Journal of Computing & Biomedical Informatics
ISSN: 2710 - 1606

Volume 02 Issue 01
2021

Research Article

Assessment of Network & Processor Virtualization in Cloud Computing

Rashid Mahmood[1*], Muhammad Imran[1] and Sayyid Kamran Hussain[2]

[1]Department of Computer Science,Bahauddin Zakariya University, Multan, Pakistan
[2]Department of Computer Science, University of Sargodha, Sub Campus Bhakkar, Pakistan
[*]Corresponding Author: Rshid Mahmood. Email: rashidbzu075@gmail.com

Received: November 02, 2020 **Accepted:** February 19, 2021 **Published:** March 15, 2021

Abstract: Cloud Computing is an emerging field in applied computer science. Cloud computing depends on virtualization, where a sole physical resource is virtualized into numerous virtual resources. Resources that are being virtualized include physical storage, processing units, volatile memories, network, and further physical resources. Processor and network virtualization offer several advantages like lessening hardware cost, energy consumption, and the human struggle for administration and management of resources. In this article, the primary concentration was on the evaluation and assessment of network virtualization between different cloud computing hypervisors and to contrast their evaluated results. This comparison is based on a few experiments performed on Citrix XenServer, Microsoft Hyper-V, and VMware vSphere hypervisors. XenServer uses Open vSwitch for its network virtualization whereas Hyper-V and vSphere use Microsoft network virtualization and vSphere switch for managing their network virtualization respectively. We have found that virtual networks are further scalable and secure than physical networks. We can assign multiple virtual interfaces or connections to a single virtual machine. In light of various experiments, in which we have compared non-virtualized with virtualized scenarios, for all these above-mentioned hypervisors, we found out that we can use virtual networks with and without VLAN Tags or VLAN Ids. Hyper-V provides further flexibility and options regarding VLAN Tag in comparison with XenServer or vSphere. We have also found that XenServer and vSphere give us extra network throughput in comparison with Hyper-V in terms of TCP and UDP whereas Round Trip Time (RTT) in the virtual network of all hypervisors are high in comparison with the non-virtual environment. Network CPU usage is also measured and found that vSphere and XenServer together consume more CPU during network activity in comparison with Hyper-V. We also found that processor virtualization did not majorly result from the average schedule time in comparison with a non-virtualized machine. We consequently conclude that in general a small reduction in the performance, in the case of virtualization is insignificant as compared with the advantages we get from virtualization when using cloud hypervisors. For the application of virtualization, this work will motivate to set up of data centers using a virtualized environment.

Keywords: Virtual Machine, Virtual network, Network virtualize environment, Cloud Computing

1. Introduction

In this section, we will be discussing much basic knowledge required for fully understanding the cloud computing. We will be creating this section as a required base knowledge for ongoing our discussion in the coming sections of this report. We can mainly divide this section into sub sections. In 1st section, we will be discussing the term "Cloud" itself and "virtualization" will be discussed in 2nd section. 3rd section we will summarize the rest of the article. In the current era, almost all of us have heard of the latest hardware technologies and enhancements, made by the technologist, from dissimilar vendors like Intel, Apple, AMD, and other manufacturers. The latest machines are equipped with enhanced and updated technology. These machines are much faster and much capable of doing stuff in comparison to those earlier machines which were in use like 15 or 20 years ago. If we do various research on the improvement and enhancement process of these technologies in detail based on which these latest hardware equipment are built, we will realize that hardware improvement has almost touched its peak. The continuous evolvement has already made hardware equipment so dense and complex that further improvement is not an easy task to achieve, and even if possible it will not be inexpensive and cost-effective. Therefore, manufacturers and technologists have come to a point where they are extending their research finding a way to obtain maximum output from the existing hardware instead of making it more dense and complex. Cloud computing has also emerged as a result of these researches.

2. Materials and Methods

For many of us, Cloud Computing is a very fascinating word and several are still not sure what it is. Cloud computing is a computing paradigm. According to Vaquero et al. [1], cloud computing is built on the concepts of virtualization, distributed computing, or utility computing. In simple words, cloud computing is a distributed computing system. Before we get into further details that what we do with these cloud systems or in fact with cloud computing, we will describe the term cloud computing. There are various definitions for cloud computing but none is concrete and complete. Every definition differs from the other, with a focus on one or additional aspects of cloud computing. The main focus of all these definition revolves around scalability, resources monitoring, automation and optimization, collaboration, expensiveness, utility computing, user-friendliness, services, virtualization of hardware and software, and many more [1]. On the other hand, in [1] authors had proposed a self-explanatory definition for the cloud, in which they had considered all the features and factors a cloud system could hold. According to Vaquero et al. [1], Clouds are the large pools of easily usable and accessible virtualized resources (such as hardware, development platforms, and/or services). These resources can be dynamically reconfigured to adjust to a variable load (scale), allowing also for optimum resource utilization. This pool of resource is typically exploited by a pay-per-use model in which guarantees are offered by the infrastructure provider by means of customized services level agreements (SLAs)". From then on top of the mentioned features and definitions, we can assume that the main characteristics of a cloud system are virtualization, service models, and scalability.

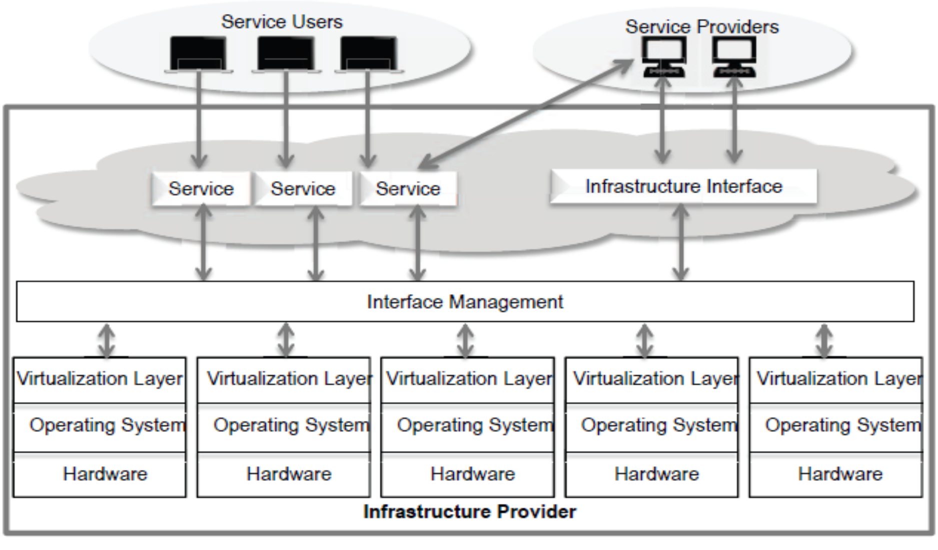


Figure 1: Service Provider, Infrastructure Provider, and Service User Service Provider and Service User Relation

On basis of the discussion made in the prior section, we can say that the cloud is a system that provides services to its user. These services are offered to users by different Service Providers (SP), whereas Service Provider is an entity that established and manages the cloud system as well as offers services to the users. These services are usually accessible to users or subscribers over the internet. Cloud tries to outsource these offered services to the cloud Infrastructure Providers (IP) as a service. Due to this, computing resources are transferred from SPs to IPs, resulting in flexibility and reduction in cost to SPs, Figure 1 depicts the above discussion [1] [2].

Based on cloud services, we can categorize these services into three different models. Infrastructure as a Service (IaaS): Infrastructure providers usually have data centers with multiple physical machines running various hypervisors on them which can support multiple virtual machines. These virtual machines contain virtualized resources. Usually, a user connects to one or more virtual machines using the internet. For users, these virtual machines act as physical machines, which they can remotely access. [1] [3].

Platform as a Service (PaaS): In contrast to IaaS, in PaaS user is provided with a computational platform. The User connects to the system using various sorts of the software interface and uses the system. Hardware resource demand and management are kept transparent to the user. Microsoft Azure and Google App Engine are examples of PaaS [1] [3]. Software as a Service (SaaS): In this model; software, business application, or web utilities are provided to the user as services. The User just connects to these services and uses them. Dropbox is an example of SaaS [1] [3].

