Virtual Design Master

Season 5

Challenge 1

We Can Rebuild Earth… We Have the Technology

Table of Contents

Executive Summary

Project Overview

We are working on building an army of robots to carry out the terraforming efforts. The new version of the HumanityLink application will add the features needed for the scheduling, operations, and maintenance of the robot fleet, in an effort to rebuild our home world.

In our effort to rebuild Earth, we are designing HumanityLink to span a resilient 3-site architecture [**RQ01**]. Resiliency and redundancy are paramount at each site, as we need to support 24/7 operations for our three types of terraforming robots: terrestrial, air, and water-based drones for operations such as dredging [**RQ02**]. An architecture must support unknown workloads; it must be able to scale in all directions [**RQ03**]. Coastal locations [**R01**], with multiple connectivity, power, and cooling options [**A10**, **A11**], have been chosen to begin rebuilding, to facilitate these requirements.

No budget constraints have been identified.

Intended Audience

Any engineer willing, or capable of reviewing, implementing, or maintaining this design. HumanityLink and the future of humanity needs all the technical help it can get.

Project Summary

Requirements

#	Description
RQ01	Resilient 3-site architecture
RQ02	24/7 uptime for scheduling, operations, and maintenance of the robot fleet
RQ03	Able to scale, easily, in any direction (up, out).

Table 1

Assumptions

#	Description
A01	Each site will run an independent "instance" of the terraforming module of HumanityLink, including separate frontend, middleware, and backend (data) services
A02	Each robot can return to base, autonomously, for updates and maintenance
A03	Each site's instance is completely self-reliant, and has independent hardware from other sites
A04	Terraforming data-sets, instructions, and conditions will be unique to each site
A05	Each site will share operational data to improve task efficiencies across all sites (metadata, task/routing optimizations)

A06	Autonomous drone ships are available for long-distance transport of equipment to other sites, for use mitigating [**R01**]
A07	Each site has the ability to assume half of another site's working robots for assimilation into local operations in the event of an entire site failure, scaling resources appropriately
A08	NTP, DNS is available at each site
A09	A documented RBAC process exists for adds/modifications/removals
A10	Redundant power and cooling available
A11	Dual connections (MPLS, Internet) at each site
A12	Design and deployment will be on Earth, with Earth-like environmental conditions
A13	The zombies have been eradicated

Table 2

Constraints

#	Description
C01	x86_64 Architecture
C02	Earth-like conditions

Table 3

Risks

#	Description	Mitigation
R01	Each site is costal, introducing natural hazards that are unique to that geographic feature	Drone ships for displacement of drones, backup and disaster recovery plan
R02	Cisco ASR 1002-HX provides no hardware-level redundancy	Redundant hardware (Internet, MPLS)
R03	Unknown workload	Scalable infrastructure

Table 4

Description

The infrastructure will consist of matching platforms, based off Cisco's FlexPod with OpenStack architecture, across three locations [**RQ01**]:

1. Sydney, Australia
2. Tokyo, Japan
3. Seattle, Washington, USA.

Because each site has independent, site-specific data on terraforming operations, most data is not actively shared between each site [**A01**]. Robots at each site will have individual data downloaded to a local unit at the start of a shift, and the robot will automatically return-to-base at the end of its working period [**A02**]. The operation of each site is not dependent on the operations of the others [**A03**]. Each site will store its own set of working data, unique to that site [**A04**]. Shared data between sites will include operational data to help determine where efficiencies can be made in operations throughout all sites [**A05**], as well as a backup location as detailed in **Table 5**.

Each site will be running at 50% capacity, to provide adequate robot assimilation of 50% of a single site's robot in the event of a failure, leaving room for resource overhead when planning operations for more drones.

In the event of a site failure, by natural disaster or otherwise, the remaining sites will provide enough physical, compute, storage, and networking capacity to assume control of the remaining robots for the remaining respective sites [**A07**].

The sites were specifically placed on the same large body of water for initial architecture rollout, testing, and disaster recovery abilities. Further expansion based off this model can be further refined to other unique geographical topologies as required. Autonomous drone ships [**A06**] will provide transport of robots and drones long-distance for equal disbursement between the remaining sites in the event a natural disaster can be predicted (hurricane, earthquake/tsunami, volcano, or otherwise) and evacuation possible.

Each site will employ a standard design based on Cisco's FlexPod Datacenter with RHEL OpenStack platform 6.0, featuring end-to-end hardware-level redundancy with Cisco UCS and NetApp high-availability features, to allow for scalability (up || out) in compute, storage, and/or networking if/when required [**RQ03**].

Each site will utilize the next-in-line location for backups of configuration and VM data based on the following table:

Source	Destination
Sydney, Australia	Tokyo, Japan
Tokyo, Japan	Seattle, Washington, USA
Seattle, Washington, USA	Sydney, Australia

Table 5

In the event of a non-recoverable failure of a site, the remaining drones will be distributed, and backup data can be restored to the original location once the infrastructure is rebuilt and/or brought back online.

