Comparing performance of HyperV and VMware considering Network Isolation in virtual machines

Vishrutha Adla

Submitted as part of the requirements for the degree
of MSc in Cloud Computing
at the School of Computing,
National College of Ireland
Dublin, Ireland.

September 2013

Supervisor Mr. Vikas Sahni

Declaration

I confirm that the work contained in this MSc Dissertation report has been composed solely by myself and has not been accepted in any previous application for a degree. All sources of information have been specifically acknowledged and all verbatim extracts are distinguished by quotation marks.

Signed:..... Date: 13-09-13

Vishrutha Adla

Acknowledgement

I would like to express my sincere gratitude to my supervisor, Vikas Sahni, for the continuous support and advice provided in every way throughtout the dissertation. Your patience and support helped me finish this dissertation and your valuable guidance helped me explore challenging tasks. I am thankful to you and could not have wished for a better supervisor.

I would like to appreciate the staff from the School of Computing. The support and help I received throughout the completion of my Masters was very much appreciable.

A special thanks to all staff at NCI, your support and helpfulness in all areas was very much appreciated.

I am thankful to my parents, Shai Reddy Adla and Indira for the love and support you showered on me whole my life. To my brother Vishesh Reddy Adla, you always supported and encouraged me.

Abstract

Cloud computing is the set of resources and services which are offered through the internet. These resources are delivered globally from data centers located throughout the world. Cloud services use virtualisation technology to provide better resource management. The biggest challenge facing cloud computing is security as resources are shared between various users. In addition, recent technological improvements in virtualisation are also the reason for breaches in the security of cloud services. To overcome this problem to some extent, isolation of virtual machines is provided. This will ensure security and protection of applications and cloud services from malicious attacks that arises due to resource sharing in virtualisation.

Virtual machines are enabled with VLANs and their performance is measured using benchmarking tools. Network isolation is crucial to enhance the performance of virtual machines present in any hypervisors. Isolation of virtual machines will greatly remove the overheads of one VM influencing the performance of another virtual machine. Results delivered from the tests indicate that there is little difference between the two hypervisors in terms of performance, when they are considered in individual VMs. However, VMware proves to deliver better performance than HyperV. Virtual machine isolation at hardware-level and network isolation is considered suitable to resource allocation for virtual machines and to improve their performances.

Keywords: Hypervisors, virtual machines, virtualization, network virtualization, network isolation, performance, VLANs, HyperV, VMware ESXi.

Contents

Declaration ii

Acknowledgement iii

Abstract iv

1 Introduction 1
- 1.1 Overview: . . . 1
- 1.2 Research Question: . . . 1
- 1.3 Aim: . . . 2
- 1.4 Objective: . . . 2
- 1.5 Research Structure: . . . 3

2 Background 4
- 2.1 Introduction: . . . 4
- 2.2 Hypervisors: . . . 4
  - 2.2.1 Types of Hypervisors: . . . 5
  - 2.2.2 Risks of Hypervisors: . . . 5
- 2.3 Security: . . . 6
  - 2.3.1 Security Breaches: . . . 7
- 2.4 Network: . . . 7
  - 2.4.1 VLANs: . . . 8
  - 2.4.2 VPN: . . . 9
  - 2.4.3 Overlay Network: . . . 9
  - 2.4.4 Active Networks: . . . 9
- 2.5 Isolation: . . . 9
  - 2.5.1 What is Isolation? . . . 9
  - 2.5.2 Purpose of Isolation: . . . 10
  - 2.5.3 Isolation in Hypervisors: . . . 10
  - 2.5.4 Uses of Isolation: . . . 10

2.5.5 Benefits of Isolation: . . . 10
2.5.6 Types of Isolation: . . . 10
2.5.7 Network Isolation: . . . 11
2.5.8 Uses of Network Isolation: . . . 12
2.5.9 Background on Network Isolation: . . . 12
2.6 Resource allocation in isolated VMs: . . . 13
2.7 Virtualization: . . . 15
2.7.1 Types of Virtualization: . . . 16
2.7.2 Network Virtualization: . . . 16
2.7.3 Background on Network Virtualization: . . . 17
2.8 Performance: . . . 17
2.8.1 Network Performance: . . . 17
2.8.2 CPU Performance: . . . 17
2.8.3 Memory Performance: . . . 17
2.8.4 Disk Performance: . . . 17
2.9 Conclusion: . . . 18

3 Literature Review **19**
3.1 Introduction: . . . 19
3.2 Isolation: . . . 19
3.2.1 Why Network Isolation? . . . 20
3.3 Providing Isolation between virtual machines and related issues: . . . 21
3.4 Benefits of VM Isolation: . . . 23
3.5 Hypervisors: . . . 24
3.5.1 Hypervisors Available: . . . 25
3.6 Tools for performance measurements: . . . 25
3.6.1 Network Performance: . . . 25
3.6.2 Disk Performance: . . . 26
3.6.3 Memory Performance: . . . 27
3.6.4 CPU Performance: . . . 28
3.7 Conclusion: . . . 28

4 Design **29**
4.1 Introduction: . . . 29
4.2 System Description: . . . 29
4.3 VLAN: . . . 30
4.4 HyperV: . . . 31
4.5 VMware ESXi: . . . 32
4.6 Pseudo IP addresses for virtual machines: . . . 33

4.7 Hardware and Software Configurations: . . . 34
4.7.1 Hardware Configurations: . . . 34
4.7.2 Software Configurations: . . . 34
4.8 Tool for Network performance: . . . 35
4.9 Tool for Disk performance : . . . 35
4.10 Tool for CPU performance: . . . 35
4.11 Tool for Memory performance: . . . 36
4.12 Conclusion: . . . 36

5 Implementation **37**
5.1 Introduction: . . . 37
5.2 Logical networks in HyperV: . . . 37
5.3 Installation of Tools: . . . 39
5.3.1 Iperf Tool Installation: . . . 39
5.3.2 Passmark Tool Installation: . . . 39
5.3.3 IOzone Tool Installation: . . . 40
5.3.4 Memtest Tool Installation: . . . 41
5.4 Benchmark readings for tools: . . . 42
5.4.1 Benchmark readings for Iperf: . . . 42
5.4.2 Benchmark readings for Passmark: . . . 43
5.4.3 Benchmark readings for IOzone: . . . 44
5.4.4 Benchmark readings for Memtest: . . . 45
5.5 Logical networks in VMware: . . . 46
5.6 Benchmark readings for tools: . . . 47
5.6.1 Benchmark readings for Iperf: . . . 47
5.6.2 Benchmark readings for Passmark: . . . 47
5.6.3 Benchmark readings for IOzone: . . . 47
5.6.4 Benchmark readings for Memtest: . . . 48
5.7 Conclusion: . . . 48

6 Evaluation **49**
6.1 Introduction: . . . 49
6.2 Graph charts for performance test on HyperV: . . . 49
6.2.1 Iperf analysis: . . . 49
6.2.2 Passmark analysis: . . . 50
6.2.3 IOzone analysis: . . . 50
6.2.4 Memtest analysis: . . . 51
6.3 Graph charts for performance test on VMware ESXi: . . . 51
6.3.1 Iperf analysis: . . . 52

6.3.2 Passmark analysis: . . . 52
6.3.3 IOzone analysis: . . . 53
6.3.4 Memtest analysis: . . . 53
6.4 Conclusion: . . . 56

7 Conclusions **57**
7.1 Overview: . . . 57
7.2 Further Work: . . . 58

Bibliography **59**

A Appendix **61**
A.0.1 VM creations on HyperV and VMware ESXi: . . . 61
A.1 Tool Implementation: . . . 65
A.1.1 Configuration of HyperV and VMware ESXi: . . . 68

List of Figures

2.1 Types of Hypervisors 5
2.2 Type 1 Hypervisor 6
2.3 Type 2 Hypervisor 6
2.4 Hypervisor Attacks 7
2.5 Architecture of VLAN 8
2.6 Virtual machines traffic distribution 12
2.7 CPU utilization among different types of VMs 14
2.8 Performance of Workloads and their CPU Utilization 15

4.1 VLAN Architecture 31
4.2 HyperV VLAN Architecture 32
4.3 VMware VLAN Architecture 33
4.4 Virtualization Charactersitics Considerations 33

5.1 Networking Architecture 38
5.2 Logical networks 38
5.3 Iperf Installation 39
5.4 Passmark Installation 40
5.5 IOzone Installation 41
5.6 Memtest Installation 41
5.7 Iperf test 42
5.8 Passmark test 43
5.9 IOzone test 44
5.10 Memtest test 45
5.11 Networking Architecture 46

6.1 Bandwidth performance of VMs 50
6.2 Passmark performance of VMs 50
6.3 IOzone performance of VMs 51
6.4 Memory performance of VMs 51

6.5 Iperf performance of VMs 52
6.6 Passmark performance of VMs 53
6.7 IOzone performance of VMs 53
6.8 Memtest performance of VMs 54
6.9 Performance of VMs on HyperV and VMware by Iperf 54
6.10 Performance of VMs on HyperV and VMware by Passmark 55
6.11 Performance of VMs on HyperV and VMware by IOzone 55
6.12 Performance of VMs on HyperV and VMware by Memtest 56

A.1 VM creation 61
A.2 VM creation 62
A.3 logical network connection 62
A.4 logical network connection 63
A.5 Assigning IP address 63
A.6 Iperf command for server connection 65
A.7 Iperf command for client connection 65
A.8 Passmark execution 66
A.9 Passmark single threaded operations execution 66
A.10 IO command for output in microseconds 67
A.11 IOzone command for read/write 67
A.12 Memtest execution 68

List of Tables

3.1 Available Tools for Network Performance 26
3.2 Tools Available for Disk Performance 27
3.3 Memory Performance Tools 27
3.4 List of Available CPU Performance Tools 28

4.1 Hardware Configurations 34
4.2 Software Configurations 34

5.1 Different Bandwidth Readings by Iperf in VMs 42
5.2 Iperf Benchmark Readings for vish2 43
5.3 Different Single Threaded Operations Readings by Passmark in VMs . . 43
5.4 Various Passmark operations readings for vish2 44
5.5 Different Read Operations Readings by IOzone in VMs 45
5.6 Different RAM Readings by Memtest in VMs 46
5.7 Different Bandwidth Readings by Iperf in VMs 47
5.8 Different Single Threaded Operations Readings by Passmark in VMs . . 47
5.9 Different Read Operations Readings by IOzone in VMs 48
5.10 Different RAM Readings by Memtest in VMs 48

Chapter 1

Introduction

1.1 Overview:

Virtualisation is beneficiary in many ways with shared computing resources providing improvements in stability and reliability of systems. Some security problems have appeared with the development of virtualisation technology. It is commonly defined as the technology that introduces a software abstraction layer between the hardware, OS and the applications running over it. This abstraction layer is called virtual machine monitor or hypervisor. When considering security for virtualised environments, isolation of the virtual machines appears as a good solution in terms of resource sharing, as this is very much common in virtualisation.

Performance isolation, network isolation and high utilization are the key requirements when sharing resources such as network bandwidth. Performance isolation proves to be difficult as resources are shared and performance of one VM influences the performance of the other.

To minimise this problem, network isolation is considered. When network isolation is considered the sharing of resources in the virtual machines is difficult which will ensure fair resources allocation for them and in turn performance is also improved.

1.2 Research Question:

Cloud computing has become the most popular concept in terms of resource management as well as provisioning resources to the internet services. It has become an

important and vital part of internet services these days. When dealing with provisioning resources, there may occur a problem of security called isolation. There is a need to focus on this problem of security as most of the resources requested by users may be underutilized or over utilized. Moreover, user data is also stored on cloud which raises a question of user data privacy. When proper isolation is not provided, it can degrade the performance of either the hypervisor(virtual machine monitor) or the virtual machines. These basic concepts on which the research is based upon are elaborated in chapter 2 of the thesis.

Therefore, this thesis focuses on performing comparative analysis on virtual machines residing on two hypervisors namely HyperV and VMware ESXi that are provided by one of the top virtualization technologies, Microsoft and VMware. To complete comparison, network isolation is considered among virtual machines.

1.3 Aim:

The aim of this thesis is to conduct performance tests on virtual machines when they are connected to a separate VLAN by creating logical networks. As virtual machines reside on these hypervisors, their behaviour is tested for hardware parameters like network, CPU, disk, memory which will determine the performance. In order to proceed with performance test, virtual machines are connected to separate VLANs. Enabling individual VLANs characterises network isolation feature for virtual machines. This feature of network isolation enabled in VMs is used to perform tests on them. Isolating VMs in network is identified as a crucial part of the thesis.

1.4 Objective:

Network Isolation is one of the isolation strategies present in the security aspects of the cloud. As the name indicates, isolation is referred to as the segregation of hardware resources such as memory, disk and network. There is a possibility of improvement in the performance when the hardware resources allocated to the VMs are adequately utilized by them.

Thus, to make it more comprehensible, one of the resources(that is network) is held in isolation to test the performance of virtual machines. Network isolation is demonstrated by virtualising the network parameter like local area network, network adapters and creating logical networks.

1.5 Research Structure:

The main structure of the research is divided into seven parts. Chapter 1 Background explains the basics of isolation, hypervisors and network virtualization. Literature review is covered under chapter 3 which focuses on presenting a review of the work on which the dissertation is based upon. Following chapter 3 is the design of setting up required logical networks for both HyperV and VMware ESXi which is discussed in chapter 4.

Consecutively, after design is the part of implementing the above discussed by taking some benchmarking tools for resources mentioned above. Chapter 5 characterized by the implementation of benchmarking tools that help test the performance. The approach for comparative analysis that is conducted to produce results is evaluated in chapter 6. Finally, conclusion is discussed in chapter 7.

Chapter 2

Background

2.1 Introduction:

This chapter covers the concepts of hypervisors, their types, risks associated with hypervisors, isolation concepts and network virtualization. The aim of conducting research background is to develop a basic idea of isolation, its underlying features and concepts. Various works have also been done previously on these topics. The purpose of presenting a background is to divide the research area into sections to clearly understand each concept.

Hypervisors, their types, architecture and the associated risks are explained in sections 2.1 to 2.1.2. Security and its breaches are discussed in sections 2.3 and 2.3.1. Section 2.4 presents an overview of network and its types are presented in sections 2.4.1 to 2.4.3. Isolation, its types, purpose, uses and benefits are elaborated in sections 2.5.1 to 2.5.6. Network Isolation is explained in 2.5.7. Virtualization and its types are discussed in sections 2.6. Finally, section 2.7 presents an overview of performance measurement techniques/ tools.

2.2 Hypervisors:

This section provides an overview of hypervisors. Hypervisor otherwise called as virtual machine monitor (VMM) is virtualisation software platform for computer hardware and software. Multiple operating systems can be run concurrently on a host with the help of hypervisors. Physical resources of the computing system are hidden from the operating system by using hypervisors. Hypervisors act as an abstraction layer between the

hardware and the operating system as well as the applications which run on it.

Hardware resources are managed by this thin layer(VMM) that exports a uniform interface to the VMs present on upper layers. Virtual machine instances creation, migration and termination is done by the hypervisors. If hypervisor is breached in a virtualised environment then the virtual machines are compromised due to it. Hypervisors are prone to the risk of attacks from malicious guest VMs(virtual machines). Any malicious user can gain access and control to the virtual machines of the legitimate user.

2.2.1 Types of Hypervisors:

There are generally two types of hypervisors namely Type 1 and Type 2. Hypervisor running directly on the system hardware is called Type 1 hypervisor. Hypervisor(VMM) provides virtual resources which are used by the virtual system in this case. Hypervisor running on the host operating system is called as Type 2 hypervisor. Virtualization services like I/O device support and memory management are provided by this hypervisor. Type 1 hypervisor is also called as bare metal hypervisor.Type 2 hypervisor is also called as hosted hypervisor.

Type of VMM(Hypervisor)	Architecture
VMware	Bare-metal/monolithic(type-1)
MS HyperV R2	Bare-metal/Microkernel
KVM(Ubuntu enterprise server)	Hosted
Xen(Centos)	Bare-metal/microkernel

Figure 2.1: Types of Hypervisors

Below figures illustrates the architecture for the above mentioned hypervisor types.

2.2.2 Risks of Hypervisors:

This section highlights the risks pertaining to hypervisors. Hypervisors are prone to risk of attacks from the malicious guest VMs. The two types of attacks are VM escape and Rootkit in hypervisors.