Virtualization

In the previous sections, we have been using the term "Virtualization" repeatedly. But what do we mean by "Virtualization"? According to Mike Adams, director of product marketing at VMware a pioneer company in virtualization and cloud software and services, "Virtualization software makes it possible to run several operating systems and several applications on the same server at a similar time. It enables businesses to decrease IT costs while increasing the efficiency, utilization, and flexibility of their existing computer hardware." [4].

On the basis of Adam's description, we can say that virtualization means an environment where resources are executed in a controlled virtual environment rather than on actual physical resources. By the use of virtualization, a sole physical machine resource including main memory, microprocessor, and storage can be utilized as multiple virtual processors, memory, and storage. These virtual resources can keep running in parallel coming about the improved efficiency of the hardware. [5]. Virtual resources are managed and separated from physical resources by a virtual machine monitor (VMM) [4].

Virtualization Is Not Cloud Computing

Generally, there is a common misconception that virtualization is actually cloud computing, however, in reality, this is not the case [6]. According to Adams, "Virtualization is a foundational element of cloud computing and helps deliver on the value of cloud computing. Cloud computing is the delivery of shared computing resources, software or data as a service and on-demand through the internet". [4]. This confusion generally arises because of a deep relationship between virtualization and cloud computing. Virtualization and cloud computing collectively provide various services, like private clouds [4]. Virtualization, which is further specifically referred to as a virtual machine (VM), is built on top of physical computing resources. These VMs have their own virtual storage drive and network connectivity, whereas the cloud is responsible for determining how these virtualized resources are allocated, delivered, and presented. A cloud environment can be set up without even using virtualization. But with virtualized resources, it is more scalable as compared to the non-virtualized environment [6] [7].

Rick Philips, vice president of computer solutions at an IT firm Weiden hammer, elaborated on the dissimilarity between virtualization and cloud computing. According to him, "The difference is that a true cloud provides the self-service capability, elasticity, automated management, scalability, and pay-as-you-go service that is not inherent in virtualization." [4]. Thus, we can say that virtualization is itself not a cloud, but in fact, the cloud is based on virtualization.

Virtualization in Cloud Computing

Up to now, we have acquired needed knowledge about virtualization; so we can conclude that virtual resources in cloud computing are mainly contained virtualized processing units, memory, storage, and network. There are a number of articles discussing these virtual units. In a cloud environment, there are a number of virtual resources which collectively act as a cloud system and are used to deliver services to the end client or users. For example, figure 2 represents an extremely essential instance of a cloud environment, in this cloud environment different virtual resources, are being used to give administrations to clients.
In order to provide services for these virtual resources, or if we specifically use the term VM, works collaboratively.

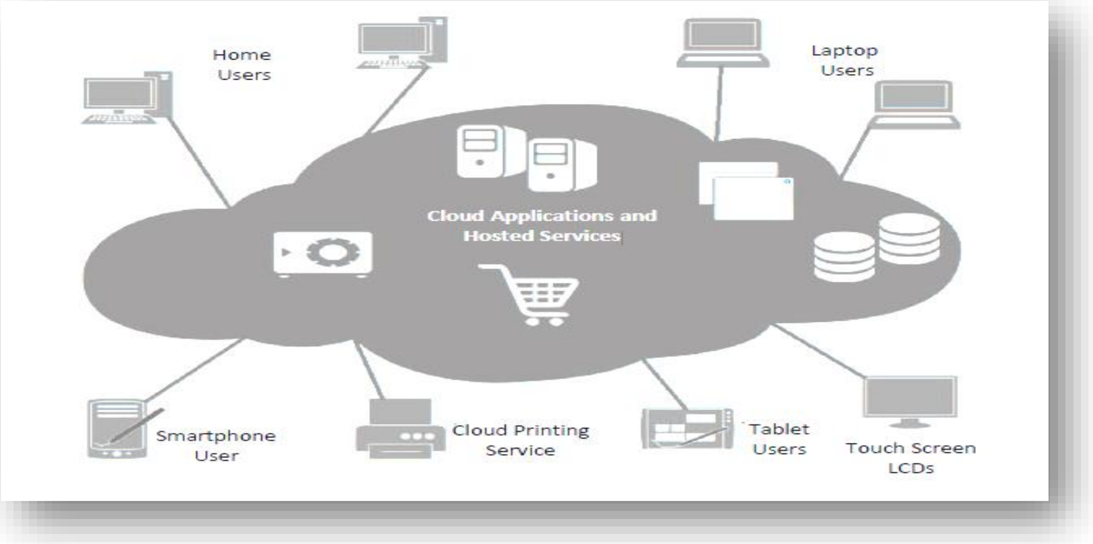


Figure 2: Cloud Environment

If these VMs collaborate with each other then there should be several sorts of communication mechanisms because without collaboration it would not possible. The answer is yes, there is a mechanism, named as a network or further specifically network virtualization [5]. If there is communication then there should also be various ways to manage and control this communication. The virtual environment has its own benefit but various challenges too, like scaling, migration, isolation, and security [8]. These challenges require being addressed [8].

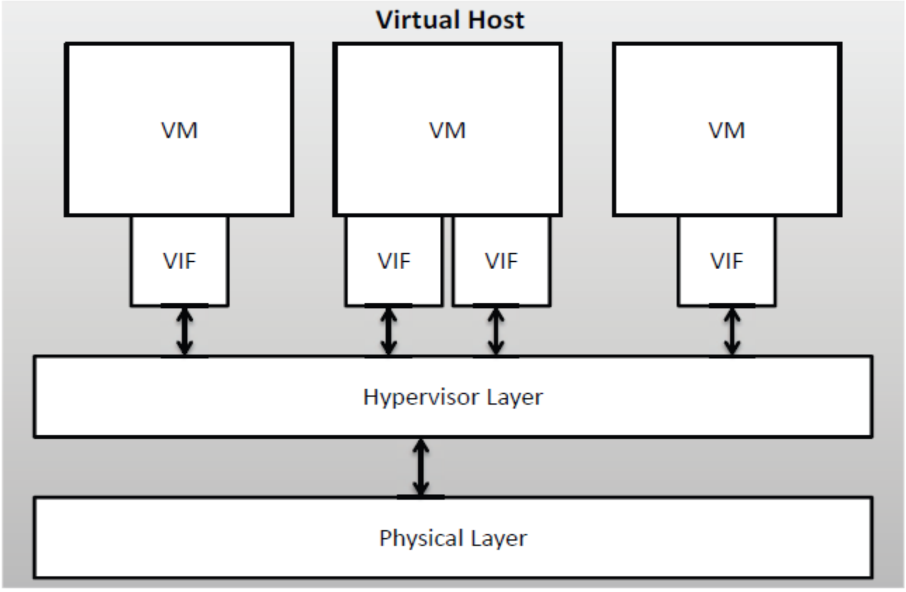


Figure 3: Virtual Network in Hypervisor

In that we consider physical networks, we can track and control network communication among nodes utilizing tools and indicated hardware, yet in the virtual environment, it is quite difficult. In the virtual network, communication is being directed at the virtual layer, where it could conceivably go from any physical network hardware, making it much further hard to track and control the network communication [5] [8] [9]. Network virtualization promises to tackle all these challenges.

3. Related Work

In this, we shall explain the work which has been done previously in the domain of cloud computing and virtualization and how their work change from our work. At the end of this section, we will also explain our contribution toward network virtualization. The main focus in research made in past revolves around the mechanisms for the implementation of the algorithms, virtualization, and other related areas in the cloud computing environment. In this section, our major focus will be on describing the work which has been done in previous research on network virtualization.

A virtual machine (VM) is an imitation of a particular computer system and acts as a virtual computing resource in a cloud environment. As traditional network models cannot control and monitor the interconnection between these virtual resources, so a strong mechanism is required i.e. network virtualization. Network virtualization provides extra benefits and features over physical networks, like software level, manage, and management.

