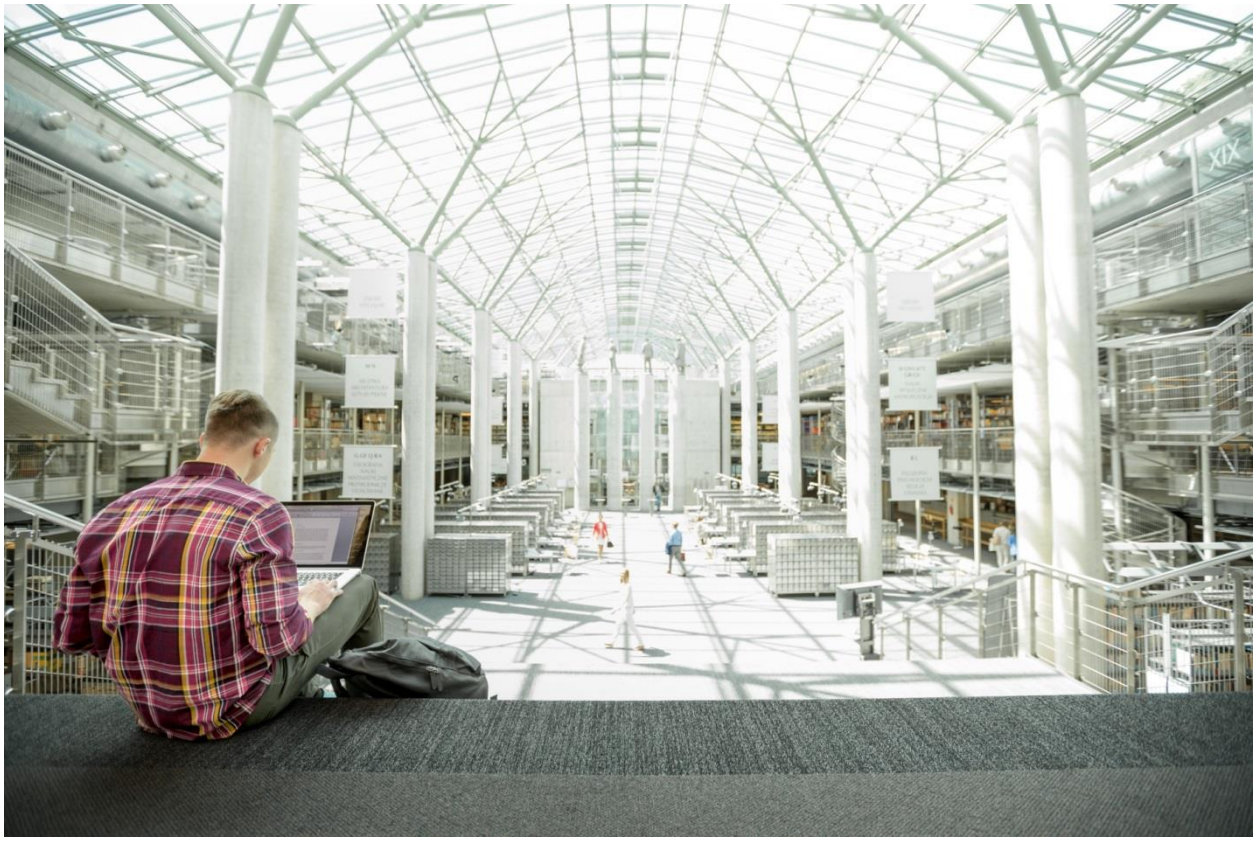


FlashStack Datacenter with VMware Horizon 7.4 and VMware vSphere 6.5 U1 with Cisco UCS Manager 3.2 for 6000 Seats

Cisco Validated Design for a 6000 Seat Virtual Desktop Infrastructure Built on Cisco UCS B200 M5 and Cisco UCS Manager 3.2 with Pure Storage //X70 Array, VMware Horizon 7.4 and VMware vSphere 6.5 U1 Hypervisor Platform

Last Updated: July 25, 2018

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, visit:

http://www.cisco.com/go/designzone.

Table of Contents

Executive Summary

Cisco Validated Designs include systems and solutions that are designed, tested, and documented to facilitate and improve customer deployments. These designs incorporate a wide range of technologies and products into a portfolio of solutions that have been developed to address the business needs of customers. Cisco, Pure and VMware have partner to deliver this document, which serves as a specific step-by-step guide for implementing this solution. This Cisco Validated Design provides an efficient architectural design that is based on customer requirements. The solution that follows is a validated approach for deploying Cisco, Pure and VMware technologies as a shared, high performance, resilient, virtual desktop infrastructure.

This document provides a reference architecture and design guide for up to a 6000 seat mixed workload end user computing environment on FlashStack Datacenter with Cisco UCS and Pure Storage® Flash Array //X70 with 100% NVMe flash modules. The solution includes VMware Horizon server-based Remote Desktop Windows Sever 2016 Hosted sessions, VMware Horizon persistent Microsoft Windows 10 virtual desktops and VMware Horizon non-persistent Microsoft Windows 10 virtual desktops on VMware vSphere 6.5.

The solution is a predesigned, best-practice data center architecture built on the FlashStack reference architecture. The FlashStack Datacenter used in this validation includes Cisco Unified Computing System (UCS), the Cisco Nexus® 9000 family of switches, Cisco MDS 9000 family of Fibre Channel (FC) switches and Pure All-NVMe //X70 system.

This solution is 100 percent virtualized on fifth generation Cisco UCS B200 M5 blade servers, booting VMware vSphere 6.5 Update 1 through FC SAN from the //X70 storage array. The virtual desktop sessions are powered by VMware Horizon 7.4. VMware Horizon Remote Desktop Server Hosted Sessions (2430 RDS Server sessions) and 1190 VMware Horizon Instant-Clone, 1190 VMware Horizon Linked-Clone and 1190 Full Clones Window 10 desktops (3570 virtual desktops) were provisioned on the Pure Storage //X70 storage array. Where applicable the document provides best practice recommendations and sizing guidelines for customer deployment of this solution.

This solution delivers the design for 6000 user payload with 6 fewer blade servers than previous 6000 seat solutions on fourth generation Cisco UCS Blade Servers making it more efficient and cost effective in the data center due to increased solution density. Further rack efficiencies were gained from a storage standpoint as all 6000 users were hosted on a single 3U //X70 storage array while previous large-scale FlashStack Cisco Validated Designs with VDI used a Pure Storage 3U base chassis along with a 2U expansion shelf.

The solution provides outstanding virtual desktop end user experience as measured by the Login VSI 4.1.32 Knowledge Worker workload running in benchmark mode.

The 6000-seat solution provides a large-scale building block that can be replicated to confidently scale out to tens of thousands of users.

Solution Overview

Introduction

The current industry trend in data center design is towards shared infrastructures. By using virtualization along with pre-validated IT platforms, enterprise customers have embarked on the journey to the cloud by moving away from application silos and toward shared infrastructure that can be quickly deployed, thereby increasing agility and reducing costs. Cisco, Pure storage, and VMware have partnered to deliver this Cisco Validated Design, which uses best of breed storage, server and network components to serve as the foundation for desktop virtualization workloads, enabling efficient architectural designs that can be quickly and confidently deployed.

Audience

The audience for this document includes, but is not limited to; sales engineers, field consultants, professional services, IT managers, partner engineers, and customers who want to take advantage of an infrastructure built to deliver IT efficiency and enable IT innovation.

Purpose of this Document

This document provides a step-by-step design, configuration and implementation guide for the Cisco Validated Design for a large-scale VMware Horizon 7 mixed workload solution with Pure Storage //X70 array, Cisco UCS Blade Servers, Cisco Nexus 9000 series Ethernet switches and Cisco MDS 9000 series Fibre channel switches.

What's New?

This is the first VMware Horizon desktop virtualization Cisco Validated Design with Cisco UCS 5th generation servers and Pure X-Series system.

It incorporates the following features:

- Cisco UCS B200 M5 blade servers with Intel Xeon Scalable Family processor and 2666 MHz memory
- Validation of Cisco Nexus 9000 with Pure Storage //X70 system
- Validation of Cisco MDS 9000 with Pure Storage //X70 system
- Support for the Cisco UCS 3.2(2f) release and Cisco UCS B200-M5 servers
- Support for the latest release of Pure Storage //X70 hardware and Purity//FA v5.0.2
- A Fibre Channel storage design supporting SAN LUNs
- Cisco UCS Inband KVM Access
- Cisco UCS vMedia client for vSphere Installation

- Cisco UCS Firmware Auto Sync Server policy
- VMware vSphere 6.5 U1 Hypervisor
- VMware Horizon 7 RDS Hosted server sessions on Windows Server 2016
- VMware Horizon 7 non-persistent Instant-Clone and Linked-Clone Windows 10 virtual machines
- VMware Horizon 7 persistent Full Clones Windows 10 virtual machines

The data center market segment is shifting toward heavily virtualized private, hybrid and public cloud computing models running on industry-standard systems. These environments require uniform design points that can be repeated for ease of management and scalability.

These factors have led to the need for predesigned computing, networking and storage building blocks optimized to lower the initial design cost, simplify management, and enable horizontal scalability and high levels of utilization.

The use cases include:

- Enterprise Data Center
- Service Provider Data Center
- Large Commercial Data Center

Solution Summary

This Cisco Validated Design prescribes a defined set of hardware and software that serves as an integrated foundation for both VMware Horizon RDSH server desktop sessions based on Microsoft Server 2016, VMware Horizon VDI persistent virtual machines and VMware Horizon VDI non–persistent virtual machines based on Windows 10 operating system.

The mixed workload solution includes Pure Storage FlashArray //X70®, Cisco Nexus® and MDS networking, the Cisco Unified Computing System (Cisco UCS®), VMware Horizon® and VMware vSphere® software in a single package. The design is space optimized such that the network, compute, and storage required can be housed in one data center rack. Switch port density enables the networking components to accommodate multiple compute and storage configurations of this kind.

The infrastructure is deployed to provide Fibre Channel-booted hosts with block-level access to shared storage. 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.

The combination of technologies from Cisco Systems, Inc., Pure Storage Inc. and VMware Inc. produced a highly efficient, robust and affordable desktop virtualization solution for a hosted virtual desktop and hosted shared desktop mixed deployment supporting different use cases. Key components of this solution include the following:

- More power, same size. Cisco UCS B200 M5 half-width blade with dual 18-core 2.3 GHz Intel ® Xeon ® Scalable Family Gold (6140) processors and 768 GB of memory for VMware Horizon Desktop hosts supports more virtual desktop workloads than the previously released generation processors on the same hardware. The Intel 18-core 2.3 GHz Intel ® Xeon ® Gold Scalable Family (6140) processors used in this study provided a balance between increased per-blade capacity and cost.
- Fewer servers. Because of the increased compute power in the Cisco UCS B200 M5 servers, we supported the 6000 seat design with 16 percent fewer servers compared to previous generation Cisco UCS B200 M4s.
- Fault-tolerance with high availability built into the design. The various designs are based on using one Unified Computing System chassis with multiple Cisco UCS B200 M5 blades for virtualized desktop and infrastructure workloads. The design provides N+1 server fault tolerance for hosted virtual desktops, hosted shared desktops and infrastructure services.
- Stress-tested to the limits during aggressive boot scenario. The 6000-user mixed RDS hosted virtual sessions and VDI pooled shared desktop environment booted and registered with the VMware Horizon 7 Administrator in under 20 minutes, providing our customers with an extremely fast, reliable cold-start desktop virtualization system.
- Stress-tested to the limits during simulated login storms. All 6000 simulated users logged in and started running workloads up to steady state in 48-minutes without overwhelming the processors, exhausting memory or exhausting the storage subsystems, providing customers with a desktop virtualization system that can easily handle the most demanding login and startup storms.
- Ultra-condensed computing for the datacenter. The rack space required to support the system is less than a single 42U rack, conserving valuable data center floor space.

- All Virtualized: This Cisco Validated Design (CVD) presents a validated design that is 100 percent virtualized on VMware ESXi 6.5. All of the virtual desktops, user data, profiles, and supporting infrastructure components, including Active Directory, Provisioning Servers, SQL Servers, VMware Horizon Connection Servers, VMware Horizon Composer Server, VMware Horizon Replica Servers, VMware Horizon Remote Desktop Server Hosted sessions and VDI virtual machine desktops. This provides customers with complete flexibility for maintenance and capacity additions because the entire system runs on the FlashStack converged infrastructure with stateless Cisco UCS Blade servers and Pure FC storage.
- Cisco maintains industry leadership with the new Cisco UCS Manager 3.2(2f) software that simplifies scaling, guarantees consistency, and eases maintenance. Cisco's ongoing development efforts with Cisco UCS Manager, Cisco UCS Central, and Cisco UCS Director insure that customer environments are consistent locally, across Cisco UCS Domains and across the globe, our software suite offers increasingly simplified operational and deployment management and it continues to widen the span of control for customer organizations' subject matter experts in compute, storage and network.
- Our 40G unified fabric story gets additional validation on Cisco UCS 6300 Series Fabric Interconnects as Cisco runs more challenging workload testing, while maintaining unsurpassed user response times.
- Pure All-NVMe //X70 storage array provides industry-leading storage solutions that efficiently handle the most demanding I/O bursts (for example, login storms), profile management, and user data management, deliver simple and flexible business continuance, and help reduce storage cost per desktop.
- Pure All-NVMe //X70 storage array provides a simple to understand storage architecture for hosting all user data components (VMs, profiles, user data) on the same storage array.
- Pure Storage software enables to seamlessly add, upgrade or remove storage from the infrastructure to meet the needs of the virtual desktops.
- Pure Storage Management UI for VMware vSphere hypervisor has deep integrations with vSphere, providing easy-button automation for key storage tasks such as storage repository provisioning, storage resize, data deduplication, directly from vCenter.
- VMware Horizon 7. Latest and greatest virtual desktop and application product. VMware Horizon 7 follows a new unified product architecture that supports both hosted-shared desktops and applications (RDS) and complete virtual desktops (VDI). This new VMware Horizon release simplifies tasks associated with large-scale VDI management. This modular solution supports seamless delivery of Windows apps and desktops as the number of users increase. In addition, Horizon enhancements help to optimize performance and improve the user experience across a variety of endpoint device types, from workstations to mobile devices including laptops, tablets, and smartphones.
- Optimized to achieve the best possible performance and scale. For hosted shared desktop sessions, the best performance was achieved when the number of vCPUs assigned to the VMware 7 RDS virtual machines did not exceed the number of hyper-threaded (logical) cores available on the server. In other words, maximum performance is obtained when not overcommitting the CPU resources for the virtual machines running virtualized RDS systems.

- Provisioning desktop machines made easy. Remote Desktop Server Hosted (RDSH) shared virtual machines and VMware Horizon 7, Microsoft Windows 10 virtual machines were created for this solution using VMware Instant and Composer pooled desktops.

Cisco Desktop Virtualization Solutions: Data Center

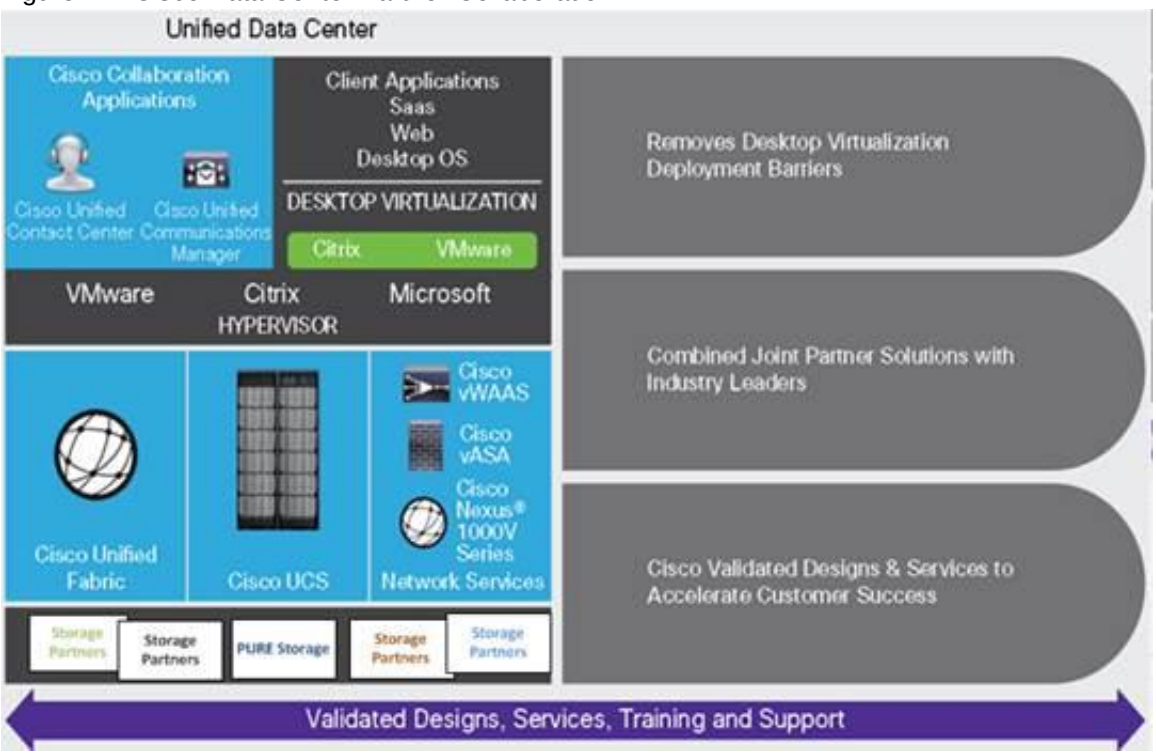
The Evolving Workplace

Today's IT departments are facing a rapidly evolving workplace environment. The workforce is becoming increasingly diverse and geographically dispersed, including offshore contractors, distributed call center operations, knowledge and task workers, partners, consultants, and executives connecting from locations around the world at all times.

This workforce is also increasingly mobile, conducting business in traditional offices, conference rooms across the enterprise campus, home offices, on the road, in hotels, and at the local coffee shop. This workforce wants to use a growing array of client computing and mobile devices that they can choose based on personal preference. These trends are increasing pressure on IT to ensure protection of corporate data and prevent data leakage or loss through any combination of user, endpoint device, and desktop access scenarios (Figure 1).

These challenges are compounded by desktop refresh cycles to accommodate aging PCs and bounded local storage and migration to new operating systems, specifically Microsoft Windows 10 and productivity tools, specifically Microsoft Office 2016.

Figure 1 Cisco Data Center Partner Collaboration



Some of the key drivers for desktop virtualization are increased data security and reduced TCO through increased control and reduced management costs.

Cisco Desktop Virtualization Focus

Cisco focuses on three key elements to deliver the best desktop virtualization data center infrastructure: simplification, security, and scalability. The software combined with platform modularity provides a simplified, secure, and scalable desktop virtualization platform.

Simplified

Cisco UCS provides a radical new approach to industry-standard computing and provides the core of the data center infrastructure for desktop virtualization. Among the many features and benefits of Cisco UCS are the drastic reduction in the number of servers needed and in the number of cables used per server, and the capability to rapidly deploy or re-provision servers through Cisco UCS service profiles. With fewer servers and cables to manage and with streamlined server and virtual desktop provisioning, operations are significantly simplified. Thousands of desktops can be provisioned in minutes with Cisco UCS Manager Service Profiles and Cisco storage partners' storage-based cloning. This approach accelerates the time to productivity for end users, improves business agility, and allows IT resources to be allocated to other tasks.

Cisco UCS Manager automates many mundane, error-prone data center operations such as configuration and provisioning of server, network, and storage access infrastructure. In addition, Cisco UCS B-Series Blade Servers and C-Series Rack Servers with large memory footprints enable high desktop density that helps reduce server infrastructure requirements.

Simplification also leads to more successful desktop virtualization implementation. Cisco and its technology partners like VMware Technologies and Pure Storage have developed integrated, validated architectures, including predefined converged architecture infrastructure packages such as FlashStack. Cisco Desktop Virtualization Solutions have been tested with VMware vSphere, VMware Horizon.

Secure

Although virtual desktops are inherently more secure than their physical predecessors, they introduce new security challenges. Mission-critical web and application servers using a common infrastructure such as virtual desktops are now at a higher risk for security threats. Inter–virtual machine traffic now poses an important security consideration that IT managers need to address, especially in dynamic environments in which virtual machines, using VMware vMotion, move across the server infrastructure.

Desktop virtualization, therefore, significantly increases the need for virtual machine–level awareness of policy and security, especially given the dynamic and fluid nature of virtual machine mobility across an extended computing infrastructure. The ease with which new virtual desktops can proliferate magnifies the importance of a virtualization-aware network and security infrastructure. Cisco data center infrastructure (Cisco UCS and Cisco Nexus Family solutions) for desktop virtualization provides strong data center, network, and desktop security, with comprehensive security from the desktop to the hypervisor. Security is enhanced with segmentation of virtual desktops, virtual machine–aware policies and administration, and network security across the LAN and WAN infrastructure.

Scalable

Growth of a desktop virtualization solution is all but inevitable, so a solution must be able to scale, and scale predictably, with that growth. The Cisco Desktop Virtualization Solutions built on FlashStack Datacenter infrastructure supports high virtual-desktop density (desktops per server), and additional servers and

storage scale with near-linear performance. FlashStack Datacenter provides a flexible platform for growth and improves business agility. Cisco UCS Manager Service Profiles allow on-demand desktop provisioning and make it just as easy to deploy dozens of desktops as it is to deploy thousands of desktops.

Cisco UCS servers provide near-linear performance and scale. Cisco UCS implements the patented Cisco Extended Memory Technology to offer large memory footprints with fewer sockets (with scalability to up to 1 terabyte (TB) of memory with 2- and 4-socket servers). Using unified fabric technology as a building block, Cisco UCS server aggregate bandwidth can scale to up to 80 Gbps per server, and the northbound Cisco UCS fabric interconnect can output 2 terabits per second (Tbps) at line rate, helping prevent desktop virtualization I/O and memory bottlenecks. Cisco UCS, with its high-performance, low-latency unified fabric-based networking architecture, supports high volumes of virtual desktop traffic, including high-resolution video and communications traffic. In addition, Cisco storage partners Pure help maintain data availability and optimal performance during boot and login storms as part of the Cisco Desktop Virtualization Solutions. Recent Cisco Validated Designs for End User Computing based on FlashStack solutions have demonstrated scalability and performance, with up to 6000 desktops up and running in 20 minutes.

FlashStack Datacenter provides an excellent platform for growth, with transparent scaling of server, network, and storage resources to support desktop virtualization, data center applications, and cloud computing.

Savings and Success

The simplified, secure, scalable Cisco data center infrastructure for desktop virtualization solutions saves time and money compared to alternative approaches. Cisco UCS enables faster payback and ongoing savings (better ROI and lower TCO) and provides the industry's greatest virtual desktop density per server, reducing both capital expenditures (CapEx) and operating expenses (OpEx). The Cisco UCS architecture and Cisco Unified Fabric also enables much lower network infrastructure costs, with fewer cables per server and fewer ports required. In addition, storage tiering and deduplication technologies decrease storage costs, reducing desktop storage needs by up to 50 percent.

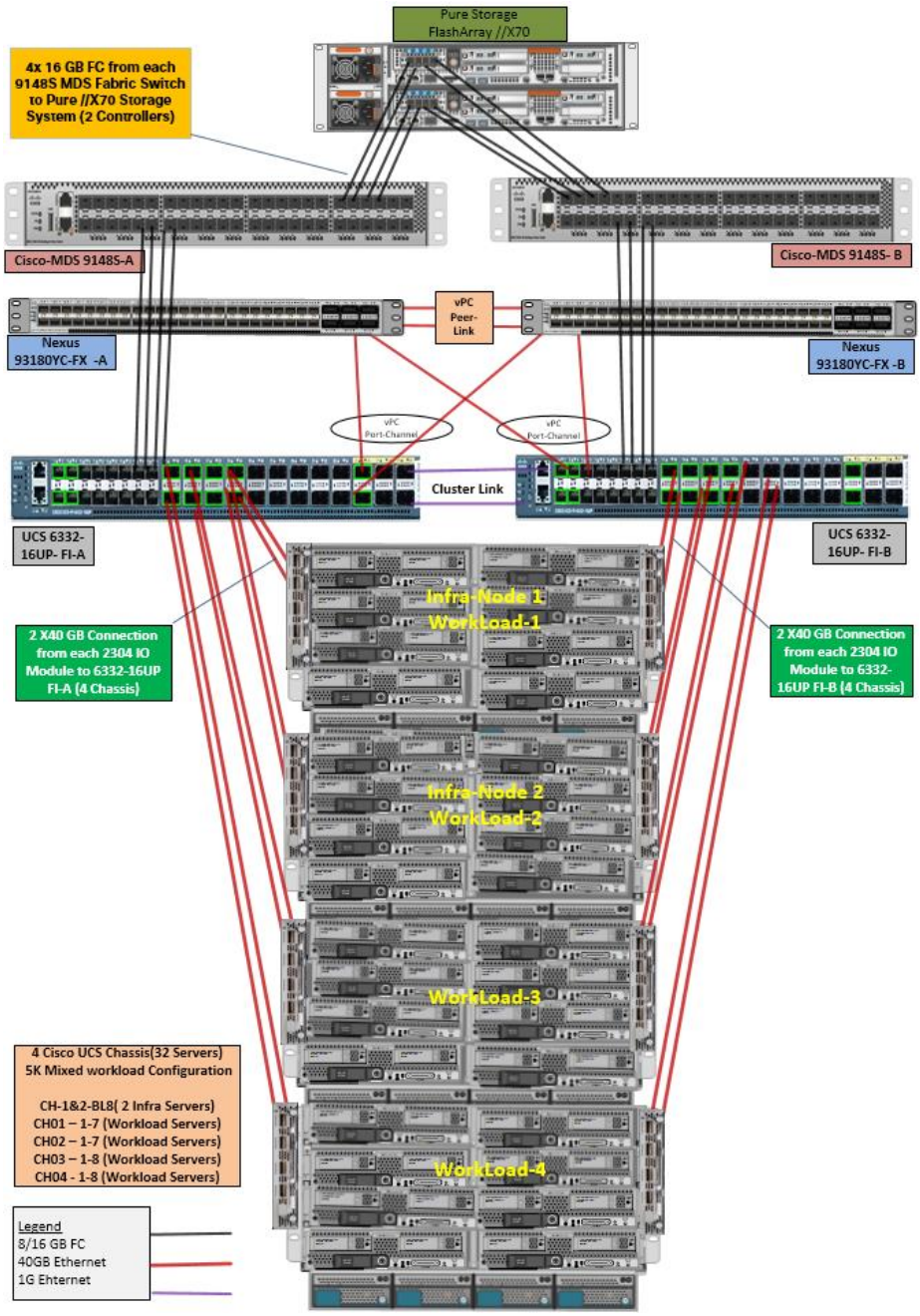
The simplified deployment of Cisco UCS for desktop virtualization accelerates the time to productivity and enhances business agility. IT staff and end users are more productive more quickly, and the business can respond to new opportunities quickly by deploying virtual desktops whenever and wherever they are needed. The high-performance Cisco systems and network deliver a near-native end-user experience, allowing users to be productive anytime and anywhere.

The ultimate measure of desktop virtualization for any organization is its efficiency and effectiveness in both the near term and the long term. The Cisco Desktop Virtualization Solutions are very efficient, allowing rapid deployment, requiring fewer devices and cables, and reducing costs. The solutions are also very effective, providing the services that end users need on their devices of choice while improving IT operations, control, and data security. Success is bolstered through Cisco's best-in-class partnerships with leaders in virtualization and storage, and through tested and validated designs and services to help customers throughout the solution lifecycle. Long-term success is enabled through the use of Cisco's scalable, flexible, and secure architecture as the platform for desktop virtualization.

Physical Topology

Figure 2 illustrates the FlashStack System architecture.

Figure 2 Flash-Stack Solution Reference Architecture



The reference hardware configuration includes:

- Two Cisco Nexus 93180YC-FX switches
- Two Cisco MDS 9148S 16GB Fibre Channel switches
- Two Cisco UCS 6332-16UP Fabric Interconnects
- Four Cisco UCS 5108 Blade Chassis
- Two Cisco UCS B200 M4 Blade Servers (2 Server hosting Infrastructure VMs)

- Thirty Cisco UCS B200 M5 Blade Servers (for workload)
- One Pure Storage FlashArray //X70 with All-NVMe DirectFlash Modules

For desktop virtualization, the deployment includes VMware Horizon 7 running on VMware vSphere 6.5.

The design is intended to provide a large-scale building block for both VMware Horizon RDS Hosted server sessions and Windows 10 non- persistent and persistent VDI desktops in the following ratio:

- 2430 Remote Desktop Server Hosted (RDSH) desktop sessions
- 2380 VMware Horizon Windows 10 non-persistent virtual desktops
- 1190 VMware Horizon Windows 10 persistent virtual desktops

The data provided in this document will allow our customers to adjust the mix of RDSH and VDI desktops to suite their environment. For example, additional blade servers and chassis can be deployed to increase compute capacity, additional 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. This procedure covers everything from physical cabling to network, compute and storage device configurations.

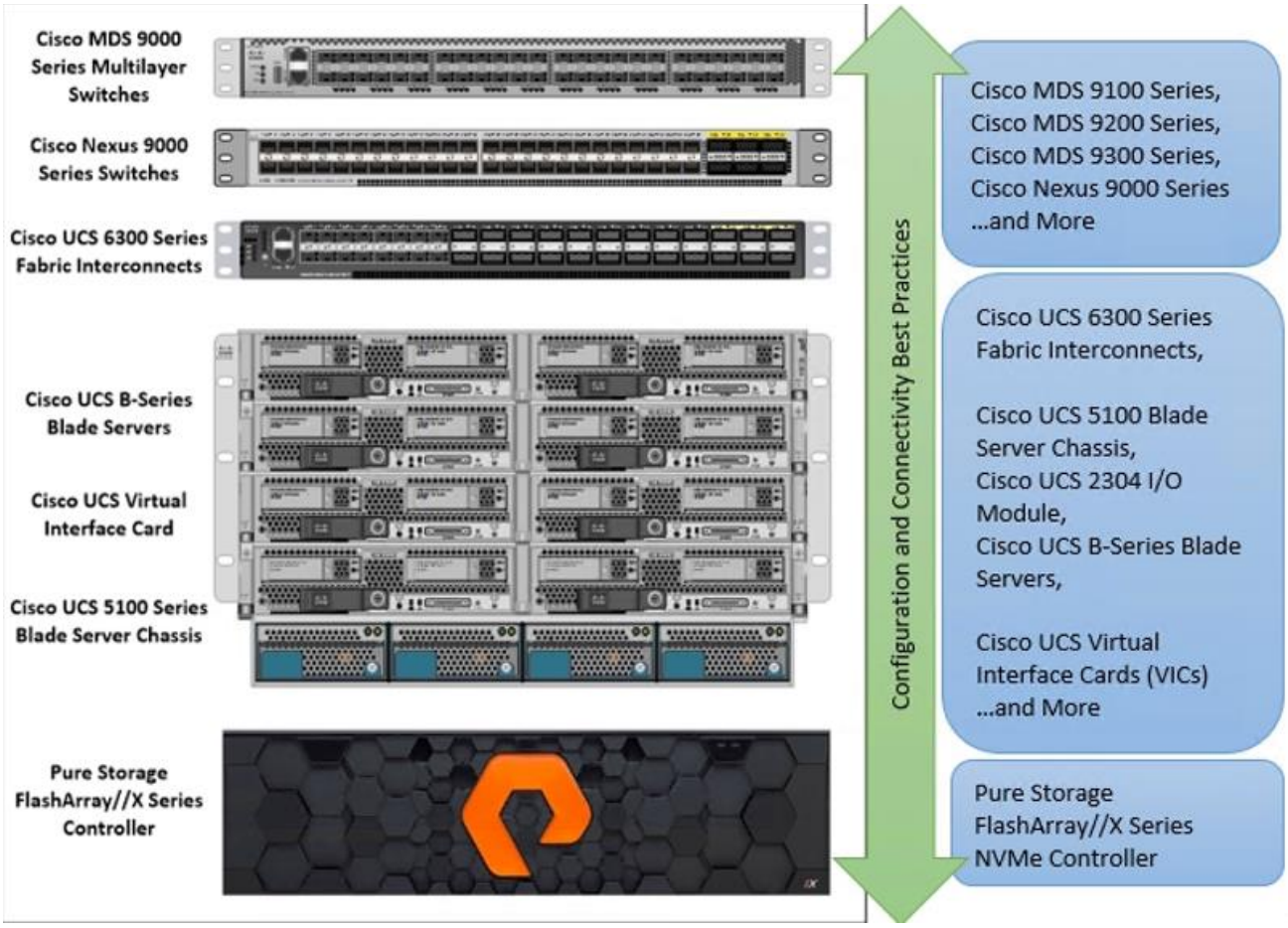
What is FlashStack?

The FlashStack platform, developed by Cisco and Pure Storage, is a flexible, integrated infrastructure solution that delivers pre-validated storage, networking, and server technologies. 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 is a best practice datacenter architecture that includes the following components:

- Cisco Unified Computing System
- Cisco Nexus Switches
- Cisco MDS Switches
- Pure Storage FlashArray

Figure 3 FlashStack Systems Components



As shown in Figure 3, these components are connected and configured according to best practices of both Cisco and Pure Storage and provide the ideal platform for running a variety of enterprise database workloads with confidence. FlashStack can scale up for greater performance and capacity (adding compute, network, or storage resources individually as needed), or it can scale out for environments that require multiple consistent deployments.

The reference architecture covered in this document leverages the Pure Storage FlashArray//X70 Controller with NVMe based DirectFlash modules for Storage, Cisco UCS B200 M5 Blade Server for Compute, Cisco Nexus 9000 and Cisco MDS 9100 series for the switching element and Cisco Fabric Interconnects 6300 series for System Management. As shown in Figure 3, FlashStack Architecture can maintain consistency at scale. Each of the component families shown in (Cisco UCS, Cisco Nexus, Cisco MDS, Cisco FI and Pure Storage) offers platform and resource options to scale the infrastructure up or down, while supporting the same features and functionality that are required under the configuration and connectivity best practices of FlashStack.

FlashStack Solution Benefits

FlashStack provides a jointly supported solution by Cisco and Pure Storage. Bringing a carefully validated architecture built on superior compute, world-class networking, and the leading innovations in all flash storage. The portfolio of validated offerings from FlashStack includes but is not limited to the following:

- Consistent Performance and Scalability
  - Consistent sub-millisecond latency with 100 percent NVMe enterprise flash storage
  - Consolidate hundreds of enterprise-class applications in a single rack
  - Scalability through a design for hundreds of discrete servers and thousands of virtual machines, and the capability to scale I/O bandwidth to match demand without disruption
  - Repeatable growth through multiple FlashStack CI deployments
- Operational Simplicity
  - Fully tested, validated, and documented for rapid deployment
  - Reduced management complexity
  - No storage tuning or tiers necessary
  - 3x better data reduction without any performance impact
- Lowest TCO
  - Dramatic savings in power, cooling and space with Cisco UCS and 100 percent Flash
  - Industry leading data reduction
  - Free FlashArray controller upgrades every three years with Forever Flash™
- Mission Critical and Enterprise Grade Resiliency
  - Highly available architecture with no single point of failure
  - Non-disruptive operations with no downtime
  - Upgrade and expand without downtime or performance loss
  - Native data protection: snapshots and replication

Cisco and Pure Storage have also built a robust and experienced support team focused on FlashStack solutions, from customer account and technical sales representatives to professional services and technical support engineers. The support alliance between Pure Storage and Cisco gives customers and channel services partners direct access to technical experts who collaborate with cross vendors and have access to shared lab resources to resolve potential issues.

What's New in this FlashStack Release

This CVD of the FlashStack release introduces new hardware with the Pure Storage FlashArray//X, that is 100 percent NVMe enterprise class all-flash array along with Cisco UCS B200 M5 Blade Servers featuring the Intel Xeon Scalable Family of CPUs. This is the second Oracle RAC Database deployment Cisco Validated Design with Pure Storage. It incorporates the following features:

- Pure Storage FlashArray //X

- Cisco UCS B200 M5 Blade Servers
- VMware vSphere 6.5 U1 and VMware Horizon 7.4

Configuration Guidelines

This Cisco Validated Design provides details for deploying a fully redundant, highly available 6000 seat mixed workload virtual desktop solution with VMware on a FlashStack Datacenter architecture. Configuration guidelines are provided that refer the reader to which redundant component is being configured with each step. For example, storage controller 01and storage controller 02 are used to identify the two Pure Storage FlashArray //X70 controllers that are provisioned with this document, Cisco Nexus A or Cisco Nexus B identifies the pair of Cisco Nexus switches that are configured and Cisco MDS A or Cisco MDS B identifies the pair of Cisco MDS switches that are configured. The pair of Cisco UCS 6332-16UP Fabric Interconnects are similarly configured as FI-A and FI-B.

Additionally, this document details the steps for provisioning multiple Cisco UCS hosts, and these are identified sequentially: VM-Host-Infra-01, VM-Host-Infra-02, VM-Host-RDSH-01, VM-Host-VDI-01 and so on. Finally, to indicate that you should include information pertinent to your environment in a given step, <text> appears as part of the command structure.

Solution Components

This section describes the components used in the solution outlined in this solution.

Cisco Unified Computing System

Cisco UCS Manager provides unified, embedded management of all software and hardware components of the Cisco Unified Computing System™ (Cisco UCS) through an intuitive GUI, a command-line interface (CLI), and an XML API. The manager provides a unified management domain with centralized management capabilities and can control multiple chassis and thousands of virtual machines.

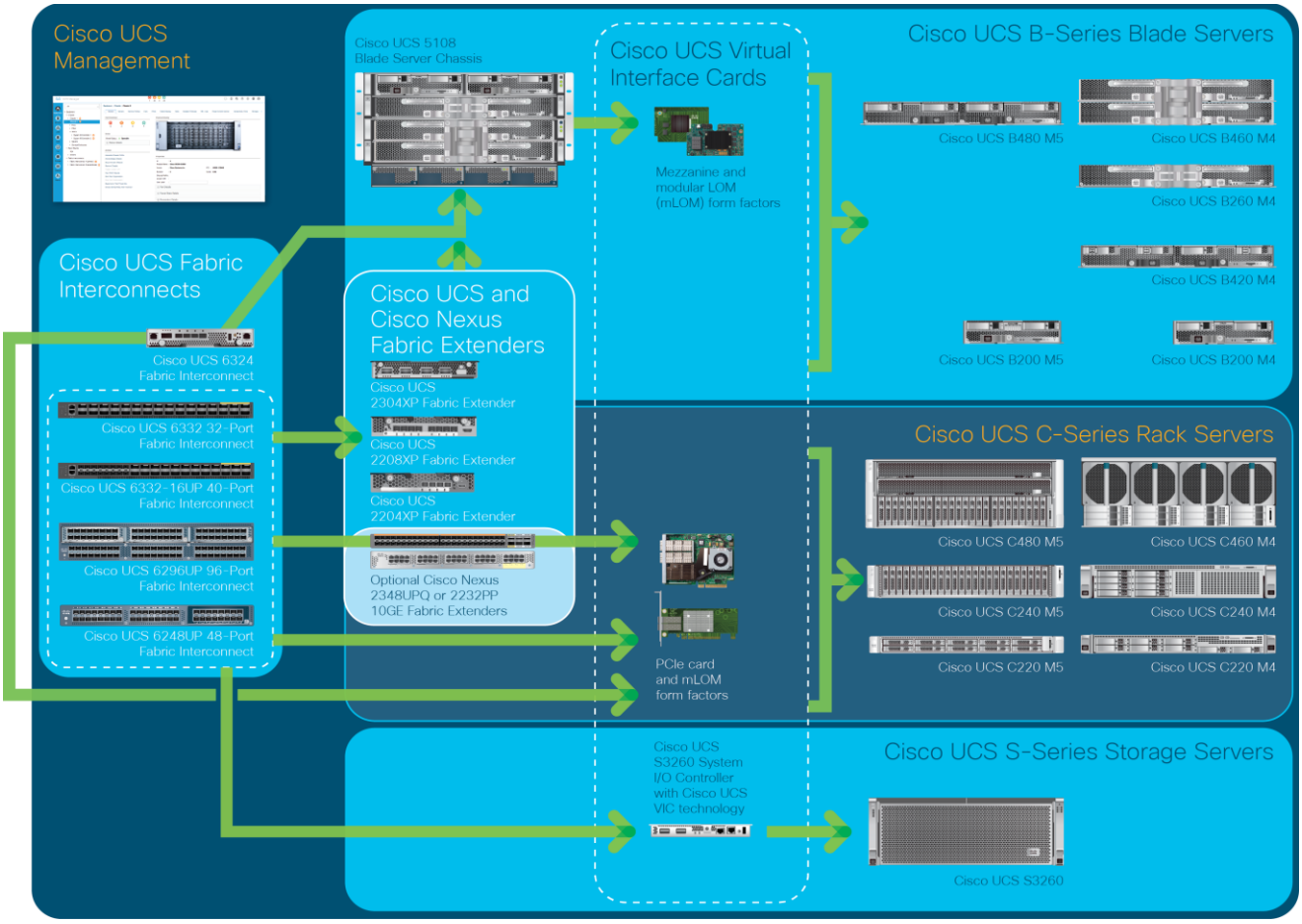
Cisco UCS is a next-generation data center platform that unites computing, networking, and storage access. The platform, optimized for virtual environments, is designed using open industry-standard technologies and aims to reduce total cost of ownership (TCO) and increase business agility. The system integrates a low-latency; lossless 40 Gigabit Ethernet unified network fabric with enterprise-class, x86-architecture servers. It is an integrated, scalable, multi-chassis platform in which all resources participate in a unified management domain.

Cisco Unified Computing System Components

The main components of Cisco UCS are:

- Compute: The system is based on an entirely new class of computing system that incorporates blade servers based on Intel® Xeon® Scalable Family processors.
- Network: The system is integrated on a low-latency, lossless, 40-Gbps unified network fabric. This network foundation consolidates LANs, SANs, and high-performance computing (HPC) networks, which are separate networks today. The unified fabric lowers costs by reducing the number of network adapters, switches, and cables needed, and by decreasing the power and cooling requirements.
- Virtualization: The system unleashes the full potential of virtualization by enhancing the scalability, performance, and operational control of virtual environments. Cisco security, policy enforcement, and diagnostic features are now extended into virtualized environments to better support changing business and IT requirements.
- Storage access: The system provides consolidated access to local storage, SAN storage, and network-attached storage (NAS) over the unified fabric. With storage access unified, Cisco UCS can access storage over Ethernet, Fibre Channel, Fibre Channel over Ethernet (FCoE), and Small Computer System Interface over IP (iSCSI) protocols. This capability provides customers with choice for storage access and investment protection. In addition, server administrators can pre-assign storage-access policies for system connectivity to storage resources, simplifying storage connectivity and management and helping increase productivity.
- Management: Cisco UCS uniquely integrates all system components, enabling the entire solution to be managed as a single entity by Cisco UCS Manager. The manager has an intuitive GUI, a CLI, and a robust API for managing all system configuration processes and operations.

Figure 4 Cisco Data Center Overview



Cisco UCS is designed to deliver:

- Reduced TCO and increased business agility
- Increased IT staff productivity through just-in-time provisioning and mobility support
- A cohesive, integrated system that unifies the technology in the data center; the system is managed, serviced, and tested as a whole
- Scalability through a design for hundreds of discrete servers and thousands of virtual machines and the capability to scale I/O bandwidth to match demand
- Industry standards supported by a partner ecosystem of industry leaders

Cisco UCS Manager provides unified, embedded management of all software and hardware components of the Cisco Unified Computing System across multiple chassis, rack servers, and thousands of virtual machines. Cisco UCS Manager manages Cisco UCS as a single entity through an intuitive GUI, a command-line interface (CLI), or an XML API for comprehensive access to all Cisco UCS Manager Functions.

Cisco UCS Fabric Interconnect

The Cisco UCS 6300 Series Fabric Interconnects are a core part of Cisco UCS, providing both network connectivity and management capabilities for the system. The Cisco UCS 6300 Series offers line-rate, low-latency, lossless 40 Gigabit Ethernet, FCoE, and Fibre Channel functions.

The fabric interconnects provide the management and communication backbone for the Cisco UCS B-Series Blade Servers and Cisco UCS 5100 Series Blade Server Chassis. All chassis, and therefore all blades, attached to the fabric interconnects become part of a single, highly available management domain. In addition, by supporting unified fabric, the Cisco UCS 6300 Series provides both LAN and SAN connectivity for all blades in the domain.

For networking, the Cisco UCS 6300 Series uses a cut-through architecture, supporting deterministic, low-latency, line-rate 40 Gigabit Ethernet on all ports, 2.4 plus terabit (Tb) switching capacity, and 320 Gbps of bandwidth per chassis IOM, independent of packet size and enabled services. The product series supports Cisco low-latency, lossless, 40 Gigabit Ethernet unified network fabric capabilities, increasing the reliability, efficiency, and scalability of Ethernet networks. The fabric interconnects support multiple traffic classes over a lossless Ethernet fabric, from the blade server through the interconnect. Significant TCO savings come from an FCoE-optimized server design in which network interface cards (NICs), host bus adapters (HBAs), cables, and switches can be consolidated.

Figure 5 Cisco UCS 6300 Series Fabric Interconnect – 6332-16UP

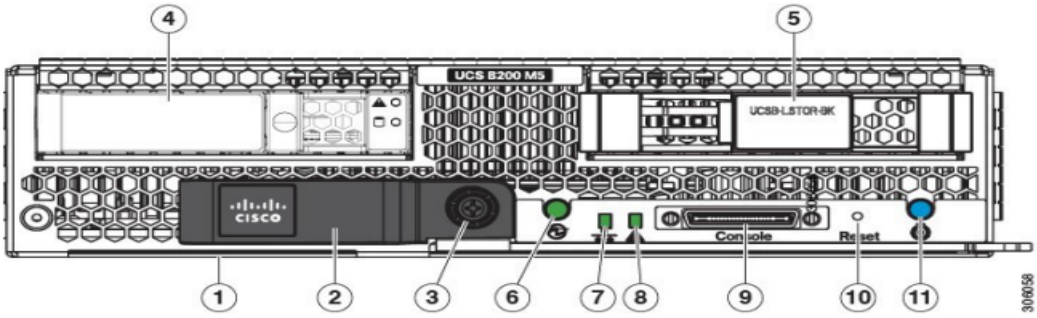
Cisco UCS B200 M5 Blade Server

The Cisco UCS B200 M5 Blade Server (Figure 6 and 0) is a density-optimized, half-width blade server that supports two CPU sockets for Intel Xeon processor 6140 Gold series CPUs and up to 24 DDR4 DIMMs. It supports one modular LAN-on-motherboard (LOM) dedicated slot for a Cisco virtual interface card (VIC) and one mezzanine adapter. In additions, the Cisco UCS B200 M5 supports an optional storage module that accommodates up to two SAS or SATA hard disk drives (HDDs) or solid-state disk (SSD) drives. You can install up to eight Cisco UCS B200 M5 servers in a chassis, mixing them with other models of Cisco UCS blade servers in the chassis if desired. Latest features of Cisco UCS Virtual Interface Cards (VICs)

Figure 6 Cisco UCS B200 M5 Front View


UCS B200 M3
cisco
Console
Reset

Figure 7 Cisco UCS B200 M5 Back View



1	Asset pull tag Each server has a plastic tag that pulls out of the front panel. The tag contains the server serial number as well as the product ID (PID) and version ID (VID). The tag also allows you to add your own asset tracking label without interfering with the intended air flow.	7	Network link status
2	Blade ejector handle	8	Blade health LED
3	Ejector captive screw	9	Console connector[1]
4	Drive bay 1	10	Reset button access
5	Drive bay 2	11	Locater button and LED
6	Power button and LED		

Notes:

1. A KVM I/O Cable plugs into the console connector, it can be ordered as a spare. The KVM I/O Cable in included with every Cisco UCS 5100 Series blade server chassis accessory kit

Cisco UCS combines Cisco UCS B-Series Blade Servers and C-Series Rack Servers with networking and storage access into a single converged system with simplified management, greater cost efficiency and agility, and increased visibility and control. The Cisco UCS B200 M5 Blade Server is one of the newest servers in the Cisco UCS portfolio.

The Cisco UCS B200 M5 delivers performance, flexibility, and optimization for data centers and remote sites. This enterprise-class server offers market-leading performance, versatility, and density without compromise for workloads ranging from web infrastructure to distributed databases. The Cisco UCS B200 M5 can quickly deploy stateless physical and virtual workloads with the programmable ease of use of the Cisco UCS Manager software and simplified server access with Cisco® Single Connect technology. Based on the Intel Xeon processor 6140 Gold product family, it offers up to 3 TB of memory using 128GB DIMMs, up to two disk drives, and up to 320 Gbps of I/O throughput. The Cisco UCS B200 M5 offers exceptional levels of performance, flexibility, and I/O throughput to run your most demanding applications.

In addition, Cisco UCS has the architectural advantage of not having to power and cool excess switches, NICs, and HBAs in each blade server chassis. With a larger power budget per blade server, it provides uncompromised expandability and capabilities, as in the new Cisco UCS B200 M5 server with its leading memory-slot capacity and drive capacity.

The Cisco UCS B200 M5 provides:

- Latest Intel® Xeon® Scalable processors with up to 28 cores per socket
- Up to 24 DDR4 DIMMs for improved performance

- Intel 3D XPoint-ready support, with built-in support for next-generation nonvolatile memory technology
- Two GPUs
- Two Small-Form-Factor (SFF) drives
- Two Secure Digital (SD) cards or M.2 SATA drives
- Up to 80 Gbps of I/O throughput

Main Features

The Cisco UCS B200 M5 server is a half-width blade. Up to eight servers can reside in the 6-Rack-Unit (6RU) Cisco UCS 5108 Blade Server Chassis, offering one of the highest densities of servers per rack unit of blade chassis in the industry. You can configure the Cisco UCS B200 M5 to meet your local storage requirements without having to buy, power, and cool components that you do not need.

The Cisco UCS B200 M5 provides these main features:

- Up to two Intel Xeon Scalable CPUs with up to 28 cores per CPU
- 24 DIMM slots for industry-standard DDR4 memory at speeds up to 2666 MHz, with up to 3 TB of total memory when using 128-GB DIMMs
- Modular LAN On Motherboard (mLOM) card with Cisco UCS Virtual Interface Card (VIC) 1340, a 2-port, 40 Gigabit Ethernet, Fibre Channel over Ethernet (FCoE)–capable mLOM mezzanine adapter
- Optional rear mezzanine VIC with two 40-Gbps unified I/O ports or two sets of 4 x 10-Gbps unified I/O ports, delivering 80 Gbps to the server; adapts to either 10- or 40-Gbps fabric connections
- Two optional, hot-pluggable, hard-disk drives (HDDs), solid-state drives (SSDs), or NVMe 2.5-inch drives with a choice of enterprise-class RAID or pass-through controllers
- Cisco FlexStorage local drive storage subsystem, which provides flexible boot and local storage capabilities and allows you to boot from dual, mirrored SD cards
- Support for up to two optional GPUs
- Support for up to one rear storage mezzanine card

For more information about Cisco UCS B200 M5, see the Cisco UCS B200 M5 Blade Server Specsheet.

Table 1 Ordering Information

Part Number	Description
UCSB-B200-M5	UCS B200 M5 Blade w/o CPU, mem, HDD, mezz
UCSB-B200-M5-U	UCS B200 M5 Blade w/o CPU, mem, HDD, mezz (UPG)
UCSB-B200-M5-CH	UCS B200 M5 Blade w/o CPU, mem, HDD, mezz, Drive bays, HS

Cisco UCS VIC1340 Converged Network Adapter

The Cisco UCS Virtual Interface Card (VIC) 1340 (Figure 8) is a 2-port 40-Gbps Ethernet or dual 4 x 10-Gbps Ethernet, Fibre Channel over Ethernet (FCoE)-capable modular LAN on motherboard (mLOM) designed exclusively for the M5 generation of Cisco UCS B-Series Blade Servers. When used in combination with an optional port expander, the Cisco UCS VIC 1340 capabilities is enabled for two ports of 40-Gbps Ethernet.

The Cisco UCS VIC 1340 enables a policy-based, stateless, agile server infrastructure that can present over 256 PCIe standards-compliant interfaces to the host that can be dynamically configured as either network interface cards (NICs) or host bus adapters (HBAs). In addition, the Cisco UCS VIC 1340 supports Cisco® Data Center Virtual Machine Fabric Extender (VM-FEX) technology, which extends the Cisco UCS fabric interconnect ports to virtual machines, simplifying server virtualization deployment and management.

Figure 8 Cisco UCS VIC 1340

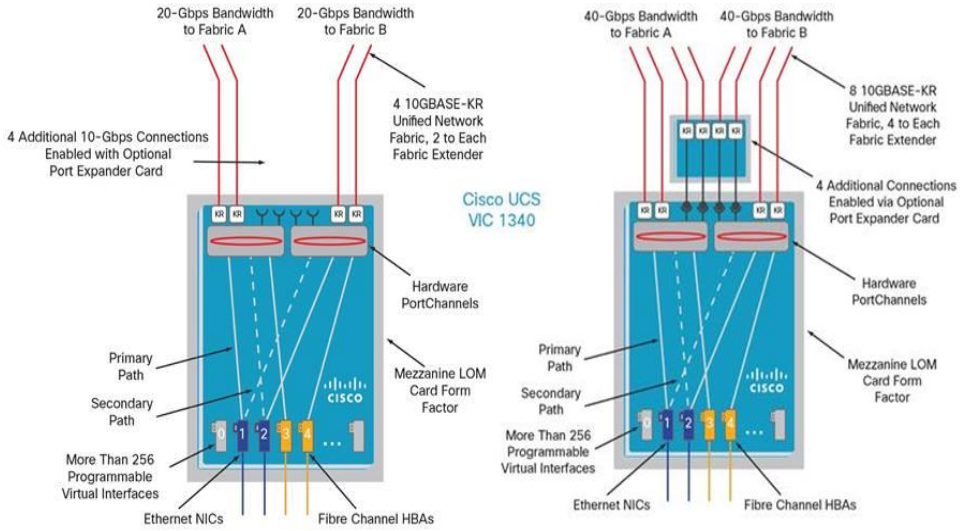


Figure 8 illustrates the Cisco UCS VIC 1340 Virtual Interface Cards Deployed in the Cisco UCS B-Series B200 M5 Blade Servers.

Cisco Switching

Cisco Nexus 93180YC-FX Switches

The 93180YC-EX Switch provides a flexible line-rate Layer 2 and Layer 3 feature set in a compact form factor. Designed with Cisco Cloud Scale technology, it supports highly scalable cloud architectures. With the option to operate in Cisco NX-OS or Application Centric Infrastructure (ACI) mode, it can be deployed across enterprise, service provider, and Web 2.0 data centers.

Architectural Flexibility

- Includes top-of-rack or middle-of-row fiber-based server access connectivity for traditional and leaf-spine architectures
- Leaf node support for Cisco ACI architecture is provided in the roadmap
- Increase scale and simplify management through Cisco Nexus 2000 Fabric Extender support

Feature Rich

- Enhanced Cisco NX-OS Software is designed for performance, resiliency, scalability, manageability, and programmability
- ACI-ready infrastructure helps users take advantage of automated policy-based systems management
- Virtual Extensible LAN (VXLAN) routing provides network services
- Rich traffic flow telemetry with line-rate data collection
- Real-time buffer utilization per port and per queue, for monitoring traffic micro-bursts and application traffic patterns

Highly Available and Efficient Design

- High-density, non-blocking architecture
- Easily deployed into either a hot-aisle and cold-aisle configuration
- Redundant, hot-swappable power supplies and fan trays

Simplified Operations

- Power-On Auto Provisioning (POAP) support allows for simplified software upgrades and configuration file installation
- An intelligent API offers switch management through remote procedure calls (RPCs, JSON, or XML) over a HTTP/HTTPS infrastructure
- Python Scripting for programmatic access to the switch command-line interface (CLI)
- Hot and cold patching, and online diagnostics

Investment Protection

A Cisco 40 Gb bidirectional transceiver allows reuse of an existing 10 Gigabit Ethernet multimode cabling plant for 40 Gigabit Ethernet Support for 1 Gb and 10 Gb access connectivity for data centers migrating access switching infrastructure to faster speed. The following is supported:

- 1.8 Tbps of bandwidth in a 1 RU form factor
- 48 fixed 1/10/25-Gbps SFP+ ports
- 6 fixed 40/100-Gbps QSFP+ for uplink connectivity
- Latency of less than 2 microseconds
- Front-to-back or back-to-front airflow configurations
- 1+1 redundant hot-swappable 80 Plus Platinum-certified power supplies
- Hot swappable 3+1 redundant fan trays

Figure 9 Cisco Nexus 93180YC-EX Switch

Cisco MDS 9148S Fiber Channel Switch

The Cisco MDS 9148S 16G Multilayer Fabric Switch is the next generation of the highly reliable Cisco MDS 9100 Series Switches. It includes up to 48 auto-sensing line-rate 16-Gbps Fibre Channel ports in a compact easy to deploy and manage 1-rack-unit (1RU) form factor. In all, the Cisco MDS 9148S is a powerful and flexible switch that delivers high performance and comprehensive Enterprise-class features at an affordable price.

Cisco MDS 9148S has a pay-as-you-grow model which helps you scale from a 12 port base license to a 48 port with an incremental 12-port license. This helps customers to pay and activate only the required ports.

Cisco MDS 9148S has a dual power supply and FAN trays to provide physical redundancy. The software features, such as ISSU and ISSD, helps with upgrading and downgrading code with-out reloading the switch and without interrupting the live traffic.

Figure 10 Cisco 9148S MDS Fibre Channel Switch

Benefits

- Flexibility for growth and virtualization
- Easy deployment and management
- Optimized bandwidth utilization and reduced downtime

- Enterprise-class features and reliability at low cost

Features

- PowerOn Auto Provisioning and intelligent diagnostics
- In-Service Software Upgrade and dual redundant hot-swappable power supplies for high availability
- Role-based authentication, authorization, and accounting services to support regulatory requirements
- High-performance interswitch links with multipath load balancing
- Smart zoning and virtual output queuing
- Hardware-based slow port detection and recovery

Specifications at-a-Glance

Performance and Port Configuration

- 2/4/8/16-Gbps auto-sensing with 16 Gbps of dedicated bandwidth per port
- Up to 256 buffer credits per group of 4 ports (64 per port default, 253 maximum for a single port in the group)
- Supports configurations of 12, 24, 36, or 48 active ports, with pay-as-you-grow, on-demand licensing

Advanced Functions

- Virtual SAN (VSAN)
- Inter-VSAN Routing (IVR)
- Port Channel with multipath load balancing
- Flow-based and zone-based QoS

Hypervisor and Desktop Broker

This Cisco Validated Design includes VMware vSphere 6.5 and VMware Horizon 7.4

VMware vSphere 6.5

VMware provides virtualization software. VMware's enterprise software hypervisors for servers VMware vSphere ESX, vSphere ESXi, and VSphere–are bare-metal hypervisors that run directly on server hardware without requiring an additional underlying operating system. VMware vCenter Server for vSphere provides central management and complete control and visibility into clusters, hosts, virtual machines, storage, networking, and other critical elements of your virtual infrastructure.

VMware vSphere 6.5 introduces many enhancements to vSphere Hypervisor, VMware virtual machines, vCenter Server, virtual storage, and virtual networking, further extending the core capabilities of the vSphere platform.

Today VMware announced vSphere 6.5, which is one of the most feature rich releases of vSphere in quite some time. The vCenter Server Appliance is taking charge in this release with several new features which we'll cover in this blog article. For starters, the installer has gotten an overhaul with a new modern look and feel. Users of both Linux and Mac will also be ecstatic since the installer is now supported on those platforms along with Microsoft Windows. If that wasn't enough, the vCenter Server Appliance now has features that are exclusive such as:

- Migration
- Improved Appliance Management
- VMware Update Manager
- Native High Availability
- Built-in Backup / Restore

VMware vSphere Client

With VMware vSphere 6.5, a fully supported version of the HTML5-based vSphere Client that will run alongside the vSphere Web Client. The vSphere Client is built into vCenter Server 6.5 (both Windows and Appliance) and is enabled by default. While the HTML-5 based vSphere Client does not have full feature parity, the team has prioritized many of the day-to-day tasks of administrators and continue to seek feedback on items that will enable customers to use it full time. The vSphere Web Client continues to be accessible via "http://<vcenter_fqdn>/vsphere-client" while the vSphere Client is reachable via "http://<vcenter_fqdn>/ui". VMware is periodically updating the vSphere Client outside of the normal vCenter Server release cycle. To make sure it is easy and simple for customers to stay up to date the vSphere Client will be able to be updated without any effects to the rest of vCenter Server.

Some of the benefits of the new vSphere Client:

- Clean, consistent UI built on VMware's new Clarity UI standards (to be adopted across our portfolio)
- Built on HTML5 so it is truly a cross-browser and cross-platform application
- No browser plugins to install/manage
- Integrated into vCenter Server for 6.5 and fully supported
- Fully supports Enhanced Linked Mode
- Users of the Fling have been extremely positive about its performance

VMware ESXi 6.5 Hypervisor

VMware vSphere 6.5 introduces a number of new features in the hypervisor:

Scalability Improvements

- ESXi 6.5 dramatically increases the scalability of the platform. With vSphere Hypervisor 6.0, clusters can scale to as many as 64 hosts, up from 32 in previous releases. With 64 hosts in a cluster, vSphere 6.0 can support 8000 virtual machines in a single cluster. This capability enables greater consolidation ratios, more efficient use of VMware vSphere Distributed Resource Scheduler (DRS), and fewer clusters that must be separately managed. Each vSphere Hypervisor 6.5 instance can support up to

480 logical CPUs, 12 terabytes (TB) of RAM, and 1024 virtual machines. By using the newest hardware advances, ESXi 6.5 enables the virtualization of applications that previously had been thought to be non-virtualizable.

Security Enhancements

- ESXi 6.5 offers these security enhancements:
  - Account management: ESXi 6.5 enables management of local accounts on the ESXi server using new ESXi CLI commands. The capability to add, list, remove, and modify accounts across all hosts in a cluster can be centrally managed using a vCenter Server system. Previously, the account and permission management functions for ESXi hosts were available only for direct host connections. The setup, removal, and listing of local permissions on ESXi servers can also be centrally managed.
  - Account lockout: ESXi Host Advanced System Settings have two new options for the management of failed local account login attempts and account lockout duration. These parameters affect Secure Shell (SSH) and vSphere Web Services connections, but not ESXi direct console user interface (DCUI) or console shell access.
  - Password complexity rules: In previous versions of ESXi, password complexity changes had to be made by manually editing the /etc/pam.d/passwd file on each ESXi host. In vSphere 6.0, an entry in Host Advanced System Settings enables changes to be centrally managed for all hosts in a cluster.
  - Improved auditability of ESXi administrator actions: Prior to vSphere 6.0, actions at the vCenter Server level by a named user appeared in ESXi logs with the vpxuser username: for example, [user=vpxuser]. In vSphere 6.5, all actions at the vCenter Server level for an ESXi server appear in the ESXi logs with the vCenter Server username: for example, [user=vpxuser: DOMAIN\User]. This approach provides a better audit trail for actions run on a vCenter Server instance that conducted corresponding tasks on the ESXi hosts.
  - Flexible lockdown modes: Prior to vSphere 6.5, only one lockdown mode was available. Feedback from customers indicated that this lockdown mode was inflexible in some use cases. With vSphere 6.5, two lockdown modes are available:
    - In normal lockdown mode, DCUI access is not stopped, and users on the DCUI access list can access the DCUI.
    - In strict lockdown mode, the DCUI is stopped.
    - Exception users: vSphere 6.0 offers a new function called exception users. Exception users are local accounts or Microsoft Active Directory accounts with permissions defined locally on the host to which these users have host access. These exception users are not recommended for general user accounts, but they are recommended for use by third-party applications–for service accounts, for example–that need host access when either normal or strict lockdown mode is enabled. Permissions on these accounts should be set to the bare minimum required for the application to perform its task and with an account that needs only read-only permissions on the ESXi host.
  - Smart card authentication to DCUI: This function is for U.S. federal customers only. It enables DCUI login access using a Common Access Card (CAC) and Personal Identity Verification (PIV). The ESXi host must be part of an Active Directory domain.

VMware Horizon Version 7

Enterprise IT organizations are tasked with the challenge of provisioning Microsoft Windows apps and desktops while managing cost, centralizing control, and enforcing corporate security policy. Deploying Windows apps to users in any location, regardless of the device type and available network bandwidth, enables a mobile workforce that can improve productivity. With VMware Horizon 7, IT can effectively control app and desktop provisioning while securing data assets and lowering capital and operating expenses.

VMware Horizon

VMware Horizon desktop virtualization solutions built on a unified architecture so they are simple to manage and flexible enough to meet the needs of all your organization's users. You use the same architecture and management tools to manage public, private, and hybrid cloud deployments as you do for on premises deployments

- VMware Horizon Virtual machines and RDSH known as server-based hosted sessions: These are applications hosted from Microsoft Windows servers to any type of device, including Windows PCs, Macs, smartphones, and tablets. Some VMware editions include technologies that further optimize the experience of using Windows applications on a mobile device by automatically translating native mobile-device display, navigation, and controls to Windows applications; enhancing performance over mobile networks; and enabling developers to optimize any custom Windows application for any mobile environment.
- VMware Horizon RDSH session users also known as server-hosted desktops: These are inexpensive, locked-down Windows virtual desktops hosted from Windows server operating systems. They are well suited for users, such as call center employees, who perform a standard set of tasks.

Advantages of Using VMware Horizon

VMware Horizon 7 provides the following new features and enhancements:

- Instant Clones
  - A new type of desktop virtual machines that can be provisioned significantly faster than the traditional Composer linked clones.
  - A fully functional desktop can be provisioned in two seconds or less.
  - Recreating a desktop pool with a new OS image can be accomplished in a fraction of the time it takes a Composer desktop pool because the parent image can be prepared well ahead of the scheduled time of pool recreation.
  - Clones are automatically rebalanced across available datastores.
  - View storage accelerator is automatically enabled.
- VMware Blast Extreme
  - VMware Blast Extreme is now fully supported on the Horizon platform.
  - Administrators can select the VMware Blast display protocol as the default or available protocol for pools, farms, and entitlements.

  - End users can select the VMware Blast display protocol when connecting to remote desktops and applications.
  - VMware Blast Extreme features include:
    - TCP and UDP transport support
    - H.264 support for the best performance across more devices
    - Reduced device power consumption for longer battery life
    - NVIDIA GRID acceleration for more graphical workloads per server, better performance, and a superior remote user experience
- True SSO
  - For VMware Identity Manager integration, True SSO streamlines the end-to-end login experience. After users log in to VMware Identity Manager using a smart card or an RSA SecurID or RADIUS token, users are not required to also enter Active Directory credentials in order to use a remote desktop or application.
  - Uses a short-lived Horizon virtual certificate to enable a password-free Windows login.
  - Supports using either a native Horizon Client or HTML Access.
  - System health status for True SSO appears in the Horizon Administrator dashboard.
  - Can be used in a single domain, in a single forest with multiple domains, and in a multiple-forest, multiple-domain setup.
- Smart Policies
  - Control of the clipboard cut-and-paste, client drive redirection, USB redirection, and virtual printing desktop features through defined policies.
  - PCoIP session control through PCoIP profiles.
  - Conditional policies based on user location, desktop tagging, pool name, and Horizon Client registry values.
- Configure the clipboard memory size for VMware Blast and PCoIP sessions

  Horizon administrators can configure the server clipboard memory size by setting GPOs for VMware Blast and PCoIP sessions. Horizon Client 4.1 users on Windows, Linux, and Mac OS X systems can configure the client clipboard memory size. The effective memory size is the lesser of the server and client clipboard memory size values.
- VMware Blast network recovery enhancements

  Network recovery is now supported for VMware Blast sessions initiated from iOS, Android, Mac OS X, Linux, and Chrome OS clients. Previously, network recovery was supported only for Windows client sessions. If you lose your network connection unexpectedly during a VMware Blast session, Horizon Client attempts to reconnect to the network and you can continue to use your remote desktop or application. The network recovery feature also supports IP roaming, which means you can resume your VMware Blast session after switching to a WiFi network.
- Configure Horizon Administrator to not remember the login name

Horizon administrators can configure not to display the Remember user name check box and therefore not remember the administrator's login name.

- Allow Mac OS X users to save credentials

  Horizon administrators can configure Connection Server to allow Horizon Client Mac OS X systems to remember a user's user name, password, and domain information. If users choose to have their credentials saved, the credentials are added to the login fields in Horizon Client on subsequent connections.

- Microsoft Windows 10
  - Windows 10 is supported as a desktop guest operating system
  - Horizon Client runs on Windows 10
  - Smart card is supported on Windows 10
  - The Horizon User Profile Migration tool migrates Windows 7, 8/8.1, Server 2008 R2, or Server 2012 R2 user profiles to Windows 10 user profiles.

- RDS Desktops and Hosted Apps
  - View Composer. View Composer and linked clones provide automated and efficient management of RDS server farms.
  - Graphics Support. Existing 3D vDGA and GRID vGPU graphics solutions on VDI desktops have been extended to RDS hosts, enabling graphics-intensive applications to run on RDS desktops and Hosted Apps.
  - Enhanced Load Balancing. A new capability provides load balancing of server farm applications based on memory and CPU resources.
  - One-Way AD Trusts. One-way AD trust domains are now supported. This feature enables environments with limited trust relationships between domains without requiring Horizon Connection Server to be in an external domain.

- Cloud Pod Architecture (CPA) Enhancements
  - Hosted App Support. Support for application remoting allows applications to be launched using global entitlements across a pod federation.
  - HTML Access (Blast) Support. Users can use HTML Access to connect to remote desktops and applications in a Cloud Pod Architecture deployment.

- Access Point Integration
  - Access Point is a hardened Linux-based virtual appliance that protects virtual desktop and application resources to allow secure remote access from the Internet. Access Point provides a new authenticating DMZ gateway to Horizon Connection Server. Smart card support on Access Point is available as a Tech Preview. Security server will continue to be available as an alternative configuration. For more information, see Deploying and Configuring Access Point.

- FIPS
  - Install-time FIPS mode allows customers with high security requirements to deploy Horizon 6.
- Graphics Enhancements
  - AMD vDGA enables vDGA pass-through graphics for AMD graphics hardware.
  - 4K resolution monitors (3840x2160) are supported.
- Horizon Administrator Enhancements
  - Horizon Administrator shows additional licensing information, including license key, named user and concurrent connection user count.
  - Pool creation is streamlined by letting Horizon administrators to clone existing pools.
- Additional Features
  - Support for IPv6 with VMware Blast Extreme on security servers.
  - Horizon Administrator security protection layer. See the VMware Knowledge Base (KB) article 2144303 for more information: https://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2144303
  - Protection against inadvertent pool deletion.
  - RDS per-device licensing improvements.
  - Support for Intel vDGA.
  - Support for AMD Multiuser GPU Using vDGA.
  - More resilient upgrades.
  - Display scaling for Windows Horizon Clients.
  - DPI scaling is supported if it is set at the system level and the scaling level is greater than 100.

What are VMware RDS Hosted Sessions?

The following describes the VMware RDS Hosted Sessions:

- An RDS host is a server computer that hosts applications and desktop sessions for remote access. An RDS host can be a virtual machine or a physical server.
- An RDS host has the Microsoft Remote Desktop Services role, the Microsoft Remote Desktop Session Host service, and Horizon Agent installed. Remote Desktop Services was previously known as Terminal Services. The Remote Desktop Session Host service allows a server to host applications and remote desktop sessions. With Horizon Agent installed on an RDS host, users can connect to applications and desktop sessions by using the display protocol PCoIP or Blast Extreme. Both protocols provide an optimized user experience for the delivery of remote content, including images, audio and video.

- The performance of an RDS host depends on many factors. For information on how to tune the performance of different versions of Windows Server, see http://msdn.microsoft.com/library/windows/hardware/gg463392.aspx.
- Horizon 7 supports at most one desktop session and one application session per user on an RDS host.
- When users submit print jobs concurrently from RDS desktops or applications that are hosted on the same RDS host, the ThinPrint server on the RDS host processes the print requests serially rather than in parallel. This can cause a delay for some users. Note that the print server does not wait for a print job to complete before processing the next one. Print jobs that are sent to different printers will print in parallel.
- If a user launches an application and also an RDS desktop, and both are hosted on the same RDS host, they share the same user profile. If the user launches an application from the desktop, conflicts may result if both applications try to access or modify the same parts of the user profile, and one of the applications may fail to run properly.
- The process of setting up applications or RDS desktops for remote access involves the following tasks:
- Installing Applications
  - If you plan to create application pools, you must install the applications on the RDS hosts. If you want Horizon 7 to automatically display the list of installed applications, you must install the applications so that they are available to all users from the Start menu. You can install an application at any time before you create the application pool. If you plan to manually specify an application, you can install the application at any time, either before or after creating an application pool.
- Important
  - When you install an application, you must install it on all the RDS hosts in a farm and in the same location on each RDS host. If you do not, a health warning will appear on the View Administrator dashboard. In such a situation, if you create an application pool, users might encounter an error when they try to run the application.
  - When you create an application pool, Horizon 7 automatically displays the applications that are available to all users rather than individual users from the Start menu on all of the RDS hosts in a farm. You can choose any applications from that list. In addition, you can manually specify an application that is not available to all users from the Start menu. There is no limit on the number of applications that you can install on an RDS host.

Farms, RDS Hosts, Desktop and Application Pools

With VMware Horizon, you can create desktop and application pools to give users remote access to virtual machine-based desktops, session-based desktops, physical computers, and applications. Horizon takes advantage of Microsoft Remote Desktop Services (RDS) and VMware PC-over-IP (PCoIP) technologies to provide high-quality remote access to users.

- RDS Hosts

  - RDS hosts are server computers that have Windows Remote Desktop Services and View Agent installed. These servers host applications and desktop sessions that users can access remotely. To use RDS desktop pools or applications, your end users must have access to Horizon Client 3.0 or later software.

- Desktop Pools
  - There are three types of desktop pools: automated, manual, and RDS. Automated desktop pools use a vCenter Server virtual machine template or snapshot to create a pool of identical virtual machines. Manual desktop pools are a collection of existing vCenter Server virtual machines, physical computers, or third-party virtual machines. In automated or manual pools, each machine is available for one user to access remotely at a time. RDS desktop pools are not a collection of machines, but instead, provide users with desktop sessions on RDS hosts. Multiple users can have desktop sessions on an RDS host simultaneously.

- Application Pools
  - Application pools let you deliver applications to many users. The applications in application pools run on a farm of RDS hosts.

- Farms
  - Farms are collections of RDS hosts and facilitate the management of those hosts. Farms can have a variable number of RDS hosts and provide a common set of applications or RDS desktops to users. When you create an RDS desktop pool or an application pool, you must specify a farm. The RDS hosts in the farm provide desktop and application sessions to users.

Some of the latest VMware Horizon features and enhancements are:

- Flash Redirection
  You can compile a black list to ensure that the URLs specified in the list will not be able to redirect Flash content. You must enable the GPO setting FlashMMRUrlListEnableType to use either a white list or black list.

- Horizon Agent Policy Settings
  - The VMwareAgentCIT policy setting enables remote connections to Internet Explorer to use the Client's IP address instead of the IP address of the remote desktop machine.
  - The FlashMMRUrlListEnableType and FlashMMRUrlList policy settings specify and control the white list or black list that enables or disables the list of URLs from using Flash Redirection.

- Horizon PowerCLI
  - View PowerCLI is deprecated. Horizon PowerCLI replaces View PowerCLI and includes cmdlets that you can use with VMware PowerCLI.
  - For more information about Horizon PowerCLI cmdlets, read the *VMware PowerCLI Cmdlets Reference.*
  - For information on the API specifications to create advanced functions and scripts to use with Horizon PowerCLI, see the API Reference at the VMware Developer Center

  - For more information on sample scripts that you can use to create your own Horizon PowerCLI scripts, see the Horizon PowerCLI community on GitHub.

- Horizon 7 for Linux desktops enhancements

- UDP based Blast Extreme connectivity

  User Datagram Protocol (UDP) is enabled by default in both the client and the agent. Note that Transmission Control Protocol (TCP) connectivity will have a better performance than UDP on the Local Area Network (LAN). UDP will have better performance than TCP over Wide Area Network (WAN). If you are on a LAN, disable the UDP feature to switch to using TCP to get better connectivity performance.

- KDE support

  K Desktop Environment (KDE) support is now also available on CentOS 7, RHEL 7, Ubuntu 14.04, Ubuntu 16.04, and SLED 11 SP4 platforms.

- MATE support

  MATE desktop environment is supported on Ubuntu 14.04 and 16.04 virtual machines.

- Hardware H.264 Encoder

  The hardware H.264 encoder is now available and used when the vGPU is configured with the NVIDIA graphics card that has the NVIDIA driver 384 series or later installed on it.

- Additional platforms support

  RHEL 7.4 x64 and CentOS 7.4 x64 are now supported.

- Remote Desktop Operating System

  The following remote desktop operating systems are now supported:

  - Windows 10 version 1607 Long-Term Servicing Branch (LTSB)
  - Windows Server 2016

Horizon Agent

HTML5 Multimedia Redirection

You can install the HTML5 Multimedia Redirection feature by selecting the HTML5 Multimedia Redirection custom setup option in the Horizon Agent installer. With HTML5 Multimedia Redirection, if an end user uses the Chrome browser, HTML5 multimedia content is sent from the remote desktop to the client system, reducing the load on the ESXi host. The client system plays the multimedia content and the user has a better audio and video experience.

- SHA-256 support

  Horizon Agent has been updated to support the SHA-256 cryptographic hash algorithm. SHA-256 is also supported in Horizon Client 4.6 and Horizon 7 version 7.2 and later.

- Improved USB redirection with User Environment Manager

  The default User Environment Manager timeout value has been increased. This change ensures that the USB redirection smart policy takes effect even when the login process takes a long time. With Horizon Client 4.6, the User Environment Manager timeout value is configured only on the agent and is sent from the agent to the client.

  You can now bypass User Environment Manager control of USB redirection by setting a registry key on the agent machine. This change ensures that smart card SSO works on Teradici zero clients.

- Composer

  For enhanced security, you can enable the digest access authentication method for Composer.

Horizon Connection Server Enhanced Features

Horizon Help Desk Tool

- View application and process names and resource use within a virtual or published desktop to identify which applications and process are using up machine resources.
- View event log information about the user's activities.
- View updated metrics such as Horizon Client version and the Blast protocol.
- View additional session metrics such as the VM information, CPU, or memory usage.
- You can assign predefined administrator roles to Horizon Help Desk Tool administrators to delegate the troubleshooting tasks between administrator users. You can also create custom roles and add privileges based on the predefined administrator roles.
- You can verify the product license key for Horizon Help Desk Tool and apply a valid license.
- Monitoring - If the event database shuts down, Horizon Administrator maintains an audit trail of the events that occur before and after the event database shutdown.

Instant Clones

- You can create dedicated instant-clone desktop pools.
- Windows Server operating systems are supported for instant clones in this release. For an updated list of supported Windows Server operating systems, see the VMware Knowledge Base (KB) article 2150295.
- You can copy, paste, or enter the path for the AD tree in the AD container field when you create an instant-clone desktop pool.
- If there are no internal VMs in all four internal folders created in vSphere Web Client, these folders are unprotected and you can delete these folders.
- You can use the enhanced instant-clone maintenance utility `IcUnprotect.cmd` to unprotect or delete template, replica, or parent VMs or folders from vSphere hosts.

- Instant clones are compatible with Storage DRS (sDRS). Therefore, instant clones can reside in a datastore that is part of an sDRS cluster.

Cloud Pod Architecture

- The total session limit is increased to 140,000.
- The site limit is increased to 7.
- You can configure Windows Start menu shortcuts for global entitlements. When an entitled user connects to a Connection Server instance in the pod federation, Horizon Client for Windows places these shortcuts in the Start menu on the user's Windows client device.

Published Desktops and Application Pools

- You can restrict access to entitled desktop pools, application pools, global entitlements, and global application entitlements from certain client computers.
- You can configure Windows start menu shortcuts for entitled desktop and application pools. When an entitled user connects to a Connection Server instance, Horizon Client for Windows places these shortcuts in the Start menu on the user's Windows client device.

Virtual Desktops and Desktop Pools

- Blast Extreme provides network continuity during momentary network loss on Windows clients.
- Performance counters displayed using PerfMon on Windows agents for Blast session, imaging, audio, CDR, USB, and virtual printing provide an accurate representation of the current state of the system that also updates at a constant rate.

Customer Experience Improvement Program

Details about the data collected through CEIP and the purposes for which it is used by VMware can be found at the Trust Assurance Center.

- Security

  With the USB over Session Enhancement SDK feature, you do not need to open TCP port 32111 for USB traffic in a DMZ-based security server deployment. This feature is supported for both virtual desktops and published desktops on RDS hosts.

- Database Support

  The Always On Availability Groups feature for Microsoft SQL Server 2014 is supported in this release of Horizon 7.

See the VMware Horizon 7.4 Release Notes for more information.

Supported Windows 10 Operating Systems

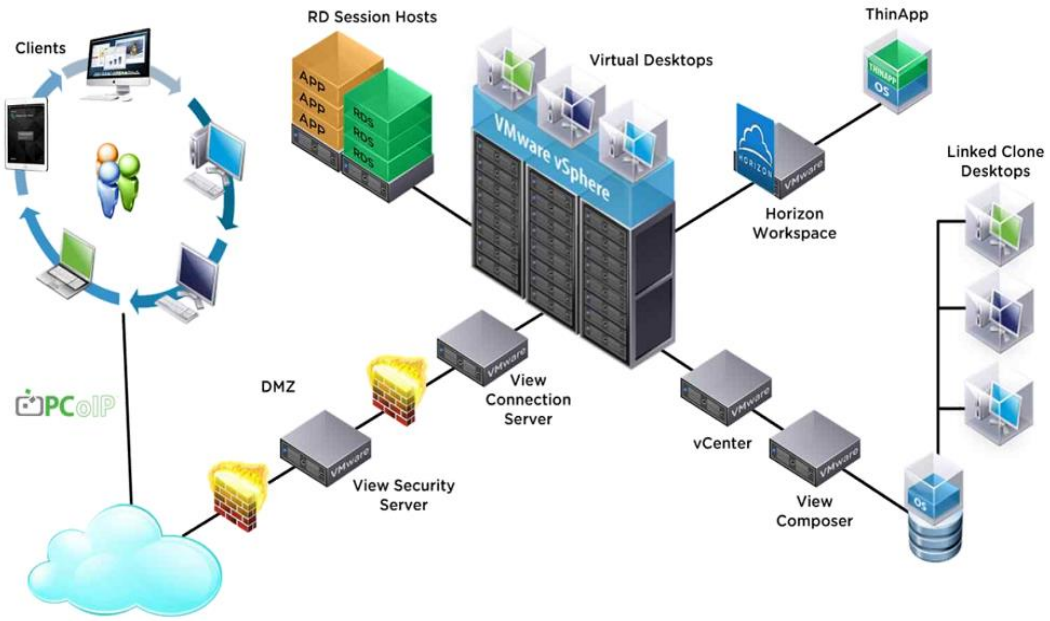
Horizon 7 version 7.4 supports the following Windows 10 operating systems:

- Windows 10 version 1507 (RTM) Long-Term Servicing Branch (LTSB)
- Windows 10 version 1607 Long-Term Servicing Branch (LTSB)

- Windows 10 version 1607 Enterprise Current Branch (CBB)
- Windows 10 version 1703 Semi Annual Channel (broad deployment) Current Branch (CBB)
- For the complete list of supported Windows 10 on Horizon including all VDI (Full Clones, Linked and Instant clones) click the below link.

Windows 10 LTSB version 1607 was used in this study.

Figure 11 Logical Architecture of VMware Horizon

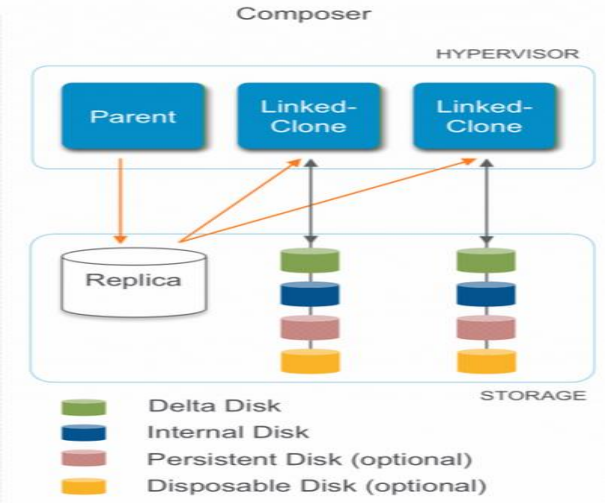


VMware Horizon Composer

VMware Horizon Composer is a feature in Horizon that gives administrators the ability to manage virtual machine pools or the desktop pools that share a common virtual disk. An administrator can update the master image, then all desktops using linked clones of that master image can also be patched. Updating the master image will patch the cloned desktops of the users without touching their applications, data or settings.

The VMware View Composer pooled desktops solution's infrastructure is based on software-streaming technology. After installing and configuring the composed pooled desktops, a single shared disk image (Master Image) is taken a snapshot of the OS and application image, and then storing that snapshot file accessible to host(s).

Figure 12 VMware Horizon Composer



Desktop Virtualization Design Fundamentals

An ever growing and diverse base of user devices, complexity in management of traditional desktops, security, and even Bring Your Own (BYO) device to work programs are prime reasons for moving to a virtual desktop solution.

VMware Horizon Design Fundamentals

VMware Horizon 7 integrates Remote Desktop Server Hosted sessions users and VDI desktop virtualization technologies into a unified architecture that enables a scalable, simple, efficient, mixed users and manageable solution for delivering Windows applications and desktops as a service.

Users can select applications from an easy-to-use “store” that is accessible from tablets, smartphones, PCs, Macs, and thin clients. VMware Horizon delivers a native touch-optimized experience via PCoIP or Blast Extreme high-definition performance, even over mobile networks.

Horizon VDI Pool and RDSH Servers Pool

Collections of identical Virtual Machines (VMs) or physical computers are managed as a single entity called a Desktop Pool. In this CVD, VM provisioning relies on VMware View Composer aligning with VMware Horizon Connection Server with vCenter Server components. In this CVD, machines in the Pools are configured to run either a Windows Server 2016 OS (for RDS Hosted shared sessions) and a Windows 10 Desktop OS (for pooled VDI desktops)

Figure 13 VMware Horizon Design Overview

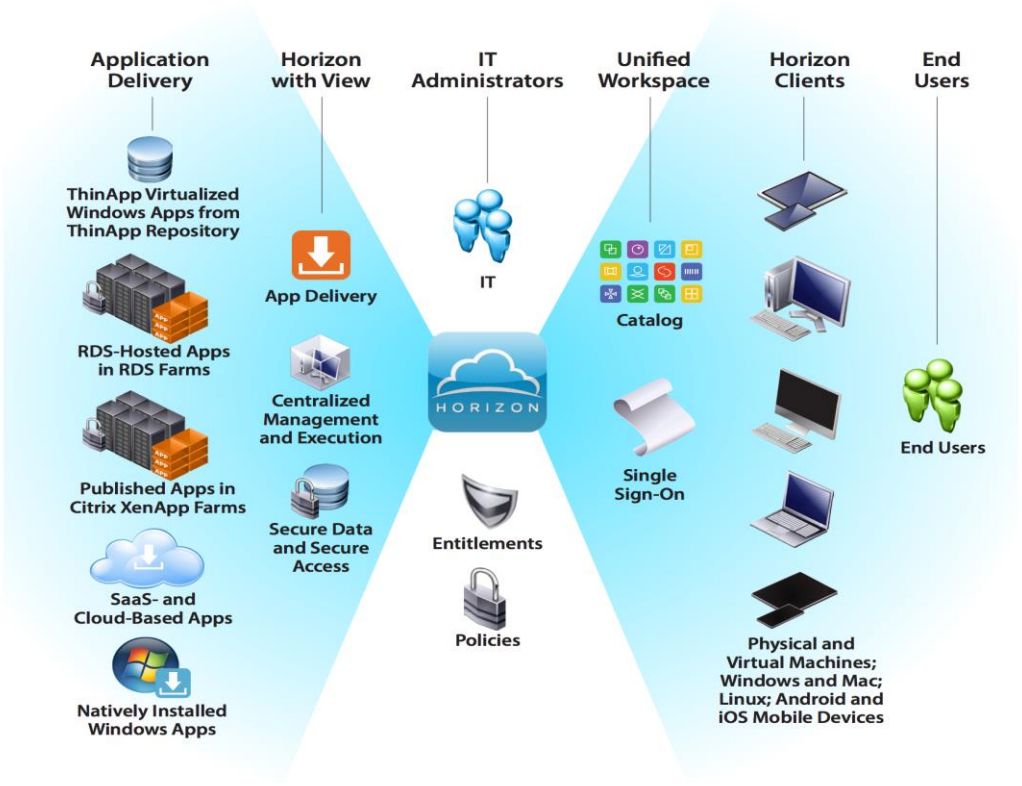
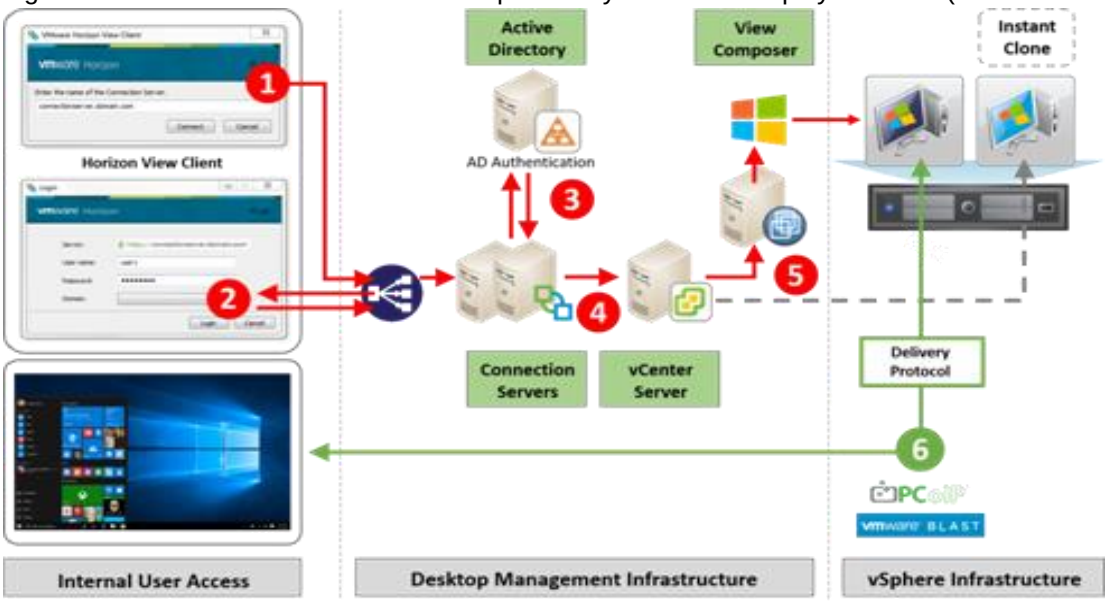


Figure 14 Horizon VDI and RDSH Desktop Delivery Based on Display Protocol (PCoIP/Blast/RDP)



Pure Storage FlashArray //X70

The Pure Storage FlashArray family delivers purpose-built, software-defined all-flash power and reliability for businesses of every size. FlashArray is all-flash enterprise storage that is up to 10X faster, more space

and power efficient, more reliable, and far simpler than other available solutions. Critically, FlashArray also costs less, with a TCO that's typically 50 percent lower than traditional performance disk arrays.

At the top of the FlashArray line is FlashArray//X – the first mainstream, 100 percent NVMe, enterprise-class all-flash array. //X represents a higher performance tier for mission-critical databases, top-of-rack flash deployments, and Tier 1 application consolidation. It is optimized for the lowest-latency workloads and delivers an unprecedented level of performance density that makes possible previously unattainable levels of consolidation.

FlashArray//X provides microsecond latency, 1PB in 3U, and GBs of bandwidth, with rich data services, proven 99.9999 percent availability (inclusive of maintenance and generational upgrades), 2X better data reduction versus alternative all-flash solutions, and DirectFlash™ global flash management. Further, //X is self-managing and plug-n-play, thanks to unrivalled Pure1® Support and the cloud-based, machine-learning predictive analytics of Pure1 Meta. Finally, FlashArray//X, like the rest of the FlashArray line, has revolutionized the 3-5 year storage refresh cycle by eliminating it: Pure's Evergreen™ Storage model provides a subscription to hardware and software innovation that enables organizations to expand and enhance their storage for 10 years or more.

Figure 15 Pure Storage FlashArray //X70

At the heart of FlashArray//X is the Purity Operating Environment software. Purity enables organizations to enjoy Tier 1 data services for all workloads, completely non-disruptive operations, and the power and efficiency of DirectFlash. Moreover, Purity includes enterprise-grade data security, comprehensive data protection options, and complete business continuity via ActiveCluster multi-site stretch cluster – all included with every array.

Figure 16 Pure Storage FlashArrays

Pure Storage FlashArray sets the benchmark for all-flash enterprise storage arrays. It delivers the following:

Consistent Performance FlashArray delivers consistent <1ms average latency. Performance is optimized for the real-world applications workloads that are dominated by I/O sizes of 32K or larger vs. 4K/8K hero performance benchmarks. Full performance is maintained even under failures/updates.

Less Cost than Disk Inline de-duplication and compression deliver 5 – 10x space savings across a broad set of I/O workloads including Databases, Virtual Machines and Virtual Desktop Infrastructure. With VDI workloads data reduction is typically > 10:1

Disaster Recovery Built-In FlashArray offers native, fully-integrated, data reduction-optimized backup and disaster recovery at no additional cost. Setup disaster recovery with policy-based automation within minutes. In addition, recover instantly from local, space-efficient snapshots or remote replicas.

Mission-Critical Resiliency FlashArray delivers >99.999 percent proven availability, as measured across the Pure Storage installed base and does so with non-disruptive everything without performance impact.

Architecture and Design Considerations for Desktop Virtualization

There are many reasons to consider a virtual desktop solution such as an ever growing and diverse base of user devices, complexity in management of traditional desktops, security, and even Bring Your Own Device (BYOD) to work programs. The first step in designing a virtual desktop solution is to understand the user community and the type of tasks that are required to successfully execute their role. The following user classifications are provided:

- Knowledge Workers today do not just work in their offices all day – they attend meetings, visit branch offices, work from home, and even coffee shops. These anywhere workers expect access to all of their same applications and data wherever they are.
- External Contractors are increasingly part of your everyday business. They need access to certain portions of your applications and data, yet administrators still have little control over the devices they use and the locations they work from. Consequently, IT is stuck making trade-offs on the cost of providing these workers a device vs. the security risk of allowing them access from their own devices.
- Task Workers perform a set of well-defined tasks. These workers access a small set of applications and have limited requirements from their PCs. However, since these workers are interacting with your customers, partners, and employees, they have access to your most critical data.
- Mobile Workers need access to their virtual desktop from everywhere, regardless of their ability to connect to a network. In addition, these workers expect the ability to personalize their PCs, by installing their own applications and storing their own data, such as photos and music, on these devices.
- Shared Workstation users are often found in state-of-the-art university and business computer labs, conference rooms or training centers. Shared workstation environments have the constant requirement to re-provision desktops with the latest operating systems and applications as the needs of the organization change, tops the list.

After the user classifications have been identified and the business requirements for each user classification have been defined, it becomes essential to evaluate the types of virtual desktops that are needed based on user requirements. There are essentially five potential desktops environments for each user:

- Traditional PC: A traditional PC is what typically constitutes a desktop environment: physical device with a locally installed operating system.
- Hosted Shared Desktop: A hosted, server-based desktop is a desktop where the user interacts through a delivery protocol. With hosted, server-based desktops, a single installed instance of a server operating system, such as Microsoft Windows Server 2016, is shared by multiple users simultaneously. Each user receives a desktop "session" and works in an isolated memory space. Remoted Desktop Server Hosted Server sessions: A hosted virtual desktop is a virtual desktop running on a virtualization layer (ESX). The user does not work with and sit in front of the desktop, but instead the user interacts through a delivery protocol.
- Published Applications: Published applications run entirely on the VMware Horizon RDS hosted server virtual machines and the user interacts through a delivery protocol. With published applications, a

single installed instance of an application, such as Microsoft Office, is shared by multiple users simultaneously. Each user receives an application "session" and works in an isolated memory space.

- Streamed Applications: Streamed desktops and applications run entirely on the user's local client device and are sent from a server on demand. The user interacts with the application or desktop directly but the resources may only available while they are connected to the network.
- Local Virtual Desktop: A local virtual desktop is a desktop running entirely on the user's local device and continues to operate when disconnected from the network. In this case, the user's local device is used as a type 1 hypervisor and is synced with the data center when the device is connected to the network.
- For the purposes of the validation represented in this document, both VMware Horizon hosted virtual desktops and Remote Desktop Server Hosted sessions were validated. Each of the sections provides some fundamental design decisions for this environment.

Understanding Applications and Data

When the desktop user groups and sub-groups have been identified, the next task is to catalog group application and data requirements. This can be one of the most time-consuming processes in the VDI planning exercise, but is essential for the VDI project's success. If the applications and data are not identified and co-located, performance will be negatively affected.

The process of analyzing the variety of application and data pairs for an organization will likely be complicated by the inclusion cloud applications, for example, SalesForce.com. This application and data analysis is beyond the scope of this Cisco Validated Design, but should not be omitted from the planning process. There are a variety of third party tools available to assist organizations with this crucial exercise.

Project Planning and Solution Sizing Sample Questions

Now that user groups, their applications and their data requirements are understood, some key project and solution sizing questions may be considered.

General project questions should be addressed at the outset, including:

- Has a VDI pilot plan been created based on the business analysis of the desktop groups, applications and data?
- Is there infrastructure and budget in place to run the pilot program?
- Are the required skill sets to execute the VDI project available? Can we hire or contract for them?
- Do we have end user experience performance metrics identified for each desktop sub-group?
- How will we measure success or failure?
- What is the future implication of success or failure?

Below is a short, non-exhaustive list of sizing questions that should be addressed for each user sub-group:

- What is the desktop OS planned? Windows 8 or Windows 10?

- 32 bit or 64 bit desktop OS?
- How many virtual desktops will be deployed in the pilot? In production? All Windows 8/10?
- How much memory per target desktop group desktop?
- Are there any rich media, Flash, or graphics-intensive workloads?
- Will VMware Horizon RDSH be used for Hosted Shared Server applications planned? Are they are any applications installed?
- What is the desktop OS planned for RDS Server Roles? Windows server 2012 or Server 2016?
- Will VMware Horizon Composer or Instant Clones or another method be used for virtual desktop deployment?
- What is the hypervisor for the solution?
- What is the storage configuration in the existing environment?
- Are there sufficient IOPS available for the write-intensive VDI workload?
- Will there be storage dedicated and tuned for VDI service?
- Is there a voice component to the desktop?
- Is anti-virus a part of the image?
- What is the SQL server version for database? SQL server 2012 or 2016?
- Is user profile management (for example, non-roaming profile based) part of the solution?
- What is the fault tolerance, failover, disaster recovery plan?
- Are there additional desktop sub-group specific questions?

Hypervisor Selection

VMware vSphere has been identified the hypervisor for both RDS Hosted Sessions and VDI based desktops:

- VMware vSphere: VMware vSphere comprises the management infrastructure or virtual center server software and the hypervisor software that virtualizes the hardware resources on the servers. It offers features like Distributed Resource Scheduler, vMotion, high availability, Storage vMotion, VMFS, and a multi-pathing storage layer. More information on vSphere can be obtained at the VMware web site.

For this CVD, the hypervisor used was VMware ESXi 6.5 Update 1.

Server OS and Desktop OS Machines configured in this CVD to support Remote Desktop Server Hosted (RDSH) shared sessions and Hosted Virtual Desktops (both non-persistent and persistent).

Designing a VMware Horizon Environment for a Mixed Workload

With VMware Horizon 7 the method you choose to provide applications or desktops to users depends on the types of applications and desktops you are hosting and available system resources, as well as the types of users and user experience you want to provide.

Table 2 Designing a VMware Horizon Environment

Server OS machines	You want: Inexpensive server-based delivery to minimize the cost of delivering applications to a large number of users, while providing a secure, high-definition user experience. Your users: Perform well-defined tasks and do not require personalization or offline access to applications. Users may include task workers such as call center operators and retail workers, or users that share workstations. Application types: Any application.
Desktop OS machines	You want: A client-based application delivery solution that is secure, provides centralized management, and supports a large number of users per host server (or hypervisor), while providing users with applications that display seamlessly in high-definition. Your users: Are internal, external contractors, third-party collaborators, and other provisional team members. Users do not require off-line access to hosted applications. Application types: Applications that might not work well with other applications or might interact with the operating system, such as .NET framework. These types of applications are ideal for hosting on virtual machines. Applications running on older operating systems such as Windows XP or Windows Vista, and older architectures, such as 32-bit or 16-bit. By isolating each application on its own virtual machine, if one machine fails, it does not impact other users.
Remote PC Access	You want: Employees with secure remote access to a physical computer without using a VPN. For example, the user may be accessing their physical desktop PC from home or through a public WIFI hotspot. Depending upon the location, you may want to restrict the ability to print or copy and paste outside of the desktop. This method enables BYO device support without migrating desktop images into the datacenter. Your users: Employees or contractors that have the option to work from home, but need access to specific software or data on their corporate desktops to perform their jobs remotely. Host: The same as Desktop OS machines. Application types: Applications that are delivered from an office computer

	and display seamlessly in high definition on the remote user's device.

For the Cisco Validated Design described in this document, a mix of Remote Desktop Server Hosted sessions (RDSH) using RDS based Server OS and VMware Horizon pooled Linked Clone Virtual Machine Desktops using VDI based desktop OS machines were configured and tested.

The mix consisted of a combination of both use cases. The following sections discuss design decisions relative to the VMware Horizon deployment, including the CVD test environment.

Solution Design

Products Deployed

The architecture deployed is highly modular. While each customer's environment might vary in its exact configuration, the reference architecture contained in this document once built, can easily be scaled as requirements and demands change. This includes scaling both up (adding additional resources within a Cisco UCS Domain) and out (adding additional Cisco UCS Domains and Pure Storage FlashArrays).

The FlashStack Datacenter solution includes Cisco networking, Cisco UCS and Pure Storage FlashArray //X, which efficiently fit into a single data center rack, including the access layer network switches.

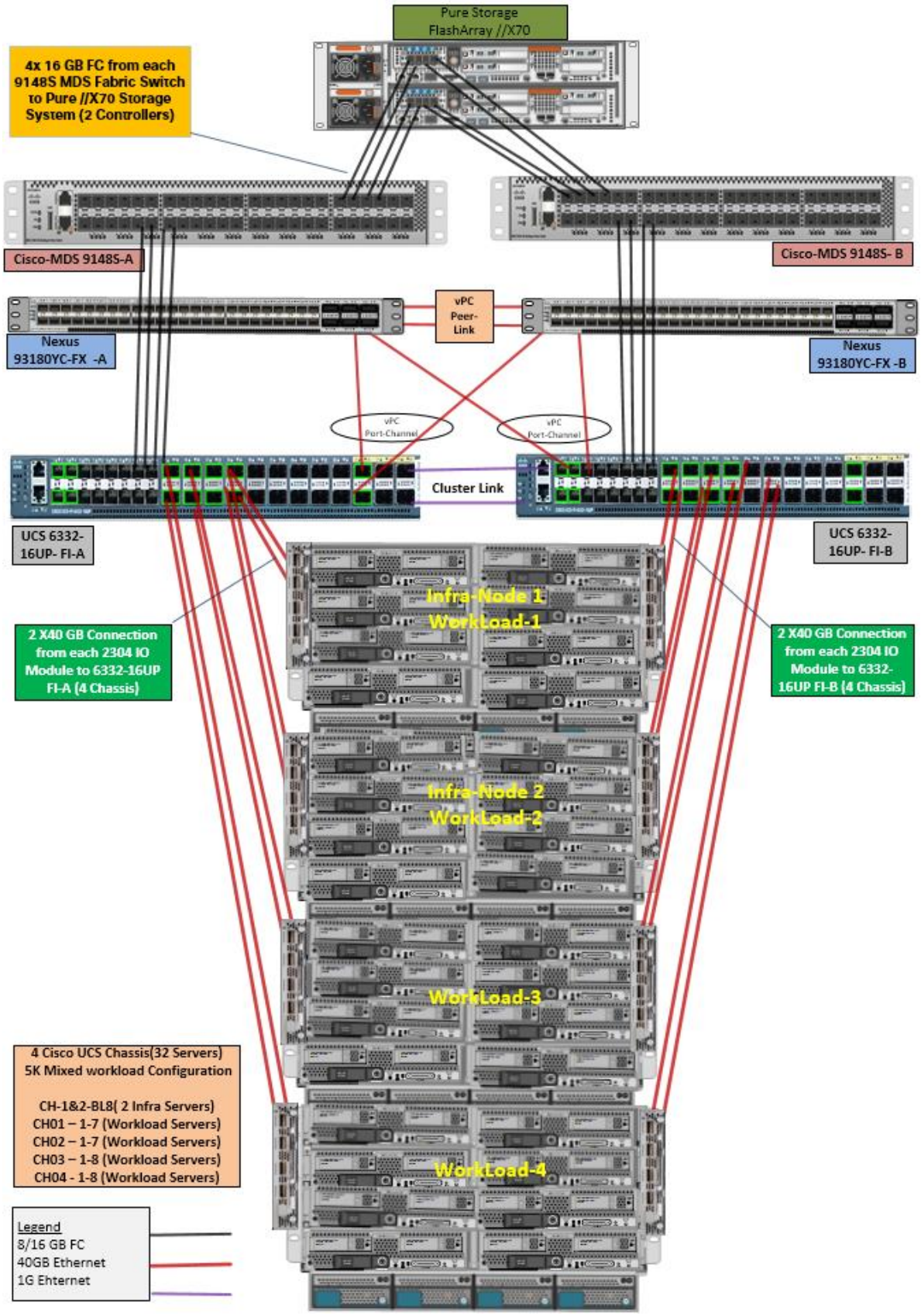
This CVD details the deployment of 6000 users for a mixed VMware Horizon desktop workload featuring the following software:

This document details the deployment of the multiple configurations extending to 6000 users for a mixed Horizon workload featuring the following software:

- VMware vSphere ESXi 6.5 Update 1 Hypervisor
- Microsoft SQL Server 2016
- VMware Horizon 7 Shared Remote Desktop Server Hosted Sessions (RDSH) on Pure Storage FlashArray //X70 FC storage
- VMware Horizon 7 Non-Persistent Virtual Desktops (VDI) on Pure Storage FlashArray //X70 FC storage
- VMware Horizon 7 Persistent Virtual Desktops (VDI) on Pure Storage FlashArray //X70 FC storage
- VMware Horizon 7 Connection Server and Additional Replica Servers
- VMware Horizon 7 Composer Server
- Microsoft Windows Server 2016 for Infrastructure
- Microsoft Windows Server 2016 for RDS Server Roles Configuration
- Windows 10 64-bit virtual machine Operating Systems for Non- Persistent and Persistent virtual machine users

Figure 17 details the physical hardware and cabling deployed to enable this solution.

Figure 17 Virtual Desktop Workload Architecture for the 6000 Seat on VMware Horizon 7 on FlashStack



Hardware Deployed

The solution contains the following hardware as shown in Figure 17.

- Cisco Nexus 93180YC-FX Layer 2 Access Switches (2)
- Cisco MDS 9148S 16Gb Fibre Channel Switches (2)
- Cisco UCS 5108 Blade Server Chassis with two Cisco UCS-IOM-2304 IO Modules per Chassis (4)
- Cisco UCS B200 M4 Blade servers with Intel Xeon E5-2660v4 2-GHz 14-core processors, 256GB 2400MHz RAM, and one Cisco VIC1340 mezzanine card for the hosted infrastructure, providing N+1 server fault tolerance (2)
- Cisco UCS B200 M5 Blade Servers with Intel Xeon Gold 6140 2.30-GHz 18-core processors, 768GB 2666MHz RAM, and one Cisco VIC1340 mezzanine card for the VMware Horizon Remote Desktop Server Hosted Sessions workload, providing N+1 server fault tolerance at the workload cluster level (10)
- Cisco UCS B200 M5 Blade Servers with Intel Xeon Gold 6140 2.30-GHz 18-core processors, 768GB 2666MHz RAM, and one Cisco VIC1340 mezzanine card for the VMware Horizon Full Clones VDI desktops workload, providing N+1 server fault tolerance at the workload cluster level (20)
- Pure Storage FlashArray //X70 with dual redundant controllers, with Twenty 1.92TB DirectFlash NVMe drives

Table 3 lists the software and firmware version used in the study.

Table 3 Software and Firmware Versions

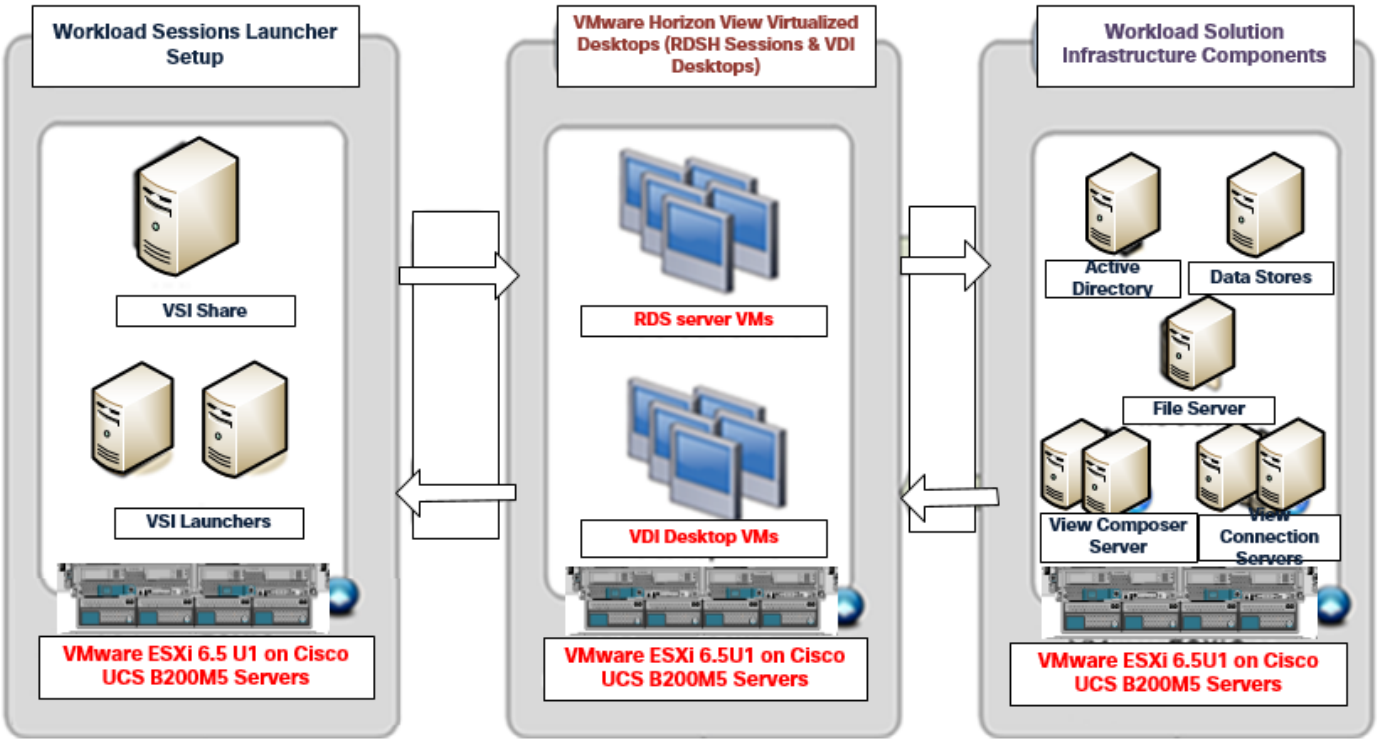
Vendor	Product / Component	Version / Build / Code
Cisco	UCS Component Firmware	3.2(2f) bundle release
Cisco	UCS Manager	3.2(2f) bundle release
Cisco	UCS B200 M4 Blades	3.2(2f) bundle release
Cisco	VIC 1340	4.2(2b)
VMware	VMware Horizon	7.4.0
VMware	VMware Composer Server	7.4.0-7312595
VMware	vCenter Server Appliance	6.5.0-8024368
VMware	vSphere ESXi 6.5 Update 1	6.5.0-7967591
Pure Storage	FlashArray //X70	Purity//FA v5.0.2

Logical Architecture

The logical architecture of this solution is designed to support up to 6000 users within four Cisco UCS 5108 Blade server chassis containing 30 blades, which provides physical redundancy for the blade servers for each workload type.

Figure 18 illustrates the logical architecture of the test environment, including the Login VSI session launcher self-contained end user experience benchmarking platform.

Figure 18 Logical Architecture Overview



This document is intended to allow you to fully configure your environment. In this process, various steps require you to insert customer specific naming conventions, IP addresses and VLAN schemes, as well as to record appropriate MAC address and WWPN and WWNN for FC connectivity.

Figure 19 identifies the server roles in the 32 server deployment to support the 6000 seat workload. We also break out the infrastructure virtual machine fault tolerant design.

Figure 19 Server, Location, and Purpose

Chassis/Host	Purpose
CH01-Blade01	RDSH
CH01-Blade02	RDSH
CH01-Blade03	RDSH
CH01-Blade04	Persistent
CH01-Blade05	Persistent
CH01-Blade06	Non-Persistent
CH01-Blade07	Non-Persistent
CH01-Blade08	Infra01
CH02-Blade01	RDSH
CH02-Blade02	RDSH
CH02-Blade03	RDSH
CH02-Blade04	Persistent
CH02-Blade05	Persistent
CH02-Blade06	Non-Persistent
CH02-Blade07	Non-Persistent
CH02-Blade08	Infra02
CH03-Blade01	RDSH
CH03-Blade02	RDSH
CH03-Blade03	Persistent
CH03-Blade04	Persistent
CH03-Blade05	Persistent
CH03-Blade06	Non-Persistent
CH03-Blade07	Non-Persistent
CH03-Blade08	Non-Persistent
CH04-Blade01	RDSH
CH04-Blade02	RDSH
CH04-Blade03	Persistent
CH04-Blade04	Persistent
CH04-Blade05	Persistent
CH04-Blade06	Non-Persistent
CH04-Blade07	Non-Persistent
CH04-Blade08	Non-Persistent

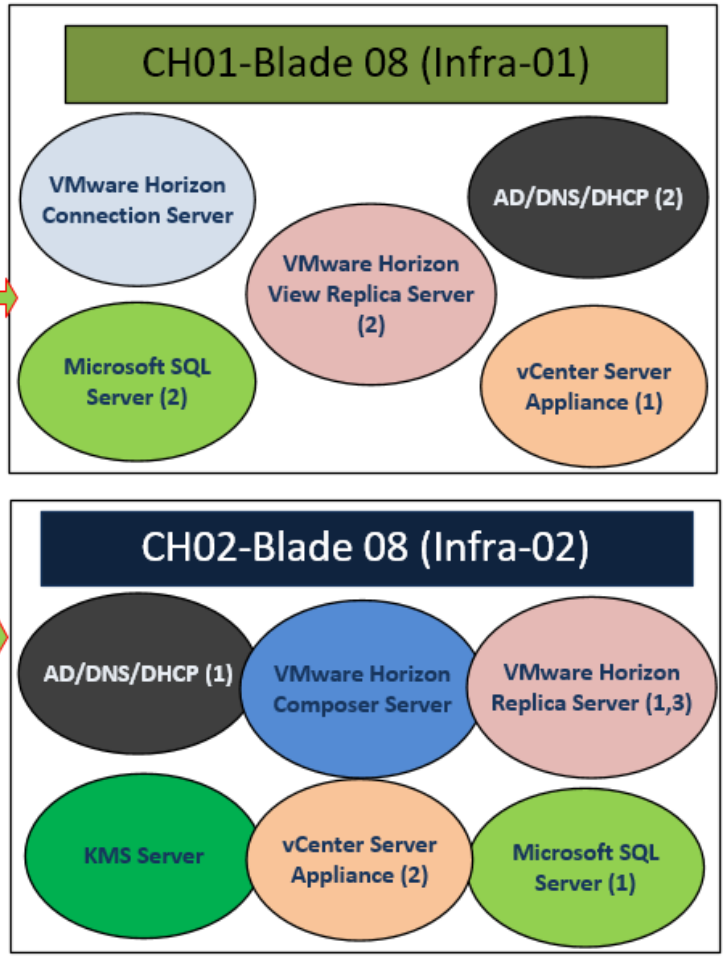


Table 4 lists the virtual machine deployments on the hardware platform.

Table 4 Virtual Machine Deployment Architecture

Server name	Location	Purpose

Server name	Location	Purpose
CH01-Blade 8 CH02-Blade 8	Physical – Chassis 1, 2	ESXi 6.5 Hosts Infrastructure VMs Windows 2016-R2, vCenter Server Appliance, VMware Horizon Connection Servers, View Replica Servers, View Composer Server, Active Directory Domain Controllers, SQL Server and Key Management Server.
CH01-Blade 1-3 CH02-Blade 1-3 CH03-Blade 1-2 CH04-Blade 1-2	Physical – Chassis 1, 2	ESXi 6.5 Hosts 72x VMware Horizon Server 2016 RDSH Server VMs (2430 RDS Server Sessions)
CH01-Blade 4-7 CH02-Blade 4-7 CH03-Blade 3-8 CH04-Blade 3-8	Physical – Chassis 1,2, 3, 4	ESXi 6.5 Hosts 3570x VMware Horizon VDI (3 Pools consist of Non-Persistent and Persistent virtual machines) VMs

VLANs

The VLAN configuration recommended for the environment includes a total of six VLANs as outlined in Table 5 .

Table 5 VLANs configured in this study

VLAN Name	VLAN ID	VLAN Purpose
Default	1	Native VLAN
In-Band-Mgmt	70	In-Band management interfaces
Infra-Mgmt	71	Infrastructure Virtual Machines
VCC/VM-Network	72	RDSH, Persistent and Non-Persistent
vMotion	73	VMware vMotion
OOB-Mgmt	164	Out of Band management interfaces

VSANs

We configured two virtual SANs for communications and fault tolerance in this design.

Table 6 VSANs Configured in this Study

VSAN Name	VSAN ID	Purpose
VSAN 100	100	VSAN for Primary SAN communication
VSAN 101	101	VSAN for Secondary SAN communication

Solution Cabling

The following sections detail the physical connectivity configuration of the FlashStack 6000 seat VMware Horizon 7 environment.

The information provided in this section is as a reference for cabling the physical equipment in this Cisco Validated Design environment. To simplify cabling requirements, the tables include both local and remote device and port locations.

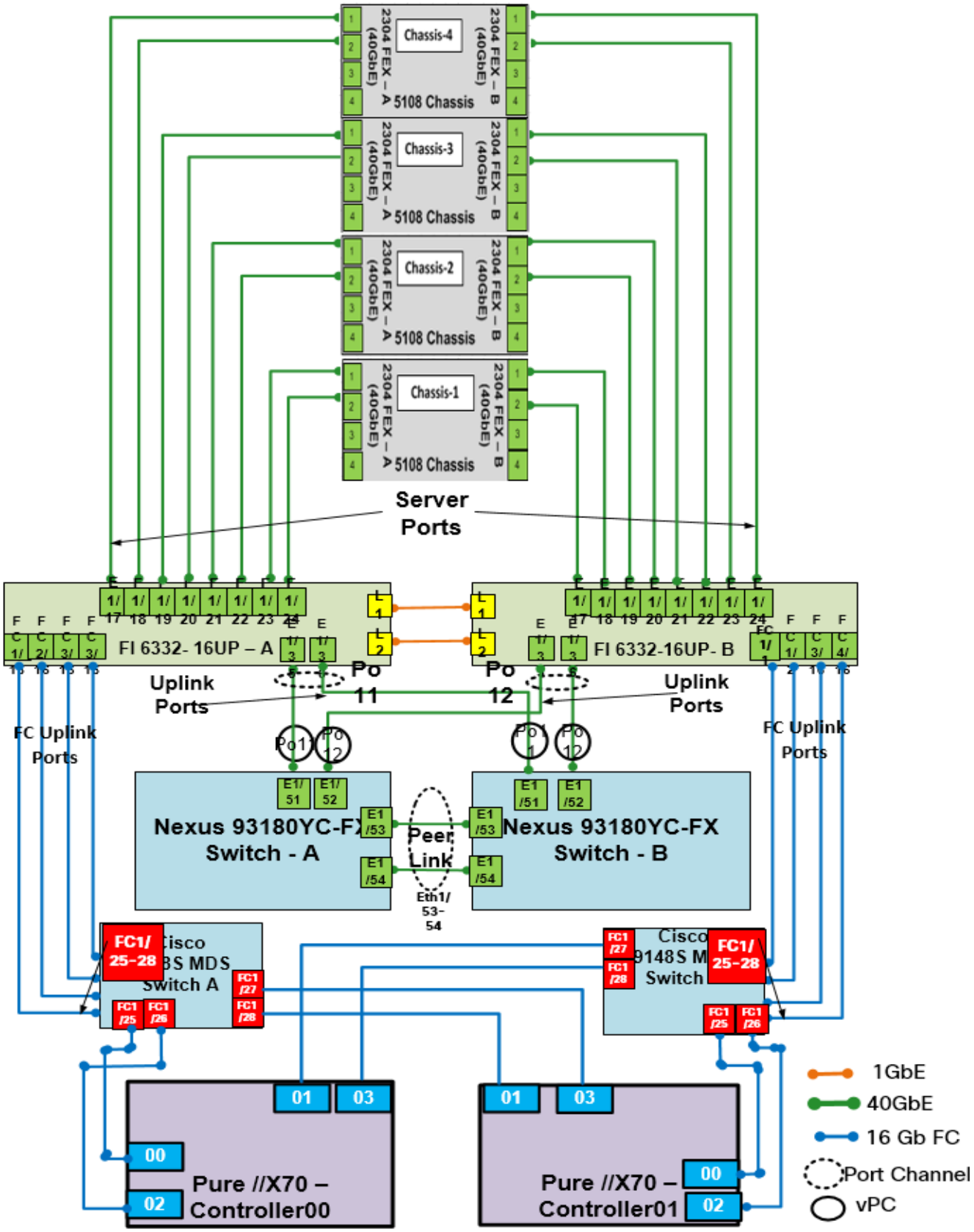
The tables in this section contain the details for the prescribed and supported configuration of the Pure Storage FlashArray //X70 storage array to the Cisco 6332-16UP Fabric Interconnects via Cisco MDS 9148S FC switches.

This document assumes that out-of-band management ports are plugged into an existing management infrastructure at the deployment site. These interfaces will be used in various configuration steps.

Be sure to follow the cabling directions in this section. Failure to do so will result in necessary changes to the deployment procedures that follow because specific port locations are mentioned.

Figure 20 shows a cabling diagram for a VMware Horizon configuration using the Cisco Nexus 9000, Cisco MDS 9100 Series, and Pure Storage //X70 array.

Figure 20 FlashStack 6000 Seat Cabling Diagram



Cisco Unified Computing System Base Configuration

This section details the Cisco UCS configuration that was done as part of the infrastructure build out. The racking, power, and installation of the chassis are described in the Cisco UCS Manager Getting Started Guide and it is beyond the scope of this document. For more information about each step, refer to the following document, Cisco UCS Manager - Configuration Guides.

Cisco UCS Manager Software Version 3.2(2f)

This document assumes you are using Cisco UCS Manager Software version 3.2(2f). To upgrade the Cisco UCS Manager software and the Cisco UCS 6332-16UP Fabric Interconnect software to a higher version of the firmware,) refer to Cisco UCS Manager Install and Upgrade Guides.

Configure Fabric Interconnects at Console

To configure the fabric Interconnects, complete the following steps:

1. Connect a console cable to the console port on what will become the primary fabric interconnect.

2. If the fabric interconnects was previously deployed and you want to erase it to redeploy, follow these steps:

   a. Login with the existing user name and password.

      # connect local-mgmt
      # erase config
      # yes (to confirm)

3. After the fabric interconnect restarts, the out-of-box first time installation prompt appears, type “console” and press Enter.

4. Follow the Initial Configuration steps as outlined in Cisco UCS Manager Getting Started Guide. When configured, Login to UCSM IP Address via Web interface to perform base Cisco UCS configuration.

Configure Fabric Interconnects for a Cluster Setup

To configure the Cisco UCS Fabric Interconnects, complete the following steps:

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

2. Connect to the console port on the first Fabric Interconnect.

3. Review the settings on the console. Answer yes to Apply and Save the configuration.

4. Wait for the login prompt to make sure the configuration has been saved to Fabric Interconnect A.

5. Connect the console port on the second Fabric Interconnect, configure secondary FI.

Figure 21 Initial Setup of Cisco UCS Manager on Primary Fabric Interconnect

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

Enter the setup mode; setup newly or restore from backup. (setup/restore) ? setup

You have chosen to setup a new Fabric interconnect. Continue? (y/n): y

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

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

Is this Fabric interconnect part of a cluster(select 'no' for standalone)? (yes/no) [n]: yes

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

Enter the system name:  VCC-AAD17

Physical Switch Mgmt0 IP address : 10.29.164.246

Physical Switch Mgmt0 IPv4 netmask : 255.255.255.0

IPv4 address of the default gateway : 10.29.164.1

Cluster IPv4 address : 10.29.164.245

Configure the DNS Server IP address? (yes/no) [n]:

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

Join centralized management environment (UCS Central)? (yes/no) [n]:

Following configurations will be applied:

  Switch Fabric=A
  System Name=VCC-AAD17
  Enforced Strong Password=no
  Physical Switch Mgmt0 IP Address=10.29.164.246
  Physical Switch Mgmt0 IP Netmask=255.255.255.0
  Default Gateway=10.29.164.1
  Ipv6 value=0

  Cluster Enabled=yes
  Cluster IP Address=10.29.164.245
  NOTE: Cluster IP will be configured only after both Fabric Interconnects are initialized.
        UCSM will be functional only after peer FI is configured in clustering mode.

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

Configuration file - Ok


Cisco UCS 6300 Series Fabric Interconnect
VCC-AAD17-A login:
```

Figure 22 Initial Setup of Cisco UCS Manager on Secondary Fabric Interconnect

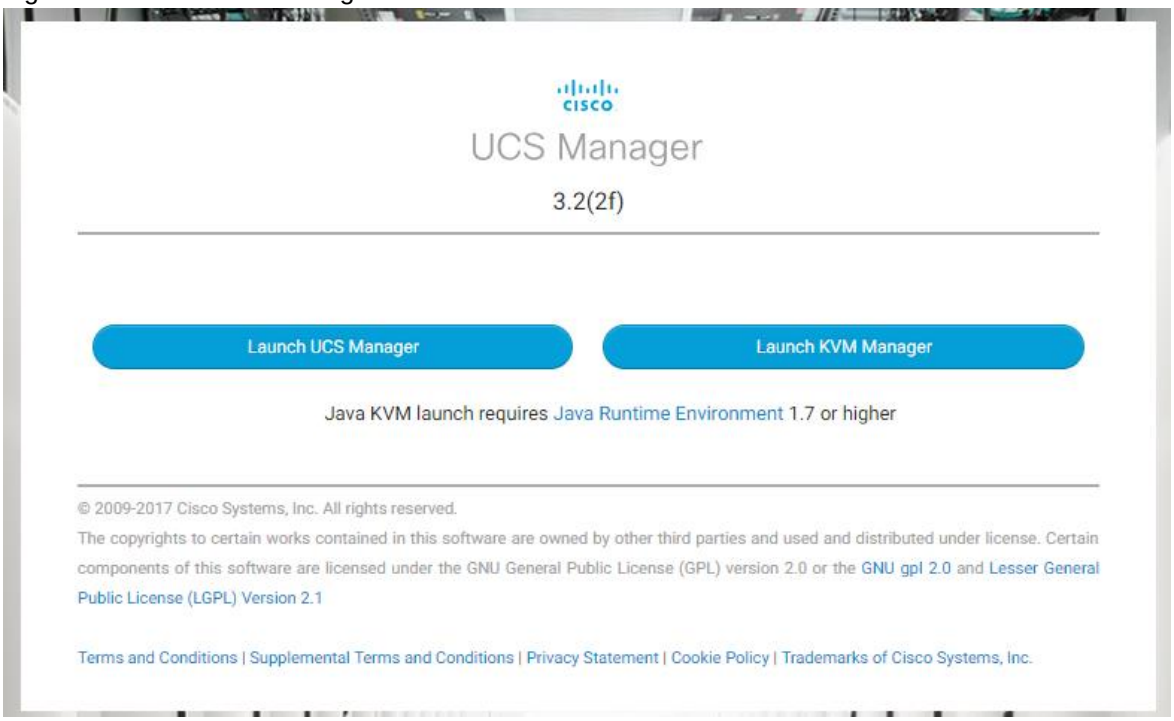
```
Enter the configuration method. (console/gui) ? console

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 Mgmt0 IPv4 Address: 10.29.164.246
  Peer Fabric interconnect Mgmt0 IPv4 Netmask: 255.255.255.0
  Cluster IPv4 address          : 10.29.164.245