Sites will be inter-linked using Cisco's Intelligent WAN (IWAN) hybrid concepts and design. This will allow for encrypted communications for up to 1000 dual-homed locations within a single domain. This scalability will allow our site-independent architecture to scale the number of sites for future use. Each site will have dual (count: 2) edge routers: one serving the MPLS transport, and the second Internet transport for redundancy against hardware failures [**R02**, **A11**]. Each router connects to a pair of distribution layer switches for additional redundancy. Primary internet traffic will be directed out from each site, internal (site/site) traffic will be routed over the MPLS initially, with failover to the Internet/DMVPN path. Conversely, the reverse failover method used for internet based traffic.

Design Summary

Conceptual Overview

Cisco FlexPod Site Architecture

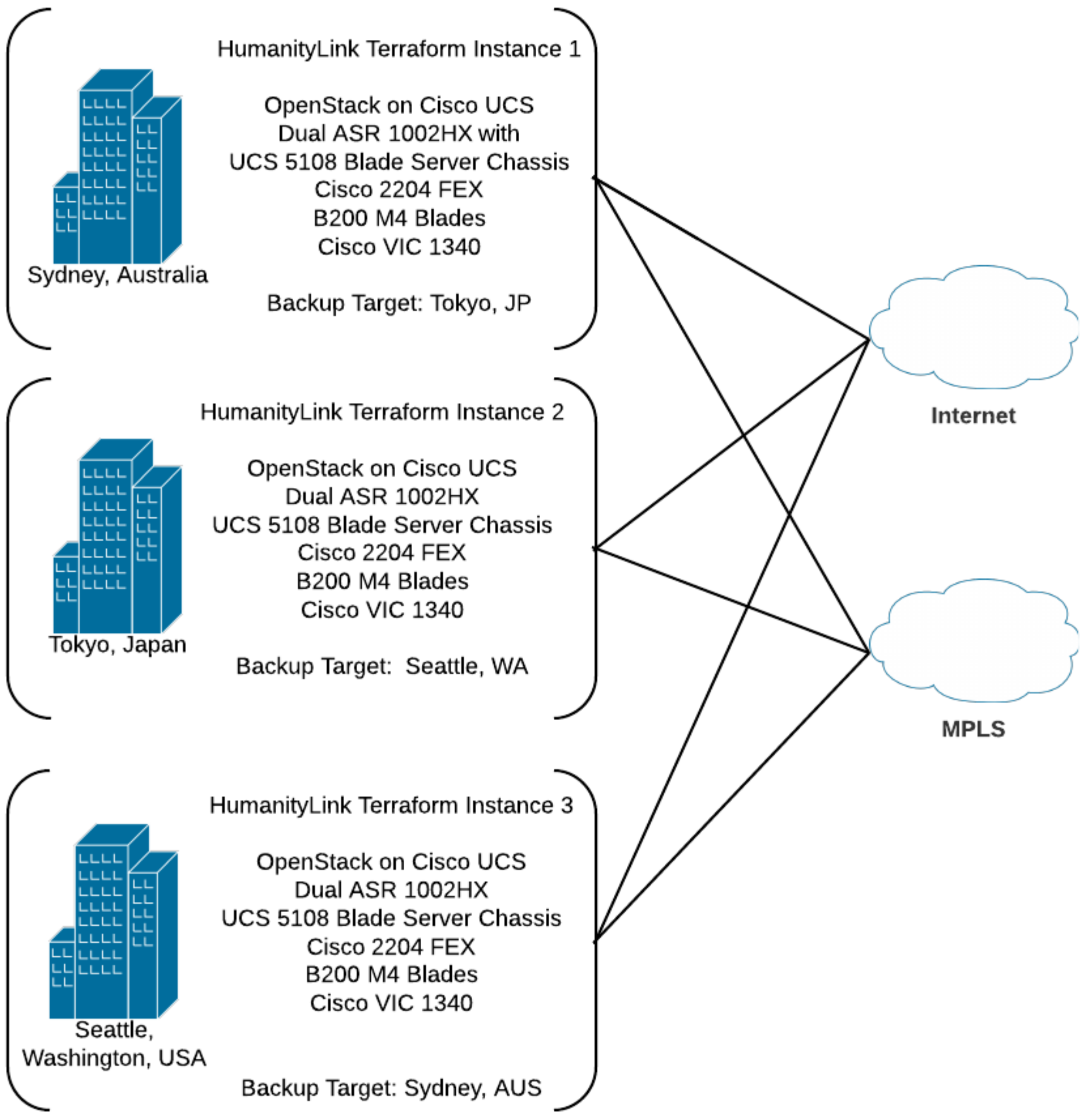
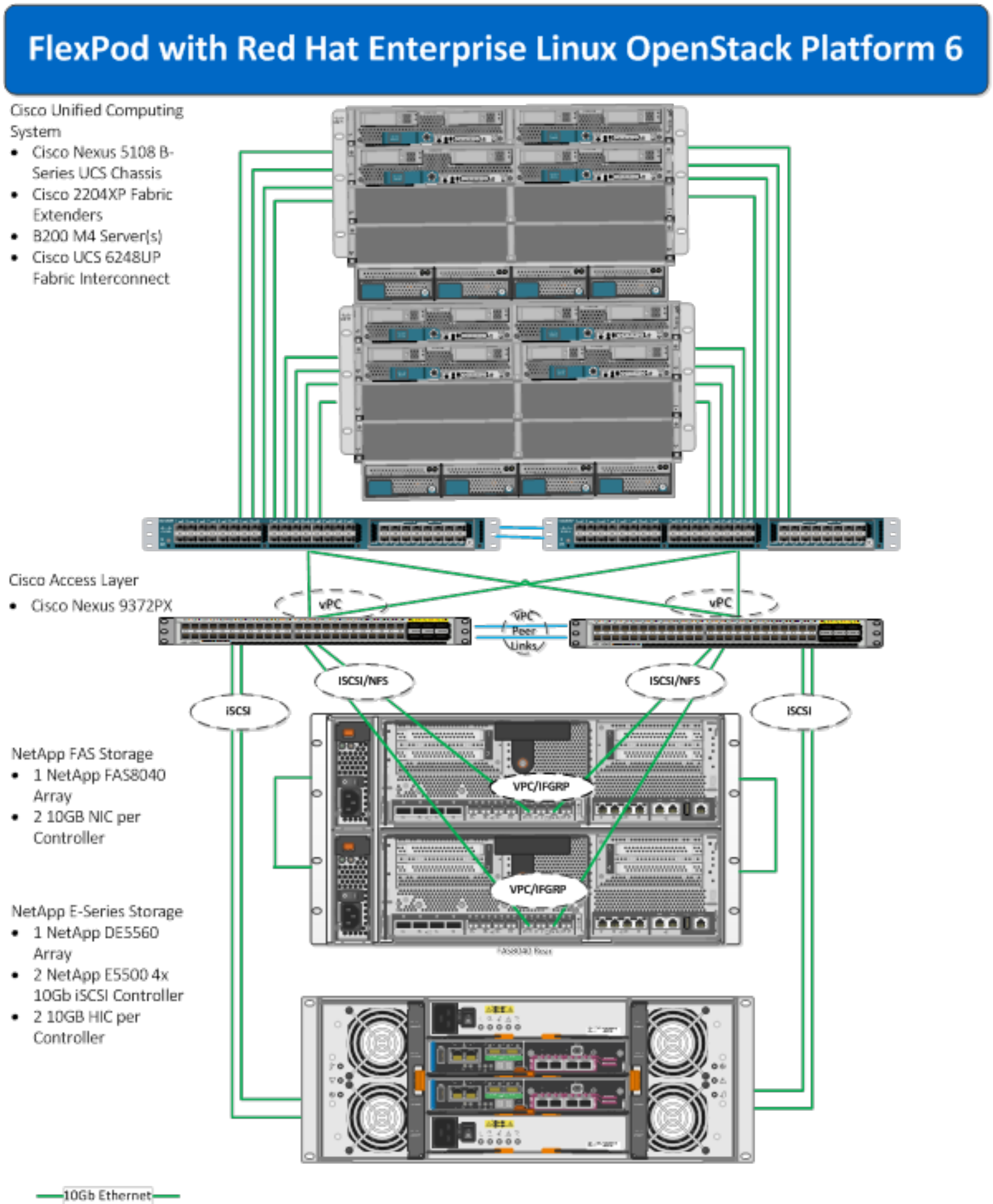


