Tommi Hokkanen

Evaluation of Virtual Firewall in Private Cloud

Helsinki Metropolia University of Applied Sciences

Master’s Degree

Information Technology

Master’s Thesis

17 December 2016

Author(s) Title Number of Pages Date	Tommi Hokkanen Evaluation of Virtual Firewall in Private Cloud 73 pages + 2 appendices 29 November 2016
Degree	Master of Engineering
Degree Programme	Information Technology
Instructor(s)	Antti Koivumäki, Principal Lecturer Pasi Ulkuniemi, Director Corporate Service Delivery and Operations

The Master's Thesis study evaluates the deployment and performance of a virtual firewall appliance in a private cloud setting. The objective was to study virtual firewall technology in the perspective of a network service provider. The scope of the study was limited to virtual firewall appliance deployment and network throughput benchmarking with various network configurations. Other aspects of virtual firewalls, such as security functions, are not discussed in detail. The benchmarks contained the following virtual firewall use-cases: stateful firewall throughput, UTM services throughput, IPSec VPN throughput and SSL VPN throughput.

As the background for the study, key concepts of virtualization, network virtualization technologies and network performance testing methods were researched. Also, the current state of the network service provider's firewall product portfolio was analyzed. Based on this background work, the virtual firewall under test was chosen to be Fortinet's Fortigate-VM. Then, methodology for virtual firewall performance testing was created. The performance measurements were carried out in a laboratory network purpose-built for the case. The laboratory network consisted of two server machines and a switch. A Microsoft Hyper-V hypervisor was installed on one of the servers and virtual network was created. Virtual network consisted of virtual firewall appliance, virtual switches and Linux virtual machines.

The results showed that virtual firewall throughput performance was close to that of the same vendor's (Fortinet) low-end physical appliance firewalls. In addition, it was found that virtual firewall deployment and configuration was practical and straightforward. No major issues were encountered in any part of the deployment. The conclusion was drawn that virtual firewall network performance is feasible and the technology is ready for production use. The developed test method will be replicated in other deployment scenarios in the future. In the next phase of the virtual firewall productization, the performance of a virtual firewall appliance deployed in Microsoft Azure and Amazon Web Services public clouds will be evaluated using the developed method.

Keywords	Throughput, Virtualization, Virtual Firewall

Contents

Appendices

List of Tables

List of Figures

List of Abbreviations

ARP Address resolution protocol. Protocol for mapping IP addresses to physical MAC addresses.

BER Bit-error-rate. Number of bit errors in data transfer per time unit.

BOOTP Bootstrap protocol. Protocol for automatic network configuration.

CFI Canonical format indicator. Describes MAC address notation.

CIR Committed information rate. Bandwidth for a virtual circuit guaranteed by the service provider.

CLI Command-line interface. An interface for issuing commands to computer programs in text format.

CPE Customer premises equipment. Device connected to service provider’s network located in customer premises.

CPU Central processing unit. Electronic circuitry that performs computing operations.

DAS Direct attached storage. Digital storage directly attached to the accessing device.

DHCP Dynamic host configuration protocol. Protocol for distributing network configuration parameters.

DMZ Demilitarized zone. Network segment for hosting services open to the Internet.

DUT Device under test. Refers to device undergoing functional testing.

EIR Excess information rate. Magnitude of burst above committed information rate.

FDV Frame delay variation. Difference of frame delay between packets. Also known as jitter.

FLR Frame loss ratio. Amount of lost frames in a data transfer.

FTP Frame transfer protocol. Protocol for transferring files between computers.

GUI Graphical user interface. An interface for issuing commands to computer programs by the use of graphical elements.

HTTP Hypertext transfer protocol. Protocol for transferring data in between WWW-servers and browsers.

IAAS Infrastructure as a Service. Form of cloud computing providing scalable resources over the Internet.

IP Internet protocol. Part of TCP/IP protocol suite used for delivering datagrams across network boundaries.

IPSEC Internet protocol security. Protocol suite for securing IP packet delivery by encryption and authentication methods.

LAN Local area network. Network covering small local area, such as home or office.

MPLS Multiple protocol label switching. Data transfer technique that utilizes labels instead of network addresses for packet redirection.

NAS Network attached storage. Method for serving files over network with a server that is directly connected to network.

NAT Network address translation. Method of mapping IP address space to another.

NIC Network interface controller. Computer component for connecting the computer to network.

NSP Network service provider. Business organization providing network access services to customers.

OS Operating system. Computer system software that manages access to hardware and software resources.

PAAS Platform as a Service. Category of cloud computing where application platform is hosted by a service provider.

RFC Request for comments. Memos developed by Internet Engineering Taskforce containing technical and organizational notes about the Internet.

SAAS Software as a Service. Subscription based software delivery model where software is centrally hosted.

SAN Storage area network. Network that provides access to a data storage.

SCSI Small computer system interface. Standard for transferring data between computer and peripheral devices, such as hard drives.

SCTP Stream control transmission protocol. Protocol for transporting multiple data stream simultaneously.

SGW Security gateway. Refers of NGFW product service provided by network service provider who is a stakeholder in this thesis.

SLA Service level agreement. Agreement of service quality between service provider and customer.

SSL Secure sockets layer. Cryptographic protocol for providing communications security over networks. Newer version of the protocol is known as transport layer security (TLS).

TCP Transmission control protocol. Part of TCP/IP protocol suite that provides reliable, ordered and error-checked data stream delivery in IP networks.

UDP User datagram protocol. Connectionless protocol for transmitting data streams in IP networks.

VID VLAN identifier. Numerical value referring to virtual local area network instance.

VLAN Virtual local area network. Technology for dividing physical network to several logical LAN segments.

VM Virtual machine. Emulation of computer system created by files.

VMM Virtual machine monitor. Computer software, firmware or hardware that creates and runs virtual machines. Also known as hypervisor.

VPN Virtual private networking. Method for securely connecting two or more trusted networks together over the Internet.

VRF Virtual routing and forwarding. Technology for dividing router resources to multiple routing tables.

WAN Wide area network. Network covering large geographical area.

WLAN Wireless local area network. Technology for connecting networking devices without physical cables.

1 Introduction

In modern enterprises and organizations, the network services are concentrated in facilities called “data centers”. The data center facility hosts the physical servers, network equipment and other components necessary to produce services. The data center provider (the company itself or a service provider) manages the physical aspects for the hosted equipment, such as electricity, cooling, security and access rights. Network level access to the services residing in data centers is controlled by hardware based appliances, network firewalls. Network firewall devices are placed strategically in the data center, so that incoming and outgoing network traffic must traverse the firewall before reaching its destination. This way the traffic can be inspected and controlled to mitigate security risks.

Recently it has been found out that the traditional approach to data center security has proven to be inadequate (Hossain, 2014). One important reason for this is that the use of *virtualization* technologies in on-premises data centers has reached great popularity and is now the prevalent way of producing network services (Gartner, Inc., 2016). In virtualized environments the services are hosted on *virtual machines* (VMs). Virtual machines are software equivalents of physical computers. In effect, virtual machines are collections of files that represent different parts of physical computers. In a virtualized environment a single physical computer server can host several VMs and network traffic to and between VMs is often contained to this server.

The problem of the trend in using virtualization technologies is that the traditional network firewalls never see the VM to VM traffic. This means that a large part of the enterprise traffic is uninspected and uncontrolled, unless the traffic is artificially re-routed to network firewall and back (process known as “hair-pinning”) (Hossain, 2014). This fact can open up new attack vectors for malicious intruders. Another problem is that because VM to VM traffic is not logged in the monitoring systems, the possible security incidents are not identified and attacks can continue unnoticed.

To tackle these issues, firewall manufacturers have developed *virtual firewalls* (VFs). VF is a virtual machine that has network firewall capabilities. A VF can be installed on the same physical machine as other virtual machines and can be configured so that all traffic to and between virtual machines traverses the virtual firewall. The virtual firewall approach suggests several benefits over traditional firewalls, for example:

- Faster deployment time
- Enhanced network security
- Better visibility to virtual machine traffic

This thesis was done for a Finnish network service provider (NSP). The NSP offers data center, hosting, networking and firewall services to its customers. At the moment the NSP is using physical next-generation firewalls (NGFWs) to produce its firewall services. For customers, two firewall deployment models are offered, cloud-based and CPE-based. The NSP is not using virtual firewalls in any department yet. For the NSP the virtual firewalls could provide new revenue in the form of new services provided to customers, offer savings in hardware costs and lower overall firewall maintenance and change request work time. However, the virtual firewall technology is rather new and it is unclear if the VF can be feasibly deployed to secure virtualized environments. It is also unknown if VF network performance is comparable to dedicated hardware firewall appliances. Therefore, there was a need to study virtual firewall technology and establish whether this technology is ready for production use.

The goal of this thesis is to establish if virtual firewall technology is mature enough to be ready for productization. The aim of the study is to answer the research problem:

> *Can Virtual Firewall offer similar network performance as cloud-based firewall service?*

In order to answer to this question this thesis concentrates on the following subjects:

- Virtual firewall appliance deployment in a private cloud
- Measuring the basic network performance of virtual firewall

Other aspects of this issue, such as network service deployment time, enhancement in network security and visibility or the protection of virtualization platform are out the scope of the present study. The results of the thesis will act as groundwork for virtual firewall service productization. In further phases of the productization process, virtual firewalls will be deployed in Microsoft Azure and in Amazon Web Services public cloud environments. The network performance tests are then run on these public cloud environments and the results are compared to the performance results obtained in this thesis. Next, the composition of the thesis is discussed. First, in chapter two (*Background*), the study introduces the background from the company's point of view for this case. Next, in chapter three (*Literary Review*), the theoretical background is built in the form of a literary review. The literary review discusses key technologies and methods essential for firewall virtualization and network performance testing. Then, in chapter four (*Current State Analysis of NSP Cloud Based Firewall Service*), the current state of the company's firewall service portfolio is described. The thesis continues with chapter five (*Solution Needs Analysis*), which discusses the solution design and components. Next, chapter six (*Performance Test Measurements*) describes and presents the conducted network performance measurements. The measurement results are then analyzed in chapter seven (*Measurement Analysis*). The study ends with chapter eight (*Discussion and Conclusion*), where the study itself and the obtained results are discussed and conclusions are presented.

2 Background

The NSP is offering various networking services from its data centers for customers. The services are produced mainly in two ways: either with dedicated physical appliances, or with virtual appliances. Network traffic between customers and services is segregated by the use of network virtualization technologies.

When a physical appliance is used to offer the service, its network traffic is forwarded to a data center network firewall. In this case, the traffic segmentation is performed with *Virtual Local Area Networks* (VLANs). In contrast, when virtual appliances are used to offer services, the traffic is routed between VM and data center firewall. In this case the traffic is segregated on two levels: on switch and router. On the switch level, the segregation method is *Virtual Local Area Networks* (VLANs). On the router level the segregation method is *Virtual Routing and Forwarding* (VRF) tables.

The traffic flow from VM to firewall is rather simple in case the firewall instance dedicated for the customer resides in the same data center as the virtualization host. In this case there are four hops between a VM and a firewall. In practice, this means configurations need to be done in four places: on VM, virtual switch, physical switch and on firewall. Figure 1 illustrates the traffic flow in this case.