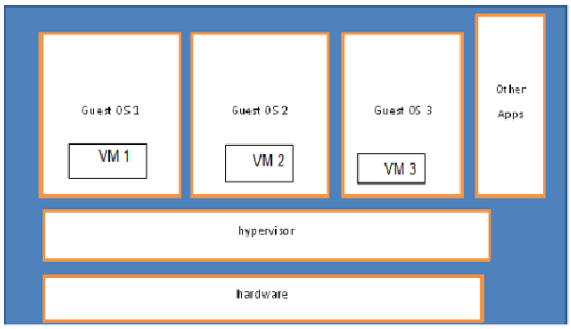

Figure 2.2: Type 1 Hypervisor

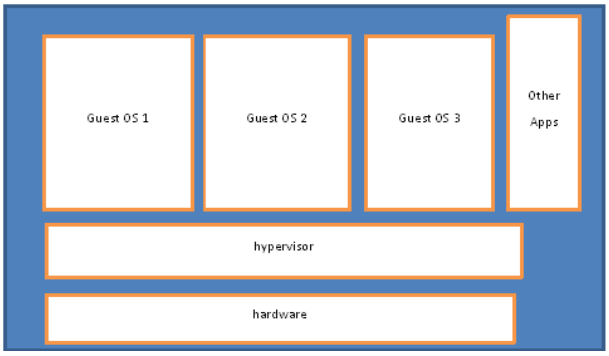

Figure 2.3: Type 2 Hypervisor

VM Escape attack occurs when the isolation layer is broke due to the intervention of an attacker's program running in a VM. This happens when the attacker tries to access hypervisor's root privileges instead of the VM privileges. Rootkit hypervisor attack occurs when the existing OS in a host is compromised to a VM initiating hypervisor by the VM-based rootkits. Control over the resources is gained by the new guest OS which assumes itself running as the host OS. Other attacks due to the cloud infrastructure breach and sharing of resources are Distributed Denial of Service(DDoS), man-in-the-middle attacks and IP spoofing. Other cloud users can also cause the risk of attacks on the information specific to the cloud subscribers.

The table below shows the attacks and their characteristics briefly,

2.3 Security:

Security in the cloud environments poses to be a great challenge because of the multi-tenancy nature (where multiple tenants try to access the same application), accessing sensitive and critical applications. Various security policies like access control, system

Attacks	Characteristics
Distributed Denial of service attack(DDoS)	When multiple requests are sent by the attacker trying to access the data exchange or any type of communication existing between the virtual machines, the cloud is overloaded by requests. This causes Denial of Service or Distributed Denial of service attack to the servers. A service is denied for the virtual machines involved in the data exchange.
Man-in-the-middle attack	Due to the poor configuration of SSL(Secure socket layer), the data exchange between any two VMs can be accessed illegally by any attacker.
Zombie attack	The virtual machines are flooded when an attacker tries and sends requests from innocent hosts in the network through the internet. The hosts are termed as Zombies.
Phishing attack	Phishing attack occurs when the user is redirected to a false link by manipulating a web link to access the sensitive data like password, personal details etc.

Figure 2.4: Hypervisor Attacks

portability, software security that includes virtualization technology, host operating system, guest operating system and data encryption and hardware security that includes backup, server location, firewall ensure to provide a secure cloud environment. The absence of security standards specific to cloud computing leads to the application of conventional security concepts.

2.3.1 Security Breaches:

The virtualisation developments lead to many security problems that appeared to be a great challenge. Security in the hypervisors is considered to be important. For example, when the virtual machines have access to same memory, there is a possibility of information flow breach. In this way, the files can be copied from one VM to the other VM(virtual machine). This problem arises due to networking and file sharing. When the resources are shared among the virtual machines, security is breached. Various attacks like VM escape, VM-level attacks, isolation failure, and Distributed denial of service attacks, man-in-the middle attacks occur due to breach in the security. An overview of isolation, its purpose and types of isolation is explained in the following sections.

2.4 Network:

Network is defined as a term which end points of a session are connected together. It is simply explained as a method where information is transferred from one endpoint to the other endpoint. To describe further, until the information that needs to reach the

receiving stack is delivered at this receiving end-point, the information is sent down the network.

Hardware devices like switches or routers are used to interconnect the transmissions lines. These transmission lines are the representation of end-to-end physical paths of the network. Transmission lines can be either wired or wireless links. Network provides end-to-end sessions with distinct characteristics depending on the desired type of communication, like connection-oriented, connection-less sessions and quality of service(QoS)guarantees for data transfer process. This logical network is categorised into four types. They are: VLANs, VPNs, overlay networks, and active and programmable networks.

2.4.1 VLANs:

A group of logically networked hosts contained in a single broadcast domain regardless of the physical connectivity is called as a VLAN(virtual local area network). Local area network comprises of network hosts situated in a particular area or a building. Thus there is a VLAN ID assigned to all the frames in a VLAN. The network administration, management, and reconfiguration of VLANs are simpler than the physical counterparts. The VLANs are based on logical connectivity rather than physical connectivity. Not only this, another advantage of using VLANs is they provide isolation. VLANs have been used for providing virtual networks isolation on a single physical network in the networking space.

An architecture of VLAN is illustrated in the below figure,

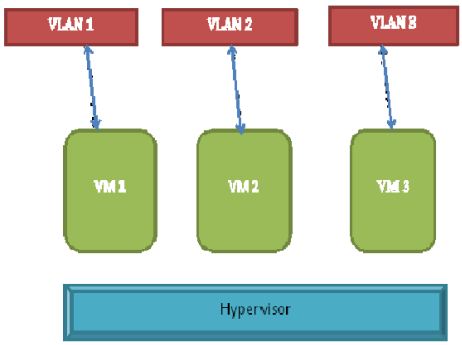

Figure 2.5: Architecture of VLAN

2.4.2 VPN:

Multiple sites are connected by using private and secured tunnels over the networks like public or shared communication networks with the help of a dedicated network called as Virtual private network(VPN). Geographically distributed sites of a single corporate enterprise are connected by VPN. VPNs are classified as layer 1 VPN, layer 2 VPN, layer 3 VPN and higher-layer VPNs.

2.4.3 Overlay Network:

A network that is built on top of one or more existing physical networks is called as an overlay network. In the existing internet, this overlay network is implemented in the application layer of OSI(open standard interconnect) layers.

2.4.4 Active Networks:

Within the confinement of existing networks, new services are dynamically deployed at runtime using active networks. Customization of network services at the packet transport and flexibility are offered by active networks.

2.5 Isolation:

This section gives an overview of isolation concept, the underlying uses and types of isolation.

2.5.1 What is Isolation?

Isolation is the property of security which separates either the virtual machines or the programs running in it from one another. A virtual machine is an efficient, isolated duplicate of a real machine according to Popek and Goldberg. Virtual machine is a logical unit of the partitioned hardware platform. This hardware is partitioned so that multiple OSs(operating systems) can be run on the same hardware in parallel. Hypervisors must be ensured of isolation between the hosted virtualized machines. This implies that one guest operating system cannot have resources more than granted for it. In the next section, purpose of isolation is presented.

2.5.2 Purpose of Isolation:

Isolation limits the usage of resources between the virtual machines of hypervisors thus reducing the overhead of resource sharing. Due to sharing of the resources, files/ programs present in one virtual machine are copied into the other virtual machine. So isolation avoids this problem to maximum extent. VMM (virtual machine monitor) protection is provided by isolation.

2.5.3 Isolation in Hypervisors:

This part of the Isolation section emphasises on providing isolation in hypervisors. Isolation is avoided in some virtualizations. This scenario rises when applications which are designed for one operating system have to run in another operating system. This yields to exploiting the security bearer in both the OSs.

2.5.4 Uses of Isolation:

Users are provided with isolation from physical capacity of virtual machine movement between computers and clouds. Isolation can be implemented at Hardware level by either direct access to the hardware or by the hardware-assisted execution. It can also be implemented by the hypervisor base approach.

2.5.5 Benefits of Isolation:

Virtual machines can have their own hardware resources allocated to them avoiding any malicious VM from utilizing all the resources. Isolation helps to segregate the user data from one another. It helps to avoid overhead of network traffic flow.

2.5.6 Types of Isolation:

There are various types of isolation mechanisms in virtualization. They are:

1. Security Isolation: This type of isolation provides security features between the virtual machines. For example, when a malicious program attacks a virtual machine, other virtual machines should work normally.

2. Resource Isolation: This type of isolation between the virtual machines ensures security as well as the resource isolation. Information is not disclosed between the

virtual machines which mean that one virtual machine cannot see the information present in another virtual machine.

3. Performance isolation: This type of isolation provides performance, resource, security isolation between the virtual machines. Performance isolation also separates the network traffic of services running in virtual machines. The traffic flowing in one service should not affect the traffic running in other service assuming that they are separated by a physical switch.

Types of Isolation Adding further to this, isolation is also divided into three types based on the isolation provided at individual VM. They are hypervisor-based approach, hardware-assisted isolated execution approach and the third is direct hardware access. In the hypervisor-based approach, the hypervisor can be either hardened or minimised. Hardware functions are used by the system to maintain isolation in hardware-assisted isolated execution. In direct hardware access type of approach, the hardware is directly accessed in-order to provide the necessary isolation. Based on the operating system and applications that run on VMs, isolation is classified as hard isolation and soft isolation. Hard isolation is provided by the VMs when the applications hosted on same physical machine share the resources. Soft isolation is provided when the system is underutilized and there is need for resource sharing. There is another type of isolation called as network isolation which is described in the next section.

2.5.7 Network Isolation:

This section gives description about the network isolation phenomenon. In this type of isolation, the network is separated for each virtual machine. There are two ways of providing this type of isolation: network traffic isolation and network security isolation. The user traffic needs to be isolated which will ensure security, higher bandwidth. Implementation of VLANs for specific tenants also ensures security among the virtual machines. In case of network security isolation, database traffic must be authenticated against breaches when the network is in consolidated environments.

The four networking modes such as system, static, managed, managed-no VLAN are the networking modes on which network isolation can be performed. Networking, OS and hypervisor-based machine are the three levels on which machine instance isolation can be provided. The next section covers virtualization.

Below figure depicts the traffic distribution of virtual machines.

The above fig 1.1 gives the description of Virtual machines traffic distribution. For VM

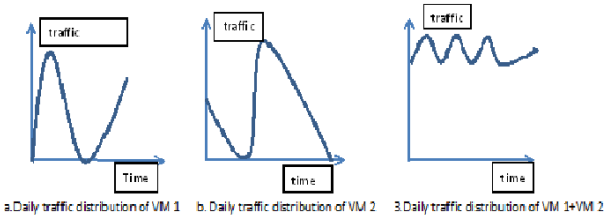

Figure 2.6: Virtual machines traffic distribution

1, the traffic distribution is sometimes high and sometimes low depending on time. For VM 2, traffic distribution is constant for a certain period of time and changes after the period crosses. For VM 1+ VM 2, traffic distribution is fluctuating at certain traffic though the time changes.

2.5.8 Uses of Network Isolation:

Computer resources isolation is a major issue in cloud computing. Isolation of network for the cloud subscribers is considered to be mainly a security issue among this. Since the cloud infrastructure is shared with different organizations, there is a risk of multi-tenancy.

Techniques such as network virtualisation can be used to isolate users from each other. This is possible by the use of VLANs(Virtual Local Area Networks) by which logical network segmentation is done. Traffic is segregated and isolated between different user groups or subnets by the partitioning of VLAN. Network Isolation is possible by this implementation of VLANs. Different VLANS connected to VMs belong to different domains on the network switch. VLANs configuration on the network switch is done in this way to provide network isolation and secure connectivity between VMs same as well as different domains.

2.5.9 Background on Network Isolation:

Network isolation concept has been implemented in various cloud technologies to improve the utilization and flexibility of hardware resources allocation for virtual machines. It has been implemented in public as well as the private clouds. VIOLIN is

a virtual network architecture, which provides network isolation by creating isolated virtual networks[4]. A software-based utility called SoftUDC also provides network isolation which relies on VMM running on every server. It uses VNETs for isolating networks[13]. TVDc is a technology where isolation of networks is provided by associating VLANs with VMs with same label[1]. To implement this isolation, the cloud environment chosen is Orangecloud of Cloud Competency Center at National College of Ireland in this thesis.

2.6 Resource allocation in isolated VMs:

This section covers the resources allocation consideration in isolated virtual machines.

Sharing of resources between the virtual machines will lead to security problem. The virtual machines are controlled by the virtual machine monitor leading to a serious consequence.

IaaS(Infrastructure as a Service) services are delivered using shared resources, which may not be designed to provide strong isolation for multi-tenant architectures. This may affect the overall architecture of Cloud by allowing one tenant to interfere in the other, and hence affecting its normal operation. This type of threat affects IaaS .Physical resources such as memory, disk, network devices of the host machine are shared by the VMs.

IaaS and PaaS providers are provided with the opportunity for sharing of physical resources. Physical infrastructure is utilized well with increasing need to deploy complex, distributed, interactive, real-time applications over the virtualized infrastructure. To avoid decreasing the customer satisfaction, CPU and I/O isolation must be provided between the two classes of workloads.

Concurrently running VMs may interfere with each other that a stable performance level cannot be guaranteed to each one of them without an appropriate support for isolation.

Additionally, sharing infrastructure and administration has clear economic benefits. In most cloud services, these properties are delivered via virtualization, where a fraction of the CPU, disk, and memory resources on a physical machine are dedicated to each application running on it.

Users manage the operating systems and applications running in the VMs while the placement of virtual machines and the operations of the physical infrastructure are

managed by the cloud provider. There is hard isolation and soft isolation provided by VMs as there is no resource sharing or interactions between

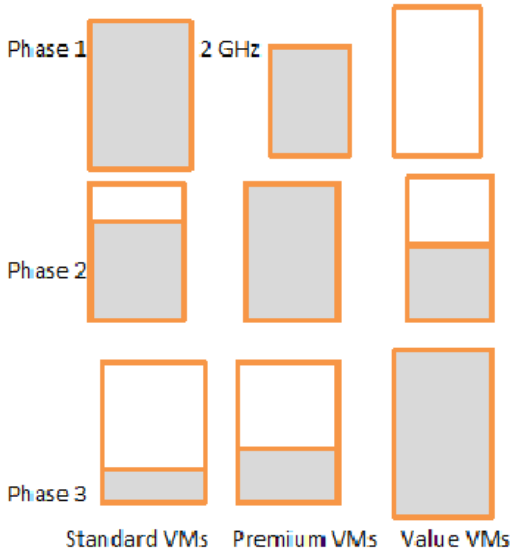

Figure 2.7: CPU utilization among different types of VMs

The CPU utilization depends on three different types of VMs. They are Standard VMs, Premium VMs, Value VMs. They further describe saying that standard VMs are those which provide performance predictability and isolation at the intermediate level. Premium VMs are those that provide performance predictability at highest level.

Finally Value VMs provide less performance guarantees. In the above Fig 2.7 shows CPU utilization and resource demand of the three types of VMs. The CPU utilization is denoted in grey area. The premium VM is underutilizes in the phase 1 demanding only 1.5 GHz CPU. On the other hand CPU utilization for Standard VM is 100 percent of 2 GHz CPU. The value VM has no resources. In the phase 2, premium VM increases the demand to 100 percent and in phase 3, the demand is reduced for standard VM and premium VM is also the same while the value VM increases its demand.

Hard isolation is resource sharing when more than database is co-located on the virtual machine which will cause over-provisioning and commensurate loss of performance. On the other hand, soft isolation is allowing sharing of resource when there is underutilization of resources.

Most modern operating systems essential part are the disk I/O schedulers which improve disk utilization and achieve better application performance. Inter-VM fairness and improving the overall disk throughput in the system has been focused by a recent work on disk I/O scheduling for virtualized environment. Shared disk usage in virtualized environments is considered to impact the virtualized environments as its ability to enforce isolation and fairly utilizing I/O resources of virtual machines share among applications is focused.

Workload	Major resources used	Net I/O(KB/sec)	Response time in (ms)	CPU in %
1 KB	CPU	2018	1.52	97.50
4 KB	CPU	7034	5.46	97.46
10 KB	Network	11045	2.36	70.44
30 KB	Network	11293	2.52	54.87
50 KB	Network	11934	2.7	49.62
70 KB	Network	11255	2.8+	47.10

Figure 2.8: Performance of Workloads and their CPU Utilization

The performance of VMs would be affected when the virtual machines compete for shared resources.

2.7 Virtualization:

This section describes about virtualization. Virtualization has become a widely used technology in todays computing. It has replaced the use of physical machine thereby optimizing hardware utilization. It introduces virtual machines which share the resources of a physical machine. It also adds an abstraction layer called hypervisor (virtual machine monitor) between hardware and software.

Virtualization is also implemented at network-level. Sharing of physical devices and enforcing isolation between the VMs of different users in a cloud environment is enabled by the virtualization software. Computer resources are abstracted when virtualization is introduced. Relating to this, Virtualization is considered to be the key element as it enables optimized hardware utilization aiding isolated virtual machines to share the resources of a physical machine.