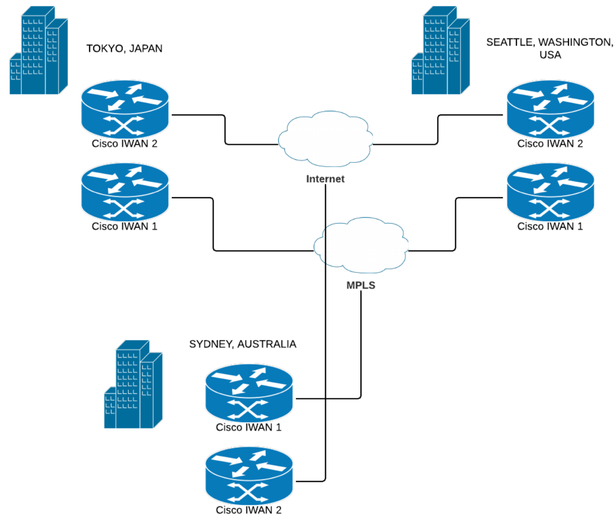
Figure 1

Logical

FlexPod with Red Hat Enterprise Linux OpenStack Platform 6



IWAN Topology Overview



OpenStack Topology

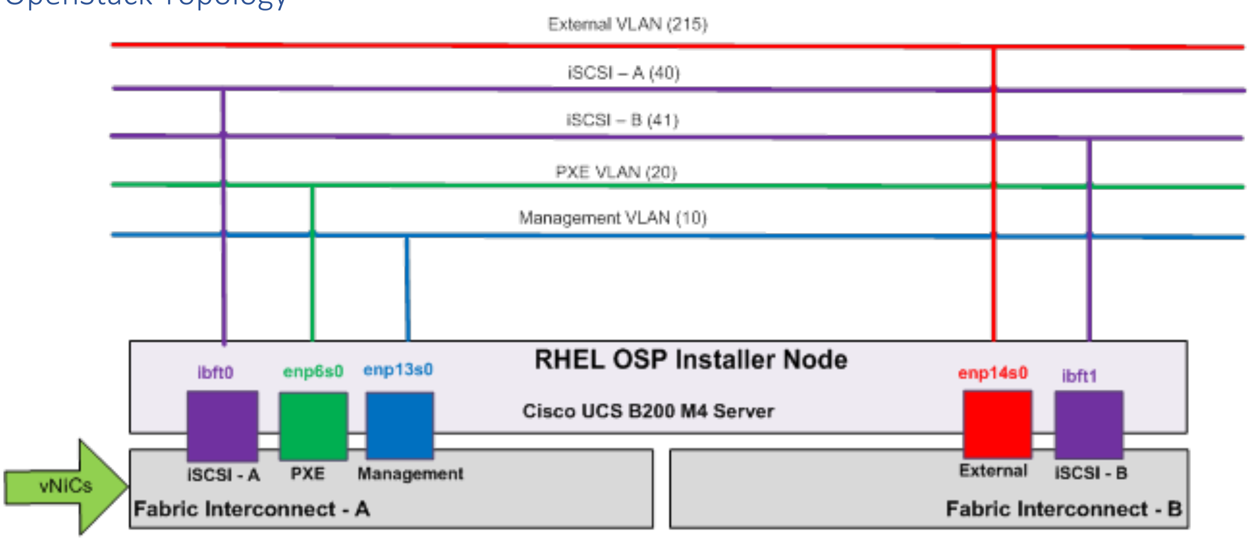


Figure 2 - Platform Installer Node vNIC

- Two iSCSI vNICS are used to provide iSCSI LUN for boot from SAN. They will be providing multiple paths to the boot LUN.
- The PXE vNIC is used for PXE/Provisioning network. This network is used by the Red Hat Enterprise Linux OpenStack Platform Installer to build OpenStack controller and compute hosts. Installer uses this network to boot hosts using PXE, deploy and configure hosts based on their roles (Controller or Compute). The PXE vNIC is mapped to fabric "A" and can dynamically failover to fabric "B" for redundancy.
- The Management vNIC is created for hosts management and also carries OpenStack public API traffic. This is also mapped to fabric "A" and dynamically failover to fabric "B".

- The MCAS vNIC is created to carry OpenStack Management, Cluster Management, Admin API, and Storage Clustering traffic. MCAS vNIC is mapped to fabric “A” and can dynamically failover to fabric “B”, in case any failure occurs on fabric “A” or uplink connectivity of fabric “A”.
- The VM-Traffic vNIC is mapped to Fabric “B” and is configured to trunk provider, tenant, and external VLAN. VM-Traffic vNIC is dynamically failover to fabric “A” in case of failure of fabric “B”.
- The NS–A and NS–B (abbreviation for Network Storage) vNICs are configured to trunk NFS (VLAN 30) and Swift (VLAN 50 and 51) traffic. These vNICs are not configured to failover to the surviving Fabric, instead the host will be configured to treat these two vNICs as bonded interfaces and performs host level interface failover for redundancy as shown in *Figure 3*

