Bachelor’s thesis

Degree in Information Technology

2019

Alexander Enberg

Osaiasi Katiloka Foleti

CREATION OF A PRIVATE CLOUD INFRASTRUCTURE

Building a Foundation for Cloud Services

BACHELOR’S THESIS | ABSTRACT

TURKU UNIVERSITY OF APPLIED SCIENCES

Degree in Information Technology

2019 | 66

Alexander Enberg, Osaiasi Foleti

CREATION OF A PRIVATE CLOUD INFRASTRUCTURE

Building a Foundation for Cloud Services

This thesis focuses on our journey to create a functioning hardware-based infrastructure for a cloud environment. The project was conceived by Alexander Enberg, Osaiasi Foleti and approved by the Turku University of Applied Sciences' (TUAS) Internet Technology lab directors and instructors after proposing a new direction derived from the original project scope. The current structure consists of proprietary software for management-plane utilization and open-source software as a means of cost-effective approaches to cloud creation. The hardware is provided for by the TUAS Internet Technology lab, allowing us to match the hardware requirements stipulated by the software we are using.

The aim of this project is to investigate both open-source and proprietary solutions in cloud infrastructure creation, delineating the creation methodology and the scalability of the platform. The objective thus is to develop a functional cloud infrastructure that would be serviceable in a production environment. This thesis should cover the topics of installation and maintenance of the cloud infrastructure and explore methods to scale growing operations in small/home office environments.

This thesis was written by two students: Alexander Enberg and Osaiasi Foleti. Alexander was in charge of the software that kept the cloud environment running. He wrote the associated software text and most of the literature heavy parts found in this thesis. Osaiasi oversaw the hardware side of the cloud environment and the associated description and text also found in this thesis.

The outcome of this project is to have an operational cloud infrastructure as a proof-of-concept model to show our knowledge. This can be used as a basis for other cloud projects for TUAS to utilize should the project show potential for further development if local hosting of services is necessary.

KEYWORDS:

cloud, engineering, Cisco, HP, SAN, VMware

CONTENT

APPENDICES

Appendix 1: Docker Installation
Appendix 2: Pfsense system information
Appendix 3: Differences between the vSphere licenses

LIST OF ABBREVIATIONS

AMD-V / VT-X	Advanced Micro Dynamics Virtualization/Virtualization Technology Intel
API	Application Programming Interface
AWS	Amazon Web Service
BGP	Border Gateway Protocol
BIOS	Basic Input/output Operating System
CC	Cluster Controller
CI	Converged Infrastructure
CLC	Cloud Controller
CPU	Central Processing Unit
DHCP	Dynamic Host Configuration Protocol
DVMRP	Distance Vector Multicast Routing Protocol
DRS(VMware)	Distributed Resource Scheduler is a utility that balances computing workloads with available resources in a virtualized environment
EBS	Elastic Block Storage
EC2	Elastic Computing Cloud
EIGRP	Enhanced Interior Gateway Routing Protocol
EMI	EUCALYPTUS Machine Image(s)
EUCALYPTUS	Elastic Utility Computing Architecture for Linking Your Programs to Useful Systems
EsXI	VMware EsXi is a bare metal hypervisor that installs easily on to your server and partitions it into multiple virtual machines
FT	Fault Tolerance is a property that enables a system to continue operating properly in the event of the failure of some of its components
FC	Fibre-Channel
GB	Gigabyte
Gb	Gigabit
GbE	Gigabit Ethernet

GBIC	Gigabit interface converter
HA	High Availability, an agreed level of operational performance for a higher than normal period.
HPE	Hewlett Packard Enterprise
HSRP	Hot Standby Router Protocol
IAAS	Infrastructure as a Service
IB	InfiniBand
IEEE	Institute of Electrical and Electronics Engineers
IGMP	Internet Group Management Protocol
IGRP	Interior Gateway Routing Protocol
I/O	Input/output
ISCSI	Internet Small Computer System Interface
ISL	Inter-Switch Link
ISO	An ISO image is a disk image of an optical disc. It is an archive file that contains everything that would be written to an optical disc, sector by sector, including the optical disc file system.
KVM	Kernel-based Virtual Machine
LACP	Link Aggregation Control Protocol
MSA	Modular SAN Array
NFS	Network File System, protocol
NIST	National Institute of Standards and Technology
NAT	Network Address Translation
OS	Operating System
OSPF	Open Shortest Path First
PAAS	Platform as a Service
PII	Personally Identifiable Information
PIM-DM	Protocol-Independent Multicast - Dense Mode
PIM-SM	Protocol-Independent Multicast - Sparse Mode
RIP	Routing Information Protocol
RMON	Remote Network Monitoring

SAAS	Software as a Service
SAN	Storage Area Network
SCSI	Small Computer System Interface
SFP	Small form-factor pluggable
SNMP	Simple Network Management Protocol

1 INTRODUCTION

"If you have built castles in the air, your work need not be lost; that is where they should be. Now put the foundations under them."

Henry David Thoreau, Walden

"Cloud computing, the long-held dream of computing as a utility, has the potential to transform a large part of the IT industry, making the software even more attractive as a service and shaping the way IT hardware is designed and purchased." (Armbrust et al., 2018)

"Natural clouds are expanding, contracting and definitely elastic based on the number of winds hoisting them." (Salam et al, 2015) The virtual cloud is similarly shrinking, expanding and elastic based on resources and their usage, cloud tenants' resources and their demands. Physical resources (networking, storage, computing, etc.) inside or across data centers do not change so rapidly. The elasticity is something the cloud handles which is built in with the help of software, not the hardware. The ever so classic promise of the cloud is to make computing resources available on demand. This means, in theory, that a cloud needs to be scalable when it boils down to business growing/shrinking. (Salam et al, 2015)

2 OBJECTIVES

The goal of this thesis is to create a functioning hardware-based infrastructure for a cloud environment, covering both software-and infrastructure-wise and its appurtenant equipment.

All the work was done inside Turku University of Applied Sciences premises. We had the utmost privilege of using all the machinery we needed, such as networking hardware and specialized server enclosures. In the end, we still could have used much more than we did. During the winter of 2018 from January till the end of May 2018, we worked on creating the infrastructure and the configurations needed for the accommodated software, VMware being a large part of the development including open-source software such as Ubuntu Server operating systems and Docker container provisioners. The end result was a fully-functional infrastructure for a cloud, and its use can be used as a foundation for a localized network within the Turku University of Applied Sciences. We wanted to explore the many possibilities of creating this environment, apropos our cloud's potential use and how we would further expand.

3 Cloud Models

We chose the private model of deployment for our cloud environment. There are currently four deployment models that are recognized: public, private, hybrid and community.

3.1 Public Clouds

The public cloud model is owned by the cloud provider and end user has hardware and software made available for them. As far as the terminology goes, a private cloud is both a 'pay-as-you-go' and 'on-demand' model.

In an 'on-demand' model end-users can use software, storage, and other resources instantly and in many a case without limits. Usually, this process of resource addition is performed in live environments through a transition process that does not affect current operations. Examples of large public clouds are Google apps, Microsoft Azure and Amazon EC2.

3.2 Private Clouds

The private cloud model infrastructure is owned by one organization and is not shared by other companies either. To narrow down private cloud even further, there are two subcategories of a privately-owned cloud: externally hosted private cloud and on-premise cloud.

The externally hosted private cloud gives the companies the opportunity to buy exclusive externally hosted private clouds (Figure 1). The features of an external cloud are that a cloud provider can give exclusive access to a company and in this way, the expenses are not placed solely on the company that buys the cloud service. The cloud provider maintains security and IT infrastructure. The provider must also guarantee confidentiality, privacy, and exclusiveness of the cloud that is offered.

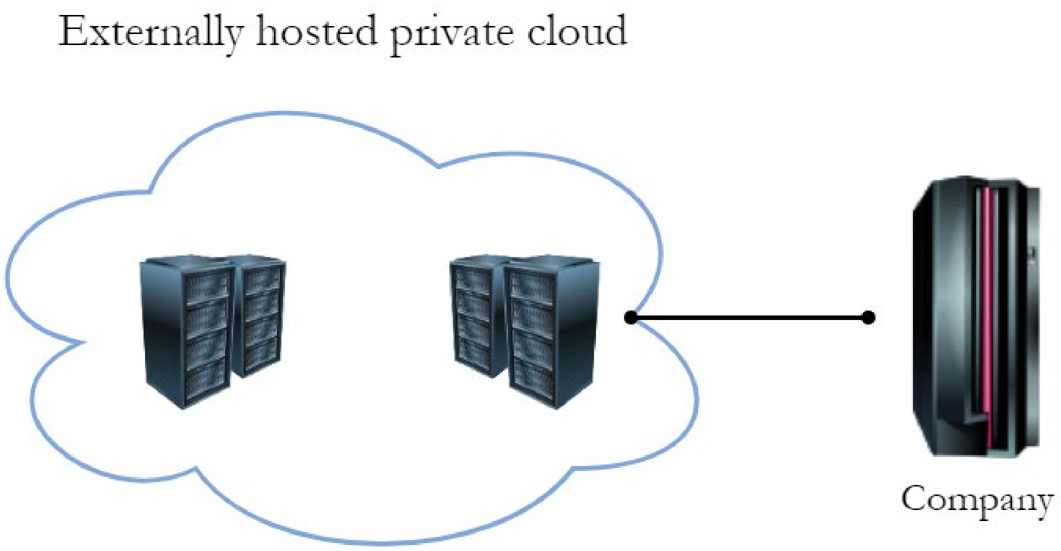


Figure 1. Externally Hosted Private Cloud Model.

The on-premise private cloud is also known as an internal cloud since the cloud is hosted inside an organization's data center (Figure 2). The cost of such a model is much higher than any of the other since the company's building and IT infrastructure need to be built and maintained accordingly.

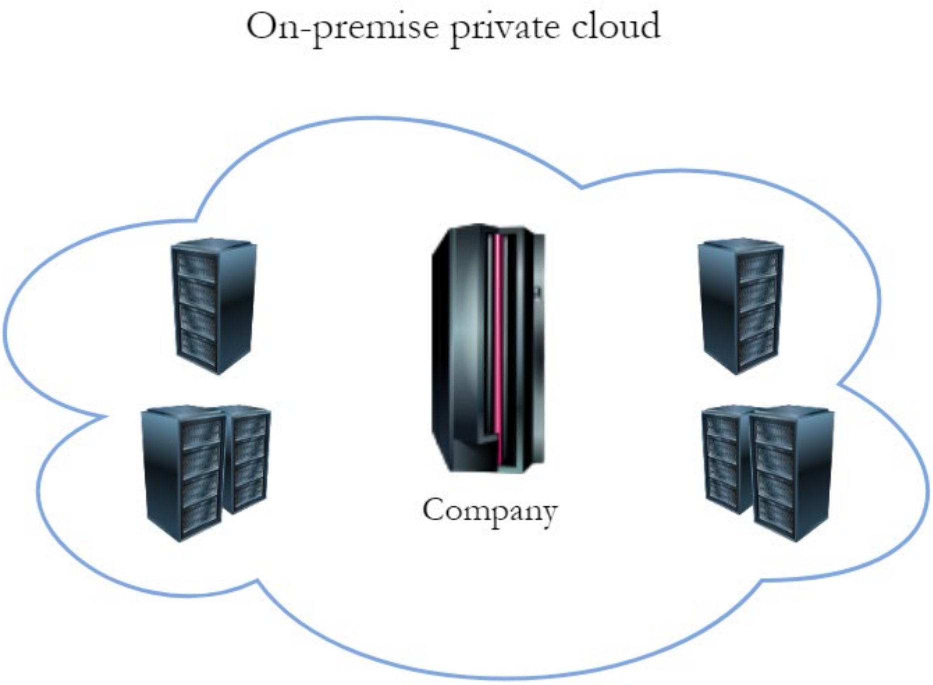


Figure 2. On-premise Private Cloud Model

3.3 Hybrid Clouds

The hybrid cloud model is a combination of the benefits of the private cloud and the public cloud (Figure 3). It uses the resources of a public cloud to expand the capacity of a private cloud as needed. If a company would like to use a hybrid cloud, e.g., to host a website, the private cloud resources are used for normal usage and during peak time hours, they take on the public cloud.

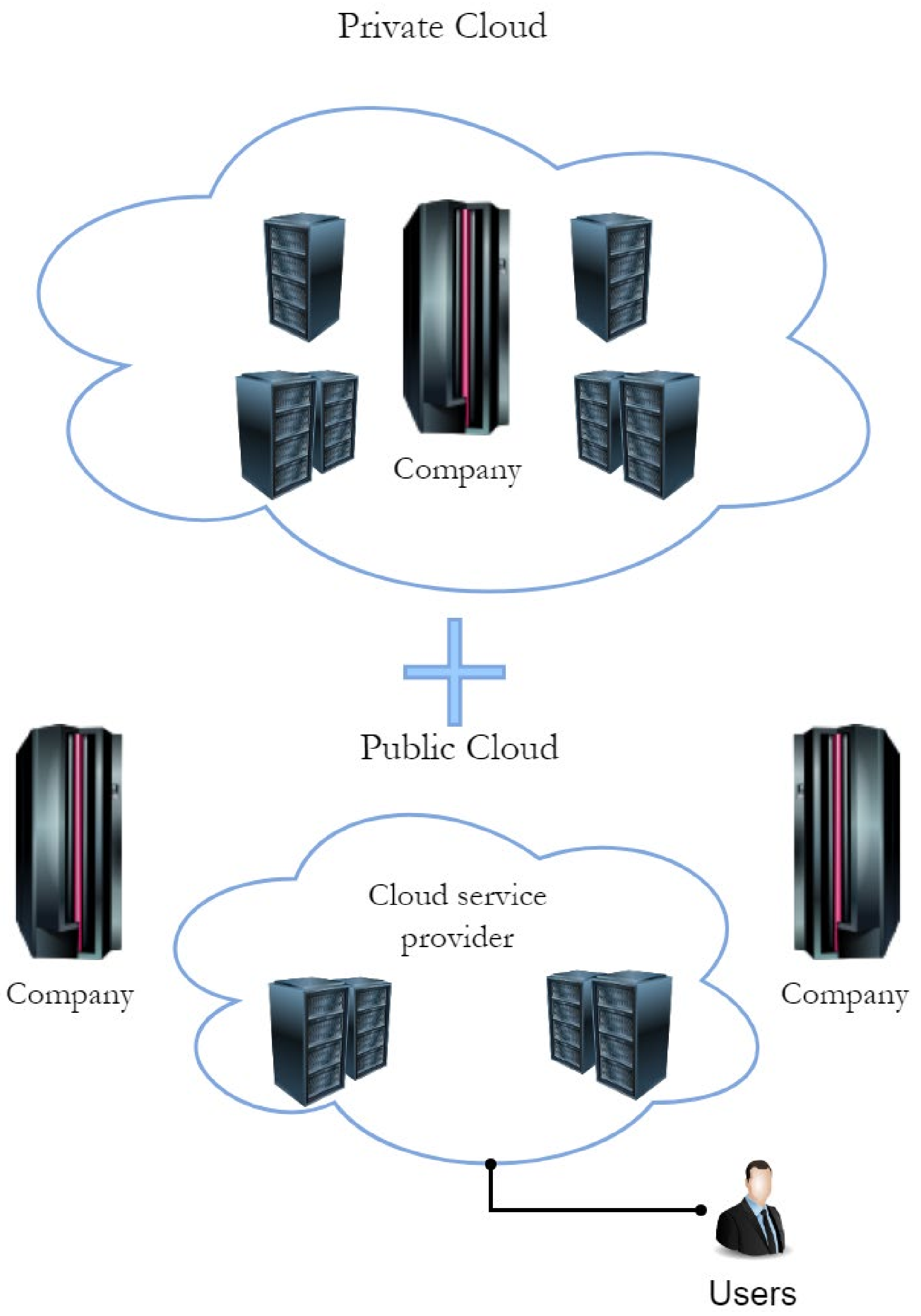


Figure 3. Hybrid Cloud Model

The community cloud model is shared by groups with similar computing concerns. A place where the community cloud can be used is among various agencies within the state government that would operate under similar guidelines. A shared community cloud can be used in this instance(Figure 4). The price of a community cloud model is cheaper than that of a private cloud model since the cost for the community cloud model is shared by community users.

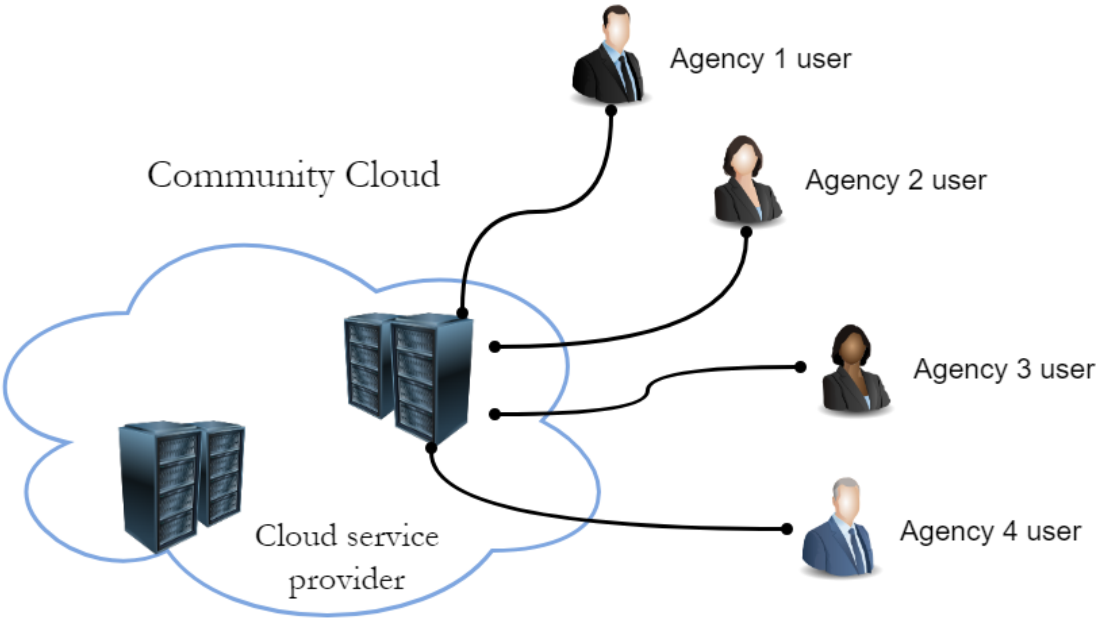


Figure 4. Community Cloud Model

Our specific model came to be ‘on-premise cloud'. Our cloud model of choice was easy to figure out since we hosted it on the Turku University of Applied Sciences premises. The sub-category became clear as well since both space, infrastructure, hardware, and software was given to us by Turku University of Applied Sciences. The security in our choice is one of the best since it is fortified in-house and not needed to rely on a second- or third-party provider. (Fox et al, 2018, pp.478-480)

4 IDENTIFICATION AND SECURITY IN THE CLOUD

Data sharing is one of the most significant applications of cloud computing. For security and privacy concerns, clients generally encrypt their data before uploading them to the cloud and also providing their identification. A system that is hosted in the cloud a user is considered being a "remote" user. Both privileged and normal users have to give their identity (online/digital) to authenticate themselves to a system. This is most often completed over a public medium or otherwise known as the internet (public cloud). It can also be executed within a shared community (community cloud) or a more private one such as a company owned (private cloud). The authentication is then established once the identity of the authenticated user is assured with a certain level of confidence. The system is then able to provide user access to the features and functions that the specific identity of the user is authorized to.

Today we cannot get far without submitting our personally identifiable information (PII) when trying to establish some sort of connection to obtain certain types of services.

A good example is when we try to obtain services from ‘Kanta.fi' where we need to submit our PII which in this case is our bank information. From here we can see what prescription we have at our disposal and what prescription has expired. If the drugs have expired, we can send an inquiry to the doctor that then renews them. When services like this are provided through the cloud, there are both benefits but also drawbacks. Examples of drawbacks would be that personal data is more easily shared amongst different hospitals and pharmacies. Although this is a drawback this also ensures that the data (when needed) is given quickly. Other drawbacks would be loss of control over one's personal data, the accuracy of said data, the aggregation of the data, what is the data used for and who is using your data. Most cloud systems have parties of multiple administrations and the provisioning of the services makes the privacy controls of cloud systems more challenging. (Vacca, 2016, pp.169-171)

4.1 Identification in the Cloud

There are quite a few methods of authentication that are used in the cloud system. The most known authentication methods are:

- Fingerprints or facial image
- Badges or keys
- Passphrases

Both single-factor and multi-factor authentication are at one's disposal. The single-factor uses one of the above-mentioned ways of authentication whereas multi-factor authentication can use all from two to three. Recently there have surfaced technologies that enable the usage of nonstandard ways to give the assurance of authentication. Using one or more of the classic factors, reliable global positioning data or network address of a computer can serve to further strengthen one's authentication. The most well-known method of authentication is of course with a username and password combination. A username is established by the user and a shared secret password that is only known only to the user and the system. Every time a user handles the same system again, he or she proves their identity by providing the correct username and password. Something that is fairly usual today is the need for a password change after a certain amount of time. Sometimes it is forced upon the user and sometimes it is only notified.

