Anglia Ruskin University

DATA CENTER AND CLOUD INFRASTRUCTURE

MOD005714

SID: 1826585

Table of Contents

PART C

PART A

Introduction:

This report is intended to guide PwnMe.Com with resource recommendations to expand their existing infrastructure. The following sections consist of calculations to rectify if the resources are under or over-utilized and recommendations have been made accordingly.

About the Infrastructure:

The current infrastructure set up consists of a hypervisor cluster it consists of 10 identical physical compute systems. Available processing power and memory per physical compute system in a cluster are equal to 19.2 GHz and 64 GB respectively. Out of 10 two systems are passive compute systems to absorb unto two system failures without any impact on the performance of services.

The calculation for the existing capacity of the infrastructure:

Total number of compute systems = 10

Number of active compute systems = 8

Number of idle compute systems = 2

Processing power of each physical compute system = 19.2 GHz

Memory capacity of each physical compute system = 64GB

It is known that the idle systems are expected to be utilized only if there is a failure with the current active systems. Hence the calculation for the overall processing power and the memory capacity for the existing infrastructure is shown below:

Overall processing power for the existing infrastructure =

Processing power of each physical compute system * Number of active compute systems

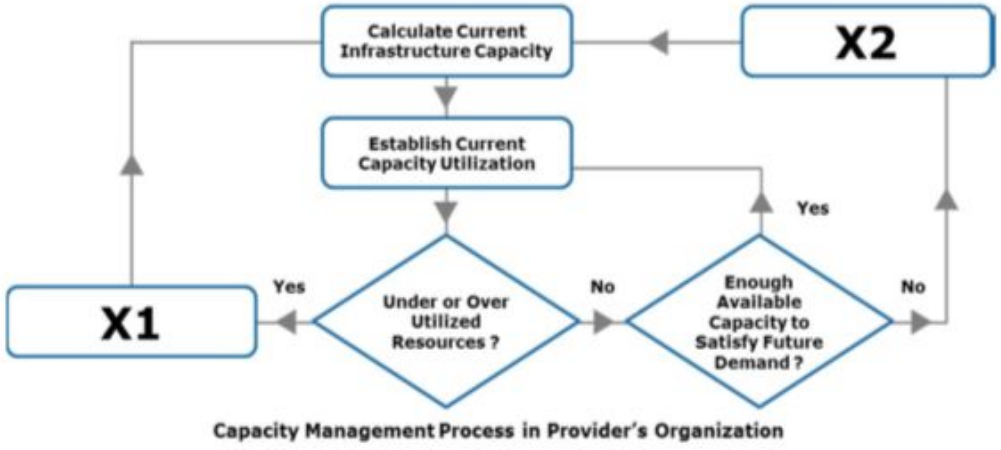
i.e, 19.2 * 8 = 153.6 GHz.

Overall Memory capacity for the existing infrastructure =

Memory capacity of each physical compute system * Number of active compute systems

i.e, 64 * 8 = 512 GB.

The working of the capacity management process is shown in the figure below:



Capacity Management Process in Provider's Organization

To determine the X1 and X2 capacity management activities from the figure above for the following two cases, the minimum and maximum capacities of the compute systems need to be calculated. It is given that the thresholds for overutilization and underutilization of resources are 70 percent and 40 percent utilization of total resource capacity respectively. Therefore,

Minimum capacity to calculate the under-utilized resources:

Processing power = Overall processing power for the existing infrastructure * 0.4

i.e, 153.6 * 0.4 = 61.44 GHz

Memory capacity = Overall Memory capacity for the existing infrastructure * 0.4

i.e, 512 * 0.4 = 204.8 GB

Maximum capacity to calculate the under-utilized resources:

Processing power = Overall processing power for the existing infrastructure * 0.7

i.e, 153.6 * 0.7 = 107.52 GHz

Memory capacity = Overall Memory capacity for the existing infrastructure * 0.7

i.e, 512 * 0.7 = 358.4 GB

Case 1: Processing power already allocated to services from the resource pool is equal to 48 GHz and memory capacity already allocated to services from the resource pool is equal to 156 GB.

In this case, it is clear from the calculation that both the processing power and the memory are under-utilized. By calculations, releasing 3 active systems and 1 passive system can be recommended without degrading the performances.

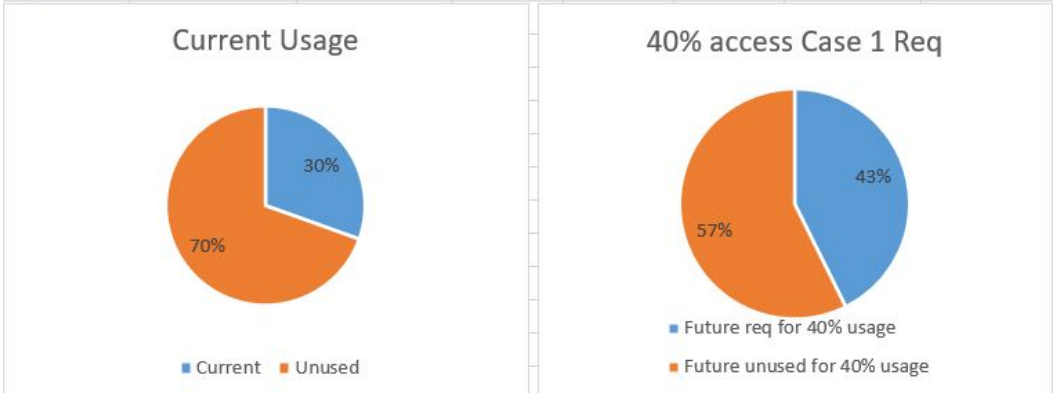
So considering having 5 compute systems:

Processing power : 19.2 * 5 = 96 GHz

Memory Capacity: 64 * 5 = 320 GB

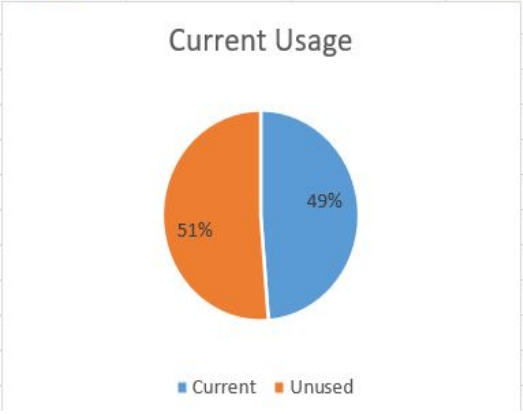
It is seen that for the current compute systems and case 1 requirements, the system is underutilized with only 30% of the systems being used.

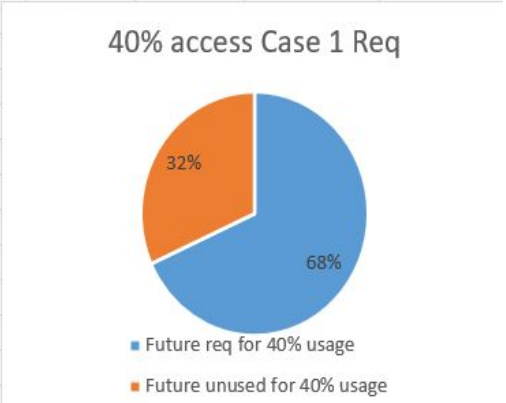
	Server inactive	Servers active	Current	Unused	Total	Future req for 40% usage	Future unused for 40% usage
Memory Capacity	2	8	156	356	512	218.4	293.6
Processing power			48	105.6	153.6	67.2	86.4



So, releasing the active compute systems to 5 will use 51% of each server which is within the performance limits. Also, releasing the passive systems to 1 is also a good idea as the active systems will be able to provide processing and memory power if required.

	Server inactive	Servers active	Current	Unused	Total	Future req for 40% usage	Future unused for 40% usage
Memory Capacity	1	5	156	164	320	218.4	101.6
Processing power			48	48	96	67.2	28.8





Case 2: Processing power already allocated to services from the resource pool is equal to 110 GHz and memory capacity already allocated to services from the resource pool is equal to 360 GB

In this case, it is clear from the calculation that both the processing power and the memory are over-utilized. This indicates that adding 3 more compute systems will satisfy future demands and avoids performance degrade.

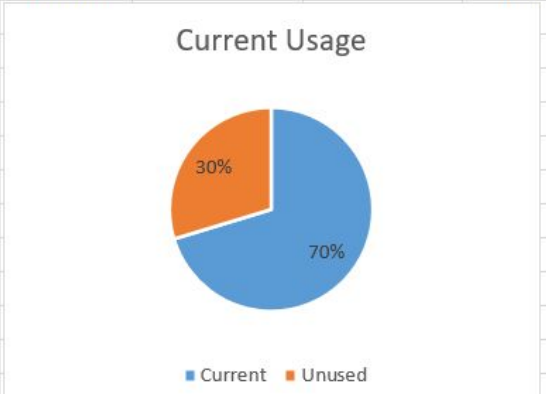
So, considering having 11 compute systems:

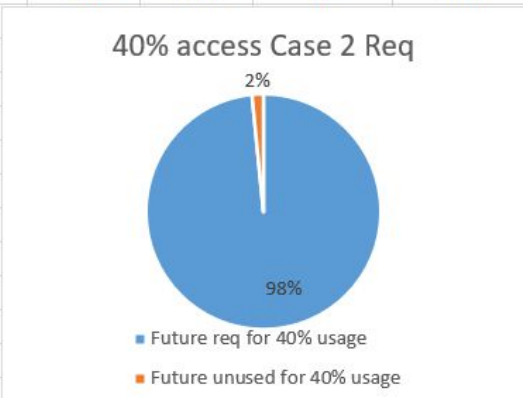
Processing power : 19.2 * 11 = 211.2 GHz

Memory Capacity: 64 * 11 = 704 GB

It is seen that for the current compute systems and case 2 requirements, the system is over utilised with 70% of the systems being used.

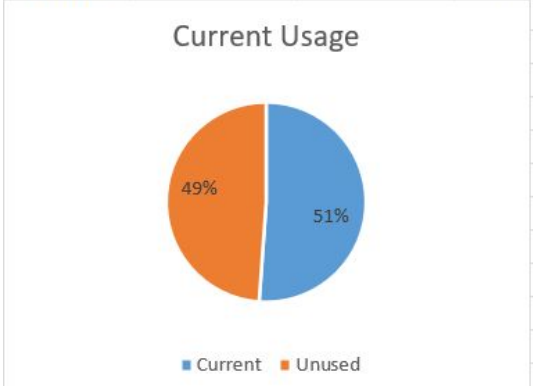
	Server inactive	Servers active	Current	Unused	Total	Future req for 40% usage	Future unused for 40% usage
Memory Capacity	2	8	360	152	512	504	8
Processing power			110	43.6	153.6	154	-0.4

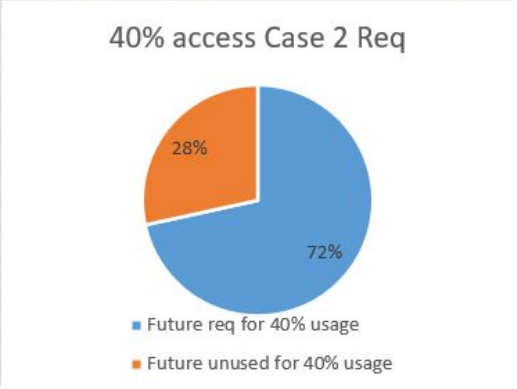




So to reduce the overload on each server, increasing active compute systems to 11 will use 51% of each server which is within the performance limits.

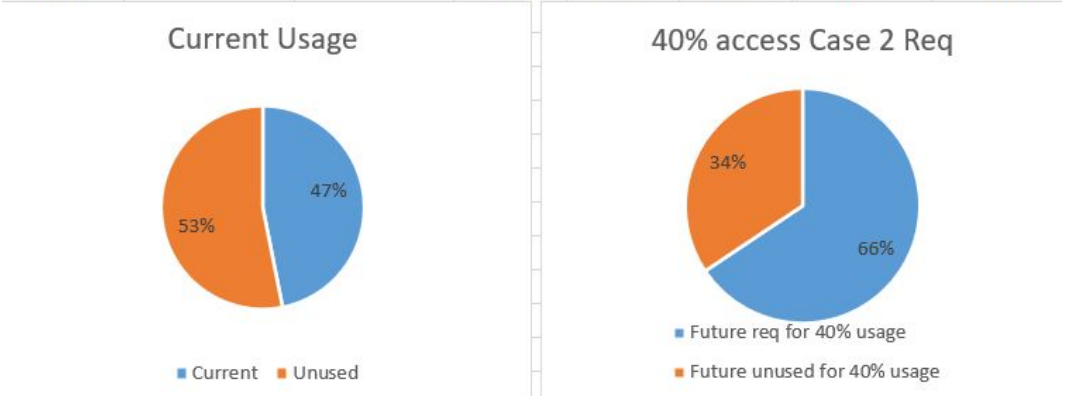
	Server inactive	Servers active	Current	Unused	Total	Future req for 40% usage	Future unused for 40% usage
Memory Capacity	2	11	360	344	704	504	200
Processing power			110	101.2	211.2	154	57.2





Considering future recommendation with the current configuration and assuming 40% of the excess of current peak demand, it would be good enough to use 12 compute systems as the utilization would be below overutilization and the performance will not reduce in the long term.

	Server inactive	Servers active	Current	Unused	Total	Future req for 40% usage	Future unused for 40% usage
Memory Capacity	2	12	360	408	768	504	264
Processing power			110	120.4	230.4	154	76.4



PART B

1. Introduction

The following report will provide the identification of current hypervisor vulnerabilities and its control measures of potential live exploits. It also provides a detailed explanation of future vulnerabilities that may cause along with its control measures to protect against unknown attacks.

2. Hypervisor Vulnerabilities

A Hypervisor or a Virtual Machine Manager (VMM) is a software virtualization technique that divides and allocates the resources on various hardware, thus allowing multiple guests operating systems (OS) to run on a single host system at the same time. There are two types of Hypervisors:

1. Bare Metal Hypervisor
2. Hosted Hypervisor.

A Hypervisor is used for placing customer data to separate resources from a multi-tenanted system and trusting the providers with administration privileges to their system; compromising this hypervisor with a malware attack or gaining root permission for an attacker would allow full access to the shared memory of the physical machine and therefore the content of all the guest virtual machines (VMs) running on the physical platform. The below-shown chart describes how severe are the hypervisor-based attacks when compared to VM based and VM image attacks. [Tank, Aggarwal and Chaubey, 2019]

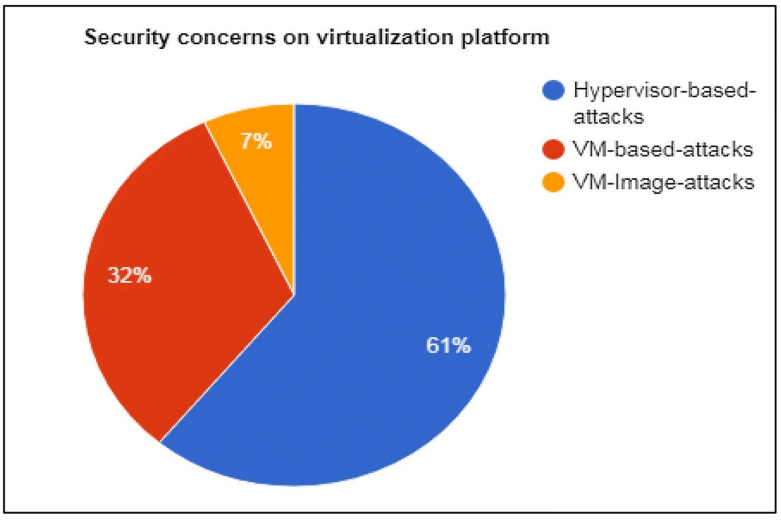


Fig 1. pie chart for security concerns on a virtualization platform (2019, p.7)

The attacks that the current hypervisor can go through are given below:

1. Blue Pill
2. Hyper jacking
3. VM escape
4. DKSM attack
5. SubVirt
6. VM Sprawling
7. Inter VM communication skills
8. Lack of visibility into virtual network traffic
9. The Cache Observing Attack

2.1 Blue Pill

The blue pill rootkit is malware that executes to gain control over computer resources as a hypervisor. The hypervisor installs without a reboot and the machine normally works, without interruption of speed or service, making it difficult to detect. The main concept of a BluePill attack is to trap a running instance of the operating system by starting a thin hypervisor and virtualizing the rest of the machine under it. The previous operating system would still keep its current links to all devices and files, but almost anything could be intercepted by the hypervisor, including hardware interrupts information requests and even the system time by the hypervisor.

2.2 HyperJacking

Hyperjacking is an attack in which a hacker takes malicious control of the hypervisor creating the virtual environment within a host of a virtual machine (VM). The purpose of the attack is to target the operating system below that of the virtual machines so that the program of the attacker can run and the applications above it on the VMs become completely unaware of its existence. It includes installing a fake malicious hypervisor capable of managing the entire server system. Standard safety measures are ineffective because the operating system does not know that the computer has been compromised. This is because the fake hypervisor runs beneath the machine.

2.3 VM Escape

Virtual machines are usually encapsulated in isolated environments. The operating systems in the virtual machine should not realize that they are virtualized, and there should be no way to break out of the virtual machine and communicate with the hypervisor of the parent. The

hypervisor break-out and interaction process is known as a VM escape. Since the hypervisor controls all virtual machine execution, an attacker who can gain access to the hypervisor can then gain control over any other virtual machine operating on the host. An attacker can then override the security controls on the virtual machine, as the hypervisor is between the physical hardware and the guest operating system.[Tank, Aggarwal and Chaubey, (2019)]

2.4 DKSM attack

A DKSM attack is a hypervisor attack that can effectively damage and falsify existing VM introspection tools. Virtual machine (VM) introspection is a powerful technique in a virtualized environment for determining the specific aspects of execution within a guest. An attack on kernel structures, that by providing false information prevents VM introspection. The presence of vulnerabilities in the guest kernel enables the attacker to potentially exploit to compromise the guest kernel and hijack the control flow. The attacker has the freedom to modify existing kernel code or inject his own code for execution by hijacking control flow. (Bahram, 2019)

A DKSM attack can manipulate these kernel data structures in three different approaches:

(1) Syntax-based manipulation where certain fields of kernel data structures are potentially added or removed

(2) Semantics-based manipulation where the semantics of the underlying data structures are changed and

(3) Multifaceted combo manipulation which effectively combines the previous two.