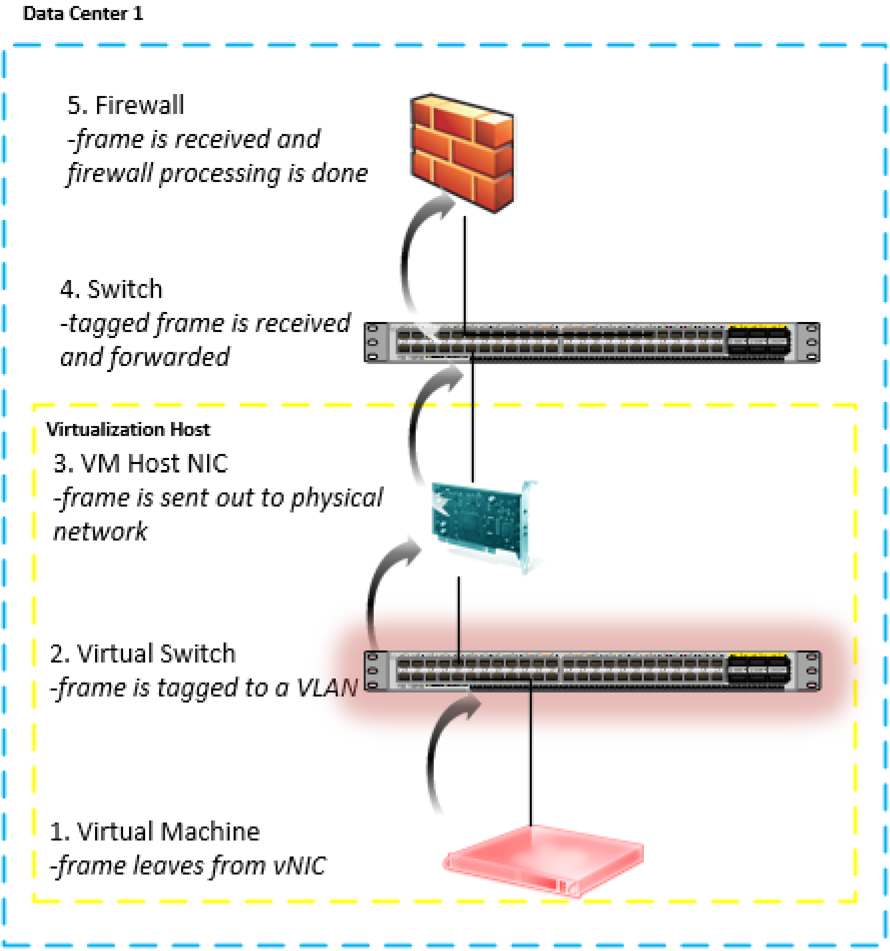

Data Center 1
5. Firewall
-frame is received and firewall processing is done
4. Switch
-tagged frame is received and forwarded
Virtualization Host
3. VM Host NIC
-frame is sent out to physical network
2. Virtual Switch
-frame is tagged to a VLAN
1. Virtual Machine
-frame leaves from vNIC

Figure 1 VM to firewall traffic flow in single data center

In many cases, however, the customer specific firewall instance resides in some other data center with the virtualization host. In that event there are eight hops between a VM and a firewall. Configurations are needed at minimum on six places: on VM, virtual switch, on both data center's routers and switches and on the firewall. The traffic flow from a VM in one data center to a firewall in another data center is illustrated in Figure 2.

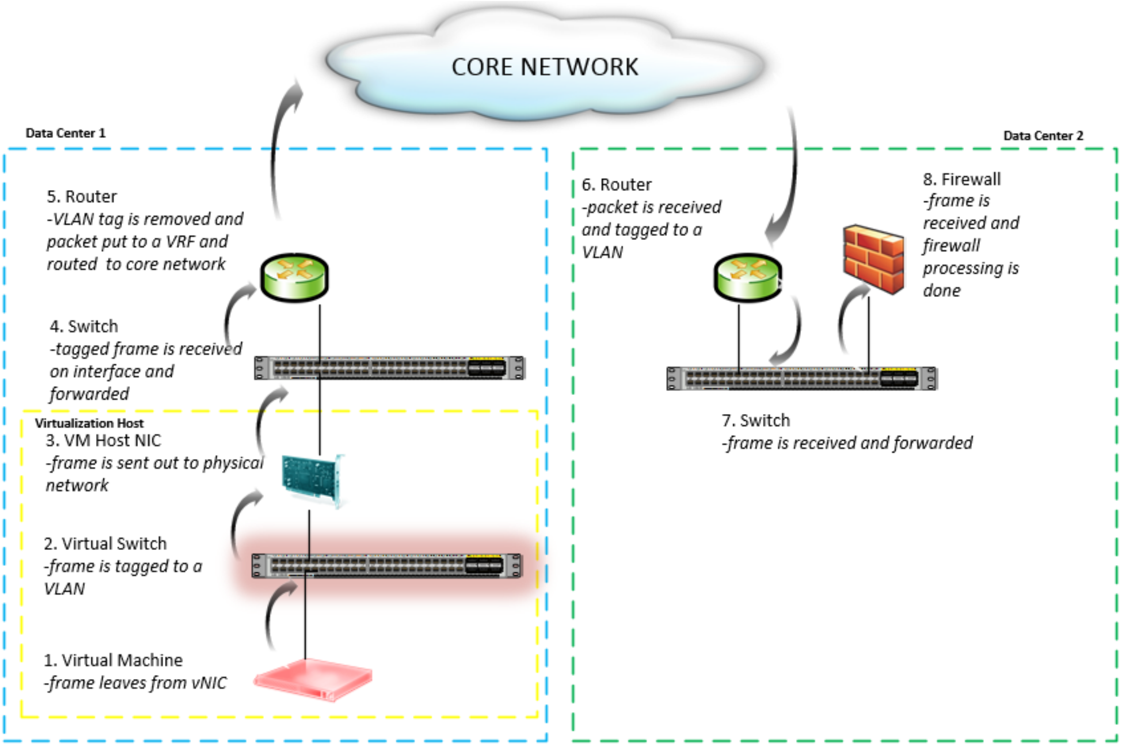

CORE NETWORK
Data Center 1
Data Center 2
5. Router
-VLAN tag is removed and packet put to a VRF and routed to core network
4. Switch
-tagged frame is received on interface and forwarded
Virtualization Host
3. VM Host NIC
-frame is sent out to physical network
2. Virtual Switch
-frame is tagged to a VLAN
1. Virtual Machine
-frame leaves from vNIC
6. Router
-packet is received and tagged to a VLAN
8. Firewall
-frame is received and firewall processing is done
7. Switch
-frame is received and forwarded

Figure 2 VM to firewall traffic flow between data centers

In practice, the networking devices (switches, routers and firewalls) in data centers are always redundant, which means configurations can be needed on up to twelve different devices. If a Virtual Firewall is used directly on the virtualization host, the traffic flow complexity and amount of configuration work could be significantly reduced. In that case, there are two hops from a VM to a Virtual Firewall and three configuration spots: VM, virtual switch and Virtual Firewall. VM to Virtual Firewall traffic flow is depicted in Figure 3.

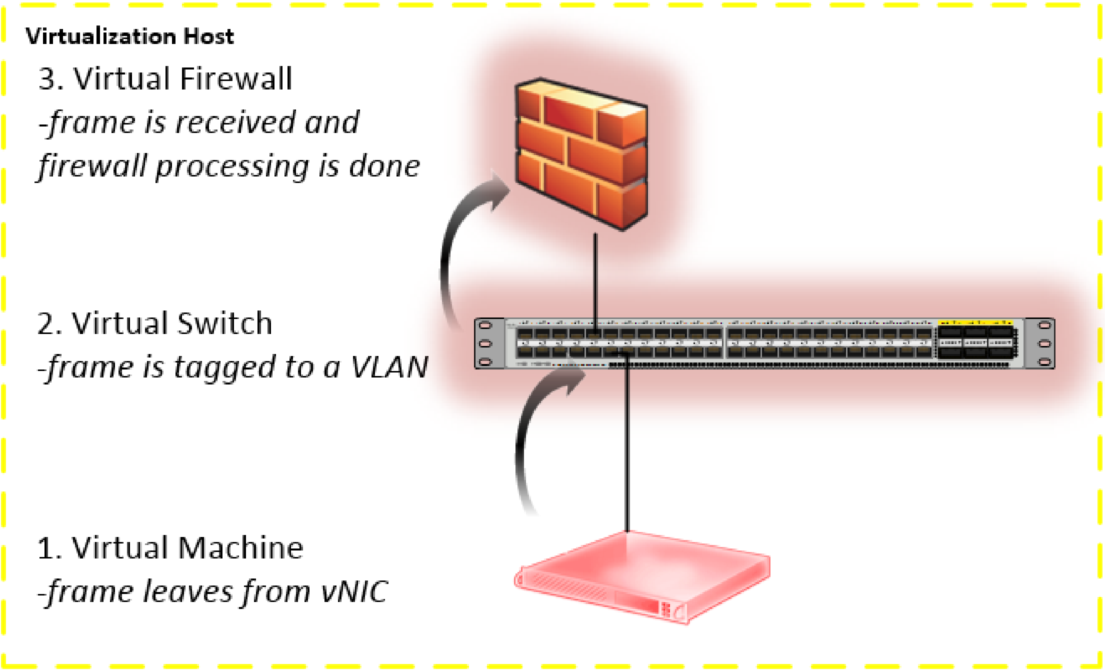


Figure 3 VM to virtual firewall traffic flow

As it seems that using Virtual Firewalls would offer important advantages over traditional network firewalls, it is interesting for the NSP to study the VF technology more closely. Good network performance is fundamental to any networking service and for that reason, performance evaluation was chosen as the paradigm for this study. The goals of this study are to:

1. Study network virtualization and network testing methods and analyse how Virtual Firewall network performance could be tested
2. Analyse current state of NSP cloud firewall services
3. Establish the metrics to be tested
4. Create a repeatable test methodology that can be used to evaluate private and public cloud deployments
5. Build a private cloud test deployment
6. Measure and evaluate the network performance of a VF

The obtained results will be utilised in the company as groundwork for virtual firewall service productization.

3 Literary Review

To build theoretical background for the study, the key concepts of virtualization, network virtualization technologies and network performance testing methods are discussed in the following sections.

3.1 Virtualization in Computing

Virtualization is a broad concept that in the context of computing generally means creating an entity that logically functions like physical resource, but actually only appears this way to the user. The term *virtualization* should not be confused with term *cloud computing* as these are different concepts (Josyula, et al., 2011, p. 3).

Resource virtualization in computing was first developed by IBM in the 1960s. At that time computing was centralized to large mainframe computer. Mainframe computing was time consuming as only one user could use the system at a time. To resolve this problem IBM invented time-sharing technology, which allowed several users to have access to the same mainframe computer at the same time. After mainframe computing time, in the 1980s, the x86 computing moved again towards distributed model. This trend started to shift again towards the centralized model in the 1990s, as the performance of x86 based servers increased to the level where most of the computing power of the servers was not utilized anymore. This excess capacity issue caused pressure to find a way of using computing resources more economically. Virtualization has been proven to be the answer to this issue. First version of x86 server virtualization was created by company called VMWare Inc. in 1999 (Brodkin, 2009).