Our cloud infrastructure has a multitude of different logins. We stored all of our passwords in KeepassX, which had its own master password that needed to be entered before gaining access to all of our machinery/software passwords. We only dealt with single-factor passwords in our infrastructure. Our plethora of software (VMware ESXi, VMware vCenter) and infrastructure (switch, HP-blade, SAN) were all accessed via single-factor authentication. (Vacca, 2016, pp.169-171)

4.2 Security in the Cloud

Cloud security is the pillar of where a safe and trustworthy network can stand. The best ways to accomplish a safe environment is to use strategies, policies, processes, best practices, and technology.

Cloud security strategies exist to enable cloud resources for their best use cases, while effectively managing risk. A company, from small to large, should account for the many different cloud security strategies that exist. Some of the strategies might be:

- Accounting for already existing IT security practices
- Being aware of potential security risks
- Understanding the organization's current and/or future cloud computing needs
- The distance/gap between the current cloud security and the desired end goal
- Eventual solutions to said distance/gap (Miller, 2018)

To help a company/organization to achieve safety in the cloud there are at least four important practices a company/organization can use.

It is recommended to perform some form of due diligence when it comes to cloud technology. A cloud consumer must have full knowledge of their network to be able to provide resilience, functionality, and security for a cloud system or application. The performing of due diligence must be done over the lifecycle of all applications and systems that are deployed in the cloud. This includes planning, development, deployment, operations and decommissioning.

- To be able to have a successful cloud deployment one must plan. The planning can be everything from selecting an appropriate system or application to move to, build in or buy from a cloud service provider. Planning also includes knowledge over the systems in-house (if the cloud is deployed in one's own company) or if the cloud is hosted elsewhere, knowledge of that system is also needed. People/staff that are involved in the development of the cloud should also be trained accordingly.
- During the development and deployment phase, the team involved with the cloud should gain/have access to details regarding correct cloud application usage.
- Since cloud computing is based on the delivery of services that often resemble existing applications, hardware, and networks. It is important and critical to secure. When developing and deploying it is important that one's organization/company reviews its security policies and current security control implementation approaches. (Faatz, 2018)

- Cloud security policies will support your overall cloud computing security strategy. Said policies should clearly explain the necessary needs to keep an online environment safe. The following are different policies of cloud security.
- **Scope** - The cloud environment and its specific environment and services.
- **Breach** - If there is a breach of security or policy, what should happen.
- **Deployment** - How cloud security will be maintained from a high-level view.
- **Compliance** - When trying to match regulatory requirements for end users, business and federal.
- **Accountability** - Ensured safe cloud computing environment/area by people with expertise.
- **Access and Identity Management** - Control over who has access to certain areas or specific information and how authentication and authorization are handled.
- **Sensitivity and Confidentiality** - Objectively analyzing confidential data sets, applications and other cloud elements.
- **Acceptable Use** - Expected standards that end users, developers and other authorized users abide by.
- **Best praxis** for Strong Cloud Security Implementation
- **Password Control** - Sharing of passwords should never be allowed. Keep passwords under lock and key in an e.g. password manager that has either a master password or a two-step verification for logging in. In this password manager, the different passwords are usually used for accessing vital parts of the system.
- **Activity Monitoring on Users** - Tracking the activity of users on your cloud system.
- **Vulnerability Scanning** - Regularly scanning for vulnerabilities and privilege-related risks. Penetration testing for determining security resilience in the real world.
- **Maintenance and Patching** - If your cloud is hosted elsewhere rather than locally, ensure that the cloud vendor has reliable patching for known vulnerabilities.
- **Network Segmentation** - Assessment of a multitenant environment must be done to secure them better. Separation of different working environments into zones which have different rights will secure the cloud. The zone that users come in contact with

first should not be able to reach certain restricted zones. This is also done to prevent hacking attempts or other breaches of security.

- **The usage of Access Control List Management** - A robust access management policy should be implemented into the system. There are usually different roles/permissions. From lowest to highest. These should be assigned to correct users to prevent wrongful usage of the systems.
- **Monitoring** - Continuous monitoring of the whole environment must be maintained.
- **Disaster Recovery** - If your cloud is hosted elsewhere rather than locally, one must know of their reporting, retention and recovery policies on backups.
- **Encryption** - The whole cloud should and must be encrypted, even at rest and in transit. (Miller, 2018)

4.3 Privacy in the Cloud

Privacy Is the ability of the individual or the group to differ from each other or information about themselves and thereby selectively reveal them. Privacy consists of three elements.

- **When:** a subject may be more concerned about revealing current or future information than previous data.
- **How:** A user may be comfortable if his/her friends can manually request their information, but the user may not like to send alerts frequently and/or automatically.
- **Scope:** A user prefers to report his information as an ambiguous region rather than an exact point.

When users are viewing sensitive data inside a cloud the cloud service can protect the users' privacy. This is done by preventing an attacker from ascertaining any of the user's behavior in the service (no direct data leakage). Oblivious RAM, or ‘ORAM' for short, is something researchers have focused on. This specific technology visits several copies of the user’s data to hide the users real visiting targets(pages). ORAM is a promising technology even in clouds but is already widely used in software protection.

There are four subcategories of different privacy issues depending on the different cloud scenarios that can be found.

- How users could be enabled to have some sort of control over their data and to be able to know when the data are stored and processed in the cloud. Also, to battle nefarious use, avoiding theft and unauthorized resale.
- Which assembly is responsible for assuring legal requirements for personal information?
- How involved cloud subcontractors are in processing which can be properly ascertained, checked and identified.
- Guarantee user data reproductions to multiple locations as a usual choice avoiding data loss, leakage and or unauthorized modification or fabrication. (Sun et al, 2014)

Cloud services can be classified into three categories: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS).

- IaaS gives the capacity for customers to rent hardware such as storage, CPU, and network. Customers can run their selected OS and applications on their rented hardware components. Customers pay for the hardware components' usage such as storage, CPU, and network. An example of an IaaS is Amazon EC2.
- PaaS gives customers the capacity to use cloud-provided programming equipment for application development and deploying them on a PaaS platform. Customers have no control of the hardware that lies underneath, and they pay only for the software components. PaaS components are e.g. OS, databases, Microsoft Azure and Google App Engine.
- SaaS gives the possibility for applications such as office software and emails to be offered as a service through a cloud provider. This gives customers the capacity to access these services through web browsers. The provider of the cloud manages and hosts the underlying hardware and software. The way the SaaS usually works is that a customer usually pays a monthly subscription fee. Google Docs and Microsoft outlook 365 are both a product of SaaS. (Fox et al, 2018, pp.477-478)

5 HARDWARE

The equipment was provided by the Turku University of Applied Sciences, and its use was proctored by the staff of the Internet Technology Lab. The equipment ranged from routers, switches, servers and the such for use in this project.

5.1 Hewlett-Packard Enterprise

Hewlett Packard Enterprise Company (HPE) is an American information technology company based in Palo Alto, California, founded in 2015 due to the splitting of the Hewlett-Packard company. HPE is a business-focused organization with two divisions: Enterprise Group and Financial Services. Of the two, Enterprise Group focuses on servers, storage, networking, as well as consulting and support of those products.

The divide was designed so that old Hewlett-Packard changed its name to HP Inc. and created a new company: Hewlett Packard Enterprise. This way, HP Inc. kept the old HP's personal computer and printing business. In 2017, HPE split off its Enterprise Services business and merged it with Computer Sciences Corporation, becoming DXC Technology. It also did the same with its software business and merged it with Micro Focus.

5.1.1 HPE BladeSystem c7000 Enclosure

The product from this enterprise that we will be focusing on is the HPE BladeSystem c7000 Enclosure (Figure 5). The enclosure provides us with all the power, cooling, storage and I/O infrastructure needed for the project. This allows us to simultaneously run 4 redundant interconnect I/O fabric connections (SAS, ISCSI, InfiniBand, Fibre Channel, Ethernet, etc.). The main purpose of the BladeSystem is to host all our virtual machines (VMs), including VMware/vSphere and vCenter and provide direct communications throughout the entire network. To the BladeSystem, a SAN storage unit can also be connected for more storage capacity.

The HPE BladeSystem c7000 Enclosure can hold up to 16 half-height server blades, 8 full-height server blades and/or 8 expansion blades (but cannot exceed a total of 16 blades). (HPE, 2018)

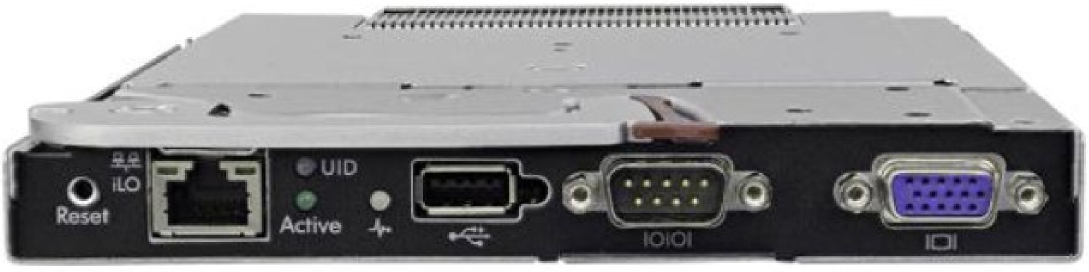
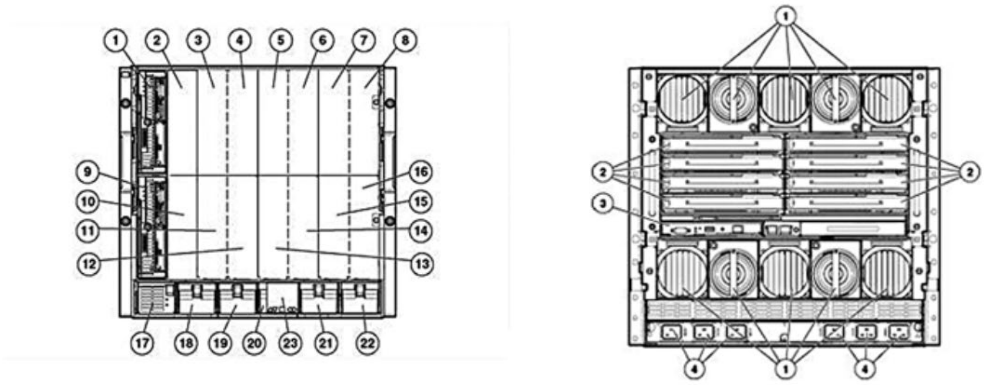


Figure 5. HPE BLc7000 Onboard Administrator with KVM Option (456204-B21) [ONLINE]. Available at: https://psnow.ext.hpe.com/doc/PSN3923547USEN.pdf

Administrators manage the blades through the Onboard Administration, which is both local and remote administration tool, allowing browser-based remote access for the HPE BladeSystem c-Class enclosures (Figure 6). The onboard administrator has multiple functions: detecting insertions and removal component, identification of components with required connectivity, power and cooling management and component controlling of remote controls and consoles.

The server blades are the separate, general-purpose ProLiant devices are hot-swappable, allowing for quick replacements should one of the devices need physical maintenance (Figure 7). On the servers, everything from local storage to virtualization happens. Optional mezzanine cards can also be installed on them to further increase the connectivity options. These mezzanine cards are used in our current setup to connect to our two networks.



HPE BladeSystem c7000 Enclosure - Front View

1-16. Device Bays 1-16

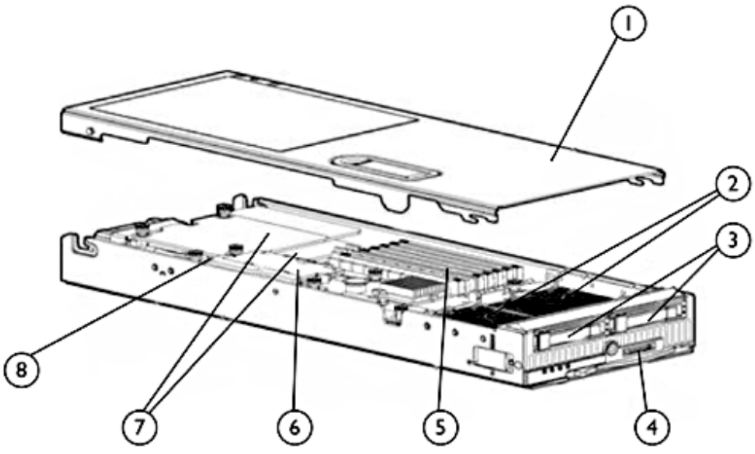
17-22. Power Supply Bays 1-6

23. Insight Display

HPE BladeSystem c7000 Enclosure - Rear View

1. Active Cool 200 Fans and Fan Bays
2. Interconnect Module Bays
3. BladeSystem Onboard Administrator (left) and optional redundant Onboard Administrator (right)
4. Power Inputs (single-phase 200-240V AC model shown)

Figure 6. HPE BladeSystem c7000 Enclosure [ONLINE]. Available at: https://h20195.www2.hpe.com/v2/GetDocument.aspx?docname=c04229580&doctype=quickspecs&doclang=EN_US&searchquery=&cc=us&lc=en pdf [Accessed 20 March 2018]



HP ProLiant BL460c Server Blade

1. Access Panel
2. Up to two Intel® Xeon® 5000 Sequence processors
3. Small form factor (SFF) hot-plug drive bays (standard BTO models)
4. Local I/O connector
5. Eight (8) PC2-5300, Fully Buffered DIMMs (DDR2-667) Memory Slots
6. HP Smart Array E200i Controller with optional battery-backed write cache (standard BTO models)
7. Two Mezzanine slots
8. Internal USB Connector (standard BTO models)

Figure 7. HP ProLiant BL460c Server Blade [ONLINE]. Available at: https://h20195.www2.hpe.com/v2/Getdocument.aspx?docname=c04111644 [Accessed 20 March 2018]

In this project, the enclosure will be housing the servers that we will designate to be our cloud nodes for our infrastructure. In the case of using the multi-node approach for installing OpenStack, we reserved six of the HP ProLiant BL460c server blades for use in the development of the cloud infrastructure. (HP, 2008)

Before the base OpenStack installation, VMware is installed initially to act as our management layer. Each ProLiant server blade hosts a VMware EsXi installation, which varies between versions 6.0 and 6.5.

5.1.2 HPE P2000 G3 Modular Smart Array

The HPE MSA hosts an 8Gb FC, 6Gb SAS, 1GbE iSCSI connected models, and an iSCSI model with four 1Gb iSCSI ports per controller (Figure 8). The arrays are 2U SAN or direct connection solutions that provide five controllers. The separate controllers are two FC's, one SAS, one 1GbE iSCSI and a four-port 1Gb iSCSI.

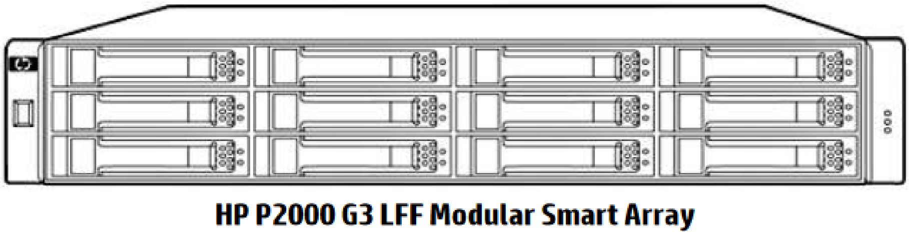


Figure 8. HP P2000 G3 Modular Smart Array, Front View [ONLINE]. Available at: https://h20195.www2.hpe.com/v2/getdocument.aspx?docname=c04168365 [Accessed 20 March 2018]

The dual-protocol P2000 G3 FC/iSCSI hosts 8Gb FC ports to support a full FC SAN (Figure 9). For our purposes, we will be utilizing the FC ports to map redundant FC connections in conjunction with a Brocade 300 FC Switch and the HPE BladeSystem c7000 Enclosure, specifically the interconnect modules located in the rear mezzanine card slots of the enclosure. The result should offer a bidirectional, dedicated connection between the SAN and the enclosure, allowing high-speed data transfer between the two devices. The outcome of this connection allows users to access their stored data within the system with little delay.

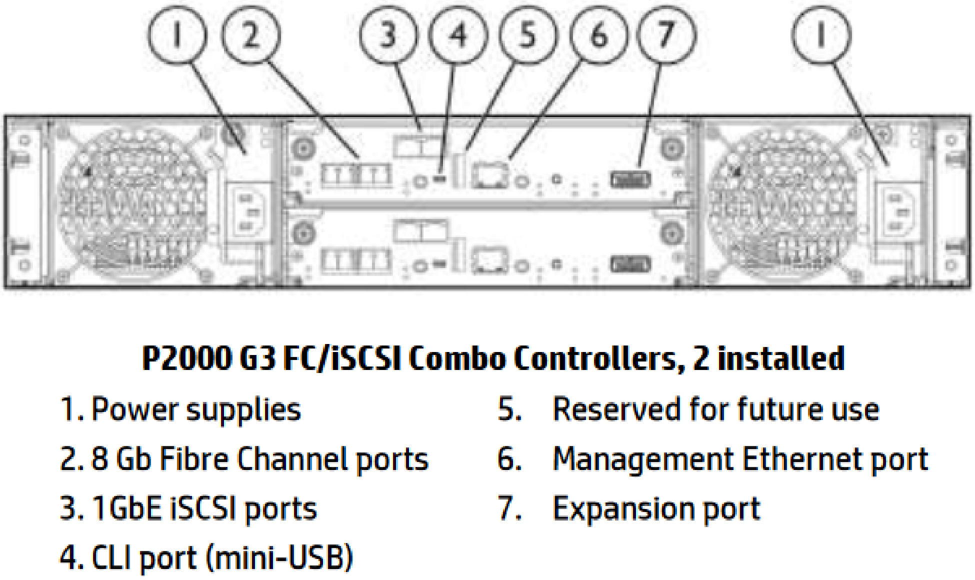

1
2
3
4
5
6
7
1
P2000 G3 FC/iSCSI Combo Controllers, 2 installed
1. Power supplies
2. 8 Gb Fibre Channel ports
3. 1GbE iSCSI ports
4. CLI port (mini-USB)
5. Reserved for future use
6. Management Ethernet port
7. Expansion port

Figure 9. HPE P2000 G3 Modular Smart Array, Rear-View [ONLINE]. Available at: https://h20195.www2.hpe.com/v2/getdocument.aspx?docname=c04168365 [Accessed 20 March 2018]

In the case of our project, we used a single Brocade 300 FC switch to handle the routing of the WWNs between the server enclosures and the SAN system. The HP ProLiant BL460c server blades are configured as hosts within the SAN, a method which allows the servers access to the storage capacity. In its current form, the 3.6TB capacity is more for theoretical purposes and light storage use than the original concept of applying a high-availability model across the cloud, using the storage array as a backup storage device for the entire cloud infrastructure.

As we are on the topic of storage, we will briefly touch upon raiding and storage provisioning techniques used in this project.

5.1.3 Raid and Raid 6

A Redundant Array of Independent Disks (RAID) is a storage technology that advertises greater availability and accuracy of disk storage through redundancy. The way RAID works is that is multiple independent sets of disk platters are housed within a single drive unit. This way multiple different surfaces of it can be accessed at a time. This allows for faster access in the following ways:

1. If a disk block is spread across multiple surfaces, then each surface can be simultaneously accessed. The access to the block itself takes less time. If a block is split into four parts onto four surfaces it can be approximately read or written four times faster.

2. If the drives are available as several independent ones, the possibility for two or more simultaneous accesses could occur. This is possible due to one block being found on one drive and another block on another drive. The drives can work independently of each other and both accesses to the block can take place at the same time rather than sequentially.

We sought to have our cloud infrastructure as redundant as possible. Redundancy also means an improvement in availability. By providing information about redundancy, so that small failures (such as bad sectors on one of the drives) does not make our entire storage network unavailable. The way we chose to provide redundancy was to duplicate everything from one drive onto another. This way we divide our RAID storage into two sets. The two sets are an exact mirror of each other. If our single sector, disk drive, surface or the entire set of disks would fail the drive would still be capable of handling the requests due to the mirrored set still being accessible. The biggest setback for this environment is that we lose exactly 50 % of our storage space.

We chose RAID 6 as our storage technology for our SAN-storage environment. The RAID 6 is relatively expensive storage but highly reliable. RAID 6 uses block pattern similar RAID 5 but it utilizes two different parity functions (to derive two different parity blocks per row) instead of one parity function. The parity works so that if one drive fails, its contents are reconstructed using one set of parity data. If another drive would fail before the other array is recovered. The contents of both the missing drives are reconstructed by the combination of the remaining data and two sets of parity. Raid 0-10 can be seen in Table 1. (Fox et al, 2018)

Table 1. Raid levels comparison chart, Raid levels comparison chart 12 June 2018, June 2018, <http://www.raid-calculator.com/raid-types-reference.aspx>.