2.5 SubVirt

SubVirt is a malware attack that allows attackers to install a virtual-machine monitor (VMM) underneath an existing operating system and use that VMM to host arbitrary malicious software. The resulting malware is called VMBR (virtual-machine based rootkit), which not only provides more control and functionality than the current system but can also completely hide all its state and activity from intrusion detection systems running in the target operating system and applications, thus making it impossible to detect.(King, et al., 2019)

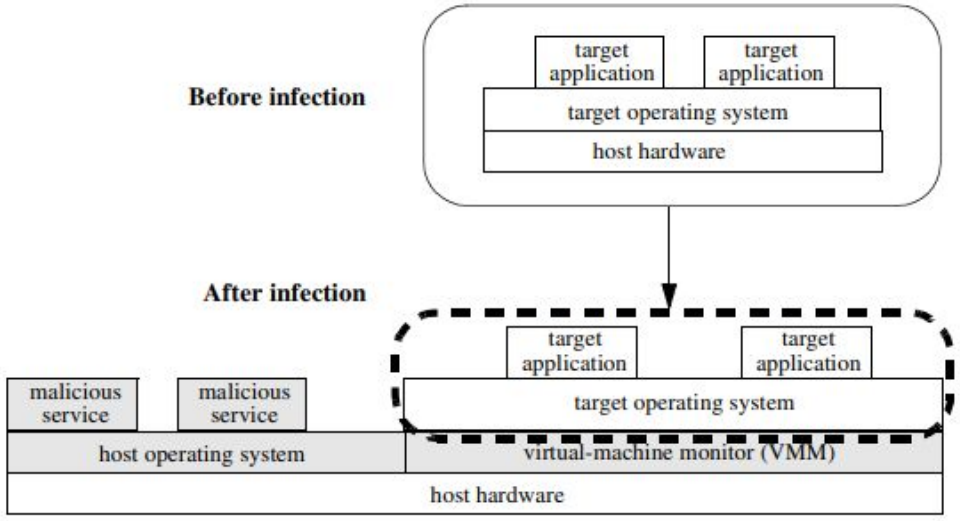


Fig 2. The figure shows how to shift an existing target system to run on a virtual machine with a virtual machine display. The figure's grey parts display the VMBR elements. (2019, p.3)

2.6 VM Sprawling

VM sprawling takes place when there is a large number of virtual machines without proper management or control in the system. That is when the number of virtual machines connected to the network exceeds the network's capabilities where the administrator can no longer manage them effectively.

2.7 Inter VM communication Skills

A typical hypervisor process consists of inter-process communication and file access. The virtualized stack contains several interacting processes such as LAMP (Linux, Apache, MySQL, Perl/Python/PHP) stack. Monitoring systems designed to detect unusual malicious activity over traditional non-virtual interfaces may be invisible.

2.8 Lack of visibility into virtual network traffic

One of the main virtualization problems is the lack of visibility in virtual networks used for virtual machine communications. As traffic flowing through virtual networks may not be visible to devices such as intrusion detection systems installed on a physical network, this poses problems when enforcing security policies. This is because of the complexity of the applications being virtualized. Network traffic flowing between virtual machines does not originate from a single host and the hypervisor is typically unable to control all virtual machine interaction. Certain tools such as WireShark can be used to monitor virtual network traffic and it is essential to use them. (Roussey, 2019)

2.9 The Cache Observing Attack

The Cache Observing Attack is an attack where the attacker allocates several neighbouring memory pages for a cache-like combined size. They then perform a series of instructions that jump through all of the page's frames, calculating the sequence's elapsed time. Then the attacker waits for a specific interval where the cache is used by the target. Timing how long it takes for the same cache sets to be refilled provides some information on how the target used the cache during that time. (*Violating Virtualization Security* (Anon., 2019))

3. Measures to prevent current vulnerabilities

A successful hypervisor attack may lead to disastrous consequences as there are numerous vulnerabilities that exist in a virtual environment. But with the right measures, one can protect data and minimize the hypervisor and virtualization security vulnerabilities. Below given is a list of control measures that can be taken to prevent against hypervisor threats :

1. Creating and managing the client's VM network away from their management network is a great way to secure virtualized environment. So, if the VM is attacked with malware, the hypervisor would be safe from the attack. (*Why hypervisor security is important* (Anon., 2018)
2. Ideally, the client and the system administrator should have access to the hypervisor console. To prevent unauthorized users from messing with VM settings and accessing your most sensitive data, one needs to set strict access restrictions on the software.
3. Off-the-shelf operating systems will have a lot of unnecessary applications and services that may raise the VM attack chances. So, one has to be knowledgeable about how to handle such operating systems and disable services.
4. Paying attention to the physical security where the server room must be monitored at all times and making sure that the physical servers are behind locked doors.
5. Because of network intrusions that affect hypervisor protection, it is highly recommended to implement firewalls and intrusion prevention systems. So that when the hypervisor is attacked from the newest hacks, the security tools track network traffic for abnormal behaviour and revert back.
6. Hypervisors must always be updated to the newest builds to defend themselves against the latest threats.

7. Integrate monitoring of hypervisors into your overall infrastructure for system management.

8. Analyzing and keeping track of hypervisor logs continuously to find out any irregularities.

9. Using security tools for monitoring the virtual environment, including the virtual servers and the network traffic between virtual machines and hosts. Make sure to perform this frequently and include them in virtual administration activities.

10. Centralized administration of all hypervisors and hypervisor hosts through management software for virtualization. Having administrative accounts set up in each hypervisor host is recommended. (Chandramouli, 2019)

11. Communicating with the hypervisor through a secure protocol such as TLS(transport layer security) or SSH(secure sockets layer).

12. The ability to monitor backbone VM network traffic is important. Conventional methods will not detect VM traffic as it is managed by internal soft switches. Nevertheless, hypervisors have active monitoring tools that should be allowed and checked. (*Common Virtualization Vulnerabilities and How to Mitigate Risks* (Anon., 2019))

13. Hypervisor attack is understood to be launched via the internet. It attacks the packet sent to the hypervisor to be precise. In that case, by monitoring and analyzing network traffic, the VMHIDS(Virtual Machine and Hypervisor Intrusion detection system) is proposed (Gajare and Sonawani, 2017) which implements the principle of anomaly-based detection to distinguish malicious packets in real-time. Continuous monitoring from hypervisor or VMs with VMHIDS allows the automated detection and blocking of suspicious events to be analyzed in real-time. since VMHIDS is placed on both VMs and hypervisor, new attacks on hypervisor can be detected easily for faster prevention.

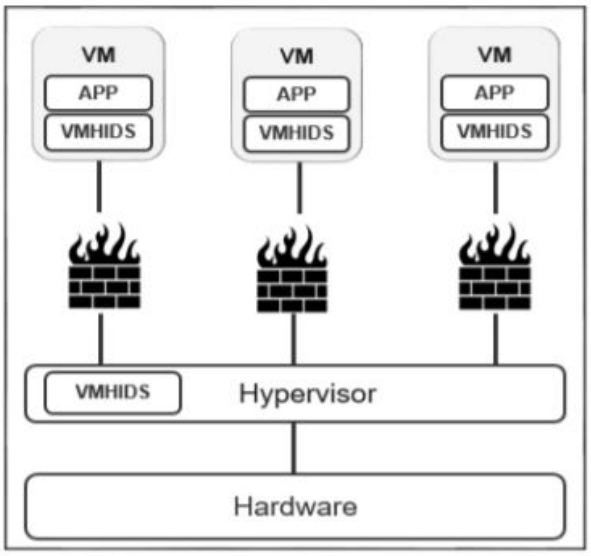


Fig 3. VM and Hypervisor IDS architecture (2017, p23)

14. Due to VM sprawl and other issues, secure access can be compromised. Ensure that authentication procedures, identity management, and logging are unbreakable.

15. Keeping load-balanced VMs on separate hypervisors will enhance service availability and provides optimal performance in the event of a hardware failure. This is called Segregate VMs. This feature is important if the client has two load-balanced servers or name servers and need to keep the VMs on separate physical machines. The client can segregate VMs from a single point control panel and also advise their cloud administrator to ensure that a VM never boots as another similar VM on the same hypervisor. (*Segregate Virtual Machine* (Anon., 2019))

4. Future Attacks

Virtualization has become essential for businesses seeking better resource supply, easier IT management, less hardware, and lower costs. Virtualization, however, is a complex and constantly evolving field that involves certain risks. Although taking concerned measures eliminates the risk of the system being compromised, but not completely secured for future attacks.

To prevent such attacks, AMD in 2016, introduced two memory encryption capabilities to protect sensitive data in multi-tenant environments in their processors.

- Secure Memory Encryption (SME) that protects memory against physical attacks like the cold boot and direct memory access attacks and
- Secure Encrypted Virtualization (SEV) which mixes memory encryption and virtualization, allowing each virtual machine to be protected from other virtual machines and underlying hypervisors and their admins.

The purpose of Secure Encrypted Virtualization is to secure the content of virtual machines from attacks on a shared virtual machine host by malicious visitors, as well as from attacks initiated by the hypervisor control software that controls all virtual machines on the host.

The paper, "The SEVerESt Of Them All: Inference Attacks Against Secure Virtual Enclaves," (Werner, et al., 2019) describes techniques that can be exploited by rogue cloud server administrators, or hypervisors hijacked by hackers, to figure out what applications are running within a SEV-protected guest virtual machine, even when its RAM is encrypted, and also extract or even inject data within those VMs. The two new attacks that can breach the confidentiality of protected enclaves are:

4.1 Register Inference Attack:

The security processor is used when SEV is allowed to automatically encrypt and decrypt memory contents on transitions between the guest and the hypervisor. All that remains unencrypted are the general-purpose registers used by virtual input/output devices in the virtual machine control block and DMA sections. Also, the hypervisor has access to the shared memory region which can force the guest to exit.

According to the security provided by designs that leave the contents of the register unprotected, the attacker can damage them. By carefully inspecting the general-purpose registers and exposing the computation that passes through them, an attacker can recover critical information about activities in the encrypted guest. Hence, one could do this by editing a target process from within the hypervisor which can cause a significant performance penalty that would be easily noticeable in the guest.

4.2 Structural Inference Attacks:

The registry status in the Virtual Machine Control Block is no longer available when SEV-ES (SEV Encrypted State) is enabled. SEV-ES not only encrypts but also preserves the VMCB's authenticity, thereby preventing attacks on the state of the register. A new structure called Guest Hypervisor Control Block (GHCB) acts as an intermediary between the guest and the hypervisor during hypercalls. VMEXITs are either labelled as Automatic (AE) or Non-Automatic (NAE); AE events do not allow the guest to reveal any state and cause an immediate transfer of control to the hypervisor. The attacker may use data provided by the Instruction Based Sampling (IBS) subsystem to identify applications running within the VM (e.g., to find out if an executed instruction was a branch, load, or store). One can also collect performance data from the virtual machine and match the observed behaviour to known

signatures of running applications. The SEV-ES attack assumes only that information is available from the quality measurement subsystem or IBS.

Resource exhaustion:

In a virtualized environment, software that uses particular physical server resources intensively may exhaust those resources and hence affect VM availability. This condition occurs because the shared environment in a physical server magnifies the severity of resource contention, especially when multiple VMs are running the same resource-intensive software at the same time—as in anti-virus scanning. (*Best Practices for Mitigating Risks in Virtualized Environments* (Anon., 2019))

5. Measures to prevent future vulnerabilities:

The vulnerabilities cannot be eliminated from software changes but design changes must be done. The paper, “Security Analysis of Encrypted Virtual Machines” (Hetzelt and Buhren, 2018) discusses mitigations to thwart attacks and proposes design changes for further versions of SEV such as:

- Encrypted general-purpose registers.
- No access to the vmcb after an initial configuration.
- Memory protection against hypervisor access.

The hypervisor must never be able to see the general-purpose registers as they leak sensitive guest data on any vmexit. A guest has no control over exits to the hypervisor, therefore the hardware must enforce the authentication of general-purpose registers. It introduces another challenge as some guest operations allow the hypervisor to read the registers for general purposes. The vmcb must contain decode assists for events such as the hypervisor attempting to emulate the access, will not be able to read it when the general-purpose

registers are encrypted. This is to ensure that malicious hypervisors cannot force a guest to reveal register content through decode assists.

Usually, the vmcb is configured only once during the initial setup while, with some exceptions, a good hypervisor does not have to change the vmcb at runtime. However, the fact that SEV allows users to change the vmcb, imposes a security risk because it allows users to divert the guest's control flow by setting an arbitrary instruction pointer. For this, altering the existing state caching mechanism to allow for creating a write-once vmcb is suggested. Currently, the content of the vmcb is already cached to improve context switch performance.

Writing to the guest’s memory by the untrusted hypervisor is dangerous. The absence of memory authentication makes the door open for fault injection and replay attacks. Integrity trees are the most common way of protecting memory from unauthorized access. They induce significant overhead performance and memory space. It is sufficient to prevent the hypervisor from using mechanisms such as CIP(Critical Infrastructure Protection) to write encrypted guest memory. However, the exclusion of pages from access to hypervisors requires non-trivial changes to both the guest operating system and the hypervisor.

6. Conclusion

As a result, the report has accomplishes its objectives of identifying the existing and future hypervisor vulnerabilities along with its control measures. Using monitoring and network security tools, administrators will be able to monitor their virtual environments early and detect unusual behaviours. Minimizing potential attack surfaces by reducing potential access points would make it harder to access a digital network. Some operating systems or hypervisors will have additive features that may not be required or used by an enterprise to increase a VM's attack surface. A system administrator should also set limits on who can access the hypervisor from the remote and console. Finally, another best practice is blocking access to physical servers, as access to the physical server is a relatively easy way to access the hypervisors.

7. References:

1. Adrian Winkles 2019, Security Student Notes(University site) [Online] Available at <https://canvas.anglia.ac.uk/courses/12041/files/848908?module_item_id=477297> [Accessed 18 Nov. 2019]
2. Bahram, S., Jiang, X., Wang, Z., Grace, M., Li, J. and Xu, D. (2019). *DKSM: Subverting Virtual Machine Introspection for Fun and Profit*. [online] Research Gate. Available at:
3. Bob Cromwell: Travel, Linux, Cybersecurity. (2019). *Virtualization Security*. [online] Available at: https://cromwell-intl.com/cybersecurity/virtualization.html [Accessed 6 Dec. 2019].
4. Chandramouli, R. (2019). *Security Recommendations for Hypervisor Deployment on Servers*. [online] Nvlpubs.nist.gov. Available at: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-125A.pdf [Accessed 5 Dec. 2019].
5. Claburn, T. (2019). *AMD's SEV tech that protects cloud VMs from rogue servers may as well stand for... Still Extremely Vulnerable*. [online] Theregister.co.uk. Available at: https://www.theregister.co.uk/2019/07/10/amd_secure_enclave_vulnerability/ [Accessed 6 Dec. 2019].
6. Consulting, M. and Long, R. (2019). *The Risk of Virtualization - Concerns and Controls*. [online] MHA Consulting. Available at: https://www.mha-it.com/2017/06/20/risk-of-virtualization/ [Accessed 9 Dec. 2019].
7. Devhost.com. (2019). *Segregate Virtual Machine - Features - DevHost Cloud Hosting*. [online] Available at: https://www.devhost.com/features/segregate-virtual-machine/ [Accessed 8 Dec. 2019].
8. Downloads.cloudsecurityalliance.org. (2015). *Best Practices for Mitigating Risks in Virtualized Environments*. [online] Available at:

https://downloads.cloudsecurityalliance.org/whitepapers/Best_Practices_for%20_Mitigating_Risks_Virtual_Environments_April2015_4-1-15_GLM5.pdf [Accessed 4 Dec. 2019].

9. En.wikipedia.org. (2019). *Hyperjacking*. [online] Available at: https://en.wikipedia.org/wiki/Hyperjacking [Accessed 8 Dec. 2019].
10. Gajare, S. and Sonawani, P. (2017). *Safeguard Hypervisor attacks in cloud computing*. [online] Ijseas.com. Available at: http://ijseas.com/volume3/v3i12/ijseas20171204.pdf [Accessed 4 Dec. 2019].
11. Hetzelt, F. and Buhren, R. (2018). *Security Analysis of Encrypted Virtual Machines*. [online] Arxiv.org. Available at: https://arxiv.org/pdf/1612.01119v1.pdf [Accessed 9 Dec. 2019].

https://www.researchgate.net/publication/232642261_DKSM_Subverting_Virtual_Machine_Introspection_for_Fun_and_Profit [Accessed 4 Dec. 2019].

12. HyTrust. (2019). *Protecting Your Hypervisor: When the Enemy Breaks Through the Line, Command and Control Is the New Soft Underbelly*. [online] Available at: https://www.hytrust.com/blog/protecting-your-hypervisor-when-the-enemy-breaks-through-the-line-command-and-control-is-the-new-soft-underbelly/ [Accessed 9 Dec. 2019].
13. King, S., Chen, P., Wang, Y., Verbowski, C., Wang, H. and Lorch, J. (2019). *SubVirt: Implementing malware with virtual machines*. [online] Researchgate. Available at: https://www.researchgate.net/publication/220713559_SubVirt_Implementing_malware_with_virtual_machines [Accessed 6 Dec. 2019].
14. Millman, R. (2016). *Researchers query hypervisor security in future AMD Zen processors*. [online] Scmagazineuk.com. Available at: https://www.scmagazineuk.com/researchers-query-hypervisor-security-future-amd-zen-processors/article/1475724 [Accessed 5 Dec. 2019].

15. Morbitzer, M., Huber, M., Horsch, J. and Wessel, S. (2019). *SEVered: Subverting AMD's Virtual Machine Encryption*. [online] Arxiv.org. Available at: https://arxiv.org/pdf/1805.09604.pdf [Accessed 5 Dec. 2019].

16. Myerson, J. (2018). *Secure encrypted virtualization: How is this technology exploited?*. [online] SearchSecurity. Available at: https://searchsecurity.techtarget.com/answer/Secure-encrypted-virtualization-How-is-this-technology-exploited [Accessed 9 Dec. 2019].

17. Myerson, J. (2019). *How can a hypervisor deployment avoid security risks?*. [online] SearchCloudSecurity. Available at: https://searchcloudsecurity.techtarget.com/answer/How-can-a-hypervisor-deployment-avoid-security-risks [Accessed 6 Dec. 2019].

18. Officer), E., Cybersecurity), G., Communications), J., Mark Nunnikhoven (Vice President, C., Rik Ferguson (VP, S. and Strategies), W. (2019). *Hypervisors Bring New Capabilities and New Risks* -. [online] Blog.trendmicro.com. Available at: https://blog.trendmicro.com/hypervisors-bring-new-capabilities-and-new-risks/ [Accessed 6 Dec. 2019].