It should be noted that because of the nature of virtualization technology the reference materials used for this thesis are largely produced by companies developing virtualization products. These reference materials were chosen because they offer the latest view on technology and also the amount of completely vendor independent information is scarce and often outdated. In this chapter type-1 and type-2 hypervisor differences and aspects are discussed in section *Virtualization Types*, then technical concepts essential to hypervisors are discussed in sections *Isolation*, *Efficiency, Resource Control* and *Emulation*.

3.1.1 Virtualization Types

In computing, resource virtualization is most commonly carried out by creation of *virtual machines.* Virtual machine is a software equivalent of physical computer that is an:

efficient, isolated duplicate of the real machine. (Popek & Goldberg, 1974, p. 413)

In order to run virtual machines on real machines, there must be a software layer between real hardware and virtual machine. This piece of software has been traditionally called a *virtual machine monitor* (VMM). The term has later been replaced with term *hypervisor.* VMM or hypervisor has three essential properties:

1. Provides environment for programs which is essentially identical with the original machine.
2. Programs run on this environment show at worst only minor decreases in speed.
3. VMM is in complete control of system resources (Popek & Goldberg, 1974, p. 413).

The virtual machine monitor concept is illustrated in Figure 4.

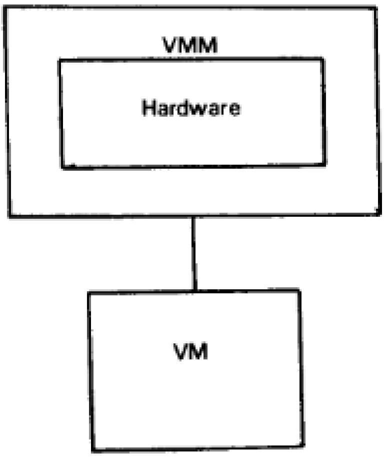


Figure 4 The virtual machine monitor (Popek & Goldberg, 1974, p. 413).

Modern hypervisors can be broadly divided into two categories, type-1 and type-2 hypervisors. Next, type-1 and type-2 hypervisor characteristics are discussed in more detail.

3.1.2 Type-1 Hypervisors

Type-1 virtualization is also known as full virtualization or bare-metal virtualization. The concept for VMM (type-1 hypervisor) was initially introduced in 1974 by Popek and Goldberg in their article *Formal Requirements for Virtualizable Third Generation Architectures*.

In type-1 virtualization all aspects of hardware resources are virtualized and presented to virtual machines as if they were "real". All operating system calls are sent to the virtual resource (i.e. to virtual CPU) and these calls are then translated by the hypervisor to the underlying physical hardware. (VMWare, Inc, 2007).

A common example of type-1 virtualization is server virtualization. Server virtualization means that functions of physical servers are converted to virtual machines (files) and these files are then run on dedicated virtualization platforms, hypervisors. Hypervisors are in effect servers that are running virtualization software that enables the server resources to be divided between virtual machines. For example, the following resources can be virtualized:

- Operating systems
- Central Processing Units (CPUs)
- Memory
- Network interfaces
- Storage devices.

Figure 5 depicts the differences between physical and virtual servers (Josyula, et al., 2011, pp. 3-4).

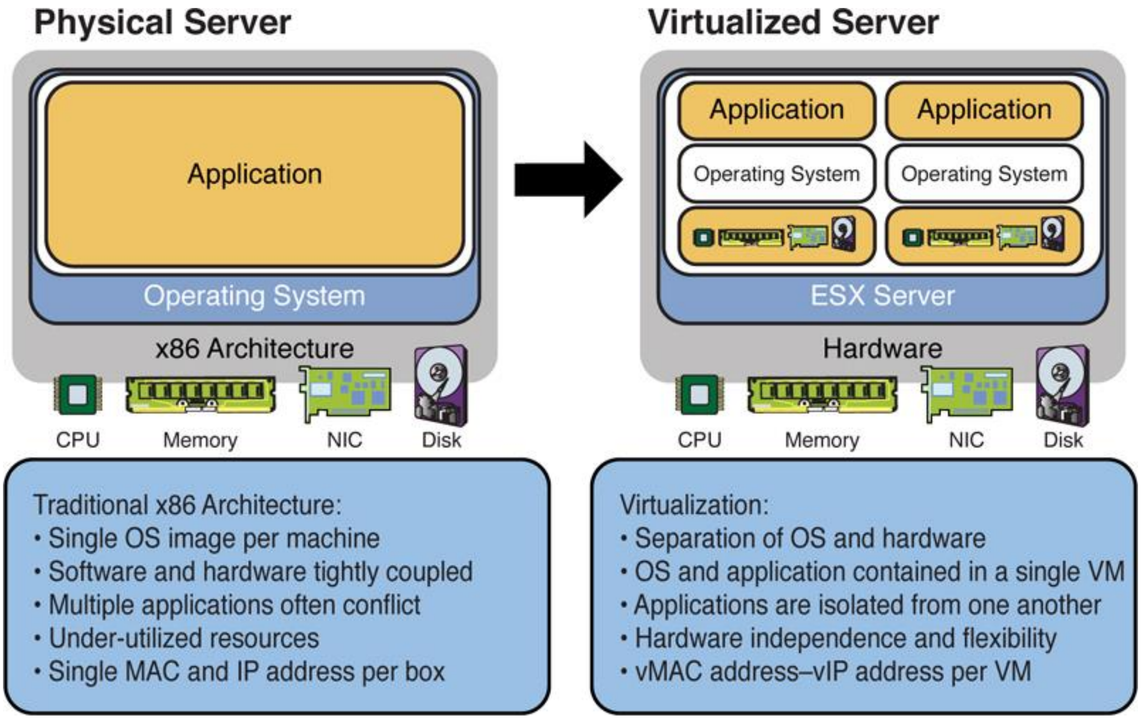

Physical Server
Application
Operating System
x86 Architecture
CPU
Memory
NIC
Disk
Virtualized Server
Application
Application
Operating System
Operating System
ESX Server
Hardware
CPU
Memory
NIC
Disk
Traditional x86 Architecture:
• Single OS image per machine
• Software and hardware tightly coupled
• Multiple applications often conflict
• Under-utilized resources
• Single MAC and IP address per box
Virtualization:
• Separation of OS and hardware
• OS and application contained in a single VM
• Applications are isolated from one another
• Hardware independence and flexibility
• vMAC address–vIP address per VM

Figure 5 Server virtualization example *(Josyula, et al., 2011, p. 4)*

In this thesis the virtualization platform used is type-1 hypervisor (Microsoft Hyper-V).

3.1.3 Type-2 Hypervisors

Type-2 virtualization is also known as paravirtualization, or operating system assisted virtualization. It is a lightweight virtualization technique that does not require the underlying hardware to support virtualization natively. The paravirtualized guest OS however needs to be modified to support virtualization. In contrast type-1 virtualization can support any OS, as in type-1 virtualization the guest OS is agnostic to virtualization. Paravirtualization hypervisor was first developed for Linux operating system by Xen Project team.

Paravirtualization software creates communication channels between hypervisors and guest operating systems by special PV drivers. Interaction between hardware and paravirtualized guest OS is depicted in Figure 6 (Xen.org, 2015).

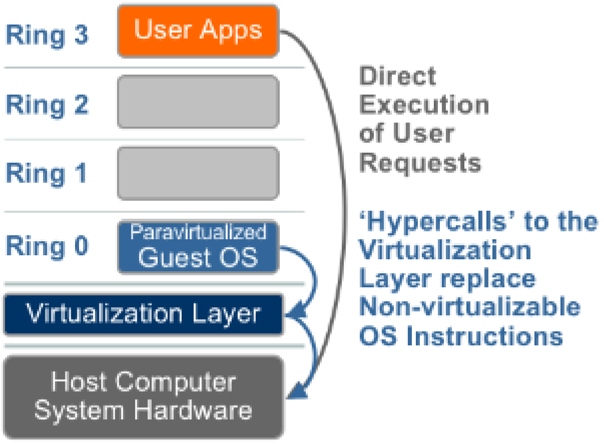

Ring 3
User Apps
Ring 2
Ring 1
Ring 0
Paravirtualized Guest OS
Virtualization Layer
Host Computer System Hardware
Direct Execution of User Requests
'Hypercalls' to the Virtualization Layer replace Non-virtualizable OS Instructions

Figure 6 Paravirtualization driver interaction (VMWare, Inc, 2007)

Because OS kernel needs to be modified to support paravirtualization, the OS support for it is limited. For popular OS versions the support is quite comprehensive, for example VirtualBox open source virtualization platform version 5.0 supports the following OS's at the time of writing (list from VirtualBox online manual):

- **Windows** hosts:
  - Windows Vista SP1 and later (32-bit and 64-bit).
  - Windows Server 2008 (64-bit)
  - Windows Server 2008 R2 (64-bit)
  - Windows 7 (32-bit and 64-bit)
  - Windows 8 (32-bit and 64-bit)
  - Windows 8.1 (32-bit and 64-bit)
  - Windows 10 RTM build 10240 (32-bit and 64-bit)
  - Windows Server 2012 (64-bit)
  - Windows Server 2012 R2 (64-bit)
- **Mac OS X** hosts (64-bit):
  - 10.8 (Mountain Lion)
  - 10.9 (Mavericks)
  - 10.10 (Yosemite)
  - 10.11 (El Capitan)
- **Linux** hosts (32 and 64-bit):
  - Ubuntu 10.04 to 15.04
  - Debian GNU/Linux 6.0 ("Squeeze") and 8.0 ("Jessie")
  - Oracle Enterprise Linux 5, Oracle Linux 6 and 7

  - Redhat Enterprise Linux 5, 6 and 7
  - Fedora Core / Fedora 6 to 22
  - Gentoo Linux
  - openSUSE 11.4, 12.1, 12.2, 13.1
  - Mandriva 2011
- **Solaris** hosts (64-bit):
  - Solaris 11
  - Solaris 10 (U10 and higher) (Oracle, 2016).

Compared to full virtualization, paravirtualization has lower virtualization overhead. Lower overhead generally means better performance, but it should be noted that workload affects performance greatly. (VMWare, Inc, 2007)

3.1.4 Hypervisor Characteristics

There are four basic charasterictics of hypervisors, isolation, efficiency, resource control and emulation. These characteristics are discussed in the following sections.

3.1.4.1 Isolation

In virtualization, same physical resources are shared by multiple virtual machines. The hypervisor software layer is responsible for sharing hardware between different VMs. This is done by creating *logical partitions of real objects.* These logical partitions are then conglomerated as instances of virtual objects. An example of this is a hard disk drive which can be partitioned to several logical parts which represent parts of the physical hard disk (Jithin & Chandran, 2014).

In the virtualized environment each virtual resource dedicated to a VM is isolated from other VMs. This is crucial from performance point of view; it prevents one VM consuming all server resources. Isolation is also mandatory in security point of view; it guarantees that VMs have only access to resources appointed to them which mitigates data leakage between VMs. Because of isolation one VM operating system failure has no effect on other virtual machines running on the same physical host (VMWare Inc., 2016).

3.1.4.2 Efficiency

Virtual processor instructions must be executed predominantly directly by the real processor in order to satisfy the efficiency requirements. In modern CPUs the efficiency is further enhanced by technologies that natively support virtualization techniques. Examples of these technologies are *Intel VT-x* for Intel processors and *AMD-V* for AMD processors.

