1 | orarac1 | 23-Apr-24 15:09 | 23-Apr-24 18:11 | 24-Apr-24 18:15 | 21.0.0.0.0 | 1,444.70 | 3,506.57 | 27.11 | 2.43 | Linux x86 64-bi |
| 2 | shcdb2 | orarac2 | 23-Apr-24 15:09 | 23-Apr-24 18:11 | 24-Apr-24 18:15 | 21.0.0.0.0 | 1,444.68 | 2,899.84 | 27.11 | 2.01 | Linux x86 64-bi |
| 3 | shcdb3 | orarac3 | 23-Apr-24 15:09 | 23-Apr-24 18:11 | 24-Apr-24 18:15 | 21.0.0.0.0 | 1,444.68 | 2,959.72 | 27.11 | 2.05 | Linux x86 64-bi |
| 4 | shcdb4 | orarac4 | 23-Apr-24 15:09 | 23-Apr-24 18:11 | 24-Apr-24 18:15 | 21.0.0.0.0 | 1,444.68 | 2,898.62 | 27.11 | 2.01 | Linux x86 64-bi |
| 5 | shcdb5 | orarac5 | 23-Apr-24 15:09 | 23-Apr-24 18:11 | 24-Apr-24 18:15 | 21.0.0.0.0 | 1,444.70 | 2,898.81 | 27.11 | 2.01 | Linux x86 64-bi |
| 6 | shcdb6 | orarac6 | 23-Apr-24 15:09 | 23-Apr-24 18:11 | 24-Apr-24 18:15 | 21.0.0.0.0 | 1,444.68 | 2,899.05 | 27.11 | 2.01 | Linux x86 64-bi |
| 7 | shcdb7 | orarac7 | 23-Apr-24 15:09 | 23-Apr-24 18:11 | 24-Apr-24 18:15 | 21.0.0.0.0 | 1,444.68 | 2,899.93 | 27.11 | 2.01 | Linux x86 64-bi |
| 8 | shcdb8 | orarac8 | 23-Apr-24 15:09 | 23-Apr-24 18:11 | 24-Apr-24 18:15 | 21.0.0.0.0 | 1,444.70 | 2,898.90 | 27.11 | 2.01 | Linux x86 64-bi |

Open Pluggable Databases at Begin Snap: 3, End Snap: 3

The following screenshot captured from Oracle AWR report shows the SHCDB database "IO Profile" for the "Reads/s" and "Writes/s" requests for the entire duration of the test. As the screenshots shows, the Total MB (Read and Write Per Second) were around "21,238 MB/s" (20,673 MB/s Reads/s & 565 MB/s Writes/s) for the SHPDB database while running this workload.



IO Profile (Global) DB/Inst: SHCDB/shcdb1 Snaps: 27-63

| Statistic | Read+Write/s | Reads/s | Writes/s |
|---|---|---|---|
| Total Requests | 25,364.16 | 22,666.61 | 2,697.55 |
| Database Requests | 25,279.51 | 22,595.29 | 2,684.21 |
| Optimized Requests | 0.00 | 0.00 | 0.00 |
| Redo Requests | 4.68 | N/A | 4.68 |
| Total (MB) | 21,238.89 | 20,673.87 | 565.02 |
| Database (MB) | 21,237.62 | 20,672.76 | 564.86 |
| Optimized Total (MB) | 0.00 | 0.00 | 0.00 |
| Redo (MB) | 0.03 | N/A | 0.03 |
| Database (blocks) | 2,718,415.41 | 2,646,113.61 | 72,301.80 |
| Via Buffer Cache (blocks) | 14,096.37 | 13,858.19 | 238.18 |
| Direct (blocks) | 2,704,319.04 | 2,632,255.42 | 72,063.62 |

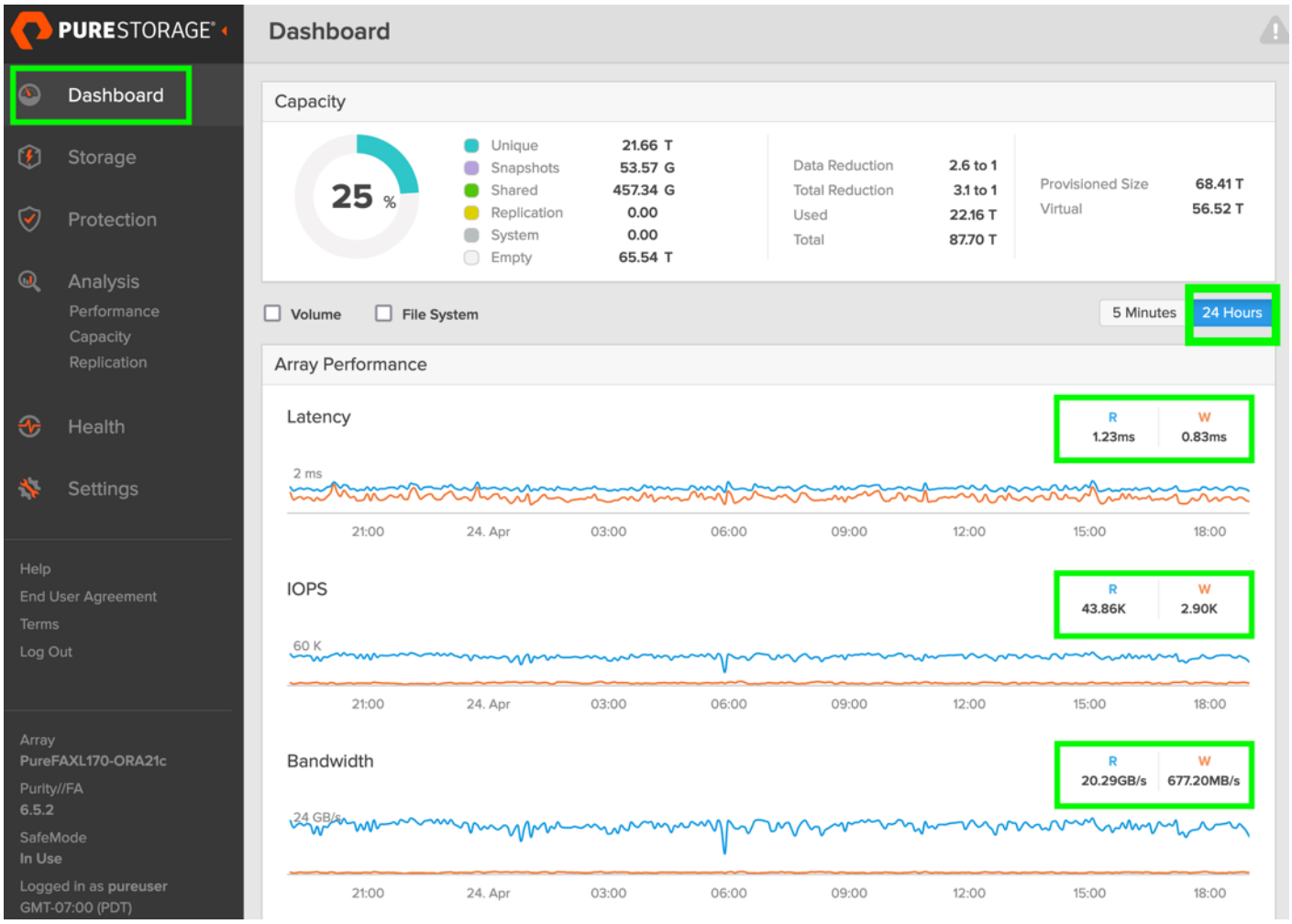
The following screenshot shows “Top Timed Events” for this container database SHCDB for the entire duration of the test while running SwingBench SH workload for 24-hours.

Top Timed Events DB/Inst: SHCDB/shcdb1 Snaps: 27-63

-> Instance '*' - cluster wide summary
-> '*' Waits, %Timeouts, Wait Time Total(s) : Cluster-wide total for the wait event
-> '*' 'Wait Time Avg' : Cluster-wide average computed as (Wait Time Total / Event Waits)
-> '*' Summary 'Avg Wait Time ' : Per-instance 'Wait Time Avg ' used to compute the following statistics
-> '*' [Avg/Min/Max/Std Dev] : average/minimum/maximum/standard deviation of per-instance 'Wait Time Avg'
-> '*' Cnt : count of instances with wait times for the event

| | Wait | | Event | | Wait Time | | | Summary Avg Wait Time | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| I# | Class | Event | Waits | %Timeouts | Total(s) | Avg Wait | %DB time | Avg | Min | Max | Std Dev | Cnt |
| * | | DB CPU | N/A | N/A | 1,019,924.27 | | 71.24 | | | | | 8 |
| | User I/O | direct path read temp | 416,523,433 | 0.0 | 232,773.86 | 558.85us | 16.26 | 559.00us | 540.18us | 600.50us | 21.07us | 8 |
| | User I/O | direct path read | 1.786443E+09 | 0.0 | 106,581.36 | 59.66us | 7.44 | 59.66us | 54.29us | 82.57us | 9.48us | 8 |
| | User I/O | direct path write temp | 238,377,952 | 0.0 | 22,736.88 | 95.38us | 1.59 | 95.35us | 82.80us | 102.12us | 6.43us | 8 |
| | User I/O | db file scattered read | 9,513,923 | 0.0 | 22,661.36 | 2.38ms | 1.58 | 2.37ms | 2.15ms | 2.51ms | 114.31us | 8 |
| | Cluster | gc cr multi block grant | 13,797,381 | 0.0 | 9,796.23 | 710.01us | 0.68 | 711.47us | 658.82us | 827.51us | 55.58us | 8 |
| | User I/O | local write wait | 20,061,116 | 0.0 | 7,409.34 | 369.34us | 0.52 | 369.28us | 363.99us | 374.29us | 3.71us | 8 |
| | System I/O | db file parallel write | 20,379,060 | 0.0 | 6,330.64 | 310.64us | 0.44 | 310.60us | 306.81us | 316.02us | 3.88us | 8 |
| | Other | gcs enter server mode | 276 | 0.0 | 3,972.45 | 14.39 s | 0.28 | 15.38 s | 9069.19ms | 28.04 s | 5693.22ms | 8 |
| | System I/O | control file sequential read | 6,098,213 | 0.0 | 3,916.31 | 642.21us | 0.27 | 643.17us | 608.71us | 744.32us | 43.51us | 8 |

The following screenshot captured through Pure Storage FlashArray dashboard shows the performance of the storage system while running Swingbench SH workload on single DSS database. The screenshot shows the average throughput of “20,800 MB/s (20.8 GB/s)” while running the SwingBench SH workload on one DSS database.



In this one DSS database use-case, we observed the DSS database performance was consistent throughout the test, and we did not observe any dips in performance for entire period of 24-hour test.

## Resiliency and Failure Tests

This chapter contains the following:



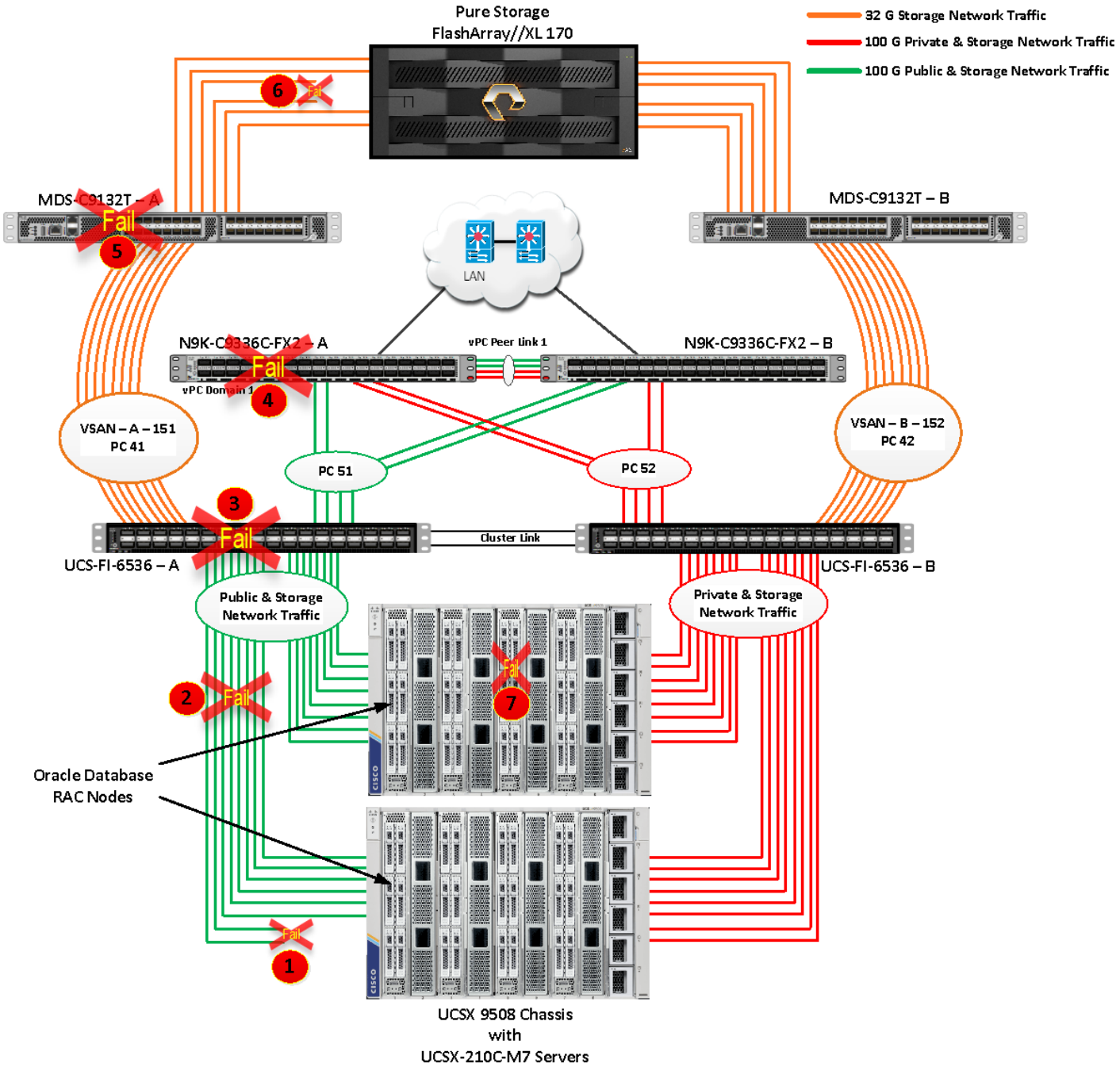
The goal of these tests was to ensure that the reference architecture withstands commonly occurring failures due to either unexpected crashes, hardware failures or human errors. We conducted many hardware (disconnect power), software (process kills) and OS specific failures that simulate the real world scenarios under stress conditions. In the destructive testing, we will also demonstrate the unique failover capabilities of Cisco UCS components used in this solution. Table 18 lists the test cases.