19. Penetration Testing Lab. (2019). *Common Virtualization Vulnerabilities and How to Mitigate Risks*. [online] Available at: https://pentestlab.blog/2013/02/25/common-virtualization-vulnerabilities-and-how-to-mitigate-risks/ [Accessed 8 Dec. 2019].

20. Rouse, M. (2019). *What is hypervisor security? - Definition from WhatIs.com*. [online] SearchCloudSecurity. Available at: https://searchcloudsecurity.techtarget.com/definition/hypervisor-security [Accessed 10 Dec. 2019].

21. Roussey, B. (2019). *REAL THREATS IN VIRTUALIZED ENVIRONMENTS: IDENTIFYING AND MITIGATING THE RISKS*. [online] Techgenix.com. Available at: http://techgenix.com/virtualization-risks/ [Accessed 4 Dec. 2019].

22. Rutkowska, J. (2019). *Introducing Blue Pill*. [online] Theinvisiblethings.blogspot.com. Available at: https://theinvisiblethings.blogspot.com/2006/06/introducing-blue-pill.html [Accessed 2 Dec. 2019].

23. Tank, D., Aggarwal, A. and Chaubey, N. (2019). *Virtualization vulnerabilities, security issues, and solutions: a critical study and comparison*. [online] Available at: https://www.researchgate.net/publication/331387774_Virtualization_vulnerabilities_security_issues_and_solutions_a_critical_study_and_comparison#pf11 [Accessed 4 Dec. 2019].

24. Techadvisory.org. (2018). *Why hypervisor security is important*. [online] Available at: https://www.techadvisory.org/2018/03/why-hypervisor-security-is-important/ [Accessed 9 Dec. 2019].

25. Vaughan-Nichols, S. (2019). *Hypervisors: The cloud's potential security Achilles heel | ZDNet*. [online] ZDNet. Available at: https://www.zdnet.com/article/hypervisors-the-clouds-potential-security-achilles-heel/ [Accessed 6 Dec. 2019].

26. Werner, J., Mason, J., Antonakakis, M., Polychronakis, M. and Monrose, F. (2019). [online] Regmedia.co.uk. Available at: https://regmedia.co.uk/2019/07/10/amd.pdf [Accessed 4 Dec. 2019].

PART C

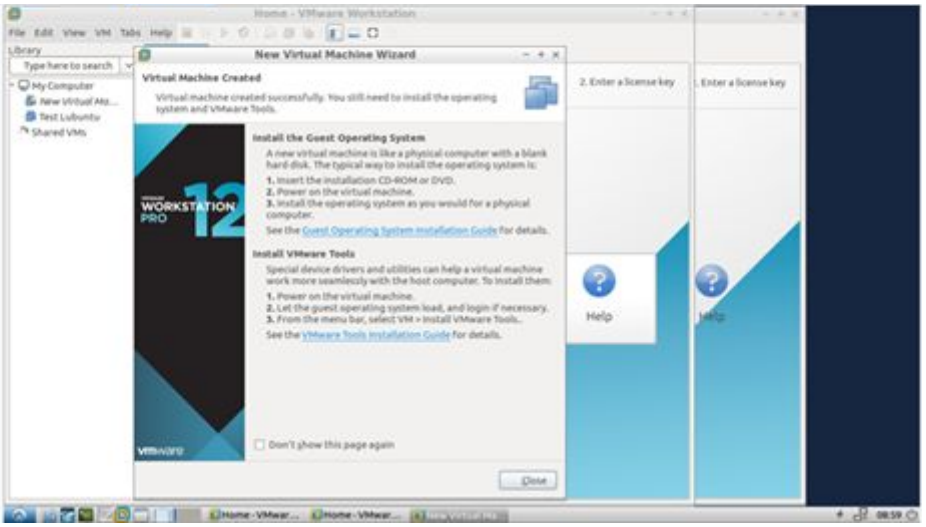
INTRODUCTION TO VIRTUALIZATION

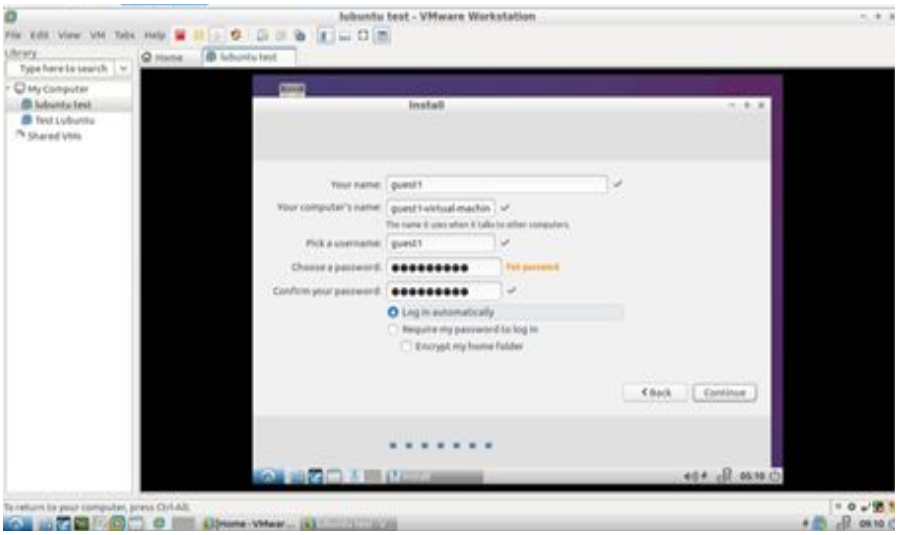
LAB 01: Creating a Virtual Machine

In this lab exercise, creating a virtual machine is practiced and learned. Along with Installing Linux Lubuntu operating system on the new virtual machine and modifying the virtual hardware settings.

The below screenshots indicate how the virtual machine is created and how to LUbuntu was created using login credentials.

Virtual machines are highly available and are easily maintainable. Virtual machines consist of multiple OS environments that run simultaneously that provides the functionality of a physical computer.

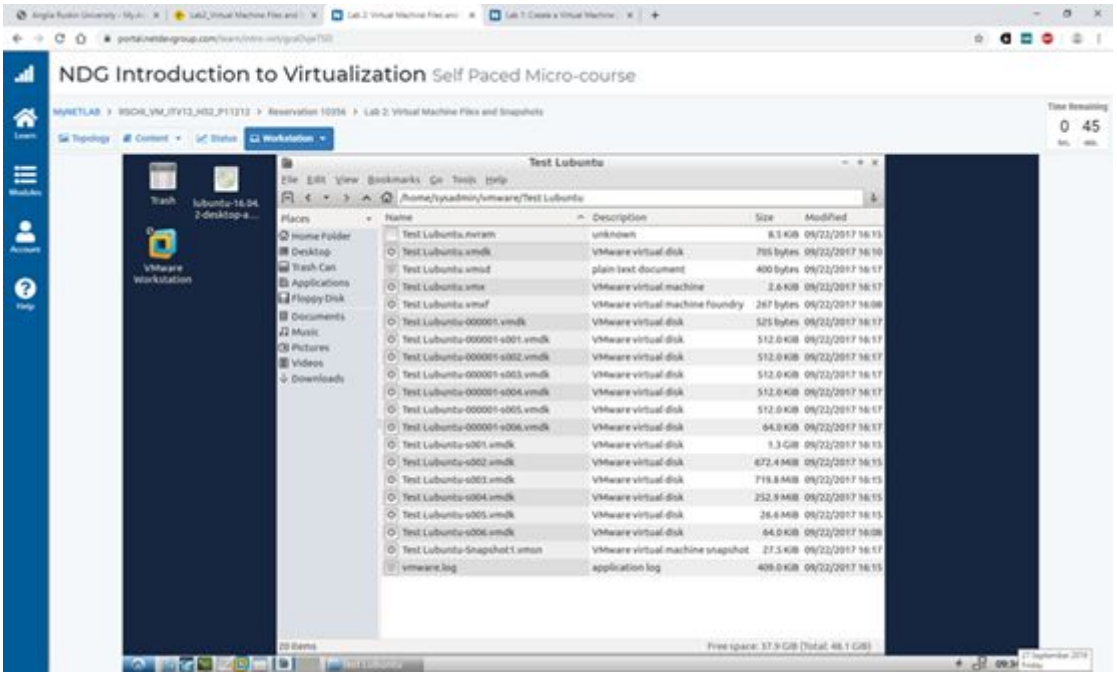


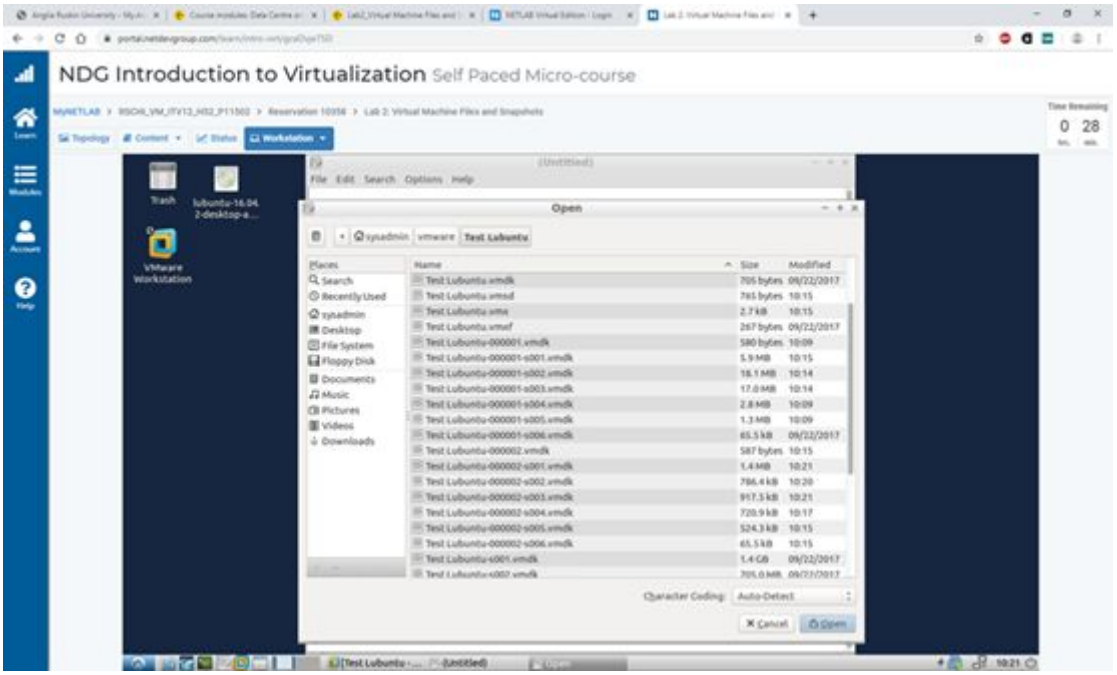

lubuntu test - VMware Workstation
Install
Your name:
Your computer's name:
Pick a username:
Choose a password:
Confirm your password:
Log in automatically
Require my password to log in
Encrypt my home folder
Back
Continue

LAB 02: Virtual Machine Files and Snapshots

In this lab exercise, it is learned how to identify the virtual machine files and where virtual machine settings are stored. It is also practiced how to work with a snapshot and what files change when a snapshot is taken as shown in the screenshots below.

The VM files have stored under the sysadmin in the Test Lubuntu file system. And snapshots are created as they are useful during recovery scenarios. A virtual machine is made up of multiple files stored on a storage device. A configuration file, virtual disc file, setting NVRAM file, and log file are the key files




NDG Introduction to Virtualization Self Paced Micro-course
Open
Test Lubuntu
Cancel
Open

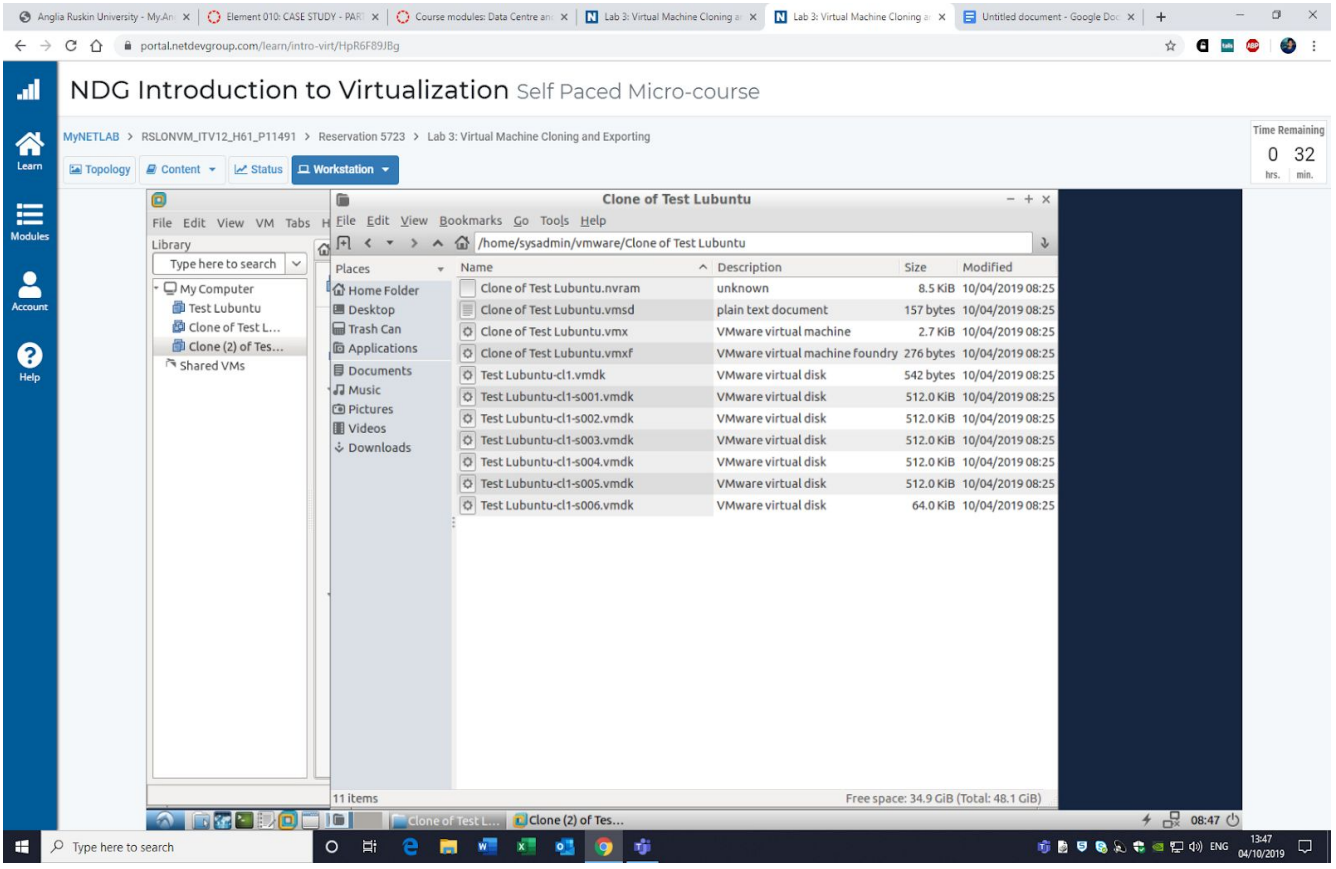
Lab 03: Virtual Machine Cloning and Exporting

In this lab exercise, the creation of both linked type and the full type clone of a virtual machine was learned and is shown in the screenshots below.

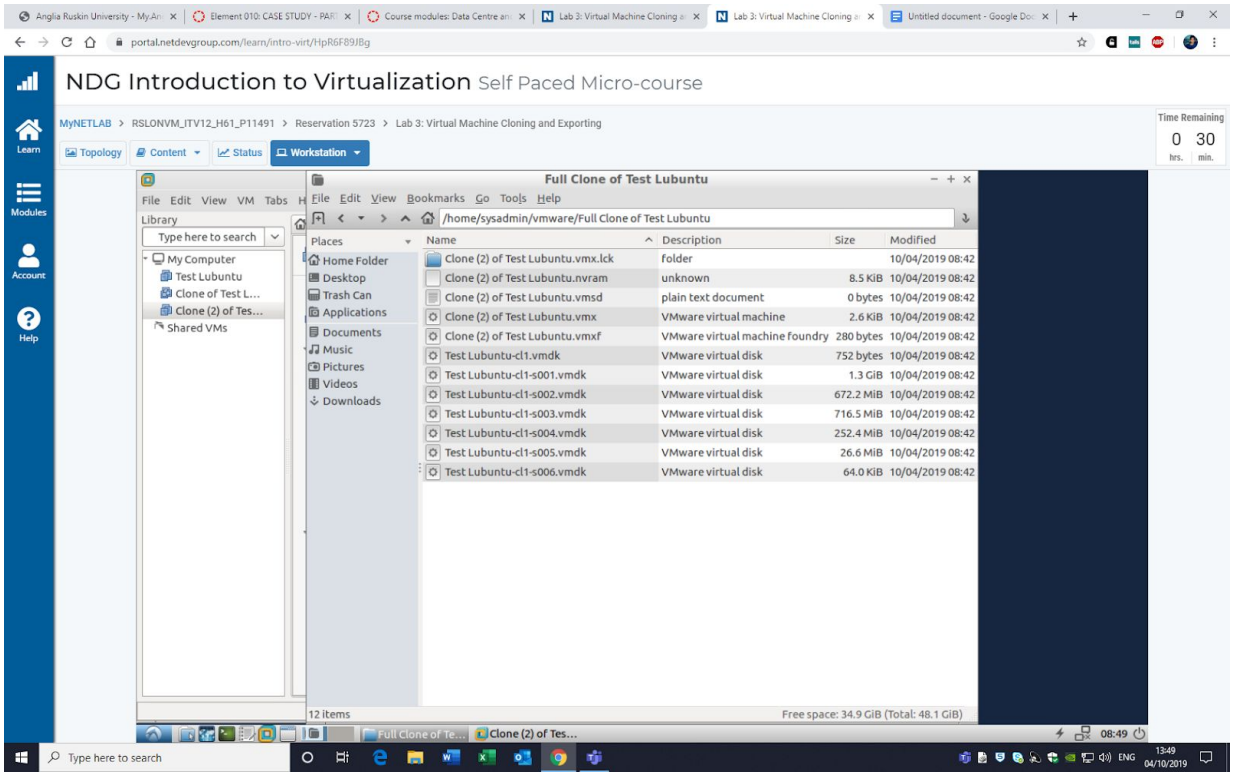
A full clone is an individual copy of a virtual machine that does not share anything after the cloning process with the parent virtual machine. The ongoing activity of a full clone is completely separate from the virtual machine of the parent.