3.1.4.3 Resource Control

The hypervisor is responsible for allocating resources to VMs. This means that VM by default cannot access any resource that is not explicitly allocated to it and hypervisor also is able to reclaim allocated resources back from VM (Popek & Goldberg, 1974, p. 413).

3.1.4.4 Emulation

Emulation means imitating something to someone. In computing emulation can be used when there is requirement to run code developed for specific hardware on another platform. As an example iPhone mobile phone code can be tested with *Simulator* software that emulates iPhone hardware. (Apple Inc., 2016)

Emulation process consumes large amount of processor resources and therefore performance in emulated environment is far from performance in non-emulated environment (Boley, 2014).

While emulation is not technically virtualization, the concept is paramount to hypervisor environments. In hypervisor context the hypervisor layer must transparently emulate physical resources to guest operating systems.

3.2 X86 Server Virtualization

In this section x86 server component virtualization methods are discussed. There are four main components that are present in practically every server. These components are CPU, memory, storage and network interface. Virtualization principles of each of these components are discussed in this section.

3.2.1 X86 CPU Virtualization

In this section the CPU virtualization is discussed. Firstly, the CPU architecture is reviewed, then CPU virtualization techniques are described and lastly, software and hardware based virtualization methods are briefly considered.

3.2.1.1 X86 CPU Architecture

In modern x86 CPU architecture the instruction sets are classified with four different privilege types called rings. This classification is known as a CPU protected mode illustrated in Figure 7.

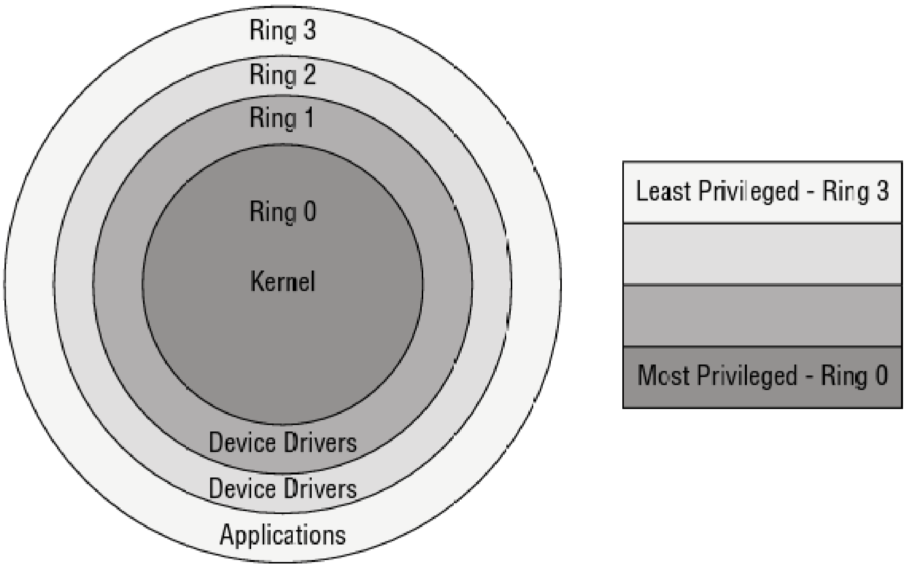


Figure 7 X86 architecture privilege levels (Liebowitz, et al., 2014, p. 65)

In CPU protected mode the ring 0 is the level with most privileges. Ring 0 is also the level where Operating System (OS) kernel and device drivers code usually is run. The code run in this ring has generally full access to hardware resources. Code run in ring 1 and 2 has more restrictions with resource access. These rings are rarely used in modern operating systems. Code run in ring 3 has the least privileges to hardware. Ring 3 is the privilege level that is given to applications and other code that is not OS kernel or device driver. (Liebowitz, et al., 2014, p. 65)

3.2.1.2 X86 CPU Virtualization

X86 processor architecture is complicated and x86 instruction set consists of more than 800 instructions. In order to virtualize the CPU, the virtual CPU events must be trapped and presented to the real CPU in understandable way. This is done with a software component called *hypervisor.* (Amit, et al., 2015)

Hypervisor is the software layer responsible of capturing the VM CPU instructions and directing them to available physical CPU processors. X86 CPUs today have multiple cores and are designed with technologies that assist virtualization. The virtual CPU of a VM sends most of its instructions directly to the physical CPU. The x86 CPU virtualization is divided further to two categories, software-based and hardware-based CPU virtualization. Software-based CPU virtualization feature is designed to run virtual CPU instructions on ring 3 privilege level. In contrast, hardware assisted virtualization is a built-in feature in modern x86 processors which provides possibility to run code in all privilege levels (rings 0-3). For Intel processors the feature is called “Intel Virtualization Technology” (Intel VT) for AMD processors it is called “AMD-V” and for VIA processors it is called “VIA VT”. (Liebowitz, et al., 2014, pp. 63-68)

3.2.2 Memory Virtualization

One of the crucial server resources that can be virtualized is the system memory. In this chapter an example of VMWare memory virtualization is discussed. Other vendors are using similar memory virtualization techniques.

Modern processors generally are capable of memory virtualization. Hypervisor host creates a uniform virtual memory address space which is then allocated to VMs as needed. The hypervisor handles address translations between virtual and physical memory. The VM has virtual and physical memory layers, where virtual memory is used by applications and physical memory is used by the operating system (OS). Figure 8 depicts the relations between physical and virtual memory allocations.

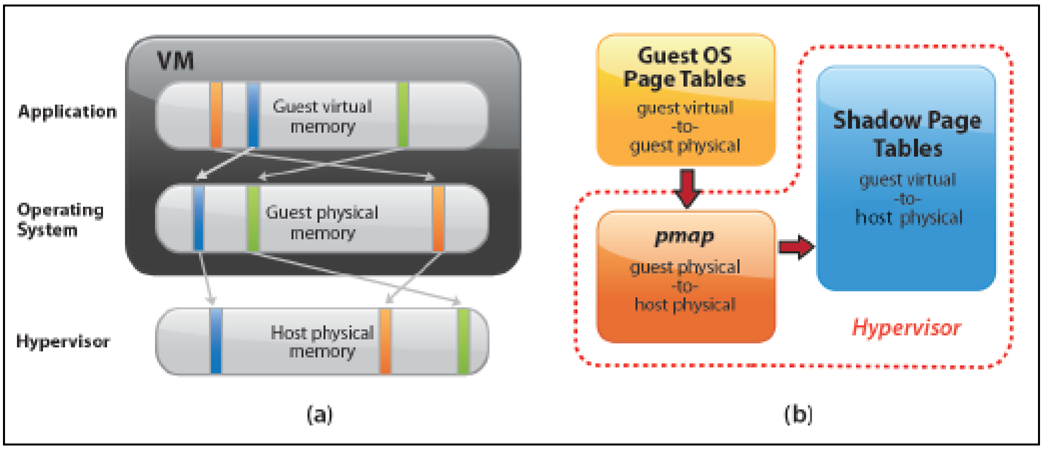


Figure 8 Virtual and physical memory relations (Fei Guo, VMWare, Inc, 2011, p. 6)

Hypervisor maintains the VM physical address space mappings in a mapping data structure called *pmap*. The application calls to VM virtual memory are then mapped to separate table called “Shadow Page Table”. Figure 8 depicts the relations of these mappings (Fei Guo, VMWare, Inc, 2011, pp. 4-6).

3.2.3 Storage Virtualization

Application data created by CPU and memory operations needs to be stored to be useful. In traditional servers the server itself contained one or more hard disks where data was stored. This method is called direct attached storage (DAS). In virtualized environments it is more common to use Storage Area Network (SAN) or Network Attached Storage (NAS). SAN or NAS is effectively a storage system that can be used over network connection. For VMs the storage virtualization is transparent. When VM OS requests storage operations it utilizes a SCSI driver which is used to access storage resources. The SCSI driver then communicates with hypervisor, which again has storage driver to contact storage controller that communicated over network to physical storage systems. Figure 9 depicts the virtual storage pathway.

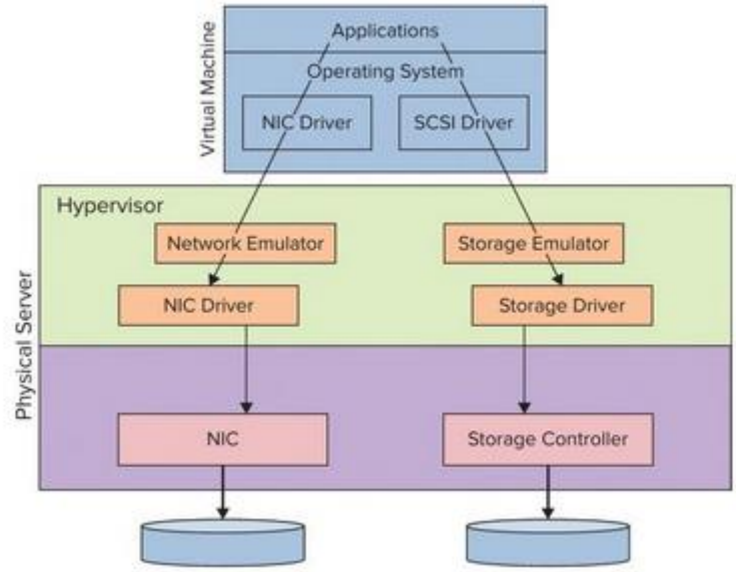


Figure 9 Virtual storage (Portnoy, 2012, p. 153)

Storage operations are critical for VMs and for this reason virtualized storage systems almost always utilize dedicated storage networks (Portnoy, 2012, p. 153).

3.2.4 NIC Virtualization

In order for the VM to communicate to outside resources (and vice versa) it needs to have network connectivity. Figure 10 portrays a simple virtual network where one virtual switch is used for VM to VM communication and one is used to communicate external resources.

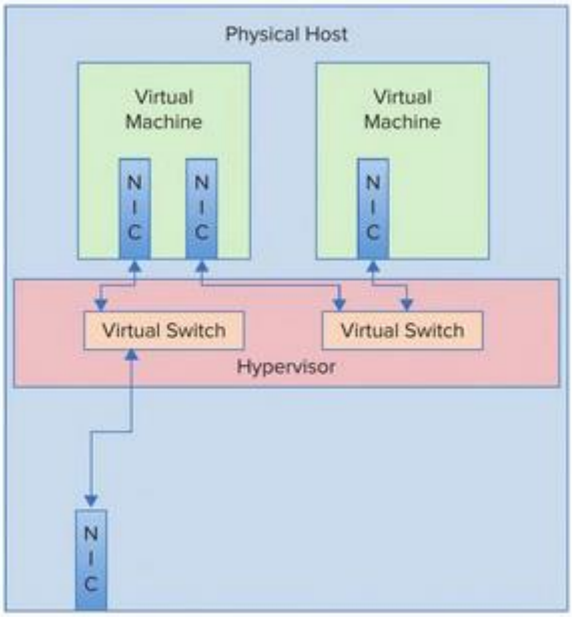


Figure 10 Virtual network example (Portnoy, 2012, p. 40)

Each VM can have one or more virtual network interface cards (NICs) which are connected to virtual switch(es) created by the hypervisor. In order to reach external networks, the virtual switch can be bound to real NIC(s) of the hypervisor server (Portnoy, 2012, pp. 39-40).