	RAID 0	**RAID 1**	**RAID 5**	**RAID 6**	**RAID 10**
Minimum Number of Disks	2	2	3	4	4
Fault Tolerance	None	1 disk	1 disk	2 disks	1 disk
Disk space overhead	None	50%	1 disk	2 disks	50%
Read speed	Fast	Fast	Slow	Slow	Fast
Write Speed	Fast	Fair	Slow	Slow	Fair
Hardware Cost	Cheap	High	High	Very High	High

5.2 Cisco Systems

An American Company founded in 1984 based in San Jose, California, USA. Cisco Systems creates routing and switching hardware for businesses and private customers. Originally based in Menlo Park, their first products were network cards and routing devices. After acquiring Crescendo Communications, network switching hardware became part of their product line. Years later, Cisco Systems helped develop for the Internet of Things (IoT) and are gradually moving their business focus towards software. (Lewis, 2019)

5.2.1 Cisco Catalyst 3560 PoE-24

In this project, the item that we have running in our network is the Cisco Catalyst 3560 PoE-24 (Figure 10). The Catalyst acts as a scalable physical access point for our management devices, allowing the addition of supplemental servers that need to be added if reserving more ProLiant server blades is not possible. The features for the Cisco Catalyst 3560 PoE-24 are:

- Routing Protocol: OSPF, IGRP, BGP-4, RIP-1, RIP-2, EIGRP, HSRP, IGMP, DVMRP, PIM-SM, static IP routing, PIM-DM
- Remote Management Protocol: SNMP 1, RMON 1, RMON 2, Telnet, SNMP 3, SNMP 2c
- Communication Mode: Half-duplex, full-duplex
- Switching Protocol: Ethernet
- Status Indicators: Link activity, port transmission speed, port duplex mode, system
- Compliant Standards- IEEE 802.3u, IEEE 802.1D, IEEE 802.1Q, IEEE 802.1p, IEEE 802.3af, IEEE 802.3x, IEEE 802.3ad (LACP), IEEE 802.1w, IEEE 802.1x, IEEE 802.1s
- Expansion Slots - 4 x SFP (mini-GBIC)
- Power Over Ethernet (PoE)

(Cisco.com, 2014)

Figure 10. Cisco Catalyst Switch [ONLINE]. Available at: https://media.cablesandkits.com/ipn/WSC356024PSSa.jpg [Accessed 20 March 2018]

5.3 Brocade

Brocade Communications Systems, Inc. is an American-based company that was formerly known for its Fibre Channel storage networks. Now Brocade is a subsidiary of Broadcom Inc. Brocade has since expanded its repertoire to include third-platform technologies (Mukherjee and Baker, 2016). These include routers and network switches that can be utilized by data centers, campuses, etc. These also include IP and Fibre

Channel storage networks, software-defined networking and Network Function Virtualization. For our purposes, we are using the eponymously named Brocade 300 Fibre Channel switch. (Matsumoto, 2017)

5.3.1 Brocade 300 FC Switch

The Brocade 300 provides an 8Gb connection and switching capabilities for FC applications and networks (Figure 11). The basic license offered allows eight FC ports to be utilized, with the ability to add an additional 16 ports upon upgrading from the initial license. The switch is made to give scalability to businesses that need to grow alongside their capacity needs. In its basic form, the switch allows the client to link with other switches via ISL Trunking for expanding the FC network, offering more connectivity with less network complexity.

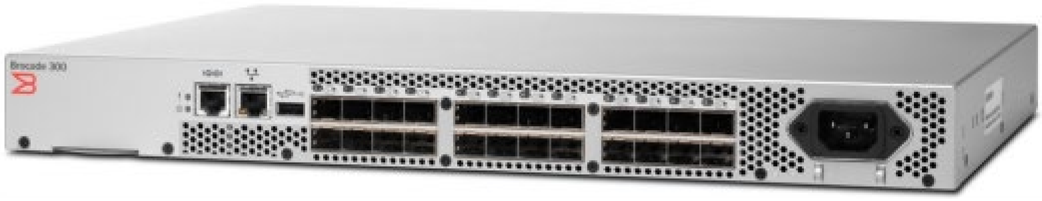


Figure 11. Brocade 300 FC Switch [ONLINE]. Available at: https://lenovopress.com/lp0044.pdf [Accessed 20 March 2018]

In this project, we will be using one FC switch for connectivity purposes. Ideally speaking, we should use two fabric switches to connect the enclosure to the SAN system for redundancy purposes. One can pipe traffic through two separate FC switches as the main switch to handle traffic and a failover connection should anything happen to the main FC switch. In the case of our project, we only used one FC switch and mapped two paths into the device, connecting both the servers within the enclosure and the SAN.

World Wide Names (WWNs) are unique identifiers for FC-capable devices. These are automatically assigned to ports that handle the traffic that use fiber optics.

5.4 Lenovo

Lenovo Group Ltd., also known as Lenovo PC International, is a Chinese multinational technology company, with two company headquarters based in Beijing, China and Morrisville, North Carolina, USA. It designs, builds and sells a wide array of technology and smart devices, from personal computers to electronic storage devices, to smart televisions. Originally founded in Beijing in November of 1984 as the New Technology Development Company for the Institute of Computing Technology of the Chinese Academy of Sciences. (Ahrens, Zhou, 2018)

5.4.1 Lenovo ThinkPad X230

For this project, we are utilizing three devices developed by Lenovo like the Lenovo ThinkCentre M90 and two Lenovo ThinkPad X230s. Of the two devices, we will first cover the ThinkPad X230 (Figure 12), as we used these laptops as an access point to manage the infrastructure. The relevant specifications will be referred to in Table 2.

Figure 12. Lenovo ThinkPad X230 [ONLINE]. Available at: http://psref.lenovo.com/syspool/Sys/PDF/withdrawnbook/ThinkPad_X230.pdf [Accessed 20 March 2018]

Table 2. Lenovo ThinkPad X230 Specs

Lenovo ThinkPad X230 Specifications	
CPU	Intel Mobile Pentium III-M 800 or 866 MHz CPU Intel Core i5-3210M processor (2 cores, 2.50GHz, 3MB cache), DDR3 memory controller (up to 1600MHz), Intel Turbo Boost 2.0 (3.10GHz), HT technology
RAM	16GB max / PC3-12800 1600MHz DDR3, non-parity, dual-channel capable, two 204-pin SO-DIMM sockets
Storage	500GB 7200rpm
Networking	Intel 82579LM Gigabit Network Connection (Lewisville), PHY, PCIe x1

5.4.2 Lenovo ThinkCentre M90 3245

The Lenovo ThinkCentre M90 3245 is a small form desktop computer (Figure 13). This computer is our Pfsense host. The resources required to run Pfsense is minimal, so even if the computer equipped with lower-end stock hardware, the M90 runs fine as a firewalling/routing device. The current relevant specifications are listed in Table 3.

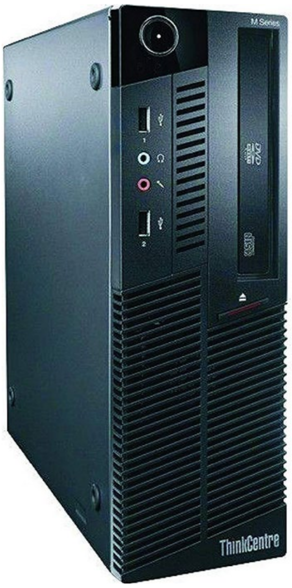


Figure 13. Lenovo ThinkCentre M90 3245 [ONLINE]. Available at: https://images-na.ssl-images-amazon.com/images/I/91v9gH28kvL._SL1500_.jpg [Accessed 20 March 2018]

Table 3. Lenovo ThinkCentre M90 3245 Specs

Lenovo ThinkCentre M90 3245 Specifications	
CPU	Intel Core i3 550 / 3.2 GHz, Dual-Core, Hyper-Threading Technology, Intel Extended Memory 64 Technology, Intel Virtualization Technology
Video	Intel HD Graphics, DisplayPort, VGA
RAM	2GB / 16GB (max), 1333 MHz, DDR3 SDRAM
Storage	320 GB
Networking	Ethernet, Fast Ethernet, Gigabit Ethernet

5.5 Fujitsu

Fujitsu Limited is a Japanese technology company based in Tokyo. In 1935, Fujitsu was established when splitting from Fuji Electric and creating a joint/venture with Furukawa Mining Company and Siemens, developing electrical equipment as their business focus. (Fujitsu.com, 2019)

5.5.1 Fujitsu Celsius W280

The Fujitsu Celsius W280 is the device used in this project (Figure 14). Originally, it was used as an IoT server to manage the device data received from the IoT devices. Once the main goal of the project changed from IoT cloud creation to cloud infrastructure development and management, the Fujitsu Celsius W280 was repurposed to become a Docker server. This was to provision tools and libraries for OpenStack once the client environment was developed.

Figure 14. Fujitsu Celsius W280[ONLINE]. Available at: https://www.pcbilliger.de/media/image/1d/af/78/fujitsu-celsius-w280-gallery.jpg [Accessed 20 March 2018]

6 SOFTWARE

The software licenses were provided by the Turku University of Applied Sciences, and its use was proctored by the staff of the Internet Technology Lab. The software ranged from firewalls, data storage management programs, virtualization technology and the such for use in this project.

6.1 KeePassX

KeePassX is a free, open-source password manager that allows the user to store passwords in a secure way. All passwords are stored in a database locked with a master key or key file. Only the master password can be used to unlock the database. The databases are encrypted using AES and Twofish encryption methods.

KeePassX is an application for individuals with extremely high demands on secure personal data management. Its interface is light, it works cross-platform and is published under GNU General Public License.

KeePassX saves different pieces of information e.g. usernames, passwords, URLs, attachments and comments in a single database. To manage KeePassX stored passwords one can use specific icons for every single entry. You can also secure your own specific password file. The password file has all the passwords you may have saved into it. Furthermore, the entries can be sorted into groups, which are customizable as well. The search function allows searching in a single group or the whole database. KeePassX also has a small utility for password generation. The password generator is highly customizable, fast and easy to use. For someone who wants to create random passwords, this feature comes in handy. The whole of the database is always encrypted with the AES (aka Rijndael) encryption algorithm which uses a 256-bit key.

Our main use of KeePassX is to store all the different passwords for our machinery. Since KeePassX works by ctrl-c and ctrl-v for password extraction it makes it safer. If a keylogger was installed onto your computer, it would only get the local login for KeePassX database login. After this, the keylogger would not be able to sniff out more passwords due to the copy-paste function. (Official keepassx.org, 2018)

6.2 Pfsense

Pfsense is an open source firewall/router computer software distribution based on FreeBSD. It is installed on a physical computer or a virtual machine (physical machine in our case) to make a dedicated firewall/router for a network. Pfsense is acclaimed for its reliability and features that can only be found in otherwise expensive commercial firewalls. Our installation of Pfsense is carried out by placing the ISO on a USB stick and hereinafter booting a computer from the USB stick. Said ISO can be found on the official website and is free to download.

Pfsense integrated graphical web-based configurator is a powerful interface which lets administrators configure advanced setups. We are required to manually add static routes, ipv4 addresses to correct interfaces and configure LAN rules. Pfsense will be covered further in our project execution section. The general requirements can be seen in Table 4.

Table 4. These are the minimum and recommended requirements for Pfsense (Official pfSense.org 2018) https://www.pfsense.org/products/#requirements

General Requirements		
	Minimum	Recommended
CPU	500 Mhz	1 Ghz
RAM	512 MB	1 GB

Requirements Specific to Individual Platforms:

- Full Install
- CD-ROM or USB for initial installation
- 1 GB hard drive

6.3 OpenStack

Developed in 2010, OpenStack was formed from the combination of two separate missions of two different entities. Rackspace Inc., an American cloud computing company, originally wanted to recreate the existing code of their cloud infrastructure. Anso Labs, another American company that offered cloud consultation and was a service provider before Rackspace acquisition, published code for Nova, a Python-based fabric controller for cloud computing. The OpenStack project was then officially announced in at the Open Source Convention (OSCON) on the 21st of July 2010. (OpenStack history, 2018)

For this project, OpenStack was planned for client use and development. The infrastructure would be installed within the VMware environment. Each node within the OpenStack installation would be an All-in-One installation to stand as its own cloud structure. Theoretically speaking, this would represent a cloud-in-cloud infrastructure. As we did not have time to fully implement this feature it will be explained in a later section.

6.4 Docker

Originally founded as dotCloud, a project named "Docker" had since garnered the attention of the company leaders, rebranding their company to Docker, Inc. to focus their attention to software containerization. Containerization in this context is a name for operating-system-level virtualization, the eponymously-named program allowing for headless initialization through isolated, single-OS kernel Linux environments. Through these instances, operation remains lightweight as the requirements for applications run through Docker are bundled together alongside the hosted application. (Docker FAQ, 2019)

In our project, we were to utilize the Docker Hub toolset to provision the libraries required for containers to run within the OpenStack or our main virtual environment. As we did not have time to fully implement this feature, this will be discussed in a later section. We will also discuss the real-world usage for how we would have used this software in our infrastructure. The installation part can be seen in Appendix 1.

6.5 VMware

In 1998 VMware was founded by a group of five IT experts. From there on out, they gave out their first product called VMware Workstation in 1999. This was followed by the VMware GSX Server in 2001.

The popularity of VMware rose quickly due to it being compatible with all the major OS's, which includes Mac OS, Linux and Microsoft Windows. The three different desktop software's are: VMware workstation which allows the user to run multiple operating systems of the same or different kind on a single physical computer. VMware fusion was designed for users on the Mac OS. Lastly, there is VMware Player which was launched as freeware to those who do not have a licensed VMware product.

VMware also made the software hypervisors which are bare-metal embedded hypervisors intended for servers. The hypervisors are made to run directly on the server hardware without any particular OS. There are a couple of different hypervisors and the most important ones are listed below. These are also the ones we used to create our infrastructure. (Techopedia.com, 2019)

6.5.1 VMware EsXi 6.0/6.5 and vSphere

EsXi is the hypervisor mentioned above. The EsXi is a very small piece of software that is installed on a single physical server or host. This software allows the user to run different operating systems, also multiple at a time. The OSs installed are completely separated from each other but are able to communicate via a network to the outside world. The outside network also guides them to the rest of the computers running on the LAN. EsXi, vSphere, and vCenter are all components of vSphere. Though EsXi server is the most important part of vSphere. The virtualization server is , in fact, the EsXi, and a 1 hypervisor if the EsXi's only function is to run virtual machines. To be able to access all the newly created virtual servers sitting on top of the EsXi, the vSphere client or vCenter comes into play. An administrator is able to connect to a specific EsXi server and access or manage the virtual machines found. The vSphere client was in our case installed both on our administrative laptops, both in Linux and Windows environment. We were able to connect from the vSphere client to our newly created EsXi servers and start managing the virtual machines. (Bipin, 2012)

6.5.2 VMware vCenter 6.0

The central management software for the whole infrastructure can be done from the vCenter. If you e.g. wanted to clone an existing virtual machine using vSphere client without the aid of the vCenter server. This cannot be done. vCenter server is quite similar to vSphere but it has more power as a server. The vSphere client is used to access the vCenter Server and then being able to manage the ESXi servers. If you only needed to manage one virtual machine, it is possible to log into a web client to its exact IP address. But when it comes to managing multiple virtual machines the vCenter will do the job. vCenter also allows for the advanced features like the already mentioned VM cloning, vMotion, High Availability, Fault Tolerance, and Distributed Resource Scheduling. (Vladan, 2019)

7 PROJECT EXECUTION

We worked closely with the staff of the Internet Technology Lab at Turku University of Applied Sciences from December 2017 - May 2018. Our project started as an IoT based network with security in mind but was scrapped when we noticed the lack of IoT devices in the school. We planned our next move very intricately because we knew that creating something from scratch as a two-man team will take a lot of time and time is something we did not have enough of. A couple of months into our project we sat down with Ossi Väänänen and re-evaluated how our thesis should be written. The first plan was to write it separately and divide the tasks 50-50 and create two logical and comprehensive theses. This plan was later changed so that we were able to create one singular thesis. This also allowed our thesis supervisor Ossi Väänänen to have his first surveilled thesis that is created by two persons. This allows us to pioneer in Turku University of Applied Sciences engineering department by creating the two-man thesis project.

The staff also gave their input whenever their time-schedule allowed them to. In an earlier course, we were given student licenses to VMware and VCenter that allowed us to get our hands on the enterprise versions for our cloud infrastructure and thesis. The enterprise licenses allowed for more virtual machines to be created and allowed for more hardware. Since VMware is subscription based per CPU core, the better the subscription the more the user can create. This would have allowed for much scaling when our infrastructure would have been ready to be turned into an IaaS. The IaaS vision would've been able to provide private environments for customers to create a certain number of virtual machines and have a set storage space. The number of virtual machines and storage space would then be adjustable via payment.

The staff engineers made it possible for us to move into the creation of a Cloud Infrastructure. They instructed us of what equipment are available to use and what is plausible to combine when planning the Cloud. Our test environment grew each week from two Lenovo ThinkPad X230 laptops and a Cisco Catalyst 3560-24 PoE switch, to having a fully-equipped infrastructure with an HPE BladeSystem c7000 enclosure and a fully-functional HPE MSA P2000 G3 Modular Smart Array System to complement the framework. Nearing the end of May 2018, we understood that the deadline was coming up. If we would have added more machinery to our infrastructure we would have been overwhelmed. Each machinery required a lot of configuration and in-depth knowledge. This also applied to the software we used e.g. VMware and OpenStack.

In April 2018 we stopped adding more software and equipment to our setup. This was due to the fact the schools' borrowed equipment needed to be cleaned and reset so they can be configured for other courses starting in August 2018. By May 2018 we managed to get all the parts of our setup moving and working for our desired cloud infrastructure. We also got OpenStack Kolla-Ansible and Docker working at this time. However, we would have wanted to continue all the way to the point where our Infrastructure could have been fully functional with OpenStack. This would ultimately make it possible to easily arrange sections of the storage and hardware for customers to use. Though this would also have required a login server and a customer-specific online environment. Hereinafter we will describe how we setup our hardware and software which ultimately became a base for our Cloud Infrastructure.

7.1 Mapping the Roadmap

We started our journey in February 2018 by receiving two Lenovo laptops and a Cisco Catalyst 3560-24 PoE switch, which was handed to us by the internet technology lab staff. At the same time, we were planning our thesis with Ossi Väänänen. The planning took roughly a month since both the approach of the thesis and the creation of a cloud is new. By the end of February, we had received the news that we can write the thesis as a two-man group and that we are allowed to borrow all the needed equipment we wanted from the Cisco Lab at Turku University of Applied Sciences. The software came a bit later which the staff also aided us getting.

7.2 The Initial Setup

The installation of the laptops and the configuration of the Cisco Catalyst 3560-24 PoE switch began. We installed windows 7 on one of the Lenovo laptops and Linux Mint on the other. We installed different OS's on the two laptops for precaution and fallback measures. This means that if something did not want to work on a Linux mint OS, we could always do it on the Windows environment. Alexander Enberg oversaw the Windows 7 Lenovo and Osaiasi Foleti used the Linux Mint laptop. Since we only had two laptops and a switch at the time, we had to make the laptops communicate with the switch with the help of USB-to-Serial adapter. We used the commands in Table 5 on the Linux Mint laptop to find and make use of the USB COM adapter.

Table 5. The console commands needed for enabling a minicom in Linux

```
sudo apt-get install minicom


dmesg | grep tty


output for a USB-to-Serial adapter


[   0.000000] console [tty0] enabled

[   5.065029] usb 4-3: pl2303 converter now attached to ttyUSB0
```

What we are interested in is the name of the serial port. In the first section above it is ttyS0, the other is ttyUSB0. which we will need in order to use Minicom. Next, we enter the following the command found in Table 6.

Table 6. The command for accessing minicom in Linux

```
sudo minicom -s
```

Next, one will want to consult the hardware vendor's manual for the bits per second, data bits, parity, stop bits, and flow control. Once these have been adjusted, one may want to also go to the modem and dialing menu and remove all information in setting options A through I. Once configured, one may "Save setup as dfl", which will save these as the default configurations for future connections (/etc/minicom/minirc.dfl). When the ttyS0 is connected it should show as an option as seen in Table 7.