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

Physical Switch Mgmt0 IP address : 10.29.164.247


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

Fri Feb 16 18:53:15 UTC 2018
Configuration file - Ok


Cisco UCS 6300 Series Fabric Interconnect
VCC-AAD17-B login:
```

6. To log into the Cisco Unified Computing System (Cisco UCS) environment, complete the following steps:

7. Open a web browser and navigate to the Cisco UCS Fabric Interconnect cluster address configured above.

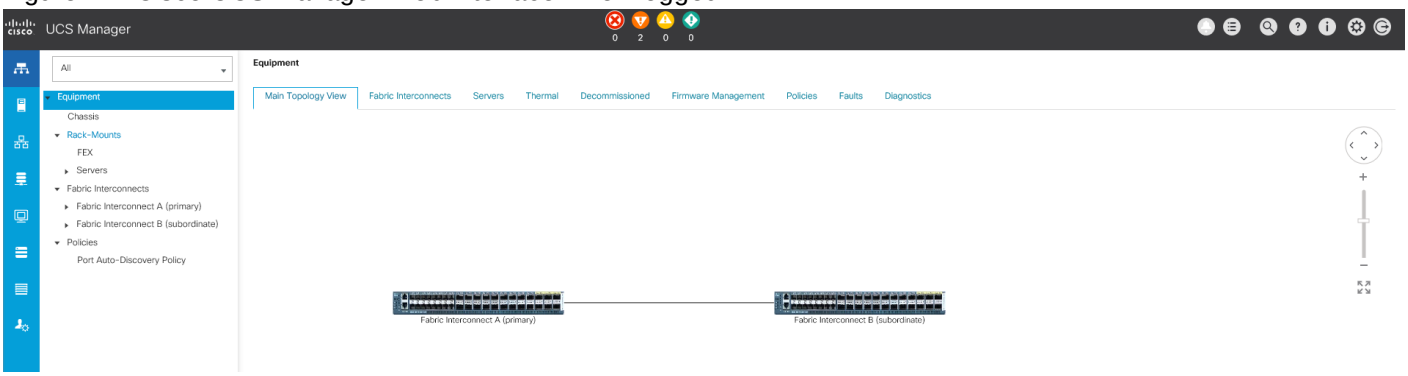
8. Click the Launch UCS Manager link to download the Cisco UCS Manager software. If prompted, accept the security certificates.

Figure 23 Cisco UCS Manager Web Interface



9. When prompted, enter the user name and password enter the password. Click “Log In” to login to Cisco UCS Manager.

Figure 24 Cisco UCS Manager Web Interface when logged in.



Configure Base Cisco Unified Computing System

The following are the high-level steps involved for a Cisco UCS configuration:

- Configure Fabric Interconnects for a Cluster Setup.

- Set Fabric Interconnects to Fibre Channel End Host Mode.
- Synchronize Cisco UCS to NTP.
- Configure Fabric Interconnects for Chassis and Blade Discovery:
- Configure Global Policies
- Configure Server Ports
- Configure LAN and SAN on Cisco UCS Manager:
- Configure Ethernet LAN Uplink Ports
- Create Uplink Port Channels to Cisco Nexus Switches
- Configure FC SAN Uplink Ports
- Configure VLAN
- Configure VSAN
- Configure IP, UUID, Server, MAC, WWNN and WWPN Pools:
- IP Pool Creation
- UUID Suffix Pool Creation
- Server Pool Creation
- MAC Pool Creation
- WWNN and WWPN Pool Creation
- Set Jumbo Frames in both the Cisco Fabric Interconnect.
- Configure Server BIOS Policy.
- Create Adapter Policy.
- Configure Update Default Maintenance Policy.
- Configure vNIC and vHBA Template
- Create Server Boot Policy for SAN Boot

Details for each step are discussed in the following sections.

Synchronize Cisco UCSM to NTP

To synchronize the Cisco UCS environment to the NTP server, complete the following steps:

1. In Cisco UCS Manager, in the navigation pane, click the Admin tab.
2. Select All > Time zone Management.

3. In the Properties pane, select the appropriate time zone in the Time zone menu.

4. Click Save Changes and then click OK.

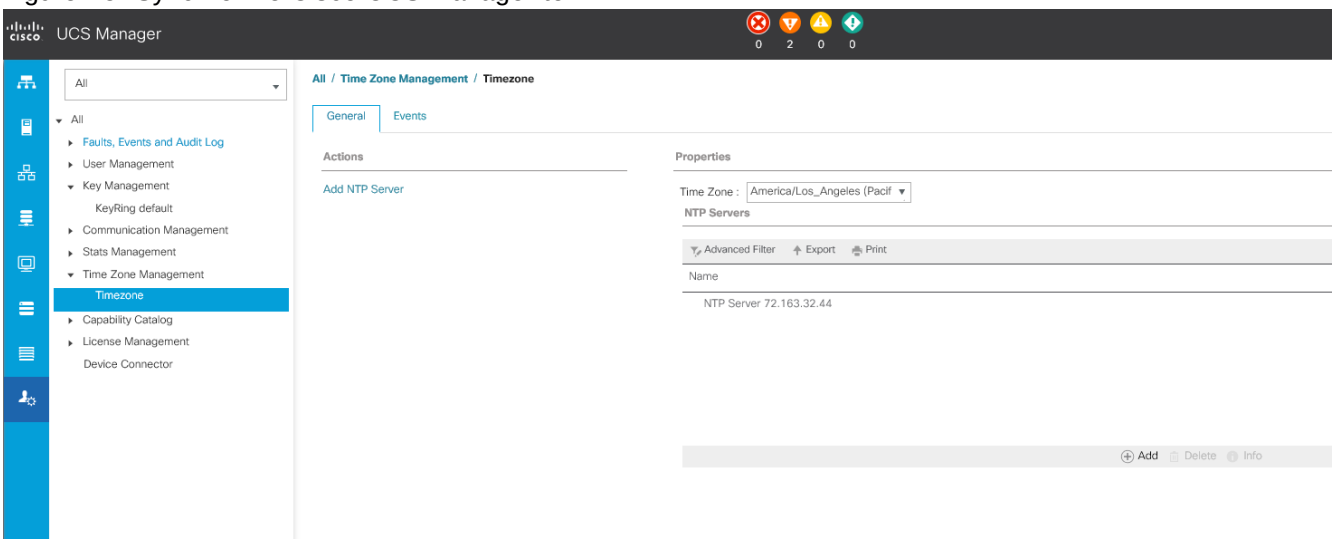
5. Click Add NTP Server.

6. Enter the NTP server IP address and click OK.

7. Click OK to finish.

8. Click Save Changes.

Figure 25 Synchronize Cisco UCS Manager to NTP



9. Configure Fabric Interconnects for Chassis and Blade Discovery

Cisco UCS 6332-16UP Fabric Interconnects are configured for redundancy. It provides resiliency in case of failures. The first step is to establish connectivity between blades and Fabric Interconnects.

Configure Global Policies

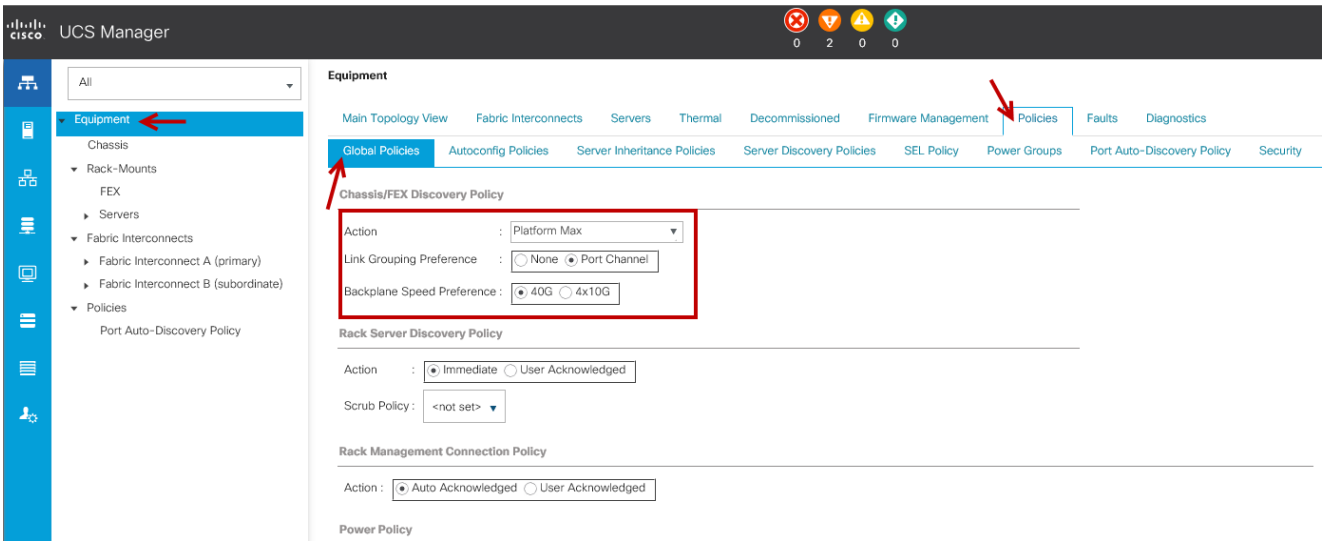
The chassis discovery policy determines how the system reacts when you add a new chassis. We recommend using the platform max value as shown. Using platform max helps ensure that Cisco UCS Manager uses the maximum number of IOM uplinks available.

To configure global policies, complete the following steps:

1. In Cisco UCS Manager; Go to Equipment > Policies (right pane) > Global Policies > Chassis/FEX Discovery Policies. As shown in the screenshot below, select Action as “Platform Max” from the drop-down list and set Link Grouping to Port Channel.

2. Click Save Changes.

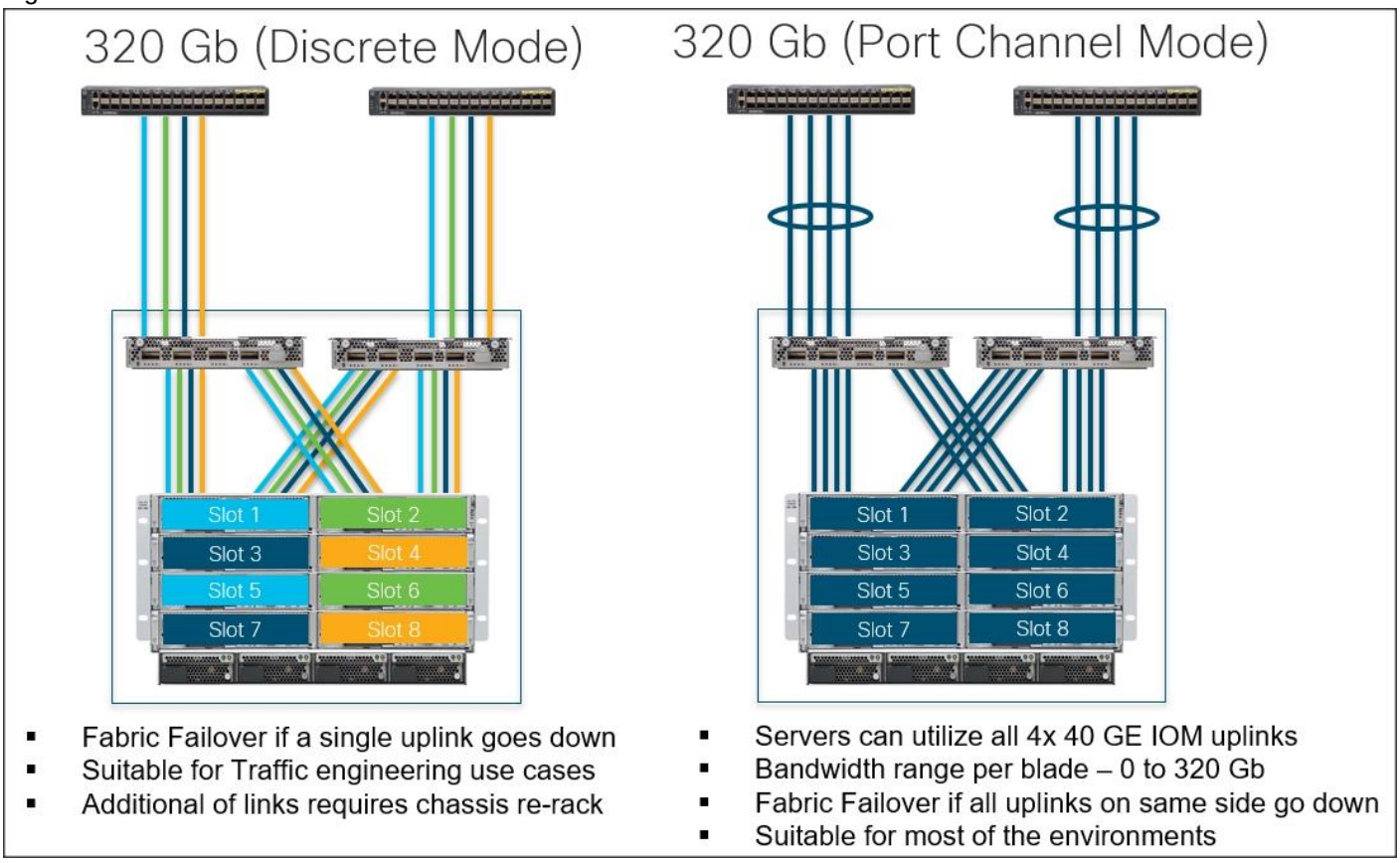
3. Click OK.



Fabric Ports: Discrete vs. Port Channel Mode

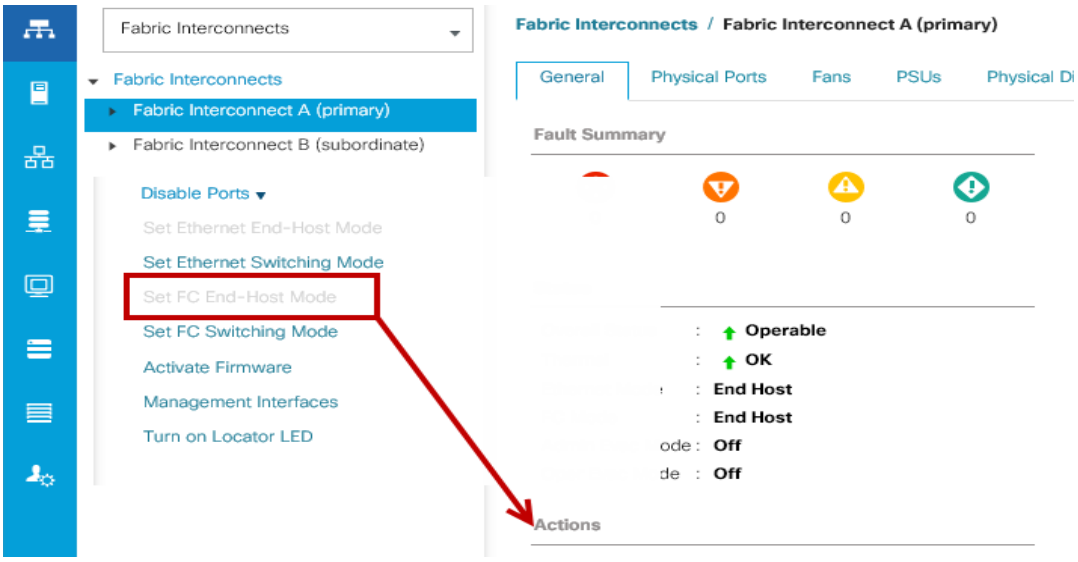
Figure 26illustrates the advantage of Discrete Vs Port-Channel mode in UCSM.

Figure 26 Port Channel versus Discrete Mode



Set Fabric Interconnects to Fibre Channel End Host Mode

In order to configure FC Uplink ports connected to Cisco UCS MDS 9148S FC switch set the Fabric Interconnects to the Fibre Channel End Host Mode. Verify that Fabric Interconnects are operating in “FC End-Host Mode.”

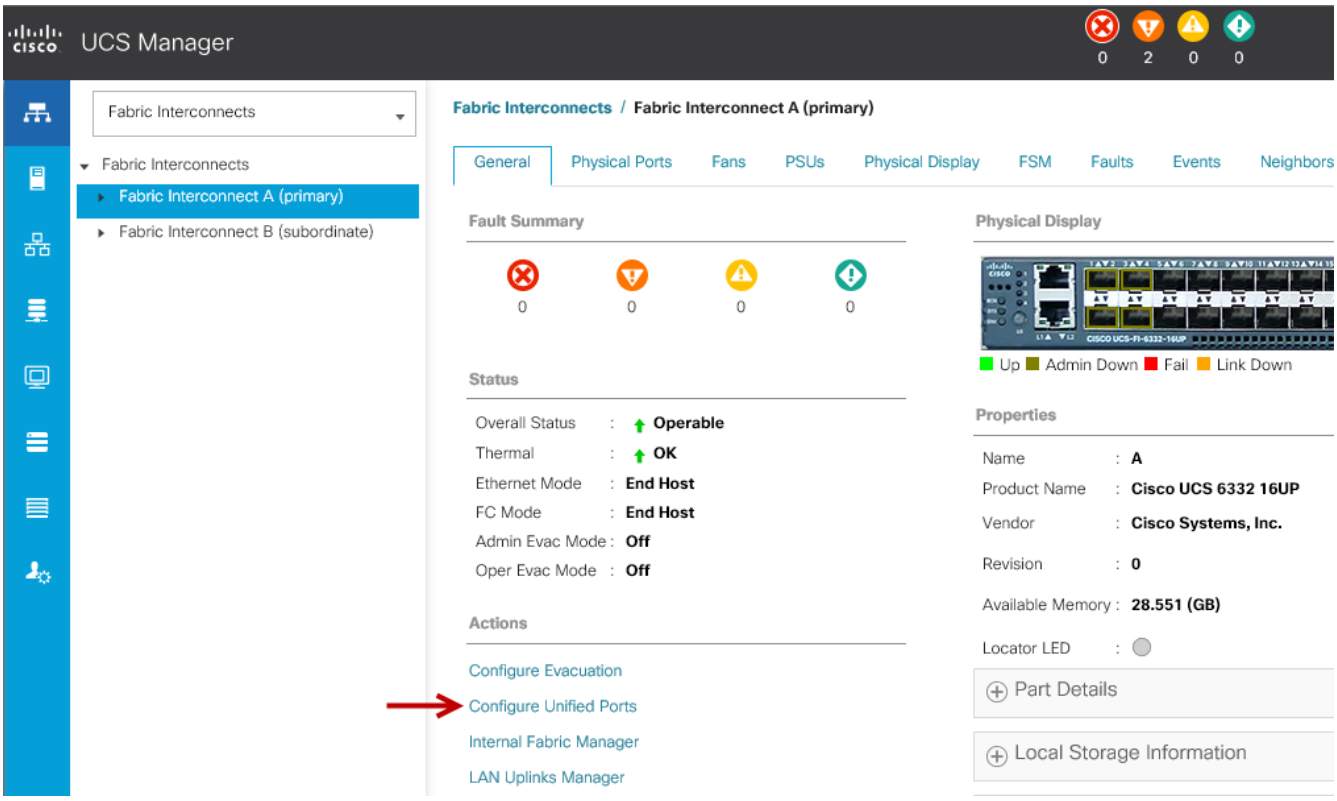


Fabric Interconnect automatically reboot if switched operational mode; perform this task on one FI first, wait for FI to come up and follow the same on second FI.

Configure FC SAN Uplink Ports

To configure Fibre Channel Uplink ports, complete the following steps:

1. Go to Equipment > Fabric Interconnects > Fabric Interconnect A > General tab > Actions pane, click Configure Unified ports.



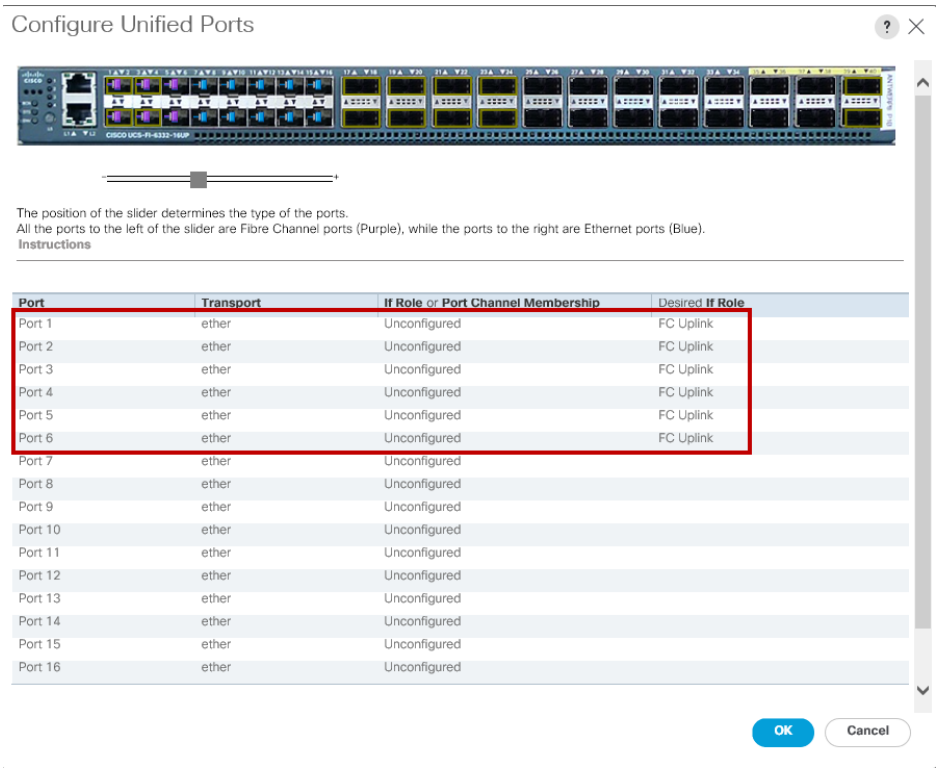
2. Click Yes to confirm in the pop-up window.

3. Move the slider to the right.

4. Click OK.

Ports to the right of the slider will become FC ports. For our study, we configured the first six ports on the FI as FC Uplink ports.

Applying this configuration will cause the immediate reboot of Fabric Interconnect and/or Expansion Module(s).

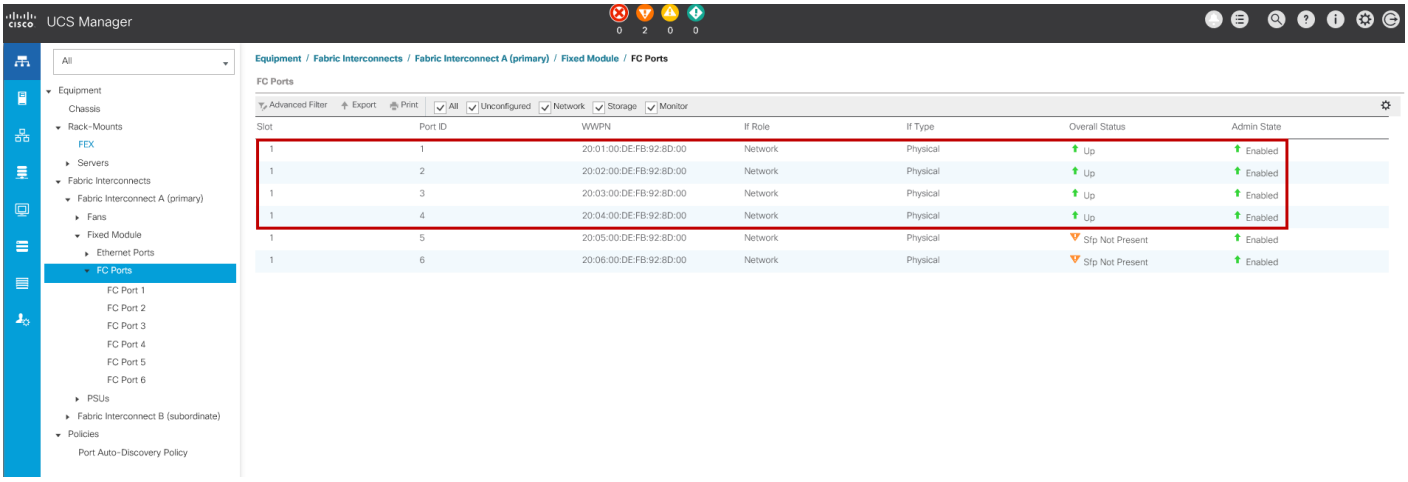


5. Click Yes to apply the changes.

6. After the FI reboot, your FC Ports configuration will look like Figure 27.

7. Follow the same steps on Fabric Interconnect B.

Figure 27 FC Uplink Ports on Fabric Interconnect A



Configure Server Ports

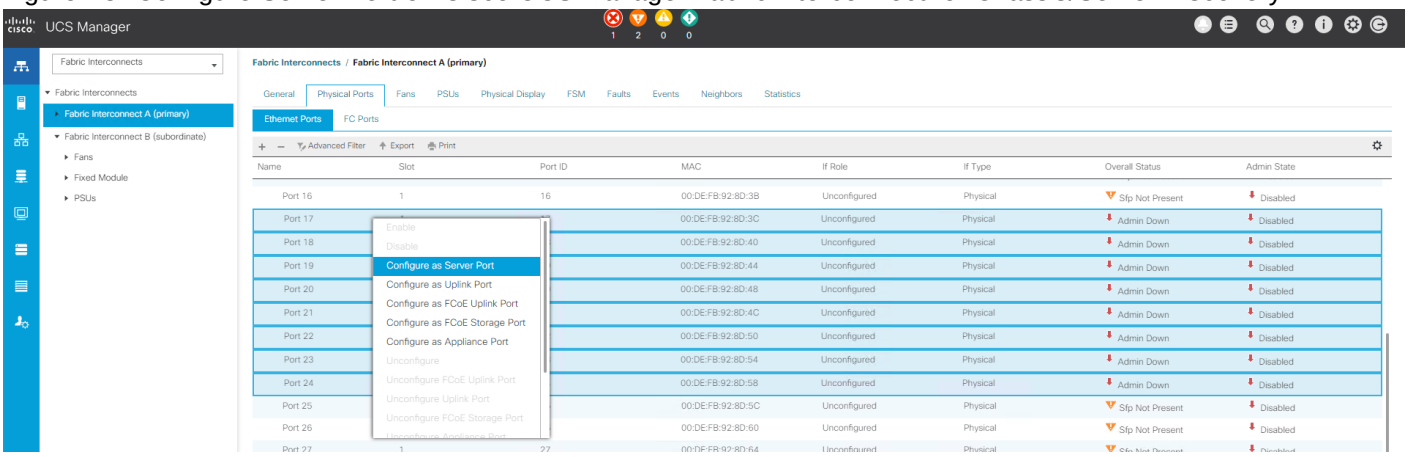
Configure Server Ports to initiate Chassis and Blade discovery. To configure server ports, complete the following steps:

1. Go to Equipment > Fabric Interconnects > Fabric Interconnect A > Fixed Module > Ethernet Ports.

2. Select the ports (for this solution ports are 17-24) which are connected to the Cisco IO Modules of the two B-Series 5108 Chassis.

3. Right-click and select “Configure as Server Port.”

Figure 28 Configure Server Port on Cisco UCS Manager Fabric Interconnect for Chassis/Server Discovery

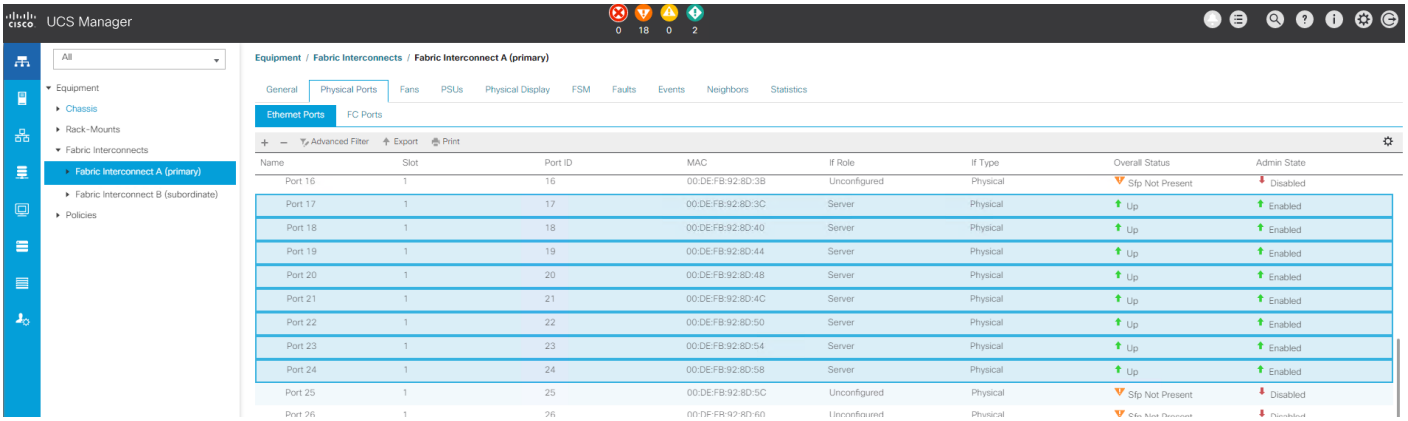


4. Click Yes to confirm and click OK.

5. Perform the same steps to configure “Server Port” on Fabric Interconnect B.

When configured, the server port will look like Figure 29 on both Fabric Interconnects.

Figure 29 Server Ports on Fabric Interconnect A

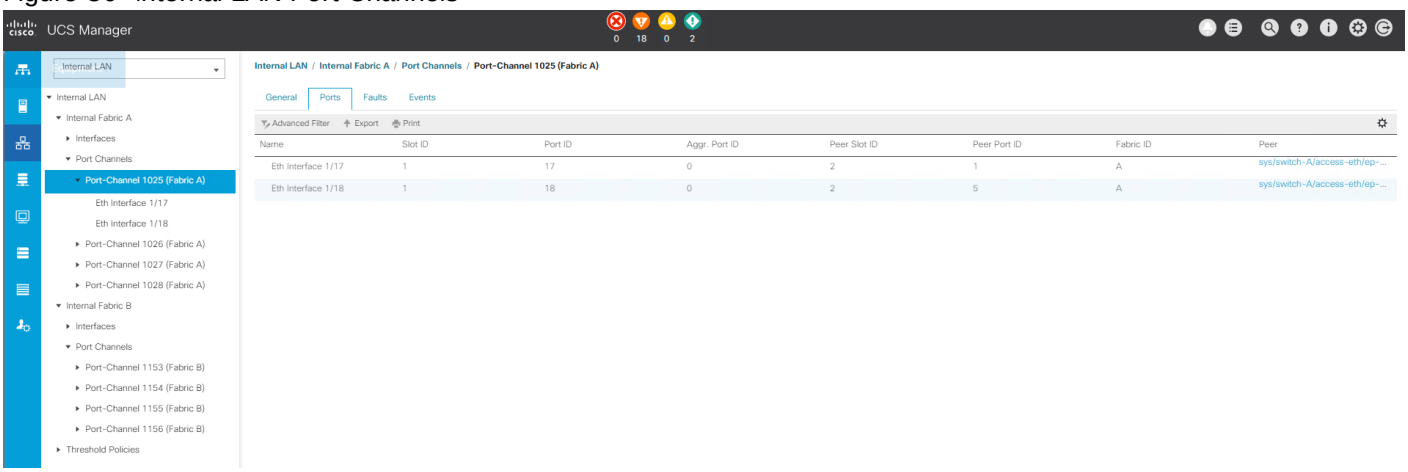


6. After configuring Server Ports, acknowledge both the Chassis. Go to Equipment >Chassis > Chassis 1 > General > Actions > select “Acknowledge Chassis”. Similarly, acknowledge the chassis 2-4.

7. After acknowledging both the chassis, re-acknowledge all the servers placed in the chassis. Go to Equipment > Chassis 1 > Servers > Server 1 > General > Actions > select Server Maintenance > select option “Re-acknowledge” and click OK. Repeat this process to re-acknowledge all eight Servers.

8. When the acknowledgement of the Servers is completed, verify the Port-channel of Internal LAN. Go to the LAN tab > Internal LAN > Internal Fabric A > Port Channels as shown in Figure 30.

Figure 30 Internal LAN Port Channels



Configure Ethernet LAN Uplink Ports

To configure network ports used to uplink the Fabric Interconnects to the Cisco Nexus switches, complete the following steps:

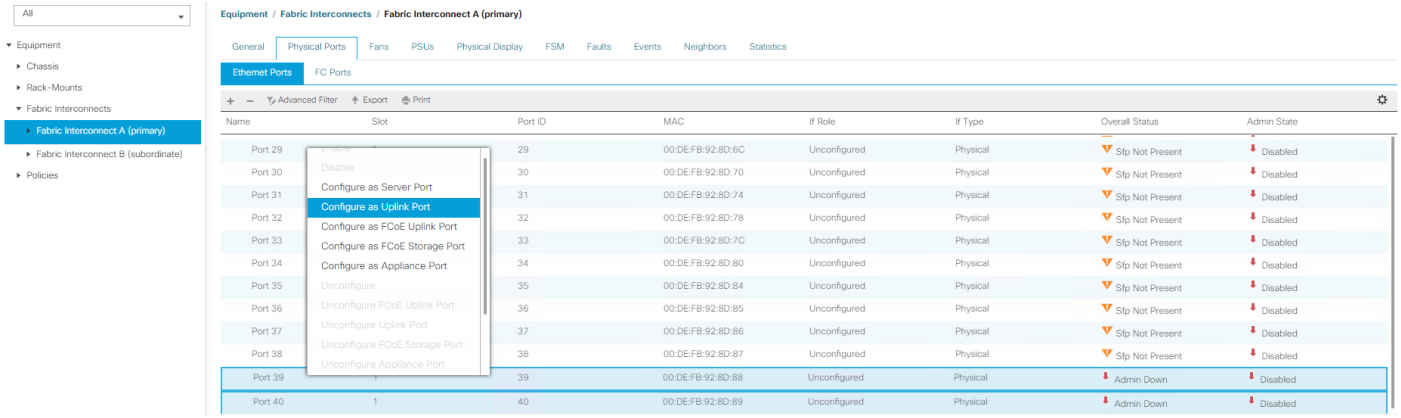
1. In Cisco UCS Manager, in the navigation pane, click the Equipment tab.

2. Select Equipment > Fabric Interconnects > Fabric Interconnect A > Fixed Module.

3. Expand Ethernet Ports.

4. Select ports (for this solution ports are 39-40) that are connected to the Nexus switches, right-click them, and select Configure as Network Port.

Figure 31 Network Uplink Port Configuration on Fabric Interconnect Configuration

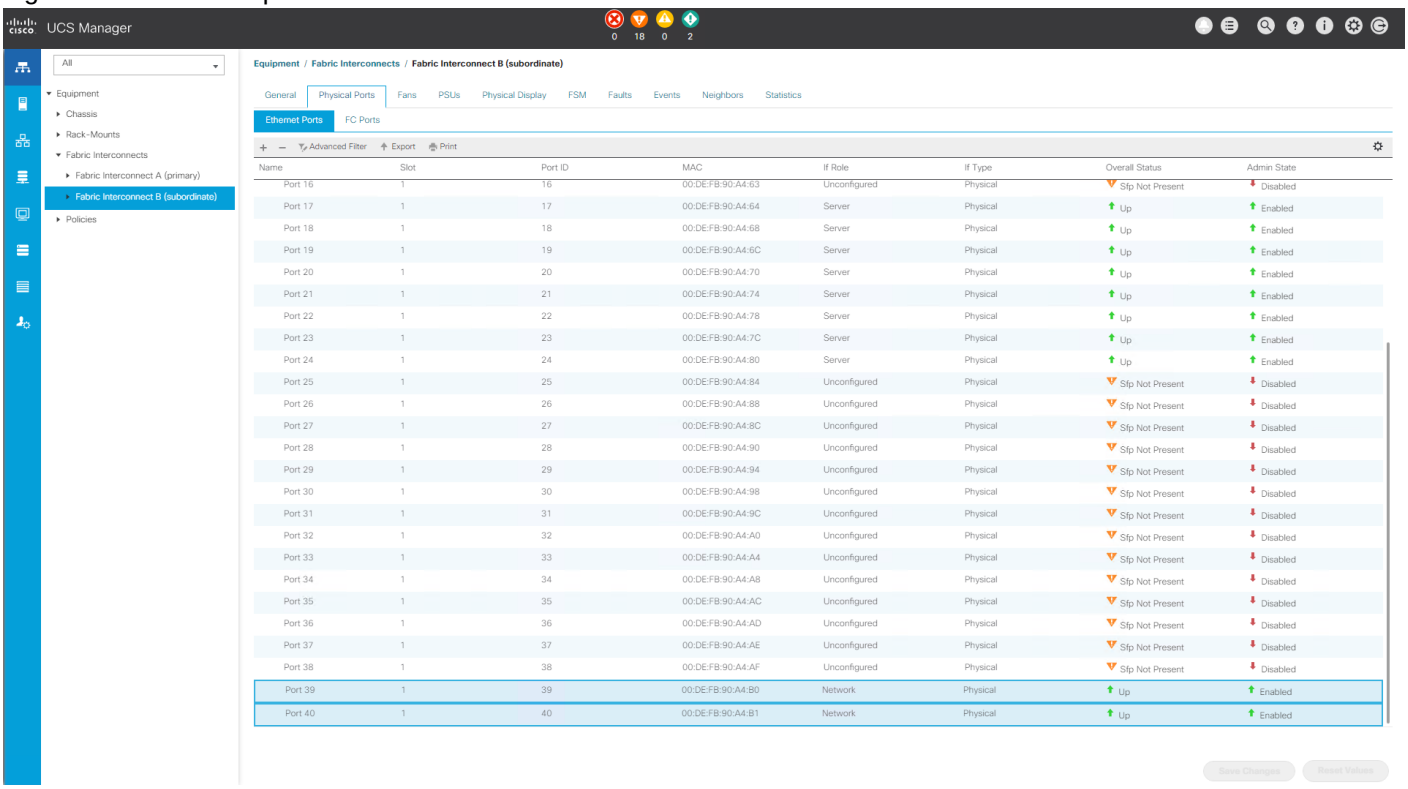


5. Click Yes to confirm ports and click OK.

6. Verify the Ports connected to Cisco Nexus upstream switches are now configured as network ports.

7. Repeat the above steps for Fabric Interconnect B. The screenshot below shows the network uplink ports for Fabric A.

Figure 32 Network Uplink Port on Fabric Interconnect



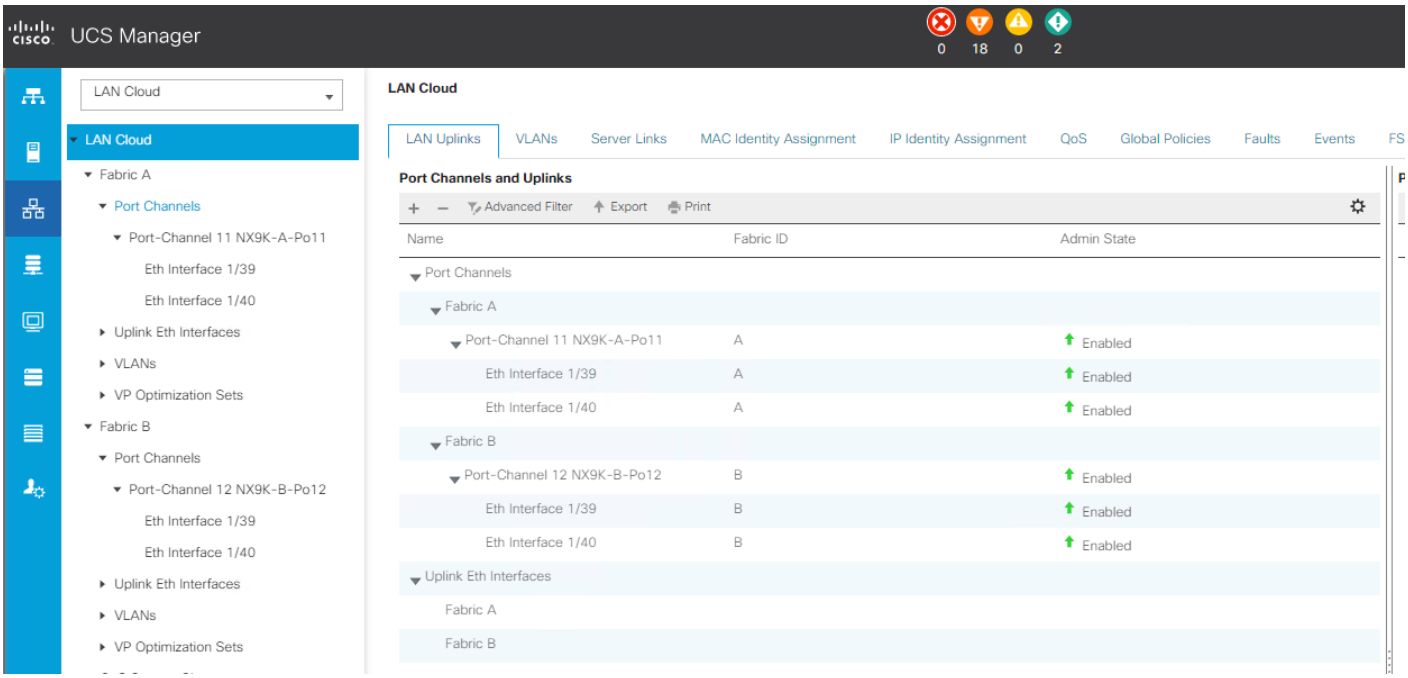
You have now created two uplink ports on each Fabric Interconnect as shown above. These ports will be used to create Virtual Port Channel in the next section.

Create Uplink Port Channels to Cisco Nexus Switches

In this procedure, two port channels were created; one from Fabric A to both Cisco Nexus 93180YC-FX switches and one from Fabric B to both Cisco Nexus 93180YC-FX switches. To configure the necessary port channels in the Cisco UCS environment, complete the following steps:

1. In Cisco UCS Manager, click the LAN tab in the navigation pane.
2. Under LAN > LAN Cloud, expand node Fabric A tree:
   a. Right-click Port Channels.
   b. Select Create Port Channel.
   c. Enter 11 as the unique ID of the port channel.

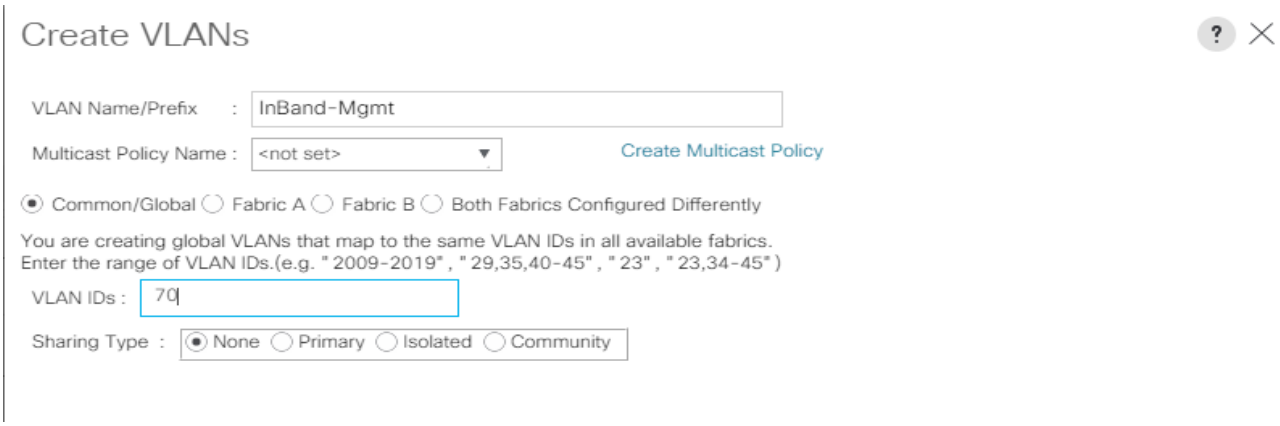
3. Enter name of the port channel.
4. Click Next.
5. Select Ethernet ports 39-40 for the port channel.
6. Click Finish.
7. Repeat steps for the Port Channel configuration on FI-B.



Configure VLAN

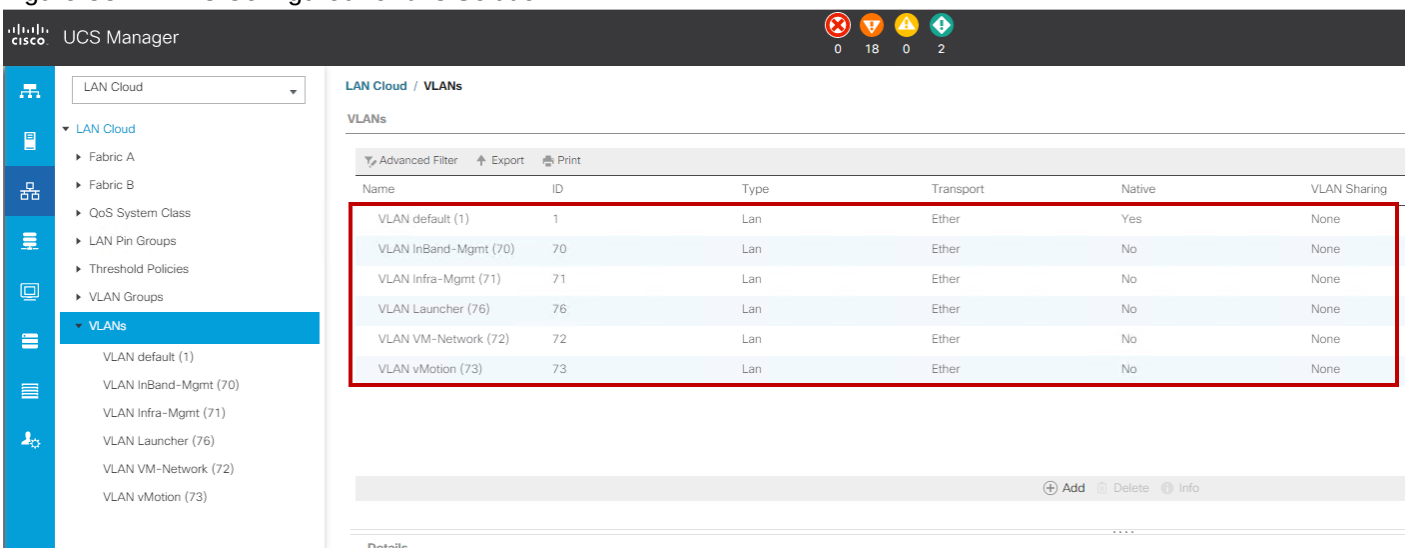
To configure the necessary virtual local area networks (VLANs) for the Cisco UCS environment, complete the following steps:

1. In Cisco UCS Manager, click the LAN tab in the navigation pane.
2. Select LAN > LAN Cloud.
3. Right-click VLANs
4. Select Create VLANs
5. Enter Public_Traffic as the name of the VLAN to be used for Public Network Traffic.
6. Keep the Common/Global option selected for the scope of the VLAN.
7. Enter 134 as the ID of the VLAN ID.
8. Keep the Sharing Type as None.



9. Follow the steps above to create required VLANs. Figure33 shows VLANs configured for this solution.

Figure 33 VLANs Configured for this Solution



It is very important to create both VLANs as global across both fabric interconnects. This makes sure the VLAN identity is maintained across the fabric interconnects in case of NIC failover.

Configure VSAN

To configure the necessary virtual storage area networks (VSANs) for the Cisco UCS environment, complete the following steps:

1. In Cisco UCS Manager, click the SAN tab in the navigation pane.
2. Select SAN > SAN Cloud.
3. Under VSANs, right-click VSANs.
4. Select Create VSANs.
5. Enter the name of the VSAN.

6. Enter VSAN ID and FCoE VLAN ID

7. Click Ok.

In this solution, we created two VSANs. VSAN-A 100 and VSAN-B 101 for SAN Boot and Storage Access.

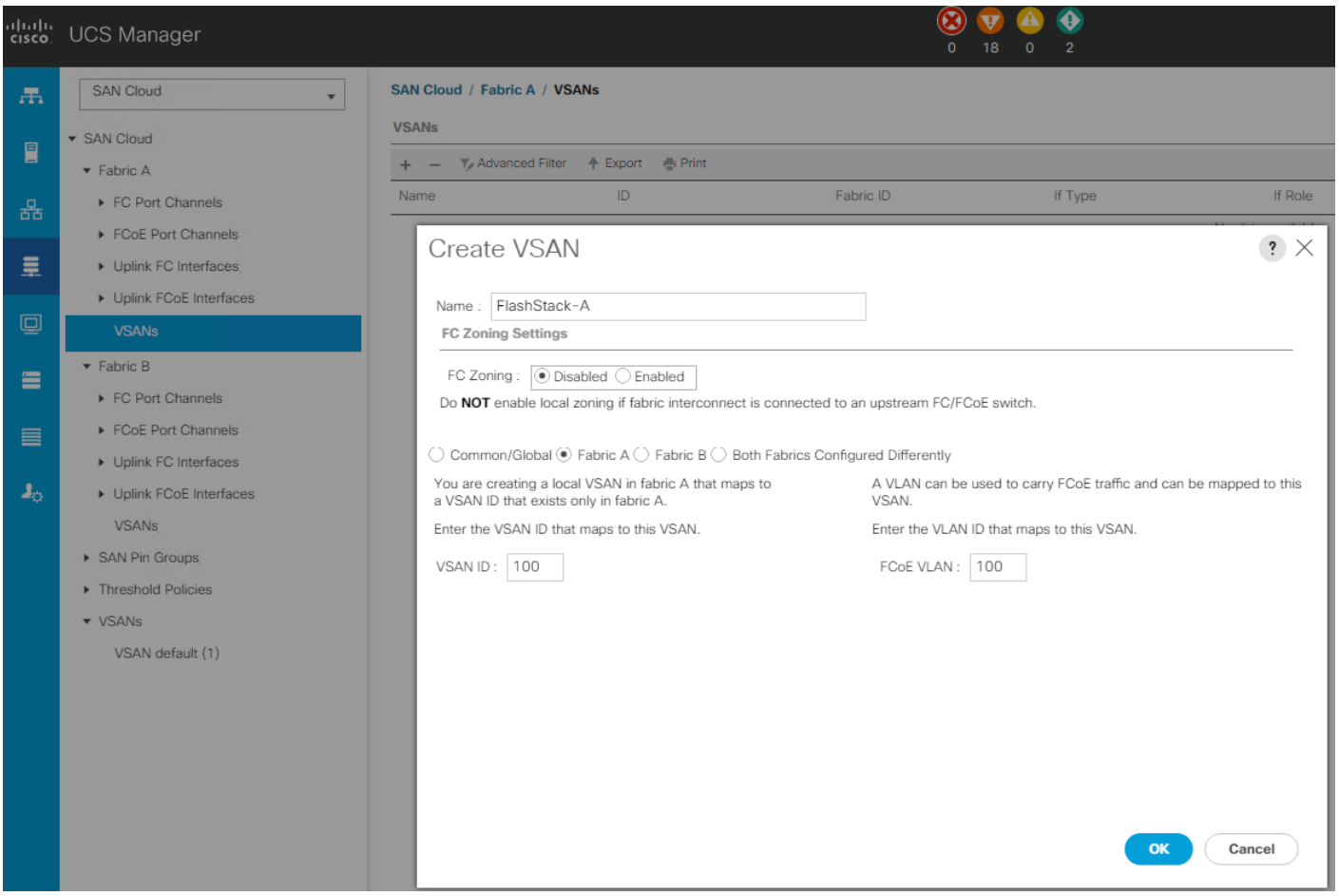
8. Select Fabric A for the scope of the VSAN:

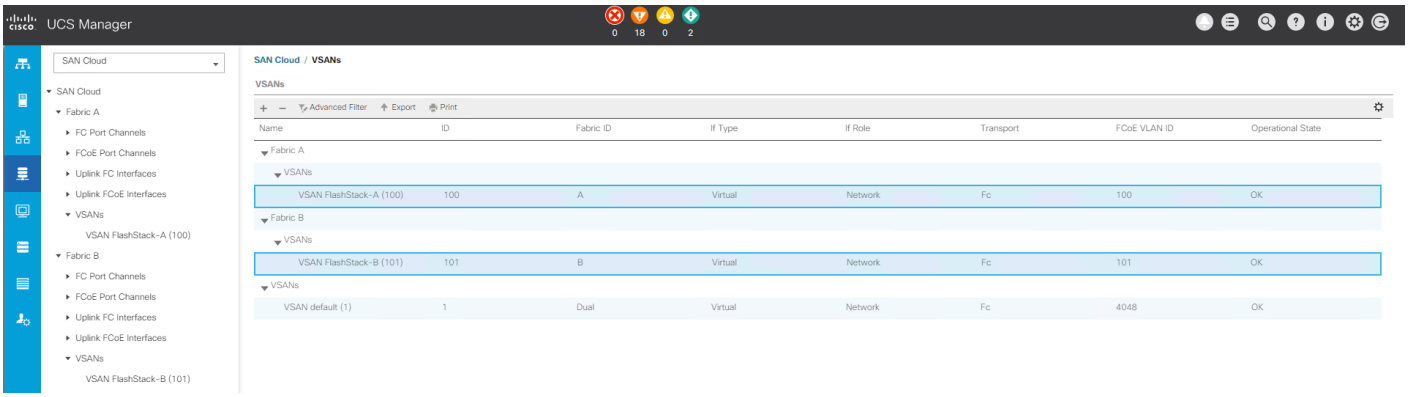
   a. Enter 100 as the ID of the VSAN.

   b. Click OK and then click OK again.

9. Repeat the above steps to create the VSANs necessary for this solution.

VSAN 100 and 101 are configured as shown below:

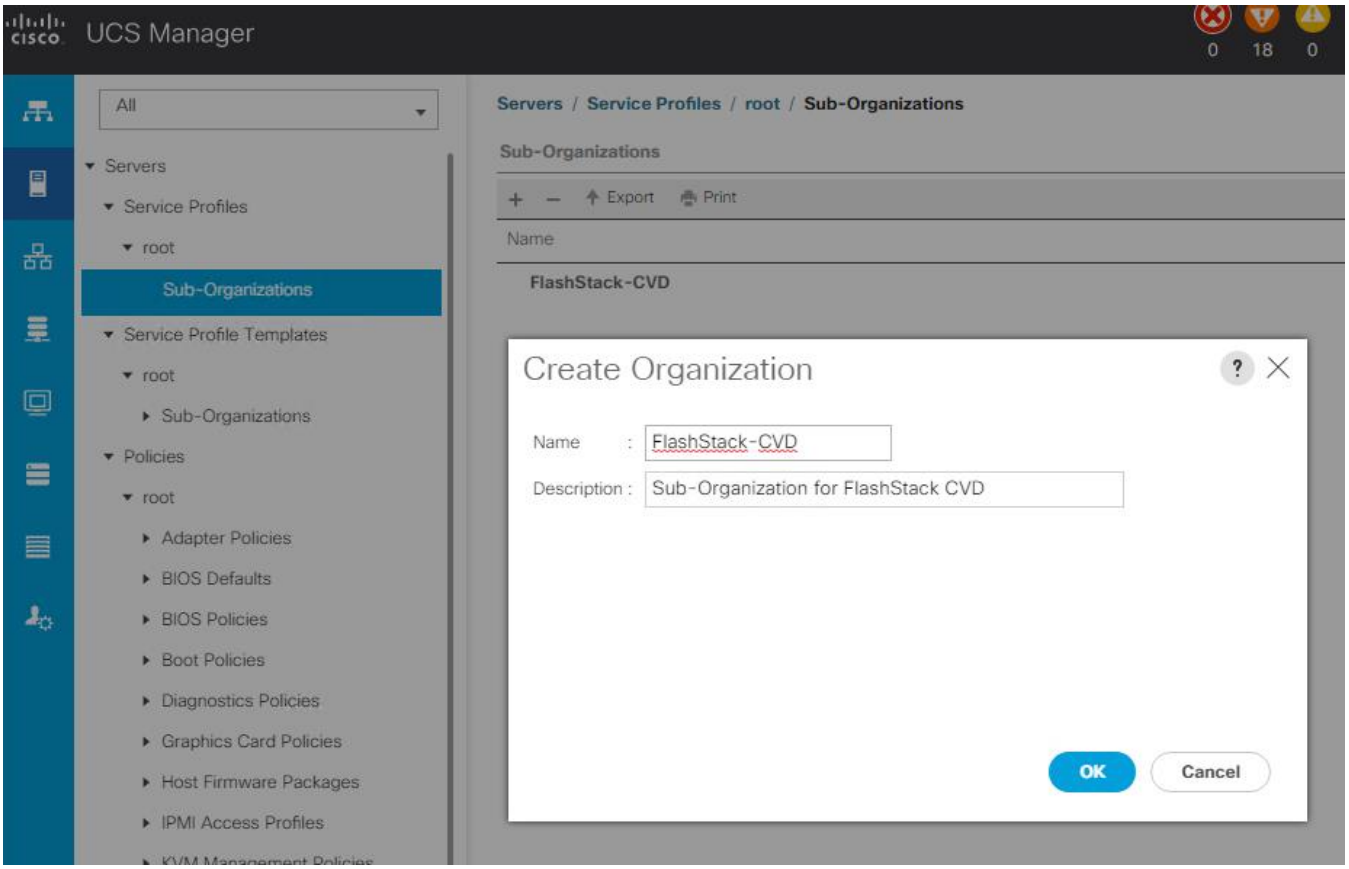




Create New Sub-Organization

To configure the necessary Sub-Organization for the Cisco UCS environment, complete the following steps:

1. In Cisco UCS Manager, click the Servers tab in the navigation pane.
2. Select root > Sub-Organization.
3. Right-click Sub-Organization.
4. Enter the name of the Sub-Organization.
5. Click Ok.



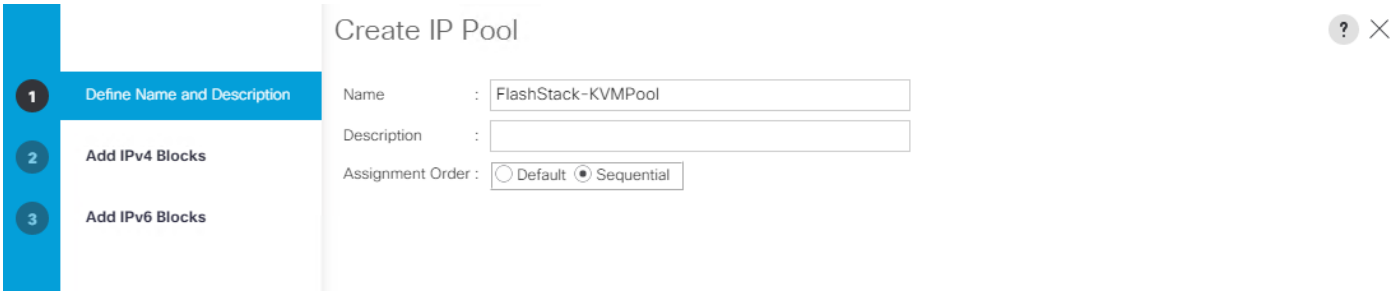
You will create pools and policies required for this solution under new “FlashStack-CVD” sub-organization created.

Configure IP, UUID, Server, MAC, WWNN, and WWPN Pools

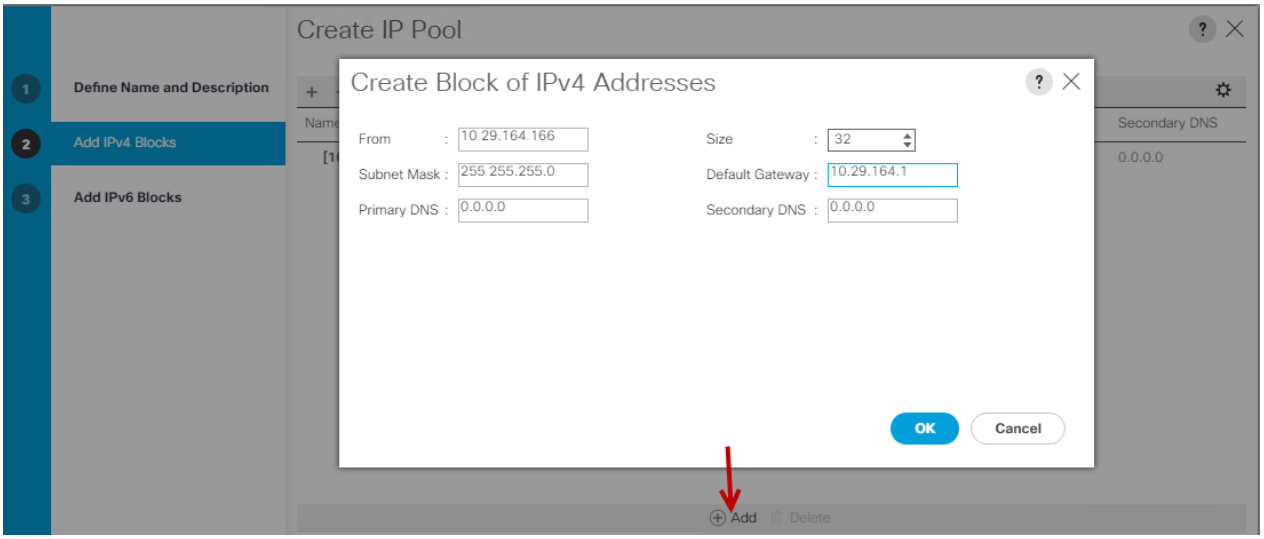
IP Pool Creation

An IP address pool on the out of band management network must be created to facilitate KVM access to each compute node in the Cisco UCS domain. To create a block of IP addresses for server KVM access in the Cisco UCS environment, complete the following steps:

1. In Cisco UCS Manager, in the navigation pane, click the LAN tab.
2. Select Pools > root > Sub-Organizations > FlashStack-CVD > IP Pools > click Create IP Pool.
3. Select option Sequential to assign IP in sequential order then click Next.



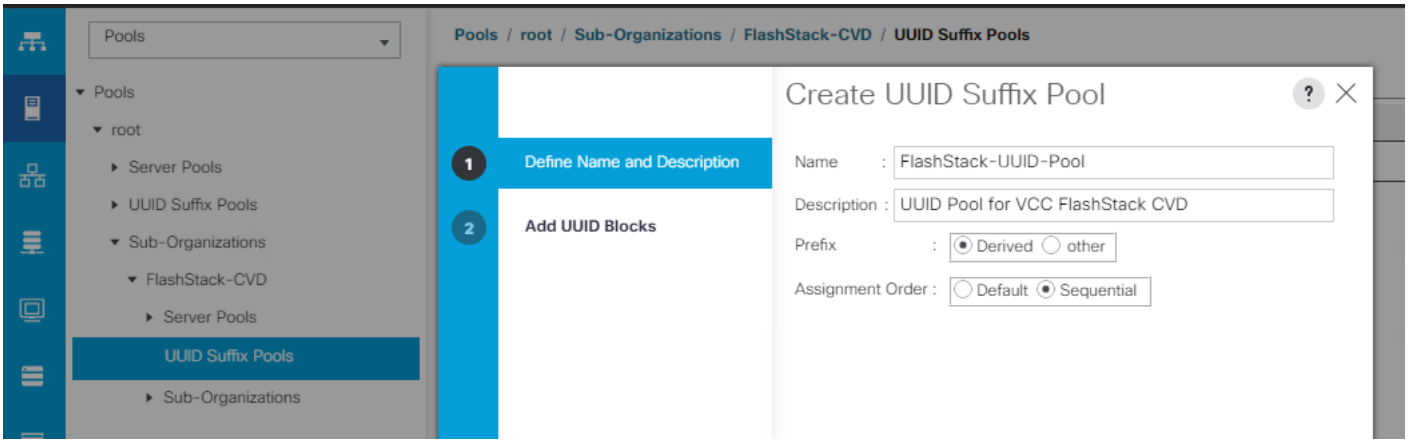
4. Click Add IPv4 Block.
5. Enter the starting IP address of the block and the number of IP addresses required, and the subnet and gateway information as shown below.



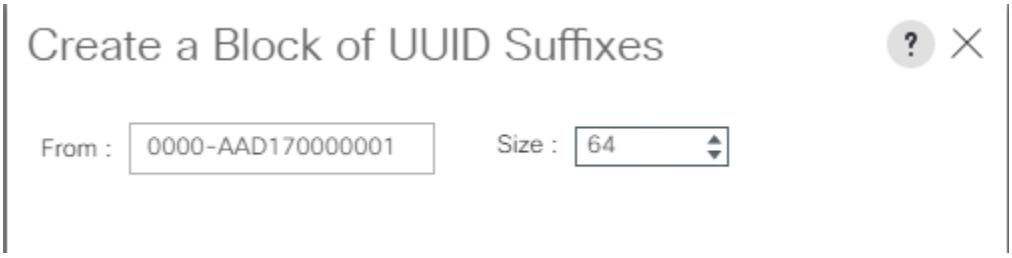
UUID Suffix Pool Creation

To configure the necessary universally unique identifier (UUID) suffix pool for the Cisco UCS environment, complete the following steps:

1. In Cisco UCS Manager, click the Servers tab in the navigation pane.
2. Select Pools > root > Sub-Organization > FlashStack-CVD
3. Right-click UUID Suffix Pools and then select Create UUID Suffix Pool.
4. Enter the name of the UUID name.
5. Optional: Enter a description for the UUID pool.
6. Keep the prefix at the derived option and select Sequential in as Assignment Order then click Next.



7. Click Add to add a block of UUIDs.
8. Create a starting point UUID as per your environment.
9. Specify a size for the UUID block that is sufficient to support the available blade or server resources.



Server Pool Creation

To configure the necessary server pool for the Cisco UCS environment, complete the following steps:

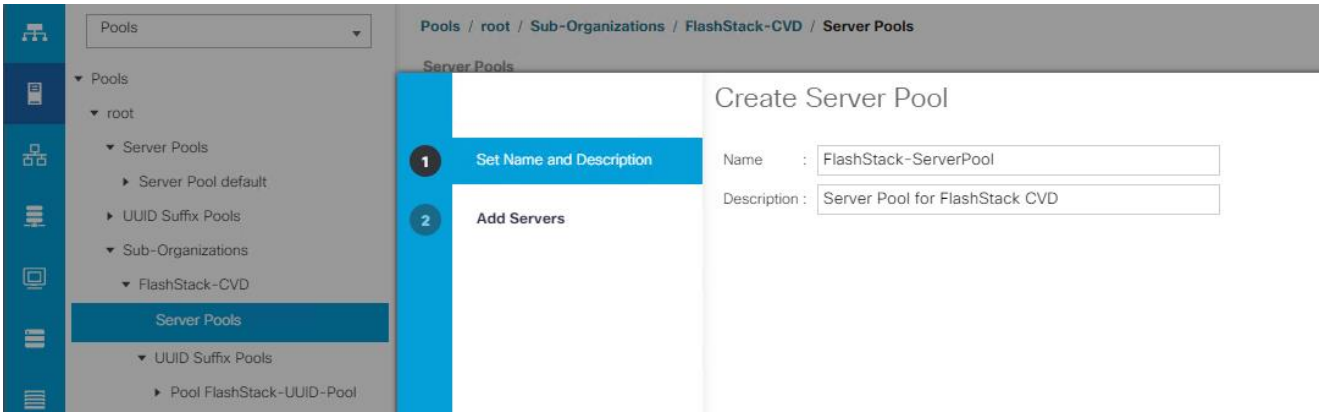
Consider creating unique server pools to achieve the granularity that is required in your environment.

1. In Cisco UCS Manager, click the Servers tab in the navigation pane.

2. Select Pools > root > Sub-Organization > FlashStack-CVD > right-click Server Pools > Select Create Server Pool.

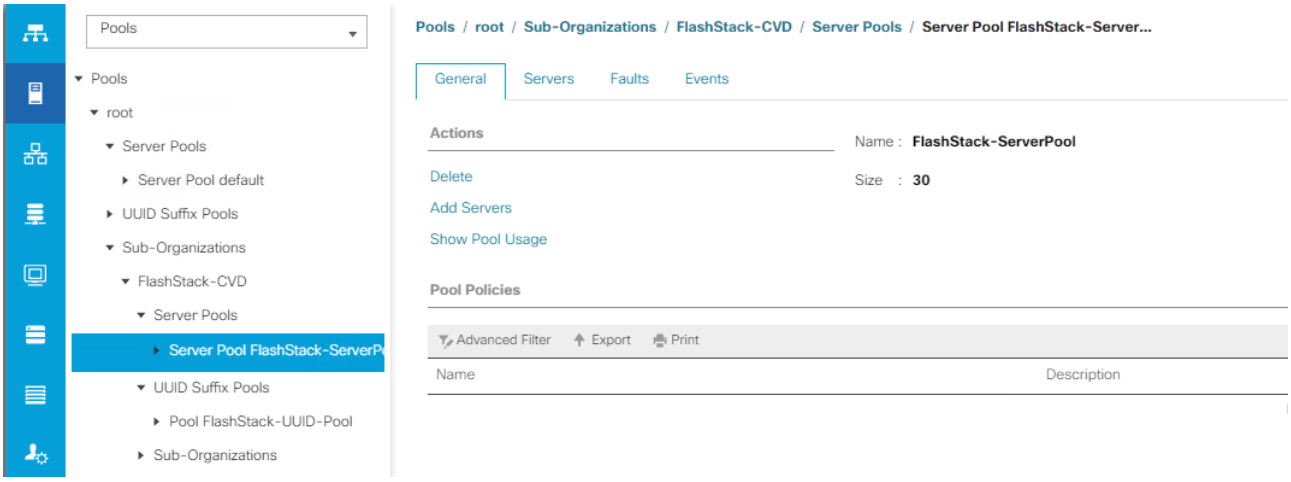
3. Enter name of the server pool.

4. Optional: Enter a description for the server pool then click Next.



5. Select servers to be used for the deployment and click > to add them to the server pool. In our case we added thirty servers in this server pool.

6. Click Finish and then click OK.



MAC Pool Creation

To configure the necessary MAC address pools for the Cisco UCS environment, complete the following steps:

1. In Cisco UCS Manager, click the LAN tab in the navigation pane.

2. Select Pools > root > Sub-Organization > FlashStack > right-click MAC Pools under the root organization.

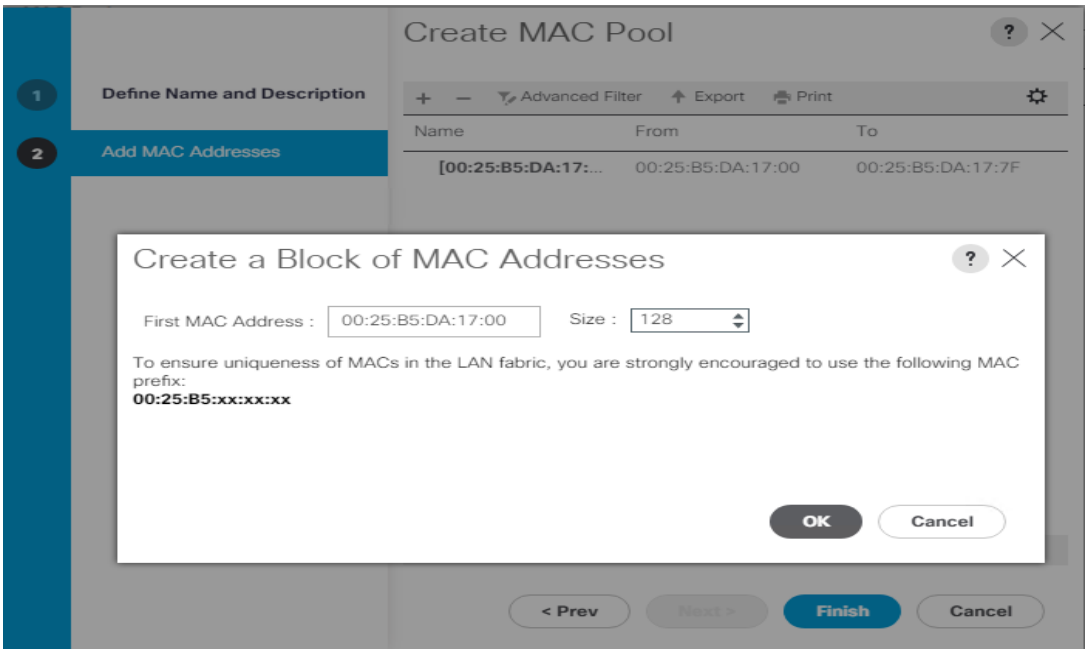
3. Select Create MAC Pool to create the MAC address pool.

4. Enter name for MAC pool. Select Assignment Order as “Sequential”.

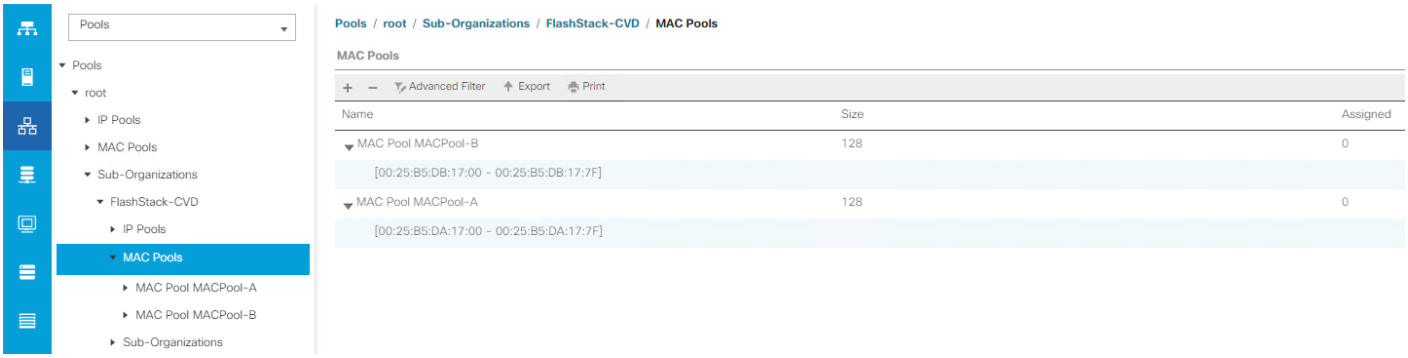
5. Enter the seed MAC address and provide the number of MAC addresses to be provisioned.

6. Click OK and then click Finish.

7. In the confirmation message, click OK.



8. Create MAC Pool B and assign unique MAC Addresses as shown below.



WWNN and WWPN Pool Creation

To configure the necessary WWNN pools for the Cisco UCS environment, complete the following steps:

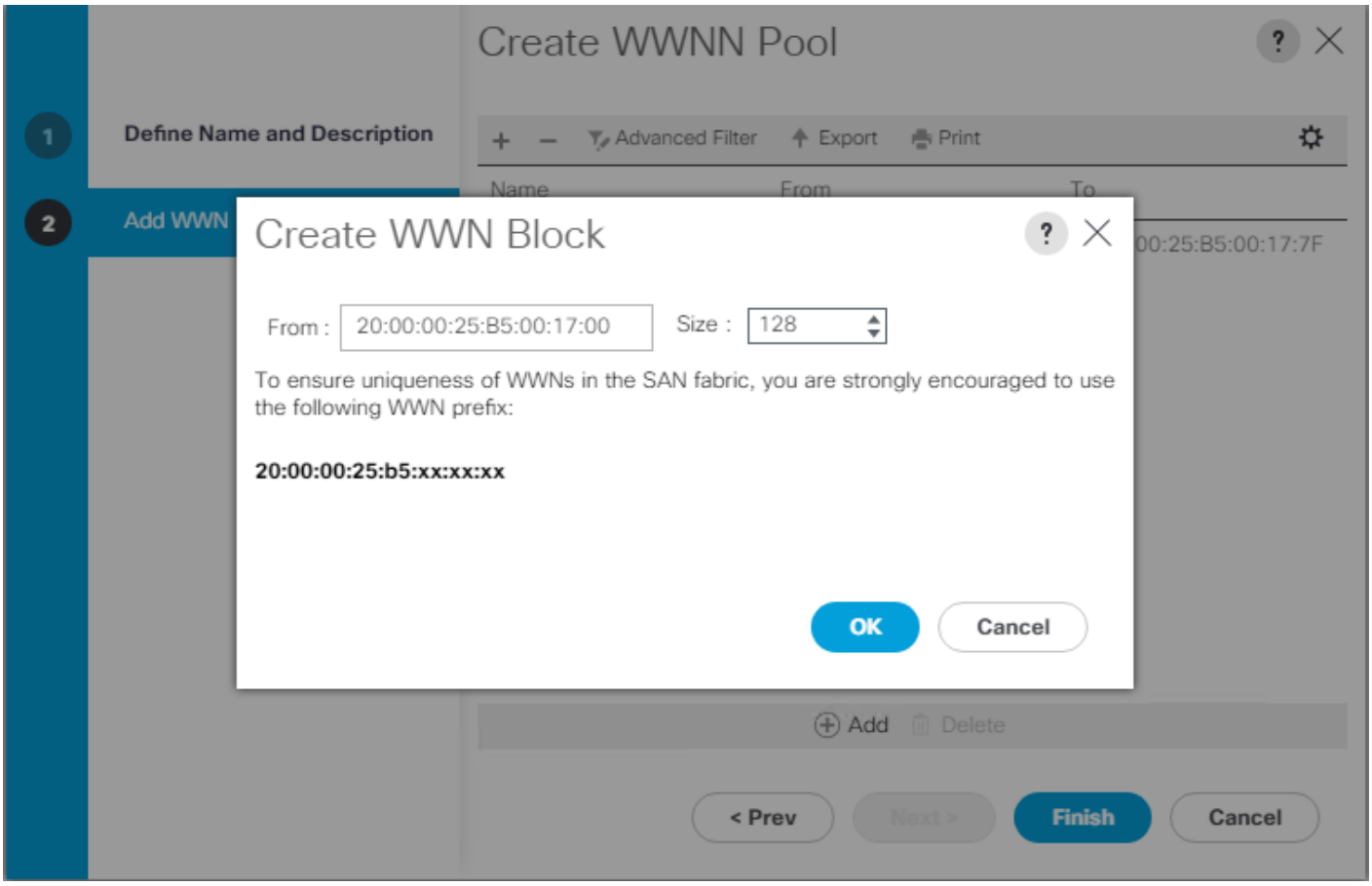
1. In Cisco UCS Manager, click the SAN tab in the navigation pane.

2. Select Pools > Root > Sub-Organization > FlashStack-CVD > WWNN Pools > right-click WWNN Pools > select Create WWNN Pool.

3. Assign name and Assignment Order as sequential.

4. Click Next and then click Add to add block of Ports.

5. Enter Block for WWN and size of WWNN Pool as shown below.



6. Click OK and then click Finish.

To configure the necessary WWPN pools for the Cisco UCS environment, complete the following steps:

We created two WWPN as WWPN-A Pool and WWPN-B as World Wide Port Name as shown below. These WWNN and WWPN entries will be used to access storage through SAN configuration.

1. In Cisco UCS Manager, click the SAN tab in the navigation pane.

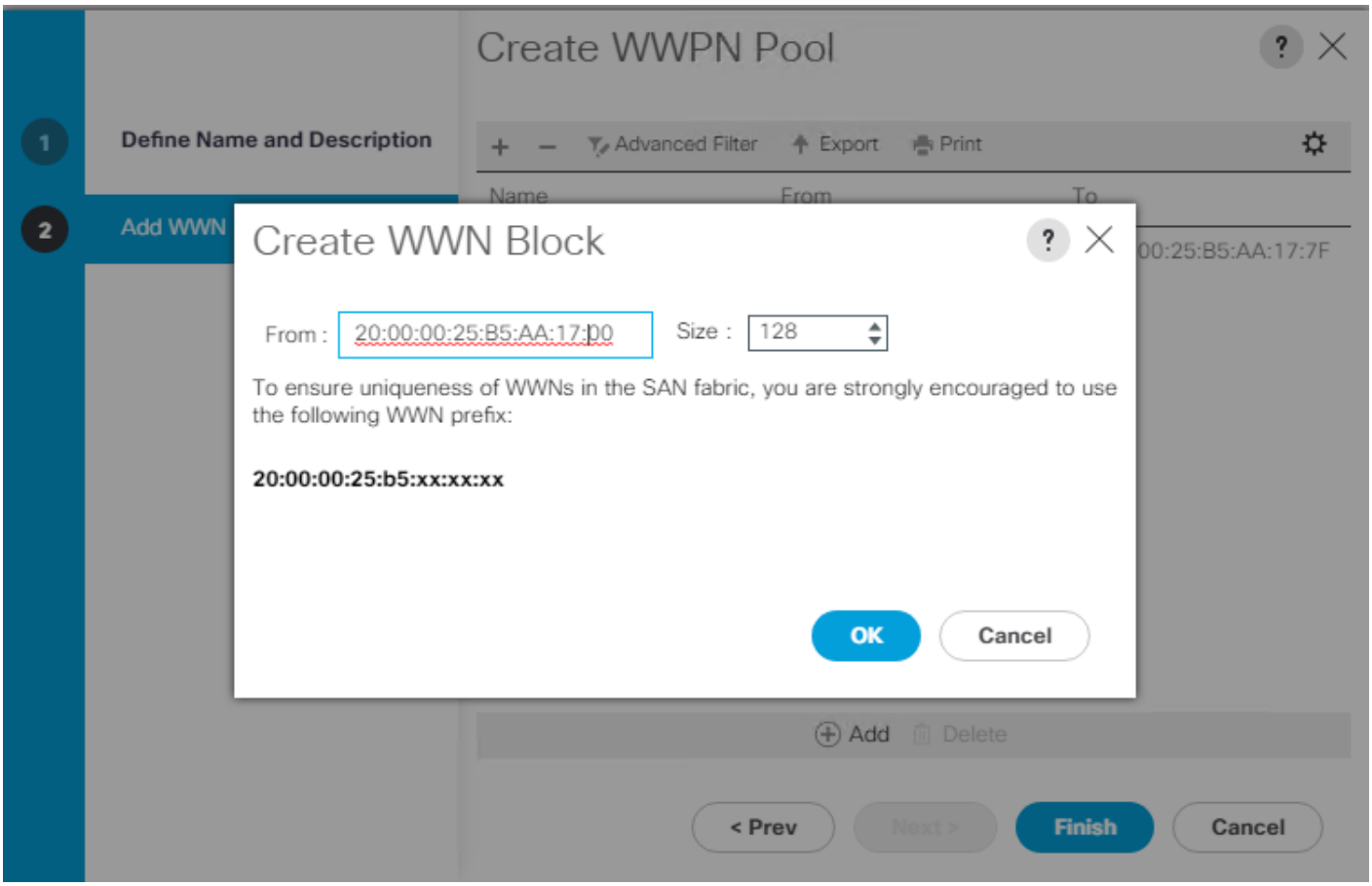
2. Select Pools > Root > WWPN Pools > right-click WWPN Pools > select Create WWPN Pool.

3. Assign name and Assignment Order as sequential.

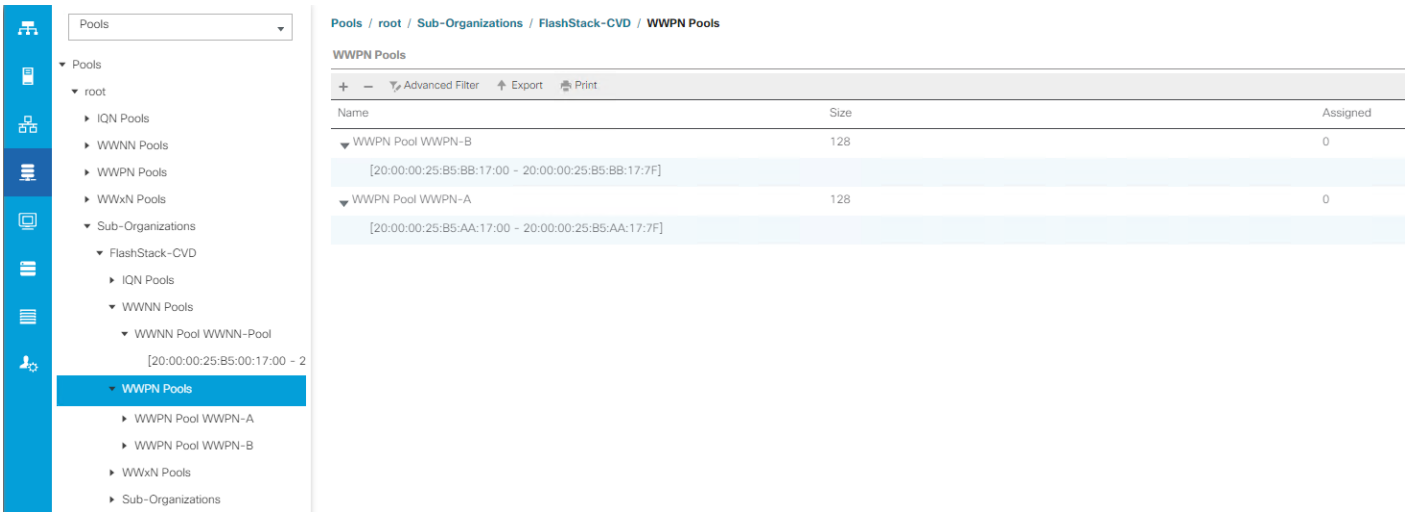
4. Click Next and then click Add to add block of Ports.

5. Enter Block for WWN and size.

6. Click OK and then click Finish.



7. Configure WWPN-Bs Pool as well and assign the unique block IDs as shown below.



Set Jumbo Frames in both the Cisco Fabric Interconnect

To configure jumbo frames and enable quality of service in the Cisco UCS fabric, complete the following steps:

1. In Cisco UCS Manager, click the LAN tab in the navigation pane.

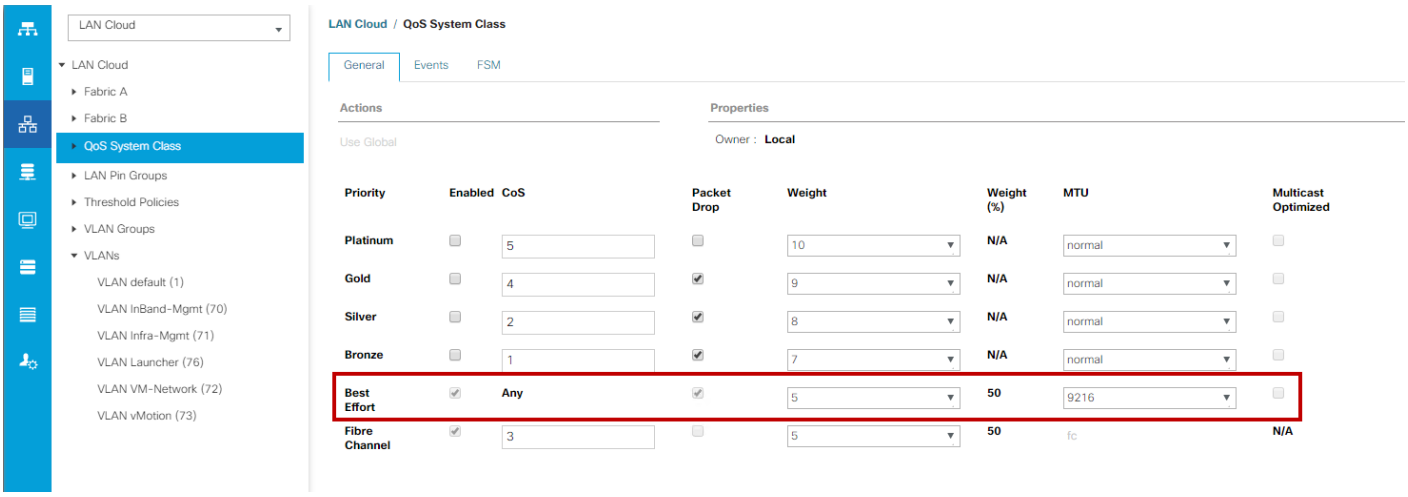
2. Select LAN > LAN Cloud > QoS System Class.

3. In the right pane, click the General tab.

4. On the Best Effort row, enter 9216 in the box under the MTU column.

5. Click Save Changes.

6. Click OK.



Create Host Firmware Package

Firmware management policies allow the administrator to select the corresponding packages for a given server configuration. These policies often include packages for adapter, BIOS, board controller, FC adapters, host bus adapter (HBA) option ROM, and storage controller properties.

To create a firmware management policy for a given server configuration in the Cisco UCS environment, complete the following steps:

1. In Cisco UCS Manager, click the Servers tab in the navigation pane.

2. Select root > Sub-Organization > FlashStack-CVD > Host Firmware Packages.

3. Right-click Host Firmware Packages.

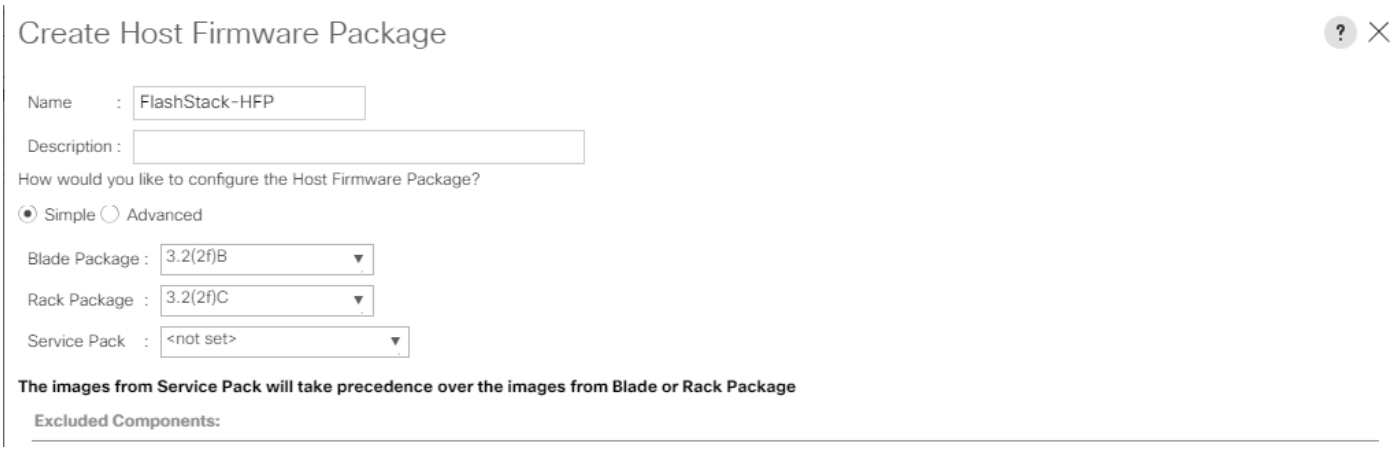
4. Select Create Host Firmware Package.

5. Enter name of the host firmware package.

6. Leave Simple selected.

7. Select the version 3.2(2f) for both the Blade Package.

8. Click OK to create the host firmware package.



Create Server Pool Policy

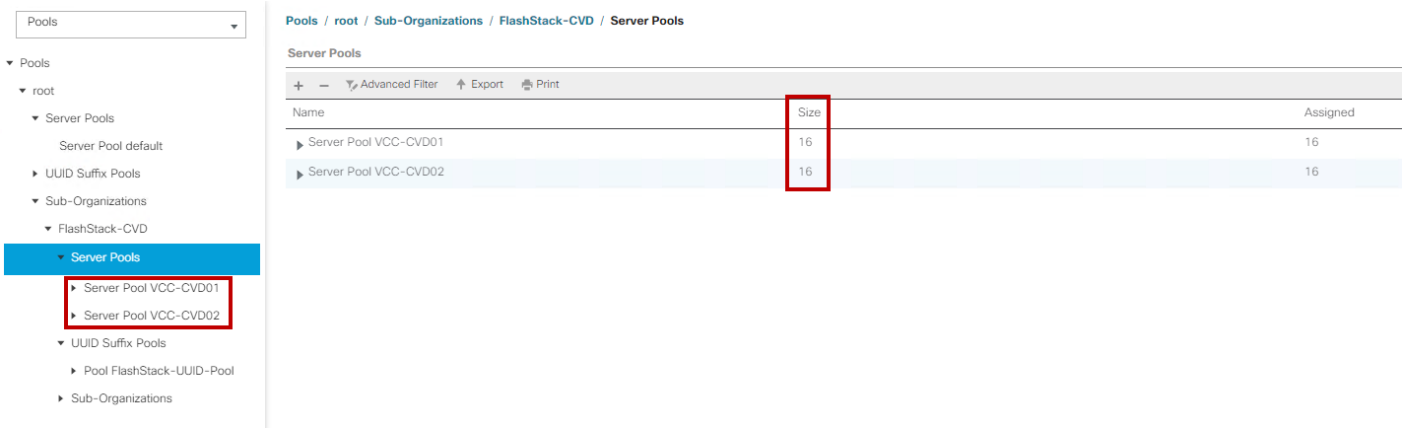
Create Server Pools Policy

Creating the Server Pool Policy requires to create Server Pool Policy and Server Pool Qualification Policy.

To create a Server Pools Policy complete following steps:

1. In Cisco UCS Manager, click the Servers tab in the navigation pane.
2. Select Pools > root > Sub-Organization > FlashStack-CVD > Server Pools.
3. Right-click Server Pools Select Create Server Pools Policy; Enter Policy name.
4. Select server from left pane to add as pooled server.

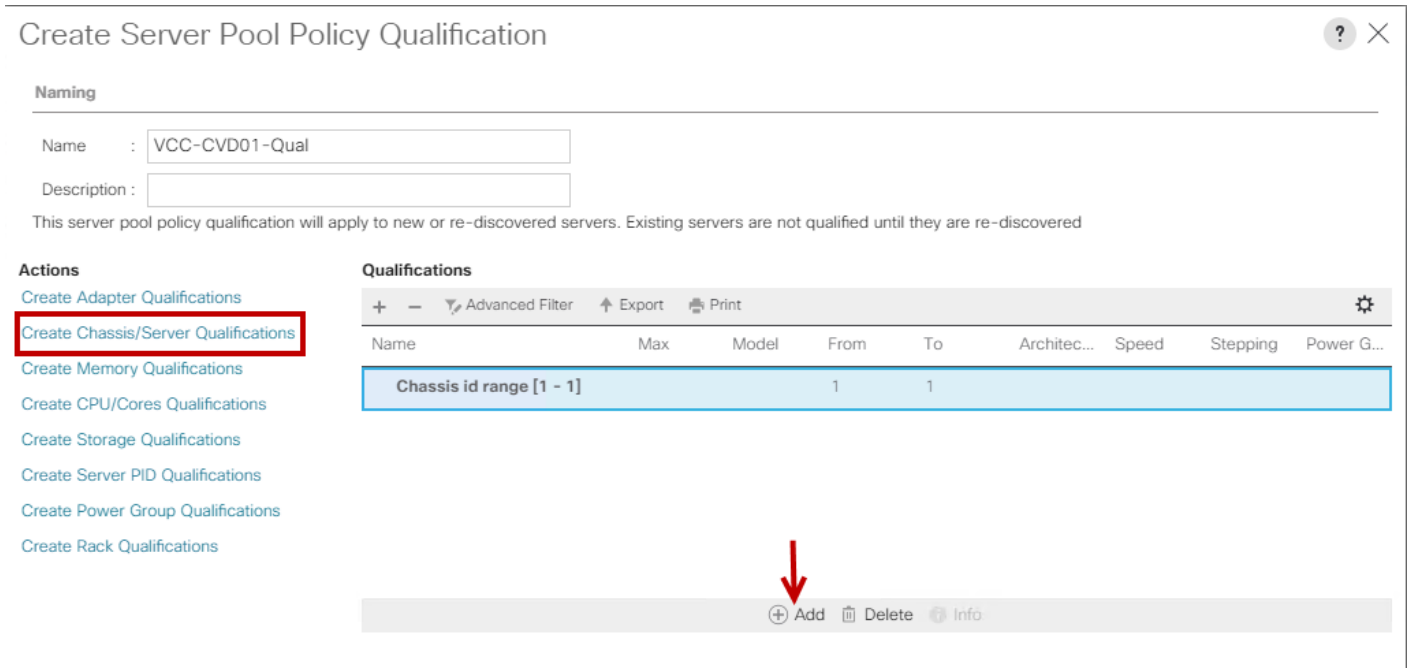
In our case we create two server Pools Policy. For “VCC-CVD01” policy we added Servers as Chassis 1 Slot 1-8 and Chassis 3 Slot 1-8 and for “VCC-CVD02” policy we added Chassis 2 Slot 1-8 and Chassis 4 Slot 1-8.



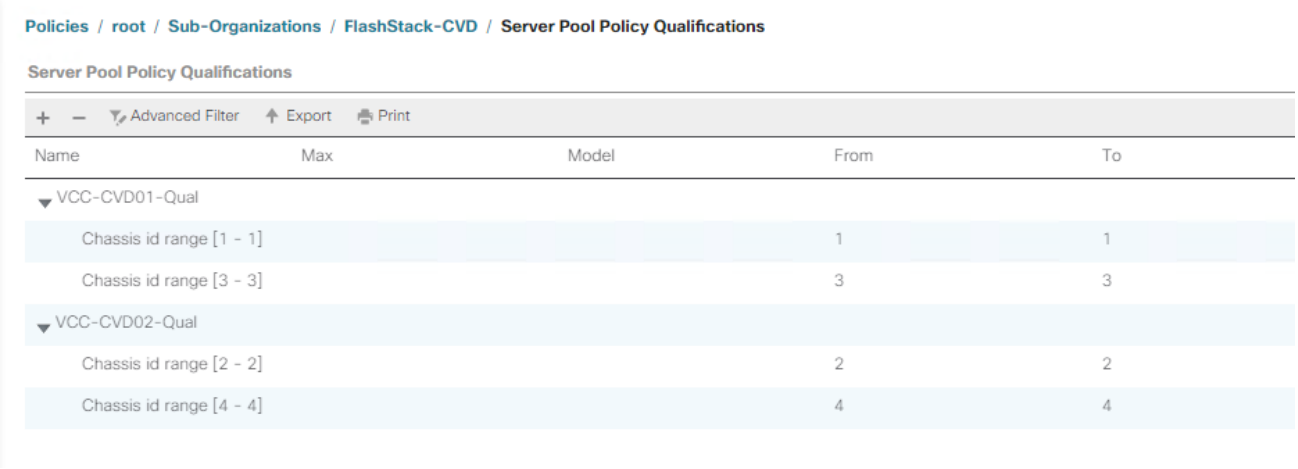
Create Server Pool Policy Qualifications

To create a Server Pool Policy Qualification Policy complete following steps:

1. In Cisco UCS Manager, click the Servers tab in the navigation pane.
2. Select Pools > root > Sub-Organization > FlashStack-CVD > Server Pool Policy Qualification.
3. Right-click Server Pools Select Create Server Pool Policy Qualification; Enter Policy name.
4. Select Chassis/Server Qualification from left pane to add in Qualifications.
5. Click Add or OK to either Add more servers to existing policy to Finish creation of Policy.

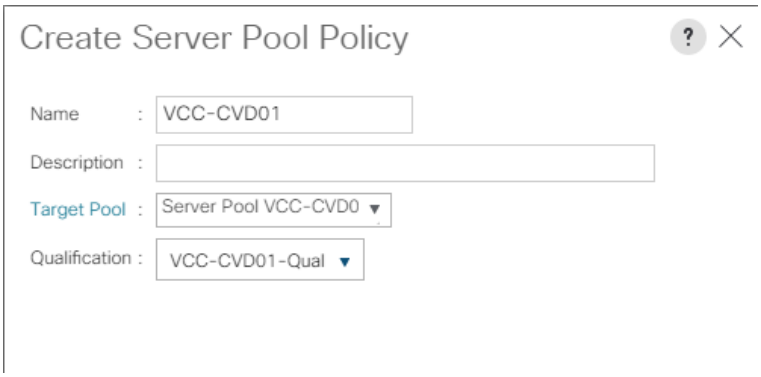


In our case we create two server Pools Policy. For “VCC-CVD01” policy we added Servers as Chassis 1 Slot 1-8 and Chassis 3 Slot 1-8 and for “VCC-CVD02” policy we added Chassis 2 Slot 1-8 and Chassis 4 Slot 1-8.



To create a Server Pool Policy complete following steps:

1. In Cisco UCS Manager, click the Servers tab in the navigation pane.
2. Select Pools > root > Sub-Organization > FlashStack-CVD > Server Pool Policies.
3. Right-click Server Pool Policies and Select Create Server Pool Policy; Enter Policy name.
4. Select Target Pool and Qualification from the drop down list.
5. Click OK.

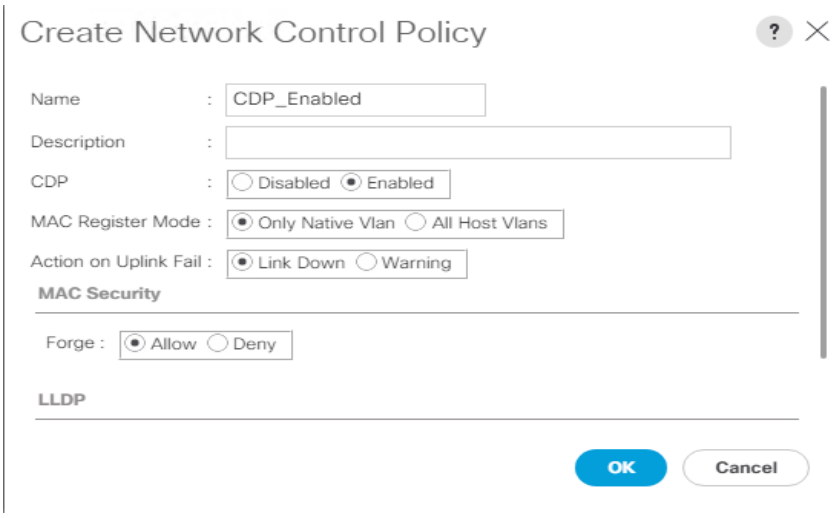


We created two Server Pool Policy to associate with Service Profile Template “VCC-CVD01” and “VCC-CVD02” as described in this section.

Create Network Control Policy for Cisco Discovery Protocol

To create a network control policy that enables Cisco Discovery Protocol (CDP) on virtual network ports, complete the following steps:

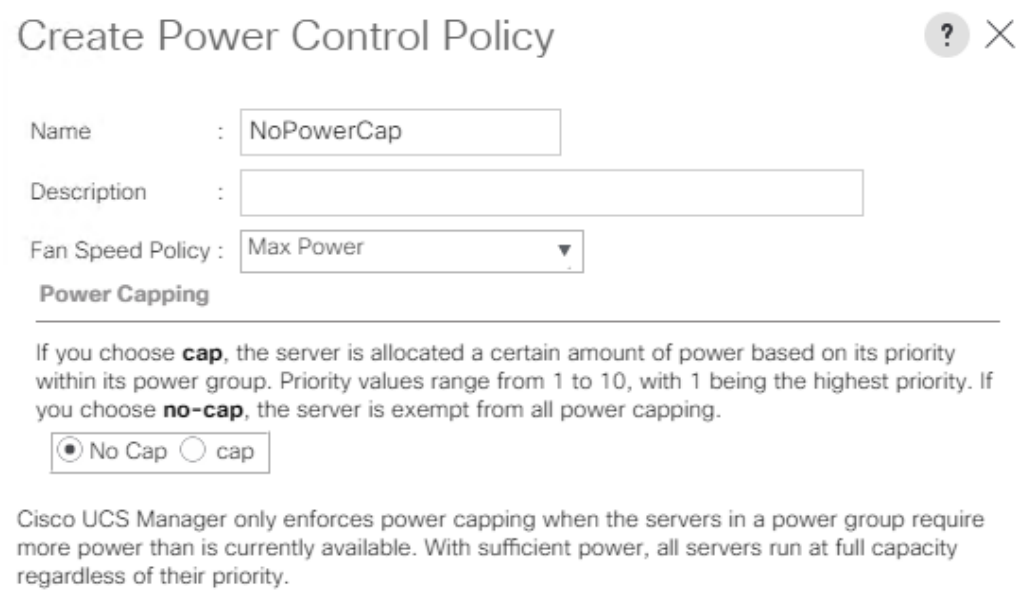
1. In Cisco UCS Manager, click the LAN tab in the navigation pane.
2. Select Policies > root > Sub-Organization > FlashStack-CVD > Network Control Policies.
3. Right-click Network Control Policies.
4. Select Create Network Control Policy.
5. Enter policy name.
6. Select the Enabled option for “CDP.”
7. Click OK to create the network control policy.



Create Power Control Policy

To create a power control policy for the Cisco UCS environment, complete the following steps:

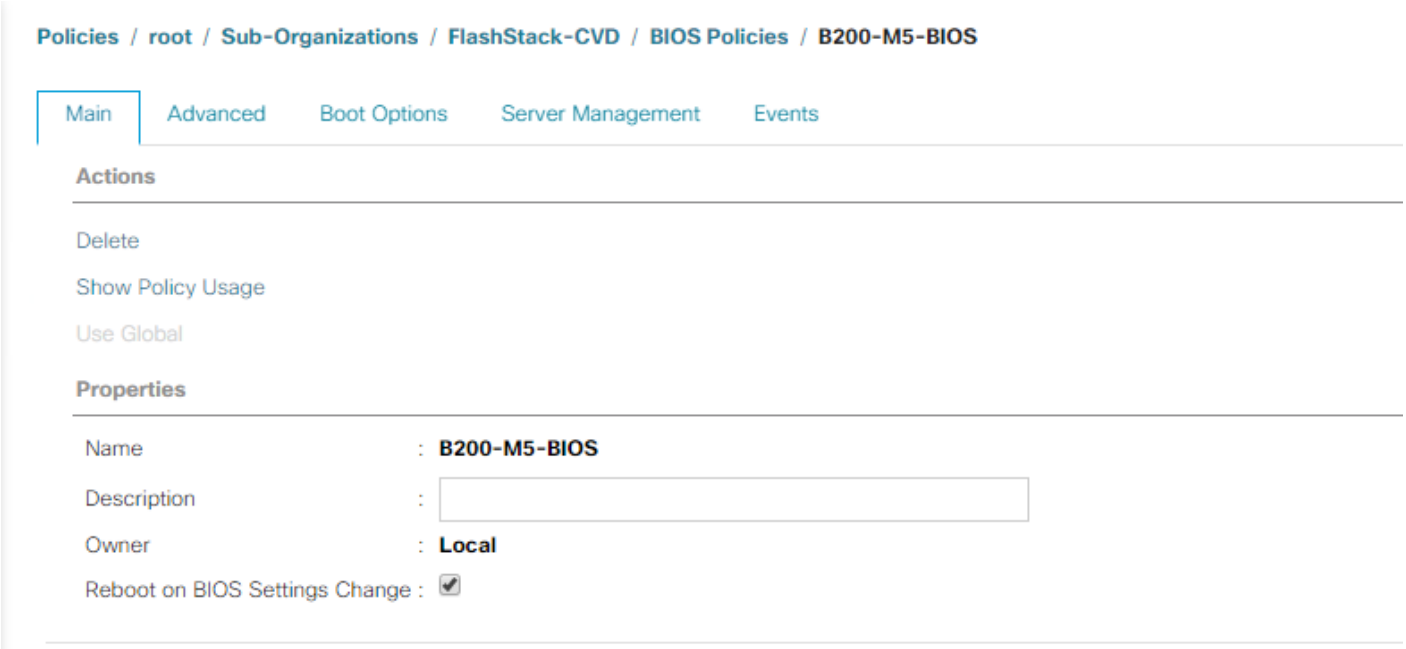
1. In Cisco UCS Manager, click the Servers tab in the navigation pane.
2. Select Policies > root > Sub-Organization > FlashStack-CVD > Power Control Policies
3. Right-click Power Control Policies.
4. Select Create Power Control Policy.
5. Select Fan Speed Policy as “Max Power”.
6. Enter NoPowerCap as the power control policy name.
7. Change the power capping setting to No Cap.
8. Click OK to create the power control policy.



Create Server BIOS Policy

To create a server BIOS policy for the Cisco UCS environment, complete the following steps:

1. In Cisco UCS Manager, click the Servers tab in the navigation pane.
2. Select Policies > root > Sub-Organization > FlashStack-CVD > BIOS Policies.
3. Right-click BIOS Policies.
4. Select Create BIOS Policy.
5. Enter B200-M5-BIOS as the BIOS policy name.
6. Leave all BIOS setting as “Platform Default.”

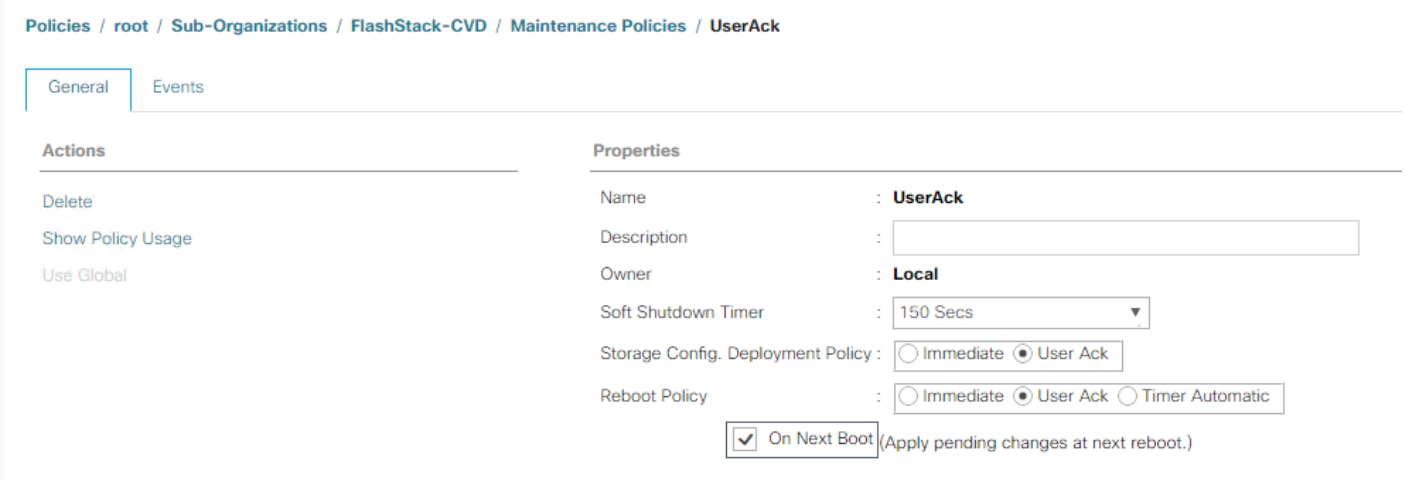


Cisco UCS M5 Server Performance Tuning guide: https://www.cisco.com/c/dam/en/us/products/collateral/servers-unified-computing/ucs-b-series-blade-servers/whitepaper_c11-740098.pdf

Configure Maintenance Policy

To update the default Maintenance Policy, complete the following steps:

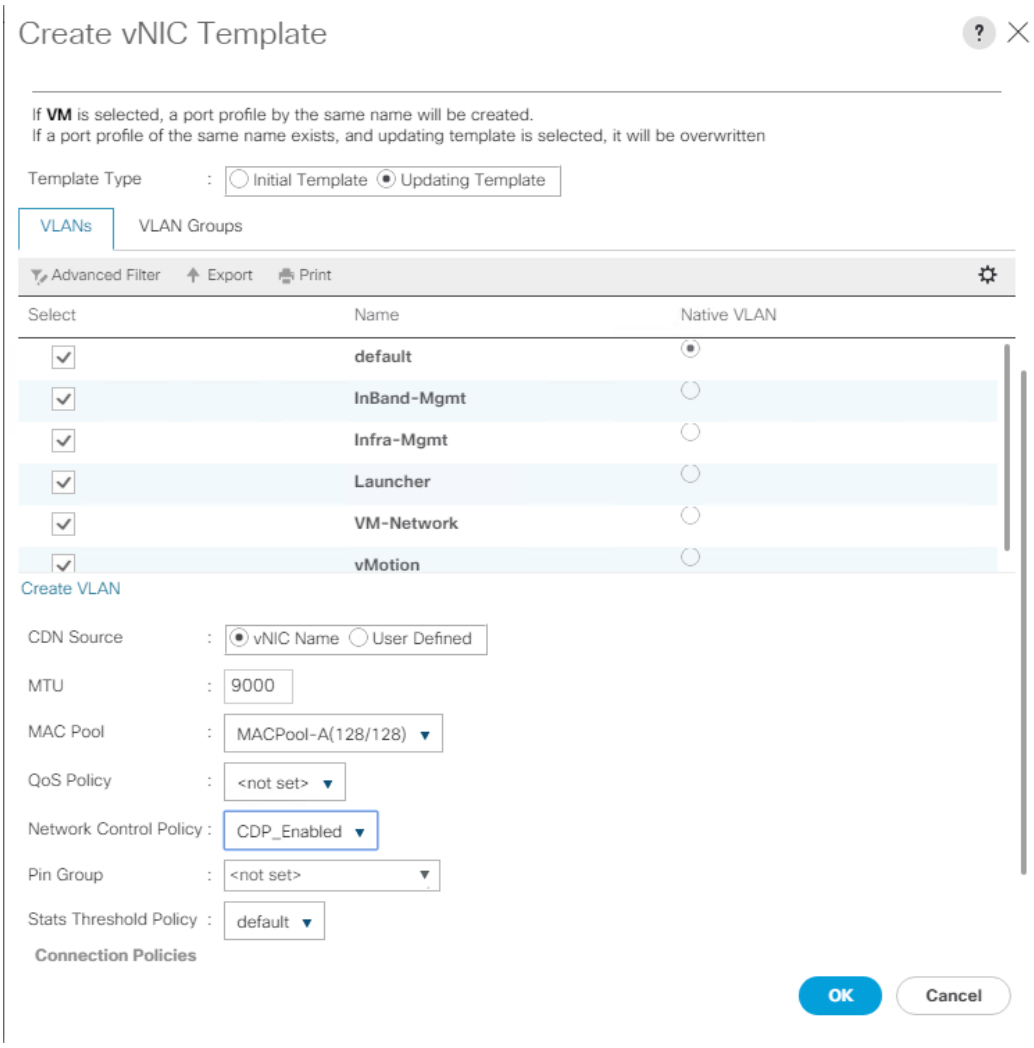
1. In Cisco UCS Manager, click the Servers tab in the navigation pane.
2. Select Policies > root > Sub-Organization > FlashStack-CVD > Maintenance Policies.
3. Right-click Maintenance Policies to create a new policy.
4. Enter name for Maintenance Policy
5. Change the Reboot Policy to User Ack.
6. Click Save Changes.
7. Click OK to accept the change.



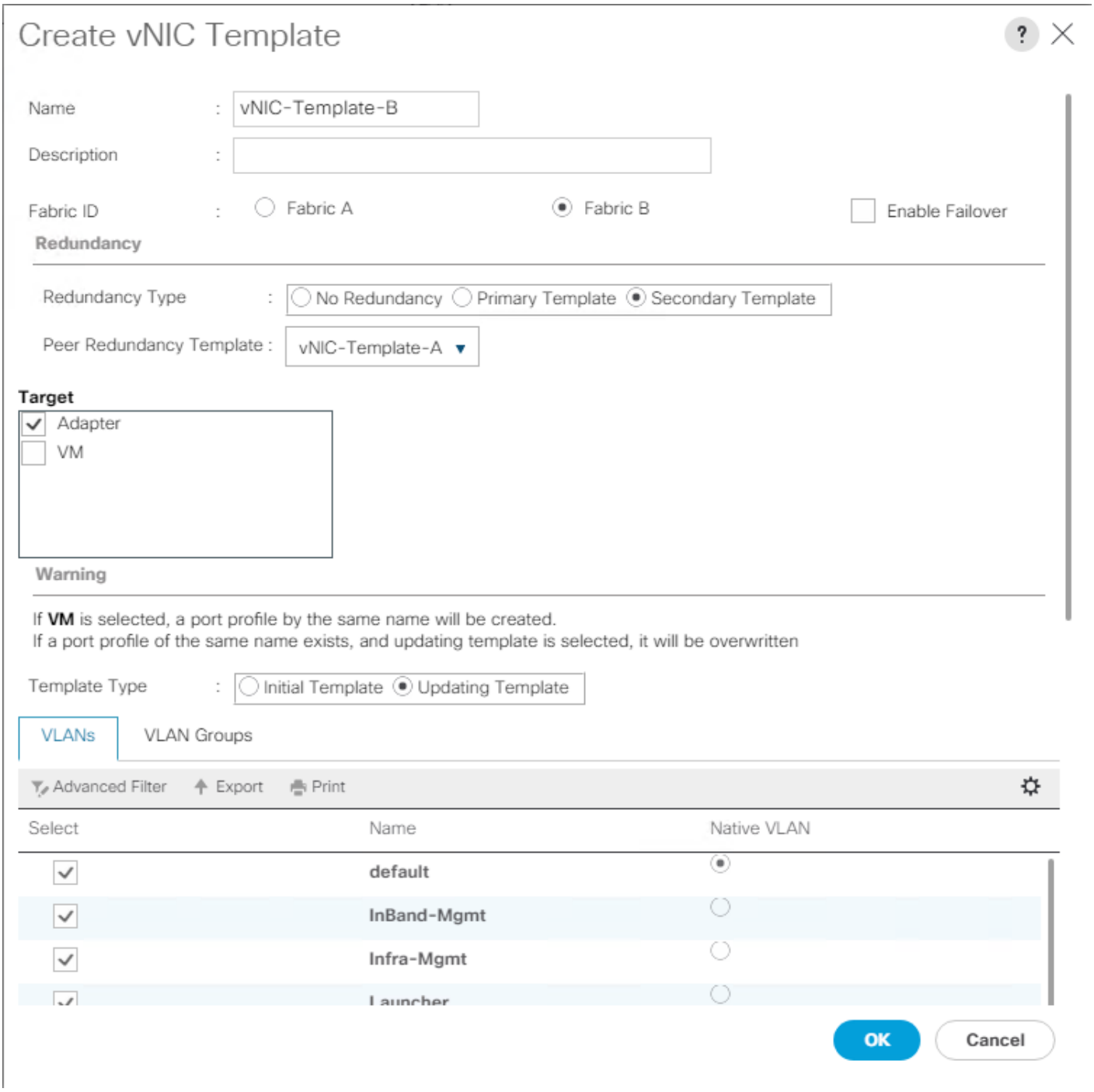
Create vNIC Templates

To create multiple virtual network interface card (vNIC) templates for the Cisco UCS environment, complete the following steps:

1. In Cisco UCS Manager, click the LAN tab in the navigation pane.
2. Select Policies > root > Sub-Organization > FlashStack-CVD > vNIC Template.
3. Right-click vNIC Templates.
4. Select Create vNIC Template.
5. Enter name for vNIC template.
6. Keep Fabric A selected. Do not select the Enable Failover checkbox.
7. Redundancy Type, Select “Primary Template.”
8. Select Updating Template as the Template Type.
9. Under VLANs, select the checkboxes for desired VLANs to add as part of the vNIC Template.
10. Set Native-VLAN as the native VLAN.
11. For MTU, enter 9000.
12. In the MAC Pool list, select MAC Pool configure for Fabric A.
13. In the Network Control Policy list, select CDP_Enabled.
14. Click OK to create the vNIC template.



15. Follow the steps above to create a vNIC Template for Fabric B. For Peer redundancy Template Select “vNIC-Template-A” created in previous step.



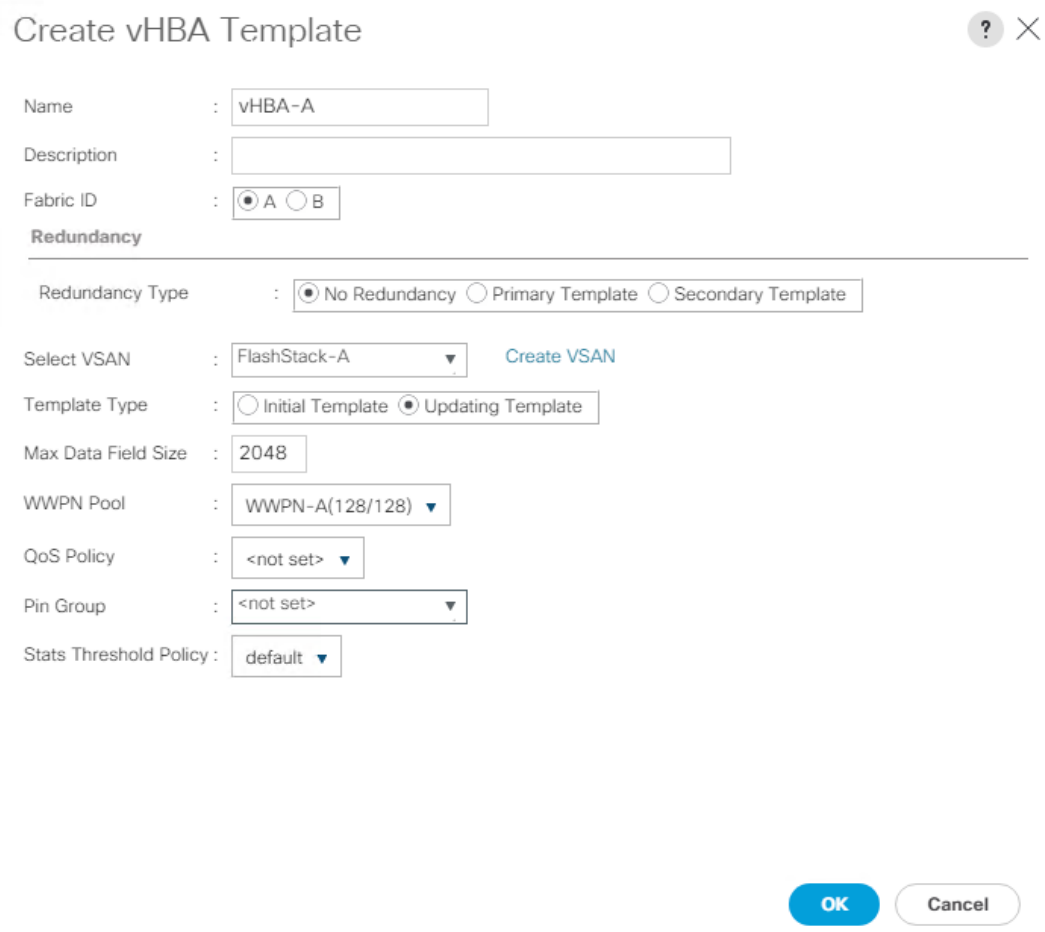
16. Verify that vNIC-Template-A Peer Redundancy Template is set to “vNIC-Template-B.”

Create vHBA Templates

To create multiple virtual host bus adapter (vHBA) templates for the Cisco UCS environment, complete the following steps:

1. In Cisco UCS Manager, click the SAN tab in the navigation pane.

2. Select Policies > root > Sub-Organization > FlashStack-CVD > vHBA Template.
3. Right-click vHBA Templates.
4. Select Create vHBA Template.
5. Enter vHBA-A as the vHBA template name.
6. Keep Fabric A selected.
7. Select VSAN created for Fabric A from the drop down.
8. Change to Updating Template.
9. For Max Data Field keep 2048.
10. Select WWPN Pool for Fabric A (created earlier) for our WWPN Pool.
11. Leave the remaining as is.
12. Click OK.



13. Follow the steps above to create a vHBA Template for Fabric B.

Create Server Boot Policy for SAN Boot

All Cisco UCS B200 M5 Blade server for workload and two Infrastructure server were set to boot from SAN for this Cisco Validated Design as part of the Service Profile template. 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, to name just a few.

We strongly recommend to use “Boot from SAN” to realize the full benefits of Cisco UCS stateless computing feature, such as service profile mobility.

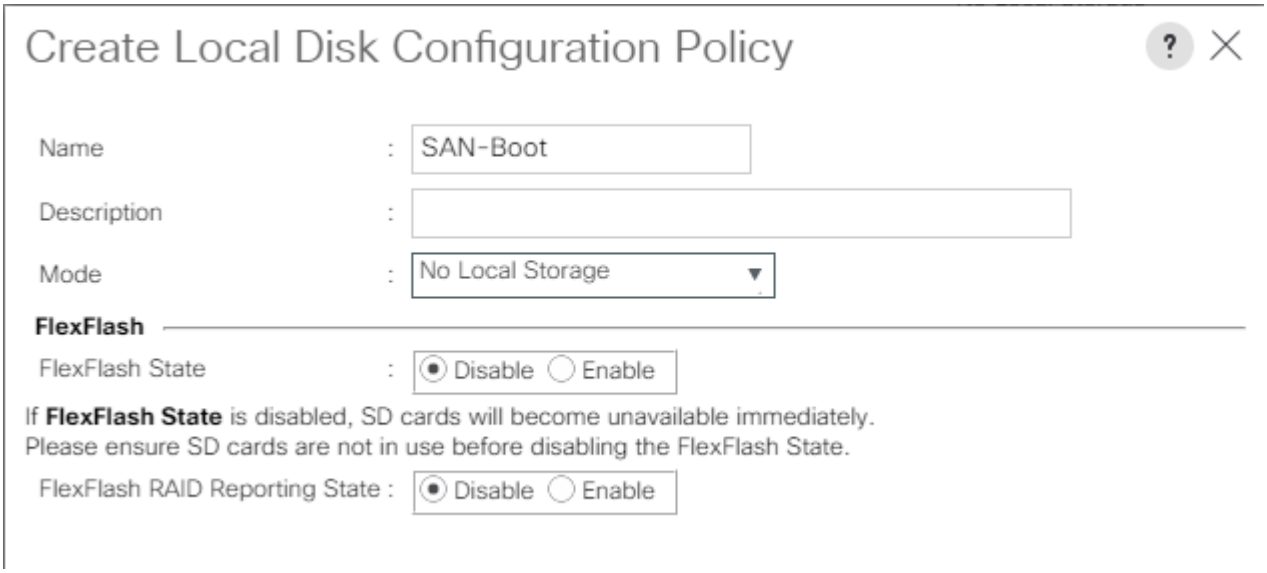
This process applies to a Cisco UCS environment in which the storage SAN ports are configured in the following section.

A Local disk configuration for the Cisco UCS is necessary if the servers in the environments have a local disk.

To configure Local disk policy, complete the following steps:

1. Go to tab Servers > Policies > root > Sub-Organization > FlashStack-CVD > right-click Local Disk Configuration Policy > Enter “SAN-Boot” as the local disk configuration policy name and change the mode to “No Local Storage.”

2. Click OK to create the policy.



As shown in the screenshot below, the Pure Storage FlashArray have eight active FC connections that go to the Cisco MDS 9148S switches. Four FC ports are connected to Cisco MDS-A and the other four FC ports are connected to Cisco MDS-B Switches. All FC ports are 16 Gb/s. The SAN Ports CT0.FC2, CT0.FC3, of Pure Storage FlashArray Controller 0 are connected to Cisco MDS Switch A and CT0.FC6, CT0.FC7 are connected to Cisco MDS Switch B. Similarly, the SAN Ports CT1.FC2, CT1.FC3, of Pure Storage FlashArray

Controller 1 are connected to Cisco MDS Switch A and CT1.FC6, CT1.FC7 are connected to Cisco MDS Switch B.

Array Ports

Port	Name	Speed	Failover	Port	Name	Speed	Failover
CT0.FC0	52:4A:93:75:DD:91:0A:00	16 Gb/s		CT1.FC0	52:4A:93:75:DD:91:0A:10	16 Gb/s	
CT0.FC1	52:4A:93:75:DD:91:0A:01	16 Gb/s		CT1.FC1	52:4A:93:75:DD:91:0A:11	16 Gb/s	
CT0.FC2	52:4A:93:75:DD:91:0A:02	16 Gb/s		CT1.FC2	52:4A:93:75:DD:91:0A:12	16 Gb/s	
CT0.FC3	52:4A:93:75:DD:91:0A:03	16 Gb/s		CT1.FC3	52:4A:93:75:DD:91:0A:13	16 Gb/s	
CT0.FC6	52:4A:93:75:DD:91:0A:06	0		CT1.FC6	52:4A:93:75:DD:91:0A:16	0	
CT0.FC7	52:4A:93:75:DD:91:0A:07	0		CT1.FC7	52:4A:93:75:DD:91:0A:17	0	

Create SAN Policy A

The SAN-A boot policy configures the SAN Primary's primary-target to be port CT0.FC2 on the Pure Storage cluster and SAN Primary's secondary-target to be port CT1.FC2 on the Pure Storage cluster. Similarly, the SAN Secondary's primary-target should be port CT1.FC3 on the Pure Storage cluster and SAN Secondary's secondary-target should be port CT0.FC3 on the Pure Storage cluster.

Log into the storage controller and verify all the port information is correct. This information can be found in the Pure Storage GUI under System > Connections > Target Ports.

You have to create SAN Primary (hba0) and SAN Secondary (hba1) in SAN-A Boot Policy by entering WWPN of Pure Storage FC Ports as detailed in the following section.

To create Boot Policies for the Cisco UCS environments, complete the following steps:

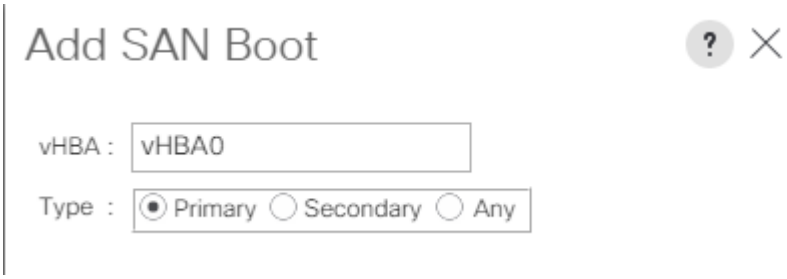
1. Go to Cisco UCS Manager and then go to Servers > Policies > root > Sub Organization > FlashStack-CVD > Boot Policies. Right-click and select Create Boot Policy.

2. Enter SAN-A as the name of the boot policy.

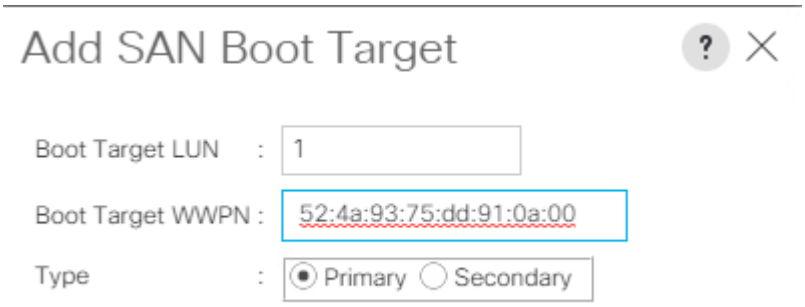
3. Expand the Local Devices drop-down menu and Choose Add CD/DVD. Expand the vHBAs drop-down list and Choose Add SAN Boot.

> The SAN boot paths and targets will include primary and secondary options in order to maximize resiliency and number of paths.

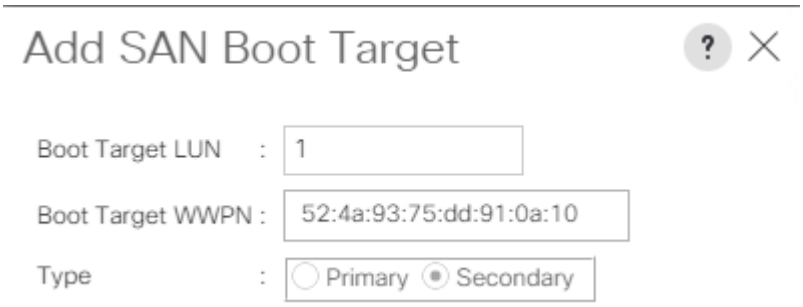
4. In the Add SAN Boot dialog box, select Type as “Primary” and name vHBA as “hba0”. Click OK to add SAN Boot.



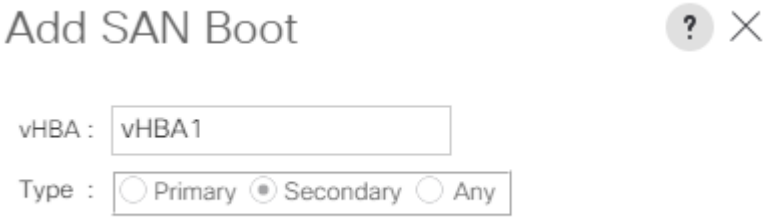
5. Select add SAN Boot Target to enter WWPN address of storage port. Keep 1 as the value for Boot Target LUN. Enter the WWPN for FC port CT0.FC0 of Pure Storage and add SAN Boot Primary Target.



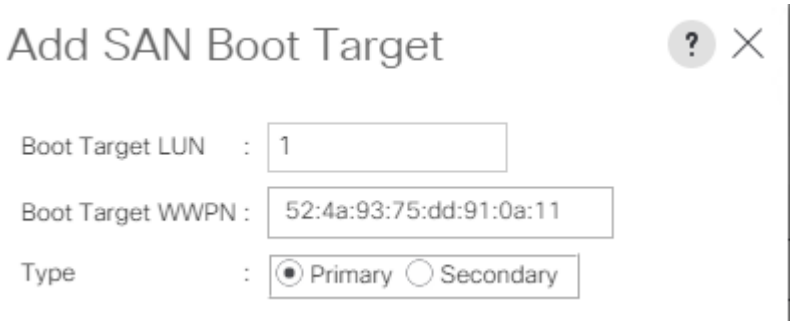
6. Add secondary SAN Boot target into same hba0, enter the boot target LUN as 1 and WWPN for FC port CT1.FC0 of Pure Storage, and add SAN Boot Secondary Target.



7. From the vHBA drop-down list and choose Add SAN Boot. In the Add SAN Boot dialog box, enter "hba1" in the vHBA field. Click OK to SAN Boot, then choose Add SAN Boot Target.

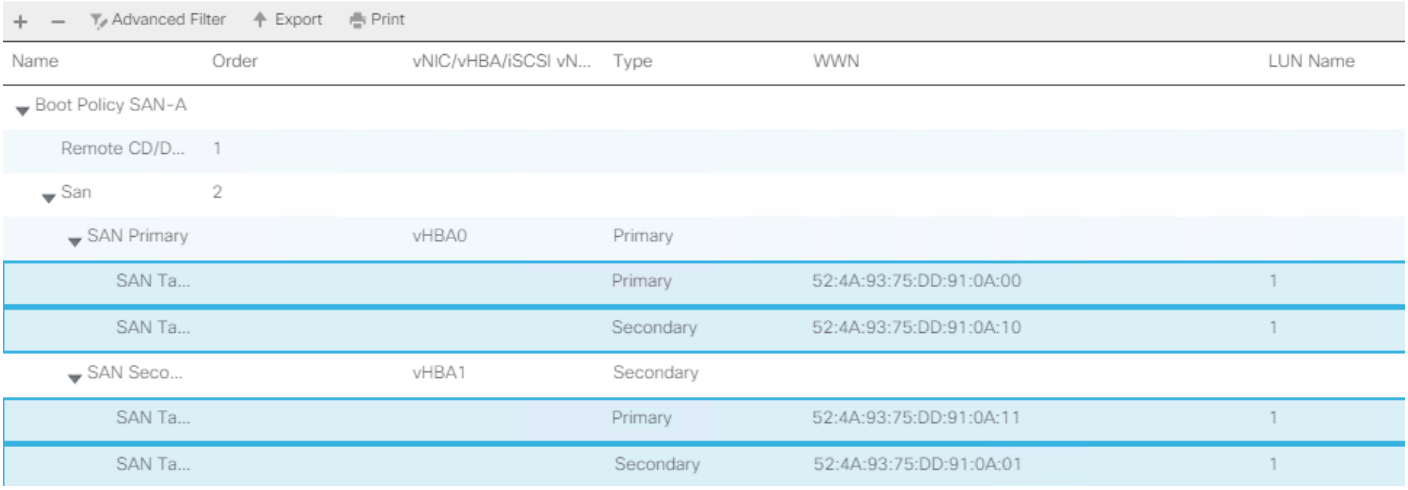


8. Keep 1 as the value for the Boot Target LUN. Enter the WWPN for FC port CT1.FC1 of Pure Storage and add SAN Boot Primary Target.



9. Add a secondary SAN Boot target into same vhba1 and enter the boot target LUN as 1 and WWPN for FC port CT0.FC1 of Pure Storage and add SAN Boot Secondary Target.

10. After creating the FC boot policies, you can view the boot order in the Cisco UCS Manager GUI. To view the boot order, navigate to Servers > Policies > Boot Policies. Click Boot Policy SAN-Boot-A to view the boot order in the right pane of the Cisco UCS Manager as shown below:



Name	Order	vNIC/vHBA/iSCSI vN...	Type	WWN	LUN Name
Boot Policy SAN-A					
Remote CD/D...	1				
San	2				
SAN Primary		vHBA0	Primary		
SAN Ta...			Primary	52:4A:93:75:DD:91:0A:00	1
SAN Ta...			Secondary	52:4A:93:75:DD:91:0A:10	1
SAN Seco...		vHBA1	Secondary		
SAN Ta...			Primary	52:4A:93:75:DD:91:0A:11	1
SAN Ta...			Secondary	52:4A:93:75:DD:91:0A:01	1

Create SAN Policy B

The SAN-B boot policy configures the SAN Primary's primary-target to be port CT0.FC6 on the Pure Storage cluster and SAN Primary's secondary-target to be port CT1.FC6 on the Pure Storage cluster. Similarly, the SAN Secondary's primary-target should be port CT1.FC7 on the Pure Storage cluster and SAN Secondary's secondary-target should be port CT0.FC7 on the Pure Storage cluster.

Log into the storage controller and verify all the port information is correct. This information can be found in the Pure Storage GUI under System > Connections > Target Ports.

You have to create SAN Primary (vHBA0) and SAN Secondary (vHBA1) in SAN-B Boot Policy by entering WWPN of Pure Storage FC Ports as explained in the following section.

To create boot policies for the Cisco UCS environments, complete the following steps:

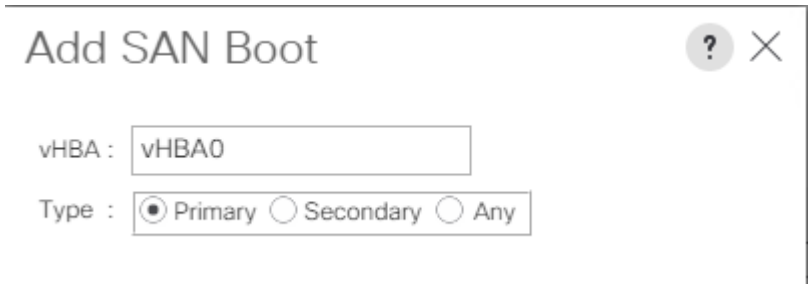
1. Go to UCS Manager and then go to tab Servers > Policies > root > Sub Organization > FlashStack-CVD > Boot Policies.

2. Right-click and select Create Boot Policy. Enter SAN-B as the name of the boot policy.

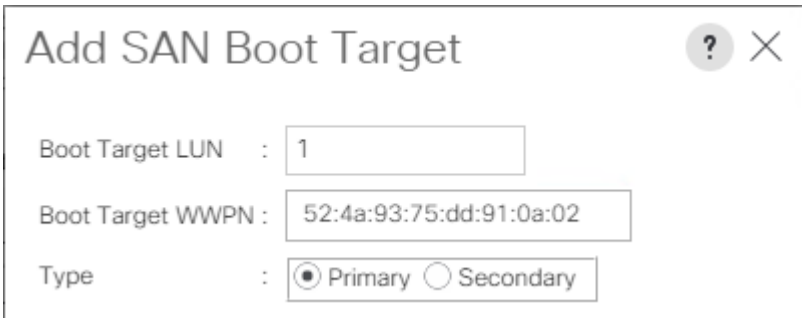
3. Expand the Local Devices drop-down list and Choose Add CD/DVD. Expand the vHBAs drop-down list and choose Add SAN Boot.

The SAN boot paths and targets will include primary and secondary options in order to maximize resiliency and number of paths.

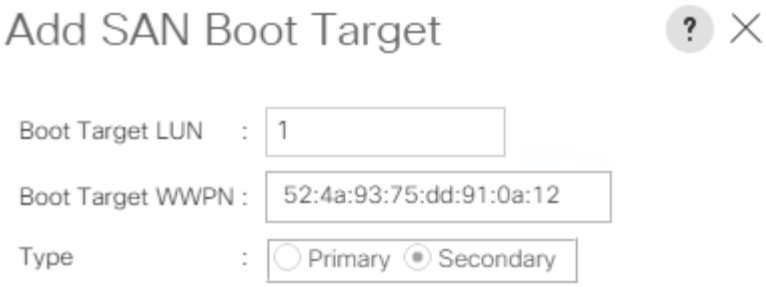
4. In the Add SAN Boot dialog box, select Type as “Primary” and name vHBA as “vHBA0”. Click OK to add SAN Boot.



5. Select Add SAN Boot Target to enter WWPN address of storage port. Keep 1 as the value for Boot Target LUN. Enter the WWPN for FC port CT0.FC2 of Pure Storage and add SAN Boot Primary Target.

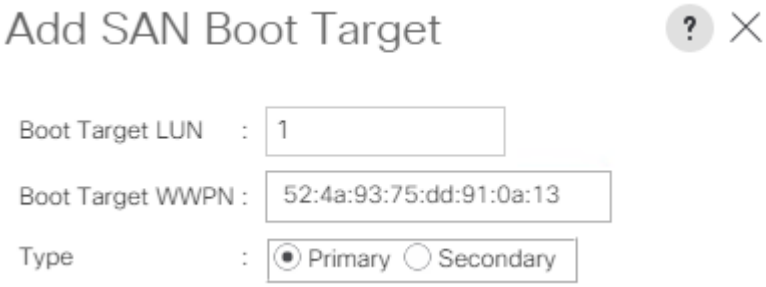


6. Add the secondary SAN Boot target into the same hba0; enter boot target LUN as 1 and WWPN for FC port CT1.FC2 of Pure Storage, and add SAN Boot Secondary Target.

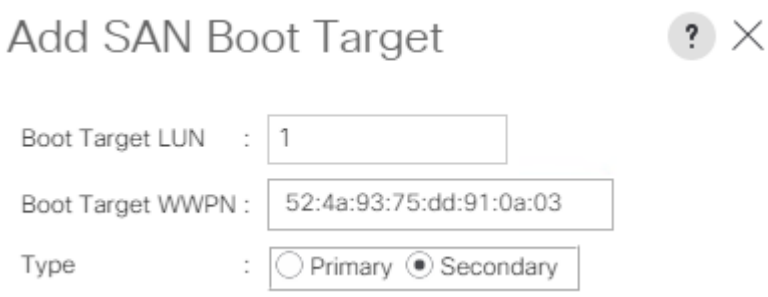


7. From the vHBA drop-down list, choose Add SAN Boot. In the Add SAN Boot dialog box, enter "hba1" in the vHBA field. Click OK to SAN Boot, then choose Add SAN Boot Target.

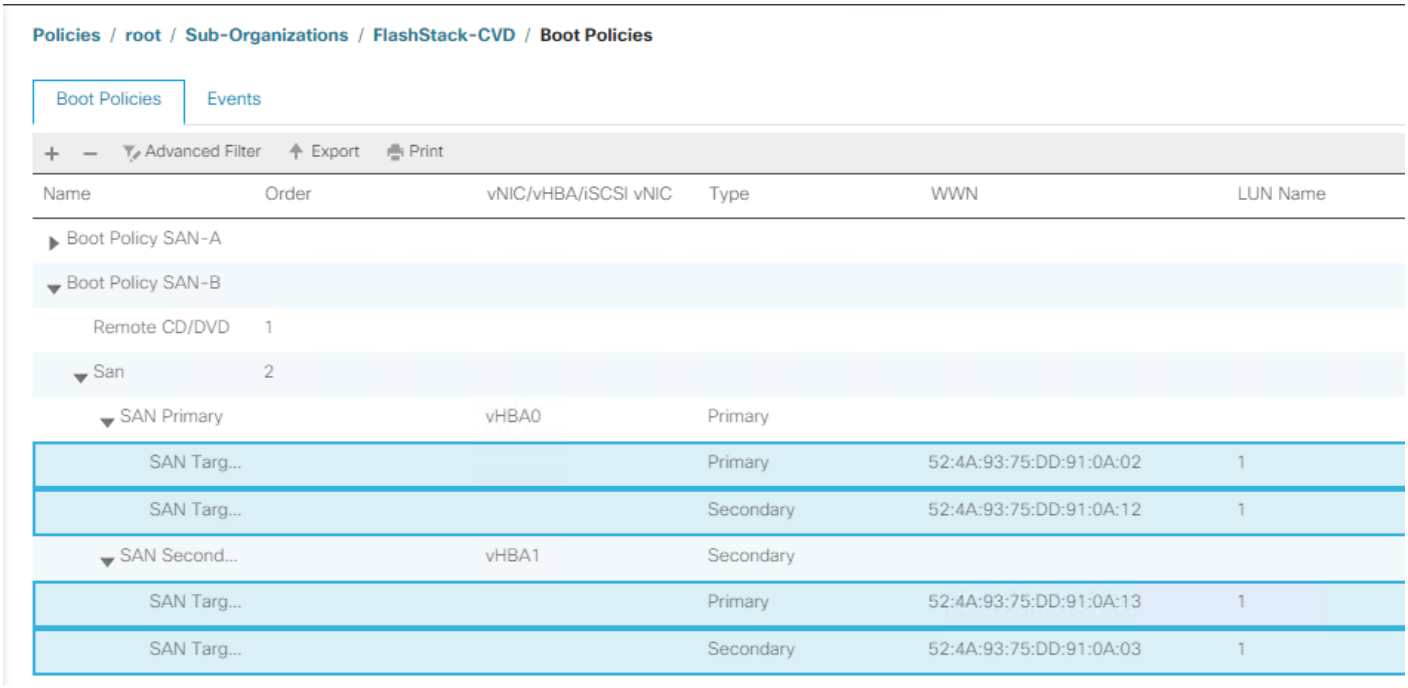
8. Keep 1 as the value for Boot Target LUN. Enter the WWPN for FC port CT1.FC3 of Pure Storage and Add SAN Boot Primary Target.



9. Add secondary SAN Boot target into same hba1 and enter boot target LUN as 1 and WWPN for FC port CT0.FC3 of Pure Storage and add SAN Boot Secondary Target.



10. After creating the FC boot policies, you can view the boot order in the Cisco UCS Manager GUI. To view the boot order, navigate to Servers > Policies > Boot Policies. Click Boot Policy SAN-Boot-B to view the boot order in the right pane of the Cisco UCS Manager as shown below:



For this solution, we created two Boot Policy as “SAN-A” and “SAN-B”. For thirty-two UCS B200 M5 blade server, you will assign first 16 Service Profiles with SAN-A to first 16 server and remaining 16 Service Profiles with SAN-B to remaining 16 server as explained in the following section.

Configure and Create a Service Profile Template

Service profile templates enable policy based server management that helps ensure consistent server resource provisioning suitable to meet predefined workload needs.

You will create two Service Profile Template. First Service profile template “VCC-CVD01” using boot policy as “SAN-A” and second Service profile template “VCC-CVD02” using boot policy as “SAN-B” to utilize all the FC ports from Pure Storage for high-availability in case of any FC links go down.

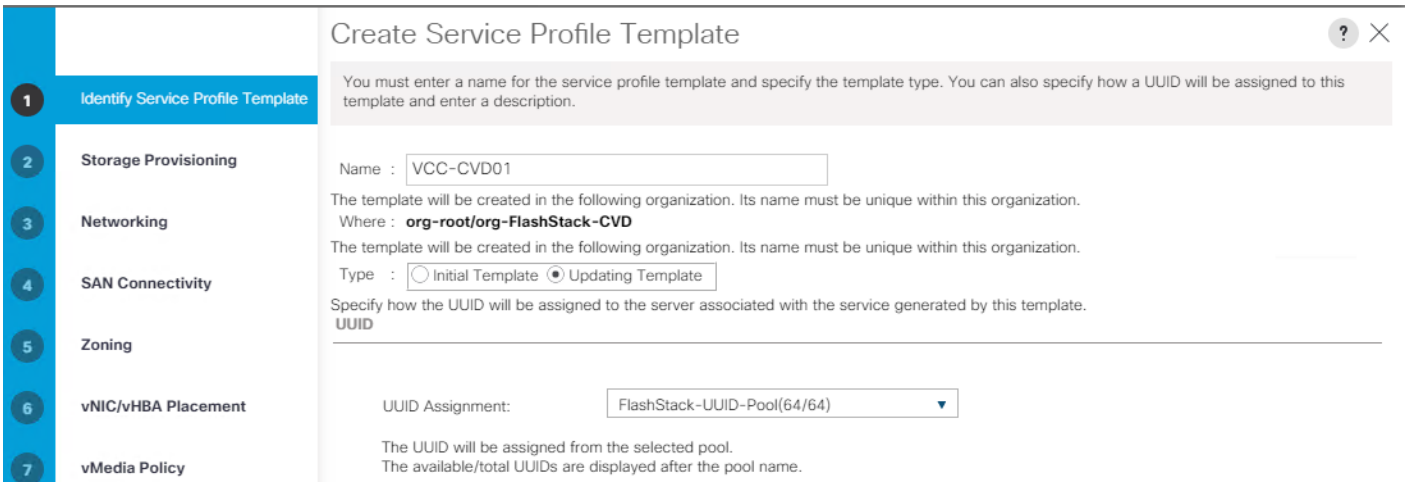
You will create the first VCC-CVD01 as explained in the following section.

Create Service Profile Template

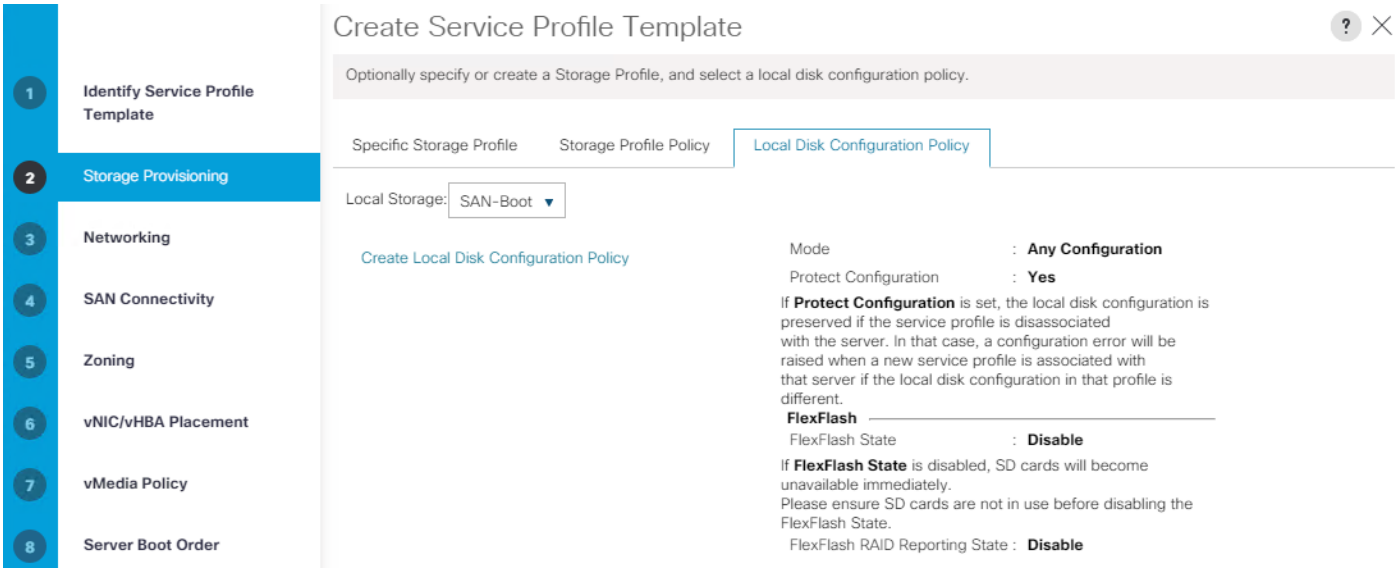
To create a service profile template, complete the following steps:

1. In the Cisco UCS Manager, go to Servers > Service Profile Templates > root Sub Organization > FlashStack-CVD > and right-click to “Create Service Profile Template” as shown below.

2. Enter the Service Profile Template name, select the UUID Pool that was created earlier, and click Next.



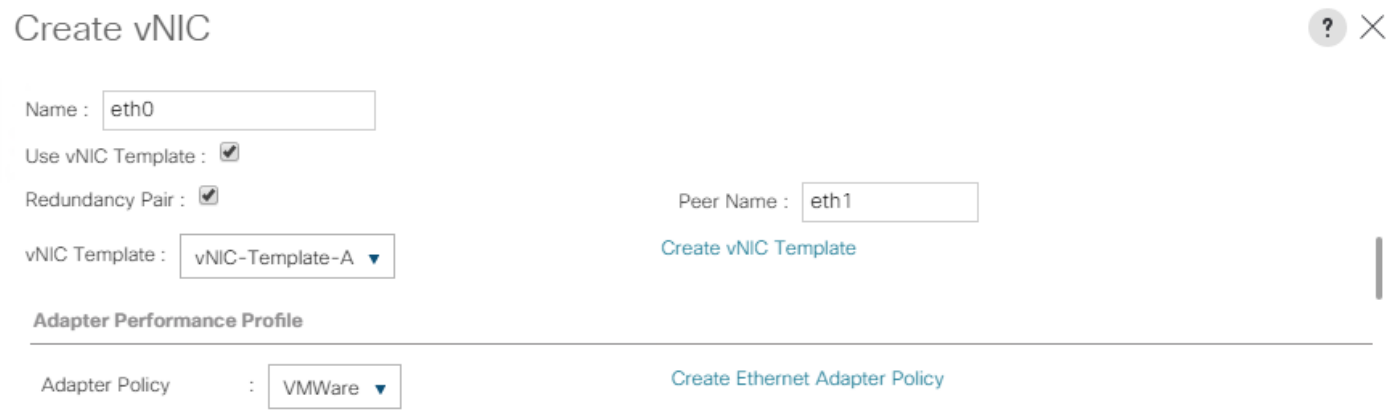
3. Select Local Disk Configuration Policy to SAN-Boot as No Local Storage.



4. In the networking window, select “Expert” and click “Add” to create vNICs. Add one or more vNICs that the server should use to connect to the LAN.

5. Now there are two vNIC in the create vNIC menu. You have given name to first vNIC as “eth0” and second vNIC as “eth1.”

6. Select vNIC Template as vNIC-Template-A and Adapter Policy as VMware, as shown below.

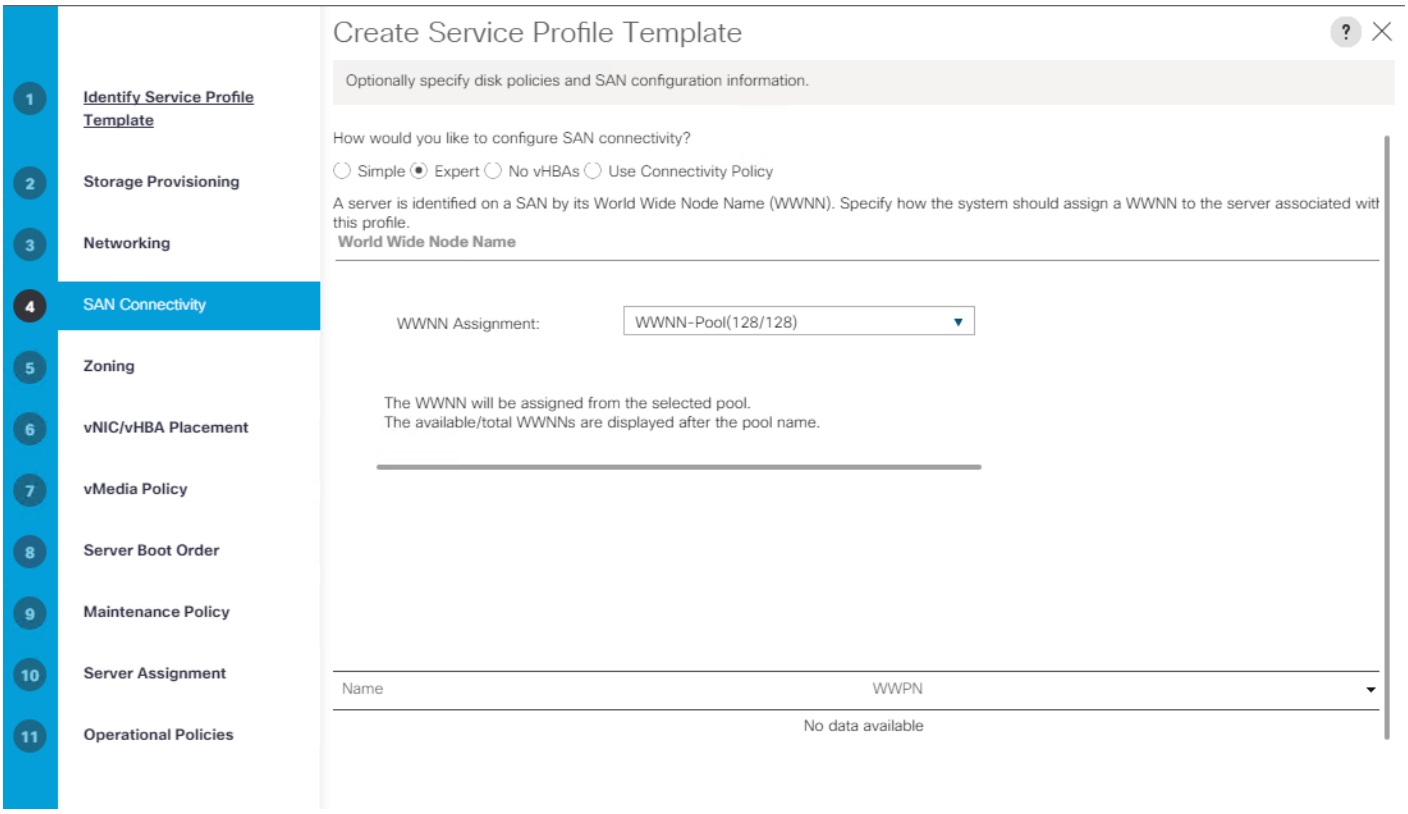


7. Select vNIC Template as vNIC-Template-B, created with the name eth1. Select the Adapter Policy as VMware for the vNIC "eth1."

eth0 and eth1 vNICs are created so that the servers can connect to the LAN.

8. When the vNICs are created, you need to create vHBAs. Click Next.

9. In the SAN Connectivity menu, select "Expert" to configure as SAN connectivity. Select WWNN (World Wide Node Name) pool, which we created earlier. Click "Add" to add vHBAs.



10. The following four HBA were created:

- vHBA0 using vHBA Template vHBA-A
- vHBA1 using vHBA Template vHBA-B
- vHBA2 using vHBA Template vHBA-A
- vHBA3 using vHBA Template vHBA-B

Figure 34 vHBA0

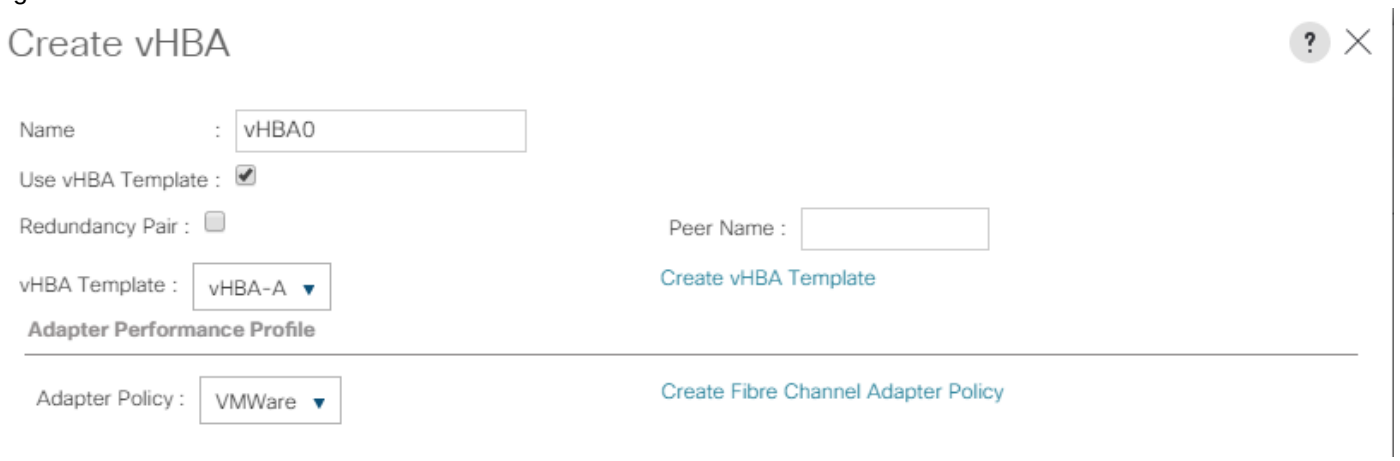
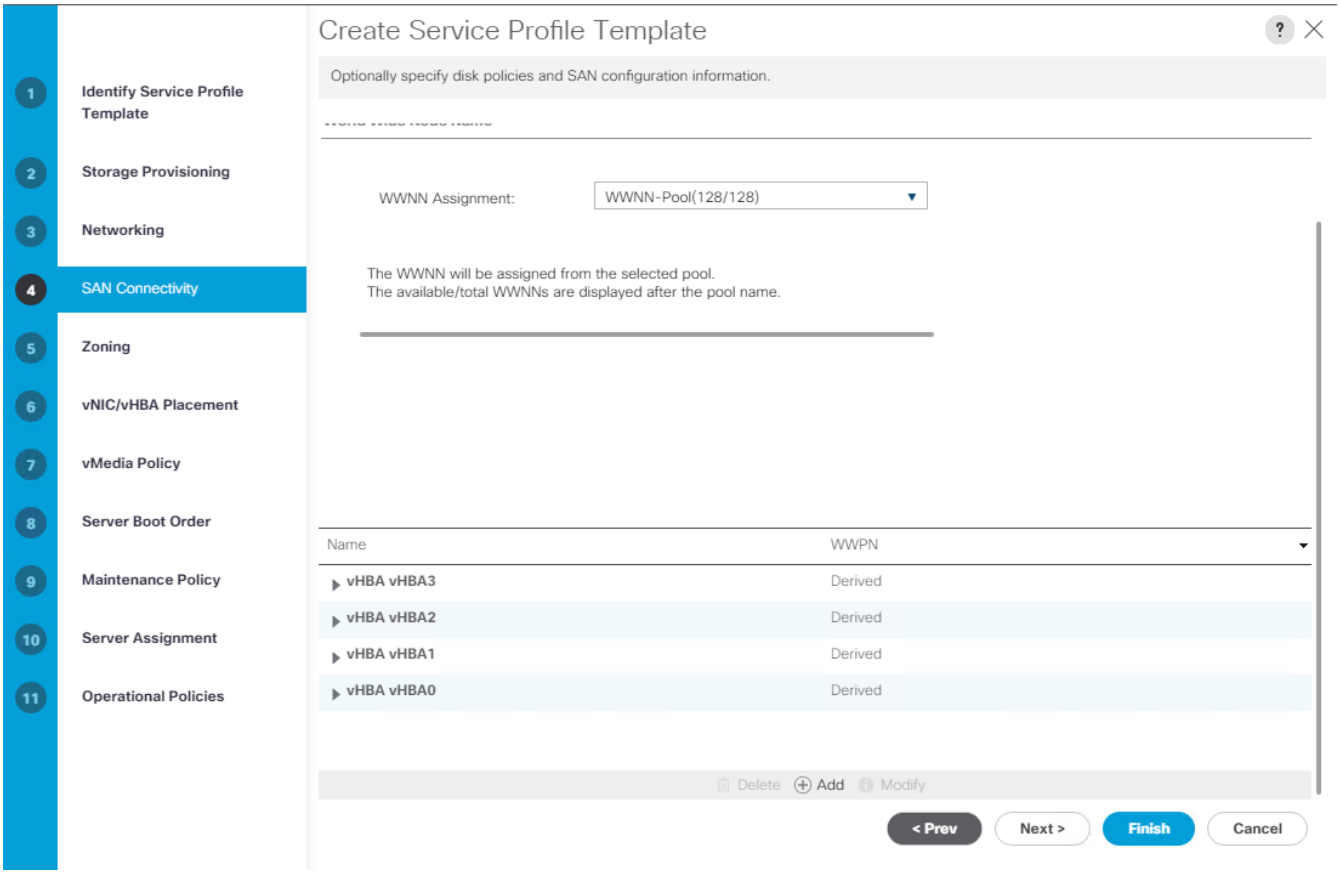


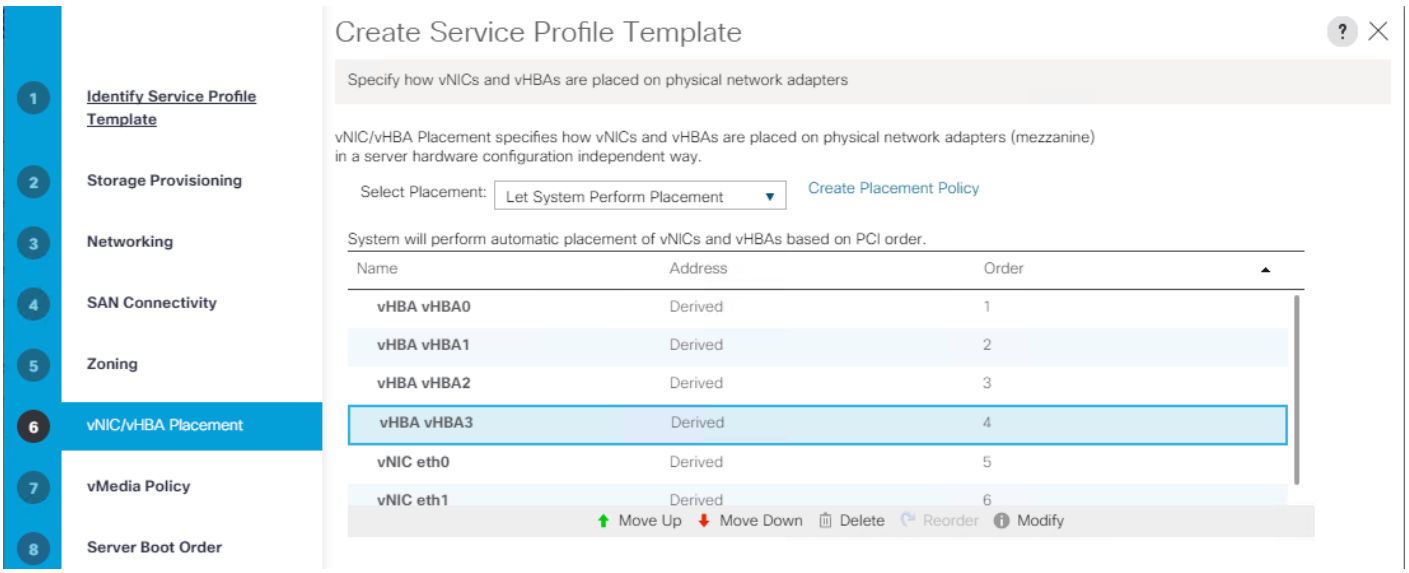
Figure 35 vHBA1

Figure 36 All vHBAs

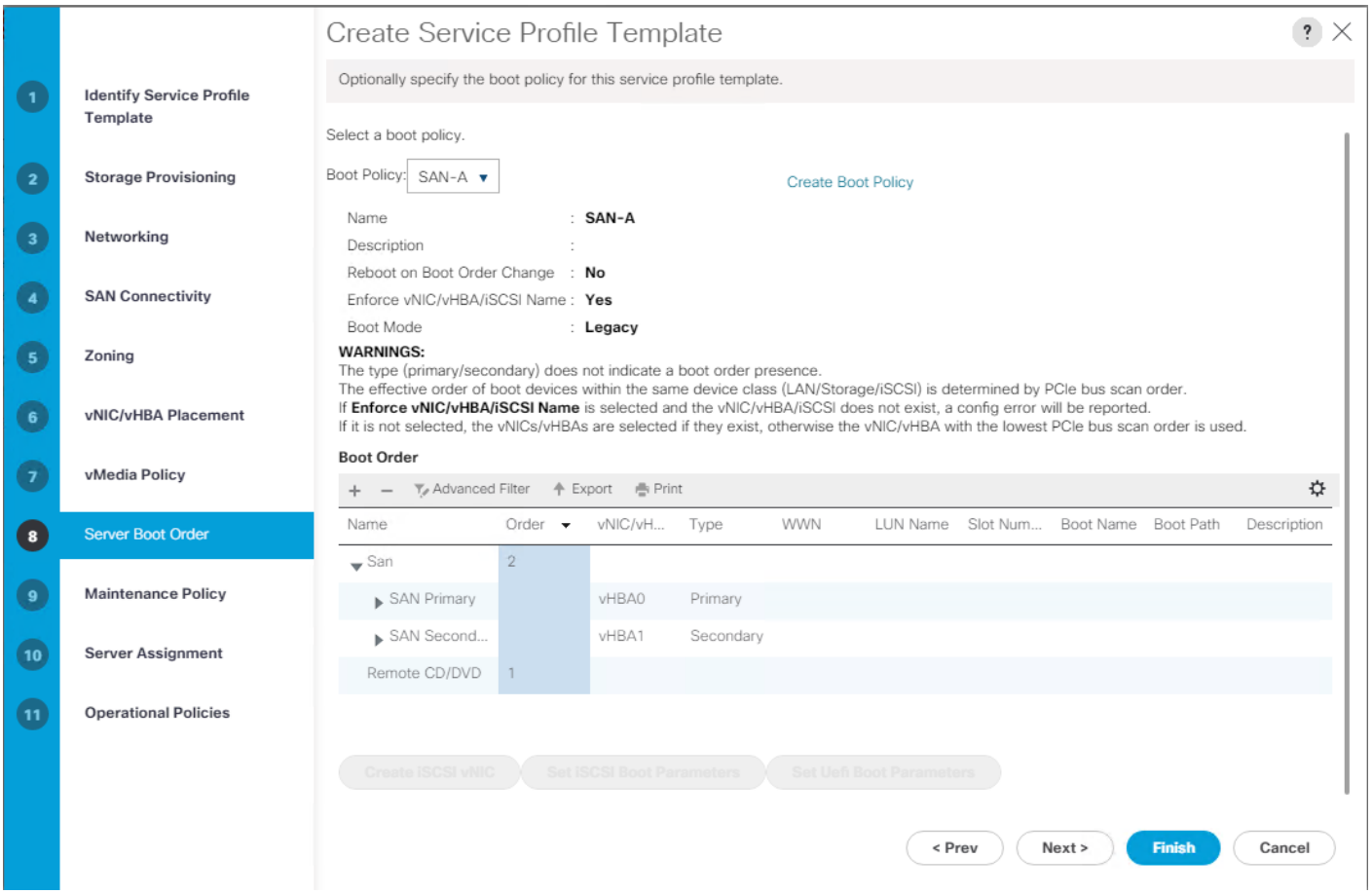


11. Skip zoning; for this FlashStack Configuration, the Cisco MDS 9148S is used for zoning.

12. Select the default option as Let System Perform Placement in the Placement Selection menu.

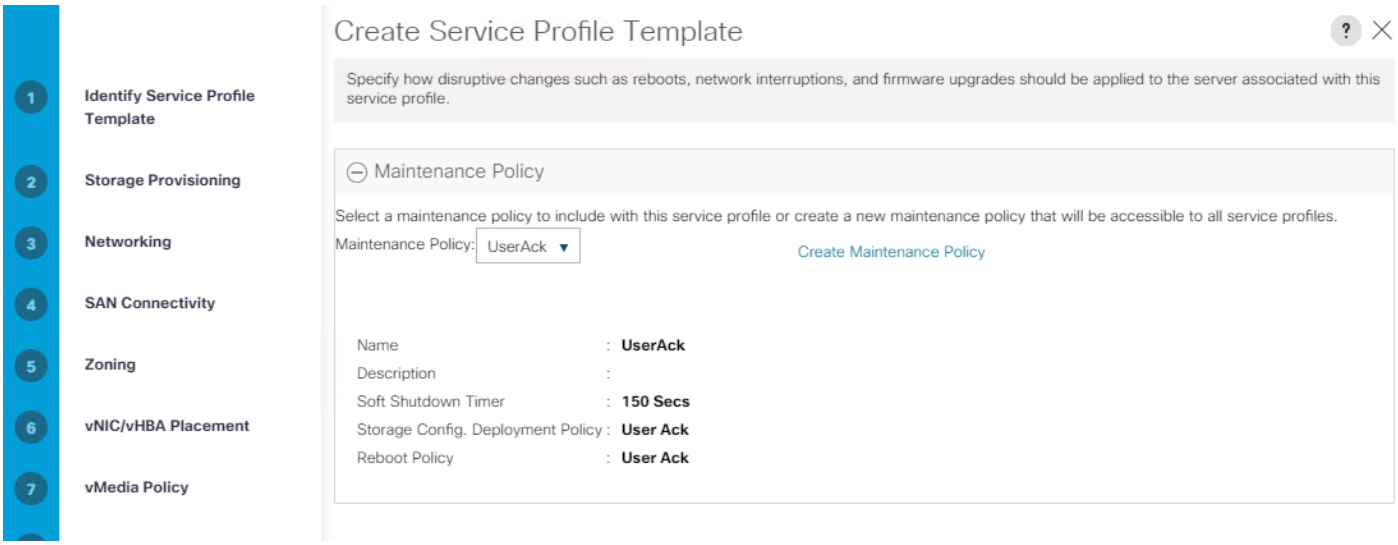


13. For the Server Boot Policy, select “SAN-A” as Boot Policy which you created earlier.



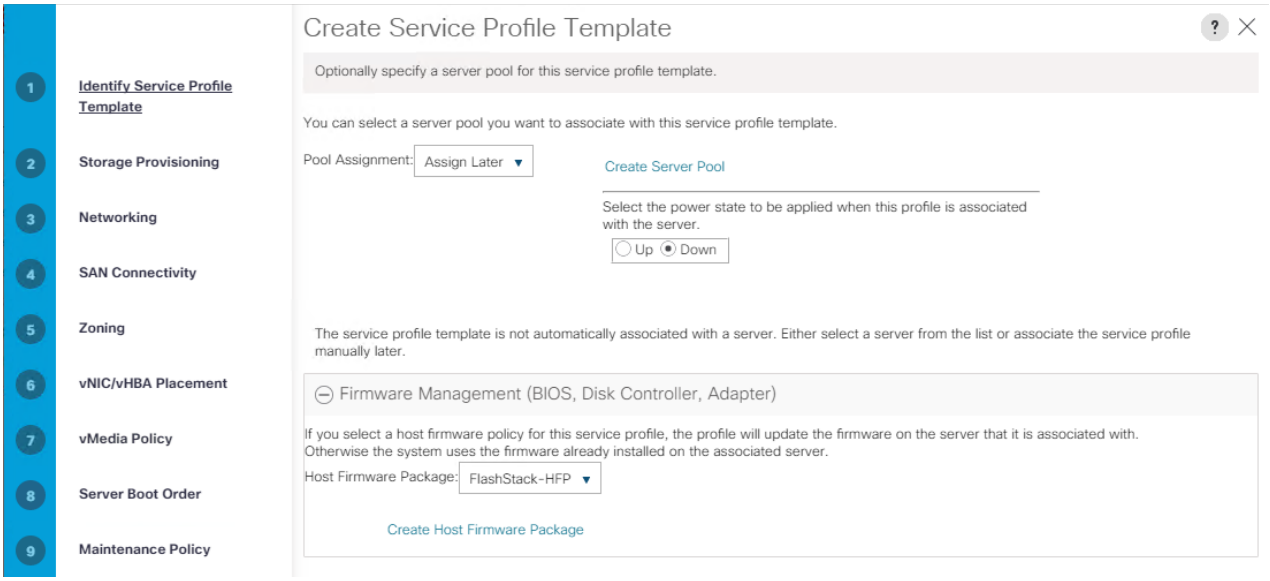
The default setting was retained for the remaining maintenance and assignment policies in the configuration. However, they may vary from site-to-site depending on workloads, best practices, and policies. For example, we created maintenance policy, BIOS policy, Power Policy as detailed below.

14. Select UserAck maintenance policy, which requires user acknowledgement prior rebooting server when making changes to policy or pool configuration tied to a service profile.

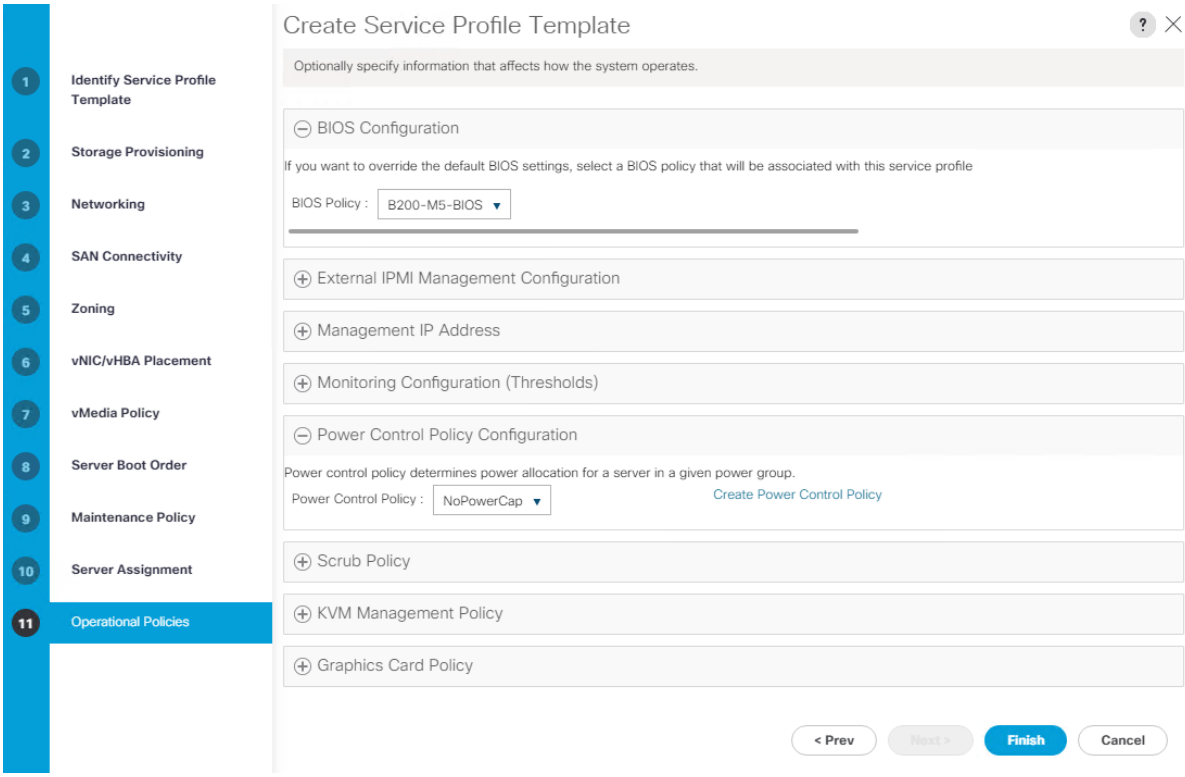


15. Select Server Pool policy to automatically assign service profile to a server that meets the requirement for server qualification based on the pool configuration.

16. On the same page; you can configure “Host firmware Package Policy” which helps to keep the firmware in sync when associated to server.



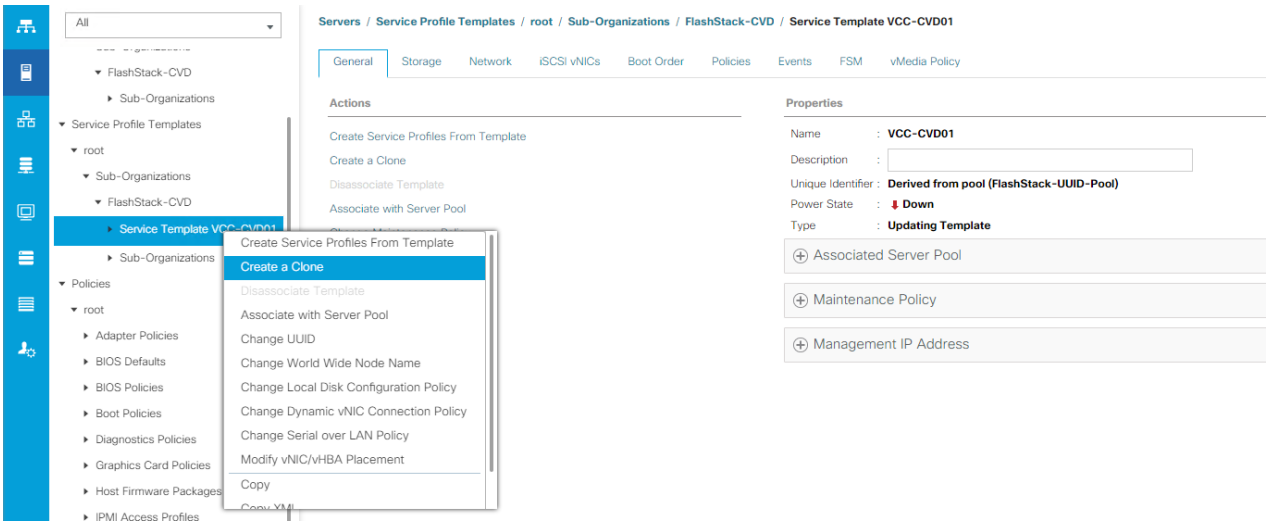
17. On the Operational Policy page, we configured BIOS policy for B200 M5 blade server, Power Control Policy with “NoPowerCap” for maximum performance and Graphics Card Policy for B200 M5 server configured with NVidia P6 GPU card.



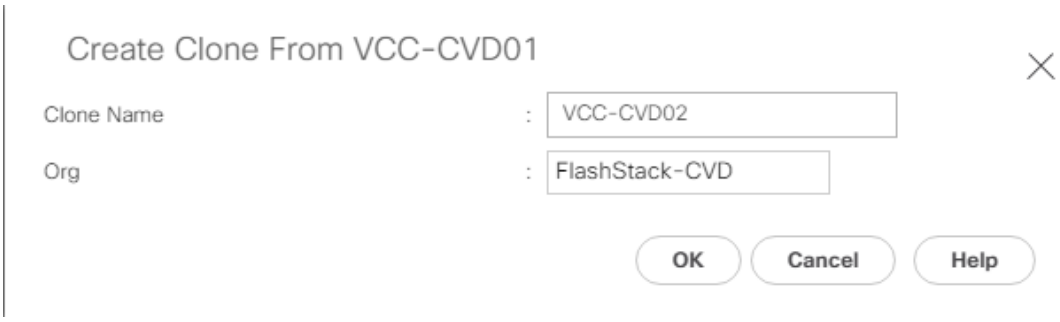
18. Click Next and then click Finish to create service profile template as “VCC-CVD01.”

Clone Service Profile Template

1. In the service profile template VCC-CVD02, modify the Boot Policy as “SAN-B” to use all the remaining FC paths of storage for high availability.

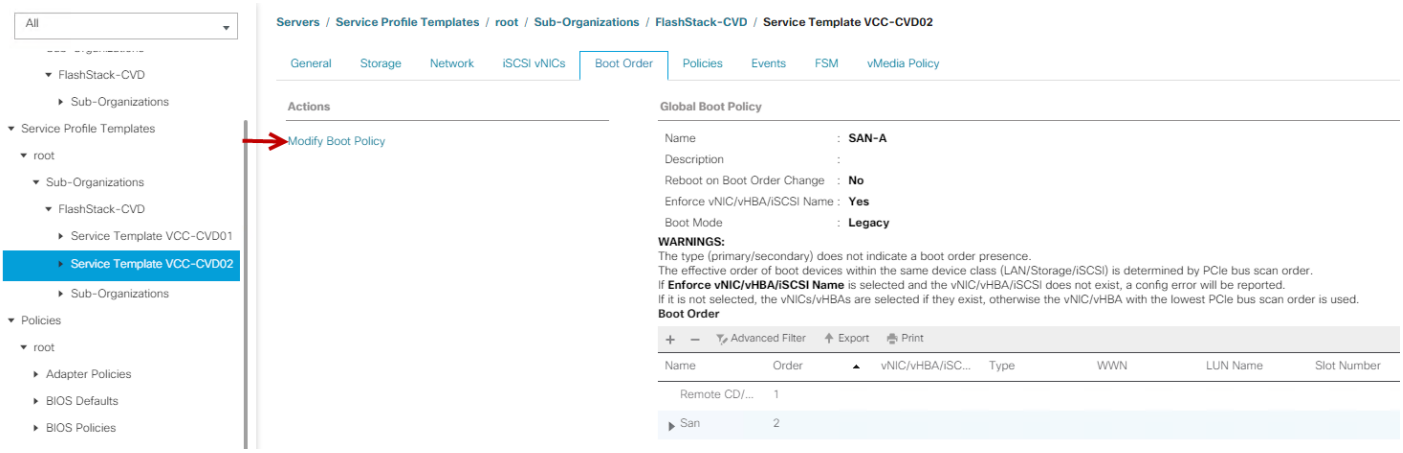


2. Enter name to create Clone from existing Service Profile template. Click OK.

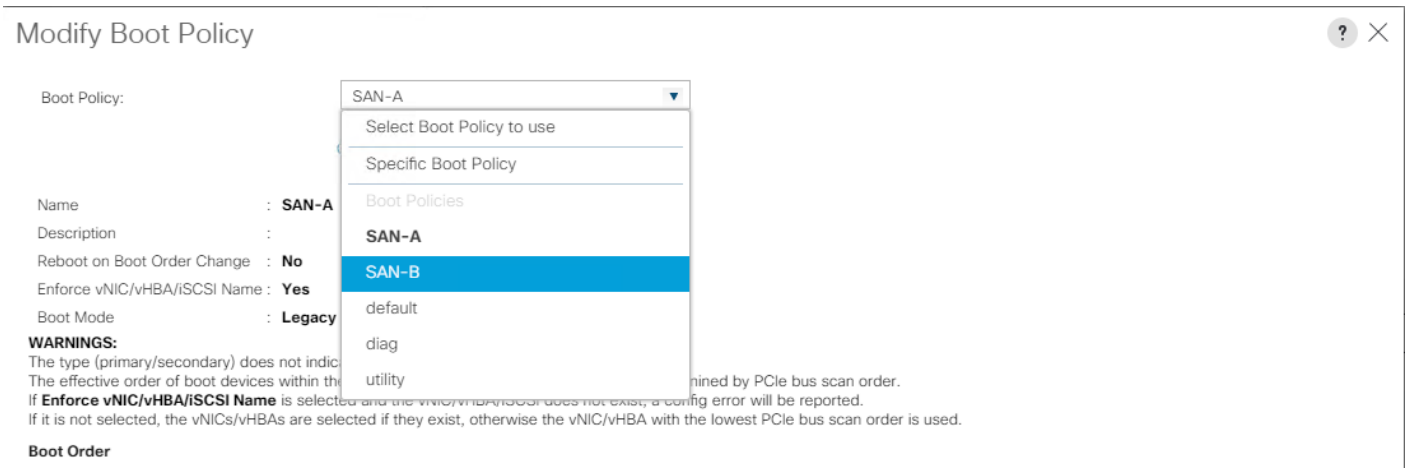


This VCC-CVD02 service profile template will be used to create the remaining sixteen service profiles for VCC workload and Infrastructure server02.

3. To change boot order from SAN-A to SAN-B for VCC-CVD02, click Cloned Service Profile template > Select Boot Order tab. Click Modify Boot Policy.



4. From the drop-down list select “SAN-B” as Boot Policy, click OK.



You have now created Service profile template “VCC-CVD01” and “VCC-CVD02” with each having four vHBAs and two vNICs.

Create Service Profiles from Template and Associate to Servers

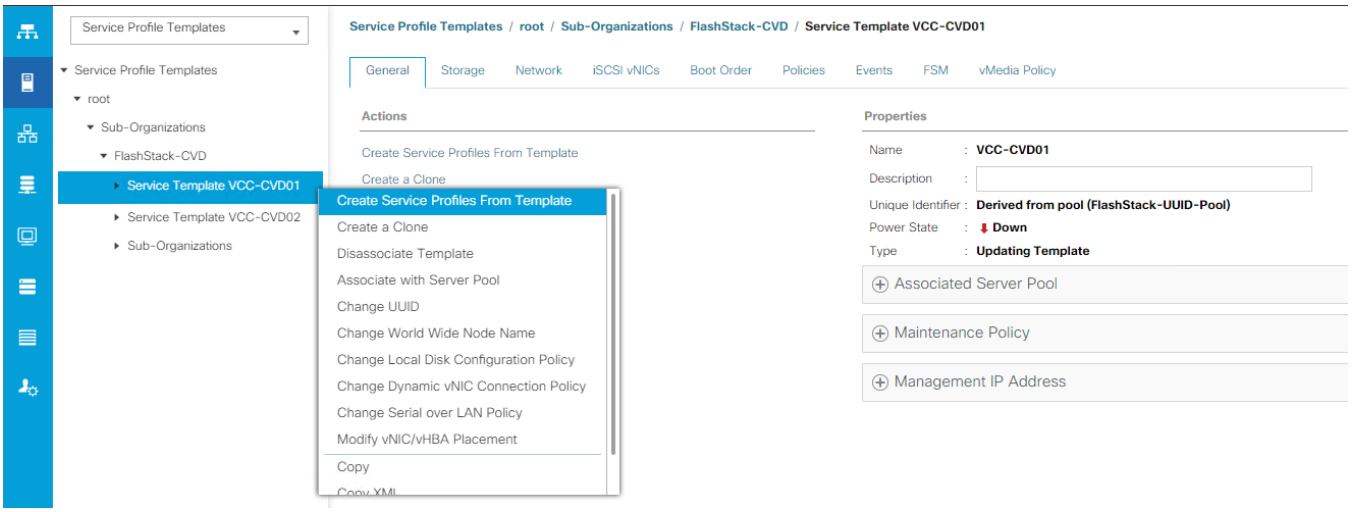
Create Service Profiles from Template

You will create sixteen Service profiles from VCC-CVD01 template and sixteen Service Profile from VCC-CVD02 template as explained in the following sections.

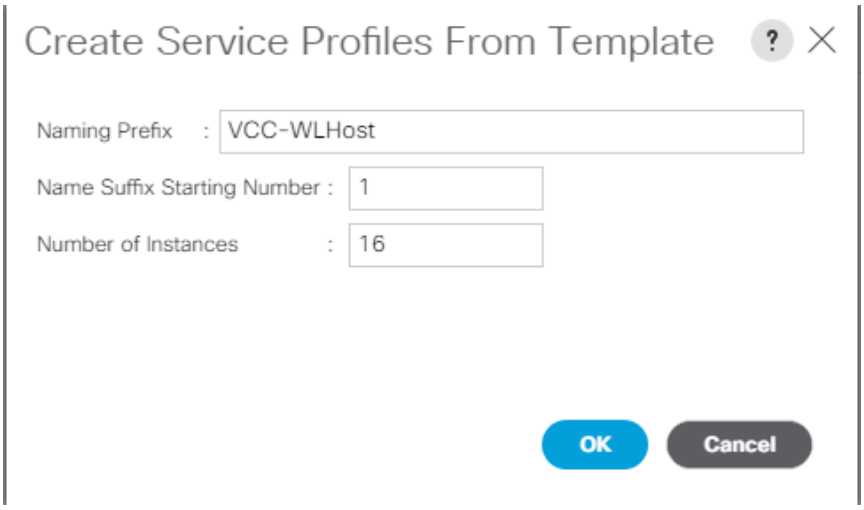
For the first fifteen workload Nodes and Infrastructure Node 01, you will create sixteen Service Profiles from Template “VCC-CVD01.” The remaining fifteen workload Nodes and Infrastructure Node 02, will require creating another sixteen Service Profiles from Template “VCC-CVD02”.

To create first four Service Profiles from Template, complete the following steps:

1. Go to tab Servers > Service Profiles > root > Sub-Organization > FlashStack-CVD and right-click “Create Service Profiles from Template”.



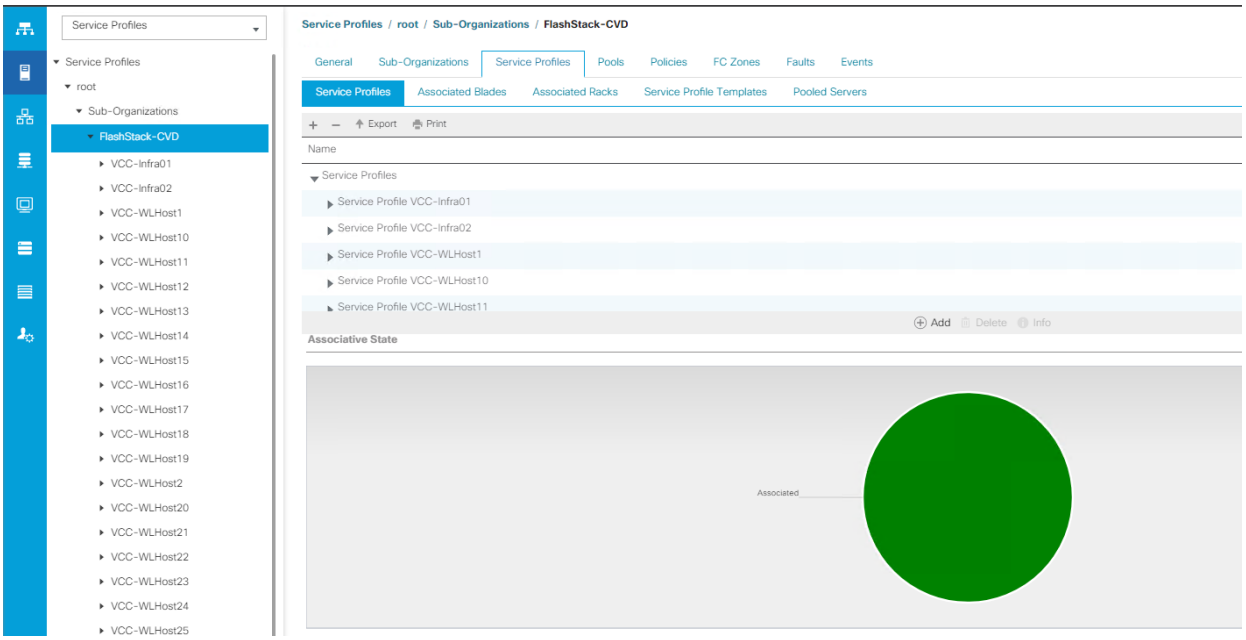
2. Select the Service profile template as “VCC-CVD01” which you created earlier and name the service profile as “VCC-WLHostX”. To create four service profiles, enter “Number of Instances” as 16 as shown below. This process will create service profiles as “VCC-WLHOST1”, “VCC-WLHOST2”, ... and “VCC-WLHOST16.”



3. Create the remaining four Service Profiles “VCC-WLHOST17”, “VCC-WLHOST18”, ... and “VCC-WLHOST32” from Template “VCC-CVD02.”

When the service profiles are created, the association of Service Profile starts automatically to servers based on the Server Pool Policies.

Service Profile association can be verified in Cisco UCS Manager > Servers > Service Profiles. Different tabs can provide details on Service profile association based on Server Pools Policy, Service Profile Template to which Service Profile is tied to, etc.



Configure Cisco Nexus 93180YC-FX Switches

The following section details the steps for the Nexus 93180YC-FX switch configuration. The details of “show run” output is listed in the Appendix.

Configure Global Settings for Cisco Nexus A and Cisco Nexus B

To set global configuration, complete the following steps on both the Nexus switches

1. Log in as admin user into the Nexus Switch A and run the following commands to set global configurations and jumbo frames in QoS:

```
conf terminal
policy-map type network-qos jumbo
class type network-qos class-default
mtu 9216
exit
class type network-qos class-fcoe
pause no-drop
mtu 2158
exit
exit
system qos
service-policy type network-qos jumbo
```

```
exit
copy run start
```

2. Log in as admin user into the Nexus Switch B and run the same above commands to set global configurations and jumbo frames in QoS.

Configure VLANs for Cisco Nexus A and Cisco Nexus B Switches

To create the necessary virtual local area networks (VLANs), complete the following steps on both Nexus switches. We created VLAN 70, 71, 72, 73 and 76. The details of “show run” output is listed in the Appendix.

1. Log in as admin user into the Nexus Switch A.

2. Create VLAN 70:

```
config terminal
VLAN 70
name InBand-Mgmt
no shutdown
exit
copy running-config startup-config
exit
```

3. Log in as admin user into the Nexus Switch B and create VLANs

Virtual Port Channel (vPC) Summary for Data and Storage Network

In the Cisco Nexus 93180YC-FX switch topology, a single vPC feature is enabled to provide HA, faster convergence in the event of a failure, and greater throughput. Cisco Nexus 93180YC-FX vPC configurations with the vPC domains and corresponding vPC names and IDs for Oracle Database Servers is shown below:

Table 7 vPC Summary

vPC Domain	vPC Name	vPC ID
70	Peer-Link	1
70	vPC Port-Channel to FI	11
70	vPC Port-Channel to FI	12

As listed in the table above, a single vPC domain with Domain ID 70 is created across two Cisco Nexus 93180YC-FX member 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 two Nexus switches in Fabric A and B.
- vPC IDs 11 and 12 are defined for traffic from Cisco UCS fabric interconnects.

Cisco Nexus 93180YC-FX Switch cabling details

Table 8 Cisco Nexus 93180YC-FX-A Cabling Information

Local Device	Local Port	Connection	Remote Device	Remote Port
Cisco Nexus 93180YC-FX Switch A	Eth1/51	40GbE	Cisco UCS fabric interconnect A	Eth1/39
	Eth1/52	40GbE	Cisco UCS fabric interconnect B	Eth1/39
	Eth1/53	40GbE	Cisco Nexus 93180YC-FX B	Eth1/53
	Eth1/54	40GbE	Cisco Nexus 93180YC-FX B	Eth1/54
	MGMT0	GbE	GbE management switch	Any

Table 9 Cisco Nexus 93180YC-FX-B Cabling Information

Local Device	Local Port	Connection	Remote Device	Remote Port
Cisco Nexus 93180YC-FX Switch B	Eth1/51	40GbE	Cisco UCS fabric interconnect A	Eth1/40
	Eth1/52	40GbE	Cisco UCS fabric interconnect B	Eth1/40
	Eth1/53	40GbE	Cisco Nexus 93180YC-FX B	Eth1/53
	Eth1/54	40GbE	Cisco Nexus 93180YC-FX B	Eth1/54
	MGMT0	GbE	GbE management switch	Any

Cisco UCS Fabric Interconnect 6332-16UP Cabling

Table 10 Cisco UCS Fabric Interconnect (FI) A Cabling Information

Local Device	Local Port	Connection	Remote Device	Remote Port
Cisco UCS FI-6332-16UP-A	FC 1/1	16G FC	Cisco MDS 9148S-A	FC 1/29
	FC 1/2	16G FC	Cisco MDS 9148S-A	FC 1/30
	FC 1/3	16G FC	Cisco MDS 9148S-A	FC 1/31
	FC ¼	16G FC	Cisco MDS 9148S-A	FC 1/32
	Eth1/17-24	40GbE	UCS 5108 Chassis IOM-A Chassis 1-4	IO Module Port1-2
	Eth1/39	40GbE	Cisco Nexus 93180YC-FX Switch A	Eth1/53

Local Device	Local Port	Connection	Remote Device	Remote Port
	Eth1/40	40GbE	Cisco Nexus 93180YC-FX Switch B	Eth1/53
	Mgmt 0	1GbE	Management Switch	Any
	L1	1GbE	Cisco UCS FI - A	L1
	L2	1GbE	Cisco UCS FI - B	L2

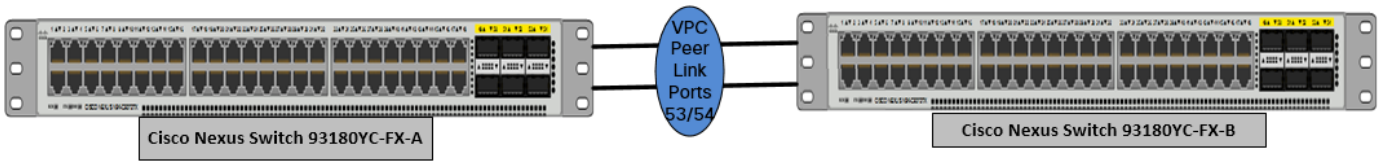
Table 11 Cisco UCS Fabric Interconnect (FI) B Cabling Information

Local Device	Local Port	Connection	Remote Device	Remote Port
Cisco UCS FI-6332-16UP-B	FC 1/1	16G FC	Cisco MDS 9148S-B	FC 1/29
	FC 1/2	16G FC	Cisco MDS 9148S-B	FC 1/30
	FC 1/3	16G FC	Cisco MDS 9148S-B	FC 1/31
	FC ¼	16G FC	Cisco MDS 9148S-B	FC 1/32
	Eth1/17-24	40GbE	UCS 5108 Chassis IOM-B Chassis 1-4	IO Module Port1-2
	Eth1/39	40GbE	Cisco Nexus 93180YC-FX Switch A	Eth1/54
	Mgmt 0	1GbE	Management Switch	Any
	L1	1GbE	Cisco UCS FI - A	L1
	L2	1GbE	Cisco UCS FI - B	L2
	Eth1/40	40GbE	Cisco Nexus 93180YC-FX Switch B	Eth1/54

Create vPC Peer-Link Between the Two Nexus Switches

To create the vPC Peer-Link, complete the following steps:

Figure 37 Nexus 93180YC-FX vPC Peer-Link



1. Log in as “admin” user into the Nexus Switch A.

For vPC 1 as Peer-link, we used interfaces 53-54 for Peer-Link. You may choose the appropriate number of ports for your needs.

To create the necessary port channels between devices, complete the following on both the Nexus Switches:

```
config terminal
feature vpc
feature lacp
vpc domain 1
peer-keepalive destination 10.29.164.234 source 10.29.164.233
exit
interface port-channel 70
description VPC peer-link
switchport mode trunk
switchport trunk allowed VLAN 1,70-73,76
spanning-tree port type network
vpc peer-link
exit
interface Ethernet1/53
description vPC-PeerLink
switchport mode trunk
switchport trunk allowed VLAN 1, 70-73,76
channel-group 70 mode active
no shutdown
exit
interface Ethernet1/54
description vPC-PeerLink
switchport mode trunk
switchport trunk allowed VLAN 1, 70-73,76
channel-group 70 mode active
no shutdown
exit
```

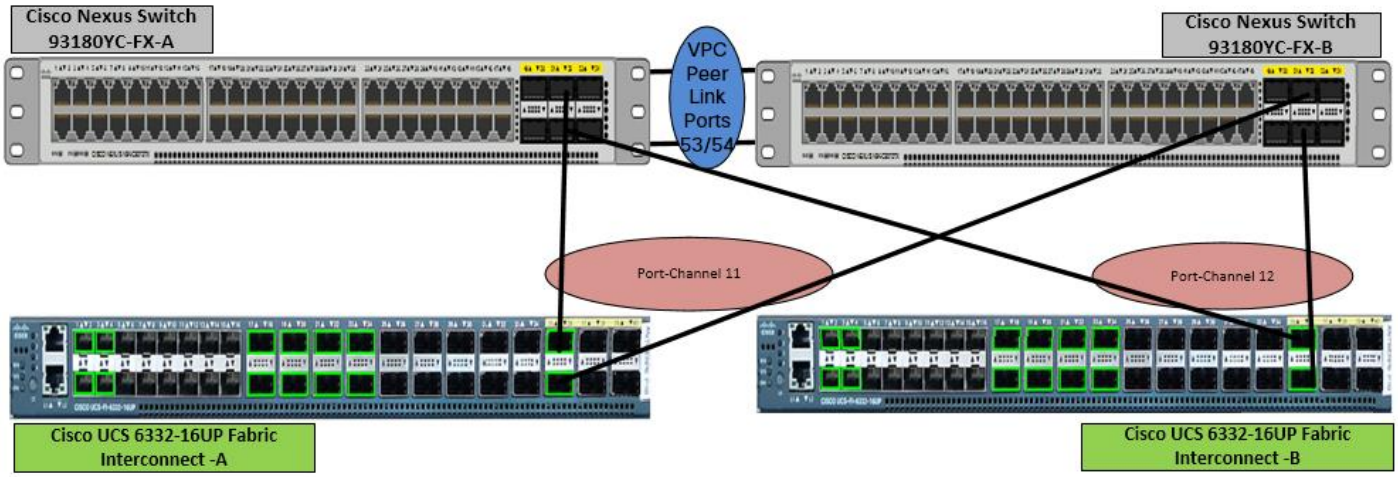
2. Log in as admin user into the Nexus Switch B and repeat the above steps to configure second nexus switch.

Make sure to change peer-keepalive destination and source IP address appropriately for Nexus Switch B.

Create vPC Configuration Between Nexus 9372PX-E and Fabric Interconnects

Create and configure vPC 11 and 12 for Data network between Nexus switches and Fabric Interconnects.

Figure 38 vPC Configuration Between Nexus Switches and Fabric Interconnects



To create the necessary port channels between devices, complete the following steps on both Nexus Switches:

1. Log in as admin user into Nexus Switch A and enter the following:

```
config Terminal
interface port-channel11
description FI-A-Uplink
switchport mode trunk
switchport trunk allowed VLAN 1,70-73,76
spanning-tree port type edge trunk
vpc 11
no shutdown
exit
interface port-channel12
description FI-B-Uplink
switchport mode trunk
switchport trunk allowed VLAN 1,70-73,76
```

```
spanning-tree port type edge trunk
vpc 12
no shutdown
exit
interface Ethernet1/51
description FI-A-Uplink
switch mode trunk
switchport trunk allowed vlan 1,70-73,76
spanning-tree port type edge trunk
mtu 9216
channel-group 11 mode active
no shutdown
exit
interface Ethernet1/52
description FI-B-Uplink
switch mode trunk
switchport trunk allowed vlan 1,70-73,76
spanning-tree port type edge trunk
mtu 9216
channel-group 12 mode active
no shutdown
exit
copy running-config startup-config
```

2. Log in as admin user into the Nexus Switch B and complete the following for the second switch configuration:

```
config Terminal
interface port-channel11
description FI-A-Uplink
switchport mode trunk
switchport trunk allowed VLAN 1,70-73,76
spanning-tree port type edge trunk
vpc 11
```

```
no shutdown
exit
interface port-channel12
description FI-B-Uplink
switchport mode trunk
switchport trunk allowed VLAN 1,70-73,76
spanning-tree port type edge trunk
vpc 12
no shutdown
exit
interface Ethernet1/51
description FI-A-Uplink
switch mode trunk
switchport trunk allowed vlan 1,70-73,76
spanning-tree port type edge trunk
mtu 9216
channel-group 11 mode active
no shutdown
exit
interface Ethernet1/52
description FI-B-Uplink
switch mode trunk
switchport trunk allowed vlan 1,70-73,76
spanning-tree port type edge trunk
mtu 9216
channel-group 12 mode active
no shutdown
exit
copy running-config startup-config
```

Verify All vPC Status is Up on Both Cisco Nexus Switches

Figure 39 vPC Description for Cisco Nexus Switch A and B

```
AAD17-NX9K-A# sh vpc brief
Legend:
                (*) - local vPC is down, forwarding via vPC peer-link

vPC domain id                     : 70
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         : 4
Peer Gateway                      : Disabled
Dual-active excluded VLANs        : -
Graceful Consistency Check        : Enabled
Auto-recovery status              : Disabled
Delay-restore status              : Timer is off.(timeout = 30s)
Delay-restore SVI status          : Timer is off.(timeout = 10s)
Operational Layer3 Peer-router    : Disabled

vPC Peer-link status
---------------------------------------------------------------------
id    Port   Status Active vlans
--    ----   ------ -------------------------------------------------
1     Po70   up     1,70-76

vPC status
----------------------------------------------------------------------------
Id    Port          Status Consistency Reason                Active vlans
--    ------------  ------ ----------- ------                ---------------
11    Po11          up     success     success               1,70-76

12    Po12          up     success     success               1,70-76

13    Po13          up     success     success               1,70-76

14    Po14          up     success     success               1,70-76
```

```
AAD17-NX9K-B# sh vpc brief
Legend:
                (*) - local vPC is down, forwarding via vPC peer-link

vPC domain id                     : 70
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         : 4
Peer Gateway                      : Disabled
Dual-active excluded VLANs        : -
Graceful Consistency Check        : Enabled
Auto-recovery status              : Disabled
Delay-restore status              : Timer is off.(timeout = 30s)
Delay-restore SVI status          : Timer is off.(timeout = 10s)
Operational Layer3 Peer-router    : Disabled

vPC Peer-link status
---------------------------------------------------------------------
id    Port   Status Active vlans
--    ----   ------ -------------------------------------------------
1     Po70   up     1,70-76

vPC status
----------------------------------------------------------------------------
Id    Port          Status Consistency Reason                Active vlans
--    ------------  ------ ----------- ------                ---------------
11    Po11          up     success     success               1,70-76

12    Po12          up     success     success               1,70-76

13    Po13          up     success     success               1,70-76

14    Po14          up     success     success               1,70-76
```

Cisco MDS 9148S FC Switch Configuration

Figure 40 illustrates the cable connectivity between the Cisco MDS 9148S and the Cisco 6332 Fabric Interconnects and Pure Storage FlashArray //X70 storage.

Figure 40 Fibre Channel Cable Connectivity from Pure Storage FlashArray //X70 to Cisco MDS 9148S to Cisco 6332-16UP Fabric Interconnects

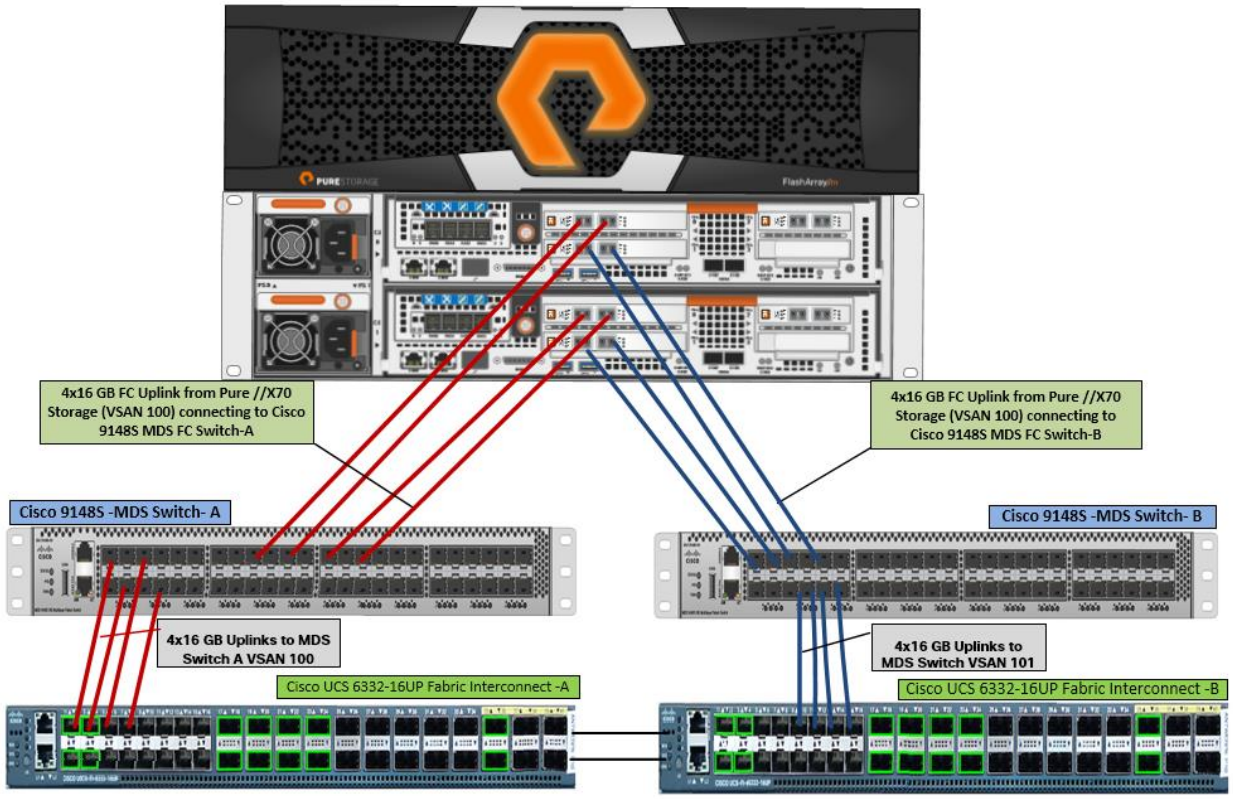


Table 12 Cisco MDS 9148S-A Cabling Information

Local Device	Local Port	Connection	Remote Device	Remote Port
Cisco MDS 9148S-A	FC1/29	16Gb FC	Pure Storage //X70 Controller 00	P00
	FC1/30	16Gb FC	PureStorage //X70 Controller 01	P01
	FC1/31	16Gb FC	Pure Storage //X70 Controller 00	P02
	FC1/32	16Gb FC	Pure Storage //X70 Controller 01	P03
	FC1/25	16Gb FC	Cisco 6332-16UP Fabric Interconnect-A	FC1/1
	FC1/26	16Gb FC	Cisco 6332-16UP Fabric Interconnect-A	FC1/2
	FC1/27	16Gb FC	Cisco 6332-16UP Fabric Interconnect-A	FC1/3
	FC1/28	16Gb FC	Cisco 6332-16UP Fabric Interconnect-A	FC1/4

Table 13 Cisco MDS 9148S-B Cabling Information

Local Device	Local Port	Connection	Remote Device	Remote Port

Local Device	Local Port	Connection	Remote Device	Remote Port
Cisco MDS 9148S-B	FC1/29	16Gb FC	Pure Storage //X70 Controller 01	P00
	FC1/30	16Gb FC	Pure Storage //X70 Controller 00	P01
	FC1/31	16Gb FC	Pure Storage //X70 Controller 01	P02
	FC1/32	16Gb FC	Pure Storage //X70 Controller 00	P03
	FC1/25	16Gb FC	Cisco 6332-16UP Fabric Interconnect-B	FC1/1
	FC1/26	16Gb FC	Cisco 6332-16UP Fabric Interconnect-B	FC1/2
	FC1/27	16Gb FC	Cisco 6332-16UP Fabric Interconnect-B	FC1/3
	FC1/28	16Gb FC	Cisco 6332-16UP Fabric Interconnect-B	FC1/4

We used four 16Gb FC connections from each Fabric Interconnect to each MDS switch. We utilized four 16Gb FC connections from the Pure Storage //X70 array controller to each MDS switch.

Pure Storage FlashArray //X70 to MDS SAN Fabric Connectivity

Pure Storage //X70 to MDS A and B Switches using VSAN 100 for Fabric A and VSAN 101 Configured for Fabric B.

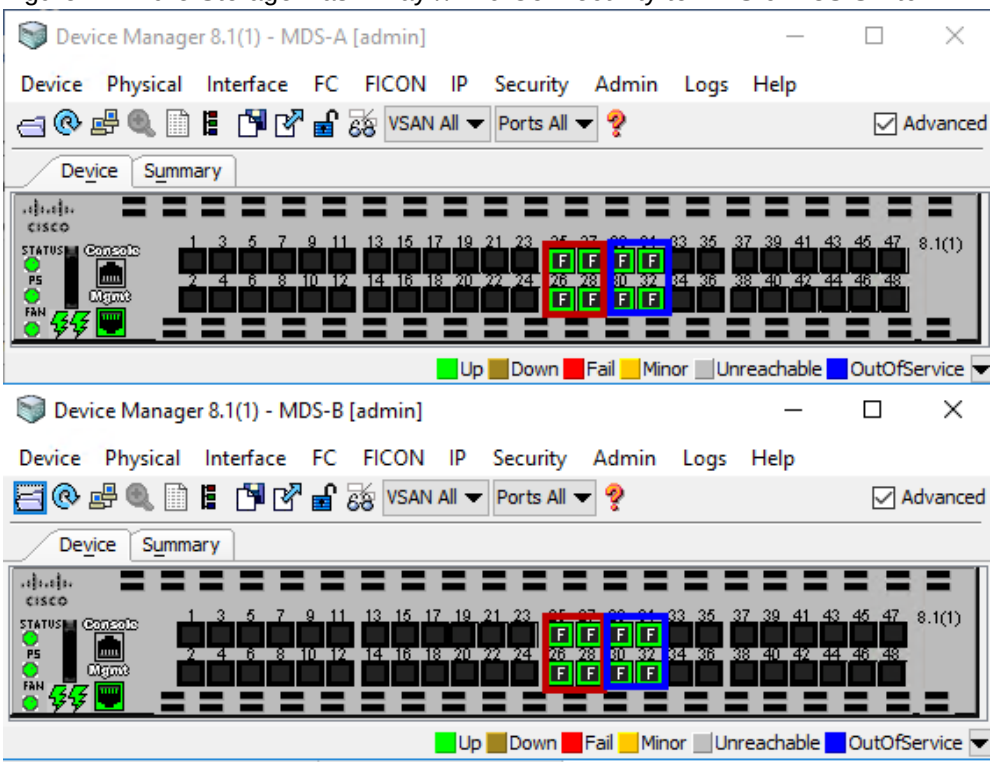
For this solution, we connected four ports (ports 25 to 28) of MDS Switch A to Pure Storage System. Similarly, we connected four ports (ports 25 to 28) of MDS Switch B to Pure Storage System as shown in the table below. All ports carry 16 Gb/s FC Traffic.

Table 14 MDS 9148S Port Connection to Pure Storage System

Local Device	Local Port	Connection	Remote Device	Remote Port
MDS Switch A	FC1/29	16Gb FC	Pure Storage //X70 Controller 01	CT0-FC0
	FC1/30	16Gb FC	Pure Storage //X70 Controller 01	CT1-FC1
	FC1/31	16Gb FC	Pure Storage //X70 Controller 02	CT0-FC2
	FC1/32	16Gb FC	Pure Storage //X70 Controller 02	CT1-FC3
MDS Switch B	FC1/29	16Gb FC	Pure Storage //X70 Controller 01	CT1-FC0
	FC1/30	16Gb FC	Pure Storage //X70 Controller 01	CT0-FC1

Local Device	Local Port	Connection	Remote Device	Remote Port
	FC1/31	16Gb FC	Pure Storage //X70 Controller 02	CT1-FC2
	FC1/32	16Gb FC	Pure Storage //X70 Controller 02	CT0-FC3

Figure 41 Pure Storage FlashArray //X70 Connectivity to MDS 9148S Switch



Configure Feature for MDS Switch A and MDS Switch B

To set feature on MDS Switches, complete the following steps on both MDS switches:

1. Log in as admin user into MDS Switch A:

```
config terminal
feature npiv
feature telnet
switchname FlashStack-MDS-A
copy running-config startup-config
```

2. Log in as admin user into MDS Switch B. Repeat the steps above on MDS Switch B.

Configure VSANs for MDS Switch A and MDS Switch B

To create VSANs, complete the following steps on both MDS switches:

1. Log in as admin user into MDS Switch A. Create VSAN 100 for Storage Traffic:

```
config terminal
VSAN database
vsan 100
vsan 100 interface fc 1/25-36
exit
interface fc 1/25-36
switchport trunk allowed vsan 100
switchport trunk mode off
port-license acquire
no shutdown
exit
copy running-config startup-config
```

2. Log in as admin user into MDS Switch B. Create VSAN 101 for Storage Traffic as per the steps outlined for create VSAN 100 for MDS Switch A:

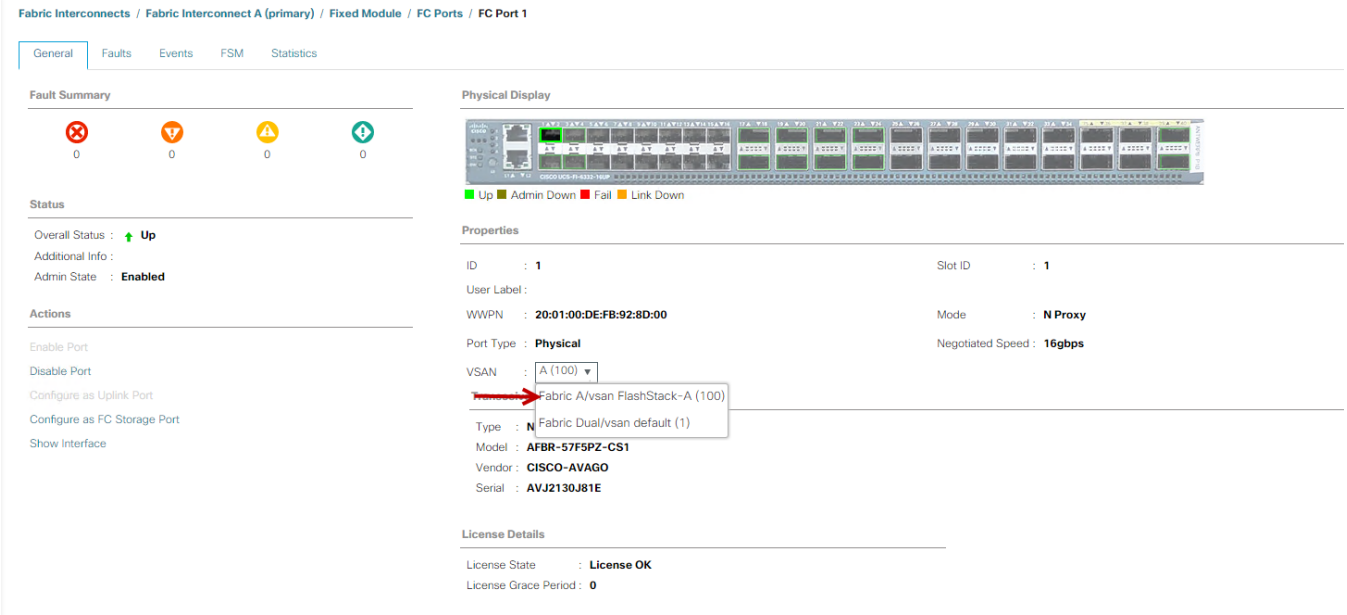
```
config terminal
VSAN database
vsan 101
vsan 101 interface fc 1/25-36
exit
interface fc 1/25-36
switchport trunk allowed vsan 100
switchport trunk mode off
port-license acquire
no shutdown
exit
copy running-config startup-config
```

Add FC Uplink ports to corresponding VSAN on Fabric Interconnect

1. In Cisco UCS Manager, in the Equipment tab, select Fabric Interconnects > Fabric Interconnect A > Fixed Module > FC Ports.

2. Select FC Port 1, on right side from the drop-down list for VSAN, Select VSAN 100.

Figure 42 VSAN Assignment on FC Uplink Ports to MDS Switch



3. Repeat these steps to Add FC Port 1-4 to VSAN 100 on Fabric A and FC Port 1-4 to VSAN 101 on Fabric B.

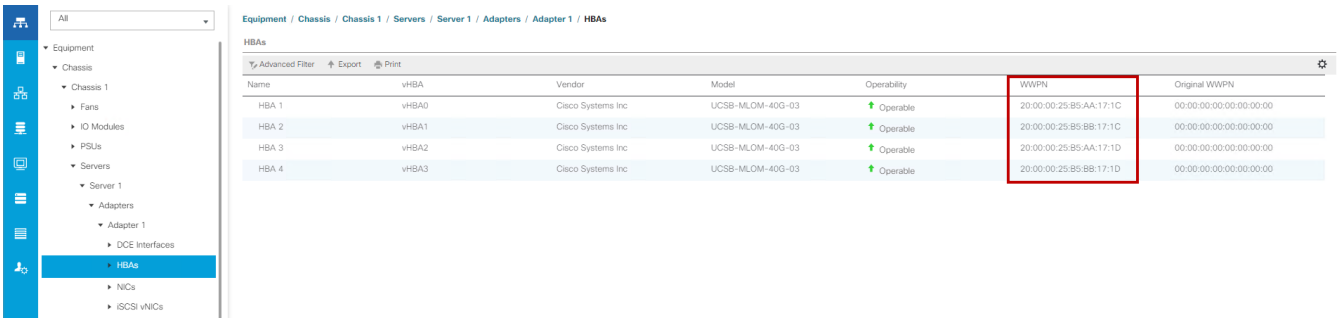
Create and Configure Fiber Channel Zoning

This procedure sets up the Fibre Channel connections between the Cisco MDS 9148S switches, the Cisco UCS Fabric Interconnects, and the Pure Storage FlashArray systems.

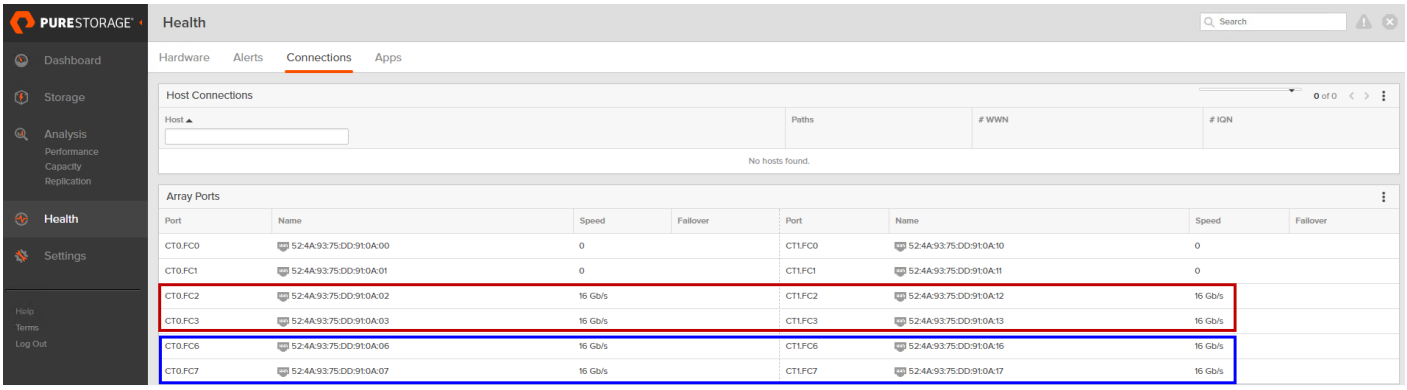
Before you configure the zoning details, decide how many paths are needed for each LUN and extract the WWPN numbers for each of the HBAs from each server. We used 4 HBAs for each Server. Two HBAs (HBA0 and HBA2) are connected to MDS Switch-A and other two HBAs (HBA1 and HBA3) are connected to MDS Switch-B.

To create and configure the fiber channel zoning, complete the following steps:

1. Log into the Cisco UCS Manager > Equipment > Chassis > Servers and select the desired server. On the right hand menu, click the Inventory tab and HBA's sub-tab to get the WWPN of HBA's as shown in the screenshot below:



2. Connect to the Pure Storage System and extract the WWPN of FC Ports connected to the Cisco MDS Switches. We have connected 8 FC ports from Pure Storage System to Cisco MDS Switches. FC ports CT0.FC2, CT1.FC2, CT0.FC3, CT1.FC3 are connected to MDS Switch-A and similarly FC ports CT0.FC6, CT1.FC6, CT0.FC7, CT1.FC7 are connected to MDS Switch-B.



Create Device Aliases for Fiber Channel Zoning

Cisco MDS Switch A

To configure device aliases and zones for the SAN boot paths as well as datapaths of MDS switch A, complete the following steps. The Appendix section regarding MDS 9148S switch provides detailed information about the “show run” configuration.

1. Log in as admin user and run the following commands:

```
conf t
device-alias database
device-alias name VCC-WLHost01-HBA0 pwwn 20:00:00:25:B5:AA:17:00
device-alias name VCC-WLHost01-HBA2 pwwn 20:00:00:25:B5:AA:17:01
device-alias name FLASHSTACK-X-CT0-FC0 pwwn 52:4a:93:75:dd:91:0a:00
device-alias name FLASHSTACK-X-CT0-FC2 pwwn 52:4a:93:75:dd:91:0a:02
device-alias name FLASHSTACK-X-CT1-FC1 pwwn 52:4a:93:75:dd:91:0a:11
device-alias name FLASHSTACK-X-CT1-FC3 pwwn 52:4a:93:75:dd:91:0a:13
```

Cisco MDS Switch B

To configure device aliases and zones for the SAN boot paths as well as datapaths of MDS switch B, complete the following steps:

1. Log in as admin user and run the following commands:

```
conf t
device-alias database
device-alias name VCC-WLHost01-HBA1 pwwn 20:00:00:25:B5:AA:17:00
```

```
device-alias name VCC-WLHost01-HBA3 pwwn 20:00:00:25:B5:AA:17:01
device-alias name FLASHSTACK-X-CT0-FC1 pwwn 52:4a:93:75:dd:91:0a:01
device-alias name FLASHSTACK-X-CT0-FC3 pwwn 52:4a:93:75:dd:91:0a:03
device-alias name FLASHSTACK-X-CT1-FC0 pwwn 52:4a:93:75:dd:91:0a:10
device-alias name FLASHSTACK-X-CT1-FC2 pwwn 52:4a:93:75:dd:91:0a:12
```

Create Zoning

Cisco MDS Switch A

To configure zones for the MDS switch A, complete the following steps:

1. Create a zone for each service profile.
2. Login as admin user and create the zone as shown below:

```
conf t
zone name FlaskStack-VCC-CVD-WLHost01 vsan 100
member pwwn 52:4a:93:75:dd:91:0a:00
member pwwn 52:4a:93:75:dd:91:0a:02
member pwwn 52:4a:93:75:dd:91:0a:11
member pwwn 52:4a:93:75:dd:91:0a:13
member pwwn 20:00:00:25:B5:AA:17:00
member pwwn 20:00:00:25:B5:AA:17:01
conf t
zoneset name FlashStack-VCC-CVD vsan 100
member FlaskStack-VCC-CVD-WLHost01
```

3. After the zone for the Cisco UCS service profile has been created, create the zone set and add the necessary members:

```
conf t
zoneset name FlaskStack-VCC-CVD vsan 100
member FlaskStack-VCC-CVD-WLHost01
```

4. Activate the zone set by running following commands:

   zoneset activate name FlaskStack-VCC-CVD vsan 100

   exit

   copy running-config startup-config

Configure Pure Storage

The design goal of the reference architecture is to best represent a real-world environment as closely as possible. The approach included features of Cisco UCS to rapidly deploy stateless servers and use Pure Storage FlashArray's boot LUNs to provision the O.S on top it. Zoning was performed on the Cisco MDS 9148S switches to enable the initiators discover the targets during boot process.

A Service Profile was created within Cisco UCS Manager to deploy the thirty-two servers quickly with a standard configuration. SAN boot volumes for these servers were hosted on the same Pure Storage FlashArray //X70. Once the stateless servers were provisioned, following process was performed to enable Rapid deployment of thirty-two nodes.

Each Server node has dedicated single LUN to install operating system and all the thirty-two server node was booted off SAN. For this solution, we have installed vSphere ESXi 6.5 U1 Cisco Custom ISO on this LUNs to create thirty node based VMware Horizon 7 solution.

Using logical servers that are disassociated from the physical hardware removes many limiting constraints around how servers are provisioned. Cisco UCS Service Profiles contain values for a server's property settings, including virtual network interface cards (vNICs), MAC addresses, boot policies, firmware policies, fabric connectivity, external management, and HA information. The service profiles represent all the attributes of a logical server in Cisco UCS model. By abstracting these settings from the physical server into a Cisco Service Profile, the Service Profile can then be deployed to any physical compute hardware within the Cisco UCS domain. Furthermore, Service Profiles can, at any time, be migrated from one physical server to another. Furthermore, Cisco is the only hardware provider to offer a truly unified management platform, with Cisco UCS Service Profiles and hardware abstraction capabilities extending to both blade and rack servers.

In addition to the service profiles, the use of Pure Storage's FlashArray's with SAN boot policy provides the following benefits:

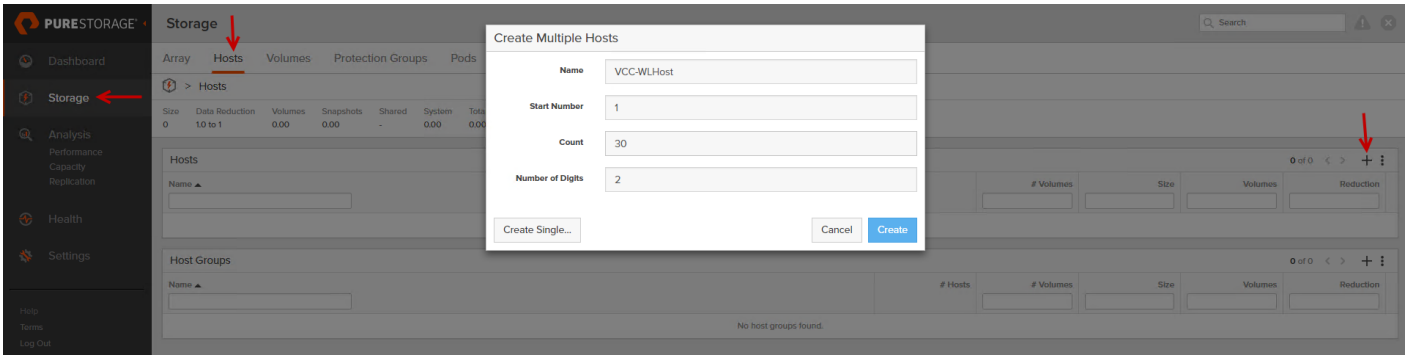
- Scalability - Rapid deployment of new servers to the environment in a very few steps.
- Manageability - Enables seamless hardware maintenance and upgrades without any restrictions. This is a huge benefit in comparison to other appliance model like Exadata.
- Flexibility - Easy to repurpose physical servers for different applications and services as needed.
- Availability - Hardware failures are not impactful and critical. In rare case of a server failure, it is easier to associate the logical service profile to another healthy physical server to reduce the impact.

Configure Host

Before using a volume (LUN) on a host, the host has to be defined on Pure FlashArray. To set up a host complete the following steps:

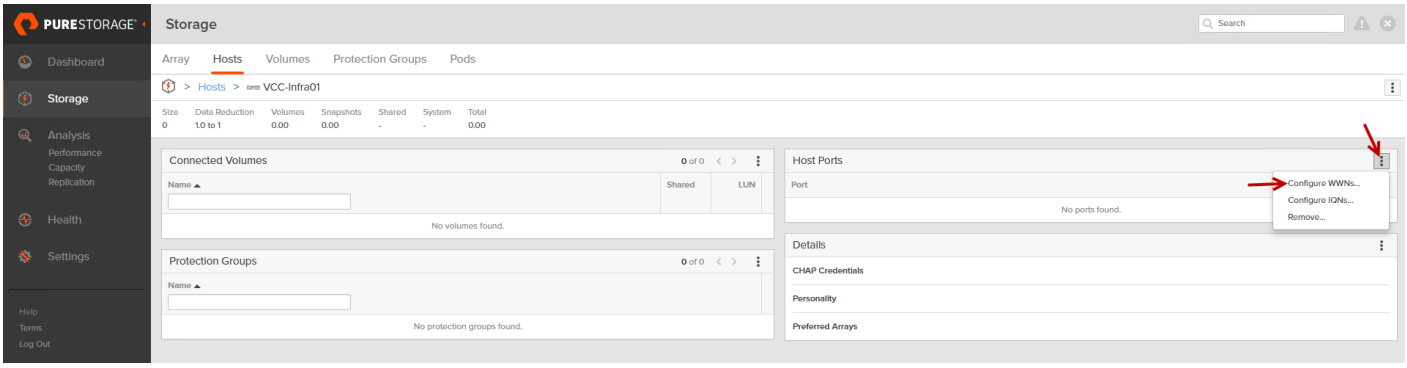
1. Log into Pure Storage dashboard.
2. In the PURE GUI, go to Storage tab.
3. Under Hosts option in the left frame, click the + sign to create a host.

4. Enter the name of the host or select Create Multiple and click Create. This will create a Host entry(s) under the Hosts category.

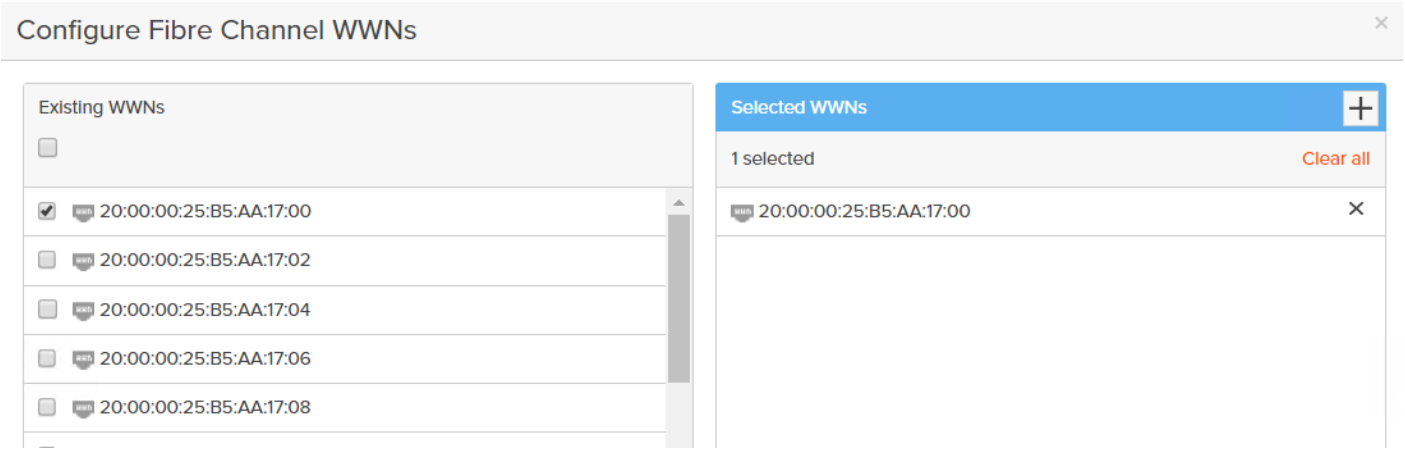


5. To update the host with the connectivity information by providing the Fibre Channel WWNs or iSCSI IQNs, click the Host that was created.

6. In the host context, click the Host Ports tab and click the settings button and select “Configure Fibre Channel WWNs” which will display a window with the available WWNs in the left side.



7. Select the list of WWNs that belongs to the host in the next window and click “Confirm.”



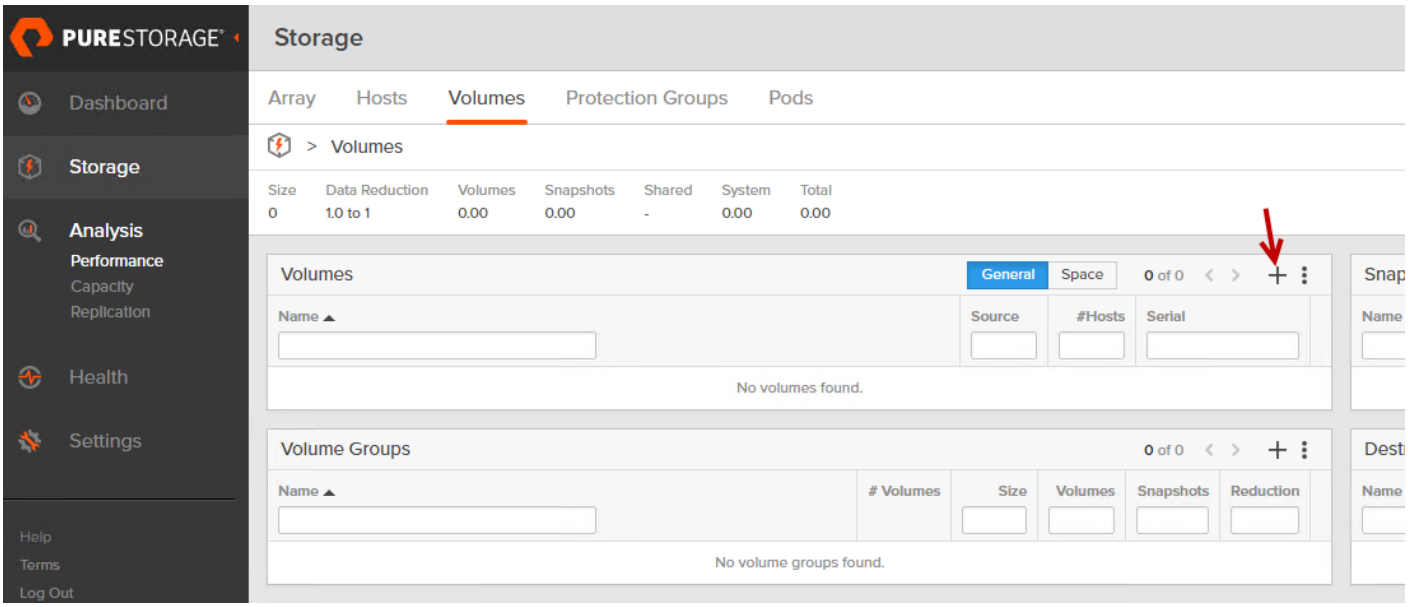
Make sure the zoning has been setup to include the WWNs details of the initiators along with the target, without which the SAN boot will not work.

WWNs will appear only if the appropriate FC connections were made and the zones were setup on the underlying FC switch.

Configure Volume

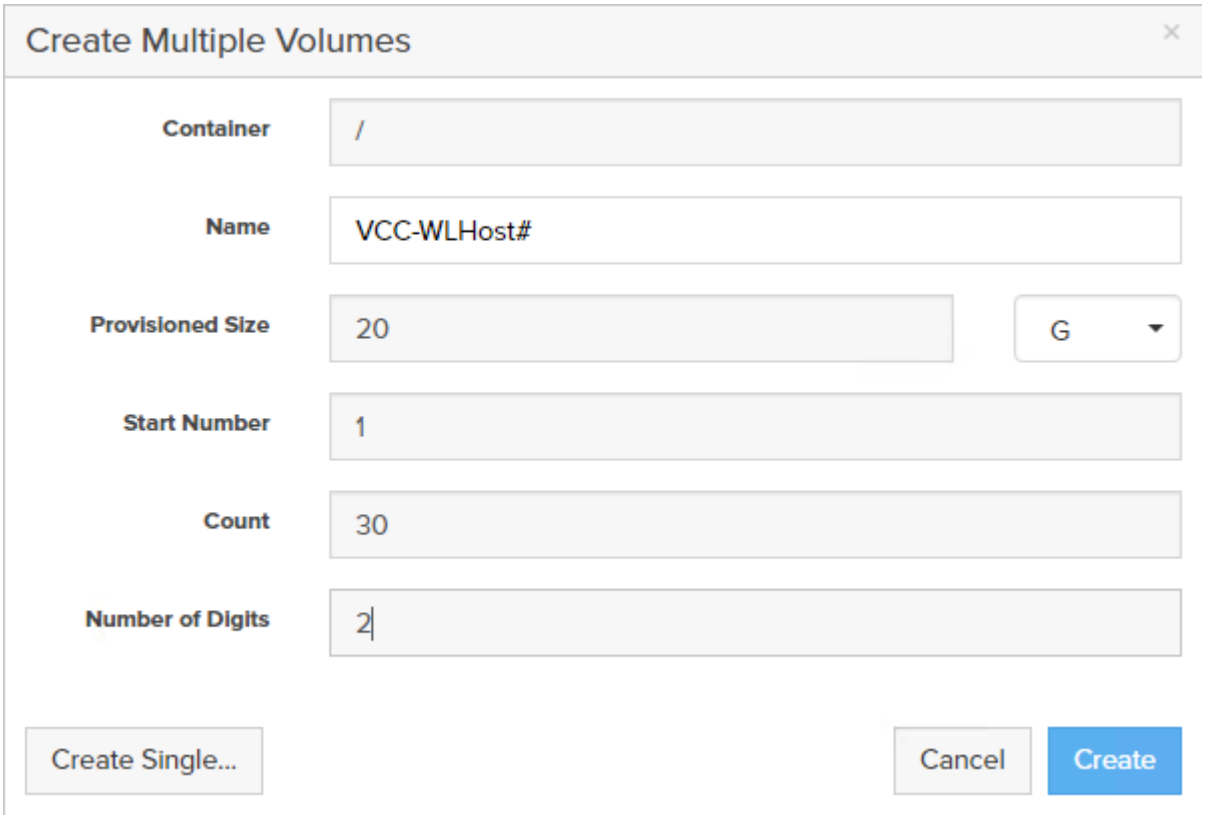
To configure a volume, complete the following steps:

1. Go to the Storage tab > Volumes > and click the + sign to “Create Volume.”

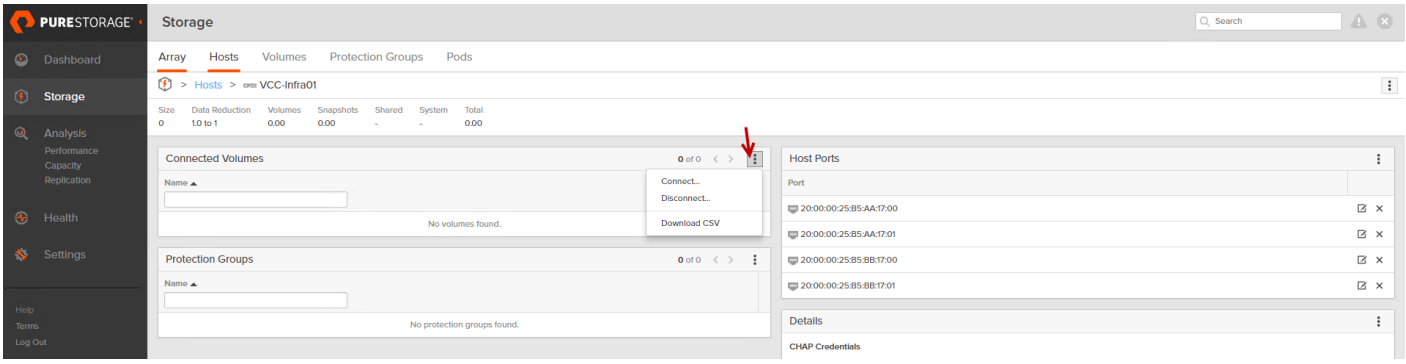


2. Provide the name of the volume, size, choose the size type (KB, MB, GB, TB, PB) and click Create to create the volume. Example creating 32 SAB boot Volume for 32 B200 M5 server configure in this solution.

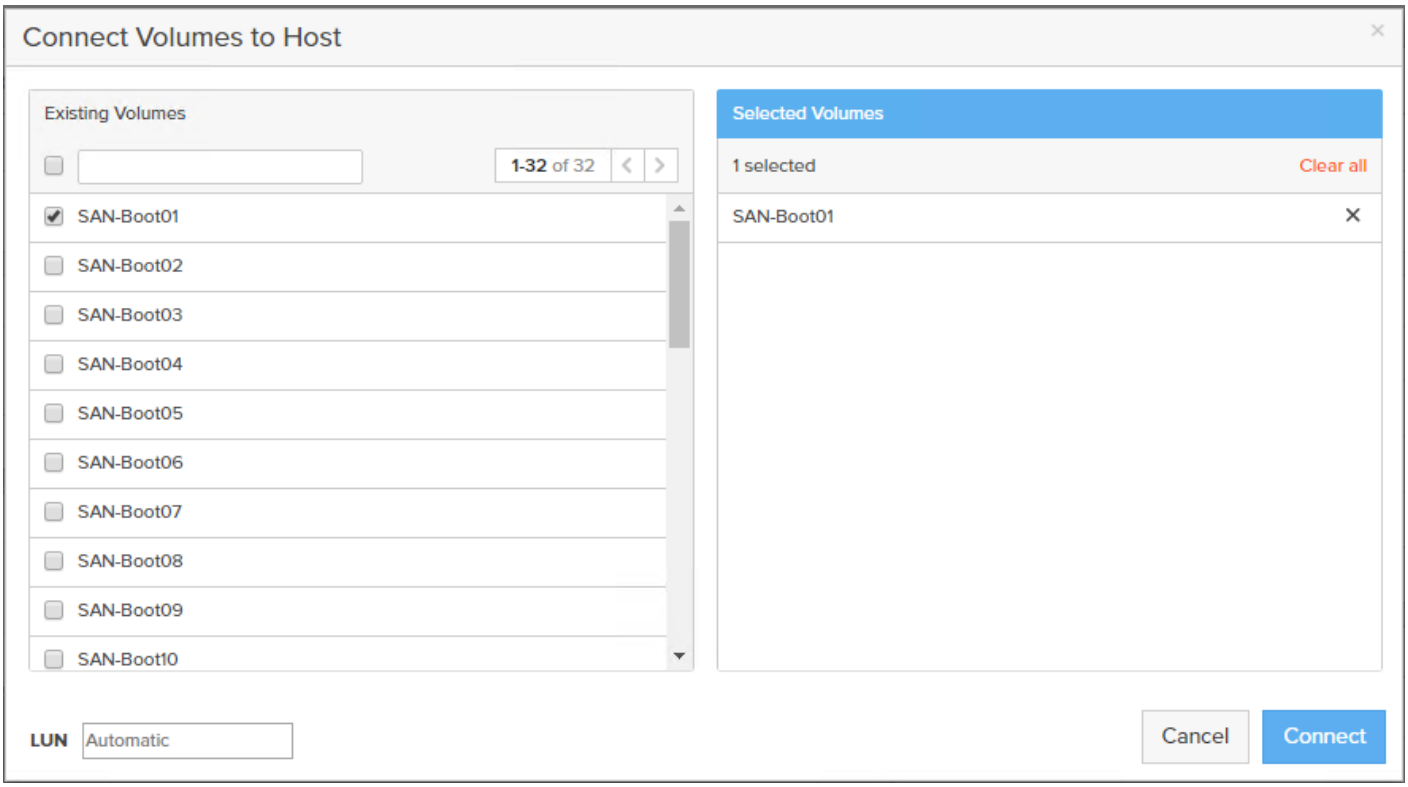
3. Two for Infrastructure and remaining thirty Servers for Horizon workload test.



4. Attach the volume to a host by going to the “Connected Hosts and Host Groups” tab under the volume context menu.



5. Select Connect; In the Connect Volumes to Host wizard select SAN-BootXX volume, click Connect.



Make sure the SAN Boot Volumes has the LUN ID “1” since this is important while configuring Boot from SAN. You will also configure the LUN ID as “1” when configuring Boot from SAN policy in Cisco UCS Manager.

More LUNs can be connected by adding connection to existing or new volume(s) to an existing node.

Install Pure Storage vSphere Web Client Plugin

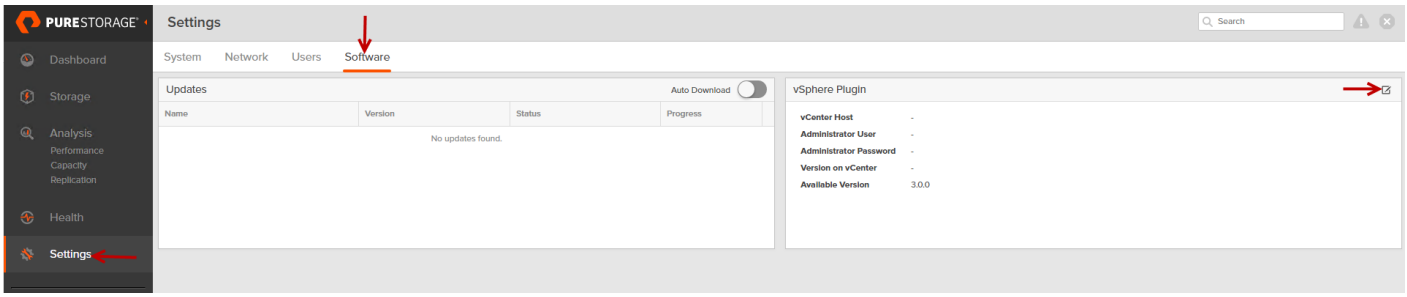
Pure Storage offers a direct plugin to the vSphere Web Client that allows for full management of a FlashArray as well as a variety of integrated menu options to provision and manage storage.

Prior to use, the Web Client Plugin must be installed and configured on the target vCenter server. There is no requirement to go to an external web site to download the plugin–it is stored on the FlashArray controllers.

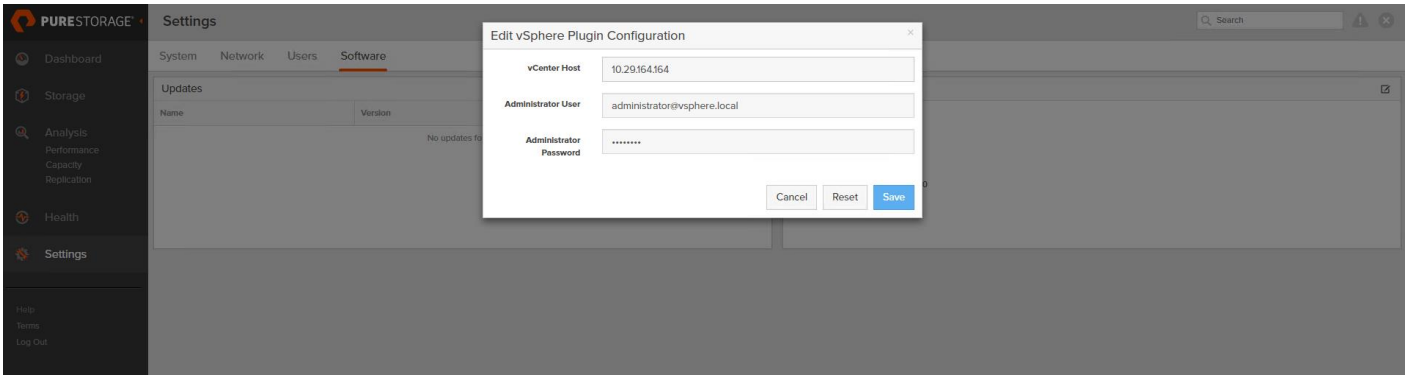
The Pure Storage vSphere Web Client Plugin is supported with both Windows and Linux-based vCenter servers. For vSphere 6.0, the 2.0.6 version or later of the plugin is required.

To install Pure Storage Plugin for vSphere Web-Client, complete the following steps:

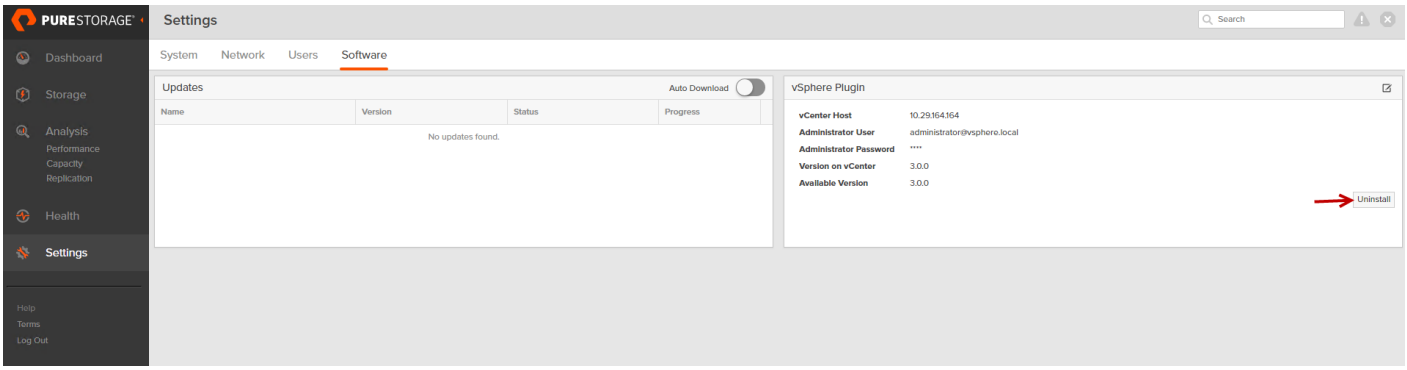
1. Log into the Pure Storage GUI and navigate to the Settings > Software > vSphere Plugin.



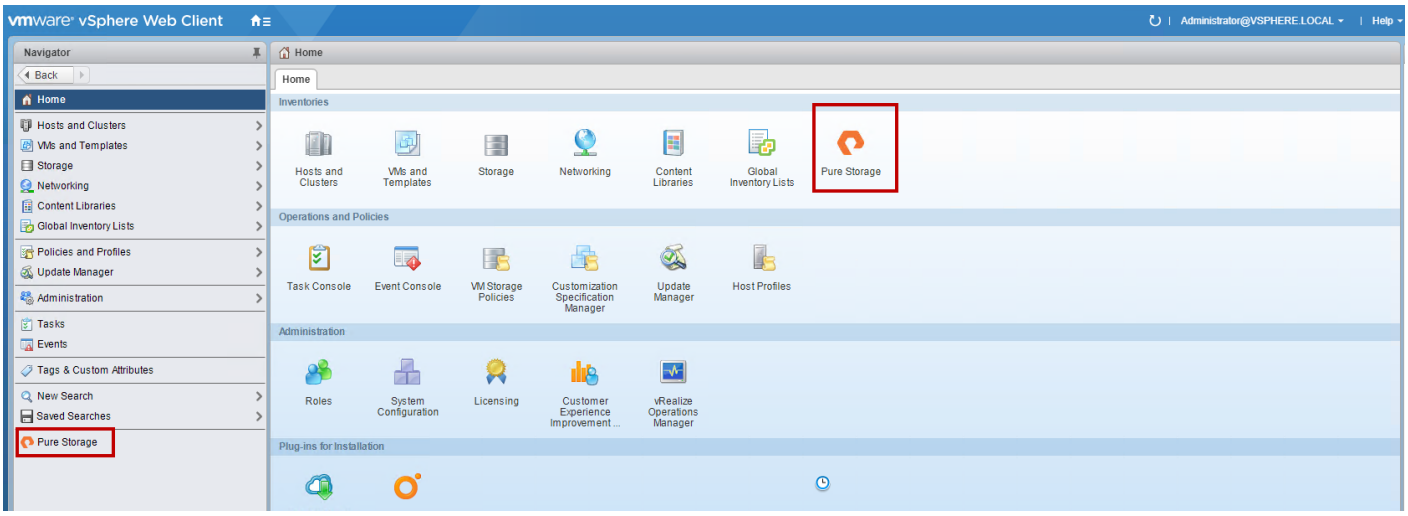
2. Enter the vCenter IP address or FQDN and the appropriate administrative credentials and click Save.



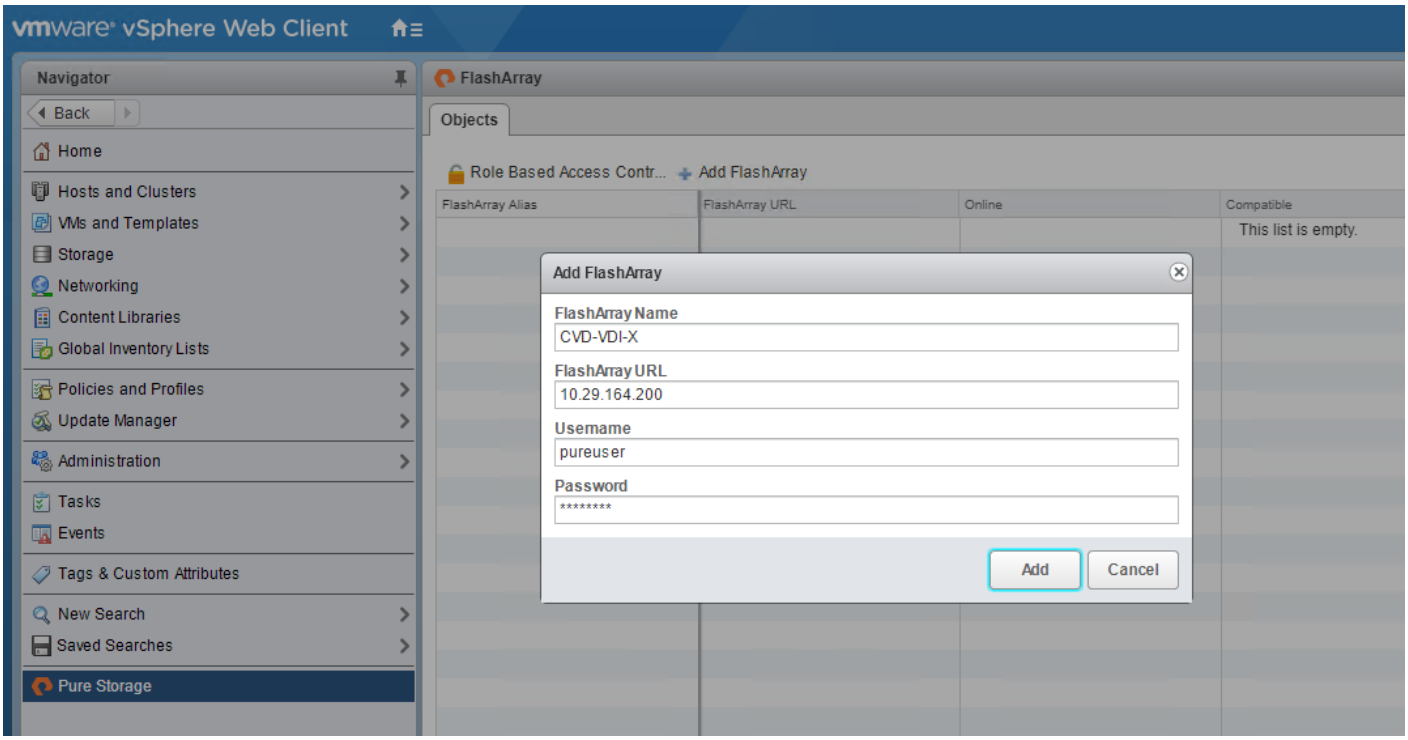
3. Click “Install” for a new Web Client install or “Update” to upgrade from an older version of the plugin.



When the plugin has been configured, it can be accessed from the vSphere Web Client interface. Login to the respective instance of the Web Client, navigate to the home screen and a new Pure Storage icon will be listed under “Inventories.”



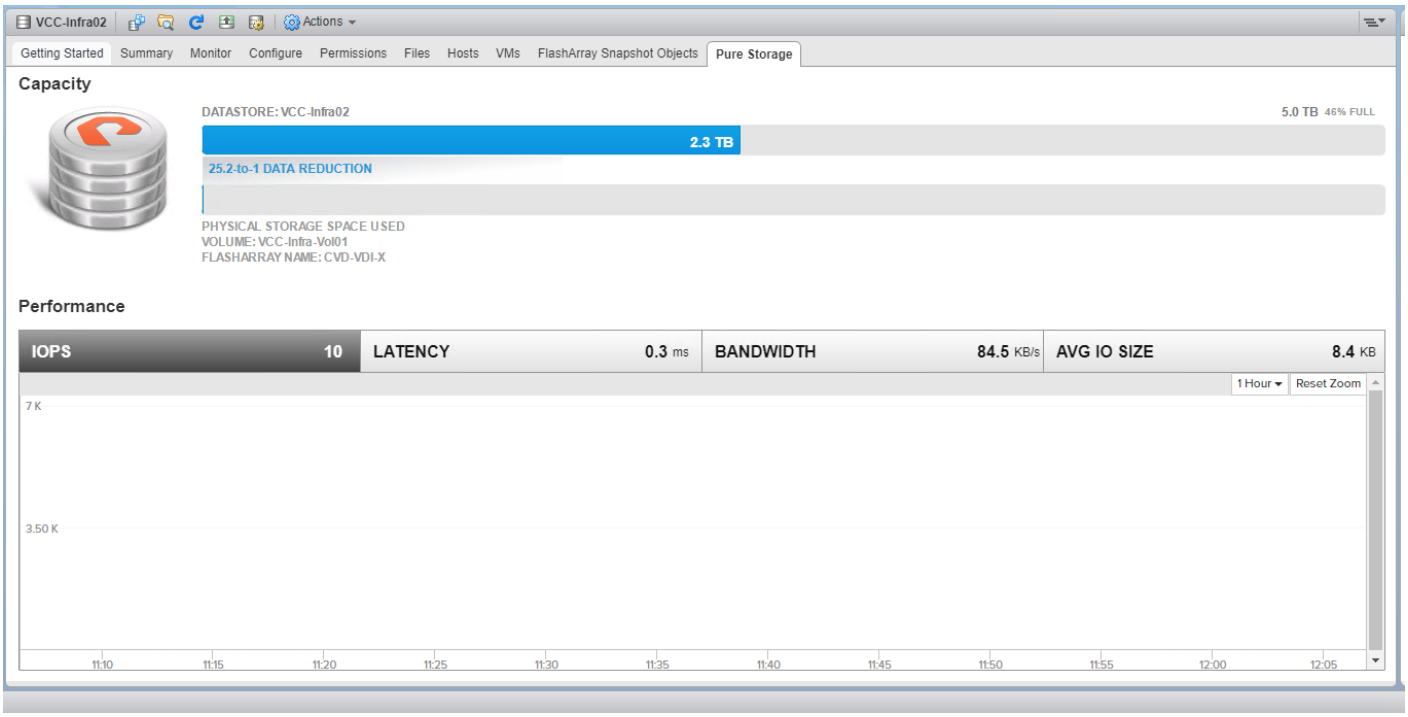
Before the plugin can be used, one or more arrays must be added. Individual FlashArrays can be added from within the home screen of the plugin by supplying a name, URL and credentials for the target array. Note that the plugin only needs to be installed once from a FlashArray on a given vCenter instance. Once it has been installed any number of other FlashArrays can be configured for use from within the plugin. Always use the virtual IP address of the FlashArray instead of a direct IP address of a controller. This will provide for Web Client connection resiliency.



From the Pure Storage plugin home page, individual arrays can be authorized for access.

One of the most basic features of the plugin is the ability to see underlying storage information in-context of a given datastore residing on a FlashArray. This is built directly into the standard datastore views in the Web Client. In the vSphere Web Client, a new tab will appear for datastores called Pure Storage.

From that tab, data reduction rates and performance information for that device is displayed.



For detaled information, see Pure Storage Plugin for vCenter WebClient.

Installing and Configuring VMware ESXi 6.5

This section provides detailed instructions for installing VMware ESXi 6.5 Update1 in an environment.

Several methods exist for installing ESXi in a VMware environment. These procedures focus on how to use the built-in keyboard, video, mouse (KVM) console and virtual media features in Cisco UCS Manager to map remote installation media to individual servers and install ESXi on boot logical unit number (LUN). Upon completion of steps outlined here ESXi hosts will be booted from their corresponding SAN Boot LUNs.

Download Cisco Custom Image for ESXi 6.5 Update 1

To download the Cisco Custom Image for ESXi 6.5 Update 1, from the VMware vSPhere Hypervisor 6.5 U1 page click “Custom ISOs” tab.

This ESXi 6.5 Cisco custom image includes updates for the fNIC and neNIC drivers. The versions that are part of this image are: neNIC: 1.0.6.0-1OEM.650.0.0.4598673; fNIC: 1.6.0.34, Build: 2494585

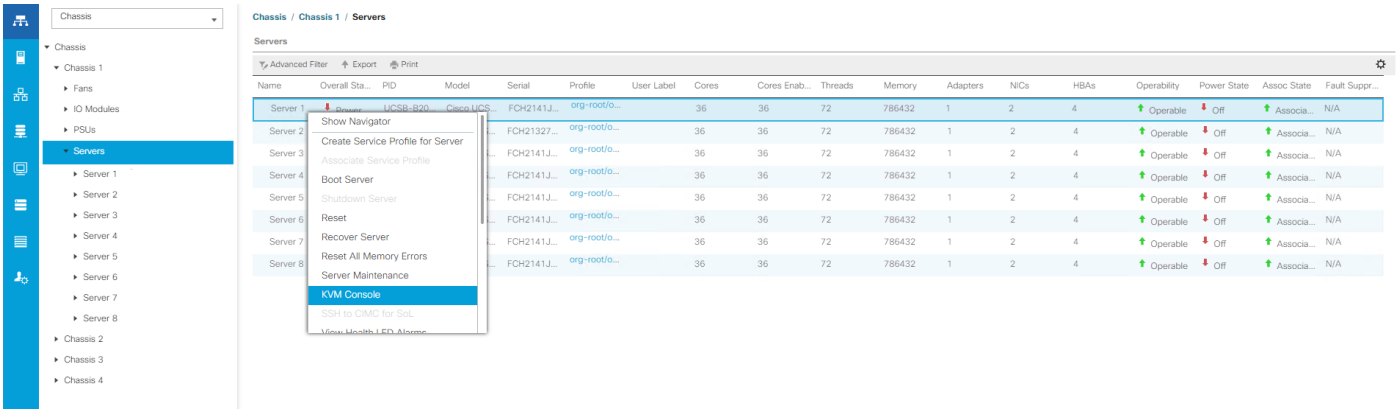
Install VMware vSphere ESXi 6.5

In order to install VMware vSphere ESXi hypervisor on Cisco UCS Server complete following steps:

1. In the Cisco UCS Manager navigation pane, click the Equipment tab.

2. Under Equipment > Chassis > Chassis 1 > Server 1.

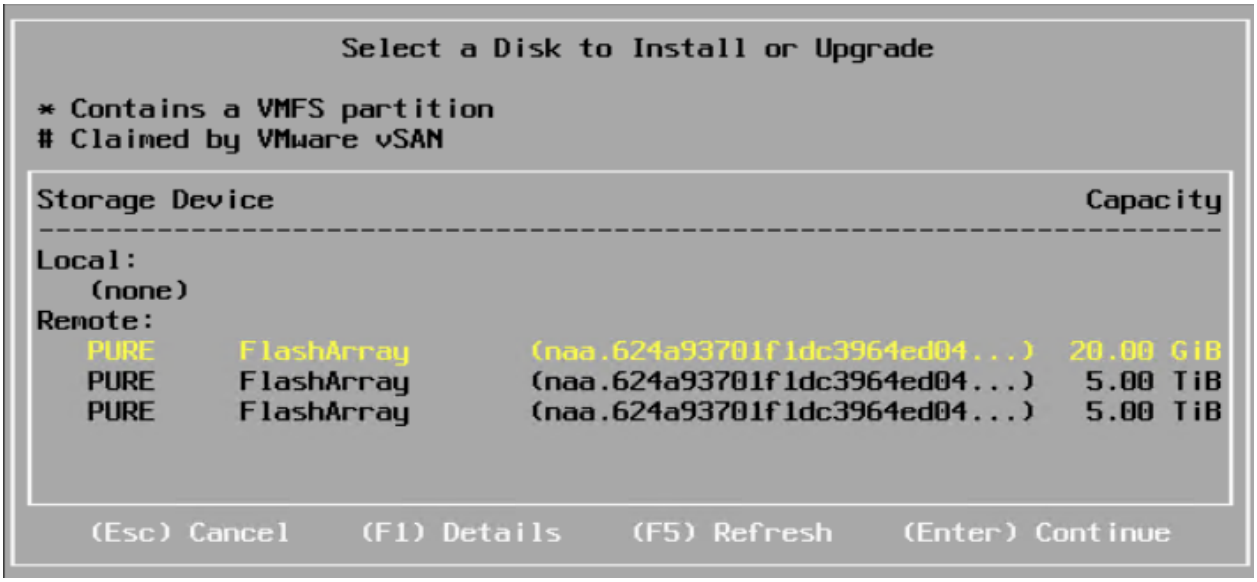
3. Right-click Server 1, Select KVM Console.



4. Click Activate Virtual Devices, mount ESXi ISO image.

5. Follow the prompts to complete installing VMware vSphere ESXi hypervisor.

6. When selecting storage device to install ESXi, select Remote LUN provisioned via Pure Storage Administrative console and access through FC connection.



Set Up Management Networking for ESXi Hosts

Adding a management network for each VMware host is necessary for managing the host and connection to vCenter Server. Please select IP address that can communicate with existing or new vCenter Server.

To configure the ESXi host with access to the management network, complete the following steps:

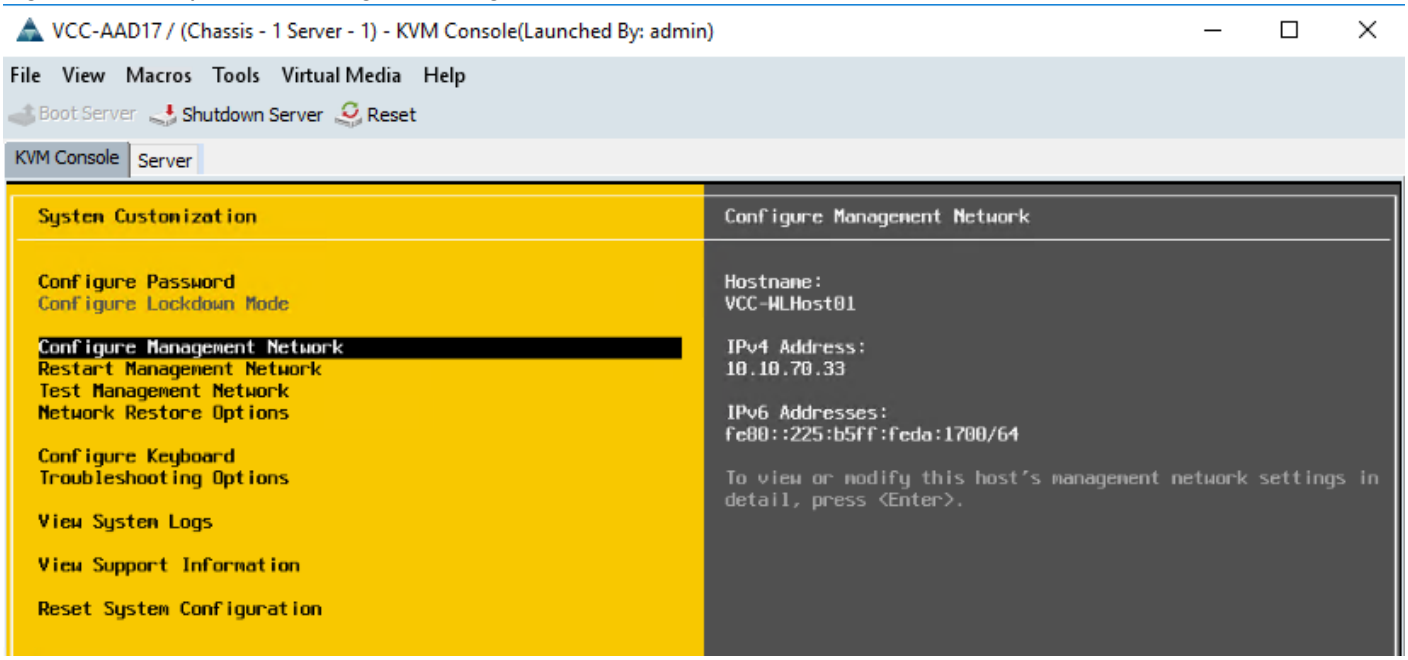
1. After the server has finished rebooting, press F2 to enter in to configuration wizard for ESXi Hypervisor.

2. Log in as root and enter the corresponding password.

3. Select the “Configure the Management Network” option and press Enter.
4. Select the VLAN (Optional) option and press Enter. Enter the VLAN In-Band management ID and press Enter.
5. From the Configure Management Network menu, select “IP Configuration” and press Enter.
6. Select “Set Static IP Address and Network Configuration” option by using the space bar. Enter the IP address to manage the first ESXi host. Enter the subnet mask for the first ESXi host. Enter the default gateway for the first ESXi host. Press Enter to accept the changes to the IP configuration.
7. IPv6 Configuration was set to automatic.
8. Select the DNS Configuration option and press Enter.
9. Enter the IP address of the primary and secondary DNS server. Enter Hostname
10. Enter DNS Suffixes.
11. Since the IP address is assigned manually, the DNS information must also be entered manually.

The steps provided varies based on the configuration. Please make thenecessary changes according to your configuration.

Figure 43 Sample ESXi Configure Management Network



Update Cisco VIC Drivers for ESXi

When ESXi is installed from Cisco Custom ISO you might have to update Cisco VIC drivers for VMware ESXi Hypervisor to match current Cisco Hardware and Software Interoperability Matrix.

Figure 44 Recommended Cisco UCS Hardware and Software Interoperability Matrix for vSphere 6.5 U1 and Cisco UCS B200 M5 on Cisco UCS Manager v3.2.2

UCS 1340 Virtual Interface Card(Cisco)	Firmware Version	4.2(2)
	Driver Version	1.0.13.0 NENIC
	Adapter BIOS	<none>
	Notes	11, 12, 20, 21, 31, 44, 66, 67
UCS 1340 Virtual Interface Card(Cisco)	Firmware Version	4.2(2)
	Driver Version	1.6.0.36 Fibre Channel
	Adapter BIOS	<none>
	Notes	11, 12, 20, 21, 31, 44, 66, 67

1. Log in to your VMware Account to download required drivers for FNIC and NENIC as per the recommendation.

   VMware ESXi 6.5 NIC nenic 1.0.13.0 Driver for Cisco nenic

   VMware ESXi 6.0 Fnic 1.6.0.36 FC Driver for Cisco fnic

2. Enable SSH on ESXi to run following commands:

```
esxcli software vib install -v /vmfs/volumes/datastore1/scsi-fnic_1.6.0.36-
1OEM.600.0.0.2494585.vib --no-sig-check
```

```
esxcli software vib install -v /vmfs/volumes/datastore1/nenic-1.0.13.0-
1OEM.650.0.0.4598673.x86_64.vib --no-sig-check
```