3.3 Network Virtualization Technologies

The rapid growth of the number of network connected devices due to the adoption of the Internet in the 1990s quickly introduced network scalability and efficiency issues. In order to control and mitigate these issues network virtualization technologies were developed. Arguably the key technologies that emerged were virtual local area networks, VLANs and virtual routing and forwarding tables, VRFs. A VLAN is designed to allow network partitioning on the switch (L2) level and VRF is a technology to allow router (L3) level partitioning (Santana, 2014, pp. loc 1542-1550). The key concepts behind these technologies are discussed in the following section.

3.3.1 Virtual Local Area Network (VLAN)

In modern LAN environments the network capable devices (PCs, laptops, servers, thin-clients, etc.) are connected to switches. In default configuration all devices connected to same switch belong to the same *broadcast domain* and all devices receive every broadcast packet that is received in any switch port. Broadcast packets have an important role in most Ethernet networks, as many important protocols utilize broadcast packets. Widely used protocols include Address Resolution Protocol (ARP), Dynamic Host Configuration Protocol (DHCP) and Bootstrap Protocol (BOOTP) to name a few (Santana, 2014, pp. loc. 1576-1607). Figure 11 illustrates the concept of a broadcast domain.

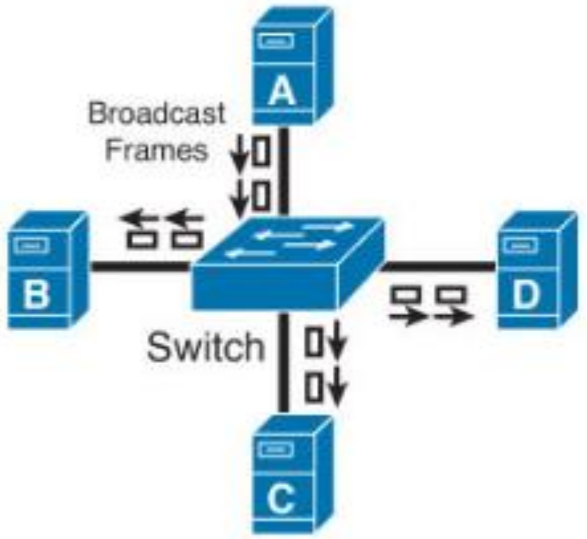


Figure 11 Switch broadcast packet forwarding (Santana, 2014, p. loc. 1583)

In small networks the amount of broadcast traffic is not an issue, but with a larger number of connected hosts the network performance degrades significantly when the number of broadcast packets increases. To enhance network performance and reduce broadcast domain size, VLANs can be configured to segment the switch ports to logical collections of several smaller broadcast domains (Santana, 2014, p. loc. 1588).

3.3.1.1 VLAN Definition

VLAN is by definition a

> Group of devices on one or more LANs that are configured to communicate as if they were attached to the same wire, when in fact they are located on a number of different LAN segments (Cisco Systems, Inc, 2013).

Figure 12 shows an example of three logically separated networks.

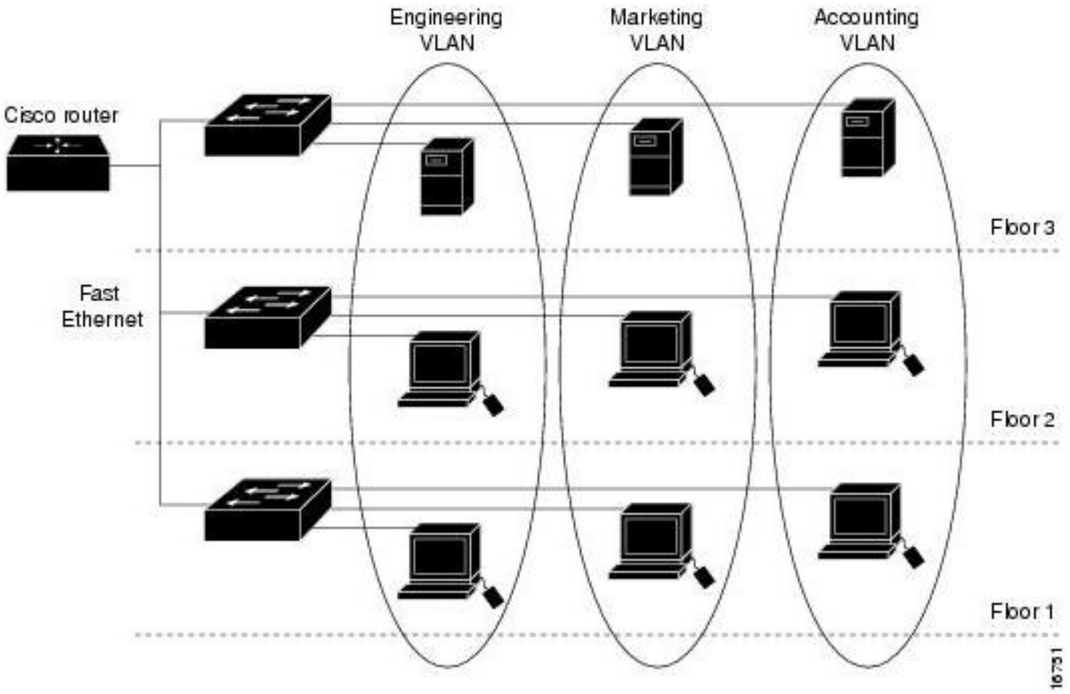


Figure 12 Example of VLAN segmentation (Cisco Systems, Inc, 2013)

The use of VLANs allows several broadcast domains to be defined in a single switch, or the VLAN can be configured across several switches. The switch ports can be allocated to different VLANs and arranged to logical segments (Cisco Systems, Inc, 2013).

3.3.1.2 802.1Q VLAN Tagging

The IEEE 802.1Q standard defines the method for VLAN tagging. The incoming frames on switch ports are assigned to different VLANs by the use of VLAN tags at the switch level. When 802.1Q VLAN tagging is enabled, the switch inserts a 4-byte tag field in the Ethernet frame received from the connected host. This process also causes the frame checksum (FCS field) of the frame to be recalculated. Figure 13 displays the frame modifications performed by the switch.

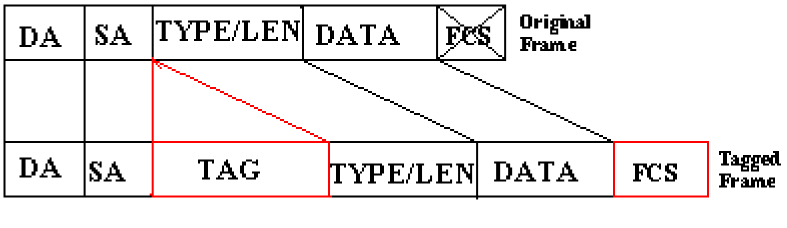
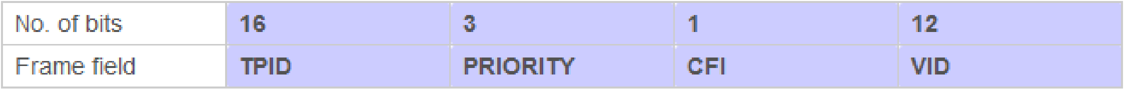


Figure 13 VLAN Ethernet frame field modifications (Cisco Systems, Inc., 2015)

Further, the inserted 802.1Q tag frame field consists of four sub-fields. The expanded tag field is illustrated in Figure 14 and the purpose of each field is explained in more detail below.

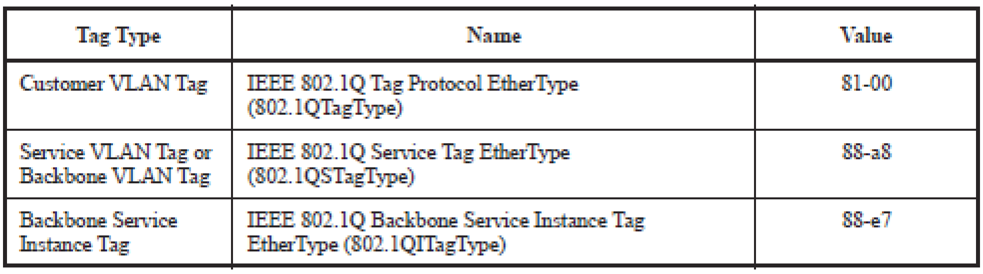


No. of bits	16	3	1	12
Frame field	TPID	PRIORITY	CFI	VID

Figure 14 802.1Q frame fields (Cisco Systems, Inc., 2015)

TPID – Tag Protocol Identifier

The frame *EtherType* is identified in the TPID field. The EtherType information is used to choose the tag decoding functions. The 802.1Q standard defines three types of EtherTypes, C-TAG, S-TAG and I-TAG. Table 1 shows tag types and associated TPID field values.



Tag Type	Name	Value
Customer VLAN Tag	IEEE 802.1Q Tag Protocol EtherType (802.1QTagType)	81-00
Service VLAN Tag or Backbone VLAN Tag	IEEE 802.1Q Service Tag EtherType (802.1QSTagType)	88-a8
Backbone Service Instance Tag	IEEE 802.1Q Backbone Service Instance Tag EtherType (802.1QITagType)	88-e7

Table 1 TPID tag types (IEEE Standards Association, 2014, p. 160)

PRIORITY

The PRIORITY field marks the frame with priority level defined by IEEE 802.1p. The field can have eight distinct values (0 to 7). (Cisco Systems, Inc., 2015)

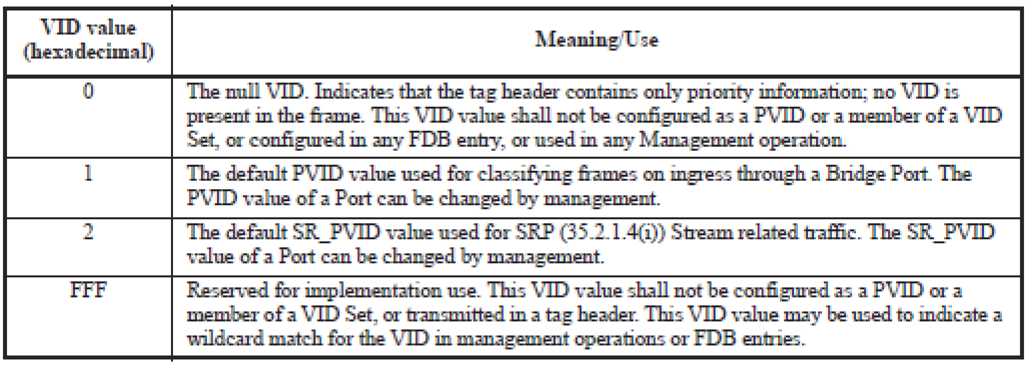
CFI – Canonical Format Indicator

CFI field can have two values, 1 or 0. Field value of 1 indicates the MAC address is expressed in noncanonical format. Field value of 0 indicates the MAC address is in canonical (unique) format. (Cisco Systems, Inc., 2015).

VID – VLAN Identifier

VID field is 12-bits in length. VID can have values between 0 and 4094. 802.1Q standard defines four reserved VID values, 0-2 and 4095. Reserved VIDs are illustrated in Table 2.