In the current year's network virtualization has gained significant attention from researchers. In [10], Chowdhury et al. have discussed network virtualization and the four existing technologies which are quite the same as the concept of network virtualization. These technologies are named virtual local area networks, virtual private networks, active and programmable networks, and overlay networks. They also identified the area and projects where network virtualization could be implemented and also talk about the architectural feature of network virtualization in a cloud. According to the authors, in order to overcome challenges and issues in the existing system and internet, the solution is to use network virtualization with the support for coexisting many current architectures, instead of redesigning all once more from scratch. We have paid attention to network virtualization evaluation in dissimilar hypervisors.

Open vSwitch (OVS) is an open-source application that is used for virtualization of network and it moreover reveals communication among VMs. Pfaff et al. [8] described how network virtualization challenges could be resolved by using open vSwitch. According to [8] between PIFs and VMs Open vSwitch act as a bridge. While acting as an L2 switch it replicates the Ethernet Switching behavior by using VLAN, RSPAN, and ACL. For managing and forwarding the states, Open vSwitch export the interface whereas, for connectivity management, it follows a very alike model to OpenFlow [11]. In [8] Pfaff et al. have also described how Open vSwitch could be used for implementing the central management, virtual private networks, mobility, distribution configuration, visibility across network hosts, and separation between IP subnet. We, then again, have concentrated on the evaluation of Open vSwitch in light of a few parameters utilizing XenServer hypervisor. In [9] Pettit et al. developed an edge-based virtual network switch. Firstly they evaluated the features provided by high-end physical switches and then tried to incorporate them in Open vSwitch. In [9] Pettit et al. also highlighted that switching packets could cause high CPU utilization. They conclude that the combination of these software and hardware offloads could boost the balance between flexibility and performance. They [9] also incorporated the Open vSwitch in XenServer and studied the behavior in terms of costs, performance, central management, visibility, and policy handling. We have focused on the evaluation of network performance and CPU utilization of virtual switches in XenServer,

Hyper-V, and vSphere

Tseng et al. [12] have proposed a network virtualization framework based on OpenFlow in which physical infrastructure is virtualized into network elements. For supervision network activities, they [12] also proposed locally manage and administered MAC addressing based Layer 2 network virtualization. Our proposed research work focuses on the assessment of virtualization using Open vSwitch in XenServer, Network Virtualization in Hyper-V, and vSphere Standard Switch in vSphere. Instead of actually implementing these switches the focus will be on the output and the variables, which may influence the performance of the system. Cloud is an infrastructure that is utilized to give services to the clients while virtualization is utilized to accomplish the highest performance. These virtual resources are created, assigned, and control dynamically. There are possibilities that these virtual resources are transferred from one data center to the other. As migrations are conducted in the live environment, network performance is very critical during the migration. [7] Proposed a new mechanism where they used a kernel-based virtual machine (KVM) linked through L2 tunnels and were connected via Open vSwitch and GRE over Ethernet protocols. This mechanism could be easily used for the relocation processes but still in the virtual environment all VMs would be sharing similar I/O ports. Dong et al. [13], have proposed the SR-IOV (Single-Root Input-output Virtualization) based structural design to handle the challenges posed by I/O performance. SR-IOV allows an I/O port to be shared by several VMs without losing performance. The proposed architecture [13] was implemented and deployed on the Virtual Machine Monitors (VMM) for SR-IOV supported devices. Moreover, those SR-IOV device drivers are highly portable to work with the VMM. They additionally [13] recommended a Dynamic network interface-switching (DNIS) plan for addressing the relocation challenges postured through virtualization. For statistics center plans utilizing XenServer and Open vSwitch, [13] can function as a manual for taking care of the movement challenges.

Soares et al. [14] have discussed the resource allocation problem. They argued that in data centers, there is no require binding the resources to the network based on the model they are following. Instead of resources should be spread throughout the network and have to be owned by the network operator. To address this resource allocation problem they proposed an algorithm through combined virtualization of network and cloud resources. Similarly, in this article, the key focus is also on virtual network performance. For migration, the dynamic allocation of resources hosted on network performance is very serious. We have conducted the experiments on virtual switches and measured the network throughput and round-trip time. For example, sharing among all virtual resources, we have also conducted a study on XenServer, Hyper-V, and vSphere hypervisors.

In [2], Popa et al. have proposed a further very interesting technique for managing the policies over cloud operations. They proposed to shift the policy and access management from the network layer to the end hosts hypervisors only and named it as CloudPolice. They argued that by using CloudPolice, it is satisfactory to just have CloudPolice for access and policy management. They claimed that in comparison with the network layer management Cloud Police can support further sophisticated access control systems. We have observed and tested this behavior in Hyper-V where we can assign VLAN Id to a complete host also as an alternative to assigning Id to VMs individually.

Lee et al. [15] suggested a productive network services placement structure, INP (In-Network Processing) that could capably move the network services preparing from individual hardware to computing resources and network devices. For switching management and control, they applied Xen hypervisor, and OpenFlow become applied as a processing module. In addition, Open vSwitch was utilized to manage the network flow. In our experiment, we have only concentrated on the security feature of Open vSwitch and evaluated it by using experiments and a network packet analyzer. The major purpose of a cloud system is to host and offer services to the end-users. For offering these services network performance is very dangerous. In [16] Duna et al. have suggested a service-oriented architecture (SOA), which obtains the brief awareness of services users towards performance by modeling and analyzing the network and computing services. They likewise evaluated SOA in the network virtualization environment and analyzed the performance of the network. At the point when SOA was connected with an in-network virtualization environment, the resultant conduct incredibly portrayed the advantages of cloud and integration networks, and that they named it as "Network as a Service" (NaaS) [17]. They additionally suggested a structure for Network as a Service (NaaS) whose significant center became on network service arrangement, description, and disclosure. [18] Has also talked about NaaS hybrid clouds. A hybrid cloud is a cloud environment where an organization provides and manages resources. Several of which resources are in-house and others are external. Kapadia et al. also identified the network requirements to adopt the hybrid cloud environment. In [19] authors have talked about a further technique for provisioning cloud services using virtualization. They proposed a method with several QoS aware service selection algorithms for composite cloud services. However, the focus of our research is not on the services being hosted on the cloud but this is will be included in our future work.

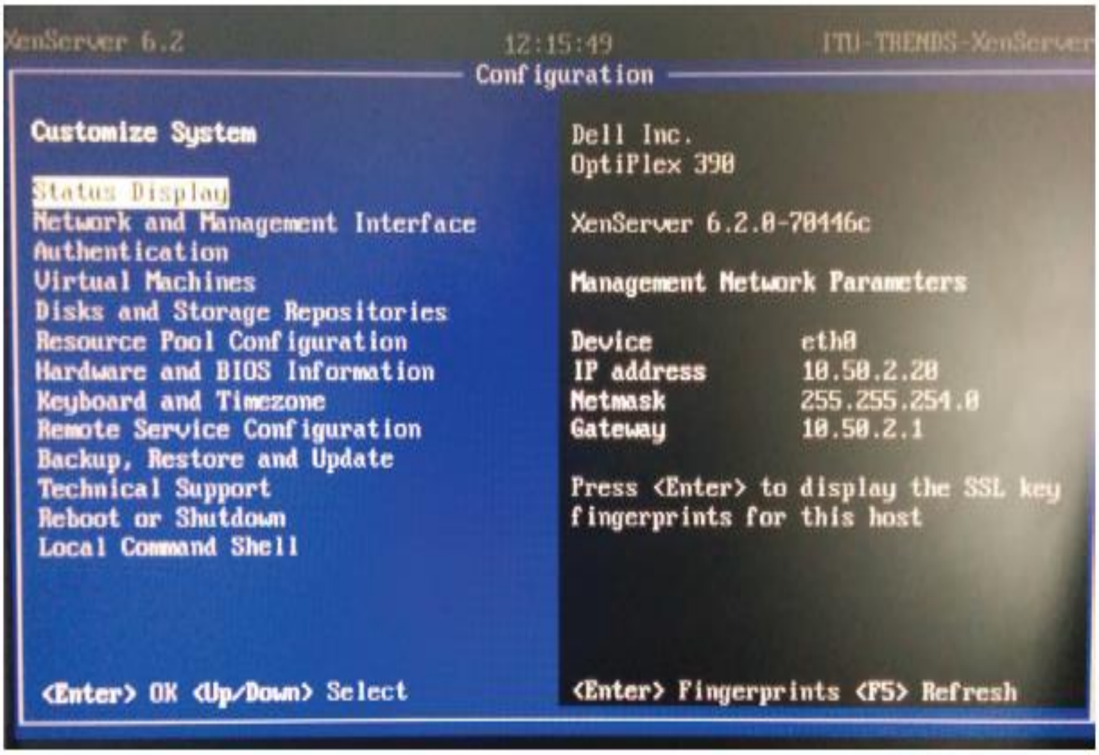
4. Results

In the last sections, we have talked about the XenServer, Hyper-V, vSphere, VMs, virtualization, Open vSwitch, XenCenter, and all other related concepts. In this part, we will be working on the experimental environment setup and evaluation section of our work. Our main focus will be on setting up the virtualized cloud environment for all these mentioned hypervisors. At first, we will be installing and configuring each hypervisor on two dissimilar physical machines. After installation, VMs will be created and configured on each hypervisor host machine and appropriate virtual networks will be assigned to these VMs.5.2. XenServer.