**Table 18. Hardware Failover Tests**

| Test Scenario | Tests Performed |
|---|---|
| Test 1: UCS-X Chassis IFM Link/Links Failure | Run the system on full Database workload.<br><br>Disconnect one or two links from any of the Chassis 1 IFM or Chassis 2 IFM by pulling it out and reconnect it after 10-15 minutes. Capture the impact on overall database performance. |
| Test 2: One of the FI Failure | Run the system on full Database workload.<br><br>Power Off one of the Fabric Interconnects and check the network traffic on the other Fabric Interconnect and capture the impact on overall database performance. |
| Test 3: One of the Nexus Switch Failure | Run the system on full Database workload.<br><br>Power Off one of the Cisco Nexus switches and check the network and storage traffic on the other Nexus switch. Capture the impact on overall database performance. |
| Test 4: One of the MDS Switch Failure | Run the system on full Database workload.<br><br>Power Off one of the Cisco MDS switches and check the network and storage traffic on the other MDS switch. Capture the impact on overall database performance. |
| Test 5: Storage Controller Links Failure | Run the system on full Database workload.<br><br>Disconnect one link from each of the Pure Storage Controllers by pulling it out and reconnect it after 10-15 minutes. Capture the impact on overall database |

| Test Scenario | Tests Performed |
| --- | --- |
| | performance. |
| Test 6: RAC Server Node Failure | Run the system on full Database workload.<br><br>Power Off one of the Linux Hosts and check the impact on database performance. |

The architecture below illustrates various failure scenario which can be occurred due to either unexpected crashes or hardware failures. The failure scenario 1 represents the Chassis IFM links failures while the scenario 2 represents the entire IFM module failure. Scenario 3 represents one of the Cisco UCS FI failures and similarly, scenario 4 and 5 represents one of the Cisco Nexus and MDS Switch failures. Scenario 6 represents the Pure Storage Controllers link failures and Scenario 7 represents one of the Server Node Failures.




**Note:** All Hardware failover tests were conducted with all three databases (SOEPDB, ENGPDB and SHPDB) running Swingbench mixed workloads.

As previously explained, we configured to carry Oracle Public Network traffic on “VLAN 135” through FI – A and Oracle Private Interconnect Network traffic on “VLAN 10” through FI – B under normal operating conditions before the failover tests. We configured FC & NVMe/FC Storage Network Traffic access from both the Fabric Interconnects to MDS Switches on VSAN 151 and VSAN 152.

The screenshots below show a complete infrastructure details of MAC address and VLAN information for Cisco UCS FI – A and FI – B Switches before failover test. Log into FI – A and type “connect nxos” then type “show mac address-table” to see all the VLAN connection on the switch:

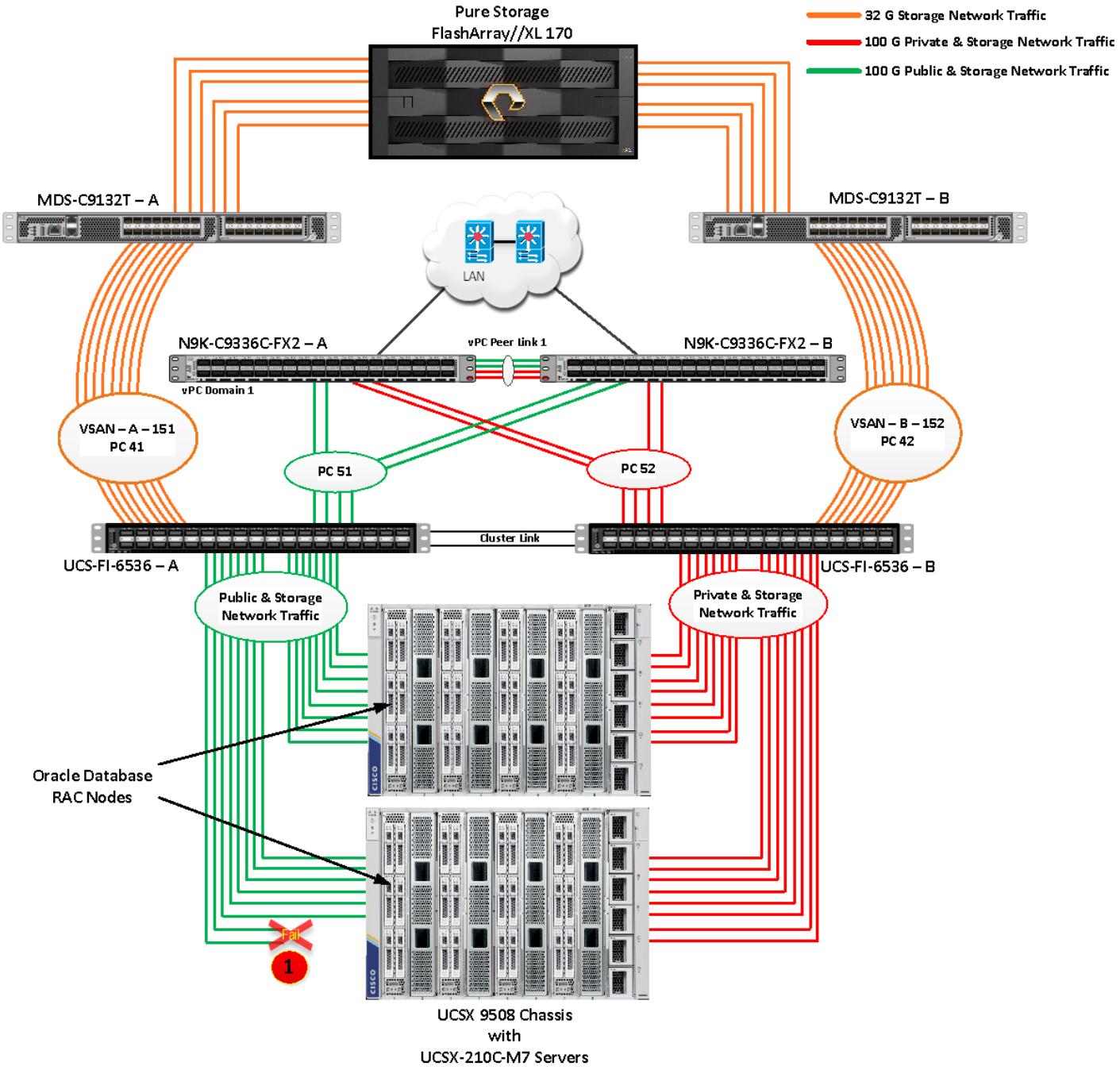
```
FS-ORA-FI-A(nx-os)# show mac address-table | grep 135
*  135     242a.04b0.9e29   dynamic  0         F      F    Eth2/1/33
*  135     242a.04b0.9e59   dynamic  0         F      F    Eth1/1/33
*  135     242a.04b0.9eb9   dynamic  0         F      F    Eth1/1/33
*  135     5451.ded5.81e1   dynamic  0         F      F    Eth1/1/33
*  135     5451.ded5.83f1   dynamic  0         F      F    Eth2/1/33
*  135     5451.ded5.87c1   dynamic  0         F      F    Eth1/1/33
*  135     5cb1.2e4a.8289   dynamic  0         F      F    Eth2/1/33
*  135     806a.00fa.1401   dynamic  0         F      F    Eth1/1/33
*  135     90eb.50cc.4581   dynamic  0         F      F    Eth2/1/33
*  135     d0dc.2c5f.9d99   dynamic  0         F      F    Eth2/1/33
*  135     0025.b5a3.5000   static   -         F      F    Veth800
*  135     0025.b5a3.5001   static   -         F      F    Veth814
*  135     0025.b5a3.5002   static   -         F      F    Veth828
*  135     0025.b5a3.5003   static   -         F      F    Veth842
*  135     0025.b5a3.5004   static   -         F      F    Veth856
*  135     0025.b5a3.5005   static   -         F      F    Veth870
*  135     0025.b5a3.5006   static   -         F      F    Veth884
*  135     0025.b5a3.5007   static   -         F      F    Veth898
```

Similarly, log into FI – B and type “connect nxos” then type “show mac address-table” to see all the VLAN connection on the switch as follows:

```
FS-ORA-FI-B(nx-os)# show mac address-table | grep 10
*   10     0025.b5b3.5000   static   -         F      F    Veth802
*   10     0025.b5b3.5001   static   -         F      F    Veth816
*   10     0025.b5b3.5002   static   -         F      F    Veth830
*   10     0025.b5b3.5003   static   -         F      F    Veth844
*   10     0025.b5b3.5004   static   -         F      F    Veth858
*   10     0025.b5b3.5005   static   -         F      F    Veth872
*   10     0025.b5b3.5006   static   -         F      F    Veth886
*   10     0025.b5b3.5007   static   -         F      F    Veth900
```

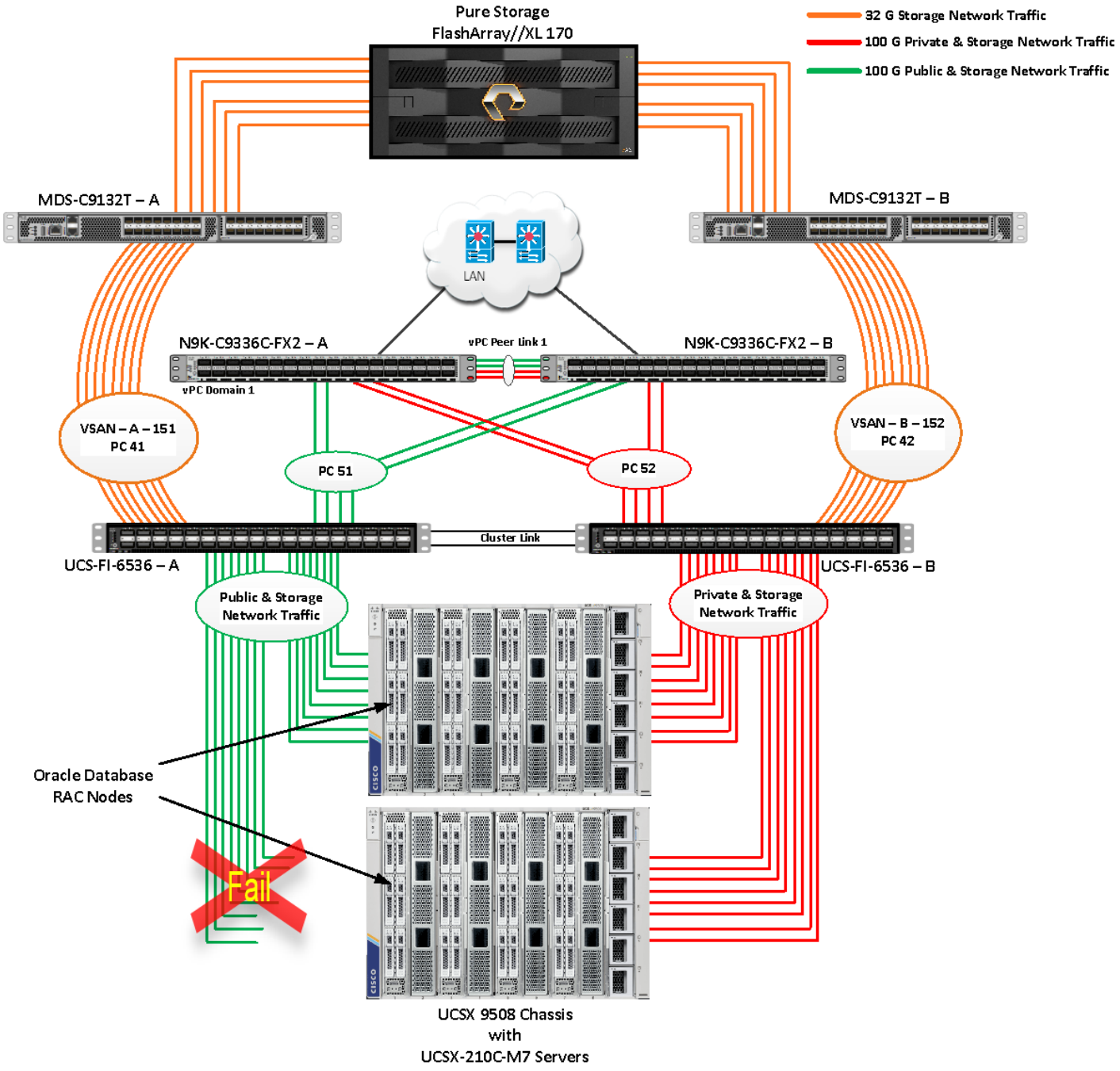
## Test 1 – Cisco UCS-X Chassis IFM Links Failure

We conducted the chassis IFM Links failure test on Cisco UCS Chassis 1 by disconnecting one of the server port link cables from the bottom chassis 1 as shown below:

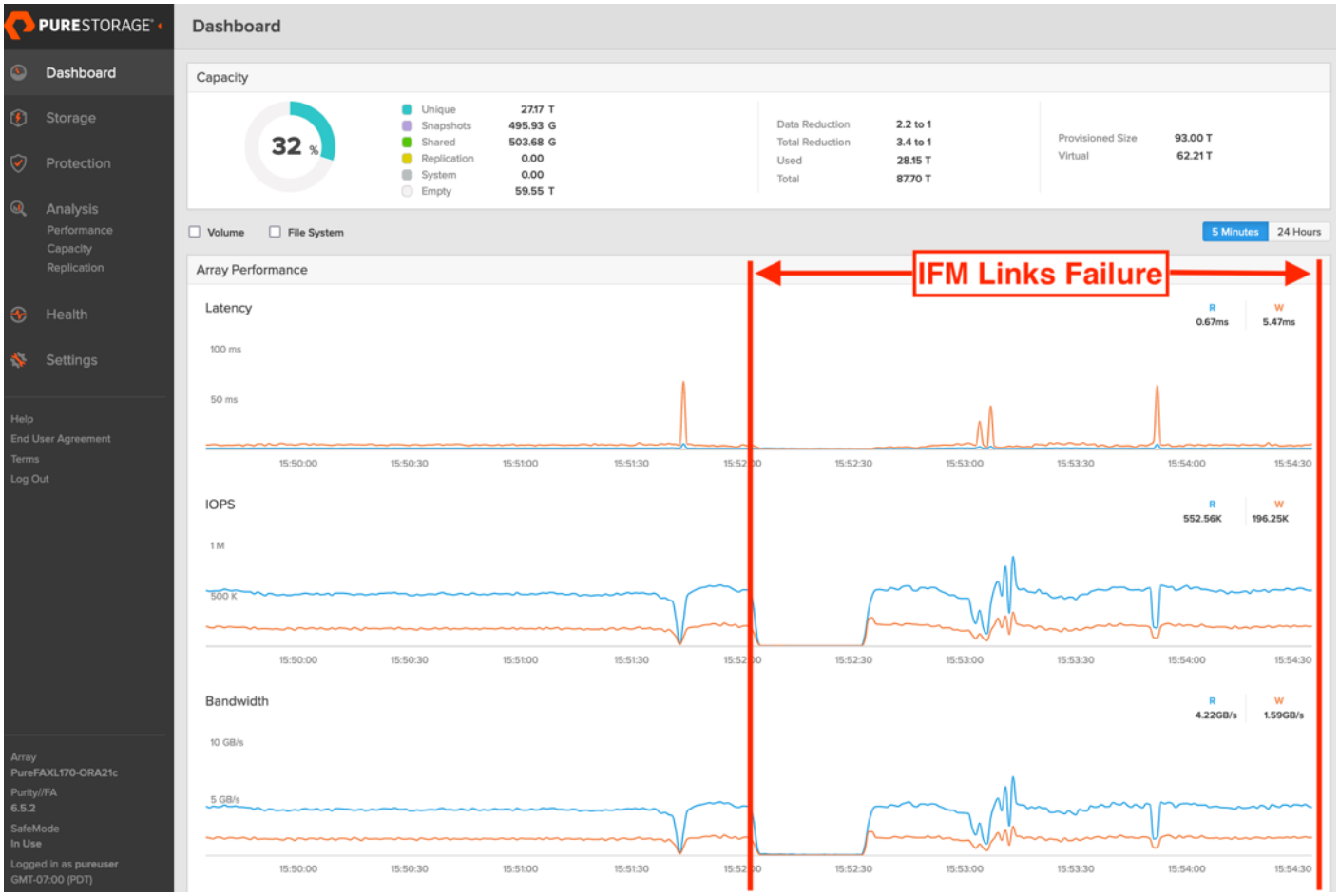



Unplug two server port cables from Chassis 1 and check all the VLAN and Storage traffic information on both Cisco UCS FIs, Database and Pure Storage. We noticed no disruption in any of the network and storage traffic and the database kept running under normal working conditions even after multiple IFM links failed from Chassis because of the Cisco UCS Port-Channel Feature.

We also conducted the IFM module test and removed the entire IFM module from one of the chassis as shown below:




The screenshot below shows the database workload performance from the storage array when the chassis IFM module links failed:

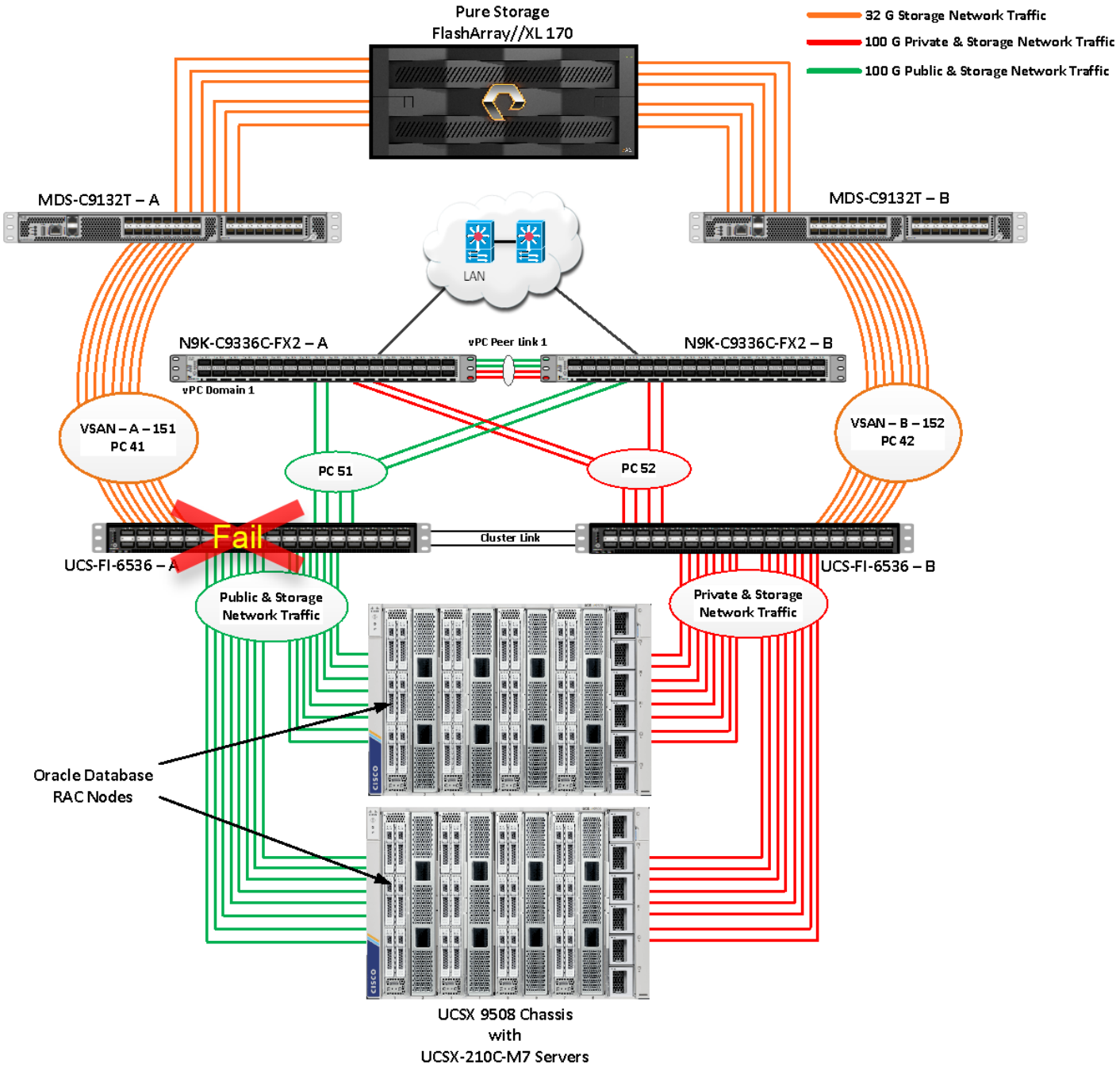



We noticed that IFM failure caused a momentary impact on the overall performance on OLTP as well as throughput of the DSS database for a few seconds but noticed that we did not see any interruption in any Private Server to Server Oracle RAC Interconnect Network, Management Public Network and Storage network traffic on IO Service Requests to the storage. We observed the database workload kept running under normal conditions throughout duration of IFM failure.

## Test 2 – One FI Failure

We conducted a hardware failure test on FI-A by disconnecting the power cable to the fabric interconnect switch.

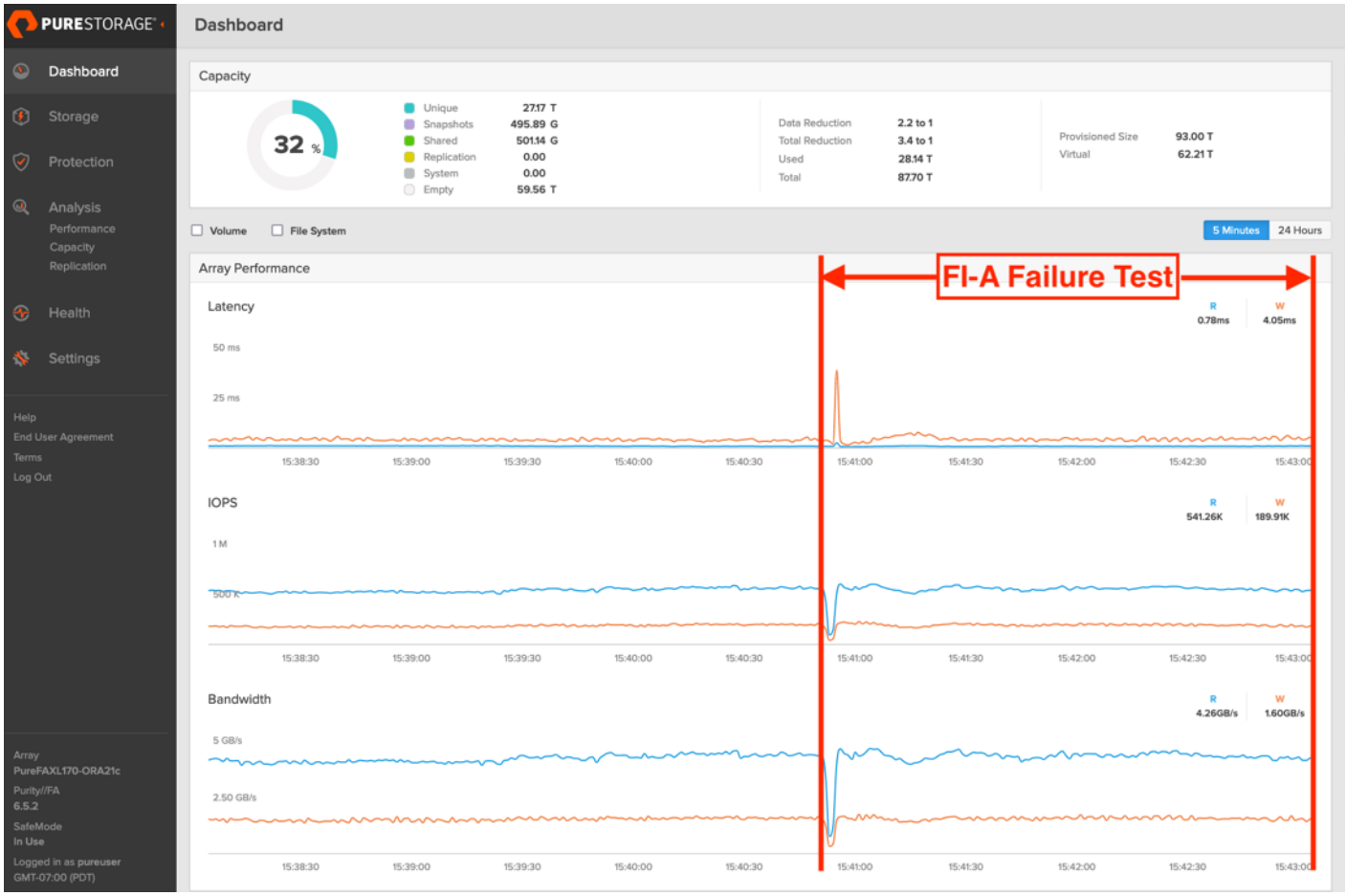
The figure below illustrates how during FI-A switch failure, the respective nodes (orarac1 to orarac4) on chassis 1 and nodes (orarac5 to orarac8) on chassis 2 will re-route the VLAN (135 – Management Network) traffic through the healthy Fabric Interconnect Switch FI-B. However, storage traffic VSANs from FI – A switch were not able to failover to FI – B because of those storage interfaces traffic is not capable of failing over to another switch.




Log into FI – B and type “connect nxos” then type “show mac address-table” to see all VLAN connection on FI – B. In the screenshot below, we noticed when the FI-A failed, all the MAC addresses of the redundant vNICs kept their VLANs network traffic going through FI-B. We observed that MAC addresses of public network vNICs (each server having 1 vNIC for VLAN 135) were failed over to other FI and database network traffic kept running under normal conditions even after failure of one of the FI.

```
FS-ORA-FI-B(nx-os)# show mac address-table
Legend:
        * - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
        age - seconds since last seen,+ - primary entry using vPC Peer-Link,
        (T) - True, (F) - False, C - ControlPlane MAC, ~ - vsan
   VLAN     MAC Address      Type      age     Secure NTFY Ports
---------+-----------------+--------+---------+------+----+------------------
*   10     0025.b5b3.5000   static   -         F      F    Veth802
*   10     0025.b5b3.5001   static   -         F      F    Veth816
*   10     0025.b5b3.5002   static   -         F      F    Veth830
*   10     0025.b5b3.5003   static   -         F      F    Veth844
*   10     0025.b5b3.5004   static   -         F      F    Veth858
*   10     0025.b5b3.5005   static   -         F      F    Veth872
*   10     0025.b5b3.5006   static   -         F      F    Veth886
*   10     0025.b5b3.5007   static   -         F      F    Veth900
*  135     04bd.97f1.da01   dynamic  0         F      F    Eth1/1/33
*  135     242a.04b0.9e59   dynamic  0         F      F    Eth1/1/33
*  135     246c.8419.9cc1   dynamic  0         F      F    Eth2/1/33
*  135     0025.b5a3.5000   static   -         F      F    Veth801
*  135     0025.b5a3.5001   static   -         F      F    Veth815
*  135     0025.b5a3.5002   static   -         F      F    Veth829
*  135     0025.b5a3.5003   static   -         F      F    Veth843
*  135     0025.b5a3.5004   static   -         F      F    Veth857
*  135     0025.b5a3.5005   static   -         F      F    Veth871
*  135     0025.b5a3.5006   static   -         F      F    Veth885
*  135     0025.b5a3.5007   static   -         F      F    Veth899
```

However, Storage Network Traffic for VSAN 151 were not able to fail-over to another FI Switch and thus we lost half of the storage traffic connectivity from the Oracle RAC Databases to Storage Array. The screenshot below shows the Pure Storage FlashArray performance of the mixed workloads on all the databases while one of the FI failed.