```
[root@VCC-SP-Infra01:~] esxcli software vib install -v /vmfs/volumes/datastore1/scsi-fnic_1.6.0.36-1O
EM.600.0.0.2494585.vib --no-sig-check
Installation Result
   Message: The update completed successfully, but the system needs to be rebooted for the changes to
 be effective.
   Reboot Required: true
   VIBs Installed: CSCO_bootbank_scsi-fnic_1.6.0.36-1OEM.600.0.0.2494585
   VIBs Removed: CSCO_bootbank_scsi-fnic_1.6.0.34-1OEM.600.0.0.2494585
   VIBs Skipped:
[root@VCC-SP-Infra01:~] esxcli software vib install -v /vmfs/volumes/datastore1/nenic-1.0.13.0-1OEM.6
50.0.0.4598673.x86_64.vib --no-sig-check
Installation Result
   Message: The update completed successfully, but the system needs to be rebooted for the changes to
 be effective.
   Reboot Required: true
   VIBs Installed: Cisco_bootbank_nenic_1.0.13.0-1OEM.650.0.0.4598673
   VIBs Removed: Cisco_bootbank_nenic_1.0.6.0-1OEM.650.0.0.4598673
   VIBs Skipped:
[root@VCC-SP-Infra01:~]
```

VMware Clusters

The following VMware Clusters were configured in two vCenter to support the solution and testing environment:

VCSA01

VDI Cluster: Pure Storage FlashArray //X70 with Cisco UCS

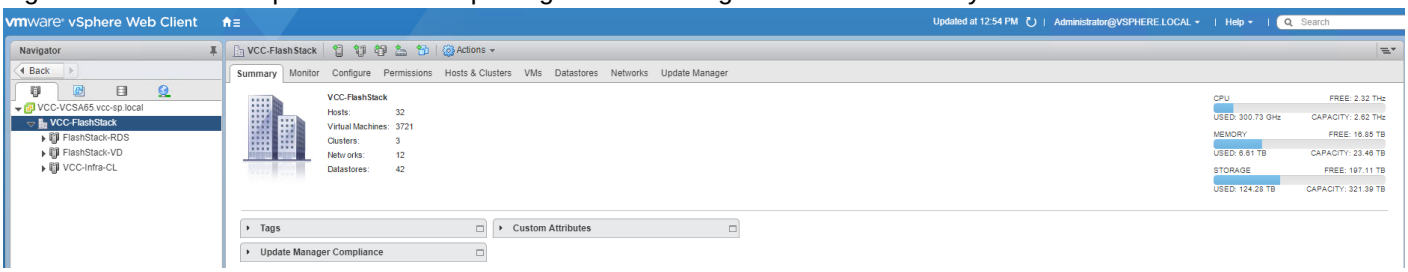
- Infrastructure Cluster: Infrastructure Virtual Machines (vCenter Server Appliance(2), Active Directory (2), DNS, DHCP, VMware Horizon Connection Servers(4), VMware Horizon Composer Server(2), Microsoft SQL Server(2), VMware Horizon User Environment Manager(2).
- RDSH: VMware Horizon RDSH (Remote Desktop Server Hosted) VMs (Windows Server 2016 RDS Roles) provisioned with VMware View Composer.
- VDI Non-Persistent: VMware Horizon VDI VMs (Windows 10 64-bit non-persistent Instant Clones virtual desktops provisioned.
- VDI Persistent: VMware Horizon VDI VMs (Windows 10 64-bit persistent virtual desktops provisioned with VMware Horizon Composer.

VCSA02

VSI Launchers Cluster

- Launcher Cluster: Login VSI Cluster (The Login VSI launcher infrastructure was connected using the same set of switches but hosted on separate SAN storage and servers)

Figure 45 VMware vSphere WebUI Reporting Cluster Configuration for this Study



Building the Virtual Machines and Environment for Workload Testing

Software Infrastructure Configuration

This section details how to configure the software infrastructure components that comprise this solution.

Install and configure the infrastructure virtual machines by following the process provided in Table 15 :

Table 15 Software Configuration for Infrastructure Virtual Machines

Configuration	Operating System	Virtual CPU	Memory	Disk Size	Network
vCenter Server Appliance (2)	VCSA- SUSE Linux	16	32	980	MGMT-VLAN
Active Directory Domain Controllers/DHCP/DNS (2)	Microsoft Windows 2016	4	12	60	Infra-VLAN

Configuration	Operating System	Virtual CPU	Memory	Disk Size	Network
VMware Horizon Connection Server (4)	Microsoft Windows 2016	4	16	60	Infra-VLAN
VMware Horizon Composer Server (2)	Microsoft Windows 2016	4	12	60	Infra-VLAN
Microsoft SQL Server 2016 (2)	Microsoft Windows 2016	4	12	60	Infra-VLAN
KMS License Server	Microsoft Windows 2016	4	8	40	Infra-VLAN

Install and Configure VMware Horizon Environment

This section details the installation of the VMware core components of the Horizon Connection Server and Replica Servers. This CVD installs 1 VMware Horizon Connection server and 3 VMware Horizon Replica Servers to support both remote desktop server hosted sessions (RDSH), non-persistent virtual desktops (VDI) based on the best practices from VMware.

Please refer to the Horizon 7 Sizing Limits and Recommendations: https://kb.vmware.com/s/article/2150348

The prerequisites for installing the Horizon Connection server or Composer server on a Windows Server. Refer to the System Requirements for Horizon Server Components for the list of supported Operating Systems. In this study, we have used Windows Server 2016 for Horizon Connection Server, Replica Servers, and Composer Server.

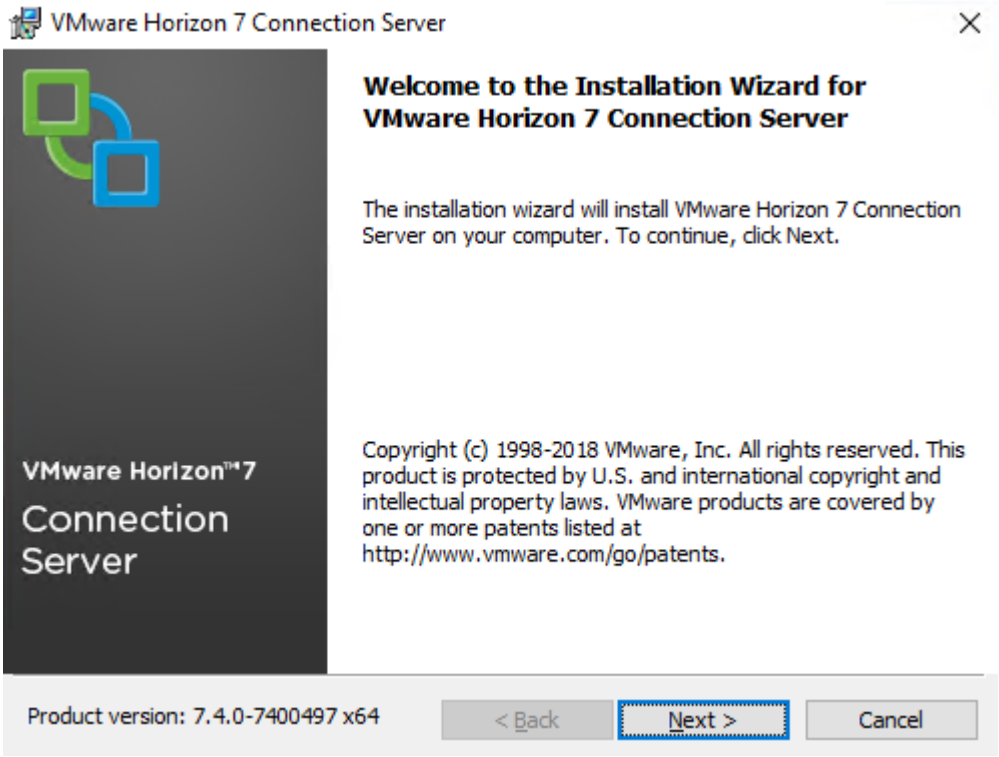
Download VMware Horizon components:

https://my.vmware.com/en/web/vmware/info/slug/desktop_end_user_computing/vmware_horizon/7_4

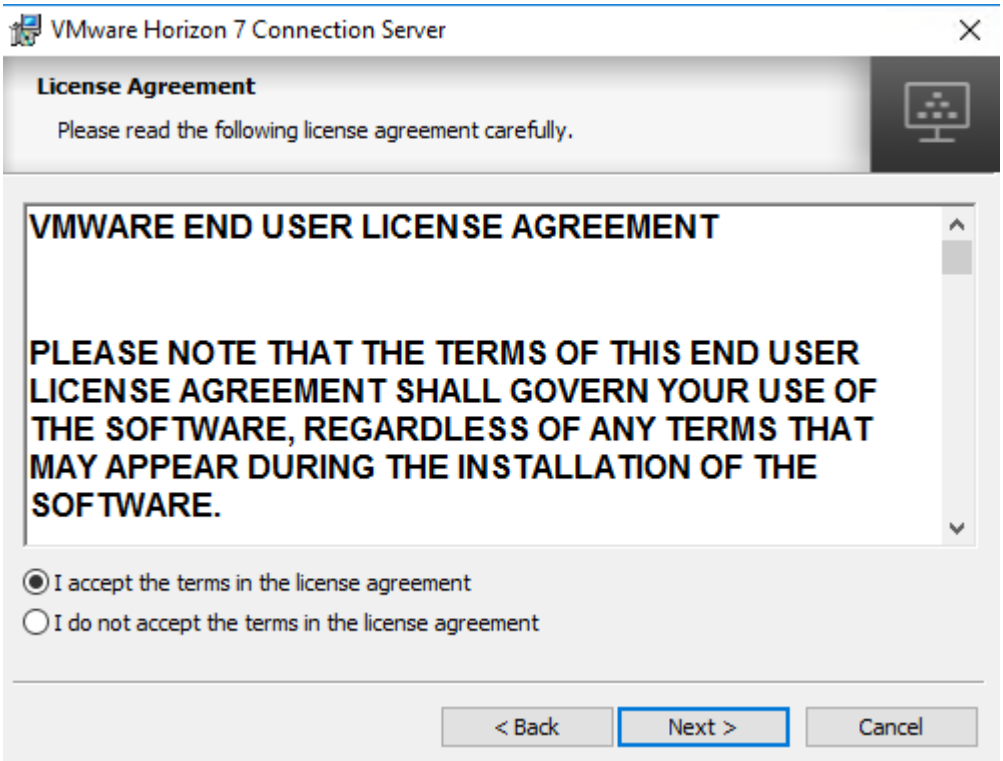
VMware Horizon Connection Server Configuration

To configure the VMware Horizon Connection Server, complete the following steps:

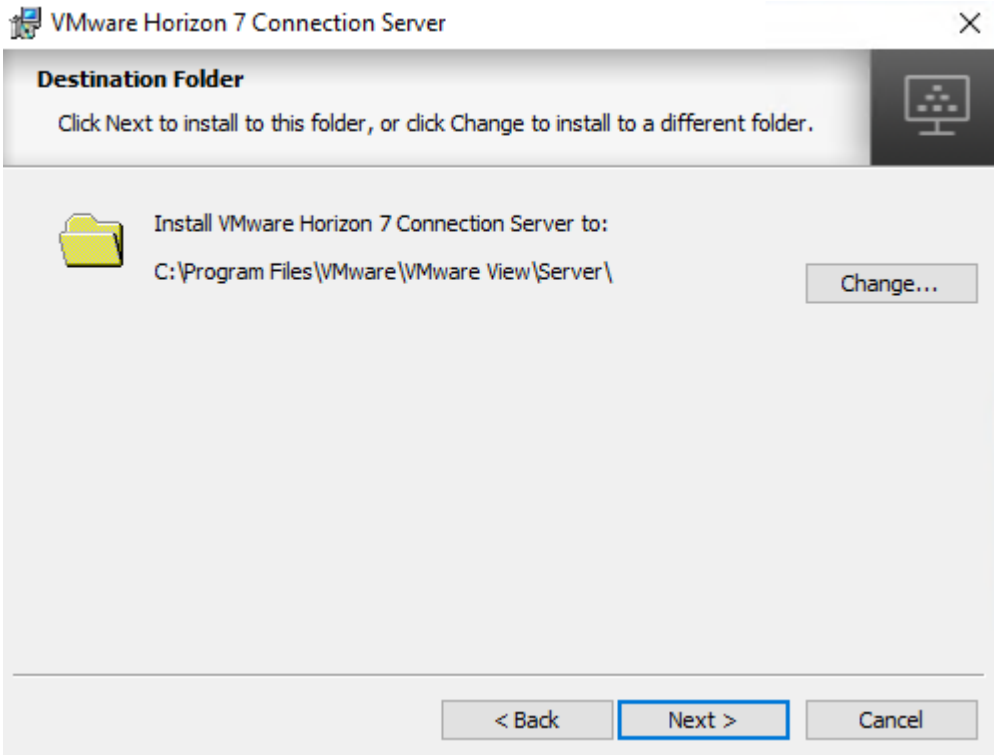
1. Run the Connection Server installer; we installed version 7.4.0-7400497. Click Next.



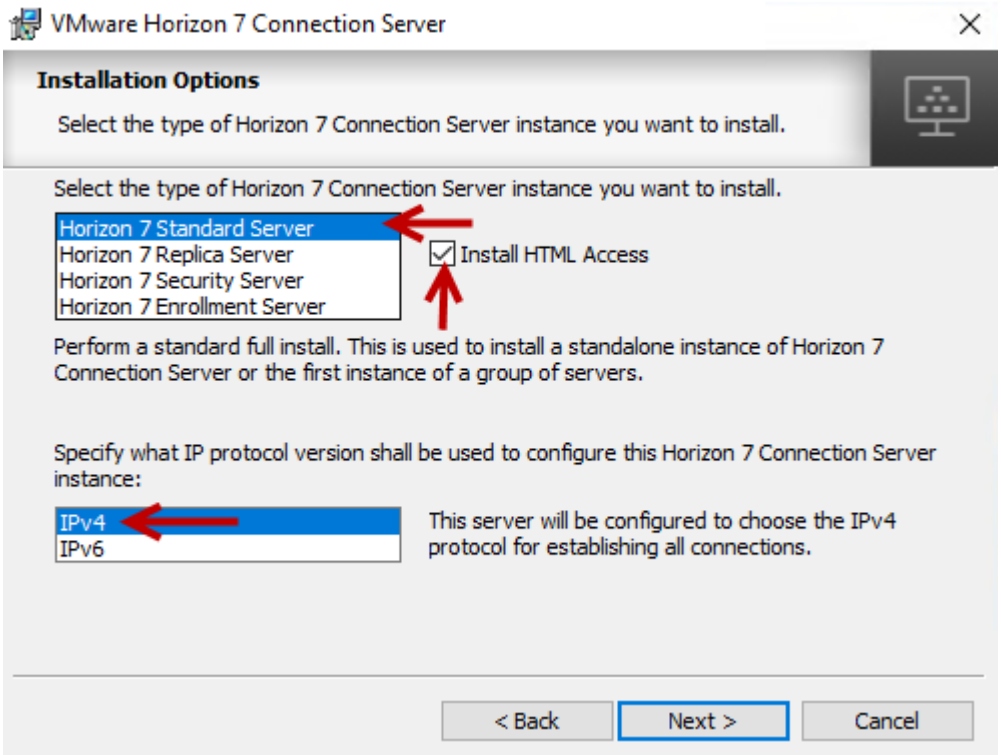
2. Accept the terms in the License Agreement. Click Next.



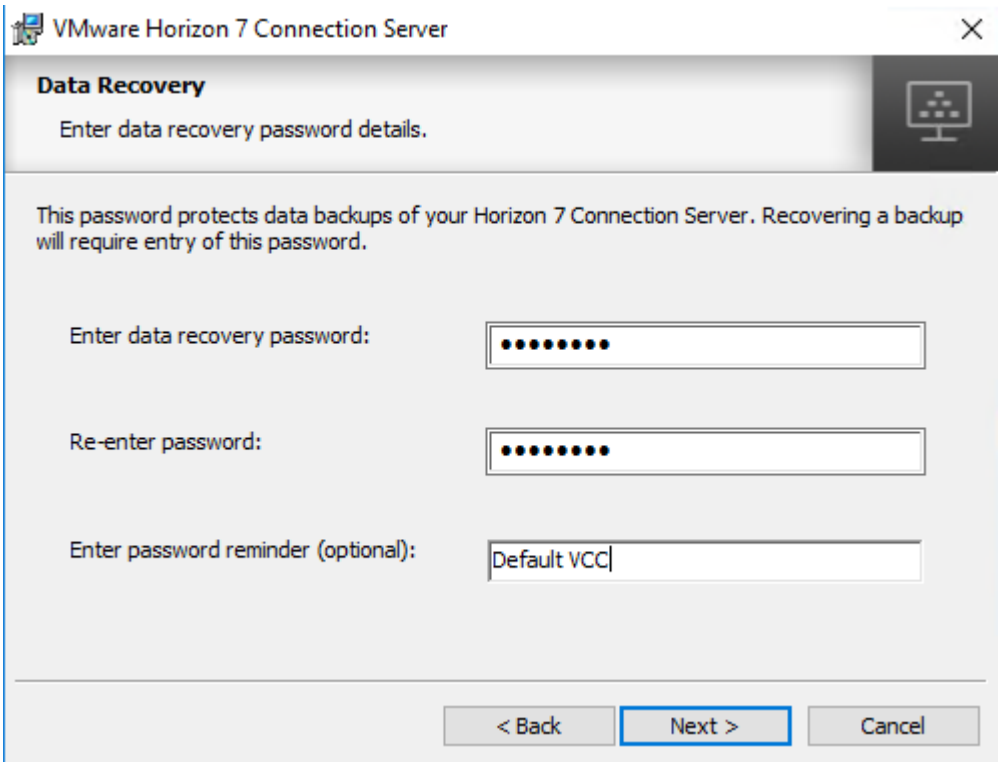
3. Click Next to select default folder or click Change to install to a different folder. We selected the default destination folder.



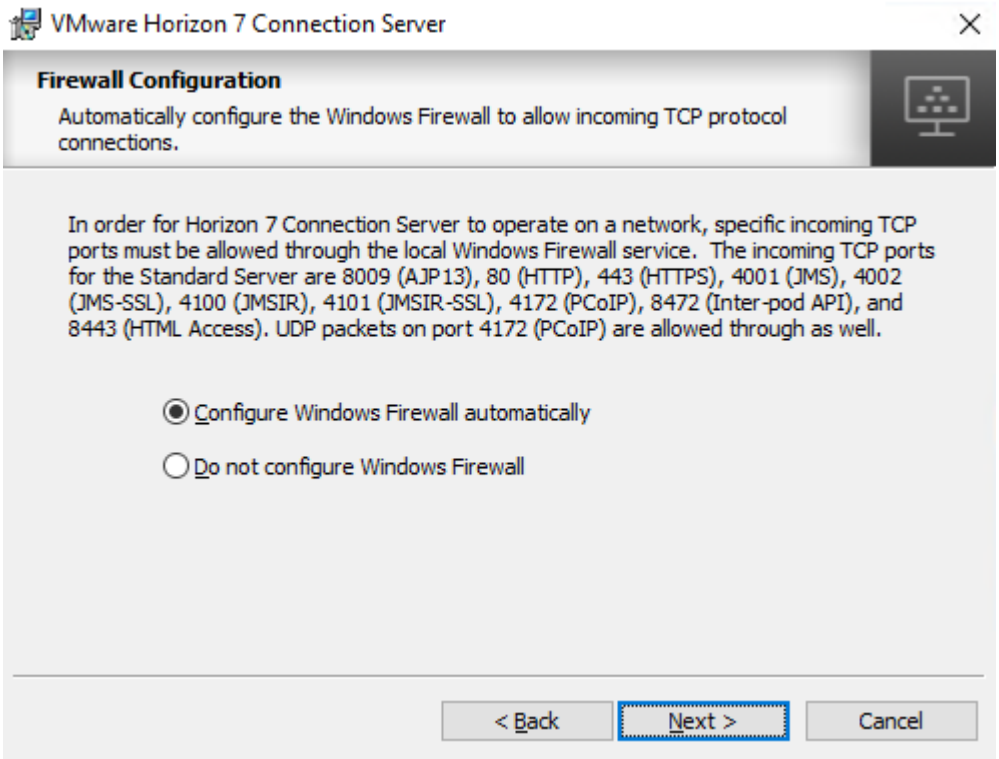
4. Select the Horiozn 7 Standard Server. Click Next.



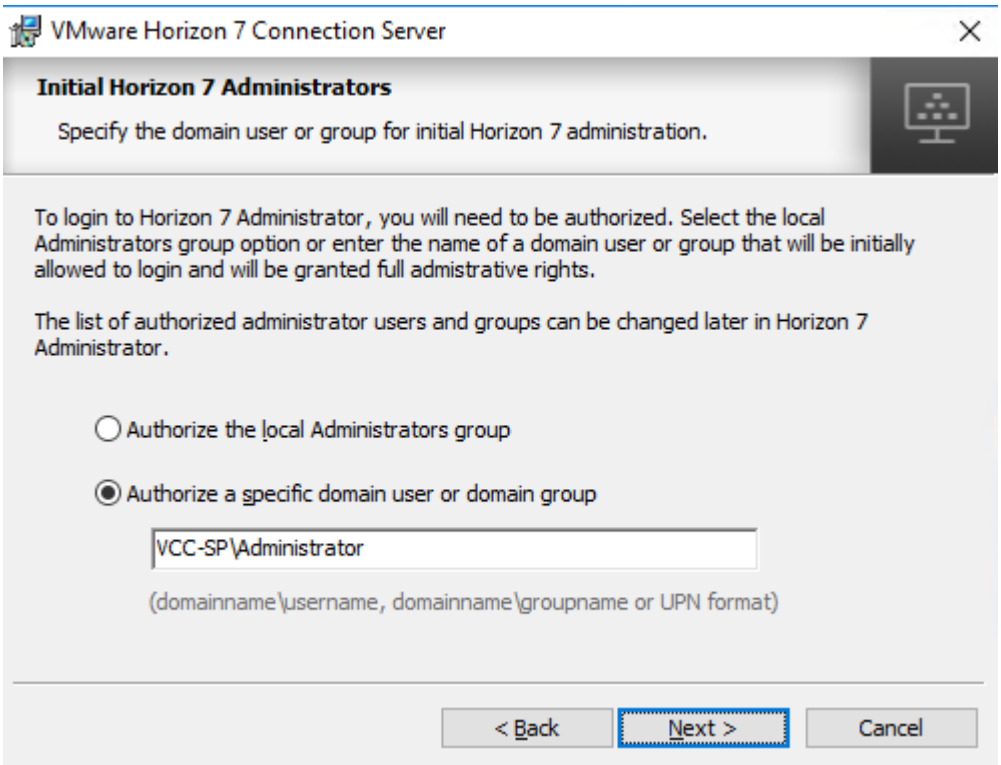
5. Enter Data Recovery Password, click Next.



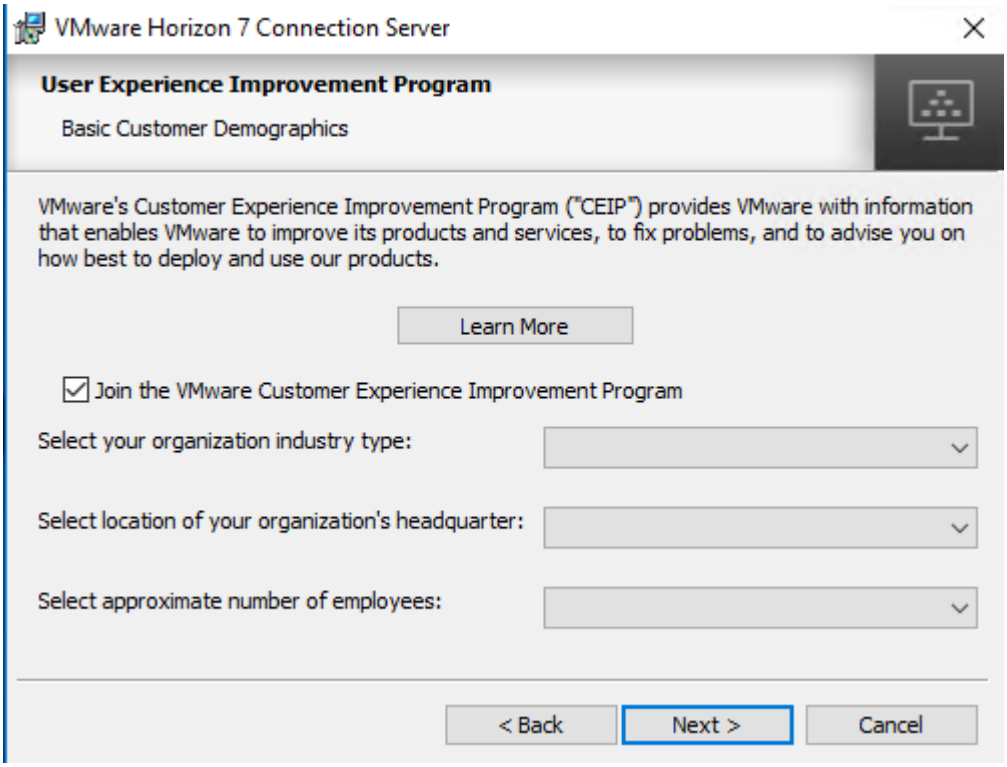
6. Configure the Windows firewall automatically. Click Next.



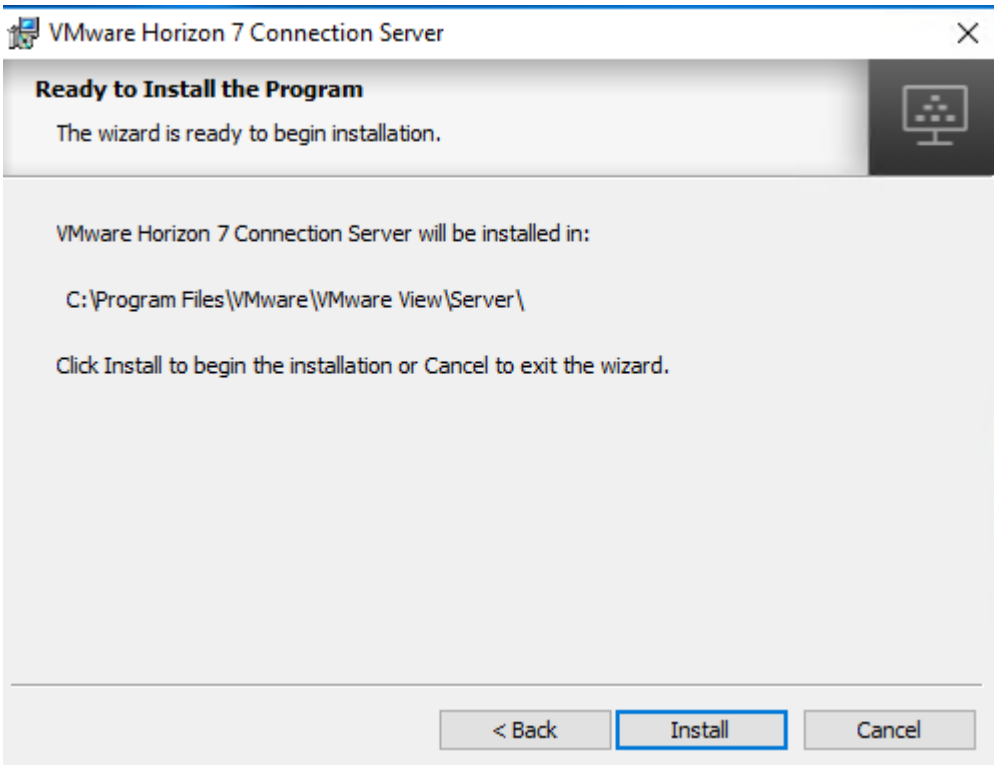
7. Specify domain user or group to Horizon 7 administration. Click Next.



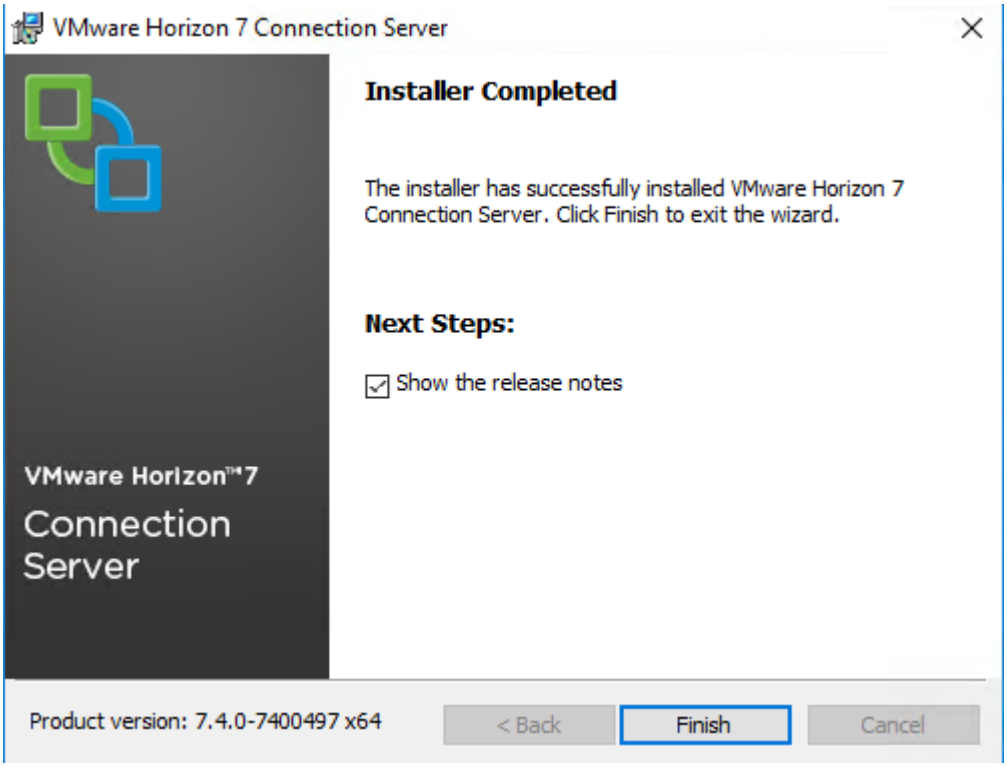
8. Click Next on Ucer Experience Improvement Program.



9. Click Install.



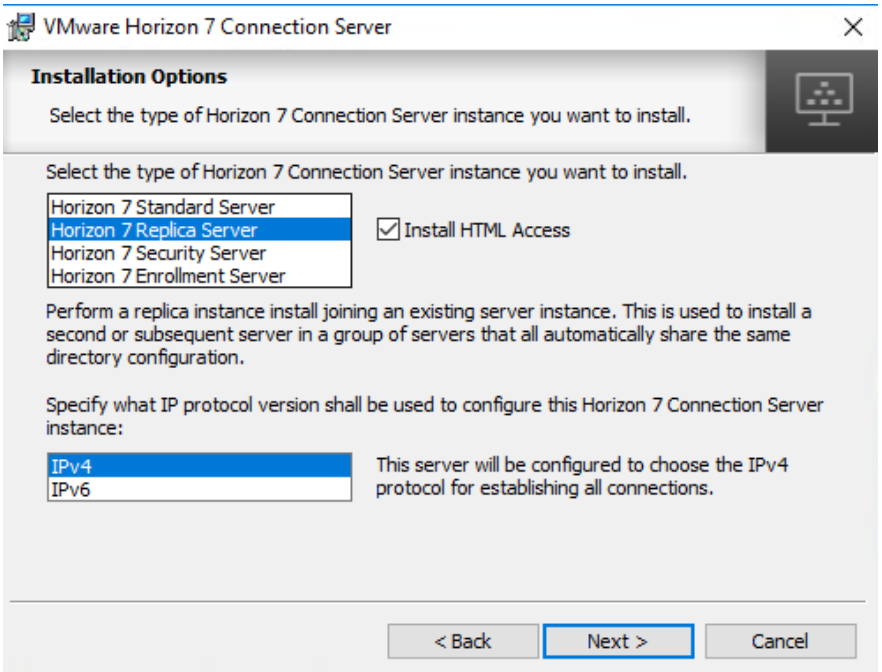
10. Click Finish to complete the Horizon Connection Server installation.



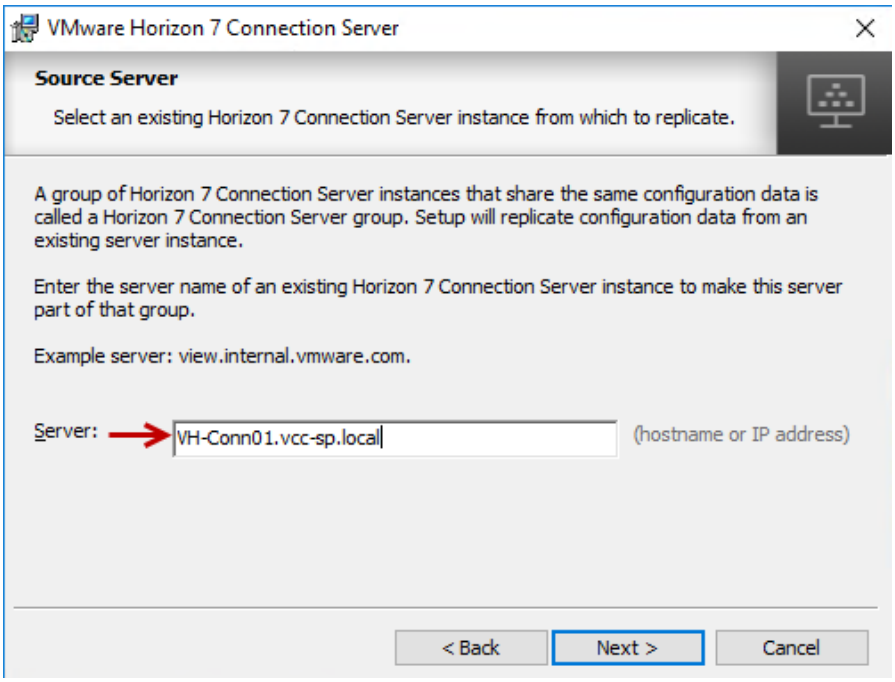
VMware Horizon Replica Server Configuration

To install Horizon Replica Server and additional Replica servers, complete the following steps:

1. Follow the steps provided above in the VMware Horizon Connection Server Configuration. During the Horizon Replica Server installation, select the Replica Server option in order to configure this server as a Replica Server (shown in the image below) and complete all the steps shown above.



2. Select Horizon Connection Server instance with Role as “Standard Server” to which all remaining Replica(s) server will be replicated to. Enter FQDN or IP Address.



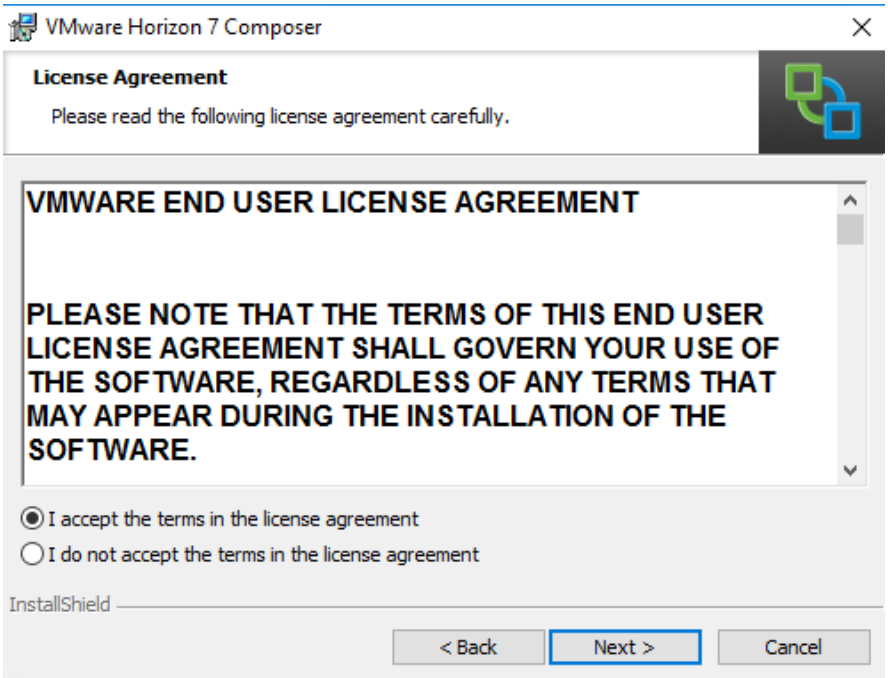
Install VMware Horizon Composer Server

To install the VMware Horizon Composer Server, complete the following steps:

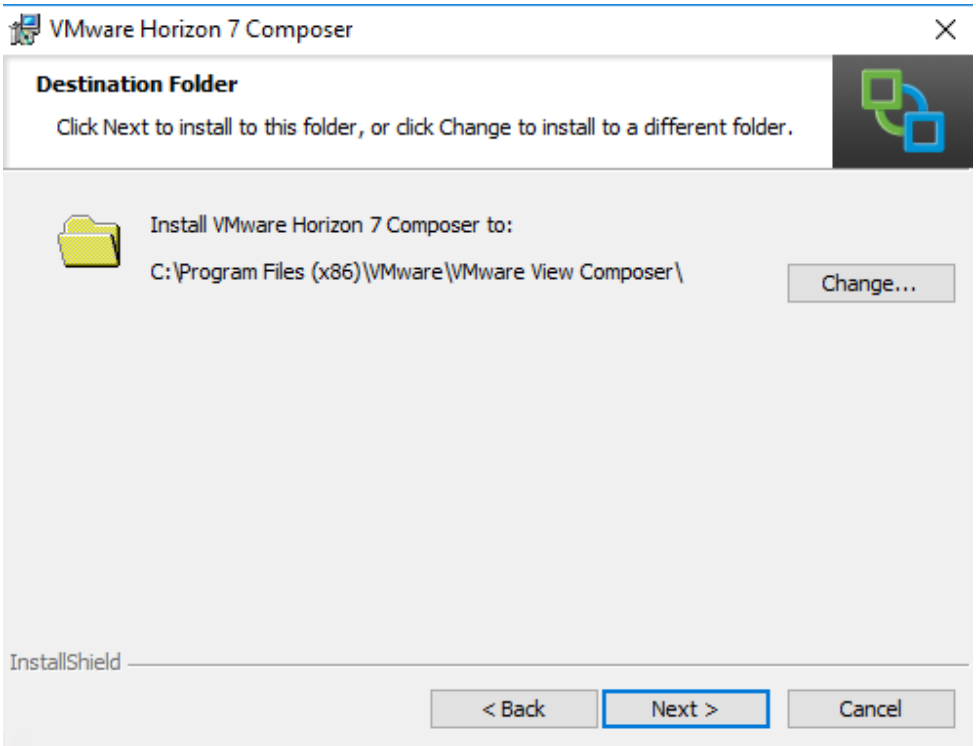
1. Run the Install Horizon Composer installer, VMware-viewcomposer-7.4.0-7312595.

2. Click Next.

3. Accept the End User License Agreement, click Next.

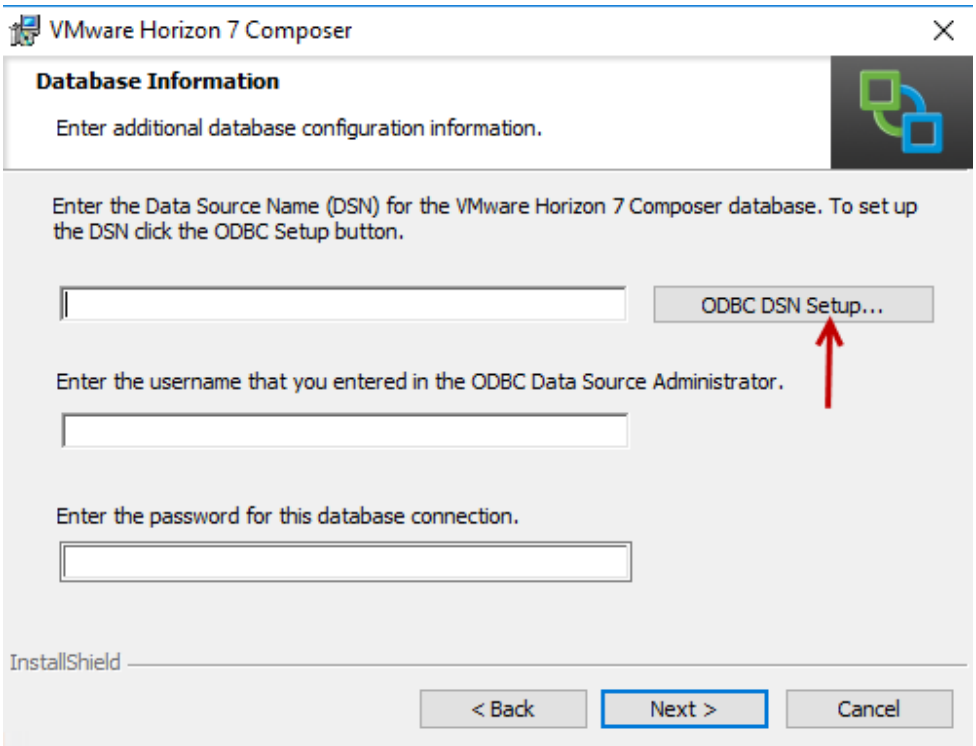


4. Click Next to select default folder or click Change to install to a different folder. We selected the default destination folder.

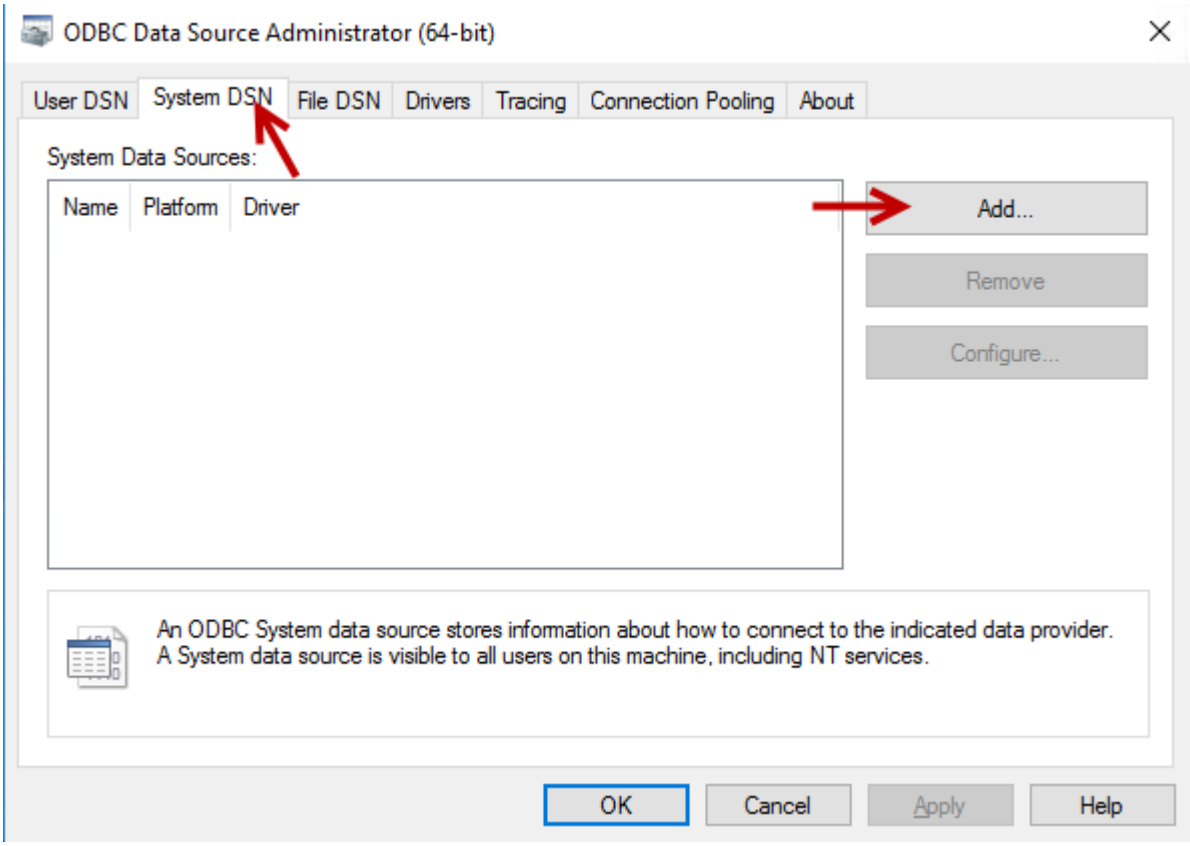


5. Enter database configuration for Horizon Composer.

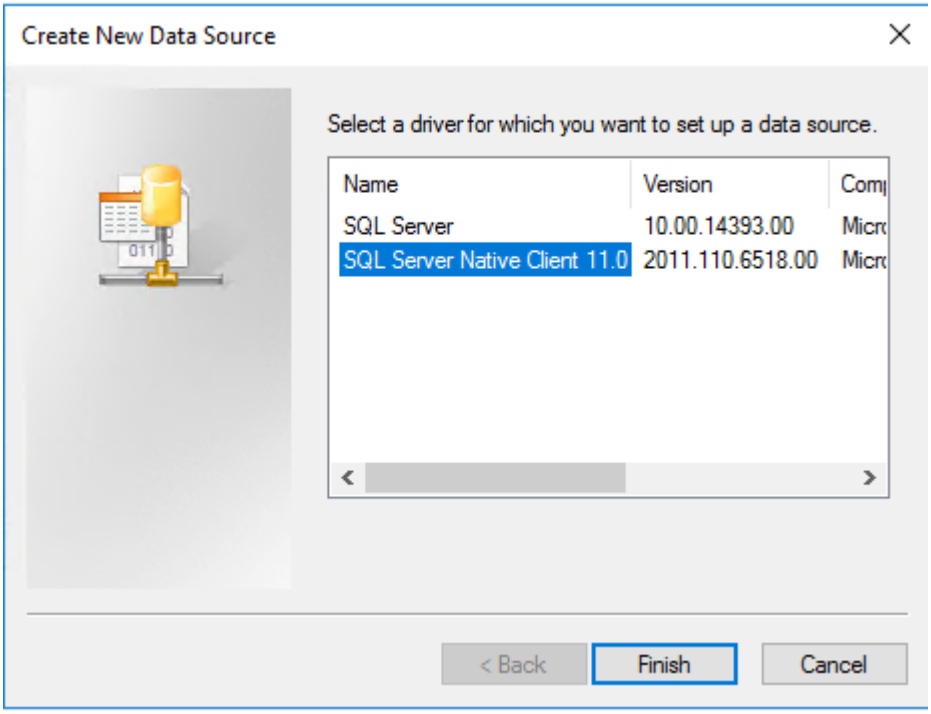
6. If not already configured, click “ODBC DNS Setup.”



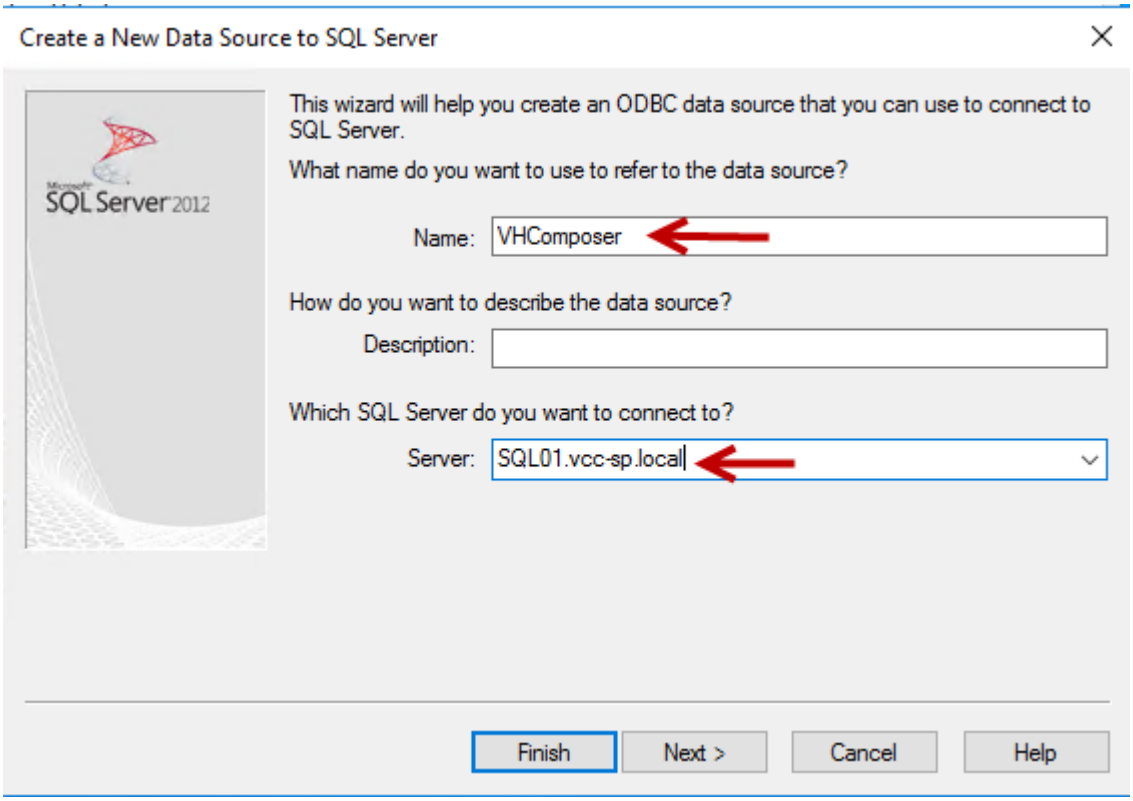
7. Click Systems DSN, then click Add.



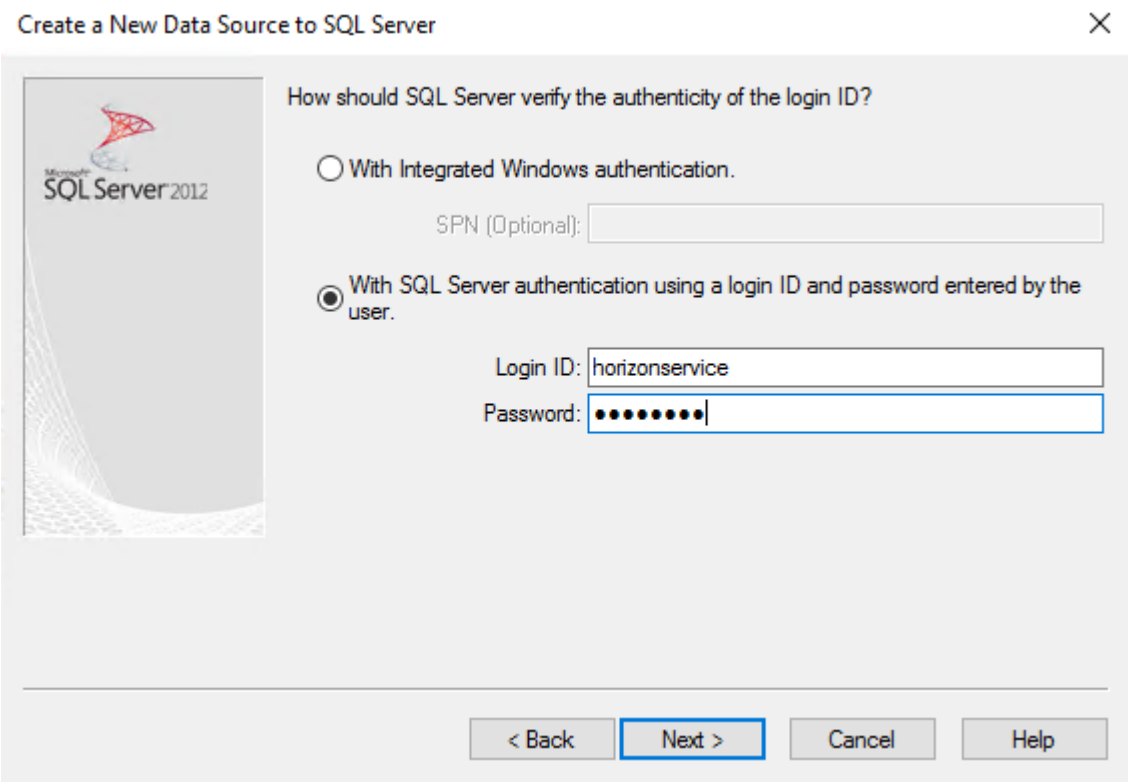
8. Select SQL Server Native Client. We installed “sqlncli x64” to connect with SQL Server 2016 SP1 database.



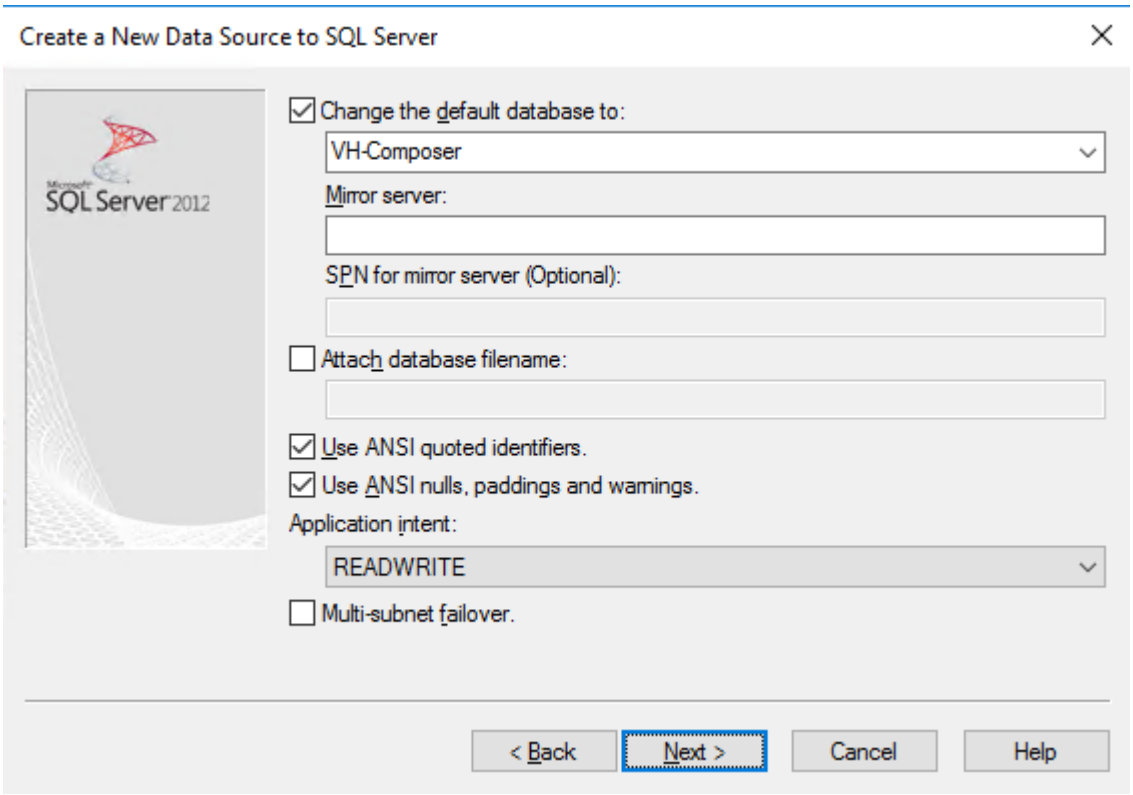
9. Enter preferred name for data source and SQL Server to connect to. Click Next.



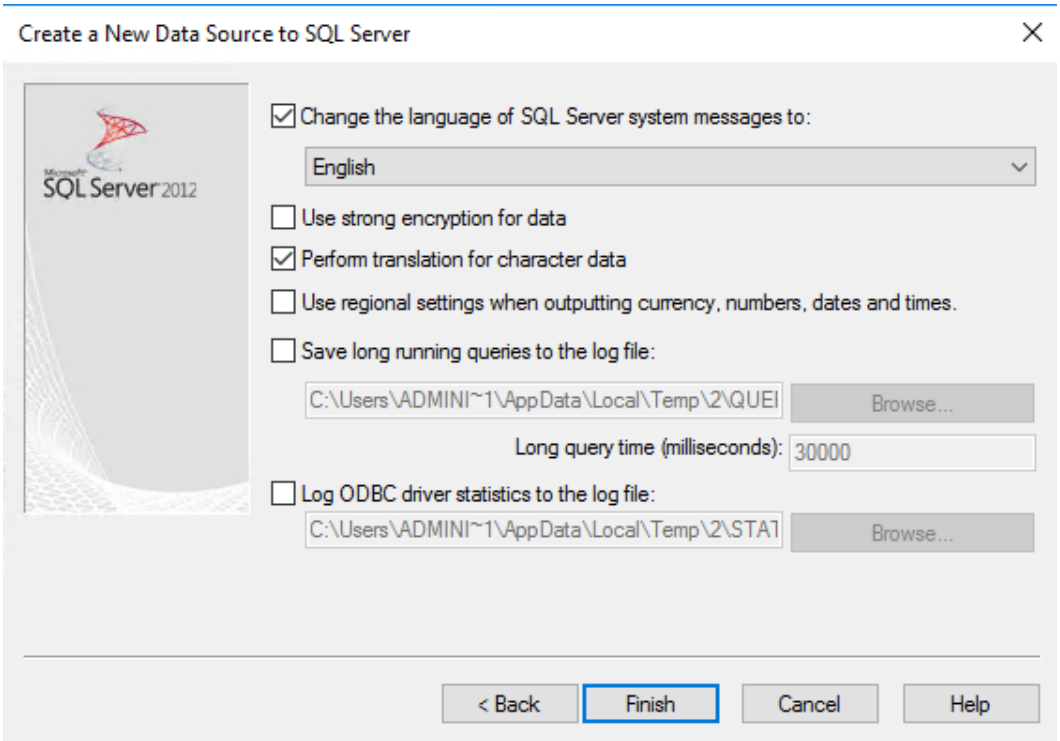
10. Select either Windows Integrated or SQL Server Authentication Login ID for verification.



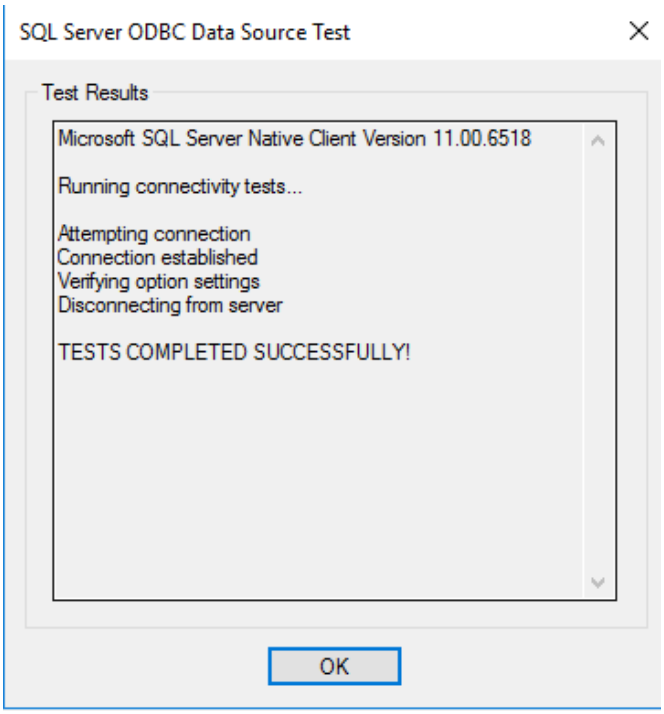
11. Change the default database to database created for VMware Horizon Composer; In our case we created “VH-Composer.” Click Next.



12. Select default Langauge, click Finish.

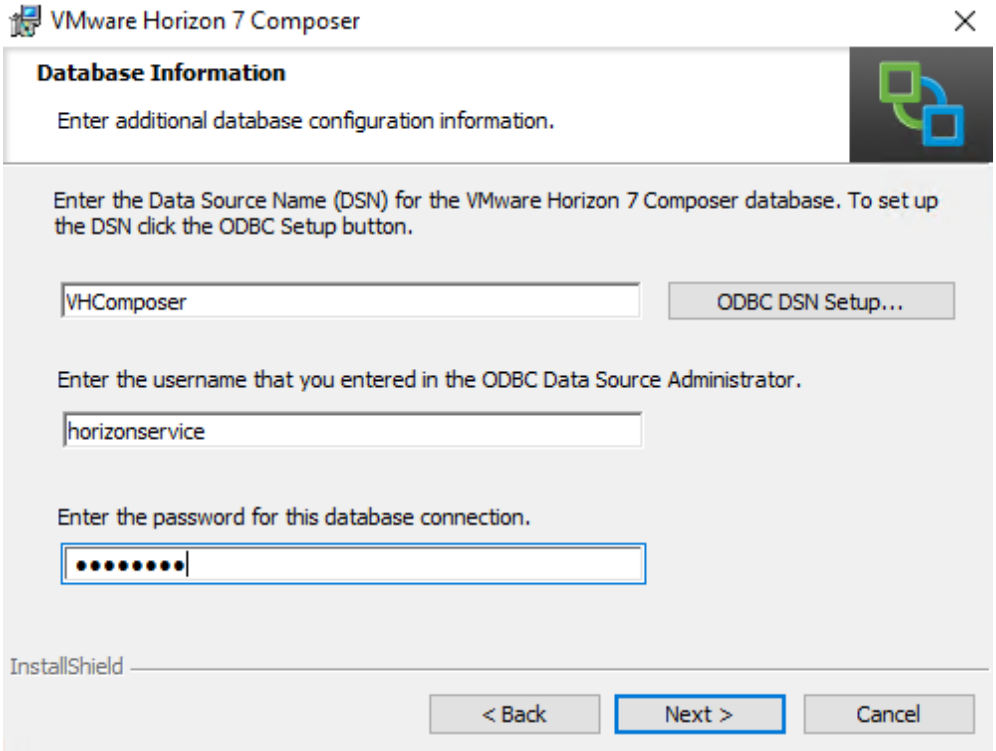


13. Click “Test Data Source” and verify that test completed successfully. Click OK with the successful completion of the test.

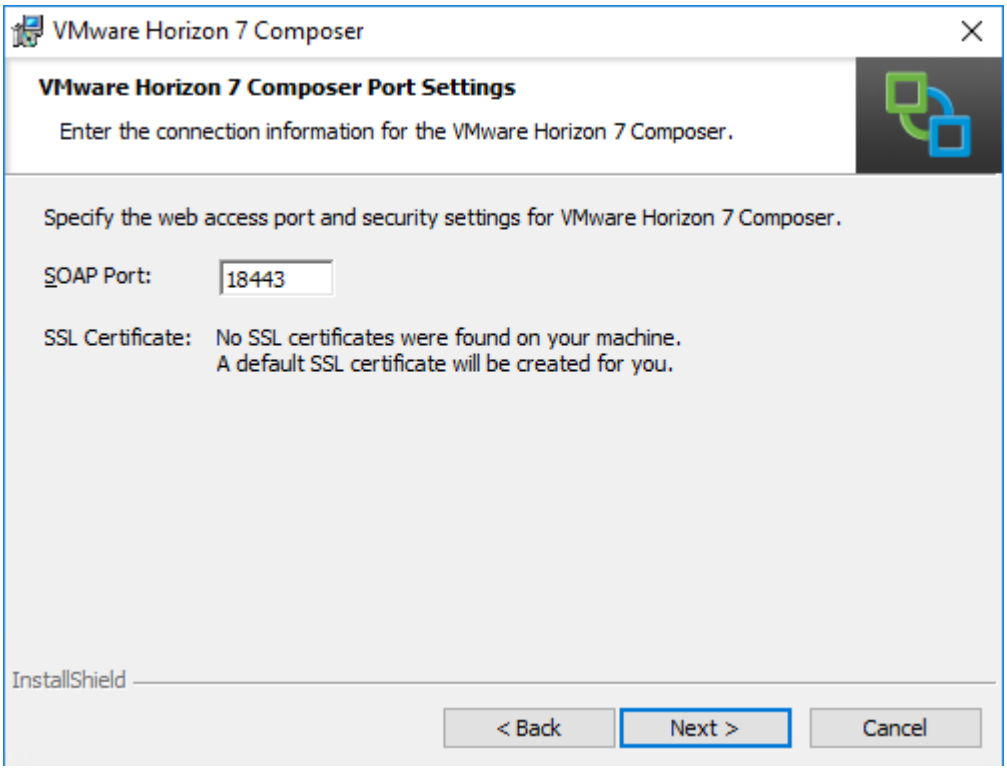


14. Click OK.

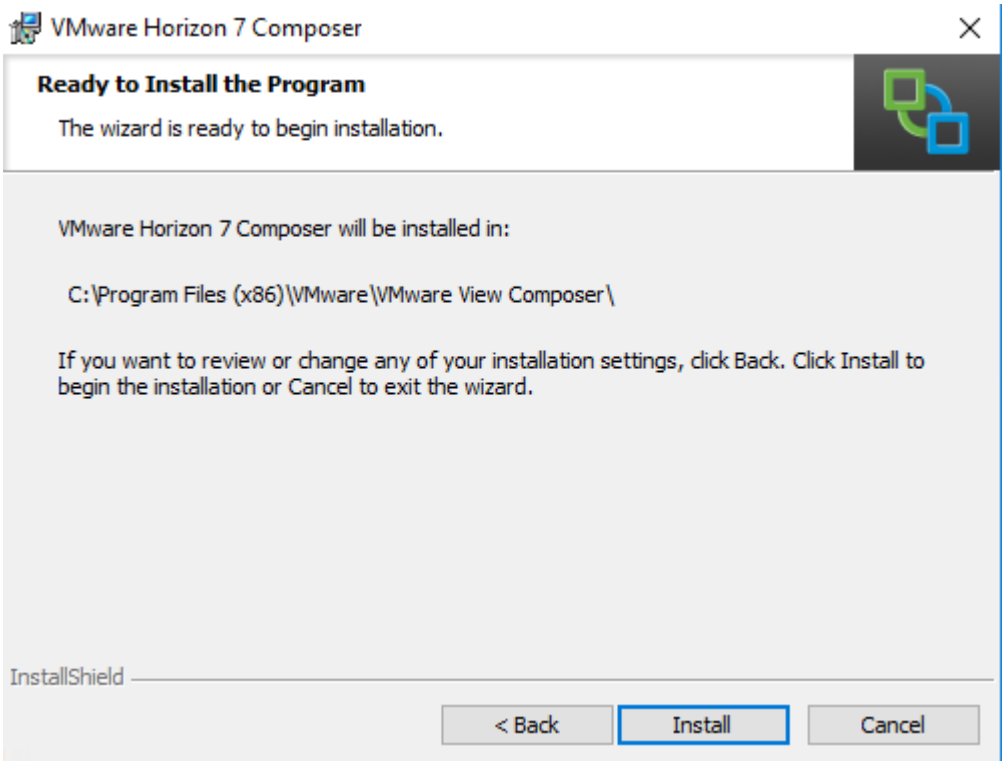
15. Enter Data Source name and login credential then click Next.



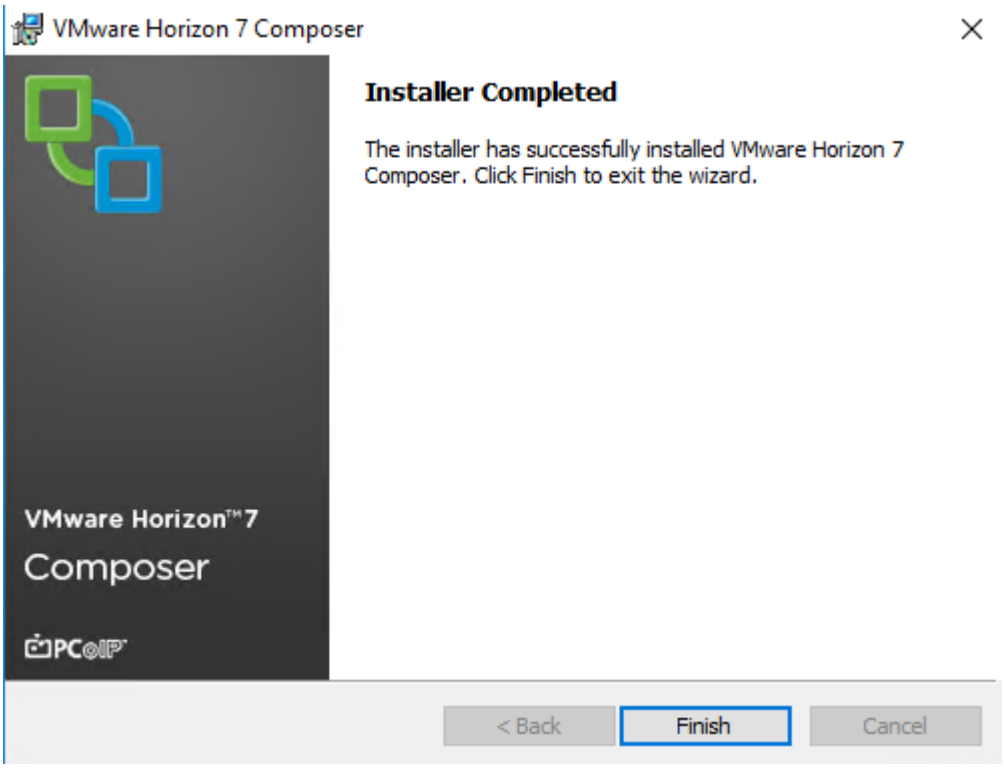
16. Leave default or specify web access port and security settings for Horizon 7 Composer. Click Next.



17. Click Install.



18. Click Finish.



19. Click Yes to restart VMware Horizon 7 Composer.

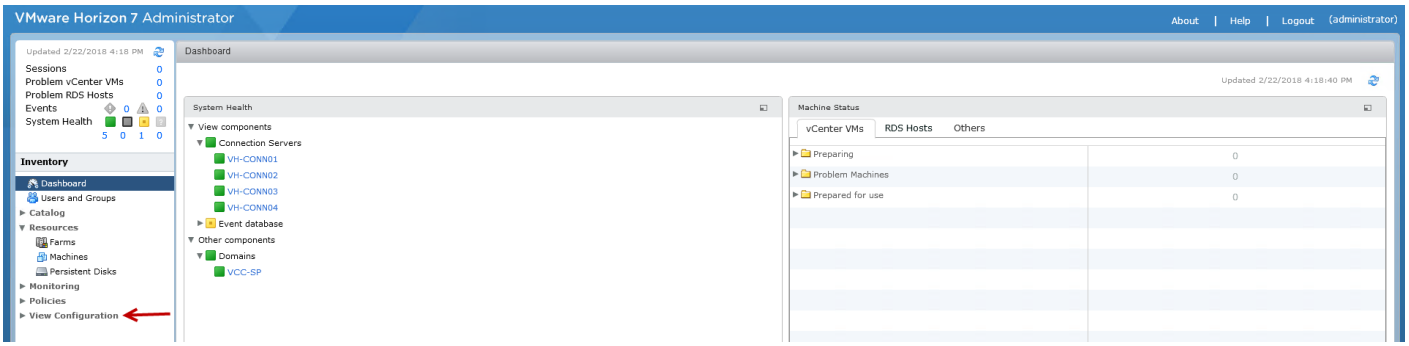
Configure the Horizon 7 Environment

To configure the VMware Horizon Administrator, complete the following steps:

1. Log into the VMware Horizon 7 Administrator Console through a Web browser and enter your credentials.

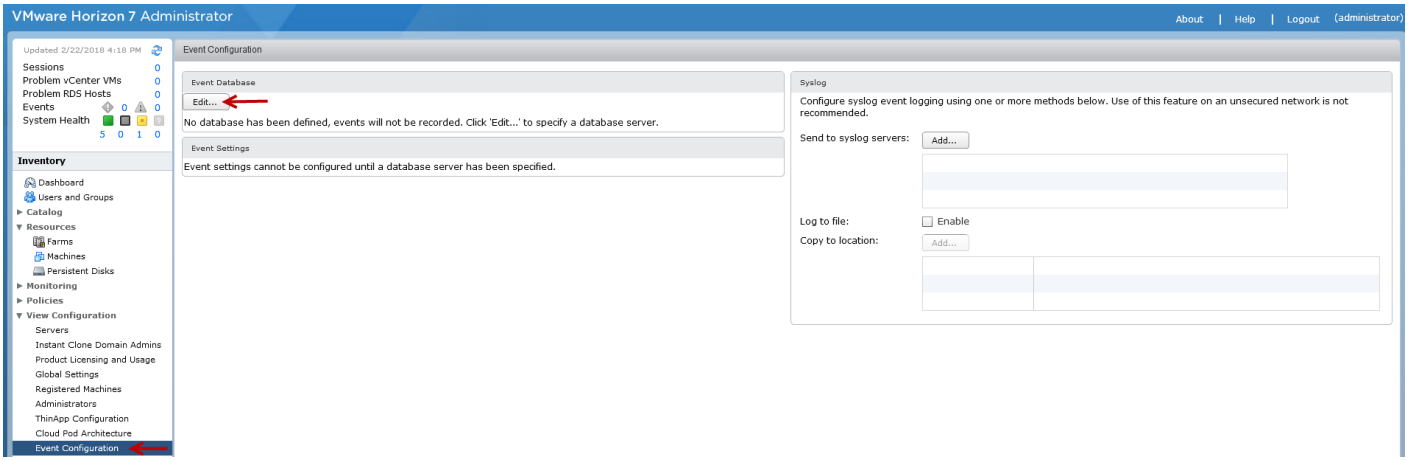
Allow the Flash content on the web browser to access the Horizon 7 Administrator console.

2. When logged in to Horizon Administrator console; it reports Connection and Replica Server(s) part of the Horizon configuration, domain. Click “View Configuration” to complete initial setup prior we can start creation of Persistent/Non-Persistent Desktop Pool or RDSH Farm.

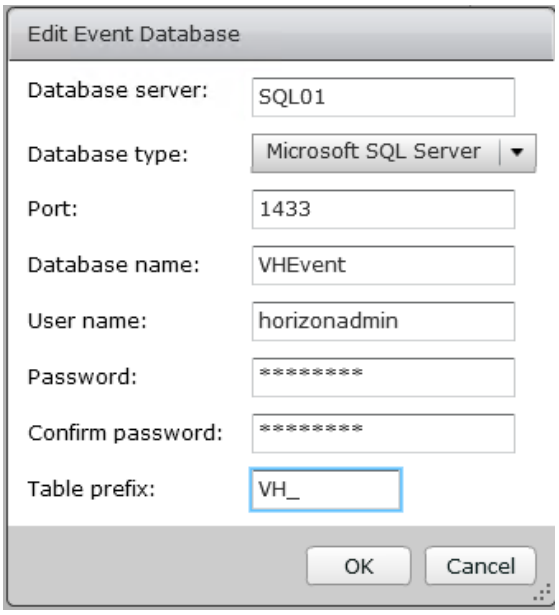


Configure Event Database

1. Click “Event Configuration” to log Events. Click Edit.

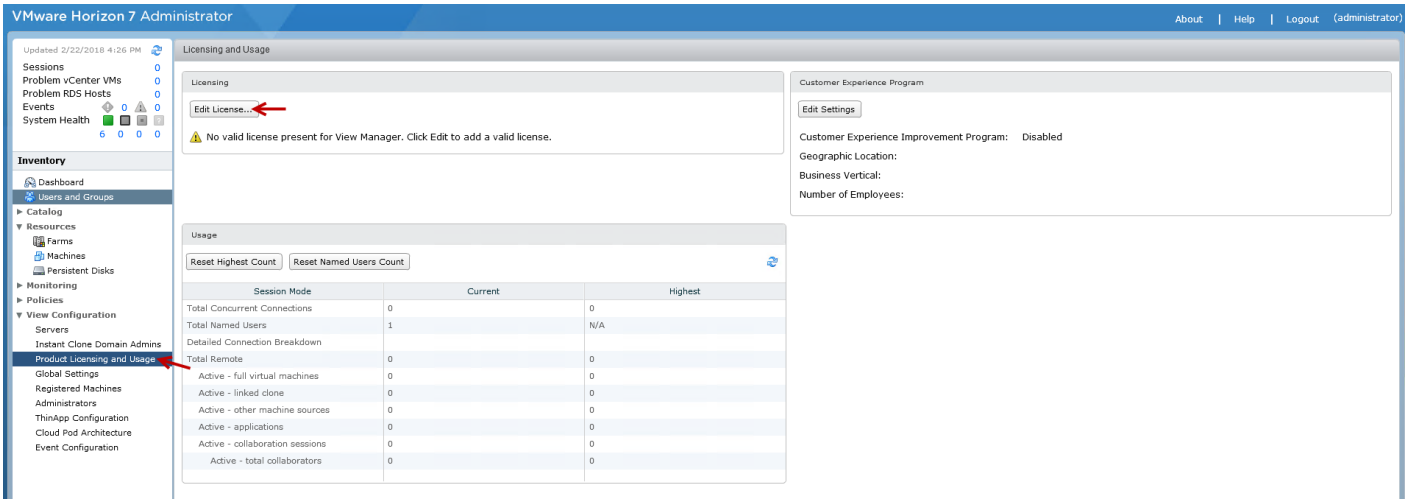


2. Enter information for SQL Server, database instance to connect, User name and Password, and Table prefix.

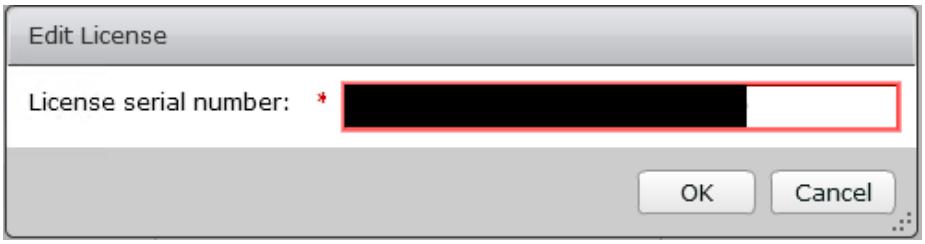


Configure Horizon 7 Licenses

1. Click “Product Licensing and Usage” and then click “Edit License.”

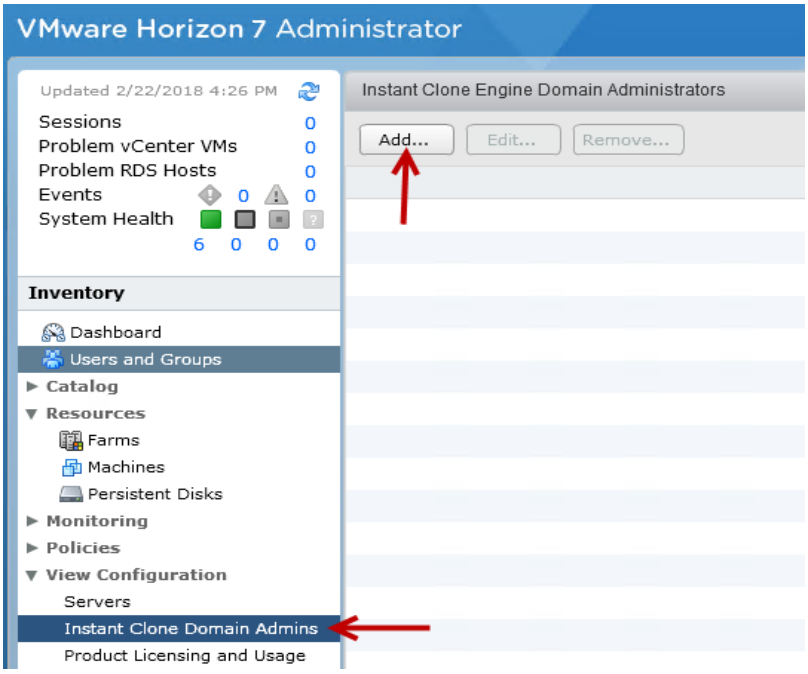


2. Enter License Serial Number. Click OK.

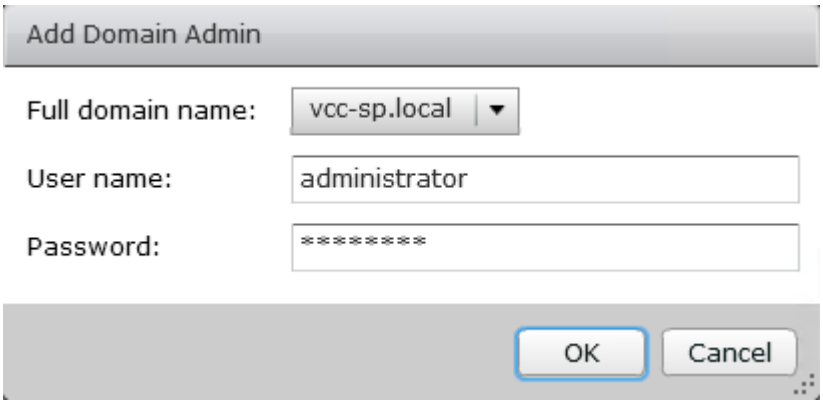


Configure Instant Clone Domain Admins

1. In the Horizon Administrator console > View Configuration > click “Instant Clone Domain Admins”, then click Add.



2. Enter User name and Password for domain. Click OK.

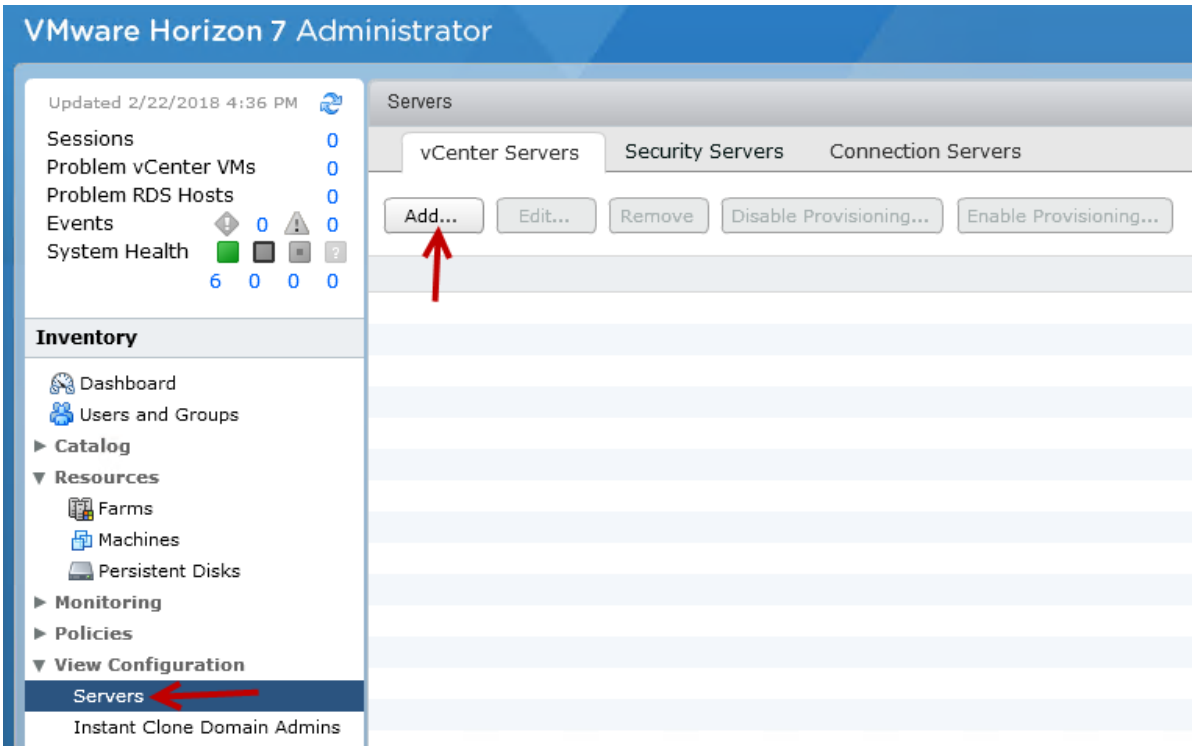


3. Click “Servers” then click Add.

Configure vCenter

In the Horizon Administrator console, complete the following steps:

1. In View Configuration, select Servers. Click the Add vCenter Server tab.



2. On vCenter Server Information page enter the following:

   a. Enter FQDN or IP address for vCenter Server

   b. User name

   c. Password

   d. Description (Optional)

3. Click Next.

vCenter Server Settings

Server address:	vcc-vcsa65.vcc-sp.local
User name:	administrator@vsphere.local
Password:	********
Description:	FlashStack VCC deployment.
Port:	443

Advanced Settings

Specify the concurrent operation limits.

Max concurrent vCenter provisioning operations:	50
Max concurrent power operations:	50
Max concurrent View Composer maintenance operations:	50
Max concurrent View Composer provisioning operations:	50
Max concurrent Instant Clone Engine provisioning operations:	50

4. Click View Certificate.

Invalid Certificate Detected

2/22/2018 4:46 PM

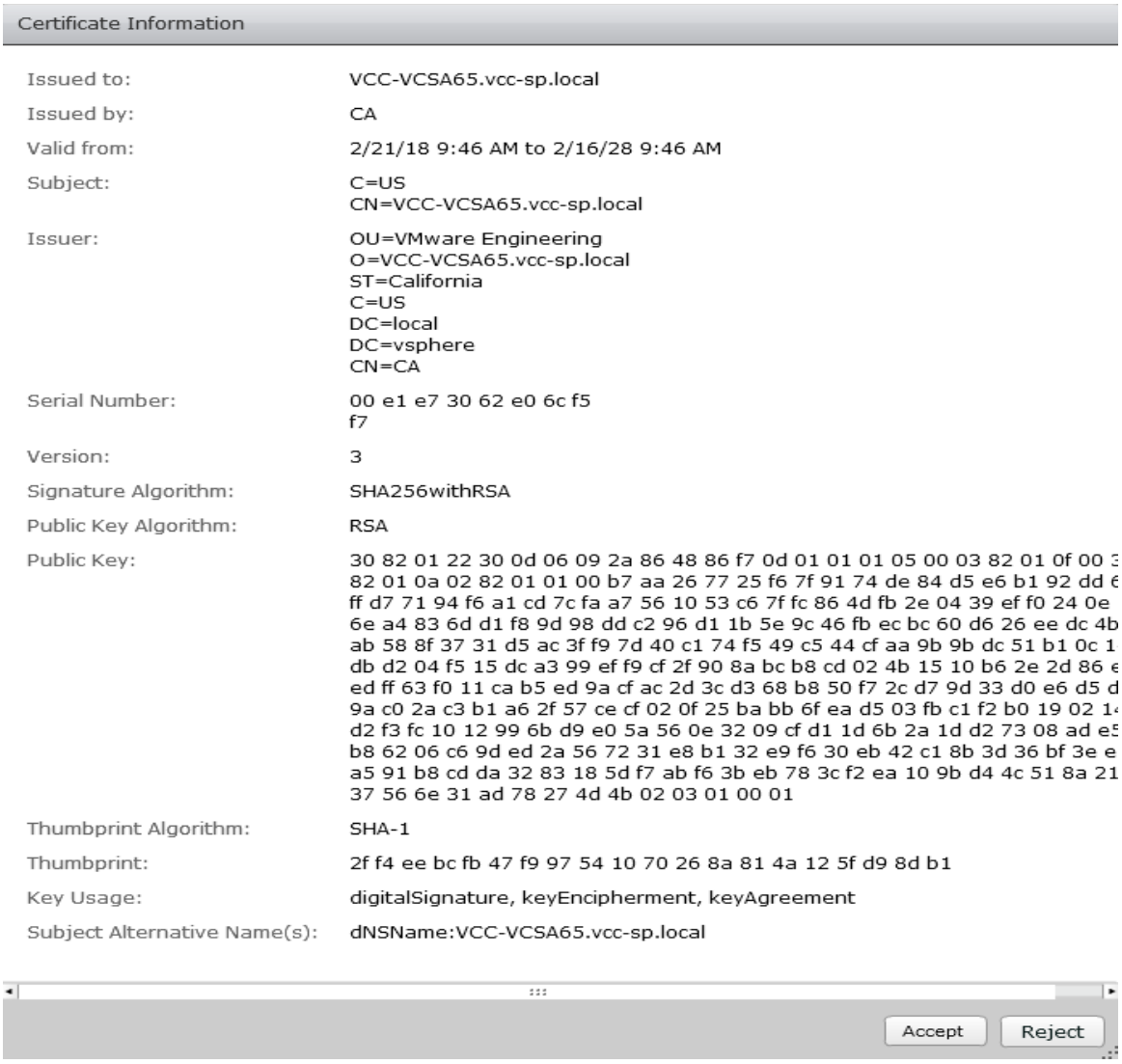
The identity of the specified vCenter Server cannot be verified for the following reasons:

- Server's certificate is not trusted.
- Server's certificate cannot be checked.

VMware recommends the use of certificates signed by a trusted Certification Authority.

View Certificate... Cancel

5. Click Accept on Certificate Information page.

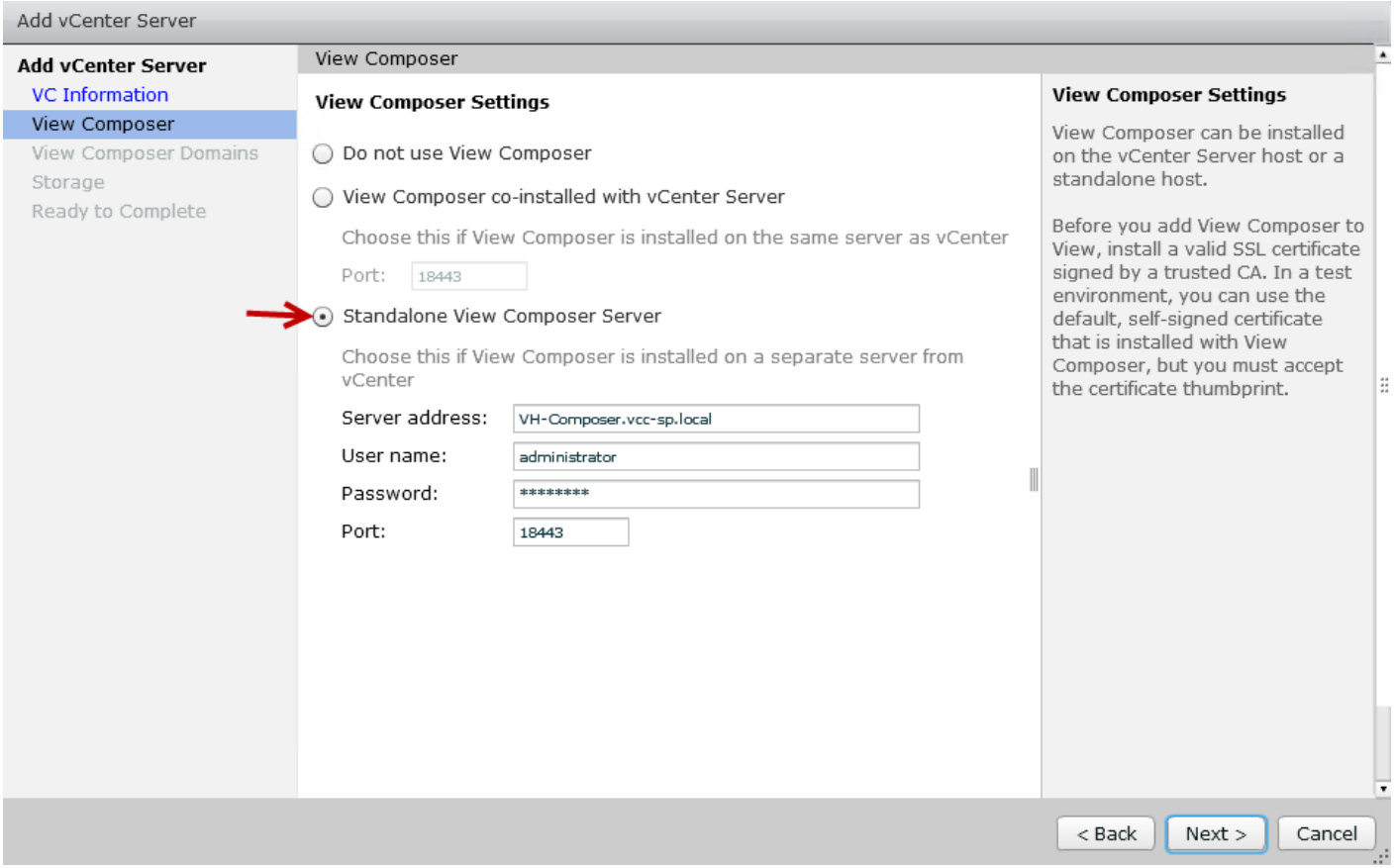


Certificate Information

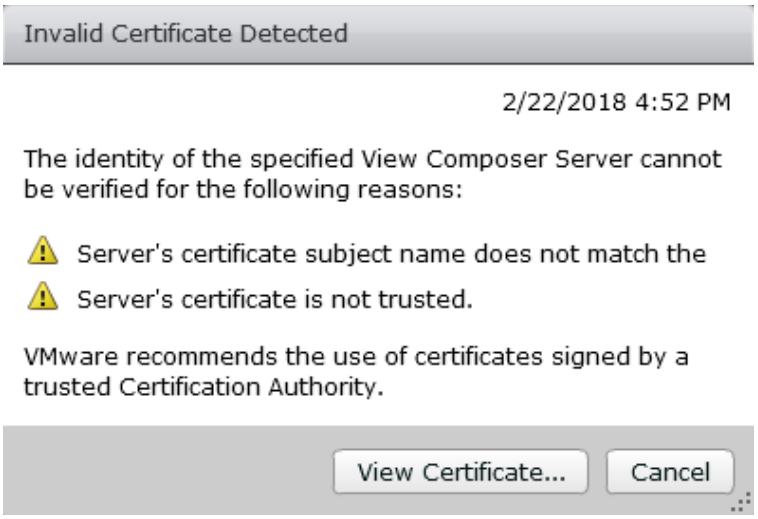
Issued to:	VCC-VCSA65.vcc-sp.local
Issued by:	CA
Valid from:	2/21/18 9:46 AM to 2/16/28 9:46 AM
Subject:	C=US CN=VCC-VCSA65.vcc-sp.local
Issuer:	OU=VMware Engineering O=VCC-VCSA65.vcc-sp.local ST=California C=US DC=local DC=vsphere CN=CA
Serial Number:	00 e1 e7 30 62 e0 6c f5 f7
Version:	3
Signature Algorithm:	SHA256withRSA
Public Key Algorithm:	RSA
Public Key:	30 82 01 22 30 0d 06 09 2a 86 48 86 f7 0d 01 01 01 05 00 03 82 01 0f 00 3 82 01 0a 02 82 01 01 00 b7 aa 26 77 25 f6 7f 91 74 de 84 d5 e6 b1 92 dd 6 ff d7 71 94 f6 a1 cd 7c fa a7 56 10 53 c6 7f fc 86 4d fb 2e 04 39 ef f0 24 0e 6e a4 83 6d d1 f8 9d 98 dd c2 96 d1 1b 5e 9c 46 fb ec bc 60 d6 26 ee dc 4b ab 58 8f 37 31 d5 ac 3f f9 7d 40 c1 74 f5 49 c5 44 cf aa 9b 9b dc 51 b1 0c 1 db d2 04 f5 15 dc a3 99 ef f9 cf 2f 90 8a bc b8 cd 02 4b 15 10 b6 2e 2d 86 e ed ff 63 f0 11 ca b5 ed 9a cf ac 2d 3c d3 68 b8 50 f7 2c d7 9d 33 d0 e6 d5 d 9a c0 2a c3 b1 a6 2f 57 ce cf 02 0f 25 ba bb 6f ea d5 03 fb c1 f2 b0 19 02 1 d2 f3 fc 10 12 99 6b d9 e0 5a 56 0e 32 09 cf d1 1d 6b 2a 1d d2 73 08 ad e5 b8 62 06 c6 9d ed 2a 56 72 31 e8 b1 32 e9 f6 30 eb 42 c1 8b 3d 36 bf 3e e a5 91 b8 cd da 32 83 18 5d f7 ab f6 3b eb 78 3c f2 ea 10 9b d4 4c 51 8a 21 37 56 6e 31 ad 78 27 4d 4b 02 03 01 00 01
Thumbprint Algorithm:	SHA-1
Thumbprint:	2f f4 ee bc fb 47 f9 97 54 10 70 26 8a 81 4a 12 5f d9 8d b1
Key Usage:	digitalSignature, keyEncipherment, keyAgreement
Subject Alternative Name(s):	dNSName:VCC-VCSA65.vcc-sp.local

Accept Reject

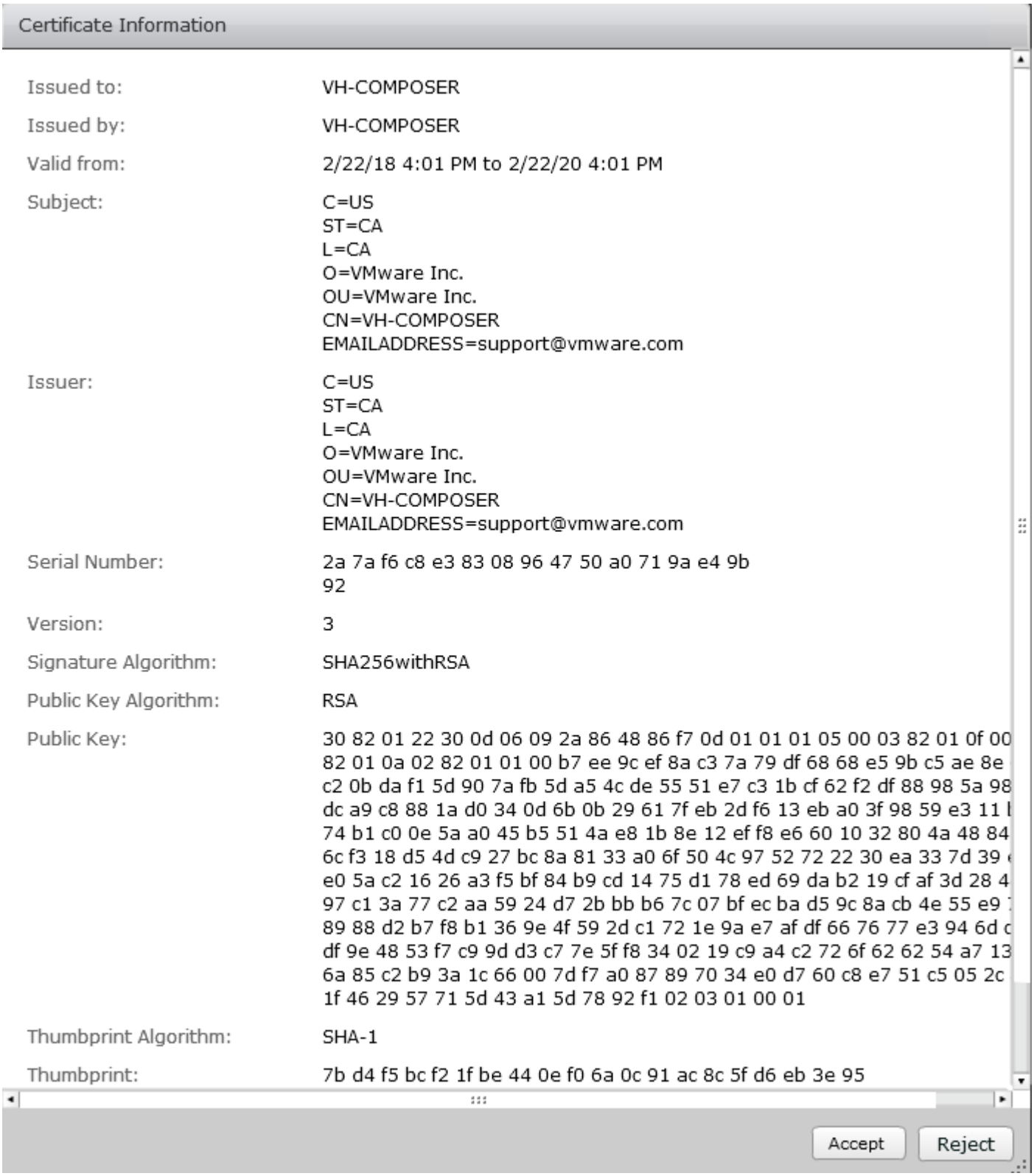
6. On View Composer settings page, select Standalone View Composer Server.
   a. Enter View Composer Server server FQDN or IP address
   b. user name
   c. password
   d. port number
7. Click Next.



8. Click \ View Certificate.



9. On Certificate Information page, click Accept.

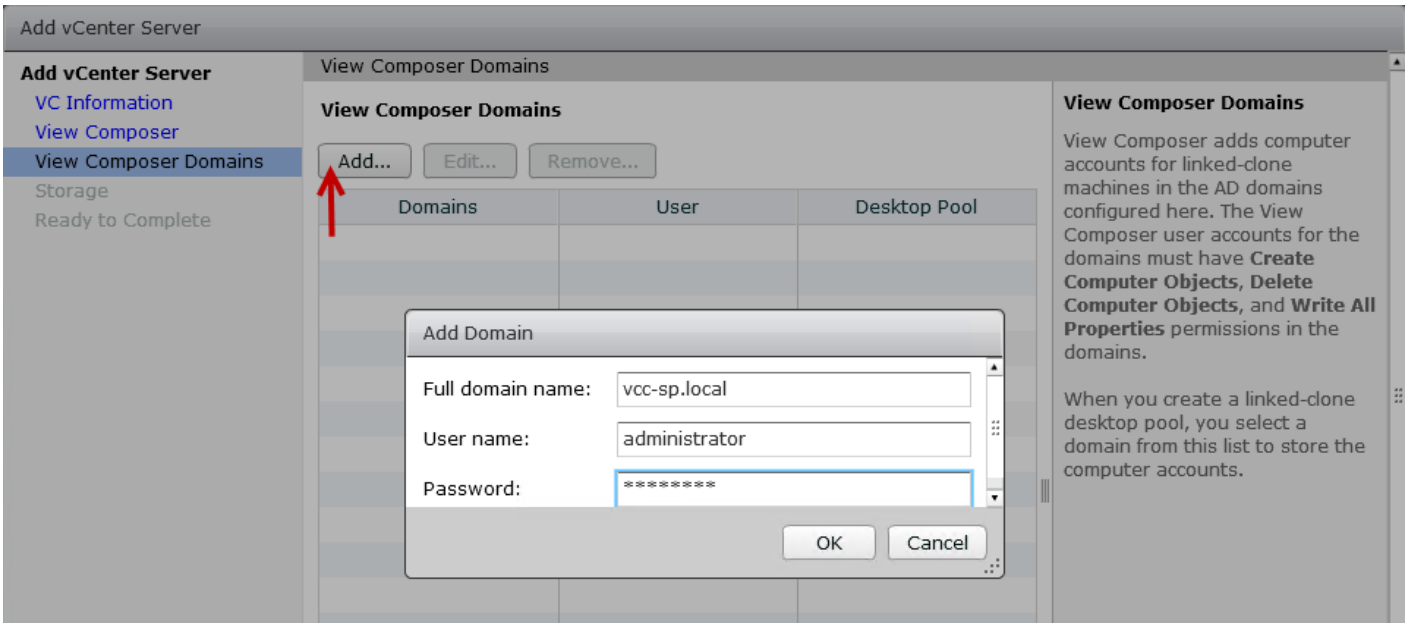


Certificate Information

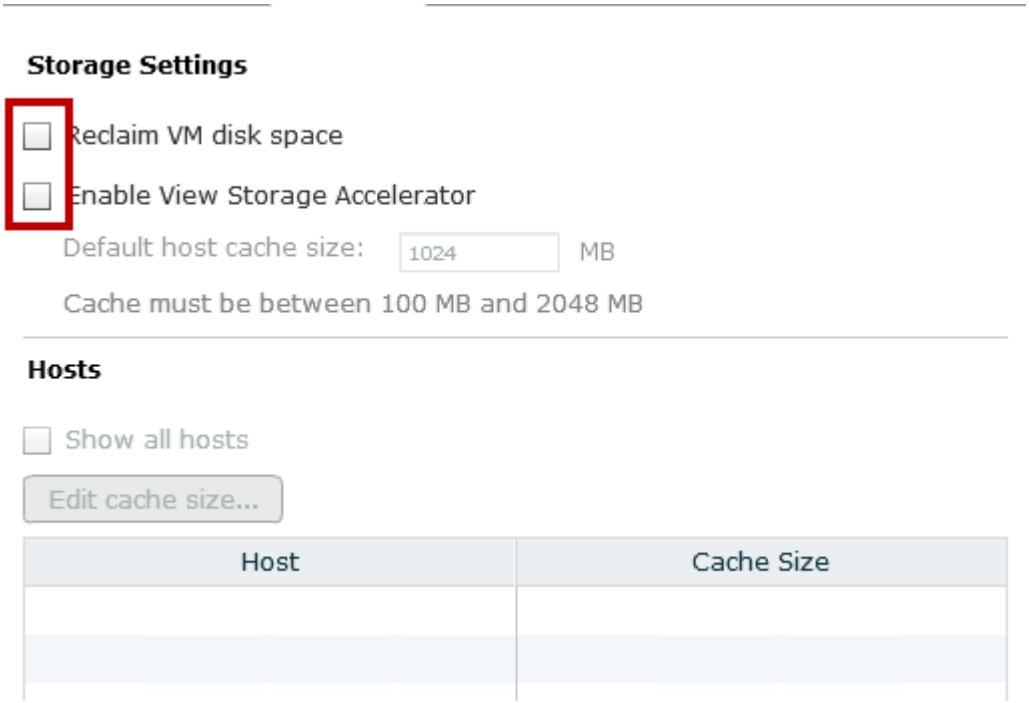
Issued to:	VH-COMPOSER
Issued by:	VH-COMPOSER
Valid from:	2/22/18 4:01 PM to 2/22/20 4:01 PM
Subject:	C=US ST=CA L=CA O=VMware Inc. OU=VMware Inc. CN=VH-COMPOSER EMAILADDRESS=support@vmware.com
Issuer:	C=US ST=CA L=CA O=VMware Inc. OU=VMware Inc. CN=VH-COMPOSER EMAILADDRESS=support@vmware.com
Serial Number:	2a 7a f6 c8 e3 83 08 96 47 50 a0 71 9a e4 9b 92
Version:	3
Signature Algorithm:	SHA256withRSA
Public Key Algorithm:	RSA
Public Key:	30 82 01 22 30 0d 06 09 2a 86 48 86 f7 0d 01 01 01 05 00 03 82 01 0f 00 82 01 0a 02 82 01 01 00 b7 ee 9c ef 8a c3 7a 79 df 68 68 e5 9b c5 ae 8e c2 0b da f1 5d 90 7a fb 5d a5 4c de 55 51 e7 c3 1b cf 62 f2 df 88 98 5a 98 dc a9 c8 88 1a d0 34 0d 6b 0b 29 61 7f eb 2d f6 13 eb a0 3f 98 59 e3 11 74 b1 c0 0e 5a a0 45 b5 51 4a e8 1b 8e 12 ef f8 e6 60 10 32 80 4a 48 84 6c f3 18 d5 4d c9 27 bc 8a 81 33 a0 6f 50 4c 97 52 72 22 30 ea 33 7d 39 e0 5a c2 16 26 a3 f5 bf 84 b9 cd 14 75 d1 78 ed 69 da b2 19 cf af 3d 28 4 97 c1 3a 77 c2 aa 59 24 d7 2b bb b6 7c 07 bf ec ba d5 9c 8a cb 4e 55 e9 89 88 d2 b7 f8 b1 36 9e 4f 59 2d c1 72 1e 9a e7 af df 66 76 77 e3 94 6d df 9e 48 53 f7 c9 9d d3 c7 7e 5f f8 34 02 19 c9 a4 c2 72 6f 62 62 54 a7 13 6a 85 c2 b9 3a 1c 66 00 7d f7 a0 87 89 70 34 e0 d7 60 c8 e7 51 c5 05 2c 1f 46 29 57 71 5d 43 a1 5d 78 92 f1 02 03 01 00 01
Thumbprint Algorithm:	SHA-1
Thumbprint:	7b d4 f5 bc f2 1f be 44 0e f0 6a 0c 91 ac 8c 5f d6 eb 3e 95

Accept Reject

10. On View Composer Domains page, click Add. Enter Domain Name, User name and Password. Click OK.



11. On Storage Settings page, enable View Storage Accelerator. Click Next.



View Storage Accelerator is automatically enabled for Instant Clone pool deployment. Clone-level CBRC is no longer needed, so you do not need to specify the level of CBRC. Master VMs and replicas still use CBRC, and the CBRC digest is calculated automatically.

12. Click Finish.

For more information about VMware Horizon Administration go to: https://docs.vmware.com/en/VMware-Horizon-7/7.4/horizon-administration.pdf

For more informantion about VMware Horizon configuration and Tuning best practices on Pure Storage go to:
https://support.purestorage.com/Solutions/VMware_Platform_Guide/zzzVMware_Horizon_View/Best_Practices%3A_Configuration_and_Tuning

Preparing the Master Image for Tested Horizon Deployment Types

This section provides guidance around creating the golden (or master) image(s) for the environment. For this CVD, the images contain the basics needed to run the Login VSI workload.

The master image for RDSH Server Farm was configured with Windows Server 2016 64 bit and master image for Persistent/Non-Persistent Desktop Pool was with Windows 10 64 bit OS listed in Table 16 .

Table 16 Master Image Configuration

Attribute	Linked-Clone/Instant-clone	Persistent/Full Clone	RDSH server
Desktop operating system	Microsoft Windows 10 Enterprise (64-bit)	Microsoft Windows 10 Enterprise (64-bit)	Microsoft Windows Server 2016 standard (64-bit)
Hardware	VMware Virtual Hardware Version 13	VMware Virtual Hardware Version 13	VMware Virtual Hardware Version 13
vCPU	2	2	8
Memory	3 GB	3 GB *	32 GB
Memory reserved	3 GB	3 GB *	32 GB
Video RAM	35 MB	35 MB	35MB
3D graphics	Off	Off	Off
NIC	1	1	1
Virtual network adapter 1	VMXNet3 adapter	VMXNet3 adapter	VMXNet3 adapter
Virtual SCSI controller 0	Paravirtual	Paravirtual	Paravirtual
Virtual disk: VMDK 1	40 GB	100 GB	40 GB
Virtual disk: VMDK 2 (non-persistent disk for Linked-Clones)	6 GB	-	-

Attribute	Linked-Clone/Instant-clone	Persistent/Full Clone	RDSH server
Virtual floppy drive 1	Removed	Removed	Removed
Virtual CD/DVD drive 1	–	–	-
Applications	• Login VSI 4.1.32 application installation • Adobe Acrobat 11 • Adobe Flash Player 16 • Doro PDF 1.82 • FreeMind • Microsoft Internet Explorer Microsoft Office 2016	• Login VSI 4.1.32 application installation • Adobe Acrobat 11 • Adobe Flash Player 16 • Doro PDF 1.82 • FreeMind • Microsoft Internet Explorer Microsoft Office 2016	• Login VSI 4.1.32 application installation • Adobe Acrobat 11 • Adobe Flash Player 16 • Doro PDF 1.82 • FreeMind • Microsoft Internet Explorer Microsoft Office 2016
VMware tools	Release 10287	Release 10287	Release 10287
VMware View Agent	Release 7.4.0-7400533	Release 7.4.0-7400533	Release 7.4.0-7400533

* For Windows 10 Desktops, we configured 3GB of RAM as amount of memory allocated is sufficient to run LoginVSI Knowledge Worker workload. Cisco UCS B200 M5 nodes were configured with 768GB of total memory for this performance study. By adding memory to each Cisco UCS node, for example twelve additional 64GB DIMMs per node, we could allocate up to 10GB of RAM per VM at the same user density.

Prepare Microsoft Windows 10 and Server 2016 with Microsoft Office 2016

Prepare your master image for one or more of the following use cases:

- VMware Horizon 7 Linked Clones
- VMware Horizon 7 Instant Clones
- VMware Horizon 7 Full clones
- VMware Horizon 7 RDSH Virtual Machines

Include Microsoft Office 2016 and other applications used by all pool users in your organization into your master image.

Apply Microsoft updates to your master images.

For this study, we added Login VSI target software to enable the use the Login VSI Knowledge Worker workload to benchmark end user experience for each use case.

Optimization of Base Windows 10 or Server 2016 Guest OS

The links below provide access to information about optimizing Windows 10 for VDI deployment:

http://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/whitepaper/vmware-view-optimizationguidewindows7-en-white-paper.pdf

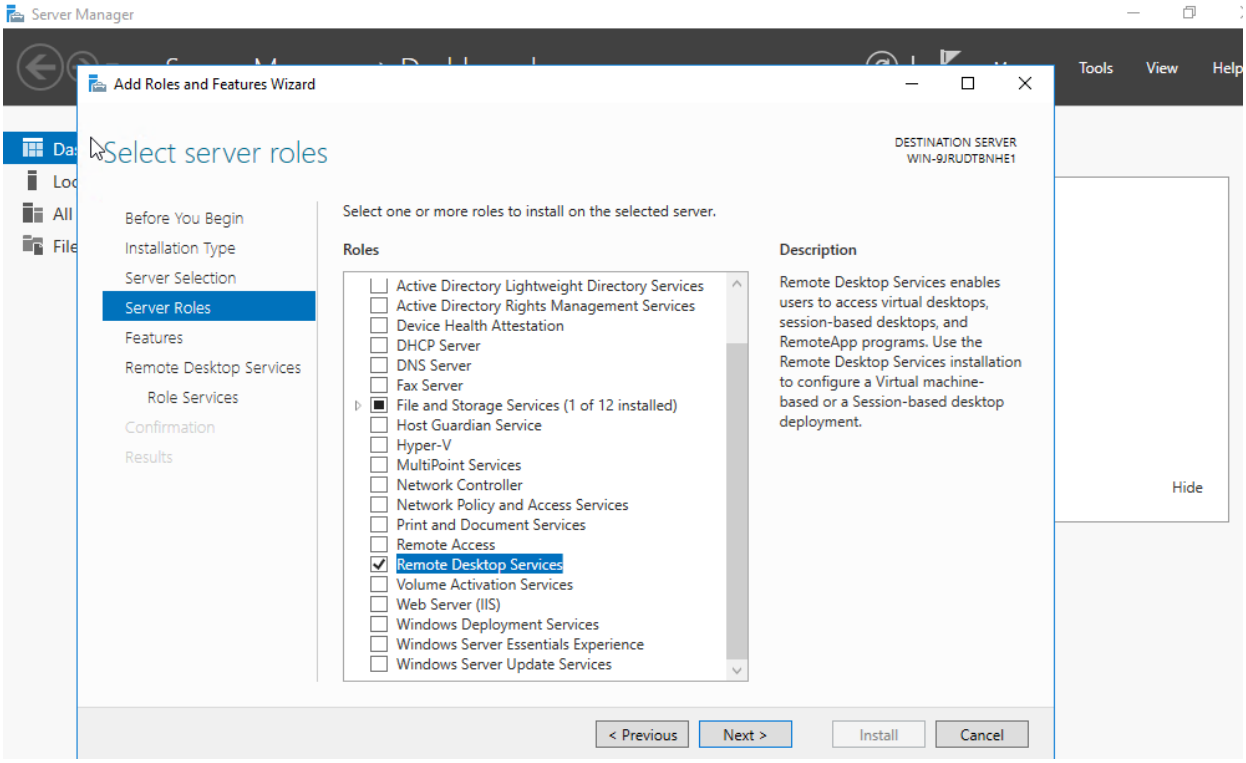
VMware Optimization tool for Windows 10 and Windows Server 2016 for RDSH: https://labs.vmware.com/flings/vmware-os-optimization-tool.

Add Remote Desktop Services on RDSH Master Image

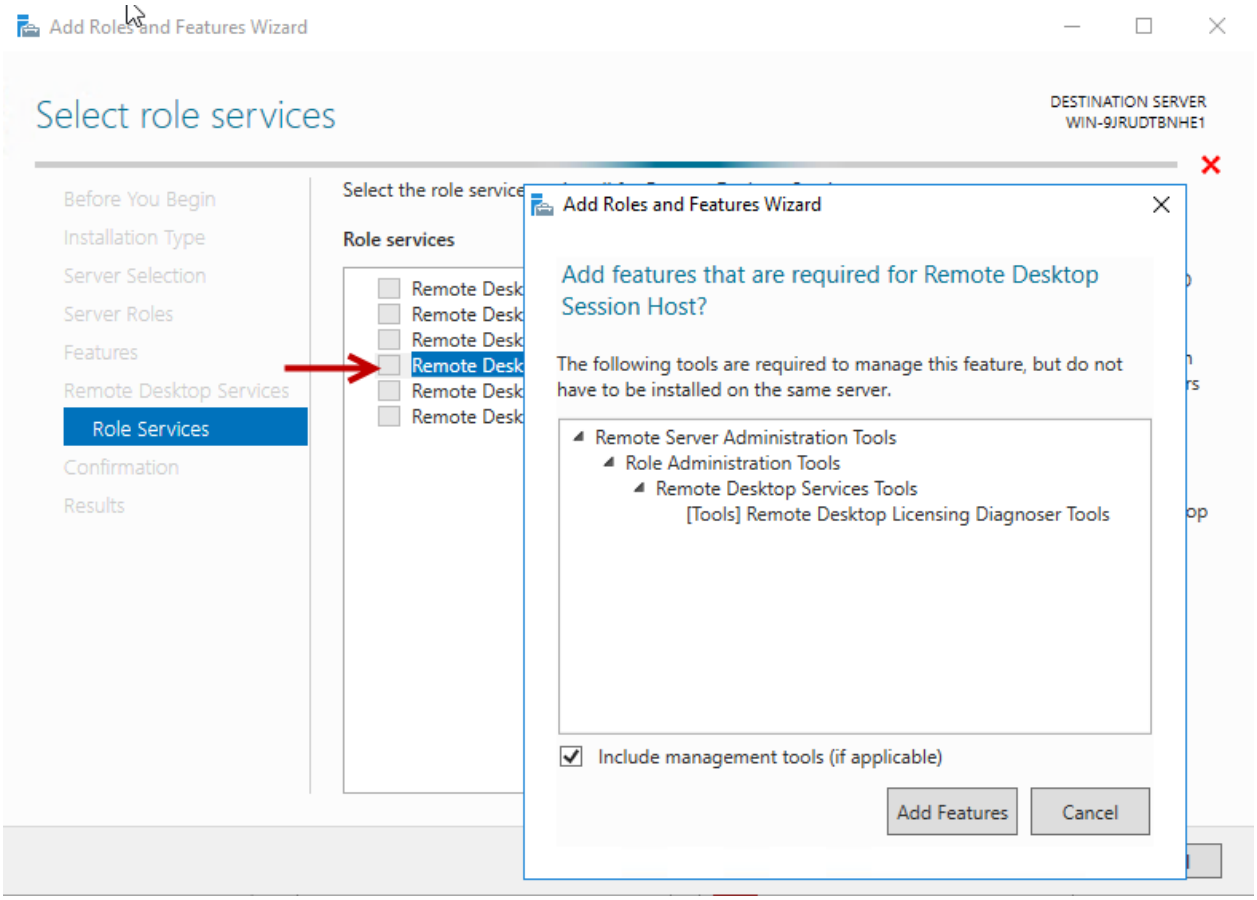
To install the Virtual Desktop Agent software for Horizon, complete the following steps:

Prior to installing the Horizon View Agent on a Microsoft Server 2016 virtual machine, you must add the Remote Desktop Services role and the Remote Desktop Session Host role service.

1. To add Remote Desktop Services role on Windows Server OS from the Server Manager, use the Add Roles and Features wizard:



2. Add “Remote Desktop Session Host” role service.

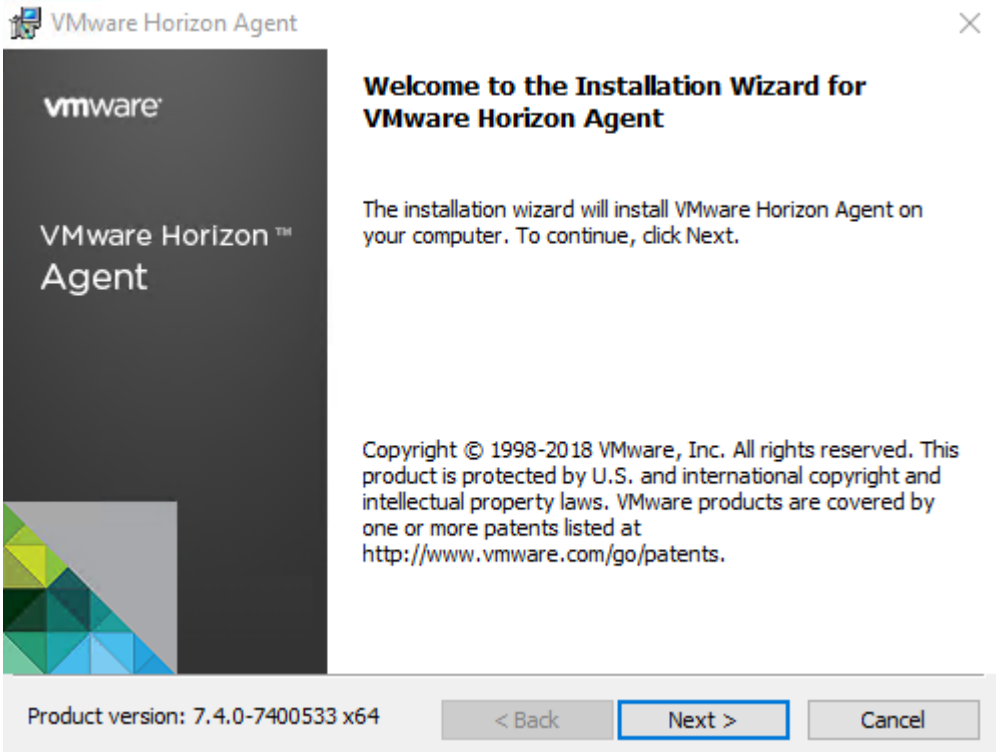


3. Click Install.

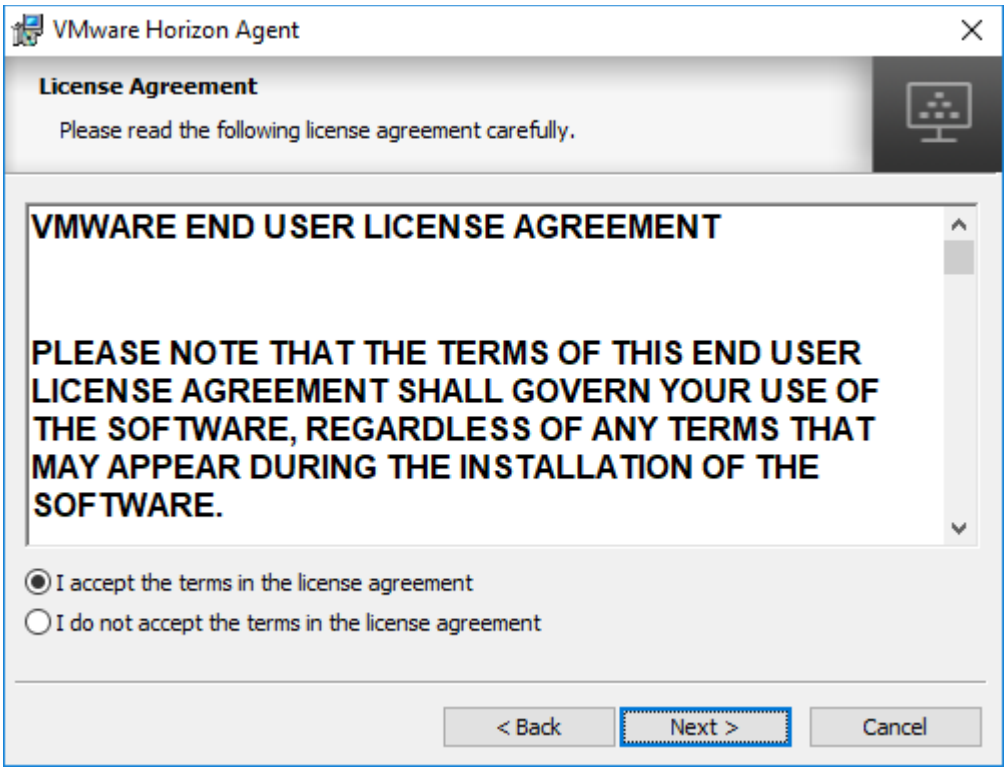
Virtual Desktop Agent Software Installation for Horizon

To install Horizon Agent on master image, complete the following steps:

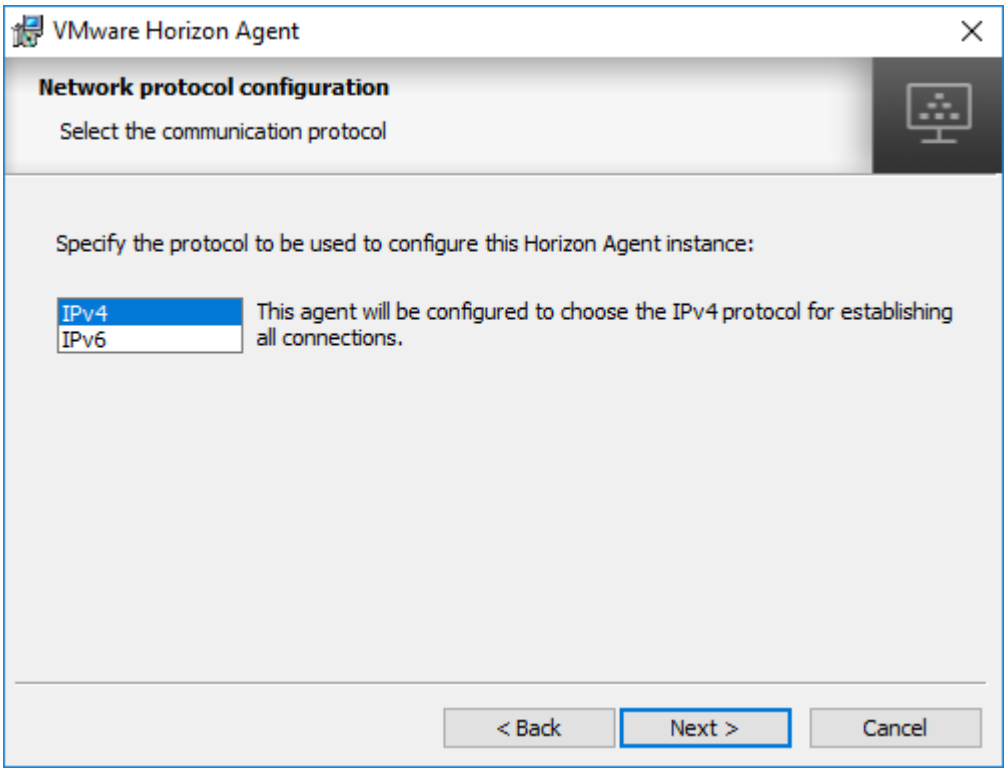
1. Open the Horizon View Agent Installer, VMware-viewagent-x86_64-7.4.0-7400533.exe. Click Next to install.



2. Review and accept the EULA Agreement. Click Next.

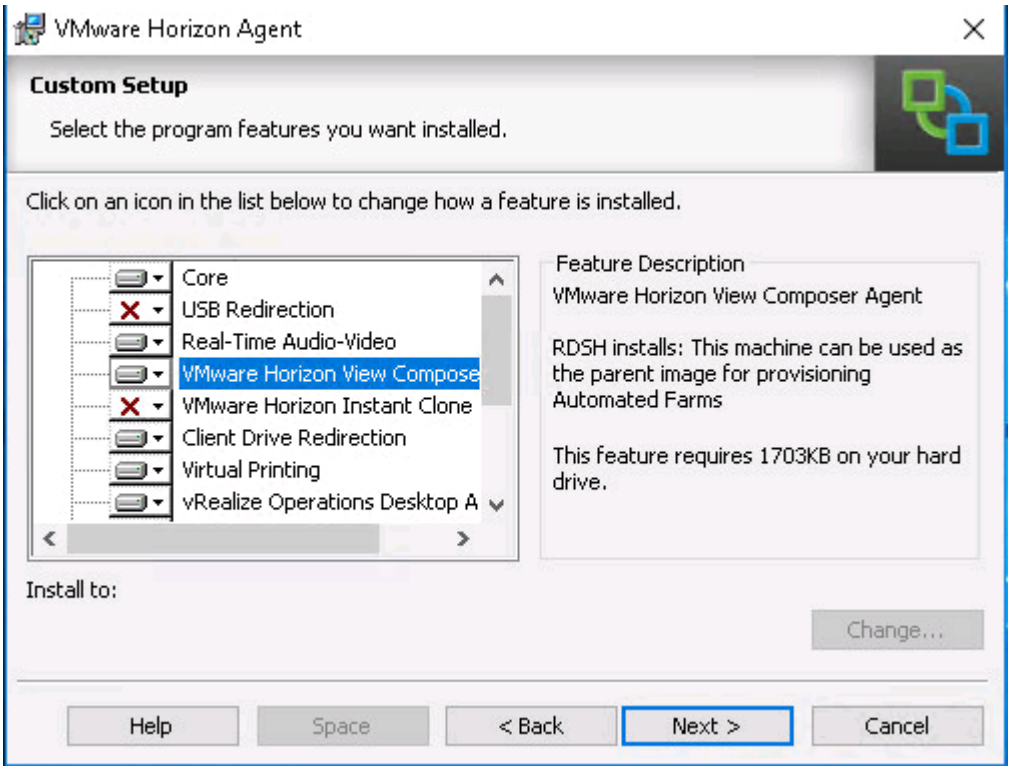


3. Select Network protocol configuration, click Next.

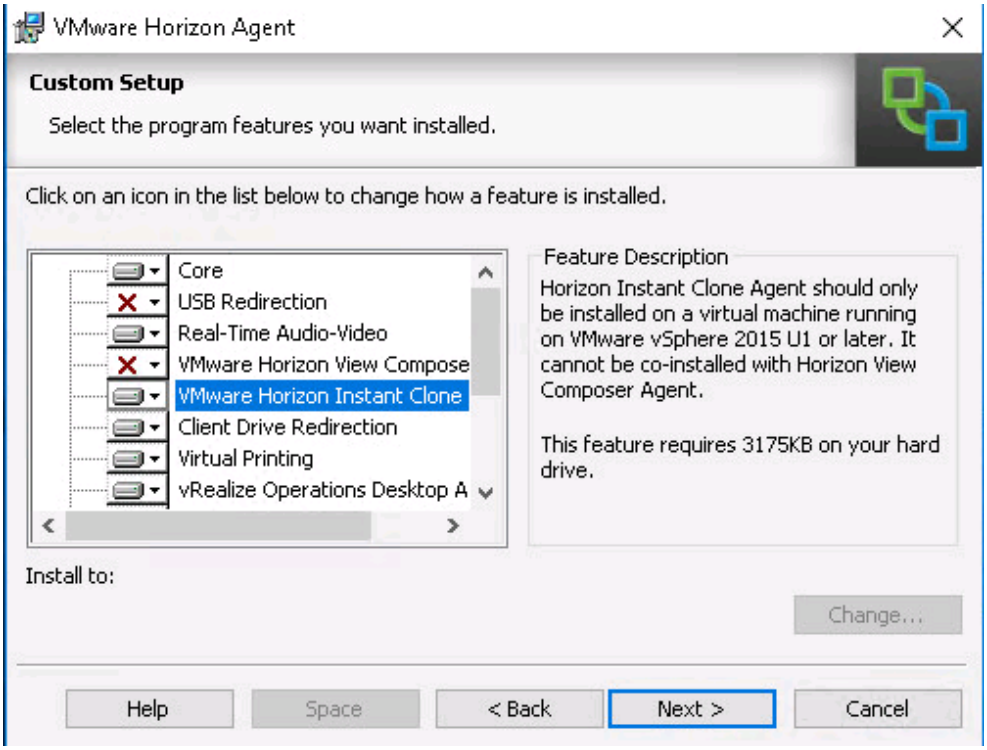


4. Based on the Desktop pool you want to create, select either View Composer Agent or Instant Clone Agent installation. Installation of both features on the same master image is not supported.

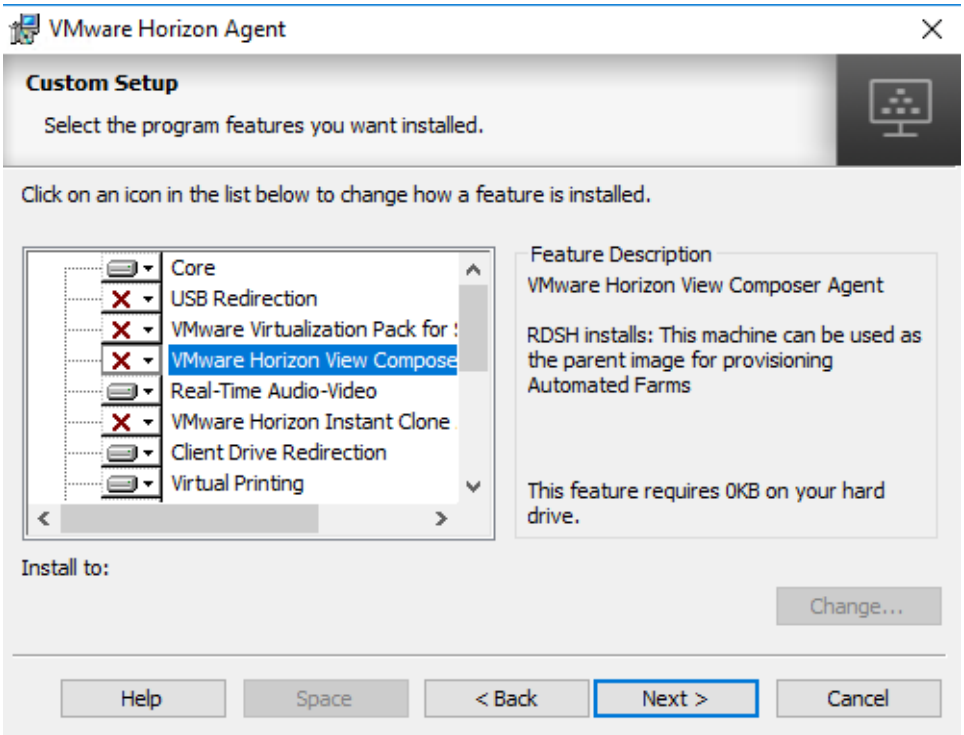
5. Enable installation of the VMware Horizon View Composer Agent for linked-clone virtual machines.



6. Disable the Horizon View Composer Agent and enable the Horizon Instant Clone Agent for Instant Clone floating assigned desktop pool creation.



7. For Full-Clone/Persistent Desktop Pool creation no need to install either VMware Horizon View Composer or VMware Horizon Instant Clone feature.

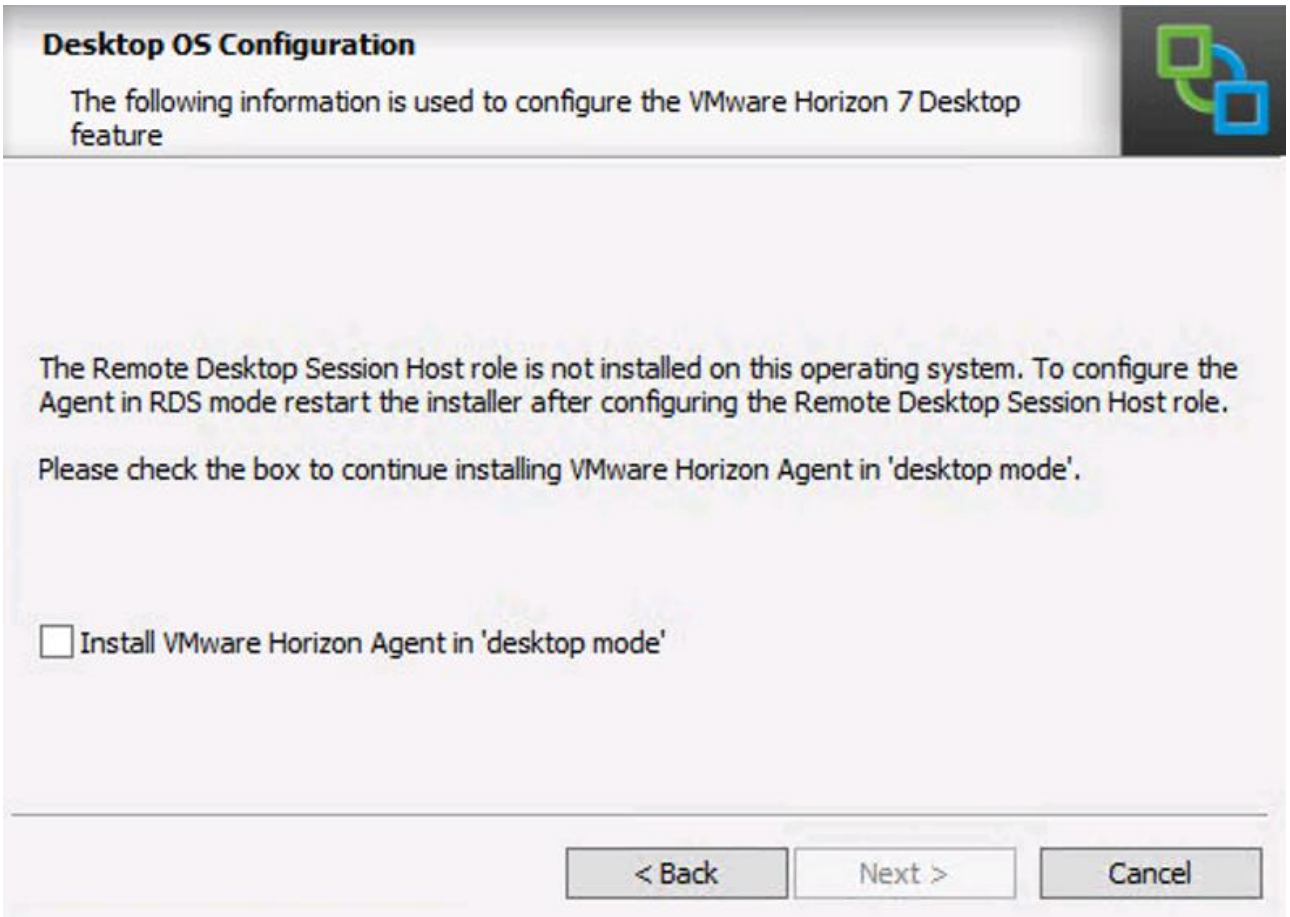


8. For RDSH Pool creation, Both Linked-Clone and Instant-Clone RDSH Farm creation is supported select either of the feature to install on RDSH master image.

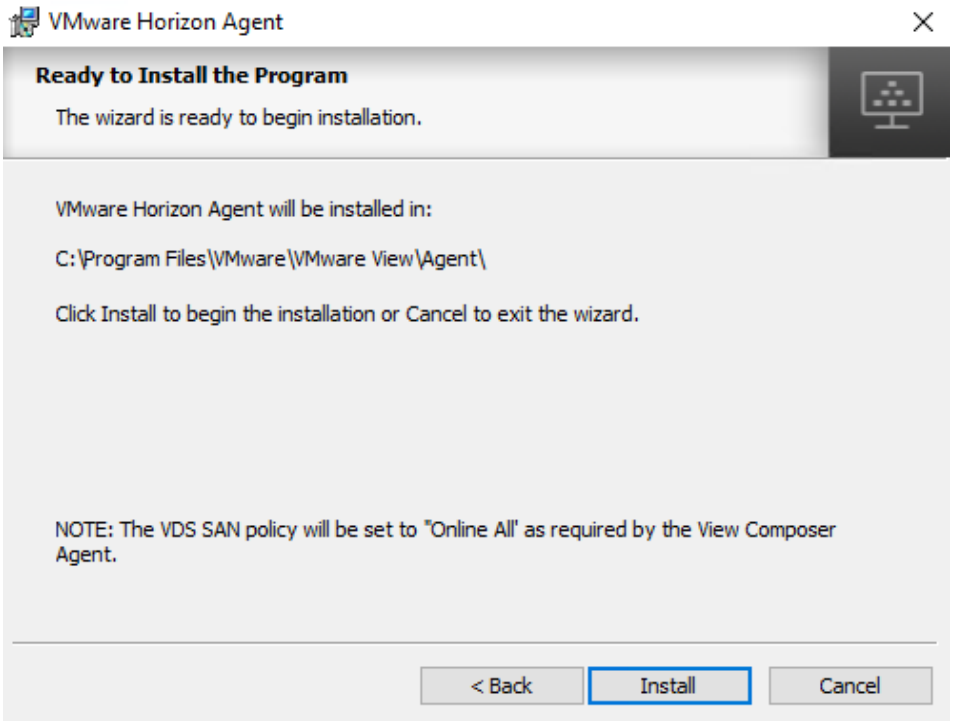
For 3D enabled RDSH deployment install “3D RDSH” feature to emable hardware 3D acceleration in RDSH sessions.

For this study we installed “VMware Horizon View Composer Agent” on the master image for RDSH which was later utilized to deploy RDSH Farm and RDSH Pool via Horizon Administrator Console as describe in section VMware Horizon Farm and Pool Creation.

9. View Agent will report “Desktop Mode” if Remote Desktop Services is not installed.



10. Click Install.



Install Additional Software

To install additional software required for your base windows image, complete the following steps:

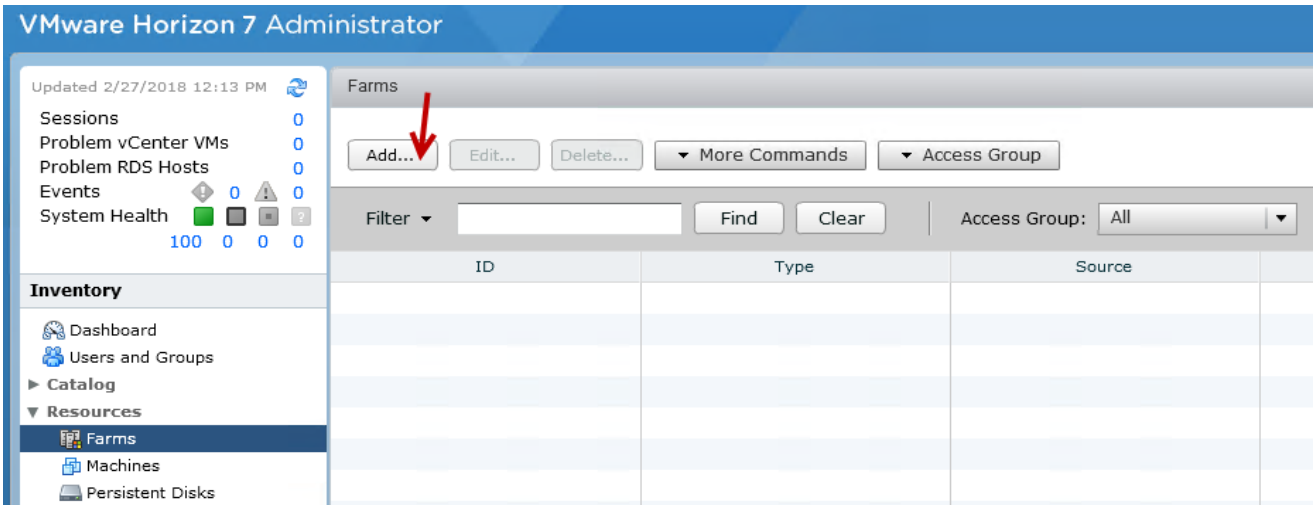
1. For testing, we installed Office 2016 64bit version.
2. Log into the VSI Target software package to facilitate workload testing.
3. Install service packs and hot fixes required for the additional software components that were added.
4. Reboot or shut down the VM as required.

VMware Horizon Farm and Pool Creation

RDSH Farm Creation

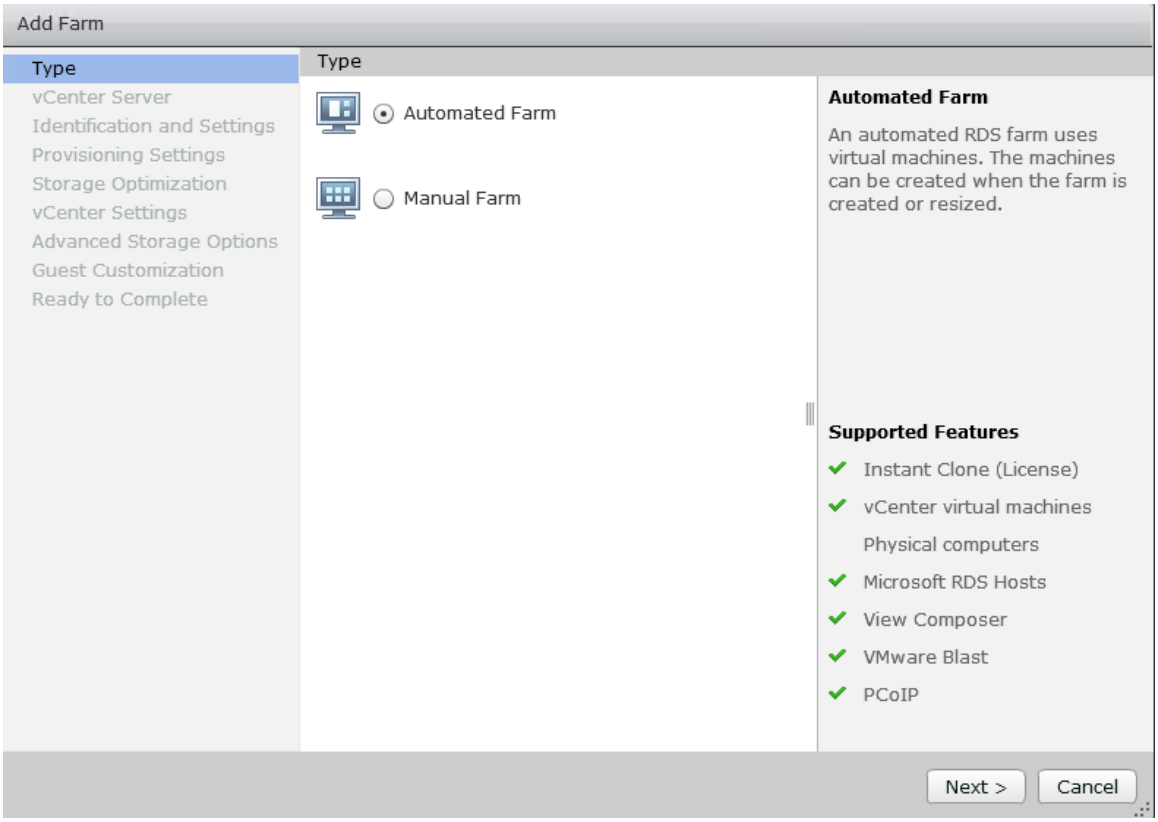
To create RDSH Farm, complete the following steps:

1. In the VMware Horizon Administration console, select Farms under the Resource node of the Inventory pane.
2. Click Add in the action pane to create a new RDSH Farm.

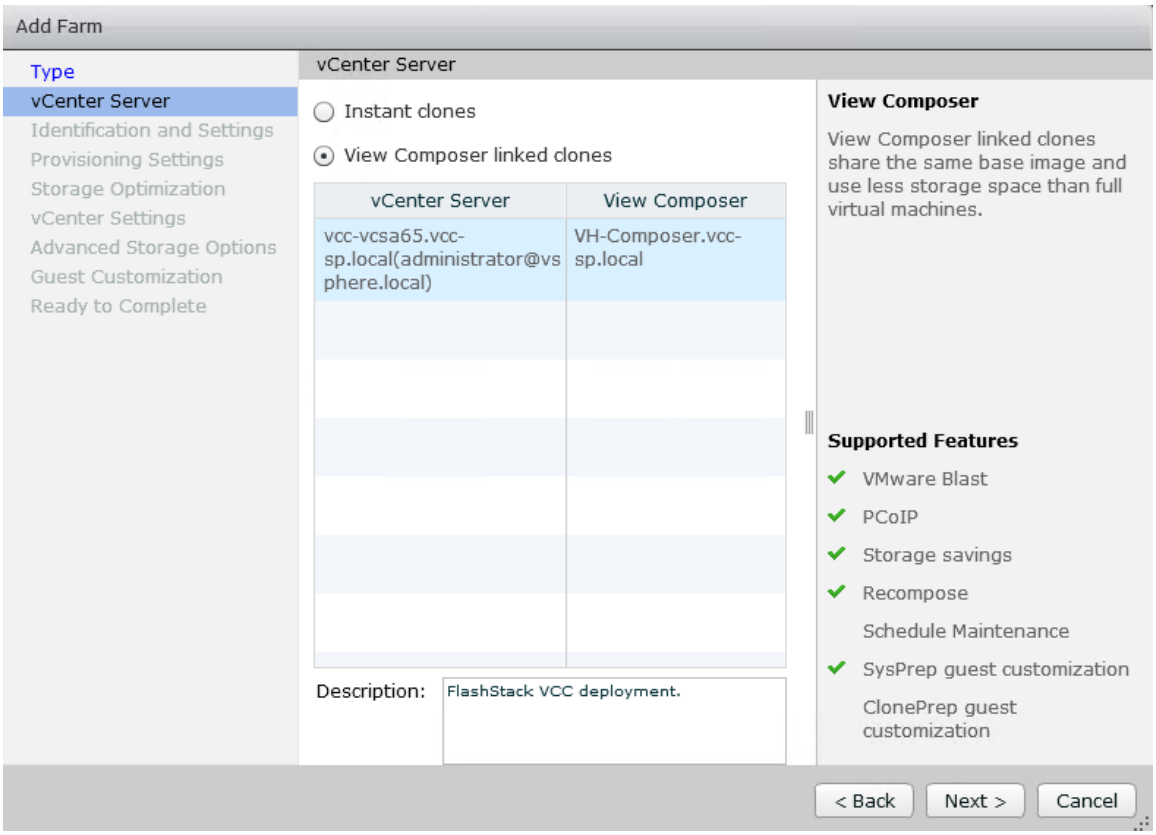


3. Select either to create an Automated or Manual Farm. In this solution, we selected Automated Farm.

A Manual Farm requires a manual registration of each RDSH server to Horizon Connection or Replica Server instance.



4. Select deployment type whether Instant-clone or Linked-Clone and vCenter server for RDSH farm.

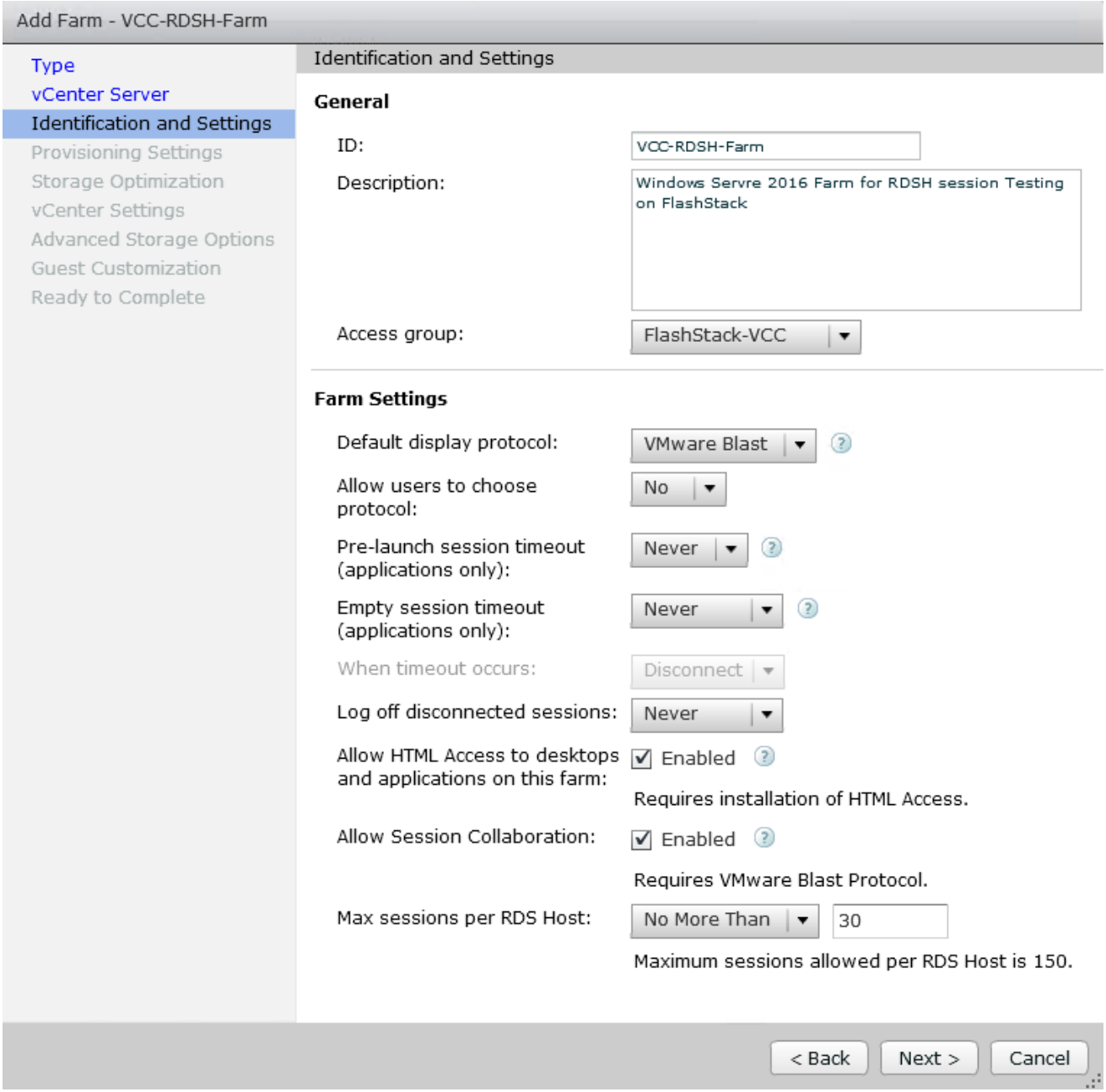


5. Enter the RDSH Farm ID, Access group, Default Display Protocol (Blast/PCoIP/RDP).

6. Select if users are allowed to change the default display protocol, Session timeout, Logoff Disconnected users.

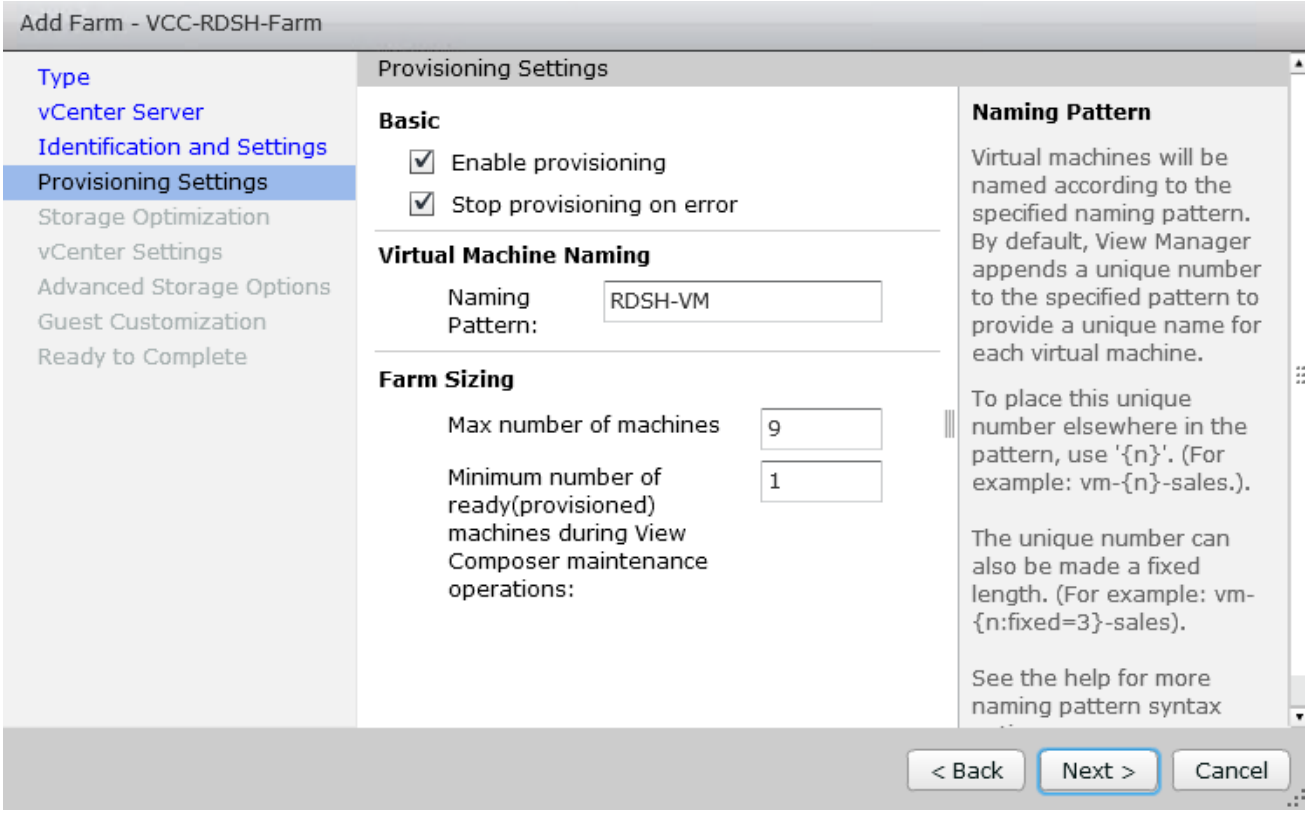
7. Enable check box to allow HTML access and Session Collaboration.

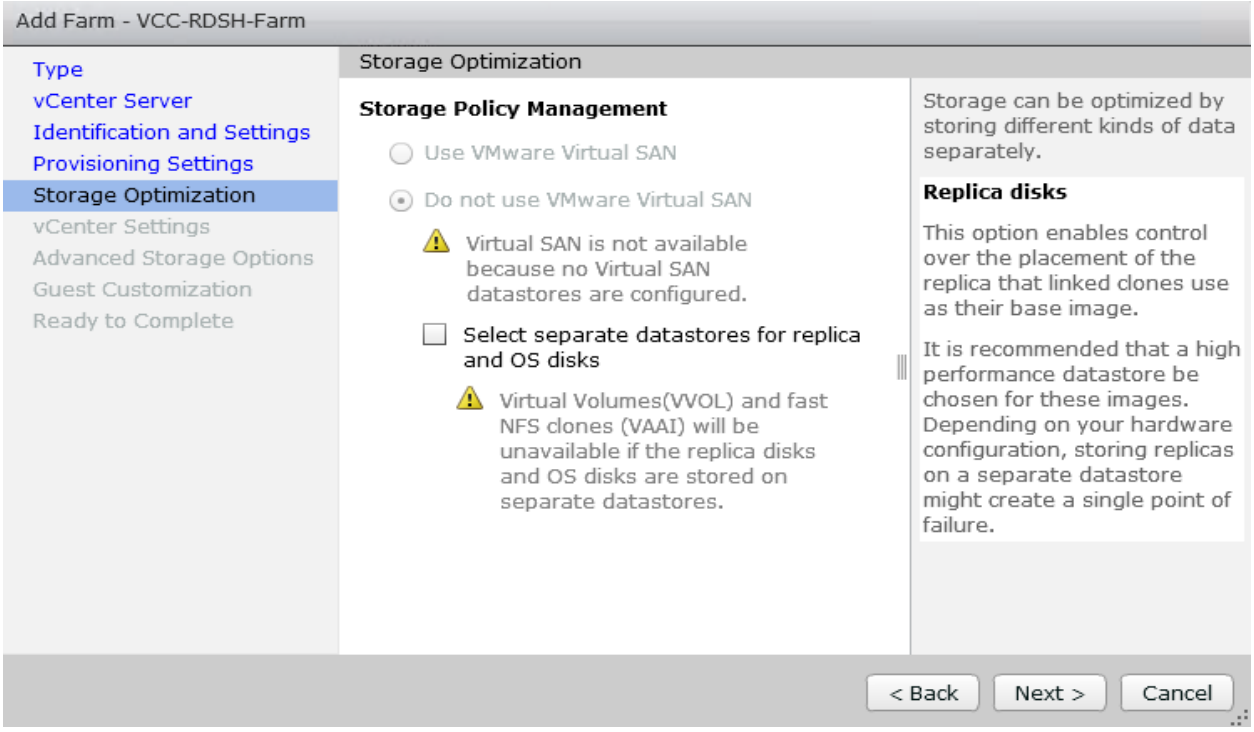
We limited our Maximum sessions per RDS Host to 30 (VMware Horizon 7 recommendation is 60 max sessions per RDSH).



8. Select the provisioning settings, naming convention for RDSH server VM to deploy, and the number of VMs to deploy.

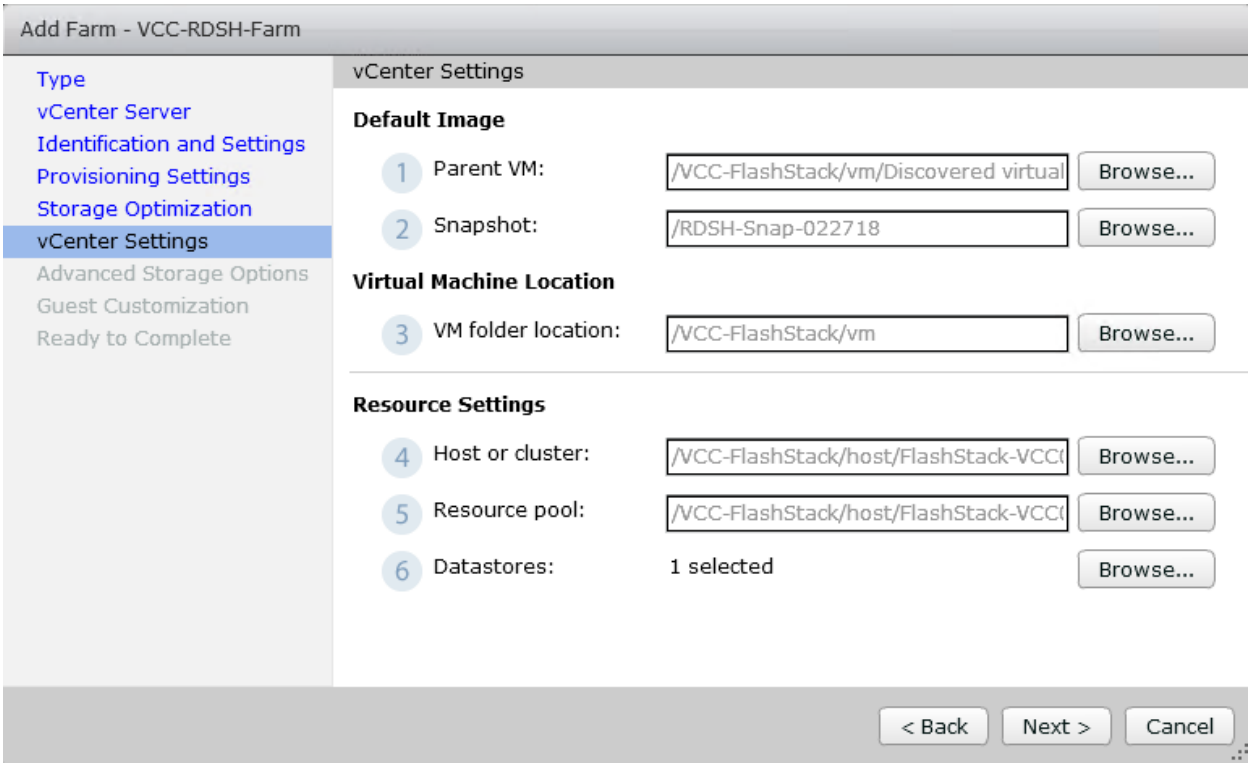
In this study, we deployed 10 RDSH virtual machine for Single Server testing and 100 RDSH virtual machines across our 10 node part of the RDSH cluster created in vCenter.



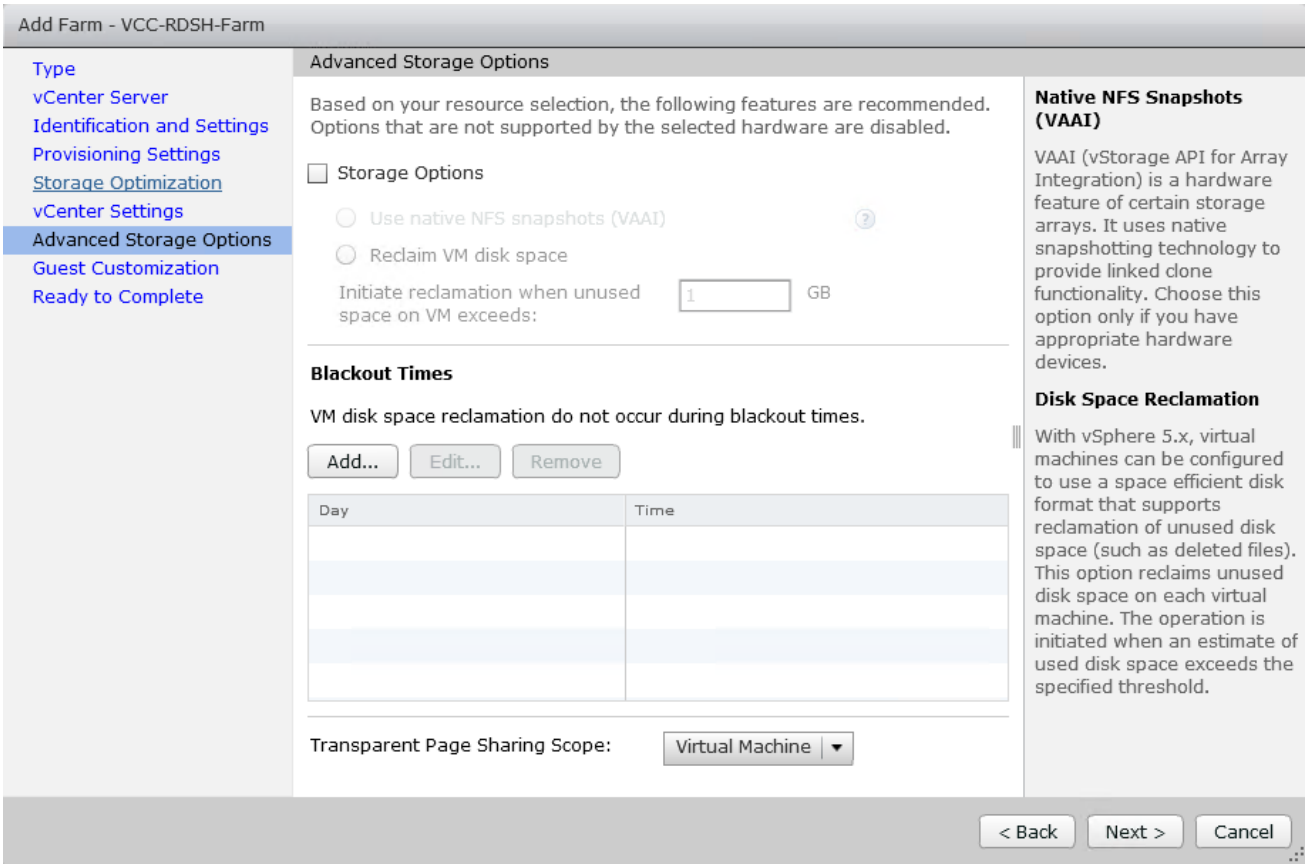


9. Select vCenter settings as asked; i.e. Master Image, snapshot, folder, Host or Cluster, resource pool, storage selection.

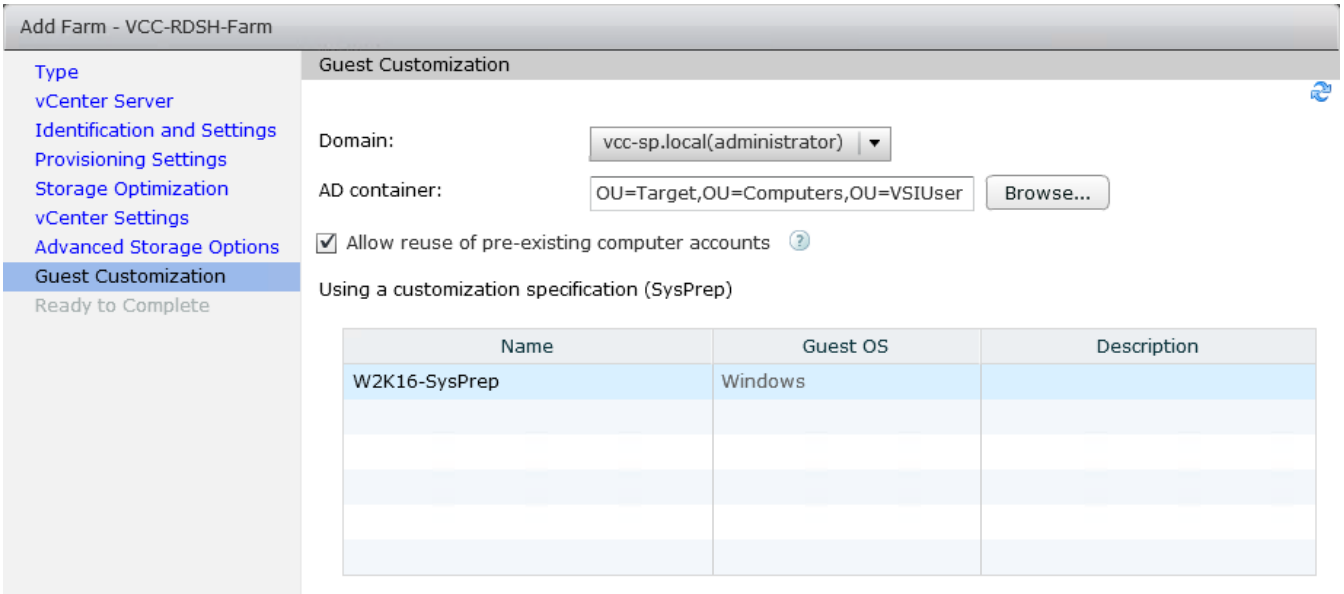
10. For Step 6 Datastores selection; select intended datastore provisioned to host RDSH VMs, choose Unbounded for the Storage Overcommit field.



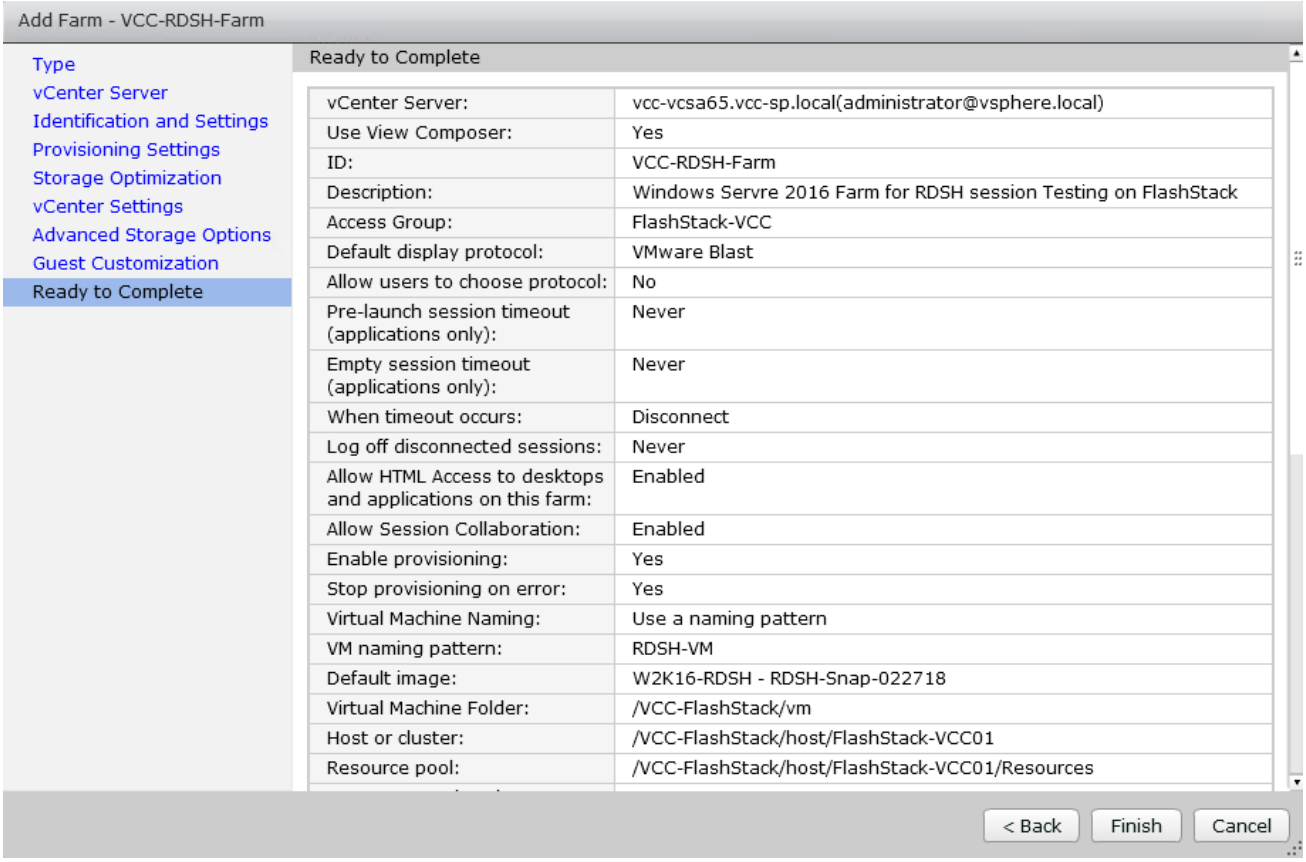
11. In the Advanced Storage Options, disable Storage Accelerator.



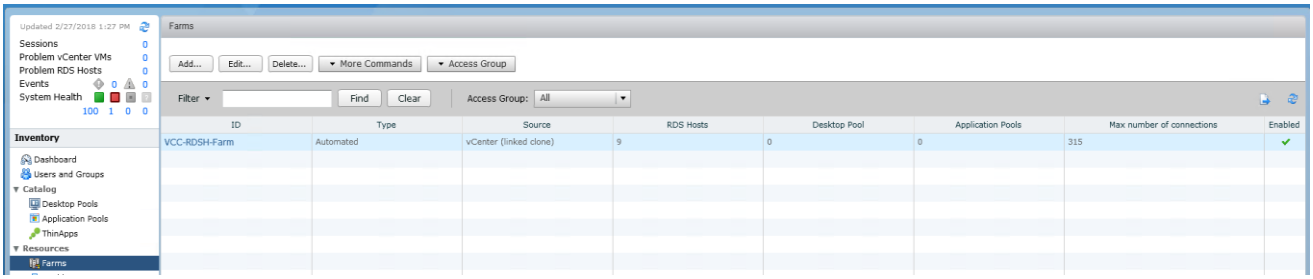
12. Select the Active Directory Domain, the Active Directory OU into which the RDSH machines will be provisioned, and the Sysprep file created as part of the customization specific configuration performed earlier.

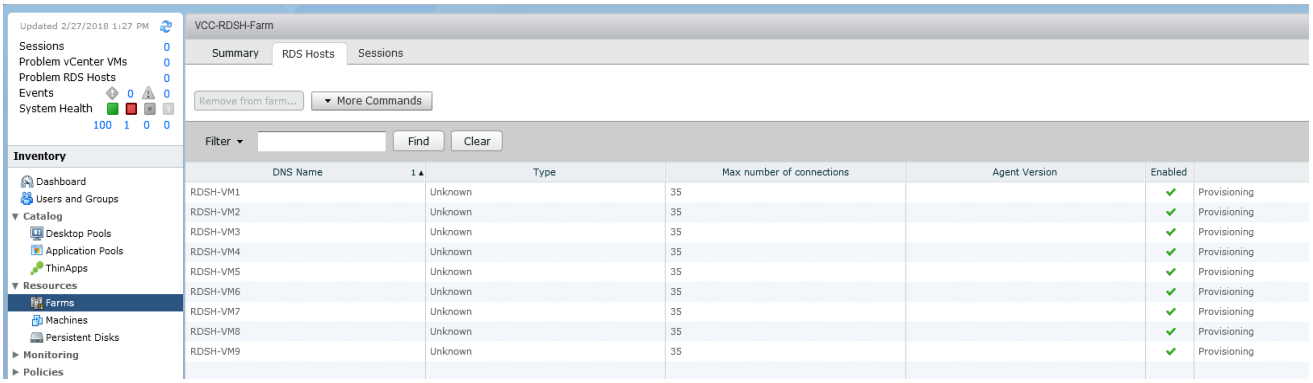


13. Review the pool creation information. Click Finish.



14. The VMware Horizon Administration console displays the status of the provisioning task and pool settings:

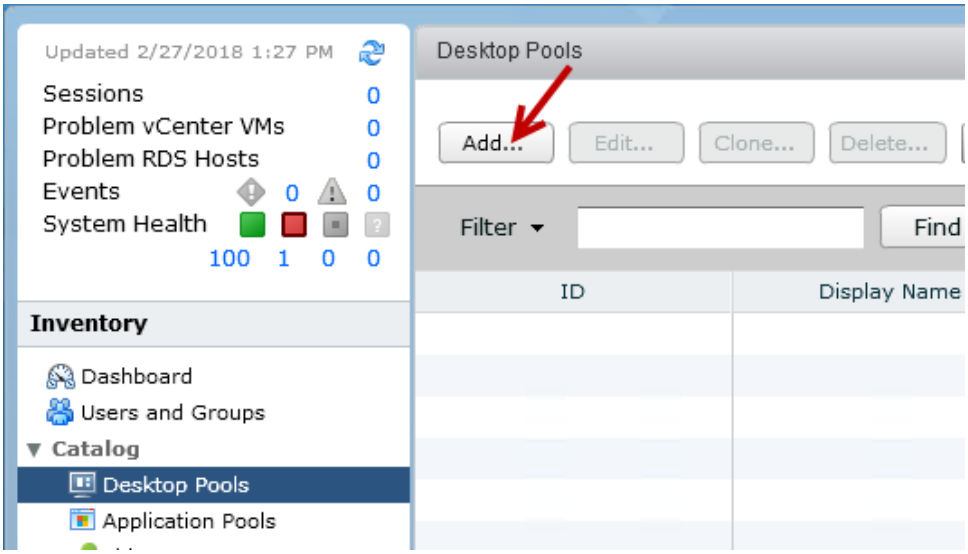




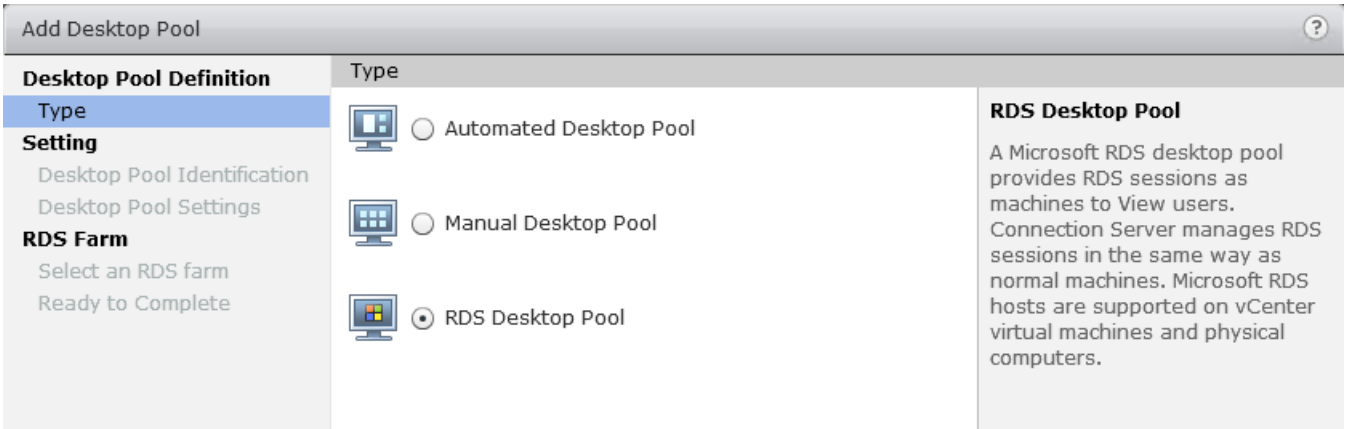
Create the Horizon 7 RDS Published Desktop Pool

To create the Horizon 7 RDS Published Desktop Pool, complete the following steps:

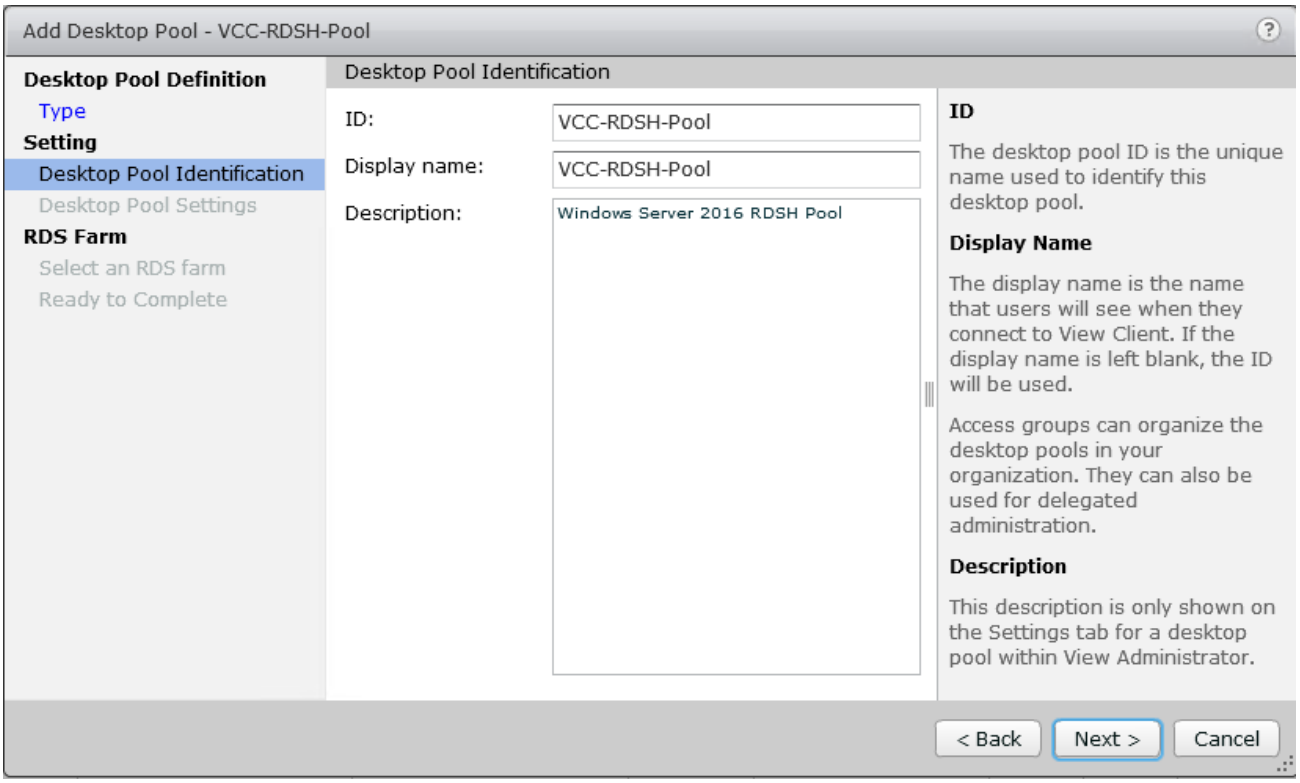
1. In the Horizon Administrator console, select Desktop Pools in the Catalog node of the Inventory pane.Click Add in the action pane.



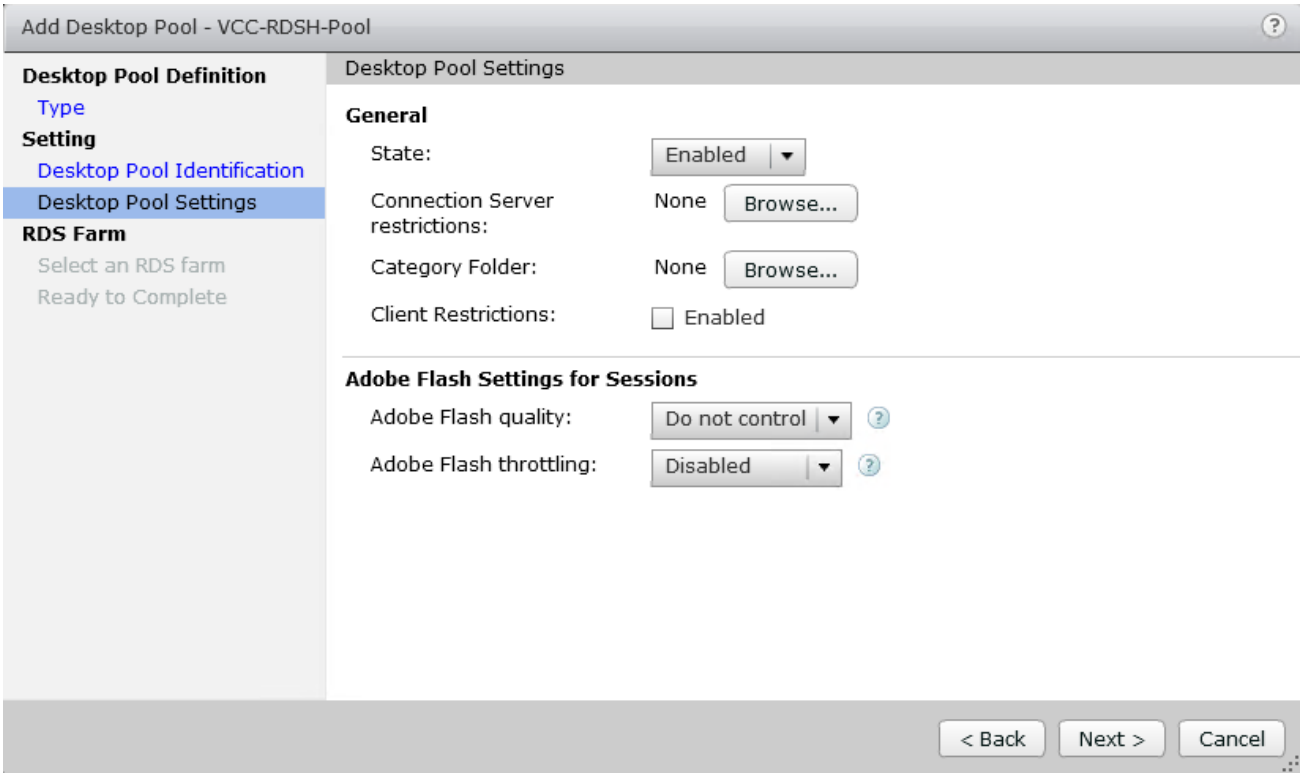
2. Select RDS Desktop pool, click Next.



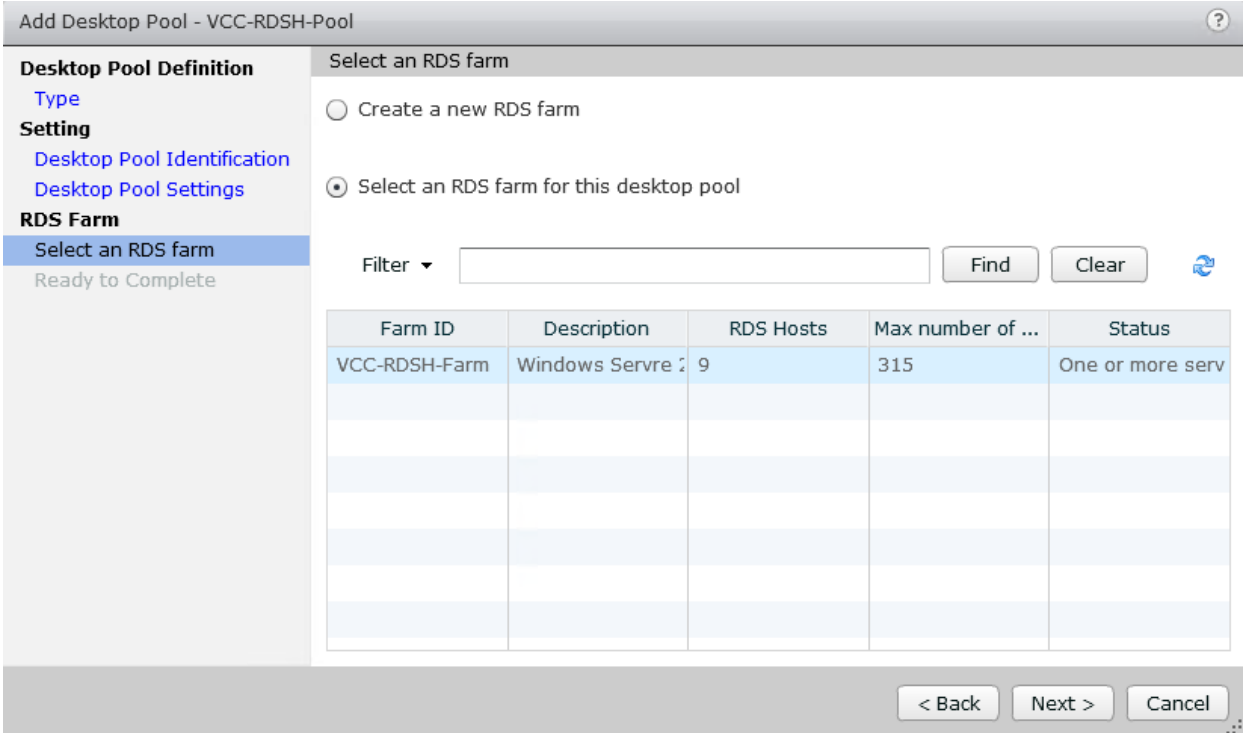
3. Enter Pool ID and Display name. Click Next.



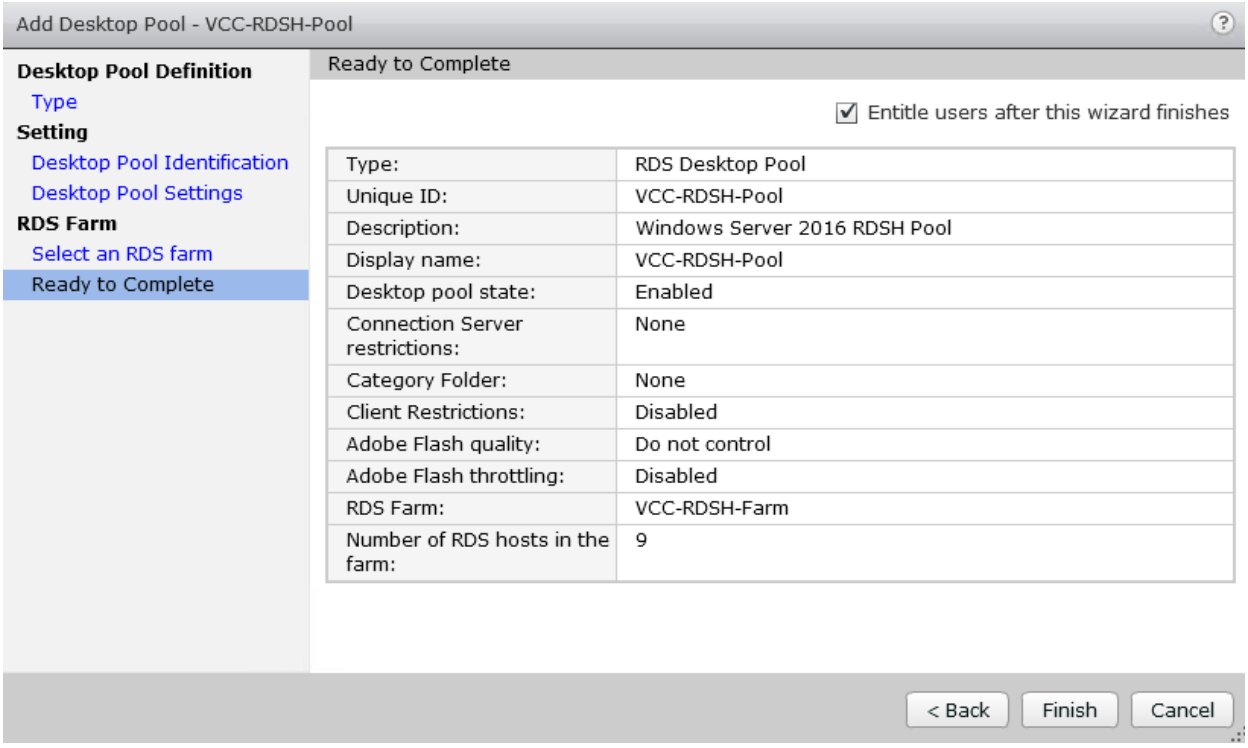
4. Accept the default settings on Desktop Pool Settings page. Click Next.



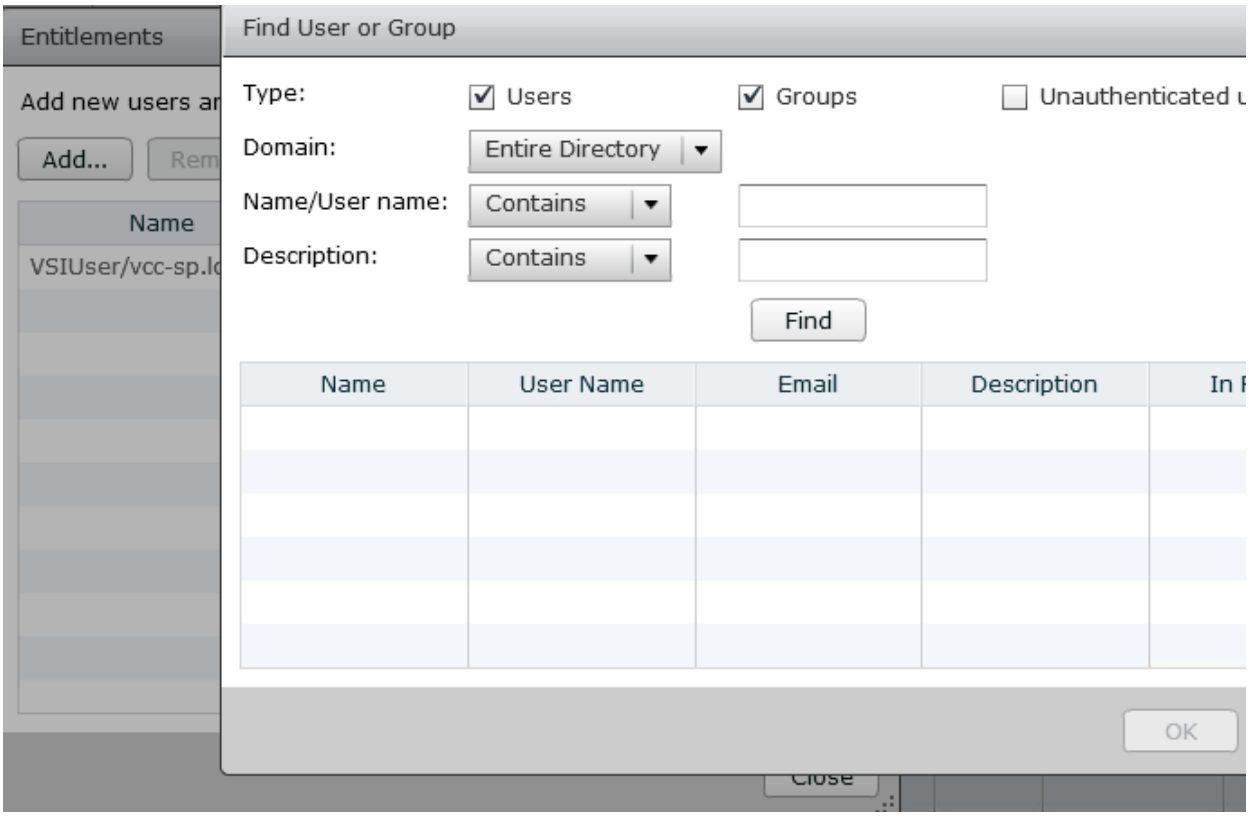
5. Click the “Select an RDS farm for this desktop pool” radio button. Select the farm created in the previous section. Click Next.



6. Review the pool settings. Select the checkbox “Entitle users after this wizard finishes” to authorize users for the newly create RDSH desktop pool. Click Finish.



7. Select the Users or Groups checkbox, use the search tools to locate the user or group to be authorized, highlight the user or group in the results box.



8. You now have a functional RDSH Farm and Desktop Pool with users identified who are authorized to utilize Horizon RDSH sessions.

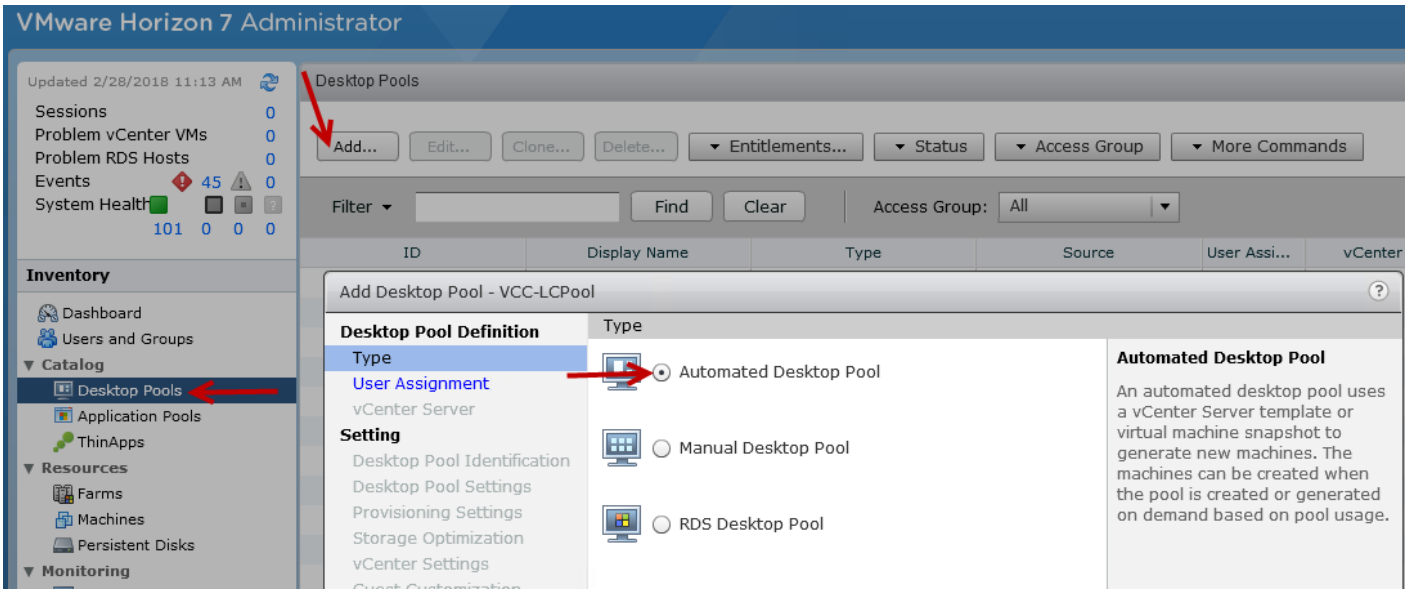
VMware Horizon Linked-Clone Windows 10 Desktop Pool Creation

To create a VMware Horizon linked-clone Windows 10 Desktop Pool, complete the following steps:

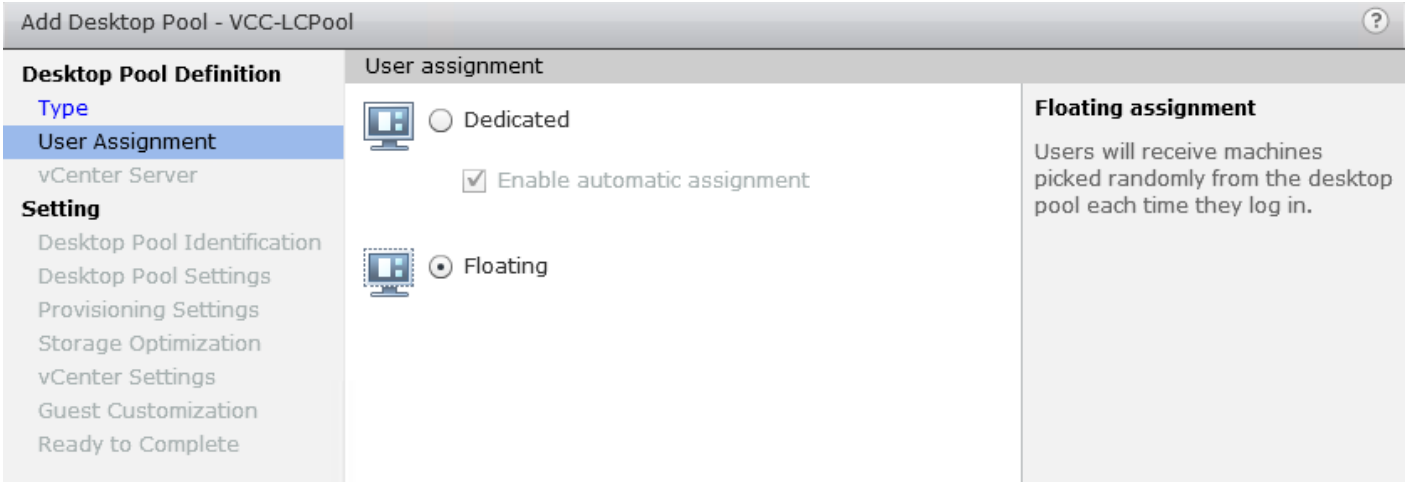
1. In Horizon Administrator console, select Desktop Pools in the Catalog node of the Inventory pane.

2. Click Add.

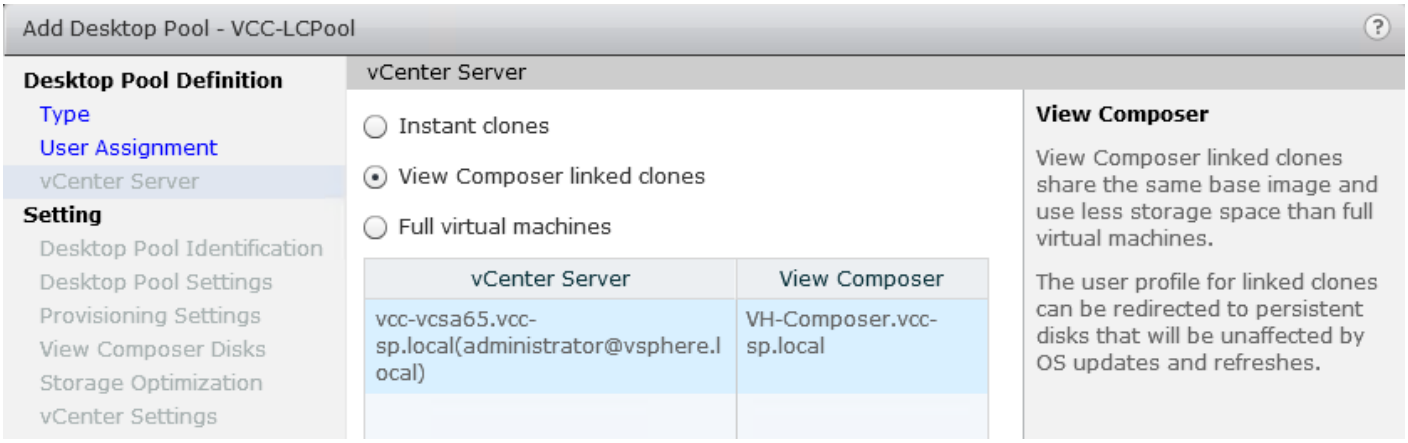
3. Select Type of Desktop Pool creation; we selected Automated Desktop Pool.



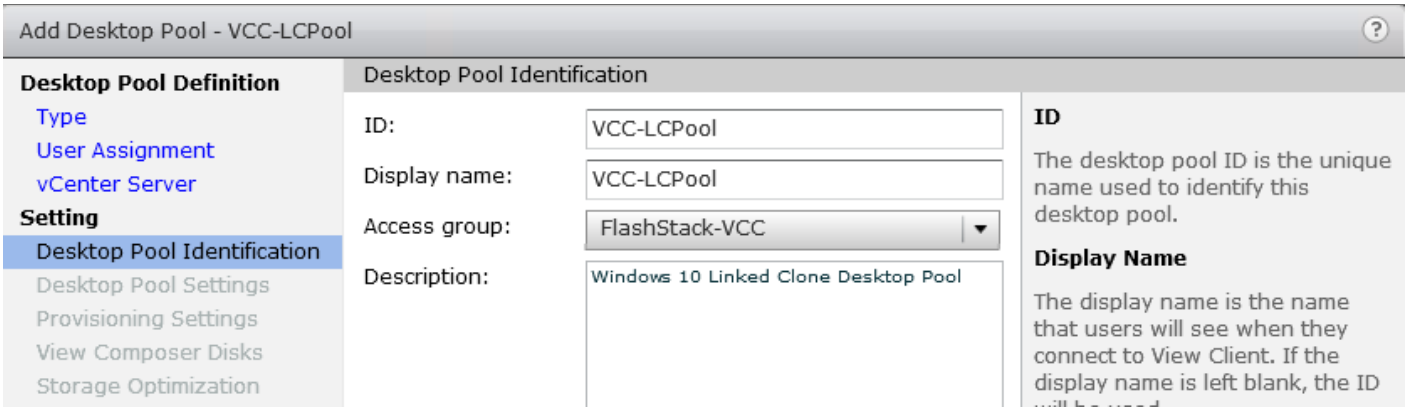
4. Select Floating or Dedicated user assignment. We created Floating assignment Pool.



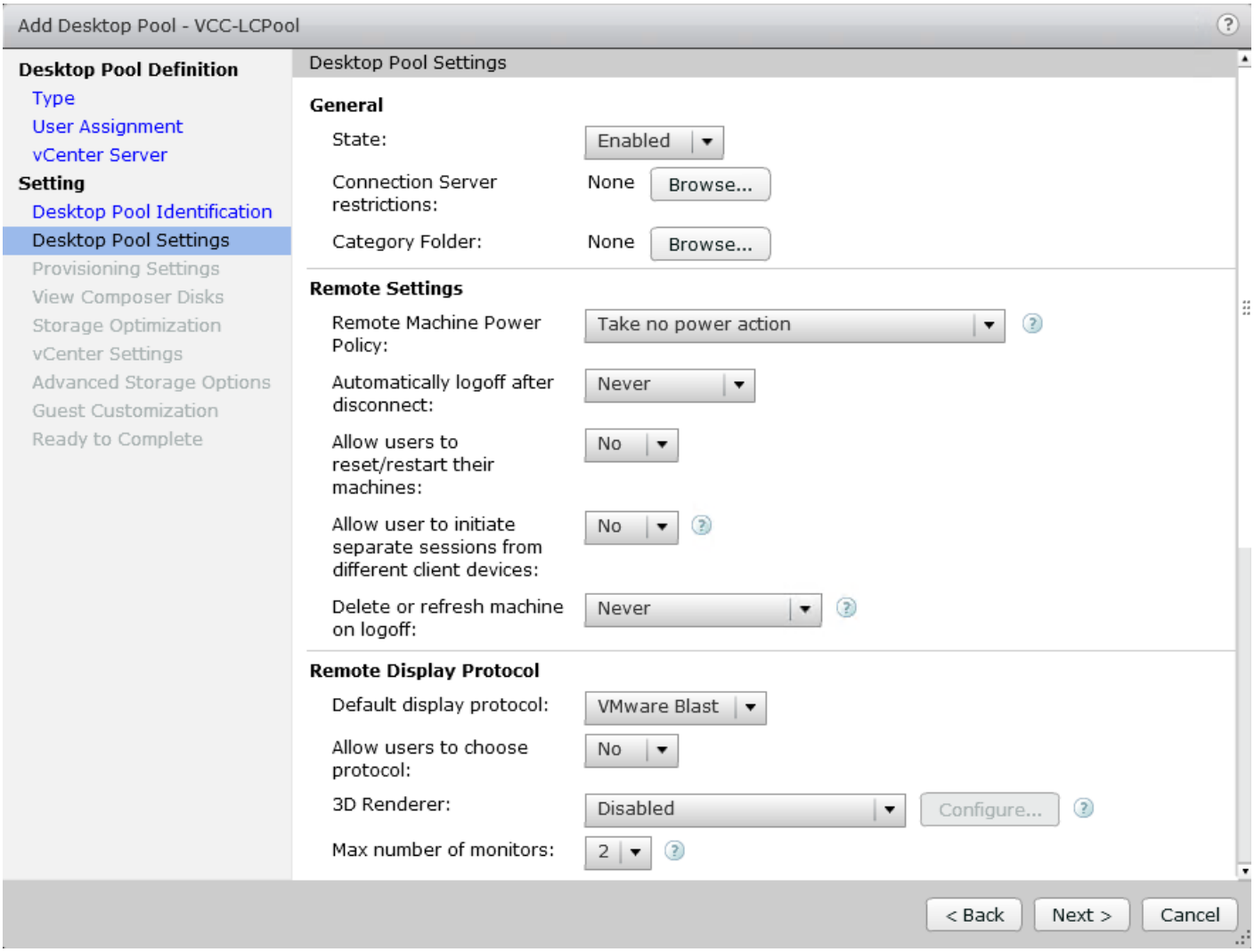
5. Select View Composer Linked Clones, select vCenter and View Composer for Linked-Clone virtual machine provisioning.



6. Enter pool identification details. Assign Access Group.

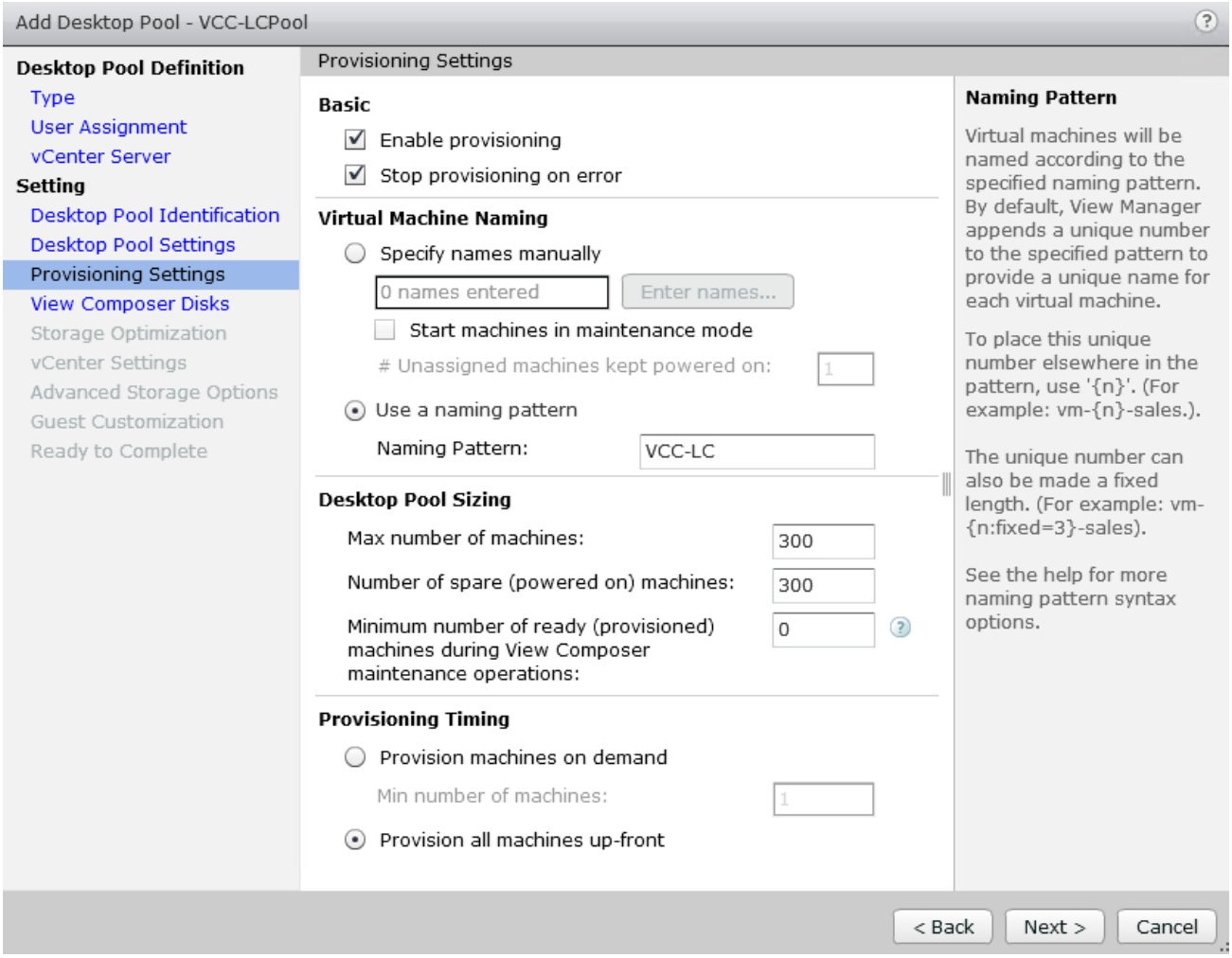


7. Select the settings for the Desktop Pool. Below is the sample configuration.

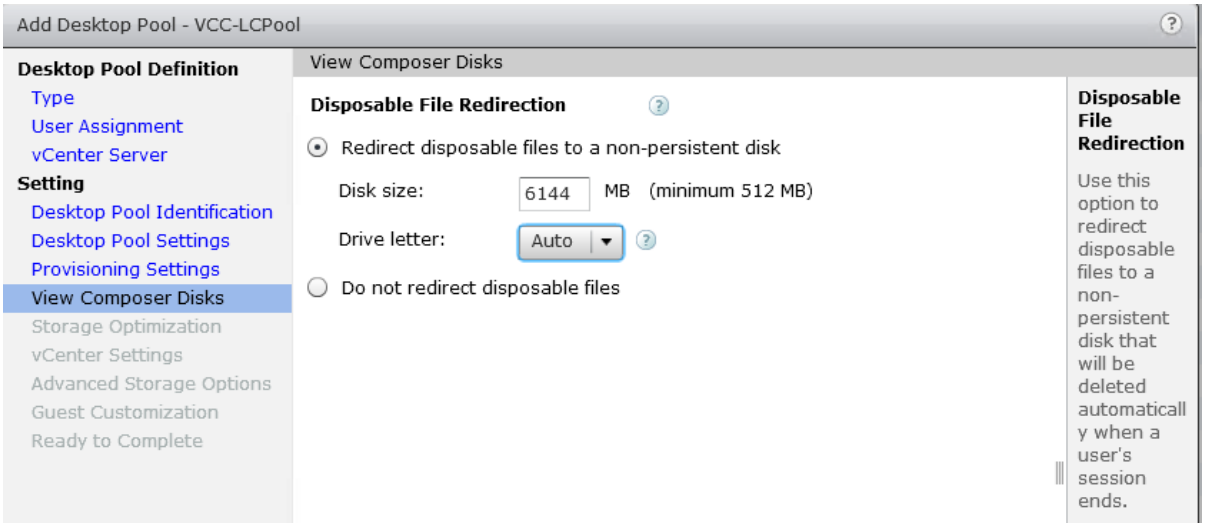


Make sure to scroll down in this dialogue to configure all options.

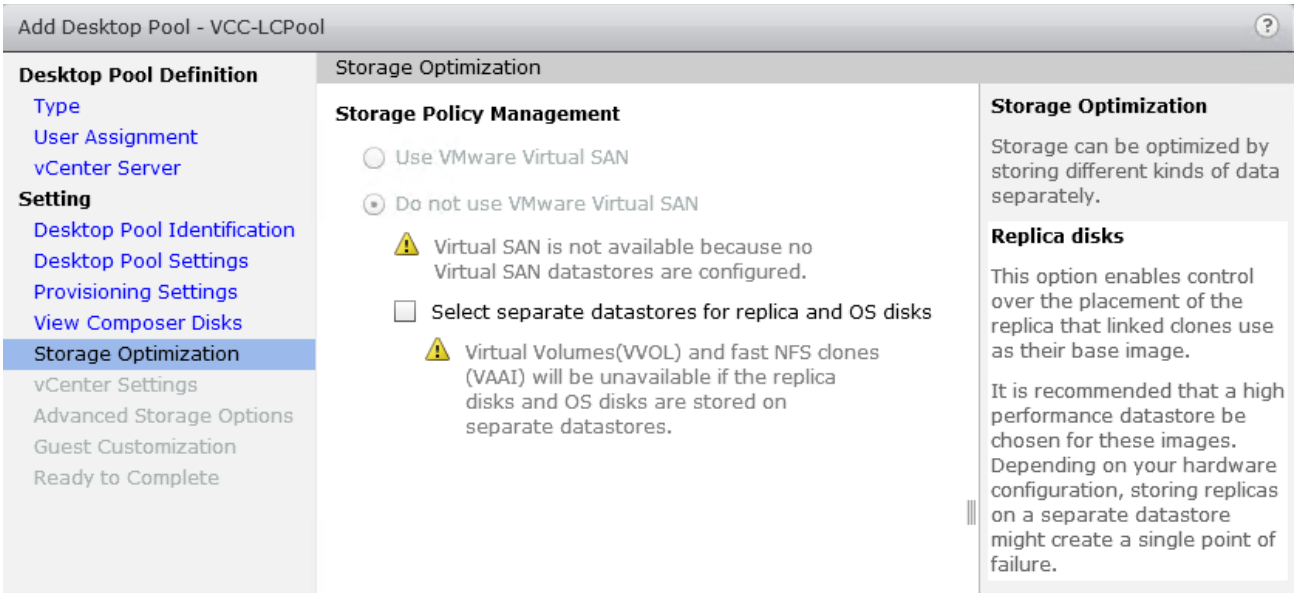
8. Select Provisioning Settings. For single server testing we provisioned 200 virtual machines.



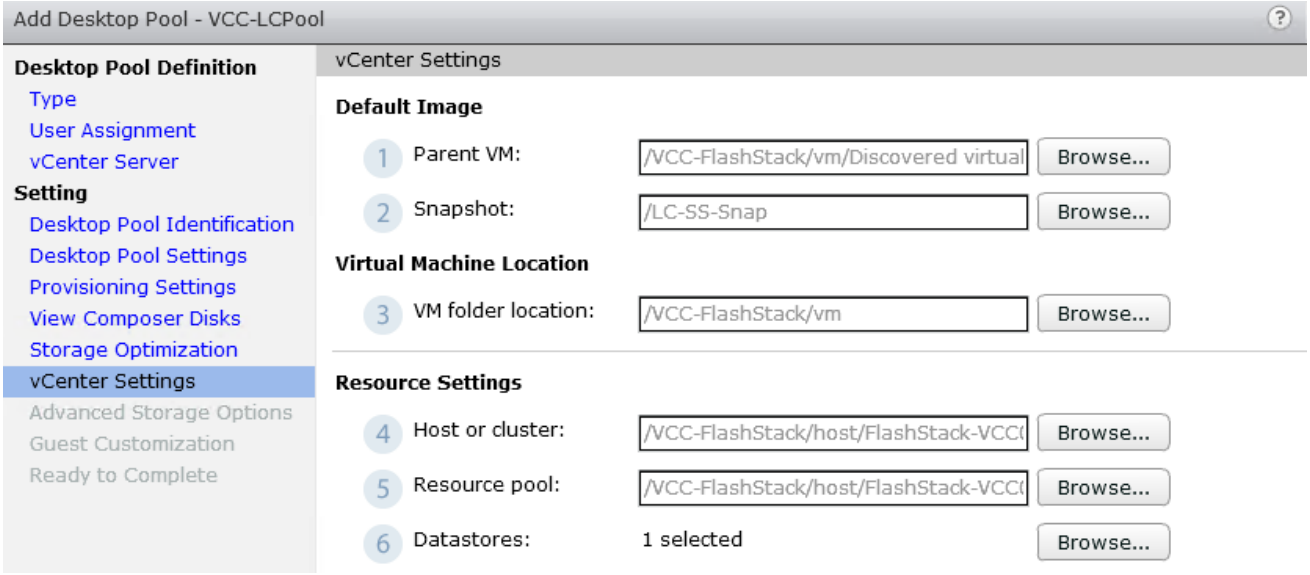
9. Select disposable (non-persistent) disk size in the View Composer Disks page.



10. Click Next on Storage Optimization.

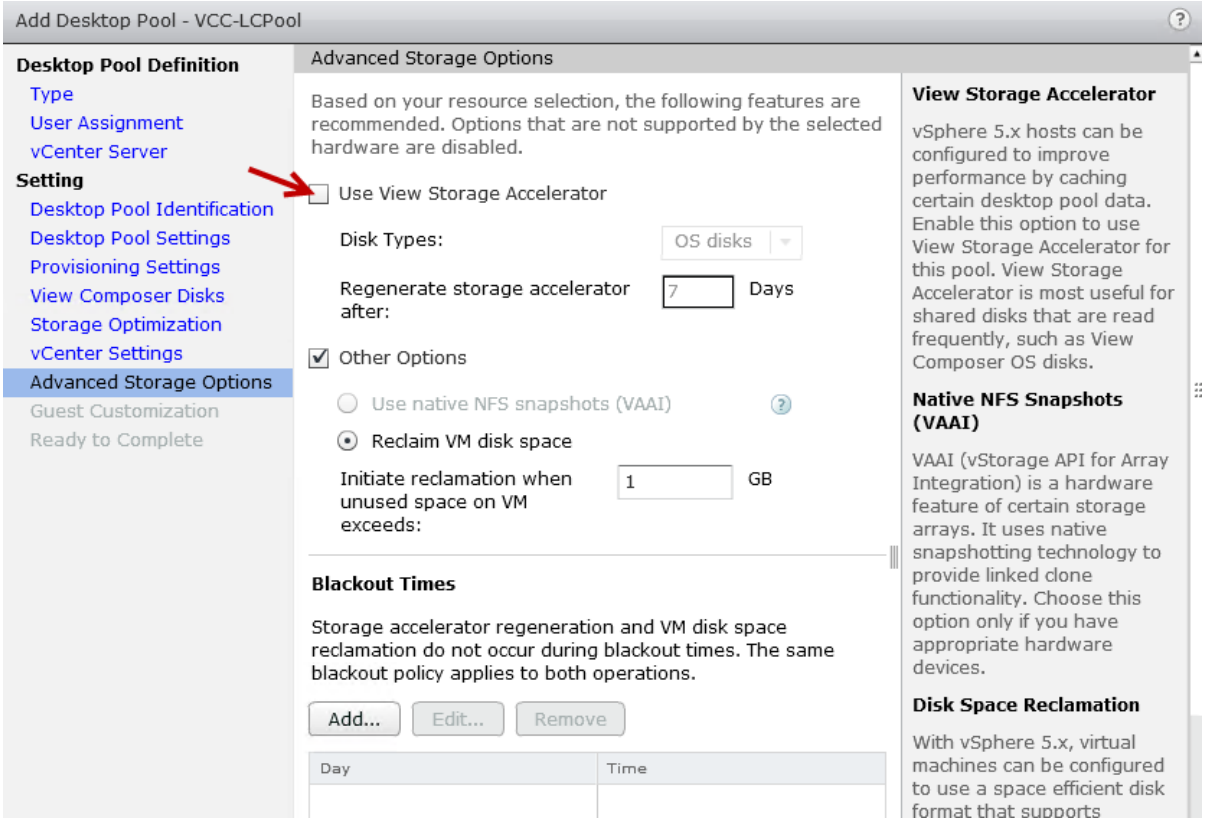


11. Select each of the required vCenter Settings by using the Browse button next to each field as intended for the desktop pool creation.

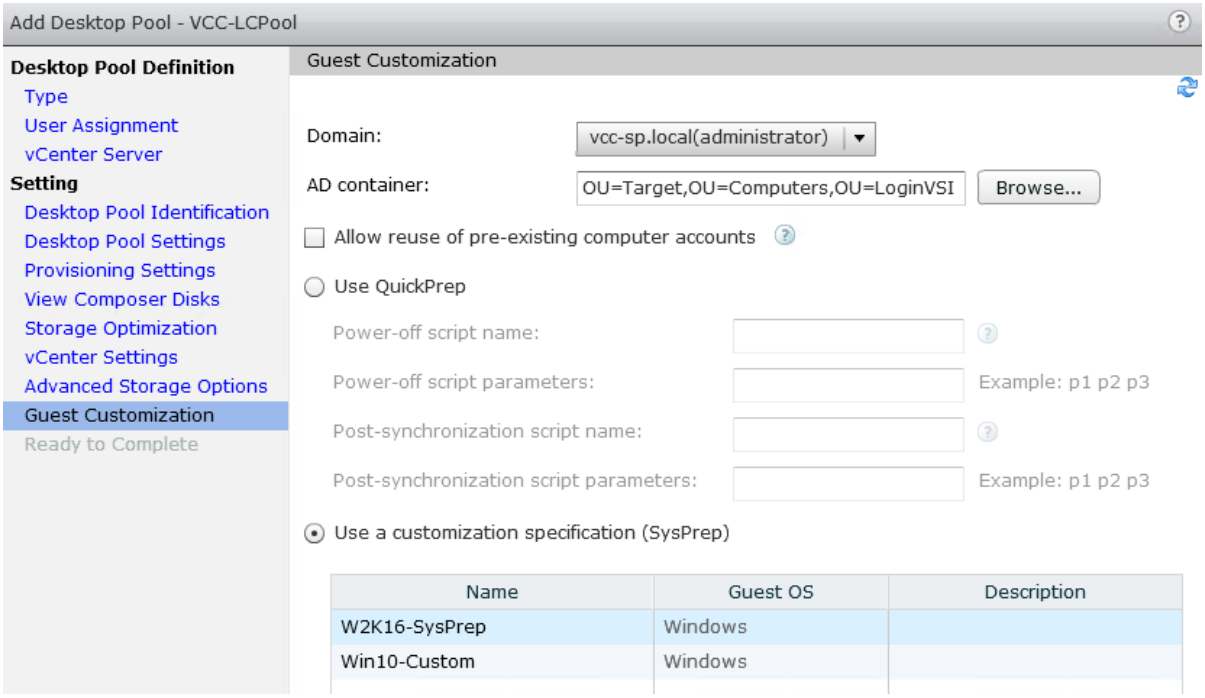


12. Set the Advanced Storage Options using the settings in the following screenshot.

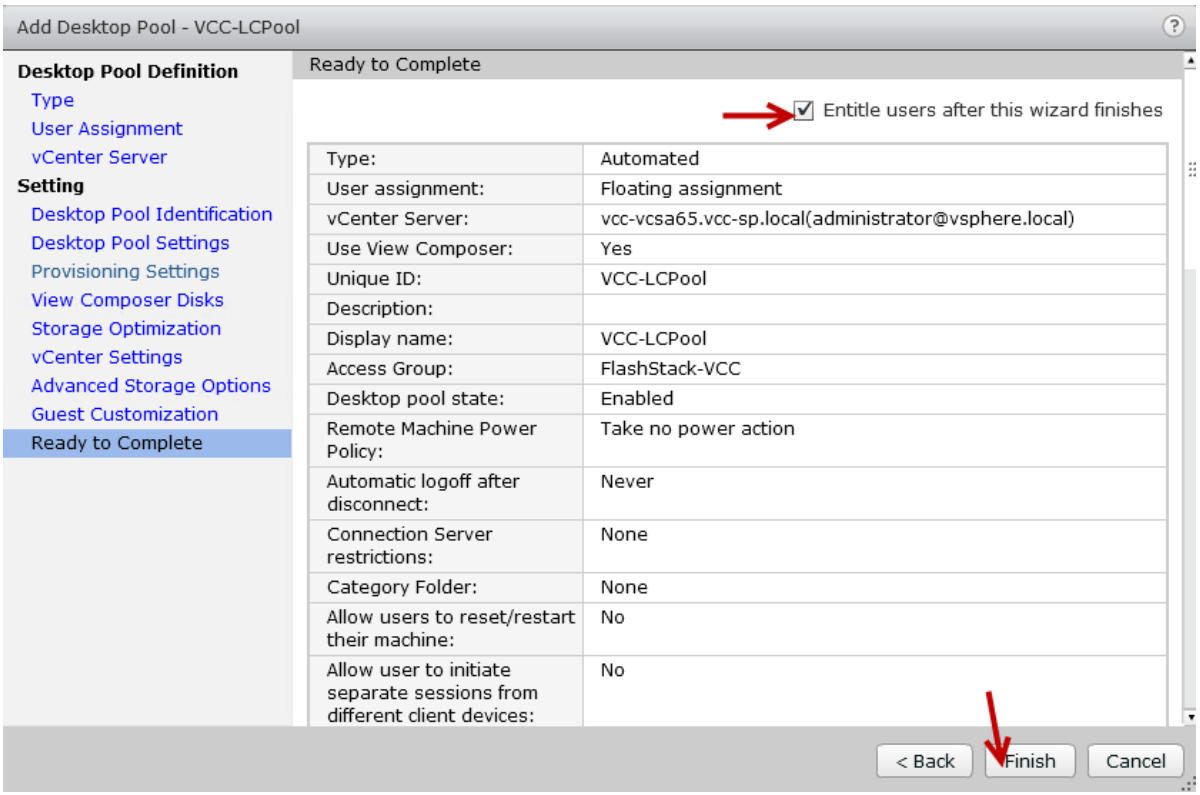
Do not enable the View Storage Accelerator.



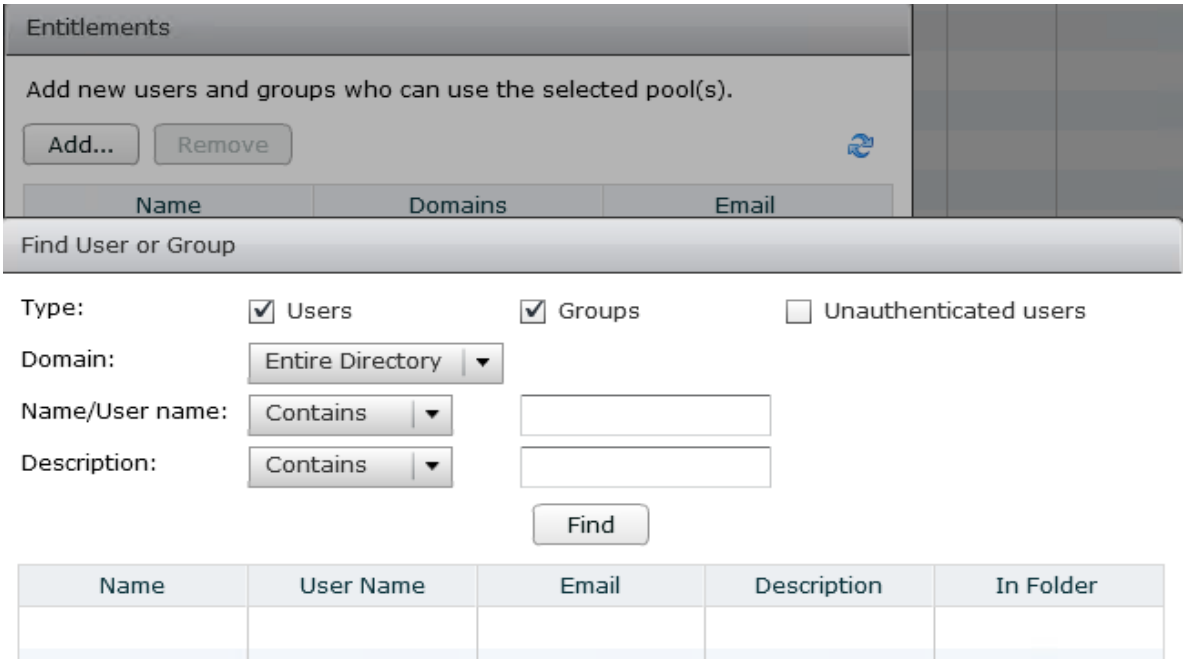
13. Select Guest Customization. Select the Active Directory domain; browse to the Active Directory (AD) container where the virtual machines will be provisioned and then select either the QuickPrep or Sysprep radio button. Highlight the Customization Spec previously prepared.



14. Select the checkbox “Entitle users after this wizard finishes” if you would like to authorize users as part of this process. Follow the instructions provided in the Create Horizon 7 RDS Desktop Pool to authorize users for the Linked Clone Pool. Click Finish to complete the Linked Clone Pool process.



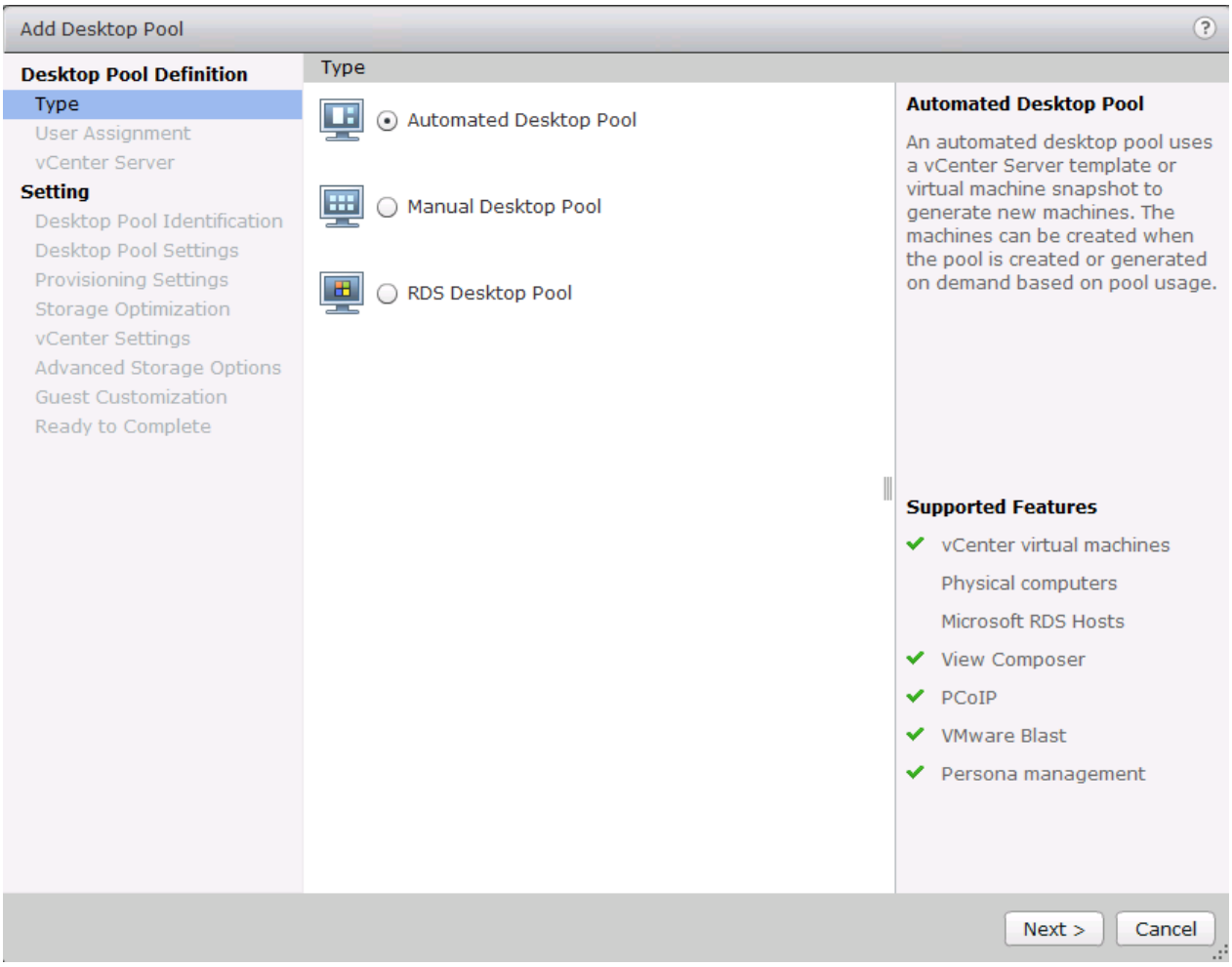
15. As part of the Entitlements, Add new users and groups who can use the selected pool(s).



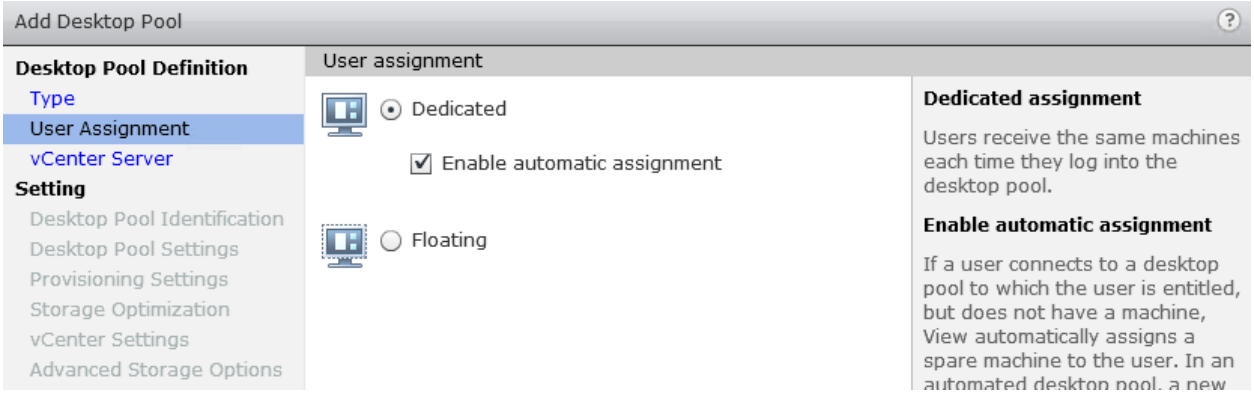
VMware Horizon Persistent Windows 10 Desktop Pool Creation

To create the VMware Horizon Persistent Windows 10 Desktop Pool, complete the following steps:

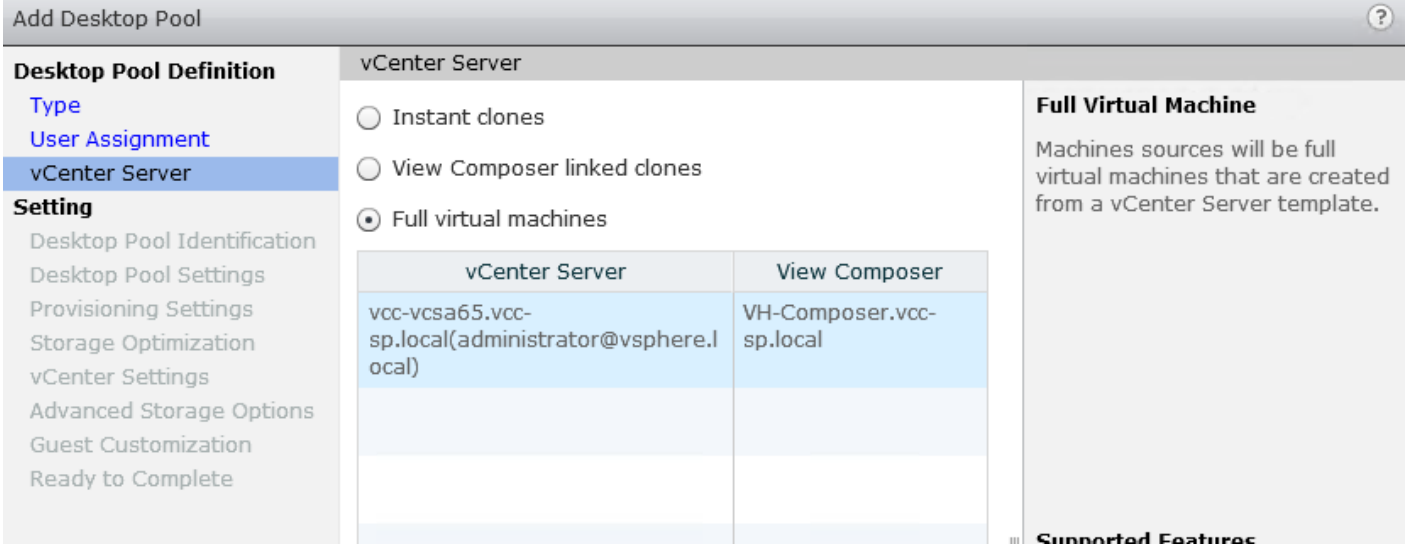
1. In Horizon Administrator console, select Desktop Pools in the Catalog node of the Inventory pane.
2. Click Add in the action pane.
3. Select assignment type for pool.
4. Click Next.



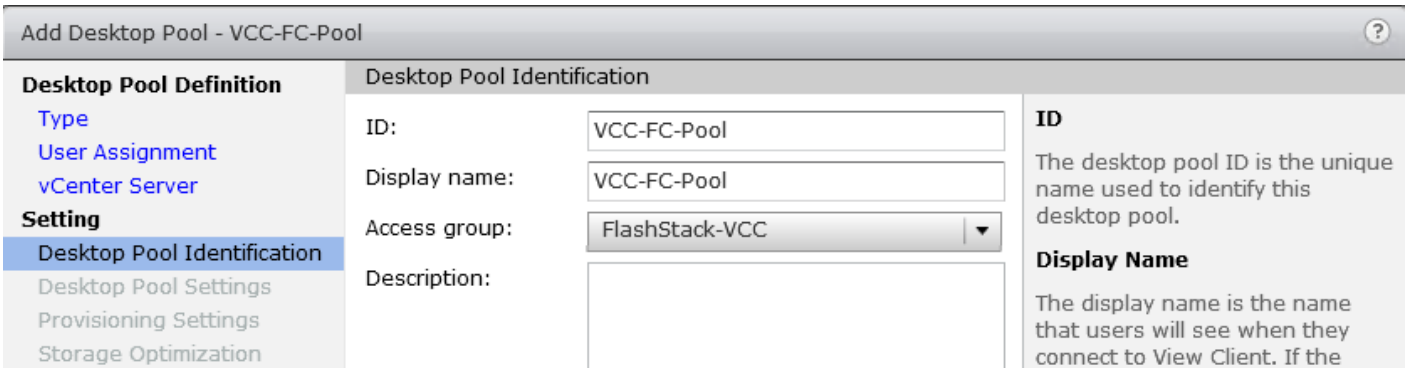
5. Select the Dedicated radio button. Select the Enable automatic assignment checkbox if desired.



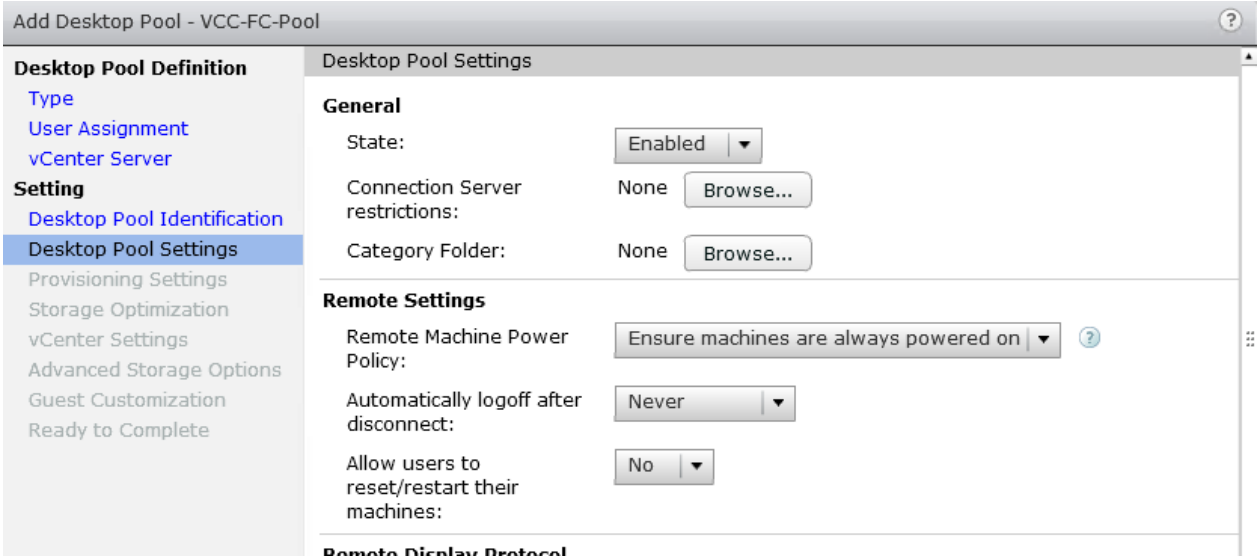
6. Select the Full Virtual Machines radio button and highlight your vCenter and Composer.



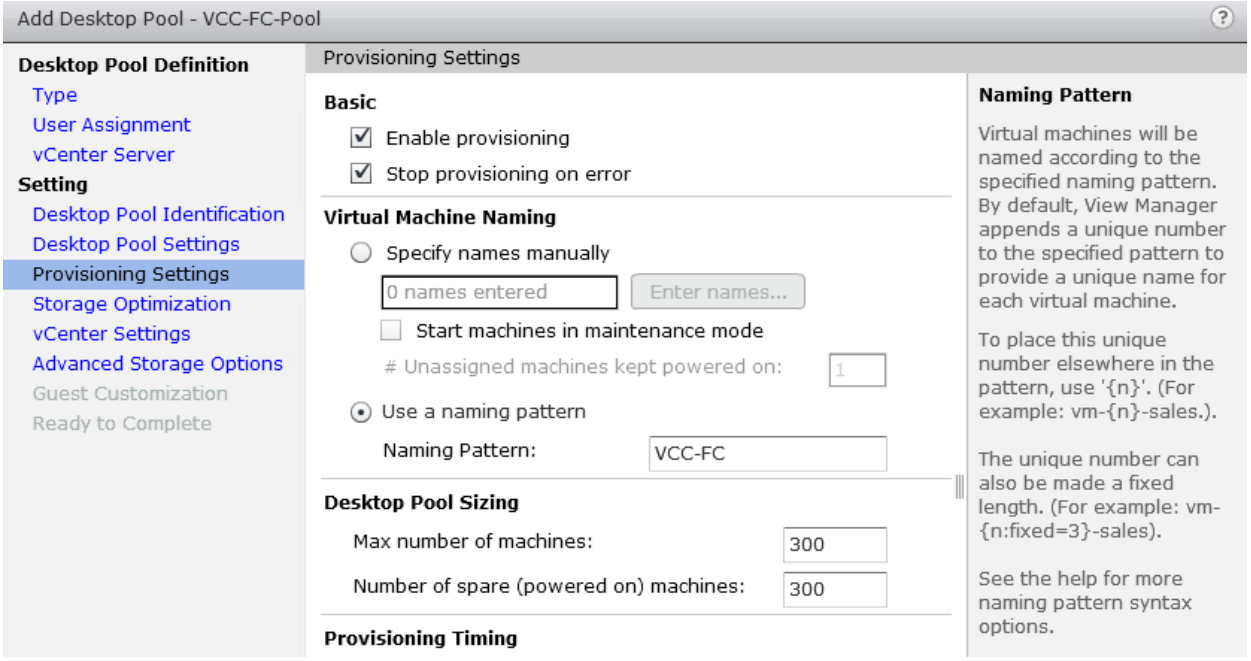
7. Enter the pool identification details.



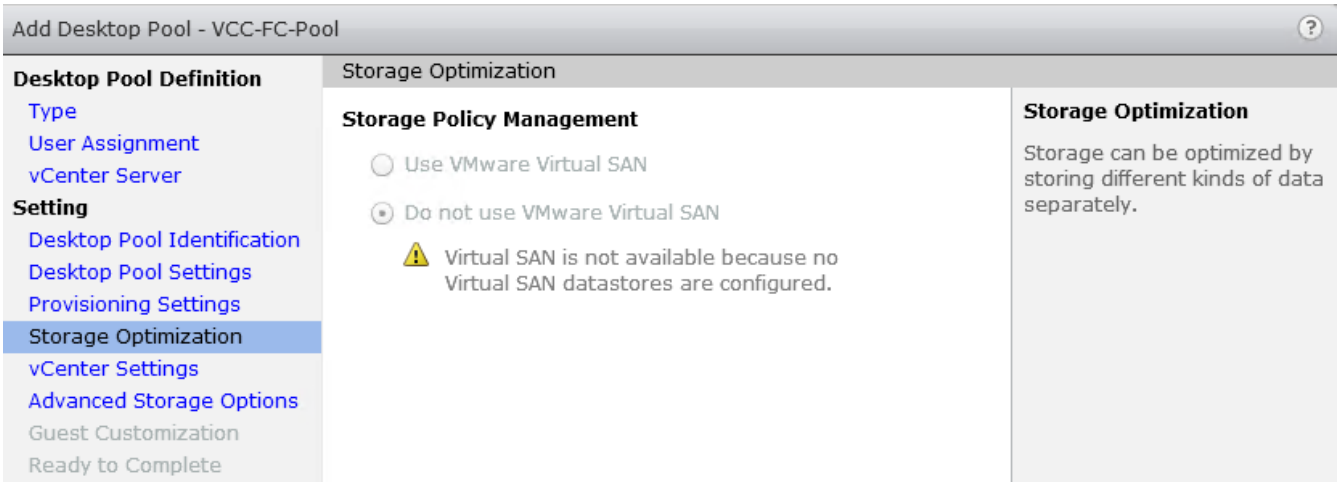
8. Select Desktop Pool settings.



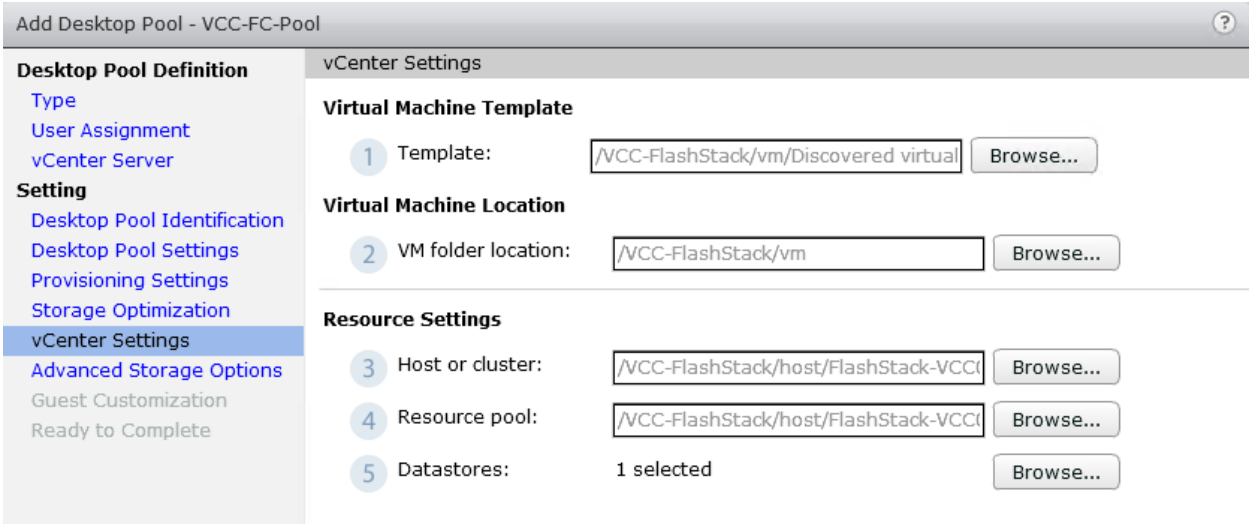
9. Select the provisioning settings to meet your requirements.



10. Click Next on the Storage Optimization page.

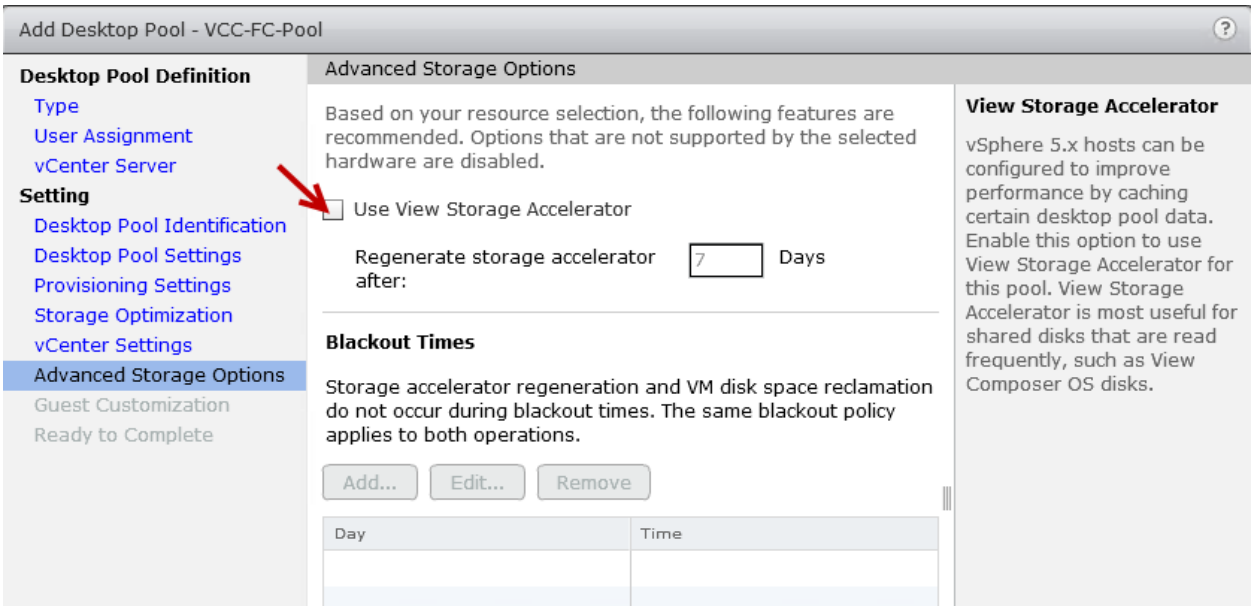


11. Select the vCenter Settings by clicking Browse for each option.

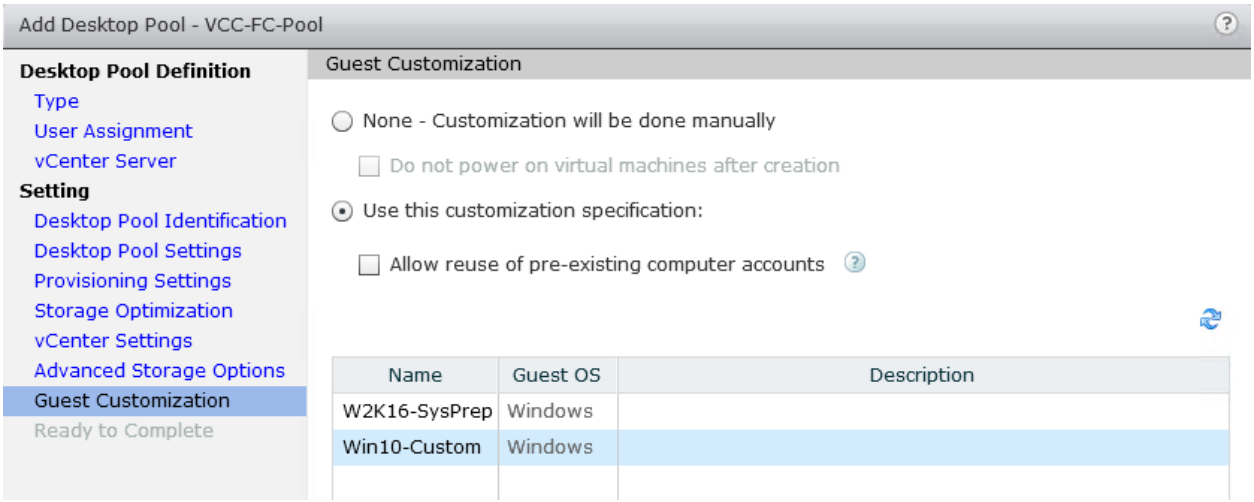


12. Select Advance Storage Options and enable the View Storage Accelerator.

Do not select View Storage Accelerator.

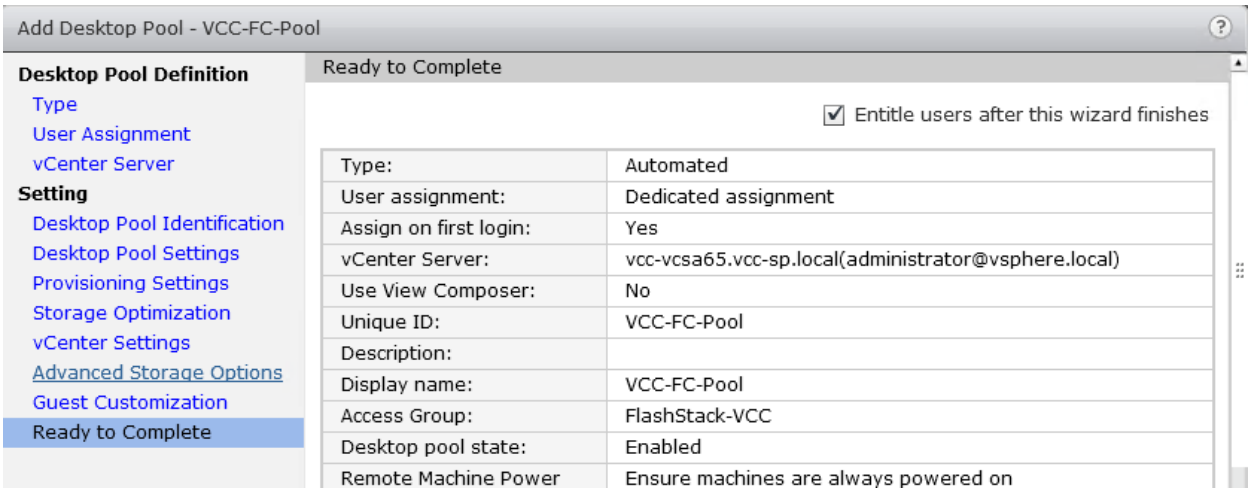


13. Select Guest optimization settings.



14. Review the summary of the pool you are creating.

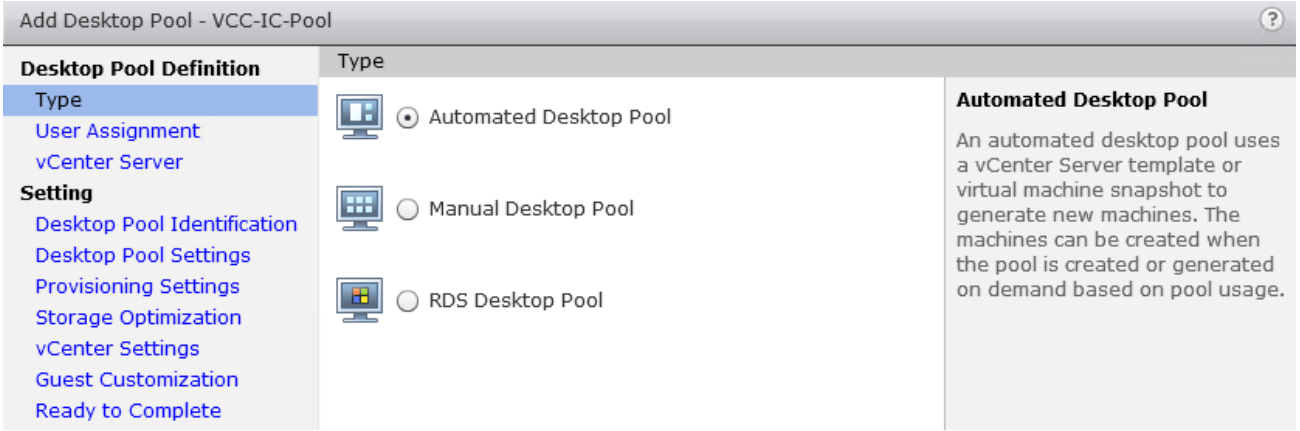
15. Select the checkbox “Entitle users after pool creation wizard completion” to authorize users for the pool.



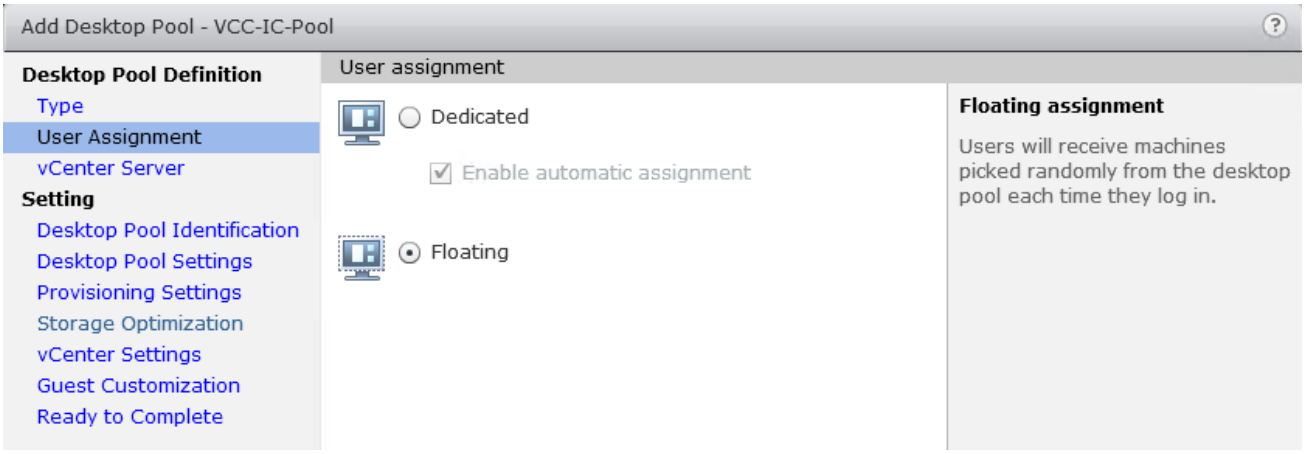
VMware Horizon Instant-Clone Windows 10 Desktop Pool Creation

To create the VMware Horizon Instant-Clone Windows 10 Desktop Pool, complete the following steps:

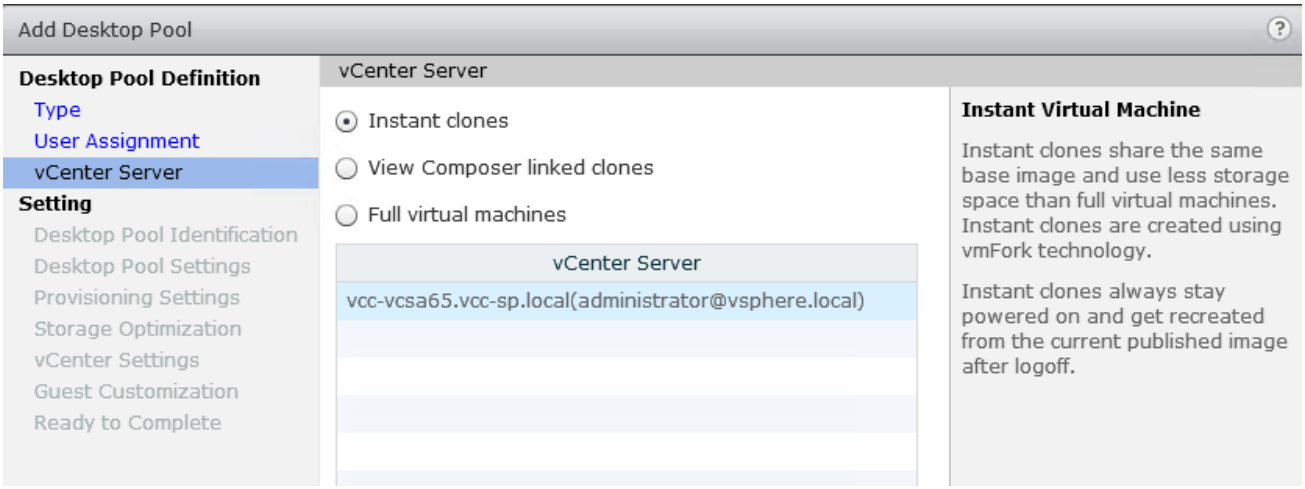
1. In Horizon Administrator console, select Desktop Pools in the Catalog node of the Inventory pane.
2. Click Add in the action pane.
3. Select Automated assignment type for pool.
4. Click Next.



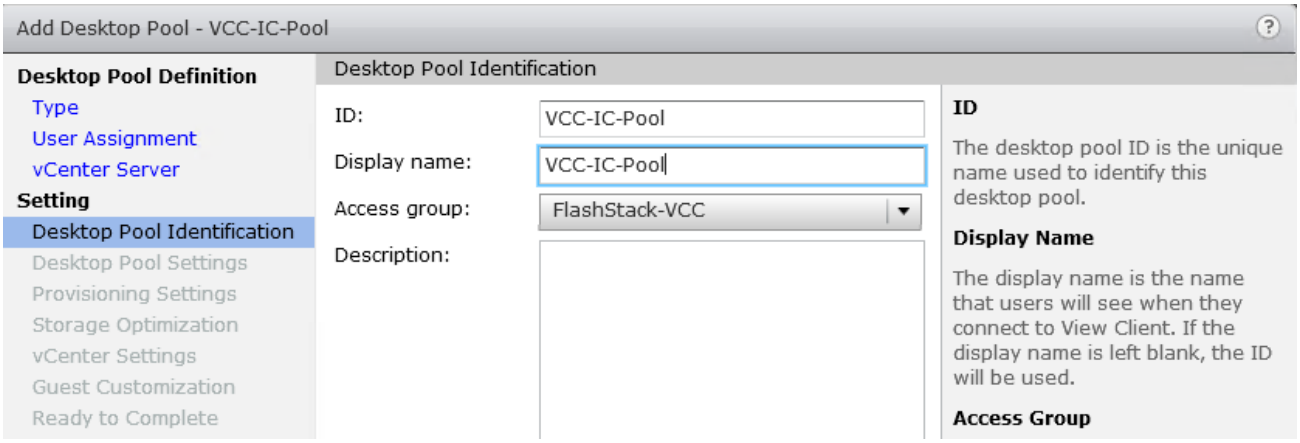
5. Select Floating or Dedicate user assignment.
6. Click Next.



7. Select Instant Clones, highlight your vCenter server, then click Next.

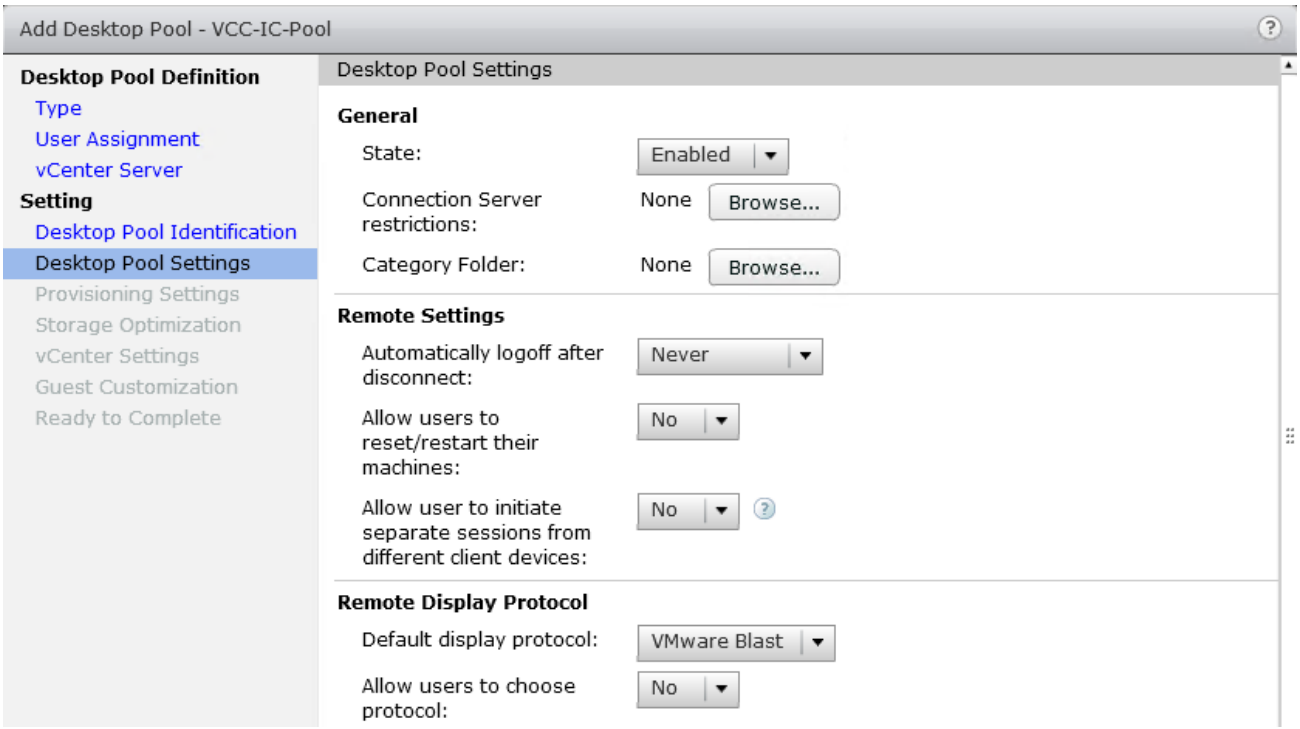


8. Enter the pool identification details.

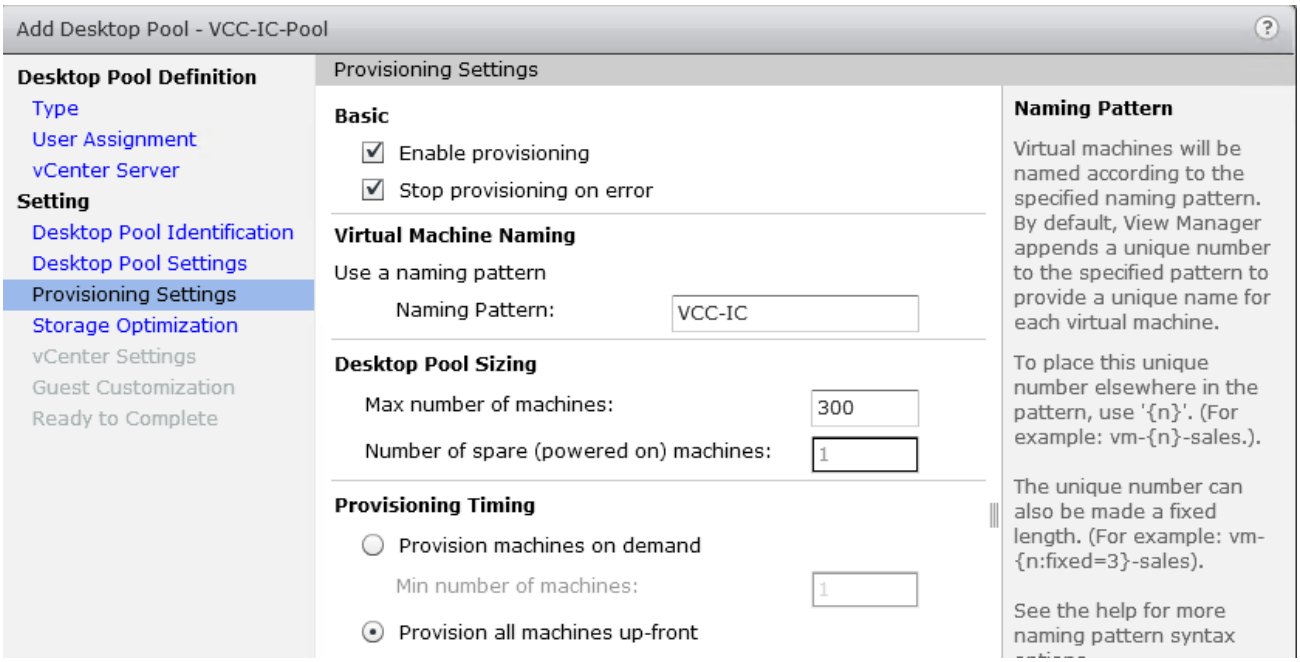


9. Select Desktop Pool settings.

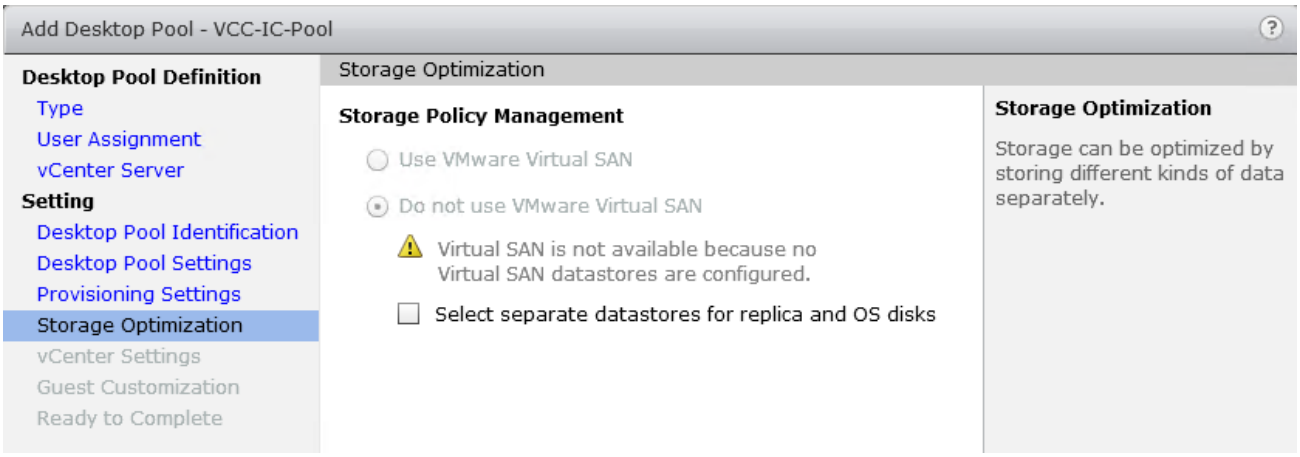
Make sure to scroll down to choose the Acrobat Flash settings.



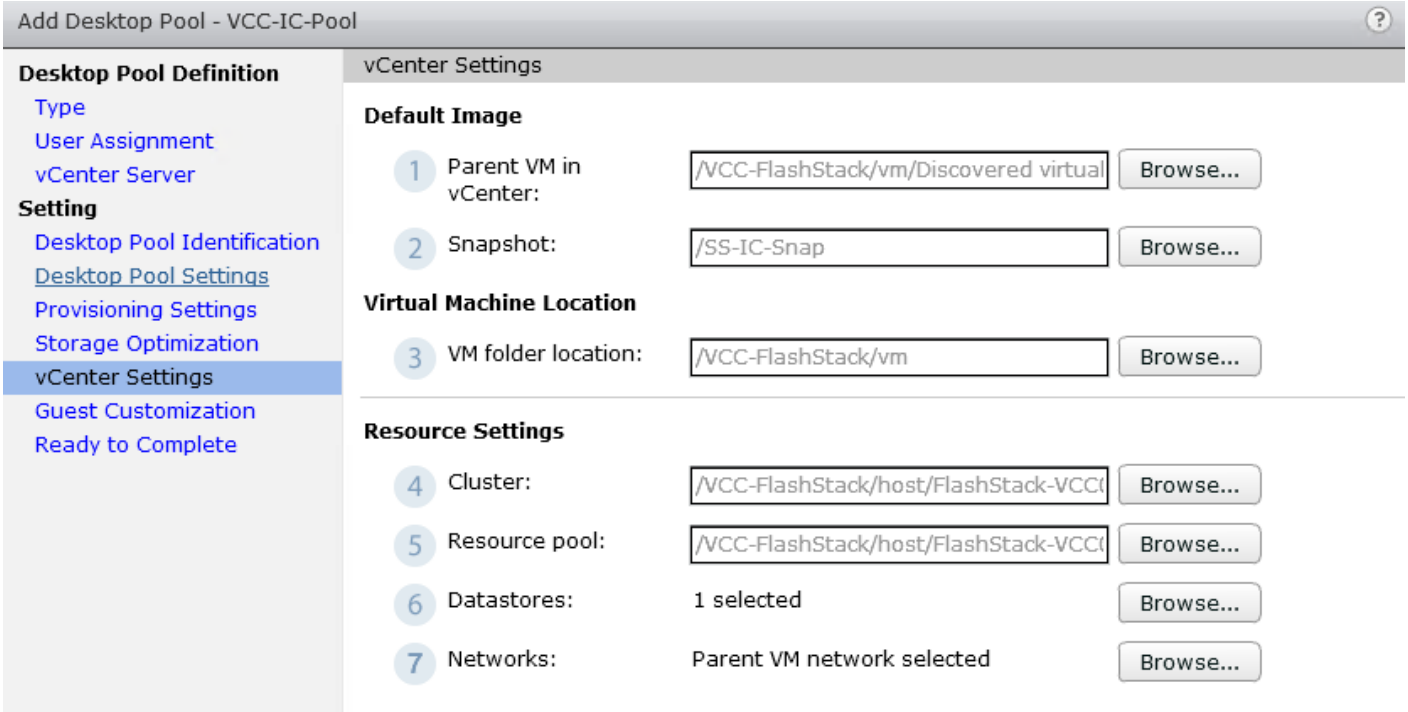
10. Select the Provisioning Settings.



11. Click Next on the Storage Optimization page.

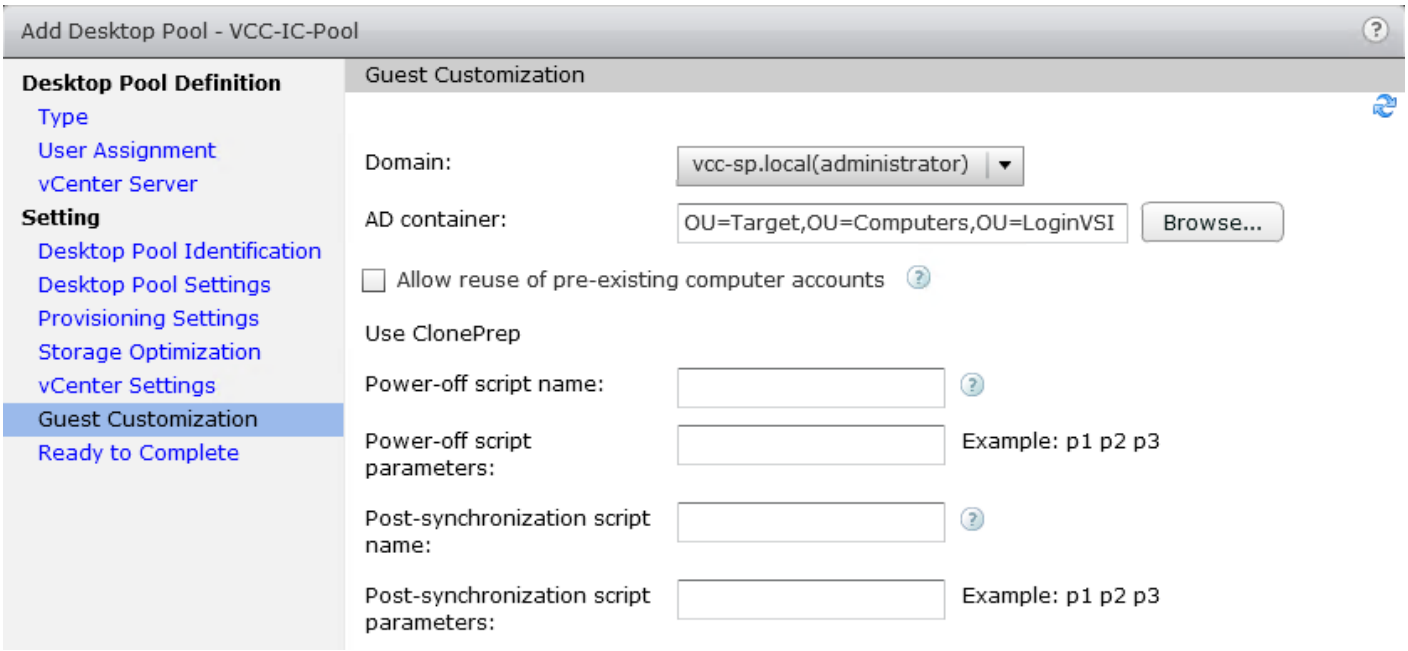


12. Select the vCenter Settings to deploy the Instant-Clone Desktop Pool.



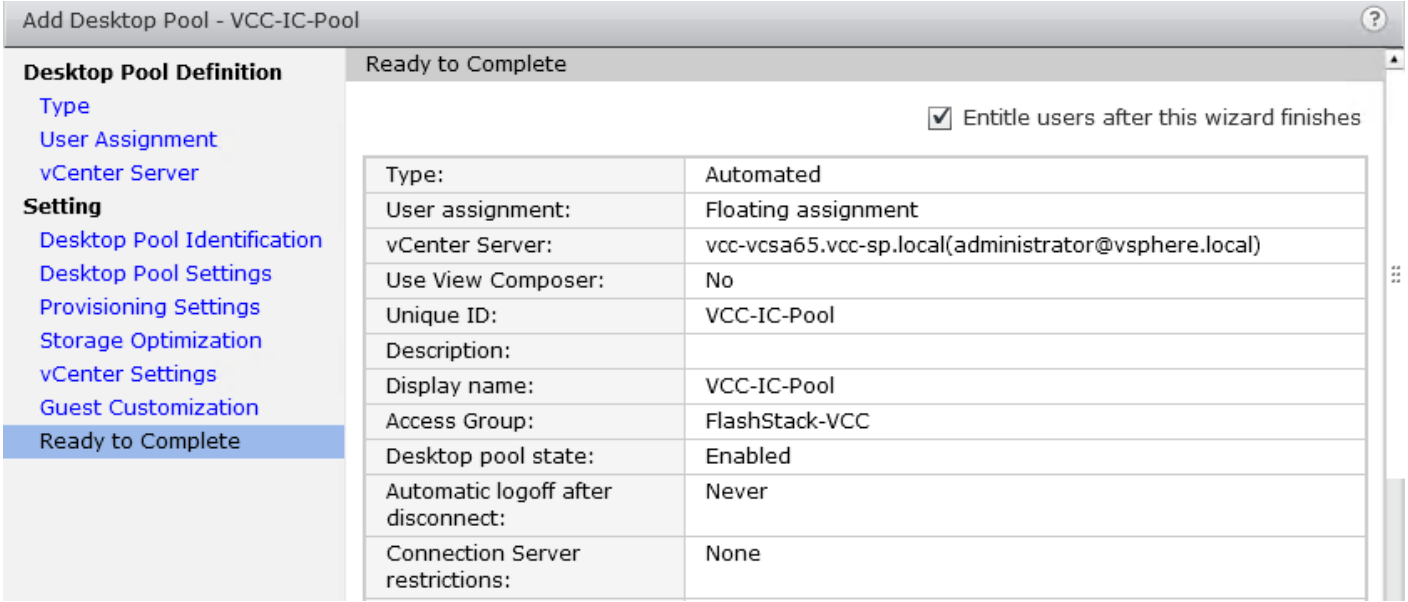
13. Select Guest Customization.

14. Browse to your Active Directory Domain and select the AD container into which you want your Instant Clone machines provisioned.



15. Review the summary of the pool configuration.

16. Select the checkbox “Entitle users after this wizard finishes” to authorize users or groups for the new pool.



17. Follow the instructions provided in the Create Horizon 7 RDS Desktop Pool to authorize users for the Linked Clone Pool.

Configuring User Profile Management

Profile management provides an easy, reliable, and high-performance way to manage user personalization settings in virtualized or physical Windows environments. It requires minimal infrastructure and

administration, and provides users with fast logons and logoffs. A Windows user profile is a collection of folders, files, registry settings, and configuration settings that define the environment for a user who logs on with a particular user account. These settings may be customizable by the user, depending on the administrative configuration. Examples of settings that can be customized are:

- Desktop settings such as wallpaper and screen saver
- Shortcuts and Start menu setting
- Internet Explorer Favorites and Home Page
- Microsoft Outlook signature
- Printers

Some user settings and data can be redirected by means of folder redirection. However, if folder redirection is not used these settings are stored within the user profile.

The first stage in planning a profile management deployment is to decide on a set of policy settings that together form a suitable configuration for your environment and users. The automatic configuration feature simplifies some of this decision-making for VMware Horizon desktop deployments. We configured Micorsoft Roaming Profile based User Profile Management and policies for this CVD's RDS and VDI users (for testing purposes) are shown below.

Install and Configure NVIDIA P6 Card

This section focuses on installing and configuring the NVIDIA P6 cards with the Cisco UCS B200 M5 servers to deploy vGPU enabled virtual desktops.

Physical Installation of P6 Card into Cisco UCS B200 M5 Server

The NVIDIA P6 graphics processing unit (GPU) card provides graphics and computing capabilities to the server. There are two supported versions of the NVIDIA P6 GPU card:

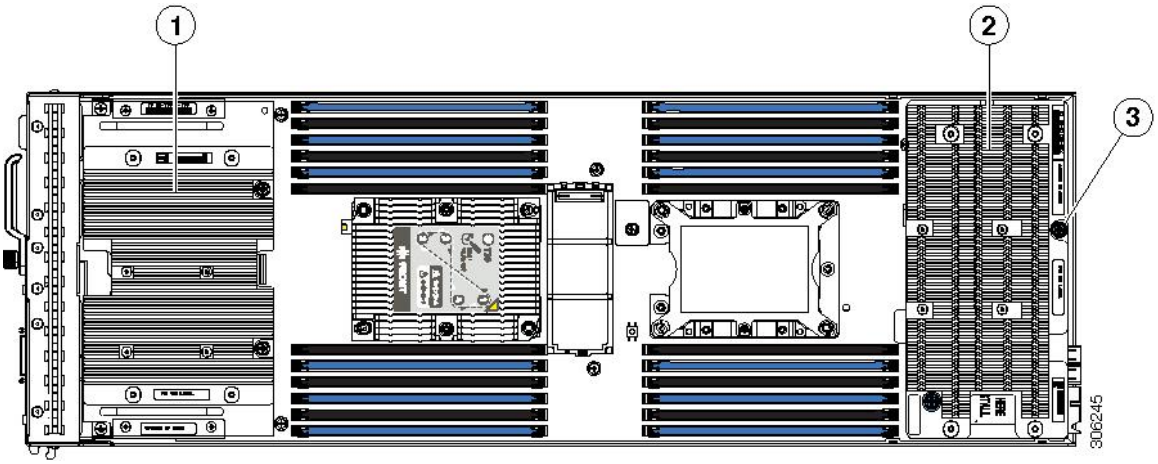
- UCSB-GPU-P6-F can be installed only in the front mezzanine slot of the server

No front mezzanine cards can be installed when the server has CPUs greater than 165 W.

- UCSB-GPU-P6-R can be installed only in the rear mezzanine slot (slot 2) of the server.

Figure 46 shows the installed NVIDIA P6 GPU in the front and rear mezzanine slots.

Figure 46 NVIDIA GPU Installed in the Front and Rear Mezzanine Slots

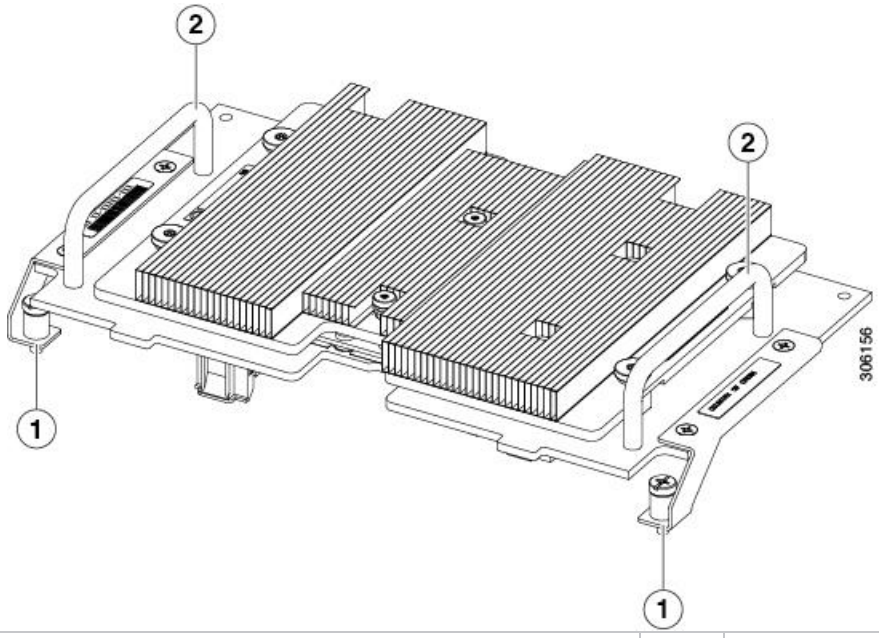


1	Front GPU	2	Rear GPU
3	Custom standoff screw	-	

Installing an NVIDIA GPU Card in the Front of the Server

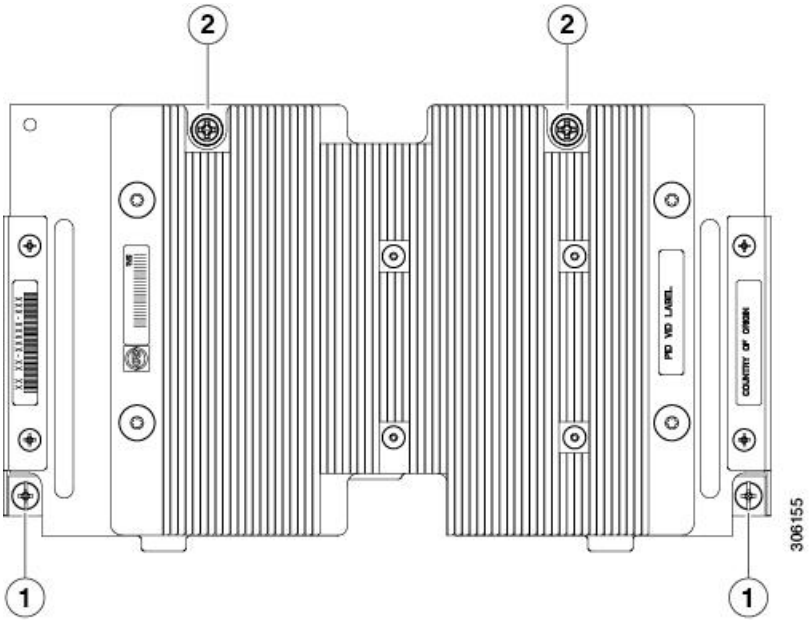
Figure 47 shows the front NVIDIA P6 GPU (UCSB-GPU-P6-F).

Figure 47 NVIDIA P6 GPU That Installs in the Front of the Server



1	Leg with thumb screw that attaches to the server motherboard at the front	2	Handle to press down on when installing the GPU

Figure 48 Top Down View of the NVIDIA P6 GPU for the Front of the Server



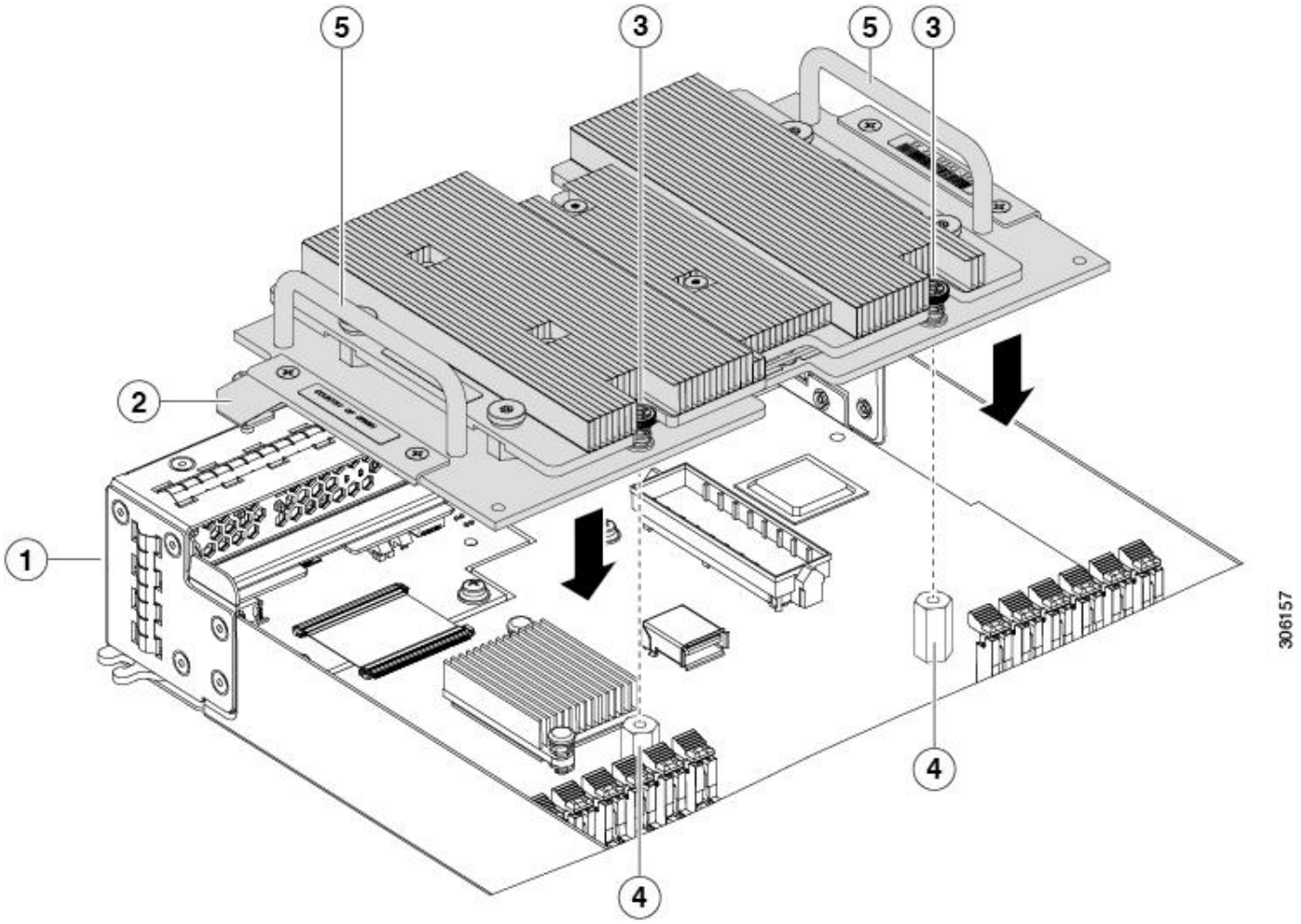
1	Leg with thumb screw that attaches to the server motherboard	2	Thumb screw that attaches to a standoff below

To install the NVIDIA GPU, complete the following steps:

Before installing the NVIDIA P6 GPU (UCSB-GPU-P6-F) in the front mezzanine slot you need to upgrade the Cisco UCS domain that the GPU will be installed into to a version of Cisco UCS Manager that supports this card. Refer to the latest version of the Release Notes for Cisco UCS Software at the following URL for information about supported hardware: http://www.cisco.com/c/en/us/support/servers-unified-computing/ucs-manager/products-release-notes-list.html. Remove the front mezzanine storage module if it is present. You cannot use the storage module in the front mezzanine slot when the NVIDIA P6 GPU is installed in the front of the server.

1. Position the GPU in the correct orientation to the front of the server (callout 1) as shown in Figure 49.
2. Install the GPU into the server. Press down on the handles (callout 5) to firmly secure the GPU.
3. Tighten the thumb screws (callout 3) at the back of the GPU with the standoffs (callout 4) on the mother-board.
4. Tighten the thumb screws on the legs (callout 2) to the motherboard.
5. Install the drive blanking panels.

Figure 49 Installing the NVIDIA GPU in the Front of the Server

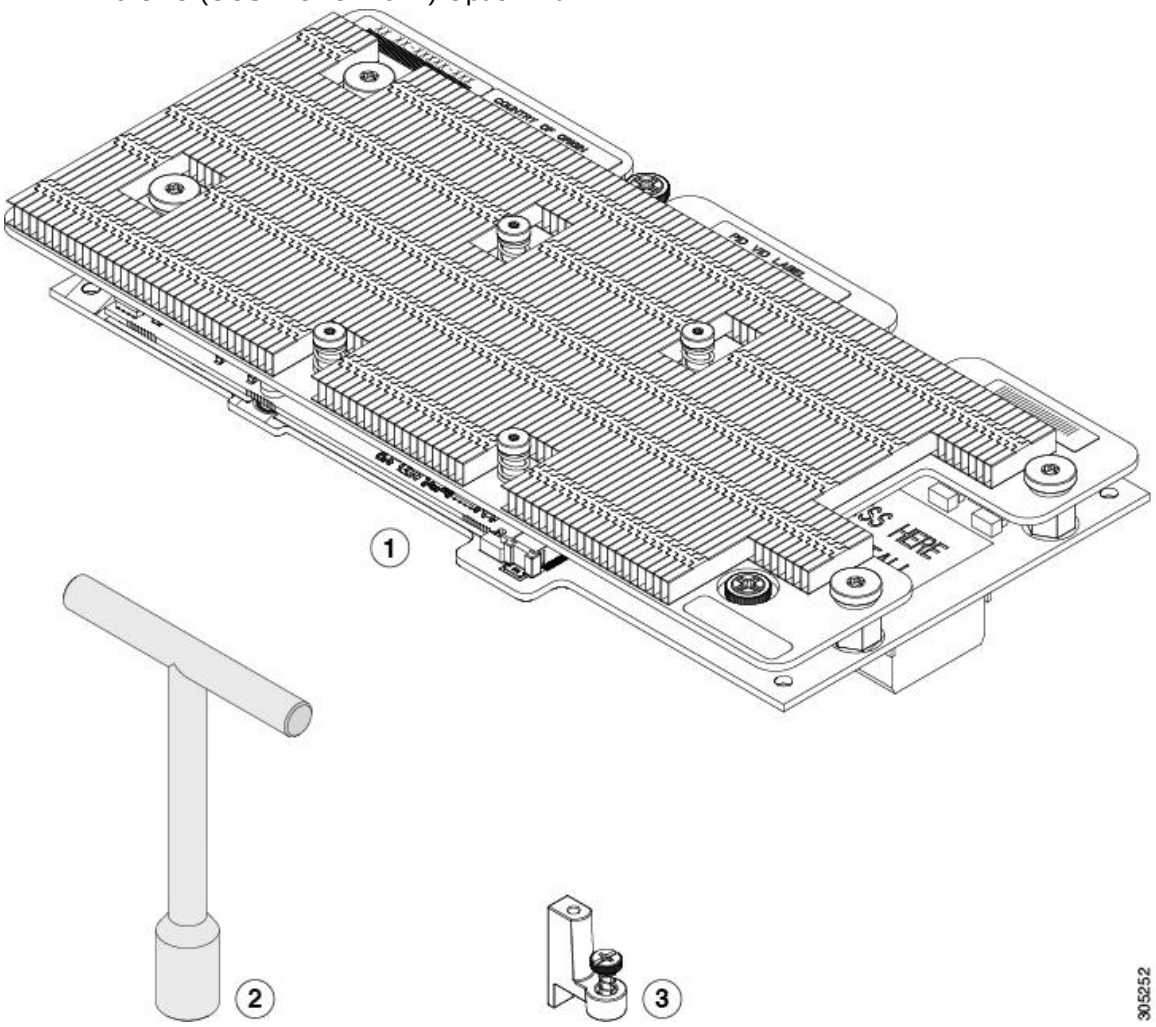


1	Front of the server	2	Leg with thumb screw that attaches to the motherboard
3	Thumbscrew to attach to standoff below	4	Standoff on the motherboard
5	Handle to press down on to firmly install the GPU	–	

Installing an NVIDIA GPU Card in the Rear of the Server

If you are installing the UCSB-GPU-P6-R to a server in the field, the option kit comes with the GPU itself (CPU and heatsink), a T-shaped installation wrench, and a custom standoff to support and attach the GPU to the motherboard. Figure 50 shows the three components of the option kit.

Figure 50 NVIDIA P6 GPU (UCSB-GPU-P6-R) Option Kit



1	NVIDIA P6 GPU (CPU and heatsink)	2	T-shaped wrench
3	Custom standoff	-	

Before installing the NVIDIA P6 GPU (UCSB-GPU-P6-R) in the rear mezzanine slot, you need to Upgrade the Cisco UCS domain that the GPU will be installed into to a version of Cisco UCS Manager that supports this card. Refer to the latest version of the *Release Notes for Cisco UCS Software* at the following URL for information about supported hardware: http://www.cisco.com/c/en/us/support/servers-unified-computing/ucs-manager/products-release-notes-list.html. Remove any other card, such as a VIC 1480, VIC 1380, or VIC port expander card from the rear mezzanine slot. You cannot use any other card in the rear mezzanine slot when the NVIDIA P6 GPU is installed.

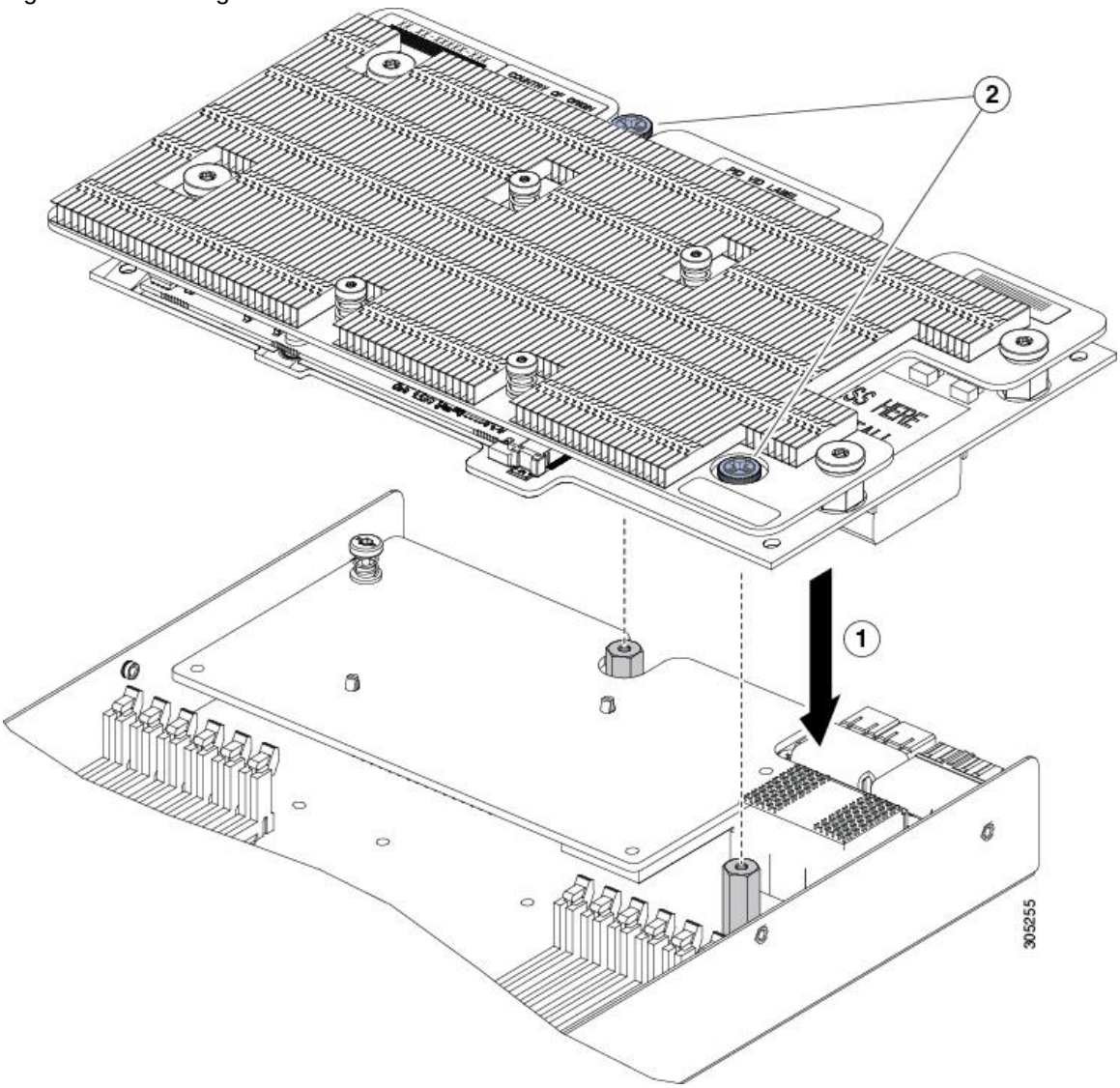
1. Use the T-shaped wrench that comes with the GPU to remove the existing standoff at the back end of the motherboard.

2. Install the custom standoff in the same location at the back end of the motherboard.

3. Position the GPU over the connector on the motherboard and align all the captive screws to the standoff posts (callout 1).

4. Tighten the captive screws (callout 2).

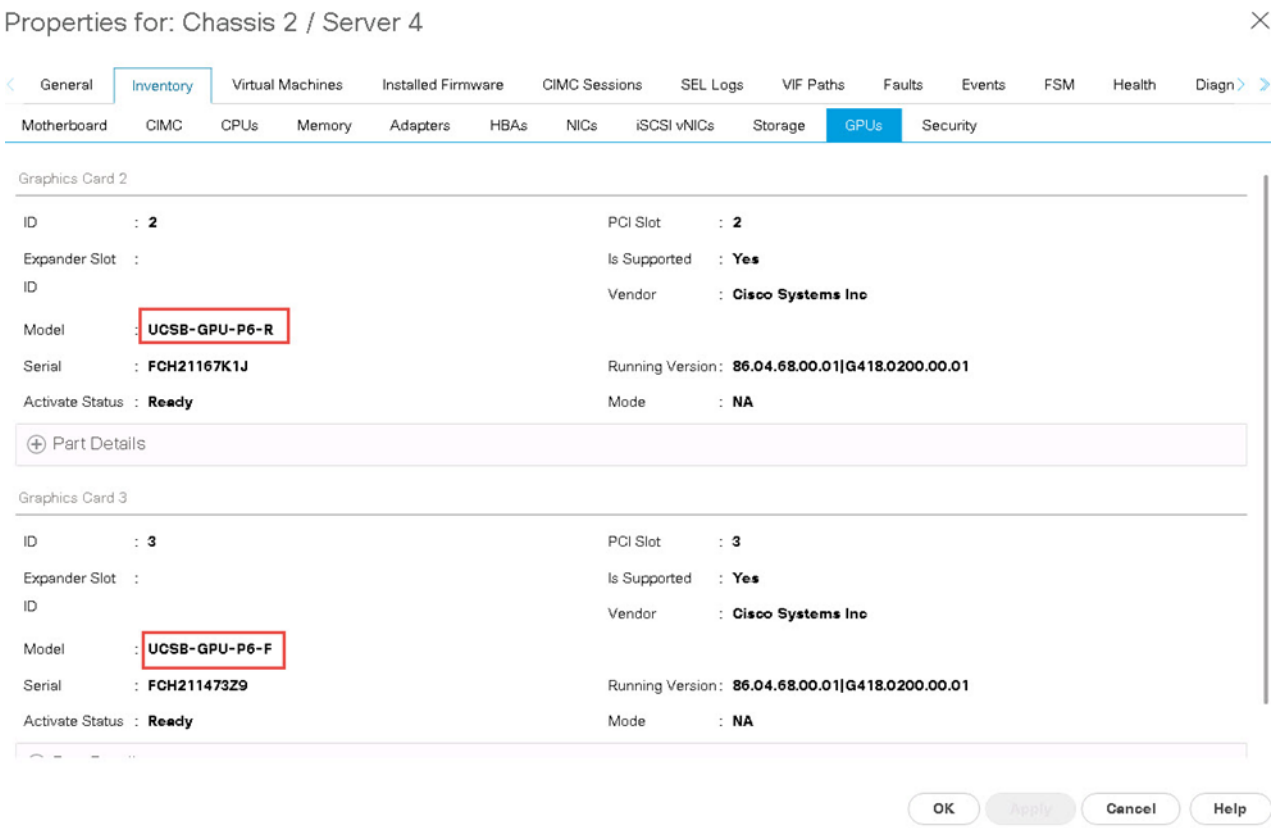
Figure 51 Installing the NVIDIA P6 GPU in the Rear Mezzanine Slot



Install the NVIDIA VMware VIB Driver

To install the NVIDIA VMware VIB driver, complete the following steps:

1. From Cisco UCS Manager, verify the GPU card has been properly installed.



2. Download the NVIDIA GRID GPU driver pack for VMware vSphere ESXi 6.5.

3. Upload the NVIDIA driver (vSphere Installation Bundle [VIB] file) to the /tmp directory on the ESXi host using a tool such as WinSCP. (Shared storage is preferred if you are installing drivers on multiple servers or using the VMware Update Manager.)

4. Log in as root to the vSphere console through SSH using a tool such as Putty.

The ESXi host must be in maintenance mode for you to install the VIB module. To place the host in maintenance mode, use the command esxcli system maintenanceMode set -enable true.

5. Enter the following command to install the NVIDIA vGPU drivers:

```
esxcli software vib install --no-sig-check -v /<path>/<filename>.VIB
```

The command should return output similar to that shown here:

```
# esxcli software vib install --no-sig-check -v /tmp/NVIDIA-VMware_ESXi_6.5_Host_Driver_384.99-
1OEM.650.0.0.4598673.vib
Installation Result
   Message: Operation finished successfully.
   Reboot Required: false
   VIBs Installed: NVIDIA_bootbank_NVIDIA-VMware_ESXi_6.5_Host_Driver_384.99-1OEM.650.0.0.4598673
   VIBs Removed:
   VIBs Skipped:
```

Although the display shows “Reboot Required: false,” a reboot is necessary for the VIB file to load and for xorg to start.

6. Exit the ESXi host from maintenance mode and reboot the host by using the vSphere Web Client or by entering the following commands:

```
#esxcli system maintenanceMode set -e false

#reboot
```

7. After the host reboots successfully, verify that the kernel module has loaded successfully using the following command:

```
#esxcli software vib list | grep -i nvidia
```

The command should return output similar to that shown here:

```
# esxcli software vib list | grep -i nvidia
NVIDIA-VMware_ESXi_6.5_Host_Driver  384.99-1OEM.650.0.0.4598673          NVIDIA
VMwareAccepted    2017-11-27
```

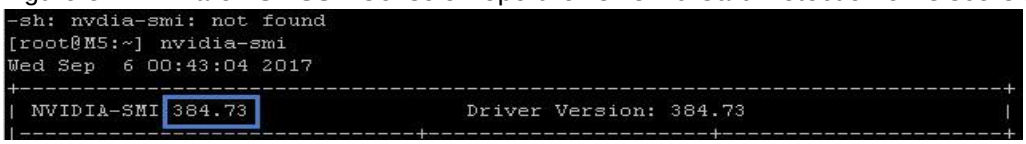
See the VMware knowledge base article for information about removing any existing NVIDIA drivers before installing new drivers: http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2033434.

8. Confirm GRID GPU detection on the ESXi host. To determine the status of the GPU card’s CPU, the card’s memory, and the amount of disk space remaining on the card, enter the following command:

```
#nvidia-smi
```

The command should return output similar to that shown in Figure 52, depending on the card used in your environment.

Figure 52 VMware ESX SSH Console Report for GPU P6 Card Detection on Cisco UCS B200 M5 Blade Server



```
[root@M5:~] nvidia-smi
Wed Sep  6 00:43:04 2017
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 384.73                 Driver Version: 384.73                    |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|===============================+======================+======================|
|   0  Tesla P6            On   | 00000000:18:00.0 Off |                  Off |
| N/A   21C    P8     9W /  90W |     41MiB / 16383MiB |      0%      Default |
+-------------------------------+----------------------+----------------------+
|   1  Tesla P6            On   | 00000000:D8:00.0 Off |                  Off |
| N/A   35C    P8    10W /  90W |     41MiB / 16383MiB |      0%      Default |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                       GPU Memory |
|  GPU       PID   Type   Process name                             Usage      |
|=============================================================================|
|  No running processes found                                                 |
+-----------------------------------------------------------------------------+
[root@M5:~]
```

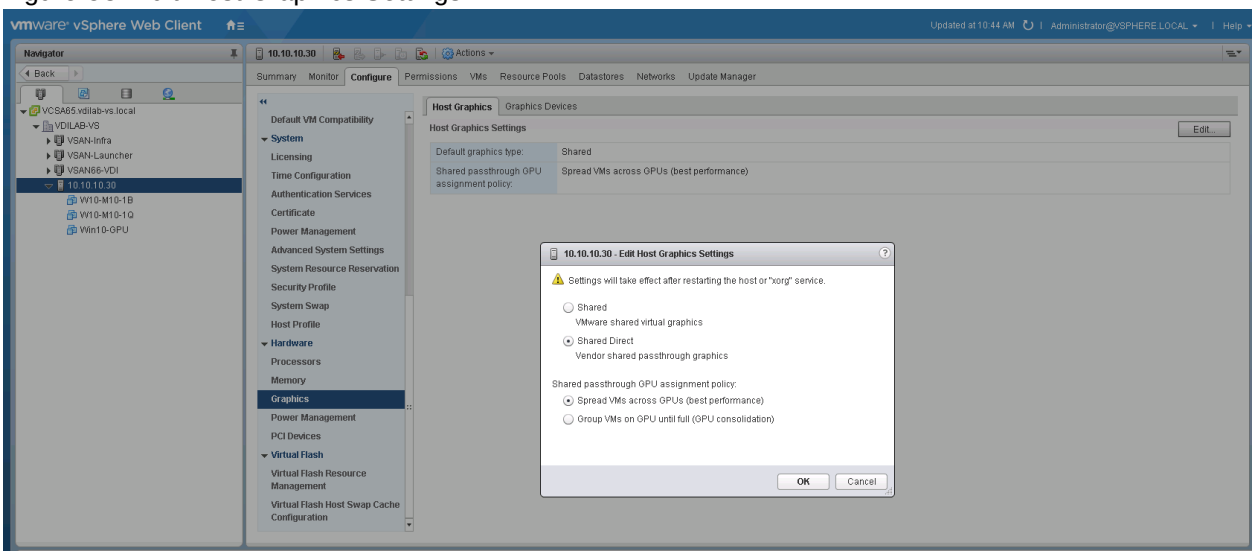
The NVIDIA system management interface (SMI) also allows GPU monitoring using the following command: nvidia-smi –l (this command adds a loop, automatically refreshing the display).

Configure a VM with a vGPU

To create the virtual machine that you will use as the VDI base image, complete the following steps:

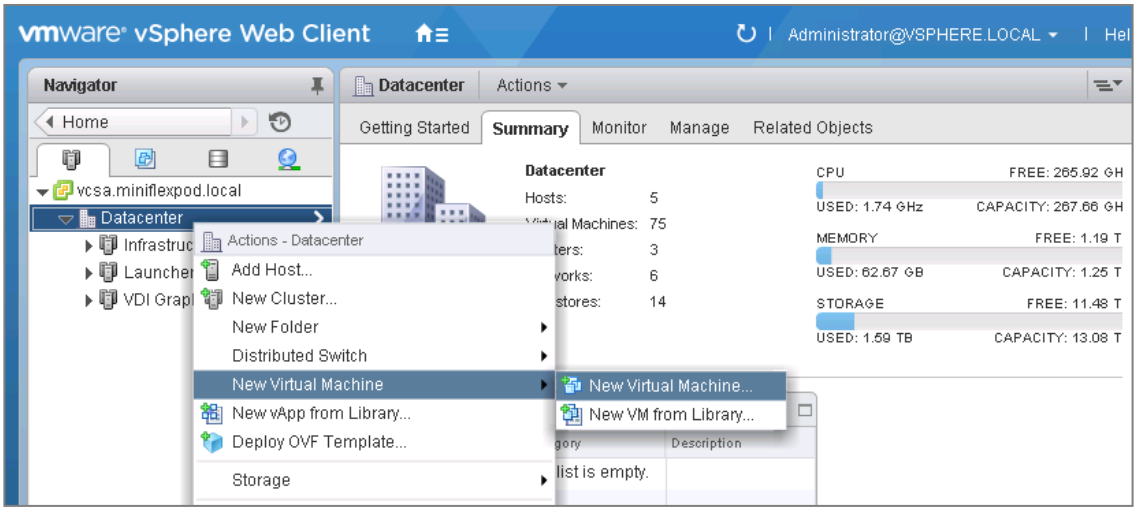
1. Select the ESXi host and click the Configure tab. From the list of options at the left, choose Graphics > Edit Host Graphics Settings. Select Shared Direct “Vendor shared passthrough graphics.” Reboot the system to make the changes effective.

Figure 53 Edit Host Graphics Settings



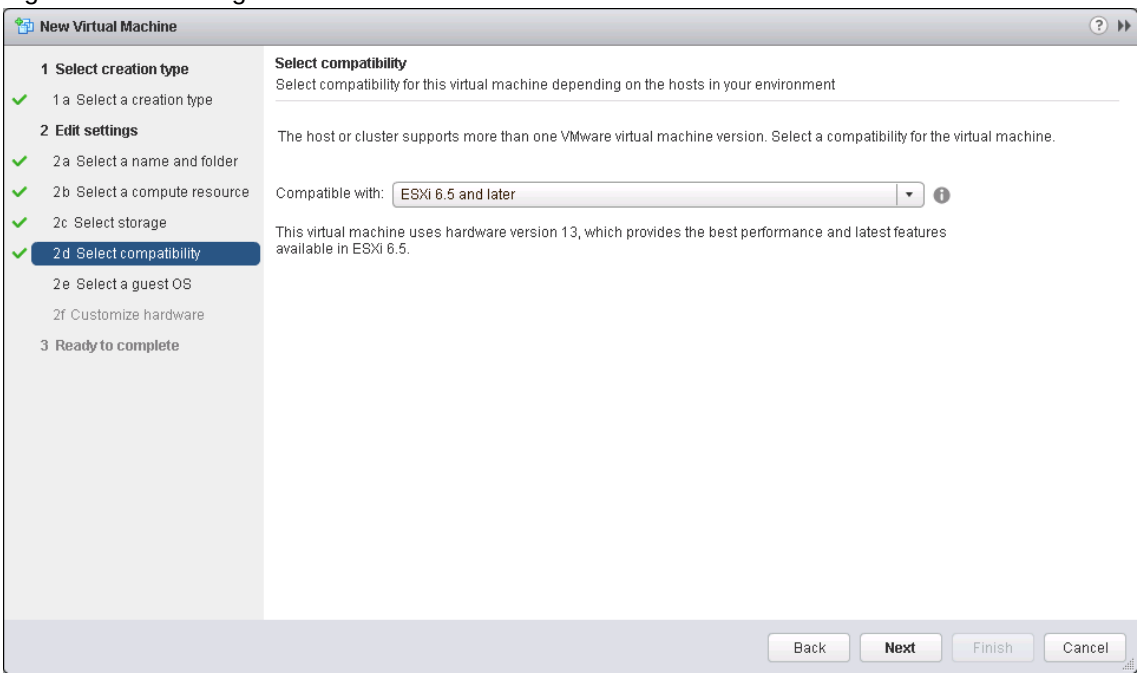
2. Using the vSphere Web Client, create a new virtual machine. To do this, right-click a host or cluster and choose New Virtual Machine. Work through the New Virtual Machine wizard. Unless another configuration is specified, select the configuration settings appropriate for your environment.

Figure 54 Creating a New Virtual Machine in VMware vSphere Web Client



3. Choose “ESXi 6.0 and later” from the “Compatible with” drop-down menu to use the latest features, including the mapping of shared PCI devices, which is required for the vGPU feature. This solution uses “ESXi 6.5 and later,” which provides the latest features available in ESXi 6.5 and virtual machine hardware Version 13.

Figure 55 Selecting Virtual Machine Hardware Version 11 or Later



4. To customize the hardware of the new virtual machine, add a new shared PCI device, select the appropriate GPU profile, and reserve all virtual machine memory.

If you are creating a new virtual machine and using the vSphere Web Client's virtual machine console functions, the mouse will not be usable in the virtual machine until after both the operating system and VMware Tools have been installed. If you cannot use the traditional vSphere Web Client to connect to the virtual machine, do not enable the NVIDIA GRID vGPU at this time.

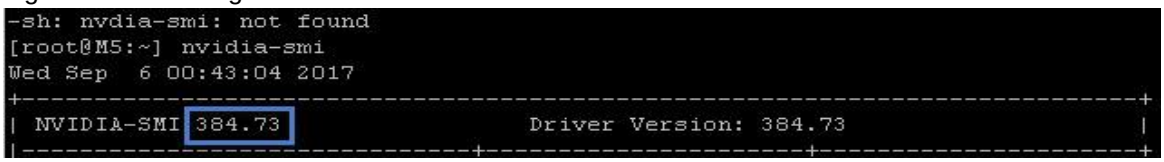
Figure 56 Adding a Shared PCI Device to the Virtual Machine to Attach the GPU Profile

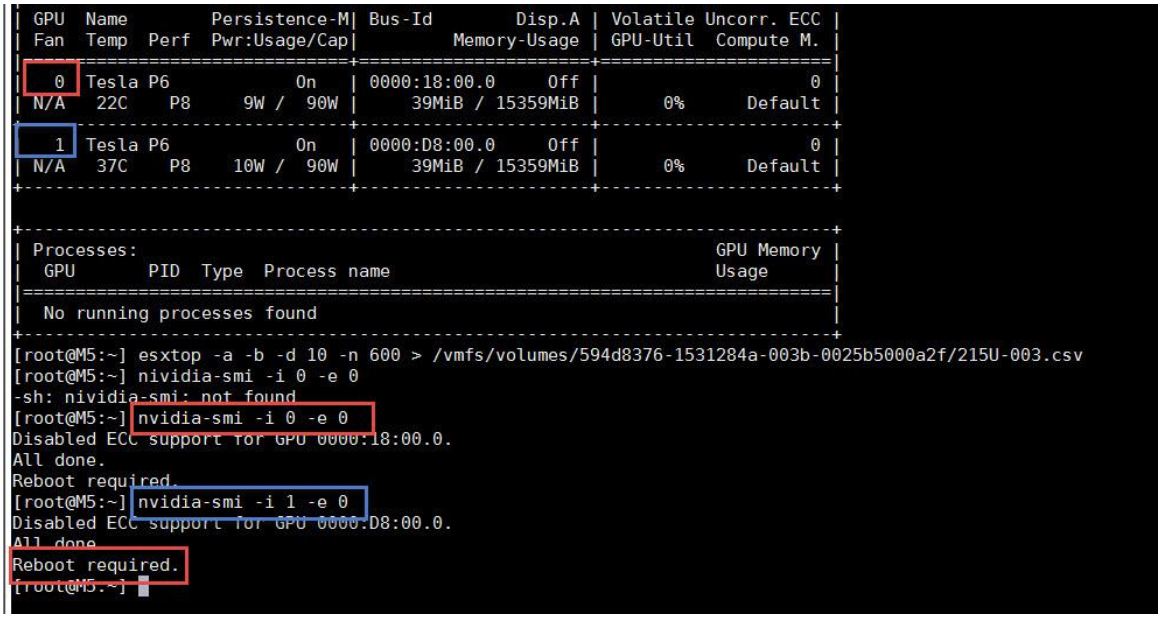
5. A virtual machine with a vGPU assigned will not start if ECC is enabled. If this is the case, as a workaround disable ECC by entering the following commands:

```
# nvidia-smi -i 0 -e 0

# nvidia-smi -i 1 -e 0
```

Use -i to target a specific GPU. If two cards are installed in a server, run the command twice as shown in the example here, where 0 and 1 each specify a GPU card.

Figure 57 Disabling ECC





6. Install and configure Microsoft Windows on the virtual machine:

   a. Configure the virtual machine with the appropriate amount of vCPU and RAM according to the GPU profile selected.

   b. Install VMware Tools.

   c. Join the virtual machine to the Microsoft Active Directory domain.

   d. Choose “Allow remote connections to this computer” on the Windows System Properties menu.

   e. Install VMware Horizon Agent with appropriate settings. Enable the remote desktop capability if prompted to do so.

f. Install Horizon Direct Connection agent.

g. Optimize the Windows OS. VMware OSOT, the optimization tool, includes customizable templates to enable or disable Windows system services and features using VMware recommendations and best practices across multiple systems. Because most Windows system services are enabled by default, the optimization tool can be used to easily disable unnecessary services and features to improve performance.

h. Restart the Windows OS when prompted to do so.

Install the GPU Drivers Inside Windows VM

It is important to note that the drivers installed with the Windows VDI desktop must match the version that accompanies the driver for the ESXi host. So if you downgrade or upgrade the ESXi host vib, you must do the same with the NVIDIA driver in your Windows master image.

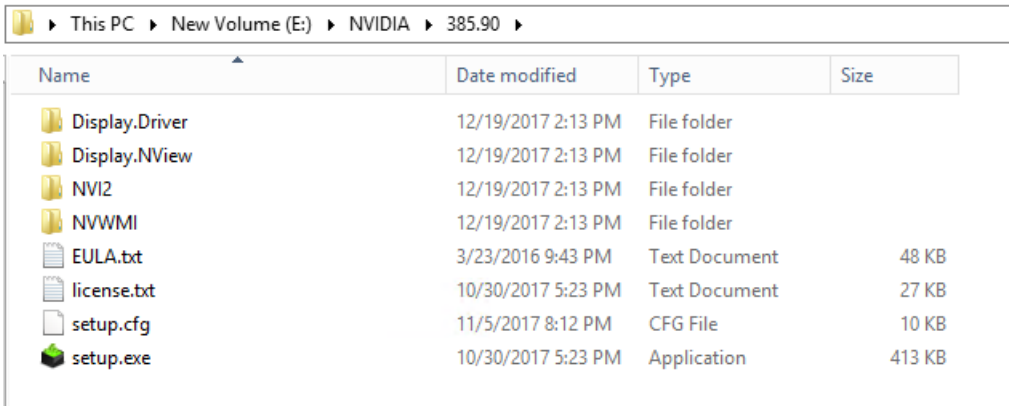
In this study we used ESXi Host Driver version 352.83 and 354.80 for the Windows VDI image. These drivers come in the same download package from NVIDIA.

To install the GPU drivers, complete the following steps:

1. Copy the Microsoft Windows drivers from the NVIDIA GRID vGPU driver pack downloaded earlier to the master virtual machine.

2. Copy the 32- or 64-bit NVIDIA Windows driver from the vGPU driver pack to the desktop virtual machine and run setup.exe.

Figure 58 NVIDIA Driver Pack



The vGPU host driver and guest driver versions need to match. Do not attempt to use a newer guest driver with an older vGPU host driver or an older guest driver with a newer vGPU host driver. In addition, the vGPU driver from NVIDIA is a different driver than the GPU pass-through driver.

3. Agree to the NVIDIA software license.

Figure 59 Agreeing to the NVIDIA Software License

4. Install the graphics drivers using the Express or Custom option (Figures 51 and 52). After the installation has completed successfully, restart the virtual machine.

Make sure that remote desktop connections are enabled. After this step, console access may not be available for the virtual machine when you connect from a vSphere Client.

Figure 60 Selecting the Express or Custom Installation Option

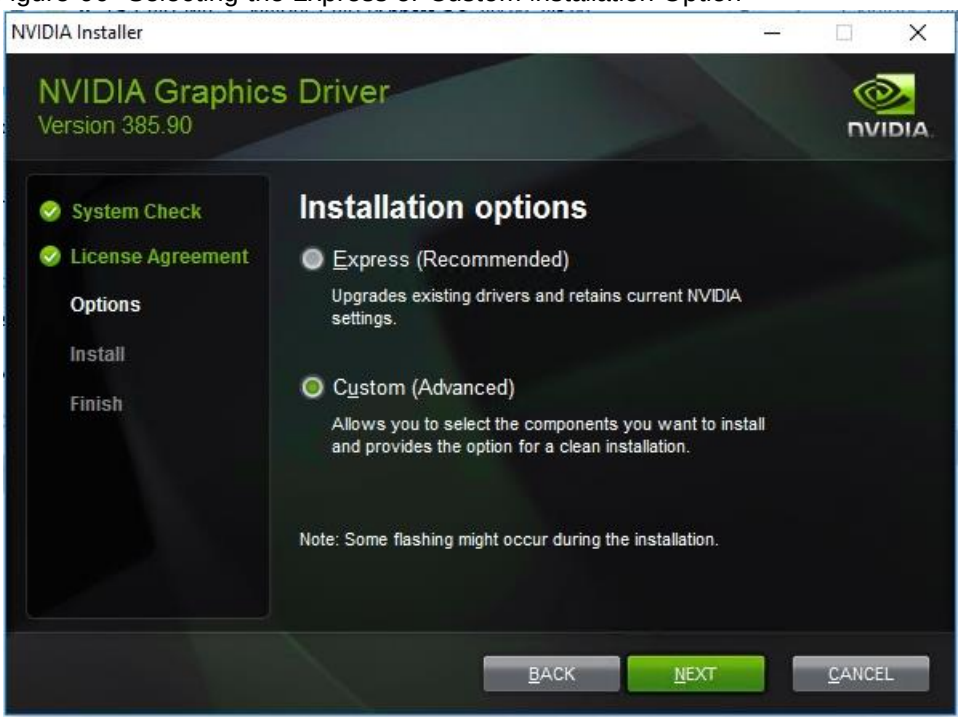


Figure 61 Components Installed During NVIDIA Graphics Driver Custom Installation Process

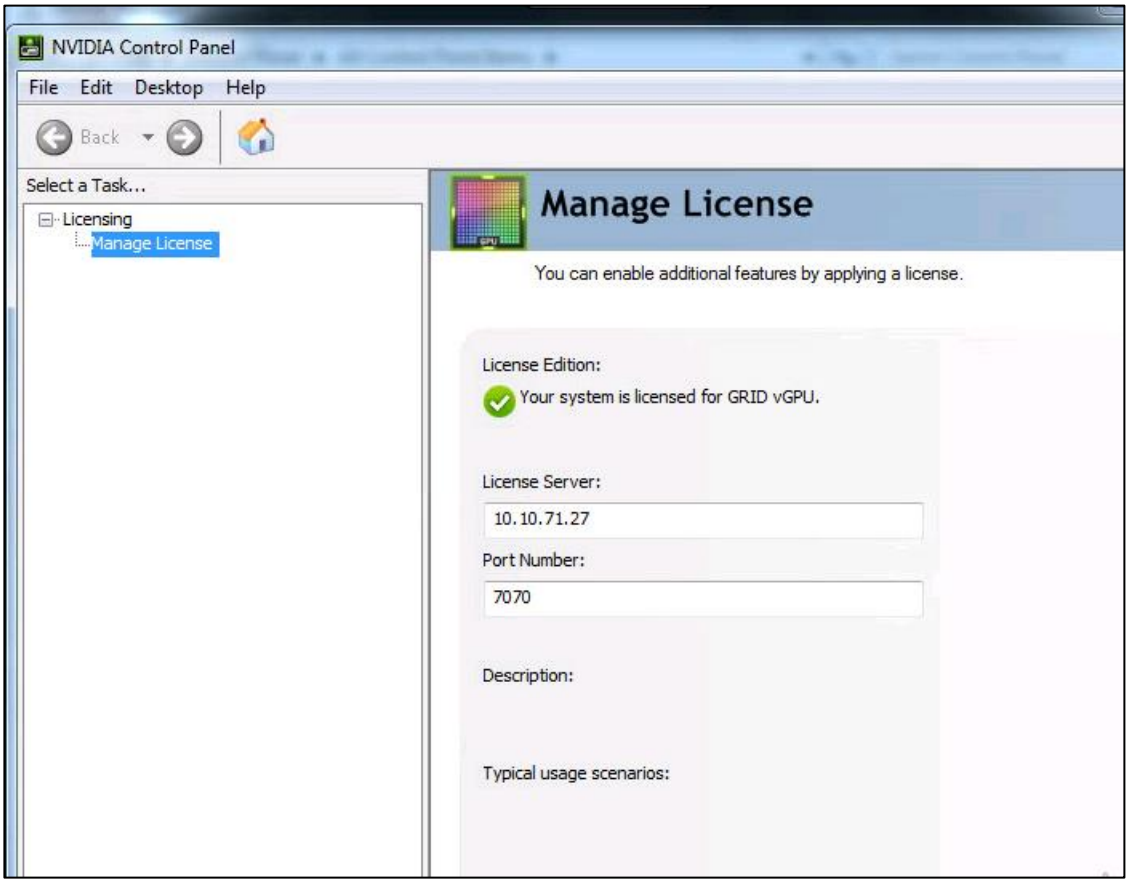
Figure 62 Resarting the Virtual Machine

Configure NVIDIA Grid License Server on Virtual Machine

When the License server is properly installed, you must point the master image to the license server so the VMs with vGPUs can obtain a license. To do so, complete the following steps:

1. In the Windows Control Panel, double-click the NVidia Control Panel.

2. In the Control Panel, enter the IP or FQDN of the Grid License Server. You will receive a result similar to the one shown below.



Cisco UCS Performance Manager

Cisco UCS Performance Manager provides visibility from a single console into Cisco UCS components for performance monitoring and capacity planning. It provides data center assurance of integrated

infrastructures and ties application performance to physical and virtual infrastructure performance. This allows you to optimize resources and deliver better service levels to your customers.

The release used in this solution features an additional component, Control Center, which is an open-source, application service orchestrator based on Docker.

Control Center greatly simplifies the installation, deployment, and management of Cisco UCS Performance Manager.

This section provides a brief introduction to Control Center, and describes how it affects Cisco UCS Performance Manager deployments.

Installing Cisco UCS Performance Manager

Installing the Control Center Master Host

To install a Cisco UCS Performance Manager appliance package as a Control Center master host, using VMware vSphere, complete the following steps:

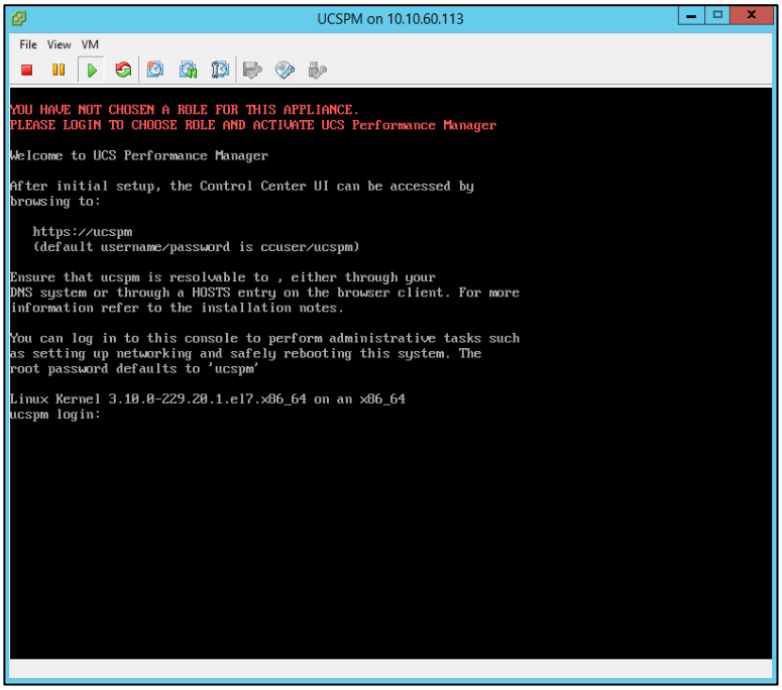
1. Download the Cisco UCS Performance Manager OVA file from the Cisco UCS Performance Manager site to your workstation.
2. Use the VMware vSphere Client to log in to vCenter as root, or as a user with superuser privileges, and then display the Home view.
3. Deploy OVF template.
4. In the Name and Location panel, provide a name and a location for the server.
5. In the Name field, enter a new name or use the default.
6. In the Inventory Location area, select a data center for the virtual machine.
7. Click Next.
8. In the Host / Cluster panel, select a host system, and then click Next.
9. In the Storage panel, select a storage system with sufficient space for your Cisco system, and then click Next.
10. In the Disk Format panel, select Thin Provision, and then click Next.
11. In the Ready to Complete panel, review the deployment settings, and then click Finish. Please do not check the check box labeled Power on after deployment.
12. Navigate to the new virtual machine's Getting Started tab, and then click the Edit virtual machine settings link.
13. In the Virtual Machine Properties dialog, select Memory in the Hardware table.
14. In the Memory Configuration area, set the Memory Size field to 64GB, and then click the OK button.

15. On the new virtual machine's Getting Started tab, click the Power on virtual machine link.

Configure the Control Center Host Mode

To configure the Control Center host mode, complete the following steps:

1. Gain access to the console interface of the Control Center host through your hypervisor console interface.



2. Log in as the root user.

3. The initial password is ucspm.

4. The system prompts you to enter a new password for root.

Passwords must include a minimum of eight characters, with at least one character from three of the following character classes: uppercase letter, lowercase letter, digit, and special.

5. The system prompts you to enter a new password for ccuser. The ccuser acount is the default account for gaining access to the Control Center browser interface.

6. Select the master role for the host.

7. In the Configure appliance menu, press the Tab key to select the Choose button.

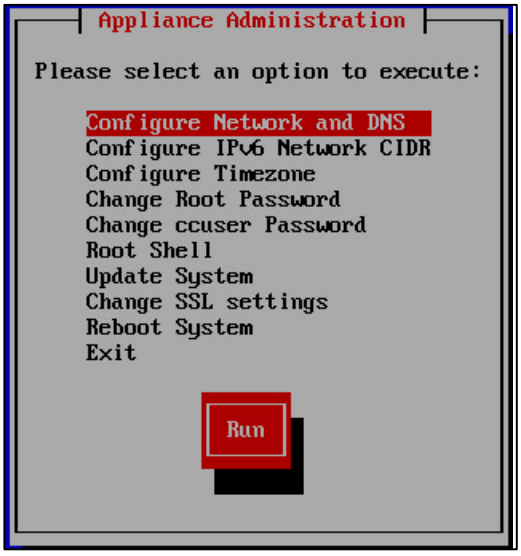
8. Press the Enter key.

The system will now restart.

Edit a Connection

The default configuration for network connections is DHCP. To configure static IPv4 addressing, complete the following steps:

1. After the systems restarts, log in as the root user.

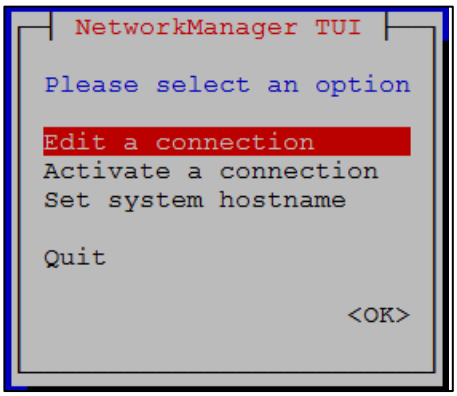


2. Select the NetworkManager TUI menu.

3. The In the Appliance Administration menu, select the Configure Network and DNS option.

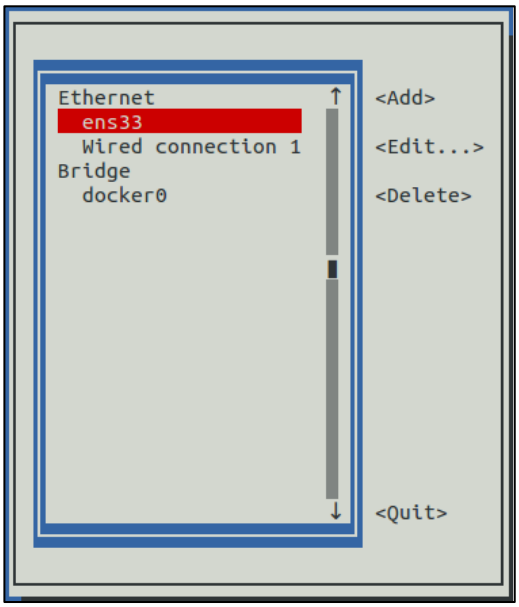
4. Press the Tab key to select the Run button.

5. Press the Enter key.

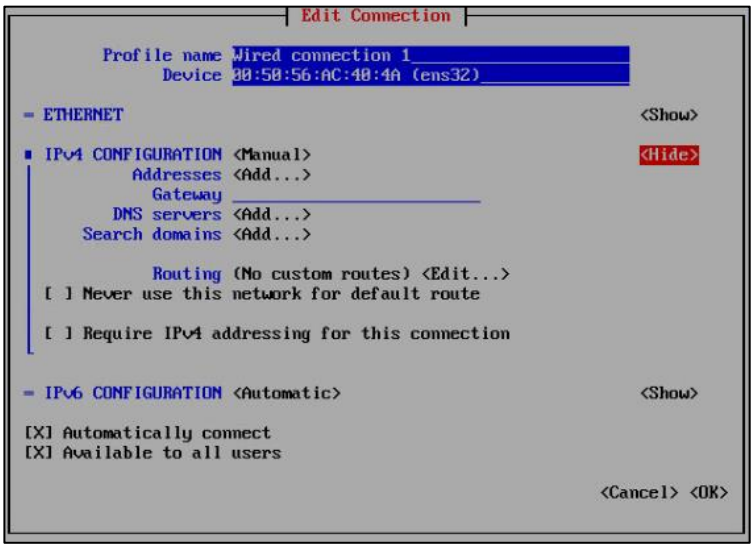


6. On the NetworkManager TUI menu, select Edit a connection, and then press the Return key.

The TUI displays the connections that are available on this host.



7. Use the down-arrow key to select Wired Connection 1 and then press the Return key.



8. Use the Tab key and the arrow keys to navigate among options in the Edit Connection screen, and use the Return key to toggle an option or to display a menu of options.

9. Optional: If the IPv4 CONFIGURATION area is not visible, select its display option (<Show>), and then press the Return key.

10. In the IPv4 CONFIGURATION area, select <Automatic>, and then press the Return key.

11. Configure static IPv4 networking.

12. Use the down arrow key to select Manual, and then press the Return key.

13. Use the Tab key or the down arrow key to select the <Add...> option next to Addresses, and then press the Return key.

14. In the Addresses field, enter an IPv4 address for the virtual machine, and then press the Return key.

15. Repeat the preceding two steps for the Gateway and DNS servers fields.

16. Use the Tab key or the down arrow key to select the <OK> option at the bottom of the Edit Connection screen, and then press the Return key.

17. In the available connections screen, use the Tab key to select the <Quit> option, and then press the Return key.

18. Reboot the operating system.

Enabling Access to Browser Interfaces

Control Center and Cisco UCS Performance Manager have independent browser interfaces served by independent web servers:

- The Control Center web server listens at HostnameOrIP:443. So, for a Control Center master host named cc-master.example.com, the hostname-based URL to use is https://cc-master.

- The Cisco UCS Performance Manager web server listens at a virtual hostname, ucspm.HostnameOrIP:443. For a Control Center master host named cc-master.example.com, the hostname-based URL to use is https://ucspm.cc-master.

To enable access to the browser interfaces by hostname, add name resolution entries to the DNS servers in your environment, or to the hosts files of individual client systems.

- On Windows client systems, the file is C:\Windows\System32\drivers\etc\hosts.
- Linux and OS/X client systems, the file is /etc/hosts.

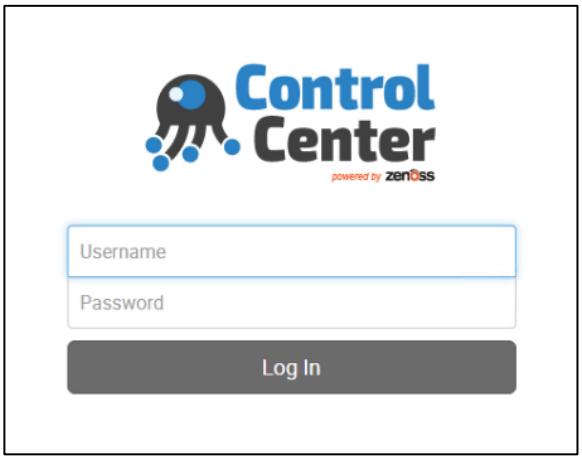
For example, the following entry identifies a Control Center master host at IP address 10.24.164.120, hostname cc-master, in the example.com domain.

10.24.164.120 cc-master.example.com cc-master ucspm.cc-master

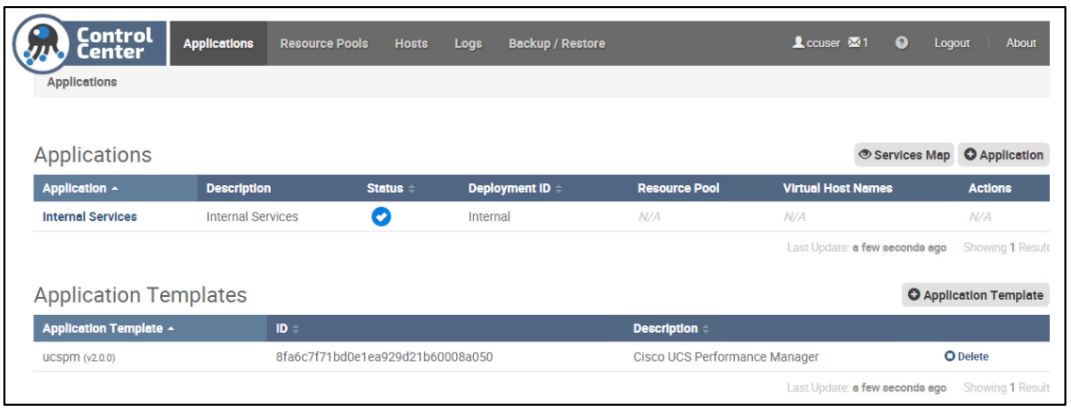
Deploy Cisco UCS Performance Manager

To log into Control Center for the first time, complete the following steps:

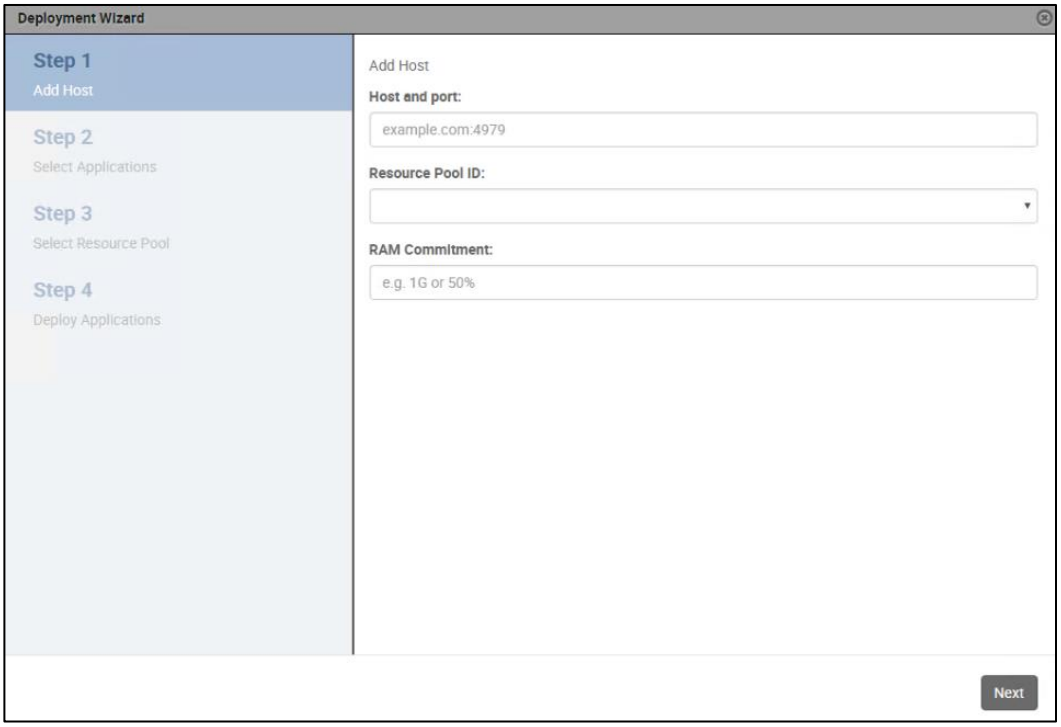
1. Display the login page of the Control Center browser interface.
2. Replace Hostname with the name of the Cisco UCS Performance Manager virtual machine.



3. At the login page, enter ccuser and its password.



4. On the Applications page, click the + Application button, located at the right side of the page.



5. In the Deployment Wizard, add the Control Center master host to the default resource pool.

6. In the Host and Port field, enter the hostname or IP address of the Control Center master host, followed by a colon character (:), and then 4979.

7. If you enter a hostname, all hosts in your Control Center cluster must be able to resolve the name, either through an entry in /etc/hosts, or through a nameserver on your network.

8. In the Resource Pool ID field, select default from the list, and then click Next.

9. In the RAM Committment field, enter the percentage of master host RAM to devote to Control Center and Cisco UCS Performance Manager.

10. The amount of RAM required for the operating system is not included in this value. Cisco recommends entering 100 in the field.

11. At the bottom of the Deployment Wizard, click Next.

12. Select the application to deploy.

13. Select ucspm.

14. At the bottom of the Deployment Wizard, click Next.

15. Select the resource pool for the application.

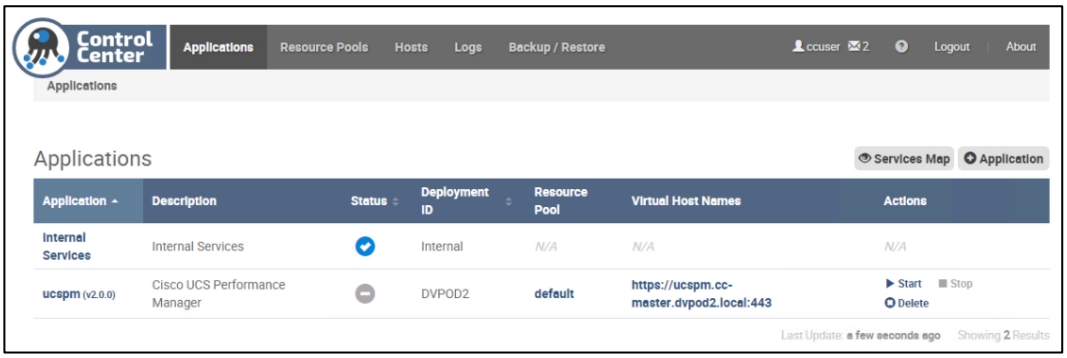
16. Select default.

17. At the bottom of the Deployment Wizard, click Next.

18. Choose a deployment ID and deploy Cisco UCS Performance Manager.

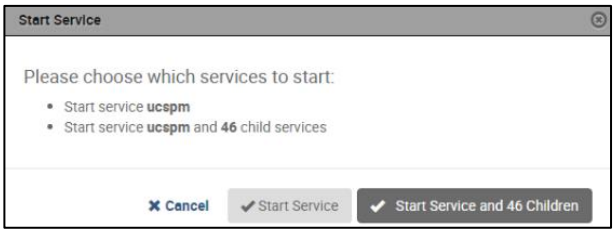
19. In the Deployment ID field, enter a name for this deployment of Cisco UCS Performance Manager.

20. At the bottom of the Deployment Wizard, click Deploy.



21. At the top of the page, click Logout. The control is located at the right side of the page.

22. In the Actions column of the Applications table, click the Start control of the ucspm row.

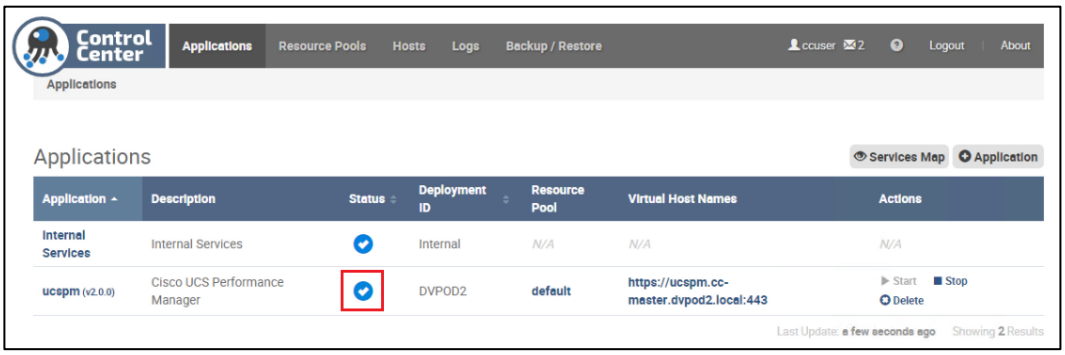


23. In the Start Service dialog, click Start Service and 46 Children button.

24. In the Application column of the Applications table, click ucspm in the ucspm row.

25. Scroll down to watch child services starting.

Typically, child services take 4-5 minutes to start. When no child services start, a red exclamation point icon displays. When child services do start, you will see Cisco UCS Performance Manager is running as shown below.



Setting up Cisco UCS Performance Manager

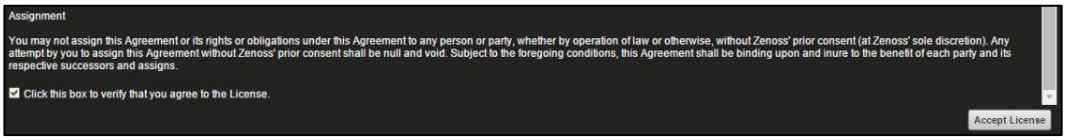
This section describes how to use the Cisco UCS Performance Manager Setup Wizard to accept the end-user license agreement, to provide your license key, define users and passwords, to set up UCS Domains, and to add additional infrastructure.

Initial Setup

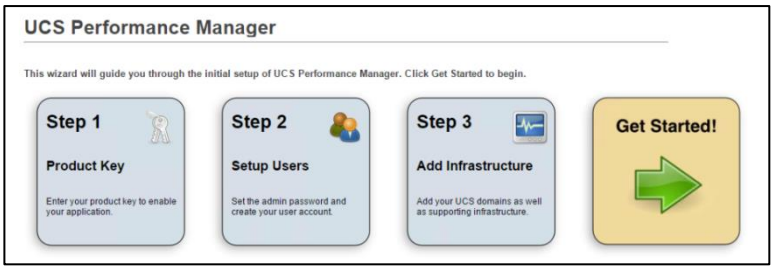
After installing Cisco UCS Performance Manager on a virtual machine, and starting it in Control Center, complete the following steps:

1. In a web browser, navigate to the login page of the Cisco UCS Performance Manager interface. Cisco UCS Performance Manager redirects the first login attempt to the Setup page, which includes the End User License Agreement (EULA) dialog.

2. Read through the agreement. At the bottom of the EULA dialog, check the check box on the left side, and then click the Accept License button on the right side.



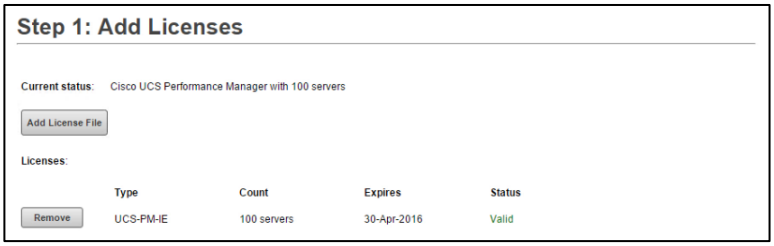
3. On the Cisco UCS Performance Manager Setup page, click Get Started!



4. On the Add Licenses page, click the Add License File button.

If you do not have your license file yet, you can use the trial version for up to 30 days. You can enter your license file at a later date through the user interface. See the "Product Licensing" section of the Cisco UCS Performance Manager Administration Guide.

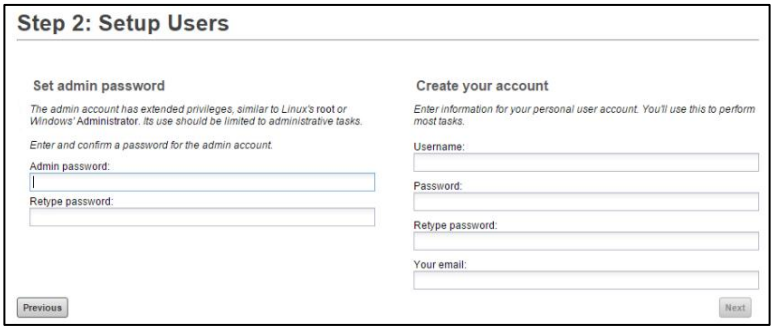
5. In the Open dialog, select your license file, and then click Open.



6. Proceed to the next task or repeat the preceding step.

7. In the Set admin password area, enter and confirm a password for the admin user account.

Passwords must contain a minimum of 8 characters, including one capital letter and one digit.



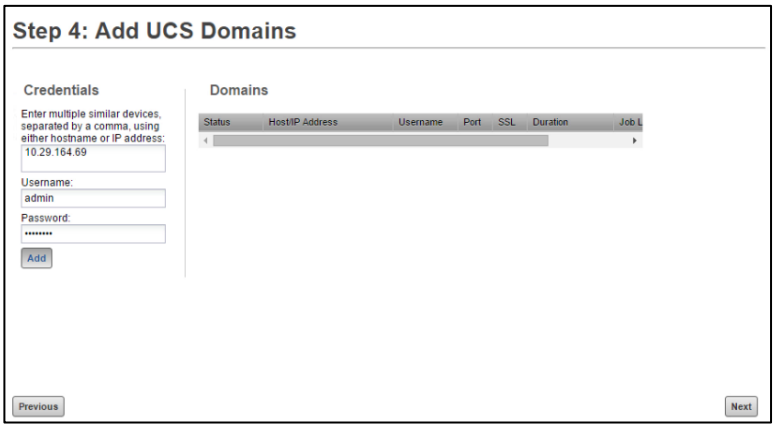
8. In the Create your account area, create one additional administrative user account name and password.

9. Click Next.

Add Cisco UCS Domains

To add the Cisco UCS Domain to Cisco UCS Performance Manager after completing the initial setup configuration, complete the following steps:

1. On the Add UCS Domains page, provide connection credentials for one or more Cisco UCS domains.



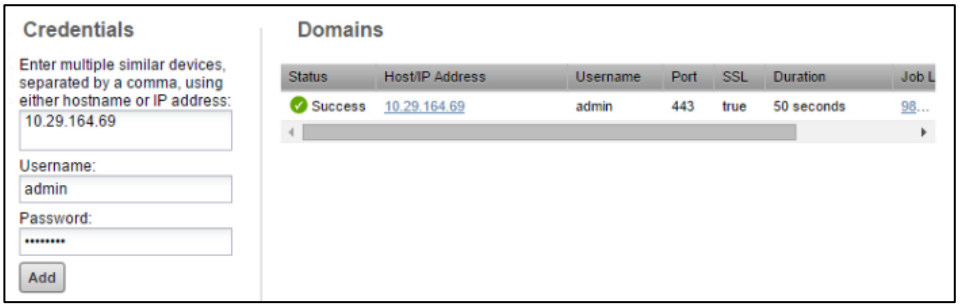
2. In the Enter multiple similar devices, separated by a comma, using either hostname or IP address field, enter the fully-qualified domain name or IP address of a UCS domain server.

3. In the Username field, enter the name of a user account in the UCS domain that is authorized for read access to the resources you plan to monitor.

4. In the Password field, enter the password of the user account specified in the preceding step.

5. Click Add.

6. Review the information in the Status column of the Domains table, and then remove a domain, add a domain, or continue.



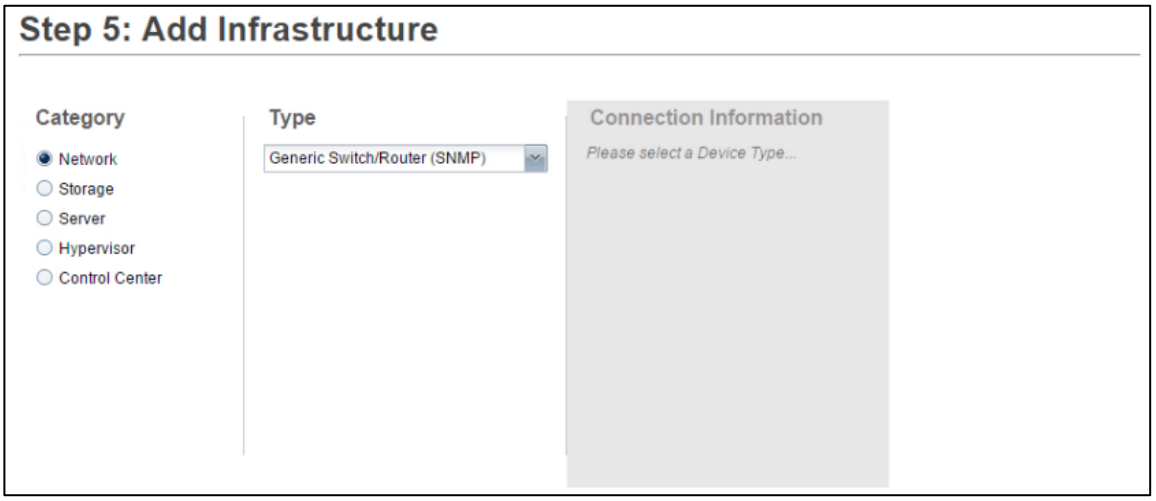
7. If the final message in the Status column is Failure, click the button in the Remove column, and then try again to add a domain.

8. If the final message in the Status column is Success, you may add another domain or continue to the next page.

9. Click Next to continue to the Add Infrastructure step.

Adding Infrastructure Devices

To add the Infrastructure Devices to Cisco UCS Performance Manager after completing the initial setup configuration, complete the following steps:



1. This step is optional. Click Finish to exit the Setup Wizard. You will then be taken to the Dashboard.

2. The Setup Wizard times out after 20 minutes if you have not completed it. You may restart Setup Wizard by closing its browser window or tab, and then logging in again. Also, you may add devices through the Add Infrastructure page at any time.

3. As it relates to this solution, other infrastructure devices that can be added include the Cisco Nexus 1000V, Pure storage, ESXi hosts using SOAP, and Windows Servers using SNMP or WinRM.

Add Nexus 9000 Series Switches

To add the Infrastructure Devices to Cisco UCS Performance Manager after completing the initial setup configuration, complete the following steps:

In order to monitor Cisco Nexus 9000 Series devices, you must first enable NX-API with the feature manager CLI command on the device. For detailed instructions on performing this task, see the following Cisco documentation: http://www.cisco.com/c/en/us/td/docs/switches/datacenter/nexus9000/sw/6-x/programmability/guide/b_Cisco_Nexus_9000_Series_NXOS_Programmability_Guide/b_Cisco_Nexus_9000_Series_NXOS_Programmability_Configuration_Guide_chapter_0101.html#concept_BCCB1EFF9C4A4138BECE9ECC0C4E38DF

1. In the Category area, select Network.

2. In the Type list, select Cisco Nexus 9000 (SNMP + Netconf).

The protocol used to gather data from the device is included in the list, in parentheses.

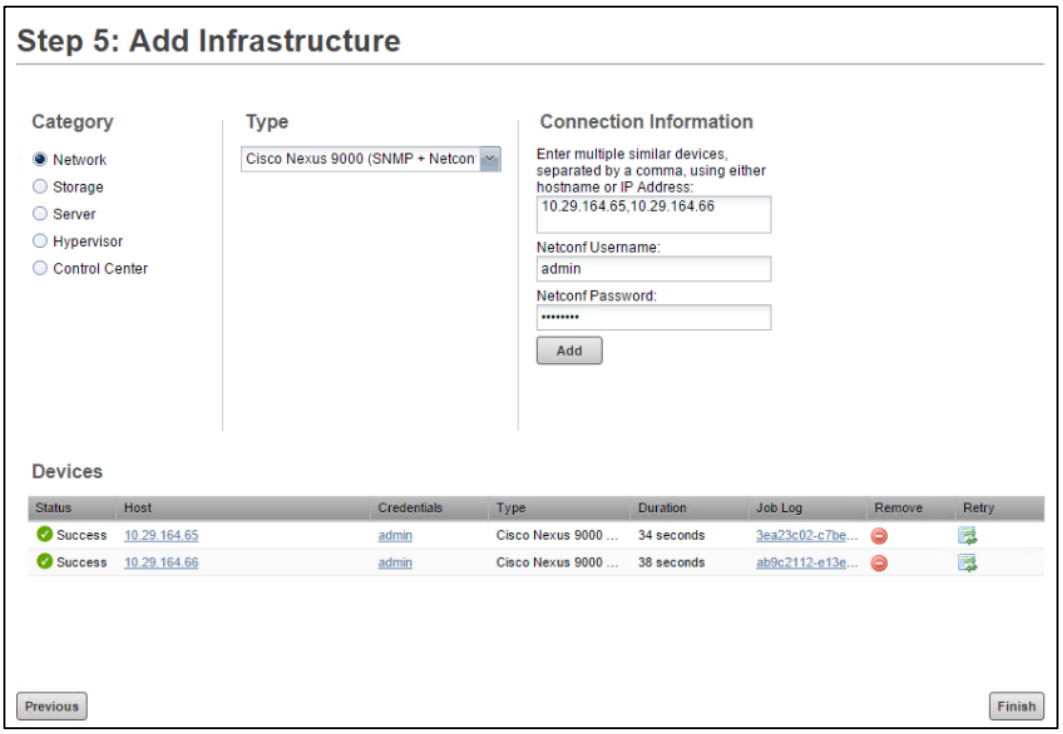
3. In the Connection Information area, specify the two 93108 switches to add.

4. In the Enter multiple similar devices, separated by a comma, using either hostname or IP Address field, enter the hostname or IP address of one or more switch or router devices on your network.

5. In the Username or Netconf Username field, enter the name of a user account on the device.

6. In the Password or Netconf Password field, enter the password of the user account specified in the previous field.

7. Click Add.



8. When finished adding network devices, click Next.

Cisco UCS Performance Manager, in addition to monitoring Cisco hardware operations, is able to monitor the vSphere environment.

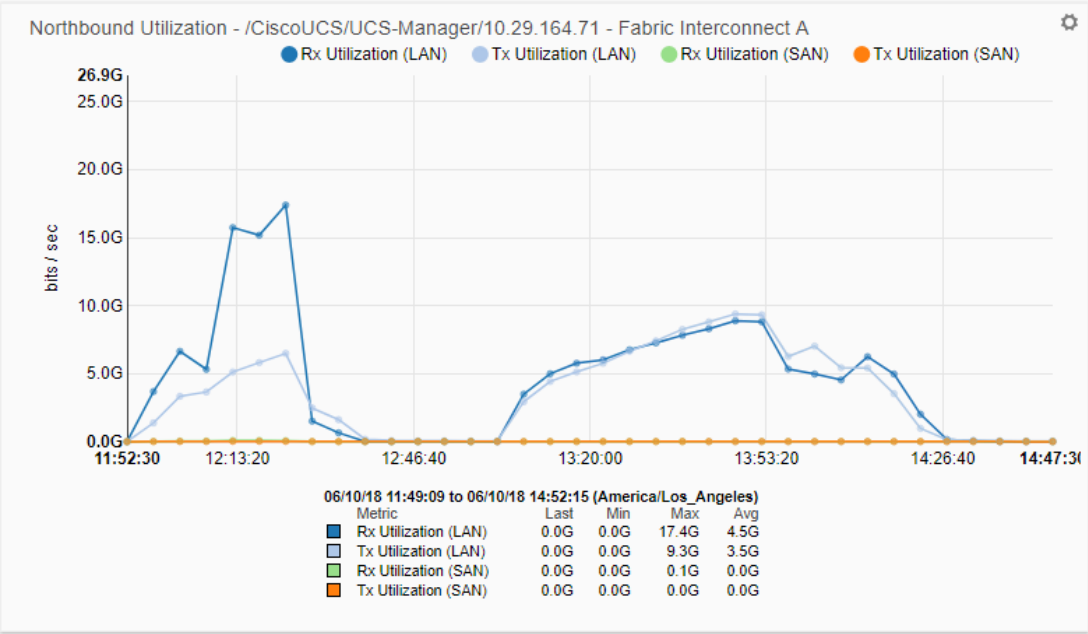
The following operational reports are available for vSphere:

- Clusters - Shows all clusters, with the count of VMs (total and powered on), hosts, and CPU/Memory utilization within each cluster.
- Datastores - Shows all datastores, with the number of connected VMs (total and powered on) and the disk space available and consumed on each datastore.
- Hosts - Shows all hosts, with the count of VMs (total and powered on), hosts, CPU/Memory reservation and utilization on each host.
- VMs - Shows all VMs, their operating system, CPU/Memory utilization, and which host/cluster they reside within.
- VMware Utilization - Provides a summary of VMs, CPU, memory, and disk utilization over a specified time interval, broken down by host.

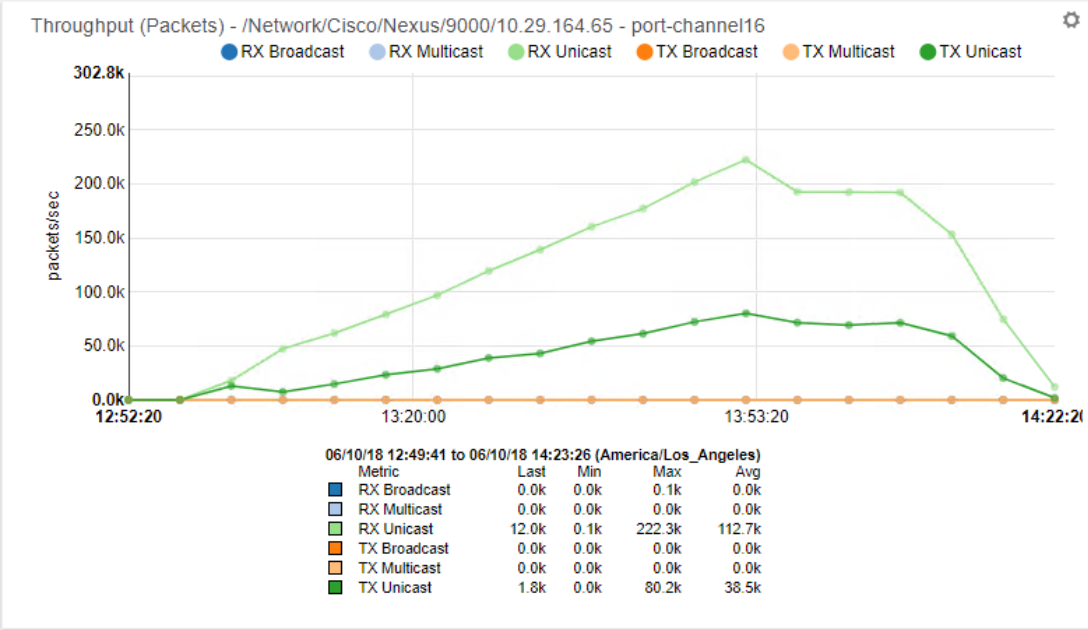
Cisco UCS Performance Manager Sample Test Data

The following samples represent just some of the useful data that can be obtained using Cisco UCS Performance Manager.

The chart shows the network usage from a Fabric Interconnect (Fabric A) during the boot storm and 6000 user mixed workload test.



The chart shows the throughput from Nexus switch (Fabric A) during the 6000 user mixed workload test.



Cisco Intersight Cloud Based Management

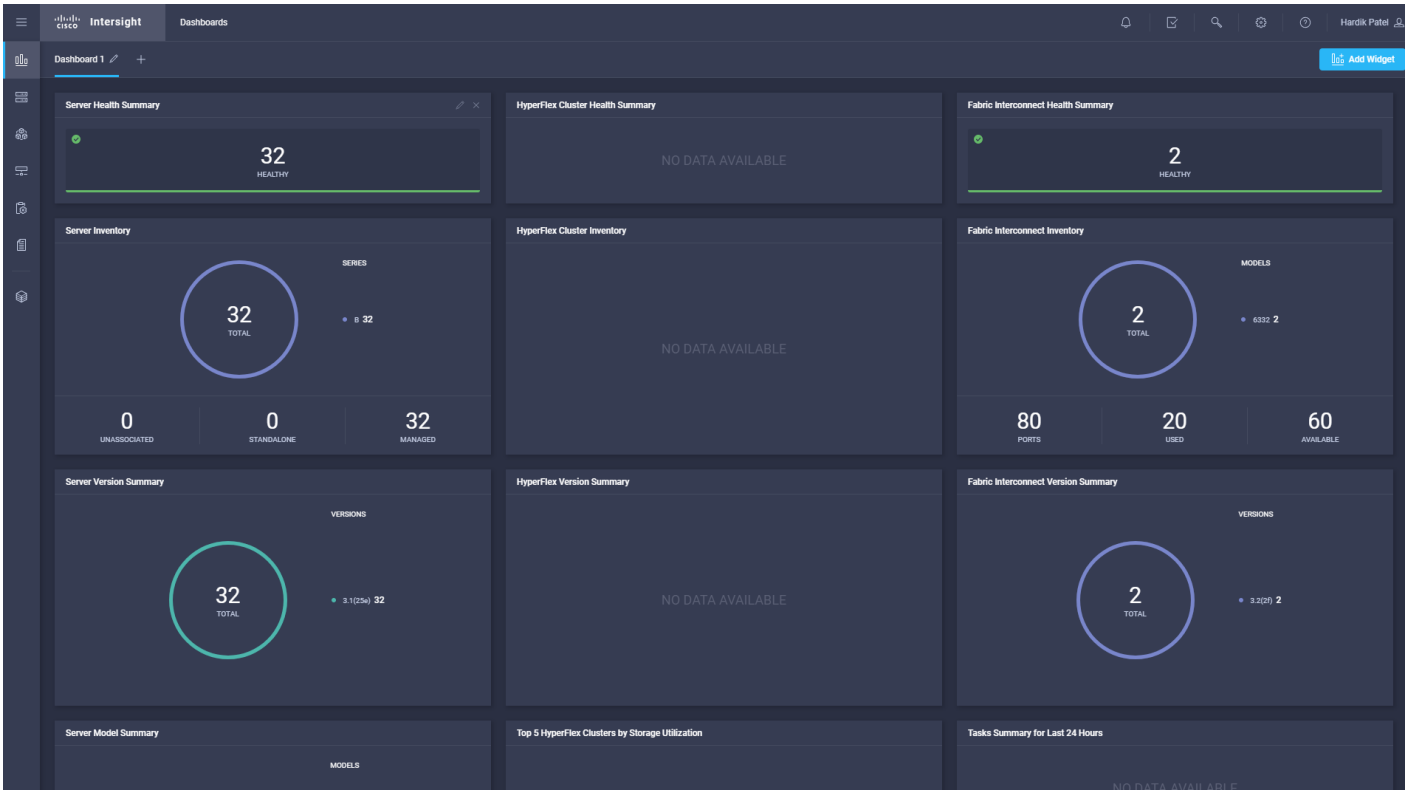
Cisco Intersight is Cisco's new systems management platform that delivers intuitive computing through cloud-powered intelligence. This platform offers a more intelligent level of management that enables IT organizations to analyze, simplify, and automate their environments in ways that were not possible with prior generations of tools. This capability empowers organizations to achieve significant savings in Total Cost of Ownership (TCO) and to deliver applications faster, so they can support new business initiates. The advantages of the model-based management of the Cisco UCS platform plus Cisco Intersight are extended

to Cisco UCS servers and Cisco HyperFlex and Cisco HyperFlex Edge systems. Cisco HyperFlex Edge is optimized for remote sites, branch offices, and edge environments.

The Cisco UCS and Cisco HyperFlex platforms use model-based management to provision servers and the associated storage and fabric automatically, regardless of form factor. Cisco Intersight works in conjunction with Cisco UCS Manager and the Cisco® Integrated Management Controller (IMC). By simply associating a model-based configuration with a resource through service profiles, your IT staff can consistently align policy, server personality, and workloads. These policies can be created once and used by IT staff with minimal effort to deploy servers. The result is improved productivity and compliance and lower risk of failures due to inconsistent configuration.

Cisco Intersight will be integrated with data center, hybrid cloud platforms and services to securely deploy and manage infrastructure resources across data center and edge environments. In addition, Cisco will provide future integrations to third-party operations tools to allow customers to use their existing solutions more effectively.

Figure 63 Cisco Intersight Includes a User-Customizable Dashboard; Example of Cisco Intersight Dashboard for FlashStack UCS Domain



Test Setup, Configuration, and Load Recommendation

In this project, we tested a single Cisco UCS B200 M5 blade in a single chassis with multiple workload types. We also ran tests on each workload type full cluster.

Cisco UCS B200 M5 Single Server Testing

Test I

Cisco UCS B200 M5 Blade Server for Single Server Scalability VMware Horizon 7 Remote Desktop Server Hosted Sessions (RDSH) with Windows Server 2016 and Window 10 Persistent/Non-Persistent Recommended Workload

Figure 64 Cisco UCS B200 M5 Blade Server for Single Server Scalability VMware Horizon 7 Remote Desktop Server Hosted Sessions (RDSH) with Windows Server 2016

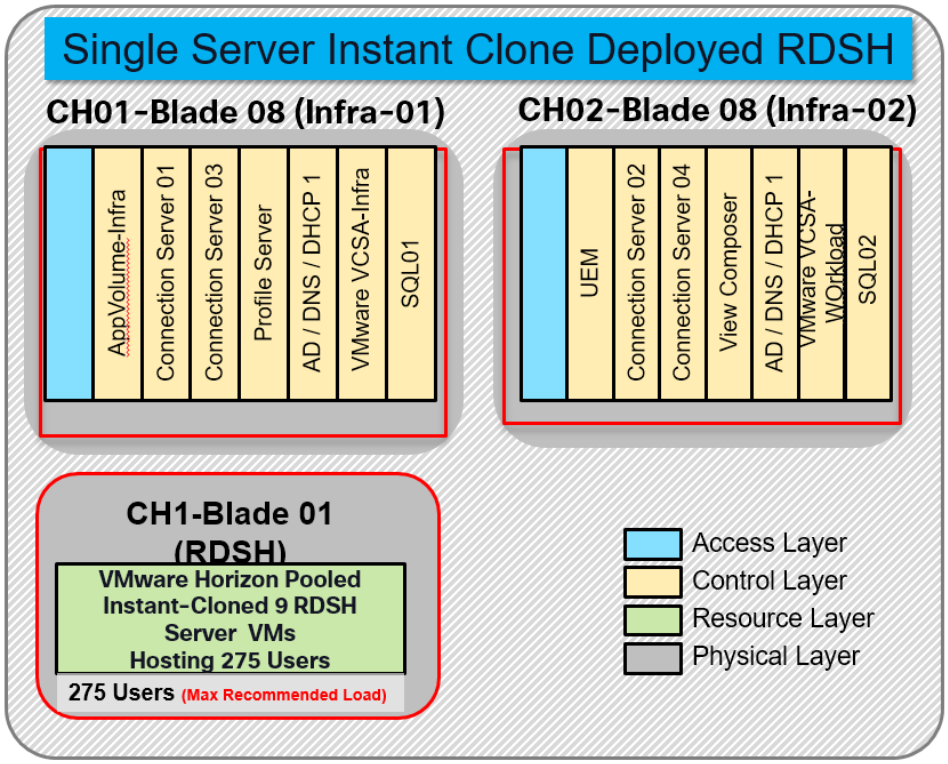


Figure 65 Cisco UCS B200 M5 Blade Server for Single Server Scalability VMware Horizon 7 VDI (Non-Persistent) Instant Clones with Windows 10 64bit OS

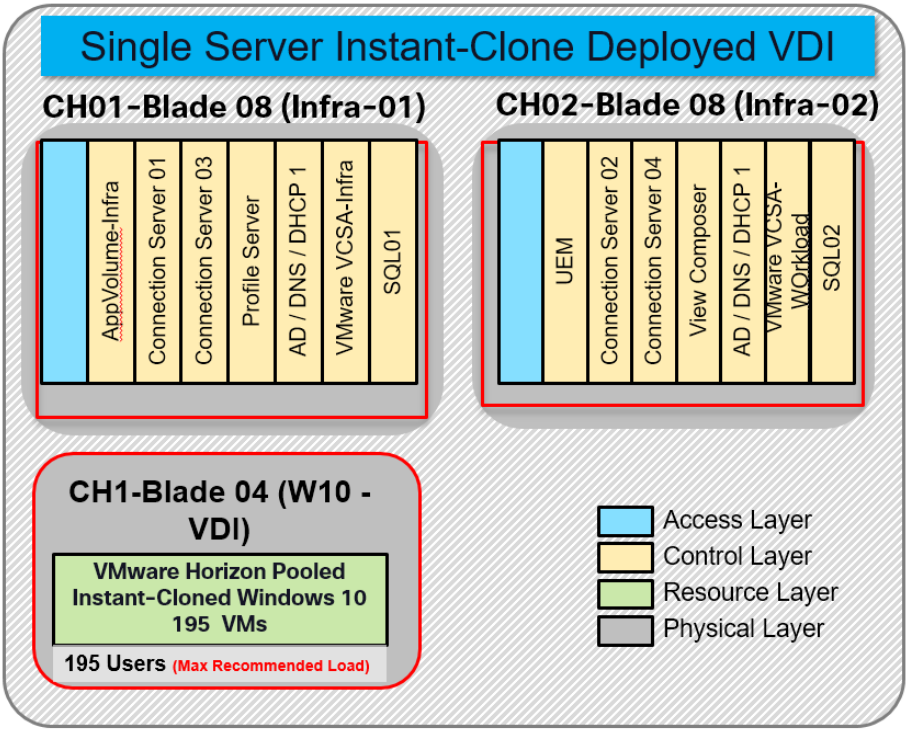


Figure 66 Cisco UCS B200 M5 Blade Server for Single Server Scalability VMware Horizon 7 VDI (Non-Persistent) Linked Clones with Windows 10 64bit OS

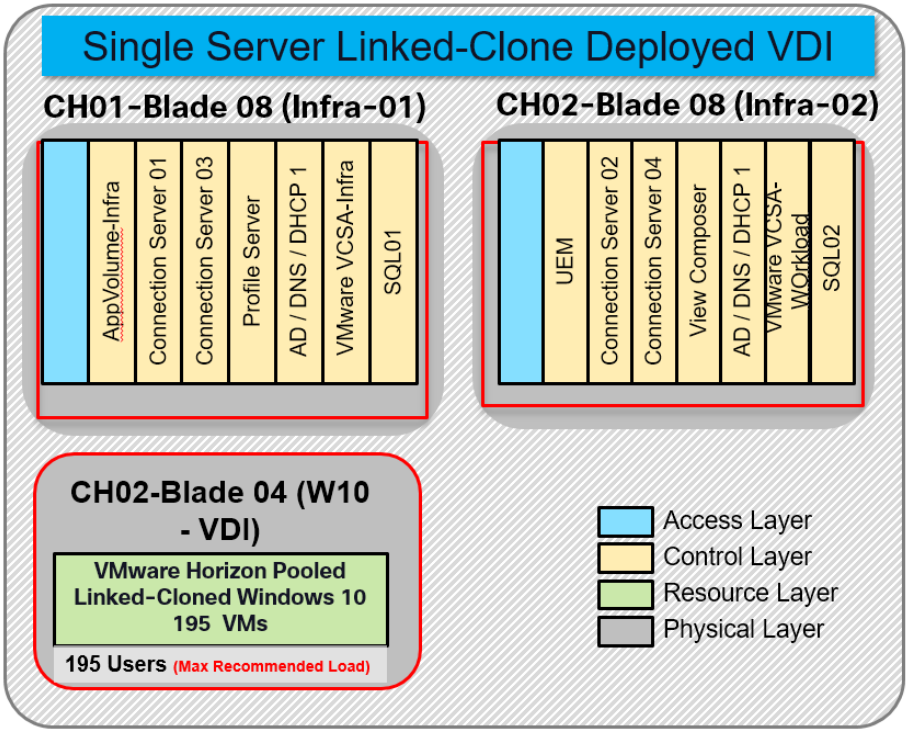
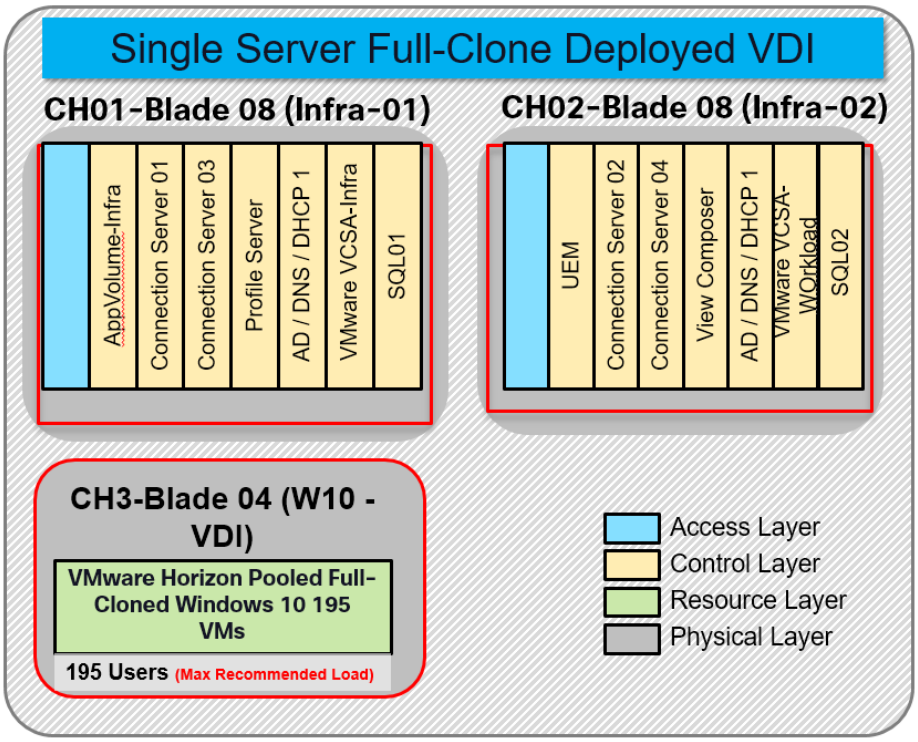


Figure 67 Cisco UCS B200 M5 Blade Server for Single Server Scalability VMware Horizon 7 VDI (Persistent) Full Clones with Windows 10 64bit OS



This test identifies the maximum recommended load a single server can support without compromising end user experience. The value determined is used to size the workload cluster for N+1 server fault tolerance.

Hardware components:

- Cisco UCS 5108 B-Series Server Chassis
- 2 Cisco UCS 6332-16UP Fabric Interconnects
- 2 Cisco UCS B200 M4 Blade Servers (2 Intel Xeon processor E5-2660 v4 CPUs at 2.0 GHz, with 256 GB of memory per blade server [16 GB x 16 DIMMs at 2400 MHz]) for all Infrastructure blades
- Cisco UCS B200 M5 Blade Server (2 Intel Xeon Scalable Family processor Gold 6140 CPUs at 2.3 GHz, with 768GB of memory per blade server [12 x 64 GB DIMMs at 2666 MHz] for workload host blade
- Cisco VIC 1340 CNA (1 per blade)
- 2 Cisco Nexus 9300 Access Switches
- Pure Storage FlashArray //X70 with All-NVMe DirectFlash Modules

Software components:

- Cisco UCS firmware 3.2.2F
- Pure Storage Purity//FA v5.0.2

- VMware ESXi 6.5 Update 1
- VMware Horizon Pooled Desktop Pool with Windows Server 2016 RDSH VMs
- VMware Horizon Pool Windows 10 Desktop Pool for Persistent/Non-Persistent Desktop VMs
- Microsoft SQL Server 2016
- Microsoft Windows 10, 2vCPU, 3GB RAM, 40 GB Disk for Non-Persistent Workload
- Microsoft Windows 10, 2vCPU, 3GB RAM, 100 GB Disk for Persistent Workload
- Microsoft Windows Server 2016, 8vCPU, 32GB RAM, 40 GB Disk for RDSH Workload
- Microsoft Office 2016
- Login VSI 4.1.32

Cisco UCS B200 M5 Configuration for Cluster Testing

Test II

Cisco UCS B200 M5 Blade Server for Cluster Scalability on VMware Horizon 7 Remote Desktop Server Hosted Sessions (RDSH) with Windows Server 2016 and Window 10 Persistent/Non-Persistent Recommended Workload

This test case validates two workload clusters using VMware Horizon 7 with 2430 RDS Hosted Server Sessions and 3570 VDI Instant Clone non- persistent and full clone persistent virtual machines. Server fault tolerance (N+1) is factored into this test scenario for each workload and infrastructure cluster.

Figure 68 RDS Cluster Test Configuration with Ten Blades

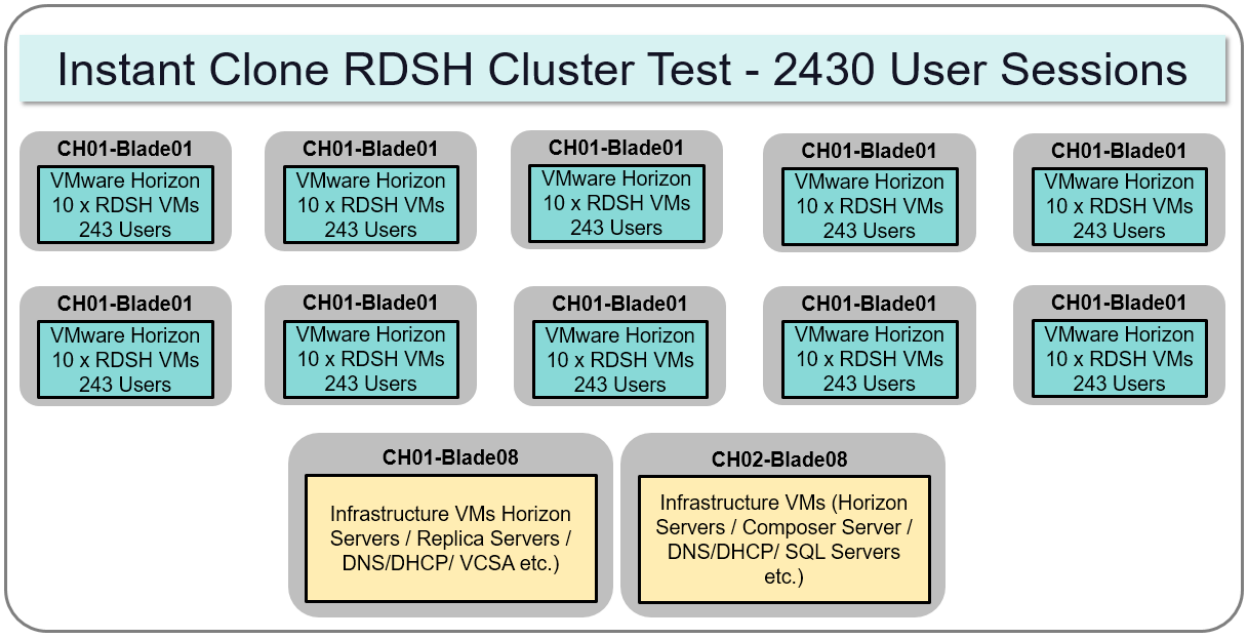
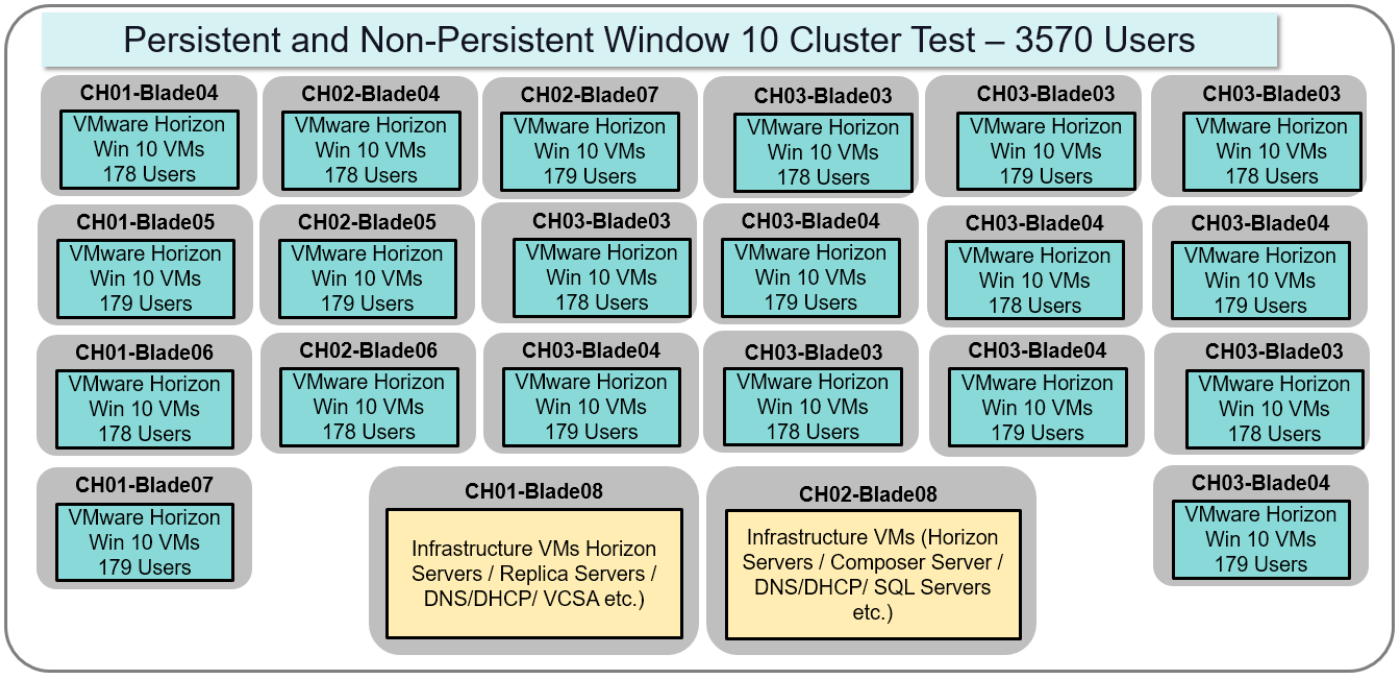


Figure 69 VDI Cluster Test with VDI Persistent/Non-Persistent Cluster Test Configuration with Twenty Blades



Hardware components:

- Cisco UCS 5108 B-Series Server Chassis
- 2 Cisco UCS 6332-16UP Fabric Interconnects
- 2 Cisco UCS B200 M4 Blade Servers (2 Intel Xeon processor E5-2660 v4 CPUs at 2.0 GHz, with 256 GB of memory per blade server [16 GB x 16 DIMMs at 2400 MHz]) for all Infrastructure blades
- Cisco UCS B200 M5 Blade Server (2 Intel Xeon Scalable Family processor Gold 6140 CPUs at 2.3 GHz, with 768GB of memory per blade server [12 x 64 GB DIMMs at 2666 MHz] for workload host blade (Ten for RDSH Cluster test and Twenty for VDI Cluster test)
- Cisco VIC 1340 CNA (1 per blade)
- 2 Cisco Nexus 9300 Access Switches
- Pure Storage FlashArray //X70 with All-NVMe DirectFlash Modules

Software components:

- Cisco UCS firmware 3.2.2f
- Pure Storage Purity//FA v5.0.2
- VMware ESXi 6.5 Update 1
- VMware Horizon Pooled Desktop Pool with Windows Server 2016 RDSH VMs
- VMware Horizon Pool Windows 10 Desktop Pool for Persistent/Non-Persistent Desktop VMs
- Microsoft SQL Server 2016

- Microsoft Windows 10, 2vCPU, 3GB RAM, 40 GB Disk for Non-Persistent Workload
- Microsoft Windows 10, 2vCPU, 3GB RAM, 100 GB Disk for Persistent Workload
- Microsoft Windows Server 2016, 8vCPU, 32GB RAM, 40 GB Disk for RDSH Workload
- Microsoft Office 2016
- Login VSI 4.1.32

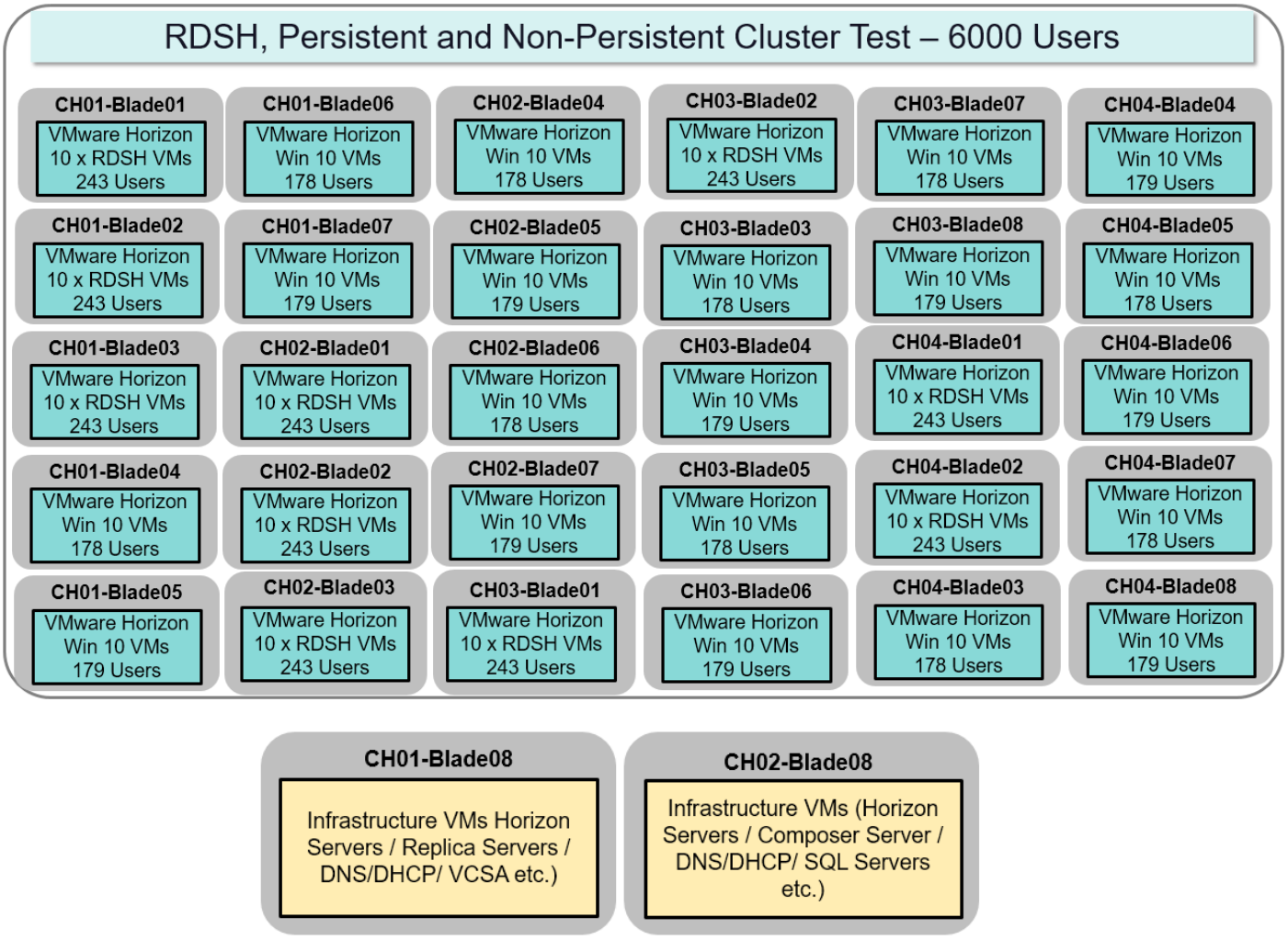
Cisco UCS Configuration for Full Scale Testing

Test III

Cisco UCS Configuration for Full Scale Testing

This test case validates thirty blades mixed workloads using VMware Horizon 7 with 2430 RDS Hosted sessions and 3570 VDI persistent/non-persistent virtual desktops for a total sum of 6,000 users. Server N+1 fault tolerance is factored into this solution for each workload and infrastructure cluster.

Figure 70 Full Scale Test Configuration with Thirty Blades



Hardware components:

- Cisco UCS 5108 B-Series Server Chassis
- 2 Cisco UCS 6332-16UP Fabric Interconnects
- 2 Cisco UCS B200 M4 Blade Servers (2 Intel Xeon processor E5-2660 v4 CPUs at 2.0 GHz, with 256 GB of memory per blade server [16 GB x 16 DIMMs at 2400 MHz]) for all Infrastructure blades
- Thirty Cisco UCS B200 M5 Blade Server (2 Intel Xeon Scalable Family processor Gold 6140 CPUs at 2.3 GHz, with 768GB of memory per blade server [12 x 64 GB DIMMs at 2666 MHz] for workload host blade
- Cisco VIC 1340 CNA (1 per blade)
- 2 Cisco Nexus 9300 Access Switches
- Pure Storage FlashArray //X70 with All-NVMe DirectFlash Modules

Software components:

- Cisco UCS firmware 3.2.2f
- Pure Storage Purity//FA v5.0.2
- VMware ESXi 6.5 Update 1
- VMware Horizon Pooled Desktop Pool with Windows Server 2016 RDSH VMs
- VMware Horizon Pool Windows 10 Desktop Pool for Persistent/Non-Persistent Desktop VMs
- Microsoft SQL Server 2016
- Microsoft Windows 10, 2vCPU, 3GB RAM, 40 GB Disk for Non-Persistent Workload
- Microsoft Windows 10, 2vCPU, 3GB RAM, 100 GB Disk for Persistent Workload
- Microsoft Windows Server 2016, 8vCPU, 32GB RAM, 40 GB Disk for RDSH Workload
- Microsoft Office 2016
- Login VSI 4.1.32

Testing Methodology and Success Criteria

All validation testing was conducted on-site within the Cisco labs in San Jose, California.

The testing results focused on the entire process of the virtual desktop lifecycle by capturing metrics during the desktop boot-up, user logon and virtual desktop acquisition (also referred to as ramp-up,) user workload execution (also referred to as steady state), and user logoff for the RDSH Servers Session under test.

Test metrics were gathered from the virtual desktop, storage, and load generation software to assess the overall success of an individual test cycle. Each test cycle was not considered passing unless all of the planned test users completed the ramp-up and steady state phases (described below) and unless all metrics were within the permissible thresholds as noted as success criteria.

Three successfully completed test cycles were conducted for each hardware configuration and results were found to be relatively consistent from one test to the next.

You can obtain additional information and a free test license from http://www.loginvsi.com

Testing Procedure

The following protocol was used for each test cycle in this study to insure consistent results.

Pre-Test Setup for Single and Multi-Blade Testing

All machines were shut down utilizing the VMware Horizon 7 Administrator Console.

All Launchers for the test were shut down. They were then restarted in groups of 10 each minute until the required number of launchers was running with the Login VSI Agent at a “waiting for test to start” state.

Test Run Protocol

To simulate severe, real-world environments, Cisco requires the log-on and start-work sequence, known as Ramp Up, to complete in 48 minutes. Additionally, we require all sessions started, whether 285 single server users or 6000 full scale test users, to become active within two minutes after the last session is launched.

In addition, Cisco requires that the Login VSI Benchmark method is used for all single server and scale testing. This assures that our tests represent real-world scenarios. For each of the three consecutive runs on single server tests, the same process was followed.

Complete the following steps:

Time 0:00:00 Start PerfMon Logging on the following systems:

- Infrastructure and VDI Host Blade servers used in test run
- All Infrastructure VMs used in test run (AD, SQL, Horizon Connection brokers, Horizon Composer, etc.)
- Time 0:00:10 Start Storage Partner Performance Logging on Storage System.
- Time 0:05: Boot RDS and/or VDI Machines using VMware Horizon 7 Administrator Console.

- Time 0:06 First machines boot.
- Time 0:35 Single Server or Scale target number of RDS Servers and/or VDI Desktop VMs reports in available state in Horizon Administrator Console.

No more than 60 Minutes of rest time is allowed after the last desktop is registered and available on VMware Horizon 7 Administrator Console dashboard. Typically a 20-30 minute rest period for Windows 10 desktops and 10 minutes for RDS VMs is sufficient.

- Time 1:35 Start Login VSI 4.1.32.1 Knowledge Worker Benchmark Mode Test, setting auto-logoff time at 900 seconds, with Single Server or Scale target number of desktop VMs utilizing sufficient number of Launchers (at 20-25 sessions/Launcher).
- Time 2:23 Single Server or Scale target number of desktop VMs desktops launched (48 minute benchmark launch rate).
- Time 2:25 All launched sessions must become active.

All sessions launched must become active for a valid test run within this window.

- Time 2:40 Login VSI Test Ends (based on Auto Logoff 900 Second period designated above).
- Time 2:55 All active sessions logged off.

All sessions launched and active must be logged off for a valid test run. The VMware Horizon 7 Administrator Dashboard must show that all desktops have been returned to the registered/available state as evidence of this condition being met.

- Time 2:57 All logging terminated; Test complete.
- Time 3:15 Copy all log files off to archive; Set virtual desktops to maintenance mode through broker; Shutdown all Windows 10 machines.
- Time 3:30 Reboot all hypervisors.
- Time 3:45 Ready for new test sequence.

Success Criteria

Our “pass” criteria for this testing follows:

Cisco will run tests at a session count level that effectively utilizes the blade capacity measured by CPU utilization, memory utilization, storage utilization, and network utilization. We will use Login VSI to launch version 4.1.5 Knowledge Worker workloads. The number of launched sessions must equal active sessions within two minutes of the last session launched in a test as observed on the VSI Management console.

The VMware Horizon Administrator Console or Horizon Connection Server Console must be monitored throughout the steady state to make sure of the following:

- All running sessions report In Use throughout the steady state

- No sessions move to unregistered, unavailable or available state at any time during steady state
- Within 20 minutes of the end of the test, all sessions on all launchers must have logged out automatically and the Login VSI Agent must have shut down.

Cisco requires three consecutive runs with results within +/-1% variability to pass the Cisco Validated Design performance criteria. For white papers written by partners, two consecutive runs within +/-1% variability are accepted. (All test data from partner run testing must be supplied along with proposed white paper.)

We will publish Cisco Validated Designs with our recommended workload following the process above and will note that we did not reach a VSImax dynamic in our testing FlashStack with Cisco UCS B200 M5 and VMware Horizon 7 on VMware ESXi 6.5 Update 1 Test Results.

The purpose of this testing is to provide the data needed to validate VMware Horizon Remote Desktop Session Hosted (RDSH) server sessions and VMware Horizon Virtual Desktop (VDI) models with VMware Horizon Composer 7 using ESXi, vCenter to virtualize Microsoft Windows 10 desktops and Microsoft Windows Server 2016 sessions on Cisco UCS B200 M5 Blade Servers using a Pure Storage FlashArray //X70 storage system.

The information contained in this section provides data points that a customer may reference in designing their own implementations. These validation results are an example of what is possible under the specific environment conditions outlined here, and do not represent the full characterization of VMware Horizon products with VMware vSphere.

Three test sequences, each containing three consecutive test runs generating the same result, were performed to establish single blade performance and multi-blade, linear scalability.

VSImax 4.1.x Description

The philosophy behind Login VSI is different to conventional benchmarks. In general, most system benchmarks are steady state benchmarks. These benchmarks execute one or multiple processes, and the measured execution time is the outcome of the test. Simply put: the faster the execution time or the bigger the throughput, the faster the system is according to the benchmark.

Login VSI is different in approach. Login VSI is not primarily designed to be a steady state benchmark (however, if needed, Login VSI can act like one). Login VSI was designed to perform benchmarks for SBC or VDI workloads through system saturation. Login VSI loads the system with simulated user workloads using well known desktop applications like Microsoft Office, Internet Explorer and Adobe PDF reader. By gradually increasing the amount of simulated users, the system will eventually be saturated. When the system is saturated, the response time of the applications will increase significantly. This latency in application response times show a clear indication whether the system is (close to being) overloaded. As a result, by nearly overloading a system it is possible to find out what its true maximum user capacity is.

After a test is performed, the response times can be analyzed to calculate the maximum active session/desktop capacity. Within Login VSI this is calculated as VSImax. When the system is coming closer to its saturation point, response times will rise. When reviewing the average response time it will be clear the response times escalate at saturation point.

This VSImax is the “Virtual Session Index (VSI).” With Virtual Desktop Infrastructure (VDI) and Terminal Services (RDS) workloads this is valid and useful information. This index simplifies comparisons and makes it possible to understand the true impact of configuration changes on hypervisor host or guest level.

Server-Side Response Time Measurements

It is important to understand why specific Login VSI design choices have been made. An important design choice is to execute the workload directly on the target system within the session instead of using remote sessions. The scripts simulating the workloads are performed by an engine that executes workload scripts on every target system, and are initiated at logon within the simulated user's desktop session context.

An alternative to the Login VSI method would be to generate user actions client side through the remoting protocol. These methods are always specific to a product and vendor dependent. More importantly, some protocols simply do not have a method to script user actions client side.

For Login VSI the choice has been made to execute the scripts completely server side. This is the only practical and platform independent solution, for a benchmark like Login VSI.

Calculating VSImax v4.1.x

The simulated desktop workload is scripted in a 48 minute loop when a simulated Login VSI user is logged on, performing generic Office worker activities. After the loop is finished it will restart automatically. Within each loop the response times of sixteen specific operations are measured in a regular interval: sixteen times in within each loop. The response times of these five operations are used to determine VSImax.

The five operations from which the response times are measured are:

- Notepad File Open (NFO)

  Loading and initiating VSINotepad.exe and opening the openfile dialog. This operation is handled by the OS and by the VSINotepad.exe itself through execution. This operation seems almost instant from an end-user's point of view.

- Notepad Start Load (NSLD)

  Loading and initiating VSINotepad.exe and opening a file. This operation is also handled by the OS and by the VSINotepad.exe itself through execution. This operation seems almost instant from an end-user's point of view.

- Zip High Compression (ZHC)

  This action copy's a random file and compresses it (with 7zip) with high compression enabled. The compression will very briefly spike CPU and disk IO.

- Zip Low Compression (ZLC)

  This action copy's a random file and compresses it (with 7zip) with low compression enabled. The compression will very briefly disk IO and creates some load on the CPU.

- CPU

  Calculates a large array of random data and spikes the CPU for a short period of time.

These measured operations within Login VSI do hit considerably different subsystems such as CPU (user and kernel), Memory, Disk, the OS in general, the application itself, print, GDI, etc. These operations are specifically short by nature. When such operations become consistently long: the system is saturated because of excessive queuing on any kind of resource. As a result, the average response times will then

escalate. This effect is clearly visible to end-users. If such operations consistently consume multiple seconds the user will regard the system as slow and unresponsive.

Figure 71 Sample of a VSI Max Response Time Graph, Representing a Normal Test

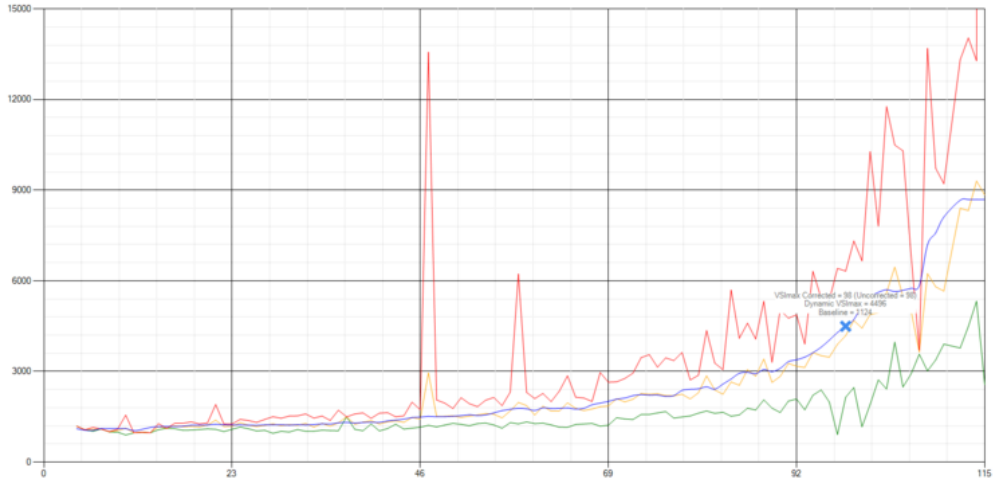
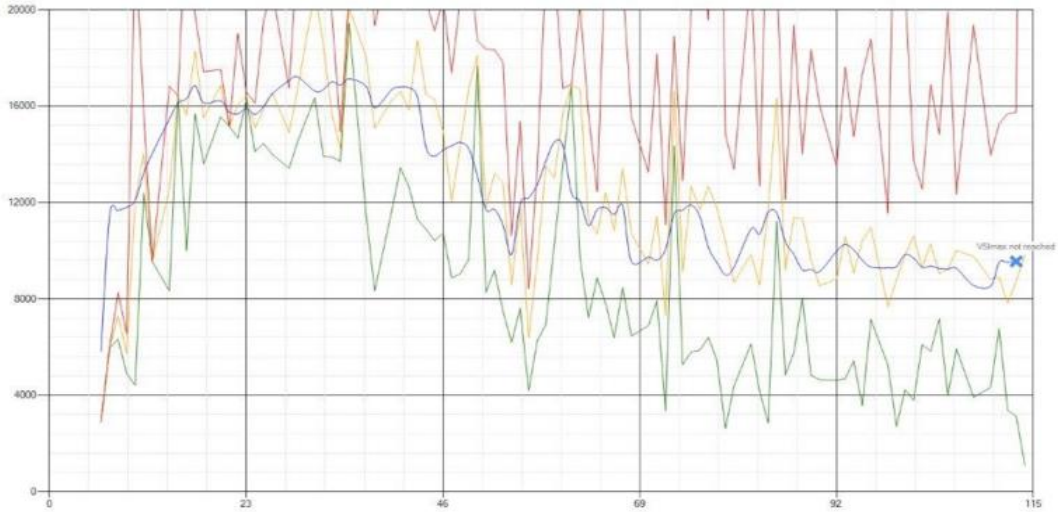


Figure 72 Sample of a VSI Test Response Time Graph (where there was an obvious performance issue)



When the test is finished, VSImax can be calculated. When the system is not saturated, and it could complete the full test without exceeding the average response time latency threshold, VSImax is not reached and the amount of sessions ran successfully.

The response times are very different per measurement type, for instance Zip with compression can be around 2800 ms, while the Zip action without compression can only take 75ms. These response time of these actions are weighted before they are added to the total. This ensures that each activity has an equal impact on the total response time.

In comparison to previous VSImax models, this weighting much better represent system performance. All actions have very similar weight in the VSImax total. The following weighting of the response times are applied.

The following actions are part of the VSImax v4.1 calculation and are weighted as follows (US notation):

- Notepad File Open (NFO): 0.75
- Notepad Start Load (NSLD): 0.2
- Zip High Compression (ZHC): 0.125
- Zip Low Compression (ZLC): 0.2
- CPU: 0.75

This weighting is applied on the baseline and normal Login VSI response times.

With the introduction of Login VSI 4.1 we also created a new method to calculate the base phase of an environment. With the new workloads (Taskworker, Powerworker, etc.) enabling 'basephase' for a more reliable baseline has become obsolete. The calculation is explained below. In total 15 lowest VSI response time samples are taken from the entire test, the lowest 2 samples are removed and the 13 remaining samples are averaged. The result is the Baseline. In short:

- Take the lowest 15 samples of the complete test

- From those 15 samples remove the lowest 2
- Average the 13 results that are left is the baseline

The VSImax average response time in Login VSI 4.1.x is calculated on the amount of active users that are logged on the system.

Always a 5 Login VSI response time samples are averaged + 40 percent of the amount of “active” sessions. For example, if the active sessions is 60, then latest 5 + 24 (=40 percent of 60) = 31 response time measurement are used for the average calculation.

To remove noise (accidental spikes) from the calculation, the top 5 percent and bottom 5 percent of the VSI response time samples are removed from the average calculation, with a minimum of 1 top and 1 bottom sample. As a result, with 60 active users, the last 31 VSI response time sample are taken. From those 31 samples the top 2 samples are removed and lowest 2 results are removed (5 percent of 31 = 1.55, rounded to 2). At 60 users the average is then calculated over the 27 remaining results.

VSImax v4.1.x is reached when the VSIbase + a 1000 ms latency threshold is not reached by the average VSI response time result. Depending on the tested system, VSImax response time can grow 2 - 3x the baseline average. In end-user computing, a 3x increase in response time in comparison to the baseline is typically regarded as the maximum performance degradation to be considered acceptable.

In VSImax v4.1.x this latency threshold is fixed to 1000ms, this allows better and fairer comparisons between two different systems, especially when they have different baseline results. Ultimately, in VSImax v4.1.x, the performance of the system is not decided by the total average response time, but by the latency is has under load. For all systems, this is now 1000ms (weighted).

The threshold for the total response time is: average weighted baseline response time + 1000ms.

When the system has a weighted baseline response time average of 1500ms, the maximum average response time may not be greater than 2500ms (1500+1000). If the average baseline is 3000 the maximum average response time may not be greater than 4000ms (3000+1000).

When the threshold is not exceeded by the average VSI response time during the test, VSImax is not hit and the amount of sessions ran successfully. This approach is fundamentally different in comparison to previous VSImax methods, as it was always required to saturate the system beyond VSImax threshold.

Lastly, VSImax v4.1.x is now always reported with the average baseline VSI response time result. For example: “The VSImax v4.1 was 125 with a baseline of 1526ms”. This helps considerably in the comparison of systems and gives a more complete understanding of the system. The baseline performance helps to understand the best performance the system can give to an individual user. VSImax indicates what the total user capacity is for the system. These two are not automatically connected and related.

When a server with a very fast dual core CPU, running at 3.6 GHZ, is compared to a 10 core CPU, running at 2,26 GHZ, the dual core machine will give an individual user better performance than the 10 core machine. This is indicated by the baseline VSI response time. The lower this score is, the better performance an individual user can expect.

However, the server with the slower 10 core CPU will easily have a larger capacity than the faster dual core system. This is indicated by VSImax v4.1.x, and the higher VSImax is, the larger overall user capacity can be expected.

With Login VSI 4.1.x a new VSImax method is introduced: VSImax v4.1. This methodology gives much better insight in system performance and scales to extremely large systems.

Test Results

For both the VMware Horizon 7 RDS Hosted Virtual Desktops and VDI virtual machines use cases, a recommended maximum workload was determined that was based on both LoginVSI Knowledge worker workload with flash end user experience measures and blade server operating parameters.

This recommended maximum workload approach allows you to determine the server N+1 fault tolerance load the blade can successfully support in the event of a server outage for maintenance or upgrade.

Our recommendation is that the Login VSI Average Response and VSI Index Average should not exceed the Baseline plus 1680 milliseconds to insure that end user experience is outstanding. Additionally, during steady state, the processor utilization should average no more than 90-95 percent.

Memory should never be oversubscribed for Desktop Virtualization workloads.

Callouts have been added throughout the data charts to indicate each phase of testing.

Table 17 Callouts for Test Results and Terminology

Test Phase	Description
Boot	Start all RDS and VDI virtual machines at the same time
Logon	The Login VSI phase of test is where sessions are launched and start executing the workload over a 48 minutes duration
Steady state	The steady state phase is where all users are logged in and performing various workload tasks such as using Microsoft Office, Web browsing, PDF printing, playing videos, and compressing files (typically for 15 minute duration)
Logoff	Sessions finish executing the Login VSI workload and logoff

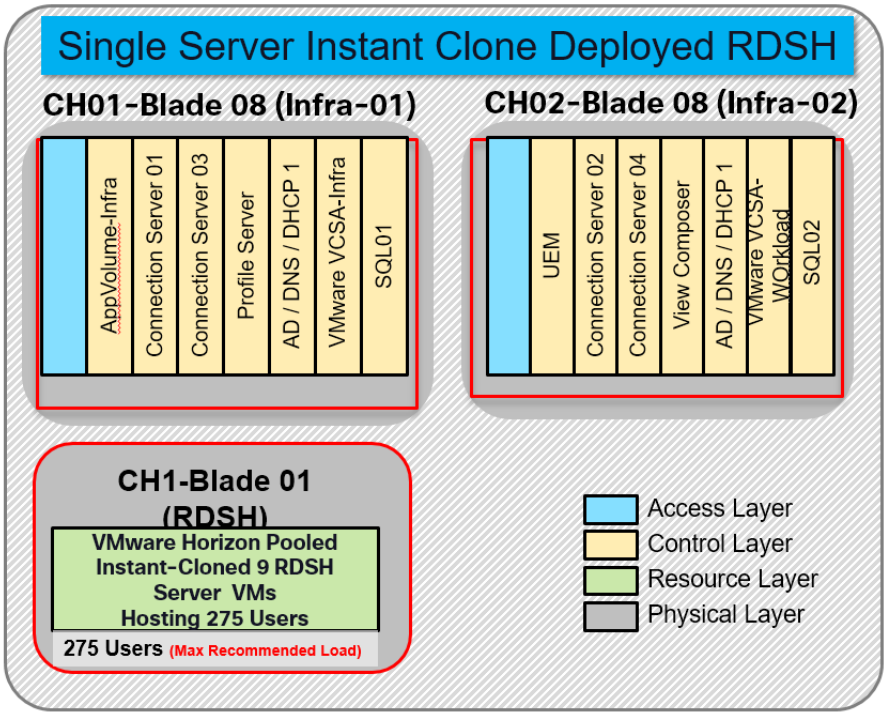
Single-Server Recommended Maximum Workload Testing

This section details the key performance metrics that were captured on the Cisco UCS host blades during the single server testing to determine the Recommended Maximum Workload per host server. The single server testing comprised of four tests:

- 275 RDS Hosted server sessions
- 195 VDI non-persistent Instant-Clone Desktop VMs
- 195 VDI non-persistent Linked-Clone Desktop VMs
- 195 VDI persistent Full-Clone Desktop VMs

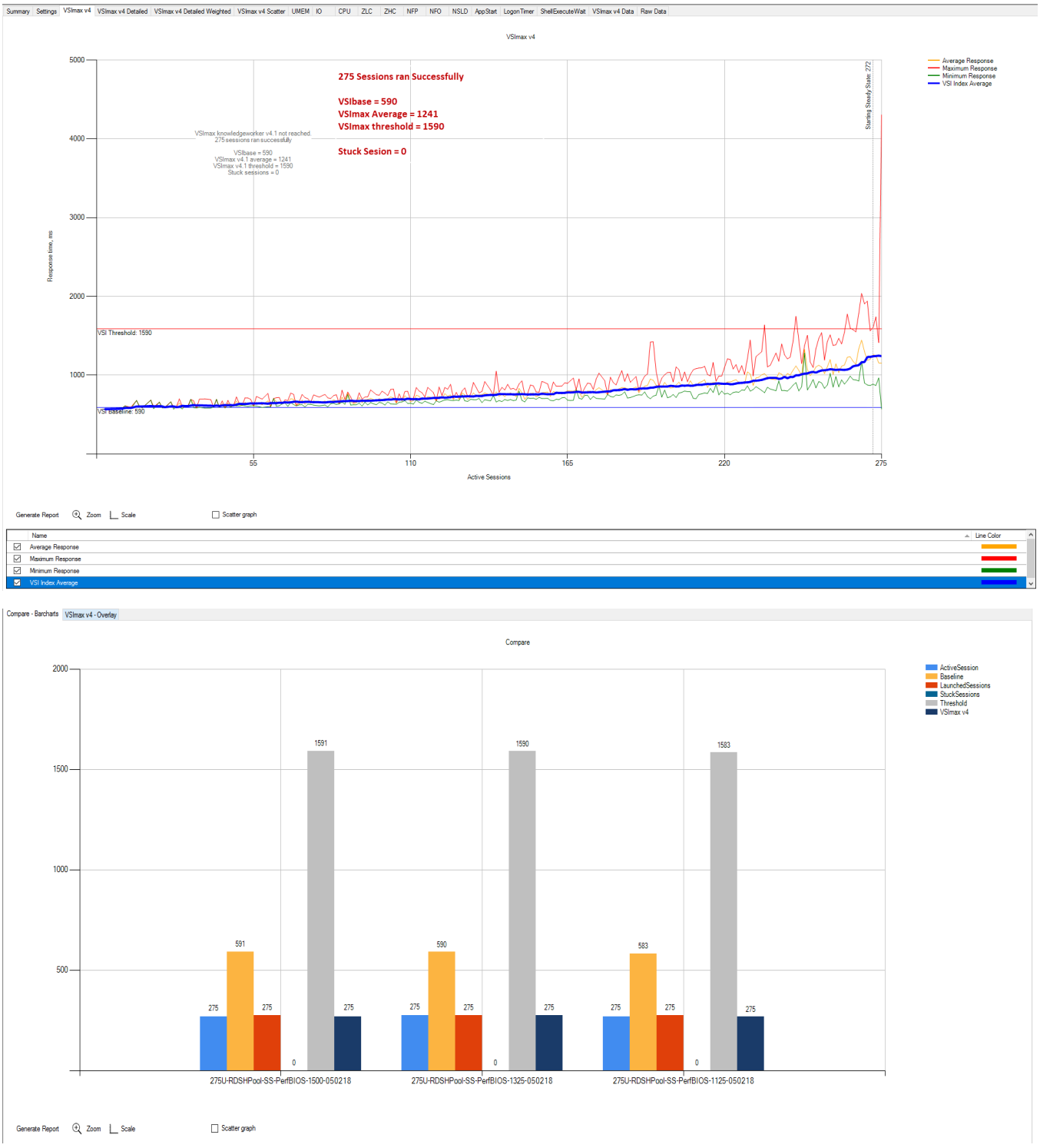
Single-Server Recommended Maximum Workload for RDS Hosted Server Sessions: 275 Users

Figure 73 Single Server Recommended Maximum Workload for RDS with 275 Users



The recommended maximum workload for a Cisco UCS B200 M5 blade server with dual 6140 Gold processors and 768GB of RAM is 275 Server 2016 Remote Desktop Server Hosted Sessions. Each dedicated blade server ran 10 Windows Server 2016 Virtual Machines. Each virtual server was configured with 8 vCPUs and 32GB RAM.

Figure 74 Single Server | VMware Horizon 7 RDS Hosted Sessions | LoginVSI Score



Performance data for the server running the workload is shown below:

Figure 75 Single Server | VMware Horizon 7 RDSH processor | Host CPU Utilization

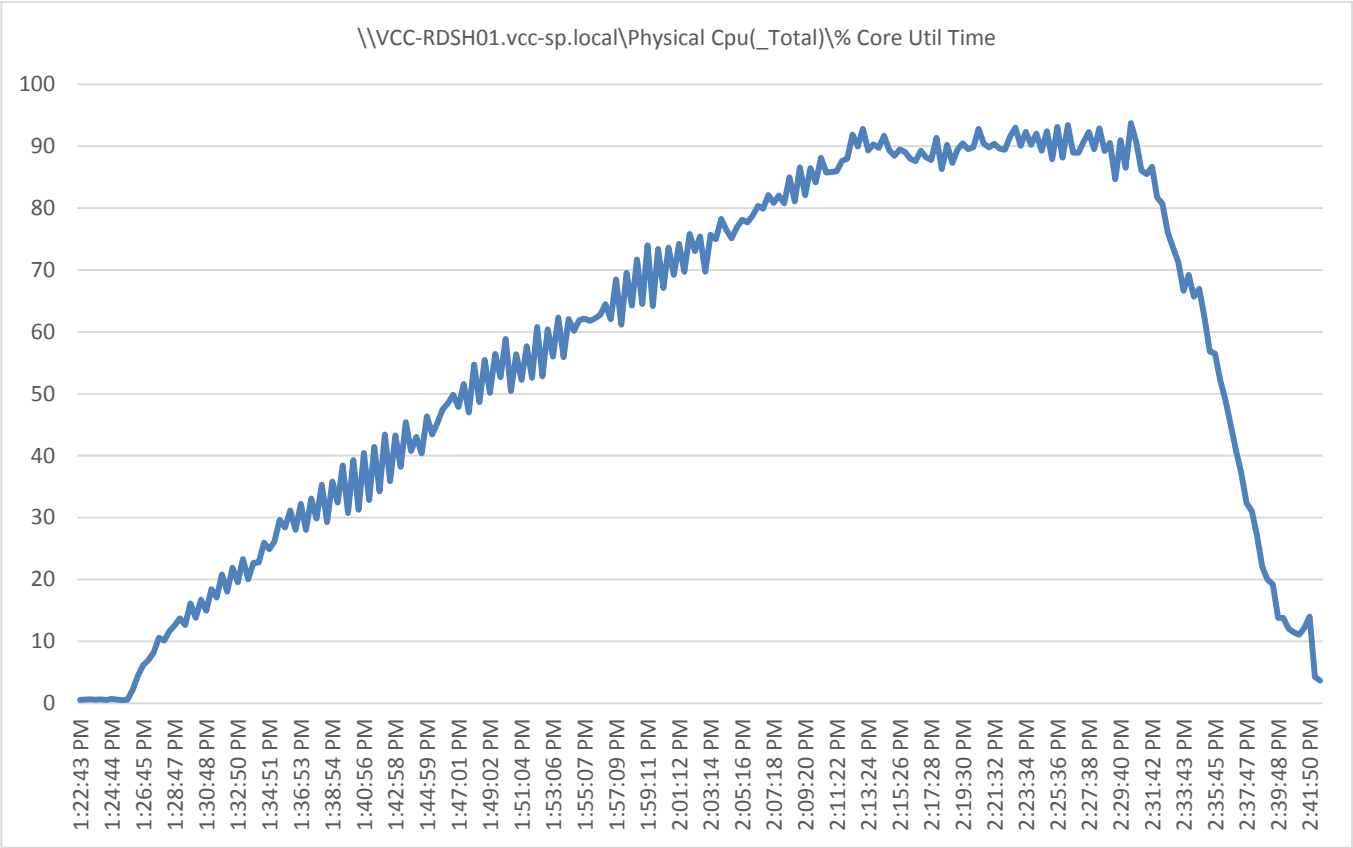

\\VCC-RDSH01.vcc-sp.local\Physical Cpu(_Total)\% Core Util Time
100
90
80
70
60
50
40
30
20
10
0
1:22:43 PM
1:24:44 PM
1:26:45 PM
1:28:47 PM
1:30:48 PM
1:32:50 PM
1:34:51 PM
1:36:53 PM
1:38:54 PM
1:40:56 PM
1:42:58 PM
1:44:59 PM
1:47:01 PM
1:49:02 PM
1:51:04 PM
1:53:06 PM
1:55:07 PM
1:57:09 PM
1:59:11 PM
2:01:12 PM
2:03:14 PM
2:05:16 PM
2:07:18 PM
2:09:20 PM
2:11:22 PM
2:13:24 PM
2:15:26 PM
2:17:28 PM
2:19:30 PM
2:21:32 PM
2:23:34 PM
2:25:36 PM
2:27:38 PM
2:29:40 PM
2:31:42 PM
2:33:43 PM
2:35:45 PM
2:37:47 PM
2:39:48 PM
2:41:50 PM

Figure 76 Single Server | VMware Horizon 7 RDSH | Host Memory Utilization

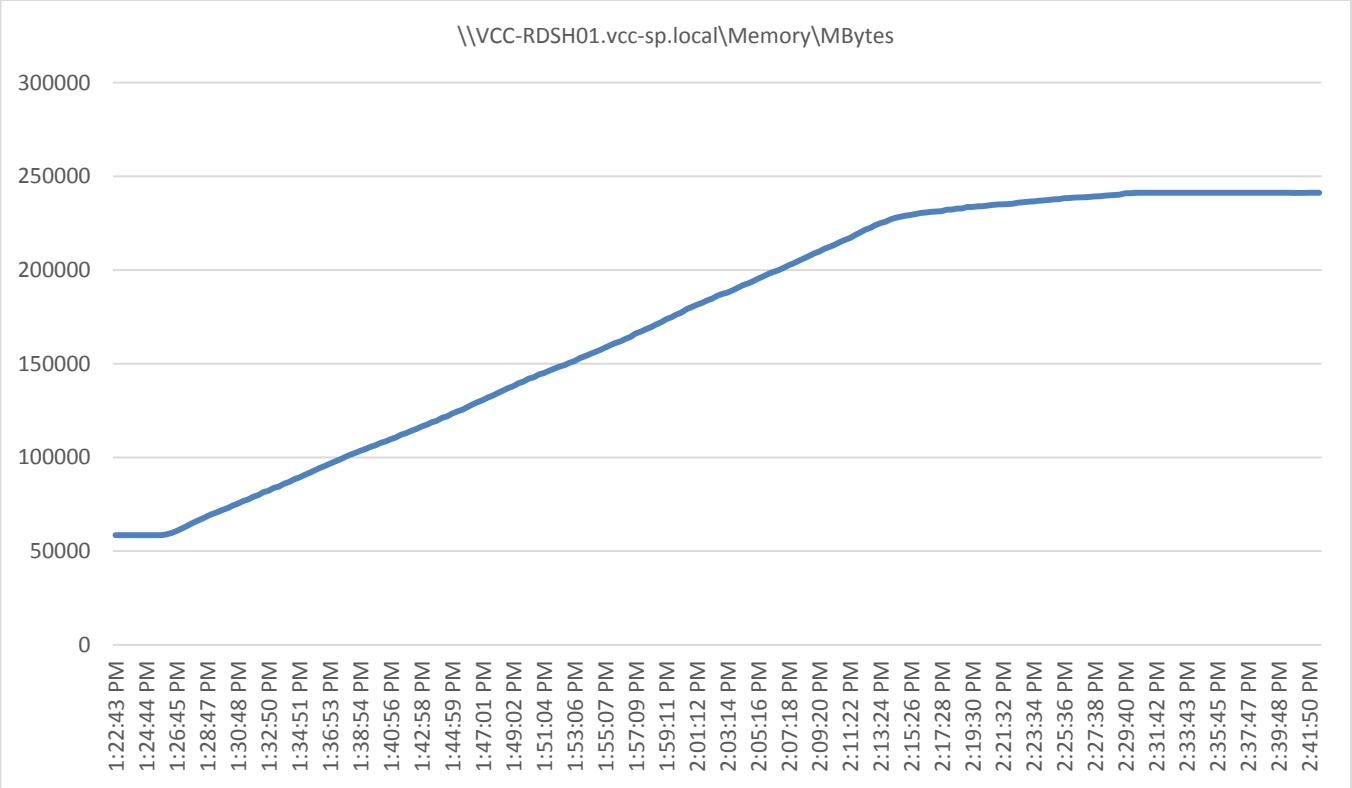
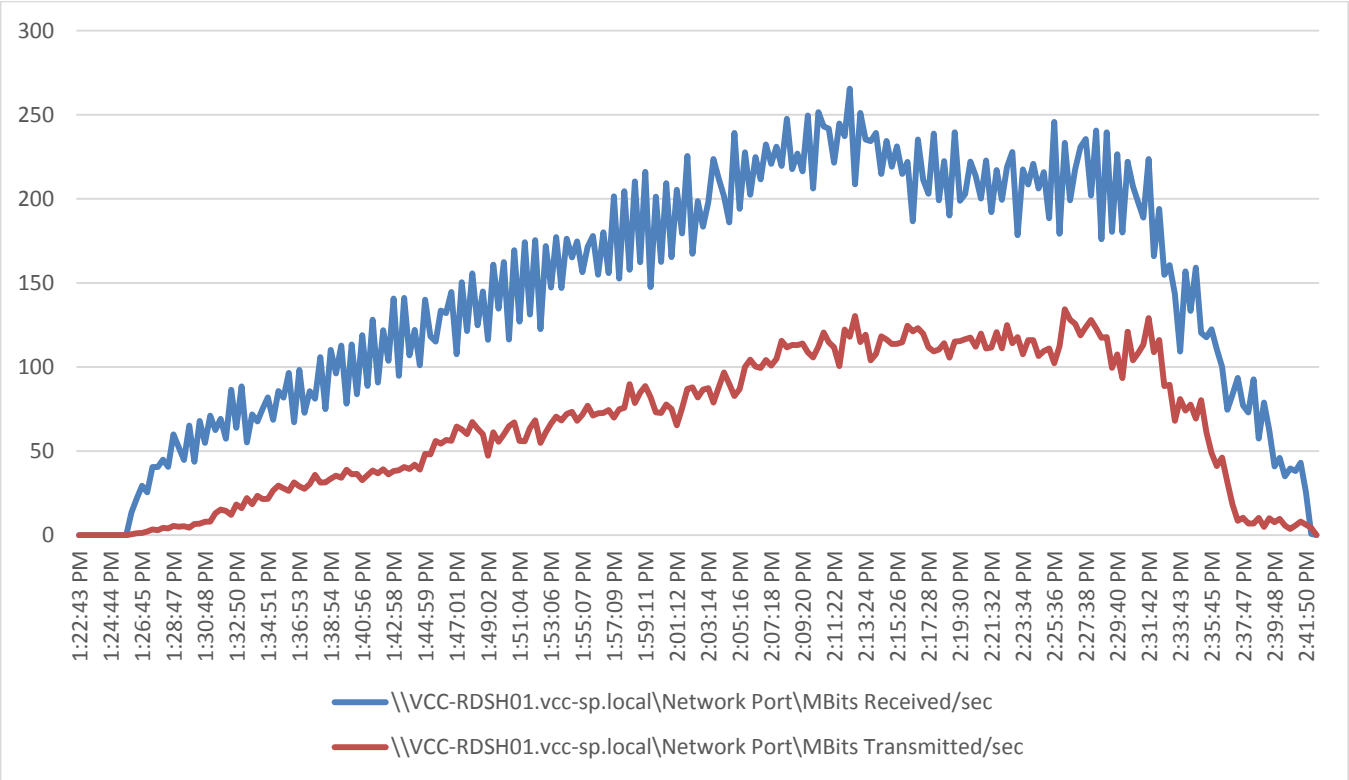
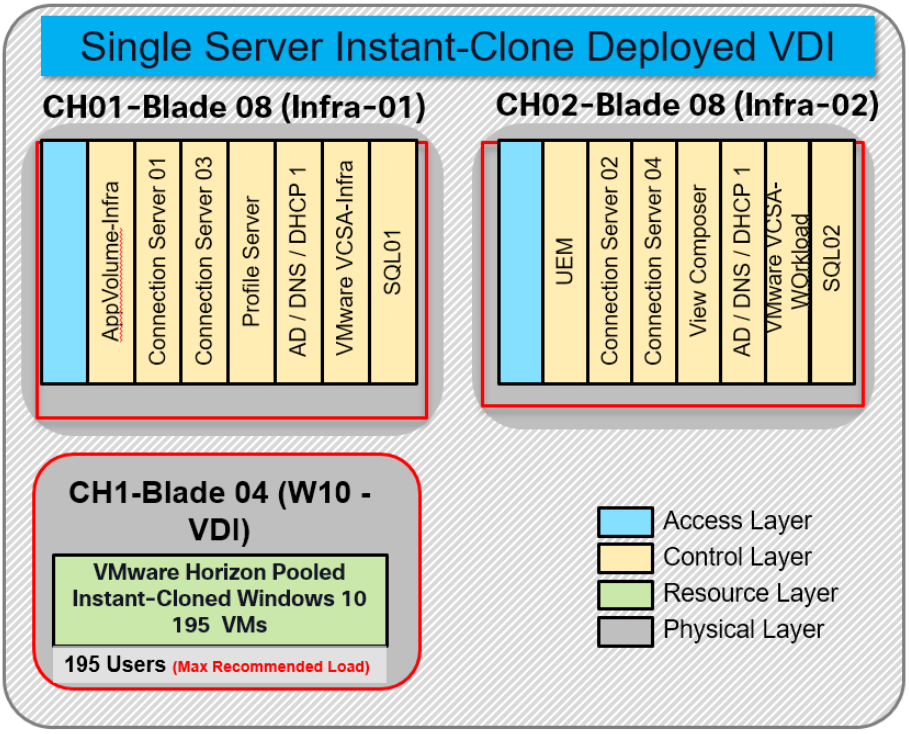


Figure 77 Single Server | VMware Horizon 7 RDSH | Host Network Utilization



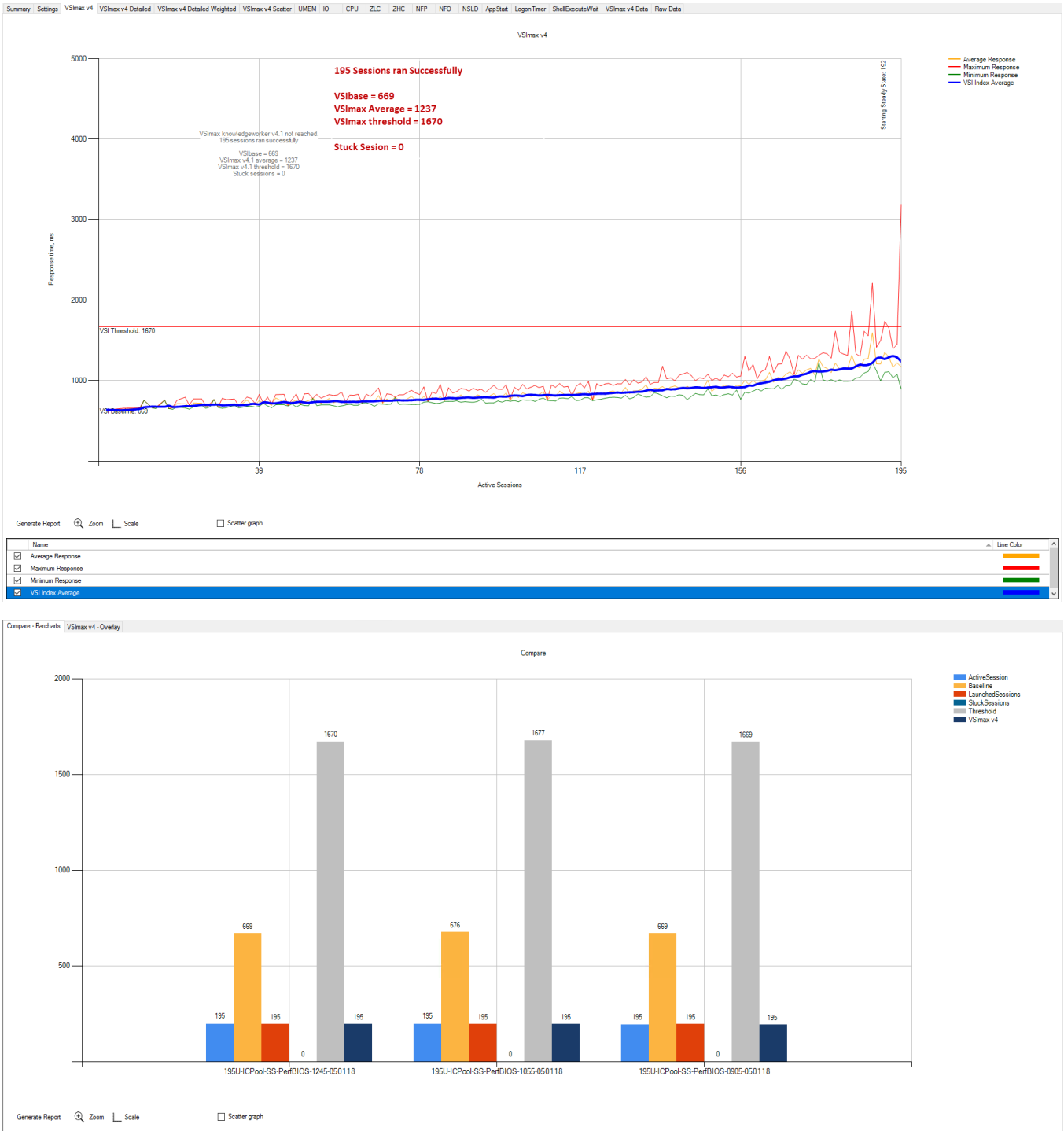
Single-Server Recommended Maximum Workload for Instant-Clone Desktop: 195 Users

Figure 78 Single Server Recommended Maximum Workload for Windows 10 Instant-Clone Desktop: 195 Users



The recommended maximum workload for a Cisco UCS B200 M5 blade server with dual 6140 processors and 768GB of RAM is 195 Windows 10 64-bit floating assigned Instant-Cloned virtual machines with 2 vCPU and 3GB RAM. The Login VSI and blade performance data is shown below.

Figure 79 Single Server | VMware Horizon 7 VDI-Non Persistent Instant-Clone Desktops | VSI Score



Performance data for the server running the workload is shown below:

Figure 80 Single Server | VMware Horizon VDI Non–Persistent Instant-Clone Desktops | Host CPU Utilization

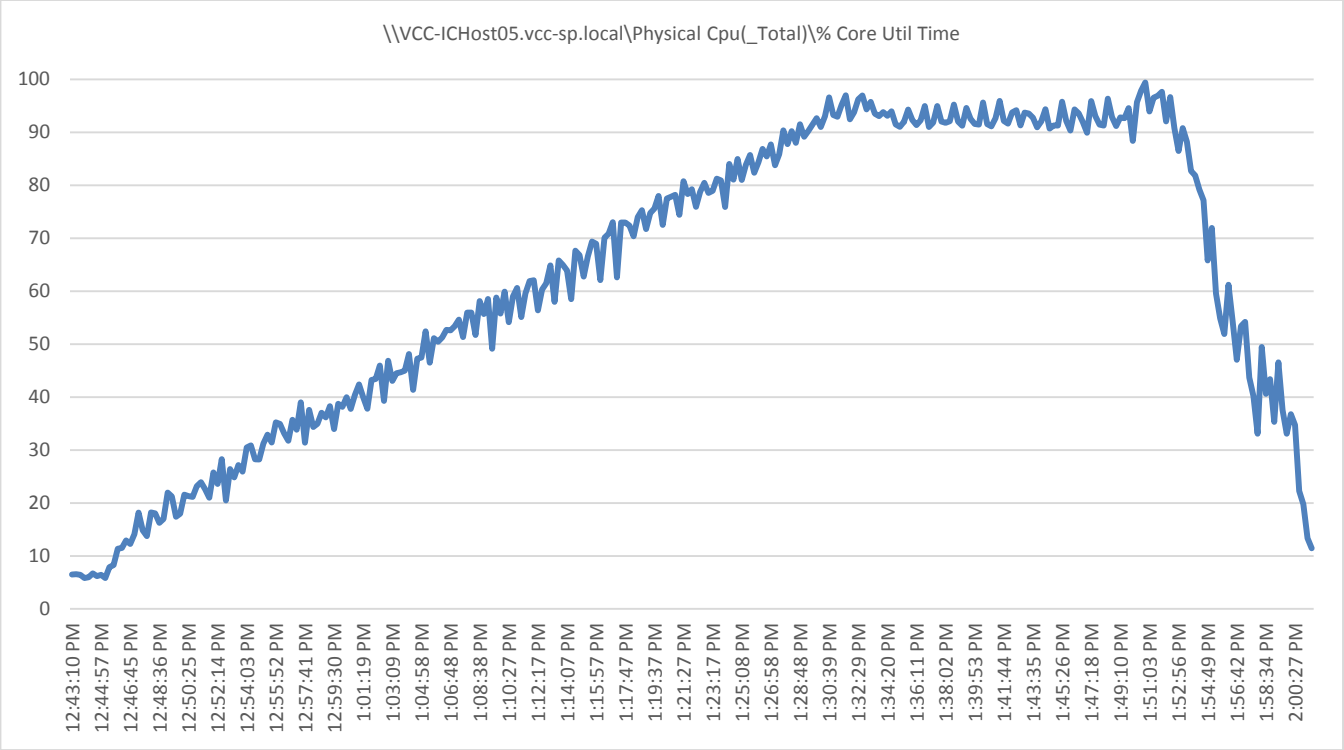


Figure 81 Single Server | VMware Horizon VDI Non –Persistent Instant-Clone Desktops | Host Memory Utilization

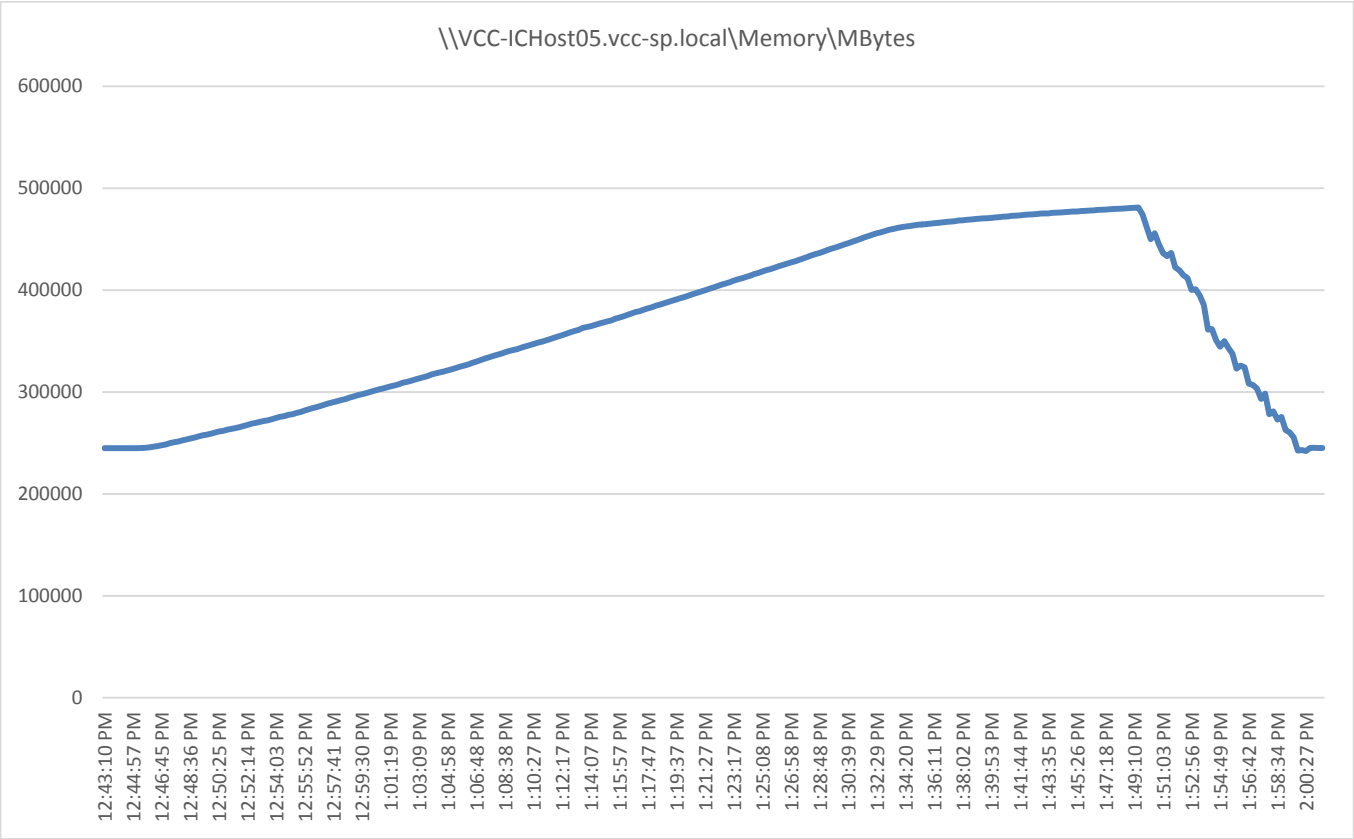
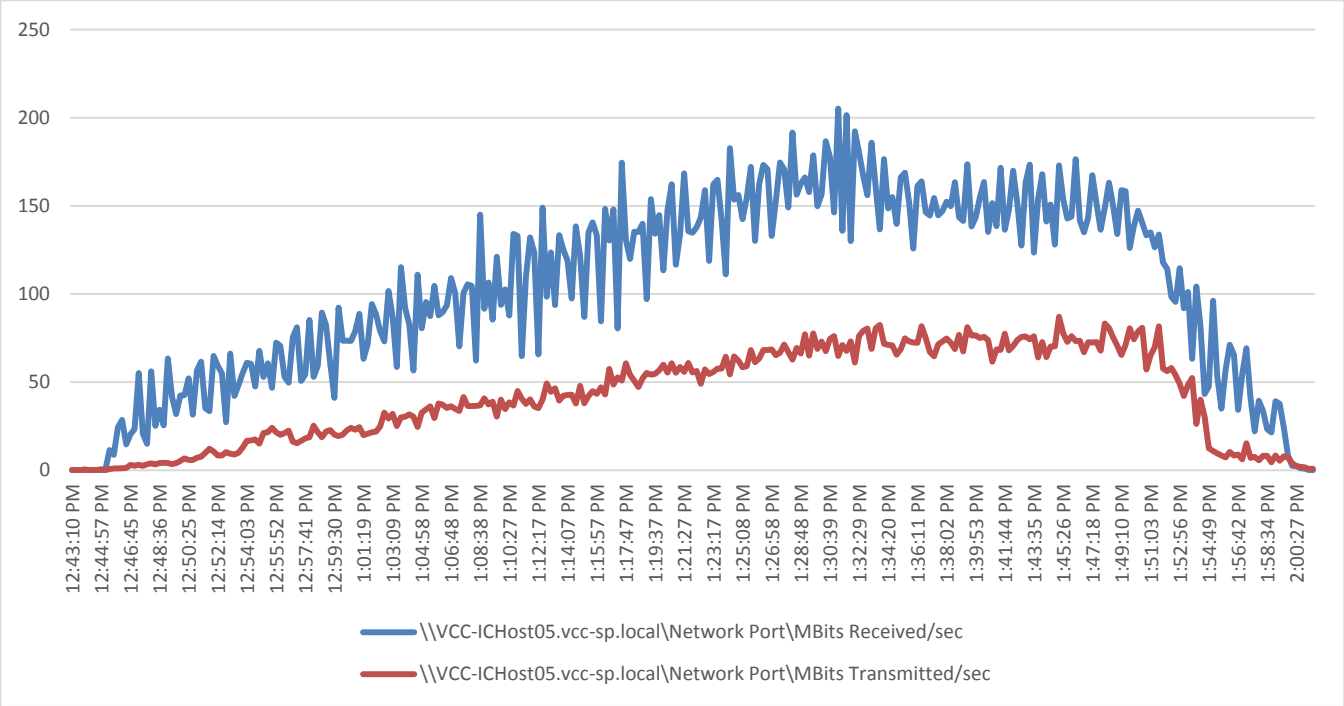
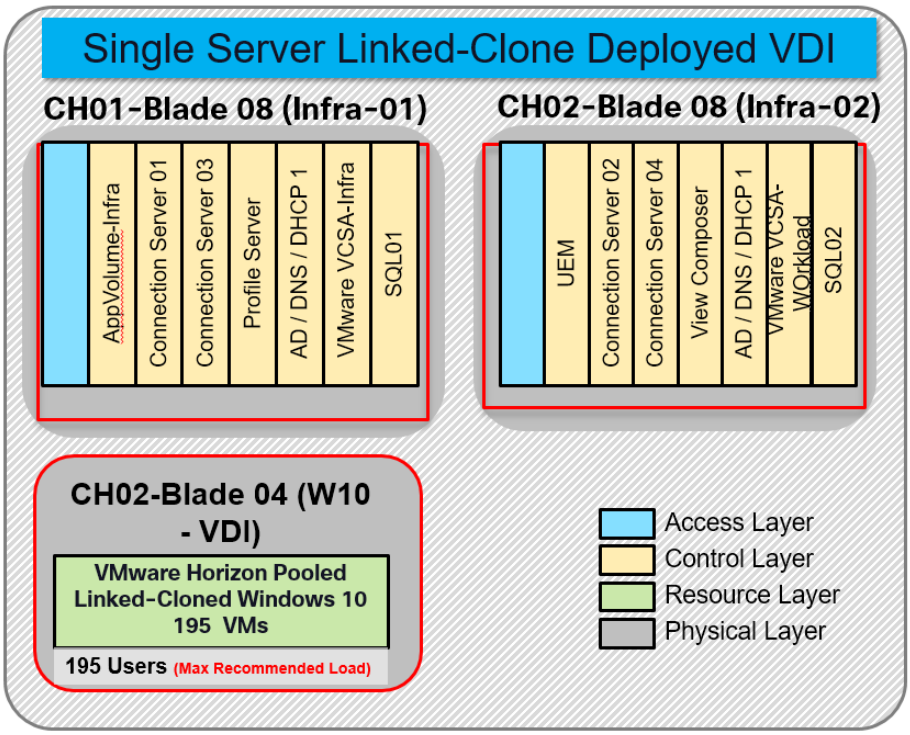


Figure 82 Single Server | VMware Horizon VDI Non –persistent Instant-Clone Desktops | Host Network Utilization



Single-Server Recommended Maximum Workload for Linked-Clone Desktop: 195 Users

Figure 83 Single Server Recommended Maximum Workload for Windows 10 Linked-Clone Desktop: 195 Users



The recommended maximum workload for a Cisco UCS B200 M5 blade server with dual 6140 processors and 768GB of RAM is 195 Windows 10 64-bit floating assigned Linked-Cloned virtual machines with 2 vCPU and 3GB RAM.

Figure 84 Single Server | VMware Horizon 7 VDI-Persistent | VSI Score

Performance data for the server running the workload is shown below:

Figure 85 Single Server | VMware Horizon VDI Non–Persistent Linked-Clone Desktops | Host CPU Utilization

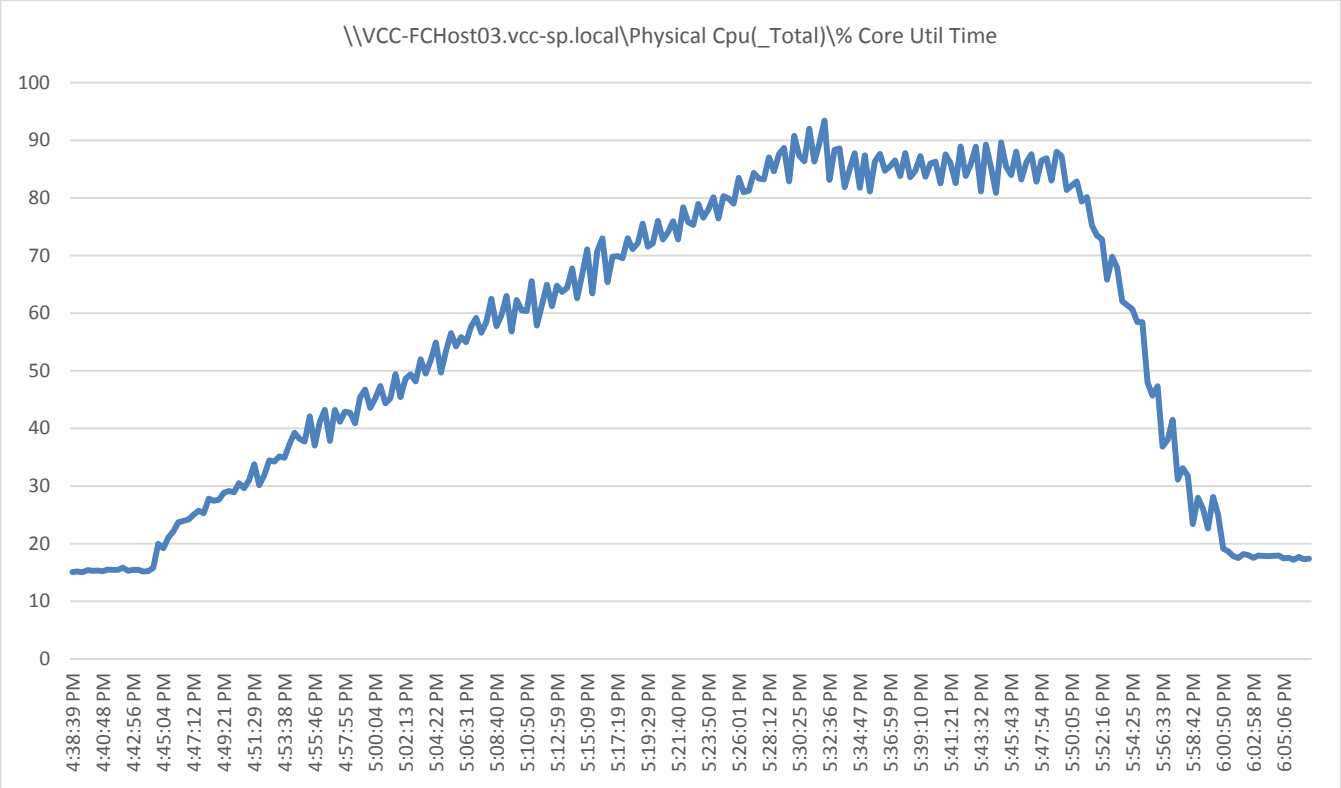


Figure 86 Single Server | VMware Horizon VDI Non –Persistent Linked-Clone Desktops | Host Memory Utilization

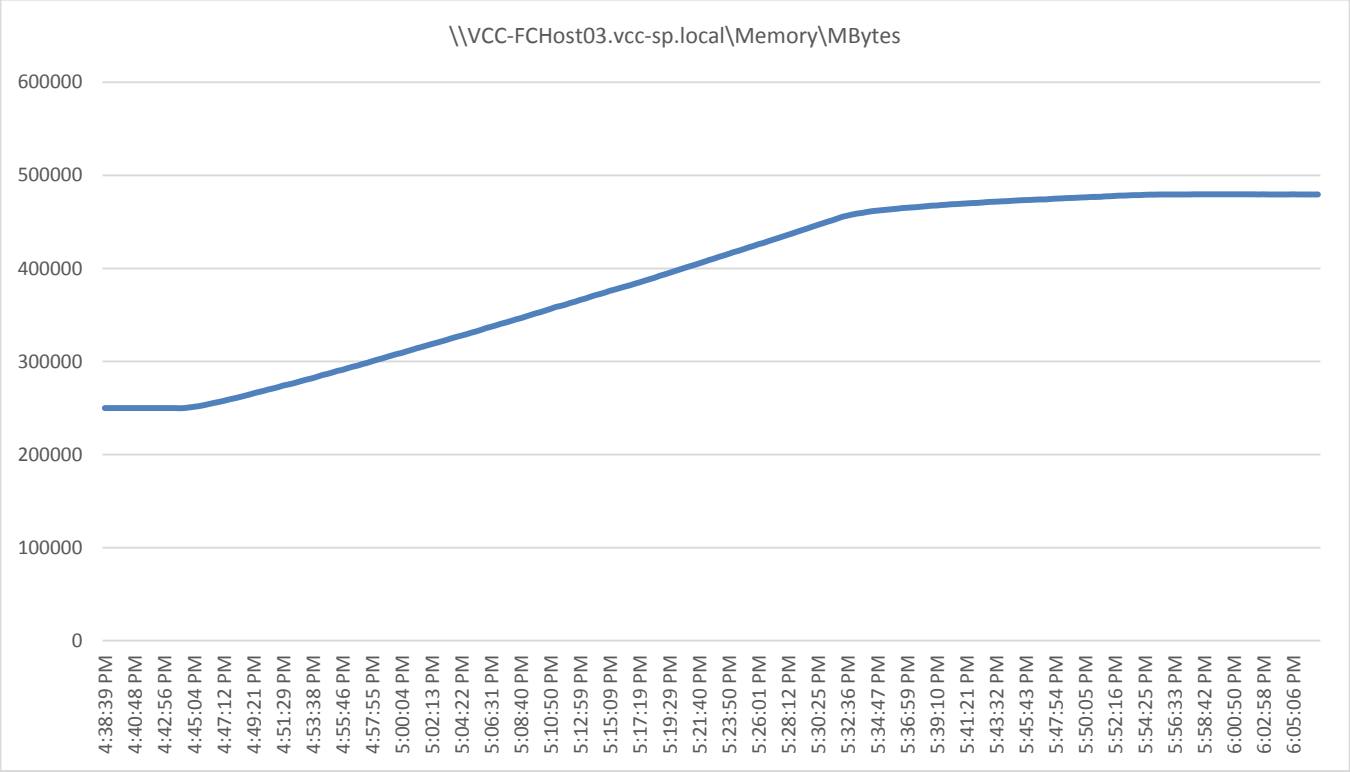
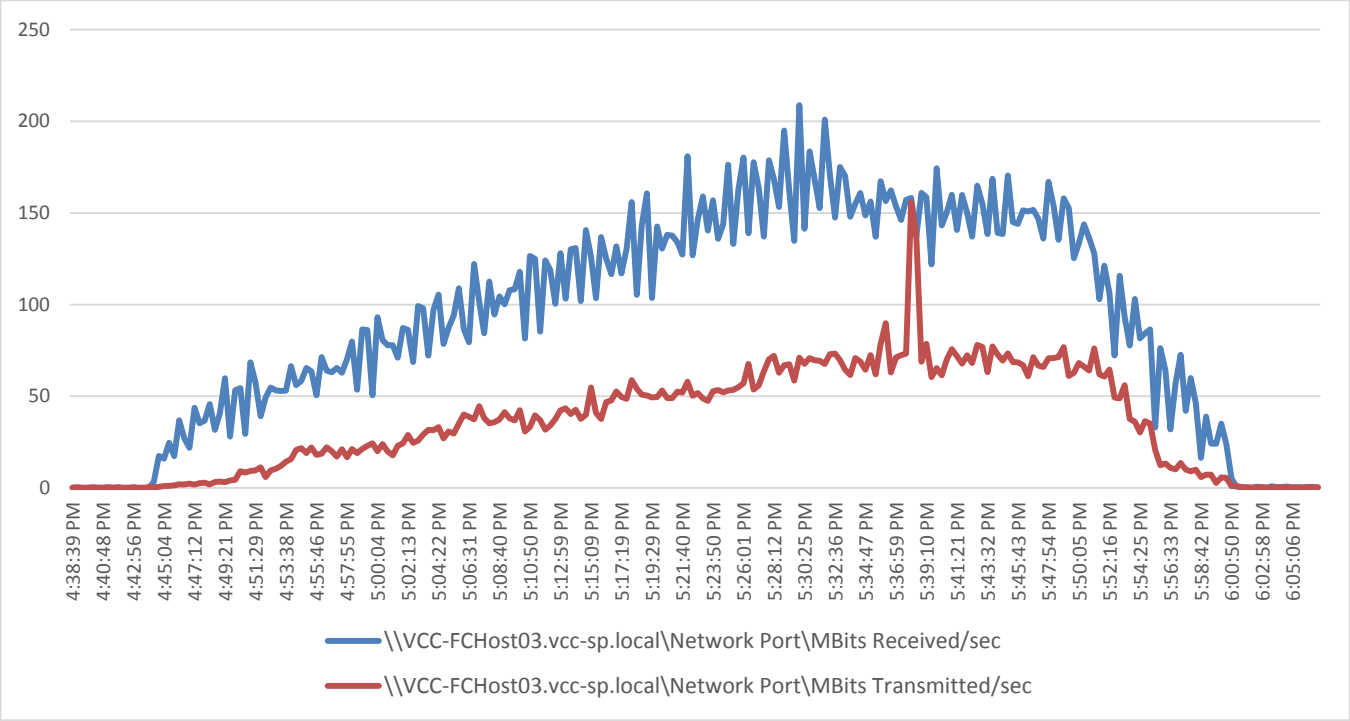
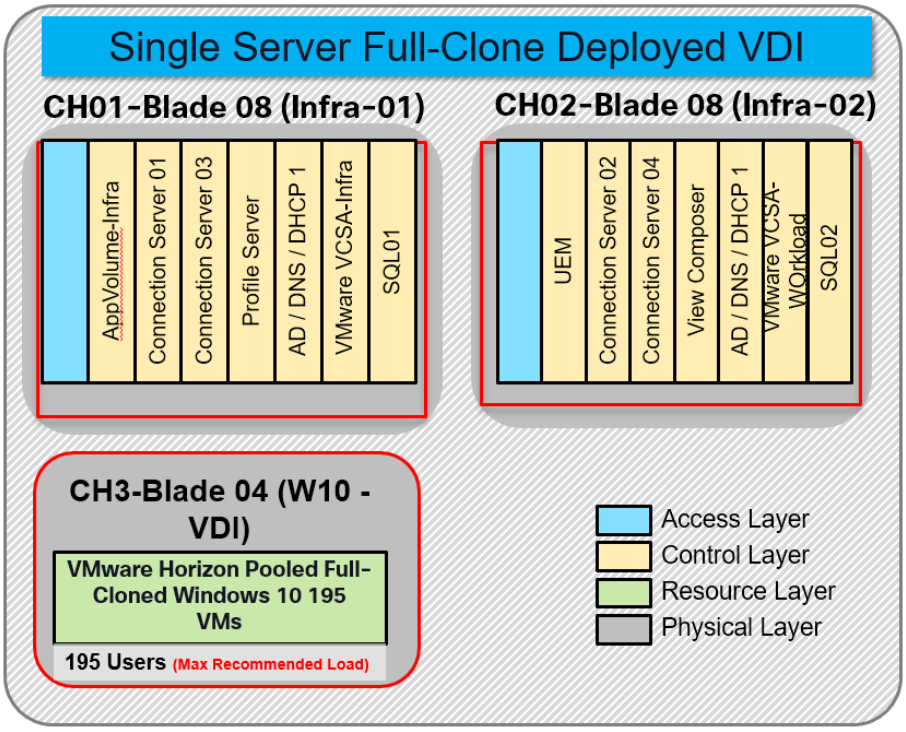


Figure 87 Single Server | VMware Horizon VDI Non –Persistent Linked-Clone Desktops | Host Network Utilization



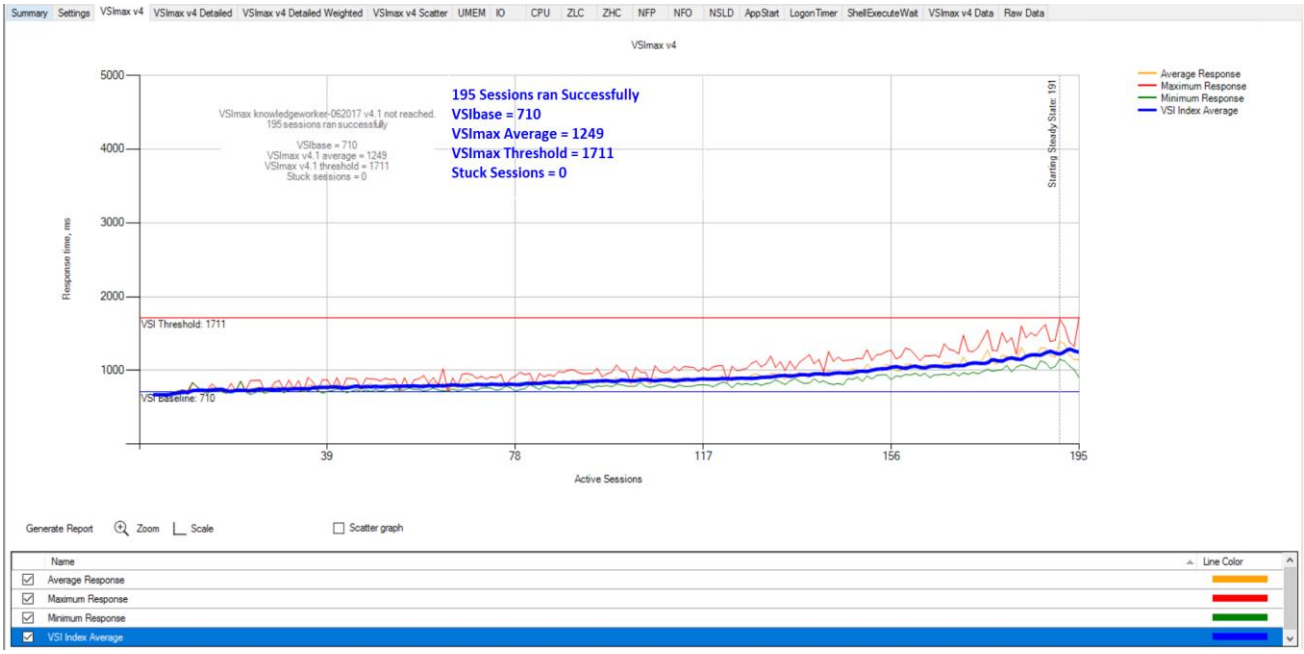
Single-Server Recommended Maximum Workload for Full-Clone Desktop: 195 Users

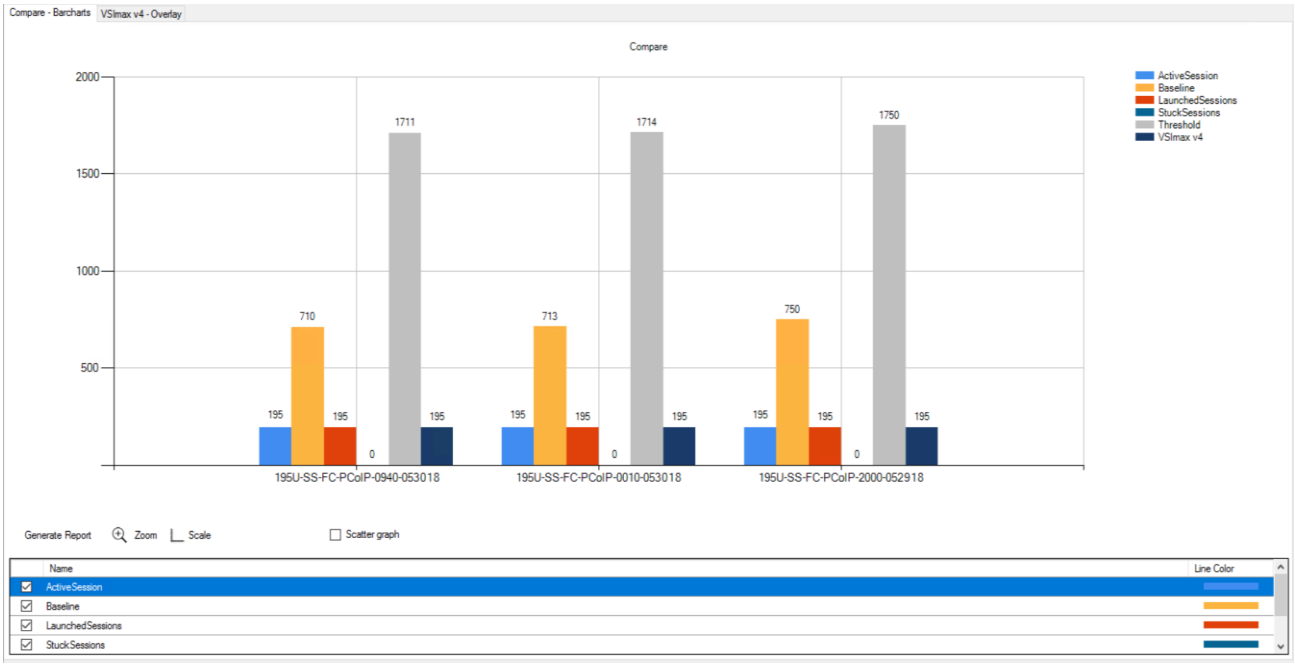
Figure 88 Single Server Recommended Maximum Workload for Windows 10 Full-Clone Desktop: 195 Users



The recommended maximum workload for a Cisco UCS B200 M5 blade server with dual 6140 processors and 768GB of RAM is 195 Windows 10 64-bit floating assigned Full-Cloned virtual machines with 2 vCPU and 3GB RAM.

Figure 89 Single Server | VMware Horizon 7 VDI-Persistent | VSI Score





Performance data for the server running the workload is shown below:

Figure 90 Single Server | VMware Horizon VDI Non–Persistent Full-Clone Desktops | Host CPU Utilization

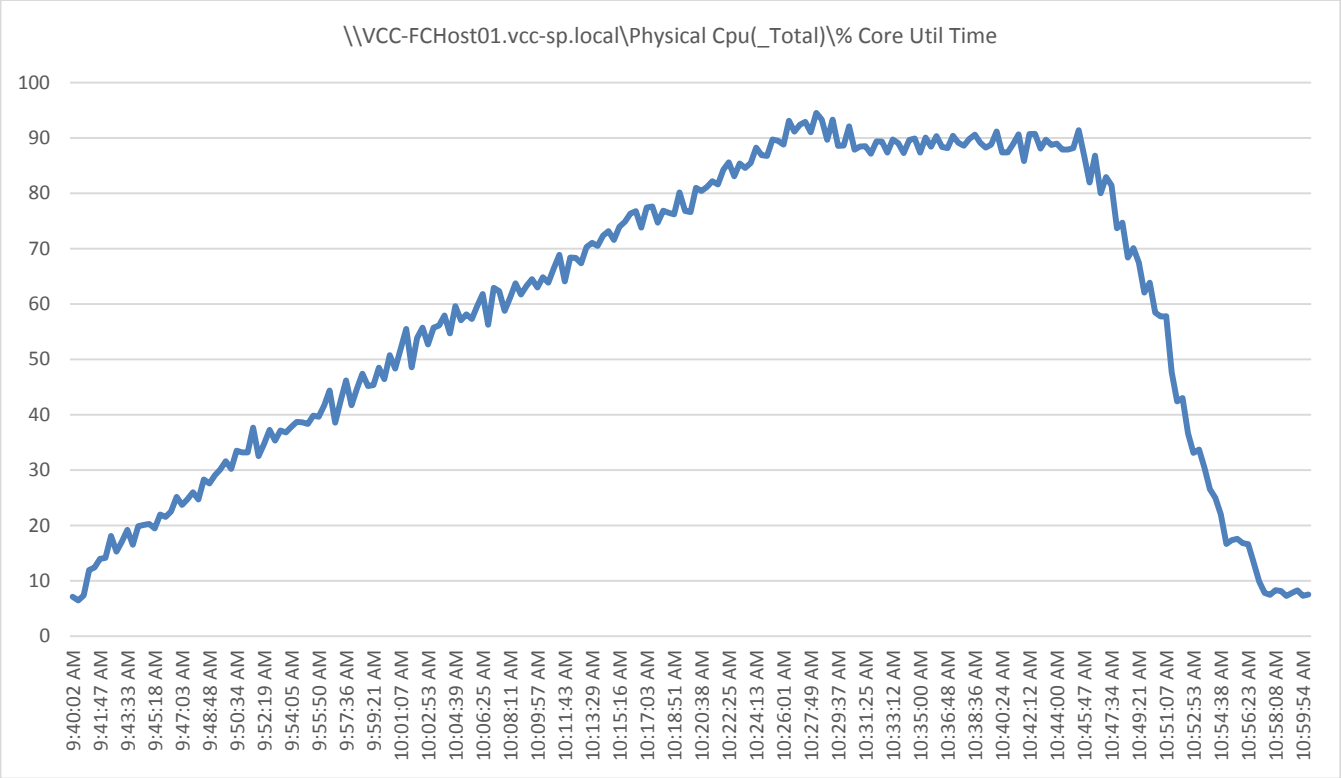


Figure 91 Single Server | VMware Horizon VDI Non –Persistent Full-Clone Desktops | Host Memory Utilization

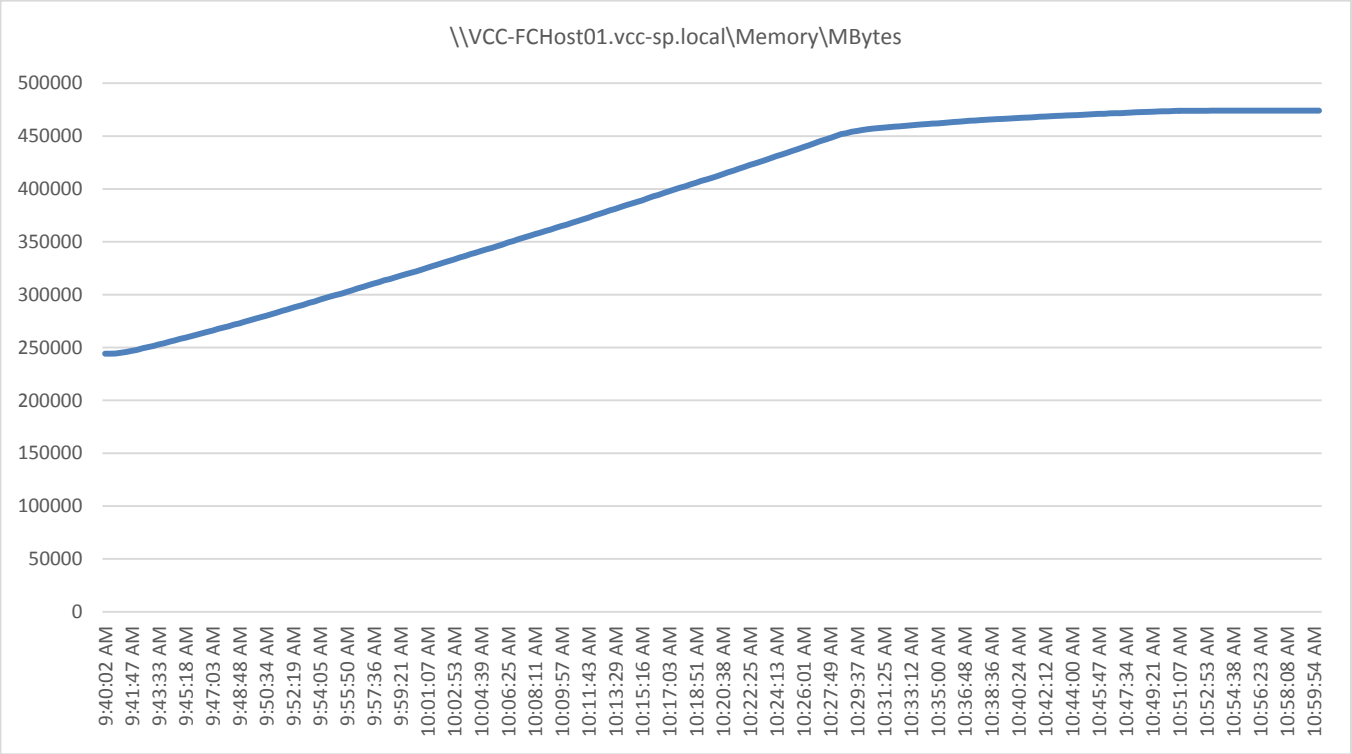
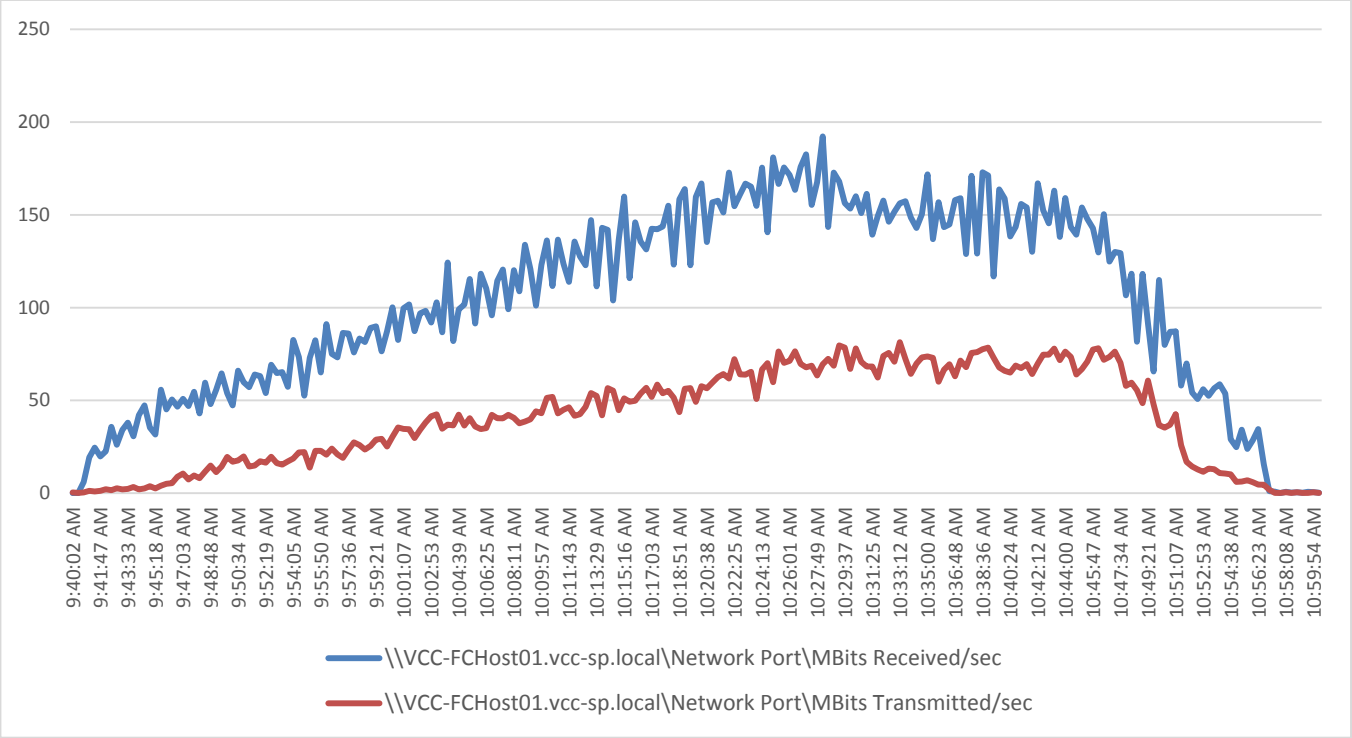


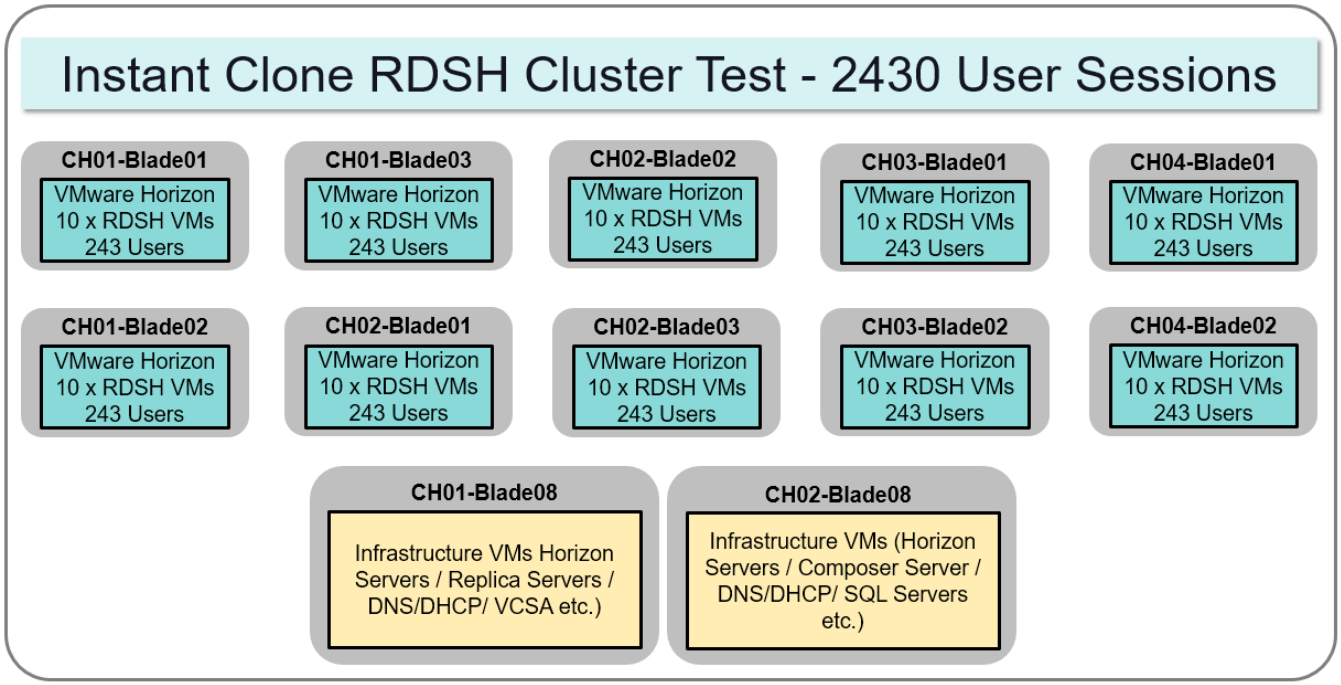
Figure 92 Single Server | VMware Horizon VDI Non –Persistent Full-Clone Desktops | Host Network Utilization



Cluster Workload Testing with 2430 RDS Users

This section details the key performance metrics that were captured on the Cisco UCS, Pure Storage FlashArray //X70 and RDS workload VMs during the RDSH Sessions testing. The cluster testing comprised of 2430 RDS Hosted sessions using 10 Cisco UCS B200 M5 workload blades.

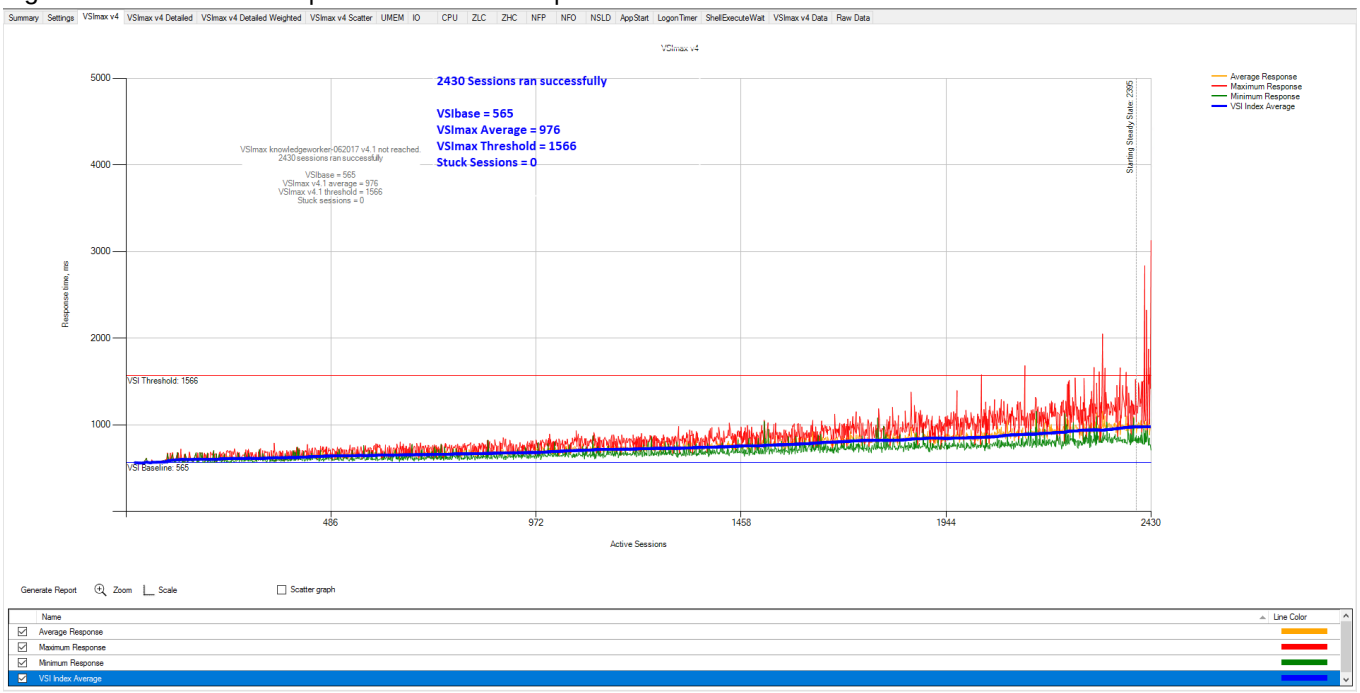
Figure 93 RDS Cluster Testing with 2430 Users



The workload for the test is 2430 RDS users. To achieve the target, sessions were launched against the single RDS cluster only. As per the Cisco Test Protocol for VDI solutions, all sessions were launched within 48 minutes (using the official Knowledge Worker Workload in VSI Benchmark Mode) and all launched sessions became active within two minutes subsequent to the last logged in session.

The configured system efficiently and effectively delivered the following results:

Figure 94 RDSH Cluster | 2430 RDSH Users | VMware Horizon RDSH VSI Score



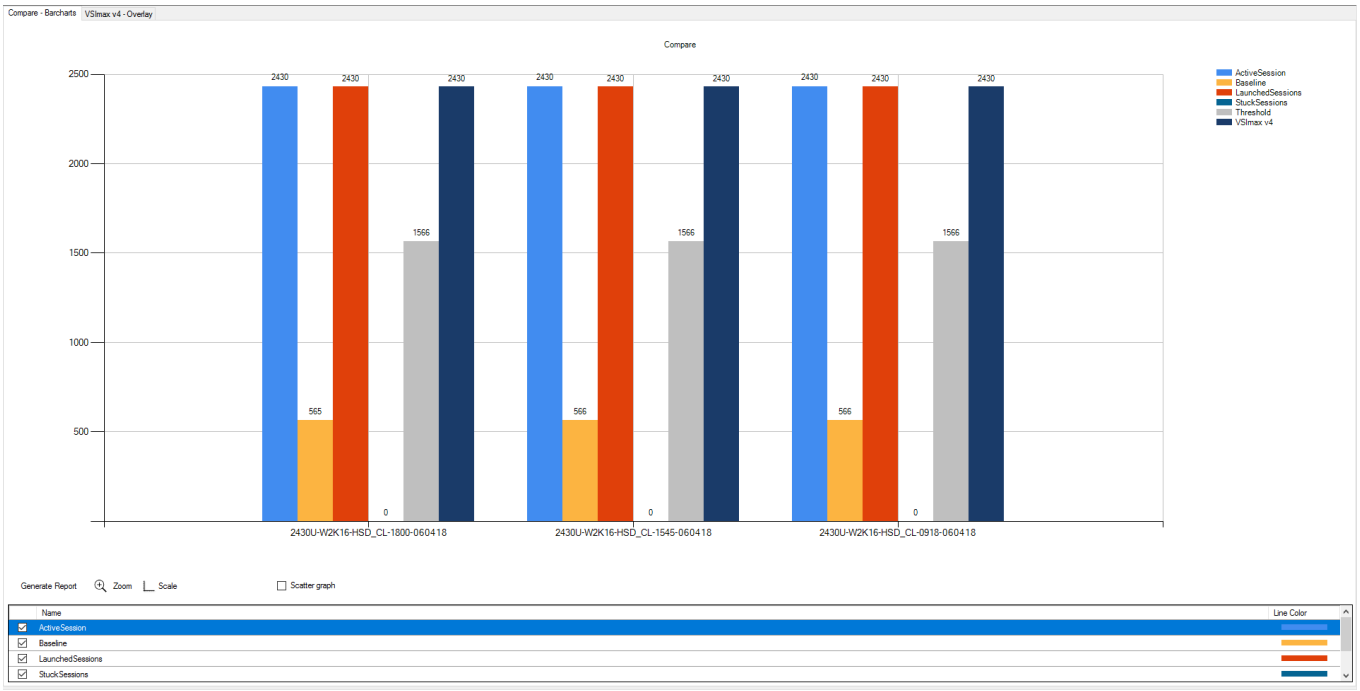


Figure 95 RDSH Cluster | 2430 RDSH Users | Host CPU Utilization

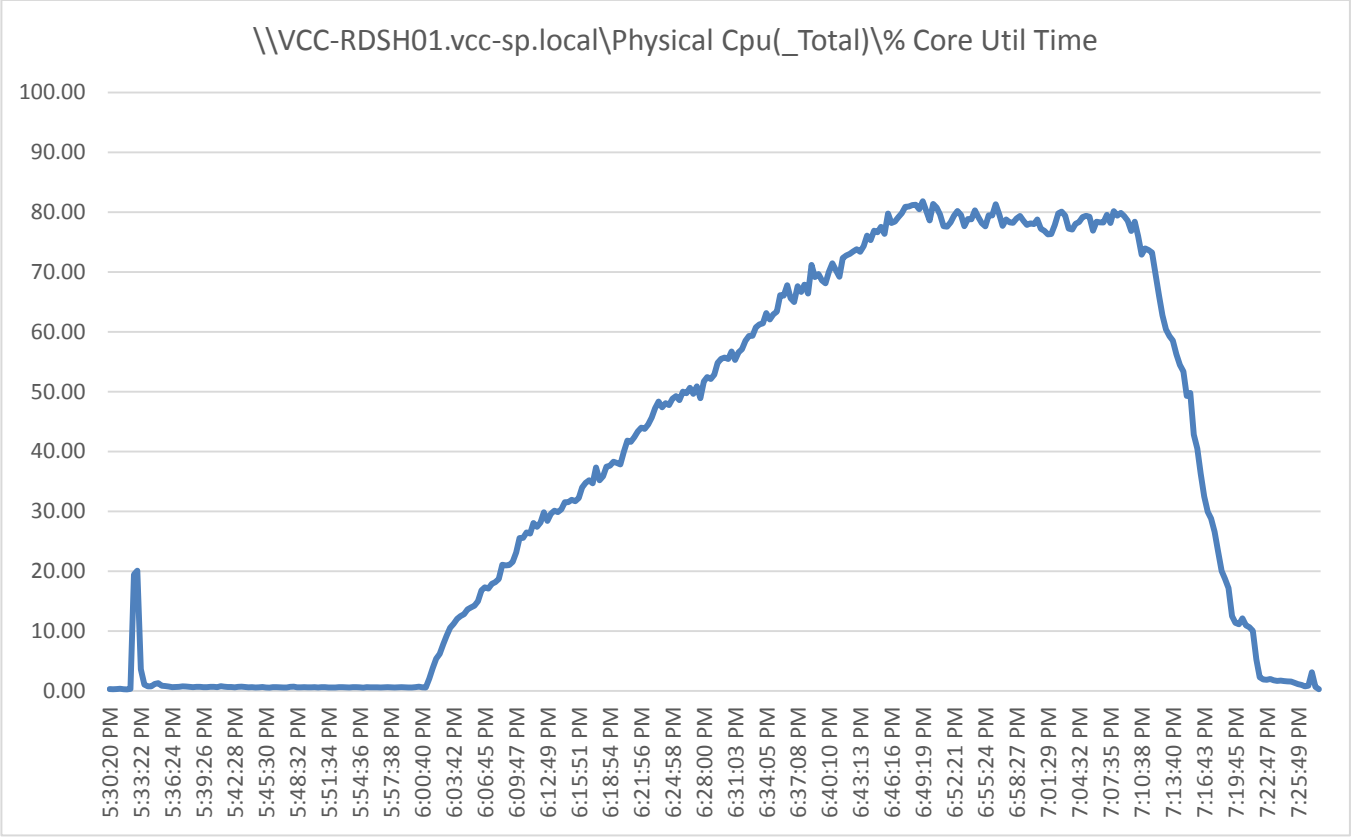


Figure 96 RDSH Cluster | 2430 RDS Users | Host Memory Utilization

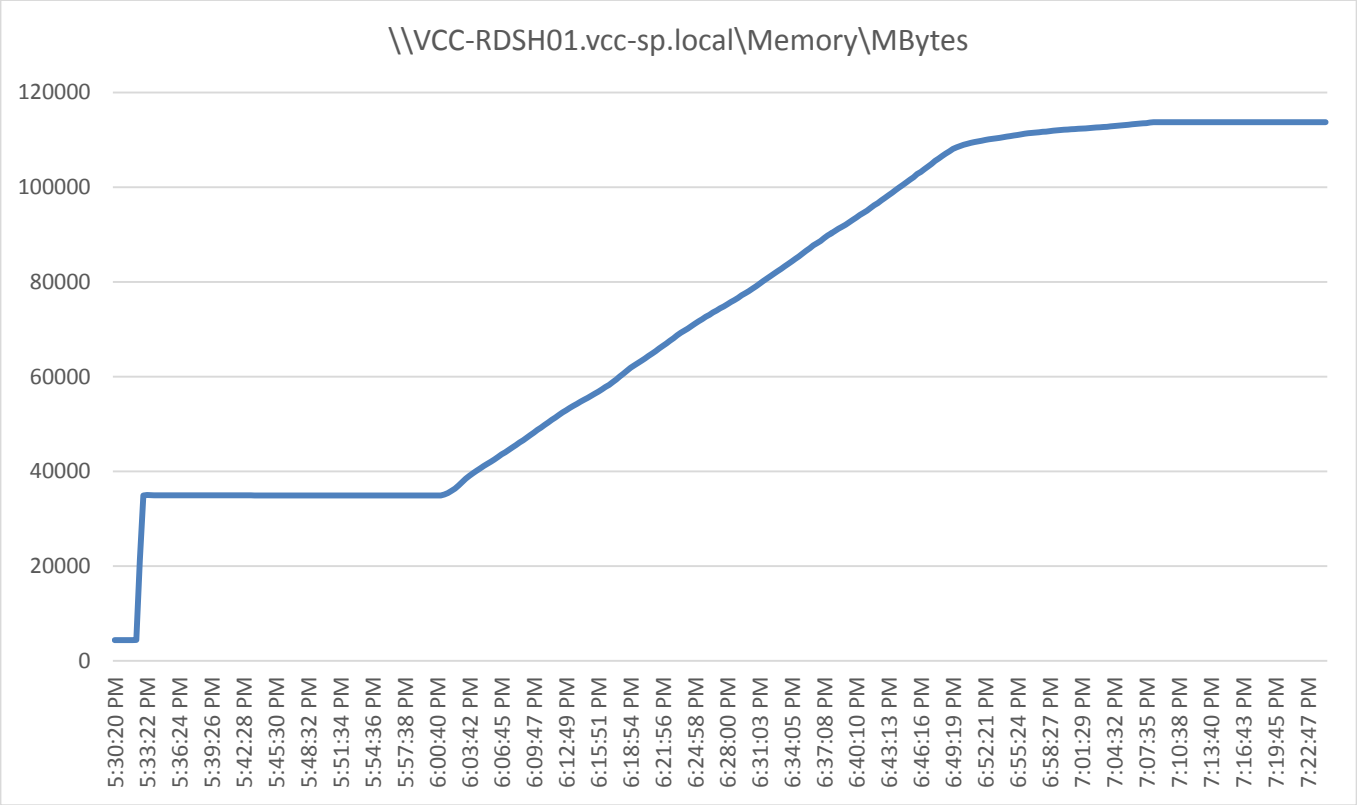


Figure 97 RDSH Cluster | 2430 RDSH Users | RDS Host | Host Network Utilization

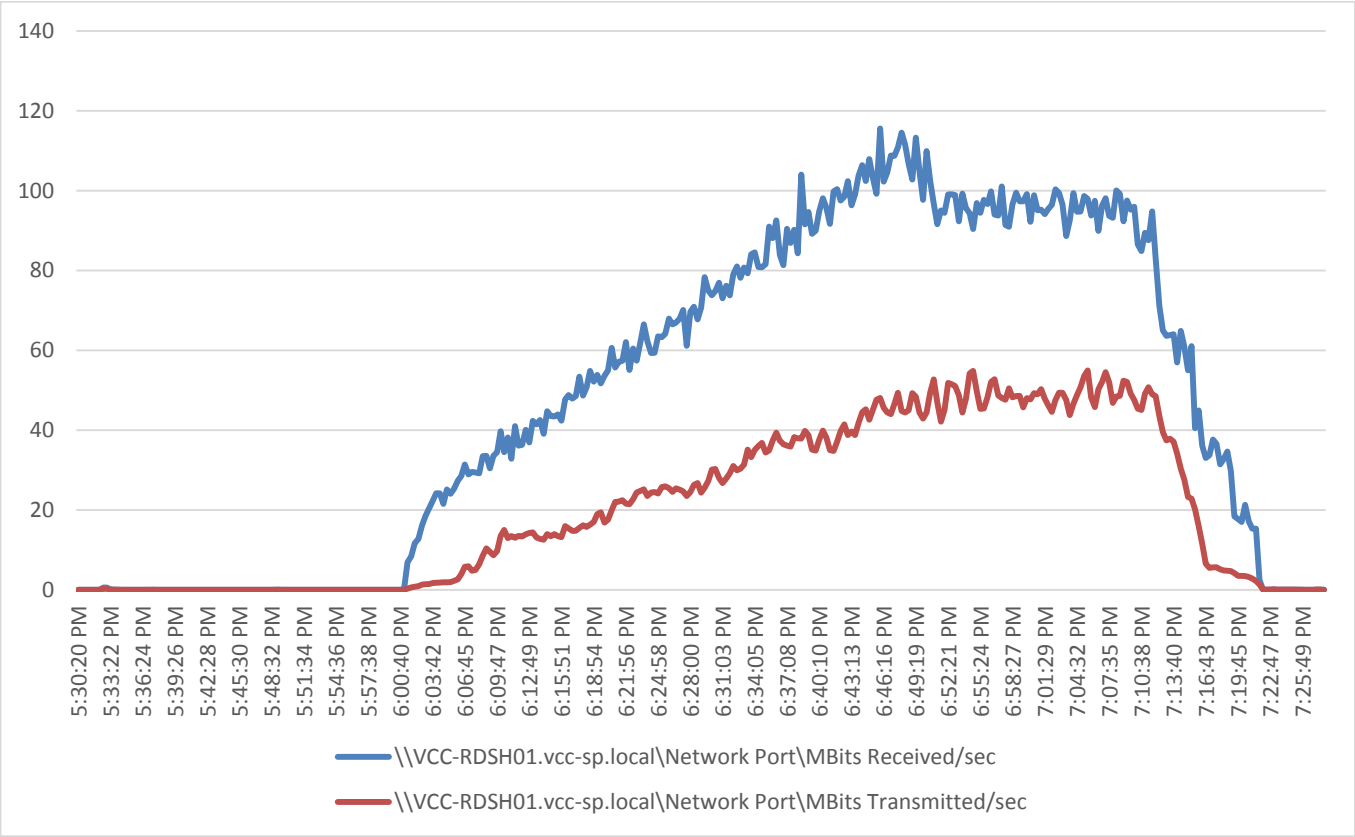


Figure 98 RDSH Cluster | 2430 RDSH Users | Host FC Adapter Commands/s

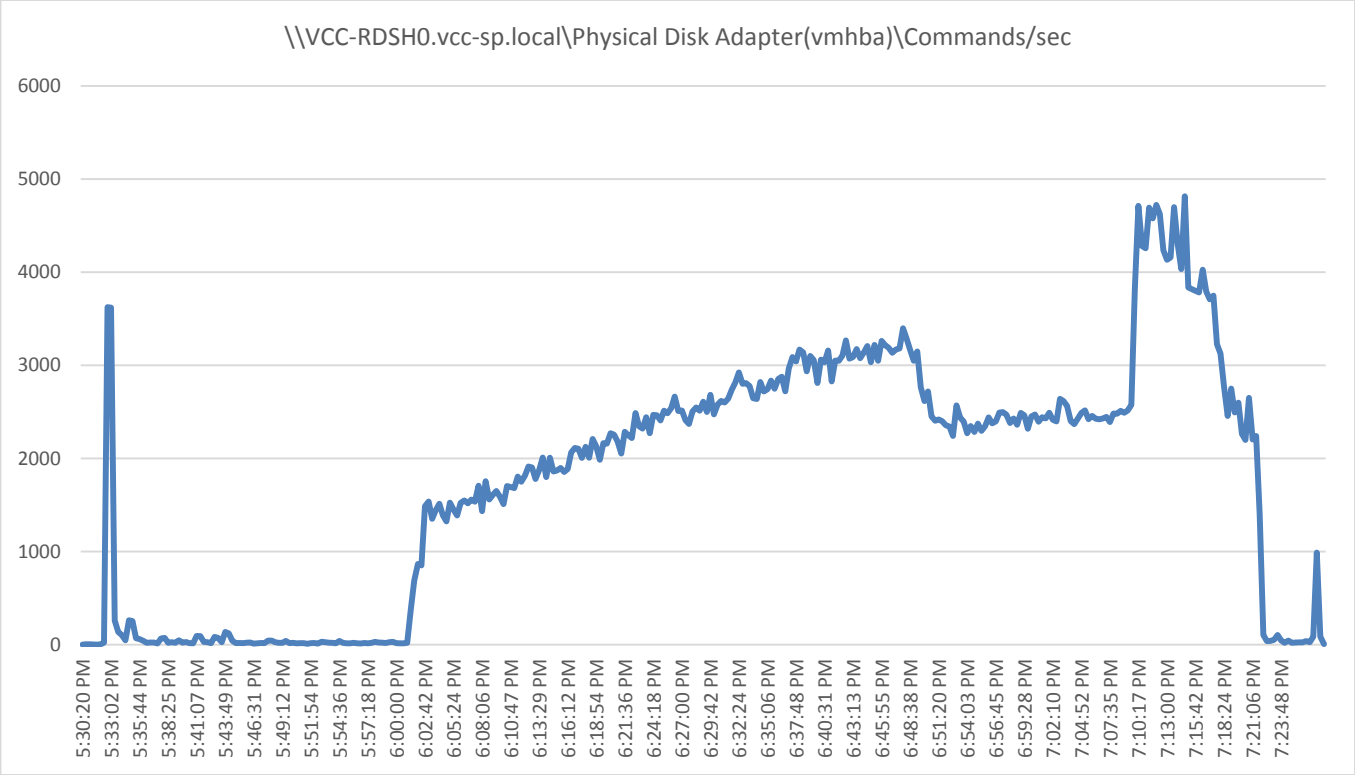


Figure 99 RDSH Cluster | 2430 RDSH Users | Host FC Adapter Reads/sec

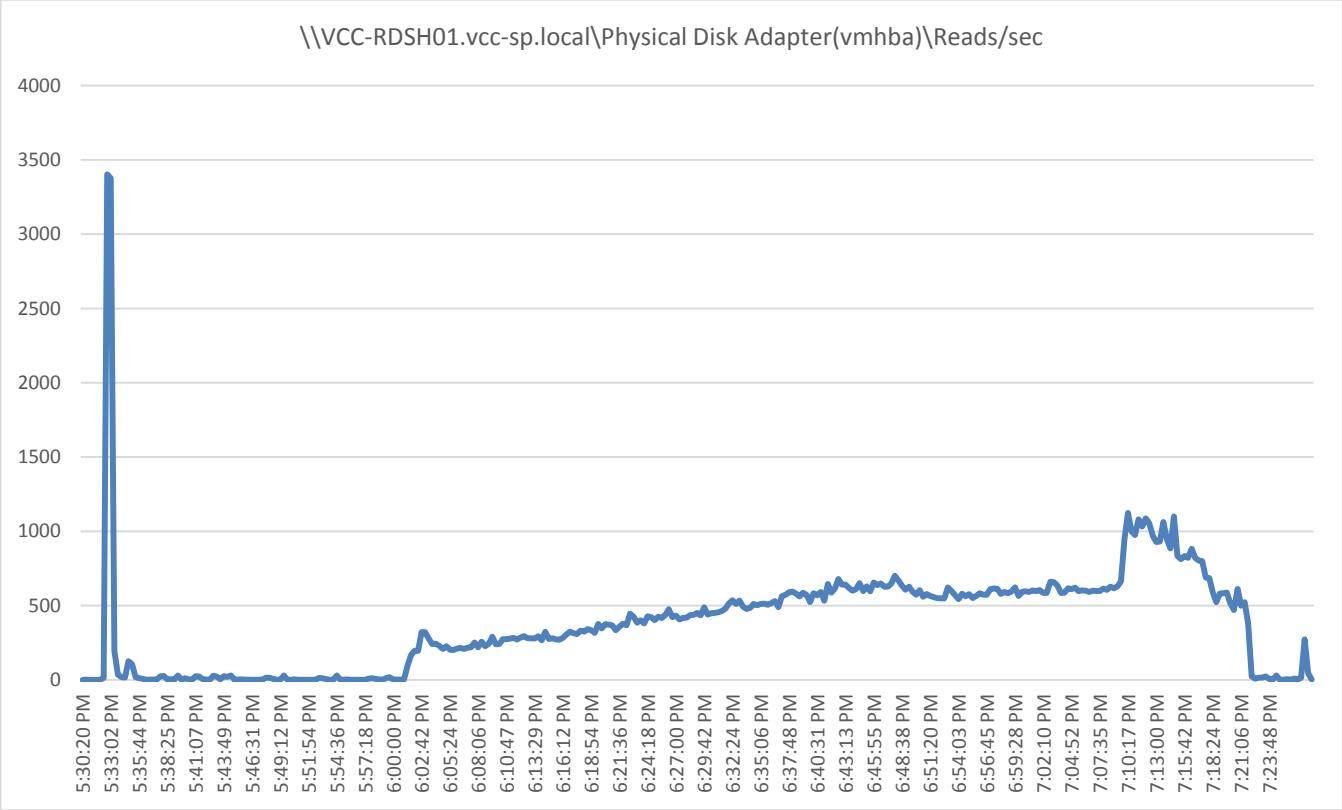

\\VCC-RDSH01.vcc-sp.local\Physical Disk Adapter(vmhba)\Reads/sec
4000
3500
3000
2500
2000
1500
1000
500
0
5:30:20 PM
5:33:02 PM
5:35:44 PM
5:38:25 PM
5:41:07 PM
5:43:49 PM
5:46:31 PM
5:49:12 PM
5:51:54 PM
5:54:36 PM
5:57:18 PM
6:00:00 PM
6:02:42 PM
6:05:24 PM
6:08:06 PM
6:10:47 PM
6:13:29 PM
6:16:12 PM
6:18:54 PM
6:21:36 PM
6:24:18 PM
6:27:00 PM
6:29:42 PM
6:32:24 PM
6:35:06 PM
6:37:48 PM
6:40:31 PM
6:43:13 PM
6:45:55 PM
6:48:38 PM
6:51:20 PM
6:54:03 PM
6:56:45 PM
6:59:28 PM
7:02:10 PM
7:04:52 PM
7:07:35 PM
7:10:17 PM
7:13:00 PM
7:15:42 PM
7:18:24 PM
7:21:06 PM
7:23:48 PM

Figure 100RDSH Cluster | 2430 RDSH Users | Host FC Adapter Writes/s

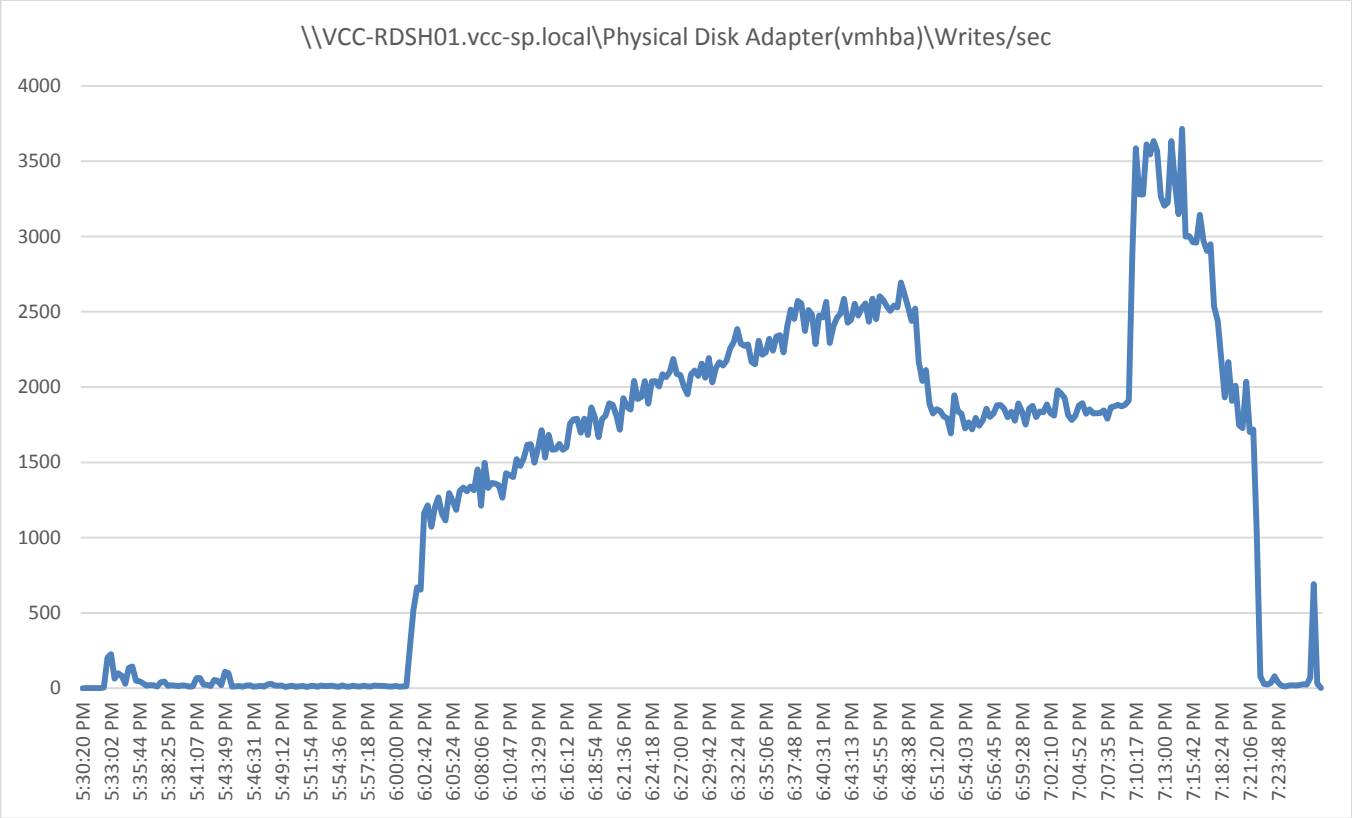


Figure 101RDSH Cluster | 2430 RDSH Users | Horizon Administrator Console Reporting 2430 Active Sessions

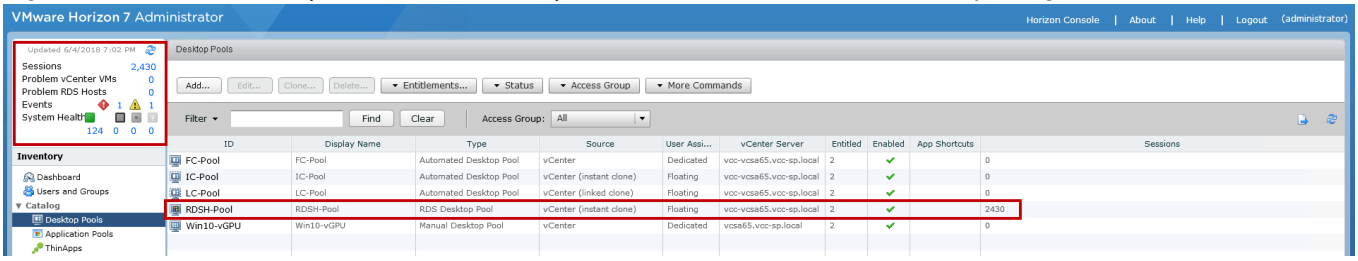
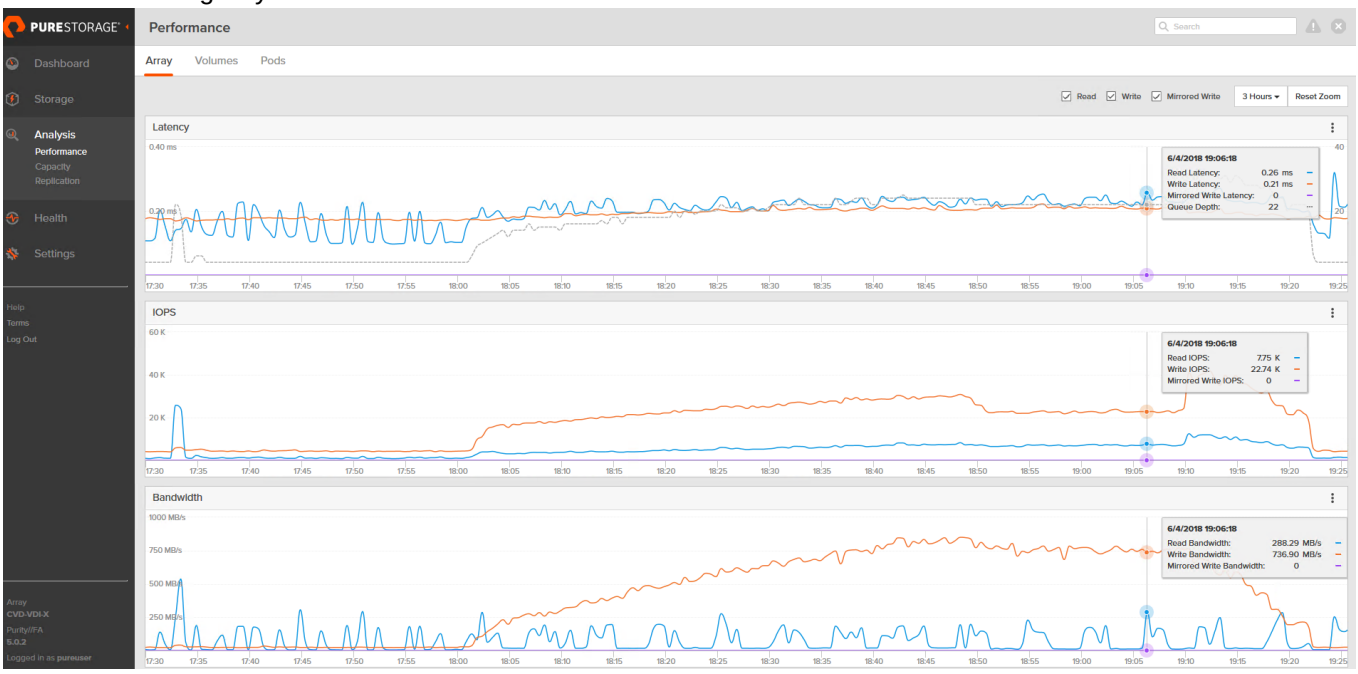


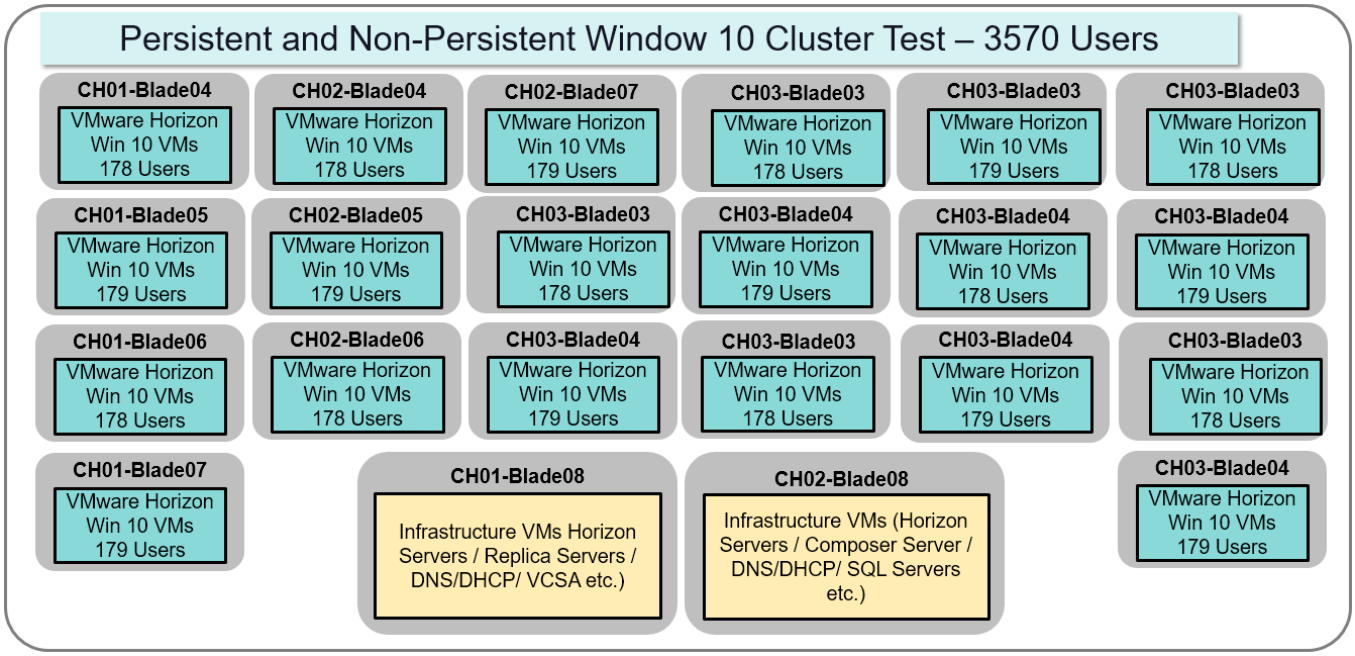
Figure 102VMware Horizon Pooled Instant-Cloned RDSH VMs running 2430 Session Users: Pure Storage //X70 Storage System Performance



Cluster Workload Testing with 3570 Persistent and Non-Persistent VDI Users

This section shows the key performance metrics that were captured on the Cisco UCS, Pure Storage //X70 storage, and Infrastructure VMs during the persistent desktop testing. The cluster testing with comprised of 3570 VDI Non- Persistent desktop sessions using 20 workload blades.

Figure 103VMware Horizon VDI Persistent and Non-Persistent Desktop VMs Cluster Testing with 3570 Users



The workload for the test is 3570 non-persistent desktop users. To achieve the target, sessions were launched against the single persistent cluster only. As per the Cisco Test Protocol for VDI solutions, all sessions were launched within 48 minutes (using the official Knowledge Worker Workload in VSI Benchmark Mode) and all launched sessions became active within two minutes subsequent to the last logged in session.

The configured system efficiently and effectively delivered the following results:

Figure 104VDI Cluster | 3570 Persistent and Non-Persistent Desktop Users | VSI Score

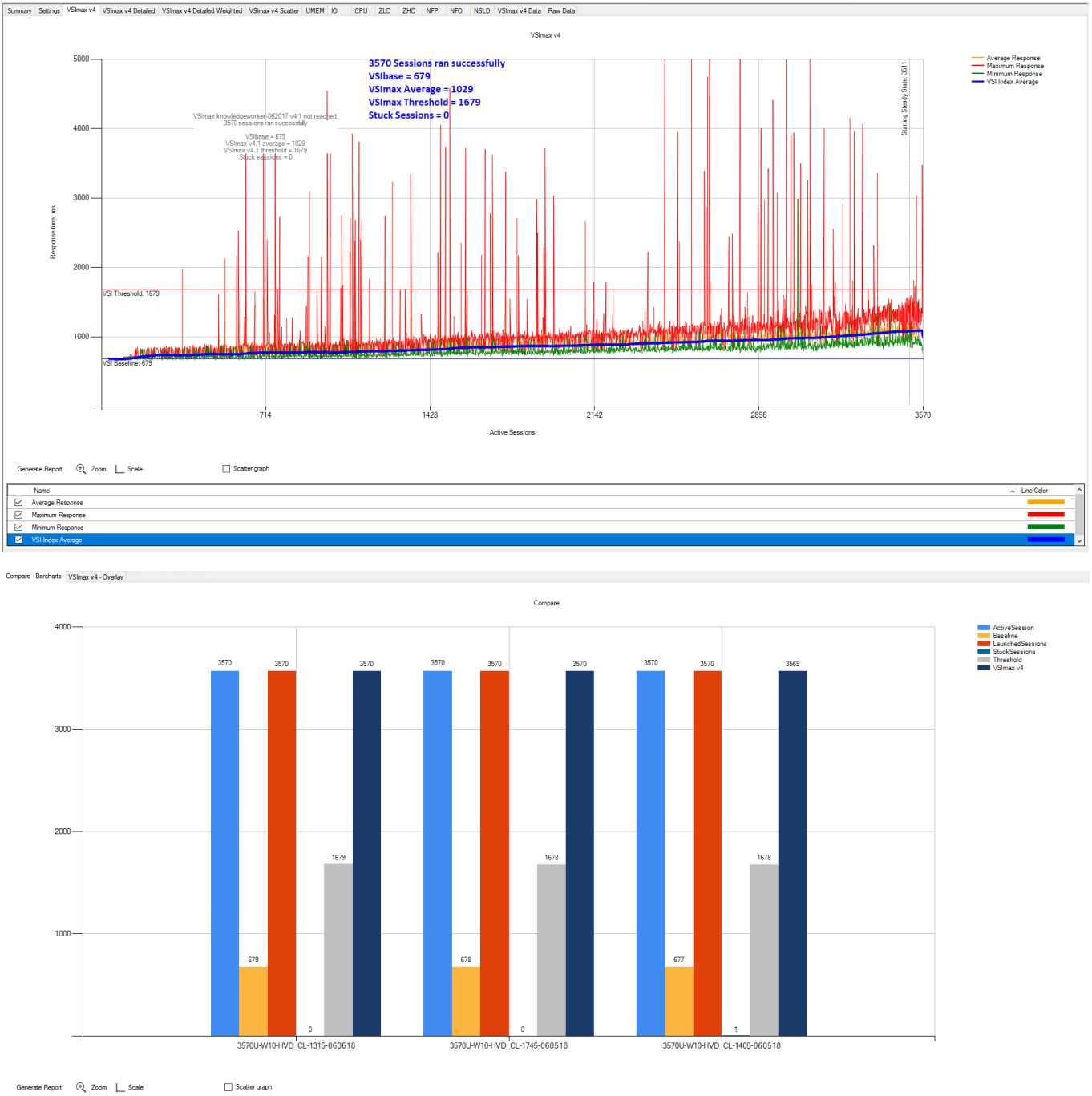


Figure 105VDI Cluster | 3570 Persistent and Non-Persistent Desktop Users | Host CPU Utilization

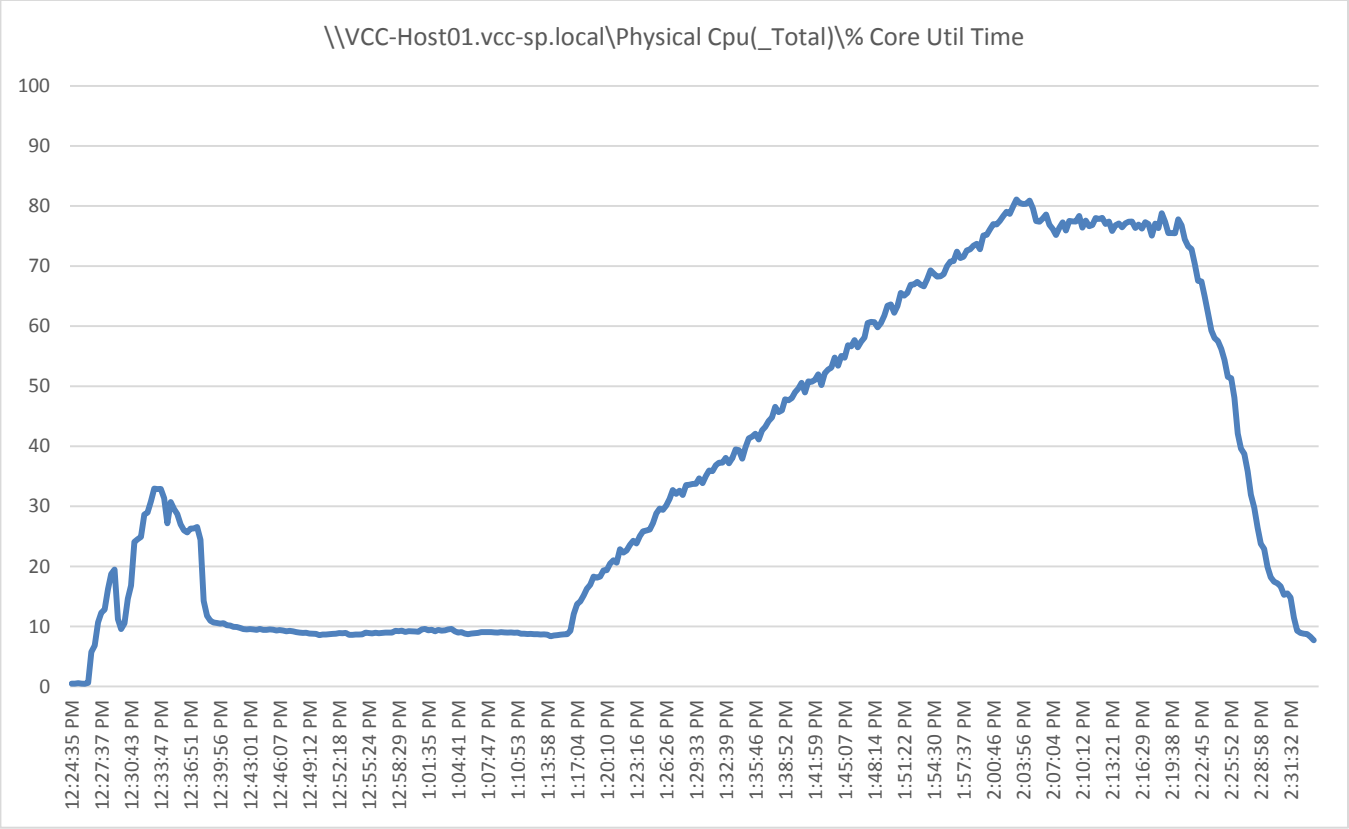


Figure 106VDI Cluster | 3570 Persistent and Non-Persistent Desktop Users | Host Memory Utilization

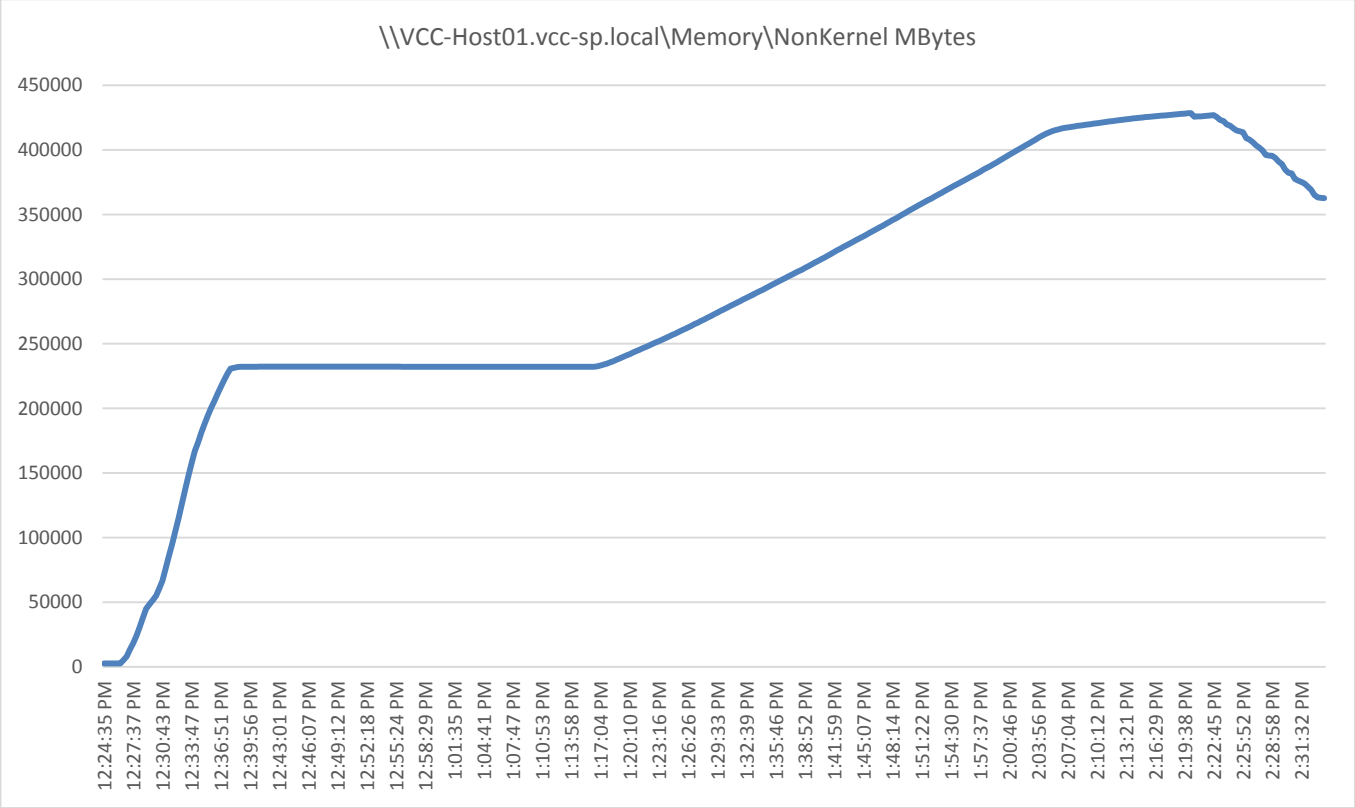


Figure 107VDI Cluster | 3570 Persistent and Non-Persistent Desktop Users | Host Network Utilization

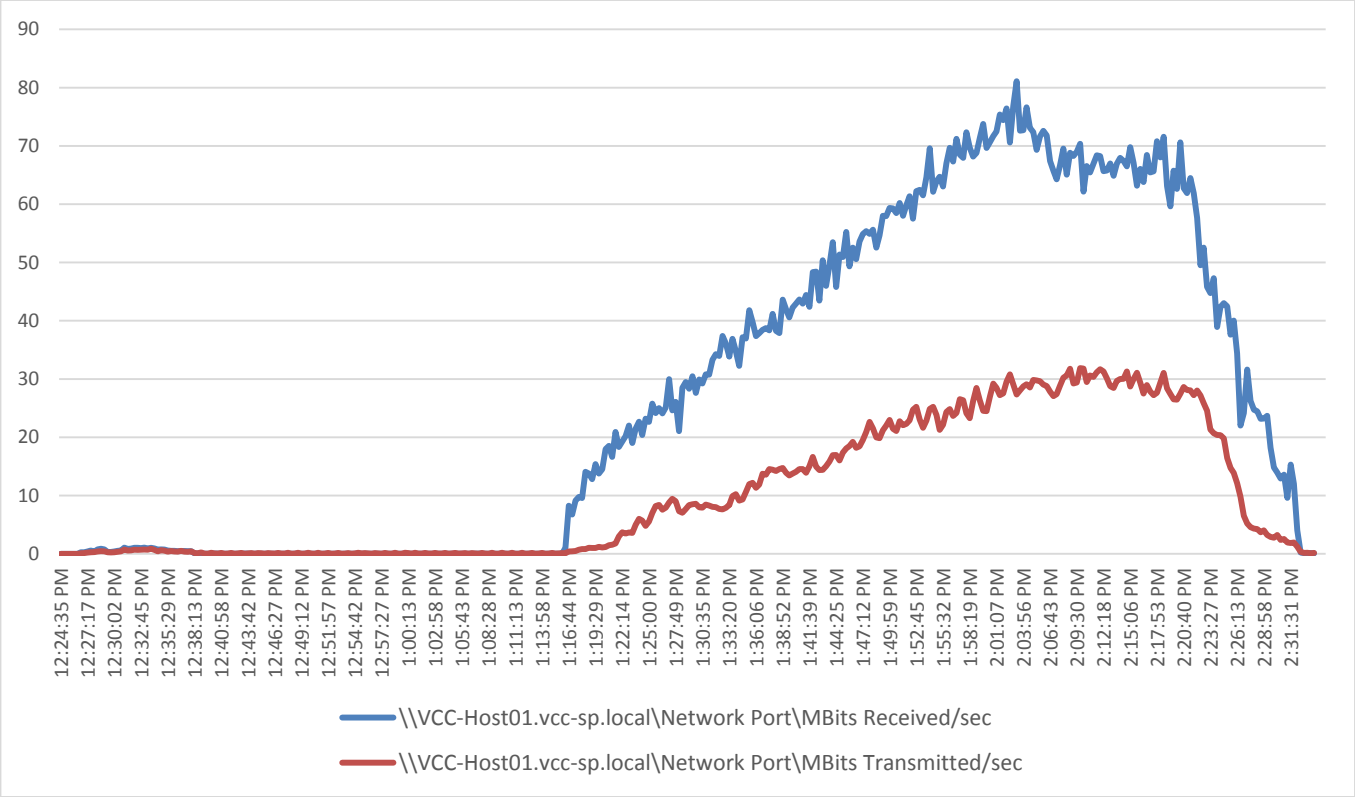


Figure 108VDI Cluster | 3570 Persistent and Non-Persistent Desktop Users | Host FC Adapter Commands/s

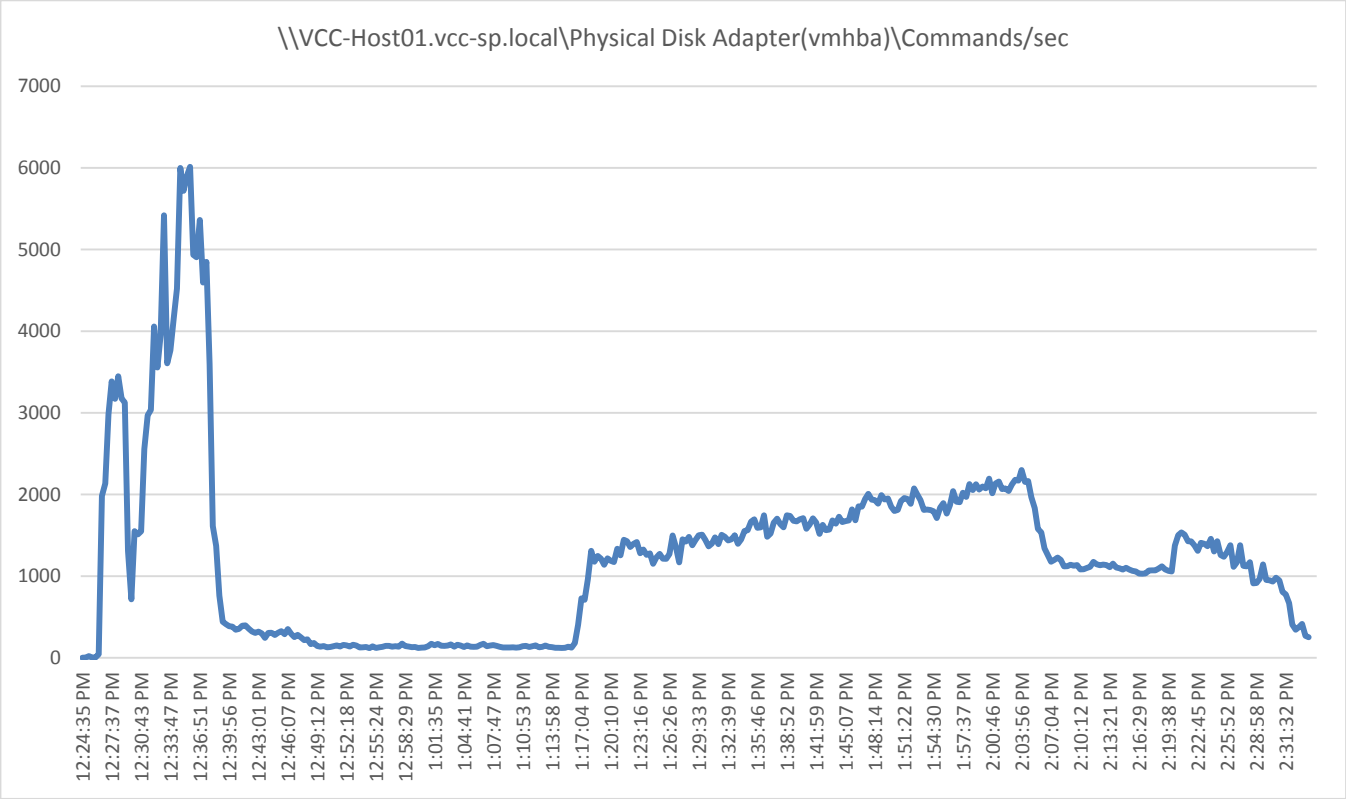


Figure 109VDI Cluster | 3570 Persistent and Non-Persistent Desktop Users | Host FC Adapter Reads/s

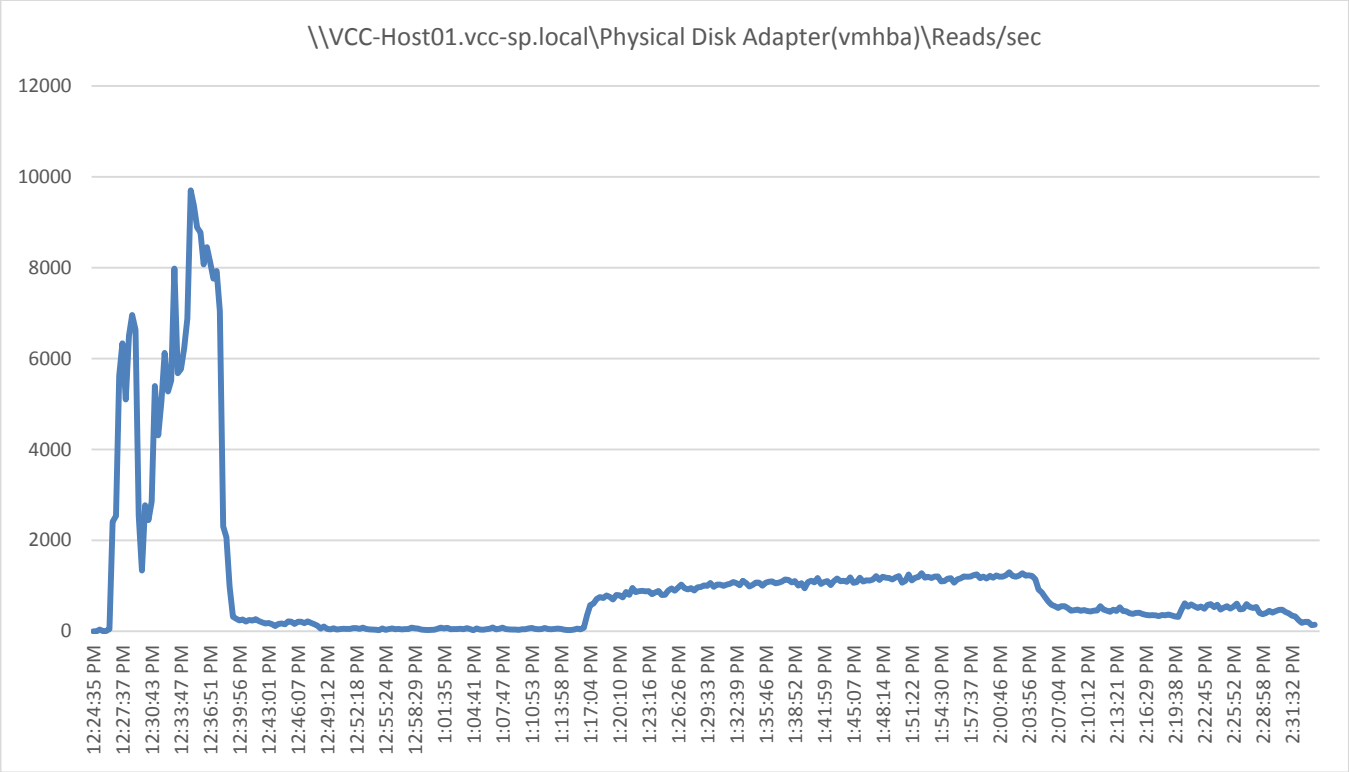


Figure 110VDI Cluster | 3570 Persistent and Non-Persistent Desktop Users | Host FC Adapter Writes/s

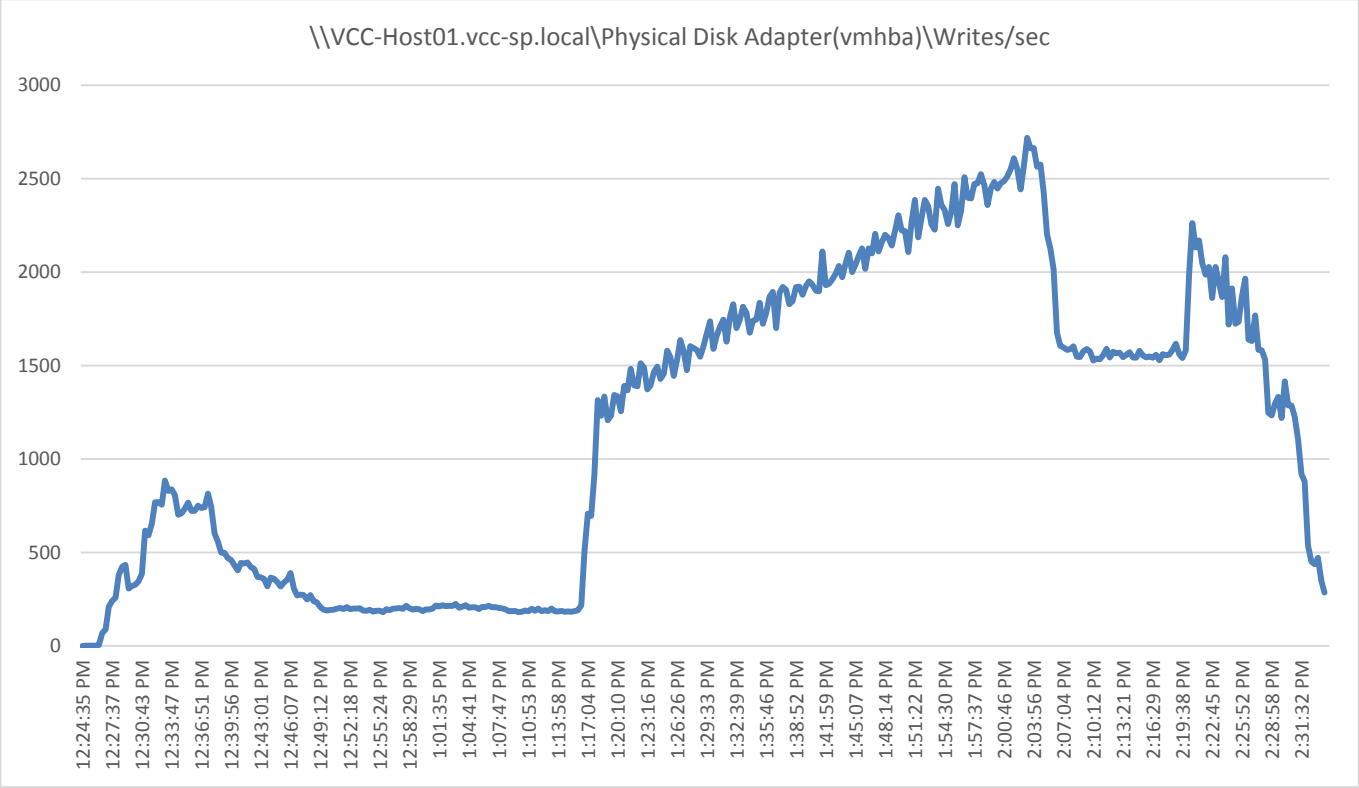
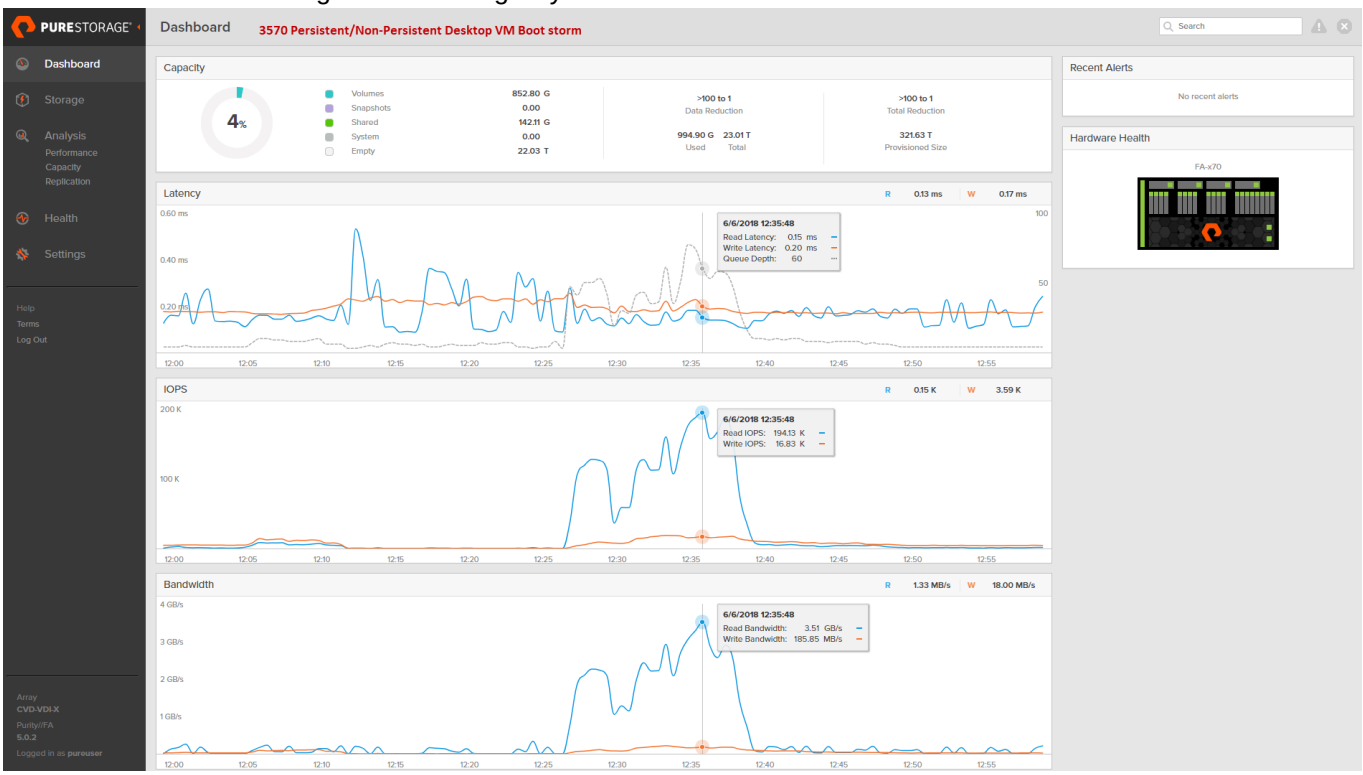


Figure 111VMware Horizon Pooled Persistent and Non-Persistent Desktop VMs Cluster Testing with 3570 Users: Pure Storage //X70 Storage System Performance



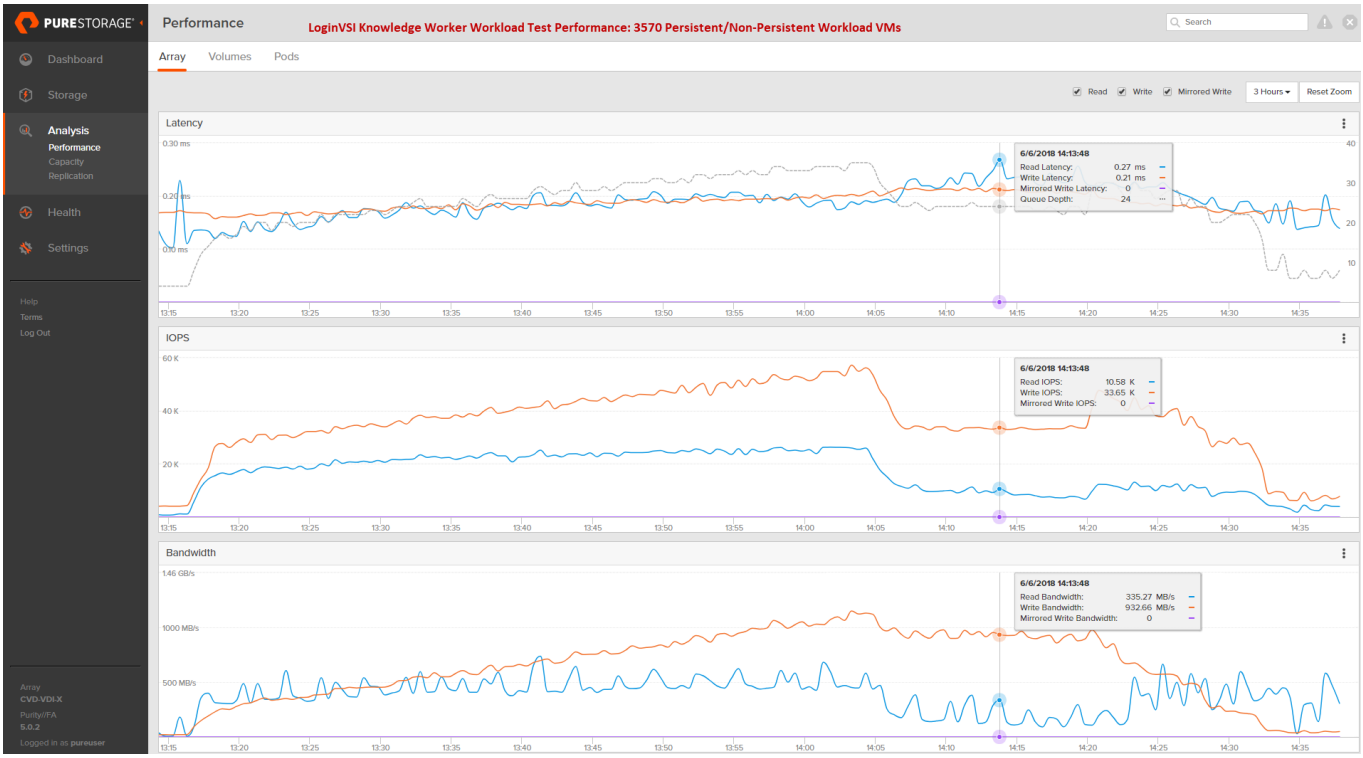
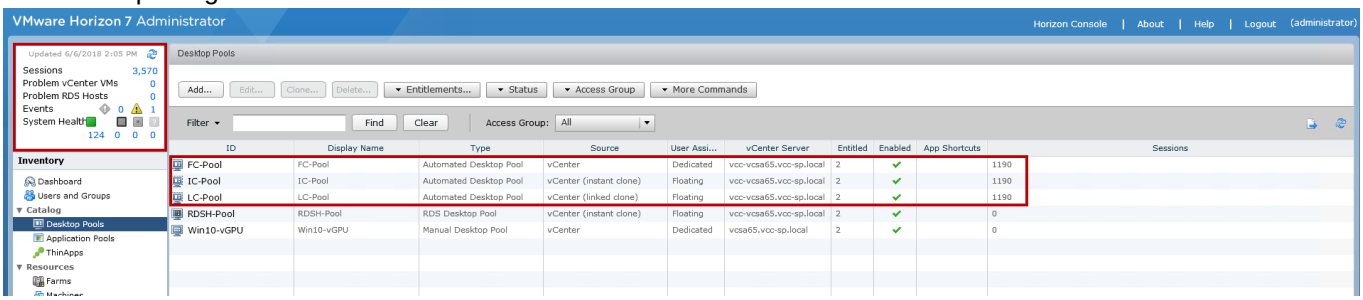


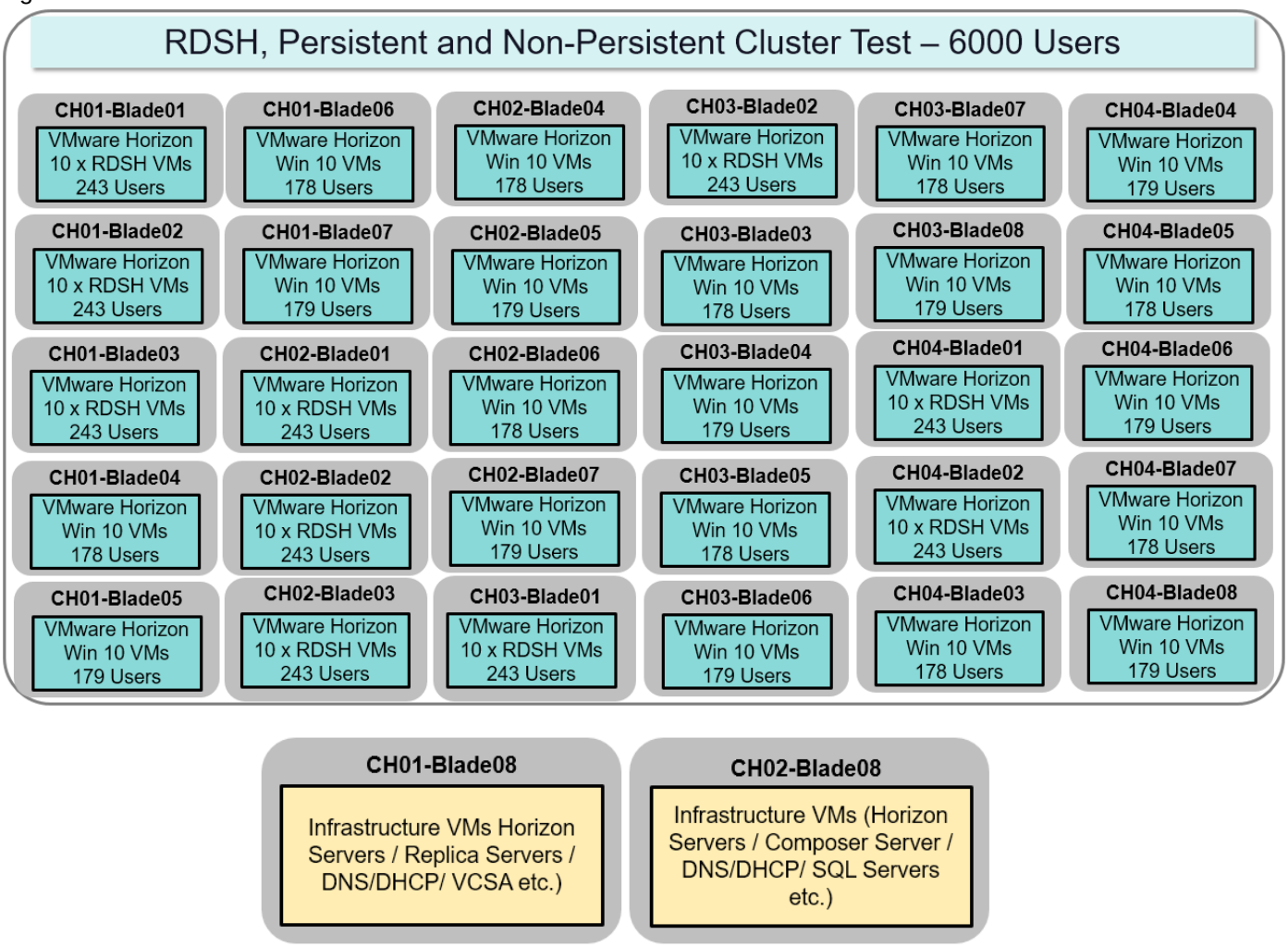
Figure 112VDI Cluster | 3570 Persistent and Non-Persistent Desktop Users Horizon Administrator Console reporting 2430 Active Sessions



Full Scale Mixed Workload Testing with 6000 Users

This section shows the key performance metrics that were captured on the Cisco UCS, Pure Storage FlashArray //X70, RDSH VMs and VDI Persistent and Non–Persistent VDI virtual machines performance monitoring during the full-scale test. The full-scale testing with 6000 users comprised of: 2430 RDS Hosted Server Sessions on 10 Cisco UCS B200 M5 blades, 3570 VDI Non-Persistent Instant-Clones, Linked-Clones and Full-Clones using 20 Cisco UCS B200 M5 blades.

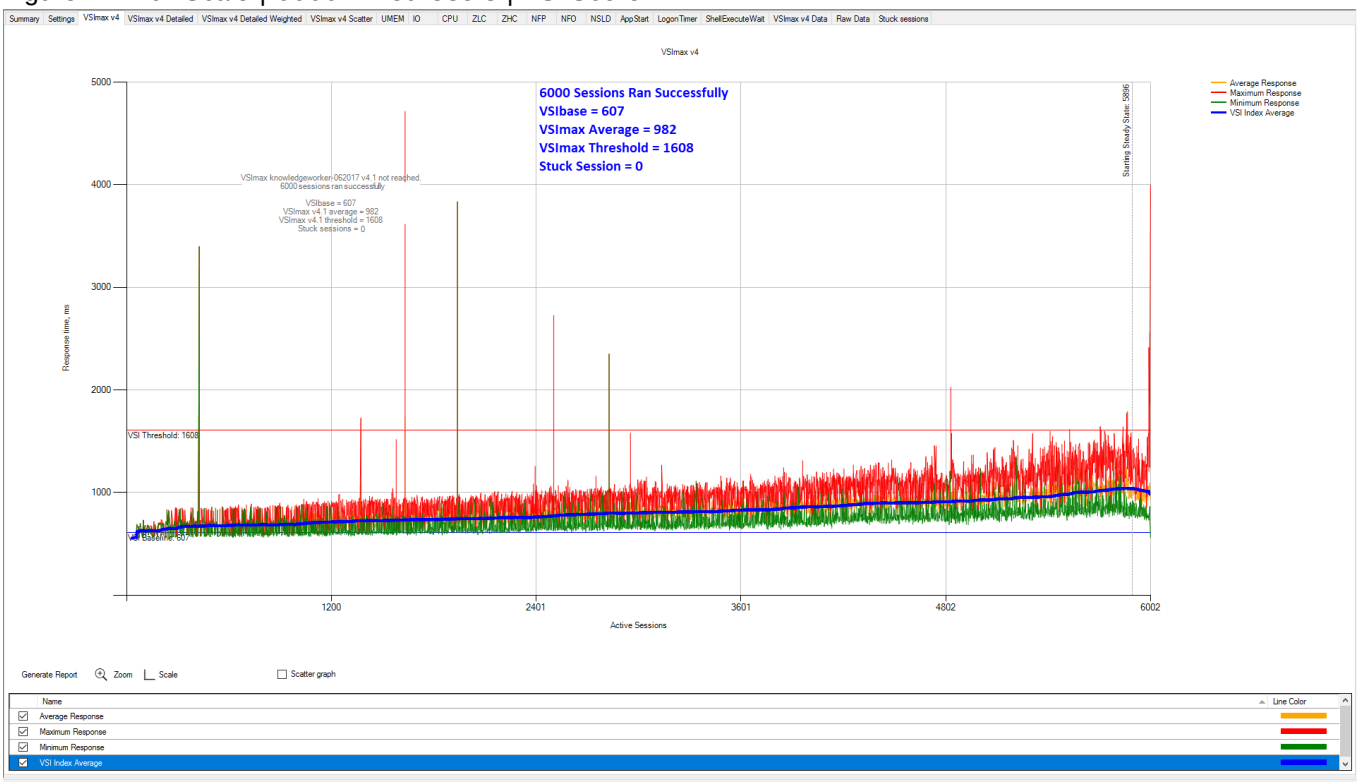
Figure 113Full Scale Horizon 7 Pooled Mixed Workload Test with 6000 Users

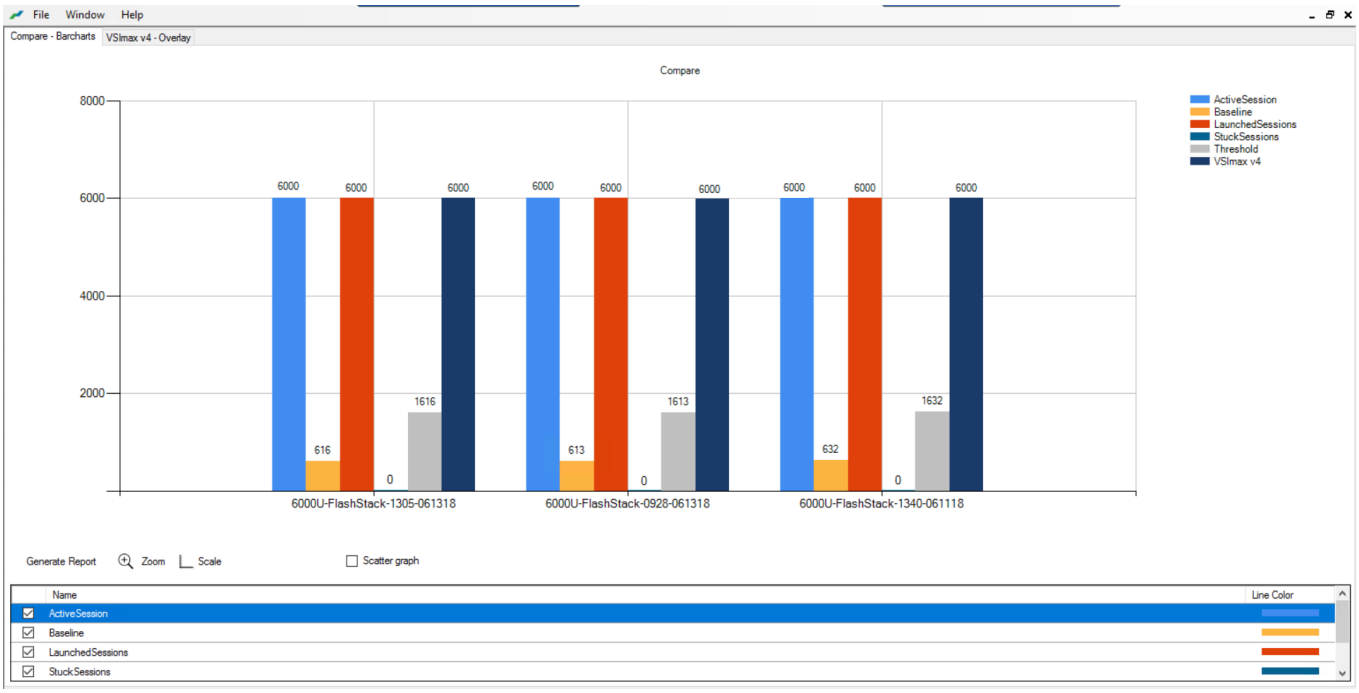


The combined mixed workload for the solution is 6000 users. To achieve the target, sessions were launched against all workload clusters concurrently. As per the Cisco Test Protocol for VDI solutions, all sessions were launched within 48 minutes (using the official Knowledge Worker Workload in VSI Benchmark Mode) and all launched sessions became active within two minutes subsequent to the last logged in session.

The configured system efficiently and effectively delivered the following results:

Figure 114Full Scale | 6000 Mixed Users | VSI Score





RDSH Server Performance Monitor Data for One Sample RDSH Server: 6000 Users Mixed Scale Testing

Figure 115Full Scale | 6000 Mixed Users | RDSH Host | Host CPU Utilization

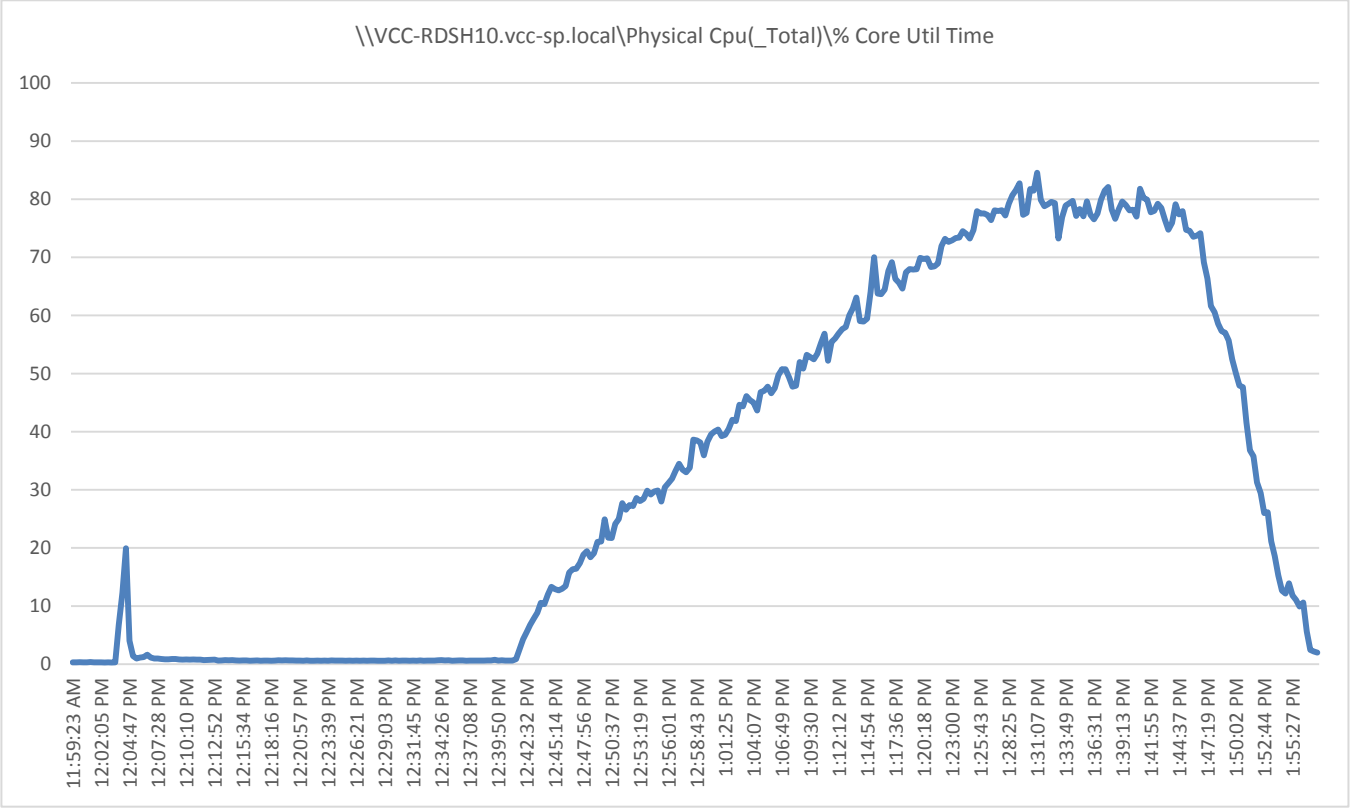


Figure 116Full Scale | 6000 Mixed Users | RDSH Host | Host Memory Utilization

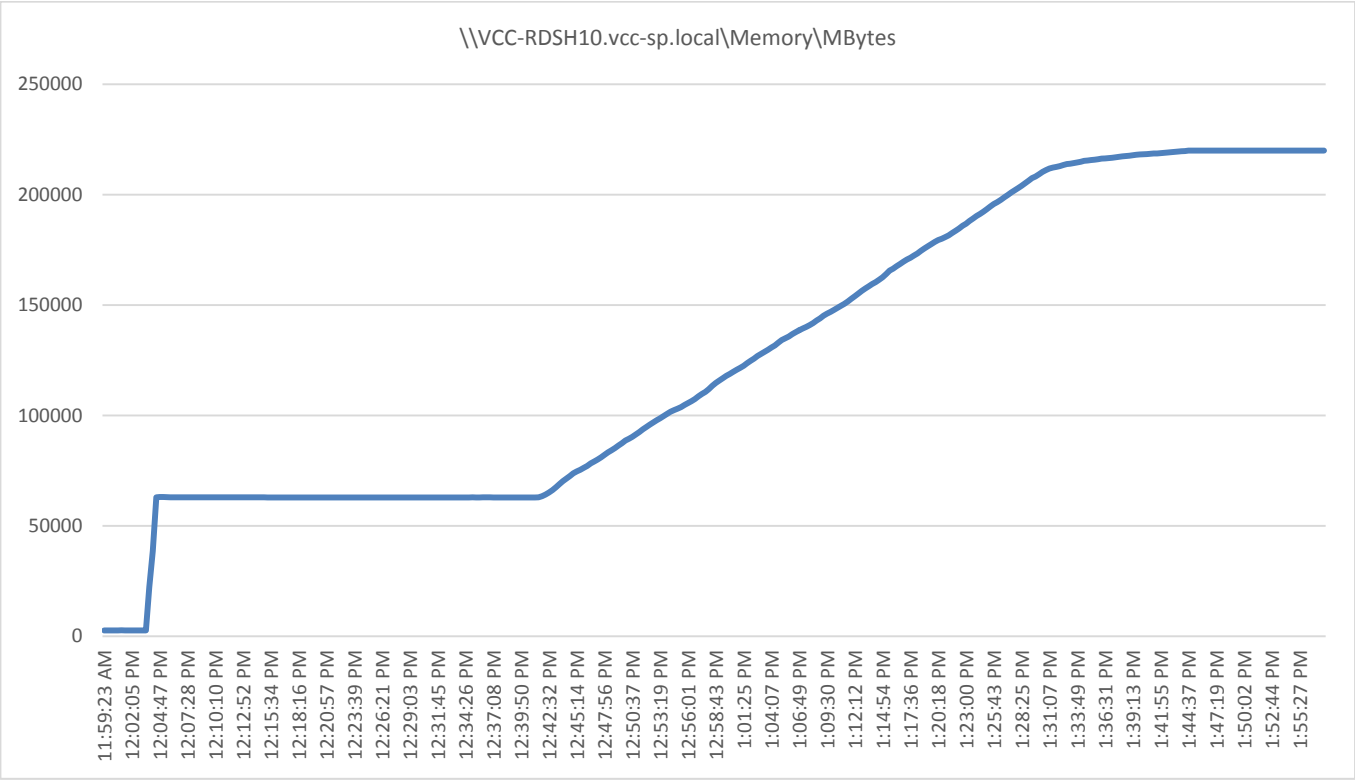


Figure 117Full Scale | 6000 Mixed Users | RDSH Host | Host Fibre Channel Network Utilization

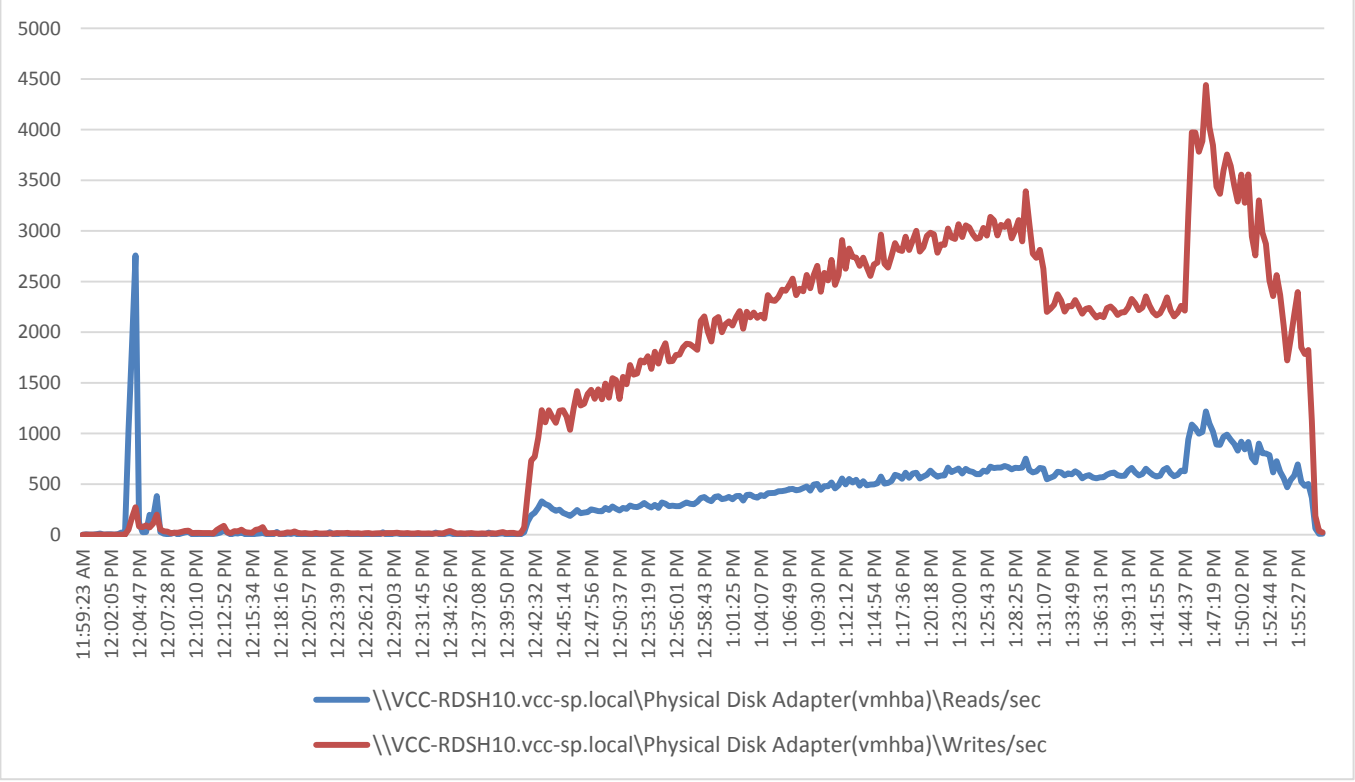
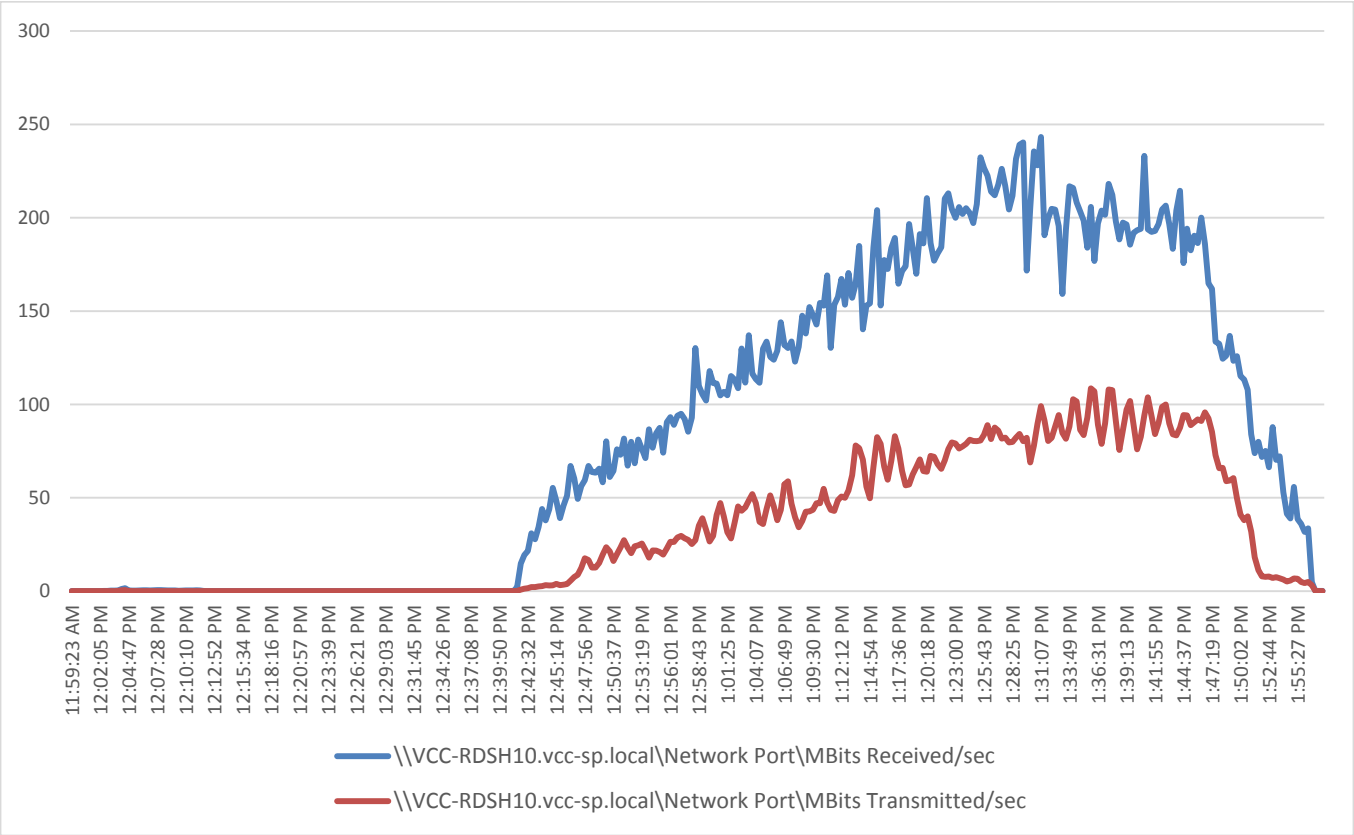


Figure 118Full Scale | 6000 Mixed Users | RDSH Host | Host Network Utilization



HVD Server Performance Monitor Data for One Sample HVD Server: 6000 Users Mixed Scale Testing

Figure 119Full Scale | 6000 Mixed Users | VDI Host | Host CPU Utilization

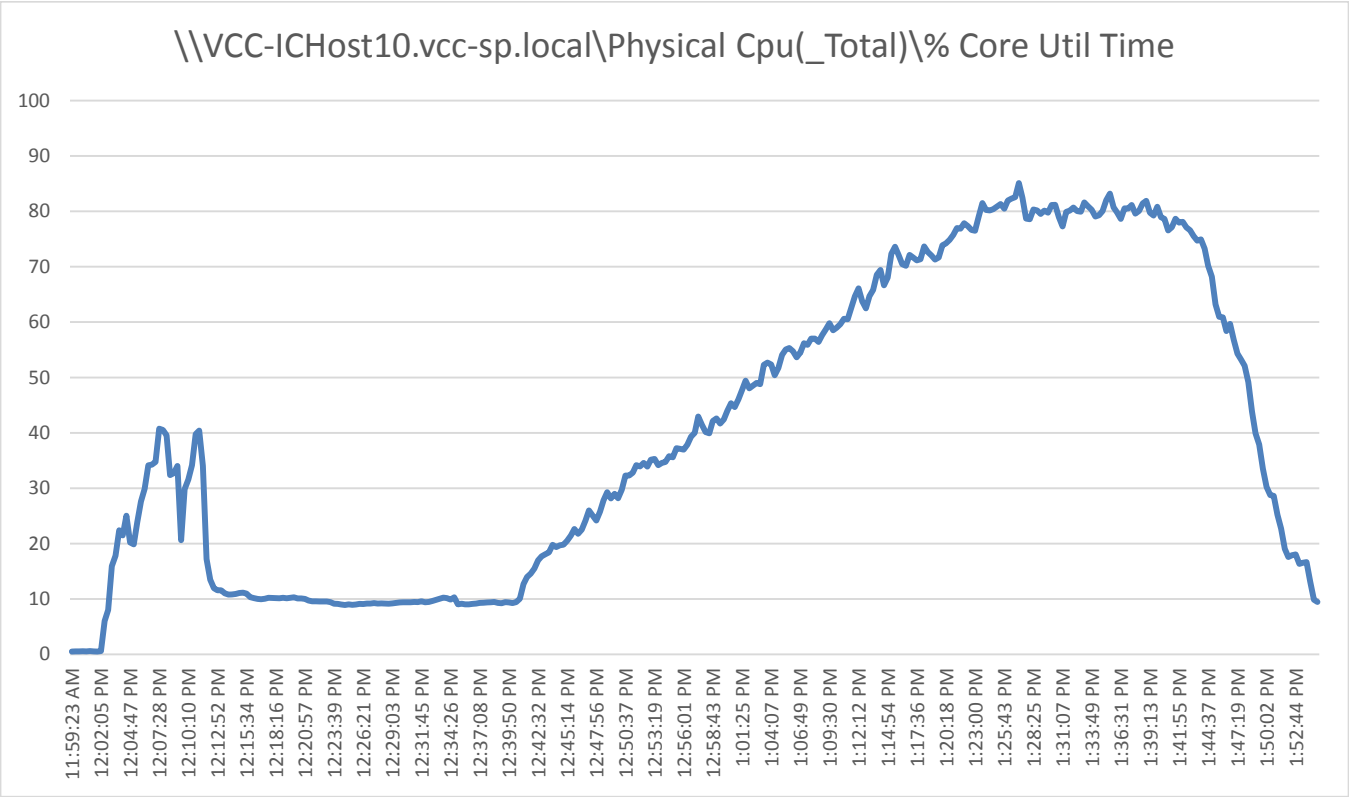


Figure 120Full Scale | 6000 Mixed Users | VDI Host | Host Memory Utilization

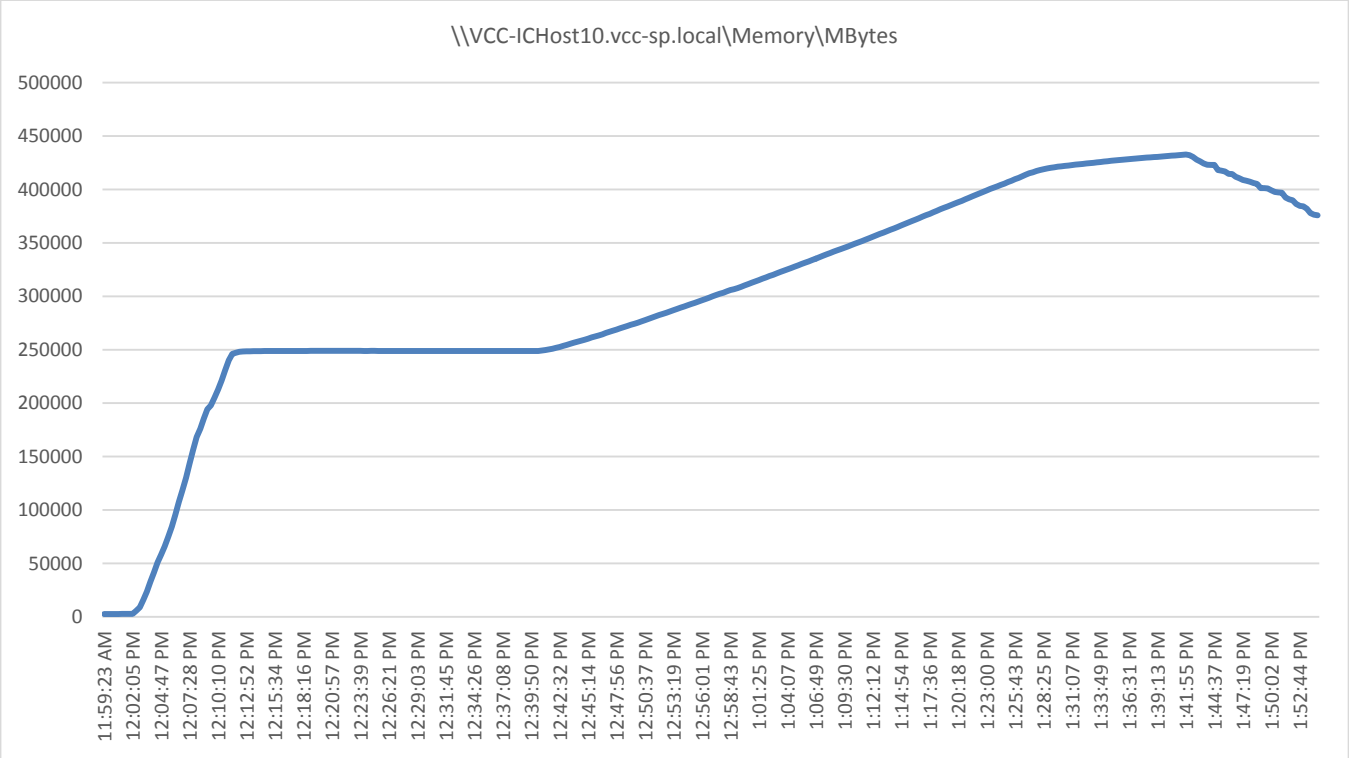


Figure 121Full Scale | 6000 Mixed Users | VDI Host | Host Fibre Channel Network Utilization

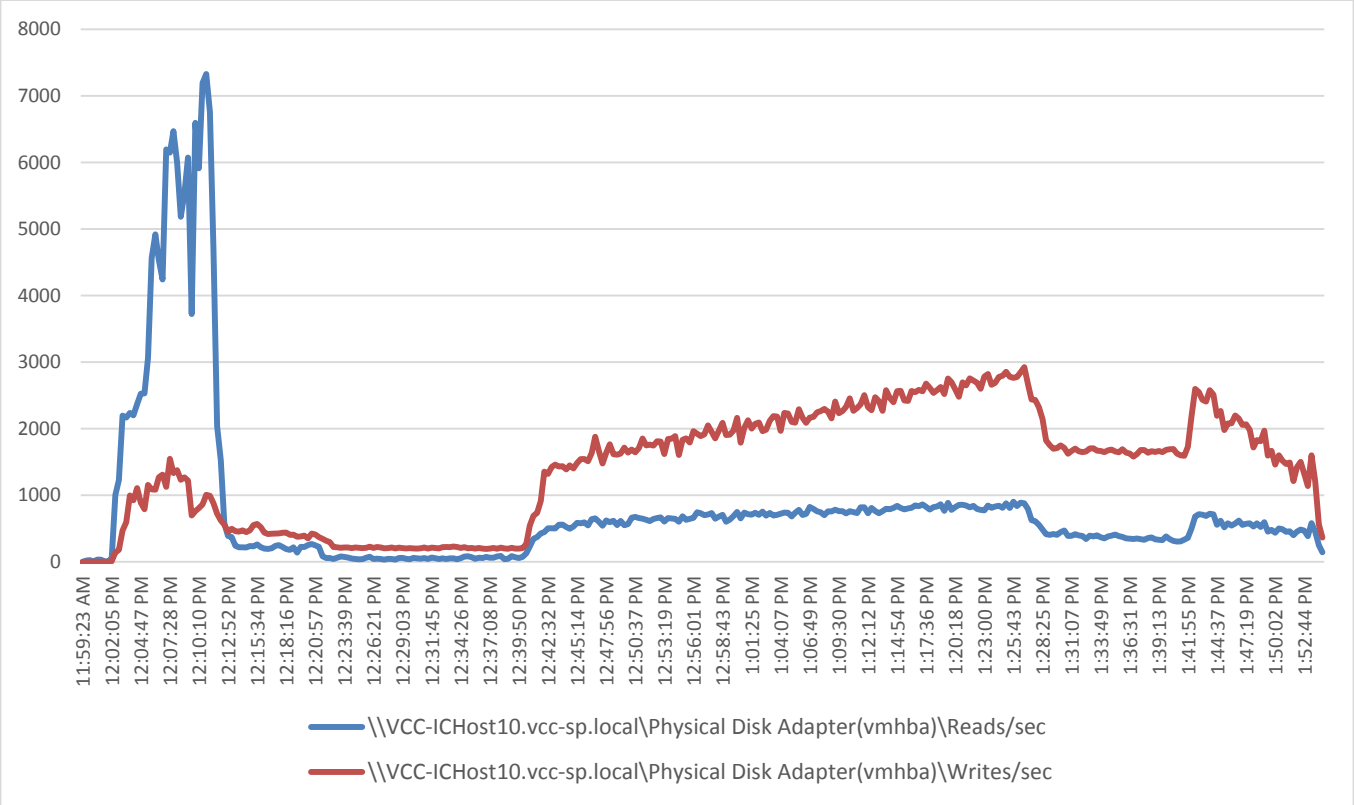


Figure 122Full Scale | 6000 Mixed Users | VDI Host | Host Network Utilization

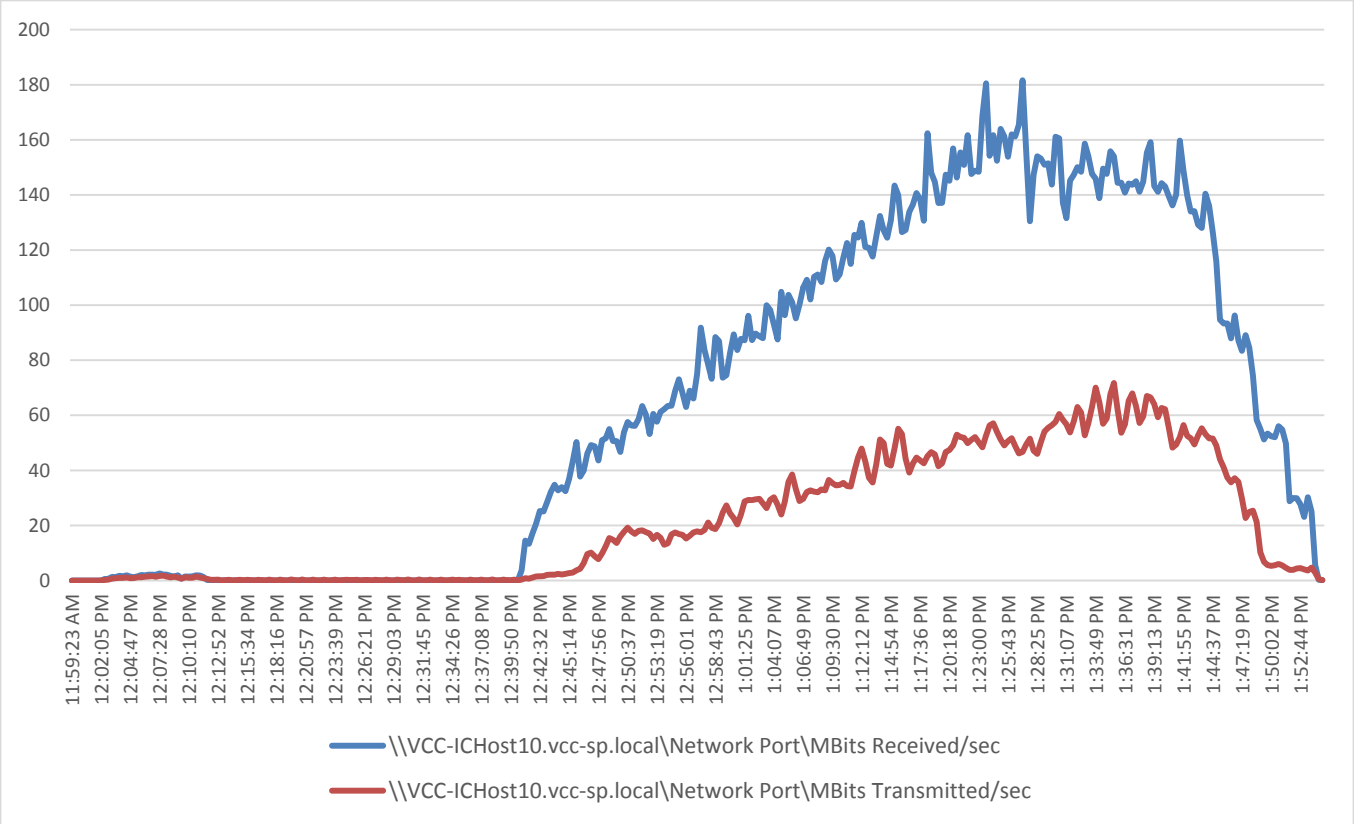
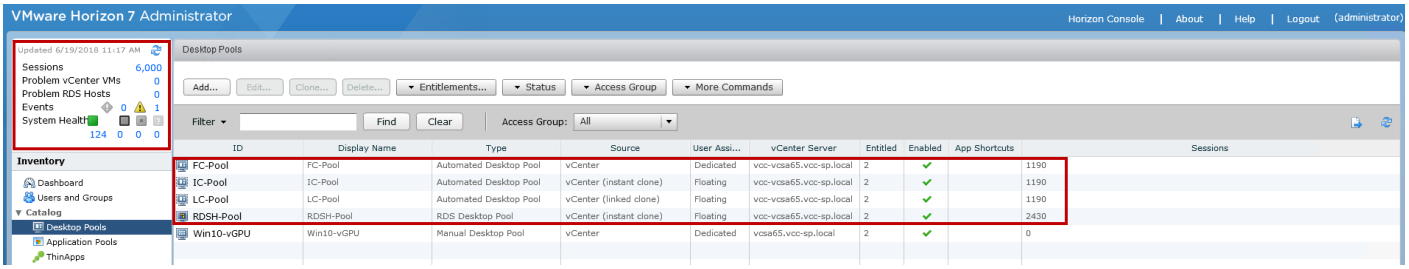


Figure 123Full Scale | 6000 Mixed Users | Horizon Administrator Console Reporting 2430 Active Sessions



Pure Storage FlashArray //X70 Storage System Graph for 6000 Users Mixed Workload test

Figure 124Full Scale 6000 Mixed User Boot Storm – Pure Storage //X70 System Web UI Performance Chart

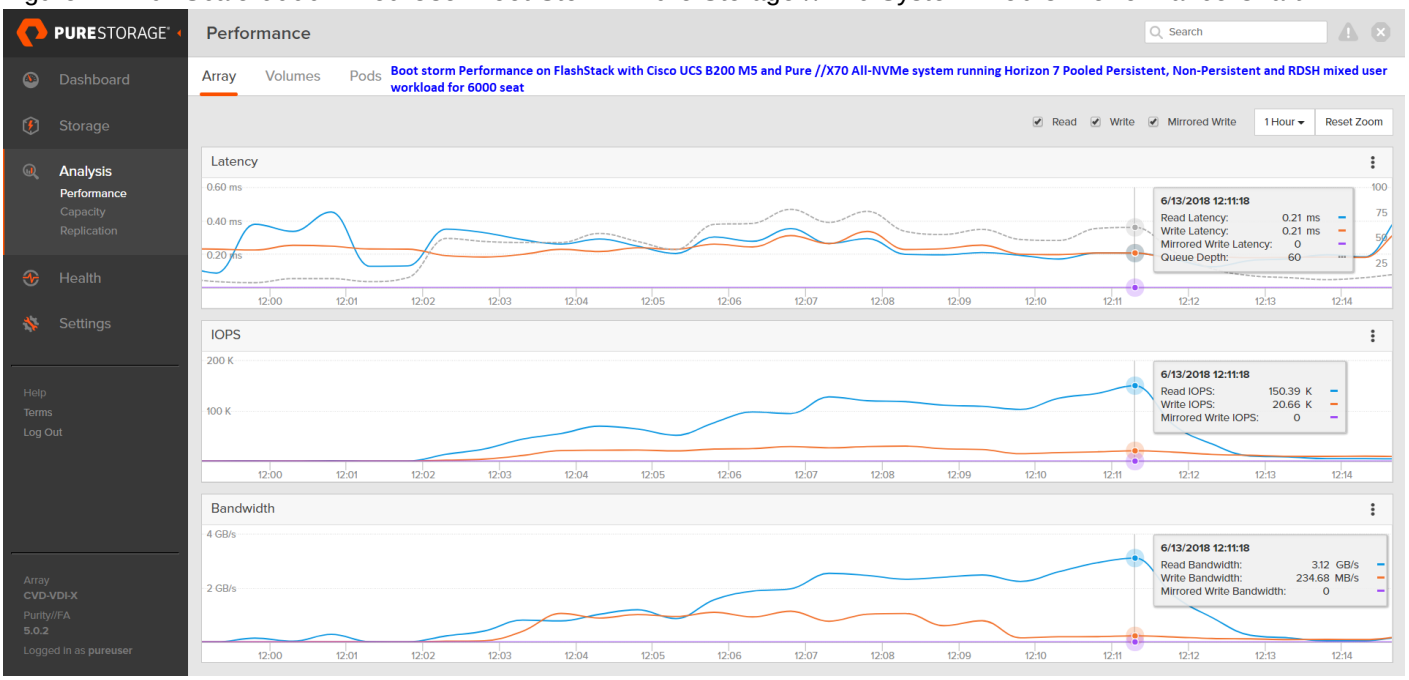
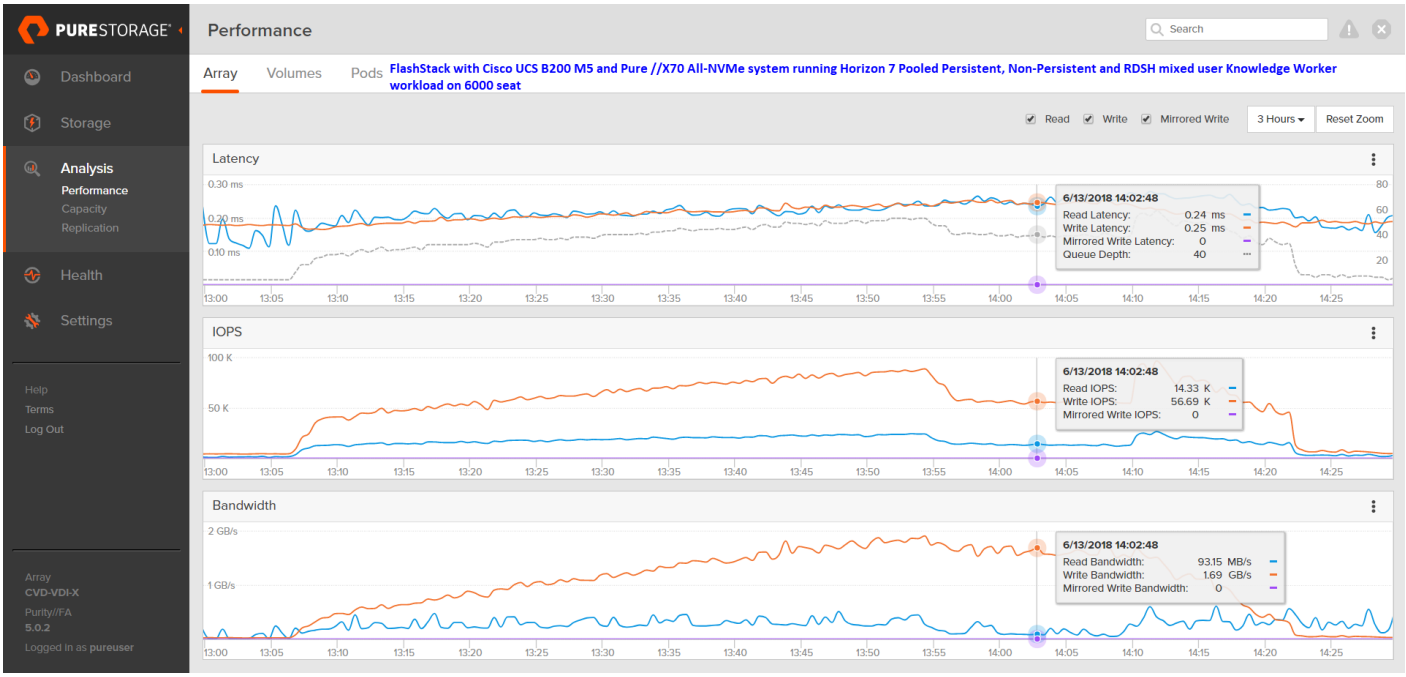


Figure 125 Full Scale 6000 Mixed User Running Knowledge Worker Workload – Pure Storage //X70 System Web UI Performance Chart



Summary

FlashStack delivers a platform for Enterprise End User Computing deployments and cloud datacenters using Cisco UCS Blade and Rack Servers, Cisco Fabric Interconnects, Cisco Nexus 9000 switches, Cisco MDS switches and FC-attached Pure Storage //X70 Storage Array. FlashStack is designed and validated using compute, network and storage best practices and high availability to reduce deployment time, project risk and IT costs while maintaining scalability and flexibility for addressing a multitude of IT initiatives. This CVD validates the design, performance, management, scalability, and resilience that FlashStack provides to customers wishing to deploy enterprise-class VDI for 6000 users at a time.

Get More Business Value with Services

Whether you are planning your next-generation environment, need specialized know-how for a major deployment, or want to get the most from your current storage, Cisco Advanced Services, Pure Storage //X70 storage and our certified partners can help. We collaborate with you to enhance your IT capabilities through a full portfolio of services that covers your IT lifecycle with:

- Strategy services to align IT with your business goals:
- Design services to architect your best storage environment
- Deploy and transition services to implement validated architectures and prepare your storage environment
- Operations services to deliver continuous operations while driving operational excellence and efficiency.

In addition, Cisco Advanced Services and Pure Storage Support provide in-depth knowledge transfer and education services that give you access to our global technical resources and intellectual property.

About the Authors

Hardik Patel, Senior Technical Marketing Engineer, Desktop Virtualization and Graphics Solutions, Cisco Systems, Inc.

Hardik is a subject matter expert on Cisco HyperFlex, Cisco Unified Computing System, Cisco Nexus Switching, VMware vSphere and VMware Horizon end user computing. Hardik is a member of the Cisco's Computer Systems Product Group team.

Acknowledgements

For their support and contribution to the design, validation, and creation of this Cisco Validated Design, we would like to acknowledge their contribution and expertise that resulted in developing this document:

- Mike Brennan, Product Manager, Desktop Virtualization and Graphics Solutions, Cisco Systems, Inc.
- Kyle Grossmiller, Solutions Architect, Pure Storage, Inc.
- Bhumik Patel, Technical Alliances Director, VMware, Inc.

References

This section provides links to additional information for each partner's solution component of this document.

Cisco UCS B-Series Servers

- http://www.cisco.com/c/en/us/products/servers-unified-computing/ucs-b-series-blade-servers/index.html
- https://www.cisco.com/c/dam/en/us/products/collateral/servers-unified-computing/ucs-b-series-blade-servers/b200m5-specsheet.pdf
- https://www.cisco.com/c/en/us/products/servers-unified-computing/ucs-b-series-blade-servers/datasheet-listing.html
- https://www.cisco.com/c/en/us/support/servers-unified-computing/ucs-b200-m5-blade-server/model.html
- https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/hw/blade-servers/B200M5.pdf

Cisco UCS Manager Configuration Guides

- http://www.cisco.com/c/en/us/support/servers-unified-computing/ucs-manager/products-installation-and-configuration-guides-list.html
- https://www.cisco.com/c/en/us/support/servers-unified-computing/ucs-manager/products-release-notes-list.html

Cisco UCS Virtual Interface Cards

- http://www.cisco.com/c/en/us/products/collateral/interfaces-modules/ucs-virtual-interface-card-1340/datasheet-c78-732517.html
- http://www.cisco.com/c/en/us/products/interfaces-modules/ucs-virtual-interface-card-1340/index.html

Cisco Nexus Switching References

- http://www.cisco.com/c/en/us/products/collateral/switches/nexus-9000-series-switches/datasheet-c78-736967.html
- https://www.cisco.com/c/en/us/products/switches/nexus-93180yc-fx-switch/index.html

Cisco MDS 9000 Service Switch References

- http://www.cisco.com/c/en/us/products/storage-networking/mds-9000-series-multilayer-switches/index.html

- http://www.cisco.com/c/en/us/products/storage-networking/product-listing.html
- https://www.cisco.com/c/en/us/products/collateral/storage-networking/mds-9148s-16g-multilayer-fabric-switch/datasheet-c78-731523.html

VMware References

- https://docs.vmware.com/en/VMware-vSphere/index.html
- https://docs.vmware.com/en/VMware-Horizon-7/7.4/rn/horizon-74-view-release-notes.html
- https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/products/horizon/vmware-horizon-7-datasheet.pdf
- https://labs.vmware.com/flings/vmware-os-optimization-tool
- https://docs.vmware.com/en/VMware-Horizon-7/index.html
- https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/techpaper/vmware-view-virtual-desktops-windows-optimization.pdf
- https://techzone.vmware.com/sites/default/files/vmware-horizon-7-view-blast-extreme-display-protocol.pdf

Microsoft References

- https://social.technet.microsoft.com/wiki/contents/articles/37890.windows-server-2016-installation.aspx
- https://support.microsoft.com/en-us/help/2833839/guidelines-for-installing-the-remote-desktop-session-host-role-service
- https://technet.microsoft.com/en-us/library/hh831447(v=ws.11).aspx

Login VSI Documentation

- https://www.loginvsi.com/documentation/Main_Page
- https://www.loginvsi.com/documentation/Start_your_first_test

Pure Storage Reference Documents

- https://www.flashstack.com/
- https://www.purestorage.com/content/dam/purestorage/pdf/datasheets/ps_ds_flasharray_03.pdf
- https://www.purestorage.com
- https://www.purestorage.com/products/evergreen-subscriptions.html
- https://www.purestorage.com/solutions/infrastructure/vdi.html

- https://www.purestorage.com/solutions/infrastructure/vdi-calculator.html

Appendix

Ethernet Network Configuration

The following section provides a detailed procedure for configuring the Cisco Nexus 9000 Switches used in this study.

Cisco Nexus 9372PX-A Configuration