Table 7. The output showing the file path for the serial device once set.

```
A -    Serial Device      : /dev/ttyS0
```

After this was done, we were able to use Cisco CLI commands in a separate window which had the minicom USB connection. This allowed us to communicate with the Cisco Catalyst 3560-24 PoE switch. The catalyst acted as the gateway for our supplementary items, allowing access to the lab network, and in turn our own cloud network. The devices that were directly connected to the Cisco Catalyst are the two Lenovo ThinkPad X230s and the Fujitsu Celsius W280. The configuration for the Cisco Catalyst was made so that VLANs were created for both the users, Alexander and Osaiasi on their ThinkPads, and for the Fujitsu Celsius. Each was configured to handle traffic from their corresponding ports, in conjunction with the set of IP addresses that we were allowed to work with. Our current setup at this time can be seen in Figure 15. The set IP addresses we were given can be seen in Table 8

Table 8. Set of IP’s given to us by Turku University of Applied Sciences staff

Network Address:	172.26.32.0
Usable Host IP Range:	172.26.32.1 - 172.26.32.6
Broadcast Address:	172.26.32.7
Total Number of Hosts:	8
Number of Usable Hosts:	6
Subnet Mask:	255.255.255.248
Wildcard Mask:	0.0.0.7
Binary Subnet Mask:	11111111.11111111.11111111.11111000
IP Class:	C
CIDR Notation:	/29
IP Type:	Private
Short:	172.26.32.1 /29
Binary ID:	10101100000110100010000000000001
Integer ID:	2887393281
Hex ID:	0xac1a2001
in-addr.arpa:	1.32.26.172.in-addr.arpa
IPv4 Mapped Address:	::ffff:ac1a.2001
6to4 Prefix:	2002:ac1a.2001::/48

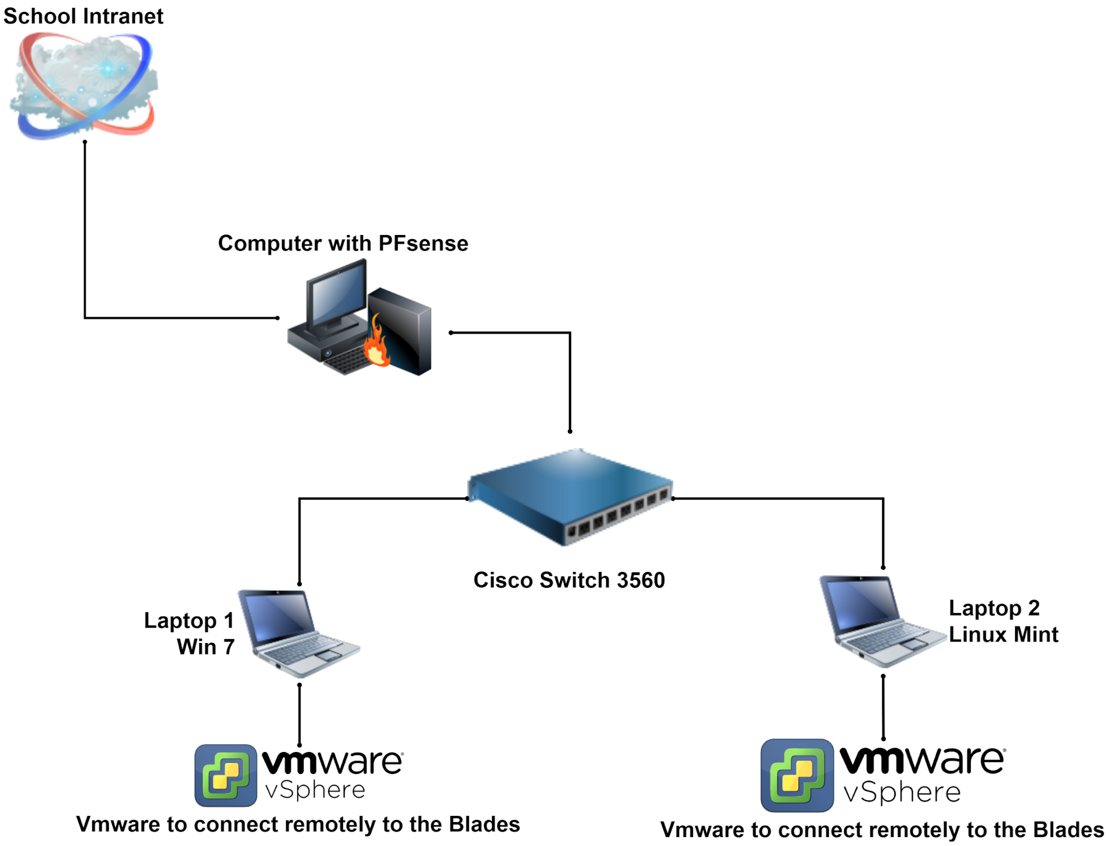


Figure 15. Our initial setup and control center

7.3 Firewalls

To amplify the reality of our environment, we got a hold of a Lenovo ThinkCentre M90 3245 and turned it into a Pfsense Firewall. The idea behind the Firewall was to protect the out- and incoming traffic of our environment. Our firewall also handled the traffic inside our network. Since we wanted this realistic approach to our cloud infrastructure, we were faced with some issues regarding other firewalls located in Turku University of Applied Sciences and the rules set for them. There are two more main firewalls in Turku University of Applied Sciences, one governing traffic going out from the school's intranet (to the outside world) and another one which is one teacher's own governed inside network. Our Cloud Infrastructure and firewall sits inside the innermost circle of the FW3 shown in Figure 16. The firewall closest to us and the one keeping us inside a limited range is the FW2. We are still able to access the internet but we're not able to e.g. create a tunnel to work on our cloud infrastructure from home. This access was blocked by both FW1-2. The firewall FW1 governs the outgoing and incoming traffic from the whole school's intranet and the outside web.

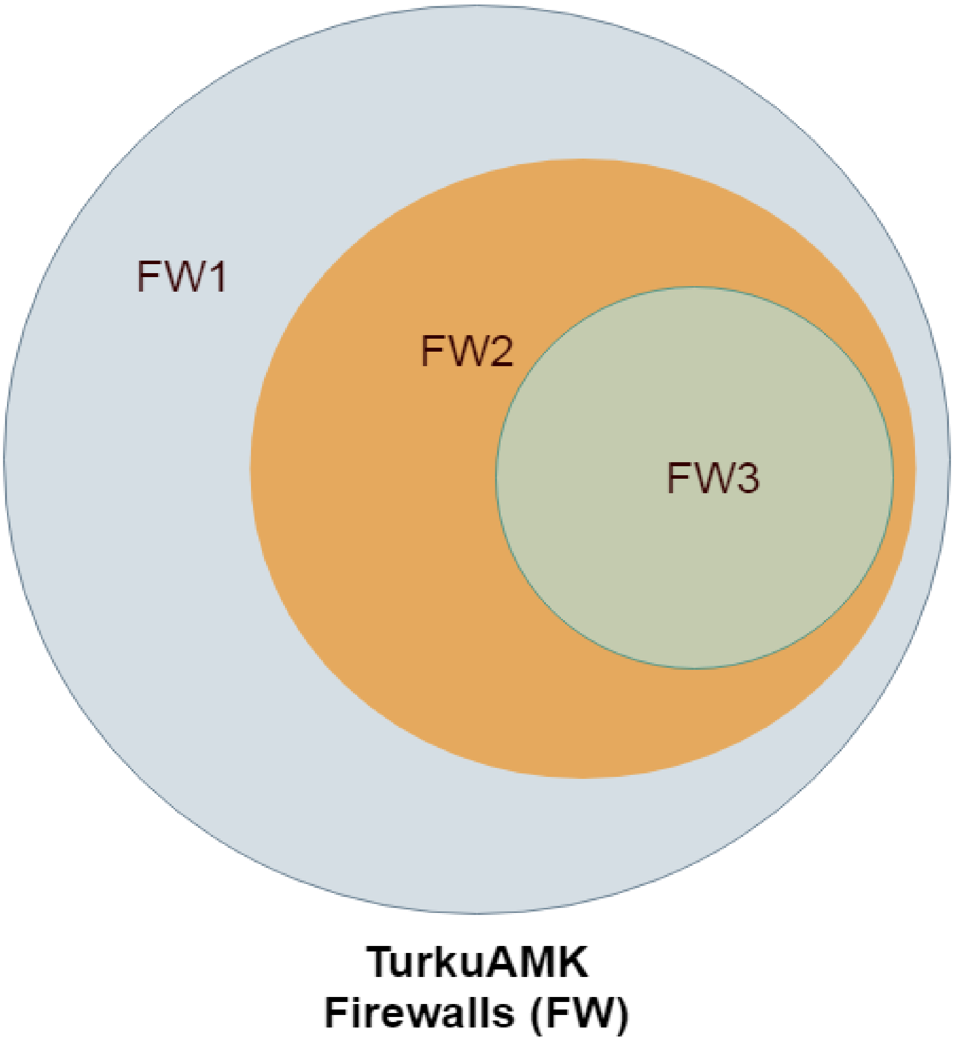
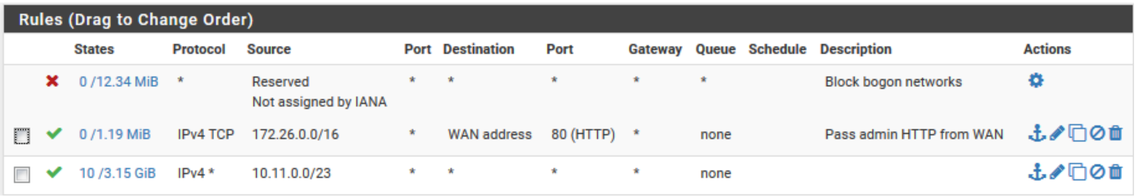


Figure 16. Nested Firewalls inside Turku University of Applied Sciences

Our setup so far consists of two laptops, a Cisco Catalyst Switch and a Fujitsu with Pfsense. We wanted to change that. We were granted access to use all the HP-blades and a SAN. The reason why the Firewalls were mentioned, FW2 especially, is that we had to figure out a way to be connected with FW2 while half of our infrastructure sits inside FW3. Luckily, we knew the engineer behind the FW2 firewall. He was able to configure his FW2 to accommodate our situation. The firewall FW2 needed to be configured so that two separate zones can communicate with each other. FW3 and FW2 were in different IP-zones. FW3 sat inside 172.26.#.# network but the FW2 firewall was in the 10.11.#.# network. All our newly acquired machinery (HP-blade and SAN) sat inside FW2 and needed to be able to communicate with our end at FW3. The FW2 needed management access to our Pfsense at FW3. This was done by using the following rules in Figure 17.

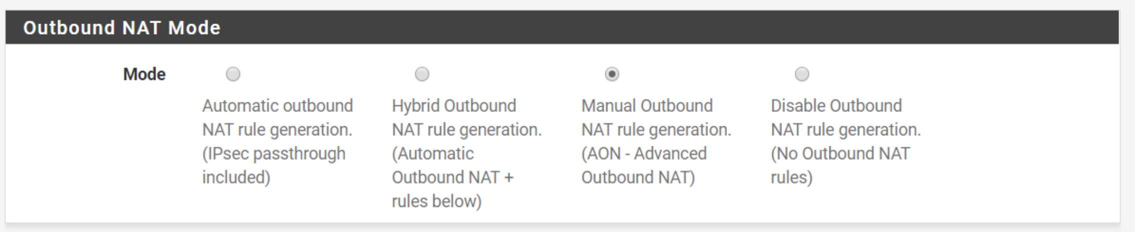


Rules (Drag to Change Order)

		States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
	✗	0 /12.34 MiB	*	Reserved Not assigned by IANA	*	*	*	*	*		Block bogon networks	⚙
□	✓	0 /1.19 MiB	IPv4 TCP	172.26.0.0/16	*	WAN address	80 (HTTP)	*	none		Pass admin HTTP from WAN	⚓✎⧉⊘🗑
□	✓	10 /3.15 GiB	IPv4 *	10.11.0.0/23	*	*	*	*	none			⚓✎⧉⊘🗑

Figure 17. The rules that were set for FW2 to gain management access to our end FW3.

To allow the traffic to be handled correctly we needed to configure NAT of the FW3. Outbound NAT does not control which interface traffic will leave, only how traffic is handled as it exits. To control which interface traffic will exit, we had to use static routes (Figure 18).



Outbound NAT Mode

Mode

○ Automatic outbound NAT rule generation. (IPsec passthrough included)

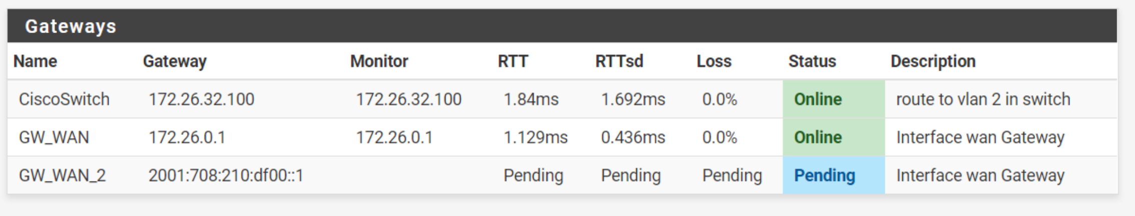
○ Hybrid Outbound NAT rule generation. (Automatic Outbound NAT + rules below)

● Manual Outbound NAT rule generation. (AON - Advanced Outbound NAT)

○ Disable Outbound NAT rule generation. (No Outbound NAT rules)

Figure 18. Outbound NAT determines how traffic leaving a Pfsense system will be translated.

To allow our Pfsense to communicate with our switch and the other network that contained the HP-Blades and SAN we had to configure the gateways on our side. Because if there are multiple paths to the internet or other networks via different gateways, the associated ones must be defined. Also, if there are static routes in play, gateways are needed to be defined (Figure 19).

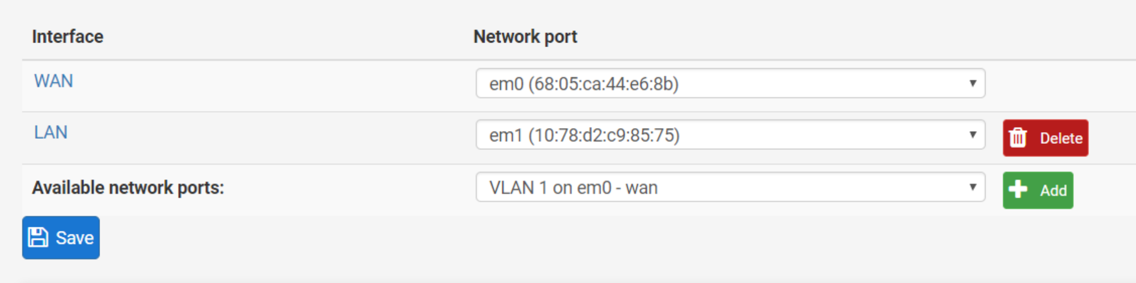


Gateways

Name	Gateway	Monitor	RTT	RTTsd	Loss	Status	Description
CiscoSwitch	172.26.32.100	172.26.32.100	1.84ms	1.692ms	0.0%	Online	route to vlan 2 in switch
GW_WAN	172.26.0.1	172.26.0.1	1.129ms	0.436ms	0.0%	Online	Interface wan Gateway
GW_WAN_2	2001:708:210:df00::1		Pending	Pending	Pending	Pending	Interface wan Gateway

Figure 19. Gateways that are in use to allow our different networks to communicate

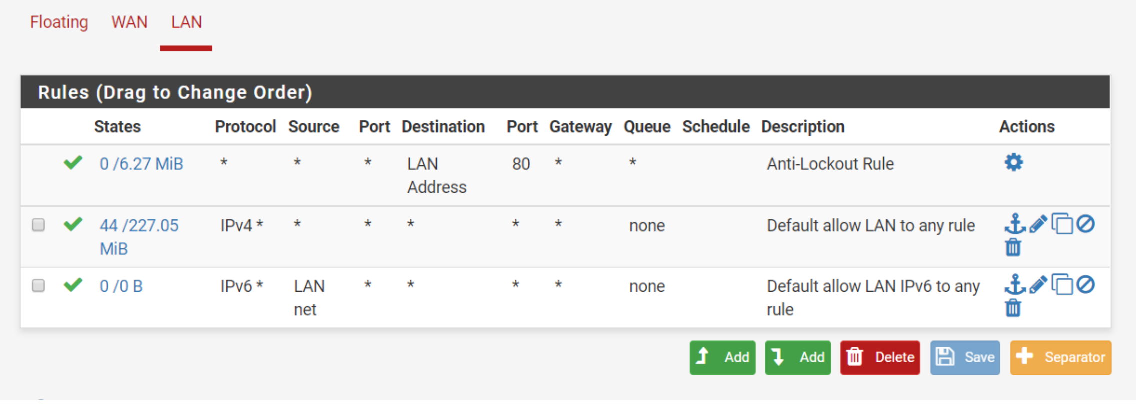
For our Pfsense to be able to work with two networks simultaneously, it had to be equipped with two network cards. These can be seen in Figure 20. Network card em0 was used for the WAN interface. We were able to get online by using this interface. The em1 was used for our inside intranet. By using multiple network cards, it lessens the strain on packet communication and lowers the error margin.



Interface	Network port
WAN	em0 (68:05:ca:44:e6:8b)
LAN	em1 (10:78:d2:c9:85:75)
Available network ports:	VLAN 1 on em0 - wan

Figure 20. Our two separate network ports. WAN (em0) for outside and LAN (em1) for inside network

We needed to apply an Anti-Lockout rule (Figure 21). This rule allowed us to stay logged in rather than have a 10-minute timer. The other two “default allow LAN to any rules” were set by Pfsense by default.

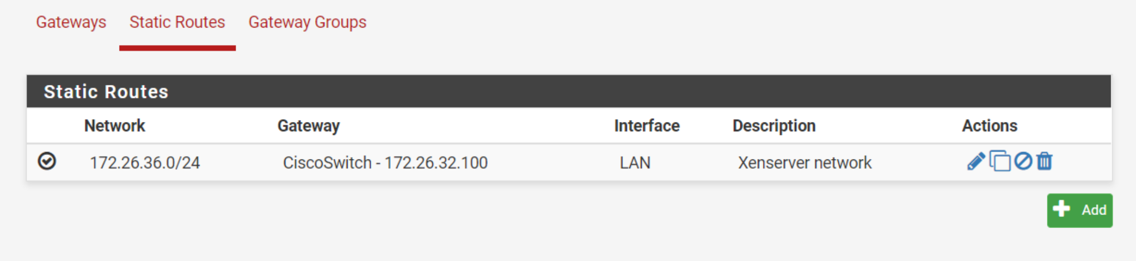


Rules (Drag to Change Order)

		States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
	✓	0 /6.27 MiB	*	*	*	LAN Address	80	*	*		Anti-Lockout Rule	
□	✓	44 /227.05 MiB	IPv4 *	*	*	*	*	*	none		Default allow LAN to any rule	
□	✓	0 /0 B	IPv6 *	LAN net	*	*	*	*	none		Default allow LAN IPv6 to any rule	

Figure 21. We applied an Anti-Lockout rule to stay logged in

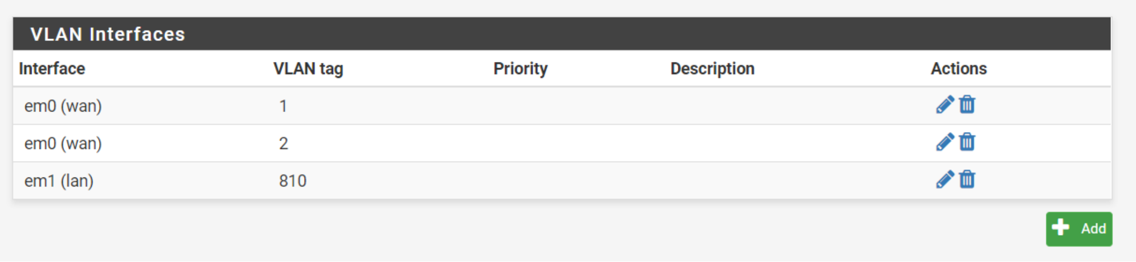
Figure 22 shows our static route between the Pfsense firewall and our Cisco Switch VLAN. This static route allows traffic that comes from another network to find its way into our switch and also the other way around. So that traffic hailing from our laptops, switch and other connected equipment also can move past the switch and through the Pfsense. All the VLANS used inside our Pfsense can be seen in Figure 23. The general information about our Pfsense can be seen in Appendix 2



Static Routes

	Network	Gateway	Interface	Description	Actions
✓	172.26.36.0/24	CiscoSwitch - 172.26.32.100	LAN	Xenserver network	

Figure 22. This was the route for access between the server network and our management network



VLAN Interfaces

Interface	VLAN tag	Priority	Description	Actions
em0 (wan)	1			
em0 (wan)	2			
em1 (lan)	810			

Figure 23. The interfaces linking both the server and management network

7.4 VMware/vSphere and vCenter

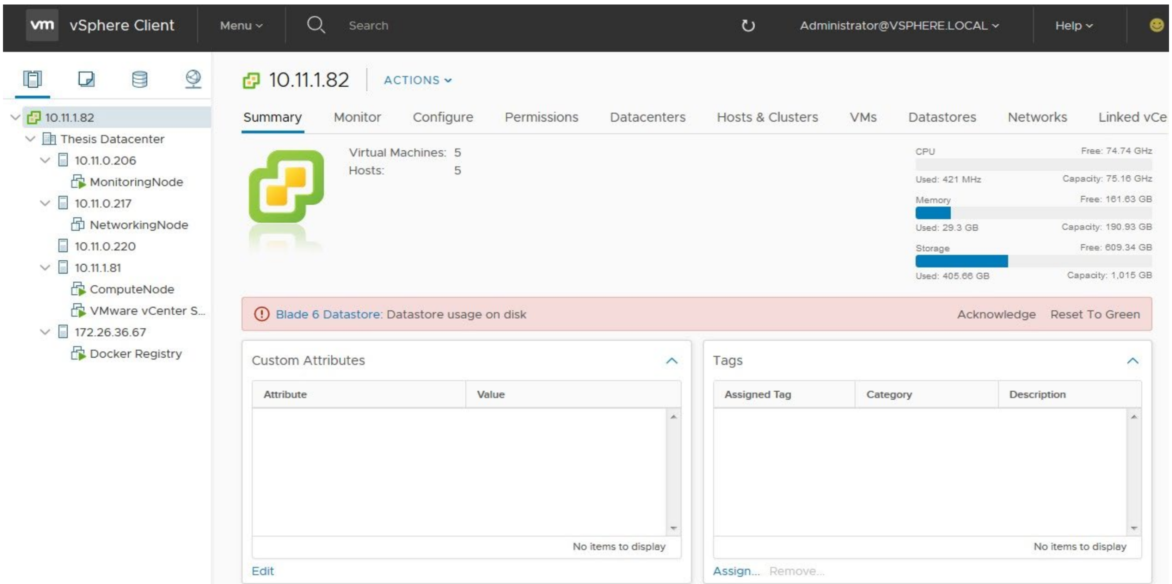
Our next part of the journey included finding something that could give a great overview of our whole cloud environment. We ended up going for VMware, vSphere, and vCenter. We also needed to upgrade the VMware license in order to support the size of our Cloud Infrastructure. Since the free version was not up to the task. As mentioned before, we still had our student credentials for an earlier course available. This allowed us for the Enterprise versions of the different VMware products. The Enterprise versions support larger infrastructure and have more management options. The difference between the standard edition and the enterprise versions can be seen in Appendix 3.