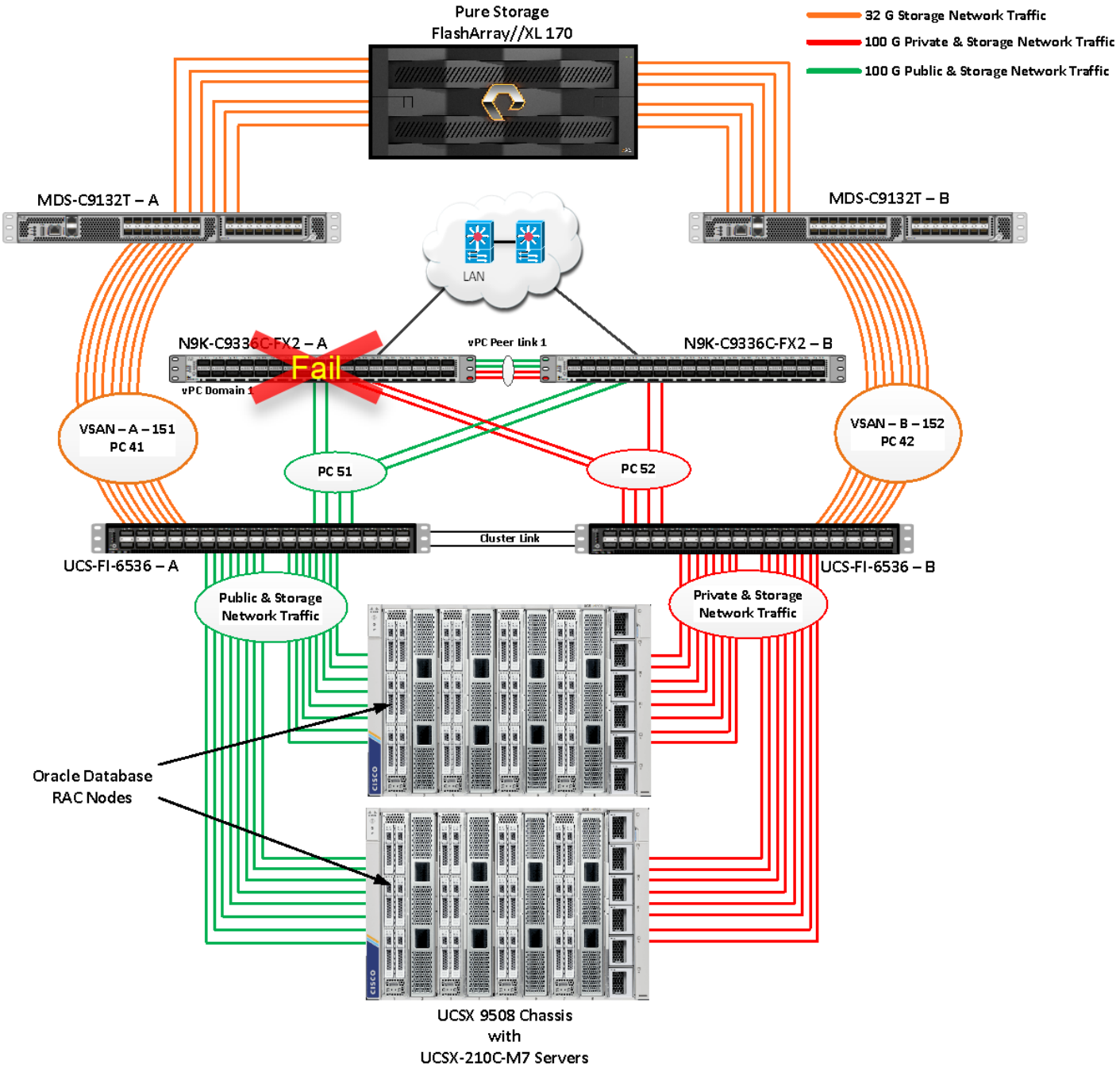

We also monitored and captured databases and its performance during this FI failure test through database alert log files and AWR reports. When we disconnected the power from FI – A, it caused a momentary impact on performance on the overall total IOPS, latency on OLTP as well as throughput on the DSS database for a few seconds but noticed that we did not see any interruption in any Private Server to Server Oracle RAC Interconnect Network, Management Public Network and Storage network traffic on IO Service Requests to the storage. We observed the database workload kept running under normal conditions throughout duration of FI failure.

We noticed this behavior because each server node can failover vNICs from one fabric interconnect switch to another fabric interconnect switch but there is no vHBA storage traffic failover from one fabric interconnect switch to another fabric interconnect switch. Therefore, in case of any one fabric interconnect failure, we would lose half of the number of vHBAs or storage paths and consequently we observe momentary databases performance impact for few seconds on the overall system as shown in the graph (above).

After plugging back power cable to FI-A Switch, the respective nodes (orarac1 to orarac4) on chassis 1 and nodes (orarac5 to orarac8) on chassis 2 will route back the MAC addresses and its VLAN public network and storage network traffic to FI-A. After FI – A arrives in normal operating state, all the nodes to storage connectivity, the operating system level multipath configuration will bring back all the path back to active and database performance will resume to peak performance.

## Test 3 - Cisco Nexus Switch Failure

We conducted a hardware failure test on Cisco Nexus Switch-A by disconnecting the power cable to the Cisco Nexus Switch and checking the public, private and storage network traffic on Cisco Nexus Switch-B and the overall system as shown below:




The screenshot below shows the vpc summary on Cisco Nexus Switch B while Cisco Nexus A was down.

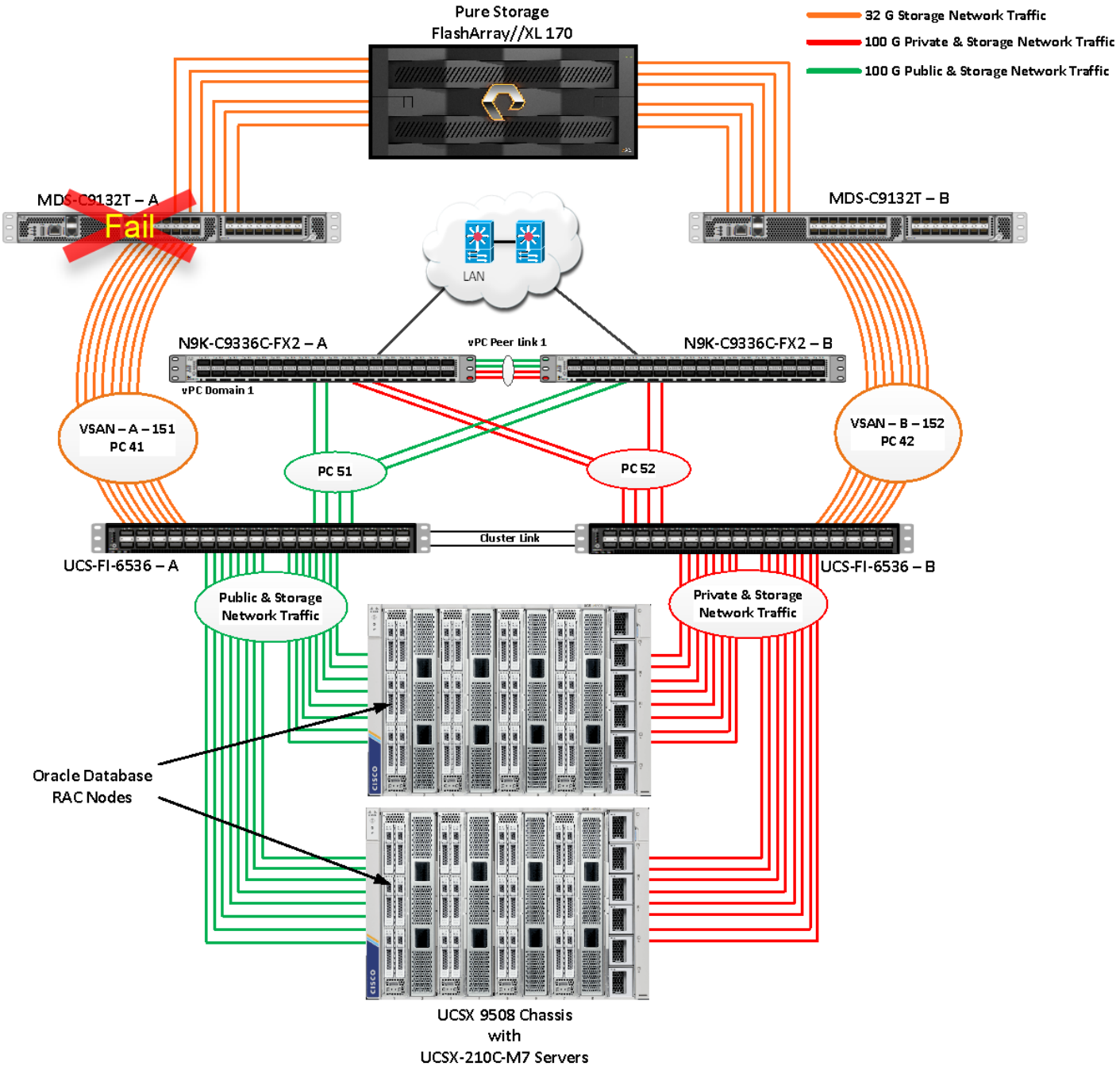
```
FS-ORA-N9K-B# show vpc brief
Legend:
                (*) - local vPC is down, forwarding via vPC peer-link

vPC domain id                     : 1
Peer status                       : peer link is down
vPC keep-alive status             : Suspended (Destination IP not reachable)
Configuration consistency status  : success
Per-vlan consistency status       : success
Type-2 consistency status         : success
vPC role                          : primary
Number of vPCs configured         : 2
Peer Gateway                      : Disabled
Dual-active excluded VLANs        : -
Graceful Consistency Check        : Enabled
Auto-recovery status              : Enabled, timer is off.(timeout = 240s)
Delay-restore status              : Timer is off.(timeout = 30s)
Delay-restore SVI status          : Timer is off.(timeout = 10s)
Operational Layer3 Peer-router    : Disabled
Virtual-peerlink mode             : Disabled

vPC Peer-link status
---------------------------------------------------------------------
id    Port   Status Active vlans
--    ----   ------ --------------------------------------------------
1     Po1    down   -

vPC status
----------------------------------------------------------------------------
Id    Port          Status Consistency Reason                Active vlans
--    ------------  ------ ----------- ------                ---------------
51    Po51          up     success     Type checks were      1,10,135
                                       bypassed for the vPC
52    Po52          up     success     Type checks were      1,10,135
                                       bypassed for the vPC

Please check "show vpc consistency-parameters vpc <vpc-num>" for the
consistency reason of down vpc and for type-2 consistency reasons for
any vpc.
```

When we disconnected the power from Cisco Nexus-A Switch, it caused no impact on database performance of the overall total IOPS, latency on OLTP as well as throughput of the DSS database and noticed no interruption in the overall Private Server to Server Oracle RAC Interconnect Network, Management Public Network, and storage network traffic on I/O Service Requests to the storage.

Such as FI failure tests, we observed no impact overall on all three databases performance and all the VLAN network traffic were going through other active Cisco Nexus switch B and databases workload kept running under normal conditions throughout the duration of Nexus failure. After plugging back the power cable back into Cisco Nexus-A Switch, Nexus Switch returns to normal operating state and database performance continue peak performance.

### Test 4 - Cisco MDS Switch Failure

We conducted a hardware failure test on Cisco MDS Switch-A by disconnecting the power cable to the MDS Switch and checking the public, private and storage network traffic on Cisco MDS Switch-B and the overall system as shown below:



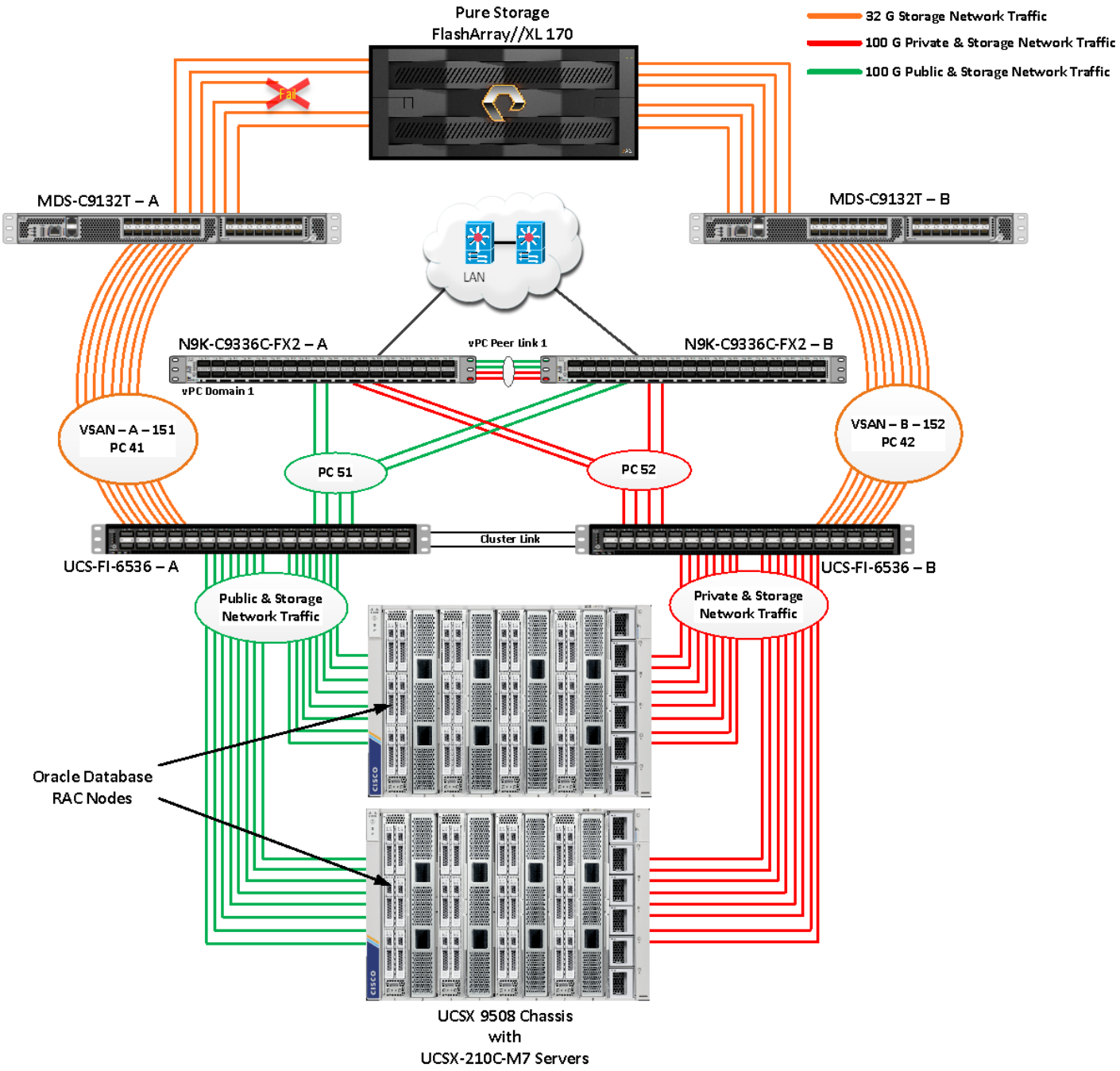

Like FI failure tests, we observed some impact on all three databases performance as we lost half of the VSAN (VSAN-A 151) traffic. While VSAN-A (151) stays locally into the switch and only carry storage traffic through the MDS switch A, VSAN-A doesn't failover to MDS Switch B therefore we reduced server to storage connectivity into half during MDS Switch A failure. However, failure in MDS Switch did not cause any disruption to Private and Public Network Traffic.