A linked clone is a replica of a virtual machine that continuously shares virtual discs with the virtual machine of the parent. It preserves disc space and allows multiple virtual machines to use the same software installation.

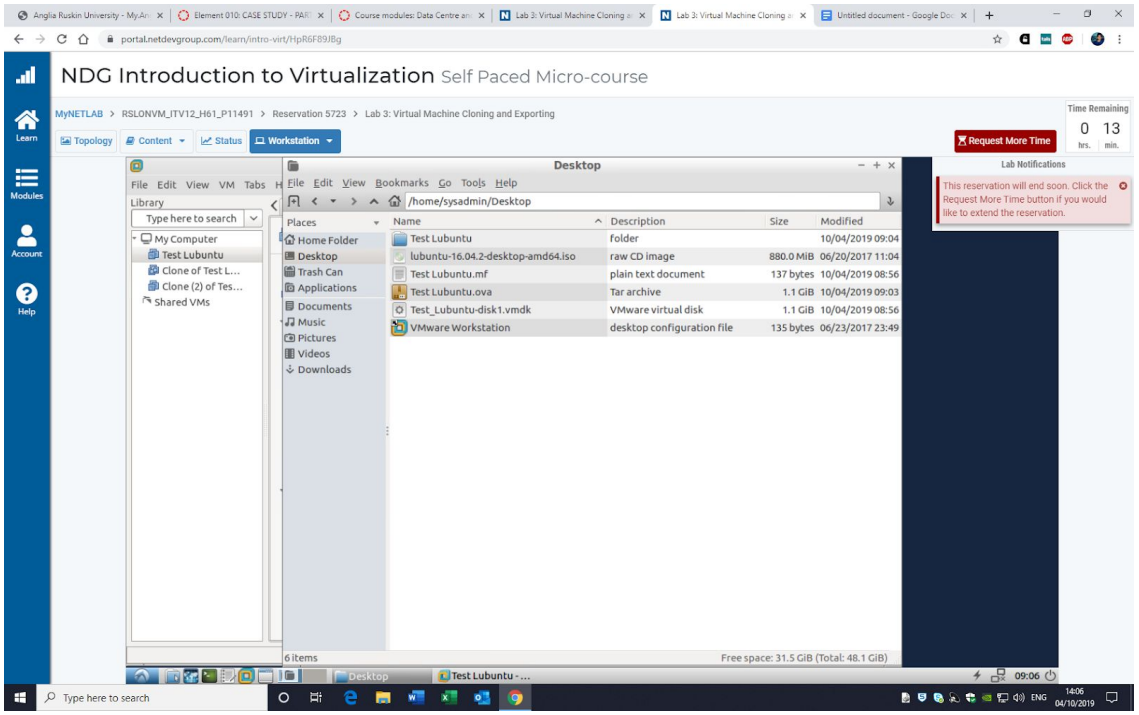
1. Create a Linked Virtual Clone

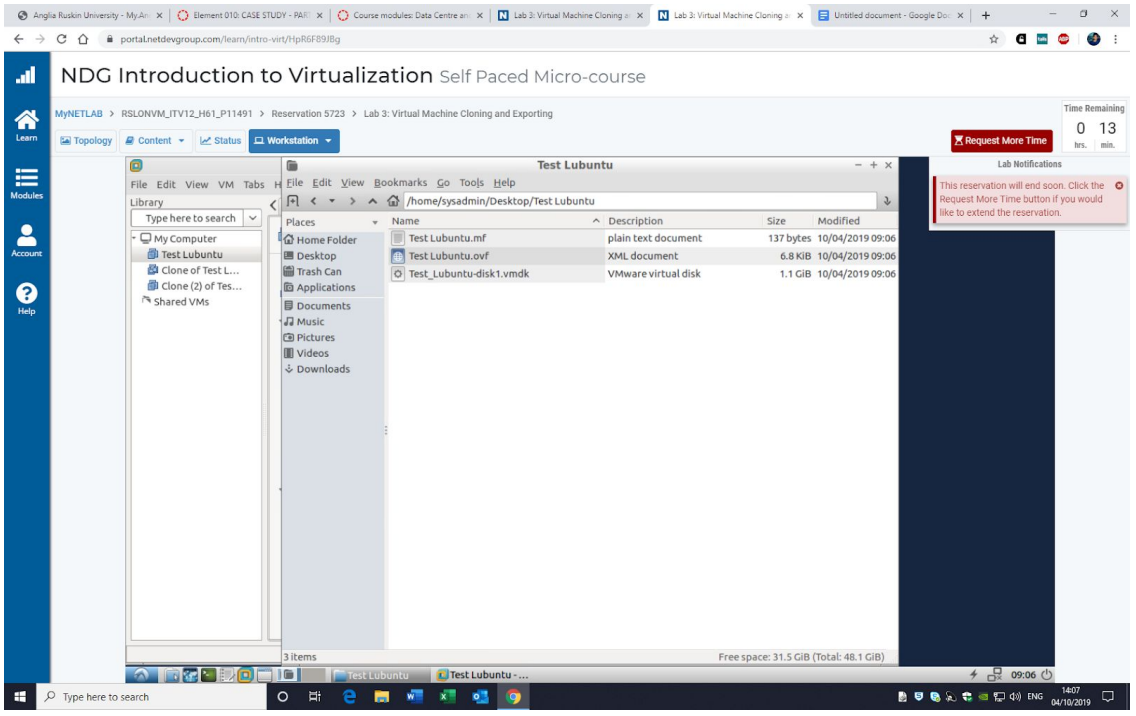


2. Create a Full Virtual Machine Clone



3. Exporting a Virtual Machine

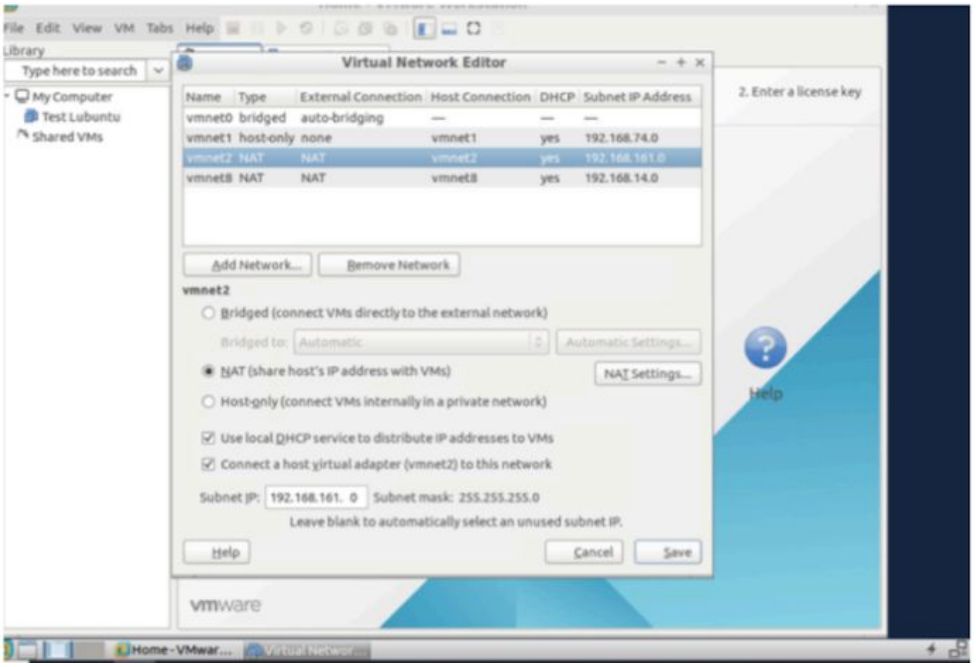

NDG Introduction to Virtualization Self Paced Micro-course
MyNETLAB > RSLONVM_ITV12_H61_P11491 > Reservation 5723 > Lab 3: Virtual Machine Cloning and Exporting
Topology Content Status Workstation
Time Remaining 0 hrs. 13 min.
Request More Time
Lab Notifications
This reservation will end soon. Click the Request More Time button if you would like to extend the reservation.
Desktop
File Edit View Bookmarks Go Tools Help
/home/sysadmin/Desktop
Library
Type here to search
My Computer
Test Lubuntu
Clone of Test L...
Clone (2) of Tes...
Shared VMs
Places
Home Folder
Desktop
Trash Can
Applications
Documents
Music
Pictures
Videos
Downloads
Name Description Size Modified
Test Lubuntu folder 10/04/2019 09:04
lubuntu-16.04.2-desktop-amd64.iso raw CD image 880.0 MiB 06/20/2017 11:04
Test Lubuntu.mf plain text document 137 bytes 10/04/2019 08:56
Test Lubuntu.ova Tar archive 1.1 GiB 10/04/2019 09:03
Test_Lubuntu-disk1.vmdk VMware virtual disk 1.1 GiB 10/04/2019 08:56
VMware Workstation desktop configuration file 135 bytes 06/23/2017 23:49
6 items
Free space: 31.5 GiB (Total: 48.1 GiB)
Desktop
Test Lubuntu - ...
09:06
Type here to search
ENG
14:06
04/10/2019

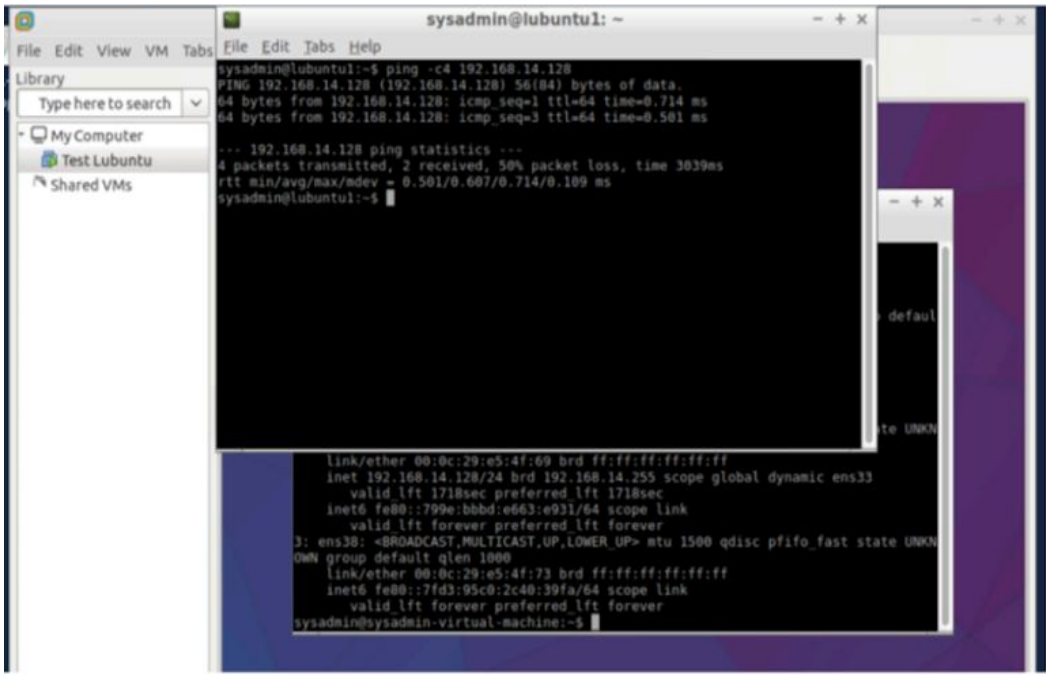

NDG Introduction to Virtualization Self Paced Micro-course
MyNETLAB > RSLONVM_ITV12_H61_P11491 > Reservation 5723 > Lab 3: Virtual Machine Cloning and Exporting
Topology Content Status Workstation
Time Remaining 0 hrs. 13 min.
Request More Time
Lab Notifications
This reservation will end soon. Click the Request More Time button if you would like to extend the reservation.
Test Lubuntu
File Edit View Bookmarks Go Tools Help
/home/sysadmin/Desktop/Test Lubuntu
Library
Type here to search
My Computer
Test Lubuntu
Clone of Test L...
Clone (2) of Tes...
Shared VMs
Places
Home Folder
Desktop
Trash Can
Applications
Documents
Music
Pictures
Videos
Downloads
Name Description Size Modified
Test Lubuntu.mf plain text document 137 bytes 10/04/2019 09:06
Test Lubuntu.ovf XML document 6.8 KiB 10/04/2019 09:06
Test_Lubuntu-disk1.vmdk VMware virtual disk 1.1 GiB 10/04/2019 09:06
3 items
Free space: 31.5 GiB (Total: 48.1 GiB)
Test Lubuntu
Test Lubuntu - ...
09:06
Type here to search
ENG
14:07
04/10/2019

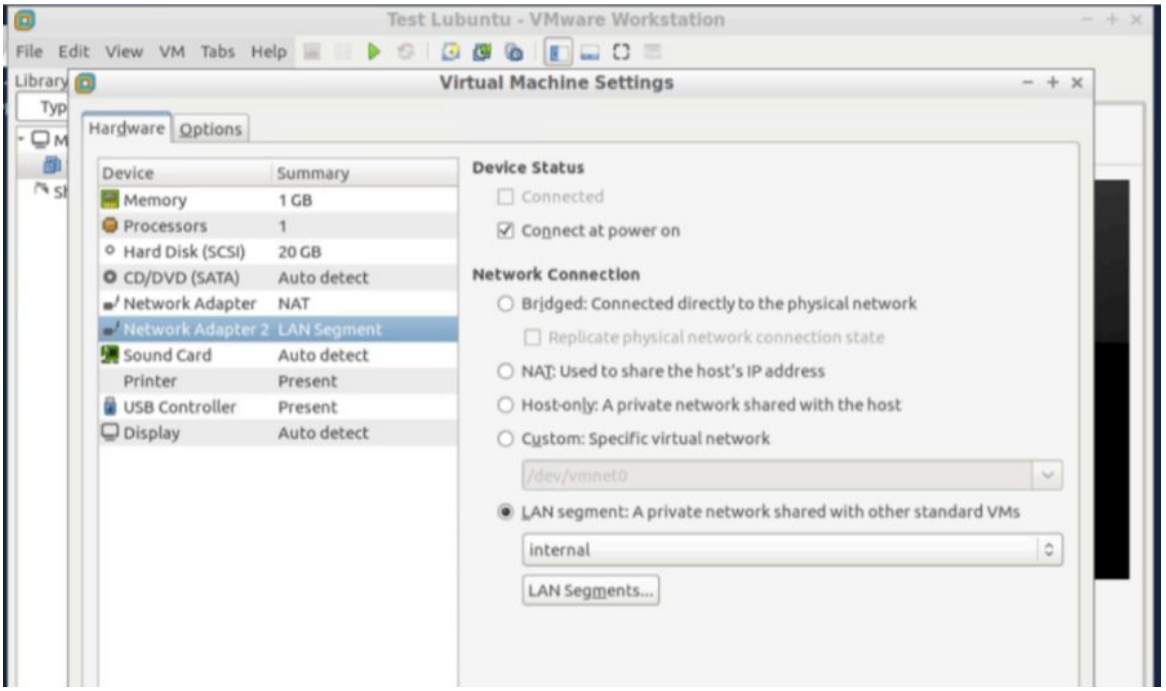
LAB 04: Virtual Machine Networking

In this lab exercise, identifying the different network types available in the Workstation environment and how to set up a new vmnet was practised and learned. It is also learned how to set up NAT rules and DHCP and how to show LAN segments and adapter settings as shown in the screenshots below.

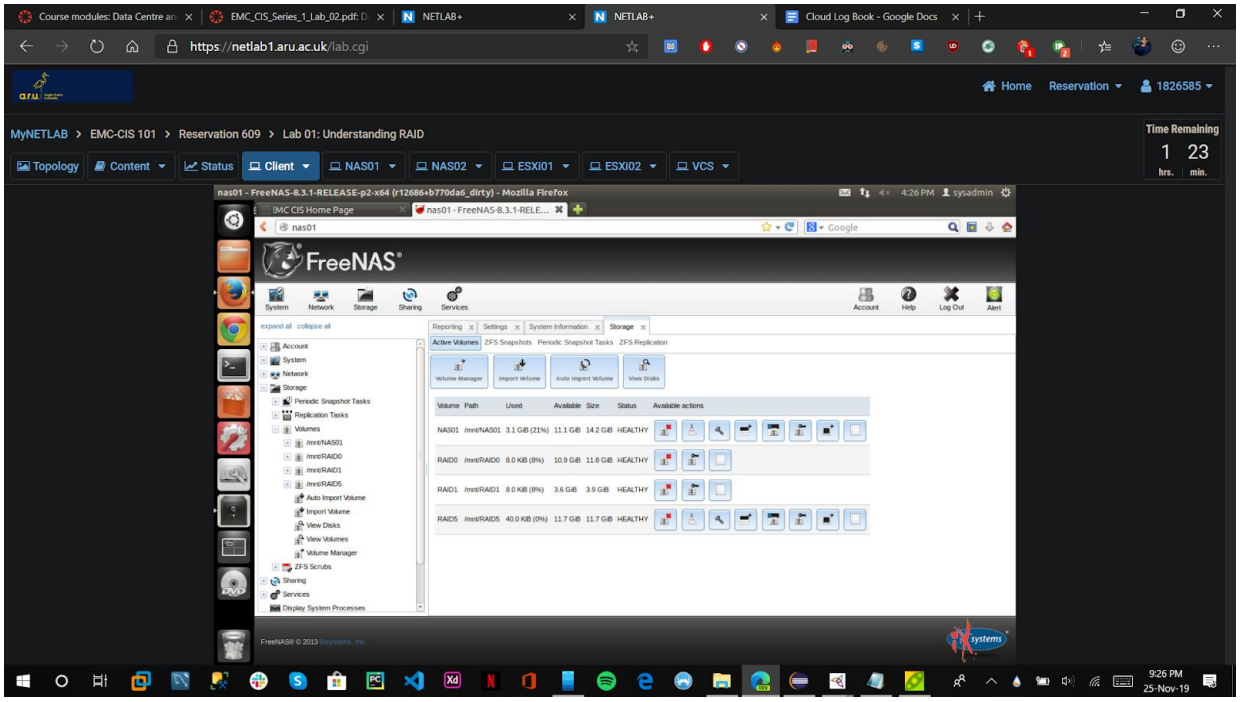
After the Exercise, it is understood that VMware Workstation's network features are very useful for software and application development and testing, such as reducing bandwidth or triggering packet loss scenarios.