2.7.1 Types of Virtualization:

This section concentrates on eliciting the types of virtualizations which can be divided into three types based on the x86 architecture as:

1. Full virtualization: In this type of virtualization, hypervisor runs on top of the host operating system. This implies that VMs, applications and the guest OS run on top of a virtual hardware which is provided by the hypervisor.

2. Para virtualization: In this type of virtualization, the guest OS needs to be modified unlike in full virtualization.

Adding to the above, virtualization can also be divided based on various possibilities. According to the possibilities and needs of the users, virtualization is divided as follows:

1. Process virtualization: This virtualization provides an interface between the applications and the underlying system. Example of this virtualization is Java virtual machine.

2. Server virtualization: Hardware is virtualized in this type of virtualization where multiple OSes can run on a physical machine simultaneously.

3. Storage virtualization: In this type of virtualization, all the physical resources scattered over a network are abstracted to create a logical storage.

There is also another type of virtualization called network virtualization which is discussed in the next section.

2.7.2 Network Virtualization:

Network virtualization is the concept where the virtual private networks (VPN) and VLANs (virtual LANs) are constituted in it. Virtual LANs are the distinct local networks sharing same physical infrastructure. Virtual networks are the basic entities in network virtualization. It comprises of virtual topology which is the collection of virtual nodes collected together by a virtual link. The Internet service providers(ISPs) is divided into two independent entities namely Infrastructure providers(InPs) managing the physical infrastructure and service providers(SPs) which create virtual networks. This is termed as network virtualization. Multiple heterogeneous virtual networks are coexisted on a shared infrastructure provider(InP) network. As a result, flexibility and better manageability are offered by network virtualization.

2.7.3 Background on Network Virtualization:

Network virtualization emerged as a result of experimenting on the networks to provide new capabilities of performance. It was previously implemented for the military communication where there had to be delay/disruption/disconnection tolerant network infrastructure needed for effective communication.

2.8 Performance:

This section discusses the performance benchmarks used to measure various parameters. These benchmark measurements help to recognize the utilization of network, CPU, disk and memory resources that can be allocated for virtual machines.

2.8.1 Network Performance:

Network performance is a major measurement in any virtualization. The impact of virtualization on network performance experienced by users must be measured to characterize the networking performance of virtual machine instances. Network bandwidth and network traffic is monitored and measured by using network performance benchmark.

2.8.2 CPU Performance:

CPU performance measurement measures the speed at which the CPU(processor) operates. For this, various tools can be used.

2.8.3 Memory Performance:

Memory performance refers to the RAM bandwidth measurement and its capacity to hold data.

2.8.4 Disk Performance:

Disk performance relates to the capacity at which the read/write operations are carried out by the disk. Benchmark tools are used to measure the disk speed.

2.9 Conclusion:

This chapter discussed the research background of the dissertation. The next chapter will thus cover the literature review conducted on the tools for performance and related work carried out on isolation of virtual machines.

Chapter 3

Literature Review

3.1 Introduction:

The literature presented in this thesis reviews about the tools used for performance, previous work conducted on network isolation. Performance can be measured when network isolation is provided in virtual machines running on hypervisors. The purpose of this review is to consider isolation of network in virtual machines which can fairly allocate resources.

The main body of this work is split into sections. Isolation and the question of considering it are covered in section 3.2 and 3.2.1 respectively. Providing isolation between virtual machines, related issues and benefits of isolation are elicited in section 3.3 and section 3.4. Section 3.5 and section 3.5.1 gives revision of hypervisors and available hypervisors. Tools used for measuring performance parameters such as network, disk, memory and CPU are reviewed in section 3.6.1, section 3.6.2, section 3.6.3, section 3.6.4 respectively. The chapter is concluded in section 3.7. The next chapter focuses on presenting hypervisors and tools chosen to elaborate design sections in the thesis.

3.2 Isolation:

This section provides a description about the basics of isolation. Isolation is referred to as the process of separating either virtual machines or the programs running in it from each other. When multiple virtual machines situated parallel to each other are to be operated on the same physical server, they must be isolated[8].

3.2.1 Why Network Isolation?

This section deals with the isolation of virtual machines in terms of network and performance.

Network isolation is considered for the virtual machines and then the performance for them is estimated to enable the VMs to have greater performance in terms of network bandwidth and latency. To achieve performance isolation between internet applications on virtualized servers is important but challenging say [16].

The authors also continue saying that performance isolation should be aligned with the incentive to maximize the overall system utility, which includes the service-level utility of customer applications and the utility of server energy consumption [16]. He continues adding that workload intensity may also be the reason for the energy consumption characteristics to be dependent on it. A self-adaptive approach is needed to ensure the robustness of the performance isolation against the application heterogeneity and dynamic workload variations that will respond to the performance changes.

A machine learning based online adaption of the performance interference and energy models of the system is considered which will achieve the performance isolation [16].

[27] presents a benchmark for performance isolation that quantifies the degree to which a virtualization system limits the impact of a misbehaving virtual machine on other well-behaving VMs which are running on the same physical machine. They further continue saying that an access control mechanism is presented and the added performance cost is acceptable which will enforce isolation and limits sharing between VMs. [27]adds that performance isolation is provided in virtual machine monitor. By performance isolation between the virtual machines will avoid the performance changes of one virtual machine not to affect or influence the performance of other virtual machine. Many different kinds of benchmarks are designed to assess the performance isolation between virtual machines such as memory-intensive, CPU-intensive, disk-intensive and network-intensive benchmarks [27].

[28] say that if workload of one VM increases fast in a host, other VMs performance must not be influenced to ensure service providers wont breach SLA. A cloud provider should be able to provision network resources for its tenants starting with bandwidth as bit per second to enable customer to set clear expectations of network.

[12]say that distributed services which have a demanding network component unlike CPU intensive jobs require network isolation which is vital. [21] says that the applications' performance, which are running in one VM would be affected by the neighbour VM's applications. This happens when the network I/O workloads are at high rates in

the virtual machines.

[14] indicates that isolation is important for virtual system in the network. The author also signifies the importance of implementing isolation in the cloud. Isolation helps providing better sharing of network resources. Hence it is interpreted that isolation also tries to reduce the network traffic for increasing functionality to access the virtual resources.

3.3 Providing Isolation between virtual machines and related issues:

This section covers security aspect of considering isolation of VMs as it is of key importance for ensuring security in VMs.

By isolating virtual machines the programs that are running in one VM cannot be seen by the program running in another VM. Virtualisation separates one VM (virtual machine) from another VM running on the same physical hardware. Therefore we can understand that virtualization provides isolation of VMs [24].

Due to the rapid development of cloud technology, providing isolation between virtual machine on the same physical platform has been an active area of research in the past decade[29]. The authors also indicate that there are generally three approaches for providing isolation of virtual machines in the cloud environment. They are: Hypervisor based approach wherein the hypervisor is minimized or hardened, hardware assisted isolated execution wherein the hardware functions are used by systems to maintain the isolation and thirdly direct hardware access is used to provide the necessary isolation[29]. By this we can understand that isolation of virtual machines is the important concern in terms of security.

However, another important component of VM isolation that is to ensure that the applications running in one VM do not have access to the applications which are running in another VM. This means that isolation should be carefully configured and maintained; otherwise it will prove to be a threat. If isolation is carefully configured and maintained, it can avoid the attack on one VM which would prevent access to either the VMs in the same environment or to the underlying host machine[24].

[27]in their article emphasize that consideration of virtual machine isolation is the most basic security requirement in a virtualized environment. They say that significant security risks will be prevalent if isolation is not ensured between the virtual machines.

One of the main goals of hypervisors is to ensure that isolation occurs in the host machine. By this isolation, guest systems will not access more resources than it has been granted in terms of memory usage, both by the guests or the host systems suggest[26]. They continue explaining that breach in the isolation will lead to the following attacks:

Denial of service, is an attack where all the computing capacities of the host are utilized by one VM preventing the other VMs from running correctly. Distributed denial of service attack is one such kind of attack. In the host system, there is a chance of guest machines (virtual machines) to impose denial of service attack to the other guest machines residing in the same system. This attack is described as an attack when all the resources are utilized by the guest machine or virtual machine preventing other VMs to utilize the resources. Therefore services are denied to the other virtual machines that are requesting for resources.

System Halt is a specifically crafted instruction. It causes the VM or the hypervisor to crash.

VM escape is one of the attacks where the attacker gains access to other system. It will help the attacker to read, write, or execute the contents of memory of the host or guest systems. Isolation layer is broke by the attacker's program running in one VM allowing the attacker to interact directly with the hypervisor[20].

[26]further discusses isolation with several examples of hypervisors like KVM for linux-based virtualisation solutions which use Mandatory Access Control (MAC) in sVirt a virtualisation which is used by RedHat. They explain that assigning a distinct label to each VM and its processes will allow that VM to read and modify only its corresponding memory contents and prevents any access of other VMs from accessing each other. Therefore, this will not allow VMs to access the one which has been compromised.

In spite of this much analysis indicating that isolation is necessary to avoid illegal access and in order to support applications designed for one operating system to be operated on another operating system [24] ,some virtualization avoids isolation which will completely exploit the security bearers in both the operating systems. The author adds that the system gives virtual machines an unlimited access to the hosts resources (like file system and networking devices) where there is no isolation between the host and the VMs. The file system becomes vulnerable in this case.

Though many articles from different authors specified on providing enough isolation, there is considerable amount of breach in the isolation due to bad configuration or design flaws within the hypervisor. This conclusion is drawn after referring to the above review.

3.4 Benefits of VM Isolation:

This section deals with the benefits associated with the isolation of virtual machines.

[20] argue that firewall or any antivirus software just can't be installed on a cloud-based virtual machine. They also argue that the vast amount of traffic originating from a hypervisor running 10 virtualized servers cannot be inspected and filtered by the physical firewalls. They continue saying that at the click of the button, VMs start, stop, and move from hypervisor to hypervisor with ease whatever may be the protection chosen to handle these activities. Also VMs number increases in the data center and this will become harder to manage and protect them if unauthorized people gain access to the hypervisor, lack of control may be the advantage for them to modify all the VMs housed there.

[20] further adds that like their physical counterparts, even virtual machines are vulnerable. Virtual machines must be isolated from other network segments to adequately protect them and to prevent them from both internal and external threats deep inspection at the network level should be implemented. By using secure remote access technologies like IPSec or SSL VPN, unauthorized external access should be protected. Also the attacker may be disabled from injecting malicious code in the neighbours VM by providing strong isolation between VMs. This is in the case of service injection type of attack[20].

[17] say that running sensitive applications in their own virtual machines is one of the frequently proposed approaches to protect those applications. This is a benefit of isolation as the sensitive applications which are prone to attacks will be protected. He further says that multiple virtual machines management puts an administrative burden on users as the users will be in confusion to use which VM for specific task and installation task is also multiplied and if any VM is compromised, it must be rolled back. By isolating VMs both usability and security can be improved.

[29] in their article proposed a framework for isolating and protecting individual virtual machine at hardware-level. With this approach, confidentiality and integrity of VMs in a multicore environment can be maintained by the user even in the presence of malicious attacks from both within and outside the cloud infrastructure. The authors explain that a particular RAM is used for the isolation between the host and VM they say is readily available in current generation of processors.

Run-time confidentiality and integrity can be provided by isolating individual VMs from legacy host and neighbouring VM. By separating processor memory and I/O devices this type of isolation can be achieved. The memory content and control flows can be

protected with such isolation[29]. They also continue that isolation can be provided at the processor level wherein each guest VM is dedicated individual cores.

Therefore different users own their separate processors. The user VM memory is isolated from the malicious legacy host by a SMRAM protection. Extended page table is used to isolate memory from neighbouring VM which requires privilege instructions to change and does not allow the memory content of neighbouring VM to be tampered.

Hardware provides Isolation of I/O devices like hardware devices. This will make the virtualized hardware appear as the real system to the software system. Individual VMs are assigned different physical devices or hardware virtualized devices which achieves isolation of I/O devices.

3.5 Hypervisors:

This section concentrates on giving description about hypervisors. There are many researches being conducted on hypervisors and its development is making considerable contribution to cloud computing.

[24] gives a clear and comprehendible definition about hypervisor. The author demonstrates that hypervisor is an abstraction layer that separates the hardware, operating system and the applications running on top of it. It is also called as VMM(virtual machine monitor). He adds that the physical resources of the computing system are hidden from the operating system. Since the VMM controls hardware resources, multiple OSs can be run in parallel on the same hardware. The refers to the hardware platform partitions as virtual machines(VMs).

In conjunction to the above statement [27] notifies that virtual machines are completely controlled by the VMM. [5] also agrees adding that VMM is an abstraction layer making calls to the real machine from the virtual machine. The author also signifies that different VM operating systems and configurations can be allowed by hypervisor.

Records of all loads are kept by hypervisors like memory usage, storage usage of each VM, through its dedicated software components. It also manages software management like CPU,memory, I/O device shared by VMs [8].

However, to continue with providing isolation, hypervisor plays a vital role. Since hypervisor controls the entire VMs hypervisors providing isolation also depends on the hypervisor used.

3.5.1 Hypervisors Available:

Hypervisors are classified into two types as emphasized by [26]. Type 1 and Type 2 are the two types of hypervisors. Hypervisors installed directly over hardware are type 1 and those installed on top of host operating system are type 2 hypervisors. The authors[26] refers to these types giving example for type 1 as VMware ESX, Xen and type 2 as VirtualBox, QEMU.

In addition to the above hypervisor types [8] implies that there is another type called type 3 referred to as Hybrid VM. The combination of type 1 and type 2 is Type 3 hypervisor. In this type, server OS is utilized for its process execution and resource management while another part of the server is utilized for hardware resource allocation of VMM.

There are many hypervisors available which provide isolation like HyperV developed by Microsoft corporation, VMware ESXi by VMware, Inc, Xen developed by University of Cambridge, KVM(Kernel Virtual Machine)by Qumranet, OpenVZ developed by Parallel Inc.

3.6 Tools for performance measurements:

This section provides information about the tools or techniques that are used for measuring the performance of various benchmark parameters like CPU, memory, Disk, Network.

Benchmarking tools help measure the performance characteristics. Testing performance of hardware features is gaining importance which makes the evaluation of performance measurements in the thesis understandable. [3] gives clear implication that a particular hardware feature of a computer can be assessed by using these benchmarking tools. The authors imply that performance comparison can be determined with the help of these benchmark results. They continue further adding that performance results can be similar when performance characteristics of two machines are similar.

3.6.1 Network Performance:

This section lists the various benchmarking tools available for network performance.

S No.	Tool	Software/OS	Version	Size
1	Netperf	Windows,Linux	2.6.0	6.14 MB
2	Uperf	Windows,Linux	1.0.4	202.8 KB
3	Iperf	Windows all	2.0.5-2	1.2 MB
4	Netmeter	Windows all	1.1.3 /1.1.41 Beta	600 KB
5	PRTG	Windows 7,2008 R2/8/server 2012	13.2.3.2284	117 MB
6	LAN speed test	Windows 7,XP,server 2003,2008	V3.0	182 KB
7	Sockperf	Windows all	2.5.215	85.9 KB
8	LANBench	windows all	1.1.0	1.9 MB

Table 3.1: Available Tools for Network Performance

Sources:
Netperf-www.netperf.org
Uperf-www.uperf.org
Iperf-staff.science.uva.nl/~jblom/gigaport/tools/test-tools.html
Netmeter-www.smallnetbuilder.com/lanwan/lanwan-howto/
30366-measuring-network-performance-netmeter
PRTG-www.paessler.com/prtg
LAN speed test-www.totusoft.com/lanspeed.html
Sockperf-www.findbestopensource.com/product/sockperf
LANBench-www.zachsaw.com/?pg=lanbenchtcp-network-benchmark

The above table demonstrates various benchmarking tools available for network performance. Iperf tool is used for measuring TCP and UDP bandwidth performance [11]. This statement is substantiated by [19] who adds indicating that Iperf tool measures bandwidth of network link. Uperf is another tool which measures TCP communication [23]. There is another tool for network monitoring which is a commercial network analyser tool called PRTG [6].

3.6.2 Disk Performance:

This section lists the various benchmarking tools available for Disk performance.

S No.	Tool	Software/OS	Version	Size
1	Crystal DiskMark	Windows 2000/XP/2003/Vista	3.0.2f	1.6 MB
2	DiskBench	Windows all	2.6.2.0	154 KB
3	IOzone Filesystem benchmark	Windows 2000/XP/Vista	3.405	3 MB
4	Datamarck	Windows Xp/Vista	0.0.3	570.13 KB
5	ATTO benchmark	Windows all	2.47	236 KB

Table 3.2: Tools Available for Disk Performance

Sources:
Crystal Disk mark-www.majorgeeks.com/mg/sortname/benchmarking.html
DiskBench-www.majorgeeks.com/mg/sortname/benchmarking.html
Datamarck-www.download32.com/datamarck-i158870.html
ATTO benchmark-www.majorgeeks.com/mg/sortname/benchmarking.html
IOzone Filesystem benchmark-www.iozone.org

IOzone tool is used to measure the performance of NFSv4.0 and NFSv4.1 which are file system components [15]. Disk intensive tests are conducted by IOzone agrees [25]

3.6.3 Memory Performance:

This section lists the various benchmarking tools available for Memory performance.