After we had worked out our firewall issues, we were able to start on the HP blade environment which would become our servers for handling most of the computing. Our initial task was to clean all the blades since they were used by earlier students. After this was done, we needed to connect all the blades together so that they were able to communicate. Now that they communicated with each other we were able to start adding our needed VMware software

Our goal was to create a set of virtual machines or even an endless amount of them, as long as our storage was sufficiently large enough. This required both vSphere EsXi 6.0/6.5 and hypervisors. We also needed something that was able to hold them all together as a centralized management system, vCenter. Lastly, we wanted a web client for our vCenter, to be able to configure our system from an online browser. vSphere provided us with that web client.

We started out with EsXi because we were most familiar with it and version 6.0 worked on most of the targeted computers. We also had the installation discs ready from the earlier courses. Our first initial tests on installing EsXi 6.0 were successful. These installations were on a couple of local computers after which we moved on to install the EsXi 6.0 on the blade servers.

As we obtained more and more computers/servers to work with the EsXI 6.0 we were confident that the current student license worked well for us. The next step was to get a student license for the vCenter. All the VMware software solutions work well together and need also need certain software to work in unison. With this I mean: how we need vSphere to access the ESXi and the web client for vCenter, and vCenter for cloning existing virtual machines. A summary view of our vSphere can be seen in Figure 24 and the networking overview in Figure 25.



Virtual Machines: 5
Hosts: 5

CPU Free: 74.74 GHz
Used: 421 MHz Capacity: 75.16 GHz
Memory Free: 161.63 GB
Used: 29.3 GB Capacity: 190.93 GB
Storage Free: 609.34 GB
Used: 405.66 GB Capacity: 1,015 GB

Blade 6 Datastore: Datastore usage on disk

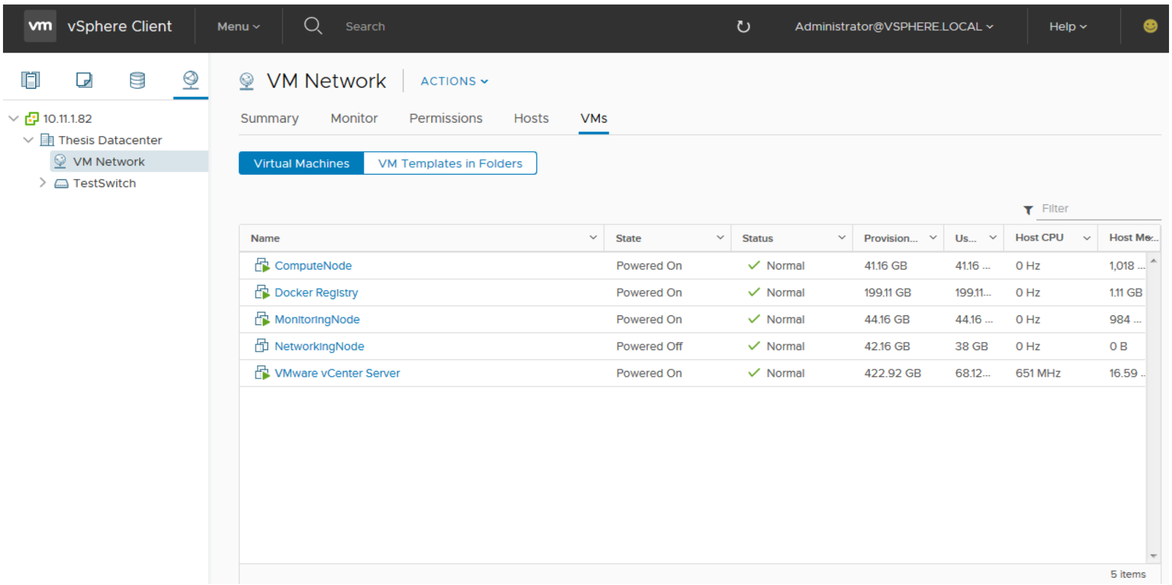
Attribute	Value

No items to display

Assigned Tag	Category	Description

No items to display

Figure 24. Summary overview of our vSphere Client

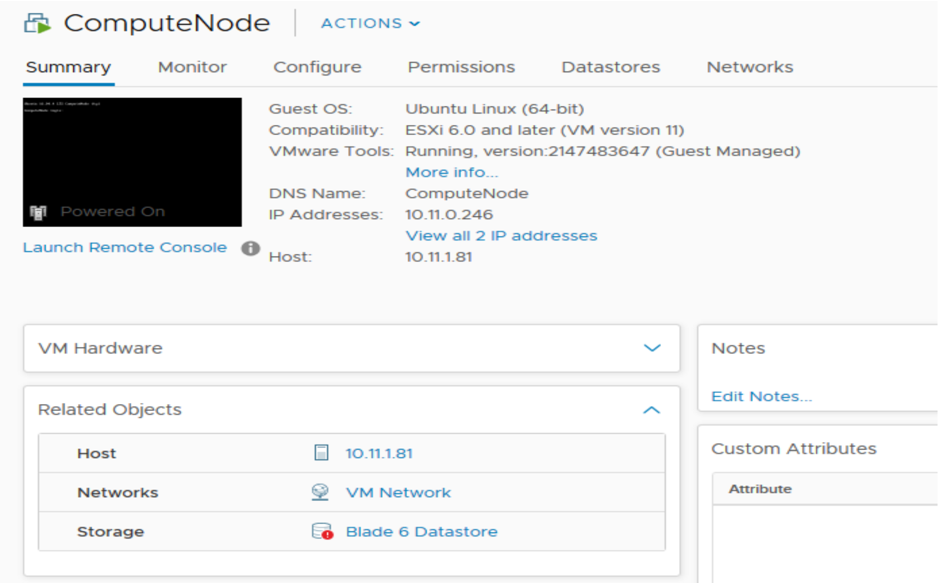


Name	State	Status	Provision...	Us...	Host CPU	Host Me...
ComputeNode	Powered On	✓ Normal	41.16 GB	41.16 ...	0 Hz	1,018 ...
Docker Registry	Powered On	✓ Normal	199.11 GB	199.11...	0 Hz	1.11 GB
MonitoringNode	Powered On	✓ Normal	44.16 GB	44.16 ...	0 Hz	984 ...
NetworkingNode	Powered Off	✓ Normal	42.16 GB	38 GB	0 Hz	0 B
VMware vCenter Server	Powered On	✓ Normal	422.92 GB	68.12...	651 MHz	16.59 ...

5 items

Figure 25. The network overview of our vSphere Client

When we installed the vCenter we already had the HP blades to work with. In the HP blade system, the hardware specifications differed from each other. Of the six servers reserved, server blade six was selected to become the “brain” of the operation since it was able to handle large amounts of our current data and even more if so needed. We named the “brain-blade” compute node, which can be seen in Figure 26. This was done due to the additional components the specific blade-server was equipped with, compared to the other servers.

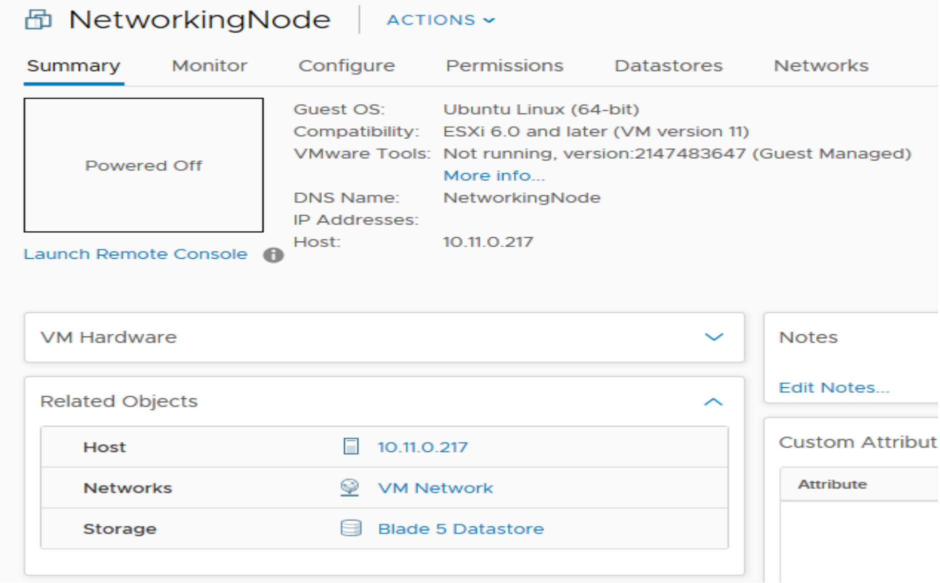


Guest OS: Ubuntu Linux (64-bit)
Compatibility: ESXi 6.0 and later (VM version 11)
VMware Tools: Running, version:2147483647 (Guest Managed)
More info...
DNS Name: ComputeNode
IP Addresses: 10.11.0.246
View all 2 IP addresses
Host: 10.11.1.81

Host	10.11.1.81
Networks	VM Network
Storage	Blade 6 Datastore

Figure 26. Summary of the Compute Node

Server blade six is equipped with two Quad-Core Intel Xeon processors and 65.5 GB RAM. The low-end servers that were available for use instead contained a single Quad-Core Intel Xeon that ran at 2.6 GHz along with 22.5 GB RAM. The other specific nodes we created were the networking node (Figure 27) and the monitoring node (Figure 28). The idea behind the compute node was to mainly use it for larger computational needs. The other nodes would also be used but the compute would mainly be active in scenarios where larger computational power is needed. The plan behind the network node was for it to be handling the networking/logging over our infrastructure. The monitoring node would have all the tools for in-depth infrastructure and cloud monitoring. Our logs would then be stored onto our SAN.

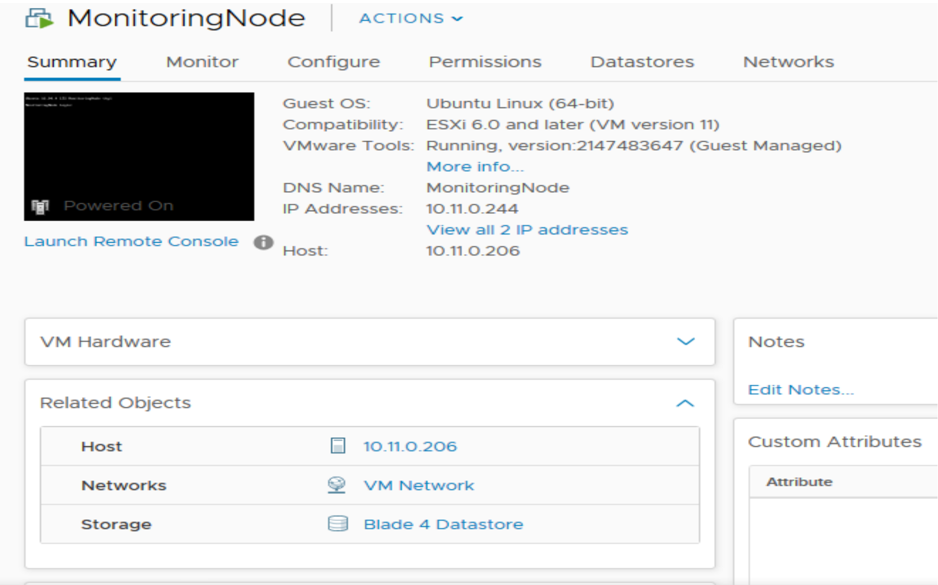


Guest OS: Ubuntu Linux (64-bit)
Compatibility: ESXi 6.0 and later (VM version 11)
VMware Tools: Not running, version:2147483647 (Guest Managed)
More info...
DNS Name: NetworkingNode
IP Addresses:
Host: 10.11.0.217

Related Objects

Host	10.11.0.217
Networks	VM Network
Storage	Blade 5 Datastore

Figure 27. Summary of the Networking Node



Guest OS: Ubuntu Linux (64-bit)
Compatibility: ESXi 6.0 and later (VM version 11)
VMware Tools: Running, version:2147483647 (Guest Managed)
More info...
DNS Name: MonitoringNode
IP Addresses: 10.11.0.244
View all 2 IP addresses
Host: 10.11.0.206

Related Objects

Host	10.11.0.206
Networks	VM Network
Storage	Blade 4 Datastore

Figure 28. Summary of the Monitoring Node

As we were not able to completely finish this part of our environment, so that we could've tested it with many real users. We cannot be sure how well our environment would sustain larger amounts of users or vast amounts of VM creations. With only us in the systems, there were no noticeable problems when the system was running. A more in-depth overview of the HP blade environment can be seen in chapter 12.1.1.

We started out with vSphere EsXi and added vCenter on top of that. Our main idea with the vCenter was to add as many virtual machines as possible in the close-by blades and virtual machines in our “control center”. This would then allow us to control all the virtual machines within our whole cloud environment. With the correct IP address setups both from the vCenter side and our control center, allowed us to fully configure vCenter from the control center which can be seen in Figure 15. We were able to fully configure everything e.g. adding more virtual machines, creating virtual hardware space, removing machines and fully monitor the whole operation. The completed HP-blade environment can be seen in Figure 29.

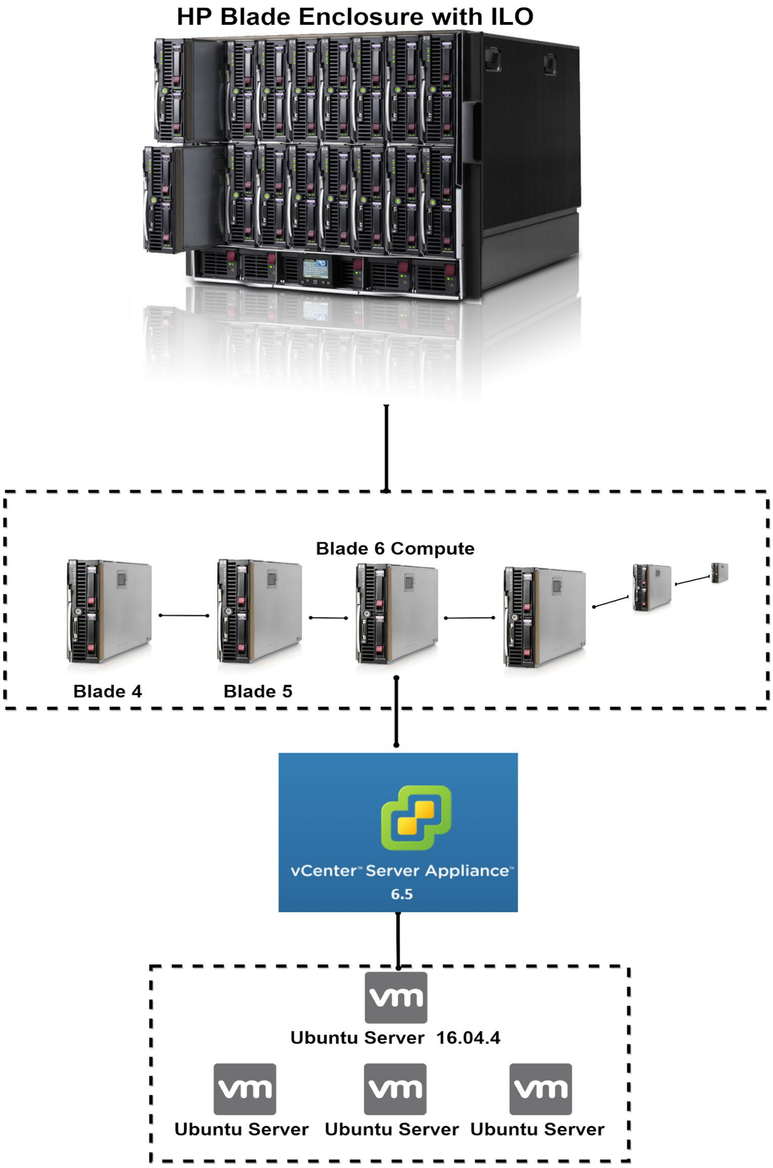
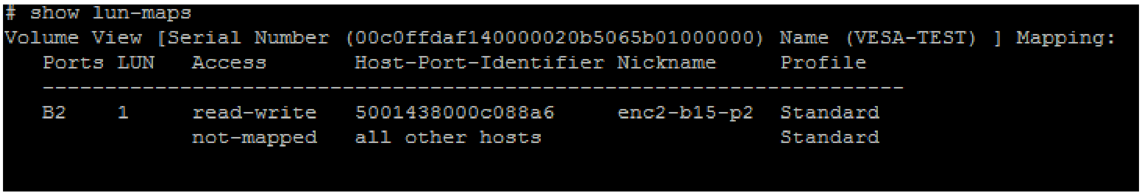


Figure 29. The HP-blade environment

7.5 Fiber-Channel Connections and Storage Arrays

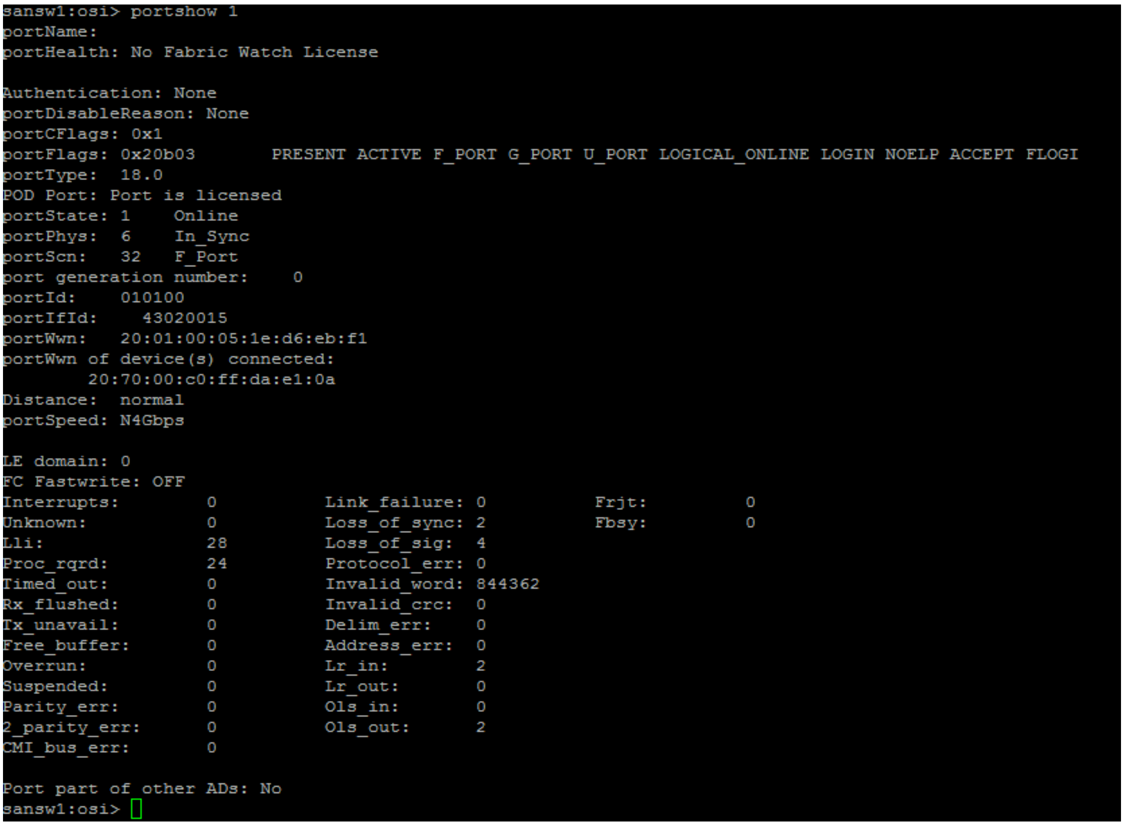
The HP P2000 MSA configuration we have set to allow communication between the blade system and the storage array include a set of multi-mode fiber-channel cables the physical connection as well as a fiber-channel switch to manage the data flow between the two devices. The Brocade 300 is our tool of choice to manage our fiber-channel connections between the two systems, and the HP P2000 MSA was physically connected via the 8GB ports located on the system's rear. Each of the FC/iSCSI combo controller ports route to the same controller, and through this we were able to set up a backup route should the initial connection be lost. In practice, this would be represented with a second fiber-channel management device to act as the failover path should the main connection lose functionality.