We also recorded performance of the databases from the storage dashboard where we observed momentary impact on performance on overall IOPS, latency on OLTP as well as throughput on DSS database for few seconds.

After plugging back power cable to MDS Switch A, the operating system level multipath configuration will bring back all the path back to active and database performance will resume to peak performance.

## Test 5 – Storage Controller Links Failure

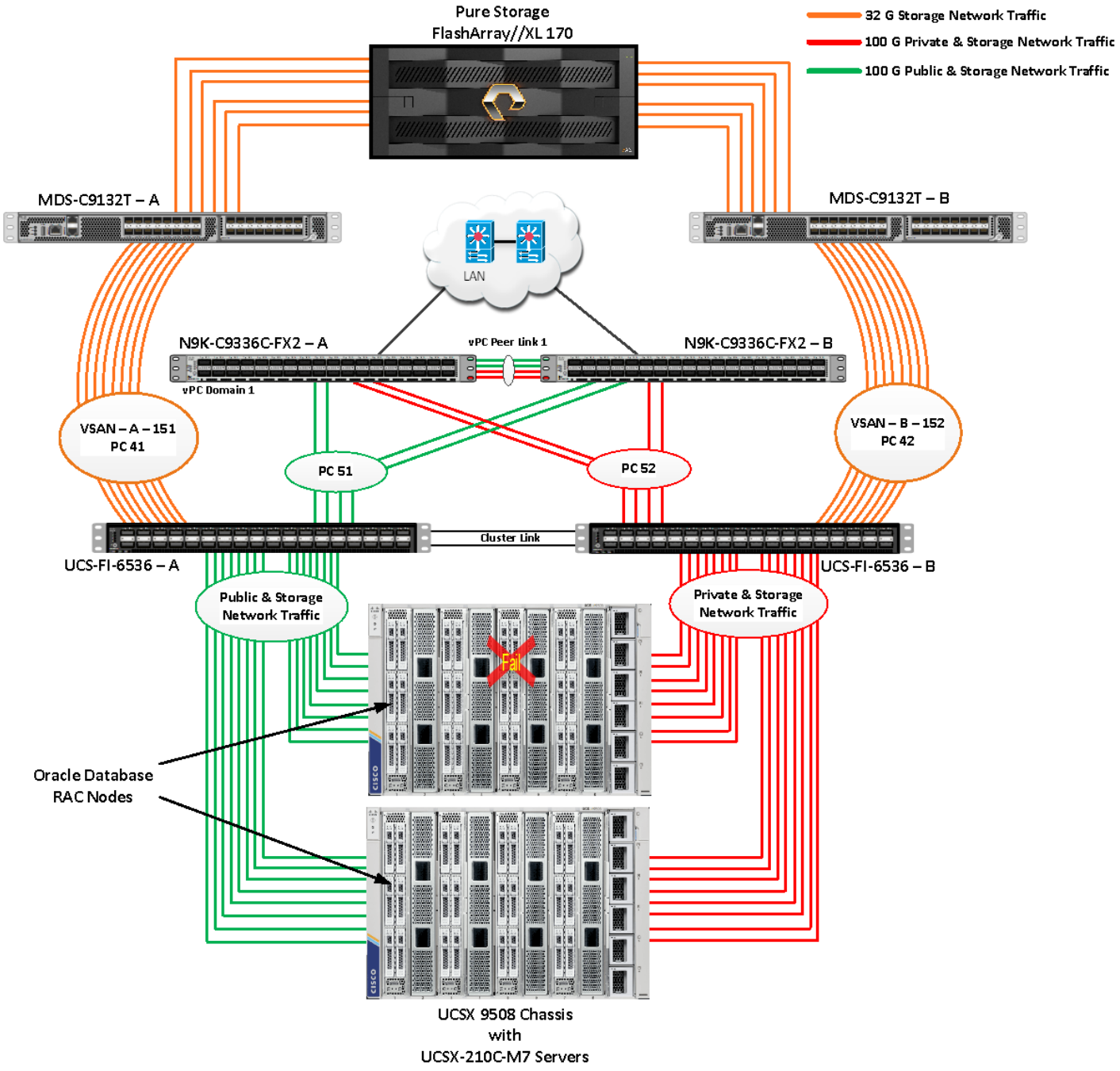
We performed storage controller link failure test by disconnecting two of the FC 32G links from the Pure Storage FlashArray from one of the storage controllers as shown below:



Like FI and MDS failure tests, storage link failure did not cause any disruption to Private, Public and Storage Network Traffic. After plugging back FC links to storage controller, MDS Switch and Storage array links comes back online, and the operating system level multipath configuration will bring back all the path back to active and database performance will resume to peak performance.

### Test 6 – Oracle RAC Server Node Failure

In this test, we started the SwingBench workload test run on all eight RAC nodes, and then during run, we powered down one node from the RAC cluster to check the overall system performance. We didn't observe any performance impact on overall database IOPS, latency and throughput after losing one node from the system.




We completed an additional failure scenario and validated that there is no single point of failure in this reference design.

## Summary

FlashStack is a converged infrastructure solution jointly developed by Cisco and Pure Storage. It combines computing, networking, and storage components into a pre-validated, integrated architecture designed to simplify data center operations and improve efficiency.

Key features of FlashStack include:

- All-flash storage: FlashStack utilizes Pure Storage's all-flash arrays, providing high performance with consistent sub-millisecond latency.
- AI-based management: FlashStack uses artificial intelligence for infrastructure management, improving business outcomes and simplifying operations.
- Flexible consumption models: It offers as-a-service consumption options, allowing organizations to align costs with usage.
- Validated designs: FlashStack provides pre-validated designs for popular workloads, reducing deployment risks and simplifying implementation.
- Cloud integration: The solution supports hybrid cloud environments, offering cloud-like agility and pay-as-you-use pricing.

FlashStack is an ideal platform for the architecture of mission critical database workloads such as Oracle RAC. The combination of Cisco UCS, Pure Storage and Oracle Real Application Cluster Database architecture can accelerate your IT transformation by enabling faster deployments, greater flexibility of choice, efficiency, high availability, and lower risk. The FlashStack Datacenter solution is a validated approach for deploying Cisco and Pure Storage System technologies and products to build shared private and public cloud infrastructure.

If you're interested in understanding the FlashStack design and deployment details, including the configuration of various elements of design and associated best practices, refer to Cisco Validated Designs for FlashStack, here:

https://www.cisco.com/c/en/us/solutions/design-zone/data-center-design-guides/data-center-design-guides-all.html#FlashStack

https://support.purestorage.com/bundle/m_flashstack_reference_architectures/page/FlashStack/FlashStack_Reference_Architectures/topics/concept/c_50th_celebration_of_flashstack_cisco_validated_designs.html

## About the Author

**Hardikkumar Vyas, Technical Marketing Engineer, CSPG UCS Product Management and Data Center Solutions Engineering Group, Cisco Systems, Inc.**

Hardikkumar Vyas is a Solution Architect in Cisco System's Cloud and Compute Engineering Group for configuring, implementing, and validating infrastructure best practices for highly available Oracle RAC databases solutions on Cisco UCS Servers, Cisco Nexus Products, and various Storage Technologies. Hardikkumar holds a master's degree in electrical engineering and has over 12 years of experience working with Oracle RAC Databases and associated applications. Hardikkumar's focus is developing database solutions on different platforms, perform benchmarks, prepare reference architectures, and write technical documents for Oracle RAC Databases on Cisco UCS Platforms.

## Acknowledgements

For their support and contribution to the design, validation, and creation of this Cisco Validated Design, the authors would like to thank:

- Tushar Patel, Distinguished Technical Marketing Engineer, Cisco Systems, Inc.
- Vijay Kulari, Solution Architecture, Pure Storage Systems

## Appendix

This appendix contains the following:



### Compute

Cisco Intersight: https://www.intersight.com

Cisco Intersight Managed Mode: https://www.cisco.com/c/en/us/td/docs/unified_computing/Intersight/b_Intersight_Managed_Mode_Configuration_Guide.html

Cisco Unified Computing System: http://www.cisco.com/en/US/products/ps10265/index.html

Cisco UCS 6536 Fabric Interconnects: https://www.cisco.com/c/en/us/products/collateral/servers-unified-computing/ucs6536-fabric-interconnect-ds.html

### Network

Cisco Nexus 9000 Series Switches: http://www.cisco.com/c/en/us/products/switches/nexus-9000-series-switches/index.html

Cisco MDS 9132T Switches: https://www.cisco.com/c/en/us/products/collateral/storage-networking/mds-9100-series-multilayer-fabric-switches/datasheet-c78-739613.html

### Storage

Pure Storage: https://www.purestorage.com/products/unified-block-file-storage.html

## Interoperability Matrix

Cisco UCS Hardware Compatibility Matrix: https://ucshcltool.cloudapps.cisco.com/public/