Server Installation

In this section, we will be explaining the experimental setup for XenServer hypervisors. Machines with similar configurations are used for all hypervisor host servers.

As a first step, we grabbed the ISO file from the Citrix website. This file can be downloaded from [34]. After the download completes, wrote this image on a USB drive and mode it bootable. We have used XenServer version 6.2.0 in our experiments. This version is delivered with the Open vSwitch form 1.4.6. XenServer depends on the Linux kernel, and its installation is not different than any other Linux flavor installation. But there is a restriction in hardware requirements. The on top of mentioned version is only installable on 64-bit x86 machines. At the start, we were trying to install it on 32-bit x86 machines booting process started without creating any trouble. XenServer can be installed on a machine with a max of ITB and min 1GB of RAM, with max 19 and, min 1 with 100Mb of faster NICs, max 160 logical processors, and min 100MB of disk space. For running Windows-based VMs, an Intel VT or AMD-V 64-bit x86-based system with one or more CPU(s) is required [36].



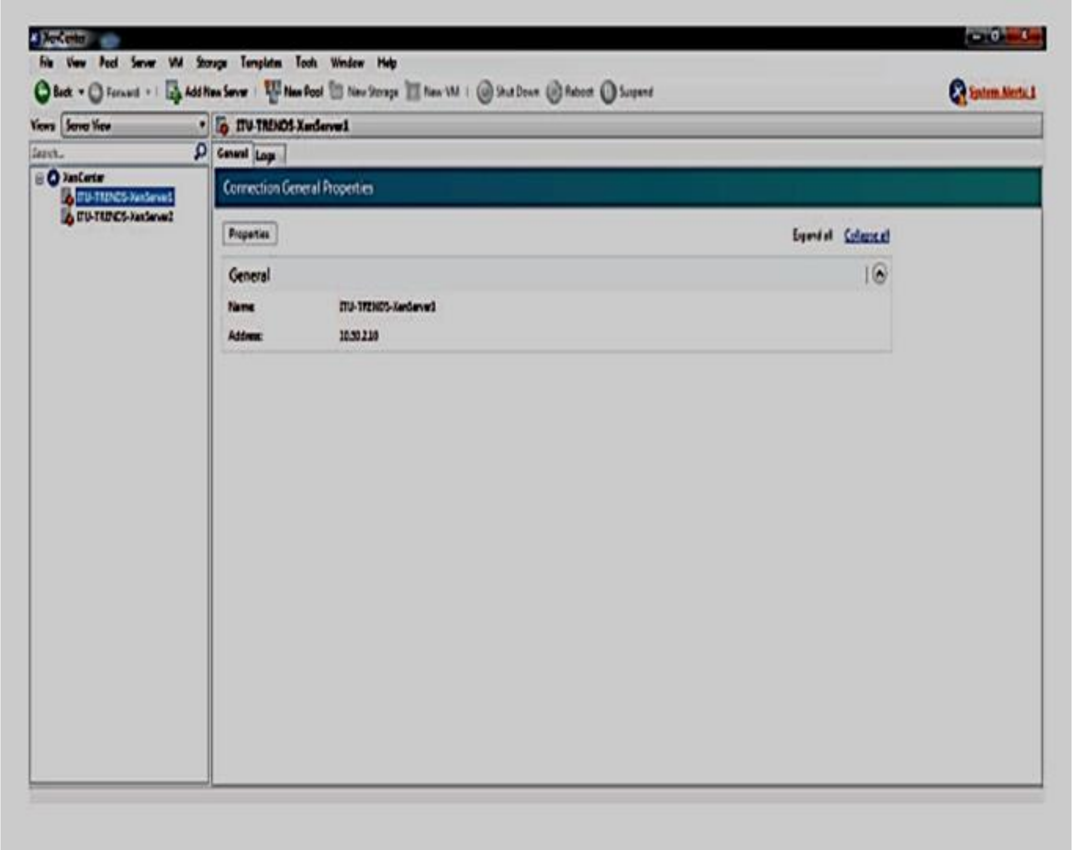
```
XenServer 6.2                     12:15:49                 ITU-TRENDS-XenServer
                               Configuration

 Customize System                       Dell Inc.
                                        OptiPlex 390
 Status Display
 Network and Management Interface       XenServer 6.2.0-70446c
 Authentication
 Virtual Machines                       Management Network Parameters
 Disks and Storage Repositories
 Resource Pool Configuration            Device          eth0
 Hardware and BIOS Information          IP address      10.50.2.20
 Keyboard and Timezone                  Netmask         255.255.254.0
 Remote Service Configuration           Gateway         10.50.2.1
 Backup, Restore and Update
 Technical Support                      Press <Enter> to display the SSL key
 Reboot or Shutdown                     fingerprints for this host
 Local Command Shell


 <Enter> OK <Up/Down> Select            <Enter> Fingerprints <F5> Refresh
```

Figure 4: XenServer

After booting up, the installation process starts. In contrast to other operating systems, we cannot produce custom-sized partitions in XenServer. XenServer manages the storage drive and file systems itself. It is advised to install XenServer on that machine that carries on data or at least make the backup of your data prior to installing XenServer, as XenServer installation will wipe out all the data on your storage drive.

As the first step, we will require to accept the end-user license agreement and click on the clean installation. If you have multiple local hard disks, choose a primary disk for the installation. In our case, we were only having one hard drive and a similar drive will be used for storing VMs. After that, select the appropriate installation source, we will be using local media for this purpose. After this, we will be prompted to enter the root password. This password will be used for all operation that requires admin privileges and to connect to the server using XenCenter, at a later stage. After this step, the network configuration stage comes. If you have multiple NICs, select one which you wished to be used for management purposes. In our case, we were having only 1NIC (PIF). We can configure the network using either Automatic configuration (DHCP) or static configuration. After this, we will be setting up the hostname for the server configuration. This name and UUID should be unique for all the servers. At the end select the appropriate time zone and start the installation process. This will take some time, as the installation process completes removing the installation medium and restarting the system [36]. For a detailed installation guide refer to [36]. In Figure 5 XenServer screenshot has been shown.

In this section, we will be downloading and installing the XenCenter. We can download it by writing the XenServer host IP address in the browser URL address bar of the machine where XenCenter has to be installed. This is a windows based utility, which will be used to interact with the XenServer host [36]. This tool provides us with centralized control, where with just one machine we can interact with all of our XenServer hosts. Installation is pretty simple without any difficult step which needs any explanation. After installation, we will add and connect to our XenServer hosts using this tool. We will be prompted to enter the root password which we gave during host installation. The below fig shows the screenshot of the XenCenter.