The logical unit number (LUN) maps are needed to route information from the iSCSI initiator and the iSCSI target. Our initiators are located within the HP c7000 blade system, and our target was the HP P2000 MSA. We repurposed an existing LUN map named VESA-TEST seen in Figure 30 Following this, we connected the corresponding physical ports mapped to the Brocade switch, and cross-referenced the ports shown. This was done through show commands to find connections within the Brocade switch to match the WWNs between the HP P2000 and the HP c7000 as is shown in Figure 31, and 32.



```
# show lun-maps
Volume View [Serial Number (00c0ffdaf140000020b5065b01000000) Name (VESA-TEST) ] Mapping:
   Ports LUN   Access       Host-Port-Identifier Nickname     Profile
   ---------------------------------------------------------------------
   B2    1     read-write   5001438000c088a6     enc2-b15-p2  Standard
               not-mapped   all other hosts                   Standard
```

Figure 30. the LUN-maps of the Brocade switch



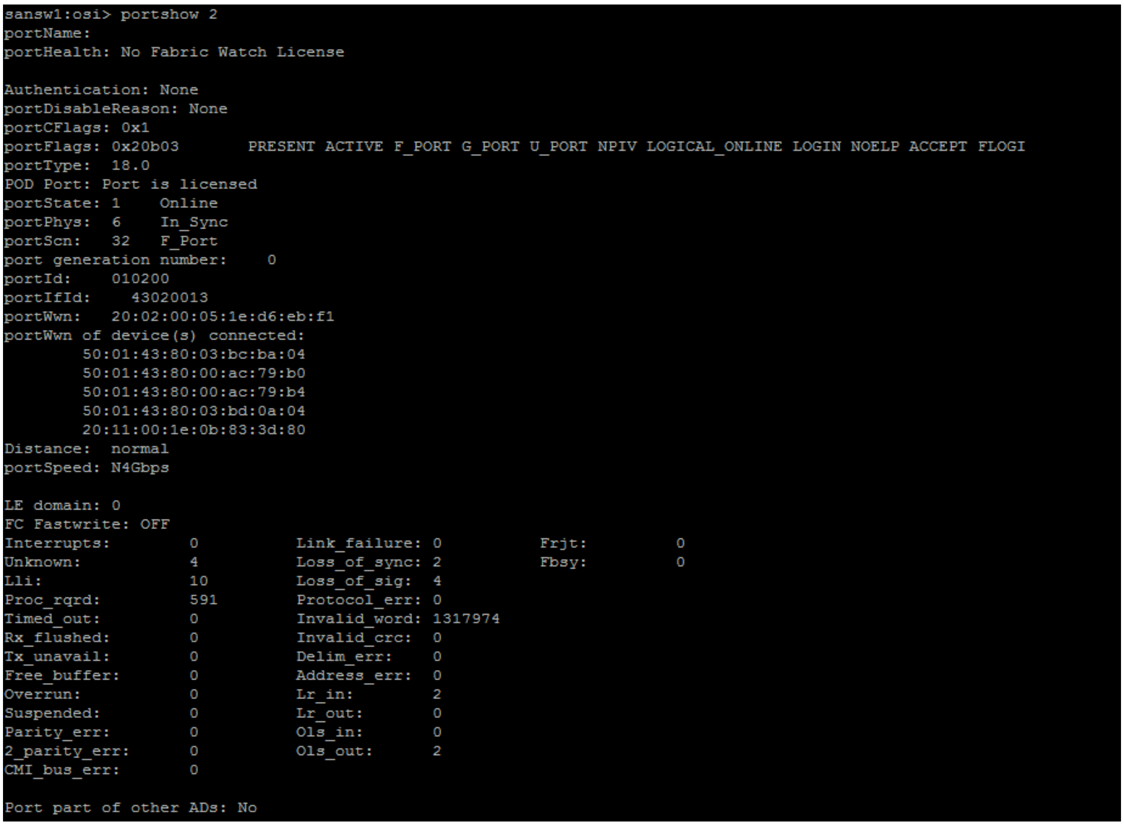
```
sansw1:osi> portshow 1
portName:
portHealth: No Fabric Watch License

Authentication: None
portDisableReason: None
portCFlags: 0x1
portFlags: 0x20b03       PRESENT ACTIVE F_PORT G_PORT U_PORT LOGICAL_ONLINE LOGIN NOELP ACCEPT FLOGI
portType:  18.0
POD Port: Port is licensed
portState: 1    Online
portPhys:  6    In_Sync
portScn:   32   F_Port
port generation number:    0
portId:    010100
portIfId:    43020015
portWwn:   20:01:00:05:1e:d6:eb:f1
portWwn of device(s) connected:
        20:70:00:c0:ff:da:e1:0a
Distance:  normal
portSpeed: N4Gbps

LE domain: 0
FC Fastwrite: OFF
Interrupts:        0          Link_failure: 0          Frjt:         0
Unknown:           0          Loss_of_sync: 2          Fbsy:         0
Lli:               28         Loss_of_sig:  4
Proc_rqrd:         24         Protocol_err: 0
Timed_out:         0          Invalid_word: 844362
Rx_flushed:        0          Invalid_crc:  0
Tx_unavail:        0          Delim_err:    0
Free_buffer:       0          Address_err:  0
Overrun:           0          Lr_in:        2
Suspended:         0          Lr_out:       0
Parity_err:        0          Ols_in:       0
2_parity_err:      0          Ols_out:      2
CMI_bus_err:       0

Port part of other ADs: No
sansw1:osi> 
```

Figure 31. Brocade portshow results 1, SAN WWN



```
sansw1:osi> portshow 2
portName:
portHealth: No Fabric Watch License

Authentication: None
portDisableReason: None
portCFlags: 0x1
portFlags: 0x20b03       PRESENT ACTIVE F_PORT G_PORT U_PORT NPIV LOGICAL_ONLINE LOGIN NOELP ACCEPT FLOGI
portType:  18.0
POD Port: Port is licensed
portState: 1    Online
portPhys:  6    In_Sync
portScn:   32   F_Port
port generation number:    0
portId:    010200
portIfId:    43020013
portWwn:   20:02:00:05:1e:d6:eb:f1
portWwn of device(s) connected:
        50:01:43:80:03:bc:ba:04
        50:01:43:80:00:ac:79:b0
        50:01:43:80:00:ac:79:b4
        50:01:43:80:03:bd:0a:04
        20:11:00:1e:0b:83:3d:80
Distance:  normal
portSpeed: N4Gbps

LE domain: 0
FC Fastwrite: OFF
Interrupts:        0          Link_failure: 0          Frjt:         0
Unknown:           4          Loss_of_sync: 2          Fbsy:         0
Lli:               10         Loss_of_sig:  4
Proc_rqrd:         591        Protocol_err: 0
Timed_out:         0          Invalid_word: 1317974
Rx_flushed:        0          Invalid_crc:  0
Tx_unavail:        0          Delim_err:    0
Free_buffer:       0          Address_err:  0
Overrun:           0          Lr_in:        2
Suspended:         0          Lr_out:       0
Parity_err:        0          Ols_in:       0
2_parity_err:      0          Ols_out:      2
CMI_bus_err:       0

Port part of other ADs: No
```

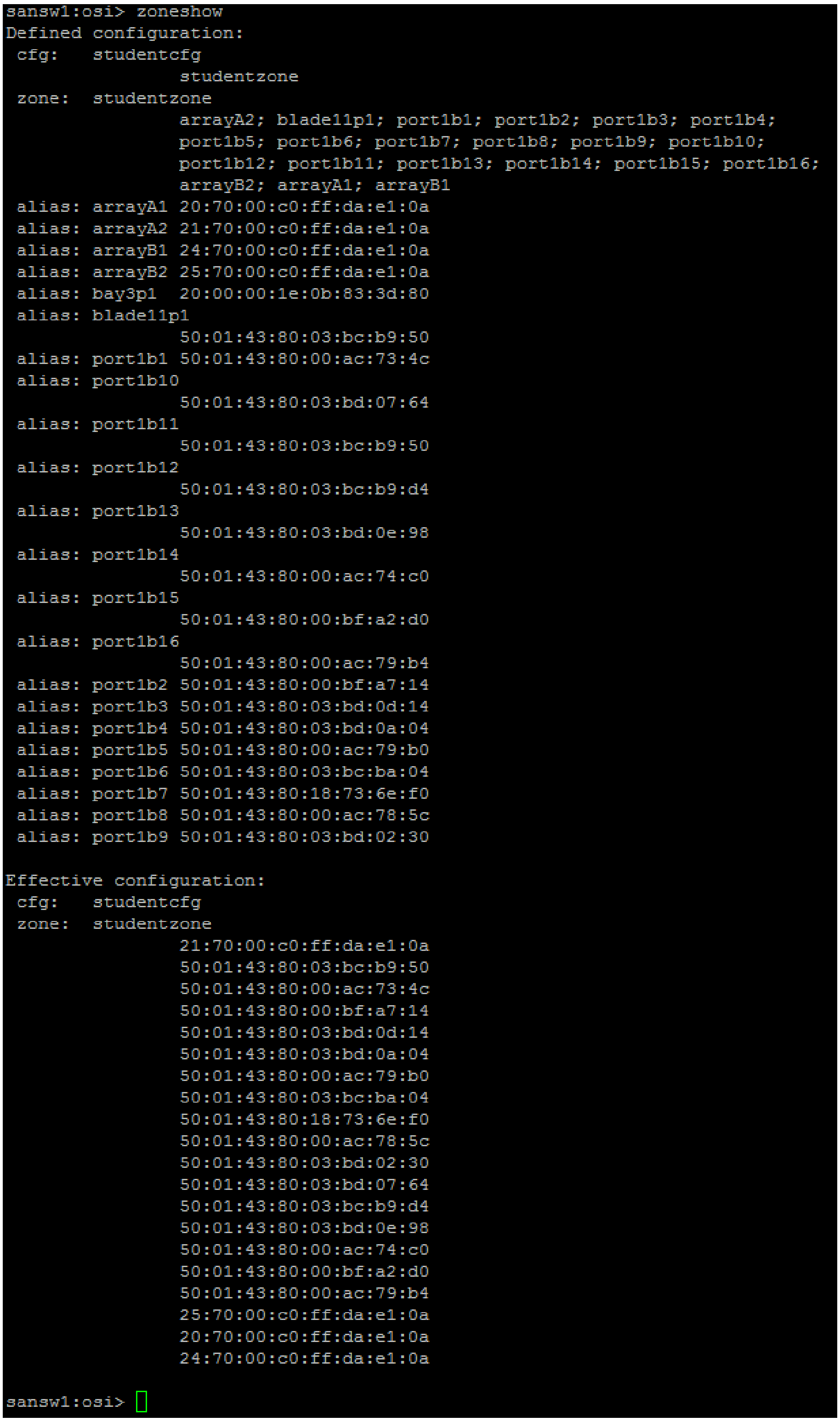
Figure 32. Brocade portshow results 2, enclosure WWNs

The Brocade 300 handled all physical fiber connections between the HP c7000 and the HP P2000, with the FC/iSCSI combo controllers handling the SAN port connection and the Virtual Connect module in the HP c7000 midplane using multi-mode fiber-channel cables.

Assigned to the SAN is the WWN 20:70:00:c0:ff:da:e1:0a, and was given the alias, or a secondary name for easier identification, ‘arrayA1' to denote the physical connection to the controller port A1 on the SAN device. This example is shown in figures below where the commands ‘portshow 1' and ‘portshow 2' invokes a report of the current running configurations for each respective port. Included in the resulting report are the WWNs connected to each end device; the SAN for port 1 as previously discussed and the multiple HP ProLiant BL460c server blades are connected to port 2.

For a connection to be established, WWNs from both devices must share the same “zone”. FC zoning is used to both improve security and simplify networking by partitioning a part of the FC network for use of specified traffic. This is done by first creating a zone using the command ‘zonecreate', followed by a name for the zone and a list of WWNs that will be included within the zone. The results can be viewed using the command ‘zoneshow', which prints a report of the existing zones within the FC switch, including the WWNs used in each zone.

Figure 33 displays both the saved configuration, named ‘studentcfg’ listed in as well as the zone saved within the configuration, aptly named ‘studentzone ‘. Each listed WWN is a named separately using an alias, giving a more understandable naming convention outside of the purely numeric WWNs. The ‘arrayA1’ and similarly named WWNs originate from the HP P2000, referencing the storage array network and the connected port, arrayA1 in particular if we refer to the example earlier. All four connections of the FC/iSCSI combo controller are mapped in the zone, as well as all 16 HP ProLiant BL460c server blades that reside within the HP c7000 server blade enclosure. Although the throughput of the FC ports could support up to 8Gb speeds, our small form-factor pluggable (SFP) transceiver would only output a max of 4Gbps, limiting transfer speeds should large amounts of data need to be moved.



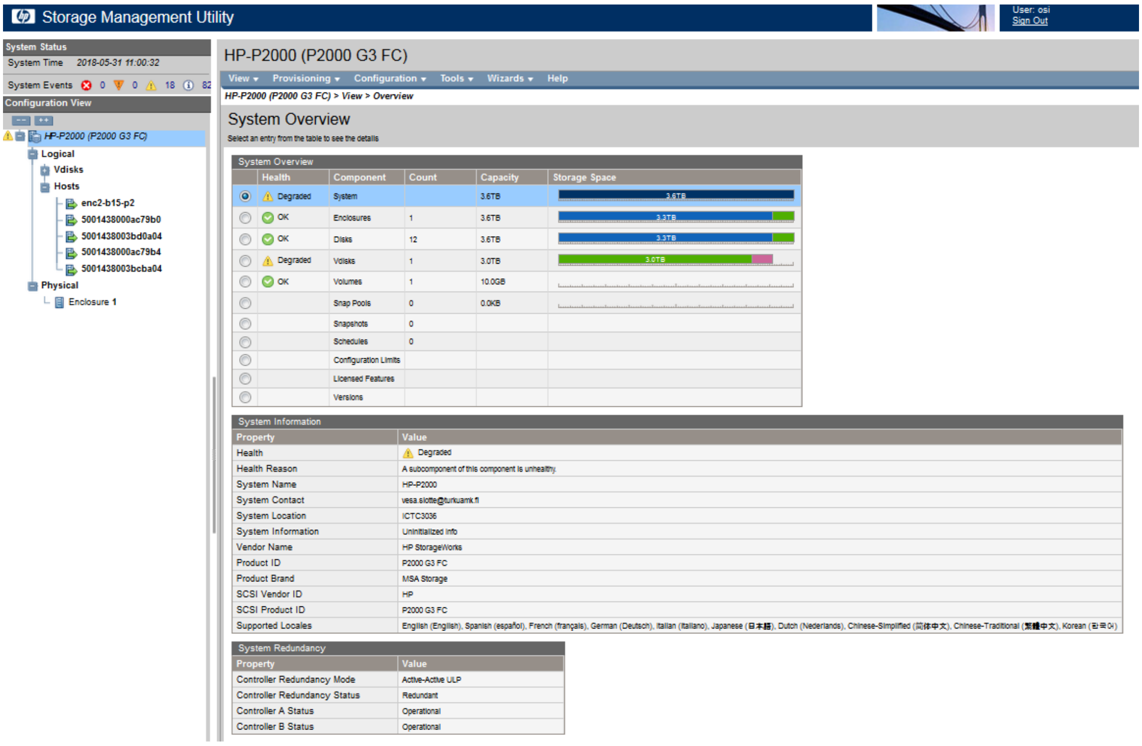
```
sansw1:osi> zoneshow
Defined configuration:
 cfg:   studentcfg
                studentzone
 zone:  studentzone
                arrayA2; blade11p1; port1b1; port1b2; port1b3; port1b4;
                port1b5; port1b6; port1b7; port1b8; port1b9; port1b10;
                port1b12; port1b11; port1b13; port1b14; port1b15; port1b16;
                arrayB2; arrayA1; arrayB1
 alias: arrayA1 20:70:00:c0:ff:da:e1:0a
 alias: arrayA2 21:70:00:c0:ff:da:e1:0a
 alias: arrayB1 24:70:00:c0:ff:da:e1:0a
 alias: arrayB2 25:70:00:c0:ff:da:e1:0a
 alias: bay3p1  20:00:00:1e:0b:83:3d:80
 alias: blade11p1
                50:01:43:80:03:bc:b9:50
 alias: port1b1 50:01:43:80:00:ac:73:4c
 alias: port1b10
                50:01:43:80:03:bd:07:64
 alias: port1b11
                50:01:43:80:03:bc:b9:50
 alias: port1b12
                50:01:43:80:03:bc:b9:d4
 alias: port1b13
                50:01:43:80:03:bd:0e:98
 alias: port1b14
                50:01:43:80:00:ac:74:c0
 alias: port1b15
                50:01:43:80:00:bf:a2:d0
 alias: port1b16
                50:01:43:80:00:ac:79:b4
 alias: port1b2 50:01:43:80:00:bf:a7:14
 alias: port1b3 50:01:43:80:03:bd:0d:14
 alias: port1b4 50:01:43:80:03:bd:0a:04
 alias: port1b5 50:01:43:80:00:ac:79:b0
 alias: port1b6 50:01:43:80:03:bc:ba:04
 alias: port1b7 50:01:43:80:18:73:6e:f0
 alias: port1b8 50:01:43:80:00:ac:78:5c
 alias: port1b9 50:01:43:80:03:bd:02:30

Effective configuration:
 cfg:   studentcfg
 zone:  studentzone
                21:70:00:c0:ff:da:e1:0a
                50:01:43:80:03:bc:b9:50
                50:01:43:80:00:ac:73:4c
                50:01:43:80:00:bf:a7:14
                50:01:43:80:03:bd:0d:14
                50:01:43:80:03:bd:0a:04
                50:01:43:80:00:ac:79:b0
                50:01:43:80:03:bc:ba:04
                50:01:43:80:18:73:6e:f0
                50:01:43:80:00:ac:78:5c
                50:01:43:80:03:bd:02:30
                50:01:43:80:03:bd:07:64
                50:01:43:80:03:bc:b9:d4
                50:01:43:80:03:bd:0e:98
                50:01:43:80:00:ac:74:c0
                50:01:43:80:00:bf:a2:d0
                50:01:43:80:00:ac:79:b4
                25:70:00:c0:ff:da:e1:0a
                20:70:00:c0:ff:da:e1:0a
                24:70:00:c0:ff:da:e1:0a

sansw1:osi> 
```

Figure 33. Brocade zoneshow results

Once connected via mapping LUN maps represented and zoning through the Brocade 300, the active server blades would then appear in the management page for the HP P2000 SAN shown in Figure 34 The configuration view would show both the physical and logical connections represented in Figure 35. The physical connection being the connection to the HP c7000 server blade enclosure midplane, and the logical connections being represented by the each of the mapped servers' blades residing inside the enclosure. This can also be found in the "portshow" results within the Brocade command terminal in Figures 36 and 37.