## Cisco MDS Switch Configuration

```
FS-ORA-MDS-A# show running-config


!Command: show running-config

!No configuration change since last restart

!Time: Sat June 13 10:45:28 2024


version 9.3(2)

power redundancy-mode redundant

system default switchport trunk mode auto

system default switchport mode F

feature fport-channel-trunk

role name default-role

  description This is a system defined role and applies to all users.

  rule 5 permit show feature environment

  rule 4 permit show feature hardware

  rule 3 permit show feature module

  rule 2 permit show feature snmp

  rule 1 permit show feature system

ip domain-lookup

ntp server 72.163.32.44

vsan database

  vsan 151 name "VSAN-FI-A"

no device-alias mode enhanced

device-alias database

  device-alias name ORARAC-1-FC-HBA0 pwwn 20:00:00:25:b5:ab:30:00
```

```
device-alias name ORARAC-2-FC-HBA0 pwwn 20:00:00:25:b5:ab:30:0a
device-alias name ORARAC-3-FC-HBA0 pwwn 20:00:00:25:b5:ab:30:14
device-alias name ORARAC-4-FC-HBA0 pwwn 20:00:00:25:b5:ab:30:1e
device-alias name ORARAC-5-FC-HBA0 pwwn 20:00:00:25:b5:ab:30:28
device-alias name ORARAC-6-FC-HBA0 pwwn 20:00:00:25:b5:ab:30:32
device-alias name ORARAC-7-FC-HBA0 pwwn 20:00:00:25:b5:ab:30:3c
device-alias name ORARAC-8-FC-HBA0 pwwn 20:00:00:25:b5:ab:30:46
device-alias name ORARAC1-NVMe-HBA2 pwwn 20:00:00:25:b5:ab:30:02
device-alias name ORARAC1-NVMe-HBA4 pwwn 20:00:00:25:b5:ab:30:04
device-alias name ORARAC1-NVMe-HBA6 pwwn 20:00:00:25:b5:ab:30:06
device-alias name ORARAC1-NVMe-HBA8 pwwn 20:00:00:25:b5:ab:30:08
device-alias name ORARAC2-NVMe-HBA2 pwwn 20:00:00:25:b5:ab:30:0c
device-alias name ORARAC2-NVMe-HBA4 pwwn 20:00:00:25:b5:ab:30:0e
device-alias name ORARAC2-NVMe-HBA6 pwwn 20:00:00:25:b5:ab:30:10
device-alias name ORARAC2-NVMe-HBA8 pwwn 20:00:00:25:b5:ab:30:12
device-alias name ORARAC3-NVMe-HBA2 pwwn 20:00:00:25:b5:ab:30:16
device-alias name ORARAC3-NVMe-HBA4 pwwn 20:00:00:25:b5:ab:30:18
device-alias name ORARAC3-NVMe-HBA6 pwwn 20:00:00:25:b5:ab:30:1a
device-alias name ORARAC3-NVMe-HBA8 pwwn 20:00:00:25:b5:ab:30:1c
device-alias name ORARAC4-NVMe-HBA2 pwwn 20:00:00:25:b5:ab:30:20
device-alias name ORARAC4-NVMe-HBA4 pwwn 20:00:00:25:b5:ab:30:22
device-alias name ORARAC4-NVMe-HBA6 pwwn 20:00:00:25:b5:ab:30:24
device-alias name ORARAC4-NVMe-HBA8 pwwn 20:00:00:25:b5:ab:30:26
device-alias name ORARAC5-NVMe-HBA2 pwwn 20:00:00:25:b5:ab:30:2a
device-alias name ORARAC5-NVMe-HBA4 pwwn 20:00:00:25:b5:ab:30:2c
device-alias name ORARAC5-NVMe-HBA6 pwwn 20:00:00:25:b5:ab:30:2e
device-alias name ORARAC5-NVMe-HBA8 pwwn 20:00:00:25:b5:ab:30:30
device-alias name ORARAC6-NVMe-HBA2 pwwn 20:00:00:25:b5:ab:30:34
device-alias name ORARAC6-NVMe-HBA4 pwwn 20:00:00:25:b5:ab:30:36
```

```
  device-alias name ORARAC6-NVMe-HBA6 pwwn 20:00:00:25:b5:ab:30:38
  device-alias name ORARAC6-NVMe-HBA8 pwwn 20:00:00:25:b5:ab:30:3a
  device-alias name ORARAC7-NVMe-HBA2 pwwn 20:00:00:25:b5:ab:30:3e
  device-alias name ORARAC7-NVMe-HBA4 pwwn 20:00:00:25:b5:ab:30:40
  device-alias name ORARAC7-NVMe-HBA6 pwwn 20:00:00:25:b5:ab:30:42
  device-alias name ORARAC7-NVMe-HBA8 pwwn 20:00:00:25:b5:ab:30:44
  device-alias name ORARAC8-NVMe-HBA2 pwwn 20:00:00:25:b5:ab:30:48
  device-alias name ORARAC8-NVMe-HBA4 pwwn 20:00:00:25:b5:ab:30:4a
  device-alias name ORARAC8-NVMe-HBA6 pwwn 20:00:00:25:b5:ab:30:4c
  device-alias name ORARAC8-NVMe-HBA8 pwwn 20:00:00:25:b5:ab:30:4e
  device-alias name PureFAXL170-ORA21c-CT0-FC04 pwwn 52:4a:93:7a:7e:04:85:04
  device-alias name PureFAXL170-ORA21c-CT0-FC06 pwwn 52:4a:93:7a:7e:04:85:06
  device-alias name PureFAXL170-ORA21c-CT0-FC32 pwwn 52:4a:93:7a:7e:04:85:80
  device-alias name PureFAXL170-ORA21c-CT1-FC04 pwwn 52:4a:93:7a:7e:04:85:14
  device-alias name PureFAXL170-ORA21c-CT1-FC06 pwwn 52:4a:93:7a:7e:04:85:16
  device-alias name PureFAXL170-ORA21c-CT1-FC32 pwwn 52:4a:93:7a:7e:04:85:90

device-alias commit

system default zone distribute full
zone smart-zoning enable vsan 151
zoneset distribute full vsan 151
!Active Zone Database Section for vsan 151

zone name ORARAC-1-Boot-A vsan 151
    member pwwn 20:00:00:25:b5:ab:30:00 init
    !           [ORARAC-1-FC-HBA0]
    member pwwn 52:4a:93:7a:7e:04:85:04 target
    !           [PureFAXL170-ORA21c-CT0-FC04]
```

```
    member pwwn 52:4a:93:7a:7e:04:85:14 target
    !           [PureFAXL170-ORA21c-CT1-FC04]

zone name ORARAC-2-Boot-A vsan 151
    member pwwn 20:00:00:25:b5:ab:30:0a init
    !           [ORARAC-2-FC-HBA0]
    member pwwn 52:4a:93:7a:7e:04:85:04 target
    !           [PureFAXL170-ORA21c-CT0-FC04]
    member pwwn 52:4a:93:7a:7e:04:85:14 target
    !           [PureFAXL170-ORA21c-CT1-FC04]

zone name ORARAC-3-Boot-A vsan 151
    member pwwn 20:00:00:25:b5:ab:30:14 init
    !           [ORARAC-3-FC-HBA0]
    member pwwn 52:4a:93:7a:7e:04:85:04 target
    !           [PureFAXL170-ORA21c-CT0-FC04]
    member pwwn 52:4a:93:7a:7e:04:85:14 target
    !           [PureFAXL170-ORA21c-CT1-FC04]

zone name ORARAC-4-Boot-A vsan 151
    member pwwn 20:00:00:25:b5:ab:30:1e init
    !           [ORARAC-4-FC-HBA0]
    member pwwn 52:4a:93:7a:7e:04:85:04 target
    !           [PureFAXL170-ORA21c-CT0-FC04]
    member pwwn 52:4a:93:7a:7e:04:85:14 target
    !           [PureFAXL170-ORA21c-CT1-FC04]

zone name ORARAC-5-Boot-A vsan 151
    member pwwn 20:00:00:25:b5:ab:30:28 init
```

```
    !           [ORARAC-5-FC-HBA0]
    member pwwn 52:4a:93:7a:7e:04:85:04 target
    !           [PureFAXL170-ORA21c-CT0-FC04]
    member pwwn 52:4a:93:7a:7e:04:85:14 target
    !           [PureFAXL170-ORA21c-CT1-FC04]

zone name ORARAC-6-Boot-A vsan 151
    member pwwn 20:00:00:25:b5:ab:30:32 init
    !           [ORARAC-6-FC-HBA0]
    member pwwn 52:4a:93:7a:7e:04:85:04 target
    !           [PureFAXL170-ORA21c-CT0-FC04]
    member pwwn 52:4a:93:7a:7e:04:85:14 target
    !           [PureFAXL170-ORA21c-CT1-FC04]

zone name ORARAC-7-Boot-A vsan 151
    member pwwn 20:00:00:25:b5:ab:30:3c init
    !           [ORARAC-7-FC-HBA0]
    member pwwn 52:4a:93:7a:7e:04:85:04 target
    !           [PureFAXL170-ORA21c-CT0-FC04]
    member pwwn 52:4a:93:7a:7e:04:85:14 target
    !           [PureFAXL170-ORA21c-CT1-FC04]

zone name ORARAC-8-Boot-A vsan 151
    member pwwn 20:00:00:25:b5:ab:30:46 init
    !           [ORARAC-8-FC-HBA0]
    member pwwn 52:4a:93:7a:7e:04:85:04 target
    !           [PureFAXL170-ORA21c-CT0-FC04]
    member pwwn 52:4a:93:7a:7e:04:85:14 target
    !           [PureFAXL170-ORA21c-CT1-FC04]
```

```
zone name ORARAC-1-NVMe-A1 vsan 151
    member pwwn 20:00:00:25:b5:ab:30:02 init
    !           [ORARAC1-NVMe-HBA2]
    member pwwn 20:00:00:25:b5:ab:30:04 init
    !           [ORARAC1-NVMe-HBA4]
    member pwwn 20:00:00:25:b5:ab:30:06 init
    !           [ORARAC1-NVMe-HBA6]
    member pwwn 20:00:00:25:b5:ab:30:08 init
    !           [ORARAC1-NVMe-HBA8]
    member pwwn 52:4a:93:7a:7e:04:85:06 target
    !           [PureFAXL170-ORA21c-CT0-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:16 target
    !           [PureFAXL170-ORA21c-CT1-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:80 target
    !           [PureFAXL170-ORA21c-CT0-FC32]
    member pwwn 52:4a:93:7a:7e:04:85:90 target
    !           [PureFAXL170-ORA21c-CT1-FC32]


zone name ORARAC-2-NVMe-A1 vsan 151
    member pwwn 20:00:00:25:b5:ab:30:0c init
    !           [ORARAC2-NVMe-HBA2]
    member pwwn 20:00:00:25:b5:ab:30:0e init
    !           [ORARAC2-NVMe-HBA4]
    member pwwn 20:00:00:25:b5:ab:30:10 init
    !           [ORARAC2-NVMe-HBA6]
    member pwwn 20:00:00:25:b5:ab:30:12 init
    !           [ORARAC2-NVMe-HBA8]
    member pwwn 52:4a:93:7a:7e:04:85:06 target
```

```
    !           [PureFAXL170-ORA21c-CT0-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:16 target
    !           [PureFAXL170-ORA21c-CT1-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:80 target
    !           [PureFAXL170-ORA21c-CT0-FC32]
    member pwwn 52:4a:93:7a:7e:04:85:90 target
    !           [PureFAXL170-ORA21c-CT1-FC32]

zone name ORARAC-3-NVMe-A1 vsan 151
    member pwwn 20:00:00:25:b5:ab:30:16 init
    !           [ORARAC3-NVMe-HBA2]
    member pwwn 20:00:00:25:b5:ab:30:18 init
    !           [ORARAC3-NVMe-HBA4]
    member pwwn 20:00:00:25:b5:ab:30:1a init
    !           [ORARAC3-NVMe-HBA6]
    member pwwn 20:00:00:25:b5:ab:30:1c init
    !           [ORARAC3-NVMe-HBA8]
    member pwwn 52:4a:93:7a:7e:04:85:06 target
    !           [PureFAXL170-ORA21c-CT0-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:16 target
    !           [PureFAXL170-ORA21c-CT1-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:80 target
    !           [PureFAXL170-ORA21c-CT0-FC32]
    member pwwn 52:4a:93:7a:7e:04:85:90 target
    !           [PureFAXL170-ORA21c-CT1-FC32]

zone name ORARAC-4-NVMe-A1 vsan 151
    member pwwn 20:00:00:25:b5:ab:30:20 init
    !           [ORARAC4-NVMe-HBA2]
```

```
    member pwwn 20:00:00:25:b5:ab:30:22 init
    !           [ORARAC4-NVMe-HBA4]
    member pwwn 20:00:00:25:b5:ab:30:24 init
    !           [ORARAC4-NVMe-HBA6]
    member pwwn 20:00:00:25:b5:ab:30:26 init
    !           [ORARAC4-NVMe-HBA8]
    member pwwn 52:4a:93:7a:7e:04:85:06 target
    !           [PureFAXL170-ORA21c-CT0-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:16 target
    !           [PureFAXL170-ORA21c-CT1-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:80 target
    !           [PureFAXL170-ORA21c-CT0-FC32]
    member pwwn 52:4a:93:7a:7e:04:85:90 target
    !           [PureFAXL170-ORA21c-CT1-FC32]

zone name ORARAC-5-NVMe-A1 vsan 151
    member pwwn 20:00:00:25:b5:ab:30:2a init
    !           [ORARAC5-NVMe-HBA2]
    member pwwn 20:00:00:25:b5:ab:30:2c init
    !           [ORARAC5-NVMe-HBA4]
    member pwwn 20:00:00:25:b5:ab:30:2e init
    !           [ORARAC5-NVMe-HBA6]
    member pwwn 20:00:00:25:b5:ab:30:30 init
    !           [ORARAC5-NVMe-HBA8]
    member pwwn 52:4a:93:7a:7e:04:85:06 target
    !           [PureFAXL170-ORA21c-CT0-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:16 target
    !           [PureFAXL170-ORA21c-CT1-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:80 target
```

```
    !           [PureFAXL170-ORA21c-CT0-FC32]
    member pwwn 52:4a:93:7a:7e:04:85:90 target
    !           [PureFAXL170-ORA21c-CT1-FC32]


zone name ORARAC-6-NVMe-A1 vsan 151
    member pwwn 20:00:00:25:b5:ab:30:34 init
    !           [ORARAC6-NVMe-HBA2]
    member pwwn 20:00:00:25:b5:ab:30:36 init
    !           [ORARAC6-NVMe-HBA4]
    member pwwn 20:00:00:25:b5:ab:30:38 init
    !           [ORARAC6-NVMe-HBA6]
    member pwwn 20:00:00:25:b5:ab:30:3a init
    !           [ORARAC6-NVMe-HBA8]
    member pwwn 52:4a:93:7a:7e:04:85:06 target
    !           [PureFAXL170-ORA21c-CT0-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:16 target
    !           [PureFAXL170-ORA21c-CT1-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:80 target
    !           [PureFAXL170-ORA21c-CT0-FC32]
    member pwwn 52:4a:93:7a:7e:04:85:90 target
    !           [PureFAXL170-ORA21c-CT1-FC32]


zone name ORARAC-7-NVMe-A1 vsan 151
    member pwwn 20:00:00:25:b5:ab:30:3e init
    !           [ORARAC7-NVMe-HBA2]
    member pwwn 20:00:00:25:b5:ab:30:40 init
    !           [ORARAC7-NVMe-HBA4]
    member pwwn 20:00:00:25:b5:ab:30:42 init
    !           [ORARAC7-NVMe-HBA6]
```

```
    member pwwn 20:00:00:25:b5:ab:30:44 init
    !           [ORARAC7-NVMe-HBA8]
    member pwwn 52:4a:93:7a:7e:04:85:06 target
    !           [PureFAXL170-ORA21c-CT0-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:16 target
    !           [PureFAXL170-ORA21c-CT1-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:80 target
    !           [PureFAXL170-ORA21c-CT0-FC32]
    member pwwn 52:4a:93:7a:7e:04:85:90 target
    !           [PureFAXL170-ORA21c-CT1-FC32]

zone name ORARAC-8-NVMe-A1 vsan 151
    member pwwn 20:00:00:25:b5:ab:30:48 init
    !           [ORARAC8-NVMe-HBA2]
    member pwwn 20:00:00:25:b5:ab:30:4a init
    !           [ORARAC8-NVMe-HBA4]
    member pwwn 20:00:00:25:b5:ab:30:4c init
    !           [ORARAC8-NVMe-HBA6]
    member pwwn 20:00:00:25:b5:ab:30:4e init
    !           [ORARAC8-NVMe-HBA8]
    member pwwn 52:4a:93:7a:7e:04:85:06 target
    !           [PureFAXL170-ORA21c-CT0-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:16 target
    !           [PureFAXL170-ORA21c-CT1-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:80 target
    !           [PureFAXL170-ORA21c-CT0-FC32]
    member pwwn 52:4a:93:7a:7e:04:85:90 target
    !           [PureFAXL170-ORA21c-CT1-FC32]
```

```
zoneset name ORARAC-A vsan 151
    member ORARAC-1-Boot-A
    member ORARAC-2-Boot-A
    member ORARAC-3-Boot-A
    member ORARAC-4-Boot-A
    member ORARAC-1-NVMe-A1
    member ORARAC-2-NVMe-A1
    member ORARAC-3-NVMe-A1
    member ORARAC-4-NVMe-A1
    member ORARAC-5-Boot-A
    member ORARAC-6-Boot-A
    member ORARAC-7-Boot-A
    member ORARAC-8-Boot-A
    member ORARAC-5-NVMe-A1
    member ORARAC-6-NVMe-A1
    member ORARAC-7-NVMe-A1
    member ORARAC-8-NVMe-A1

zoneset activate name ORARAC-A vsan 151
do clear zone database vsan 151
!Full Zone Database Section for vsan 151

zone name ORARAC-1-Boot-A vsan 151
    member pwwn 20:00:00:25:b5:ab:30:00 init
    !           [ORARAC-1-FC-HBA0]
    member pwwn 52:4a:93:7a:7e:04:85:04 target
    !           [PureFAXL170-ORA21c-CT0-FC04]
    member pwwn 52:4a:93:7a:7e:04:85:14 target
    !           [PureFAXL170-ORA21c-CT1-FC04]
```

```
zone name ORARAC-2-Boot-A vsan 151
    member pwwn 20:00:00:25:b5:ab:30:0a init
    !           [ORARAC-2-FC-HBA0]
    member pwwn 52:4a:93:7a:7e:04:85:04 target
    !           [PureFAXL170-ORA21c-CT0-FC04]
    member pwwn 52:4a:93:7a:7e:04:85:14 target
    !           [PureFAXL170-ORA21c-CT1-FC04]

zone name ORARAC-3-Boot-A vsan 151
    member pwwn 20:00:00:25:b5:ab:30:14 init
    !           [ORARAC-3-FC-HBA0]
    member pwwn 52:4a:93:7a:7e:04:85:04 target
    !           [PureFAXL170-ORA21c-CT0-FC04]
    member pwwn 52:4a:93:7a:7e:04:85:14 target
    !           [PureFAXL170-ORA21c-CT1-FC04]

zone name ORARAC-4-Boot-A vsan 151
    member pwwn 20:00:00:25:b5:ab:30:1e init
    !           [ORARAC-4-FC-HBA0]
    member pwwn 52:4a:93:7a:7e:04:85:04 target
    !           [PureFAXL170-ORA21c-CT0-FC04]
    member pwwn 52:4a:93:7a:7e:04:85:14 target
    !           [PureFAXL170-ORA21c-CT1-FC04]

zone name ORARAC-1-NVMe-A1 vsan 151
    member pwwn 20:00:00:25:b5:ab:30:02 init
    !           [ORARAC1-NVMe-HBA2]
    member pwwn 20:00:00:25:b5:ab:30:04 init
```

```
    !           [ORARAC1-NVMe-HBA4]
    member pwwn 20:00:00:25:b5:ab:30:06 init
    !           [ORARAC1-NVMe-HBA6]
    member pwwn 20:00:00:25:b5:ab:30:08 init
    !           [ORARAC1-NVMe-HBA8]
    member pwwn 52:4a:93:7a:7e:04:85:06 target
    !           [PureFAXL170-ORA21c-CT0-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:16 target
    !           [PureFAXL170-ORA21c-CT1-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:80 target
    !           [PureFAXL170-ORA21c-CT0-FC32]
    member pwwn 52:4a:93:7a:7e:04:85:90 target
    !           [PureFAXL170-ORA21c-CT1-FC32]

zone name ORARAC-2-NVMe-A1 vsan 151
    member pwwn 20:00:00:25:b5:ab:30:0c init
    !           [ORARAC2-NVMe-HBA2]
    member pwwn 20:00:00:25:b5:ab:30:0e init
    !           [ORARAC2-NVMe-HBA4]
    member pwwn 20:00:00:25:b5:ab:30:10 init
    !           [ORARAC2-NVMe-HBA6]
    member pwwn 20:00:00:25:b5:ab:30:12 init
    !           [ORARAC2-NVMe-HBA8]
    member pwwn 52:4a:93:7a:7e:04:85:06 target
    !           [PureFAXL170-ORA21c-CT0-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:16 target
    !           [PureFAXL170-ORA21c-CT1-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:80 target
    !           [PureFAXL170-ORA21c-CT0-FC32]
```

```
    member pwwn 52:4a:93:7a:7e:04:85:90 target
    !           [PureFAXL170-ORA21c-CT1-FC32]


zone name ORARAC-3-NVMe-A1 vsan 151
    member pwwn 20:00:00:25:b5:ab:30:16 init
    !           [ORARAC3-NVMe-HBA2]
    member pwwn 20:00:00:25:b5:ab:30:18 init
    !           [ORARAC3-NVMe-HBA4]
    member pwwn 20:00:00:25:b5:ab:30:1a init
    !           [ORARAC3-NVMe-HBA6]
    member pwwn 20:00:00:25:b5:ab:30:1c init
    !           [ORARAC3-NVMe-HBA8]
    member pwwn 52:4a:93:7a:7e:04:85:06 target
    !           [PureFAXL170-ORA21c-CT0-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:16 target
    !           [PureFAXL170-ORA21c-CT1-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:80 target
    !           [PureFAXL170-ORA21c-CT0-FC32]
    member pwwn 52:4a:93:7a:7e:04:85:90 target
    !           [PureFAXL170-ORA21c-CT1-FC32]


zone name ORARAC-4-NVMe-A1 vsan 151
    member pwwn 20:00:00:25:b5:ab:30:20 init
    !           [ORARAC4-NVMe-HBA2]
    member pwwn 20:00:00:25:b5:ab:30:22 init
    !           [ORARAC4-NVMe-HBA4]
    member pwwn 20:00:00:25:b5:ab:30:24 init
    !           [ORARAC4-NVMe-HBA6]
    member pwwn 20:00:00:25:b5:ab:30:26 init
```

```
    !           [ORARAC4-NVMe-HBA8]
    member pwwn 52:4a:93:7a:7e:04:85:06 target
    !           [PureFAXL170-ORA21c-CT0-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:16 target
    !           [PureFAXL170-ORA21c-CT1-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:80 target
    !           [PureFAXL170-ORA21c-CT0-FC32]
    member pwwn 52:4a:93:7a:7e:04:85:90 target
    !           [PureFAXL170-ORA21c-CT1-FC32]

zone name ORARAC-5-Boot-A vsan 151
    member pwwn 20:00:00:25:b5:ab:30:28 init
    !           [ORARAC-5-FC-HBA0]
    member pwwn 52:4a:93:7a:7e:04:85:04 target
    !           [PureFAXL170-ORA21c-CT0-FC04]
    member pwwn 52:4a:93:7a:7e:04:85:14 target
    !           [PureFAXL170-ORA21c-CT1-FC04]

zone name ORARAC-6-Boot-A vsan 151
    member pwwn 20:00:00:25:b5:ab:30:32 init
    !           [ORARAC-6-FC-HBA0]
    member pwwn 52:4a:93:7a:7e:04:85:04 target
    !           [PureFAXL170-ORA21c-CT0-FC04]
    member pwwn 52:4a:93:7a:7e:04:85:14 target
    !           [PureFAXL170-ORA21c-CT1-FC04]

zone name ORARAC-7-Boot-A vsan 151
    member pwwn 20:00:00:25:b5:ab:30:3c init
    !           [ORARAC-7-FC-HBA0]
```

```
    member pwwn 52:4a:93:7a:7e:04:85:04 target
    !           [PureFAXL170-ORA21c-CT0-FC04]
    member pwwn 52:4a:93:7a:7e:04:85:14 target
    !           [PureFAXL170-ORA21c-CT1-FC04]

zone name ORARAC-8-Boot-A vsan 151
    member pwwn 20:00:00:25:b5:ab:30:46 init
    !           [ORARAC-8-FC-HBA0]
    member pwwn 52:4a:93:7a:7e:04:85:04 target
    !           [PureFAXL170-ORA21c-CT0-FC04]
    member pwwn 52:4a:93:7a:7e:04:85:14 target
    !           [PureFAXL170-ORA21c-CT1-FC04]

zone name ORARAC-5-NVMe-A1 vsan 151
    member pwwn 20:00:00:25:b5:ab:30:2a init
    !           [ORARAC5-NVMe-HBA2]
    member pwwn 20:00:00:25:b5:ab:30:2c init
    !           [ORARAC5-NVMe-HBA4]
    member pwwn 20:00:00:25:b5:ab:30:2e init
    !           [ORARAC5-NVMe-HBA6]
    member pwwn 20:00:00:25:b5:ab:30:30 init
    !           [ORARAC5-NVMe-HBA8]
    member pwwn 52:4a:93:7a:7e:04:85:06 target
    !           [PureFAXL170-ORA21c-CT0-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:16 target
    !           [PureFAXL170-ORA21c-CT1-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:80 target
    !           [PureFAXL170-ORA21c-CT0-FC32]
    member pwwn 52:4a:93:7a:7e:04:85:90 target
```

```
    !           [PureFAXL170-ORA21c-CT1-FC32]


zone name ORARAC-6-NVMe-A1 vsan 151
    member pwwn 20:00:00:25:b5:ab:30:34 init
    !           [ORARAC6-NVMe-HBA2]
    member pwwn 20:00:00:25:b5:ab:30:36 init
    !           [ORARAC6-NVMe-HBA4]
    member pwwn 20:00:00:25:b5:ab:30:38 init
    !           [ORARAC6-NVMe-HBA6]
    member pwwn 20:00:00:25:b5:ab:30:3a init
    !           [ORARAC6-NVMe-HBA8]
    member pwwn 52:4a:93:7a:7e:04:85:06 target
    !           [PureFAXL170-ORA21c-CT0-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:16 target
    !           [PureFAXL170-ORA21c-CT1-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:80 target
    !           [PureFAXL170-ORA21c-CT0-FC32]
    member pwwn 52:4a:93:7a:7e:04:85:90 target
    !           [PureFAXL170-ORA21c-CT1-FC32]


zone name ORARAC-7-NVMe-A1 vsan 151
    member pwwn 20:00:00:25:b5:ab:30:3e init
    !           [ORARAC7-NVMe-HBA2]
    member pwwn 20:00:00:25:b5:ab:30:40 init
    !           [ORARAC7-NVMe-HBA4]
    member pwwn 20:00:00:25:b5:ab:30:42 init
    !           [ORARAC7-NVMe-HBA6]
    member pwwn 20:00:00:25:b5:ab:30:44 init
    !           [ORARAC7-NVMe-HBA8]
```

```
    member pwwn 52:4a:93:7a:7e:04:85:06 target
    !           [PureFAXL170-ORA21c-CT0-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:16 target
    !           [PureFAXL170-ORA21c-CT1-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:80 target
    !           [PureFAXL170-ORA21c-CT0-FC32]
    member pwwn 52:4a:93:7a:7e:04:85:90 target
    !           [PureFAXL170-ORA21c-CT1-FC32]


zone name ORARAC-8-NVMe-A1 vsan 151
    member pwwn 20:00:00:25:b5:ab:30:48 init
    !           [ORARAC8-NVMe-HBA2]
    member pwwn 20:00:00:25:b5:ab:30:4a init
    !           [ORARAC8-NVMe-HBA4]
    member pwwn 20:00:00:25:b5:ab:30:4c init
    !           [ORARAC8-NVMe-HBA6]
    member pwwn 20:00:00:25:b5:ab:30:4e init
    !           [ORARAC8-NVMe-HBA8]
    member pwwn 52:4a:93:7a:7e:04:85:06 target
    !           [PureFAXL170-ORA21c-CT0-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:16 target
    !           [PureFAXL170-ORA21c-CT1-FC06]
    member pwwn 52:4a:93:7a:7e:04:85:80 target
    !           [PureFAXL170-ORA21c-CT0-FC32]
    member pwwn 52:4a:93:7a:7e:04:85:90 target
    !           [PureFAXL170-ORA21c-CT1-FC32]


interface mgmt0
  ip address 10.29.135.57 255.255.255.0
```

```
interface port-channel41
  switchport trunk allowed vsan 151
  switchport description Port-Channel-FI-A-MDS-A
  switchport rate-mode dedicated
  switchport trunk mode off

vsan database
  vsan 151 interface port-channel41
  vsan 151 interface fc1/9
  vsan 151 interface fc1/10
  vsan 151 interface fc1/11
  vsan 151 interface fc1/12
  vsan 151 interface fc1/13
  vsan 151 interface fc1/14
  vsan 151 interface fc1/15
  vsan 151 interface fc1/16
  vsan 151 interface fc1/17
  vsan 151 interface fc1/18
  vsan 151 interface fc1/19
  vsan 151 interface fc1/20
  vsan 151 interface fc1/21
  vsan 151 interface fc1/22
  vsan 151 interface fc1/23
  vsan 151 interface fc1/24
switchname FS-ORA-MDS-A
cli alias name autozone source sys/autozone.py
line console
line vty
```

```
boot kickstart bootflash:/m9100-s6ek9-kickstart-mz.9.3.2.bin
boot system bootflash:/m9100-s6ek9-mz.9.3.2.bin

interface fc1/1
  switchport description ORA21C-FI-A-1/35/1
  switchport trunk mode off
  port-license acquire
  channel-group 41 force
  no shutdown

interface fc1/2
  switchport description ORA21C-FI-A-1/35/2
  switchport trunk mode off
  port-license acquire
  channel-group 41 force
  no shutdown

interface fc1/3
  switchport description ORA21C-FI-A-1/35/3
  switchport trunk mode off
  port-license acquire
  channel-group 41 force
  no shutdown

interface fc1/4
  switchport description ORA21C-FI-A-1/35/4
  switchport trunk mode off
  port-license acquire
  channel-group 41 force
```

```
  no shutdown

interface fc1/5
  switchport description ORA21C-FI-A-1/35/5
  switchport trunk mode off
  port-license acquire
  channel-group 41 force
  no shutdown

interface fc1/6
  switchport description ORA21C-FI-A-1/35/6
  switchport trunk mode off
  port-license acquire
  channel-group 41 force
  no shutdown

interface fc1/7
  switchport description ORA21C-FI-A-1/35/7
  switchport trunk mode off
  port-license acquire
  channel-group 41 force
  no shutdown

interface fc1/8
  switchport description ORA21C-FI-A-1/35/8
  switchport trunk mode off
  port-license acquire
  channel-group 41 force
  no shutdown
```

```
interface fc1/17
  switchport trunk allowed vsan 151
  switchport description PureFAXL170-ORA21c-CT0.FC4
  switchport trunk mode off
  port-license acquire
  no shutdown

interface fc1/18
  switchport trunk allowed vsan 151
  switchport description PureFAXL170-ORA21c-CT1.FC4
  switchport trunk mode off
  port-license acquire
  no shutdown

interface fc1/19
  switchport trunk allowed vsan 151
  switchport description PureFAXL170-ORA21c-CT0.FC6
  switchport trunk mode off
  port-license acquire
  no shutdown

interface fc1/20
  switchport trunk allowed vsan 151
  switchport description PureFAXL170-ORA21c-CT1.FC6
  switchport trunk mode off
  port-license acquire
  no shutdown
```

```
interface fc1/21
  switchport trunk allowed vsan 151
  switchport description PureFAXL170-ORA21c-CT0.FC32
  switchport trunk mode off
  port-license acquire
  no shutdown

interface fc1/22
  switchport trunk allowed vsan 151
  switchport description PureFAXL170-ORA21c-CT1.FC32
  switchport trunk mode off
  port-license acquire
  no shutdown

ip default-gateway 10.29.135.1
```