Name: ITU-TRENDS-XenServer1
Address: 10.20.2.10

Figure 5: XenCenter

Up to this point, we have successfully installed and configured the XenServer. Now the next important step is to create virtual machines. After connecting to XenServer from XenCenter, we can create and control VMs on that server. There is a different option, we can use for creating VMs on XenServer. XenServer is shipped with several VM templates; we can use any template we are interested in. These templates are available for many Windows, Linux, and UNIX flavors. We can use pre-configured templates or install from a CD/DVD or locally stored ISO images [37]. We can also use other options like physical-to-virtual conversion (P2V), cloning an existing virtual machine, or importing an exported VM [37].

In P2V, an operating system installed on a physical machine is converted into a virtual machine with all of its configurations and files. After conversion, it is copied to the XenServer repository. In closing, we can create several copies of the existing VM on a very similar server. In XenCenter, we can export the VM to the machine where a center is running and can be imported is any other server [37].

Xen Tools:

In our experimental setup, we have created VMs using ISO images and cloning. After the creation of VM, we need to install Xen Tools on them. An important point is worth mentioning here we should keep in mind while creating VMs. Xen instruments are not accessible for all operating systems. If the guest operating system isn't upheld by XenServer and Xen devices, we might have the capacity to make VM, however, won't have the capacity to utilize it properly. Since VMs used Xen tools to fill in as drivers so that XenServer and Virtual Machine can connect with one another. A rundown of a supported operating system can be found at [37]. We can configure VMs with much additional functionality, but in our case, this is the only thing we will need.

Virtual Network Configuration

After successful installation of Xen tools, if we go to the network tab on XenCenter we can notice that a network connection (VIF) has been added to the VM. But if we want to configure VLAN properly with all security, QoS, and different features, we have to add VIF and appoint them to the VMs. We can create a new VLAN network connection using the Command Line Interface (CLI) of the XenServer. Login into the shell and using commands we will create the VLAN connections. With these connections, we can likewise allocate VLAN tags. Forgiving security these tags are very useful, as VMs allotted with tag VLAN association mean connection can just talk with those machines having comparable tag connection, otherwise VM will not be discoverable to requesting VM. The commands for creating a tagged VLAN connection can be found at [41]. After creating network connections, we can assign those connections to VMs. This can be done using both CLI and XenCenter interfaces. We can also set a QoS rate there if we wish. We can also assign multiple VIFs, we will use the term VIF for the connections associated with the VMs, to individual VMs. We can assign the IP address and other information either using DHCP or static addresses. But for successful communication, subnet addresses should be the same.

Hyper-V

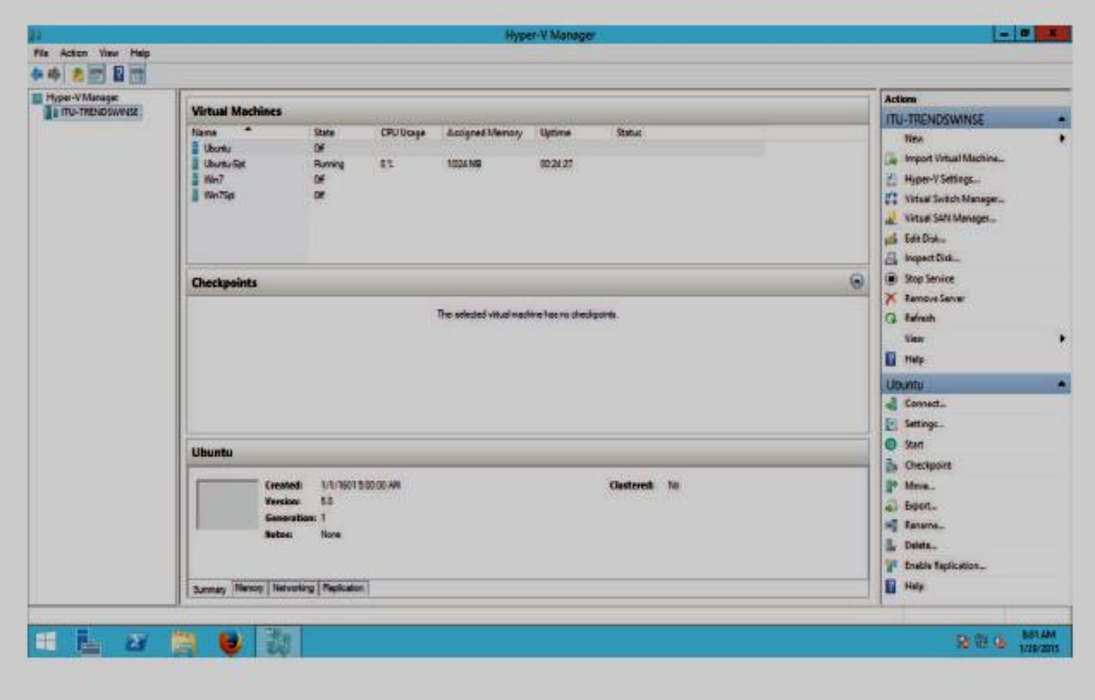
In this section, we will be describing the experimental setup created for Hyper-V hypervisor hosts. We have configured Hyper-V on top of Windows Server 2012.

Windows Server 2012 Installation

The first step towards Hyper-V installation is to install the Microsoft Windows Server 2012. We downloaded the evaluation copy of Windows Server 2012 and installed it on the machine with similar configurations as used for XenServer. For installation purposes, we made a USB disk bootable using the downloaded ISO image file. The installation for the windows server is pretty simple just like any other Microsoft Windows. Once the installation process of the windows server completes, we logged in to our system. The next step is to continue the Hyper-V on the windows server.

Hyper-V Installation

Hyper-V will be added to the windows server as a server role. Using the server manager of the windows server, we clicked on add new role to add Hyper-V role. Before you, being page, verified our destination server and network environment which are prepared for the role and feature we are perusing to install and clicked next. On the next screen we choose the installation type, we can choose Role-based or feature-based installation and moved to the next screen of the wizard. On the choose destination server page, we selected the very same server on which installation is being conducted. Selected Hyper-V as the role on the next screen and proceeded further with the installation. To add the tools that will be used to create and control virtual machines, we clicked on add features option. On the create virtual switch page, virtual machine migration page, and default stores page we selected the appropriate options and proceeded further with the installation.



Name	State	CPU Usage	Assigned Memory	Uptime	Status
Ubuntu	Off				
Ubuntu-Sjt	Running	0 %	1024 MB	00:24:27	
Win7	Off				
Win7Sjt	Off				

The selected virtual machine has no checkpoints.

Ubuntu

Created: 1/1/1601 5:00:00 AM
Version: 5.0
Generation: 1
Notes: None
Clustered: No

Figure 6: Hyper-V Manager

Once done with all steps we finished the installation and restarted the destination server. After the system is booted again after the restart we can verify the installation by opening the all servers page in server manager of Windows Server by selecting a server on which we have installed Hyper-V, and viewing the roles and features tile on the page for the selected server.

Client Hyper-V

To enable the Client Hyper-V we went to the program section in Control Panel or Windows Server enabled the Hyper-V from there. We may need to restart the system and we are done with the installation process. Figure 7 shows the screenshot of client Hyper-V. We can also call Client Hyper-V as Hyper-V manager.

Hyper-V Virtual Machine

To create VMs on Hyper-V, we opened the Hyper-V Manager and selected the computer from the navigation pane. After opening the New Virtual Machine wizard specified the name and location of the VM. On the networking configuration page, we connected the virtual machine to the switch we created during the installation of Hyper-V. On the Connect Virtual Hard Disk and Installation Options page, we select an image file (an . ISO file) select to create the new virtual hard disk. At the end of the summary page, verified our selections and then click finished to create the VM.