S No.	Tool	Software/OS	Version	Size
1	BenchMem	Windows all	0.1.5	12 KB
2	Memtest86	Windows all	V5.0 Beta	2.12 MB
3	Memtest	Windows all	4.0	13 KB
4	Memtach	Windows all	0.93 Alpha	246 KB

Table 3.3: Memory Performance Tools

Sources:
Benchmem-www.majorgeeks.com/mg/sortname/benchmarking.html
Memtach-www.majorgeeks.com/mg/sortname/benchmarking.html
Memtest86-www.memtest.org
Memtest-www.benchmarkhq.ru/english.html?/

LMbench provides memory benchmark by comparing performance of different VMs [11]. [7] uses Memtest to measure the performance of AC clusters. Memtest86 is another tool for memory testing signifies [22].

3.6.4 CPU Performance:

This section lists the various benchmarking tools available for CPU performance.

S No.	Tool	Software/O	Version	Size
1	Passmark	Windows all	V8.0	23 MB
2	SPECvirt-sc2013	Windows all	1.00	401 KB
3	Former CPU mark	Windows all	2.2	149 KB
4	OCB(open CPU benchmark)	Windows all	0.1.01.0714	67 KB
5	CPU-M benchmark	Windows XP/2003/Vista/2008/7	1.4.0.0	673 KB
6	Novabench	Windows all	3.04	11.7 MB
7	Linpack	windows all,linux	11.1.0.002	7 MB

Table 3.4: List of Available CPU Performance Tools

Sources:
SPECvirt-sc2013-www.spec.org/virt-sc2013/
Former CPU mark- www.softpedia.com/get/System/Benchmarks/CPUMark.shtml
OCB(open CPU benchmark)-sourceforge.net/projects/opencpubench/
CPU-M benchmark-www.majorgeeks.com/mg/sortname/benchmarking.html
Novabench-www.majorgeeks.com/mg/sortname/benchmarking.html
Passmark-www.passmark.com
Linpack-www.software.intel.com

[9] emphasizes Passmark benchmark is used for CPU test.[18] notifies PCmark is the tool for CPU performance.[2]Dacapo is the becnhmarking tool to test CPU. To benchmark the CPU performance of supercomputers a tool called Linpack is used. Fortran source code is used to take metrics[10].

3.7 Conclusion:

In this literature review, the tools which can be used for performance benchmarks are specified. There are various tools that can be used for performance measurements. The next chapter will discuss about tools and hypervisors specifically chosen for the thesis.

Chapter 4

Design

4.1 Introduction:

This chapter emphasizes on demonstrating the design features that are chosen for the thesis. The main body of this chapter is composed of sections. Section 4.2 gives system description. VLAN, HyperV, VMware, Pseudo IP addresses for virtual machines are distinguished in section 4.3, section 4.4, section 4.5, and section 4.6 respectively. Hardware and software requirements are enumerated in section 4.7.1 and 4.7.2. Tools picked for network, disk, cpu, memory performances in thesis are listed in section 4.8, section 4.9, section 4.10, section 4.11. Finally, section 4.12 draws a conclusion.

4.2 System Description:

The hypervisor is present, which is a layer of software for controlling the virtual machines. It is necessary to isolate the network of the virtual machines as the user data resides on the cloud infrastructure and maintaining the confidentiality and integrity of user data is important. Also data traffic is generated by the VMs. One of the major security concerns is to isolate the networks of individual cloud subscribers.

The best approach to prevent the virtual machine from consuming all the resources is to limit the resources allocated to the guests. One possible method to limit the resources is to isolate the resources like network, memory, CPU. Isolating network interfaces is considered here. Also the performance of the virtual machines is compared for VMware and HyperV virtualization technologies. For this, the resources like CPU utilization of virtual machines and are considered to show performance.

Hypervisors used in this thesis are HyperV and VMware ESXi upon which guest OSs are run.Orangecloud is the cloud environment used for this experiment. System Center 2012 Virtual Machine Manager(SCVMM) is the management portal for creating, managing and operating the VMs. HyperV is the hypervisor on which SCVMM runs.VMware vsphere client is the management portal on orange cloud which is used to create, manage and control Guest VMs on it. ESXi is the hypervisor on which vsphere client runs.

4.3 VLAN:

Various network resources include VLAN (virtual local area network) which provides isolation to a group of virtual machines that reside on the same host machine.

VLAN is used for traffic isolation and segregation between the different user groups or subnets. Moreover VLAN IDs supported by the switches are limited to maximum of 4,094. Instead of using two or more VMs per VLAN, one VLAN can be used for one VM.

In the Orange cloud environment, when the logical network is created in the System Center 2012 VMM, VLANs are assigned to each virtual machine. For example, the VLAN Id for logical network(prod team network) in the host server of SCVMM is 320. In this thesis, Prod team network is the logical network to which the VLAN401 pool is attached in HyperV. Like this, three VLAN namely, VLAN401, VLAN402, VLAN403 are created and thus a guest VMs is attached to each VLAN. In VMware, vSwitch is the standard virtual switch to which VLANs created are attached. Each guest VM is thus linked to one VLAN.

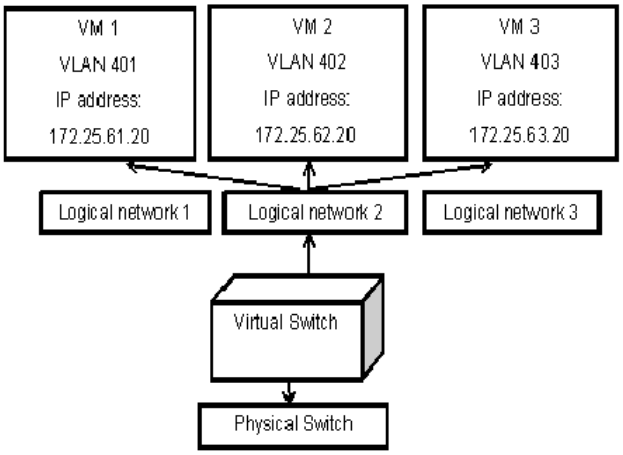

Figure 4.1: VLAN Architecture

4.4 HyperV:

This section provides description about Microsoft's HyperV hypervisor used in the implementation of isolating VMs and thereby comparing the performances considering hardware features.

One of the drawbacks in HyperV is that multiple logical subnets cannot be spanned. This limits the number of nodes within a VLAN. Other limitations for using VLAN are cloud-based environments require readdressing the virtual machines. Also IP address of the VM was responsible for maintaining security management and policies of VM. There are certain limitations on the memory and processors for HyperV. Virtual processors are limited to four virtual processors and memory is up to 64 GB of memory.

In the earlier, when only one NIC was needed to create an external virtual network on HyperV. The network performance was not affected much as only few VMs were created. But now as number of VMs increase, more bandwidth and better network performance for HyperV role is important. It is recommended to have multiple NICs available for HyperV server role as there is plenty of network traffic. HyperV also provides network traffic monitoring. An internal switch is available which allows the access to other internally hosted virtual machines by the VMs.

By virtual switch, network isolation is provided to a greater level. Misconfigurations can allow the breach in the security and attacker comes into action.

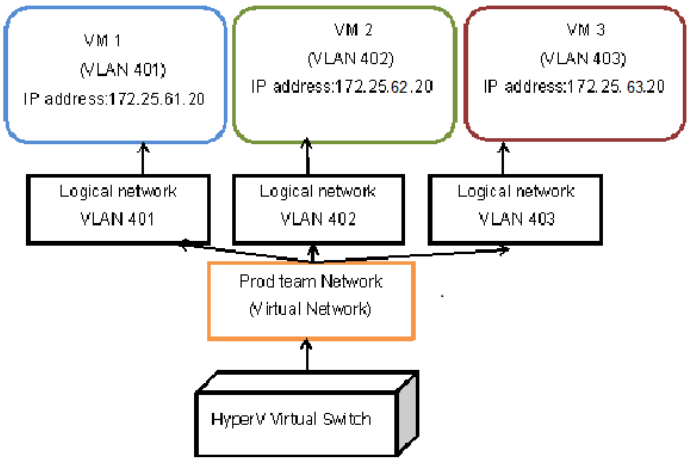

Figure 4.2: HyperV VLAN Architecture

4.5 VMware ESXi:

This section describes VMware's ESXi hypervisor which is used for the implementation of isolating virtual machines by enabling separate VLANs.

Over the years, VMware's hypervisors evolved rapidly. It has emerged from its basic release ESX classic to ESXi. Some additional features are built into the VMkernel of ESXi hypervisor. It has much smaller management console. VMKernel, VMkernel extensions, VMM are the components of VMware.

vSphere Distributed Switch helps abstracting the physical network features. VLAN is assigned by the virtual machine port group of the vSwitch.

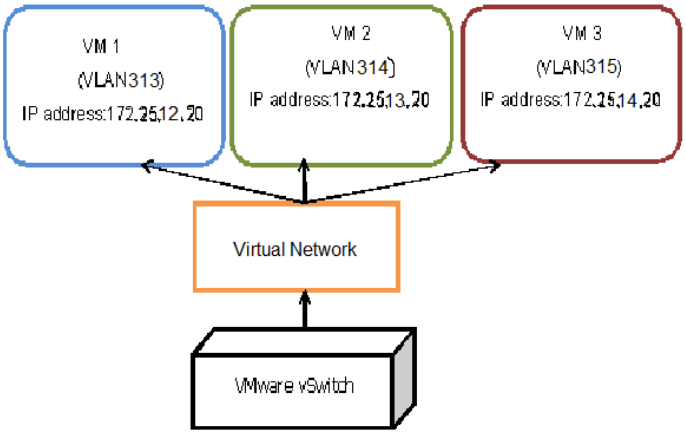

Figure 4.3: VMware VLAN Architecture

4.6 Pseudo IP addresses for virtual machines:

Here the randomly-allocated IP address is provided to the VM when VMs are communicating with each other. The actual IP address allocations are not changed.

Virtualization Technology	Hosting machine for virtual machines(guest machines)	Virtualization Environment	Performance characteristics	Network isolation mechanism
HyperV	Same host machine	Orange cloud	CPU utilization, network bandwidth,memory	Virtual switch
VMware	Same host	Orange cloud	CPU utilization,network bandwidth,memory	vCloud

Figure 4.4: Virtualization Charactersitics Considerations

In the above diagram, the virtualization technologies VMware and HyperV are taken to show the performance in both by comparing them. The virtual machines are present in the same hosting machine as there is minimum chance of vulnerability to distinct host. OrangeCloud is used as a virtualization environment

Various performance metrics like the CPU utilization, network bandwidth and memory are used in both the cases to compare their individual performances. HyperV uses a network isolation mechanism, virtual switch. VMware uses vSwitch and Distributed

vSwitch which are part of VMware for network isolation mechanism. vCloud director, which is a technological solution by VMware to build private cloud, controls both these switches.

4.7 Hardware and Software Configurations:

Both HyperV and VMware ESXi have identical hardware configurations. The software configurations are similar in both the hypervisors.

4.7.1 Hardware Configurations:

This section gives the list of hardware components.

CPU speed	2.50GHz
Processor type	Intel(R) Xeon(R) CPU E5-26400
Host Disk space	2 TB
Guest Disk space	40 GB
Host Memory usage	24530.48 MB
Guest Memory usage	4 GB

Table 4.1: Hardware Configurations

4.7.2 Software Configurations:

This section gives the list of Software components.

Operating System	Windows server 2012/Windows server 2008 R2
Hypervisor	HyperV, VMware ESXi
Network performance Tool	Iperf
CPU performance Tool	Passmark
Memory performance Tool	Memtest
Disk performance Tool	IOzone Filesystem Benchmark

Table 4.2: Software Configurations

4.8 Tool for Network performance:

This section gives overview of the tool used for comparing the network performance. Iperf benchmarking tool is used for this purpose. It is used for measuring TCP and UDP bandwidth performance. This tool tries to measure the network connections of virtual machines. VMs that send data packets and VMs that receive data packets are on the same physical machine. Iperf tool measures this type of network connections also. Written in C++, this tool creates TCP and UDP data streams to measure the throughput of a network that carries them. It can measure throughput between two ends that is either unidirectional or bi-directional. Iperf tool can run on Linux, Unix and Windows platforms. This particular tool is implemented in the thesis as it evaluates the bandwidth over test period of time, size of data transferred.

4.9 Tool for Disk performance :

This section describes the tool used for measuring Disk performance. IOzone Filesystem Benchmark is the tool used for this purpose. IOzone measures reading and writing performances of a file. IOzone is chosen as it gives broader filesystem performance which means that it is portable to any machine. This makes tool to work on any operating system platforms like Windows, Linux etc. The I/O operations are taken as basic workloads for measuring performance of filesystems. Read, write, re-read, re-write are some of the file operations. Written in ANSII C, it can measure the performance of both single and multiple streams.

4.10 Tool for CPU performance:

This section explains Passmark benchmark tool which is used in the thesis to measure CPU performance. Passmark gives better performance when used in VMware. Passmark is a reliable tool as VMware ESXi is one of the hypervisors used for implementation. It does various operations on system like CPU speed tests, Floating point operations etc. It compares speeds of different processors like AMD opteron, Intel Core2 Quad, Intel Core i7 and mobile CPUs.

4.11 Tool for Memory performance:

This section provides details about benchmarking tool used for memory performance. Memtest is the tool to test RAM. Memtest is advantageous to use as it has flexibility of using memory at low level. This tool is designed for x-86 architectures. It does read-write operations on memory to check for any errors.

4.12 Conclusion:

This section described about the hypervisors used for implementing isolation in VMs. The implementation of performance tools that are used in this thesis will be covered in next chapter.

Chapter 5

Implementation

5.1 Introduction:

This section encompasses the implementation details. The structure of this chapter is divided into sections that will cover implementation of tools on HyperV and ESXi hypervisors. The tools are run for performance tests on virtual machines residing on both of them. Both Section 5.1 and section 5.5 describe about logical networks setup on HyperV and ESXi. Installation of tools is covered in section 5.3. Implementation of tools in virtual machines located on HyperV is explained in section 5.4. Section 5.6 describes tools implemented in VMs residing on ESXi. Lastly, conclusion is covered in section 5.7. The following section will discuss networks created in HyperV.

5.2 Logical networks in HyperV:

Logical networks are created in hyperV. HyperV virtual switch is used for this purpose. The logical networks lie on this virtual switch. VLAN 401, VLAN 402, VLAN 403 are the three logical networks created. They have IP addresses assigned to them from the IP pool. These logical networks are connected to the PROD TEAM virtual network on orangecloudenci environment. Virtual machines vish2, vish1, vish 3 connected to VLAN 401, VLAN 402, VLAN 403 VLANs respectively.

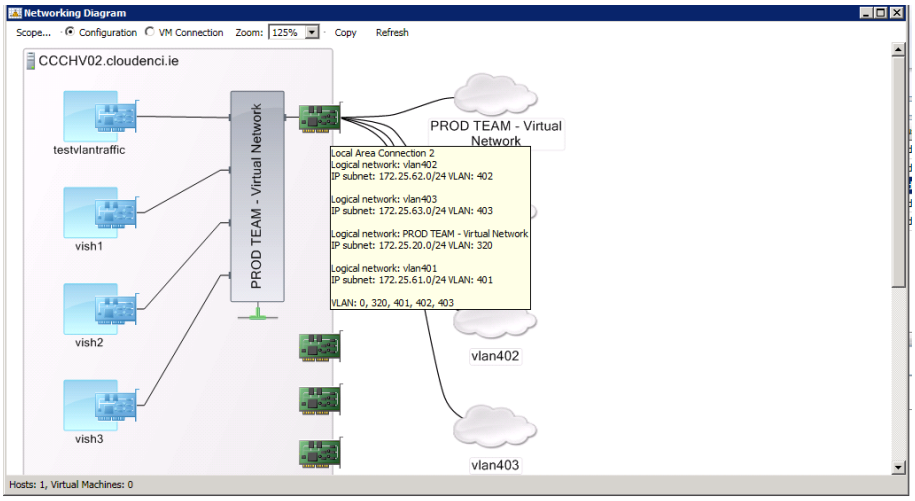

Figure 5.1: Networking Architecture

Above figure demonstrates networking architecture of logical networks.