File Edit View VM Tabs
Library
Type here to search
My Computer
Test Lubuntu
Shared VMs
sysadmin@lubuntu1: ~
File Edit Tabs Help
sysadmin@lubuntu1:~$ ping -c4 192.168.14.128
PING 192.168.14.128 (192.168.14.128) 56(84) bytes of data.
64 bytes from 192.168.14.128: icmp_seq=1 ttl=64 time=0.714 ms
64 bytes from 192.168.14.128: icmp_seq=3 ttl=64 time=0.501 ms
--- 192.168.14.128 ping statistics ---
4 packets transmitted, 2 received, 50% packet loss, time 3039ms
rtt min/avg/max/mdev = 0.501/0.607/0.714/0.109 ms
sysadmin@lubuntu1:~$
link/ether 00:0c:29:e5:4f:69 brd ff:ff:ff:ff:ff:ff
inet 192.168.14.128/24 brd 192.168.14.255 scope global dynamic ens33
valid_lft 1718sec preferred_lft 1718sec
inet6 fe80::799e:bbbd:e663:e931/64 scope link
valid_lft forever preferred_lft forever
3: ens38: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKN
OWN group default qlen 1000
link/ether 00:0c:29:e5:4f:73 brd ff:ff:ff:ff:ff:ff
inet6 fe80::7fd3:95c0:2c40:39fa/64 scope link
valid_lft forever preferred_lft forever
sysadmin@sysadmin-virtual-machine:~$


Test Lubuntu - VMware Workstation
File Edit View VM Tabs Help
Library
Virtual Machine Settings
Hardware Options
Device Summary
Memory 1 GB
Processors 1
Hard Disk (SCSI) 20 GB
CD/DVD (SATA) Auto detect
Network Adapter NAT
Network Adapter 2 LAN Segment
Sound Card Auto detect
Printer Present
USB Controller Present
Display Auto detect
Device Status
Connected
Connect at power on
Network Connection
Bridged: Connected directly to the physical network
Replicate physical network connection state
NAT: Used to share the host's IP address
Host-only: A private network shared with the host
Custom: Specific virtual network
/dev/vmnet0
LAN segment: A private network shared with other standard VMs
internal
LAN Segments...

Lab 05: Understanding RAID



In this lab exercise, the creation of RAID 0, 1, and 5 in a FreeNAS system was learned as shown in the screenshot. The fundamental differences between these RAID implementations have been understood.

Raid 0 has a minimum of 2 disks and provides a high performance during temporary data needs. There is no mirroring and parity and hence there are no data protection capabilities. Hence using this for any critical system is avoided.

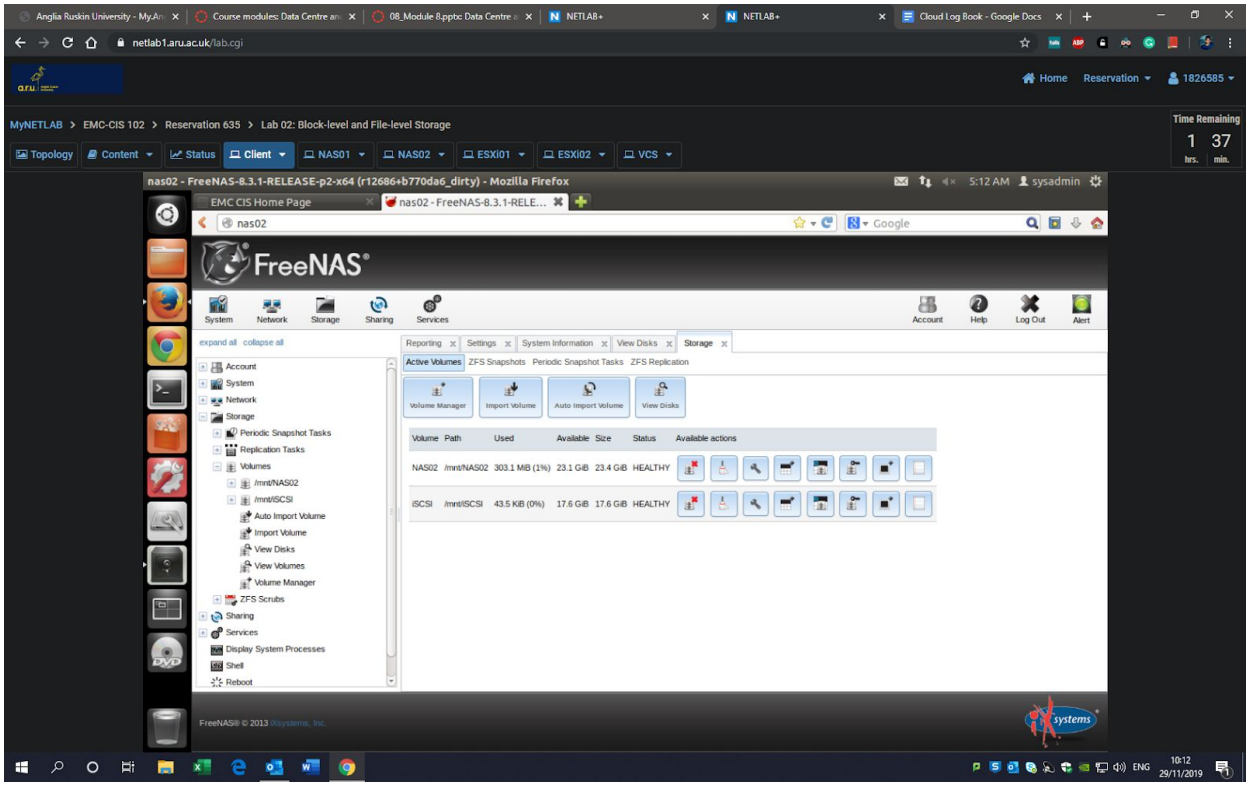
RAID 1 does not provide much performance gain as there is no striping and parity. It has good redundancy as RAID1 supports mirroring.

RAID 5 provides good performance as it utilizes striping and parity. Using RAID5 for a heavy read oriented system is recommended as the write operations will be slow.

The differences between the size and available for each volume are due to the overhead required by the file system to manage the data stored on disk(s).

Lab 06: Block-level and File-level Storage

1.2 Create a Volume for Our iSCSI Target



In this lab, configuring the common protocols such as iSCSI and NFS for block-level and file-level storage systems has been learned and the differences between SAN and NAS have also been understood.

Block-level systems enable the client to manage the storage space while File-level storage enables the server to manage the storage space. iSCSI is a block-level storage protocol that is also referred to as a Storage Area Network(SAN). NFS is a file-level storage protocol that is

also referred to as Network Attached Storage(NAS). NAS accesses data as files, while A SAN stores data at the block level.

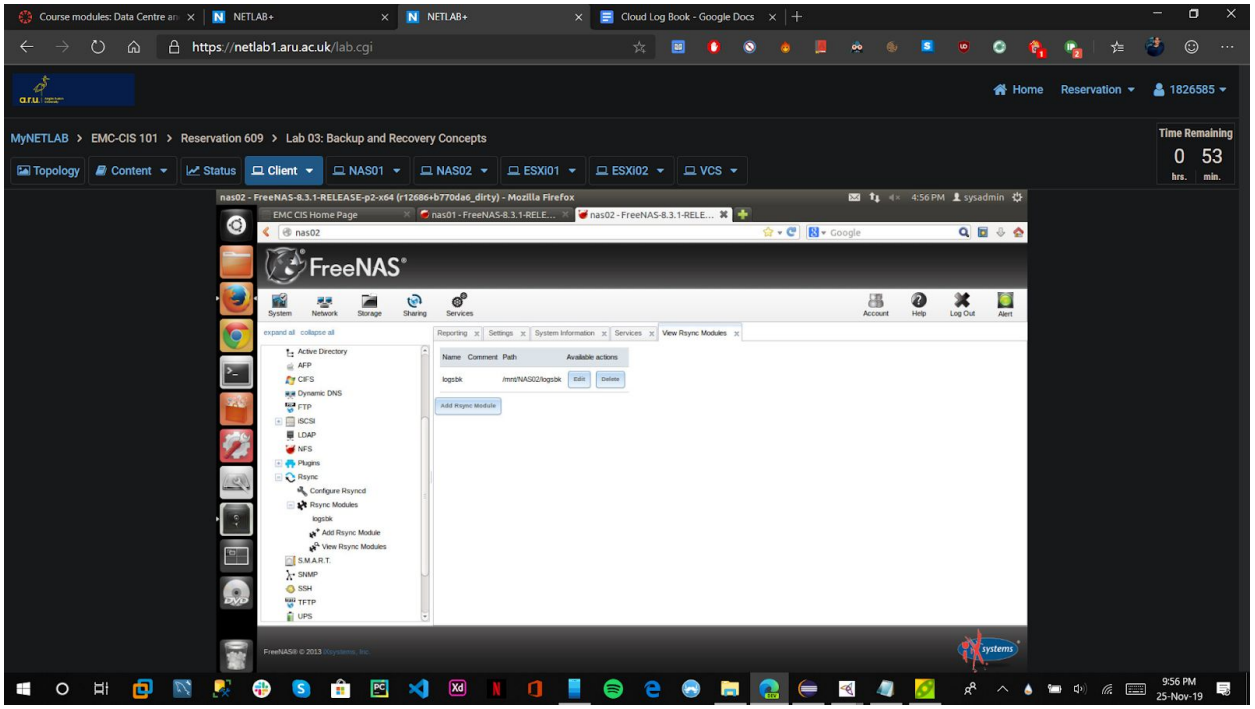
Lab 07: Backup and Recovery Concepts

In this exercise, configuring the NAS02 rsync service to accept the rsync push from NAS01 has been learned. Synchronizing and verifying data from NAS01 to NAS02 are practised as shown in the screenshots below. It is also learned that how snapshots are created and are used for point-in-time recovery.

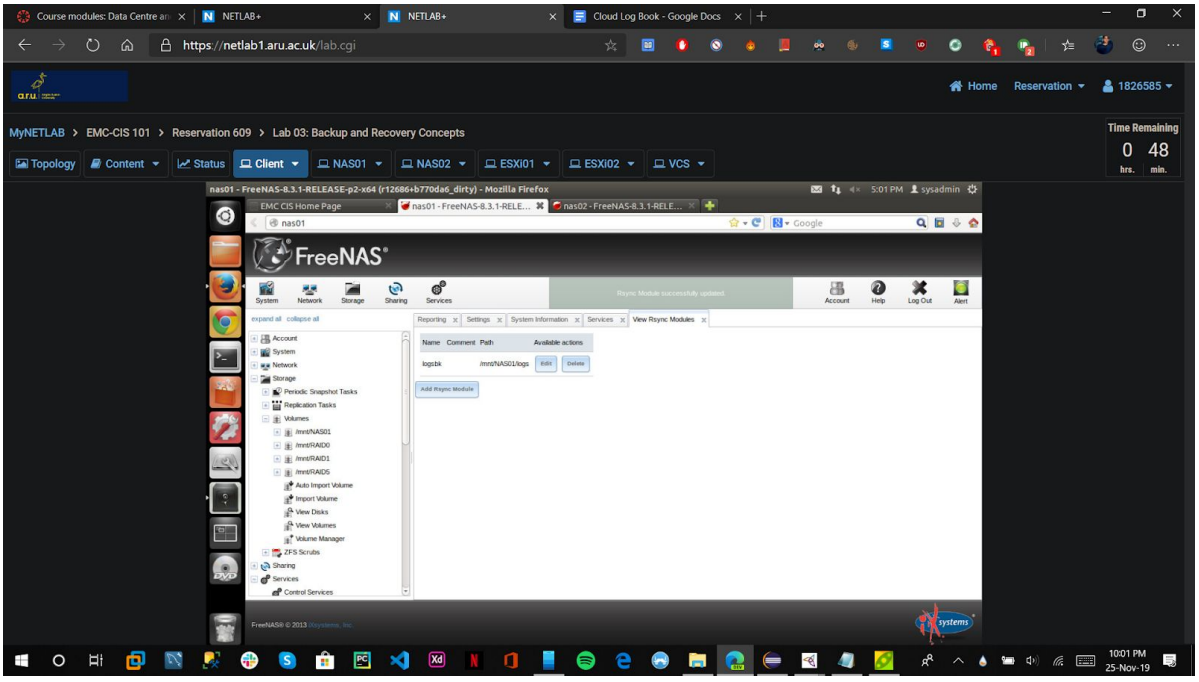
The rsync backup allows clients in copying and synchronizing files both locally and remotely across Unix-based systems. The rsync always provides a single backup to restore. To overcome this issue, snapshots are practised.

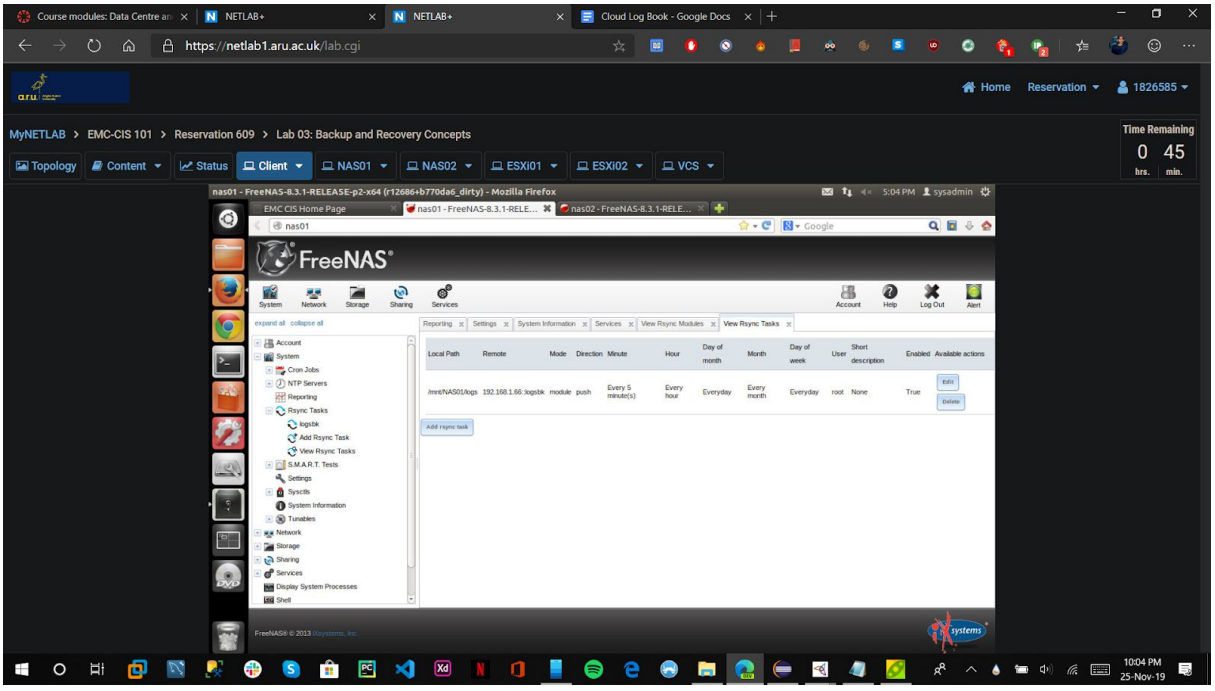
A snapshot is not equivalent to a backup copy and it does not store the data on its own, but only defines where and how the data has been stored and organized.

1.2 Configure rsync on NAS02



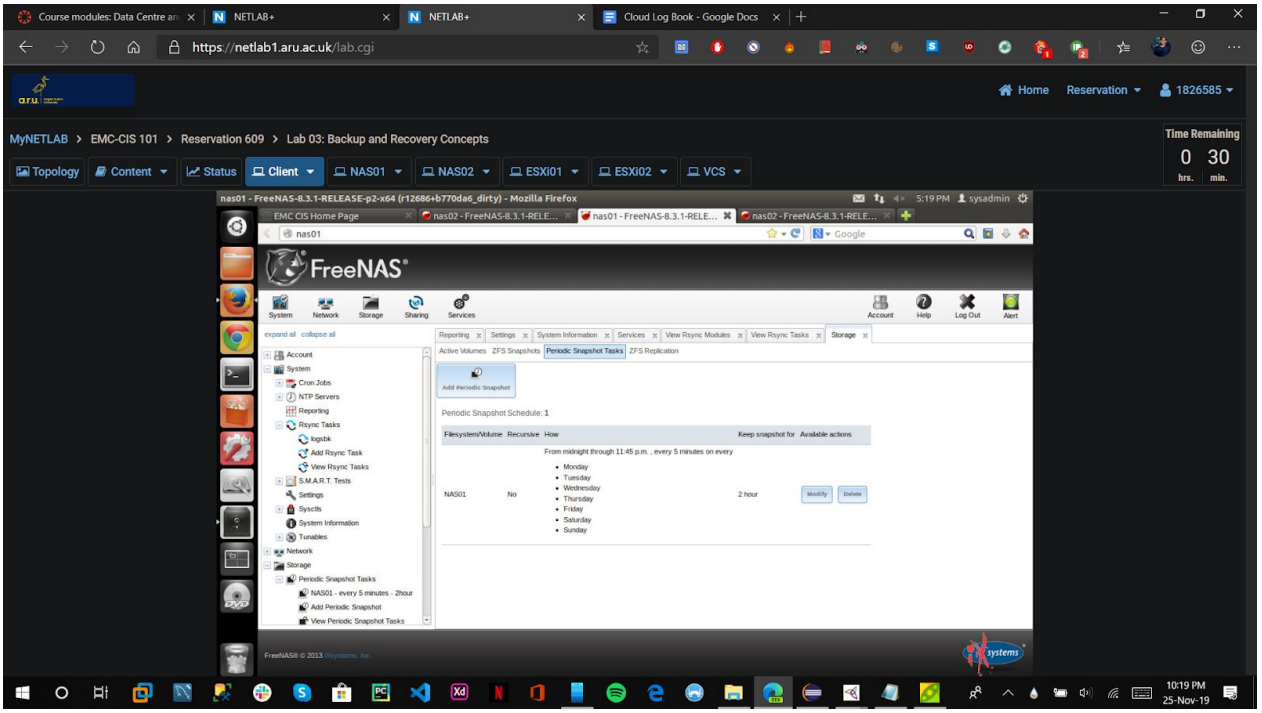
1.3 Configure the rsync and rsync Tasks on NAS01