Virtual Network Configuration

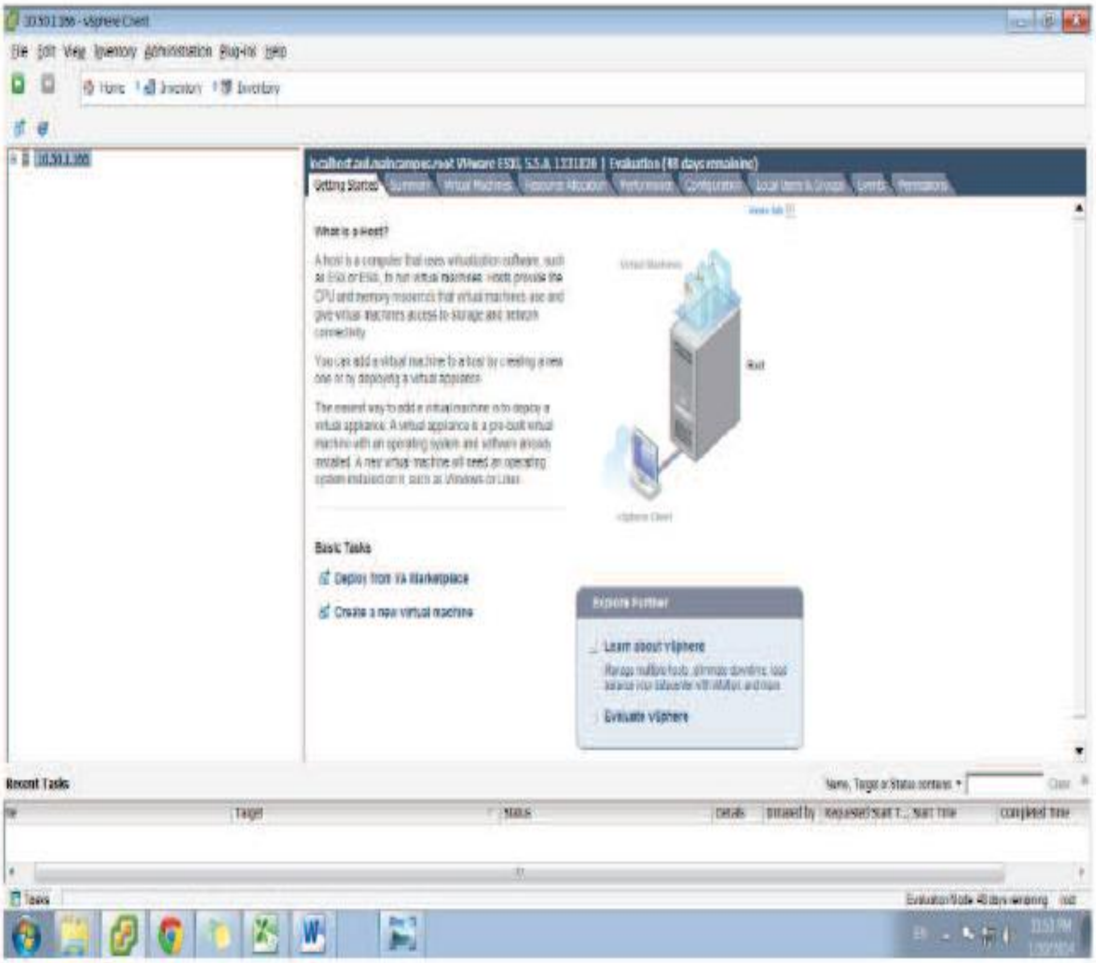
Hyper-V adds a virtual switch installation and connects to its PIF automatically. We also created other virtual connections later using Hyper-V manager for our experiments which will be discussed in the next section and assigned these connections to the VMs accordingly. We can make the virtual connection and can tag them VLAN using Hyper-V manager. VLAN Id behaves similarly to the VLAN tag in XenServer.

5. Discussion

In this section, we will be discussing the experimental setup for VMware vSphereEsxi installation and configuration.

vSphere Installation

vSphere is freely available for non-commercial use on VMware online link for download. But the process for downloading the vSphere is similar to their purchasing procedure. We registered on their site and requested the vSphere download. We get the response from them in about a week and then proceeded with downloading the ISO file using our registered account. Before anyone proceeds with ESXi installation, it is worth mentioning that vSphere installation required a minimum of one supported network adapter.



What is a Host?

A host is a computer that uses virtualization software, such as ESX or ESXi, to run virtual machines. Hosts provide the CPU and memory resources that virtual machines use and give virtual machines access to storage and network connectivity.

You can add a virtual machine to a host by creating a new one or by deploying a virtual appliance.

The easiest way to add a virtual machine is to deploy a virtual appliance. A virtual appliance is a pre-built virtual machine with an operating system and software already installed. A new virtual machine will need an operating system installed on it, such as Windows or Linux.

Figure 7: vSphere Client

If the system doesn't have supported hardware installation cannot be preceded. A compatibility guide can be consulted on their site before proceeding further. We have used the 5.0 version of vSphereEsxi and vSphere client. After creating the bootable USB using the download ISO image file, we proceeded with the installation. Installation is pretty much simple just like any other Linux installation and similar to Xen-Server installation. Once the installation completes, we can use the CLI interface on host machines where vSphere is installed or we can use the vSphere client to interact with the servers. We opted for the vSphere client.

VMware vSphere Client

We downloaded the vSphere Client again using the very same link we used for downloading the vSphere. vSphere Client is a windows based application and we installed it on another system with similar specifications as per host machines. After installation connects it to the host using the IP address and root password for further interacting with the server. Figure 8 shows the screenshot of the vSphere Client.

vSphere Virtual Machine

We created the new VM using the vSphere client. Entered name and location for the new VM and later selected the host where we want to run our VM. We can also select cluster instead of the host in the case of data centers. Assign resource pool to the VM in the next screen of the wizard. After creating the VM we started the VM execution using the same vSphere client and proceeded with the installation of guest operating systems on the VM. Once installation of guest OS completes, we need to install VMware Tools just like XenTools on the VM.

Virtual Network Configuration

In vSphere virtual network is managed by vSphere standard switch or vSphere distributed switch, we have opted for the standard switch. By default virtual connection is created for each physical switch. We created other virtual connections with VLAN Id also for our experimental purpose and assigned them to our VMs as we proceeded with our experiments.

Experimental Configuration

After performing all the actions discussed earlier in this section, we are done with setting up our experimental environment, installation, and configurations. In this section, we will be summarizing the experimental configuration, which we will be using for our experiments and evaluations in the coming section.

Table 1: Hypervisors Configuration (Hardware and Software

Component	Hardware Configuration	Software Configuration
XenServer	Intel Core i3 4 GB RAM 500 GB Hard Drive	Citrix XenServer 6.2 Open Switch
Hyper-V Server	Intel Core i3 4 GB RAM 500 GB Hard Drive	Windows Server 2012 Hyper-V
vSphere Server	Intel Core i3 4 GB RAM 500 GB Hard	VMware vSphereEsxi 5.0
Virtual Machines (Windows)	2 Intel i3 Cores 2 GB RAM 100 GB Hard Drive	Windows 7 (64 bit)
Virtual Machines (Linux)	2 Intel i3 Cores 1 GB RAM 100 GB Hard Drive	Ubuntu 12.04 (64 bit)
Remote Host	Intel Core i3 4 GB RAM 500 GB Hard Drive	Windows 7 XenCenter vSphere Center