```
AAD17-NX9K-A(config)# sh running-config

!Command: show running-config

!Time: Thu Feb 22 18:41:17 2018

version 7.0(3)I7(2)

switchname AAD17-NX9K-A

class-map type network-qos class-fcoe

match qos-group 1

class-map type network-qos class-all-flood

match qos-group 2

class-map type network-qos class-ip-multicast

match qos-group 2

policy-map type network-qos jumbo

  class type network-qos class-fcoe

    mtu 2158

  class type network-qos class-default

    mtu 9216

install feature-set fcoe-npv

vdc AAD17-NX9K-A id 1

  allow feature-set fcoe-npv

  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
feature-set fcoe-npv

feature telnet
cfs eth distribute
feature interface-vlan
feature hsrp
feature lacp
feature dhcp
feature vpc
feature lldp

no password strength-check
username admin password 5 $5$d3vc8gvD$hmf.YoRRPcqZ2dDGV2IaVKYZsPSPls8E9bpUzMciMZ0  role
network-admin
ip domain-lookup
system default switchport
class-map type qos match-all class-fcoe
policy-map type qos jumbo
  class class-default
    set qos-group 0
system qos
  service-policy type network-qos jumbo
copp profile lenient
snmp-server user admin network-admin auth md5 0xc9a73d344387b8db2dc0f3fc624240ac priv
0xc9a73d344387b8db2dc0f3fc624
```

```
240ac localizedkey
rmon event 1 description FATAL(1) owner PMON@FATAL
rmon event 2 description CRITICAL(2) owner PMON@CRITICAL
rmon event 3 description ERROR(3) owner PMON@ERROR
rmon event 4 description WARNING(4) owner PMON@WARNING
rmon event 5 description INFORMATION(5) owner PMON@INFO
ntp server 10.10.70.2 use-vrf default
ntp peer 10.10.70.3 use-vrf default
ntp server 72.163.32.44 use-vrf management
ntp logging
ntp master 8

vlan 1,70-76
vlan 70
  name InBand-Mgmt-SP
vlan 71
  name Infra-Mgmt-SP
vlan 72
  name VM-Network-SP
vlan 73
  name vMotion-SP
vlan 74
  name Storage_A-SP
vlan 75
  name Storage_B-SP
vlan 76
  name Launcher-SP
service dhcp
ip dhcp relay
```

```
ip dhcp relay information option
ipv6 dhcp relay
vrf context management
  ip route 0.0.0.0/0 10.29.164.1
hardware access-list tcam region ing-racl 1536
hardware access-list tcam region ing-redirect 256
vpc domain 70
  role priority 1000
  peer-keepalive destination 10.29.164.234 source 10.29.164.233
interface Vlan1
  no shutdown
  ip address 10.29.164.241/24
interface Vlan70
  no shutdown
  ip address 10.10.70.2/24
  hsrp version 2
  hsrp 70
    preempt
    priority 110
    ip 10.10.70.1
interface Vlan71
  no shutdown
  ip address 10.10.71.2/24
  hsrp version 2
  hsrp 71
    preempt
    priority 110
    ip 10.10.71.1
interface Vlan72
```

```
  no shutdown
  ip address 10.72.0.2/20
  hsrp version 2
  hsrp 72
    preempt
    priority 110
    ip 10.72.0.1
  ip dhcp relay address 10.10.71.11
  ip dhcp relay address 10.10.71.12
interface Vlan73
  no shutdown
  ip address 10.10.73.2/24
  hsrp version 2
  hsrp 73
    preempt
    priority 110
    ip 10.10.73.1
interface Vlan74
  no shutdown
  ip address 10.10.74.2/24
  hsrp version 2
  hsrp 74
    preempt
    priority 110
    ip 10.10.74.1
interface Vlan75
  no shutdown
  ip address 10.10.75.2/24
  hsrp version 2
```

```
  hsrp 75
    preempt
    priority 110
    ip 10.10.75.1
interface Vlan76
  no shutdown
  ip address 10.10.76.2/23
  hsrp version 2
  hsrp 76
    preempt
    priority 110
    ip 10.10.76.1
  ip dhcp relay address 10.10.71.11
  ip dhcp relay address 10.10.71.12
interface port-channel10
interface port-channel11
  description FI-Uplink-D17
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  spanning-tree port type edge trunk
  mtu 9216
  service-policy type qos input jumbo
  vpc 11
interface port-channel12
  description FI-Uplink-D17
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  spanning-tree port type edge trunk
  mtu 9216
```

```
  service-policy type qos input jumbo
  vpc 12
interface port-channel13
  description FI-Uplink-D16
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  spanning-tree port type edge trunk
  mtu 9216
  service-policy type qos input jumbo
  vpc 13
interface port-channel14
  description FI-Uplink-D16
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  spanning-tree port type edge trunk
  mtu 9216
  service-policy type qos input jumbo
  vpc 14
interface port-channel70
  description vPC-PeerLink
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  spanning-tree port type network
  service-policy type qos input jumbo
  vpc peer-link

interface Ethernet1/1

interface Ethernet1/2
```

```
interface Ethernet1/3
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  mtu 9216
  channel-group 13 mode active

interface Ethernet1/4
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  mtu 9216
  channel-group 13 mode active

interface Ethernet1/5
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  mtu 9216
  channel-group 14 mode active

interface Ethernet1/6
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  mtu 9216
  channel-group 14 mode active

interface Ethernet1/7
interface Ethernet1/8
interface Ethernet1/9
interface Ethernet1/10
```

```
interface Ethernet1/11
interface Ethernet1/12
interface Ethernet1/13
interface Ethernet1/14
interface Ethernet1/15
interface Ethernet1/16
interface Ethernet1/17
interface Ethernet1/18
interface Ethernet1/19
interface Ethernet1/20
interface Ethernet1/21
interface Ethernet1/22
interface Ethernet1/23
interface Ethernet1/24
interface Ethernet1/25
interface Ethernet1/26
interface Ethernet1/27
interface Ethernet1/28
interface Ethernet1/29
interface Ethernet1/30
interface Ethernet1/31
interface Ethernet1/32
interface Ethernet1/33
interface Ethernet1/34
interface Ethernet1/35
interface Ethernet1/36
interface Ethernet1/37
interface Ethernet1/38
interface Ethernet1/39
```

```
interface Ethernet1/40
interface Ethernet1/41
interface Ethernet1/42
interface Ethernet1/43
interface Ethernet1/44
interface Ethernet1/45
interface Ethernet1/46
interface Ethernet1/47
interface Ethernet1/48
interface Ethernet1/49
interface Ethernet1/50
interface Ethernet1/51
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  mtu 9216
  channel-group 11 mode active
interface Ethernet1/52
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  mtu 9216
  channel-group 12 mode active
interface Ethernet1/53
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  channel-group 70 mode active
interface Ethernet1/54
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  channel-group 70 mode active
```

```
interface mgmt0
  vrf member management
  ip address 10.29.164.233/24
line console
line vty
boot nxos bootflash:/nxos.7.0.3.I7.2.bin
no system default switchport shutdown
```

Cisco Nexus 9372PX-B Configuration

```
AAD17-NX9K-B(config)# sh running-config
!Command: show running-config
!Time: Thu Feb 22 18:41:17 2018
version 7.0(3)I7(2)
switchname AAD17-NX9K-B
class-map type network-qos class-fcoe
match qos-group 1
class-map type network-qos class-all-flood
match qos-group 2
class-map type network-qos class-ip-multicast
match qos-group 2
policy-map type network-qos jumbo
  class type network-qos class-fcoe
    mtu 2158
  class type network-qos class-default
    mtu 9216
install feature-set fcoe-npv
vdc AAD17-NX9K-B id 1
  allow feature-set fcoe-npv
  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
feature-set fcoe-npv
feature telnet
cfs eth distribute
feature interface-vlan
feature hsrp
feature lacp
feature dhcp
feature vpc
feature lldp
no password strength-check
username admin password 5 $5$/48.OHa8$g6pOMLlwrzqxJesMYoP5CNphujBksPPRjn4l3iFfOp.  role
network-admin
ip domain-lookup
system default switchport
class-map type qos match-all class-fcoe
policy-map type qos jumbo
  class class-default
    set qos-group 0
system qos
  service-policy type network-qos jumbo
copp profile lenient
snmp-server user admin network-admin auth md5 0x6d450e3d5a3927ddee1dadd30e5f616f priv
0x6d450e3d5a3927ddee1dadd30e5
```

```
f616f localizedkey
rmon event 1 description FATAL(1) owner PMON@FATAL
rmon event 2 description CRITICAL(2) owner PMON@CRITICAL
rmon event 3 description ERROR(3) owner PMON@ERROR
rmon event 4 description WARNING(4) owner PMON@WARNING
rmon event 5 description INFORMATION(5) owner PMON@INFO
ntp peer 10.10.70.2 use-vrf default
ntp server 10.10.70.3 use-vrf default
ntp server 72.163.32.44 use-vrf management
ntp logging
ntp master 8
vlan 1,70-76
vlan 70
  name InBand-Mgmt-SP
vlan 71
  name Infra-Mgmt-SP
vlan 72
  name VM-Network-SP
vlan 73
  name vMotion-SP
vlan 74
  name Storage_A-SP
vlan 75
  name Storage_B-SP
vlan 76
  name Launcher-SP
service dhcp
ip dhcp relay
ip dhcp relay information option
```

```
ipv6 dhcp relay
vrf context management
  ip route 0.0.0.0/0 10.29.164.1
hardware access-list tcam region ing-racl 1536
hardware access-list tcam region ing-redirect 256
vpc domain 70
  role priority 2000
  peer-keepalive destination 10.29.164.233 source 10.29.164.234
interface Vlan1
  no shutdown
  ip address 10.29.164.240/24
interface Vlan70
  no shutdown
  ip address 10.10.70.3/24
  hsrp version 2
  hsrp 70
    preempt
    priority 110
    ip 10.10.70.1
interface Vlan71
  no shutdown
  ip address 10.10.71.3/24
  hsrp version 2
  hsrp 71
    preempt
    priority 110
    ip 10.10.71.1
interface Vlan72
  no shutdown
```

```
  ip address 10.72.0.3/20
  hsrp version 2
  hsrp 72
    preempt
    priority 110
    ip 10.72.0.1
  ip dhcp relay address 10.10.71.11
  ip dhcp relay address 10.10.71.12
interface Vlan73
  no shutdown
  ip address 10.10.73.3/24
  hsrp version 2
  hsrp 73
    preempt
    priority 110
    ip 10.10.73.1
interface Vlan74
  no shutdown
  ip address 10.10.74.3/24
  hsrp version 2
  hsrp 74
    preempt
    priority 110
    ip 10.10.74.1
interface Vlan75
  no shutdown
  ip address 10.10.75.3/24
  hsrp version 2
  hsrp 75
```

```
    preempt
    priority 110
    ip 10.10.75.1
interface Vlan76
  no shutdown
  ip address 10.10.76.3/23
  hsrp version 2
  hsrp 76
    preempt
    priority 110
    ip 10.10.76.1
  ip dhcp relay address 10.10.71.11
  ip dhcp relay address 10.10.71.12
interface port-channel10
interface port-channel11
  description FI-Uplink-D17
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  spanning-tree port type edge trunk
  mtu 9216
  service-policy type qos input jumbo
  vpc 11

interface port-channel12
  description FI-Uplink-D17
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  spanning-tree port type edge trunk
  mtu 9216
```

```
  service-policy type qos input jumbo
  vpc 12
interface port-channel13
  description FI-Uplink-D16
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  spanning-tree port type edge trunk
  mtu 9216
  service-policy type qos input jumbo
  vpc 13
interface port-channel14
  description FI-Uplink-D16
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  spanning-tree port type edge trunk
  mtu 9216
  service-policy type qos input jumbo
  vpc 14
interface port-channel70
  description vPC-PeerLink
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  spanning-tree port type network
  service-policy type qos input jumbo
  vpc peer-link
interface Ethernet1/1
  switchport access vlan 70
interface Ethernet1/2
interface Ethernet1/3
```

```
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  mtu 9216
  channel-group 13 mode active
interface Ethernet1/4
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  mtu 9216
  channel-group 13 mode active
interface Ethernet1/5
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  mtu 9216
  channel-group 14 mode active
interface Ethernet1/6
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  mtu 9216
  channel-group 14 mode active
interface Ethernet1/7
interface Ethernet1/8
interface Ethernet1/9

interface Ethernet1/10
interface Ethernet1/11
interface Ethernet1/12
interface Ethernet1/13
interface Ethernet1/14
interface Ethernet1/15
```

```
interface Ethernet1/16
interface Ethernet1/17
interface Ethernet1/18
interface Ethernet1/19
interface Ethernet1/20
interface Ethernet1/21
interface Ethernet1/22
interface Ethernet1/23
interface Ethernet1/24
interface Ethernet1/25
interface Ethernet1/26
interface Ethernet1/27
interface Ethernet1/28
interface Ethernet1/29
interface Ethernet1/30
interface Ethernet1/31
interface Ethernet1/32
interface Ethernet1/33
interface Ethernet1/34
interface Ethernet1/35
interface Ethernet1/36
interface Ethernet1/37
interface Ethernet1/38
interface Ethernet1/39
interface Ethernet1/40
interface Ethernet1/41
interface Ethernet1/42
interface Ethernet1/43
interface Ethernet1/44
```

```
interface Ethernet1/45
interface Ethernet1/46
interface Ethernet1/47
interface Ethernet1/48
interface Ethernet1/49
interface Ethernet1/50
interface Ethernet1/51
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  mtu 9216
  channel-group 11 mode active
interface Ethernet1/52
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  mtu 9216
  channel-group 12 mode active
interface Ethernet1/53
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  channel-group 70 mode active
interface Ethernet1/54
  switchport mode trunk
  switchport trunk allowed vlan 1,70-76
  channel-group 70 mode active
interface mgmt0
  vrf member management
  ip address 10.29.164.234/24
line console
line vty
```

```
boot nxos bootflash:/nxos.7.0.3.I7.2.bin

no system default switchport shutdown
```

Fibre Channel Network Configuration

The following section provides a detailed procedure for configuring the Cisco MDS 9100 Switches used in this study.

Cisco MDS 9148S-A Configuration

```
!Command: show running-config

!Time: Fri Mar  9 23:27:26 2018

version 8.1(1)

power redundancy-mode redundant

feature npiv

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

no password strength-check

username admin password 5 $1$DDq8vF1x$EwCSM0O3dlXZ4jlPy9ZoC.  role network-admin

ip domain-lookup

ip host MDS-A  10.29.164.238

aaa group server radius radius

snmp-server contact jnichols

snmp-server user admin network-admin auth md5 0x2efbf582e573df2038164f1422c231fe

 priv 0x2efbf582e573df2038164f1422c231fe localizedkey


snmp-server host 10.155.160.192 traps version 2c public udp-port 1163
```

```
snmp-server host 10.29.132.18 traps version 2c public udp-port 1163
snmp-server host 10.29.164.130 traps version 2c public udp-port 1163
rmon event 1 log trap public description FATAL(1) owner PMON@FATAL
rmon event 2 log trap public description CRITICAL(2) owner PMON@CRITICAL
rmon event 3 log trap public description ERROR(3) owner PMON@ERROR
rmon event 4 log trap public description WARNING(4) owner PMON@WARNING
rmon event 5 log trap public description INFORMATION(5) owner PMON@INFO
snmp-server community public group network-operator
vsan database
  vsan 3 name "SP-Launcher-A"
  vsan 100 name "FlashStack-VCC-CVD-Fabric-A"
  vsan 400 name "FlashStack-A"
device-alias database
  device-alias name X70-CT0-FC0 pwwn 52:4a:93:75:dd:91:0a:00
  device-alias name X70-CT0-FC2 pwwn 52:4a:93:75:dd:91:0a:02
  device-alias name X70-CT1-FC1 pwwn 52:4a:93:75:dd:91:0a:11
  device-alias name X70-CT1-FC3 pwwn 52:4a:93:75:dd:91:0a:13
  device-alias name Infra01-8-hba1 pwwn 20:00:00:25:b5:3a:00:4f
  device-alias name Infra02-16-hba1 pwwn 20:00:00:25:b5:3a:00:2f
  device-alias name VCC-Infra01-HBA0 pwwn 20:00:00:25:b5:aa:17:1e
  device-alias name VCC-Infra01-HBA2 pwwn 20:00:00:25:b5:aa:17:1f
  device-alias name VCC-Infra02-HBA0 pwwn 20:00:00:25:b5:aa:17:3e
  device-alias name VCC-Infra02-HBA2 pwwn 20:00:00:25:b5:aa:17:3f
  device-alias name VCC-WLHost01-HBA0 pwwn 20:00:00:25:b5:aa:17:00
  device-alias name VCC-WLHost01-HBA2 pwwn 20:00:00:25:b5:aa:17:01
  device-alias name VCC-WLHost02-HBA0 pwwn 20:00:00:25:b5:aa:17:02
  device-alias name VCC-WLHost02-HBA2 pwwn 20:00:00:25:b5:aa:17:03
  device-alias name VCC-WLHost03-HBA0 pwwn 20:00:00:25:b5:aa:17:04
  device-alias name VCC-WLHost03-HBA2 pwwn 20:00:00:25:b5:aa:17:05
```

```
device-alias name VCC-WLHost04-HBA0 pwwn 20:00:00:25:b5:aa:17:06
device-alias name VCC-WLHost04-HBA2 pwwn 20:00:00:25:b5:aa:17:07
device-alias name VCC-WLHost05-HBA0 pwwn 20:00:00:25:b5:aa:17:08
device-alias name VCC-WLHost05-HBA2 pwwn 20:00:00:25:b5:aa:17:09
device-alias name VCC-WLHost06-HBA0 pwwn 20:00:00:25:b5:aa:17:0a
device-alias name VCC-WLHost06-HBA2 pwwn 20:00:00:25:b5:aa:17:0b
device-alias name VCC-WLHost07-HBA0 pwwn 20:00:00:25:b5:aa:17:0c
device-alias name VCC-WLHost07-HBA2 pwwn 20:00:00:25:b5:aa:17:0d
device-alias name VCC-WLHost08-HBA0 pwwn 20:00:00:25:b5:aa:17:0e
device-alias name VCC-WLHost08-HBA2 pwwn 20:00:00:25:b5:aa:17:0f
device-alias name VCC-WLHost09-HBA0 pwwn 20:00:00:25:b5:aa:17:10
device-alias name VCC-WLHost09-HBA2 pwwn 20:00:00:25:b5:aa:17:11
device-alias name VCC-WLHost10-HBA0 pwwn 20:00:00:25:b5:aa:17:12
device-alias name VCC-WLHost10-HBA2 pwwn 20:00:00:25:b5:aa:17:13
device-alias name VCC-WLHost11-HBA0 pwwn 20:00:00:25:b5:aa:17:14
device-alias name VCC-WLHost11-HBA2 pwwn 20:00:00:25:b5:aa:17:15
device-alias name VCC-WLHost12-HBA0 pwwn 20:00:00:25:b5:aa:17:16
device-alias name VCC-WLHost12-HBA2 pwwn 20:00:00:25:b5:aa:17:17
device-alias name VCC-WLHost13-HBA0 pwwn 20:00:00:25:b5:aa:17:18
device-alias name VCC-WLHost13-HBA2 pwwn 20:00:00:25:b5:aa:17:19
device-alias name VCC-WLHost14-HBA0 pwwn 20:00:00:25:b5:aa:17:1a
device-alias name VCC-WLHost14-HBA2 pwwn 20:00:00:25:b5:aa:17:1b
device-alias name VCC-WLHost15-HBA0 pwwn 20:00:00:25:b5:aa:17:1c
device-alias name VCC-WLHost15-HBA2 pwwn 20:00:00:25:b5:aa:17:1d
device-alias name VCC-WLHost16-HBA0 pwwn 20:00:00:25:b5:aa:17:20
device-alias name VCC-WLHost16-HBA2 pwwn 20:00:00:25:b5:aa:17:21
device-alias name VCC-WLHost17-HBA0 pwwn 20:00:00:25:b5:aa:17:22
device-alias name VCC-WLHost17-HBA2 pwwn 20:00:00:25:b5:aa:17:23
device-alias name VCC-WLHost18-HBA0 pwwn 20:00:00:25:b5:aa:17:24
```

```
  device-alias name VCC-WLHost18-HBA2 pwwn 20:00:00:25:b5:aa:17:25
  device-alias name VCC-WLHost19-HBA0 pwwn 20:00:00:25:b5:aa:17:26
  device-alias name VCC-WLHost19-HBA2 pwwn 20:00:00:25:b5:aa:17:27
  device-alias name VCC-WLHost20-HBA0 pwwn 20:00:00:25:b5:aa:17:28
  device-alias name VCC-WLHost20-HBA2 pwwn 20:00:00:25:b5:aa:17:29
  device-alias name VCC-WLHost21-HBA0 pwwn 20:00:00:25:b5:aa:17:2a
  device-alias name VCC-WLHost21-HBA2 pwwn 20:00:00:25:b5:aa:17:2b
  device-alias name VCC-WLHost22-HBA0 pwwn 20:00:00:25:b5:aa:17:2c
  device-alias name VCC-WLHost22-HBA2 pwwn 20:00:00:25:b5:aa:17:2d
  device-alias name VCC-WLHost23-HBA0 pwwn 20:00:00:25:b5:aa:17:2e
  device-alias name VCC-WLHost23-HBA2 pwwn 20:00:00:25:b5:aa:17:2f
  device-alias name VCC-WLHost24-HBA0 pwwn 20:00:00:25:b5:aa:17:30
  device-alias name VCC-WLHost24-HBA2 pwwn 20:00:00:25:b5:aa:17:31
  device-alias name VCC-WLHost25-HBA0 pwwn 20:00:00:25:b5:aa:17:32
  device-alias name VCC-WLHost25-HBA2 pwwn 20:00:00:25:b5:aa:17:33
  device-alias name VCC-WLHost26-HBA0 pwwn 20:00:00:25:b5:aa:17:34
  device-alias name VCC-WLHost26-HBA2 pwwn 20:00:00:25:b5:aa:17:35
  device-alias name VCC-WLHost27-HBA0 pwwn 20:00:00:25:b5:aa:17:36
  device-alias name VCC-WLHost27-HBA2 pwwn 20:00:00:25:b5:aa:17:37
  device-alias name VCC-WLHost28-HBA0 pwwn 20:00:00:25:b5:aa:17:38
  device-alias name VCC-WLHost28-HBA2 pwwn 20:00:00:25:b5:aa:17:39
  device-alias name VCC-WLHost29-HBA0 pwwn 20:00:00:25:b5:aa:17:3a
  device-alias name VCC-WLHost29-HBA2 pwwn 20:00:00:25:b5:aa:17:3b
  device-alias name VCC-WLHost30-HBA0 pwwn 20:00:00:25:b5:aa:17:3c
  device-alias name VCC-WLHost30-HBA2 pwwn 20:00:00:25:b5:aa:17:3d
device-alias commit
fcdomain fcid database
  vsan 100 wwn 20:04:00:de:fb:92:8d:00 fcid 0x810600 dynamic
  vsan 100 wwn 20:01:00:de:fb:92:8d:00 fcid 0x810700 dynamic
```

```
  vsan 100 wwn 20:00:00:25:b5:aa:17:1e fcid 0x810508 dynamic
!           [VCC-Infra01-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:02 fcid 0x810607 dynamic
!           [VCC-WLHost02-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:0a fcid 0x810407 dynamic
!           [VCC-WLHost06-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:0e fcid 0x810602 dynamic
!           [VCC-WLHost08-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:26 fcid 0x810503 dynamic
!           [VCC-WLHost19-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:2e fcid 0x810401 dynamic
!           [VCC-WLHost23-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:22 fcid 0x810710 dynamic
!           [VCC-WLHost17-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:28 fcid 0x81060c dynamic
!           [VCC-WLHost20-HBA0]
 vsan 100 wwn 20:00:00:25:b5:aa:17:24 fcid 0x810703 dynamic
!           [VCC-WLHost18-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:06 fcid 0x81040e dynamic
!           [VCC-WLHost04-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:0c fcid 0x810411 dynamic
!           [VCC-WLHost07-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:08 fcid 0x810707 dynamic
!           [VCC-WLHost05-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:00 fcid 0x810611 dynamic
!           [VCC-WLHost01-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:04 fcid 0x81050a dynamic
!           [VCC-WLHost03-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:16 fcid 0x810506 dynamic
```

```
!            [VCC-WLHost12-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:10 fcid 0x810512 dynamic
!            [VCC-WLHost09-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:12 fcid 0x81060a dynamic
!            [VCC-WLHost10-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:18 fcid 0x81050b dynamic
!            [VCC-WLHost13-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:20 fcid 0x810706 dynamic
!            [VCC-WLHost16-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:2c fcid 0x810507 dynamic
!            [VCC-WLHost22-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:2a fcid 0x810605 dynamic
!            [VCC-WLHost21-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:1a fcid 0x810701 dynamic
!            [VCC-WLHost14-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:1c fcid 0x810604 dynamic
!            [VCC-WLHost15-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:14 fcid 0x810713 dynamic
!            [VCC-WLHost11-HBA0]
  vsan 100 wwn 52:4a:93:75:dd:91:0a:02 fcid 0x810800 dynamic
!            [X70-CT0-FC2]
  vsan 100 wwn 52:4a:93:75:dd:91:0a:03 fcid 0x810900 dynamic
  vsan 100 wwn 52:4a:93:75:dd:91:0a:13 fcid 0x810a00 dynamic
!            [X70-CT1-FC3]
  vsan 100 wwn 52:4a:93:75:dd:91:0a:12 fcid 0x810b00 dynamic
  vsan 100 wwn 20:00:00:25:b5:aa:17:3e fcid 0x810513 dynamic
!            [VCC-Infra02-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:3f fcid 0x81060e dynamic
!            [VCC-Infra02-HBA2]
```

```
vsan 100 wwn 20:00:00:25:b5:aa:17:1f fcid 0x810409 dynamic
!          [VCC-Infra01-HBA2]
vsan 100 wwn 20:00:00:25:b5:aa:17:03 fcid 0x810709 dynamic
!          [VCC-WLHost02-HBA2]
vsan 100 wwn 20:00:00:25:b5:aa:17:01 fcid 0x810608 dynamic
!          [VCC-WLHost01-HBA2]
vsan 100 wwn 20:00:00:25:b5:aa:17:05 fcid 0x810402 dynamic
!          [VCC-WLHost03-HBA2]
vsan 100 wwn 20:00:00:25:b5:aa:17:07 fcid 0x810408 dynamic
!          [VCC-WLHost04-HBA2]
vsan 100 wwn 20:00:00:25:b5:aa:17:0b fcid 0x81050f dynamic
!          [VCC-WLHost06-HBA2]
vsan 100 wwn 20:00:00:25:b5:aa:17:09 fcid 0x81040d dynamic
!          [VCC-WLHost05-HBA2]
vsan 100 wwn 20:00:00:25:b5:aa:17:0d fcid 0x81070a dynamic
!          [VCC-WLHost07-HBA2]
vsan 100 wwn 20:00:00:25:b5:aa:17:0f fcid 0x810410 dynamic
!          [VCC-WLHost08-HBA2]
vsan 100 wwn 20:00:00:25:b5:aa:17:21 fcid 0x810603 dynamic
!          [VCC-WLHost16-HBA2]
vsan 100 wwn 20:00:00:25:b5:aa:17:23 fcid 0x81060d dynamic
!          [VCC-WLHost17-HBA2]
vsan 100 wwn 20:00:00:25:b5:aa:17:25 fcid 0x810501 dynamic
!          [VCC-WLHost18-HBA2]
vsan 100 wwn 20:00:00:25:b5:aa:17:27 fcid 0x810711 dynamic
!          [VCC-WLHost19-HBA2]
vsan 100 wwn 20:00:00:25:b5:aa:17:29 fcid 0x810505 dynamic

!          [VCC-WLHost20-HBA2]
```

```
  vsan 100 wwn 20:00:00:25:b5:aa:17:2b fcid 0x81070c dynamic
!           [VCC-WLHost21-HBA2]
  vsan 100 wwn 20:00:00:25:b5:aa:17:2d fcid 0x810413 dynamic
!           [VCC-WLHost22-HBA2]
  vsan 100 wwn 20:00:00:25:b5:aa:17:2f fcid 0x81040c dynamic
!           [VCC-WLHost23-HBA2]
  vsan 400 wwn 20:00:00:25:b5:3a:00:4d fcid 0x680207 dynamic
!           [VDI-9-hba1]
  vsan 400 wwn 20:00:00:25:b5:3a:00:3c fcid 0x680304 dynamic
!           [VDI-32-hba1]
  vsan 100 wwn 20:00:00:25:b5:aa:17:11 fcid 0x810403 dynamic
!           [VCC-WLHost09-HBA2]
  vsan 100 wwn 20:00:00:25:b5:aa:17:13 fcid 0x810601 dynamic
!           [VCC-WLHost10-HBA2]
  vsan 100 wwn 20:00:00:25:b5:aa:17:15 fcid 0x810609 dynamic
!           [VCC-WLHost11-HBA2]
  vsan 100 wwn 20:00:00:25:b5:aa:17:17 fcid 0x81040f dynamic
!           [VCC-WLHost12-HBA2]
  vsan 100 wwn 20:00:00:25:b5:aa:17:19 fcid 0x810404 dynamic
!           [VCC-WLHost13-HBA2]
  vsan 100 wwn 20:00:00:25:b5:aa:17:1b fcid 0x810412 dynamic
!           [VCC-WLHost14-HBA2]

  vsan 100 wwn 20:00:00:25:b5:aa:17:1d fcid 0x810702 dynamic
!           [VCC-WLHost15-HBA2]
  vsan 100 wwn 20:00:00:25:b5:aa:17:34 fcid 0x810405 dynamic
!           [VCC-WLHost26-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:32 fcid 0x810606 dynamic
!           [VCC-WLHost25-HBA0]
```

```
  vsan 100 wwn 20:00:00:25:b5:aa:17:33 fcid 0x81040a dynamic
!           [VCC-WLHost25-HBA2]
  vsan 100 wwn 20:00:00:25:b5:aa:17:35 fcid 0x81050d dynamic
!           [VCC-WLHost26-HBA2]
  vsan 400 wwn 20:00:00:25:b5:3a:00:2d fcid 0x68050a dynamic
!           [VDI-10-hba1]
  vsan 100 wwn 20:00:00:25:b5:aa:17:38 fcid 0x81060b dynamic
!           [VCC-WLHost28-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:39 fcid 0x810502 dynamic
!           [VCC-WLHost28-HBA2]
  vsan 100 wwn 20:00:00:25:b5:aa:17:30 fcid 0x810610 dynamic
!           [VCC-WLHost24-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:3a fcid 0x81050c dynamic
!           [VCC-WLHost29-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:36 fcid 0x810704 dynamic
!           [VCC-WLHost27-HBA0]
  vsan 100 wwn 20:00:00:25:b5:aa:17:3c fcid 0x810504 dynamic
!           [VCC-WLHost30-HBA0]
 vsan 100 wwn 20:00:00:25:b5:aa:17:3d fcid 0x810705 dynamic
!           [VCC-WLHost30-HBA2]
  vsan 100 wwn 20:00:00:25:b5:aa:17:3b fcid 0x810712 dynamic
!           [VCC-WLHost29-HBA2]
  vsan 100 wwn 20:00:00:25:b5:aa:17:37 fcid 0x81070f dynamic
!           [VCC-WLHost27-HBA2]
  vsan 100 wwn 52:4a:93:75:dd:91:0a:00 fcid 0x810c00 dynamic
!           [X70-CT0-FC0]
  vsan 100 wwn 52:4a:93:75:dd:91:0a:11 fcid 0x810d00 dynamic
!           [X70-CT1-FC1]
  !Active Zone Database Section for vsan 100
```

```
zone name FlaskStack-VCC-CVD-WLHost01 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!          [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!          [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!          [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!          [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:00
!          [VCC-WLHost01-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:01
!          [VCC-WLHost01-HBA2]

zone name FlaskStack-VCC-CVD-WLHost02 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!          [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!          [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!          [X70-CT1-FC1]

   member pwwn 52:4a:93:75:dd:91:0a:13
!          [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:02
!          [VCC-WLHost02-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:03
!          [VCC-WLHost02-HBA2]
```

```
zone name FlaskStack-VCC-CVD-WLHost03 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!          [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!          [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!          [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!          [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:04
!          [VCC-WLHost03-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:05
!          [VCC-WLHost03-HBA2]

zone name FlaskStack-VCC-CVD-WLHost04 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00

!          [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!          [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!          [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!          [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:06
!          [VCC-WLHost04-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:07
!          [VCC-WLHost04-HBA2]
```

```
zone name FlaskStack-VCC-CVD-WLHost05 vsan 100
    member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
    member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
    member pwwn 20:00:00:25:b5:aa:17:08
!           [VCC-WLHost05-HBA0]

    member pwwn 20:00:00:25:b5:aa:17:09
!           [VCC-WLHost05-HBA2]

zone name FlaskStack-VCC-CVD-WLHost06 vsan 100
    member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
    member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
    member pwwn 20:00:00:25:b5:aa:17:0a
!           [VCC-WLHost06-HBA0]
    member pwwn 20:00:00:25:b5:aa:17:0b
!           [VCC-WLHost06-HBA2]
```

```
zone name FlaskStack-VCC-CVD-WLHost07 vsan 100
    member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
    member pwwn 52:4a:93:75:dd:91:0a:11

!           [X70-CT1-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
    member pwwn 20:00:00:25:b5:aa:17:0c
!           [VCC-WLHost07-HBA0]
    member pwwn 20:00:00:25:b5:aa:17:0d
!           [VCC-WLHost07-HBA2]

zone name FlaskStack-VCC-CVD-WLHost08 vsan 100
    member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
    member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
    member pwwn 20:00:00:25:b5:aa:17:0e
!           [VCC-WLHost08-HBA0]
    member pwwn 20:00:00:25:b5:aa:17:0f
!           [VCC-WLHost08-HBA2]
```

```
zone name FlaskStack-VCC-CVD-WLHost09 vsan 100

   member pwwn 52:4a:93:75:dd:91:0a:00
!          [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!          [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!          [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!          [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:10
!          [VCC-WLHost09-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:11
!          [VCC-WLHost09-HBA2]

zone name FlaskStack-VCC-CVD-WLHost10 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!          [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!          [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!          [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!          [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:12

!          [VCC-WLHost10-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:13
!          [VCC-WLHost10-HBA2]
```

```
zone name FlaskStack-VCC-CVD-WLHost11 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!          [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!          [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!          [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!          [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:14
!          [VCC-WLHost11-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:15
!          [VCC-WLHost11-HBA2]

zone name FlaskStack-VCC-CVD-WLHost12 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!          [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!          [X70-CT0-FC2]

   member pwwn 52:4a:93:75:dd:91:0a:11
!          [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!          [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:16
!          [VCC-WLHost12-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:17
!          [VCC-WLHost12-HBA2]
```

```
zone name FlaskStack-VCC-CVD-WLHost13 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!          [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!          [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!          [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!          [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:18
!          [VCC-WLHost13-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:19
!          [VCC-WLHost13-HBA2]


zone name FlaskStack-VCC-CVD-WLHost14 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!          [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!          [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!          [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!          [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:1a
!          [VCC-WLHost14-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:1b
!          [VCC-WLHost14-HBA2]
```

```
zone name FlaskStack-VCC-CVD-WLHost15 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!          [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!          [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!          [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!          [X70-CT1-FC3]

   member pwwn 20:00:00:25:b5:aa:17:1c
!          [VCC-WLHost15-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:1d
!          [VCC-WLHost15-HBA2]

zone name FlaskStack-VCC-CVD-Infra01 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!          [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!          [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!          [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!          [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:1e
!          [VCC-Infra01-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:1f
!          [VCC-Infra01-HBA2]
```

```
zone name FlaskStack-VCC-CVD-WLHost16 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!          [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02

!          [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!          [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!          [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:20
!          [VCC-WLHost16-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:21
!          [VCC-WLHost16-HBA2]

zone name FlaskStack-VCC-CVD-WLHost17 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!          [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!          [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!          [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!          [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:22
!          [VCC-WLHost17-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:23
!          [VCC-WLHost17-HBA2]
```

```
[K
zone name FlaskStack-VCC-CVD-WLHost18 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!          [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!          [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!          [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!          [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:24
!          [VCC-WLHost18-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:25
!          [VCC-WLHost18-HBA2]

zone name FlaskStack-VCC-CVD-WLHost19 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!          [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!          [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!          [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13

!          [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:26
!          [VCC-WLHost19-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:27
```

```
!           [VCC-WLHost19-HBA2]

zone name FlaskStack-VCC-CVD-WLHost20 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:28
!           [VCC-WLHost20-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:29
!           [VCC-WLHost20-HBA2]

zone name FlaskStack-VCC-CVD-WLHost21 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]

   member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:2a
!           [VCC-WLHost21-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:2b
```

```
!           [VCC-WLHost21-HBA2]

zone name FlaskStack-VCC-CVD-WLHost22 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:2c
!           [VCC-WLHost22-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:2d

!           [VCC-WLHost22-HBA2]

zone name FlaskStack-VCC-CVD-WLHost23 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:2e
!           [VCC-WLHost23-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:2f
```

```
!           [VCC-WLHost23-HBA2]

zone name FlaskStack-VCC-CVD-WLHost24 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]

   member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:30
!           [VCC-WLHost24-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:31
!           [VCC-WLHost24-HBA2]

zone name FlaskStack-VCC-CVD-WLHost25 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:32
!           [VCC-WLHost25-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:33
```

```
!            [VCC-WLHost25-HBA2]

zone name FlaskStack-VCC-CVD-WLHost26 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00

!            [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!            [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!            [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!            [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:34
!            [VCC-WLHost26-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:35
!            [VCC-WLHost26-HBA2]

zone name FlaskStack-VCC-CVD-WLHost27 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!            [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!            [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!            [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!            [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:36
!            [VCC-WLHost27-HBA0]
```

```
    member pwwn 20:00:00:25:b5:aa:17:37
!           [VCC-WLHost27-HBA2]

zone name FlaskStack-VCC-CVD-WLHost28 vsan 100
    member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
    member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
    member pwwn 20:00:00:25:b5:aa:17:38
!           [VCC-WLHost28-HBA0]
    member pwwn 20:00:00:25:b5:aa:17:39
!           [VCC-WLHost28-HBA2]

zone name FlaskStack-VCC-CVD-WLHost29 vsan 100
    member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
    member pwwn 52:4a:93:75:dd:91:0a:11

!           [X70-CT1-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
    member pwwn 20:00:00:25:b5:aa:17:3a
```

```
!           [VCC-WLHost29-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:3b
!           [VCC-WLHost29-HBA2]

zone name FlaskStack-VCC-CVD-WLHost30 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:3c
!           [VCC-WLHost30-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:3d
!           [VCC-WLHost30-HBA2]

zone name FlaskStack-VCC-CVD-Infra02 vsan 100

   member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:3e
```

```
!            [VCC-Infra02-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:3f
!            [VCC-Infra02-HBA2]

zoneset name FlashStack-VCC-CVD vsan 100
   member FlaskStack-VCC-CVD-WLHost01
   member FlaskStack-VCC-CVD-WLHost02
   member FlaskStack-VCC-CVD-WLHost03
   member FlaskStack-VCC-CVD-WLHost04
   member FlaskStack-VCC-CVD-WLHost05
   member FlaskStack-VCC-CVD-WLHost06
   member FlaskStack-VCC-CVD-WLHost07
   member FlaskStack-VCC-CVD-WLHost08
   member FlaskStack-VCC-CVD-WLHost09
   member FlaskStack-VCC-CVD-WLHost10
   member FlaskStack-VCC-CVD-WLHost11
   member FlaskStack-VCC-CVD-WLHost12
   member FlaskStack-VCC-CVD-WLHost13
   member FlaskStack-VCC-CVD-WLHost14
   member FlaskStack-VCC-CVD-WLHost15
   member FlaskStack-VCC-CVD-Infra01
   member FlaskStack-VCC-CVD-WLHost16
   member FlaskStack-VCC-CVD-WLHost17
   member FlaskStack-VCC-CVD-WLHost18
   member FlaskStack-VCC-CVD-WLHost19
   member FlaskStack-VCC-CVD-WLHost20
   member FlaskStack-VCC-CVD-WLHost21
   member FlaskStack-VCC-CVD-WLHost22
   member FlaskStack-VCC-CVD-WLHost23
```

```
    member FlaskStack-VCC-CVD-WLHost24
    member FlaskStack-VCC-CVD-WLHost25
    member FlaskStack-VCC-CVD-WLHost26
    member FlaskStack-VCC-CVD-WLHost27
    member FlaskStack-VCC-CVD-WLHost28
    member FlaskStack-VCC-CVD-WLHost29
    member FlaskStack-VCC-CVD-WLHost30
    member FlaskStack-VCC-CVD-Infra02
zoneset activate name FlashStack-VCC-CVD vsan 100
do clear zone database vsan 100
!Full Zone Database Section for vsan 100
zone name FlaskStack-VCC-CVD-WLHost01 vsan 100
    member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
    member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
    member pwwn 20:00:00:25:b5:aa:17:00
!           [VCC-WLHost01-HBA0]
    member pwwn 20:00:00:25:b5:aa:17:01
!           [VCC-WLHost01-HBA2]

zone name FlaskStack-VCC-CVD-WLHost02 vsan 100
    member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:02
```

```
!            [X70-CT0-FC2]

   member pwwn 52:4a:93:75:dd:91:0a:11
!            [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!            [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:02
!            [VCC-WLHost02-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:03
!            [VCC-WLHost02-HBA2]
zone name FlaskStack-VCC-CVD-WLHost03 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!            [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!            [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!            [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!            [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:04
!            [VCC-WLHost03-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:05
!            [VCC-WLHost03-HBA2]
zone name FlaskStack-VCC-CVD-WLHost04 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!            [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!            [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
```

```
!            [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!            [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:06
!            [VCC-WLHost04-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:07
!            [VCC-WLHost04-HBA2]
zone name FlaskStack-VCC-CVD-WLHost05 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!            [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!            [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!            [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!            [X70-CT1-FC3]
  member pwwn 20:00:00:25:b5:aa:17:08
!            [VCC-WLHost05-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:09
!            [VCC-WLHost05-HBA2]
zone name FlaskStack-VCC-CVD-WLHost06 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!            [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!            [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!            [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!            [X70-CT1-FC3]
```

```
    member pwwn 20:00:00:25:b5:aa:17:0a
!           [VCC-WLHost06-HBA0]
    member pwwn 20:00:00:25:b5:aa:17:0b
!           [VCC-WLHost06-HBA2]
zone name FlaskStack-VCC-CVD-WLHost07 vsan 100
    member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
    member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
    member pwwn 20:00:00:25:b5:aa:17:0c
!           [VCC-WLHost07-HBA0]
    member pwwn 20:00:00:25:b5:aa:17:0d
!           [VCC-WLHost07-HBA2]
zone name FlaskStack-VCC-CVD-WLHost08 vsan 100
    member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
    member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
    member pwwn 20:00:00:25:b5:aa:17:0e
!           [VCC-WLHost08-HBA0]
    member pwwn 20:00:00:25:b5:aa:17:0f
```

```
!            [VCC-WLHost08-HBA2]
zone name FlaskStack-VCC-CVD-WLHost09 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!            [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!            [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!            [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!            [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:10
!            [VCC-WLHost09-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:11
!            [VCC-WLHost09-HBA2]
zone name FlaskStack-VCC-CVD-WLHost10 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!            [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!            [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!            [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13

!            [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:12
!            [VCC-WLHost10-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:13
!            [VCC-WLHost10-HBA2]
```

```
zone name FlaskStack-VCC-CVD-WLHost11 vsan 100
    member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
    member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
    member pwwn 20:00:00:25:b5:aa:17:14
!           [VCC-WLHost11-HBA0]
    member pwwn 20:00:00:25:b5:aa:17:15
!           [VCC-WLHost11-HBA2]
zone name FlaskStack-VCC-CVD-WLHost12 vsan 100
    member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
    member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
    member pwwn 20:00:00:25:b5:aa:17:16
!           [VCC-WLHost12-HBA0]
    member pwwn 20:00:00:25:b5:aa:17:17
!           [VCC-WLHost12-HBA2]
zone name FlaskStack-VCC-CVD-WLHost13 vsan 100
    member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
```

```
    member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
    member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
    member pwwn 20:00:00:25:b5:aa:17:18
!           [VCC-WLHost13-HBA0]
    member pwwn 20:00:00:25:b5:aa:17:19

!           [VCC-WLHost13-HBA2]
zone name FlaskStack-VCC-CVD-WLHost14 vsan 100
    member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
    member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
    member pwwn 20:00:00:25:b5:aa:17:1a
!           [VCC-WLHost14-HBA0]
    member pwwn 20:00:00:25:b5:aa:17:1b
!           [VCC-WLHost14-HBA2]
zone name FlaskStack-VCC-CVD-WLHost15 vsan 100
    member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
```

```
    member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]

    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
    member pwwn 20:00:00:25:b5:aa:17:1c
!           [VCC-WLHost15-HBA0]
    member pwwn 20:00:00:25:b5:aa:17:1d
!           [VCC-WLHost15-HBA2]
zone name FlaskStack-VCC-CVD-Infra01 vsan 100
    member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
    member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
    member pwwn 20:00:00:25:b5:aa:17:1e
!           [VCC-Infra01-HBA0]
    member pwwn 20:00:00:25:b5:aa:17:1f
!           [VCC-Infra01-HBA2]
zone name FlaskStack-VCC-CVD-WLHost16 vsan 100
    member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
    member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
```

```
    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
    member pwwn 20:00:00:25:b5:aa:17:20
!           [VCC-WLHost16-HBA0]
    member pwwn 20:00:00:25:b5:aa:17:21
!           [VCC-WLHost16-HBA2]
zone name FlaskStack-VCC-CVD-WLHost17 vsan 100
    member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
    member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
    member pwwn 20:00:00:25:b5:aa:17:22
!           [VCC-WLHost17-HBA0]

    member pwwn 20:00:00:25:b5:aa:17:23
!           [VCC-WLHost17-HBA2]
zone name FlaskStack-VCC-CVD-WLHost18 vsan 100
    member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
    member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
```

```
    member pwwn 20:00:00:25:b5:aa:17:24
!           [VCC-WLHost18-HBA0]
    member pwwn 20:00:00:25:b5:aa:17:25
!           [VCC-WLHost18-HBA2]
zone name FlaskStack-VCC-CVD-WLHost19 vsan 100
    member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
    member pwwn 52:4a:93:75:dd:91:0a:11

!           [X70-CT1-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
    member pwwn 20:00:00:25:b5:aa:17:26
!           [VCC-WLHost19-HBA0]
    member pwwn 20:00:00:25:b5:aa:17:27
!           [VCC-WLHost19-HBA2]
zone name FlaskStack-VCC-CVD-WLHost20 vsan 100
    member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
    member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
    member pwwn 20:00:00:25:b5:aa:17:28
!           [VCC-WLHost20-HBA0]
```

```
    member pwwn 20:00:00:25:b5:aa:17:29
!           [VCC-WLHost20-HBA2]
zone name FlaskStack-VCC-CVD-WLHost21 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
    member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
    member pwwn 20:00:00:25:b5:aa:17:2a
!           [VCC-WLHost21-HBA0]
    member pwwn 20:00:00:25:b5:aa:17:2b
!           [VCC-WLHost21-HBA2]
zone name FlaskStack-VCC-CVD-WLHost22 vsan 100
    member pwwn 52:4a:93:75:dd:91:0a:00
!           [X70-CT0-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:02
!           [X70-CT0-FC2]
    member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
    member pwwn 20:00:00:25:b5:aa:17:2c

!           [VCC-WLHost22-HBA0]
    member pwwn 20:00:00:25:b5:aa:17:2d
!           [VCC-WLHost22-HBA2]
```

```
zone name FlaskStack-VCC-CVD-WLHost23 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!          [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!          [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!          [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!          [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:2e
!          [VCC-WLHost23-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:2f
!          [VCC-WLHost23-HBA2]

zone name FlaskStack-VCC-CVD-WLHost24 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!          [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!          [X70-CT0-FC2]

   member pwwn 52:4a:93:75:dd:91:0a:11
!          [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!          [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:30
!          [VCC-WLHost24-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:31
!          [VCC-WLHost24-HBA2]
```

```
zone name FlaskStack-VCC-CVD-WLHost25 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!          [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!          [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!          [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!          [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:32
!          [VCC-WLHost25-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:33
!          [VCC-WLHost25-HBA2]
zone name FlaskStack-VCC-CVD-WLHost26 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!          [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!          [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!          [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!          [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:34
!          [VCC-WLHost26-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:35
!          [VCC-WLHost26-HBA2]
zone name FlaskStack-VCC-CVD-WLHost27 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
```

```
!            [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!            [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!            [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!            [X70-CT1-FC3]

   member pwwn 20:00:00:25:b5:aa:17:36
!            [VCC-WLHost27-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:37
!            [VCC-WLHost27-HBA2]
zone name FlaskStack-VCC-CVD-WLHost28 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!            [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!            [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!            [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!            [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:38
!            [VCC-WLHost28-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:39
!            [VCC-WLHost28-HBA2]
zone name FlaskStack-VCC-CVD-WLHost29 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!            [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
```

```
!            [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!            [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!            [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:3a
!            [VCC-WLHost29-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:3b
!            [VCC-WLHost29-HBA2]
zone name FlaskStack-VCC-CVD-WLHost30 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!            [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!            [X70-CT0-FC2]
   member pwwn 52:4a:93:75:dd:91:0a:11
!            [X70-CT1-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:13
!            [X70-CT1-FC3]
   member pwwn 20:00:00:25:b5:aa:17:3c
!            [VCC-WLHost30-HBA0]
   member pwwn 20:00:00:25:b5:aa:17:3d
!            [VCC-WLHost30-HBA2]

zone name FlaskStack-VCC-CVD-Infra02 vsan 100
   member pwwn 52:4a:93:75:dd:91:0a:00
!            [X70-CT0-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:02
!            [X70-CT0-FC2]
```

```
    member pwwn 52:4a:93:75:dd:91:0a:11
!           [X70-CT1-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:13
!           [X70-CT1-FC3]
    member pwwn 20:00:00:25:b5:aa:17:3e
!           [VCC-Infra02-HBA0]
    member pwwn 20:00:00:25:b5:aa:17:3f
!           [VCC-Infra02-HBA2]

zoneset name FlashStack-VCC-CVD vsan 100
    member FlaskStack-VCC-CVD-WLHost01
    member FlaskStack-VCC-CVD-WLHost02
    member FlaskStack-VCC-CVD-WLHost03
    member FlaskStack-VCC-CVD-WLHost04
    member FlaskStack-VCC-CVD-WLHost05
    member FlaskStack-VCC-CVD-WLHost06
    member FlaskStack-VCC-CVD-WLHost07
    member FlaskStack-VCC-CVD-WLHost08
    member FlaskStack-VCC-CVD-WLHost09
    member FlaskStack-VCC-CVD-WLHost10
    member FlaskStack-VCC-CVD-WLHost11
    member FlaskStack-VCC-CVD-WLHost12
    member FlaskStack-VCC-CVD-WLHost13
    member FlaskStack-VCC-CVD-WLHost14
    member FlaskStack-VCC-CVD-WLHost15
    member FlaskStack-VCC-CVD-Infra01
    member FlaskStack-VCC-CVD-WLHost16
    member FlaskStack-VCC-CVD-WLHost17
    member FlaskStack-VCC-CVD-WLHost18
```

```
    member FlaskStack-VCC-CVD-WLHost19
    member FlaskStack-VCC-CVD-WLHost20
    member FlaskStack-VCC-CVD-WLHost21
    member FlaskStack-VCC-CVD-WLHost22
    member FlaskStack-VCC-CVD-WLHost23
    member FlaskStack-VCC-CVD-WLHost24
    member FlaskStack-VCC-CVD-WLHost25
    member FlaskStack-VCC-CVD-WLHost26
    member FlaskStack-VCC-CVD-WLHost27
    member FlaskStack-VCC-CVD-WLHost28
    member FlaskStack-VCC-CVD-WLHost29

    member FlaskStack-VCC-CVD-WLHost30
    member FlaskStack-VCC-CVD-Infra02
!Active Zone Database Section for vsan 400
zone name a300_VDI-1-hba1 vsan 400
    member pwwn 20:00:00:25:b5:3a:00:3f
!           [VDI-1-hba1]
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
zone name a300_VDI-2-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:0f
!           [VDI-2-hba1]
```

```
zone name a300_VDI-3-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!          [a300-01-0g]

   member pwwn 20:03:00:a0:98:af:bd:e8
!          [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:1f
!          [VDI-3-hba1]
zone name a300_VDI-4-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!          [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!          [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:4e
!          [VDI-4-hba1]

zone name a300_VDI-5-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!          [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!          [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:2e
!          [VDI-5-hba1]

zone name a300_VDI-6-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8

!          [a300-01-0g]
```

```
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:3e
!           [VDI-6-hba1]

zone name a300_VDI-7-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:0e
!           [VDI-7-hba1]

zone name a300_Infra01-8-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:4f
!           [Infra01-8-hba1]

zone name a300_VDI-9-hba1 vsan 400

    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:4d
!           [VDI-9-hba1]
```

```
zone name a300_VDI-10-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!          [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!          [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:2d
!          [VDI-10-hba1]

zone name a300_VDI-11-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!          [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!          [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:3d
!          [VDI-11-hba1]

zone name a300_VDI-12-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!          [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!          [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:0d
!          [VDI-12-hba1]

zone name a300_VDI-13-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!          [a300-01-0g]
```

```
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:1d
!           [VDI-13-hba1]

zone name a300_VDI-14-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:4c
!           [VDI-14-hba1]

[K
zone name a300_VDI-15-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:2c
!           [VDI-15-hba1]

zone name a300_Infra02-16-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:2f
```

```
!            [Infra02-16-hba1]

zone name a300_VDI-17-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!            [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!            [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:0c

!            [VDI-17-hba1]

zone name a300_VDI-18-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!            [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!            [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:1c
!            [VDI-18-hba1]

zone name a300_VDI-19-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!            [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!            [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:4b
!            [VDI-19-hba1]

zone name a300_VDI-20-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
```

```
!            [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!            [a300-02-0g]

   member pwwn 20:00:00:25:b5:3a:00:2b
!            [VDI-20-hba1]

zone name a300_VDI-21-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!            [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!            [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:3b
!            [VDI-21-hba1]

zone name a300_VDI-22-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!            [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!            [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:0b
!            [VDI-22-hba1]

zone name a300_VDI-23-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!            [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8

!            [a300-02-0g]
```

```
    member pwwn 20:00:00:25:b5:3a:00:1b
!           [VDI-23-hba1]

zone name a300_VDI-24-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:4a
!           [VDI-24-hba1]

zone name a300_VDI-25-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:2a
!           [VDI-25-hba1]

zone name a300_VDI-26-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]

    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:3a
!           [VDI-26-hba1]

zone name a300_VDI-27-hba1 vsan 400
```

```
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:0a
!           [VDI-27-hba1]

zone name a300_VDI-28-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:1a
!           [VDI-28-hba1]

zone name a300_VDI-29-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8

!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:49
!           [VDI-29-hba1]

zone name a300_VDI-30-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
```

```
    member pwwn 20:00:00:25:b5:3a:00:39
!           [VDI-30-hba1]

zone name a300_VDI-31-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:1e
!           [VDI-31-hba1]

zone name a300_VDI-32-hba1 vsan 400

    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:3c
!           [VDI-32-hba1]

zoneset name FlashStack_FabricA vsan 400
    member a300_VDI-1-hba1
    member a300_VDI-2-hba1
    member a300_VDI-3-hba1
    member a300_VDI-4-hba1
    member a300_VDI-5-hba1
    member a300_VDI-6-hba1
    member a300_VDI-7-hba1
    member a300_Infra01-8-hba1
```

```
    member a300_VDI-9-hba1
    member a300_VDI-10-hba1
    member a300_VDI-11-hba1
    member a300_VDI-12-hba1
    member a300_VDI-13-hba1
    member a300_VDI-14-hba1
    member a300_VDI-15-hba1
    member a300_Infra02-16-hba1
    member a300_VDI-17-hba1
    member a300_VDI-18-hba1
    member a300_VDI-19-hba1
    member a300_VDI-20-hba1
    member a300_VDI-21-hba1
    member a300_VDI-22-hba1
    member a300_VDI-23-hba1
    member a300_VDI-24-hba1
    member a300_VDI-25-hba1
    member a300_VDI-26-hba1
    member a300_VDI-27-hba1
    member a300_VDI-28-hba1
    member a300_VDI-29-hba1
    member a300_VDI-30-hba1
    member a300_VDI-31-hba1
    member a300_VDI-32-hba1

zoneset activate name FlashStack_FabricA vsan 400
do clear zone database vsan 400
!Full Zone Database Section for vsan 400
zone name a300_VDI-1-hba1 vsan 400
```

```
    member pwwn 20:00:00:25:b5:3a:00:3f

!           [VDI-1-hba1]
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]

zone name a300_VDI-2-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:0f
!           [VDI-2-hba1]

zone name a300_VDI-3-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:1f
!           [VDI-3-hba1]

zone name a300_VDI-4-hba1 vsan 400

    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
```

```
!            [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:4e
!            [VDI-4-hba1]

zone name a300_VDI-5-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!            [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!            [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:2e
!            [VDI-5-hba1]

zone name a300_VDI-6-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!            [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!            [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:3e
!            [VDI-6-hba1]

zone name a300_VDI-7-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!            [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!            [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:0e
!            [VDI-7-hba1]
```

```
zone name a300_Infra01-8-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!          [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!          [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:1e
!          [VDI-31-hba1]

zone name a300_VDI-9-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!          [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!          [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:4d
!          [VDI-9-hba1]

zone name a300_VDI-10-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!          [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!          [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:2d
!          [VDI-10-hba1]

zone name a300_VDI-11-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!          [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!          [a300-02-0g]
```

```
    member pwwn 20:00:00:25:b5:3a:00:3d
!           [VDI-11-hba1]

zone name a300_VDI-12-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:0d

!           [VDI-12-hba1]

zone name a300_VDI-13-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:1d
!           [VDI-13-hba1]

zone name a300_VDI-14-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:4c
!           [VDI-14-hba1]

zone name a300_VDI-15-hba1 vsan 400
```

```
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]

    member pwwn 20:00:00:25:b5:3a:00:2c
!           [VDI-15-hba1]

zone name a300_Infra02-16-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:2f
!           [Infra02-16-hba1]

zone name a300_VDI-17-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:0c
!           [VDI-17-hba1]

zone name a300_VDI-18-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
```

```
!            [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:1c
!            [VDI-18-hba1]

zone name a300_VDI-19-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!            [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!            [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:4b
!            [VDI-19-hba1]

zone name a300_VDI-20-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!            [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!            [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:2b
!            [VDI-20-hba1]

zone name a300_VDI-21-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!            [a300-01-0g]

   member pwwn 20:03:00:a0:98:af:bd:e8
!            [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:3b
!            [VDI-21-hba1]
```

```
zone name a300_VDI-22-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!          [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!          [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:0b
!          [VDI-22-hba1]

zone name a300_VDI-23-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!          [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!          [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:1b
!          [VDI-23-hba1]

zone name a300_VDI-24-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8

!          [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!          [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:4a
!          [VDI-24-hba1]

zone name a300_VDI-25-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!          [a300-01-0g]
```

```
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:2a
!           [VDI-25-hba1]

zone name a300_VDI-26-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:3a
!           [VDI-26-hba1]

zone name a300_VDI-27-hba1 vsan 400

    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:0a
!           [VDI-27-hba1]

zone name a300_VDI-28-hba1 vsan 400
    member pwwn 20:01:00:a0:98:af:bd:e8
!           [a300-01-0g]
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:1a
!           [VDI-28-hba1]
```

```
zone name a300_VDI-29-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!          [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!          [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:49
!          [VDI-29-hba1]


zone name a300_VDI-30-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!          [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!          [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:39
!          [VDI-30-hba1]

zone name a300_VDI-31-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!          [a300-01-0g]
   member pwwn 20:03:00:a0:98:af:bd:e8
!          [a300-02-0g]
   member pwwn 20:00:00:25:b5:3a:00:1e
!          [VDI-31-hba1]

zone name a300_VDI-32-hba1 vsan 400
   member pwwn 20:01:00:a0:98:af:bd:e8
!          [a300-01-0g]
```

```
    member pwwn 20:03:00:a0:98:af:bd:e8
!           [a300-02-0g]
    member pwwn 20:00:00:25:b5:3a:00:3c
!           [VDI-32-hba1]
zoneset name FlashStack_FabricA vsan 400
    member a300_VDI-1-hba1
    member a300_VDI-2-hba1
    member a300_VDI-3-hba1
    member a300_VDI-4-hba1
    member a300_VDI-5-hba1
    member a300_VDI-6-hba1
    member a300_VDI-7-hba1
    member a300_Infra01-8-hba1
    member a300_VDI-9-hba1
    member a300_VDI-10-hba1
    member a300_VDI-11-hba1
    member a300_VDI-12-hba1
    member a300_VDI-13-hba1
    member a300_VDI-14-hba1
    member a300_VDI-15-hba1
    member a300_Infra02-16-hba1
    member a300_VDI-17-hba1
    member a300_VDI-18-hba1
    member a300_VDI-19-hba1
    member a300_VDI-20-hba1
    member a300_VDI-21-hba1
    member a300_VDI-22-hba1
    member a300_VDI-23-hba1
    member a300_VDI-24-hba1
```

```
    member a300_VDI-25-hba1
    member a300_VDI-26-hba1
    member a300_VDI-27-hba1
    member a300_VDI-28-hba1
    member a300_VDI-29-hba1
    member a300_VDI-30-hba1
    member a300_VDI-31-hba1
    member a300_VDI-32-hba1
interface mgmt0
  ip address 10.29.164.238 255.255.255.0
vsan database
  vsan 100 interface fc1/25
  vsan 100 interface fc1/26
  vsan 100 interface fc1/27
  vsan 100 interface fc1/28
  vsan 100 interface fc1/29
  vsan 100 interface fc1/30
  vsan 100 interface fc1/31
  vsan 100 interface fc1/32
  switchname MDS-A
no terminal log-all
line console
  terminal width  80
line vty
boot kickstart bootflash:/m9100-s5ek9-kickstart-mz.8.1.1.bin
boot system bootflash:/m9100-s5ek9-mz.8.1.1.bin
interface fc1/1
interface fc1/2
interface fc1/11
```

```
interface fc1/12
interface fc1/19
interface fc1/20
interface fc1/21
interface fc1/22
interface fc1/23
interface fc1/24
interface fc1/43
interface fc1/44
interface fc1/45
interface fc1/46
interface fc1/3
interface fc1/4
interface fc1/5
interface fc1/6
interface fc1/7
interface fc1/8
interface fc1/9
interface fc1/10
interface fc1/17
interface fc1/18
interface fc1/25
interface fc1/26
interface fc1/27
interface fc1/28
interface fc1/29
interface fc1/30
interface fc1/31
interface fc1/32
```