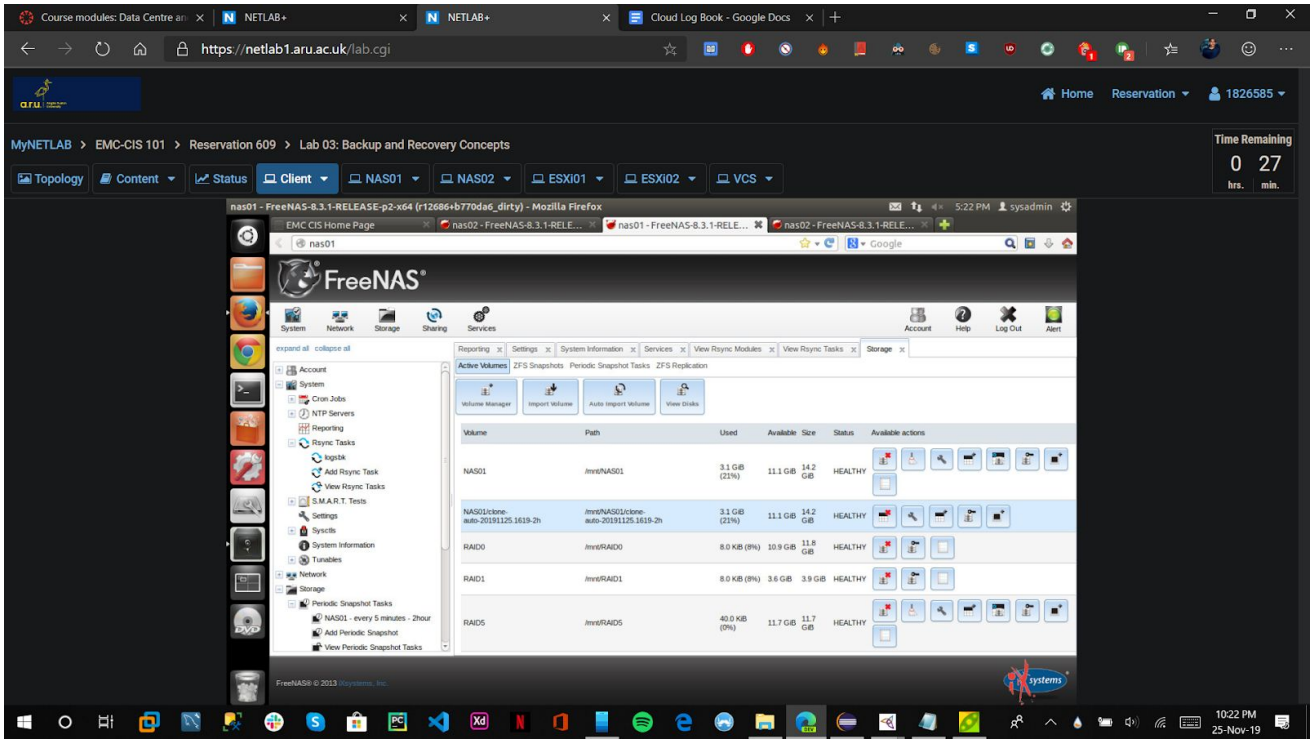


2 Create Snapshots

2.1 Creating Periodic Snapshots on NAS01



2.2 Mounting a Snapshot for Recovery on NAS01



Lab 08: Replication and Deduplication

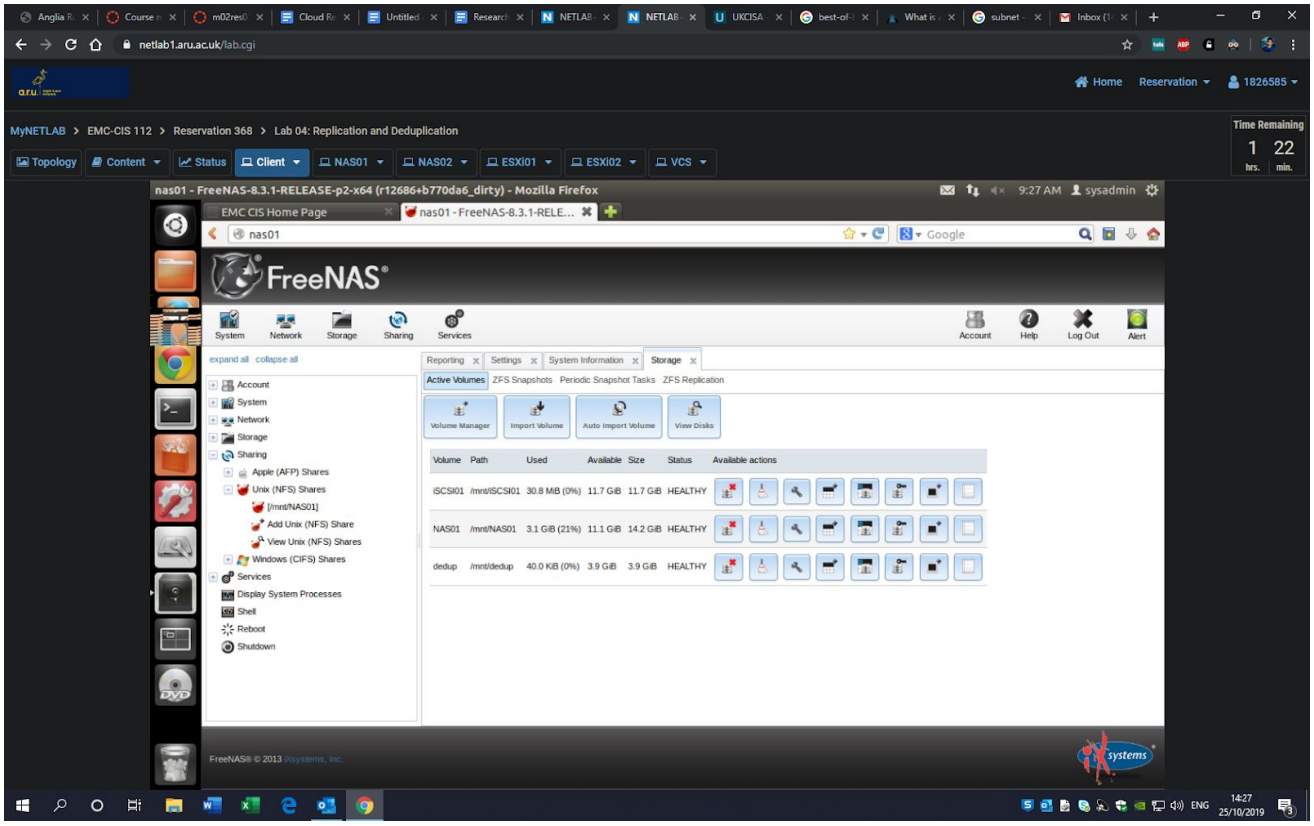
In this lab exercise, the replication and deduplication in a FreeNAS system is learned. The below screenshots show how it is done.

During deduplication on the ZFS file system, the FreeNAS uses excess RAM which reduces the performance.

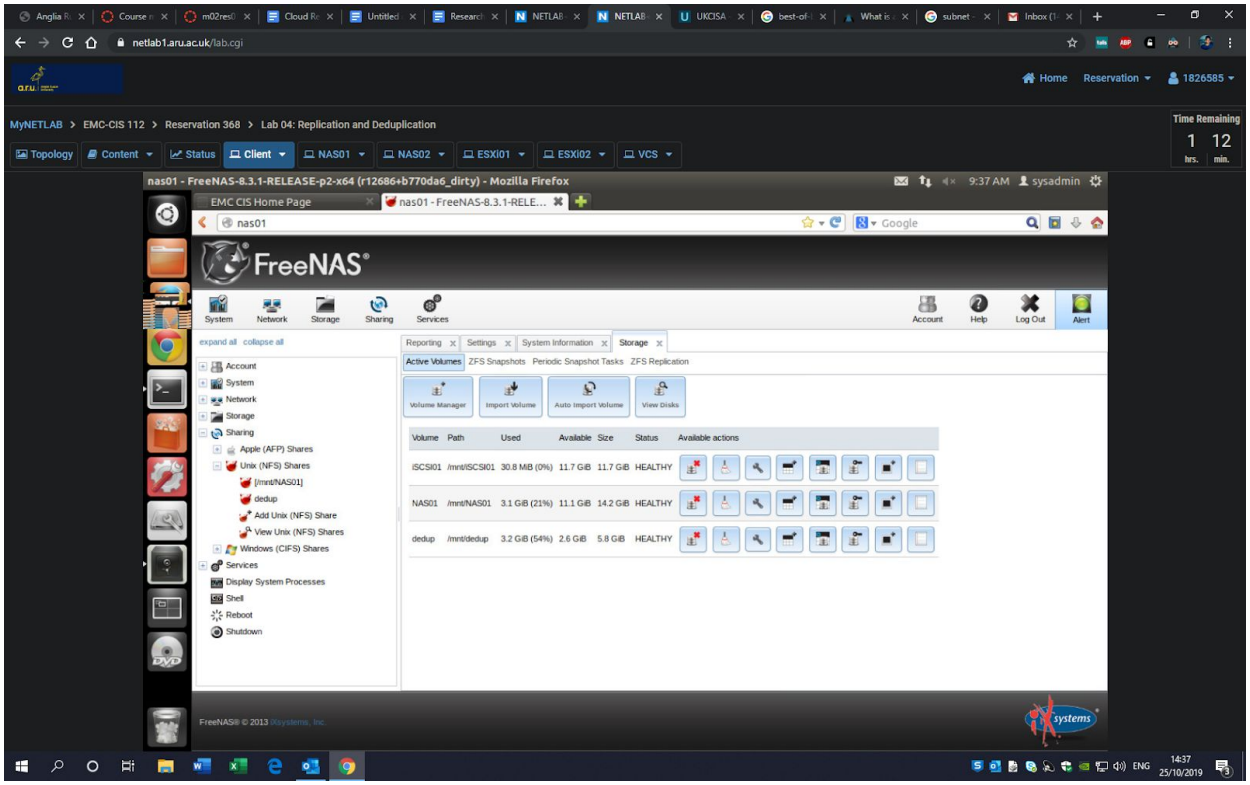
The replication of snapshots from NAS01 to NAS02 has been practiced as shown in the below screenshots.

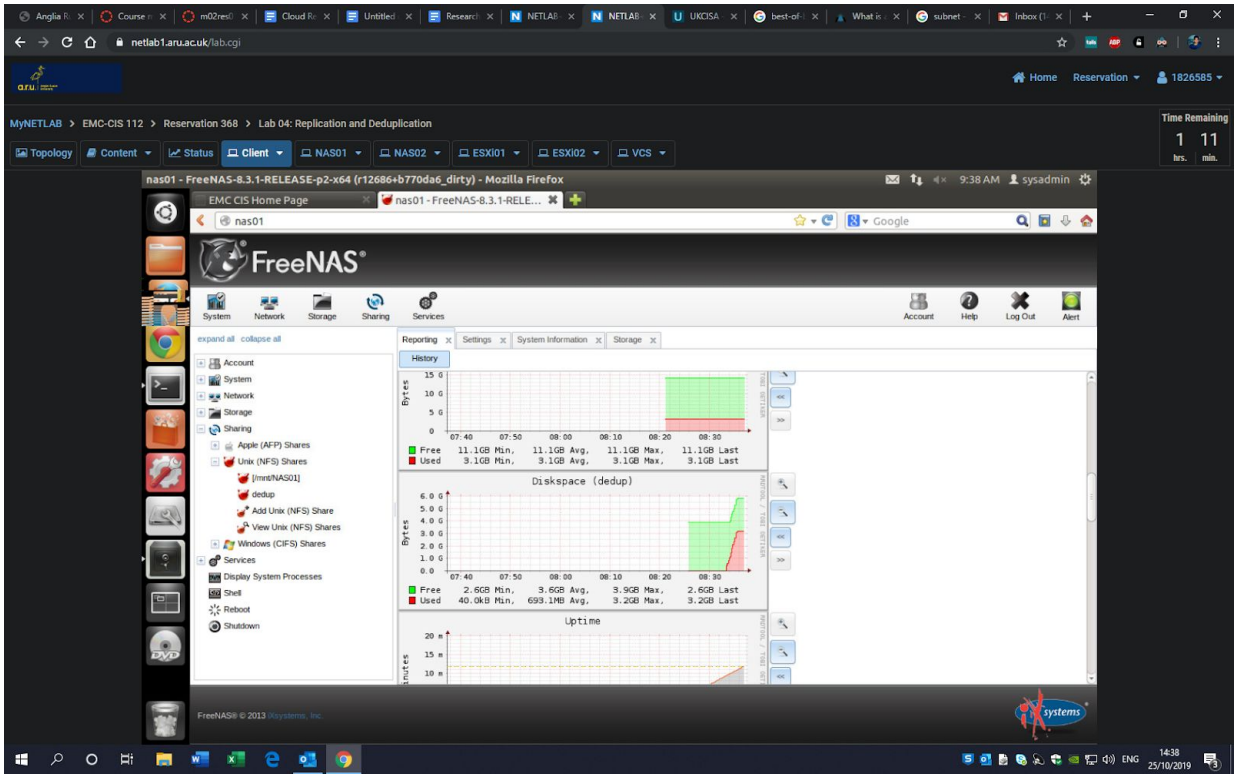
Replication of virtual machines is very important and useful in disaster recovery scenarios. This tool enables full-volume point-in-time recovery during disaster recovery.

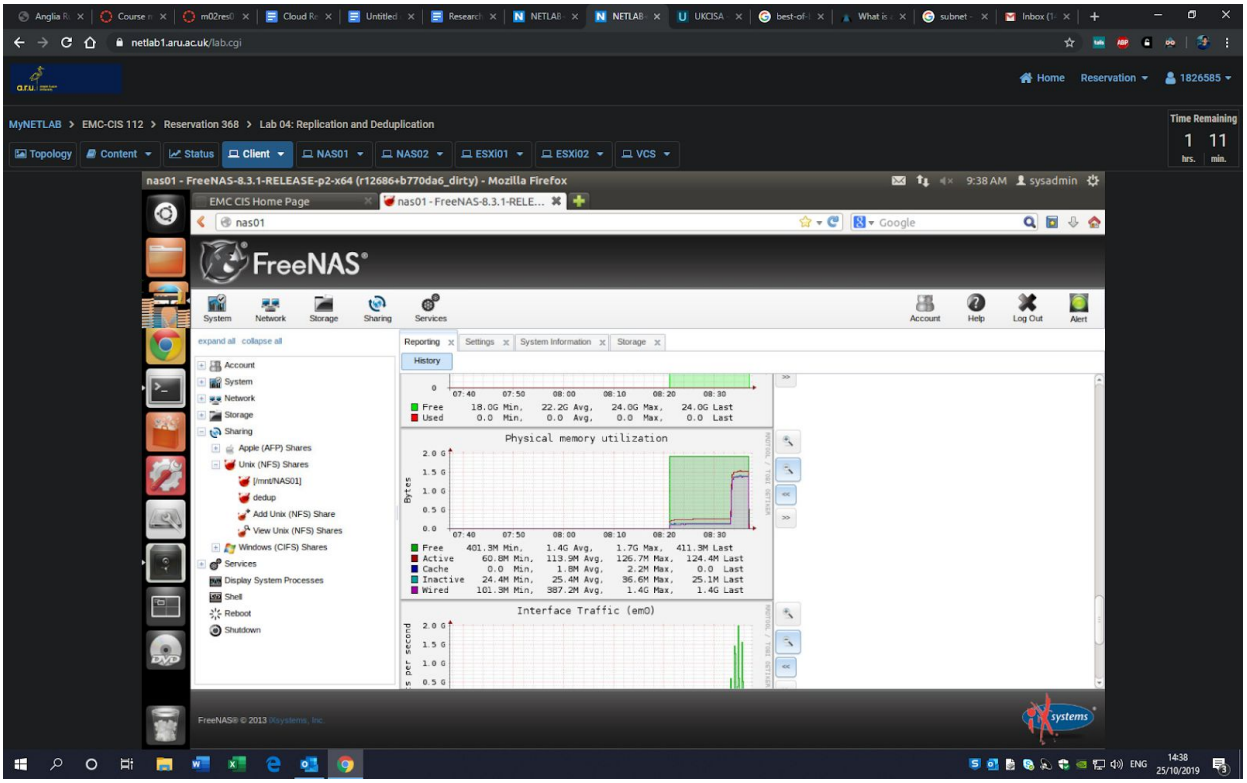
1.2 Setup the dedup Volume and the NFS Share



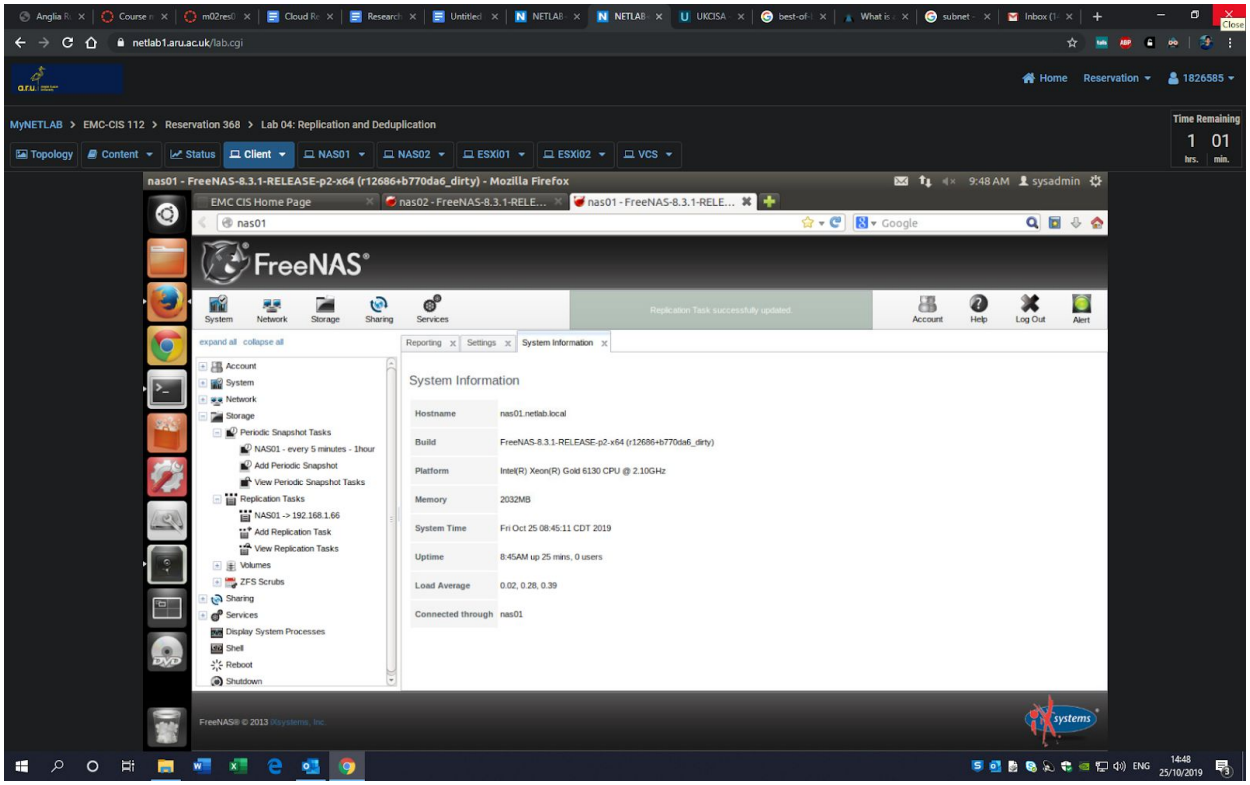
1.3 Copy the CentOS Virtual Machines from NAS01 to dedup