Each logical network has IP address pool out of which one IP address is assigned to every VM. VLAN 401 logical network has 230 available addresses. Staring address is 172.25.61.20 and ending address is 172.25.61.250. Logical network VLAN 402 has 230 available addresses. 172.25.62.20 is the starting address and ending address is 172.25.62.250. VLAN 403 logical network also has 230 available addresses. Starting address is 172.25.63.20 and 172.25.63.250 is the ending address.

Below figure illustrates above information in detailed form.

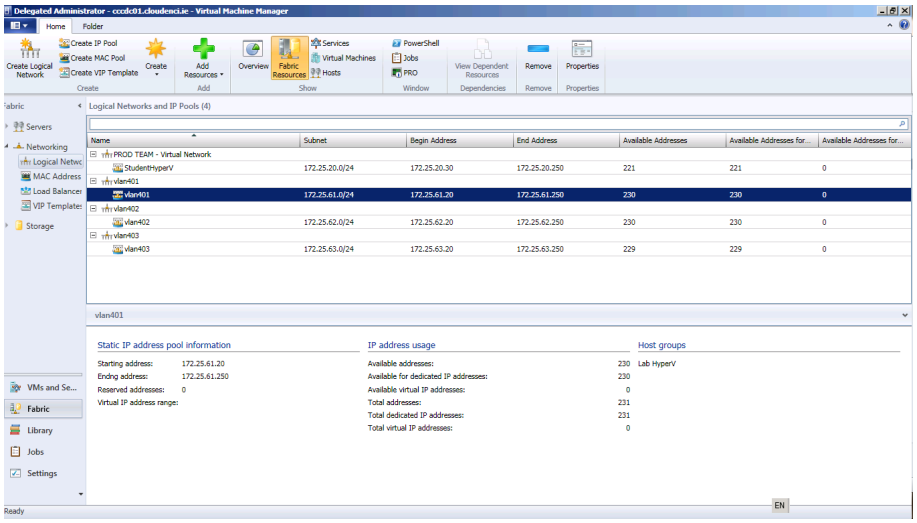

Figure 5.2: Logical networks

5.3 Installation of Tools:

5.3.1 Iperf Tool Installation:

Installation steps:

1. Iperf tool with version 2.0.5-2 is installed into windows server 2012 guest operating system.
2. Basically, the iperf 2.0.5-2 zip file is extracted into desired location.
3. It is then executed from command line changing the directory location to the specific folder where iperf files are present.

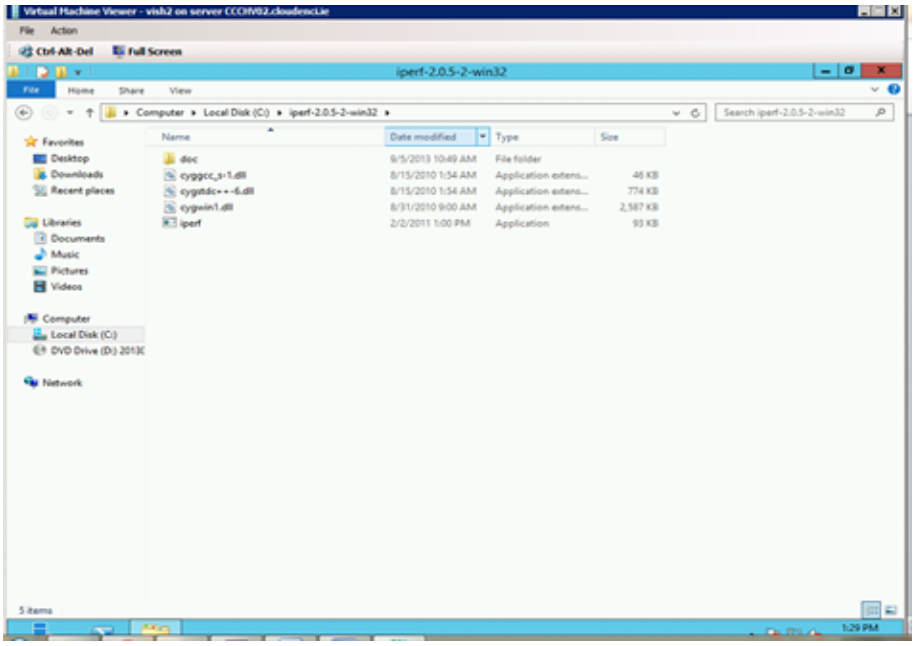

Figure 5.3: Iperf Installation

5.3.2 Passmark Tool Installation:

Installation steps:

1. Passmark toolwith version V8.0 is also installed in three guest operating systems.
2. It is installed into programs files folder.

3. The executable file is installed into the Performance test folder inside program files.

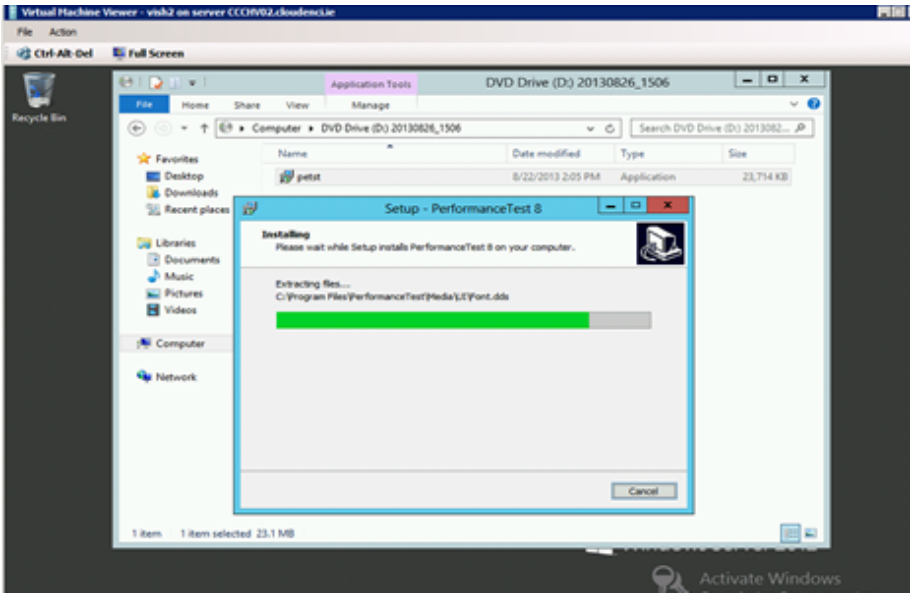

Figure 5.4: Passmark Installation

5.3.3 IOzone Tool Installation:

Installation steps:

1. IOzone tool with version 3.405 is installed from the setup file into desired location on C drive.
2. The executable file is installed into iozone3.405 folder.
3. iozone3.405 folder in present in Benchmarks folder.
4. Benchmarks folder is located in Program files(x86) folder.
5. It is executed from the command line locating the specific folders in which iozone executable file is present.

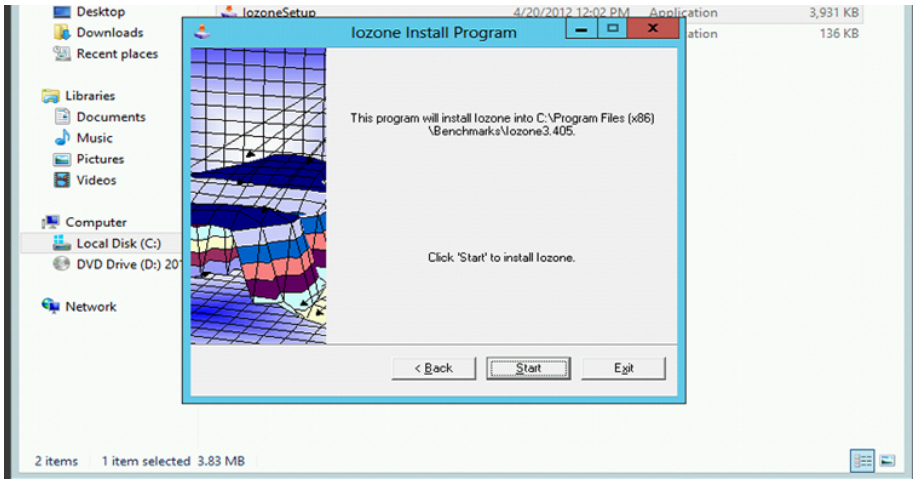

Figure 5.5: IOzone Installation

5.3.4 Memtest Tool Installation:

Installation steps:

1. Memtest tool with version 4.0 into the desired location.
2. The executable file is directly installed when it's ISO image is extracted.
3. Memtest folder contains the .exe file for Memtest.

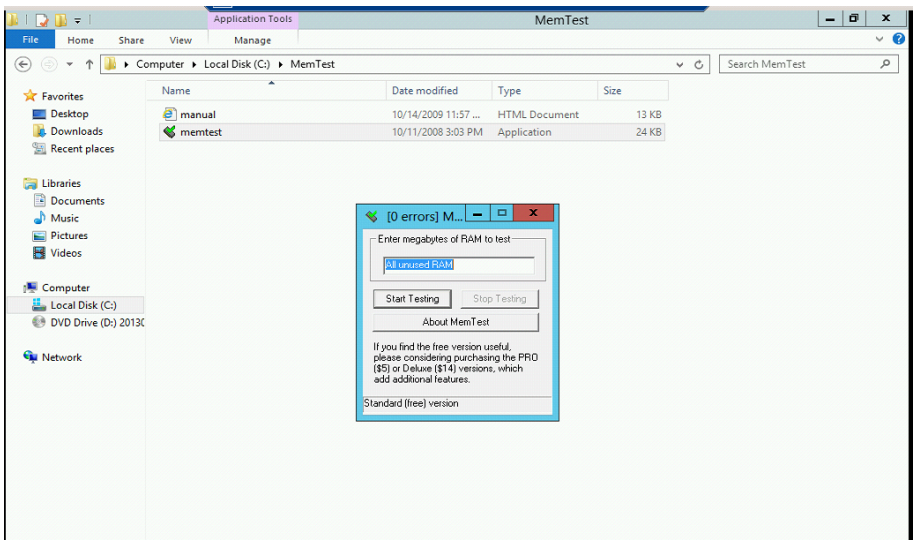

Figure 5.6: Memtest Installation

5.4 Benchmark readings for tools:

5.4.1 Benchmark readings for Iperf:

This tool is executed from command line.The commands are executed on two command prompts. Thus, the commands are "iperf -s" for server side connection and "iperf -c [IP address]" for client side connection.

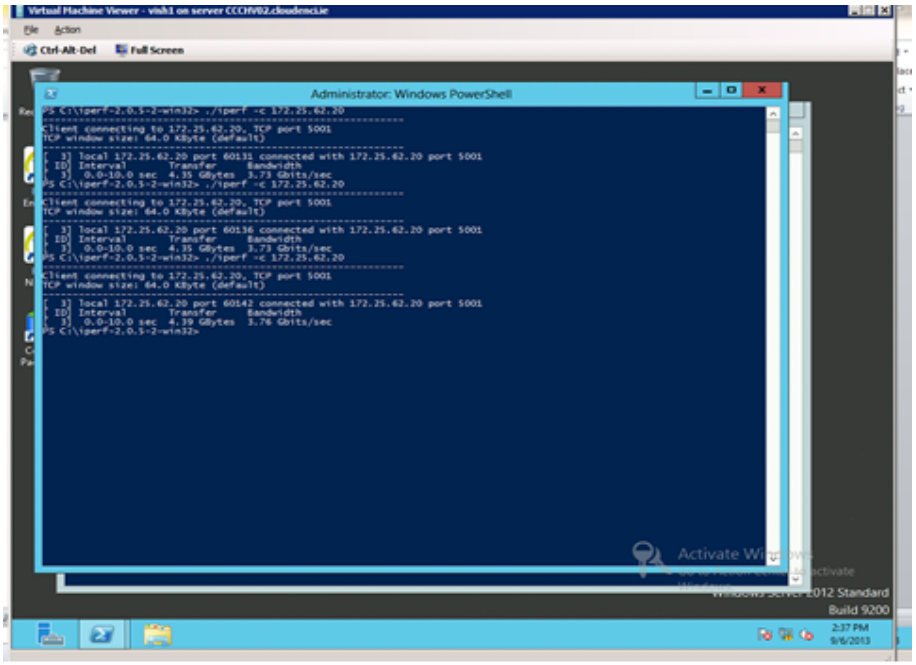

Figure 5.7: Iperf test

S No.	Readings	vish2	vish1	vish3
1	Reading 1	3.06 Gbits/sec	3.73 Gbits/sec	3.45 Gbits/sec
2	Reading 2	3.04 Gbits/sec	3.73 Gbits/sec	3.50 Gbits/sec
3	Reading 3	3.10 Gbits/sec	3.76 Gbits/sec	3.55 Gbits/sec
4	Reading 4	3.10 Gbits/sec	3.64 Gbits/sec	3.40 Gbits/sec
5	Reading 5	3.04 Gbits/sec	3.60 Gbits/sec	3.45 Gbits/sec

Table 5.1: Different Bandwidth Readings by Iperf in VMs

Above table gives bandwidth of network in each of the virtual machines. Iperf also gives the amount of data transferred at particular intervals of time. For example, if we consider a particular virtual machine vish2, it will yield results to data transferred, bandwidth rate and time intervals for under iperf tool execution leading to the following:

Parameters	vish2(Virtual Machine)
Bytes transferred	4.74 GBytes
Bandwidth rate	4.06 GBits/sec
Time intervals	10 sec

Table 5.2: Iperf Benchmark Readings for vish2

5.4.2 Benchmark readings for Passmark:

This tool is executed from the .exe file. It tests various operations such as Integer math, CPU Floating point math, prime number,extended instruction, compression, encryption, physics,sorting, single threaded operations.

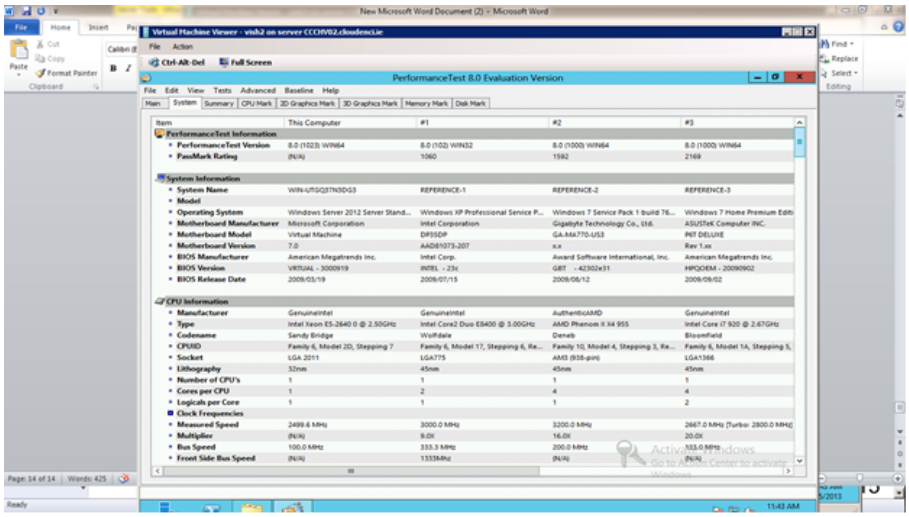

Figure 5.8: Passmark test

S No.	Readings	vish2	vish1	vish3
1	Reading 1	1402 Strings/sec	1436 Strings/sec	1455 Strings/sec
2	Reading 2	1409 Strings/sec	1429 Strings/sec	1439 Strings/sec
3	Reading 3	1408 Strings/sec	1418 Strings/sec	1441 Strings/sec
4	Reading 4	1403 Strings/sec	1415 Strings/sec	1450 Strings/sec
5	Reading 5	1408 Strings/sec	1426 Strings/sec	1456 Strings/sec

Table 5.3: Different Single Threaded Operations Readings by Passmark in VMs

The table above depicts single threaded operations conducted in each VM. The table below demonstrates various operations carried on vish2.

Operations	vish2(Virtual machine)
Integer math	1610 million operations/sec
Floating point math	1337 million operations/sec
Prime number	7.6 million primes/sec
Extended instruction	4.66 million matrices/sec
Compression	1349 KBytes/sec
Encryption	175.3 MBytes/sec
Physics	103.4 Frames/sec
Sorting	1077 Thousand Strings/sec
Single threaded	1369 Thousand Strings/sec

Table 5.4: Various Passmark operations readings for vish2

5.4.3 Benchmark readings for IOzone:

IOzone tool is executed from the command line. The commands for executing it are many such as "iozone -a" for all test operations in automatic mode."iozone -R" for generating an excel report for read and write operations. To produce read and write results in operations per second "iozone -O" command is used."iozone -N" command is used for producing results in microseconds per operation.