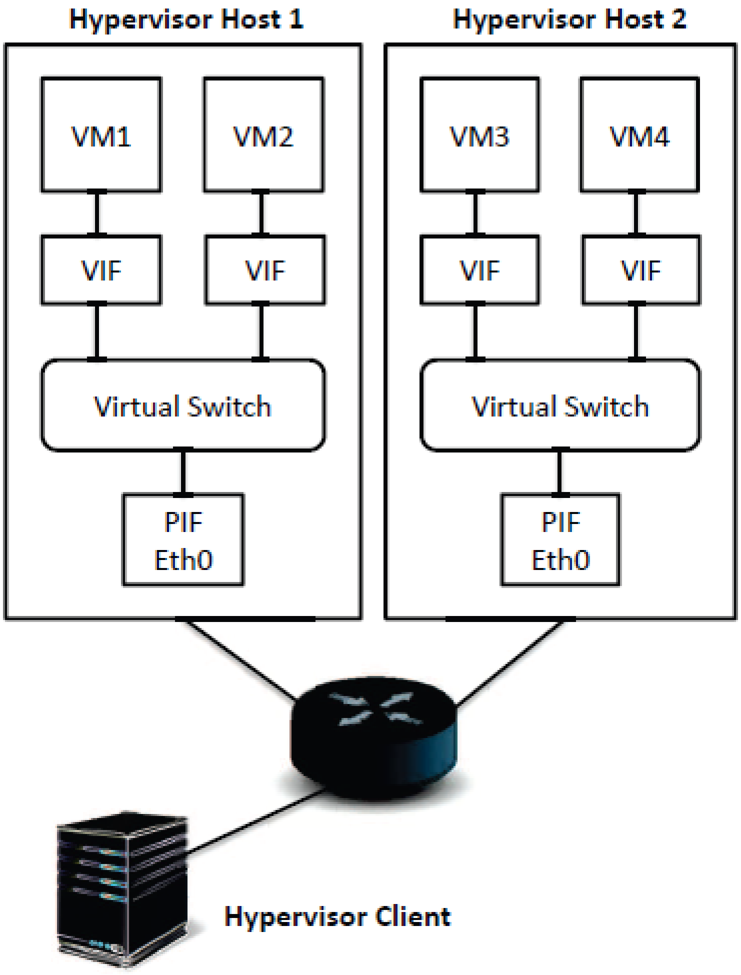


Figure 8: Experimental Setup

Table 2: Hypervisor Network Configuration

	Hypervisor Host 1	**Hypervisor Host 2**
IP Address	10.50.2.10	10.50.2.20
Default Gateway	10.50.2.1	10.50.2.1
Subnet mask	255.255.254.0	255.255.254.0
Default Time zone	Asia/Karachi	Asia/Karachi

Table 3: Virtual Machines Configurations

	VM-1	VM-2	VM-3	VM-4
OS	Ubuntu	Windows	Ubuntu	Windows
Hypervisor	Hypervisor Host 1	Hypervisor Host 1	Hypervisor Host 2	Hypervisor Host 2
IP Address	10.50.2.11	10.50.2.12	10.50.2.21	10.50.2.22
Subnet Mask	255.255.254.0	255.255.254.0	255.255.254.0	255.255.254.0
Default Gateway	10.50.2.1	10.50.2.1	10.50.2.1	10.50.2.1
Default Time zone	Asia/Karachi	Asia/Krachi	Asia/Karachi	Asia/Karachi

We have followed similar configurations and hardware for all of our three hypervisors under consideration. We have configured hypervisors on two different machines and treated them as host machines. The summary of hardware and software configuration is shown in table 5.1 for all hypervisors host machines and remote machines where the client is installed for XenServer and vSphere. In table 5.2, we have shown the summary of a network configuration for host servers. On every server, we have made two Linux, three VMs on which Ubuntu 12.04 (64 bit) was installed as a guest operating system, and on one VM Microsoft Windows 7 (64 bit) is installed as a guest operating system. Due to limited available RAM, we ran 2 VMs at max at a time. The configuration information for these VMs is shown in table 5.3. In figure 9 we have shown the general overview view of the network virtualization followed in our experimental setup.

6. Conclusions

Here, we will conclude and summarize our work. Based on our experiments and evaluations we can conclude that by using network virtualization in cloud computing hypervisors we can achieve better and enhanced network scalability and security of the network. One or numerous VIF can be assigned to a single Virtual Machine whose limit depends on the underlying hypervisor. We can ensure security and can isolate network traffic using VLAN tags or Ids. If VIF is tagged then only those VM's can communicate with each other on a virtual network having identical VLAN tags. Generally, communication is probable, ensuring security and Isolation. We also saw that Hyper-V provides the most advanced features regarding VLAN tags for network isolation. The throughput of the network that we can achieve for Transmission Control Protocol or Universal Datagram Protocol are approximately identical or with very slight distinction for VM with VLAN tag or VM without VLAN tag, yet the security advantage is the additionally preferred standpoint that we accomplish utilizing VLAN tags while not influencing the performance of the network. We can also apply quality of service checks on virtual network connections such as bandwidth rate and QoS rate. The Round-Trip Time result in a non-virtual network is marginally lower than those in a virtual network; this is due to the hypervisor layer which acts as a channel between PIFs and VIFs. Similar is the situation with CPU allocation, where numerous Virtual Machines are having a single processor as in examination with a non-virtual network where no CPU distribution is really done. This also includes latency in procedure arrangement (scheduling) while it is nearly equal to nil in the non-virtual network.

Table 4: Summary

	Best		Normal		Worst	
	Hyper - Visor	Value	Hyper – Visor	Value	Hyper - Visor	Value
Transmission Control Protocol Throughput using no VLAN tag	vSphere	11.31 MB/s	XenServer	11.30 MB/s	Hyper-V	10.69 MB/s
Transmission Control Protocol Throughput using the VLAN tag	vSphere	11.28 MB/s	XenServer	11.27 MB/s	Hyper-V	9.46 MB/s
User Datagram Protocol Throughput using no VLAN tag	XenServer	29.54 MB/s	vSphere	26.67 MB/s	Hyper-V	5.66 MB/s
User Datagram Protocol Throughput using the VLAN tag	XenServer	29.53 MB /s	vSphere	27.97 MB/s	Hyper-V	5.64 MB/s
Transmission Control Protocol CPU Usage no VLAN tag	Hyper-V	2.49%	XenServer	5.60%	vSphere	6.44%
Transmission Control Protocol CPU Usage using the VLAN tag	Hyper-V	2.14%	XenServer	3.32 %	vSphere	6.33%
User Datagram Protocol CPU Usage no VLAN tag	Hyper-V	18.08 %	vSphere	37.78%	XenServer	57.41%
User Datagram Protocol CPU Usage VLAN tag	Hyper-V	18.51%	vSphere	38.08%	XenServer	57.13%
Round Trip Time between VMs on Same Servers	vSphere	0.30ms	Hyper-V	0.31ms	XenServer	0.38ms
Round Trip Time between VMs on Different Servers	vSphere	0.40ms	XenServer	0.64ms	Hyper-V	1.03ms
CPU Sharing Among one Virtual Machine with 1 User Process	vSphere, XenServer	2 µs	vSphere, XenServer	2 µs	Hyper-V	4 µs
CPU Sharing Among two Virtual Machine with 1 User Process	vSphere, XenServer	3 µs	vSphere, XenServer	3 µs	Hyper-V	4 µs

Table 4, summarizes the overall experimental evaluations showing which hypervisor performed well in which case. In the case of Transmission Control Protocol throughput using no VLAN tag and using VLAN tag, vSphere outperforms the rest of the hypervisors and depicts the best performance with an average of 11.31 MB/s and 11.28 MB/s respectively whereas Hyper-V depicts the worst performance with an average of 10.69 MB/s and 9.46 MB/s respectively. XenServer shows a very close performance to vSphere with a difference of 0.01 MB/s for both cases. Essentially if there should arise an occurrence of UDP throughput using no VLAN tag and using VLAN tag, similar to TCP, UDP also depicts worst performance with Hyper-V hypervisors, whereas XenServer is relatively close to vSphere but yet outperforms the vSphere. In the case of TCP UDP usage with no VLAN tag and with VLAN tag Hyper-V shows the best results with an average CPU utilization of 2.49 % and 2.14 % respectively, whereas vSphere depicts the worst performance. For both cases, with and without VLAN tag XenServer shows in between performance. Just like TCP UDP usage, UDP CPU usage was best observed in Hyper-V again with an average CPU utilization of 18.08% and 18.51% respectively, whereas vSphere utilized double of Hyper-V. In the case of Round Trip Time when measured between VMs on the same server and on the different server, in both cases, vSphere outperforms all other hypervisors with an average RTT of 0.30 ms and 0.40 ms respectively. However, in the case of the same server, the difference between vSphere and Hyper-V is negligible at 0.01 ms only whereas XenServer took the most time of 0.38 ms. In the second case, Hyper-V performed worst with an average RTT of 1.03 ms. CPU virtualization and sharing them among hosted VMs on hypervisors is observed well in XenServer and vSphere for both cases. Both of these hypervisors performed equivalently outperforming Hyper-V.