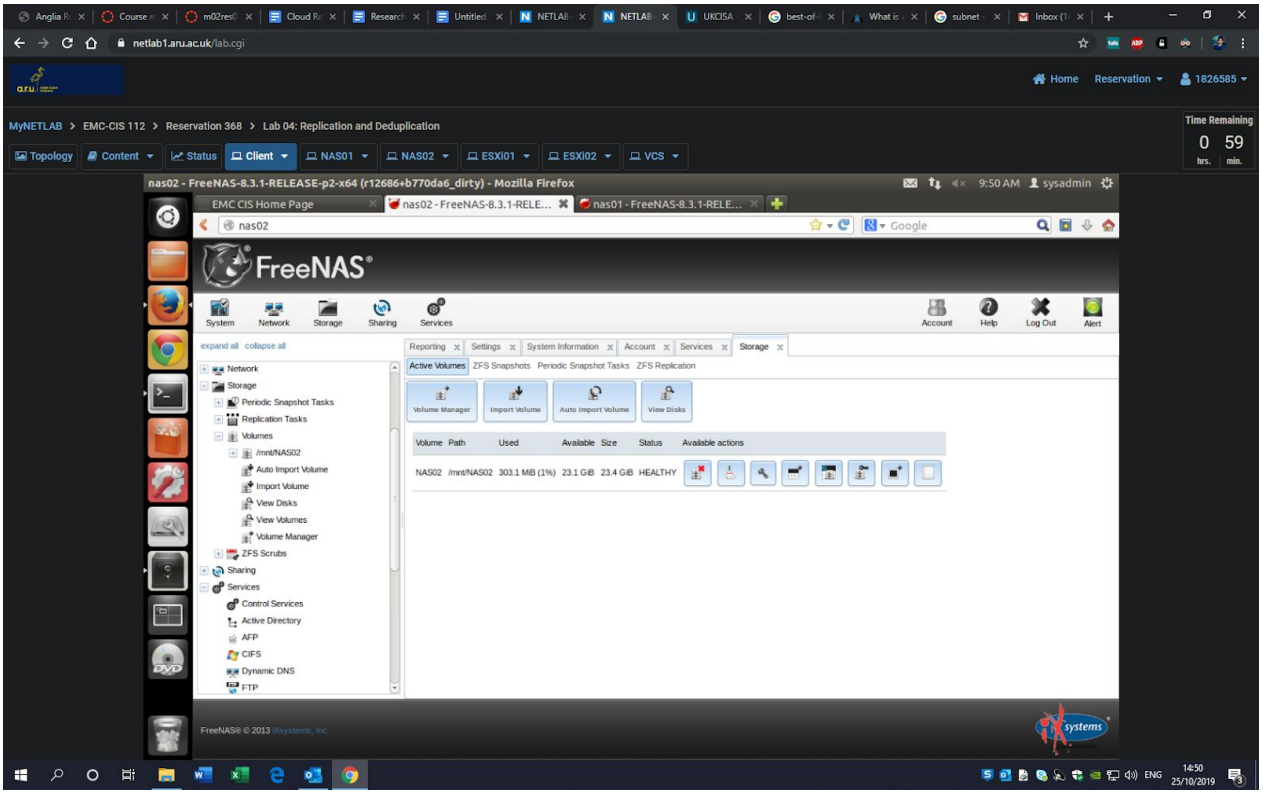

netlab1.aru.ac.uk/lab.cgi
MyNETLAB > EMC-CIS 112 > Reservation 368 > Lab 04: Replication and Deduplication
Time Remaining
1 12
nas01 - FreeNAS-8.3.1-RELEASE-p2-x64 (r12686+b770da6_dirty) - Mozilla Firefox
FreeNAS
Volume Manager
Import Volume
Auto Import Volume
View Disks
FreeNAS® © 2013 iXsystems, Inc.

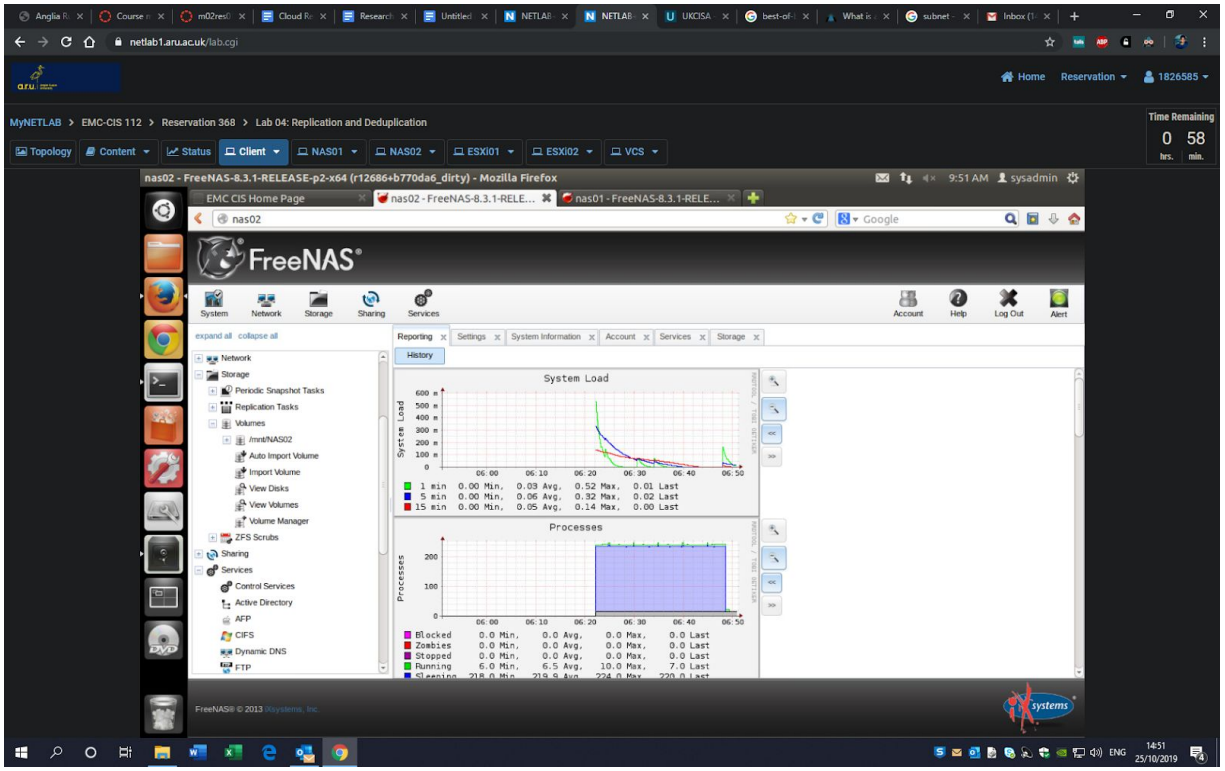

netlab1.aru.ac.uk/lab.cgi
MyNETLAB > EMC-CIS 112 > Reservation 368 > Lab 04: Replication and Deduplication
Time Remaining
1 11
nas01 - FreeNAS-8.3.1-RELEASE-p2-x64 (r12686+b770da6_dirty) - Mozilla Firefox
FreeNAS
Diskspace (dedup)
Uptime
FreeNAS® © 2013 iXsystems, Inc.

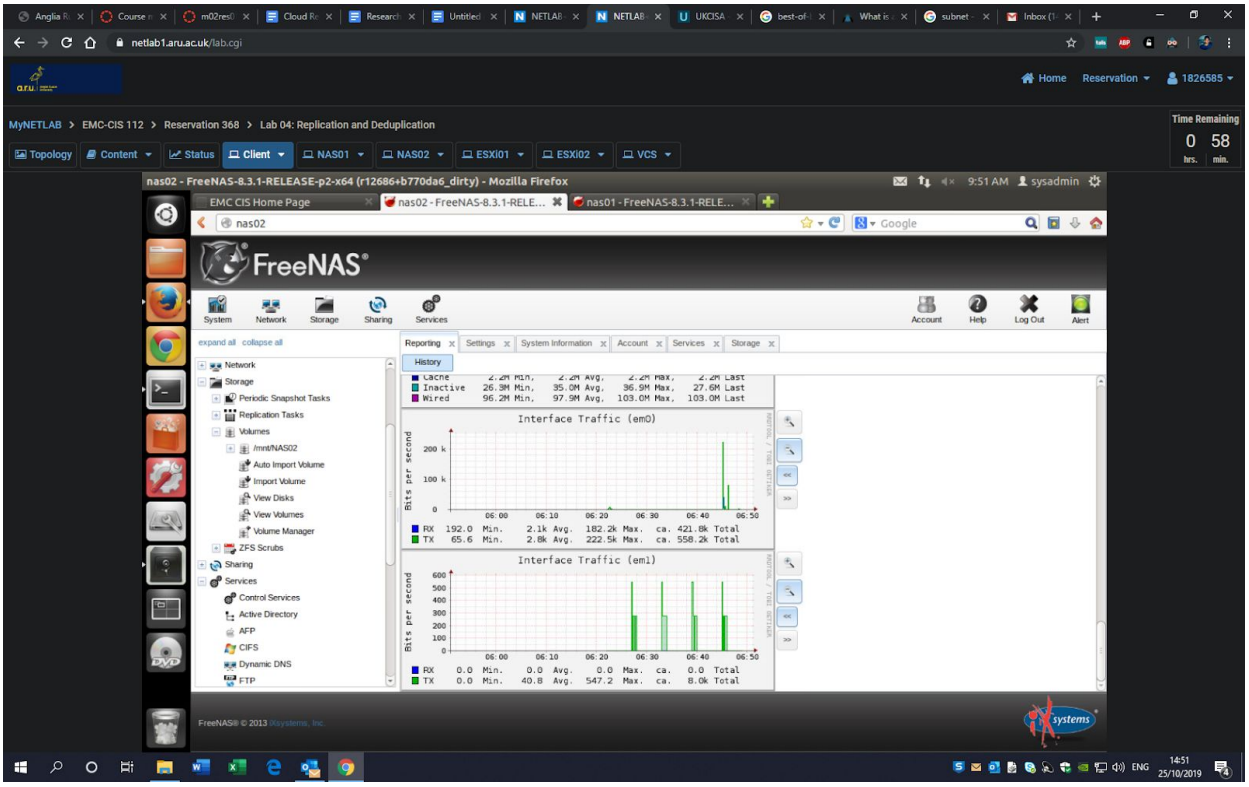


2 Setup and Configure Replication of NAS01 to NAS02


netlab1.aru.ac.uk/lab.cgi
Home
Reservation
1826585
MyNETLAB > EMC-CIS 112 > Reservation 368 > Lab 04: Replication and Deduplication
Time Remaining
1 01
hrs. min.
Topology
Content
Status
Client
NAS01
NAS02
ESXi01
ESXi02
VCS
nas01 - FreeNAS-8.3.1-RELEASE-p2-x64 (r12686+b770da6_dirty) - Mozilla Firefox
9:48 AM sysadmin
EMC CIS Home Page
nas02 - FreeNAS-8.3.1-RELE...
nas01 - FreeNAS-8.3.1-RELE...
nas01
Google
FreeNAS
System
Network
Storage
Sharing
Services
Account
Help
Log Out
Alert
Replication Task successfully updated.
expand all collapse all
Reporting
Settings
System Information
Account
System
Network
Storage
Periodic Snapshot Tasks
NAS01 - every 5 minutes - 1hour
Add Periodic Snapshot
View Periodic Snapshot Tasks
Replication Tasks
NAS01 -> 192.168.1.66
Add Replication Task
View Replication Tasks
Volumes
ZFS Scrubs
Sharing
Services
Display System Processes
Shell
Reboot
Shutdown
System Information
Hostname nas01.netlab.local
Build FreeNAS-8.3.1-RELEASE-p2-x64 (r12686+b770da6_dirty)
Platform Intel(R) Xeon(R) Gold 6130 CPU @ 2.10GHz
Memory 2032MB
System Time Fri Oct 25 08:45:11 CDT 2019
Uptime 8:45AM up 25 mins, 0 users
Load Average 0.02, 0.28, 0.39
Connected through nas01
FreeNAS® © 2013 iXsystems, Inc.
ENG
14:48
25/10/2019


netlab1.aru.ac.uk/lab.cgi
Home
Reservation
1826585
MyNETLAB > EMC-CIS 112 > Reservation 368 > Lab 04: Replication and Deduplication
Time Remaining
0 59
hrs. min.
Topology
Content
Status
Client
NAS01
NAS02
ESXi01
ESXi02
VCS
nas02 - FreeNAS-8.3.1-RELEASE-p2-x64 (r12686+b770da6_dirty) - Mozilla Firefox
9:50 AM sysadmin
EMC CIS Home Page
nas02 - FreeNAS-8.3.1-RELE...
nas01 - FreeNAS-8.3.1-RELE...
nas02
Google
FreeNAS
System
Network
Storage
Sharing
Services
Account
Help
Log Out
Alert
expand all collapse all
Reporting
Settings
System Information
Account
Services
Storage
Active Volumes
ZFS Snapshots
Periodic Snapshot Tasks
ZFS Replication
Volume Manager
Import Volume
Auto Import Volume
View Disks
Volume Path Used Available Size Status Available actions
NAS02 /mnt/NAS02 303.1 MiB (1%) 23.1 GiB 23.4 GiB HEALTHY
Network
Storage
Periodic Snapshot Tasks
Replication Tasks
Volumes
/mnt/NAS02
Auto Import Volume
Import Volume
View Disks
View Volumes
Volume Manager
ZFS Scrubs
Sharing
Services
Control Services
Active Directory
AFP
CIFS
Dynamic DNS
FTP
FreeNAS® © 2013 iXsystems, Inc.
ENG
14:50
25/10/2019

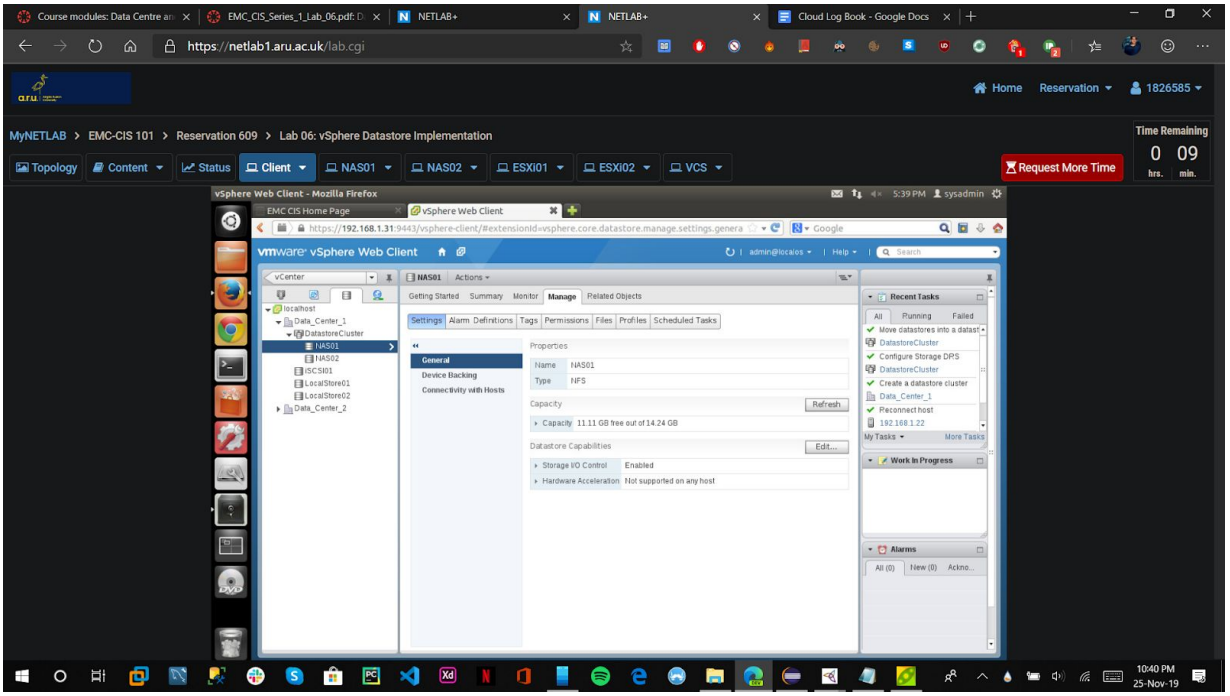

netlab1.aru.ac.uk/lab.cgi
MyNETLAB > EMC-CIS 112 > Reservation 368 > Lab 04: Replication and Deduplication
Time Remaining
0 58
nas02 - FreeNAS-8.3.1-RELEASE-p2-x64 (r12686+b770da6_dirty) - Mozilla Firefox
FreeNAS
System Load
Processes


netlab1.aru.ac.uk/lab.cgi
MyNETLAB > EMC-CIS 112 > Reservation 368 > Lab 04: Replication and Deduplication
Time Remaining
0 58
nas02 - FreeNAS-8.3.1-RELEASE-p2-x64 (r12686+b770da6_dirty) - Mozilla Firefox
FreeNAS
Interface Traffic (em0)
Interface Traffic (em1)

Lab 09: vSphere Datastore Implementation

In this lab exercise, the creation of a datastore cluster is practiced and learned. The final step is shown in the screenshot below. After the creation, one can use the vSphere Storage DRS to manage storage resources.

A Datastore cluster provides an integrated data resource management network. The system can balance the use of the datastore based on input/output operations and latency by leveraging Storage DRS within the cluster.