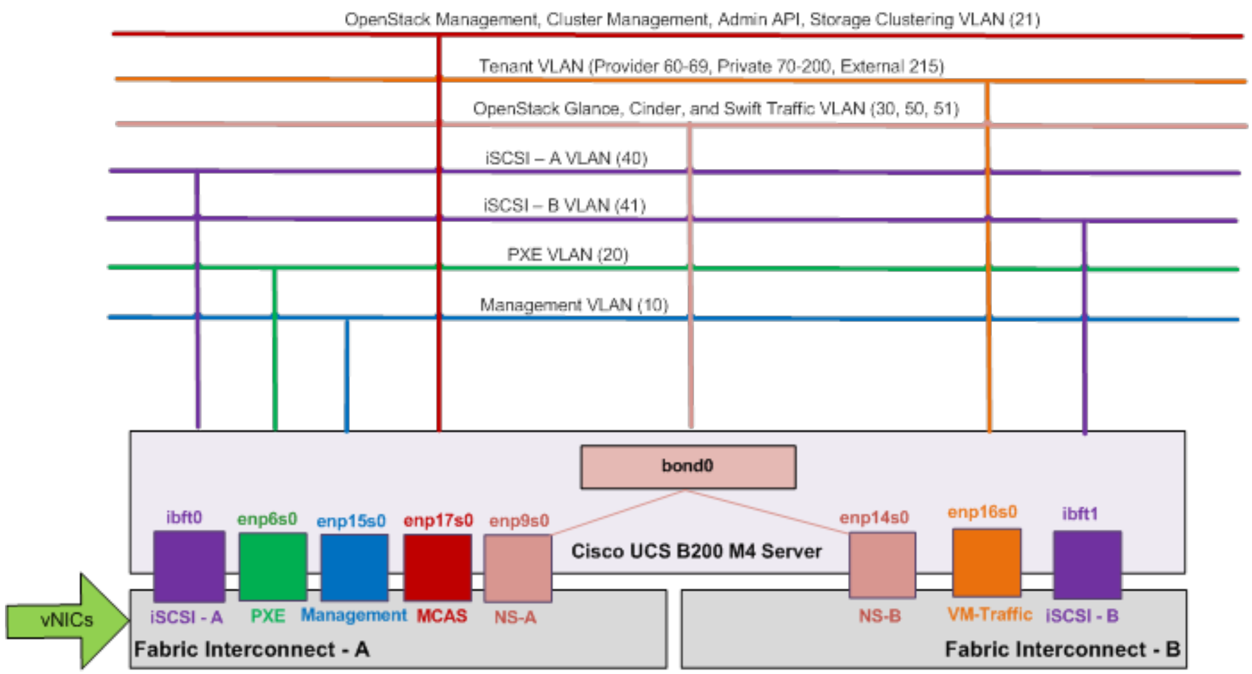


Figure 3 - RHEL OpenStack Platform Controller and Compute Node vNICs

Server pools will be utilized to divide the OpenStack server roles for easy of deployment and scalability. These pools will also decide the placement of server roles within the infrastructure. Two server pools are created *Figure 4*.

- OpenStack Controller server pool
- OpenStack Compute server pool

The Compute server pool will allow quick provisioning of additional compute hosts by adding those servers into the compute server pool, and create service profiles from the compute service profile template. These newly provisioned compute hosts can easily be added into an existing OpenStack deployment though the RHEL-OSP Installer’s web interface.

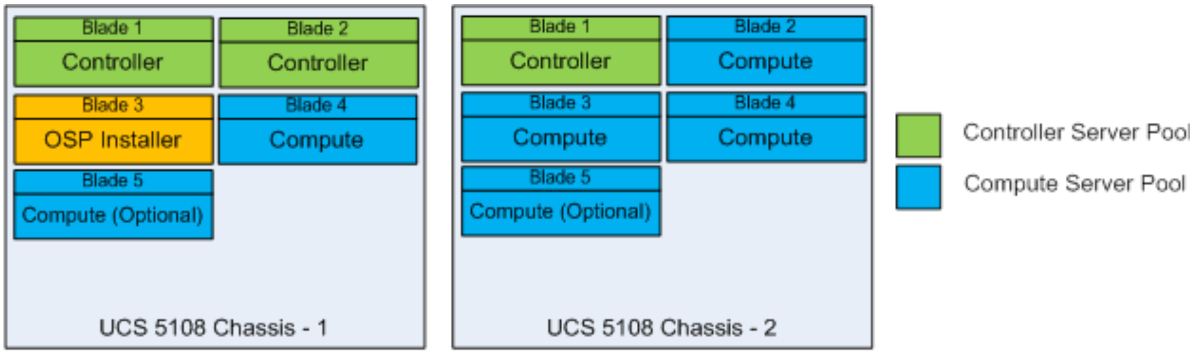


Figure 4 - OpenStack Server Pools/Role Placement

Cisco Unified Computing System

Fabric Interconnects

- Dual Cisco UCS 6200 FIs

Cisco Nexus

- Dual Cisco 9372PX

Cisco UCS 5108 Blade Server Chassis

- Cisco 2204XP Fabric Extenders
- 5 B200 M4 Server(s) – 4 active, 1 cold-standby
    - Dual Xeon E5-2699 v4 – 22c @ 2.20Ghz
    - 512GB RAM
    - Boot from SAN

Cisco VIC 1340

- 2-port 40-Gbps Ethernet

Cisco Nexus 1000v for KVM / OpenStack

NetApp Storage

FAS Storage Family

- 1 NetApp FAS8040 Array – NetApp Cinder backend, Glance image store
- 2 10GB NIC per Controller

NetApp E-Series Storage Family

- 1 NetApp DE5660 Array – OpenStack Swift
- 2 NetApp E5500 x4 10Gb iSCSI Controller
- 2 10GB HIC per Controller