System Status

System Time 2018-05-31 11:00:32

System Events ✖ 0 ▼ 0 ⚠ 18 ⓘ 82

Configuration View

- HP-P2000 (P2000 G3 FC)
  - Logical
    - Vdisks
    - Hosts
      - enc2-b15-p2
      - 5001438000ac79b0
      - 5001438003bd0a04
      - 5001438000ac79b4
      - 5001438003bcba04
  - Physical
    - Enclosure 1

HP-P2000 (P2000 G3 FC)

HP-P2000 (P2000 G3 FC) > View > Overview

System Overview

Select an entry from the table to see the details

System Overview

	Health	Component	Count	Capacity	Storage Space
●	⚠ Degraded	System		3.6TB	3.6TB
○	✓ OK	Enclosures	1	3.6TB	3.3TB
○	✓ OK	Disks	12	3.6TB	3.3TB
○	⚠ Degraded	Vdisks	1	3.0TB	3.0TB
○	✓ OK	Volumes	1	10.0GB	
○		Snap Pools	0	0.0KB	
○		Snapshots	0		
○		Schedules	0		
○		Configuration Limits			
○		Licensed Features			
○		Versions			

System Information

Property	Value
Health	⚠ Degraded
Health Reason	A subcomponent of this component is unhealthy.
System Name	HP-P2000
System Contact	vesa.slotte@turkuamk.fi
System Location	ICTC3036
System Information	Uninitialized Info
Vendor Name	HP StorageWorks
Product ID	P2000 G3 FC
Product Brand	MSA Storage
SCSI Vendor ID	HP
SCSI Product ID	P2000 G3 FC
Supported Locales	English (English), Spanish (español), French (français), German (Deutsch), Italian (italiano), Japanese (日本語), Dutch (Nederlands), Chinese-Simplified (简体中文), Chinese-Traditional (繁體中文), Korean (한국어)

System Redundancy

Property	Value
Controller Redundancy Mode	Active-Active ULP
Controller Redundancy Status	Redundant
Controller A Status	Operational
Controller B Status	Operational

Figure 34. Overview of the HP-P2000 storage array, accessed via a browser

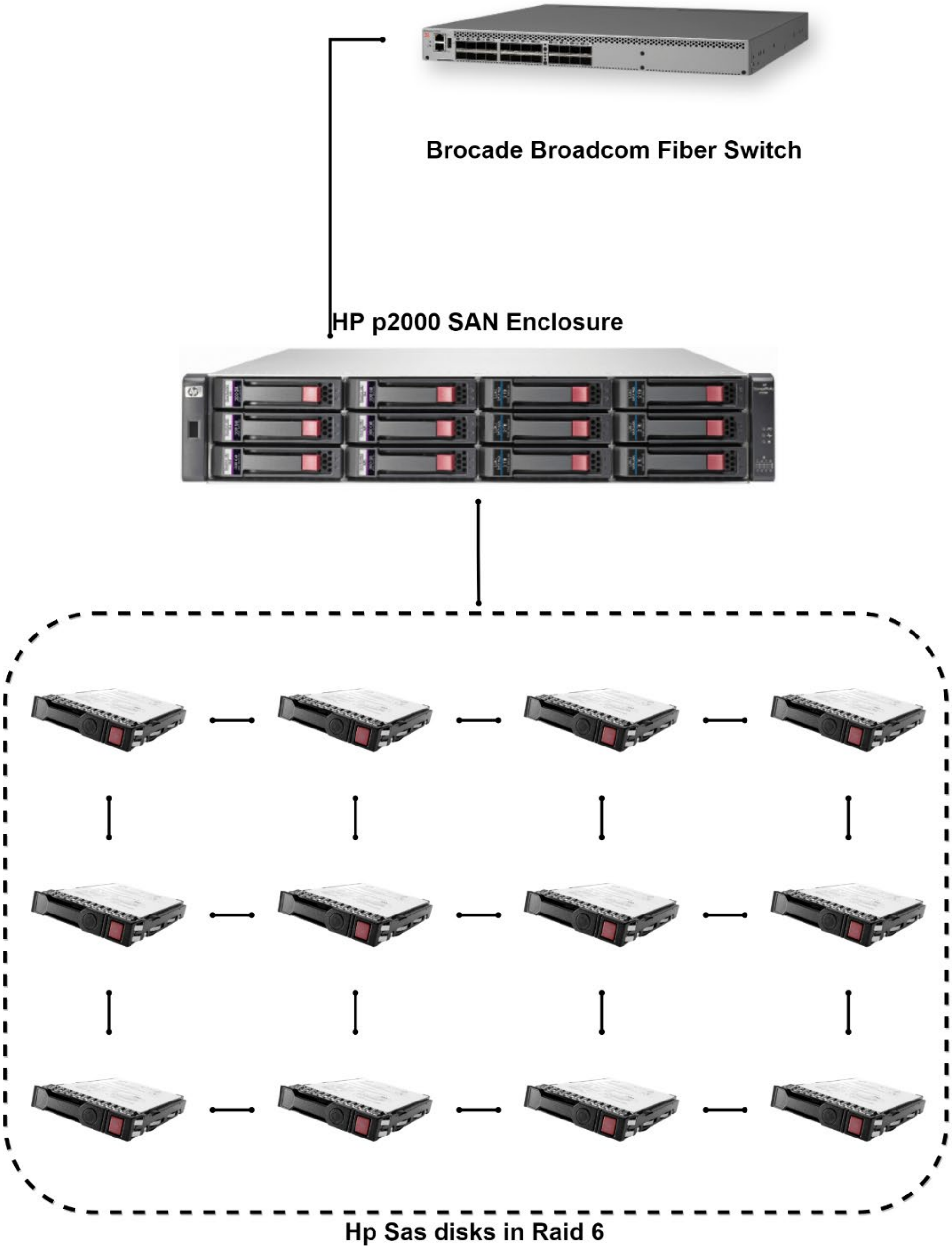
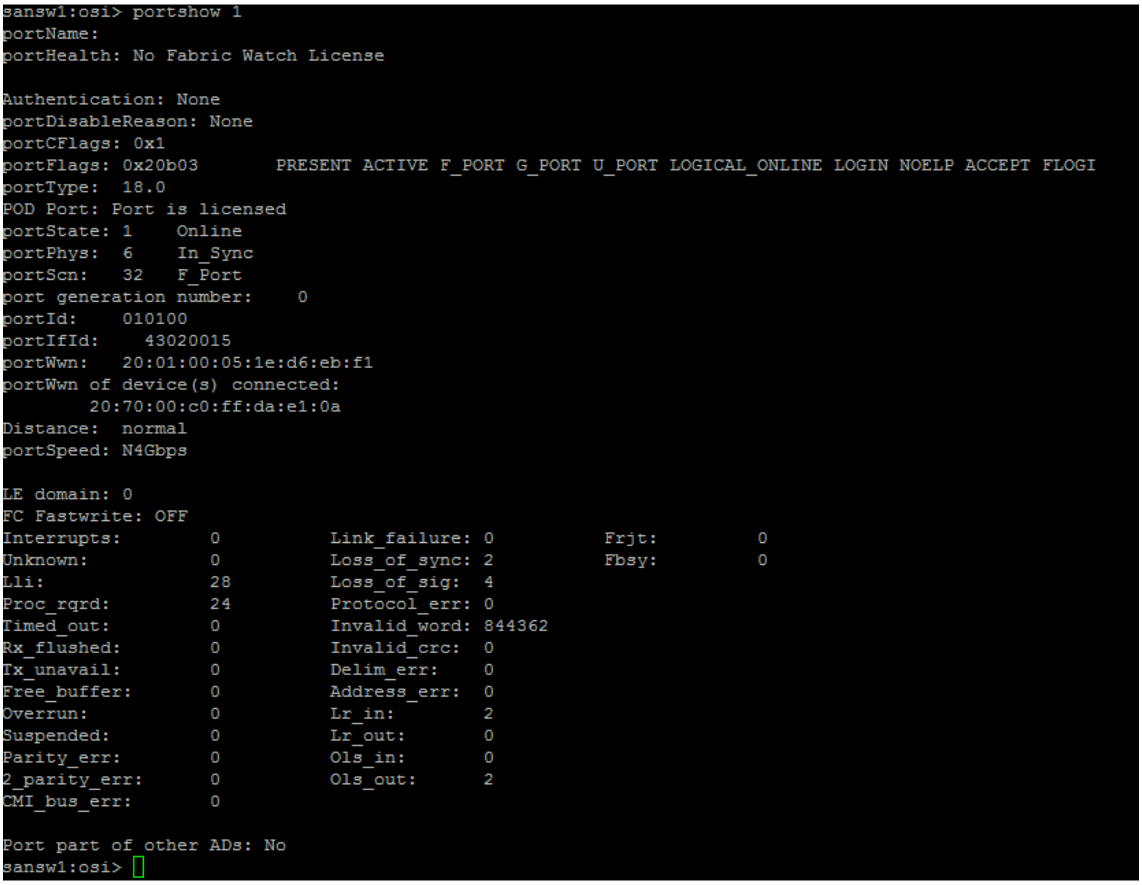


Figure 35. Overview of our SAN infrastructure



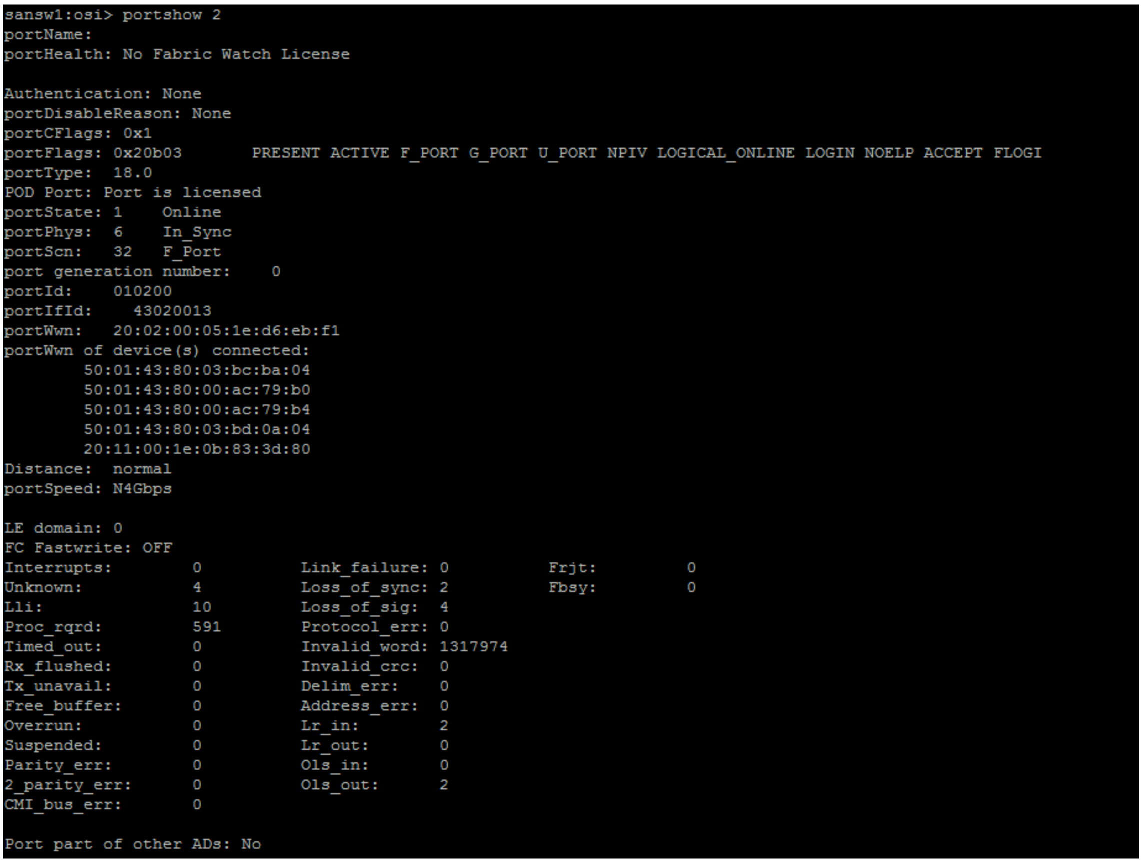
```
sansw1:osi> portshow 1
portName:
portHealth: No Fabric Watch License

Authentication: None
portDisableReason: None
portCFlags: 0x1
portFlags: 0x20b03       PRESENT ACTIVE F_PORT G_PORT U_PORT LOGICAL_ONLINE LOGIN NOELP ACCEPT FLOGI
portType:  18.0
POD Port: Port is licensed
portState: 1    Online
portPhys:  6    In_Sync
portScn:   32   F_Port
port generation number:    0
portId:    010100
portIfId:    43020015
portWwn:   20:01:00:05:1e:d6:eb:f1
portWwn of device(s) connected:
        20:70:00:c0:ff:da:e1:0a
Distance:  normal
portSpeed: N4Gbps

LE domain: 0
FC Fastwrite: OFF
Interrupts:        0          Link_failure: 0          Frjt:         0
Unknown:           0          Loss_of_sync: 2          Fbsy:         0
Lli:               28         Loss_of_sig:  4
Proc_rqrd:         24         Protocol_err: 0
Timed_out:         0          Invalid_word: 844362
Rx_flushed:        0          Invalid_crc:  0
Tx_unavail:        0          Delim_err:    0
Free_buffer:       0          Address_err:  0
Overrun:           0          Lr_in:        2
Suspended:         0          Lr_out:       0
Parity_err:        0          Ols_in:       0
2_parity_err:      0          Ols_out:      2
CMI_bus_err:       0

Port part of other ADs: No
sansw1:osi>
```

Figure 36. Brocade portshow 1 results, WWNs from the Brocade 300 switch



```
sansw1:osi> portshow 2
portName:
portHealth: No Fabric Watch License

Authentication: None
portDisableReason: None
portCFlags: 0x1
portFlags: 0x20b03      PRESENT ACTIVE F_PORT G_PORT U_PORT NPIV LOGICAL_ONLINE LOGIN NOELP ACCEPT FLOGI
portType:  18.0
POD Port: Port is licensed
portState: 1    Online
portPhys:  6    In_Sync
portScn:   32   F_Port
port generation number:    0
portId:    010200
portIfId:    43020013
portWwn:   20:02:00:05:1e:d6:eb:f1
portWwn of device(s) connected:
        50:01:43:80:03:bc:ba:04
        50:01:43:80:00:ac:79:b0
        50:01:43:80:00:ac:79:b4
        50:01:43:80:03:bd:0a:04
        20:11:00:1e:0b:83:3d:80
Distance:  normal
portSpeed: N4Gbps

LE domain: 0
FC Fastwrite: OFF
Interrupts:        0          Link_failure: 0          Frjt:         0
Unknown:           4          Loss_of_sync: 2          Fbsy:         0
Lli:               10         Loss_of_sig:  4
Proc_rqrd:         591        Protocol_err: 0
Timed_out:         0          Invalid_word: 1317974
Rx_flushed:        0          Invalid_crc:  0
Tx_unavail:        0          Delim_err:    0
Free_buffer:       0          Address_err:  0
Overrun:           0          Lr_in:        2
Suspended:         0          Lr_out:       0
Parity_err:        0          Ols_in:       0
2_parity_err:      0          Ols_out:      2
CMI_bus_err:       0

Port part of other ADs: No
```

Figure 37. Brocade portshow 2 results, WWNs from the Brocade 300 switch

8 FOR FUTURE DEVELOPMENT

The completed environment that can be seen in Figure 38, is fully-fledged and functional. Though our project only lasted around five months, we would have wanted to continue building on it. As aforementioned, the hardware used to build our cloud had to be deconstructed and cleaned before the school's official summer vacation started.

Our goal for our infrastructure was to have it momentarily available for customers. We would have needed a web server that would handle a web client. A login server that would handle SSO's and login information. Ultimately a user could be able to login to his/her own environment and choose what kind of virtual machine he or she needs. The user would be able to also choose from a variety of different operating systems and then also their hard-drive needs as well as the specifications of the virtual environment such as: CPU-power, RAM and so forth. This would have allowed the user to have full scaling capabilities in their environment. The scalability on our part would have been the adding of more hardware. The need for a better subscription from VMware would probably come about later. The greatest lack in our hardware at the time was the amount of storage and this would be the first thing that needed expanding. Since neither the equipment nor the software was paid by us this whole scenario would have purely been for testing purposes. The real user would not have paid to access our systems. We would also be in full control of what the user chooses, and we would be able to terminate their environment and user privileges at any given moment.

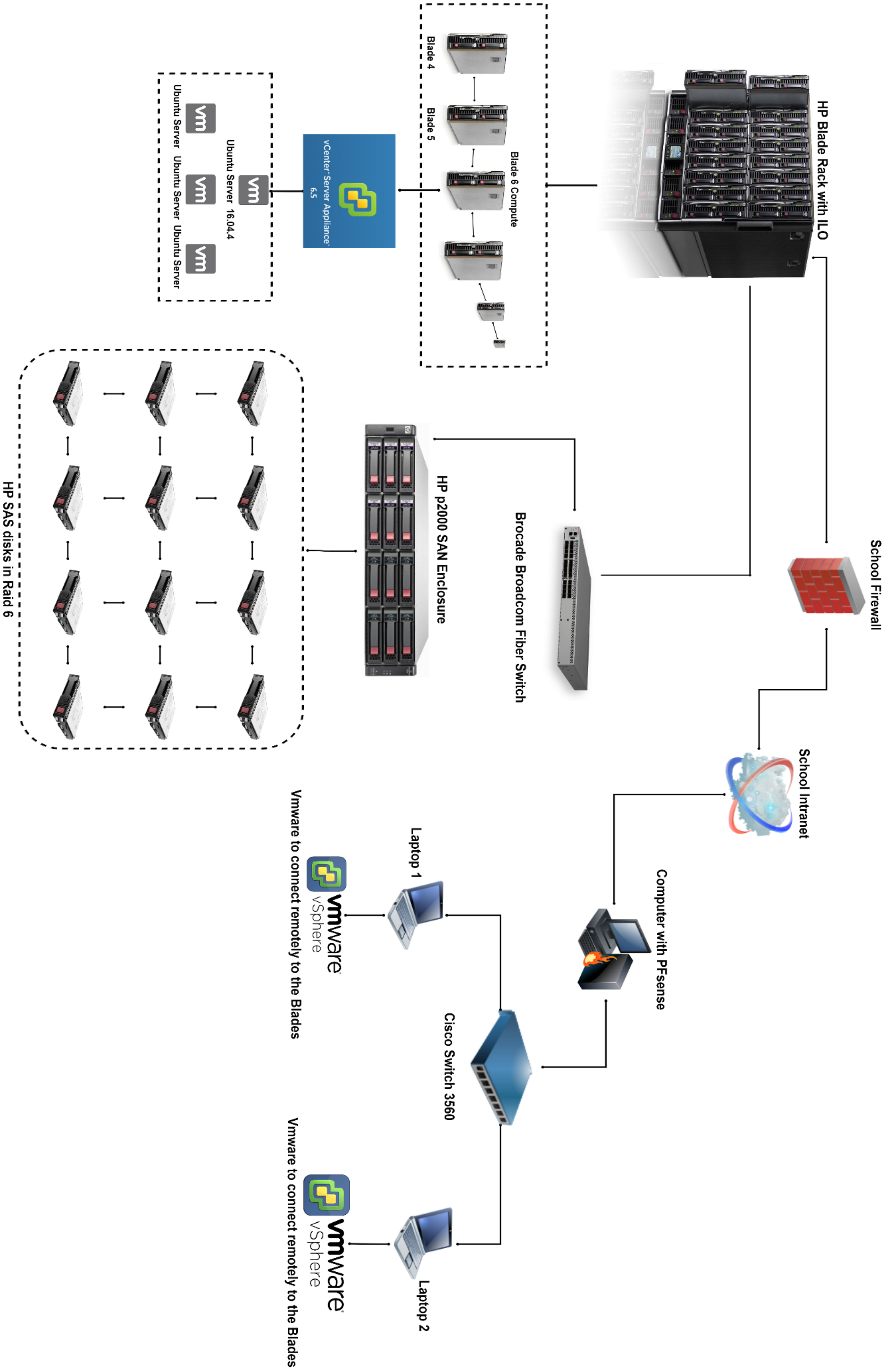


Figure 38. Complete overview of our Cloud infrastructure

8.1 Docker Registry and the OpenStack Rollout

As mentioned in the software section of the Docker introduction, we wanted to utilize this open-source software in our project. Docker was installed on the Fujitsu Celsius W280 and its storage could be found on our SAN-system. Users that would log onto their own environment inside our cloud choose, e.g., a Windows server 2012 R2 ISO. This ISO would then be automatically installed onto a VM. Docker would obtain the information that the user wants this specific OS onto their VM. It would then fetch the ISO from the SAN storage and start the installation. This could have been done by using VMware's template options. In VMware, there is an option to create templates from scratch that specifies certain resources to be used for a specific VM of one's choosing. We would have needed many templates for the many different options a user can choose from. If our cloud environment would have become a business model, our different templates would be priced. The price would be higher on the more demanding VM's that the user would choose.

On the same Fujitsu Celsius W280 we initially used for our Nas4free network area, the storage system was repurposed for another use. Now connected to our management network, we decided to establish this device as our Docker Registry server to provide the Openstack images with containers to reduce the operation resource allocation from operating system overheads. The installation was moved to our management network, so we would be able to make direct configurations easily until the server was set for remote access.

OpenStack was more of a hypothesis than an actual real-world execution. Since OpenStack is both free and can be used to create a virtual environment, we were intrigued by using this in our project. We would have been able to create our VMware environment as a purely virtual entity by the usage of OpenStack. In other words, it would mimic the operations of the VMware. Our main goal for OpenStack was to create it inside our physical infrastructure, the hierarchy of Openstack can be seen in Figure 39.