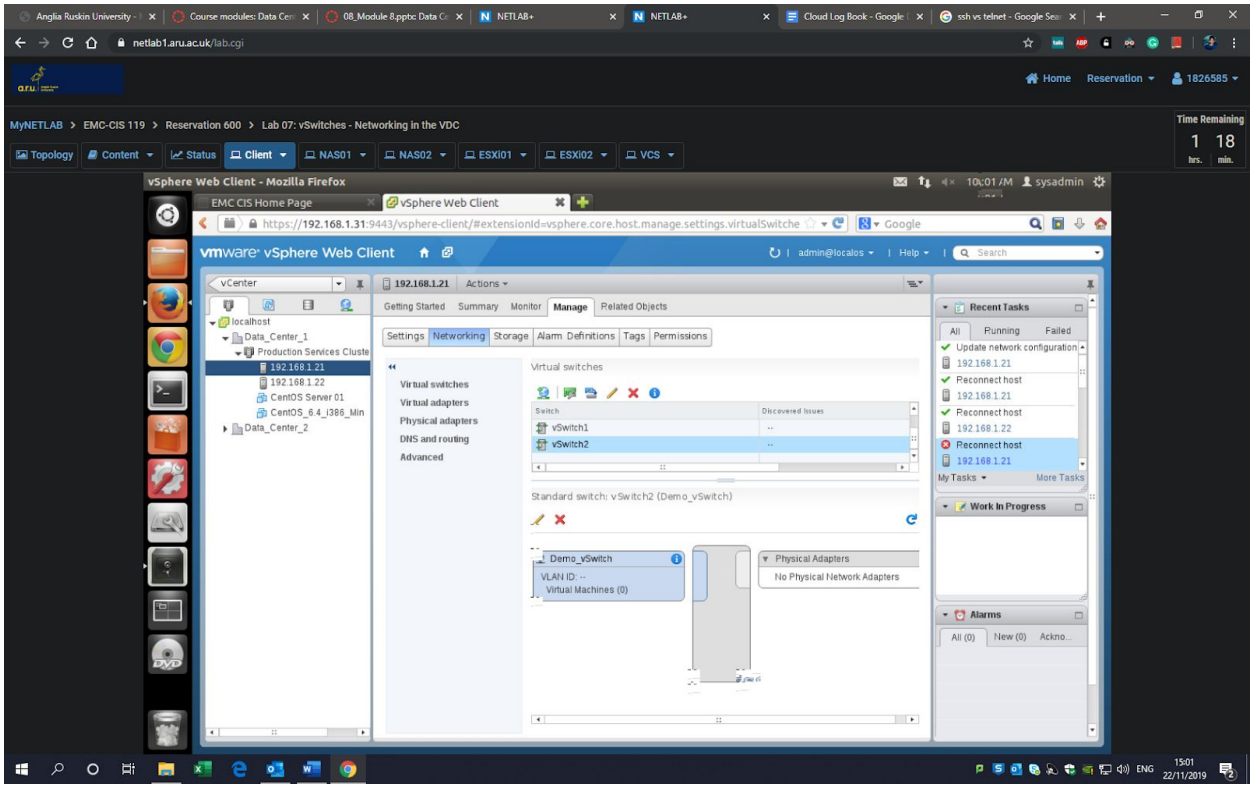
Lab 10: vSwitches – Networking in the VDC

In this lab exercise, the creation and configuration of a standard vSwitch and a distributed vSwitch were practiced and learned as shown in the screenshots below.

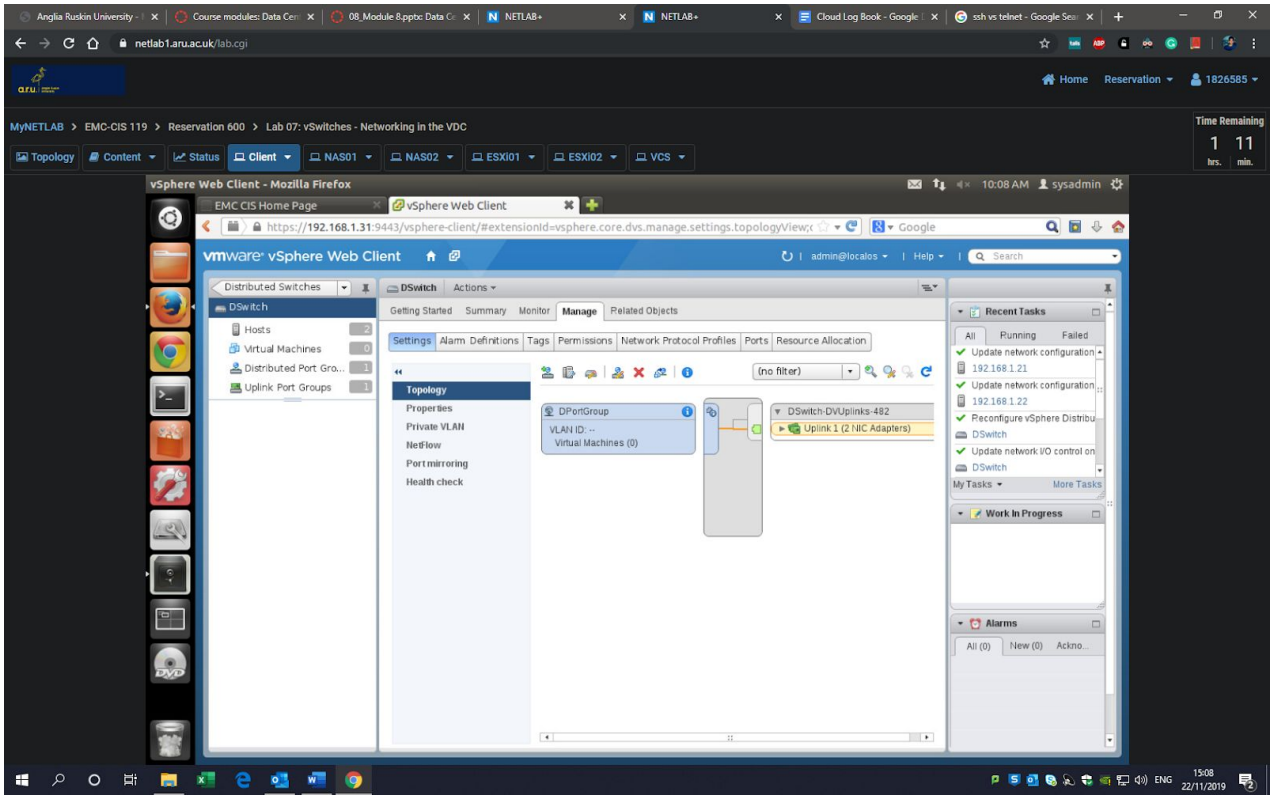
vSphere Standard Switch is used to provide network connectivity for hosts, and to handle virtual machine Traffic that works with only with one ESXi host.

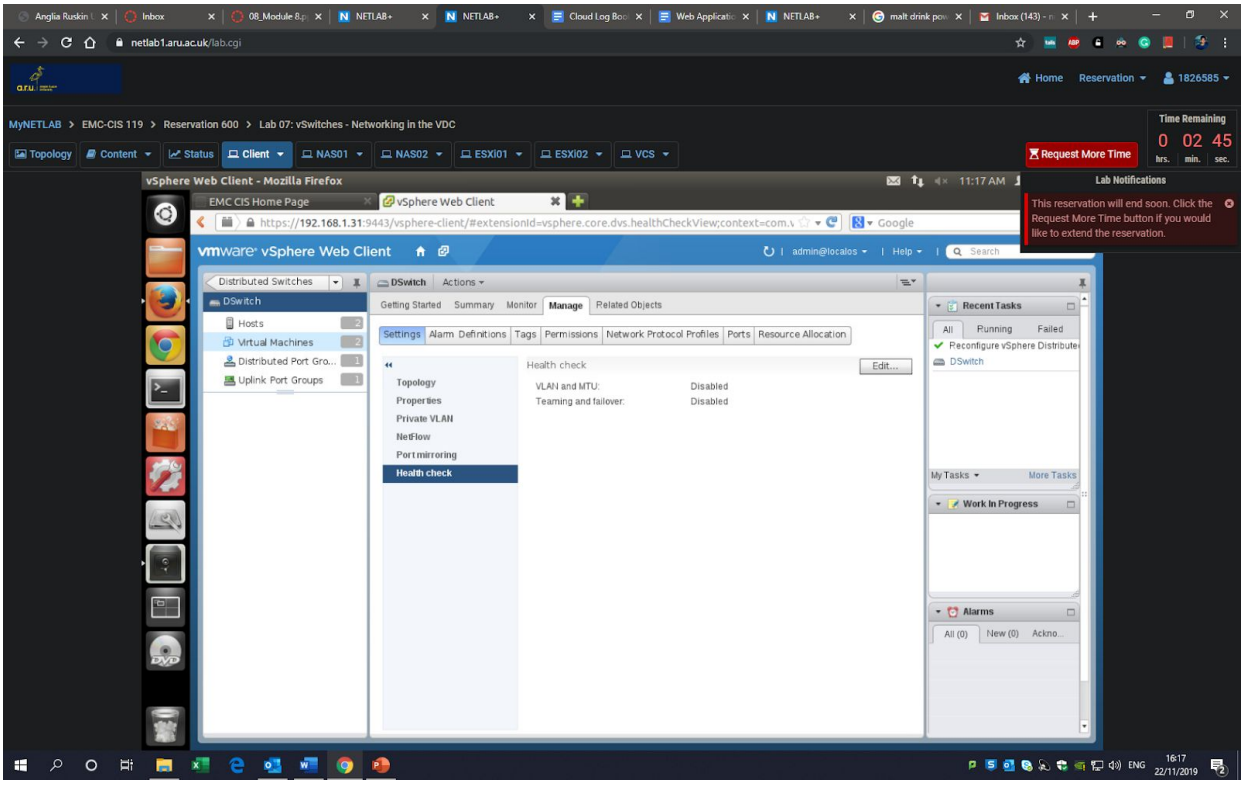
vSphere Distributed switch allows a single virtual switch to connect multiple Esxi hosts. It handles the networking configuration of multiple hosts at a time from a central place.

1. vSwitch



2. Distributed vSwitch

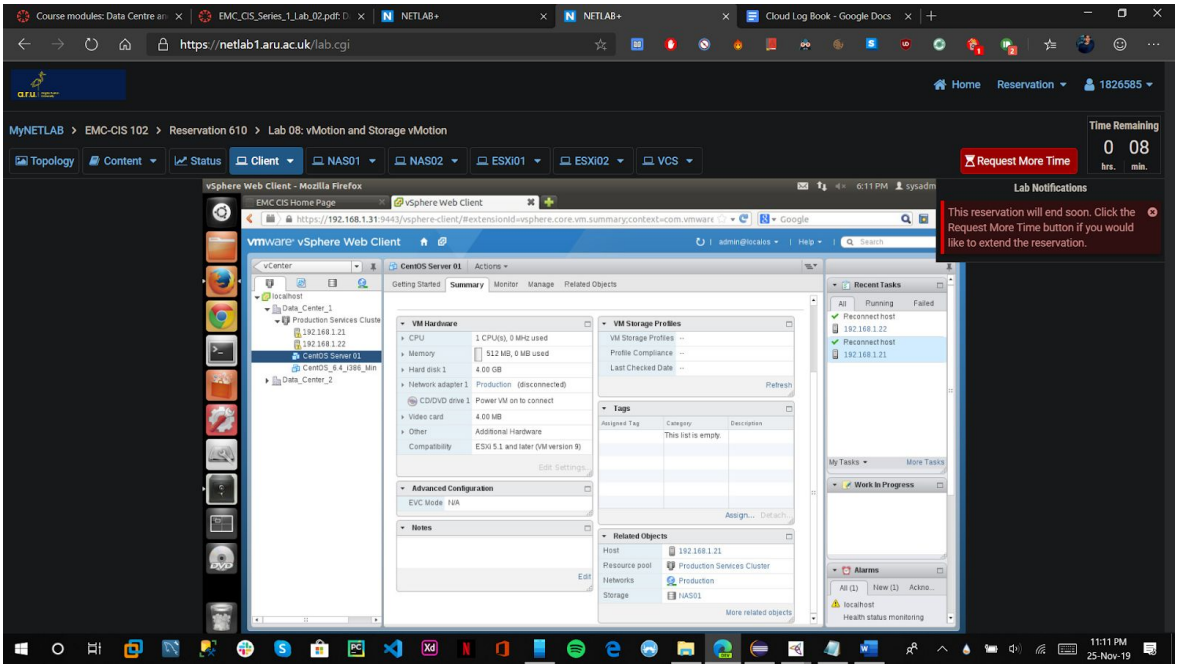

netlab1.aru.ac.uk/lab.cgi
Home
Reservation
1826585
MyNETLAB > EMC-CIS 119 > Reservation 600 > Lab 07: vSwitches - Networking in the VDC
Time Remaining
1 hrs. 11 min.
Topology
Content
Status
Client
NAS01
NAS02
ESXi01
ESXi02
VCS
vSphere Web Client - Mozilla Firefox
10:08 AM
sysadmin
EMC CIS Home Page
vSphere Web Client
https://192.168.1.31:9443/vsphere-client/#extensionId=vsphere.core.dvs.manage.settings.topologyView;c
vmware vSphere Web Client
admin@localos
Help
Search
Distributed Switches
DSwitch
Actions
Getting Started
Summary
Monitor
Manage
Related Objects
Settings
Alarm Definitions
Tags
Permissions
Network Protocol Profiles
Ports
Resource Allocation
Hosts
Virtual Machines
Distributed Port Gro...
Uplink Port Groups
Topology
Properties
Private VLAN
NetFlow
Port mirroring
Health check
(no filter)
DPortGroup
VLAN ID: --
Virtual Machines (0)
DSwitch-DVUplinks-482
Uplink 1 (2 NIC Adapters)
Recent Tasks
All
Running
Failed
Update network configuration
192.168.1.21
Update network configuration
192.168.1.22
Reconfigure vSphere Distribu
DSwitch
Update network I/O control on
DSwitch
My Tasks
More Tasks
Work In Progress
Alarms
All (0)
New (0)
Ackno...
ENG
15:08
22/11/2019

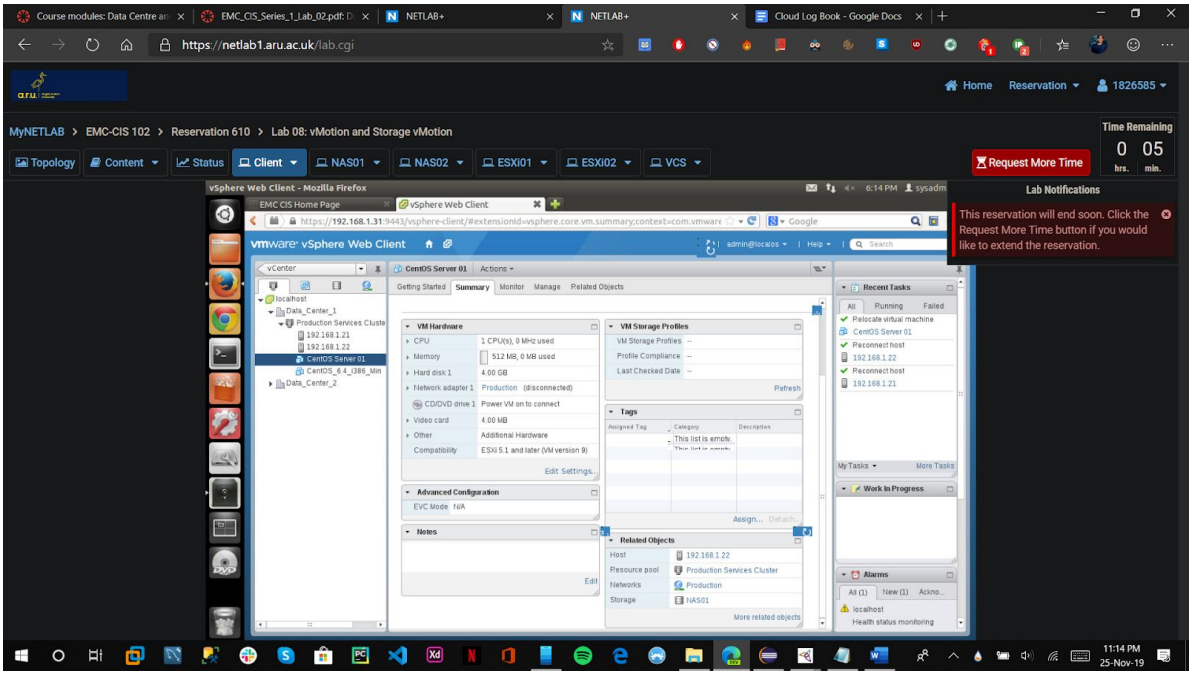

netlab1.aru.ac.uk/lab.cgi
Home
Reservation
1826585
MyNETLAB > EMC-CIS 119 > Reservation 600 > Lab 07: vSwitches - Networking in the VDC
Time Remaining
0 hrs. 02 min. 45 sec.
Topology
Content
Status
Client
NAS01
NAS02
ESXi01
ESXi02
VCS
Request More Time
vSphere Web Client - Mozilla Firefox
11:17 AM
Lab Notifications
This reservation will end soon. Click the Request More Time button if you would like to extend the reservation.
EMC CIS Home Page
vSphere Web Client
https://192.168.1.31:9443/vsphere-client/#extensionId=vsphere.core.dvs.healthCheckView;context=com.v
vmware vSphere Web Client
admin@localos
Help
Search
Distributed Switches
DSwitch
Actions
Getting Started
Summary
Monitor
Manage
Related Objects
Settings
Alarm Definitions
Tags
Permissions
Network Protocol Profiles
Ports
Resource Allocation
Hosts
Virtual Machines
Distributed Port Gro...
Uplink Port Groups
Topology
Properties
Private VLAN
NetFlow
Port mirroring
Health check
Health check
Edit...
VLAN and MTU:
Disabled
Teaming and failover:
Disabled
Recent Tasks
All
Running
Failed
Reconfigure vSphere Distribute
DSwitch
My Tasks
More Tasks
Work In Progress
Alarms
All (0)
New (0)
Ackno...
ENG
16:17
22/11/2019

Lab 11: vMotion and Storage vMotion

In this lab exercise, we will practice and learn how vMotion and Storage vMotion allows a virtual machine to migrate between compute and storage resources without being shut down or taken offline. The migrations are shown in the screenshots below.

The vMotion technology allows the powered-on virtual machines to migrate from host to host without downtime. Whereas, in a Storage vMotion two or more storage systems must be available and adequately capable of handling the load such that one unit can be taken offline. This feature makes a method in your datacenter for maintaining or updating storage systems.





2 Storage vMotion