IWAN

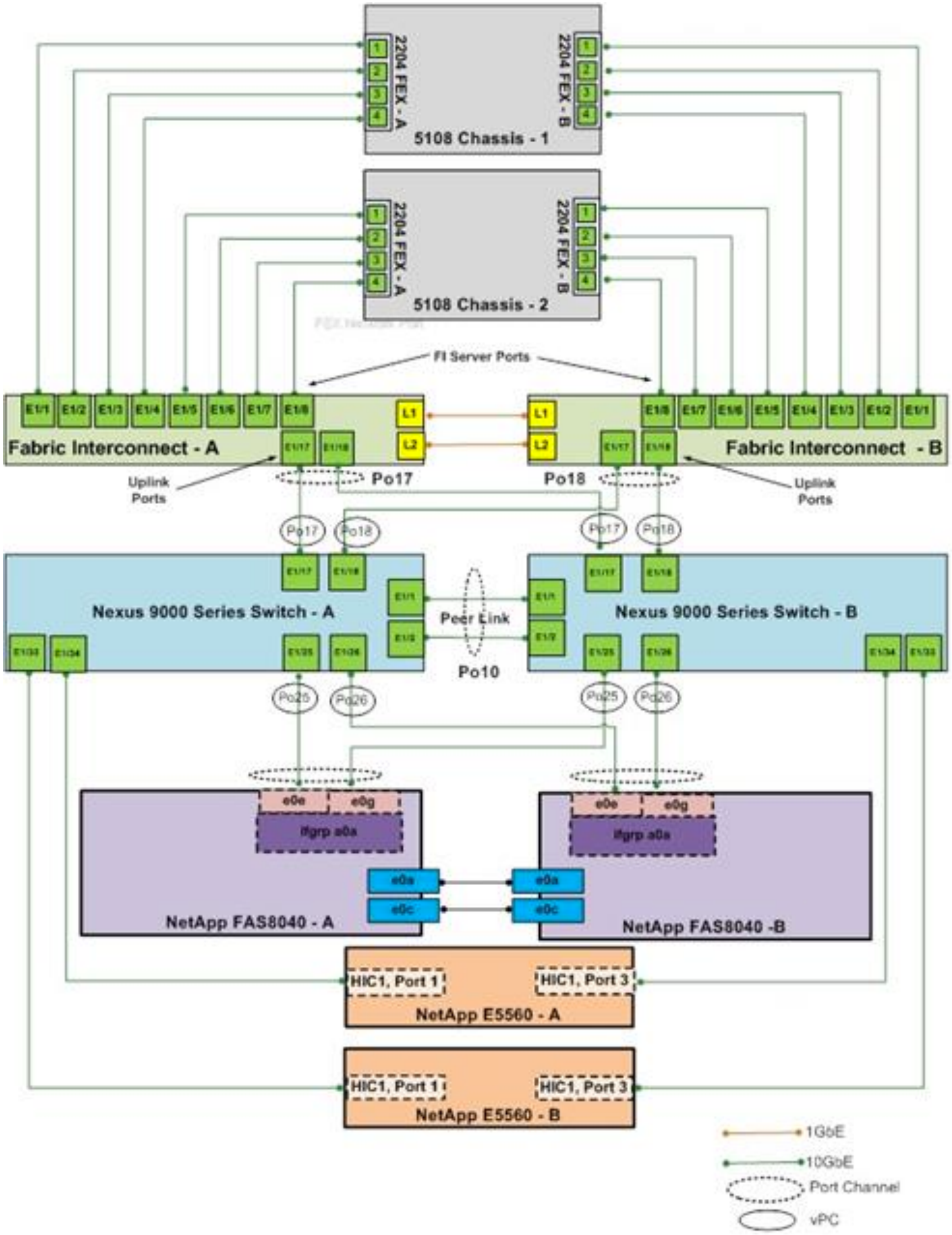
- Dual Cisco ASR 1002-HX

Per-Site Raw Resources

- 387.2Ghz processing | 176 core @ 2.20Ghz
- 2TB RAM
- 150TB Storage
- Up to 8GB/s IPSEC traffic MPLS
- Up to 8GB/s IPSEC traffic Internet

Physical

Cisco's FlexPod Datacenter with RHEL OpenStack Platform 6.0



FlexPod Cabling Detail

Local Device	Local Port	Connection	Remote Device	Remote Port
Cisco Nexus 9372 -Switch A	Eth1/1	10GbE	Cisco Nexus 9372 Series Switch B	Eth1/1
	Eth1/2	10GbE	Cisco Nexus 9372 Series Switch B	Eth1/2
	Eth1/17	10GbE	Cisco UCS Fabric Interconnect A	Eth1/17
	Eth1/18	10GbE	Cisco UCS Fabric Interconnect B	Eth1/17
	Eth1/25	10GbE	NetApp FAS8040 Node A	e0e
	Eth1/26	10GbE	NetApp FAS8040 Node B	e0e
	Eth1/34	10GbE	NetApp E5560 Controller A	Port1
	Eth1/33	10GbE	NetApp E5560 Controller B	Port1
	Eth1/23	1GBE	Management	Port Any
	MGMT0	1GbE	Cisco Catalyst 2960S	Any
Cisco Nexus 9372-Switch B	Eth1/1	10GbE	Cisco Nexus 9372 Series Switch A	Eth1/1
	Eth1/2	10GbE	Cisco Nexus 9372 Series Switch A	Eth1/2
	Eth1/17	10GbE	Cisco UCS Fabric Interconnect A	Eth1/18
	Eth1/18	10GbE	Cisco UCS Fabric Interconnect B	Eth1/18
	Eth1/25	10GbE	FAS8040 Node A	e0g
	Eth1/26	10GbE	FAS8040 Node B	e0g
	Eth1/34	10GbE	E5560 Controller A	Port2
	Eth1/33	10GbE	E5560 Controller B	Port2
	Eth1/23	1GBE	Management	Port Any
	MGMT0	100MbE	Cisco Catalyst 2960S	Any
NetApp FAS8040 Node A	e0P	1GbE	SAS shelves	ACP port
	e0a	10GbE	Cluster Connection to Node B	e0a