## Cisco Nexus Switch Configuration

```
FS-ORA-N9K-A# show running-config

!Command: show running-config
!No configuration change since last restart
!Time: Sat Jul 13 01:01:08 2024

version 9.3(7) Bios:version 05.45
switchname FS-ORA-N9K-A
policy-map type network-qos jumbo
  class type network-qos class-default
```

```
    mtu 9216
vdc FS-ORA-N9K-A id 1
  limit-resource vlan minimum 16 maximum 4094
  limit-resource vrf minimum 2 maximum 4096
  limit-resource port-channel minimum 0 maximum 511
  limit-resource u4route-mem minimum 248 maximum 248
  limit-resource u6route-mem minimum 96 maximum 96
  limit-resource m4route-mem minimum 58 maximum 58
  limit-resource m6route-mem minimum 8 maximum 8

cfs eth distribute
feature interface-vlan
feature hsrp
feature lacp
feature vpc
feature lldp
ip domain-lookup
system default switchport
system qos
  service-policy type network-qos jumbo
ntp server 72.163.32.44 use-vrf default

vlan 1,10,135
vlan 10
  name Oracle_RAC_Private_Traffic
vlan 135
  name Oracle_RAC_Public_Traffic

spanning-tree port type edge bpduguard default
```

```
spanning-tree port type network default
vrf context management
  ip route 0.0.0.0/0 10.29.135.1
port-channel load-balance src-dst l4port
vpc domain 1
  peer-keepalive destination 10.29.135.56 source 10.29.135.55
  auto-recovery


interface Vlan1

interface port-channel1
  description vPC peer-link
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  spanning-tree port type network
  vpc peer-link

interface port-channel51
  description connect to FS-ORA-FI-A
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  spanning-tree port type edge trunk
  mtu 9216
  vpc 51

interface port-channel52
  description connect to FS-ORA-FI-B
  switchport mode trunk
```

```
  switchport trunk allowed vlan 1,10,135
  spanning-tree port type edge trunk
  mtu 9216
  vpc 52

interface Ethernet1/1
  description Peer link connected to FS-ORA-N9K-B-Eth-1/1
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  channel-group 1 mode active

interface Ethernet1/2
  description Peer link connected to FS-ORA-N9K-B-Eth-1/2
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  channel-group 1 mode active

interface Ethernet1/3
  description Peer link connected to FS-ORA-N9K-B-Eth-1/3
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  channel-group 1 mode active

interface Ethernet1/4
  description Peer link connected to FS-ORA-N9K-B-Eth-1/4
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  channel-group 1 mode active
```

```
interface Ethernet1/9
  description Fabric-Interconnect-A-27
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  spanning-tree port type edge trunk
  mtu 9216
  channel-group 51 mode active

interface Ethernet1/10
  description Fabric-Interconnect-A-28
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  spanning-tree port type edge trunk
  mtu 9216
  channel-group 51 mode active

interface Ethernet1/11
  description Fabric-Interconnect-B-27
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  spanning-tree port type edge trunk
  mtu 9216
  channel-group 52 mode active

interface Ethernet1/12
  description Fabric-Interconnect-B-28
  switchport mode trunk
  switchport trunk allowed vlan 1,10,135
  spanning-tree port type edge trunk
```

```
  mtu 9216
  channel-group 52 mode active
interface Ethernet1/29
  description To-Management-Uplink-Switch
  switchport access vlan 135
  speed 1000

interface mgmt0
  vrf member management
  ip address 10.29.135.55/24
icam monitor scale

line console
line vty
boot nxos bootflash:/nxos.9.3.7.bin
no system default switchport shutdown
```