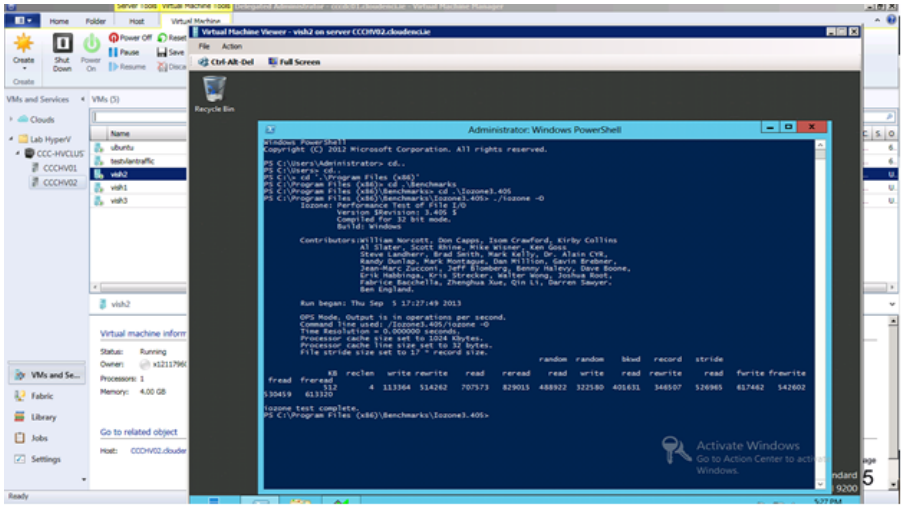

Figure 5.9: IOzone test

below table depicts the readings obtained from IOzone tool,

S No.	Readings	vish2	vish1	vish3
1	Reading 1	313648 Operations/sec	308880 Operations/sec	318645 Operations/sec
2	Reading 2	310303 Operations/sec	305562 Operations/sec	316127 Operations/sec
3	Reading 3	310981 Operations/sec	303102 Operations/sec	316674 Operations/sec
4	Reading 4	310441 Operations/sec	306762 Operations/sec	317645 Operations/sec
5	Reading 5	310556 Operations/sec	304012 Operations/sec	318462 Operations/sec

Table 5.5: Different Read Operations Readings by IOzone in VMs

5.4.4 Benchmark readings for Memtest:

Memtest tool tests the RAM performance. It gives better performance when unused RAM space is tested for any errors. The amount of RAM space to be tested is given and it will produce the percentage of RAM covered. The more it is run, the better RAM performance is generated.

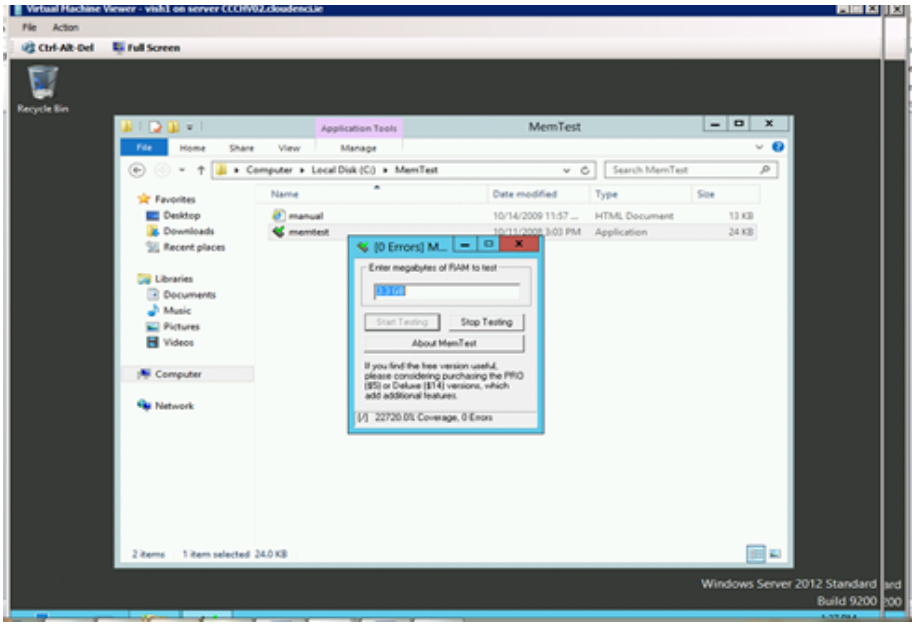

Figure 5.10: Memtest test

Table below illustrates various readings obtained from Memtest tool,

S No.	Readings	vish2	vish1	vish3
1	Reading 1	33200 percent coverage	45800 percent coverage	34226 percent coverage
2	Reading 2	33240 percent coverage	45880 percent coverage	34306 percent coverage
3	Reading 3	33296 percent coverage	45900 percent coverage	34373 percent coverage
4	Reading 4	33340 percent coverage	45776 percent coverage	34361 percent coverage
5	Reading 5	33390 percent coverage	45681 percent coverage	34340 percent coverage

Table 5.6: Different RAM Readings by Memtest in VMs

5.5 Logical networks in VMware:

VLANs in VMware ESXi are created a bit differently when compared to HyperV. vSwitch virtual switch is used to create the VLANs here. Virtual machine port groups are present in vSwitch which enables user to create port groups for managing the created VLANs. VLAN 313, VLAN 314,VLAN 315 are the three logical networks created which lie on vSwitch. They are assigned with IP addresses. Virtual machines vish1, vish2, vish3 are connected to VLAN 313, VLAN 314, VLAN 315 respectively. One IP address is assigned to each VM. Starting address of VLAN 313 is 172.25.12.20 and ending address is 172.25.12.250. VLAN 314 has starting address 172.25.13.20 and ending address 172.25.13.250. Starting address of VLAN 315 is 172.25.14.20 and ending address is 172.25.14.250.

Below figure depicts the architecture of VLANs

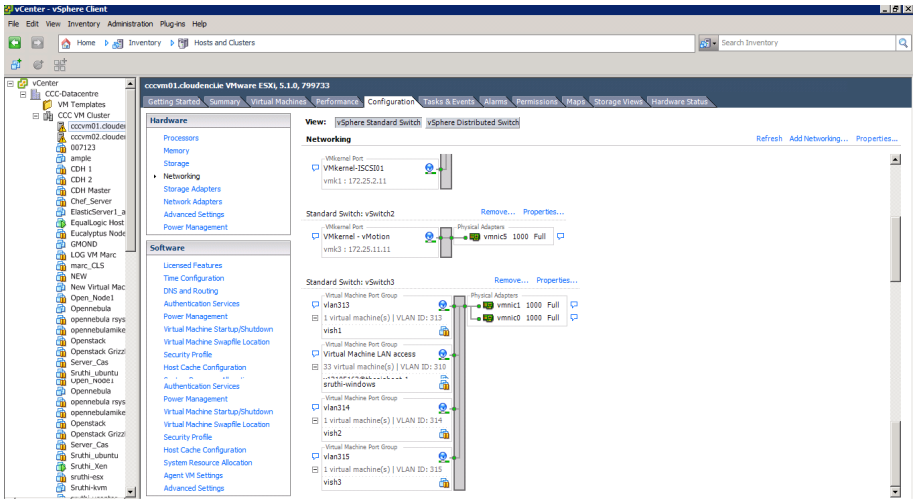

Figure 5.11: Networking Architecture

5.6 Benchmark readings for tools:

5.6.1 Benchmark readings for Iperf:

Iperf tool is executed on command line. Here the readings of client mode connection are taken into consideration. Server mode implementation will be covered in the Appendix section of the thesis.

S No.	Readings	vish2	vish1	vish3
1	Reading 1	3.47 Gbits/sec	3.30 Gbits/sec	3.43 Gbits/sec
2	Reading 2	3.43 Gbits/sec	3.27 Gbits/sec	3.93 Gbits/sec
3	Reading 3	3.55 Gbits/sec	3.38 Gbits/sec	3.99 Gbits/sec
4	Reading 4	3.43 Gbits/sec	3.25 Gbits/sec	3.98 Gbits/sec
5	Reading 5	3.57 Gbits/sec	3.30 Gbits/sec	3.96 Gbits/sec

Table 5.7: Different Bandwidth Readings by Iperf in VMs

5.6.2 Benchmark readings for Passmark:

The values generated after implementing Passmark tool are put into the table below. The readings for performance tests in every virtual machine are taken suspending all the other VMs from operation that are present on the host.

S No.	Readings	vish2	vish1	vish3
1	Reading 1	1344 Strings/sec	1348 Strings/sec	1353 Strings/sec
2	Reading 2	1342 Strings/sec	1354 Strings/sec	1347 Strings/sec
3	Reading 3	1355 Strings/sec	1336 Strings/sec	1344 Strings/sec
4	Reading 4	1335 Strings/sec	1343 Strings/sec	1352 Strings/sec
5	Reading 5	1327 Strings/sec	1346 Strings/sec	1339 Strings/sec

Table 5.8: Different Single Threaded Operations Readings by Passmark in VMs

5.6.3 Benchmark readings for IOzone:

The values of IOzone are taken implementing a command from iozone package that can give output in operations/sec. format. Below table illustrates it.

S No.	Readings	vish2	vish1	vish3
1	Reading 1	313179 Operations/sec	317355 Operations/sec	318071 Operations/sec
2	Reading 2	313393 Operations/sec	317355 Operations/sec	318181 Operations/sec
3	Reading 3	313715 Operations/sec	317245 Operations/sec	318679 Operations/sec
4	Reading 4	313769 Operations/sec	317080 Operations/sec	318790 Operations/sec
5	Reading 5	313769 Operations/sec	317025 Operations/sec	318679 Operations/sec

Table 5.9: Different Read Operations Readings by IOzone in VMs

5.6.4 Benchmark readings for Memtest:

Memtest tool is run taking all unused RAM into consideration from all VMs. A table drawn below shows various readings.

S No.	Readings(3.3 GB)	vish2	vish1	vish3
1	Reading 1	33186.0 percent coverage	33253 percent coverage	33426 percent coverage
2	Reading 2	33226 percent coverage	33226 percent coverage	33440 percent coverage
3	Reading 3	33213 percent coverage	33360 percent coverage	33413 percent coverage
4	Reading 4	33360 percent coverage	33213 percent coverage	33453 percent coverage
5	Reading 5	33173 percent coverage	33333 percent coverage	33413 percent coverage

Table 5.10: Different RAM Readings by Memtest in VMs

5.7 Conclusion:

This chapter determined tests carried out on VMs present in two hypervisors. An analysis of results gained from this part of the thesis will be evaluated in the next chapter.

Chapter 6

Evaluation

6.1 Introduction:

This chapter discusses the process of evaluation conducted to compare the performance of virtual machines created with VLAN specification on both the hypervisors. Graphs for performance tests on HyperV and VMware ESXi are demonstrated in section 6.1 and section 6.3. Finally a conclusion is provided to end this chapter.

6.2 Graph charts for performance test on HyperV:

A graph is drawn for each of the performance test conducted. This way we can evaluate the how every VM is behaving since they are assigned to VLANs.

6.2.1 Iperf analysis:

The three VMs present on HyperV cluster 1 namely, vish1, vish2, vish3 have separate VLANs VLAN 401, VLAN 402, VLAN 403 respectively. These VMs have been tested for performance. So all these VMs had given consistent performance generating identical network bandwidth for Iperf. However this analysis of bandwidth varies from VM to VM as each of them are connected to different VLAN. Therefore this kind of variation in bandwidth performance is expected from VM to VM. In the graph below, three VMs are taken. Bandwidth and time(in sec.) are the parameters for graph generated.

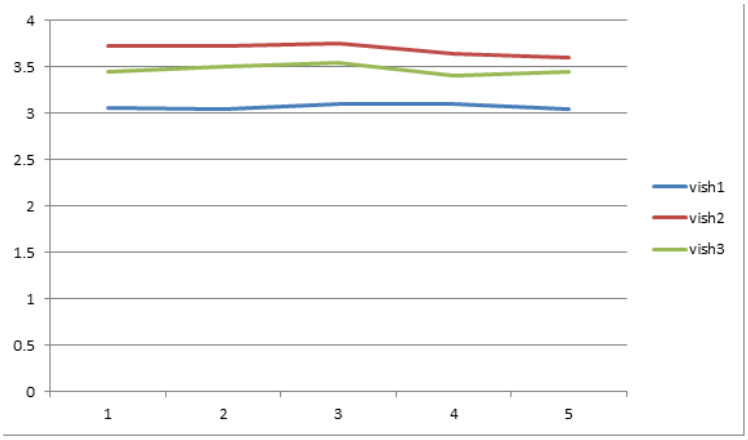

Figure 6.1: Bandwidth performance of VMs

6.2.2 Passmark analysis:

The values generated on each VM are similar to each other but they vary from VM to VM.So the three VMs show variable performance. Single threaded operations and time(in sec.) are the parameters to generate graphs.

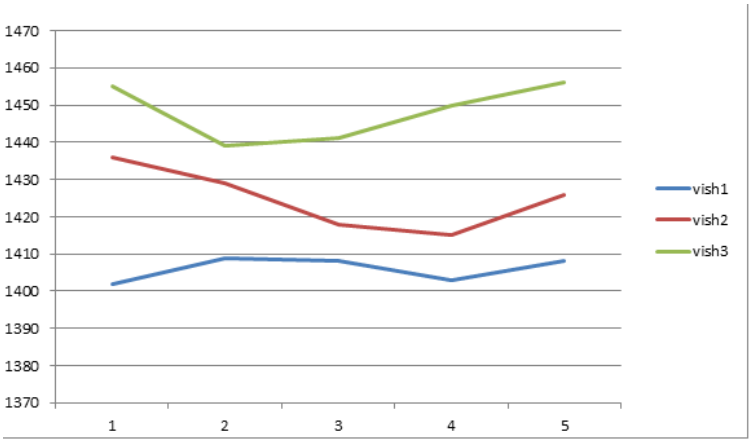

Figure 6.2: Passmark performance of VMs

6.2.3 IOzone analysis:

The values generated by IOzone are also in the same way like Iperf and Passmark. But the values resulted from Iperf and Passmark are much consistent for all the VMs. IOzone gives some variation in values which can be interpreted from the graph.

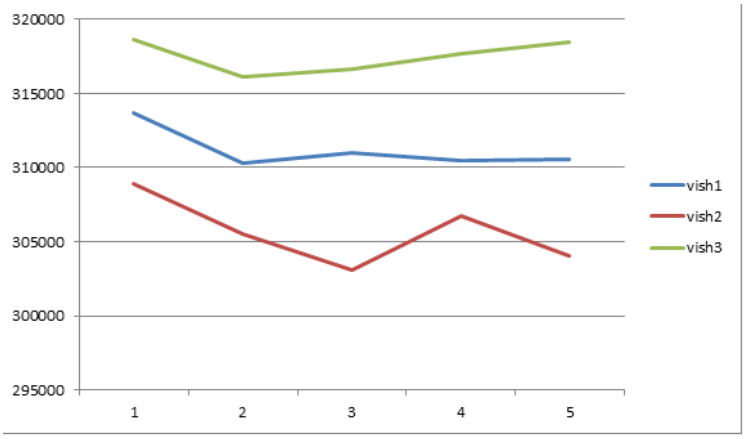

Figure 6.3: IOzone performance of VMs

6.2.4 Memtest analysis:

Memtest gives comparatively better performance than Passmark and IOzone. As the tool is implemented to test on unused RAM of 3.3 GB in every VM. Therefore, the results obtained from the test indicate that memory performance on virtual machines present on HyperV produces consistent values. Thus, memory performance is significant for the three VMs.

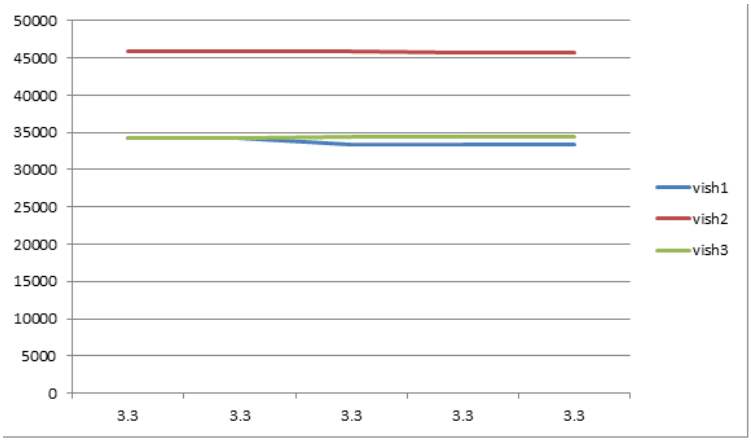

Figure 6.4: Memory performance of VMs

6.3 Graph charts for performance test on VMware ESXi:

Performance of VMs on VMware ESXi is evaluated by drawing graphs looking at the values obtained from tests. This helps in the analysis of entire performance.

6.3.1 Iperf analysis:

The VLANs 313, 314, 315 are assigned to each of the three VMs. Network bandwidth values are produced after running Iperf. There is a possibility of slight variation in the results as every VM is connected to an individual VLAN. However, below graph shows that values generated by one VM are almost nearer or same to the values generated by other VMs.