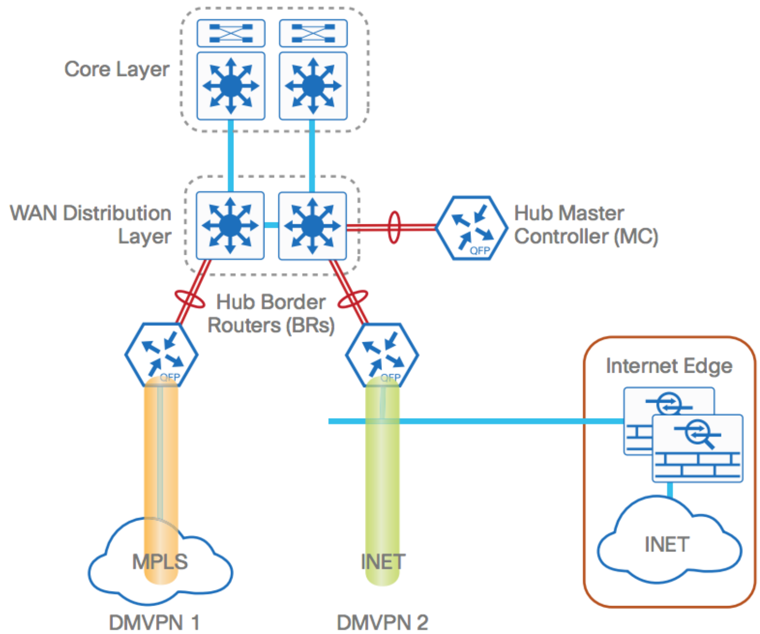
Local Device	Local Port	Connection	Remote Device	Remote Port
	e0c	10GbE	Cluster Connection to Node B	e0c
	e0e	10GbE	Cisco Nexus 9372 Series Switch A	Eth1/25
	e0g	10GbE	Cisco Nexus 9372 Series Switch B	Eth1/25
	e0M	1GbE	Cisco Catalyst 2960S	Any
NetApp FAS8040 Node B	e0P	1GbE	SAS shelves	ACP port
	e0a	10GbE	Cluster Connection to Node A	e0a
	e0c	10GbE	Cluster Connection to Node A	e0c
	e0e	10GbE	Cisco Nexus 9372 Series Switch A	Eth1/26
	e0g	10GbE	Cisco Nexus 9372 Series Switch B	Eth1/26
	e0M	1GbE	Cisco Catalyst 2960S	Any
NetApp E5560 Controller A	Controller A, HIC 1, Port 1	10GbE	Cisco Nexus 9372 Series Switch A	Eth1/34
	Controller A, HIC 1, Port 3	10GbE	Cisco Nexus 9372 Series Switch B	Eth1/34
	1GbE Management Connector 1	1GbE	Cisco Catalyst 2960S	Any
NetApp E5560 Controller B	Controller B, HIC 1, Port 1	10GbE	Cisco Nexus 9372 Series Switch A	Eth1/33
	Controller B, HIC 1, Port 3	10GbE	Cisco Nexus 9372 Series Switch B	Eth1/33
	1GbE Management Connector 1	1GbE	Cisco Catalyst 2960S	Any
Cisco UCS Fabric Interconnect A	Eth1/1	10GbE	Chassis 1 FEX A	port 1
	Eth1/2	10GbE	Chassis 1 FEX A	port 2
	Eth1/3	10GbE	Chassis 1 FEX A	port 3
	Eth1/4	10GbE	Chassis 1 FEX A	port 4
	Eth1/5	10GbE	Chassis 2 FEX A	port 1

Local Device	Local Port	Connection	Remote Device	Remote Port
	Eth1/6	10GbE	Chassis 2 FEX A	port 2
	Eth1/7	10GbE	Chassis 2 FEX A	port 3
	Eth1/8	10GbE	Chassis 2 FEX A	port 4
	Eth1/17	10GbE	Cisco Nexus 9372 A	Eth 1/17
	Eth1/18	10GbE	Cisco Nexus 9372 B	Eth 1/17
	MGMT0	1GbE	Cisco Catalyst 2960S	Any
	L1	1GbE	UCS Fabric Interconnect B	L1
	L2	1GbE	UCS Fabric Interconnect B	L2
Cisco UCS Fabric Interconnect B	Eth1/1	10GbE	Chassis 1 FEX B	port 1
	Eth1/2	10GbE	Chassis 1 FEX B	port 2
	Eth1/3	10GbE	Chassis 1 FEX B	port 3
	Eth1/4	10GbE	Chassis 1 FEX B	port 4
	Eth1/5	10GbE	Chassis 2 FEX B	port 1
	Eth1/6	10GbE	Chassis 2 FEX B	port 2
	Eth1/7	10GbE	Chassis 2 FEX B	port 3
	Eth1/8	10GbE	Chassis 2 FEX B	port 4
	Eth1/17	10GbE	Cisco Nexus 9372 A	Eth 1/18
	Eth1/18	10GbE	Cisco Nexus 9372 B	Eth 1/18
	MGMT0	1GbE	Cisco Catalyst 2960S	Any
	L1	1GbE	UCS Fabric Interconnect B	L1
	L2	1GbE	UCS Fabric Interconnect B	L2