## Multipath Configuration “/etc/multipath.conf”

```
[root@orarac1 ~]# cat /etc/multipath.conf
defaults {
        polling_interval       10
}
devices {
    device {
        vendor                      "NVME"
        product                     "Pure Storage FlashArray"
        path_selector               "queue-length 0"
        path_grouping_policy        group_by_prio
```

```
        prio                    ana
        failback                immediate
        fast_io_fail_tmo        10
        user_friendly_names     no
        no_path_retry           0
        features                0
        dev_loss_tmo            60
    }
    device {
        vendor                  "PURE"
        product                 "FlashArray"
        path_selector           "service-time 0"
        hardware_handler        "1 alua"
        path_grouping_policy    group_by_prio
        prio                    alua
        failback                immediate
        path_checker            tur
        fast_io_fail_tmo        10
        user_friendly_names     no
        no_path_retry           0
        features                0
        dev_loss_tmo            600
    }
}
multipaths {
        multipath {
                wwid            3624a93704a5561942d7640ea00011436
                alias           ORARAC1-RHEL-OS
        }
```

```
multipath {
        wwid          eui.004a5561942d764024a937ea00011f4e
        alias         ocrvote1
}
multipath {
        wwid          eui.004a5561942d764024a937ea00011f4f
        alias         ocrvote2
}
multipath {
        wwid          eui.004a5561942d764024a937ea00011f50
        alias         slobdata01
}
multipath {
        wwid          eui.004a5561942d764024a937ea00011f51
        alias         slobdata02
}
multipath {
        wwid          eui.004a5561942d764024a937ea00011f52
        alias         slobdata03
}
multipath {
        wwid          eui.004a5561942d764024a937ea00011f53
        alias         slobdata04
}
multipath {
        wwid          eui.004a5561942d764024a937ea00011f54
        alias         slobdata05
}
multipath {
```

```
        wwid            eui.004a5561942d764024a937ea00011f55
        alias           slobdata06
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011f56
        alias           slobdata07
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011f57
        alias           slobdata08
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011f58
        alias           sloblog01
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011f59
        alias           sloblog02
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011fca
        alias           soedata01
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011fcb
        alias           soedata02
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011fcc
```

```
        alias         soedata03
}
multipath {
        wwid          eui.004a5561942d764024a937ea00011fcd
        alias         soedata04
}
multipath {
        wwid          eui.004a5561942d764024a937ea00011fce
        alias         soedata05
}
multipath {
        wwid          eui.004a5561942d764024a937ea00011fcf
        alias         soedata06
}
multipath {
        wwid          eui.004a5561942d764024a937ea00011fd0
        alias         soedata07
}
multipath {
        wwid          eui.004a5561942d764024a937ea00011fd1
        alias         soedata08
}
multipath {
        wwid          eui.004a5561942d764024a937ea00011fd2
        alias         soelog01
}
multipath {
        wwid          eui.004a5561942d764024a937ea00011fd3
        alias         soelog02
```

```
}
multipath {
        wwid            eui.004a5561942d764024a937ea0001206e
        alias           engdata01
}
multipath {
        wwid            eui.004a5561942d764024a937ea0001206f
        alias           engdata02
}
multipath {
        wwid            eui.004a5561942d764024a937ea00012070
        alias           engdata03
}
multipath {
        wwid            eui.004a5561942d764024a937ea00012071
        alias           engdata04
}
multipath {
        wwid            eui.004a5561942d764024a937ea00012072
        alias           engdata05
}
multipath {
        wwid            eui.004a5561942d764024a937ea00012073
        alias           engdata06
}
multipath {
        wwid            eui.004a5561942d764024a937ea00012074
        alias           engdata07
}
```

```
multipath {
        wwid            eui.004a5561942d764024a937ea00012075
        alias           engdata08
}
multipath {
        wwid            eui.004a5561942d764024a937ea00012076
        alias           englog01
}
multipath {
        wwid            eui.004a5561942d764024a937ea00012077
        alias           englog02
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011fd4
        alias           shdata01
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011fd5
        alias           shdata02
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011fd6
        alias           shdata03
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011fd7
        alias           shdata04
}
multipath {
```

```
        wwid            eui.004a5561942d764024a937ea00011fd8
        alias           shdata05
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011fd9
        alias           shdata06
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011fda
        alias           shdata07
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011fdb
        alias           shdata08
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011fdc
        alias           shlog01
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011fdd
        alias           shlog02
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011ece
        alias           fiovol51
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011ecf
```

```
        alias         fiovol52
}
multipath {
        wwid          eui.004a5561942d764024a937ea00011ed0
        alias         fiovol53
}
multipath {
        wwid          eui.004a5561942d764024a937ea00011ed1
        alias         fiovol54
}
multipath {
        wwid          eui.004a5561942d764024a937ea00011ed2
        alias         fiovol55
}
multipath {
        wwid          eui.004a5561942d764024a937ea00011ed3
        alias         fiovol56
}
multipath {
        wwid          eui.004a5561942d764024a937ea00011ed4
        alias         fiovol57
}
multipath {
        wwid          eui.004a5561942d764024a937ea00011ed5
        alias         fiovol58
}
multipath {
        wwid          eui.004a5561942d764024a937ea00011ed6
        alias         fiovol61
```

```
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011ed7
        alias           fiovol62
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011ed8
        alias           fiovol63
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011ed9
        alias           fiovol64
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011eda
        alias           fiovol65
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011edb
        alias           fiovol66
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011edc
        alias           fiovol67
}
multipath {
        wwid            eui.004a5561942d764024a937ea00011edd
        alias           fiovol68
}
```

```
}

[root@orarac1 ~]#
```

## Configure “/etc/udev/rules.d/99-pure-storage.rules”

```
[root@orarac1 ~]# cat /etc/udev/rules.d/99-pure-storage.rules

# Recommended settings for Pure Storage FlashArray.

# Use none scheduler for high-performance solid-state storage for SCSI devices

ACTION=="add|change", KERNEL=="sd*[!0-9]", SUBSYSTEM=="block", ENV{ID_VENDOR}=="PURE",
ATTR{queue/scheduler}="none"

ACTION=="add|change", KERNEL=="dm-[0-9]*", SUBSYSTEM=="block", ENV{DM_NAME}=="3624a937*",
ATTR{queue/scheduler}="none"



# Reduce CPU overhead due to entropy collection

ACTION=="add|change", KERNEL=="sd*[!0-9]", SUBSYSTEM=="block", ENV{ID_VENDOR}=="PURE",
ATTR{queue/add_random}="0"

ACTION=="add|change", KERNEL=="dm-[0-9]*", SUBSYSTEM=="block", ENV{DM_NAME}=="3624a937*",
ATTR{queue/add_random}="0"



# Spread CPU load by redirecting completions to originating CPU

ACTION=="add|change", KERNEL=="sd*[!0-9]", SUBSYSTEM=="block", ENV{ID_VENDOR}=="PURE",
ATTR{queue/rq_affinity}="2"

ACTION=="add|change", KERNEL=="dm-[0-9]*", SUBSYSTEM=="block", ENV{DM_NAME}=="3624a937*",
ATTR{queue/rq_affinity}="2"



# Set the HBA timeout to 60 seconds

ACTION=="add|change", KERNEL=="sd*[!0-9]", SUBSYSTEM=="block", ENV{ID_VENDOR}=="PURE",
ATTR{device/timeout}="60"
```

## Configure “/etc/udev/rules.d/99-oracleasm.rules”

```
[root@orarac1 ~]# cat /etc/udev/rules.d/99-oracleasm.rules

#All volumes which starts with ocrvote* #

ENV{DM_NAME}=="ocrvote*", OWNER:="grid", GROUP:="oinstall", MODE:="660"



#All volumes which starts with dg_oradata_* #
```

```
ENV{DM_NAME}=="*data*", OWNER:="oracle", GROUP:="oinstall", MODE:="660"


#All volumes which starts with dg_oraredo_* #
ENV{DM_NAME}=="*log*", OWNER:="oracle", GROUP:="oinstall", MODE:="660"
```

## Configure “sysctl.conf”

```
[root@orarac1 ~]# cat /etc/sysctl.conf
vm.nr_hugepages=120000

# oracle-database-preinstall-21c setting for fs.file-max is 6815744
fs.file-max = 6815744

# oracle-database-preinstall-21c setting for kernel.sem is '250 32000 100 128'
kernel.sem = 250 32000 100 128

# oracle-database-preinstall-21c setting for kernel.shmmni is 4096
kernel.shmmni = 4096

# oracle-database-preinstall-21c setting for kernel.shmall is 1073741824 on x86_64
kernel.shmall = 1073741824

# oracle-database-preinstall-21c setting for kernel.shmmax is 4398046511104 on x86_64
kernel.shmmax = 4398046511104

# oracle-database-preinstall-21c setting for kernel.panic_on_oops is 1 per Orabug 19212317
kernel.panic_on_oops = 1

# oracle-database-preinstall-21c setting for net.core.rmem_default is 262144
net.core.rmem_default = 262144

# oracle-database-preinstall-21c setting for net.core.rmem_max is 4194304
net.core.rmem_max = 4194304

# oracle-database-preinstall-21c setting for net.core.wmem_default is 262144
net.core.wmem_default = 262144

# oracle-database-preinstall-21c setting for net.core.wmem_max is 1048576
net.core.wmem_max = 1048576
```

```
# oracle-database-preinstall-21c setting for net.ipv4.conf.all.rp_filter is 2
net.ipv4.conf.all.rp_filter = 2

# oracle-database-preinstall-21c setting for net.ipv4.conf.default.rp_filter is 2
net.ipv4.conf.default.rp_filter = 2

# oracle-database-preinstall-21c setting for fs.aio-max-nr is 1048576
fs.aio-max-nr = 1048576

# oracle-database-preinstall-21c setting for net.ipv4.ip_local_port_range is 9000 65500
net.ipv4.ip_local_port_range = 9000 65500
```

## Configure “oracle-database-preinstall-21c.conf”

```
[root@orarac1 ~]# cat /etc/security/limits.d/oracle-database-preinstall-21c.conf

# oracle-database-preinstall-21c setting for nofile soft limit is 1024
oracle   soft   nofile    1024

# oracle-database-preinstall-21c setting for nofile hard limit is 65536
oracle   hard   nofile    65536

# oracle-database-preinstall-21c setting for nproc soft limit is 16384
# refer orabug15971421 for more info.
oracle   soft   nproc    16384

# oracle-database-preinstall-21c setting for nproc hard limit is 16384
oracle   hard   nproc    16384

# oracle-database-preinstall-21c setting for stack soft limit is 10240KB
oracle   soft   stack    10240

# oracle-database-preinstall-21c setting for stack hard limit is 32768KB
oracle   hard   stack    32768

# oracle-database-preinstall-21c setting for memlock hard limit is maximum of 128GB on x86_64
or 3GB on x86 OR 90 % of RAM
oracle   hard   memlock    474753608

# oracle-database-preinstall-21c setting for memlock soft limit is maximum of 128GB on x86_64
or 3GB on x86 OR 90% of RAM
```

```
oracle   soft   memlock    474753608

# oracle-database-preinstall-21c setting for data soft limit is 'unlimited'
oracle   soft   data    unlimited

# oracle-database-preinstall-21c setting for data hard limit is 'unlimited'
oracle   hard   data    unlimited
```

## Feedback

For comments and suggestions about this guide and related guides, join the discussion on Cisco Community at https://cs.co/en-cvds.

## CVD Program

ALL DESIGNS, SPECIFICATIONS, STATEMENTS, INFORMATION, AND RECOMMENDATIONS (COLLECTIVELY, "DESIGNS") IN THIS MANUAL ARE PRESENTED "AS IS," WITH ALL FAULTS. CISCO AND ITS SUPPLIERS DISCLAIM ALL WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE. IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THE DESIGNS, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

THE DESIGNS ARE SUBJECT TO CHANGE WITHOUT NOTICE. USERS ARE SOLELY RESPONSIBLE FOR THEIR APPLICATION OF THE DESIGNS. THE DESIGNS DO NOT CONSTITUTE THE TECHNICAL OR OTHER PROFESSIONAL ADVICE OF CISCO, ITS SUPPLIERS OR PARTNERS. USERS SHOULD CONSULT THEIR OWN TECHNICAL ADVISORS BEFORE IMPLEMENTING THE DESIGNS. RESULTS MAY VARY DEPENDING ON FACTORS NOT TESTED BY CISCO.

**Americas Headquarters**
Cisco Systems, Inc.
San Jose, CA

**Asia Pacific Headquarters**
Cisco Systems (USA) Pte. Ltd.
Singapore

**Europe Headquarters**
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at **https://www.cisco.com/go/offices**.