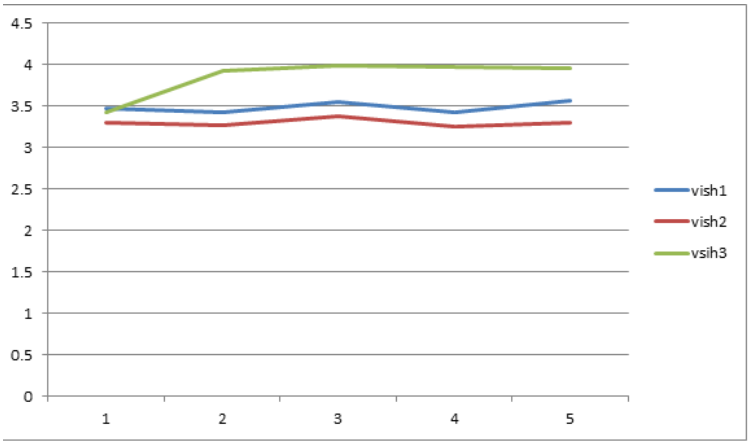

Figure 6.5: Iperf performance of VMs

6.3.2 Passmark analysis:

The values generated by Passmark are not so close enough which makes the VMs to perform differently. Passmark is the only performance test that resulted in producing distinct values. However, values generated in an individual VM are close enough to one another. Graph below shows variation but the individual VMs have consistent values.

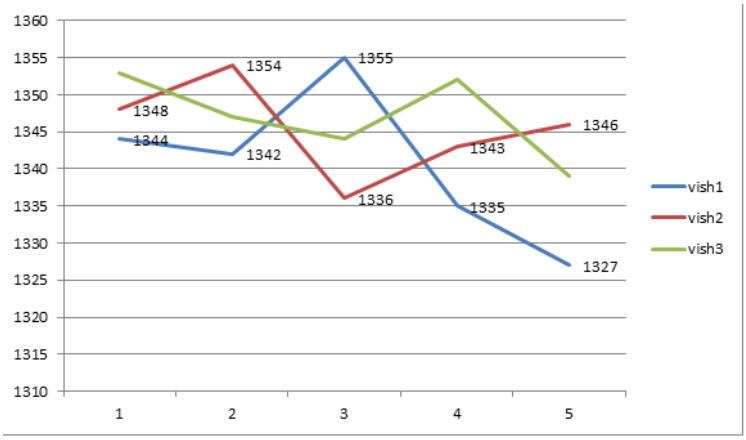

Figure 6.6: Passmark performance of VMs

6.3.3 IOzone analysis:

Disk performance of VMs on VMware ESXi given by IOzone tool is exactly same for every VM. So the analysis by below graph depicts that VMs on VMware show better performance.

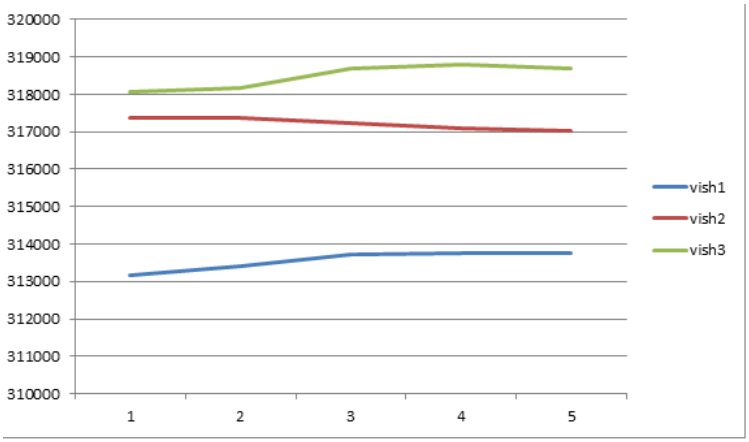

Figure 6.7: IOzone performance of VMs

6.3.4 Memtest analysis:

Memtest tool produces values that are identical to every other VM. The only difference created is some variation in values after every reading.

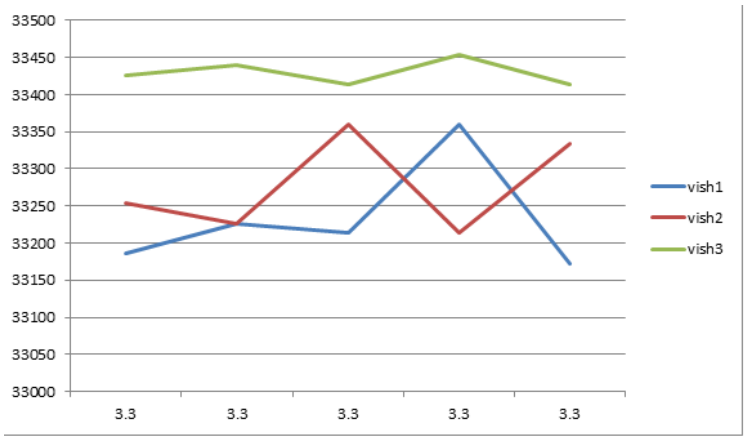

Figure 6.8: Memtest performance of VMs

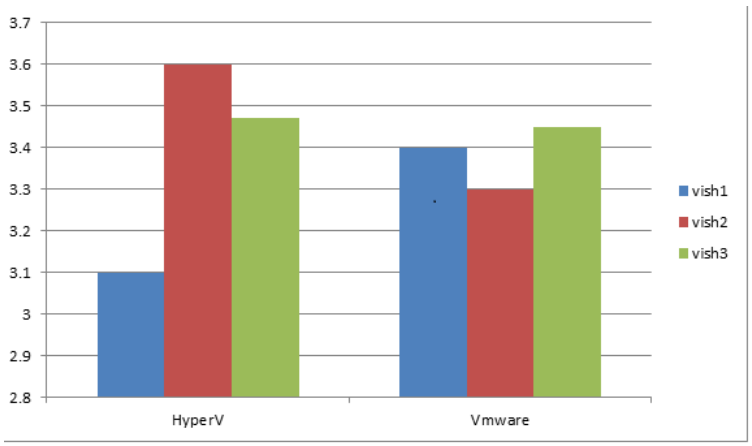

Figure 6.9: Performance of VMs on HyperV and VMware by Iperf

In the above figure, it is clearly evident that VMs on VMware ESXi show identical performance when tested with Iperf when compared to VMs on HyperV.

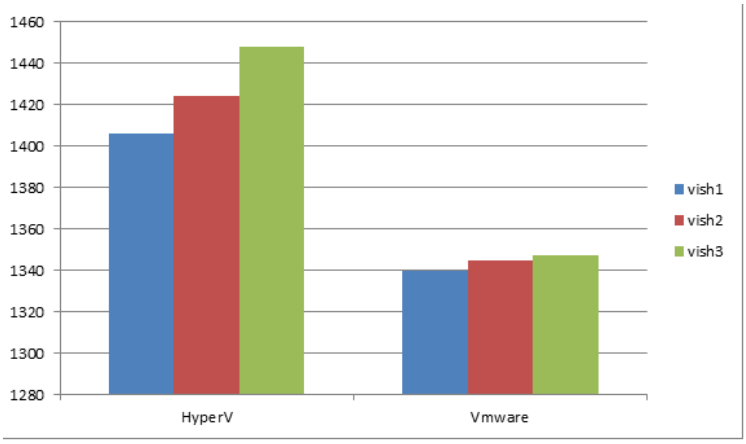

Figure 6.10: Performance of VMs on HyperV and VMware by Passmark

In the above figure, VMs on VMware ESXi demonstrate better performance for Passmark test.

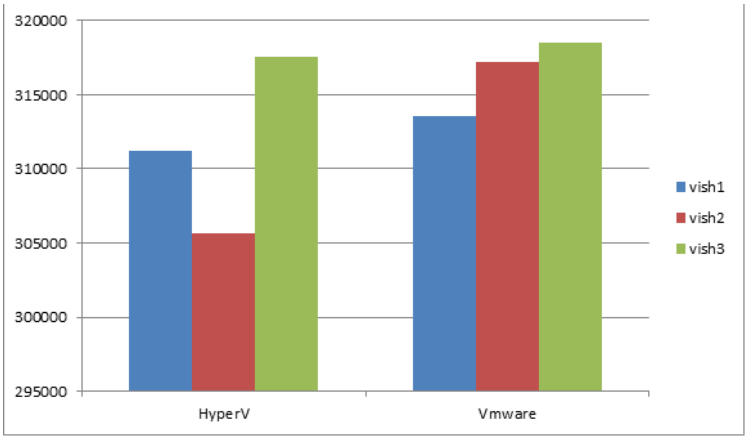

Figure 6.11: Performance of VMs on HyperV and VMware by IOzone

The above figure also depicts that tests by IOzone on both the hypervisors resulted in delivering better performance of VMware.

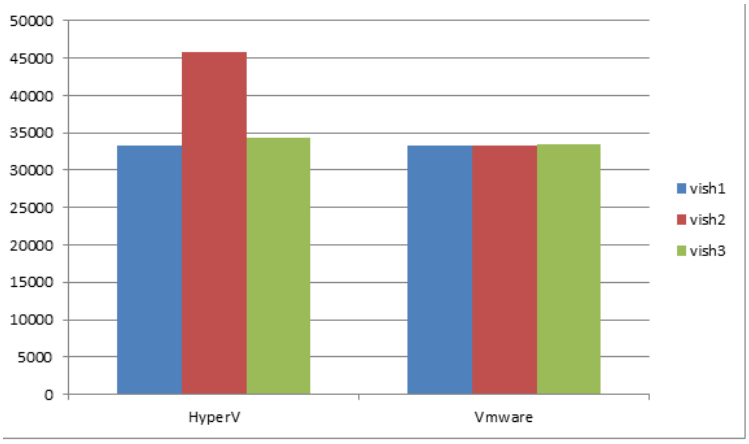

Figure 6.12: Performance of VMs on HyperV and VMware by Memtest

The figure above shows Memtest tests on two hypervisors. VMs on ESXi hypervisor give consistent performance.

6.4 Conclusion:

Evaluation focussed on analysing the results obtained from the tests conducted on virtual machines. Graphs are drawn to explain the behaviour of them against each test.

Chapter 7

Conclusions

7.1 Overview:

Network isolation is considered to compare the performance of virtual machines in VMware ESXi and HyperV virtualization technologies to understand the isolation concept further. The techniques used for network isolation are discussed and the network performance of virtual machines is discussed.

Chapter 2 provided required background details on the thesis. Concepts that are used for the thesis are explained in it. Literature review presented the previous work that had been carried in the field of network isolation. It investigated challenges in the security of virtual machines in which isolation is considered to be prominent. The next chapter design employed the structure followed to proceed with the dissertation of performance comparison on two hypervisors. This chapter explained about the logical networks created on HyperV and VMware ESXi. Following design is the implementation chapter that focussed on implementing tools on virtual machines to test performance of network, CPU, disk, memory. Performance results have been evaluated in chapter 6.

Both HyperV and VMware ESXi have uniform performances in terms of network, CPU, disk and memory tests. But VMware shows better consistency than HyperV. The values resulting from the above tests indicated that virtual machines on VMware showed readings close to one another. All the tests carried out on virtual machines on VMware ESXi yielded consistent results. The same tests when conducted on HyperV showed variable results in Passmark tool for CPU performance. Remaining tools for network, CPU, disk performance demonstrated consistent performance. By this, we can understand that there is little difference between the two hypervisors but VMware ESXi is more consistent.

7.2 Further Work:

This thesis worked on testing performance of virtual machines connected to individual VLANs on both hypervisors, HyperV and VMware ESXi. The results have been evaluated. So fulfilling the aim of dissertation which is to compare the performance of virtual machines is done. The same work can be carried further taking different hypervisors and considering many virtual machines.

Bibliography

[1] Stefan Berger, Ramón Cáceres, Dimitrios Pendarakis, Reiner Sailer, Enriquillo Valdez, Ronald Perez, Wayne Schildhauer, and Deepa Srinivasan. Tvdc: managing security in the trusted virtual datacenter. *ACM SIGOPS Operating Systems Review*, 42(1):40–47, 2008.

[2] Davide Cerotti, Marco Gribaudo, Pietro Piazzolla, and Giuseppe Serazzi. Flexible cpu provisioning in clouds: a new source of performance unpredictability. In *Quantitative Evaluation of Systems (QEST), 2012 Ninth International Conference on*, pages 230–237. IEEE, 2012.

[3] Borting Chen, Chien-Yu Lai, and Yu-Lun Huang. A performance mapping model for physical-to-virtual migration. In *Software Security and Reliability (SERE), 2013 IEEE 7th International Conference on*, pages 90–98. IEEE, 2013.

[4] NM Chowdhury and Raouf Boutaba. A survey of network virtualization. *Computer Networks*, 54(5):862–876, 2010.

[5] Jeff Daniels. Server virtualization architecture and implementation. *Crossroads*, 16(1):8–12, 2009.

[6] E Doulatyari, PA Kohan, and A Esmailpour. Quality of service support for a wifi-wimax network in a test-bed environment. In *Innovations in Information Technology (IIT), 2012 International Conference on*, pages 316–321. IEEE, 2012.

[7] Zhe Fan, Feng Qiu, Arie Kaufman, and Suzanne Yoakum-Stover. Gpu cluster for high performance computing. In *Proceedings of the 2004 ACM/IEEE conference on Supercomputing*, page 47. IEEE Computer Society, 2004.

[8] Dhruv Garg, Kamal Kant, and Abhay Bansal. Review of virtual machine migration in datacenters. *International Journal*, 3(6), 2013.

[9] Simon Grinberg and Shlomo Weiss. Architectural virtualization extensions: A systems perspective. *Computer Science Review*, 2012.

[10] Alexander Heinecke, Karthikeyan Vaidyanathan, Mikhail Smelyanskiy, Alexander Kobotov, Roman Dubtsov, Greg Henry, Aniruddha G Shet, George Chrysos, and Pradeep Dubey. Design and implementation of the linpack benchmark for single and multi-node systems based on intel® xeon phi coprocessor. In *Parallel & Distributed Processing (IPDPS), 2013 IEEE 27th International Symposium on*, pages 126–137. IEEE, 2013.

[11] Qunying Huang, Chaowei Yang, Kai Liu, Jizhe Xia, Chen Xu, Jing Li, Zhipeng Gui, Min Sun, and Zhenglong Li. Evaluating open-source cloud computing solutions for geosciences. *Computers & Geosciences*, 2013.

[12] Vimalkumar Jeyakumar, Mohammad Alizadeh, David Mazieres, Balaji Prabhakar, Changhoon Kim, and Windows Azure. Eyeq: practical network performance isolation for the multi-tenant cloud. In *Proceedings of the 4th USENIX conference on Hot Topics in Cloud Ccomputing*, pages 8–8. USENIX Association, 2012.

[13] Mahesh Kallahalla, Mustafa Uysal, Ram Swaminathan, David E. Lowell, Mike Wray, Tom Christian, Nigel Edwards, Chris I Dalton, and Frederic Gittler. Softudc: A software-based data center for utility computing. *Computer*, 37(11):38–46, 2004.

[14] Parag R Kaveri, Vinay Chavan, and Hemant Deshmukh. A study on resource oriented cloud computing. *International Journal*, 2(6), 2012.

[15] Han-gyoo Kim and Kee-cheol Lee. Experimental evaluation of pnfs protocol using network direct attached hard disk drives for mass storage system. *evolution*, 2(1), 2013.

[16] Palden Lama and Xiaobo Zhou. Ninepin: Non-invasive and energy efficient performance isolation in virtualized servers. In *Dependable Systems and Networks (DSN), 2012 42nd Annual IEEE/IFIP International Conference on*, pages 1–12. IEEE, 2012.

[17] Lorenzo Martignoni, Pongsin Poosankam, Matei Zaharia, Jun Han, Stephen McCamant, Dawn Song, Vern Paxson, Adrian Perrig, Scott Shenker, and Ion Stoica. Cloud terminal: secure access to sensitive applications from untrusted systems. In *Proc. USENIX ATC*, 2012.

[18] Goran Martinovic, Josip Balen, and Snjezana Rimac-Drlje. Impact of the host operating systems on virtual machine performance. In *MIPRO, 2010 Proceedings of the 33rd International Convention*, pages 613–618. IEEE, 2010.

[19] Gal Motika and Shlomo Weiss. Virtio network paravirtualization driver: Implementation and performance of a de-facto standard. *Computer Standards & Interfaces*, 34(1):36–47, 2012.

[20] Kashif Munir and Sellapan Palaniappan. Secure cloud architecture. *Advanced Computing: An International Journal (ACIJ), 4 (1), 9-22.*, 2013.