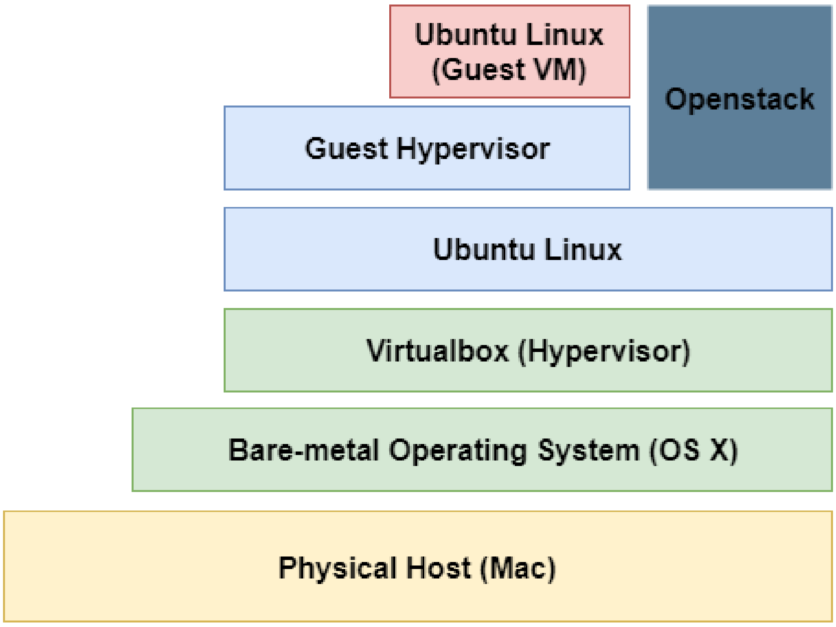


Figure 39. OpenStack Hierarchy

The OpenStack installation that we wished to apply was a Kolla-Ansible-based multi-node rollout that would have various VMs hosting each node, giving each node a dynamically allocated resource dependent on the need of the cloud. Each environment would be client-managed as its own provisioned virtual space with multi-tenant storage leasing, with scalability in mind. The deployment would traditionally be spread across different servers, each built to handle a specific task. In this format, however, each node is virtual and would be hosted in one structure for easier physical management while having the adaptability of a hot-swappable virtual environment with image management systems that runs in both staging and production.

9 CONCLUSIONS

Cloud computing is changing from day to day and has been since it was introduced to the public. New inventions related to building the infrastructure and the software handling are introduced weekly or even daily. Our infrastructure was a proof of concept that a functioning cloud environment can be created with miscellaneous hardware and software from different vendors. Since the introduction of cloud computing and its rocketing rise in demand, we were motivated to immerse ourselves into the cloud world and work out a way to create one of our own.

From start to finish, we realized that in developing our cloud we would run into multiple obstacles that would impede our progress. This resulted in the completion of our base infrastructure without the creation of the provisional OpenStack environments. From this, we found that the results were inconclusive in terms of applying a provisional cloud environment as we were unable to fully develop the project to its planned final iteration. What we did learn as cloud developer novices is the heavy resource demand in what it takes to run and maintain a cloud, without touching the baseline costs of power, cooling, space, licensing, and compliance requirements.

As this project was based on a specific set of equipment and tools, the results are limited to the scope of the lab where the project was carried out. Our use case for the results would be for utilization and/or expansion of the current cloud network or for courses geared towards creating/maintaining cloud networks. This would be performed with the intent of fostering and nurturing knowledge in an industry focused on managing large amounts of information. By now our infrastructure and the way it was built is slightly outdated. The way the technology-industry is handled today requires the utmost knowledge of the most recent creations and innovations. If the resources would have been available for us, we would have made our infrastructure hyper-converged. This means that our whole physical system would sit inside one larger case and the storage would be scrambled and saved thrice. This is something large companies are now doing. We know that our environment and the knowledge we gathered during this project will aid us in coming work and projects.

REFERENCES

Ahrens, N. & Zhou, Y. January 2013. China's Competitiveness Myth, Reality, and Lessons for the United States and Japan, [Online]. Center for Strategic & International Studies. Available at: https://csis-prod.s3.amazonaws.com/s3fs-public/130129_competitiveness_Lenovo_casestudy_Web_4.pdf Accessed 16 Apr. 2019.

Armbrust, M., Fox, A., Griffith, R., Joseph, A., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A. Stoica, I., Zaharia, M. April 2010. A View of Cloud Computing. Available at: https://cacm.acm.org/magazines/2010/4/81493-a-view-of-cloud-computing/fulltext Accessed 16 Apr. 2019.

Bipin, G. 24 August 2012. *Difference between vSphere, ESXi and vCenter*. Available at: http://www.mustbegeek.com/difference-between-vsphere-esxi-and-vcenter/ Accessed 16 Apr. 2019.

Cisco Catalyst 3560 Series 5 August 2014. Switches Data Sheet. Available at: https://www.cisco.com/c/en/us/products/collateral/switches/catalyst-3560-series-switches/product_data_sheet09186a00801f3d7d.html Accessed 16 Apr. 2019

Docker FAQ Available at: https://docs.docker.com/engine/faq/#what-does-docker-technology-add-to-just-plain-lxc Accessed 16 Apr. 2019.

Faatz, D. 12 March 2018. Best Practices for Cloud Security. [Online] Insights.sei.cmu.edu. Available at: https://insights.sei.cmu.edu/sei_blog/2018/03/best-practices-for-cloud-security.html Accessed 16 Apr. 2019.

Fox, R. & Hao, W. 2018 Internet Infrastructure Networking, Web Services, and Cloud Computing (CRC Press). pp.462-464.

Fujitsu corporate history. Available at: http://www.fujitsu.com/global/about/corporate/history/ Accessed 16 Apr. 2019.

Hewlett Packard Enterprise. 2018. Architecture and Technologies in the HPE BladeSystem c7000 Enclosure. Available at: https://h20195.www2.hpe.com/v2/GetDocument.aspx?docname=4AA4-8125ENW&doctype=Technical%20white%20paper&doclang=EN_US&searchquery=&cc=us&lc=en Accessed 16 Apr. 2019.

HP. 17 November 2008, Version 13, HP BladeSystem c-Class Interconnect quick specs. Available at: https://shop.evry.com/pdf/8C45A7CC-BA8F-4922-B37E-A86C96337BFB.pdf Accessed 16 Apr. 2019.

HPE 3 December 2018. HPE blade-systems quick specs. Available at: https://h20195.www2.hpe.com/v2/GetDocument.aspx?docname=c04229580&doctype=quickspecs&doclang=EN_US&searchquery=&cc=us&lc=en pdf Accessed 16 Apr. 2019.

Keepassx.org. 2018. KeePassX. [online] Available at: https://www.keepassx.org/ Accessed 16 Apr. 2019.

Lewis, R. *Cisco Systems*. Available at: https://www.britannica.com/topic/Cisco-Systems-Inc Accessed 16 Apr. 2019.

Matsumoto, C. 8 July 2017. *Brocade Spins Off SDN Controller Into Lumina* LightReading.com [online]. Available at: https://www.lightreading.com/automation/brocade-spins-off-sdn-controller-into-lumina/d/d-id/735228 Accessed 16 Apr. 2019.

Miller, M. 11 January 2018. Cloud Security Best Practices | BeyondTrust. [Online]. Available at: https://www.beyondtrust.com/blog/cloud-security-best-practices/ Accessed 16 Apr. 2019.

Mukherjee, S. & Baker, L. 2016. *Chipmaker Broadcom to buy network gear maker Brocade for $5.5 billion*. [online] reuters.com. Available at: https://www.reuters.com/article/us-brocade-commns-m-a-broadcom-idUSKBN12X1A8 Accessed 16 Apr. 2019.

Openstack history. 28 May 2018. Available at: https://docs.openstack.org/project-team-guide/introduction.html Accessed 16 Apr. 2019.

Raid-calculator.com. 2015. RAID Types (Levels) Reference. [online] Available at: http://www.raid-calculator.com/raid-types-reference.aspx Accessed 16 Apr. 2019.

Salam, A., Ul Haq, S. & Gilani, Z. 2015. Deploying and Managing a Cloud Infrastructure: Real World Skills for the CompTIA Cloud+ Certification and Beyond: CV0-001, John Wiley & Sons, Incorporated, Somerset. pp.2-3.

Sun, Y., Zhang, J., Xiong, Y. & Zhu, G. 14 July 2014. Data Security and Privacy in Cloud Computing. Data Security and Privacy in Cloud Computing, [Online]. 10 issue 7, 1. Available at: http://journals.sagepub.com/doi/full/10.1155/2014/190903 Accessed 16 Apr. 2019.

Techopedia.com. What is VMware? Reference. [online] Available at: https://www.techopedia.com/definition/16053/vmware Accessed 16 Apr. 2019.

Vacca, R. J. 2016. Security in the Private Cloud, CRC Press, Boca Raton. pp 169-171

Vladan, S. 11 January 2019. *What is The Difference between VMware vSphere, ESXi and vCenter*. Available at: https://www.vladan.fr/what-is-the-difference-between-vmware-vsphere-esxi-and-vcenter/ Accessed 16 Apr. 2019.

APPENDICES

Appendix 1: Docker Installation

- We installed docker registry on the 172.26.36.70 virtual machine.

- We start by installing EsXi 6.5 on our Fujitsu physical computer

- After this a connection between VCenter and EsXi is made.

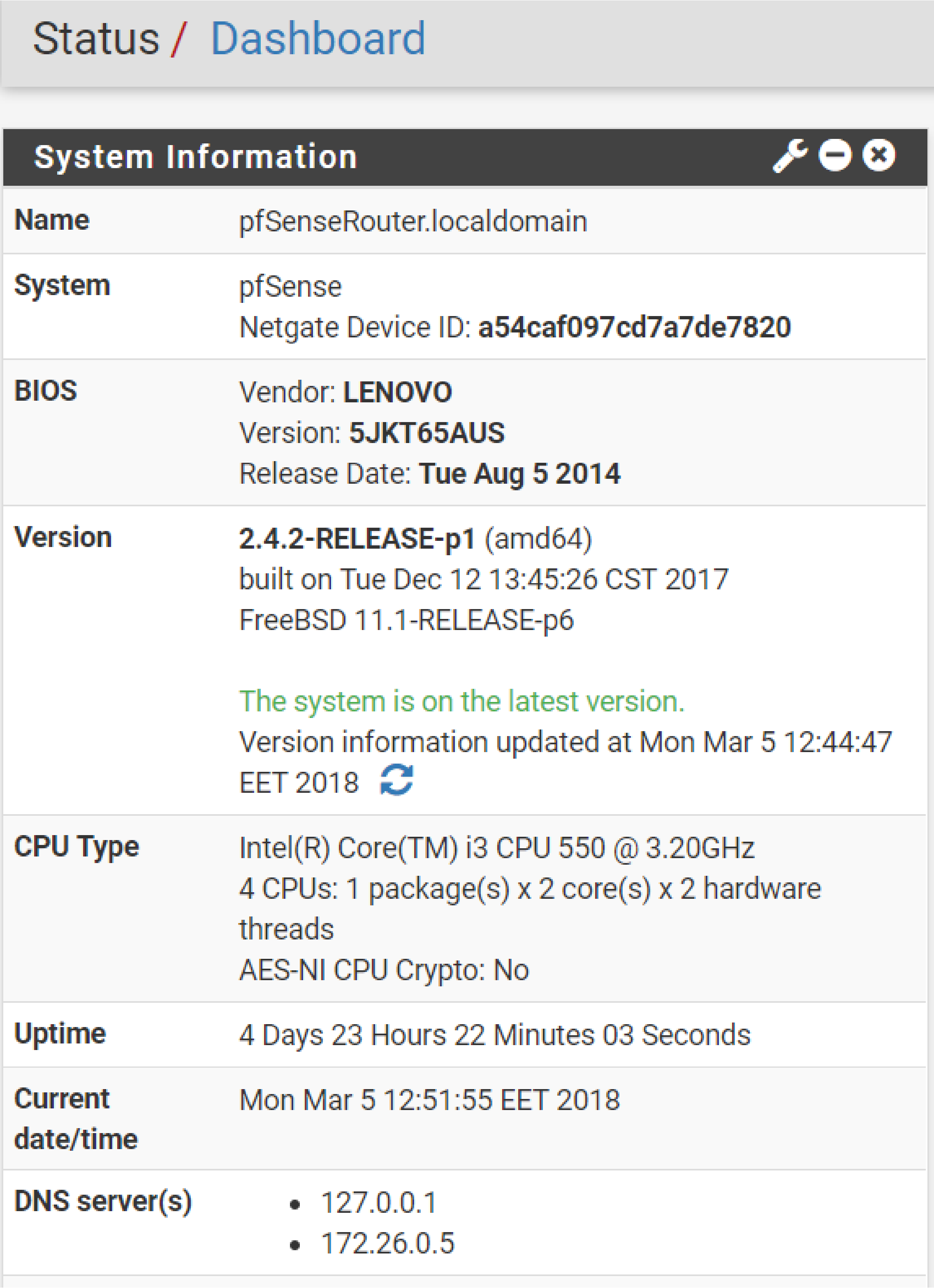
- A VM is then added via the VCenter and Ubuntu server 16.04 is installed on top of it.

- Now Kolla-Ansible is able to be installed on top of the Ubuntu server 16.04 by using the
guide: https://docs.openstack.org/kolla-ansible/latest/user/quickstart.html

- As Kolla-Ansible is installed we went on to prepare the Physical Fujitsu computer for Docker installation. The requirements needed can be found here: https://docker-curriculum.com/#prerequisites

- The guide to install docker: https://docs.docker.com/install/linux/docker-ce/ubuntu/#install-using-the-repository

Appendix 2: Pfsense system information

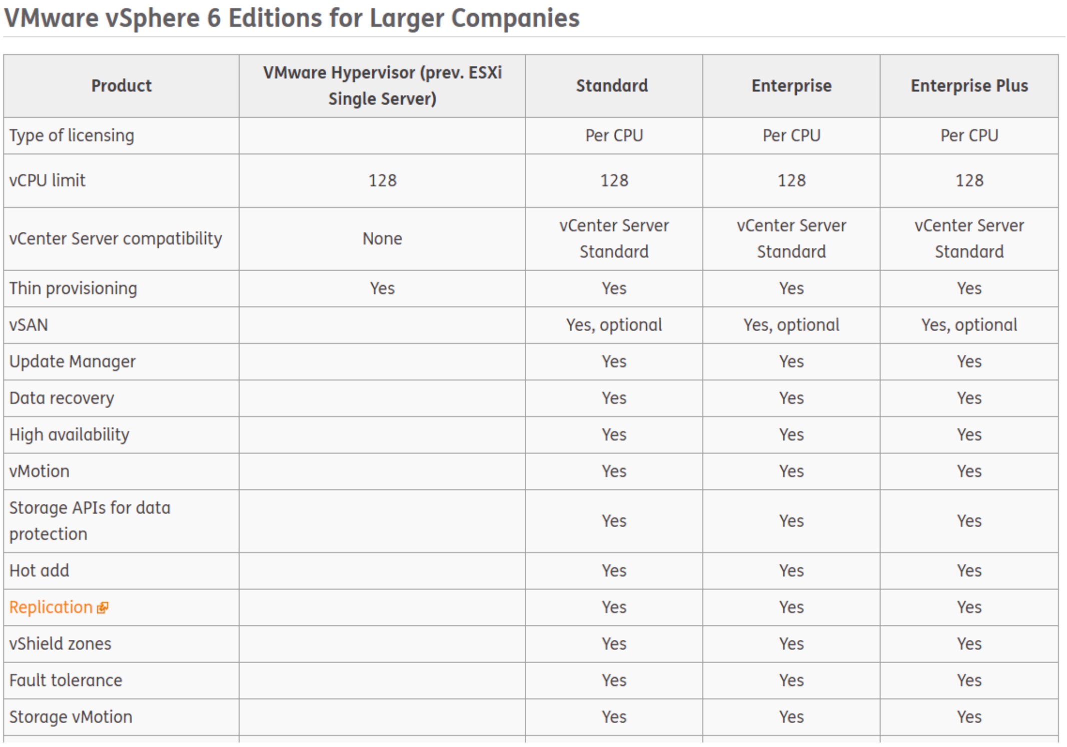


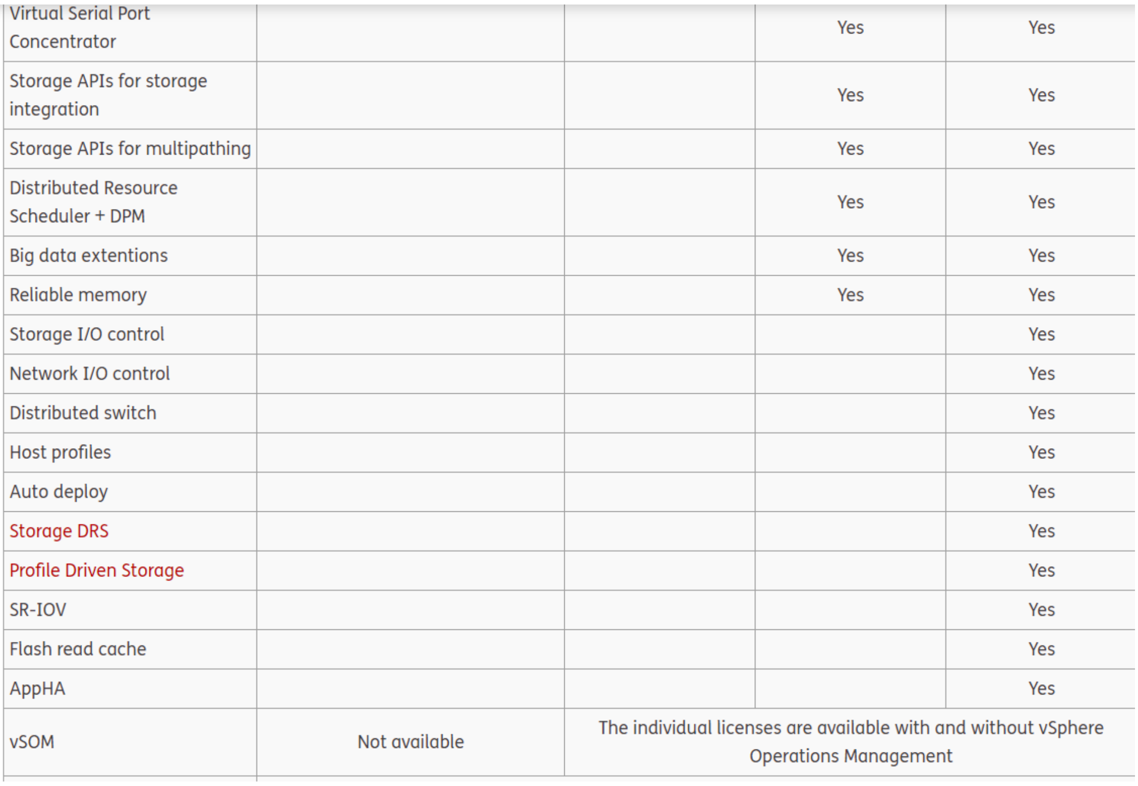
Status / Dashboard

System Information

Name	pfSenseRouter.localdomain
System	pfSense Netgate Device ID: **a54caf097cd7a7de7820**
BIOS	Vendor: **LENOVO** Version: **5JKT65AUS** Release Date: **Tue Aug 5 2014**
Version	**2.4.2-RELEASE-p1** (amd64) built on Tue Dec 12 13:45:26 CST 2017 FreeBSD 11.1-RELEASE-p6 The system is on the latest version. Version information updated at Mon Mar 5 12:44:47 EET 2018
CPU Type	Intel(R) Core(TM) i3 CPU 550 @ 3.20GHz 4 CPUs: 1 package(s) x 2 core(s) x 2 hardware threads AES-NI CPU Crypto: No
Uptime	4 Days 23 Hours 22 Minutes 03 Seconds
Current date/time	Mon Mar 5 12:51:55 EET 2018
DNS server(s)	• 127.0.0.1 • 172.26.0.5

Appendix 3: Differences between the vSphere licenses





VMware vSphere 6 Editions for Larger Companies

Product	VMware Hypervisor (prev. ESXi Single Server)	Standard	Enterprise	Enterprise Plus
Type of licensing		Per CPU	Per CPU	Per CPU
vCPU limit	128	128	128	128
vCenter Server compatibility	None	vCenter Server Standard	vCenter Server Standard	vCenter Server Standard
Thin provisioning	Yes	Yes	Yes	Yes
vSAN		Yes, optional	Yes, optional	Yes, optional
Update Manager		Yes	Yes	Yes
Data recovery		Yes	Yes	Yes
High availability		Yes	Yes	Yes
vMotion		Yes	Yes	Yes
Storage APIs for data protection		Yes	Yes	Yes
Hot add		Yes	Yes	Yes
Replication		Yes	Yes	Yes
vShield zones		Yes	Yes	Yes
Fault tolerance		Yes	Yes	Yes
Storage vMotion		Yes	Yes	Yes
Virtual Serial Port Concentrator			Yes	Yes
Storage APIs for storage integration			Yes	Yes
Storage APIs for multipathing			Yes	Yes
Distributed Resource Scheduler + DPM			Yes	Yes
Big data extentions			Yes	Yes
Reliable memory			Yes	Yes
Storage I/O control				Yes
Network I/O control				Yes
Distributed switch				Yes
Host profiles				Yes
Auto deploy				Yes
Storage DRS				Yes
Profile Driven Storage				Yes
SR-IOV				Yes
Flash read cache				Yes
AppHA				Yes
vSOM	Not available	The individual licenses are available with and without vSphere Operations Management		

Benjamin Bayer 2019. VMware vSphere 6 Editions for Smaller Environments. Available at:

https://www.thomas-krenn.com/en/wiki/VMware_vSphere_6_Editions_Overview [Accessed 1 Apr 2019]