Table 2 Reserved VID values



VID value (hexadecimal)	Meaning/Use
0	The null VID. Indicates that the tag header contains only priority information; no VID is present in the frame. This VID value shall not be configured as a PVID or a member of a VID Set, or configured in any FDB entry, or used in any Management operation.
1	The default PVID value used for classifying frames on ingress through a Bridge Port. The PVID value of a Port can be changed by management.
2	The default SR_PVID value used for SRP (35.2.1.4(i)) Stream related traffic. The SR_PVID value of a Port can be changed by management.
FFF	Reserved for implementation use. This VID value shall not be configured as a PVID or a member of a VID Set, or transmitted in a tag header. This VID value may be used to indicate a wildcard match for the VID in management operations or FDB entries.

Device vendors shall support the VIDs described in Table 2, but do not necessarily need to support the full range of VIDs between 0 and 4094 (IEEE Standards Association, 2014, p. 160).

3.3.2 Virtual Routing and Forwarding (VRF)

Broadcast domains on layer 2 LANs and VLANs are terminated on layer 3 routers. While switch level network segregation is performed by VLANs, VRFs are an IP layer technology for segregating routing tables. VRFs are often used in conjunction with MPLS networks. Several vendors also have VRF concepts that do not require the use of MPLS (i.e. Multi-VRF, Multi-VRF CE or VRF-Lite (Brocade Inc., 2005)).

Figure 15 illustrates VLAN and VRF concepts.

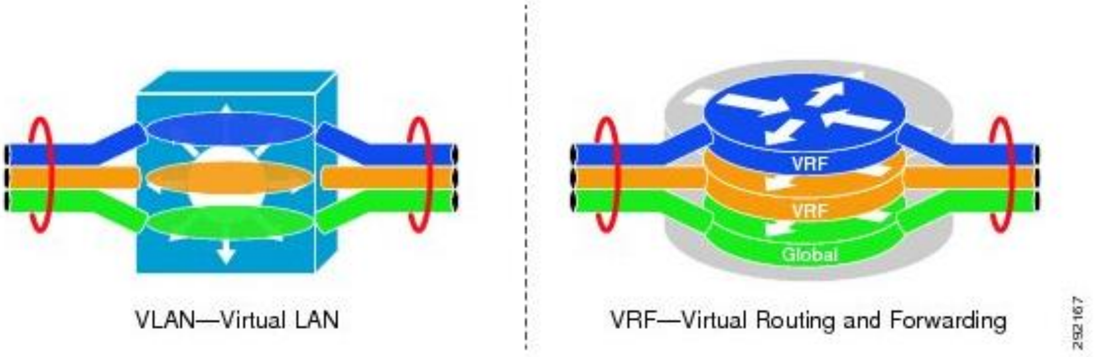


Figure 15 VLANs and VRFs comparison (Cisco Systems, Inc, 2015)

VRF instance is comprised of the following components:

- IP routing table

- Forwarding table
- Interfaces (i.e. loopbacks, physical or vlan interfaces)
- Routing protocols / rules

(Cisco Systems, Inc., 2015)

The use of VRFs allows routers to maintain multiple routing tables simultaneously. The routing tables have no interaction with each other and therefore overlapping IP addresses can be used in separate VRF instances without issues (Cisco Systems, Inc, 2014).

3.4 Cloud Computing

The cloud computing concept according to NIST definition is

> *a model for enabling ubiquitous, convenient, on- demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction* (Mell & Grance, 2011).

The cloud computing concept has been developed under ten years ago. Cloud computing popularity has grown rapidly due to the improvements in underlying technologies. Some key technologies that have advanced cloud computing popularity have been the improving availability of high-speed Internet access and the development of virtualization technologies. Virtualization allows cloud computing service providers to host virtual servers for thousands of customers. Only requirement for customers to start using cloud computing services is to have an Internet access. (Srinivasan, 2014, pp. 2-3).

There are several types of cloud computing. The basic types are:

- Software as a Service (SaaS).
- Platform as a Service (PaaS).
- Infrastructure as a Service (IaaS).

SaaS provides both server hardware and software. An example of SaaS is Microsoft Hotmail, which was launched in 1996. (Srinivasan, 2014, p. 19) PaaS provides the computing platform for applications. An example of PaaS is Google Apps service. (Srinivasan, 2014, p. 24). IaaS allows the users to create virtual machines in leased infrastructure. The VMs can be freely customized to accommodate user needs (Srinivasan, 2014, p. 26). Cloud computing can be deployed in several ways. In the next sections the most common deployment models, public cloud, private cloud and hybrid cloud are discussed.

3.4.1 Public Cloud

In the public cloud deployment model the service provider offers cloud computing services in a multitenant cloud. Public cloud is accessible over the Internet and the service provider manages all infrastructure of the service. Public cloud services are billed with the pay-as-you-go model, which allows the users to pay only for the needed services (Srinivasan, 2014, p. 30). Significant public cloud IaaS service providers include companies such as Amazon Web Services, Google, Microsoft and Rackspace. (Gartner Inc., 2016).

3.4.2 Private Cloud

A private cloud is a cloud computing service hosted only for a specific group or organization. As private cloud deployment is dedicated for the user (organization), there is more control for the infrastructure and for system management. There are several deployment models for private clouds. Private cloud infrastructure can be hosted in an organization's own data center, or it can be outsourced to a third party (a.k.a. *managed private cloud*). In both cases the infrastructure is owned by the organization. In addition to the aforementioned deployment models, a private cloud can also be completely hosted by a cloud service provider. In this case the infrastructure is owned by the service provider, but it is not shared with other customers. This deployment model is known as *hosted private cloud*. The last well-known deployment model is *Virtual Private Cloud (VPC)*. In this deployment model the infrastructure is owned by a service provider and it is shared with other customers, but customers are isolated from one another (Srinivasan, 2014, p. 31).

3.4.3 Hybrid Cloud

Hybrid cloud is a mix between private and public clouds. In this deployment model the customer's private cloud is connected to a public cloud. The benefit of this model is having the possibility to dynamically move workloads and applications to the most suitable platform. This possibility in turn can offer cost savings compared to explicitly private or public cloud deployments (Srinivasan, 2014, pp. 33-34).

3.5 Network Performance Testing Methodologies

Performance testing of network devices provides a method to understand the performance level of the tested device. This understanding gives insight to the network behaviour, optimal configurations and performance levels. (Xing, et al., 2007, p. 780).

Bit-error-rate (BER) testing was an adequate method for testing time-division multiplexing (TDM) and DS1/DS3/2M circuits. For Ethernet-networks, BER testing was proven to not being applicable and IETF RFC 2544 introduced in year 1999 quickly became de facto standard for testing Ethernet network performance. (EXFO Electro-Optical Engineering Inc., 2008).

The RFC 2544 methodology is primarily a method for testing new network devices and does not take into account real live network characteristics, such as CoS/QoS and jitter. ITU-T has created a testing methodology recommendation, Y1564 Service Activation Test, which takes the aforementioned issues into account (Omnitron Systems Technology, Inc, 2016).

3.5.1 IETF RFC 2544: Benchmarking Methodology

The RFC 2544 defines a collection of tests intended for vendors to utilize when measuring and reporting network device performance values. RFC 2544 testing methodology provides data that can be used to compare network device performance between different vendors. It should be noted that the benchmarking methodology is intended only for laboratory setups or Isolated Test Environments (ITE). The tests should not be run in production networks (Bradner, et al., 2012). In the following sections RFC 2544 test methodology is reviewed.

3.5.2 RFC 2544 Test Architecture

For optimal results the tester equipment should have separate transmitting and receiving ports. The test traffic should flow from tester transmit port to device under test (DUT) receiving port and from DUT transmit port to tester receiving port (Bradner & McQuaid, 1999). Figure 16 depicts the test topology

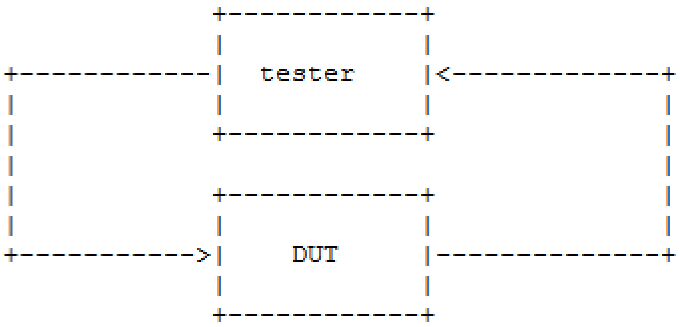


Figure 16 RFC 2544 test setup (Bradner & McQuaid, 1999, p. 3)

The RFC also defines a testing topology for two DUT network. This topology (Figure 17) is supposedly thought for testing networks with multiple media types.

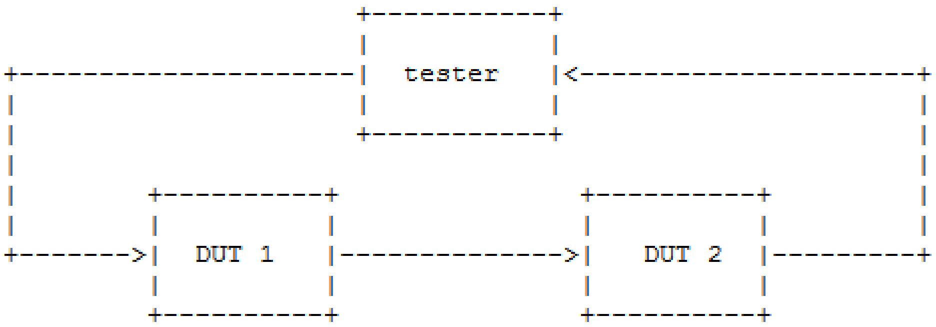


Figure 17 RFC 2544 test for multiple media types (Bradner & McQuaid, 1999, p. 3)

This topology can for example be used when simulating a connection of two LANs over a WAN link (Bradner & McQuaid, 1999, p. 3).

3.5.3 RFC 2544 Test Benchmarks

RFC 2544 defines that tests on Ethernet should be performed with the following frame sizes: 64, 128, 256, 512, 1024, 1280 and 1518 Bytes. Each frame size should be tested separately. Test methodology defines the following six testing benchmarks:

1. throughput, the maximum number of test frames successfully transmitted by DUT

2. latency, the latency introduced by DUT when frames are transmitted with maximum throughput
3. frame loss rate, maximum frame rate that DUT can support without lost frames
4. back-to-back frames, longest burst of frames the DUT can forward without lost frames
5. system recovery, time DUT takes to recover from 110 % throughput overload
6. reset, time DUT takes to recover from reset (Bradner & McQuaid, 1999, pp. 14-18).

It is to be noted that the RFC 2544 method does not take security issues into consideration in the aforementioned tests. (Bradner & McQuaid, 1999, p. 18).