VLAN Configuration Detail

VLAN Name	Variable	VLAN Purpose	VLAN ID or VLAN Range
Management	<<var_mgmt_vlan_id>>	VLAN for in-band management network. Also used for OpenStack Public API traffic	10
PXE	<<var_pxe_vlan_id>>	Provisioning network used by the RHEL-OSP Installer server for deploying RHEL 7.1 and OpenStack Platform. This network is also used for OpenStack management traffic.	20
NFS	<<var_nfs_vlan_id>>	Storage network for carrying Cinder and Glance traffic	30
iSCSI-40	<<var_iscsi_A_vlan_id>>	VLAN for iSCSI traffic for boot from SAN (Fabric A)	40
iSCSI-41	<<var_iscsi_B_vlan_id>>	VLAN for iSCSI traffic for boot from SAN (Fabric B)	41
Swift-50	<<var_swift_A_vlan_id>>	Storage VLAN for Swift traffic (Fabric A)	50
Swift-51	<<var_swift_B_vlan_id>>	Storage VLAN for Swift traffic (Fabric B)	51
Provider	<<var_provider_vlan_range>>	Tenant provider VLANs	60-69
Tenant	<<var_tenant_vlan_range>>	Tenant private networks for VM data traffic	70-200
External	<<var_external_vlan_id>>	VLAN for public network. Provide access to outside world for deploying OpenStack platform.	215
MCAS-21	<<var_mcas_vlan_id>>	VLAN for OpenStack Management, Cluster Management, Admin API, and Storage Clustering traffic.	21

Per-Site IWAN



Backup/Disaster Recovery

OpenStack's now native data-protection-as-a-service "Raksha" will be utilized for the creation and management of automatic backup policies for workloads, consistent snaps of resources, and space-efficient data streams *(see Table 5)*.

- VM(s) centric data protection service.
- Application consistent backups
- Dedupe/change block tracking at the source for efficient backups
- Point-In-Time backup copies
- A job scheduler for periodic backups
- Noninvasive backup solution that does not require service interruption during backup window

Cisco device configurations will be saved to an SFTP server, per-site, using Cisco's Auto Archive feature. Cisco UCS will utilize the scheduled full backup feature to a safe location in a similar manner. These will be backed up utilizing Raksha, with locations based on **Table 5**.

Security Considerations

Datacenter Infrastructure

- Role-Based Access Control (RBAC) credentials will be deployed on an individual basis for users in the organization through the Cisco UCS, and OpenStack environment, with a documented change-control system [**A09**]
- CHAP will be utilized whenever possible for connecting to iSCSI targets
- SSH2/AES will be utilized where possible
- Plaintext communications (telnet) access will be disabled

Inter-site Infrastructure (IWAN)

- IPsec encryption with pre-shared keys

Internet Security

- ISR 1002 HX with SHA (Security + HA) bundles

Antivirus

- Trend Micro ServerProtect (Linux)
- Trend Micro OfficeScan (Windows)

References

Cisco. (2010, December 27). *Zone-Based Policy Firewall Design and Application Guide* . Retrieved July 2017, from Cisco: http://www.cisco.com/c/en/us/support/docs/security/ios-firewall/98628-zone-design-guide.html

Cisco. (2015, July 31). *FlexPod Datacenter with Red Hat Enterprise Linux OpenStack Platform Design Guide* . Retrieved 2017, from Cisco, Inc: http://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/UCS_CVDs/flexpod_openstack_osp6_design.html

Cisco. (2016, February 11). *FlexPod Datacenter with Red Hat Enterprise Linux OpenStack Platform* . Retrieved July 2017, from FlexPod Datacenter with Red Hat Enterprise Linux OpenStack Platform : http://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/UCS_CVDs/flexpod_openstack_osp6.html

Cisco. (2017, March). *Intelligent WAN Design Summary.* Retrieved from Cisco: http://www.cisco.com/c/dam/en/us/td/docs/solutions/CVD/Mar2017/CVD-IWANDesign-MAR2017.pdf

OpenStack. (n.d.). *OpenStack Data Protection As Service ("Raksha")*. Retrieved July 2017, from OpenStack Wiki: https://wiki.openstack.org/wiki/Raksha#OpenStack_Data_Protection_As_Service.28.22Raksha.22.29