On basis of the above discussion, we can deduce that VMware vSphere could be considered as the best hypervisor, based on our experiments and evaluation, being followed by Citrix XenServer. In some cases, the vSphere performed better and in some cases, XenServer performed better than vSphere but the difference between vSphere and XenServer results are not that great and can be ignored. On this very base, we can also say that both of these hypervisors are almost equivalent in performance and can be used alternatively. In this way, utilizing Virtual switches and hypervisors we discover the virtualization impact on CPU (processor) performance and network performance. For a large portion of the performance parameters, the impact is unimportant as contrasted with non-virtualized environments. For setting up the virtualized environment the cost might increase however we get a lot of benefits as compared to the non-virtual environment. In this manner, this work inspires for building up cloud-based virtualized server farms.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. L. M. Vaquero, L. Rodero-Merino, J. Caceres, and M. Lindner. A break in the clouds: towards a cloud definition. ACM SIGCOMM Computer Communication Review, 39(1):50–55, 2008.
2. L. Popa, M. Yu, S. Y. Ko, S. Ratnasamy, and I. Stoica. Cloud police: taking access control out of the network. In Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks, page 7. ACM, 2010.
3. Cloud Computing. http://en.wikipedia.org/wiki/Cloud_computing, 2014. [Online; accessed 4-July-2014].
4. S. Angeles. Virtualization vs. Cloud Computing: What'stheDi_erence? http://www.businessnewsdaily.com/5791-virtualization-vs-cloud-computing.html, 2014. [Online; accessed 07-July-2014].
5. Z. He and G. Liang. Research and evaluation of network virtualization in the cloud computing environment. In Networking and Distributed Computing (ICNDC), 2012 Third International Conference on, pages 40{44.IEEE, 2012.
6. I. I. Center. Virtualization and Cloud Computing. http://www.intel.com/content/dam/www/public/us/en/documents/guides/cloud-computing-virtualization-building-private-iaas-guide.pdf, 2013. [Online; accessed 26-June-2014].
7. Y. Pu, Y. Deng, and A. Nakao. Cloud rack: Enhanced virtual topology migration approach with open vswitch. In Information
8. B. Pfaff, J. Pettit, K. Amidon, M. Casado, T. Koponen, and S. Shenker.Extending networking into the virtualization layer. In Hotnets, 2009.
9. J. Pettit, J. Gross, B. Pfaff, M. Casado, and S. Crosby. Virtual switching in an era of advanced edges. In 2nd Workshop on Data Center–Converged and Virtual Ethernet Switching (DC-CAVES), 2010.
10. N. Chowdhury and R. Boutaba. A survey of network virtualization. Computer Networks: The International Journal of Computer and Telecommunications Networking, 54(5):862{876, 2010.
11. M. Casado, T. Koponen, D. Moon, and S. Shenker.Rethinking packet forwarding hardware.

12. H.-M. Tseng, H.-L.Lee, J.-W.Hu, T.-L.Liu, J.-G.Chang, and W.-C. Huang.Network virtualization with cloud virtual switch. In Parallel and Distributed Systems (ICPADS), 2011 IEEE 17th International Conference on, pages 998–1003. IEEE, 2011.
13. Y. Dong, X. Yang, J. Li, G. Liao, K. Tian, and H. Guan. High performance network virtualization with sr-iov. Journal of Parallel and Distributed Computing, 72(11):1471{1480, 2012.
14. J. Soares, J. Carapinha, M. Melo, R. Monteiro, and S. Sargento. Resource allocation in the network operator's cloud: A virtualization approach. In Computers and Communications (ISCC), 2012 IEEE Symposium on, pages 000800–000805.IEEE, 2012.
15. J. Lee, J. Tourrilhes, P. Sharma, and S. Banerjee. No more middlebox: integrate processing into network. ACM SIGCOMM Computer Communication Review, 40(4):459–460, 2010.
16. Q. Duan. Modeling and performance analysis on network virtualization for composite network-cloud service provisioning. In Services (SERVICES), 2011 IEEE World Congress on, pages 548{555. IEEE, 2011.
17. Q. Duan, Y. Yan, and A. V. Vasilakos. A survey on service-oriented network virtualization toward convergence of networking and cloud computing. Network and Service Management, IEEE Transactions on, 9(4):373{392, 2012.
18. D. Kakadia and V. Varma. Network virtualization platform for hybrid cloud. In Cloud Computing Technology and Science (CloudCom), 2013 IEEE 5th International Conference on, volume 2, pages 69–74.IEEE, 2013.
19. R. Yu, X. Yang, J. Huang, Q. Duan, Y. Ma, and Y. Tanaka.Qosaware service selection in virtualization-based cloud computing. In Network Operations and Management Symposium (APNOMS), 2012 14th Asia-Pacific, pages 1–8. IEEE, 2012.
20. G. Wang and T. E. Ng.The impact of virtualization on network performance of amazon ec2 data center. In INFOCOM, 2010 Proceedings IEEE, pages 1–9. IEEE, 2010.
21. J. Hwang, S. Zeng, F.Wu, and T.Wood.A component-based performance comparison of four hypervisors. In Integrated Network Management (IM 2013), 2013 IFIP/IEEE International Symposium on,pages 269{276. IEEE, 2013.
22. A. Moniruzzaman. Analysis of memory ballooning technique for dynamic memory management of virtual machines (vms).arXiv preprint arXiv:1411.7344, 2014.
23. Introducing XenServer.http://docs.vmd.citrix.com/XenServer/4.0.1/installation/ch01s03.html. [Online; Available].
24. Xen tools. http://wiki.xen.org/wiki/Xen_tools. [Online; Available].
25. XenCenter Development. http://www.xenserver.org/partners/developing-products-for-xenserver/21-xencenter-development/88-xc-dev-home.html. [Online; Available].
26. XenServer 6.2.0 Technical FAQ. http://support.citrix.com/article/CTX137836, 2014. [Online; Available].
27. Overview of Hyper-V. https://technet.microsoft.com/en-us/library/cc816638(WS.10).aspx, 2014. [Online; accessed 29-June-2014].
28. M. Nishikiori. Server virtualization with VMware vsphere 4.Fujitsu Scientific and Technical Journal, pages 356–361, 2011.
29. Iperf. http://en.wikipedia.org/wiki/Iperf. [Online; Available].
30. What is Iperf? https://iperf.fr/. [Online; Available].
31. More tools for easy network testing: NTttcp and NetCPS. http://www.techrepublic.com/blog/data-center/ more-tools-for-easy-network-testing-ntttcp-and-netcps/.[Online; Available].
32. NTttcp Version 5.28. https://gallery.technet.microsoft.com/NTttcp-Version-528-Now-f8b12769.[Online; Available].
33. Ping (networking utility). http://en.wikipedia.org/wiki/Ping%28networking_utility%29. [Online; Available].
34. What is Wireshark. http://www.wireshark.org/docs/wsug_html_chunked/ChapterIntroduction.html#ChIntroWhatIs. [Online; Available].65 BIBLIOGRAPHY 66
35. Wireshark Frequently Asked Questions. http://www.wireshark.org/faq.html#q1.1. [Online; Available].
36. C. S. Inc. Citrix XenServer 6.2.0 Installation Guide. http://support.citrix.com/servlet/KbServlet/download/ 34970-102-704220/installation.pdf, 2013.[Online; accessed 22-June-2014].
37. Citrix XenServer 6.0 Virtual Machine Installation Guide.http://docs.vmd.citrix.com/XenServer/6.0.0/1.0/en_gb/guest.html, 2014. [Online; accessed 29-June-2014].
38. Virtual Machine? http://en.wikipedia.org/wiki/Virtualmachine. [Online; Available].
39. What Is a Virtual Machine? http://pubs.vmware.com/vsphere-50/index.jsp#com.vmware.vsphere.vm_admin.doc_50/GUID-CEFF6D89-8C19-4143-8C26-4B6D6734D2CB.html. [On-line; Available].
40. KVM Virtualized I/O Performance. whitepaperIBM&Redhat, 2013.
41. Networking. http://docs.vmd.citrix.com/XenServer/4.0.1/reference/ch03.html, 2014. [Online; accessed 29-June-2014].
42. XenServer Download. http://xenserver.org/overview-xenserver-open-source-virtualization/download.html, 2014. [Online; Available].