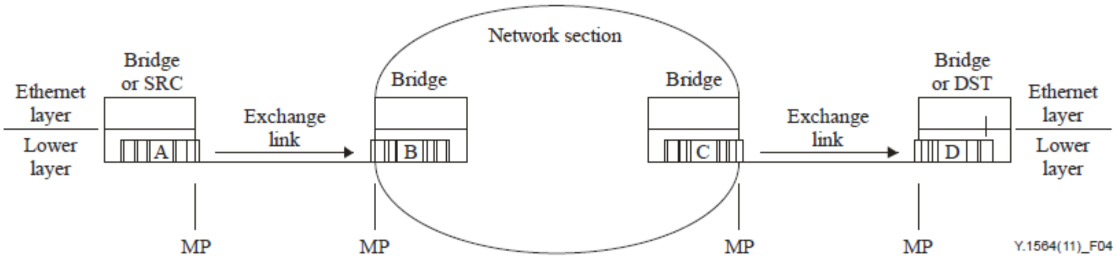
3.5.4 ITU-T Y.1564 Service Activation Testing

In present day networks the RFC 2544 testing has proven to be rather time consuming, which has led to service providers modifying their testing to include only a subset of proposed tests. Service providers now often deliver end customer services in a way that allows capacity upgrades by configurational changes only. This is accomplished by rate limiting or policing end user link based on agreed Service Level Agreement (SLA). RFC 2544 testing has proven to be inadequate for SLA based testing, mostly because it does not take traffic prioritization into account, or measure jitter. These shortcomings have led to the development of the Y.1564 test methodology.

The purpose of the Y.1564 recommendation is to provide a test methodology for evaluating proper configuration and performance of an Ethernet network. Version 1.0 of the recommendation was approved in year 2011 (ITU-T, 2016, p. i).

3.5.5 Y.1564 Test Architecture

Testing methodology in the Y.1564 recommendation varies slightly from RFC 2544 method. The Y1564 method defines four measurement points (MP), two for exiting frames and two for entering frames (ITU-T, 2016, pp. 7-8). The test architecture resembles the RFC 2544 test setup for multiple media types (see Figure 15). Figure 18 illustrates the Y.1564 testing topology.



NOTE 1 – Ethernet exit events for frames A and C.
NOTE 2 – Ethernet entry events for frames B and D.

Figure 18 Y.1564 testing topology (ITU-T, 2016, p. 8)

With this reference model, any end-to-end Ethernet services can be described ((ITU-T, 2016, pp. 7-8).

3.5.6 Y.1564 Test Procedure

The Y.1564 test method defines two objectives, service configuration and service performance validation. The configuration is validated to be working as intended before actual tests are run. The test methodology is depicted in Figure 19.

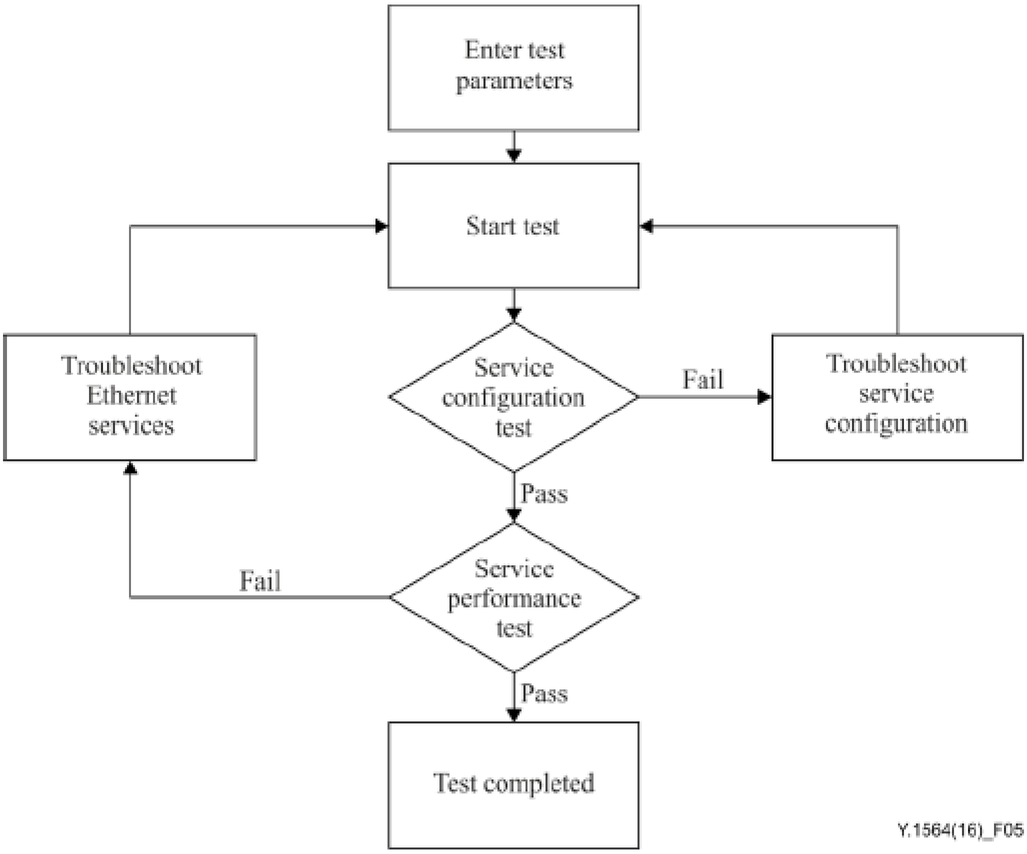
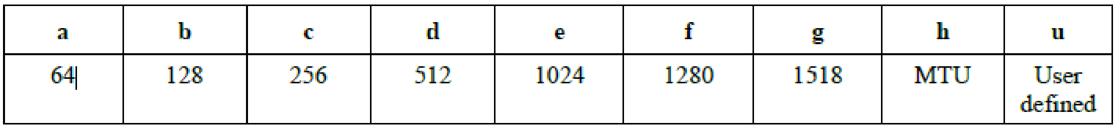


Figure 19 Y.1564 test methodology (ITU-T, 2016, p. 12)

Frame sizes in the Y.1564 performance test method allows the frame size to be fixed, user defined or a distribution of several frame sizes. The frame sizes are shown in Table 3.

Table 3 Y.1564 frame sizes



a	b	c	d	e	f	g	h	u
64	128	256	512	1024	1280	1518	MTU	User defined

The configuration validation for each service can be performed with a step load test. The test traffic ramped up to Committed Information Rate (CIR), after which the traffic rate is increased in two steps. In the first step traffic volume is increased up to Excess Information Rate (EIR) threshold, and in the second step the traffic volume is increased further. Figure 20 clarifies the step-load test concept (ITU-T, 2016, pp. 12-13).

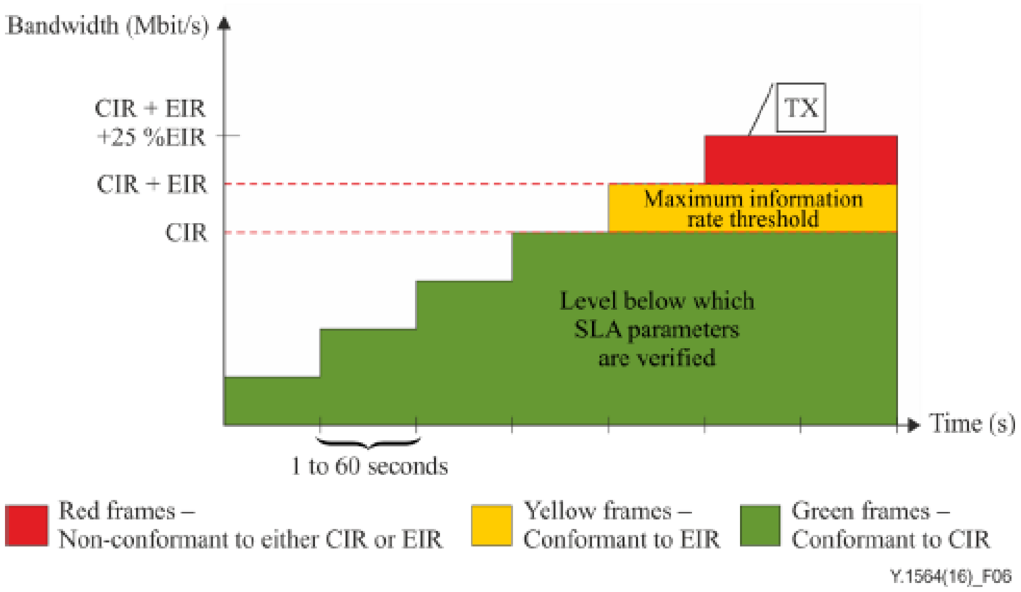


Figure 20 Y.1564 step-load test (ITU-T, 2016, p. 13)

While a service configuration test verifies the correct network operation, a performance test part further confirms the service quality level over time. Test duration per traffic direction can be 15 minutes, 2 hours or 24 hours. In addition to the aforementioned time durations a user defined duration is allowed. The performance test defines five measurement parameters:

1. information rate (IR)
2. Frame transfer delay (FTD)
3. Frame delay variation (FDV)
4. Frame loss ratio (FLR)
5. Service availability (AVAIL SHALL) (ITU-T, 2016, pp. 17-18).

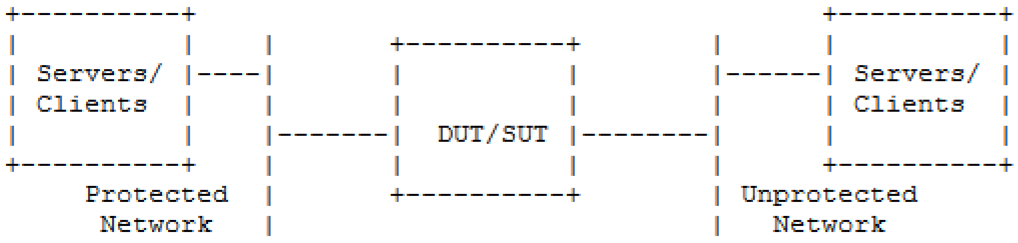
All parameters in the Y.1564 performance test can be measured simultaneously in each test traffic flow, which is considerably less time consuming than performing full RFC 2544 test (Stuart Whitehead, Anritsu Corporation, 2011, p. 4).

3.5.7 IETF RFC 3511: Firewall Performance Testing

Firewalls are often used to control network access between devices. Methodology for testing firewall performance is outlined in RFC document 3511 from year 2003: Methodology for Firewall Performance.