[21] Xing Pu, Ling Liu, Yiduo Mei, Sankaran Sivathanu, Younggyun Koh, Calton Pu, and Yuanda Cao. Who is your neighbor: Net i/o performance interference in virtualized clouds. 2012.

[22] Musfiq Rahman, Bruce R Childers, and Sangyeun Cho. Comet: Continuous online memory test. In *Dependable Computing (PRDC), 2011 IEEE 17th Pacific Rim International Symposium on*, pages 109–118. IEEE, 2011.

[23] Ellard Roush and Zoram Thanga. Zone clusters: A virtual cluster based upon solaris containers. In *Cluster Computing and Workshops, 2009. CLUSTER'09. IEEE International Conference on*, pages 1–8. IEEE, 2009.

[24] Jyotiprakash Sahoo, Subasish Mohapatra, and Radha Lath. Virtualization: A survey on concepts, taxonomy and associated security issues. In *Computer and Network Technology (ICCNT), 2010 Second International Conference on*, pages 222–226. Ieee, 2010.

[25] Gaurav Somani and Sanjay Chaudhary. Application performance isolation in virtualization. In *Cloud Computing, 2009. CLOUD'09. IEEE International Conference on*, pages 41–48. IEEE, 2009.

[26] Ivan Studnia, Eric Alata, Yves Deswarte, Mohamed Kaâniche, Vincent Nicomette, et al. Survey of security problems in cloud computing virtual machines. *Proceedings of Computer and Electronics Security Applications Rendez-vous (C&ESAR 2012)*, pages 61–74, 2012.

[27] Xiaolin Wang, Yan Sang, Yi Liu, and Yingwei Luo. Considerations on security and trust measurement for virtualized environment. *Journal of Convergence Volume*, 2(2), 2011.

[28] Jiayang Yu and Ruonan Rao. A method for solving the performance isolation problem in paas based on forecast and dynamic programming. In *Computational and Information Sciences (ICCIS), 2012 Fourth International Conference on*, pages 947–950. IEEE, 2012.

[29] Ning Zhang, Ming Li, Wenjing Lou, and Y Thomas Hou. Mushi: Toward multiple level security cloud with strong hardware level isolation. In *MILITARY COMMUNICATIONS CONFERENCE, 2012-MILCOM 2012*, pages 1–6. IEEE, 2012.

Appendix A

Appendix

A.0.1 VM creations on HyperV and VMware ESXi:

Creating virtual machine:

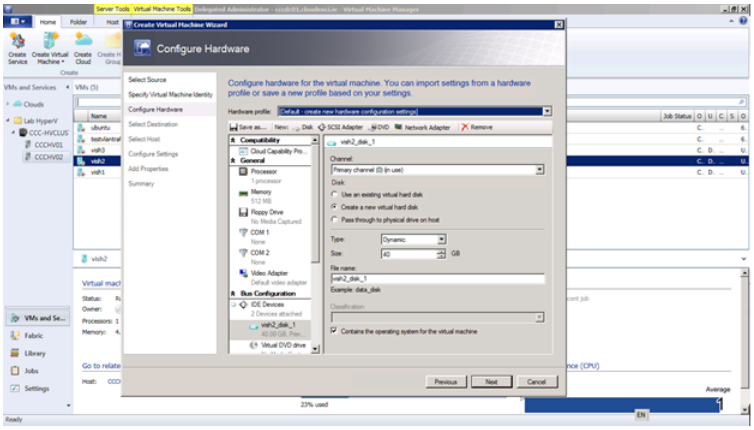

Figure A.1: VM creation

Virtual machine vish2 is created on HyperV.

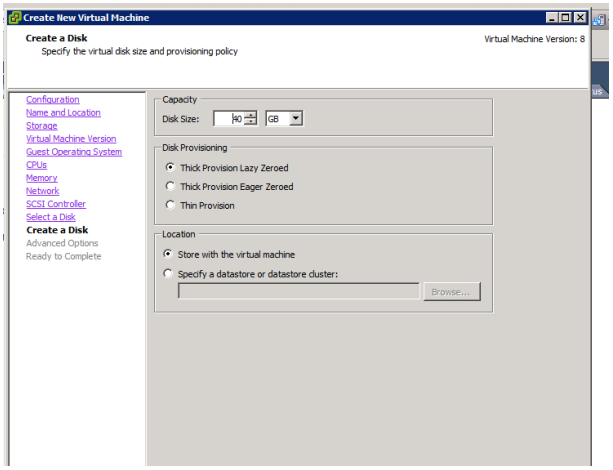

Figure A.2: VM creation

Virtual machine vish1 is created on VMware ESXi.

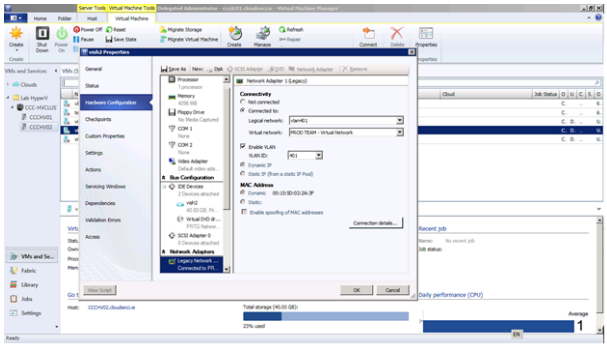

Figure A.3: logical network connection

logical network VLAN 401 and VLAN ID 401 is selected for vish2 VM.

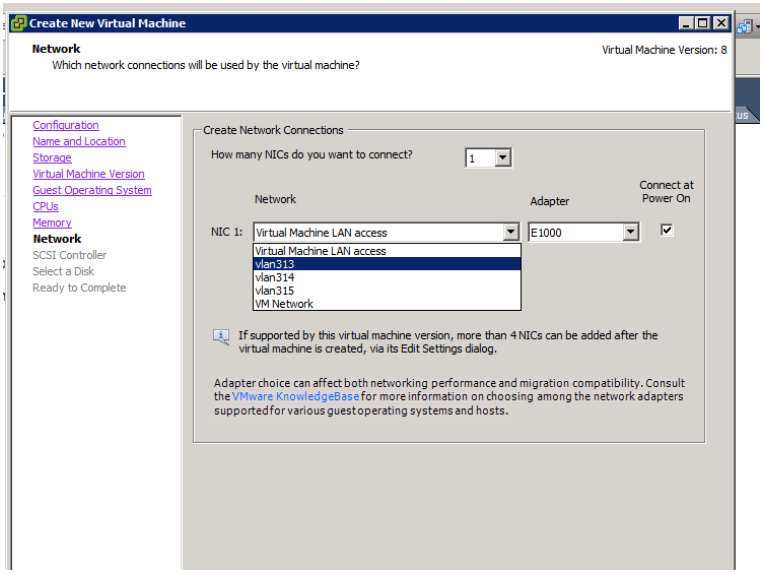

Figure A.4: logical network connection

logical network VLAN 313 is selected for vish 1 VM.

Assigning IP address to VM from IP pool:

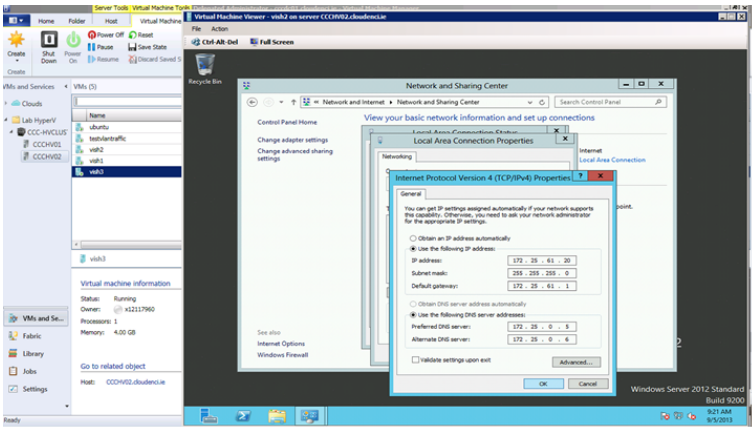

Figure A.5: Assigning IP address

VLAN 401 on HyperV:

1. Starting Address: 172.25.61.20
2. Ending Address: 172.25.61.250
3. Subnet: 172.25.61.0/24

4. Default gateway: 172.25.61.1
5. Preferred DNS server: 172.25.0.5
6. Alternate DNS server: 172.25.0.6

It is done the same way for VLAN 402 and VLAN 403 on HyperV which are allotted IP addresses as 17.25.62.20-250 and 172.25.63.20-250 respectively.

VLAN 313 on VMware ESXi:

1. Starting Address: 172.25.12.20
2. Ending Address: 172.25.12.250
3. Subnet: 172.25.12.0/24
4. Default gateway: 172.25.12.1
5. Preferred DNS server: 172.25.0.5
6. Alternate DNS server: 172.25.0.6

VLAN 314, VLAN 315 are also allotted IP addresses in the same way as above. VLAN 314 has IP addresses in the range of 172.25.13.20-250 and VLAN 315 has 172.25.14.20-250.

A.1 Tool Implementation:

Iperf:

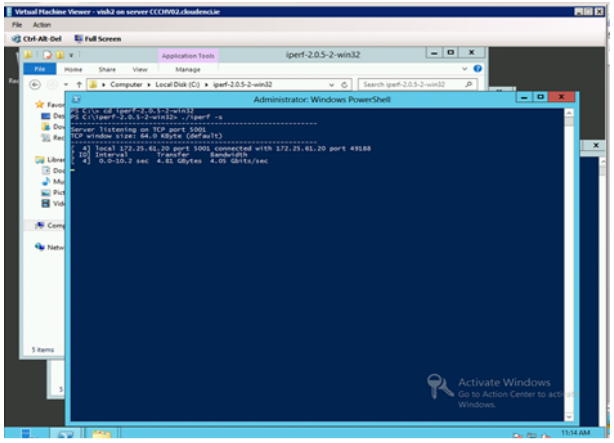

Figure A.6: Iperf command for server connection

The command iperf -s is used for establishing connection at server side.

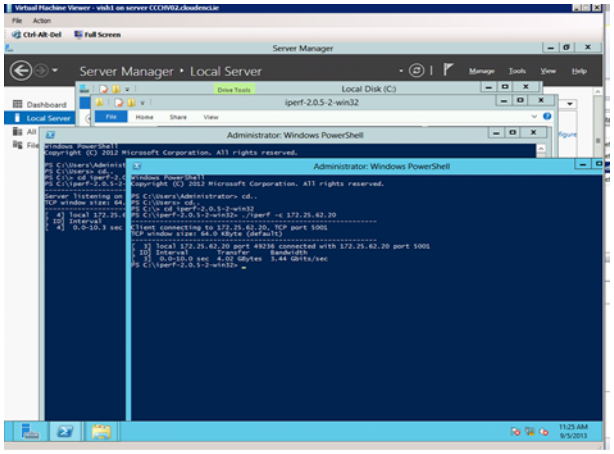

Figure A.7: Iperf command for client connection

Passmark:

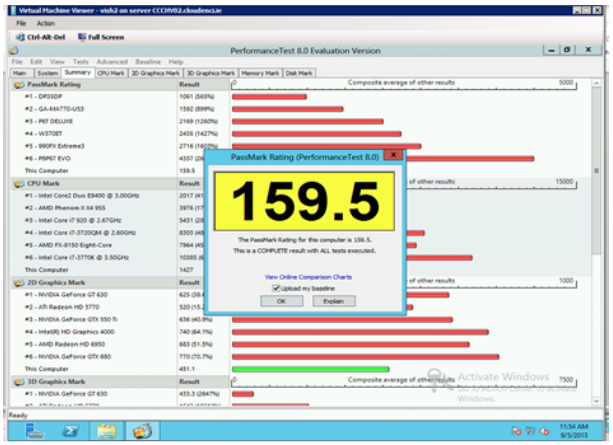

Figure A.8: Passmark execution

Passmark gives system rating also.

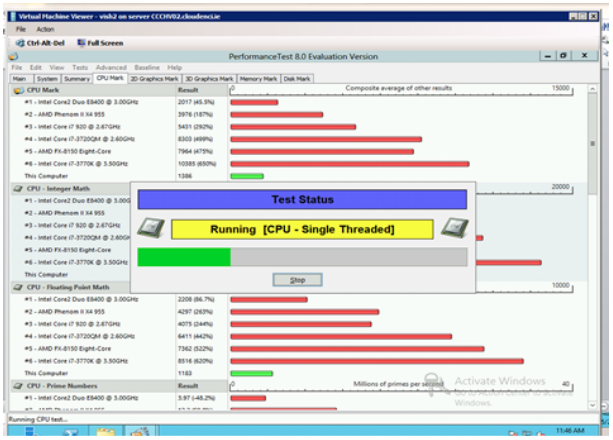

Figure A.9: Passmark single threaded operations execution

IOzone:

Other commands of iozone test:

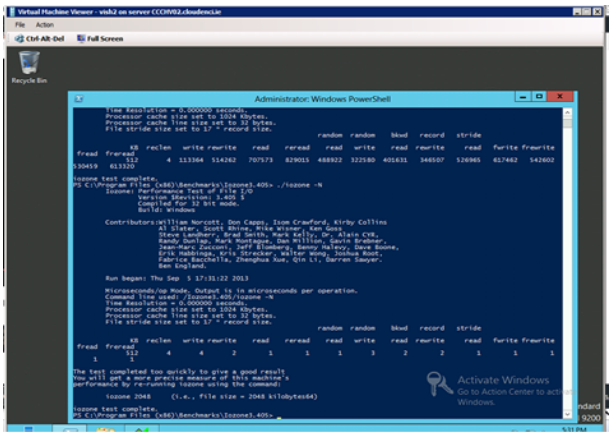

Figure A.10: IO command for output in microseconds

command is ”iozone -N”

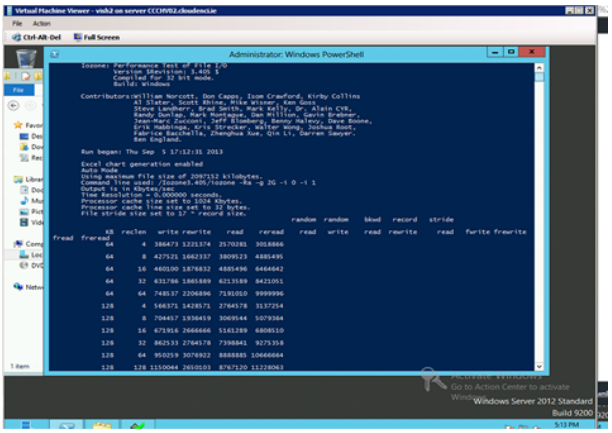

Figure A.11: IOzone command for read/write

command is ”iozone -Ra -g 2G -i 0 -i 1” that is used to test read/write tests. It does not execute rest all of the tests in order to save time.

Memtest:

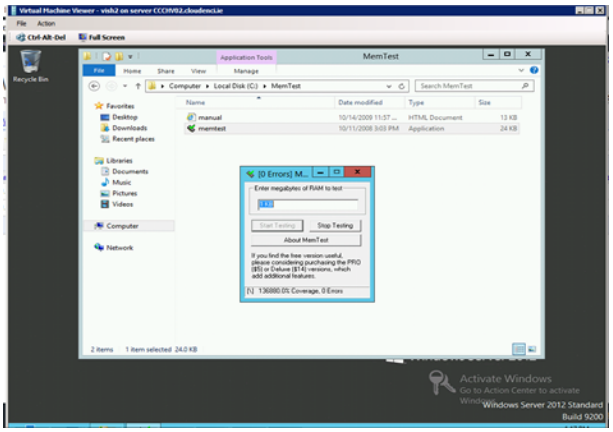

Figure A.12: Memtest execution

Memtest exection for 1 KB Memory of RAM.

A.1.1 Configuration of HyperV and VMware ESXi:

Hardware and software configuration of HyperV and VMware ESXi hypervisors on cloud.

Configuration on HyperV:

1. Processor - Intel Xeon 2.50 GHz 24 logical processors
2. Memory - 23.96 GB total, 17.86 GB available
3. Storage - 4,143.89 GB capacity, 3,843.96 GB available
4. Operating system - Microsoft Windows Server 2008 R2 Datacenter SP1
5. Virtualization software - Microsoft HyperV
6. VMM agent - Version 3.0.6005.0
7. Domain - cloudenci.ie

Configuration on VMware ESXi:

1. Processor - Intel(R) Xeon(R) CPU E5-26400@ 2.50GHz

2. Memory - 24530.48 MB total, 17277.00 MB available
3. Storage - 4.00 TB capacity,204.14 GB available
4. Operating system - Bare metal(No operating system)
5. Virtualization software - VMware ESXi
6. VMM agent - vSphere client 5.1.0,799733
7. Domain - cloudenci.ie