```
interface fc1/33
interface fc1/34
interface fc1/35
interface fc1/36
interface fc1/37
interface fc1/38
interface fc1/39
interface fc1/40
interface fc1/41
interface fc1/42
interface fc1/47
interface fc1/48
interface fc1/13
interface fc1/14
interface fc1/15
interface fc1/16
interface fc1/1
interface fc1/2
interface fc1/11
interface fc1/12
interface fc1/19
interface fc1/20
interface fc1/21
interface fc1/22
interface fc1/23
interface fc1/24
interface fc1/43
interface fc1/44
interface fc1/45
```

```
interface fc1/46

interface fc1/1
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/2
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/3
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/4
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/5
  port-license acquire
  no shutdown
interface fc1/6
  port-license acquire
  no shutdown
interface fc1/7
  port-license acquire
  no shutdown

interface fc1/8
```

```
  port-license acquire
  no shutdown
interface fc1/9
  port-license acquire
interface fc1/10
  port-license acquire
interface fc1/11
  port-license acquire
interface fc1/12
port-license acquire
interface fc1/13
  port-license acquire
  no shutdown
interface fc1/14
  port-license acquire
  no shutdown
interface fc1/15
  port-license acquire
  no shutdown
interface fc1/16
  port-license acquire
  no shutdown
interface fc1/17
  port-license acquire
  no shutdown
interface fc1/18

  port-license acquire
  no shutdown
```

```
interface fc1/19
  port-license acquire
  no shutdown

interface fc1/20
  port-license acquire
  no shutdown

interface fc1/21
  port-license acquire
  no shutdown

interface fc1/22
  port-license acquire
  no shutdown
interface fc1/23
  port-license acquire
  no shutdown
nterface fc1/24
  port-license acquire
  no shutdown
interface fc1/25

  switchport trunk allowed vsan 100
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/26
```

```
  switchport trunk allowed vsan 100
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/27
  switchport trunk allowed vsan 100
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/28
  switchport trunk allowed vsan 100
  switchport trunk mode off
  port-license acquire
  no shutdown

interface fc1/29
  switchport trunk allowed vsan 100
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/30
  switchport trunk allowed vsan 100
  switchport trunk mode off
  port-license acquire
  no shutdown

interface fc1/31
  switchport trunk allowed vsan 100
  switchport trunk mode off
```

```
  port-license acquire
  no shutdown

interface fc1/32
  switchport trunk allowed vsan 100
  switchport trunk mode off
  port-license acquire
  no shutdown

interface fc1/33
  switchport trunk allowed vsan 100
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/34
  switchport trunk allowed vsan 100
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/35
  switchport trunk allowed vsan 100
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/36
  switchport trunk allowed vsan 100
  switchport trunk mode off
  port-license acquire
```

```
  no shutdown
interface fc1/37
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/38
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/39
  port-license acquire
  no shutdown
interface fc1/40
  port-license acquire
  no shutdown
interface fc1/41
  port-license acquire
  no shutdown
interface fc1/42
  port-license acquire
  no shutdown
interface fc1/43
  port-license acquire
 no shutdown
interface fc1/44
  port-license acquire
  no shutdown
interface fc1/45
  port-license acquire
```

```
  no shutdown
interface fc1/46
  port-license acquire
  no shutdown
interface fc1/47
  port-license acquire
  no shutdown

interface fc1/48
  port-license acquire
  no shutdown
ip default-gateway 10.29.164.1
```

Cisco MDS 9148S-B Configuration

```
!Command: show running-config
!Time: Fri Mar  9 23:49:39 2018

version 8.1(1)
power redundancy-mode redundant
feature npiv
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
no password strength-check
```

```
username admin password 5 $1$OPnyy3RN$s8SLqLN3W3JPvf4rEb2CD0  role network-admin
ip domain-lookup
ip host MDS-B  10.29.164.239
aaa group server radius radius
snmp-server user admin network-admin auth md5 0xc9e1af5dbb0bbac72253a1bef037bbbe
 priv 0xc9e1af5dbb0bbac72253a1bef037bbbe localizedkey
snmp-server host 10.155.160.192 traps version 2c public udp-port 1164

rmon event 1 log trap public description FATAL(1) owner PMON@FATAL
rmon event 2 log trap public description CRITICAL(2) owner PMON@CRITICAL
rmon event 3 log trap public description ERROR(3) owner PMON@ERROR
rmon event 4 log trap public description WARNING(4) owner PMON@WARNING
rmon event 5 log trap public description INFORMATION(5) owner PMON@INFO
snmp-server community public group network-operator
vsan database
  vsan 101 name "FlashStack-VCC-CVD-Fabric-B"
fcdroplatency network 2000 vsan 1
device-alias database
  device-alias name X70-CT0-FC1 pwwn 52:4a:93:75:dd:91:0a:01
  device-alias name X70-CT0-FC3 pwwn 52:4a:93:75:dd:91:0a:03
  device-alias name X70-CT1-FC0 pwwn 52:4a:93:75:dd:91:0a:10
  device-alias name X70-CT1-FC2 pwwn 52:4a:93:75:dd:91:0a:12
  device-alias name Infra01-8-hba2 pwwn 20:00:00:25:d5:06:00:4f
  device-alias name Infra02-16-hba2 pwwn 20:00:00:25:d5:06:00:2f
  device-alias name VCC-Infra01-HBA1 pwwn 20:00:00:25:b5:bb:17:1e
  device-alias name VCC-Infra01-HBA3 pwwn 20:00:00:25:b5:bb:17:1f
  device-alias name VCC-Infra02-HBA1 pwwn 20:00:00:25:b5:bb:17:3e
  device-alias name VCC-Infra02-HBA3 pwwn 20:00:00:25:b5:bb:17:3f
  device-alias name VCC-WLHost01-HBA1 pwwn 20:00:00:25:b5:bb:17:00
```

```
device-alias name VCC-WLHost01-HBA3 pwwn 20:00:00:25:b5:bb:17:01
device-alias name VCC-WLHost02-HBA1 pwwn 20:00:00:25:b5:bb:17:02
device-alias name VCC-WLHost02-HBA3 pwwn 20:00:00:25:b5:bb:17:03
device-alias name VCC-WLHost03-HBA1 pwwn 20:00:00:25:b5:bb:17:04
device-alias name VCC-WLHost03-HBA3 pwwn 20:00:00:25:b5:bb:17:05
device-alias name VCC-WLHost04-HBA1 pwwn 20:00:00:25:b5:bb:17:06
device-alias name VCC-WLHost04-HBA3 pwwn 20:00:00:25:b5:bb:17:07
device-alias name VCC-WLHost05-HBA1 pwwn 20:00:00:25:b5:bb:17:08
device-alias name VCC-WLHost05-HBA3 pwwn 20:00:00:25:b5:bb:17:09
device-alias name VCC-WLHost06-HBA1 pwwn 20:00:00:25:b5:bb:17:0a
device-alias name VCC-WLHost06-HBA3 pwwn 20:00:00:25:b5:bb:17:0b
device-alias name VCC-WLHost07-HBA1 pwwn 20:00:00:25:b5:bb:17:0c
device-alias name VCC-WLHost07-HBA3 pwwn 20:00:00:25:b5:bb:17:0d
device-alias name VCC-WLHost08-HBA1 pwwn 20:00:00:25:b5:bb:17:0e
device-alias name VCC-WLHost08-HBA3 pwwn 20:00:00:25:b5:bb:17:0f
device-alias name VCC-WLHost09-HBA1 pwwn 20:00:00:25:b5:bb:17:10
device-alias name VCC-WLHost09-HBA3 pwwn 20:00:00:25:b5:bb:17:11
device-alias name VCC-WLHost10-HBA1 pwwn 20:00:00:25:b5:bb:17:12
device-alias name VCC-WLHost10-HBA3 pwwn 20:00:00:25:b5:bb:17:13
device-alias name VCC-WLHost11-HBA1 pwwn 20:00:00:25:b5:bb:17:14
device-alias name VCC-WLHost11-HBA3 pwwn 20:00:00:25:b5:bb:17:15
device-alias name VCC-WLHost12-HBA1 pwwn 20:00:00:25:b5:bb:17:16
device-alias name VCC-WLHost12-HBA3 pwwn 20:00:00:25:b5:bb:17:17
device-alias name VCC-WLHost13-HBA1 pwwn 20:00:00:25:b5:bb:17:18
device-alias name VCC-WLHost13-HBA3 pwwn 20:00:00:25:b5:bb:17:19
device-alias name VCC-WLHost14-HBA1 pwwn 20:00:00:25:b5:bb:17:1a
device-alias name VCC-WLHost14-HBA3 pwwn 20:00:00:25:b5:bb:17:1b
device-alias name VCC-WLHost15-HBA1 pwwn 20:00:00:25:b5:bb:17:1c
device-alias name VCC-WLHost15-HBA3 pwwn 20:00:00:25:b5:bb:17:1d
```

```
device-alias name VCC-WLHost16-HBA1 pwwn 20:00:00:25:b5:bb:17:20
device-alias name VCC-WLHost16-HBA3 pwwn 20:00:00:25:b5:bb:17:21
device-alias name VCC-WLHost17-HBA1 pwwn 20:00:00:25:b5:bb:17:22
device-alias name VCC-WLHost17-HBA3 pwwn 20:00:00:25:b5:bb:17:23
device-alias name VCC-WLHost18-HBA1 pwwn 20:00:00:25:b5:bb:17:24
device-alias name VCC-WLHost18-HBA3 pwwn 20:00:00:25:b5:bb:17:25
device-alias name VCC-WLHost19-HBA1 pwwn 20:00:00:25:b5:bb:17:26
device-alias name VCC-WLHost19-HBA3 pwwn 20:00:00:25:b5:bb:17:27
device-alias name VCC-WLHost20-HBA1 pwwn 20:00:00:25:b5:bb:17:28
device-alias name VCC-WLHost20-HBA3 pwwn 20:00:00:25:b5:bb:17:29
device-alias name VCC-WLHost21-HBA1 pwwn 20:00:00:25:b5:bb:17:2a
device-alias name VCC-WLHost21-HBA3 pwwn 20:00:00:25:b5:bb:17:2b
device-alias name VCC-WLHost22-HBA1 pwwn 20:00:00:25:b5:bb:17:2c
device-alias name VCC-WLHost22-HBA3 pwwn 20:00:00:25:b5:bb:17:2d
device-alias name VCC-WLHost23-HBA1 pwwn 20:00:00:25:b5:bb:17:2e
device-alias name VCC-WLHost23-HBA3 pwwn 20:00:00:25:b5:bb:17:2f
device-alias name VCC-WLHost24-HBA1 pwwn 20:00:00:25:b5:bb:17:30
device-alias name VCC-WLHost24-HBA3 pwwn 20:00:00:25:b5:bb:17:31
device-alias name VCC-WLHost25-HBA1 pwwn 20:00:00:25:b5:bb:17:32
device-alias name VCC-WLHost25-HBA3 pwwn 20:00:00:25:b5:bb:17:33
device-alias name VCC-WLHost26-HBA1 pwwn 20:00:00:25:b5:bb:17:34
device-alias name VCC-WLHost26-HBA3 pwwn 20:00:00:25:b5:bb:17:35
device-alias name VCC-WLHost27-HBA1 pwwn 20:00:00:25:b5:bb:17:36
device-alias name VCC-WLHost27-HBA3 pwwn 20:00:00:25:b5:bb:17:37
device-alias name VCC-WLHost28-HBA1 pwwn 20:00:00:25:b5:bb:17:38
device-alias name VCC-WLHost28-HBA3 pwwn 20:00:00:25:b5:bb:17:39
device-alias name VCC-WLHost29-HBA1 pwwn 20:00:00:25:b5:bb:17:3a
device-alias name VCC-WLHost29-HBA3 pwwn 20:00:00:25:b5:bb:17:3b
device-alias name VCC-WLHost30-HBA1 pwwn 20:00:00:25:b5:bb:17:3c
```

```
  device-alias name VCC-WLHost30-HBA3 pwwn 20:00:00:25:b5:bb:17:3d
  device-alias commit
fcdomain fcid database
    vsan 101 wwn 52:4a:93:75:dd:91:0a:02 fcid 0x2e0000 dynamic
  vsan 101 wwn 52:4a:93:75:dd:91:0a:03 fcid 0x2e0100 dynamic

!           [X70-CT0-FC3]
  vsan 101 wwn 52:4a:93:75:dd:91:0a:12 fcid 0x2e0200 dynamic
!           [X70-CT1-FC2]
  vsan 101 wwn 52:4a:93:75:dd:91:0a:13 fcid 0x2e0300 dynamic
  vsan 101 wwn 20:04:00:de:fb:90:a4:40 fcid 0x2e0400 dynamic
  vsan 101 wwn 20:02:00:de:fb:90:a4:40 fcid 0x2e0500 dynamic
  vsan 101 wwn 20:03:00:de:fb:90:a4:40 fcid 0x2e0600 dynamic
  vsan 101 wwn 20:01:00:de:fb:90:a4:40 fcid 0x2e0700 dynamic
  vsan 101 wwn 20:00:00:25:b5:bb:17:1e fcid 0x2e060e dynamic
!           [VCC-Infra01-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:02 fcid 0x2e0405 dynamic
!           [VCC-WLHost02-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:0a fcid 0x2e050f dynamic
!           [VCC-WLHost06-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:0e fcid 0x2e0409 dynamic
!           [VCC-WLHost08-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:26 fcid 0x2e0607 dynamic
!           [VCC-WLHost19-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:2e fcid 0x2e050a dynamic
!           [VCC-WLHost23-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:22 fcid 0x2e0705 dynamic
!           [VCC-WLHost17-HBA1]
```

```
  vsan 101 wwn 20:00:00:25:b5:bb:17:28 fcid 0x2e0406 dynamic

!           [VCC-WLHost20-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:24 fcid 0x2e070a dynamic
!           [VCC-WLHost18-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:06 fcid 0x2e060a dynamic
!           [VCC-WLHost04-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:0c fcid 0x2e0502 dynamic
!           [VCC-WLHost07-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:08 fcid 0x2e070c dynamic
!           [VCC-WLHost05-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:00 fcid 0x2e040f dynamic
!           [VCC-WLHost01-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:04 fcid 0x2e060b dynamic
!           [VCC-WLHost03-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:16 fcid 0x2e0612 dynamic
!           [VCC-WLHost12-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:10 fcid 0x2e0602 dynamic
!           [VCC-WLHost09-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:12 fcid 0x2e0404 dynamic
!           [VCC-WLHost10-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:18 fcid 0x2e0604 dynamic
!           [VCC-WLHost13-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:20 fcid 0x2e0709 dynamic
!           [VCC-WLHost16-HBA1]

  vsan 101 wwn 20:00:00:25:b5:bb:17:2c fcid 0x2e0601 dynamic
!           [VCC-WLHost22-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:2a fcid 0x2e0411 dynamic
```

```
!            [VCC-WLHost21-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:1a fcid 0x2e0703 dynamic
!            [VCC-WLHost14-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:1c fcid 0x2e040b dynamic
!            [VCC-WLHost15-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:14 fcid 0x2e0711 dynamic
!            [VCC-WLHost11-HBA1]
  vsan 101 wwn 52:4a:93:75:dd:91:0a:07 fcid 0x2e0800 dynamic
  vsan 101 wwn 52:4a:93:75:dd:91:0a:06 fcid 0x2e0900 dynamic
  vsan 101 wwn 52:4a:93:75:dd:91:0a:16 fcid 0x2e0a00 dynamic
  vsan 101 wwn 52:4a:93:75:dd:91:0a:17 fcid 0x2e0b00 dynamic
  vsan 101 wwn 20:00:00:25:b5:bb:17:3e fcid 0x2e0609 dynamic
!            [VCC-Infra02-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:3f fcid 0x2e040e dynamic
!            [VCC-Infra02-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:1f fcid 0x2e050b dynamic
!            [VCC-Infra01-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:03 fcid 0x2e0407 dynamic

!            [VCC-WLHost02-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:01 fcid 0x2e0704 dynamic
!            [VCC-WLHost01-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:05 fcid 0x2e0509 dynamic
!            [VCC-WLHost03-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:07 fcid 0x2e0507 dynamic
!            [VCC-WLHost04-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:0b fcid 0x2e040a dynamic
!            [VCC-WLHost06-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:09 fcid 0x2e050d dynamic
```

```
!           [VCC-WLHost05-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:0d fcid 0x2e0701 dynamic
!           [VCC-WLHost07-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:0f fcid 0x2e0608 dynamic
!           [VCC-WLHost08-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:21 fcid 0x2e0403 dynamic
!           [VCC-WLHost16-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:23 fcid 0x2e0506 dynamic
!           [VCC-WLHost17-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:25 fcid 0x2e0408 dynamic
!           [VCC-WLHost18-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:27 fcid 0x2e0508 dynamic
!           [VCC-WLHost19-HBA3]

  vsan 101 wwn 20:00:00:25:b5:bb:17:29 fcid 0x2e070f dynamic
!           [VCC-WLHost20-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:2b fcid 0x2e0707 dynamic
!           [VCC-WLHost21-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:2d fcid 0x2e0513 dynamic
!           [VCC-WLHost22-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:2f fcid 0x2e050c dynamic
!           [VCC-WLHost23-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:11 fcid 0x2e0510 dynamic
!           [VCC-WLHost09-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:13 fcid 0x2e060d dynamic
!           [VCC-WLHost10-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:15 fcid 0x2e0401 dynamic
!           [VCC-WLHost11-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:17 fcid 0x2e0712 dynamic
```

```
!          [VCC-WLHost12-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:19 fcid 0x2e0504 dynamic

!          [VCC-WLHost13-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:1b fcid 0x2e0611 dynamic
!          [VCC-WLHost14-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:1d fcid 0x2e0706 dynamic
!          [VCC-WLHost15-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:34 fcid 0x2e0505 dynamic
!          [VCC-WLHost26-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:32 fcid 0x2e0402 dynamic
!          [VCC-WLHost25-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:33 fcid 0x2e0501 dynamic
!          [VCC-WLHost25-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:35 fcid 0x2e0708 dynamic
!          [VCC-WLHost26-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:38 fcid 0x2e0412 dynamic
!          [VCC-WLHost28-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:39 fcid 0x2e0503 dynamic
!          [VCC-WLHost28-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:30 fcid 0x2e0410 dynamic
!          [VCC-WLHost24-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:3a fcid 0x2e0605 dynamic
!          [VCC-WLHost29-HBA1]
  vsan 101 wwn 20:00:00:25:b5:bb:17:36 fcid 0x2e070e dynamic
!          [VCC-WLHost27-HBA1]

  vsan 101 wwn 20:00:00:25:b5:bb:17:3c fcid 0x2e0603 dynamic
!          [VCC-WLHost30-HBA1]
```

```
  vsan 101 wwn 20:00:00:25:b5:bb:17:3d fcid 0x2e0512 dynamic
!            [VCC-WLHost30-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:3b fcid 0x2e0702 dynamic
!            [VCC-WLHost29-HBA3]
  vsan 101 wwn 20:00:00:25:b5:bb:17:37 fcid 0x2e0610 dynamic
!            [VCC-WLHost27-HBA3]
 vsan 101 wwn 52:4a:93:75:dd:91:0a:01 fcid 0x2e0c00 dynamic
!            [X70-CT0-FC1]

  vsan 101 wwn 52:4a:93:75:dd:91:0a:11 fcid 0x2e0d00 dynamic
  vsan 101 wwn 52:4a:93:75:dd:91:0a:10 fcid 0x2e0e00 dynamic
!            [X70-CT1-FC0]

!Active Zone Database Section for vsan 101
zone name FlaskStack-VCC-CVD-WLHost01 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!             [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!             [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!             [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!             [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:00
!             [VCC-WLHost01-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:01
!             [VCC-WLHost01-HBA3]

zone name FlaskStack-VCC-CVD-WLHost02 vsan 101
```

```
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]

    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:02
!           [VCC-WLHost02-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:03
!           [VCC-WLHost02-HBA3]

zone name FlaskStack-VCC-CVD-WLHost03 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:04
!           [VCC-WLHost03-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:05
!           [VCC-WLHost03-HBA3]

zone name FlaskStack-VCC-CVD-WLHost04 vsan 101
```

```
    member pwwn 52:4a:93:75:dd:91:0a:01

!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:06
!           [VCC-WLHost04-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:07
!           [VCC-WLHost04-HBA3]

zone name FlaskStack-VCC-CVD-WLHost05 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:08
!           [VCC-WLHost05-HBA1]

    member pwwn 20:00:00:25:b5:bb:17:09
!           [VCC-WLHost05-HBA3]
```

```
zone name FlaskStack-VCC-CVD-WLHost06 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:0a
!           [VCC-WLHost06-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:0b
!           [VCC-WLHost06-HBA3]

zone name FlaskStack-VCC-CVD-WLHost07 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10

!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:0c
!           [VCC-WLHost07-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:0d
!           [VCC-WLHost07-HBA3]
```

```
zone name FlaskStack-VCC-CVD-WLHost08 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:0e
!           [VCC-WLHost08-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:0f
!           [VCC-WLHost08-HBA3]

zone name FlaskStack-VCC-CVD-WLHost09 vsan 101

    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:10
!           [VCC-WLHost09-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:11
!           [VCC-WLHost09-HBA3]
```

```
zone name FlaskStack-VCC-CVD-WLHost10 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:12

!           [VCC-WLHost10-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:13
!           [VCC-WLHost10-HBA3]

zone name FlaskStack-VCC-CVD-WLHost11 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:14
!           [VCC-WLHost11-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:15
!           [VCC-WLHost11-HBA3]
```

```
zone name FlaskStack-VCC-CVD-WLHost12 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]

    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:16
!           [VCC-WLHost12-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:17
!           [VCC-WLHost12-HBA3]

zone name FlaskStack-VCC-CVD-WLHost13 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:18
!           [VCC-WLHost13-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:19
!           [VCC-WLHost13-HBA3]
```

```
zone name FlaskStack-VCC-CVD-WLHost14 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!          [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!          [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!          [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!          [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:1a
!          [VCC-WLHost14-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:1b
!          [VCC-WLHost14-HBA3]

zone name FlaskStack-VCC-CVD-WLHost15 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!          [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!          [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!          [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!          [X70-CT1-FC2]

   member pwwn 20:00:00:25:b5:bb:17:1c
!          [VCC-WLHost15-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:1d
!          [VCC-WLHost15-HBA3]
```

```
zone name FlaskStack-VCC-CVD-Infra01 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!          [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!          [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!          [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!          [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:1e
!          [VCC-Infra01-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:1f
!          [VCC-Infra01-HBA3]

zone name FlaskStack-VCC-CVD-WLHost16 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!          [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03

!          [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!          [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!          [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:20
!          [VCC-WLHost16-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:21
!          [VCC-WLHost16-HBA3]
```

```
zone name FlaskStack-VCC-CVD-WLHost17 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:22
!           [VCC-WLHost17-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:23
!           [VCC-WLHost17-HBA3]


zone name FlaskStack-VCC-CVD-WLHost18 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:24
!           [VCC-WLHost18-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:25
```

```
!           [VCC-WLHost18-HBA3]

zone name FlaskStack-VCC-CVD-WLHost19 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12

!           [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:26
!           [VCC-WLHost19-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:27
!           [VCC-WLHost19-HBA3]

zone name FlaskStack-VCC-CVD-WLHost20 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:28
!           [VCC-WLHost20-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:29
```

```
!            [VCC-WLHost20-HBA3]

zone name FlaskStack-VCC-CVD-WLHost21 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!            [X70-CT0-FC1]

   member pwwn 52:4a:93:75:dd:91:0a:03
!            [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!            [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!            [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:2a
!            [VCC-WLHost21-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:2b
!            [VCC-WLHost21-HBA3]

zone name FlaskStack-VCC-CVD-WLHost22 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!            [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!            [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!            [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!            [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:2c
!            [VCC-WLHost22-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:2d
```

```
!            [VCC-WLHost22-HBA3]

zone name FlaskStack-VCC-CVD-WLHost23 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!            [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!            [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!            [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!            [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:2e
!            [VCC-WLHost23-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:2f
!            [VCC-WLHost23-HBA3]

zone name FlaskStack-VCC-CVD-WLHost24 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!            [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!            [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!            [X70-CT1-FC0]

   member pwwn 52:4a:93:75:dd:91:0a:12
!            [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:30
!            [VCC-WLHost24-HBA1]
```

```
    member pwwn 20:00:00:25:b5:bb:17:31
!           [VCC-WLHost24-HBA3]

zone name FlaskStack-VCC-CVD-WLHost25 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:32
!           [VCC-WLHost25-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:33
!           [VCC-WLHost25-HBA3]

zone name FlaskStack-VCC-CVD-WLHost26 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01

!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:34
!           [VCC-WLHost26-HBA1]
```

```
    member pwwn 20:00:00:25:b5:bb:17:35
!           [VCC-WLHost26-HBA3]

zone name FlaskStack-VCC-CVD-WLHost27 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:36
!           [VCC-WLHost27-HBA1]

    member pwwn 20:00:00:25:b5:bb:17:37
!           [VCC-WLHost27-HBA3]

zone name FlaskStack-VCC-CVD-WLHost28 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:38
!           [VCC-WLHost28-HBA1]
```

```
    member pwwn 20:00:00:25:b5:bb:17:39
!           [VCC-WLHost28-HBA3]

zone name FlaskStack-VCC-CVD-WLHost29 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10

!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:3a
!           [VCC-WLHost29-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:3b
!           [VCC-WLHost29-HBA3]

zone name FlaskStack-VCC-CVD-WLHost30 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:3c
!           [VCC-WLHost30-HBA1]
```

```
    member pwwn 20:00:00:25:b5:bb:17:3d
!           [VCC-WLHost30-HBA3]

zone name FlaskStack-VCC-CVD-Infra02 vsan 101

    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:3e
!           [VCC-Infra02-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:3f
!           [VCC-Infra02-HBA3]

zoneset name FlashStack-VCC-CVD vsan 101
    member FlaskStack-VCC-CVD-WLHost01
    member FlaskStack-VCC-CVD-WLHost02
    member FlaskStack-VCC-CVD-WLHost03
    member FlaskStack-VCC-CVD-WLHost04
    member FlaskStack-VCC-CVD-WLHost05
    member FlaskStack-VCC-CVD-WLHost06
    member FlaskStack-VCC-CVD-WLHost07
    member FlaskStack-VCC-CVD-WLHost08
    member FlaskStack-VCC-CVD-WLHost09
```

```
member FlaskStack-VCC-CVD-WLHost10
member FlaskStack-VCC-CVD-WLHost11
member FlaskStack-VCC-CVD-WLHost12
member FlaskStack-VCC-CVD-WLHost13
member FlaskStack-VCC-CVD-WLHost14
member FlaskStack-VCC-CVD-WLHost15
member FlaskStack-VCC-CVD-Infra01
member FlaskStack-VCC-CVD-WLHost16
member FlaskStack-VCC-CVD-WLHost17
member FlaskStack-VCC-CVD-WLHost18
member FlaskStack-VCC-CVD-WLHost19
member FlaskStack-VCC-CVD-WLHost20
member FlaskStack-VCC-CVD-WLHost21
member FlaskStack-VCC-CVD-WLHost22
member FlaskStack-VCC-CVD-WLHost23
member FlaskStack-VCC-CVD-WLHost24
member FlaskStack-VCC-CVD-WLHost25
member FlaskStack-VCC-CVD-WLHost26
member FlaskStack-VCC-CVD-WLHost27
member FlaskStack-VCC-CVD-WLHost28
member FlaskStack-VCC-CVD-WLHost29
member FlaskStack-VCC-CVD-WLHost30
member FlaskStack-VCC-CVD-Infra02

zoneset activate name FlashStack-VCC-CVD vsan 101
do clear zone database vsan 101
!Full Zone Database Section for vsan 101
```

```
zone name FlaskStack-VCC-CVD-WLHost01 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:00
!           [VCC-WLHost01-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:01
!           [VCC-WLHost01-HBA3]

zone name FlaskStack-VCC-CVD-WLHost02 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]

    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:02
!           [VCC-WLHost02-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:03
!           [VCC-WLHost02-HBA3]
```

```
zone name FlaskStack-VCC-CVD-WLHost03 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:04
!           [VCC-WLHost03-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:05
!           [VCC-WLHost03-HBA3]


zone name FlaskStack-VCC-CVD-WLHost04 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:06
!           [VCC-WLHost04-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:07
!           [VCC-WLHost04-HBA3]
```

```
zone name FlaskStack-VCC-CVD-WLHost05 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]

    member pwwn 20:00:00:25:b5:bb:17:08
!           [VCC-WLHost05-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:09
!           [VCC-WLHost05-HBA3]

zone name FlaskStack-VCC-CVD-WLHost06 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:0a
!           [VCC-WLHost06-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:0b
!           [VCC-WLHost06-HBA3]
```

```
zone name FlaskStack-VCC-CVD-WLHost07 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03

!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:0c
!           [VCC-WLHost07-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:0d
!           [VCC-WLHost07-HBA3]

zone name FlaskStack-VCC-CVD-WLHost08 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:0e
!           [VCC-WLHost08-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:0f
!           [VCC-WLHost08-HBA3]
```

```
zone name FlaskStack-VCC-CVD-WLHost09 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:10
!           [VCC-WLHost09-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:11
!           [VCC-WLHost09-HBA3]

zone name FlaskStack-VCC-CVD-WLHost10 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12

!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:12
!           [VCC-WLHost10-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:13
```

```
!           [VCC-WLHost10-HBA3]

zone name FlaskStack-VCC-CVD-WLHost11 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:14
!           [VCC-WLHost11-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:15
!           [VCC-WLHost11-HBA3]

zone name FlaskStack-VCC-CVD-WLHost12 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]

   member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:16
!           [VCC-WLHost12-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:17
```

```
!           [VCC-WLHost12-HBA3]

zone name FlaskStack-VCC-CVD-WLHost13 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:18
!           [VCC-WLHost13-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:19

!           [VCC-WLHost13-HBA3]

zone name FlaskStack-VCC-CVD-WLHost14 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:1a
!           [VCC-WLHost14-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:1b
```

```
!           [VCC-WLHost14-HBA3]

zone name FlaskStack-VCC-CVD-WLHost15 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]

   member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:1c
!           [VCC-WLHost15-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:1d
!           [VCC-WLHost15-HBA3]

zone name FlaskStack-VCC-CVD-Infra01 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:1e
!           [VCC-Infra01-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:1f
```

```
!            [VCC-Infra01-HBA3]

zone name FlaskStack-VCC-CVD-WLHost16 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01

!            [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!            [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!            [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!            [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:20
!            [VCC-WLHost16-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:21
!            [VCC-WLHost16-HBA3]

zone name FlaskStack-VCC-CVD-WLHost17 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!            [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!            [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!            [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!            [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:22
!            [VCC-WLHost17-HBA1]
```

```
    member pwwn 20:00:00:25:b5:bb:17:23
!           [VCC-WLHost17-HBA3]

zone name FlaskStack-VCC-CVD-WLHost18 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:24
!           [VCC-WLHost18-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:25
!           [VCC-WLHost18-HBA3]

zone name FlaskStack-VCC-CVD-WLHost19 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10

!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:26
```

```
!            [VCC-WLHost19-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:27
!            [VCC-WLHost19-HBA3]

zone name FlaskStack-VCC-CVD-WLHost20 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!            [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!            [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!            [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!            [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:28
!            [VCC-WLHost20-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:29
!            [VCC-WLHost20-HBA3]

zone name FlaskStack-VCC-CVD-WLHost21 vsan 101

   member pwwn 52:4a:93:75:dd:91:0a:01
!            [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!            [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!            [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!            [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:2a
```

```
!            [VCC-WLHost21-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:2b
!            [VCC-WLHost21-HBA3]

zone name FlaskStack-VCC-CVD-WLHost22 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!            [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!            [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!            [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!            [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:2c

!            [VCC-WLHost22-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:2d
!            [VCC-WLHost22-HBA3]

zone name FlaskStack-VCC-CVD-WLHost23 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!            [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!            [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!            [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!            [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:2e
```

```
!            [VCC-WLHost23-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:2f
!            [VCC-WLHost23-HBA3]

zone name FlaskStack-VCC-CVD-WLHost24 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!            [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!            [X70-CT0-FC3]

   member pwwn 52:4a:93:75:dd:91:0a:10
!            [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!            [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:30
!            [VCC-WLHost24-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:31
!            [VCC-WLHost24-HBA3]

zone name FlaskStack-VCC-CVD-WLHost25 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!            [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!            [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!            [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!            [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:32
```

```
!            [VCC-WLHost25-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:33
!            [VCC-WLHost25-HBA3]


zone name FlaskStack-VCC-CVD-WLHost26 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!            [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!            [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!            [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!            [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:34
!            [VCC-WLHost26-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:35
!            [VCC-WLHost26-HBA3]

zone name FlaskStack-VCC-CVD-WLHost27 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!            [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!            [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!            [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!            [X70-CT1-FC2]
```

```
    member pwwn 20:00:00:25:b5:bb:17:36
!           [VCC-WLHost27-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:37
!           [VCC-WLHost27-HBA3]

zone name FlaskStack-VCC-CVD-WLHost28 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03
!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:38
!           [VCC-WLHost28-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:39
!           [VCC-WLHost28-HBA3]

zone name FlaskStack-VCC-CVD-WLHost29 vsan 101
    member pwwn 52:4a:93:75:dd:91:0a:01
!           [X70-CT0-FC1]
    member pwwn 52:4a:93:75:dd:91:0a:03

!           [X70-CT0-FC3]
    member pwwn 52:4a:93:75:dd:91:0a:10
!           [X70-CT1-FC0]
    member pwwn 52:4a:93:75:dd:91:0a:12
```

```
!            [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:3a
!            [VCC-WLHost29-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:3b
!            [VCC-WLHost29-HBA3]

zone name FlaskStack-VCC-CVD-WLHost30 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!            [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!            [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!            [X70-CT1-FC0]
   member pwwn 52:4a:93:75:dd:91:0a:12
!            [X70-CT1-FC2]
   member pwwn 20:00:00:25:b5:bb:17:3c
!            [VCC-WLHost30-HBA1]
   member pwwn 20:00:00:25:b5:bb:17:3d
!            [VCC-WLHost30-HBA3]


zone name FlaskStack-VCC-CVD-Infra02 vsan 101
   member pwwn 52:4a:93:75:dd:91:0a:01
!            [X70-CT0-FC1]
   member pwwn 52:4a:93:75:dd:91:0a:03
!            [X70-CT0-FC3]
   member pwwn 52:4a:93:75:dd:91:0a:10
!            [X70-CT1-FC0]
```

```
    member pwwn 52:4a:93:75:dd:91:0a:12
!           [X70-CT1-FC2]
    member pwwn 20:00:00:25:b5:bb:17:3e
!           [VCC-Infra02-HBA1]
    member pwwn 20:00:00:25:b5:bb:17:3f
!           [VCC-Infra02-HBA3]

zoneset name FlashStack-VCC-CVD vsan 101
    member FlaskStack-VCC-CVD-WLHost01
    member FlaskStack-VCC-CVD-WLHost02
    member FlaskStack-VCC-CVD-WLHost03
    member FlaskStack-VCC-CVD-WLHost04
    member FlaskStack-VCC-CVD-WLHost05
    member FlaskStack-VCC-CVD-WLHost06
    member FlaskStack-VCC-CVD-WLHost07

    member FlaskStack-VCC-CVD-WLHost08
    member FlaskStack-VCC-CVD-WLHost09
    member FlaskStack-VCC-CVD-WLHost10
    member FlaskStack-VCC-CVD-WLHost11
    member FlaskStack-VCC-CVD-WLHost12
    member FlaskStack-VCC-CVD-WLHost13
    member FlaskStack-VCC-CVD-WLHost14
    member FlaskStack-VCC-CVD-WLHost15
    member FlaskStack-VCC-CVD-Infra01
    member FlaskStack-VCC-CVD-WLHost16
    member FlaskStack-VCC-CVD-WLHost17
    member FlaskStack-VCC-CVD-WLHost18
    member FlaskStack-VCC-CVD-WLHost19
```

```
    member FlaskStack-VCC-CVD-WLHost20
    member FlaskStack-VCC-CVD-WLHost21
    member FlaskStack-VCC-CVD-WLHost22
    member FlaskStack-VCC-CVD-WLHost23
    member FlaskStack-VCC-CVD-WLHost24
    member FlaskStack-VCC-CVD-WLHost25
    member FlaskStack-VCC-CVD-WLHost26
    member FlaskStack-VCC-CVD-WLHost27
    member FlaskStack-VCC-CVD-WLHost28
    member FlaskStack-VCC-CVD-WLHost29

    member FlaskStack-VCC-CVD-WLHost30
    member FlaskStack-VCC-CVD-Infra02

interface mgmt0
  ip address 10.29.164.239 255.255.255.0
vsan database
  vsan 101 interface fc1/25
  vsan 101 interface fc1/26
  vsan 101 interface fc1/27
  vsan 101 interface fc1/28
  vsan 101 interface fc1/29
  vsan 101 interface fc1/30
  vsan 101 interface fc1/31
  vsan 101 interface fc1/32
  vsan 101 interface fc1/33
  vsan 101 interface fc1/34
  vsan 101 interface fc1/35
  vsan 101 interface fc1/36
```

```
switchname MDS-B
no terminal log-all
line console
  terminal width  80
line vty

boot kickstart bootflash:/m9100-s5ek9-kickstart-mz.8.1.1.bin
boot system bootflash:/m9100-s5ek9-mz.8.1.1.bin
interface fc1/1
interface fc1/2
interface fc1/11
interface fc1/12
interface fc1/19
interface fc1/20
interface fc1/21
interface fc1/22
interface fc1/23
interface fc1/24
interface fc1/43
interface fc1/44
interface fc1/45
interface fc1/46
interface fc1/3
interface fc1/4
interface fc1/5
interface fc1/6
interface fc1/7
interface fc1/8
interface fc1/9
```

```
interface fc1/10
interface fc1/13
interface fc1/14
interface fc1/15
interface fc1/16
interface fc1/17
interface fc1/18
interface fc1/25
interface fc1/26
interface fc1/27
interface fc1/28
interface fc1/29
interface fc1/30
interface fc1/31
interface fc1/32
interface fc1/33
interface fc1/34
interface fc1/35
interface fc1/36
interface fc1/37
interface fc1/38
interface fc1/39
interface fc1/40

interface fc1/41
interface fc1/42
interface fc1/47
interface fc1/48
```

```
interface fc1/1
interface fc1/2
interface fc1/11
interface fc1/12
interface fc1/19
interface fc1/20
interface fc1/21
interface fc1/22
interface fc1/23
interface fc1/24
interface fc1/43
interface fc1/44
interface fc1/45
interface fc1/46
interface fc1/1
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/2
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/3
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/4
  switchport trunk mode off
  port-license acquire
```

```
  no shutdown
interface fc1/5
  port-license acquire
  no shutdown
interface fc1/6
  port-license acquire
  no shutdown
interface fc1/7
  port-license acquire
  no shutdown
interface fc1/8
  port-license acquire
  no shutdown
interface fc1/9
  port-license acquire
  no shutdown
interface fc1/10
  port-license acquire
  no shutdown
interface fc1/11
  port-license acquire
interface fc1/12
  port-license acquire
interface fc1/13
  port-license acquire
  no shutdown
interface fc1/14
  port-license acquire
  no shutdown
```

```
interface fc1/15
  port-license acquire
  no shutdown
interface fc1/16
  port-license acquire
  no shutdown
interface fc1/17
  port-license acquire
  no shutdown
interface fc1/18
  port-license acquire
  no shutdown
interface fc1/19
  port-license acquire
  no shutdown
interface fc1/20
  port-license acquire
  no shutdown
interface fc1/21
  port-license acquire
  no shutdown
interface fc1/22
  port-license acquire
  no shutdown
interface fc1/23
  port-license acquire
  no shutdown

interface fc1/24
```

```
  port-license acquire
  no shutdown
interface fc1/25
  switchport trunk allowed vsan 101
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/26
  switchport trunk allowed vsan 101
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/27
  switchport trunk allowed vsan 101
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/28
  switchport trunk allowed vsan 101
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/29
  switchport trunk allowed vsan 101
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/30
  switchport trunk allowed vsan 101
```

```
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/31
  switchport trunk allowed vsan 101
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/32
  switchport trunk allowed vsan 101
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/33
  switchport trunk allowed vsan 101
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/34
  switchport trunk allowed vsan 101
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/35
  switchport trunk allowed vsan 101
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/36
```

```
  switchport trunk allowed vsan 101
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/37
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/38
  switchport trunk mode off
  port-license acquire
  no shutdown
interface fc1/39
  port-license acquire
  no shutdown
interface fc1/40
  port-license acquire
  no shutdown
interface fc1/41
  port-license acquire
  no shutdown
interface fc1/42
  port-license acquire
  no shutdown
interface fc1/43
  port-license acquire
  no shutdown
interface fc1/44
  port-license acquire
```

```
  no shutdown
interface fc1/45
  port-license acquire
  no shutdown
interface fc1/46
  port-license acquire
  no shutdown
interface fc1/47
  port-license acquire
  no shutdown
interface fc1/48
  port-license acquire
  no shutdown
ip default-gateway 10.29.164.1
```

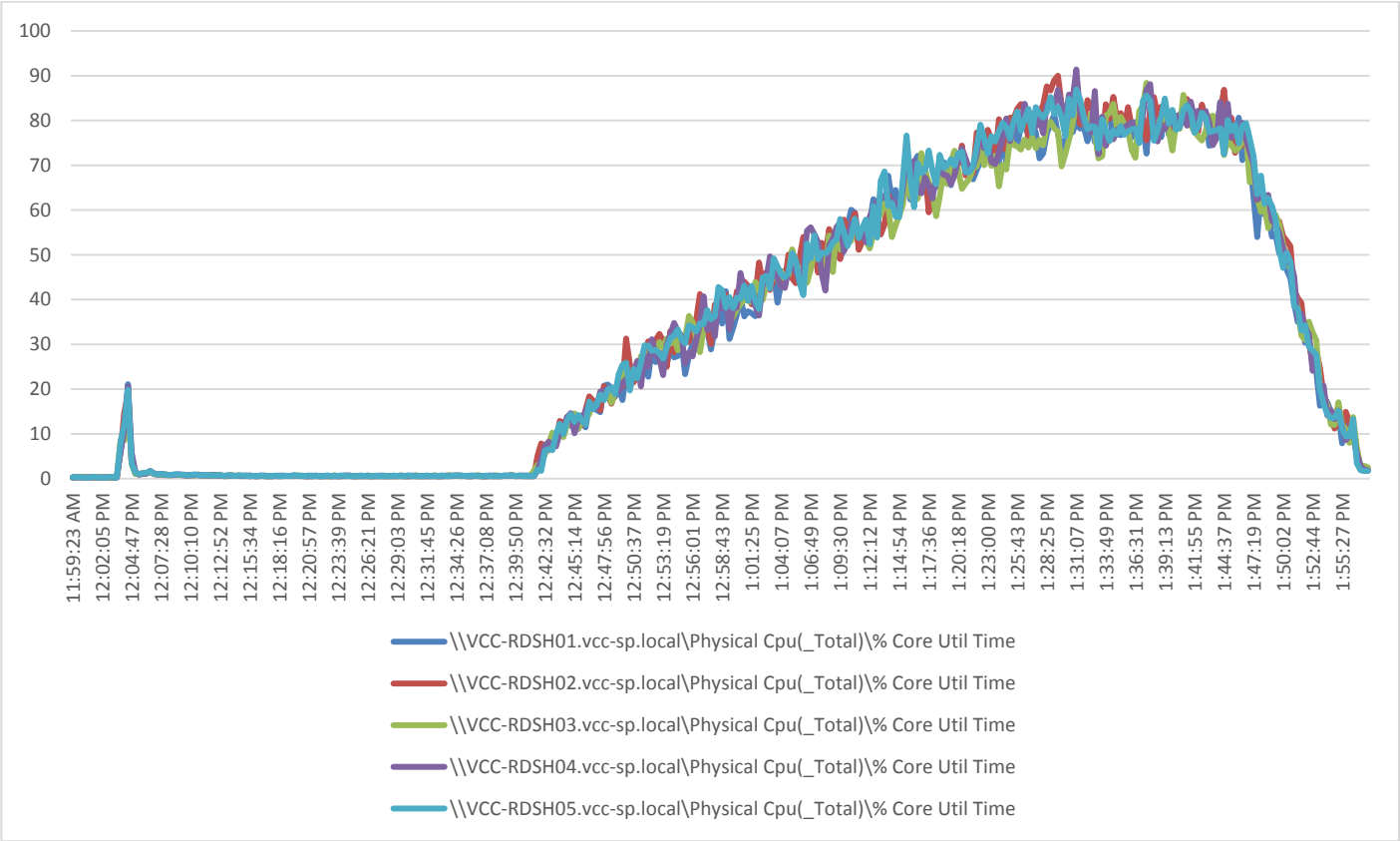
Full Scale 6000 Mixed-User Performance Chart with Boot and LoginVSI Knowledge Worker Worklaod Test

This section provides a detailed performance chart for ESXi 6.5 U1 installed on Cisco UCS B200 M5 Blade Server part of the workload test with VMware Horizon 7 deployed pooled RDSH and VDI (persistent/non-persistnent) Desktop VMs on Pure Storage //X70 system running LoginVSI v4.1.32 based knowledge worker workload part of the FlashStack Datacenter reference architecute defined here.

The charts below are defined in the set of five host in the single performance chart.

RDSH Server Performance Monitor Data for One Sample RDSH Server: 6000 Users Mixed Scale Testing

Figure 126 Full Scale | 6000 Mixed Users | RDSH Host | Host CPU Utilization


100
90
80
70
60
50
40
30
20
10
0
11:59:23 AM
12:02:05 PM
12:04:47 PM
12:07:28 PM
12:10:10 PM
12:12:52 PM
12:15:34 PM
12:18:16 PM
12:20:57 PM
12:23:39 PM
12:26:21 PM
12:29:03 PM
12:31:45 PM
12:34:26 PM
12:37:08 PM
12:39:50 PM
12:42:32 PM
12:45:14 PM
12:47:56 PM
12:50:37 PM
12:53:19 PM
12:56:01 PM
12:58:43 PM
1:01:25 PM
1:04:07 PM
1:06:49 PM
1:09:30 PM
1:12:12 PM
1:14:54 PM
1:17:36 PM
1:20:18 PM
1:23:00 PM
1:25:43 PM
1:28:25 PM
1:31:07 PM
1:33:49 PM
1:36:31 PM
1:39:13 PM
1:41:55 PM
1:44:37 PM
1:47:19 PM
1:50:02 PM
1:52:44 PM
1:55:27 PM
\\VCC-RDSH01.vcc-sp.local\Physical Cpu(_Total)\% Core Util Time
\\VCC-RDSH02.vcc-sp.local\Physical Cpu(_Total)\% Core Util Time
\\VCC-RDSH03.vcc-sp.local\Physical Cpu(_Total)\% Core Util Time
\\VCC-RDSH04.vcc-sp.local\Physical Cpu(_Total)\% Core Util Time
\\VCC-RDSH05.vcc-sp.local\Physical Cpu(_Total)\% Core Util Time

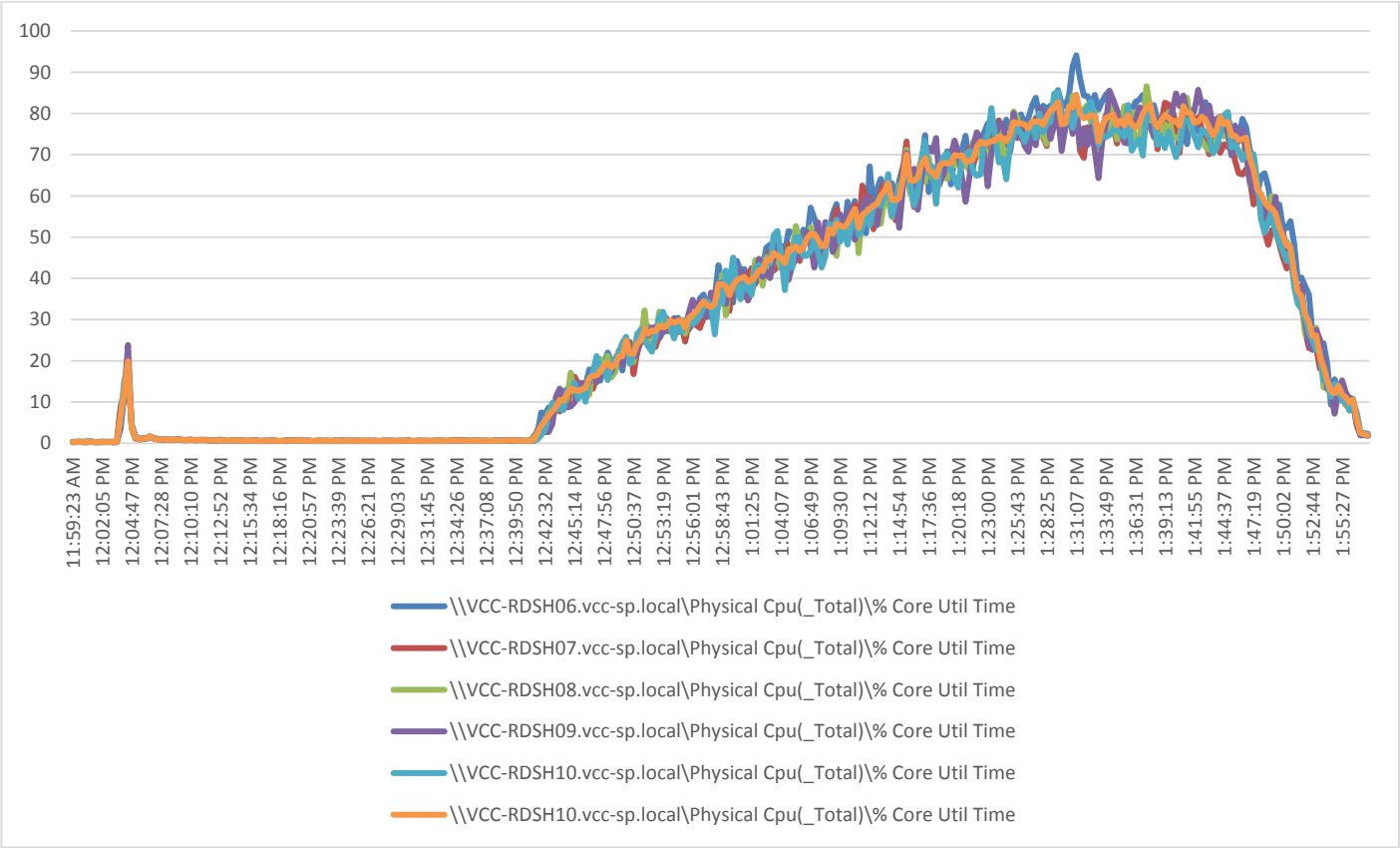
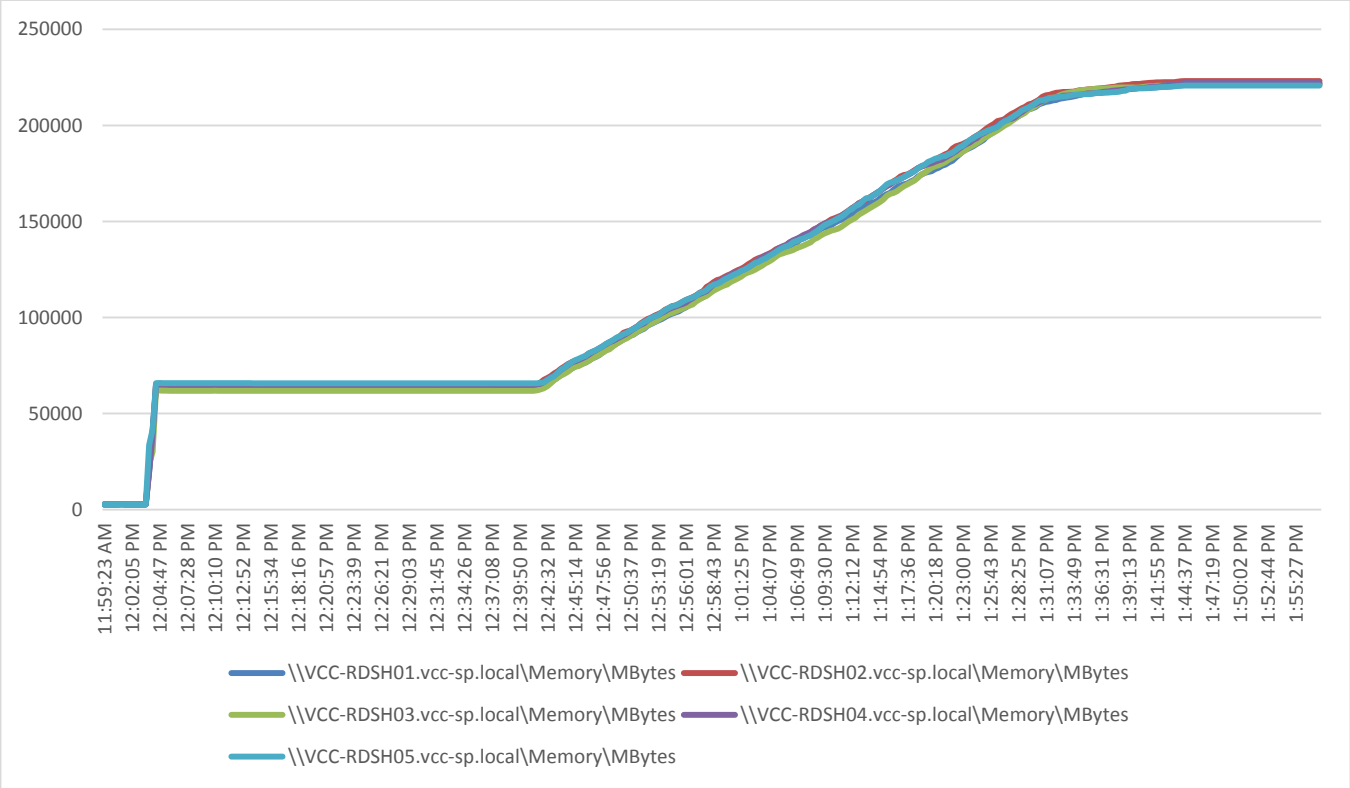

100
90
80
70
60
50
40
30
20
10
0
11:59:23 AM
12:02:05 PM
12:04:47 PM
12:07:28 PM
12:10:10 PM
12:12:52 PM
12:15:34 PM
12:18:16 PM
12:20:57 PM
12:23:39 PM
12:26:21 PM
12:29:03 PM
12:31:45 PM
12:34:26 PM
12:37:08 PM
12:39:50 PM
12:42:32 PM
12:45:14 PM
12:47:56 PM
12:50:37 PM
12:53:19 PM
12:56:01 PM
12:58:43 PM
1:01:25 PM
1:04:07 PM
1:06:49 PM
1:09:30 PM
1:12:12 PM
1:14:54 PM
1:17:36 PM
1:20:18 PM
1:23:00 PM
1:25:43 PM
1:28:25 PM
1:31:07 PM
1:33:49 PM
1:36:31 PM
1:39:13 PM
1:41:55 PM
1:44:37 PM
1:47:19 PM
1:50:02 PM
1:52:44 PM
1:55:27 PM
\\VCC-RDSH06.vcc-sp.local\Physical Cpu(_Total)\% Core Util Time
\\VCC-RDSH07.vcc-sp.local\Physical Cpu(_Total)\% Core Util Time
\\VCC-RDSH08.vcc-sp.local\Physical Cpu(_Total)\% Core Util Time
\\VCC-RDSH09.vcc-sp.local\Physical Cpu(_Total)\% Core Util Time
\\VCC-RDSH10.vcc-sp.local\Physical Cpu(_Total)\% Core Util Time
\\VCC-RDSH10.vcc-sp.local\Physical Cpu(_Total)\% Core Util Time

Figure 127 Full Scale | 6000 Mixed Users | RDSH Host | Host Memory Utilization



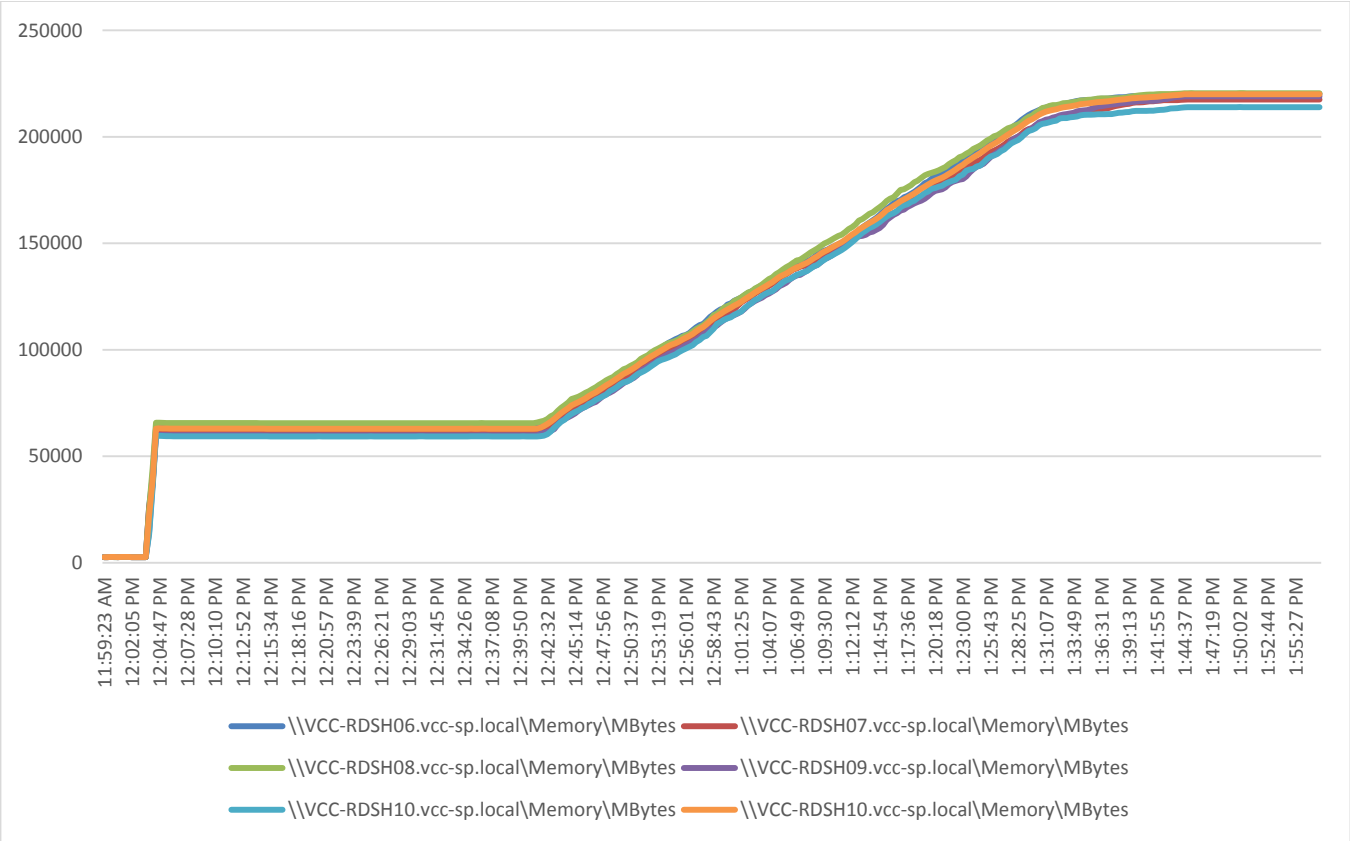
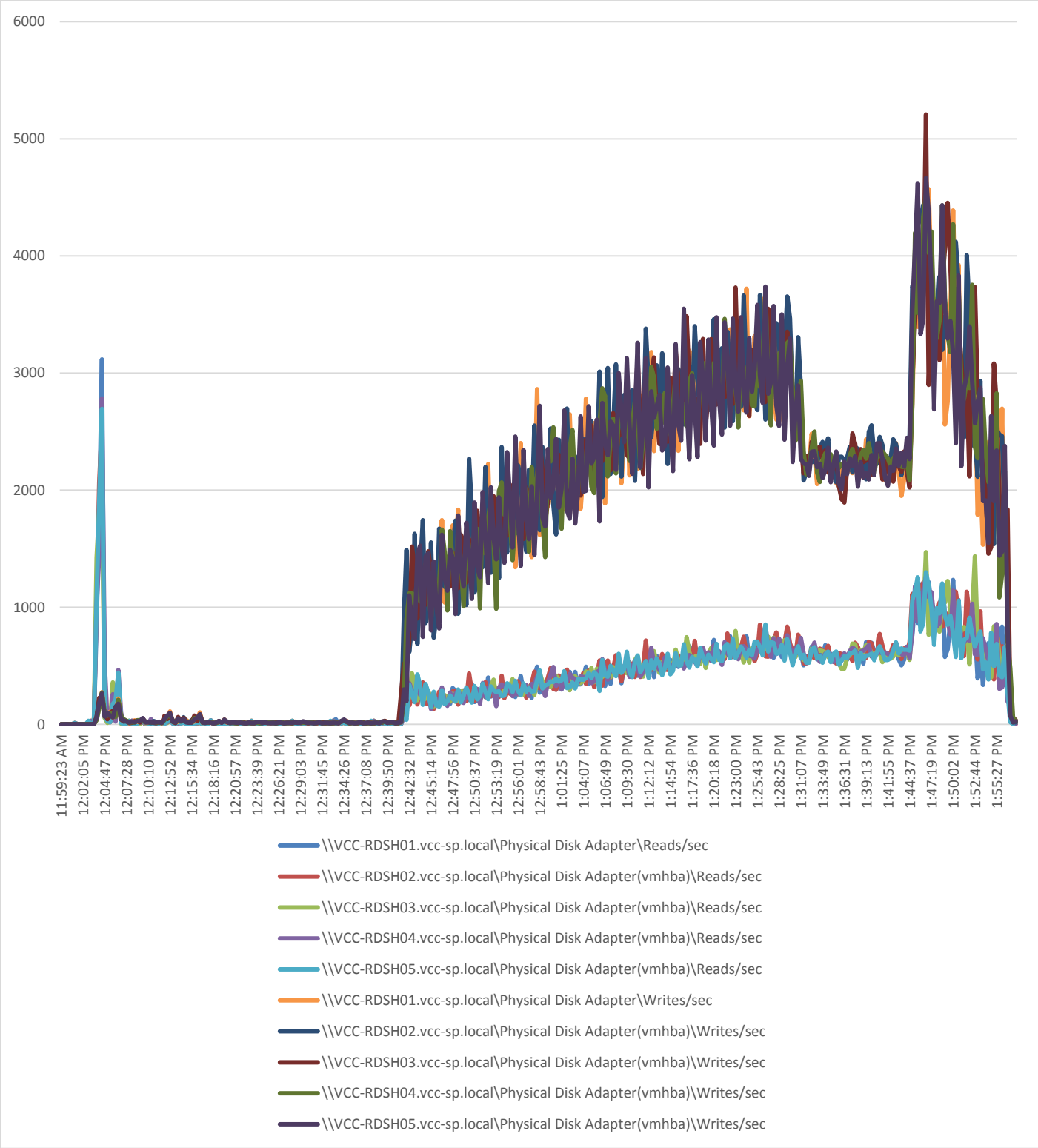


Figure 128

Figure 129Full Scale | 6000 Mixed Users | RDSH Host | Host Fibre Channel Network Utilization



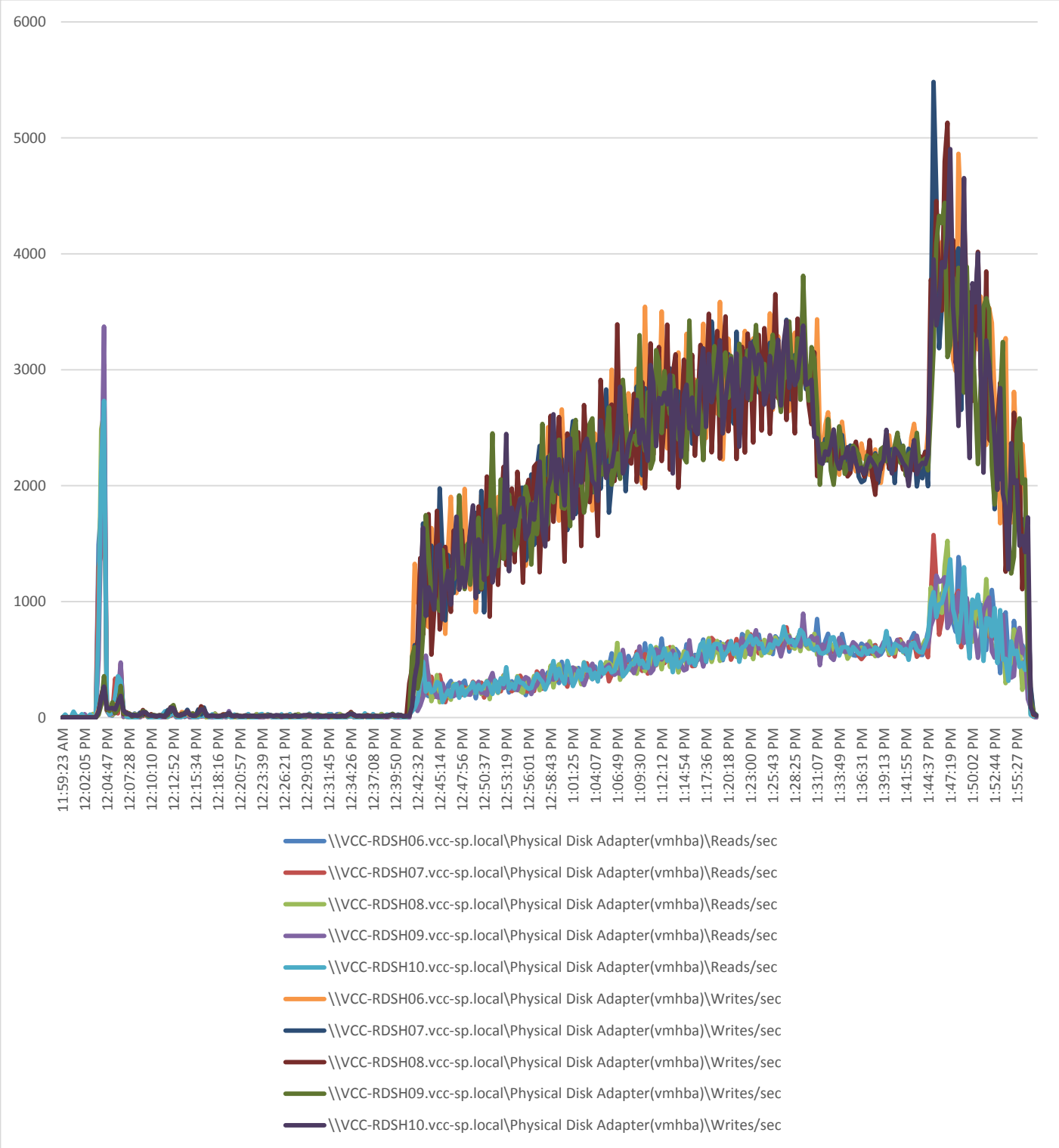
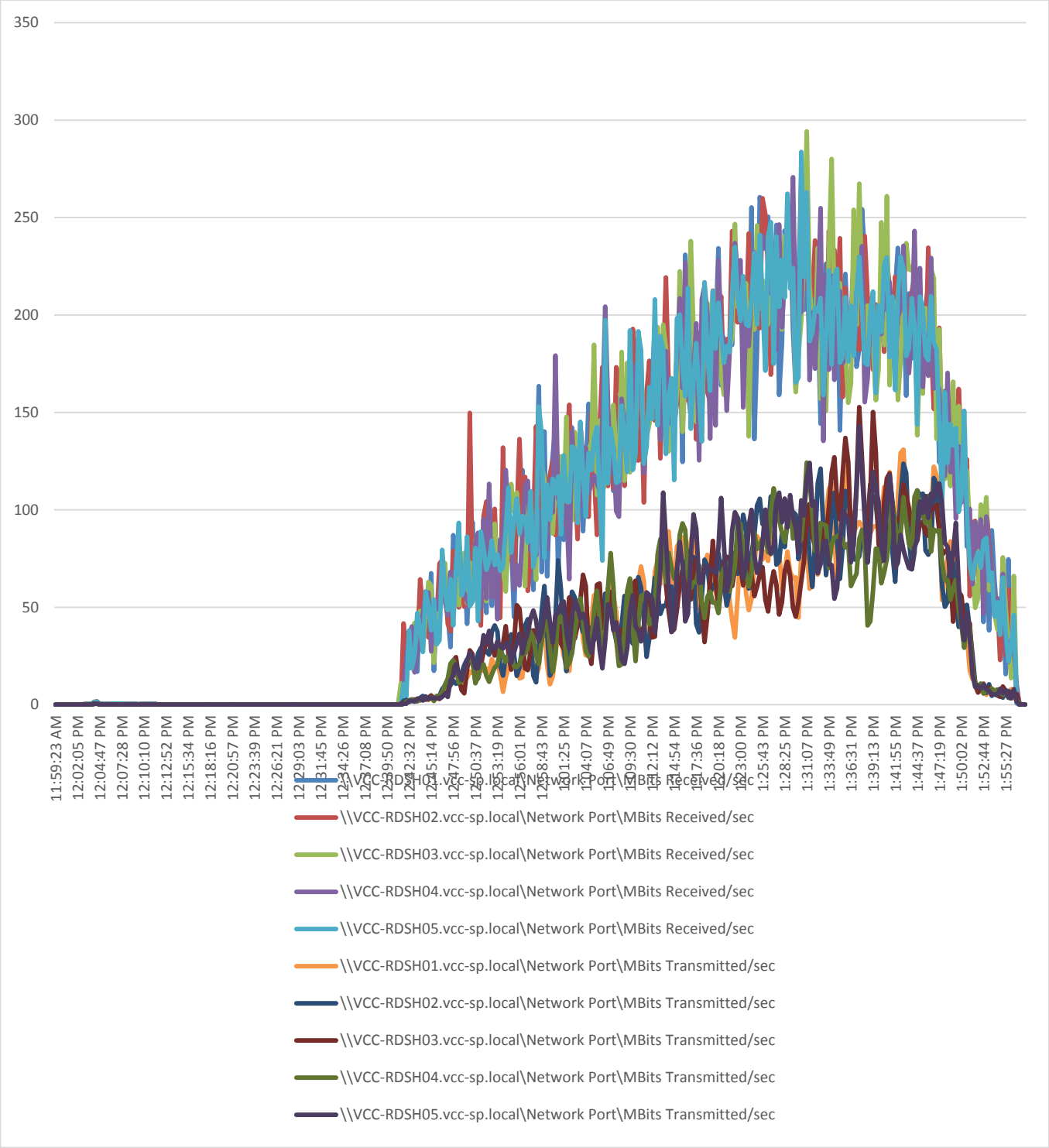
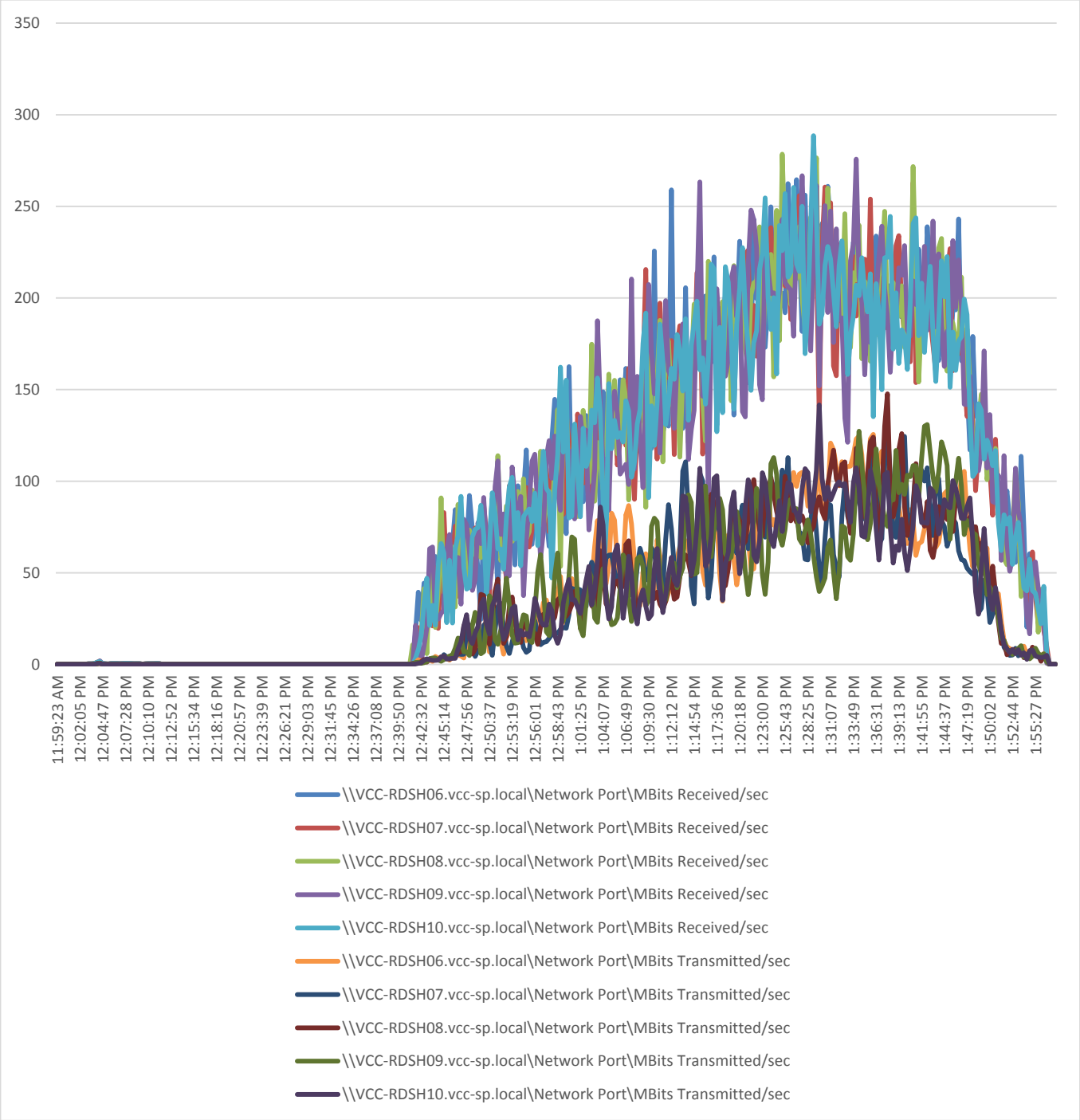

6000
5000
4000
3000
2000
1000
0
11:59:23 AM
12:02:05 PM
12:04:47 PM
12:07:28 PM
12:10:10 PM
12:12:52 PM
12:15:34 PM
12:18:16 PM
12:20:57 PM
12:23:39 PM
12:26:21 PM
12:29:03 PM
12:31:45 PM
12:34:26 PM
12:37:08 PM
12:39:50 PM
12:42:32 PM
12:45:14 PM
12:47:56 PM
12:50:37 PM
12:53:19 PM
12:56:01 PM
12:58:43 PM
1:01:25 PM
1:04:07 PM
1:06:49 PM
1:09:30 PM
1:12:12 PM
1:14:54 PM
1:17:36 PM
1:20:18 PM
1:23:00 PM
1:25:43 PM
1:28:25 PM
1:31:07 PM
1:33:49 PM
1:36:31 PM
1:39:13 PM
1:41:55 PM
1:44:37 PM
1:47:19 PM
1:50:02 PM
1:52:44 PM
1:55:27 PM
\\VCC-RDSH06.vcc-sp.local\Physical Disk Adapter(vmhba)\Reads/sec
\\VCC-RDSH07.vcc-sp.local\Physical Disk Adapter(vmhba)\Reads/sec
\\VCC-RDSH08.vcc-sp.local\Physical Disk Adapter(vmhba)\Reads/sec
\\VCC-RDSH09.vcc-sp.local\Physical Disk Adapter(vmhba)\Reads/sec
\\VCC-RDSH10.vcc-sp.local\Physical Disk Adapter(vmhba)\Reads/sec
\\VCC-RDSH06.vcc-sp.local\Physical Disk Adapter(vmhba)\Writes/sec
\\VCC-RDSH07.vcc-sp.local\Physical Disk Adapter(vmhba)\Writes/sec
\\VCC-RDSH08.vcc-sp.local\Physical Disk Adapter(vmhba)\Writes/sec
\\VCC-RDSH09.vcc-sp.local\Physical Disk Adapter(vmhba)\Writes/sec
\\VCC-RDSH10.vcc-sp.local\Physical Disk Adapter(vmhba)\Writes/sec

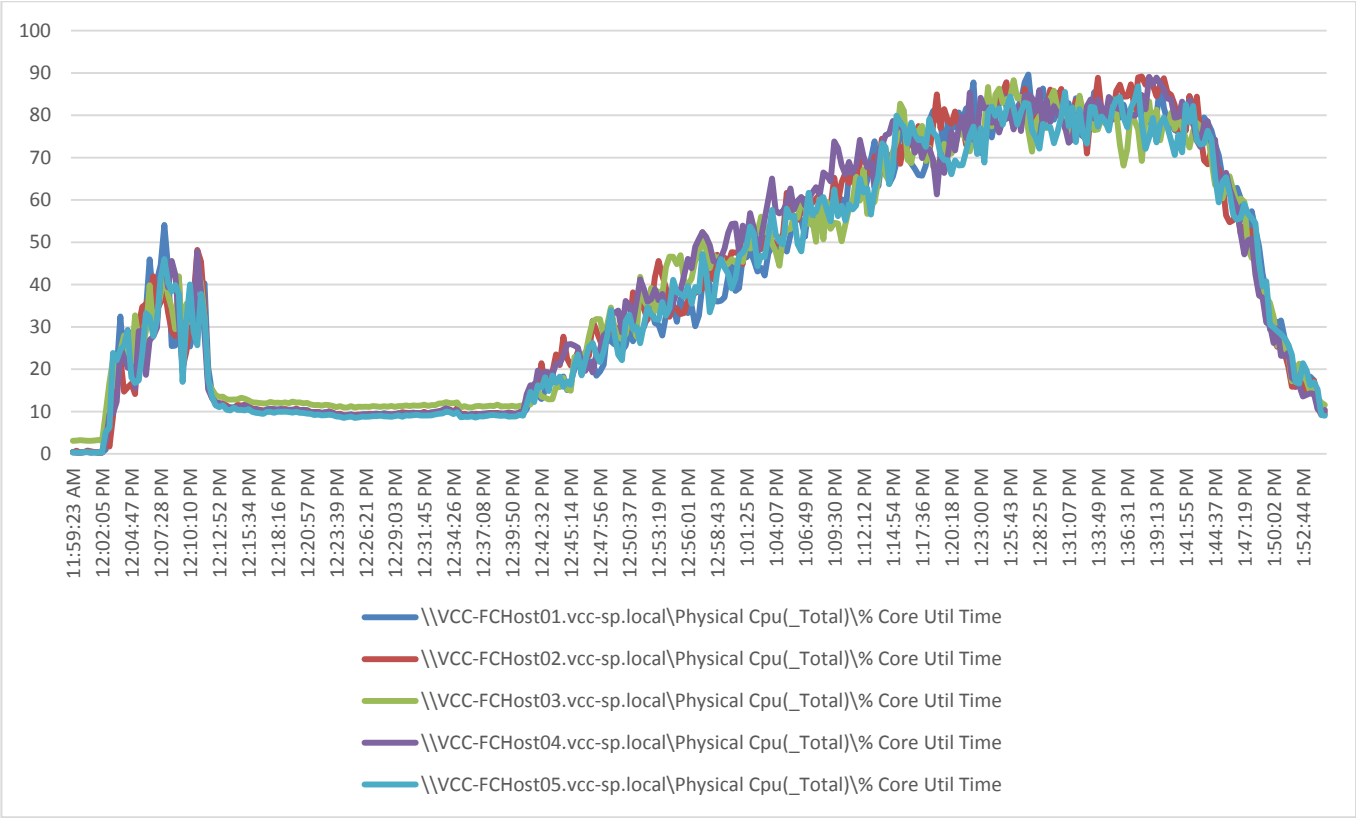
Figure 130Full Scale | 6000 Mixed Users | RDSH Host | Host Network Utilization

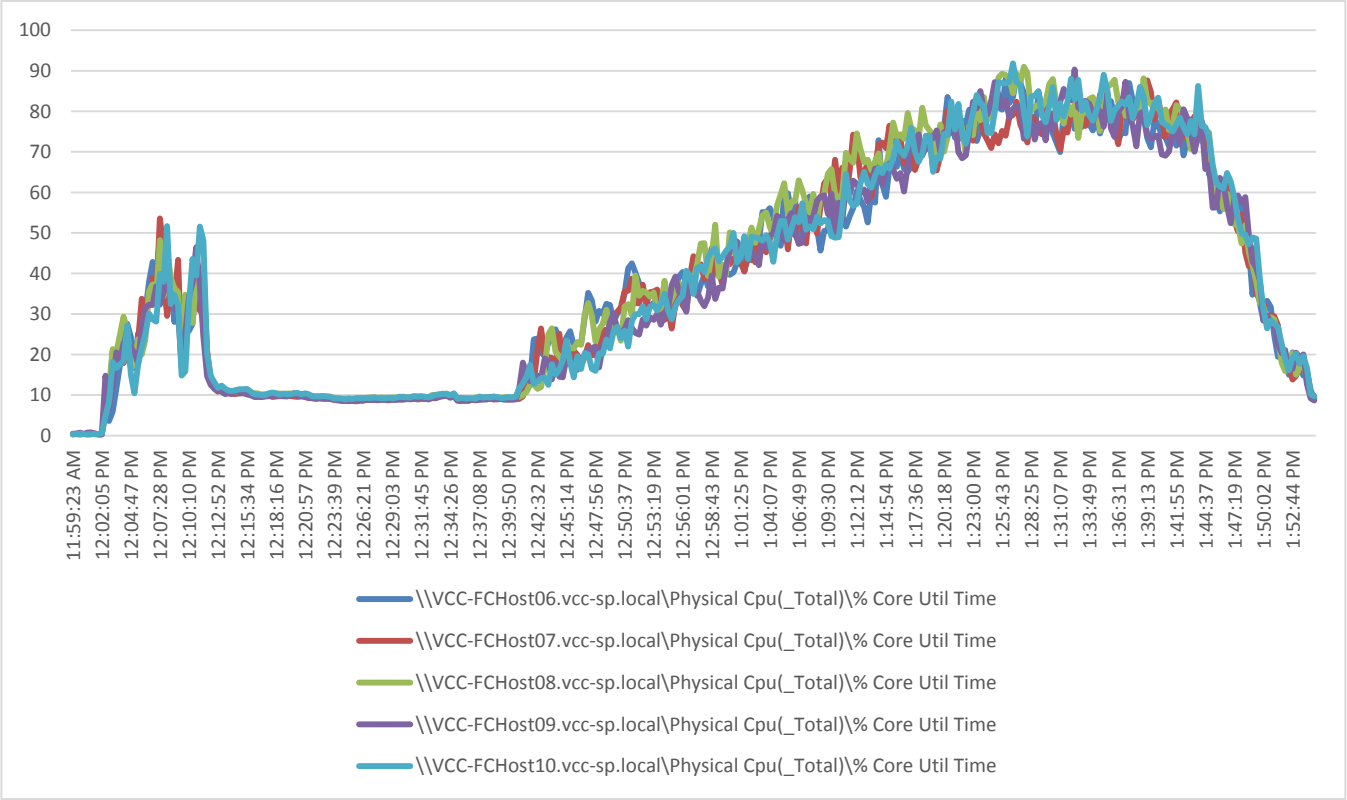


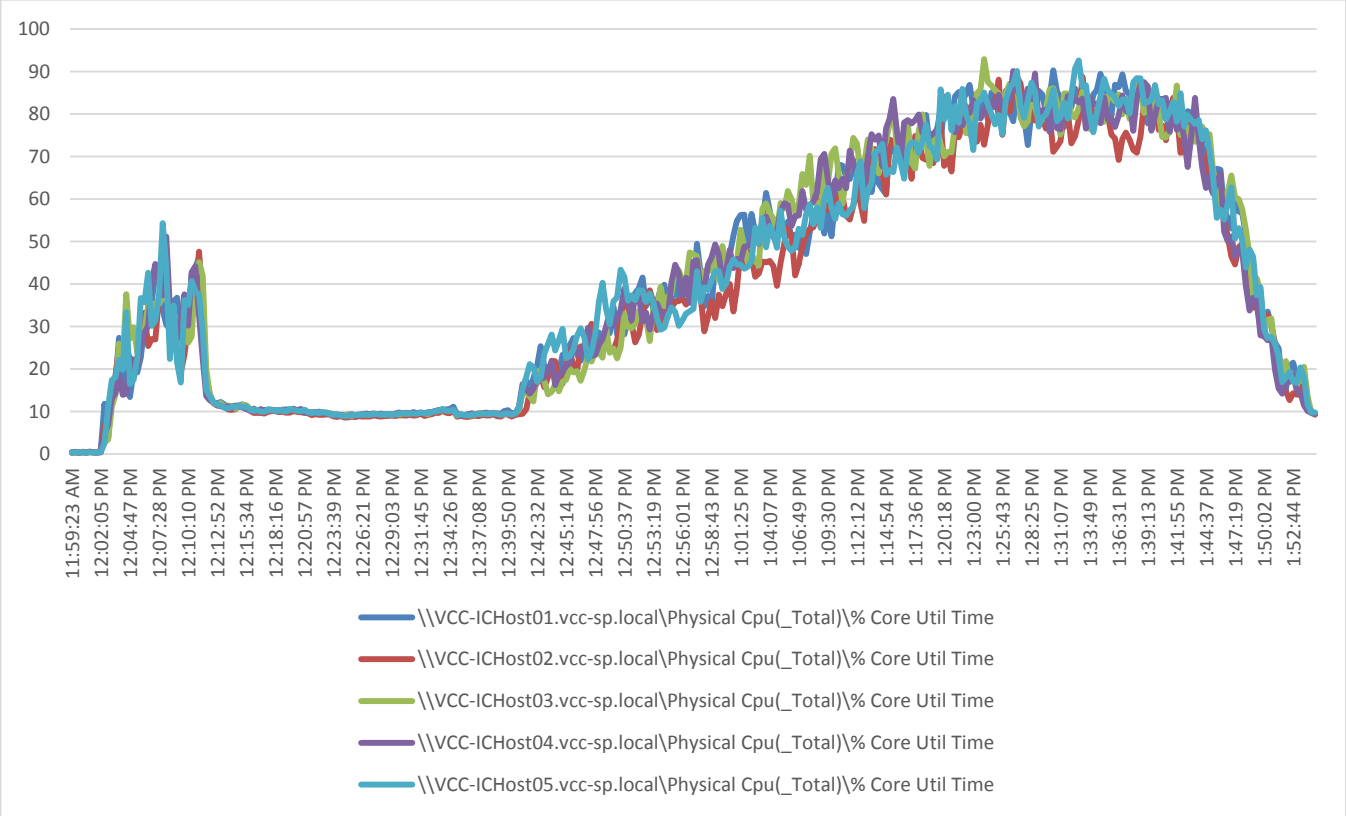

350
300
250
200
150
100
50
0
11:59:23 AM
12:02:05 PM
12:04:47 PM
12:07:28 PM
12:10:10 PM
12:12:52 PM
12:15:34 PM
12:18:16 PM
12:20:57 PM
12:23:39 PM
12:26:21 PM
12:29:03 PM
12:31:45 PM
12:34:26 PM
12:37:08 PM
12:39:50 PM
12:42:32 PM
12:45:14 PM
12:47:56 PM
12:50:37 PM
12:53:19 PM
12:56:01 PM
12:58:43 PM
1:01:25 PM
1:04:07 PM
1:06:49 PM
1:09:30 PM
1:12:12 PM
1:14:54 PM
1:17:36 PM
1:20:18 PM
1:23:00 PM
1:25:43 PM
1:28:25 PM
1:31:07 PM
1:33:49 PM
1:36:31 PM
1:39:13 PM
1:41:55 PM
1:44:37 PM
1:47:19 PM
1:50:02 PM
1:52:44 PM
1:55:27 PM
\\VCC-RDSH06.vcc-sp.local\Network Port\MBits Received/sec
\\VCC-RDSH07.vcc-sp.local\Network Port\MBits Received/sec
\\VCC-RDSH08.vcc-sp.local\Network Port\MBits Received/sec
\\VCC-RDSH09.vcc-sp.local\Network Port\MBits Received/sec
\\VCC-RDSH10.vcc-sp.local\Network Port\MBits Received/sec
\\VCC-RDSH06.vcc-sp.local\Network Port\MBits Transmitted/sec
\\VCC-RDSH07.vcc-sp.local\Network Port\MBits Transmitted/sec
\\VCC-RDSH08.vcc-sp.local\Network Port\MBits Transmitted/sec
\\VCC-RDSH09.vcc-sp.local\Network Port\MBits Transmitted/sec
\\VCC-RDSH10.vcc-sp.local\Network Port\MBits Transmitted/sec

HVD Server Performance Monitor Data for One Sample HVD Server: 6000 Users Mixed Scale Testing

Figure 131Full Scale | 6000 Mixed Users | VDI Host | Host CPU Utilization






100
90
80
70
60
50
40
30
20
10
0
11:59:23 AM
12:02:05 PM
12:04:47 PM
12:07:28 PM
12:10:10 PM
12:12:52 PM
12:15:34 PM
12:18:16 PM
12:20:57 PM
12:23:39 PM
12:26:21 PM
12:29:03 PM
12:31:45 PM
12:34:26 PM
12:37:08 PM
12:39:50 PM
12:42:32 PM
12:45:14 PM
12:47:56 PM
12:50:37 PM
12:53:19 PM
12:56:01 PM
12:58:43 PM
1:01:25 PM
1:04:07 PM
1:06:49 PM
1:09:30 PM
1:12:12 PM
1:14:54 PM
1:17:36 PM
1:20:18 PM
1:23:00 PM
1:25:43 PM
1:28:25 PM
1:31:07 PM
1:33:49 PM
1:36:31 PM
1:39:13 PM
1:41:55 PM
1:44:37 PM
1:47:19 PM
1:50:02 PM
1:52:44 PM
\\VCC-ICHost01.vcc-sp.local\Physical Cpu(_Total)\% Core Util Time
\\VCC-ICHost02.vcc-sp.local\Physical Cpu(_Total)\% Core Util Time
\\VCC-ICHost03.vcc-sp.local\Physical Cpu(_Total)\% Core Util Time
\\VCC-ICHost04.vcc-sp.local\Physical Cpu(_Total)\% Core Util Time
\\VCC-ICHost05.vcc-sp.local\Physical Cpu(_Total)\% Core Util Time

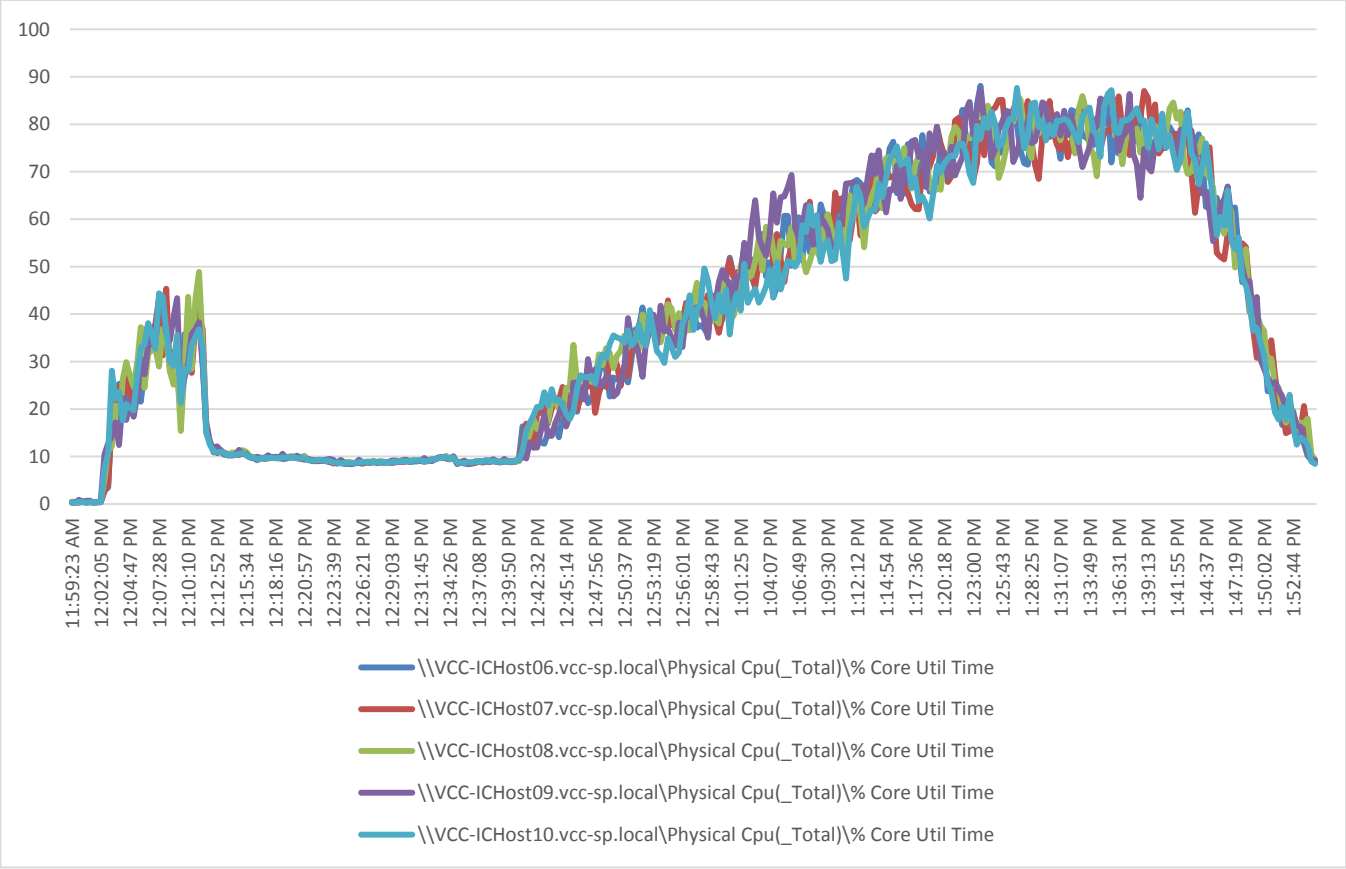
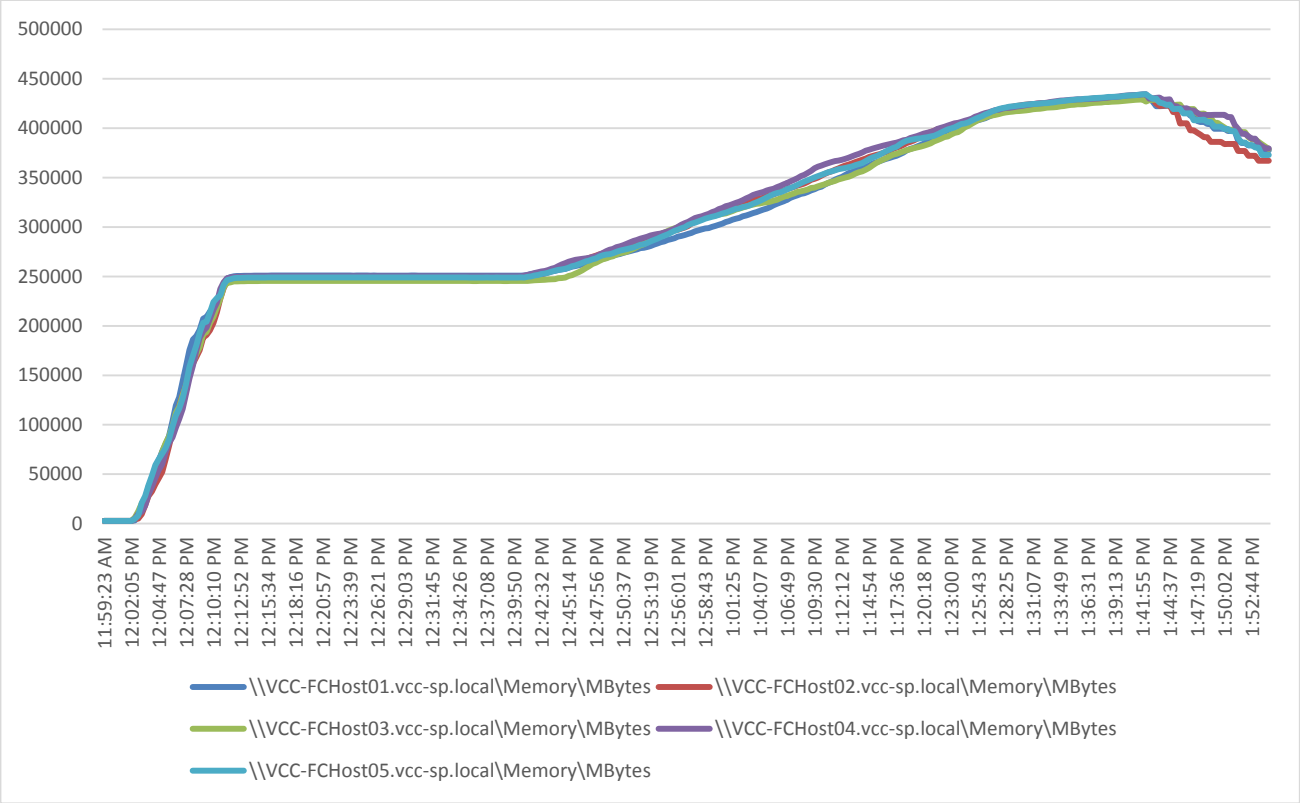
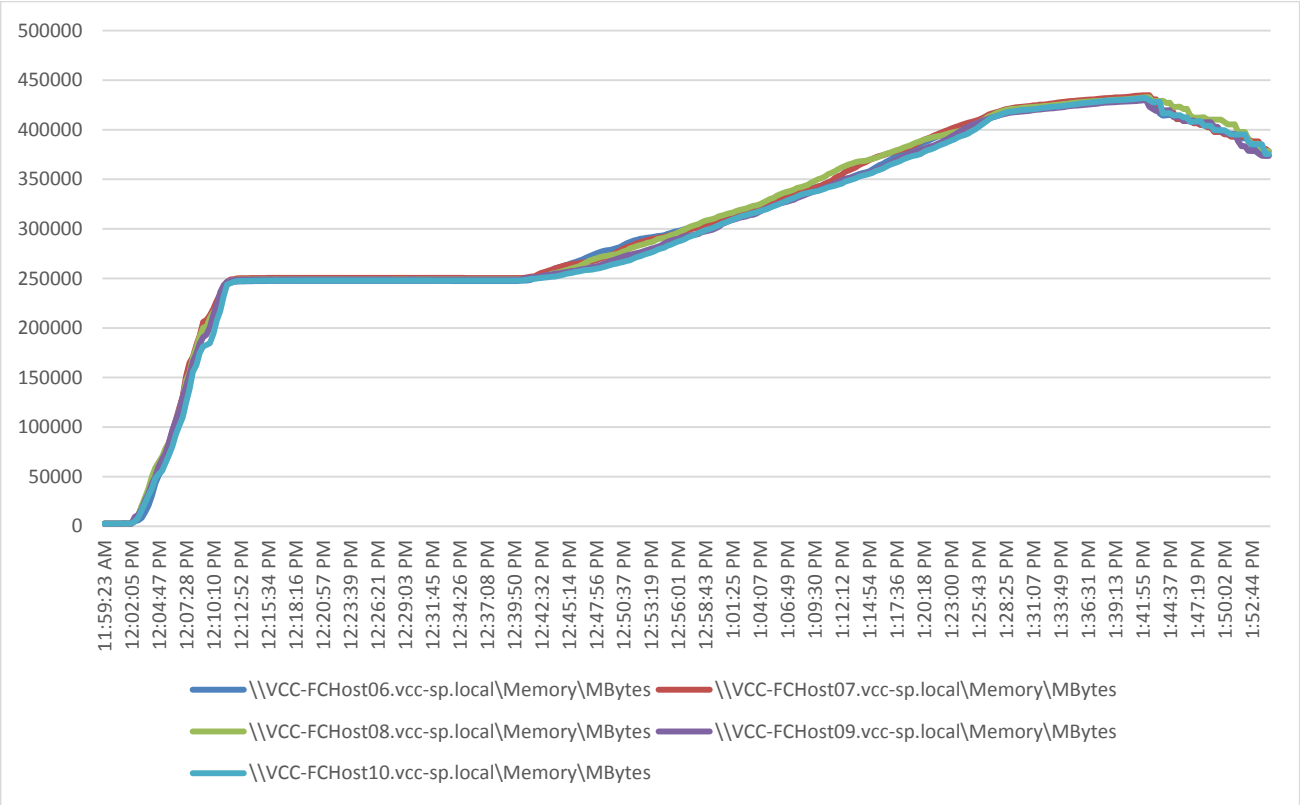
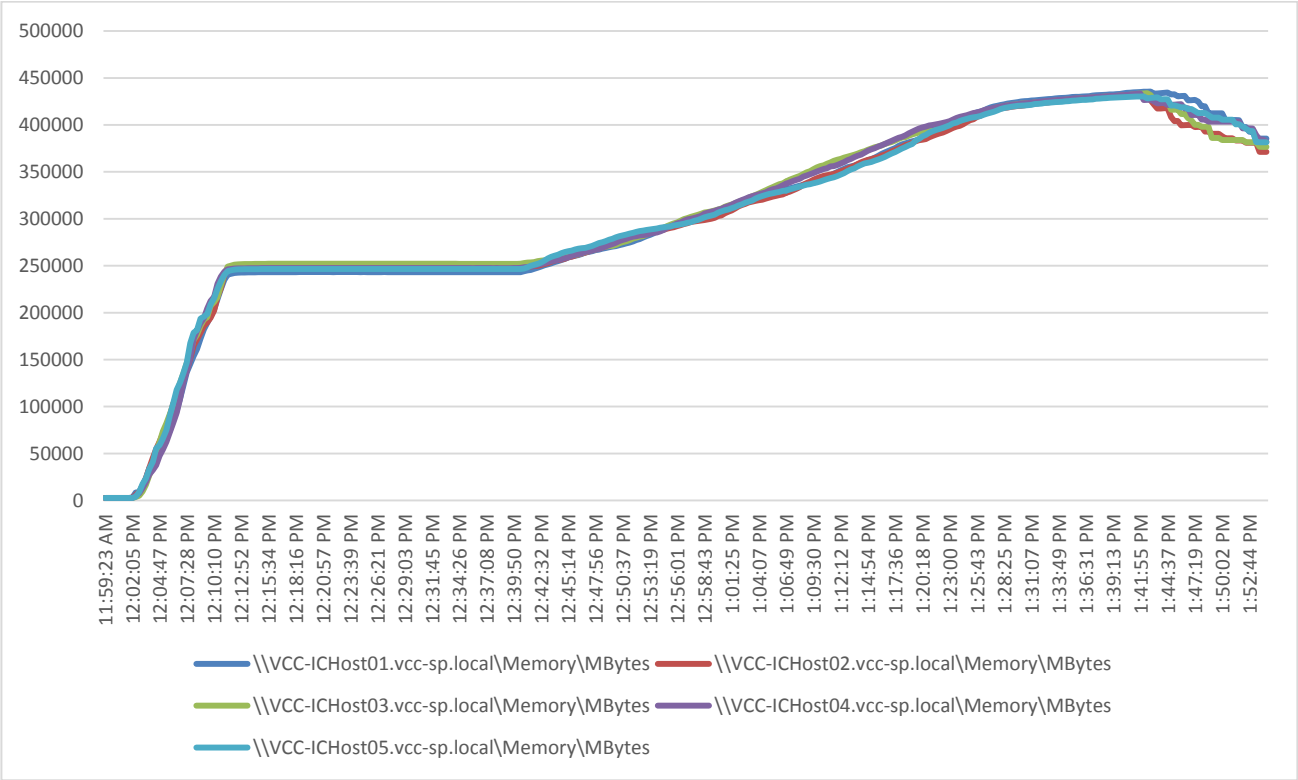

100
90
80
70
60
50
40
30
20
10
0
11:59:23 AM
12:02:05 PM
12:04:47 PM
12:07:28 PM
12:10:10 PM
12:12:52 PM
12:15:34 PM
12:18:16 PM
12:20:57 PM
12:23:39 PM
12:26:21 PM
12:29:03 PM
12:31:45 PM
12:34:26 PM
12:37:08 PM
12:39:50 PM
12:42:32 PM
12:45:14 PM
12:47:56 PM
12:50:37 PM
12:53:19 PM
12:56:01 PM
12:58:43 PM
1:01:25 PM
1:04:07 PM
1:06:49 PM
1:09:30 PM
1:12:12 PM
1:14:54 PM
1:17:36 PM
1:20:18 PM
1:23:00 PM
1:25:43 PM
1:28:25 PM
1:31:07 PM
1:33:49 PM
1:36:31 PM
1:39:13 PM
1:41:55 PM
1:44:37 PM
1:47:19 PM
1:50:02 PM
1:52:44 PM
\\VCC-ICHost06.vcc-sp.local\Physical Cpu(_Total)\% Core Util Time
\\VCC-ICHost07.vcc-sp.local\Physical Cpu(_Total)\% Core Util Time
\\VCC-ICHost08.vcc-sp.local\Physical Cpu(_Total)\% Core Util Time
\\VCC-ICHost09.vcc-sp.local\Physical Cpu(_Total)\% Core Util Time
\\VCC-ICHost10.vcc-sp.local\Physical Cpu(_Total)\% Core Util Time

Figure 132Full Scale | 6000 Mixed Users | VDI Host | Host Memory Utilization






500000
450000
400000
350000
300000
250000
200000
150000
100000
50000
0
11:59:23 AM
12:02:05 PM
12:04:47 PM
12:07:28 PM
12:10:10 PM
12:12:52 PM
12:15:34 PM
12:18:16 PM
12:20:57 PM
12:23:39 PM
12:26:21 PM
12:29:03 PM
12:31:45 PM
12:34:26 PM
12:37:08 PM
12:39:50 PM
12:42:32 PM
12:45:14 PM
12:47:56 PM
12:50:37 PM
12:53:19 PM
12:56:01 PM
12:58:43 PM
1:01:25 PM
1:04:07 PM
1:06:49 PM
1:09:30 PM
1:12:12 PM
1:14:54 PM
1:17:36 PM
1:20:18 PM
1:23:00 PM
1:25:43 PM
1:28:25 PM
1:31:07 PM
1:33:49 PM
1:36:31 PM
1:39:13 PM
1:41:55 PM
1:44:37 PM
1:47:19 PM
1:50:02 PM
1:52:44 PM
\\VCC-ICHost01.vcc-sp.local\Memory\MBytes
\\VCC-ICHost02.vcc-sp.local\Memory\MBytes
\\VCC-ICHost03.vcc-sp.local\Memory\MBytes
\\VCC-ICHost04.vcc-sp.local\Memory\MBytes
\\VCC-ICHost05.vcc-sp.local\Memory\MBytes

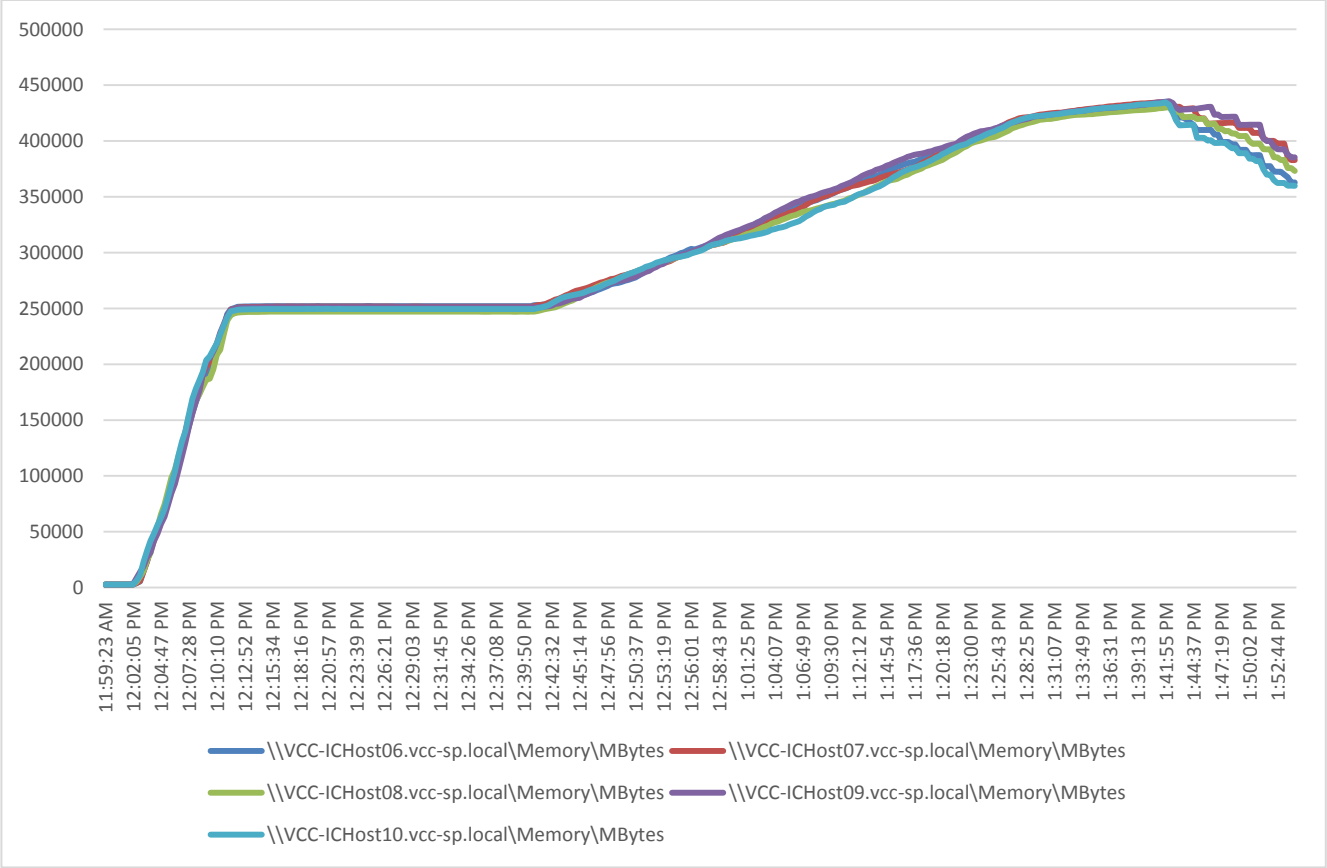
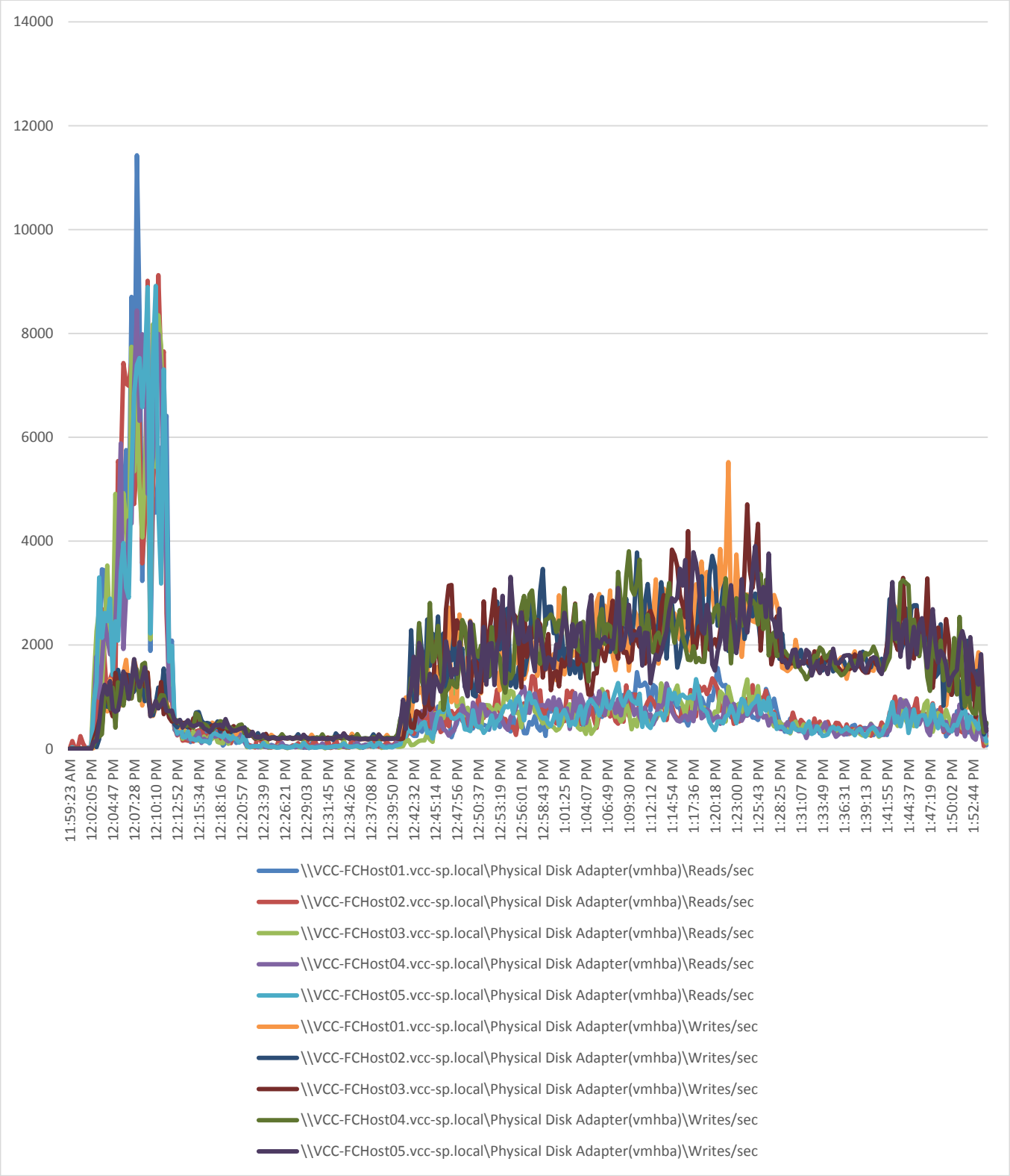
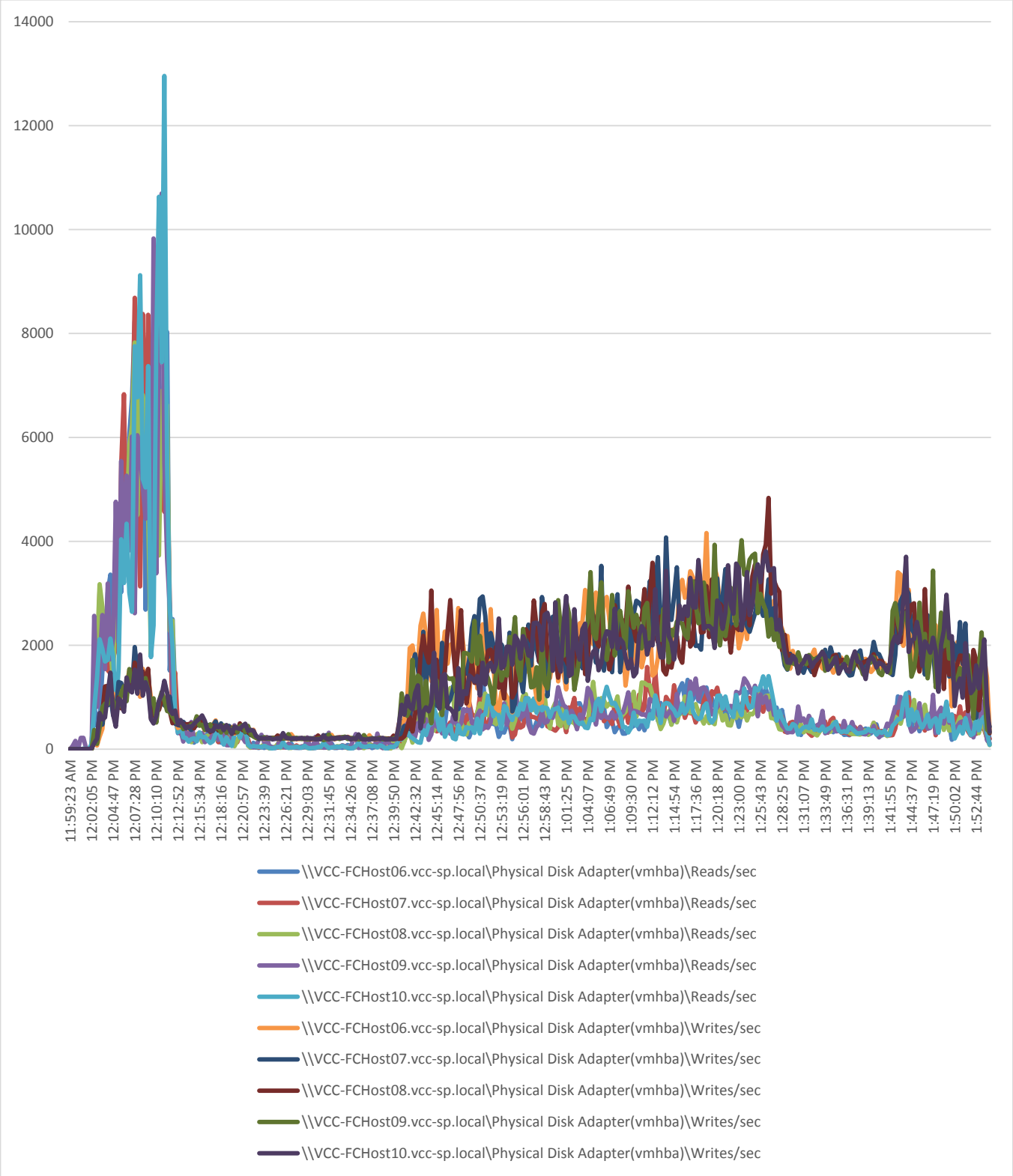
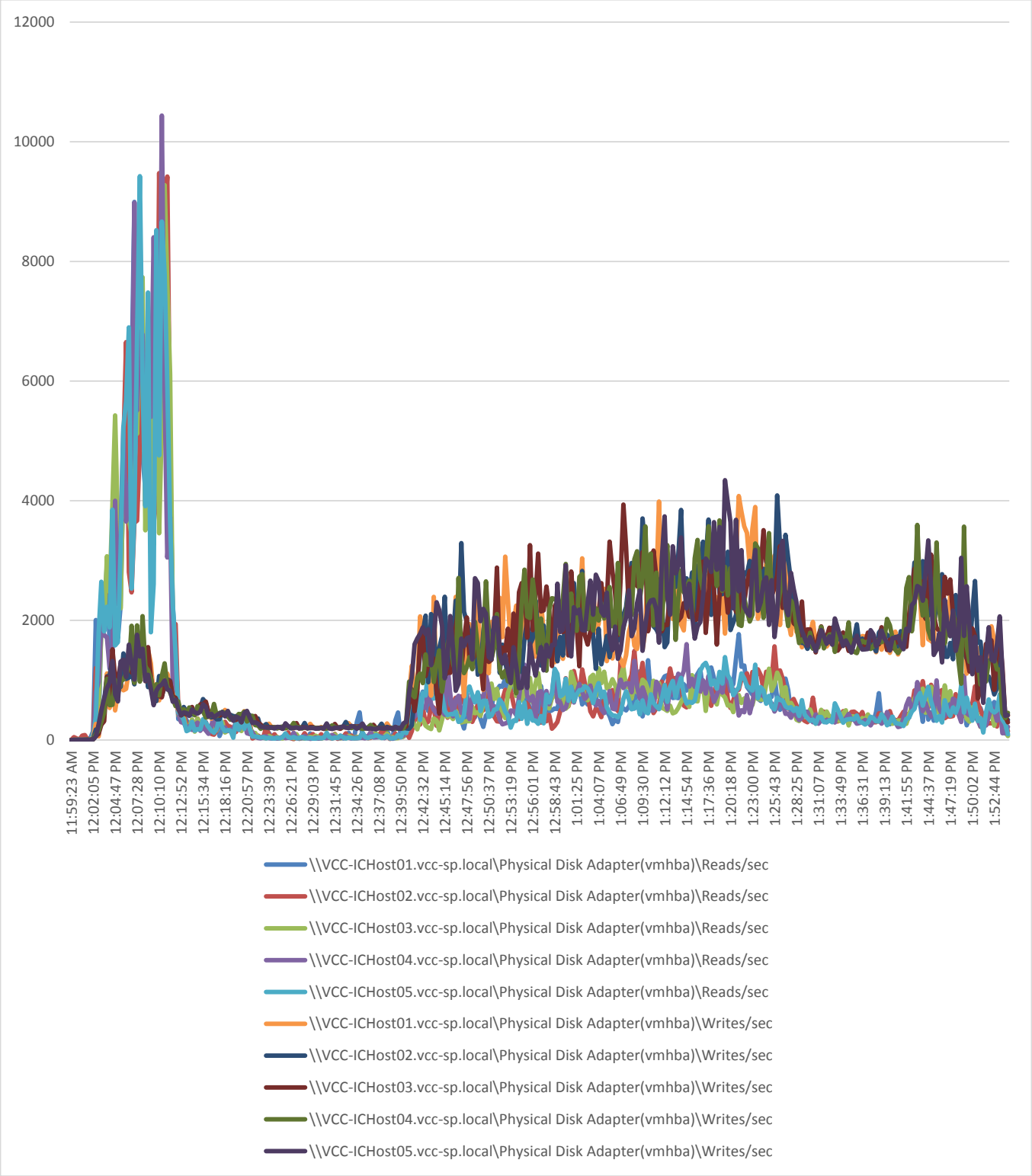

500000
450000
400000
350000
300000
250000
200000
150000
100000
50000
0
11:59:23 AM
12:02:05 PM
12:04:47 PM
12:07:28 PM
12:10:10 PM
12:12:52 PM
12:15:34 PM
12:18:16 PM
12:20:57 PM
12:23:39 PM
12:26:21 PM
12:29:03 PM
12:31:45 PM
12:34:26 PM
12:37:08 PM
12:39:50 PM
12:42:32 PM
12:45:14 PM
12:47:56 PM
12:50:37 PM
12:53:19 PM
12:56:01 PM
12:58:43 PM
1:01:25 PM
1:04:07 PM
1:06:49 PM
1:09:30 PM
1:12:12 PM
1:14:54 PM
1:17:36 PM
1:20:18 PM
1:23:00 PM
1:25:43 PM
1:28:25 PM
1:31:07 PM
1:33:49 PM
1:36:31 PM
1:39:13 PM
1:41:55 PM
1:44:37 PM
1:47:19 PM
1:50:02 PM
1:52:44 PM
\\VCC-ICHost06.vcc-sp.local\Memory\MBytes
\\VCC-ICHost07.vcc-sp.local\Memory\MBytes
\\VCC-ICHost08.vcc-sp.local\Memory\MBytes
\\VCC-ICHost09.vcc-sp.local\Memory\MBytes
\\VCC-ICHost10.vcc-sp.local\Memory\MBytes

Figure 133Full Scale | 6000 Mixed Users | VDI Host | Host Fibre Channel Network Utilization




14000
12000
10000
8000
6000
4000
2000
0
11:59:23 AM
12:02:05 PM
12:04:47 PM
12:07:28 PM
12:10:10 PM
12:12:52 PM
12:15:34 PM
12:18:16 PM
12:20:57 PM
12:23:39 PM
12:26:21 PM
12:29:03 PM
12:31:45 PM
12:34:26 PM
12:37:08 PM
12:39:50 PM
12:42:32 PM
12:45:14 PM
12:47:56 PM
12:50:37 PM
12:53:19 PM
12:56:01 PM
12:58:43 PM
1:01:25 PM
1:04:07 PM
1:06:49 PM
1:09:30 PM
1:12:12 PM
1:14:54 PM
1:17:36 PM
1:20:18 PM
1:23:00 PM
1:25:43 PM
1:28:25 PM
1:31:07 PM
1:33:49 PM
1:36:31 PM
1:39:13 PM
1:41:55 PM
1:44:37 PM
1:47:19 PM
1:50:02 PM
1:52:44 PM
\\VCC-FCHost06.vcc-sp.local\Physical Disk Adapter(vmhba)\Reads/sec
\\VCC-FCHost07.vcc-sp.local\Physical Disk Adapter(vmhba)\Reads/sec
\\VCC-FCHost08.vcc-sp.local\Physical Disk Adapter(vmhba)\Reads/sec
\\VCC-FCHost09.vcc-sp.local\Physical Disk Adapter(vmhba)\Reads/sec
\\VCC-FCHost10.vcc-sp.local\Physical Disk Adapter(vmhba)\Reads/sec
\\VCC-FCHost06.vcc-sp.local\Physical Disk Adapter(vmhba)\Writes/sec
\\VCC-FCHost07.vcc-sp.local\Physical Disk Adapter(vmhba)\Writes/sec
\\VCC-FCHost08.vcc-sp.local\Physical Disk Adapter(vmhba)\Writes/sec
\\VCC-FCHost09.vcc-sp.local\Physical Disk Adapter(vmhba)\Writes/sec
\\VCC-FCHost10.vcc-sp.local\Physical Disk Adapter(vmhba)\Writes/sec


12000
10000
8000
6000
4000
2000
0
11:59:23 AM
12:02:05 PM
12:04:47 PM
12:07:28 PM
12:10:10 PM
12:12:52 PM
12:15:34 PM
12:18:16 PM
12:20:57 PM
12:23:39 PM
12:26:21 PM
12:29:03 PM
12:31:45 PM
12:34:26 PM
12:37:08 PM
12:39:50 PM
12:42:32 PM
12:45:14 PM
12:47:56 PM
12:50:37 PM
12:53:19 PM
12:56:01 PM
12:58:43 PM
1:01:25 PM
1:04:07 PM
1:06:49 PM
1:09:30 PM
1:12:12 PM
1:14:54 PM
1:17:36 PM
1:20:18 PM
1:23:00 PM
1:25:43 PM
1:28:25 PM
1:31:07 PM
1:33:49 PM
1:36:31 PM
1:39:13 PM
1:41:55 PM
1:44:37 PM
1:47:19 PM
1:50:02 PM
1:52:44 PM
\\VCC-ICHost01.vcc-sp.local\Physical Disk Adapter(vmhba)\Reads/sec
\\VCC-ICHost02.vcc-sp.local\Physical Disk Adapter(vmhba)\Reads/sec
\\VCC-ICHost03.vcc-sp.local\Physical Disk Adapter(vmhba)\Reads/sec
\\VCC-ICHost04.vcc-sp.local\Physical Disk Adapter(vmhba)\Reads/sec
\\VCC-ICHost05.vcc-sp.local\Physical Disk Adapter(vmhba)\Reads/sec
\\VCC-ICHost01.vcc-sp.local\Physical Disk Adapter(vmhba)\Writes/sec
\\VCC-ICHost02.vcc-sp.local\Physical Disk Adapter(vmhba)\Writes/sec
\\VCC-ICHost03.vcc-sp.local\Physical Disk Adapter(vmhba)\Writes/sec
\\VCC-ICHost04.vcc-sp.local\Physical Disk Adapter(vmhba)\Writes/sec
\\VCC-ICHost05.vcc-sp.local\Physical Disk Adapter(vmhba)\Writes/sec

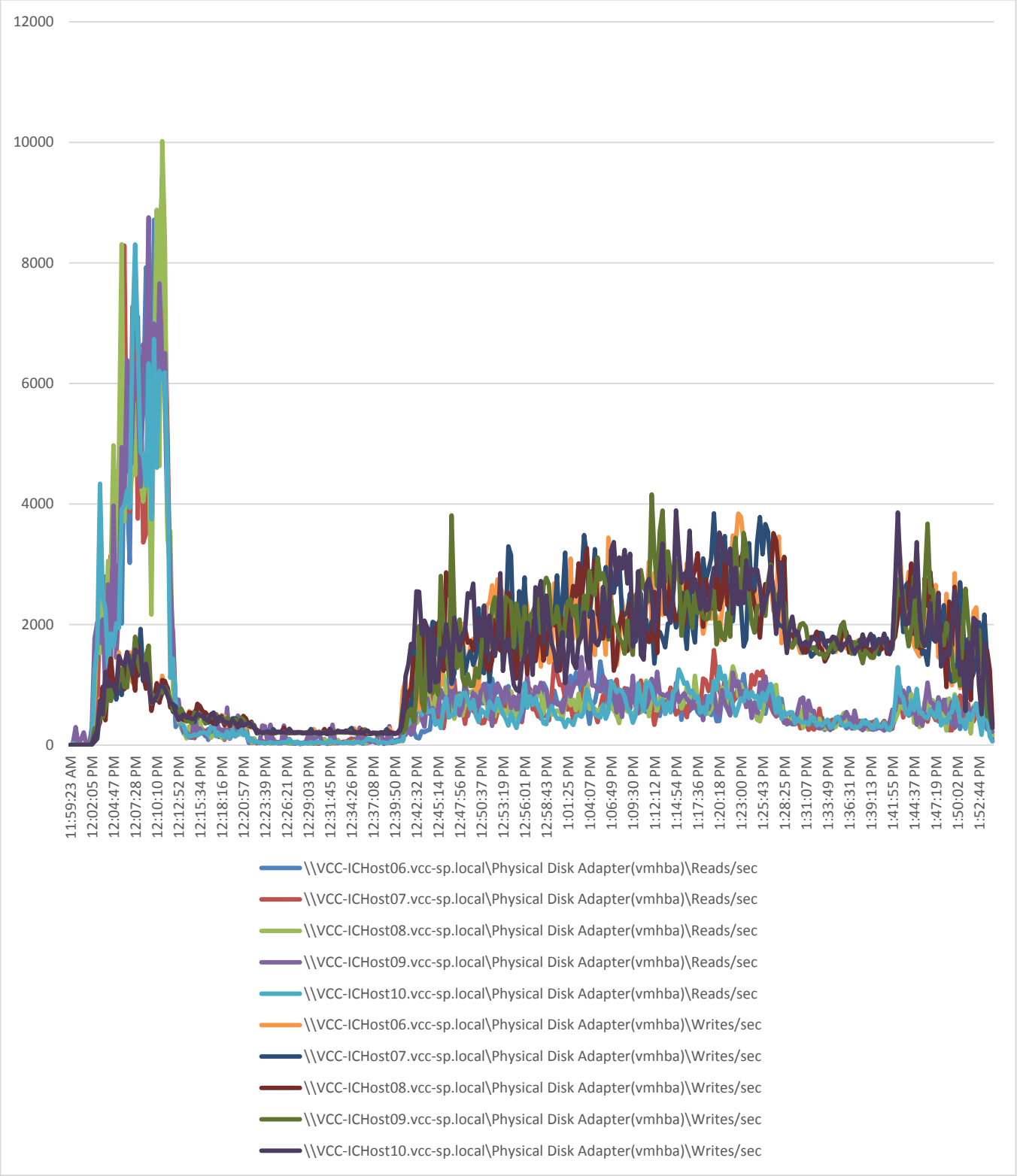
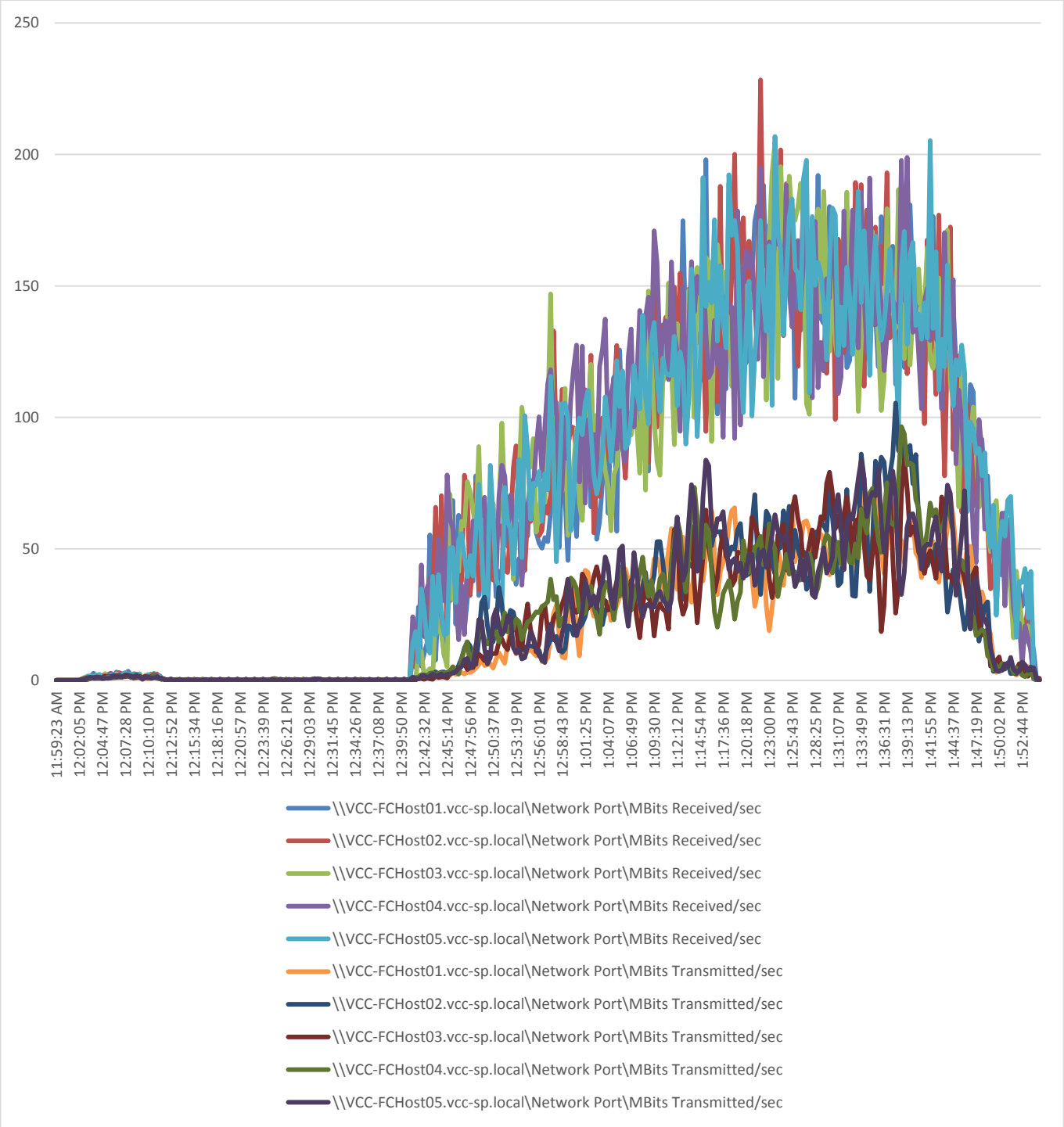
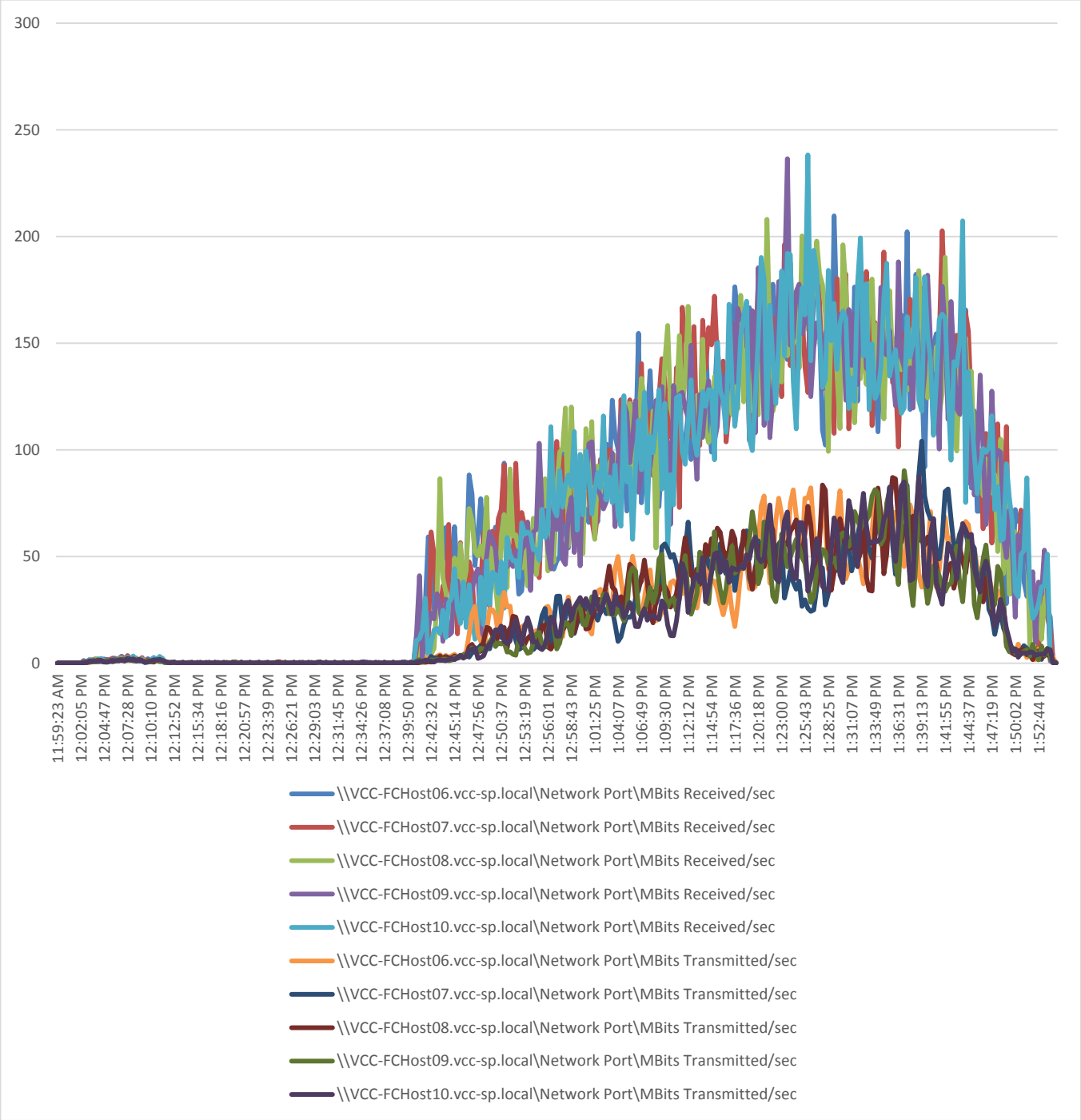
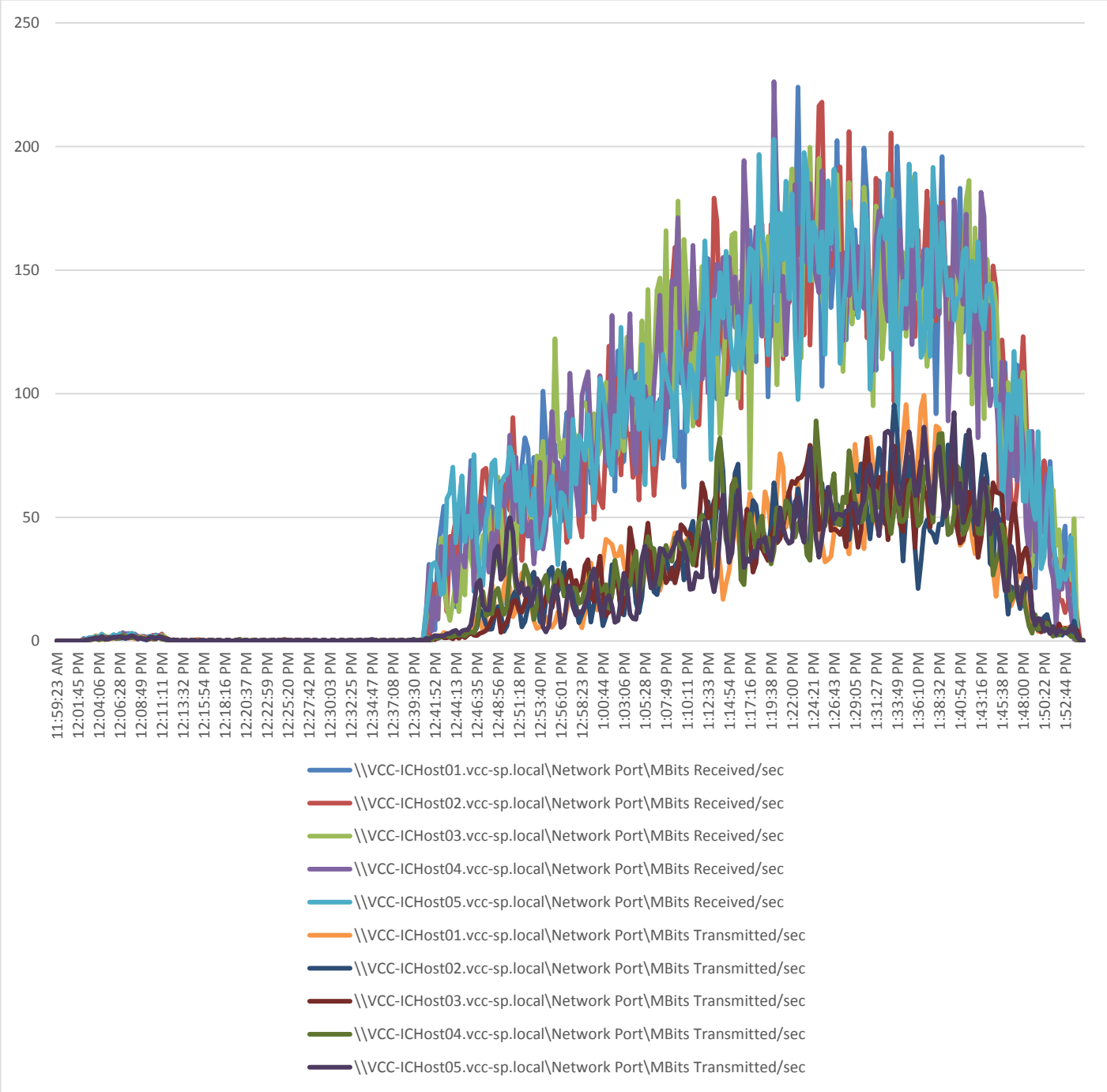
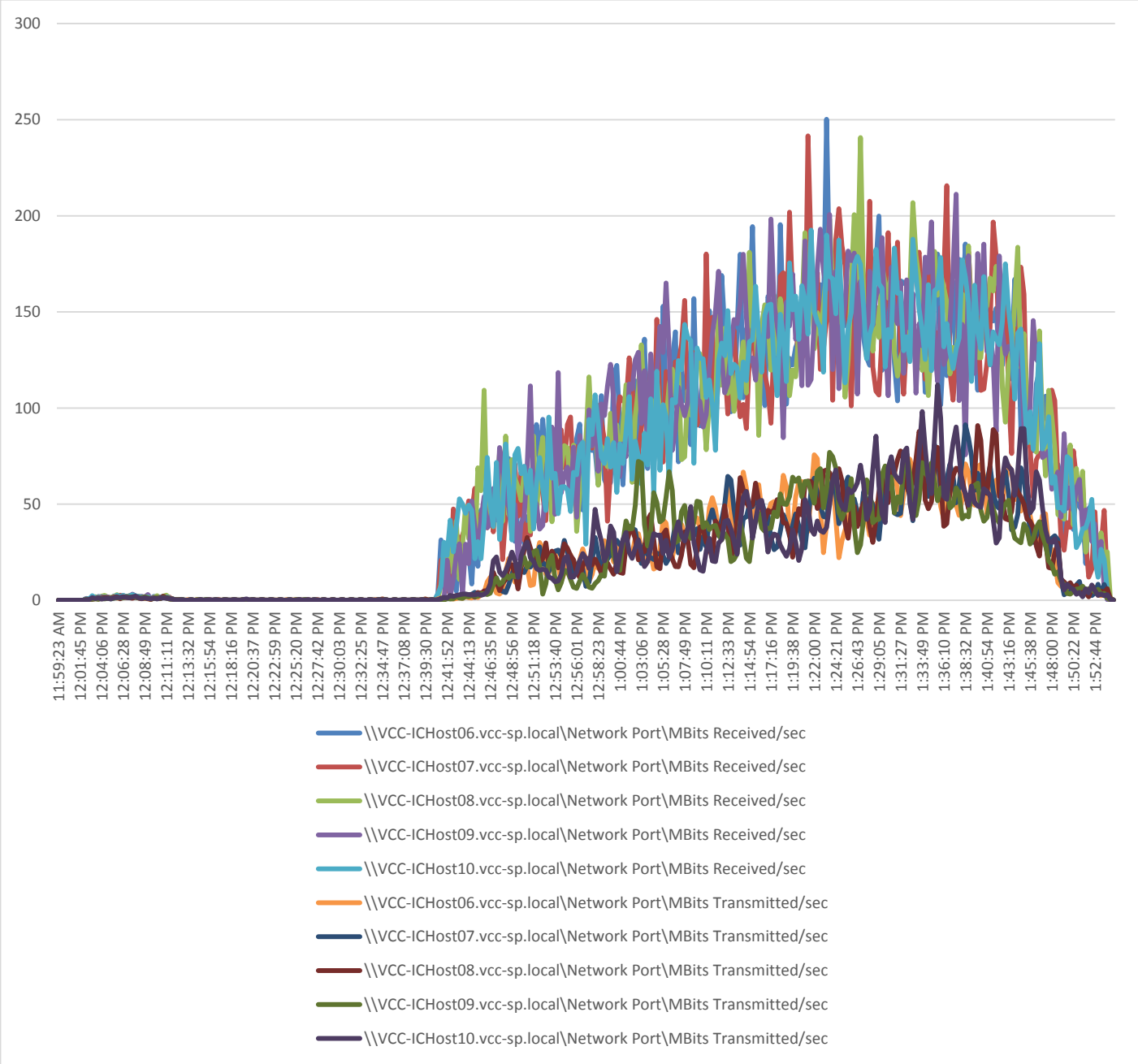

12000
10000
8000
6000
4000
2000
0
11:59:23 AM
12:02:05 PM
12:04:47 PM
12:07:28 PM
12:10:10 PM
12:12:52 PM
12:15:34 PM
12:18:16 PM
12:20:57 PM
12:23:39 PM
12:26:21 PM
12:29:03 PM
12:31:45 PM
12:34:26 PM
12:37:08 PM
12:39:50 PM
12:42:32 PM
12:45:14 PM
12:47:56 PM
12:50:37 PM
12:53:19 PM
12:56:01 PM
12:58:43 PM
1:01:25 PM
1:04:07 PM
1:06:49 PM
1:09:30 PM
1:12:12 PM
1:14:54 PM
1:17:36 PM
1:20:18 PM
1:23:00 PM
1:25:43 PM
1:28:25 PM
1:31:07 PM
1:33:49 PM
1:36:31 PM
1:39:13 PM
1:41:55 PM
1:44:37 PM
1:47:19 PM
1:50:02 PM
1:52:44 PM
\\VCC-ICHost06.vcc-sp.local\Physical Disk Adapter(vmhba)\Reads/sec
\\VCC-ICHost07.vcc-sp.local\Physical Disk Adapter(vmhba)\Reads/sec
\\VCC-ICHost08.vcc-sp.local\Physical Disk Adapter(vmhba)\Reads/sec
\\VCC-ICHost09.vcc-sp.local\Physical Disk Adapter(vmhba)\Reads/sec
\\VCC-ICHost10.vcc-sp.local\Physical Disk Adapter(vmhba)\Reads/sec
\\VCC-ICHost06.vcc-sp.local\Physical Disk Adapter(vmhba)\Writes/sec
\\VCC-ICHost07.vcc-sp.local\Physical Disk Adapter(vmhba)\Writes/sec
\\VCC-ICHost08.vcc-sp.local\Physical Disk Adapter(vmhba)\Writes/sec
\\VCC-ICHost09.vcc-sp.local\Physical Disk Adapter(vmhba)\Writes/sec
\\VCC-ICHost10.vcc-sp.local\Physical Disk Adapter(vmhba)\Writes/sec

Figure 134Full Scale | 6000 Mixed Users | VDI Host | Host Network Utilization




300
250
200
150
100
50
0
11:59:23 AM
12:02:05 PM
12:04:47 PM
12:07:28 PM
12:10:10 PM
12:12:52 PM
12:15:34 PM
12:18:16 PM
12:20:57 PM
12:23:39 PM
12:26:21 PM
12:29:03 PM
12:31:45 PM
12:34:26 PM
12:37:08 PM
12:39:50 PM
12:42:32 PM
12:45:14 PM
12:47:56 PM
12:50:37 PM
12:53:19 PM
12:56:01 PM
12:58:43 PM
1:01:25 PM
1:04:07 PM
1:06:49 PM
1:09:30 PM
1:12:12 PM
1:14:54 PM
1:17:36 PM
1:20:18 PM
1:23:00 PM
1:25:43 PM
1:28:25 PM
1:31:07 PM
1:33:49 PM
1:36:31 PM
1:39:13 PM
1:41:55 PM
1:44:37 PM
1:47:19 PM
1:50:02 PM
1:52:44 PM
\\VCC-FCHost06.vcc-sp.local\Network Port\MBits Received/sec
\\VCC-FCHost07.vcc-sp.local\Network Port\MBits Received/sec
\\VCC-FCHost08.vcc-sp.local\Network Port\MBits Received/sec
\\VCC-FCHost09.vcc-sp.local\Network Port\MBits Received/sec
\\VCC-FCHost10.vcc-sp.local\Network Port\MBits Received/sec
\\VCC-FCHost06.vcc-sp.local\Network Port\MBits Transmitted/sec
\\VCC-FCHost07.vcc-sp.local\Network Port\MBits Transmitted/sec
\\VCC-FCHost08.vcc-sp.local\Network Port\MBits Transmitted/sec
\\VCC-FCHost09.vcc-sp.local\Network Port\MBits Transmitted/sec
\\VCC-FCHost10.vcc-sp.local\Network Port\MBits Transmitted/sec


250
200
150
100
50
0
11:59:23 AM
12:01:45 PM
12:04:06 PM
12:06:28 PM
12:08:49 PM
12:11:11 PM
12:13:32 PM
12:15:54 PM
12:18:16 PM
12:20:37 PM
12:22:59 PM
12:25:20 PM
12:27:42 PM
12:30:03 PM
12:32:25 PM
12:34:47 PM
12:37:08 PM
12:39:30 PM
12:41:52 PM
12:44:13 PM
12:46:35 PM
12:48:56 PM
12:51:18 PM
12:53:40 PM
12:56:01 PM
12:58:23 PM
1:00:44 PM
1:03:06 PM
1:05:28 PM
1:07:49 PM
1:10:11 PM
1:12:33 PM
1:14:54 PM
1:17:16 PM
1:19:38 PM
1:22:00 PM
1:24:21 PM
1:26:43 PM
1:29:05 PM
1:31:27 PM
1:33:49 PM
1:36:10 PM
1:38:32 PM
1:40:54 PM
1:43:16 PM
1:45:38 PM
1:48:00 PM
1:50:22 PM
1:52:44 PM
\\VCC-ICHost01.vcc-sp.local\Network Port\MBits Received/sec
\\VCC-ICHost02.vcc-sp.local\Network Port\MBits Received/sec
\\VCC-ICHost03.vcc-sp.local\Network Port\MBits Received/sec
\\VCC-ICHost04.vcc-sp.local\Network Port\MBits Received/sec
\\VCC-ICHost05.vcc-sp.local\Network Port\MBits Received/sec
\\VCC-ICHost01.vcc-sp.local\Network Port\MBits Transmitted/sec
\\VCC-ICHost02.vcc-sp.local\Network Port\MBits Transmitted/sec
\\VCC-ICHost03.vcc-sp.local\Network Port\MBits Transmitted/sec
\\VCC-ICHost04.vcc-sp.local\Network Port\MBits Transmitted/sec
\\VCC-ICHost05.vcc-sp.local\Network Port\MBits Transmitted/sec


300
250
200
150
100
50
0
11:59:23 AM
12:01:45 PM
12:04:06 PM
12:06:28 PM
12:08:49 PM
12:11:11 PM
12:13:32 PM
12:15:54 PM
12:18:16 PM
12:20:37 PM
12:22:59 PM
12:25:20 PM
12:27:42 PM
12:30:03 PM
12:32:25 PM
12:34:47 PM
12:37:08 PM
12:39:30 PM
12:41:52 PM
12:44:13 PM
12:46:35 PM
12:48:56 PM
12:51:18 PM
12:53:40 PM
12:56:01 PM
12:58:23 PM
1:00:44 PM
1:03:06 PM
1:05:28 PM
1:07:49 PM
1:10:11 PM
1:12:33 PM
1:14:54 PM
1:17:16 PM
1:19:38 PM
1:22:00 PM
1:24:21 PM
1:26:43 PM
1:29:05 PM
1:31:27 PM
1:33:49 PM
1:36:10 PM
1:38:32 PM
1:40:54 PM
1:43:16 PM
1:45:38 PM
1:48:00 PM
1:50:22 PM
1:52:44 PM
\\VCC-ICHost06.vcc-sp.local\Network Port\MBits Received/sec
\\VCC-ICHost07.vcc-sp.local\Network Port\MBits Received/sec
\\VCC-ICHost08.vcc-sp.local\Network Port\MBits Received/sec
\\VCC-ICHost09.vcc-sp.local\Network Port\MBits Received/sec
\\VCC-ICHost10.vcc-sp.local\Network Port\MBits Received/sec
\\VCC-ICHost06.vcc-sp.local\Network Port\MBits Transmitted/sec
\\VCC-ICHost07.vcc-sp.local\Network Port\MBits Transmitted/sec
\\VCC-ICHost08.vcc-sp.local\Network Port\MBits Transmitted/sec
\\VCC-ICHost09.vcc-sp.local\Network Port\MBits Transmitted/sec
\\VCC-ICHost10.vcc-sp.local\Network Port\MBits Transmitted/sec