3.5.8 RFC 3511 Test Architecture

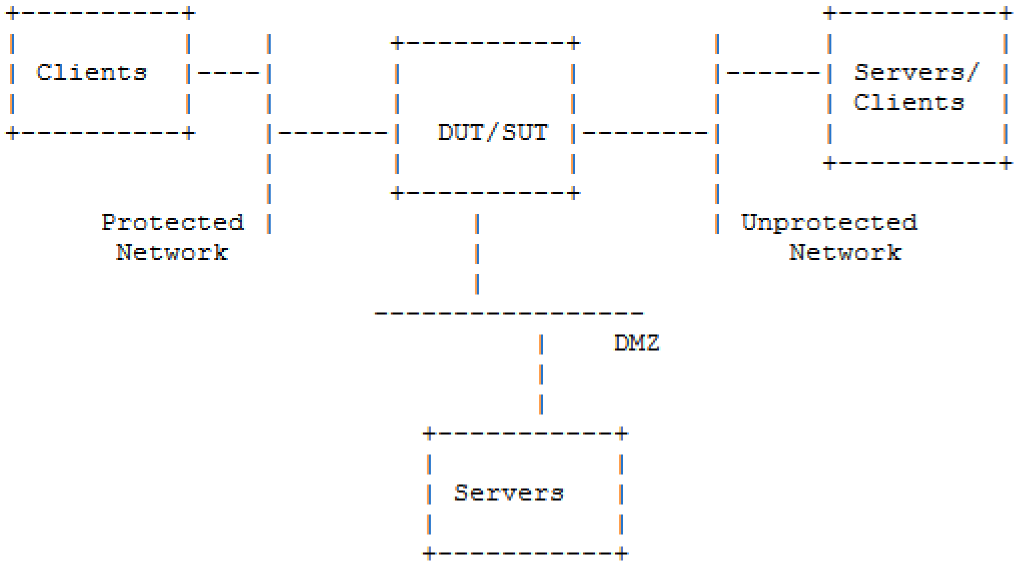
The RFC 3511 methodology defines two testing topologies, dual-homed and tri-homed. In the dual-homed topology, the firewall (DUT) is connected to protected network (LAN) and to unprotected network (Internet). The dual-homed testing topology is shown in Figure 21.



```
+-----------+                                          +-----------+
|           |    |       +-----------+         |       |           |
| Servers/  |----|       |           |         |-------| Servers/  |
| Clients   |    |       |           |         |       | Clients   |
|           |    |-------|  DUT/SUT  |---------|       |           |
+-----------+    |       |           |         |       +-----------+
     Protected   |       +-----------+         | Unprotected
      Network    |                             |    Network
```

Figure 21 RFC 3511 dual-homed topology (Hickman, et al., 2003, p. 3)

The tri-homed setup describes a network where a demilitarized zone (DMZ) segment is used to place the servers. The DMZ approach separates the servers from clients in protected networks. Figure 22 shows the tri-homed topology.



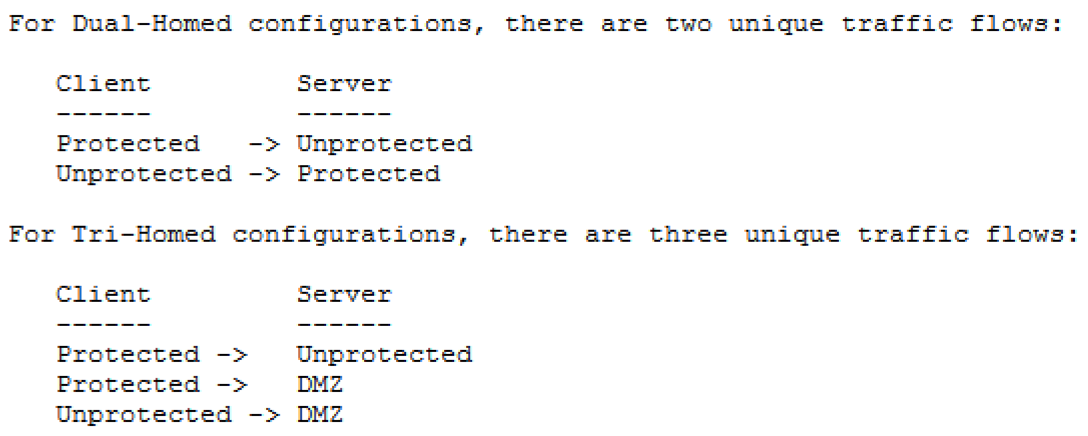
```
+-----------+                                          +-----------+
|           |    |       +-----------+         |       |           |
| Clients   |----|       |           |         |-------| Servers/  |
|           |    |       |           |         |       | Clients   |
+-----------+    |-------|  DUT/SUT  |---------|       |           |
                 |       |           |         |       +-----------+
                 |       +-----------+         |
      Protected  |             |               | Unprotected
       Network                 |                    Network
                               |
                    ------------------
                                   |    DMZ
                                   |
                                   |
                          +------------+
                          |            |
                          |  Servers   |
                          |            |
                          +------------+
```

Figure 22 RFC 3511 tri-homed topology (Hickman, et al., 2003, p. 3)

The tri-homed topology enhances security due to the fact that in this method the servers placed in the DMZ are not directly accessible from internal hosts. (Hickman, et al., 2003, p. 3).

3.5.9 RFC 3511 Test Procedure

The suggested traffic type for performance testing is HTTP 1.1 or higher. Traffic flows that should be tested are described in Figure 23.



```
For Dual-Homed configurations, there are two unique traffic flows:

   Client         Server
   ------         ------
   Protected   -> Unprotected
   Unprotected -> Protected

For Tri-Homed configurations, there are three unique traffic flows:

   Client         Server
   ------         ------
   Protected ->   Unprotected
   Protected ->   DMZ
   Unprotected -> DMZ
```

Figure 23 RFC 3511 test traffic flows

The method also suggests the following testing considerations:

- In case of client-server testing clients must make the connections in round-robin fashion.
- Traffic should be tested with Network Address Translation (NAT) enabled and with NAT disabled.
- Tests should be run with varying rule set sizes.
- Any web caching properties should be switched off .
- Any authentication process must be a part of connectivity setup.
- Any special TCP stack parameters must be noted in test report. (Hickman, et al., 2003, pp. 5-6)

The RFC 3511 testing does not necessarily need to be performed with unique physical data sources (clients/servers). It is possible to employ virtual data sources, but the test report needs to specify the number of virtual clients and servers used in testing (Hickman, et al., 2003, pp. 3-4).

3.5.10 RFC 3511 Test Benchmarks

The firewall testing methodology specifies ten test benchmarks:

- IP throughput
- TCP connection capacity
- TCP connection establishment rate
- Maximum TCP connection tear down rate
- Denial of service handling
- HTTP transfer rate
- Maximum HTTP transaction rate
- Illegal traffic handling
- IP fragmentation handling
- Latency (Hickman, et al., 2003, pp. 6-28)

The objective of the benchmark testing is to determine the throughput of network layer data traversing the DUT/SUT. The test report needs to define IP packet size and the test duration expressed in seconds. Instrument used for testing must deliver unicast IP packets to the DUT/SUT at a constant rate and the testing can be performed with unidirectional or with bi-directional traffic, or with both. The testing is effected in an iterative way; with each iteration the test instrument offers varying load to the DUT/SUT until the maximum rate where packet loss does not occur is found (Hickman, et al., 2003, pp. 6-7).

4 Current State Analysis of NSP Cloud Based Firewall Service

Currently the services in the NSPs data centers are produced so that the dedicated customer network segments are protected with network firewalls. The firewall implementation is “cloud-based”, meaning that physical network firewalls are segmented to provide isolated firewall instances for each customer. The cloud based firewall service in the NSP product portfolio is called Cloud Security Gateway (Cloud-SGW). The Cloud-SGW implementation for a customer is not protecting only Internet bound traffic; it protects also the traffic between different customer network segments. These network segments are usually dedicated to traffic from other NSP services, such as MPLS, VPN, WLAN or VM traffic.

Traffic from other services can be forwarded to cloud-SGW in two ways:

1. By switching the traffic between the SGW and device(s) that provide the service. In this method network segments are isolated with VLANs.
2. By routing the traffic between the SGW and device(s) that provide the service. In this method each network segment traffic is isolated with VLANs on the switch level and with VRFs on the router level.

Implementing a new network service for the customer often requires physical cabling work and requires configurations on data center switches and routers. In all cases cloud-SGW configuration is needed. In the following section, *3.1 NSP Security Gateway Service,* the NSP SGW services are discussed in more detail.

4.1 NSP Security Gateway Service

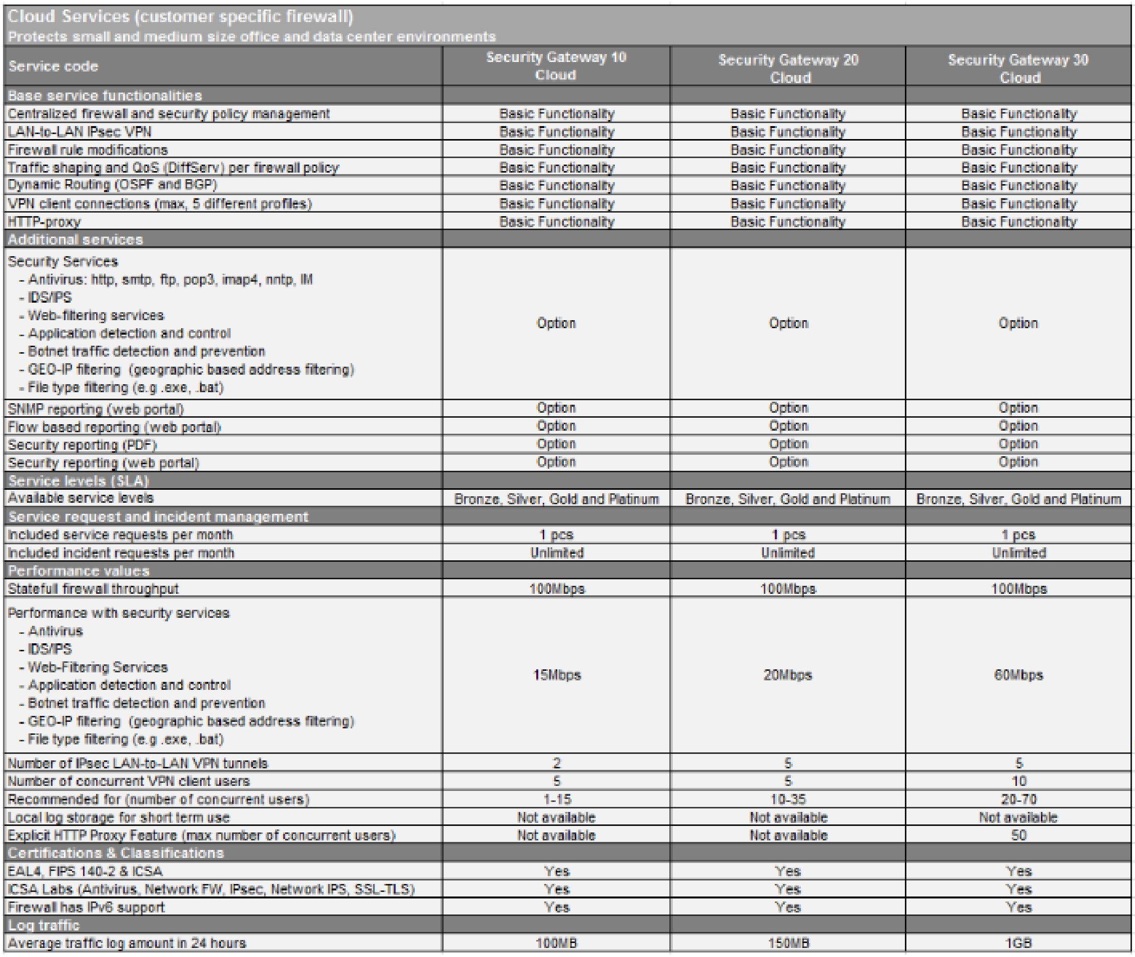
NSP Security Gateway Service (SGW) is a firewall service for enterprise customers. The service has two possible implementation methods:

1. As a cloud service via NSP data center.
2. As a dedicated CPE service in customer or third party premises.

The SGW service is typically used to protect data center and hosting resources, Internet access or internal production segments. The CPE implementation can also be used to protect remote offices and provide connectivity to enterprise WAN.

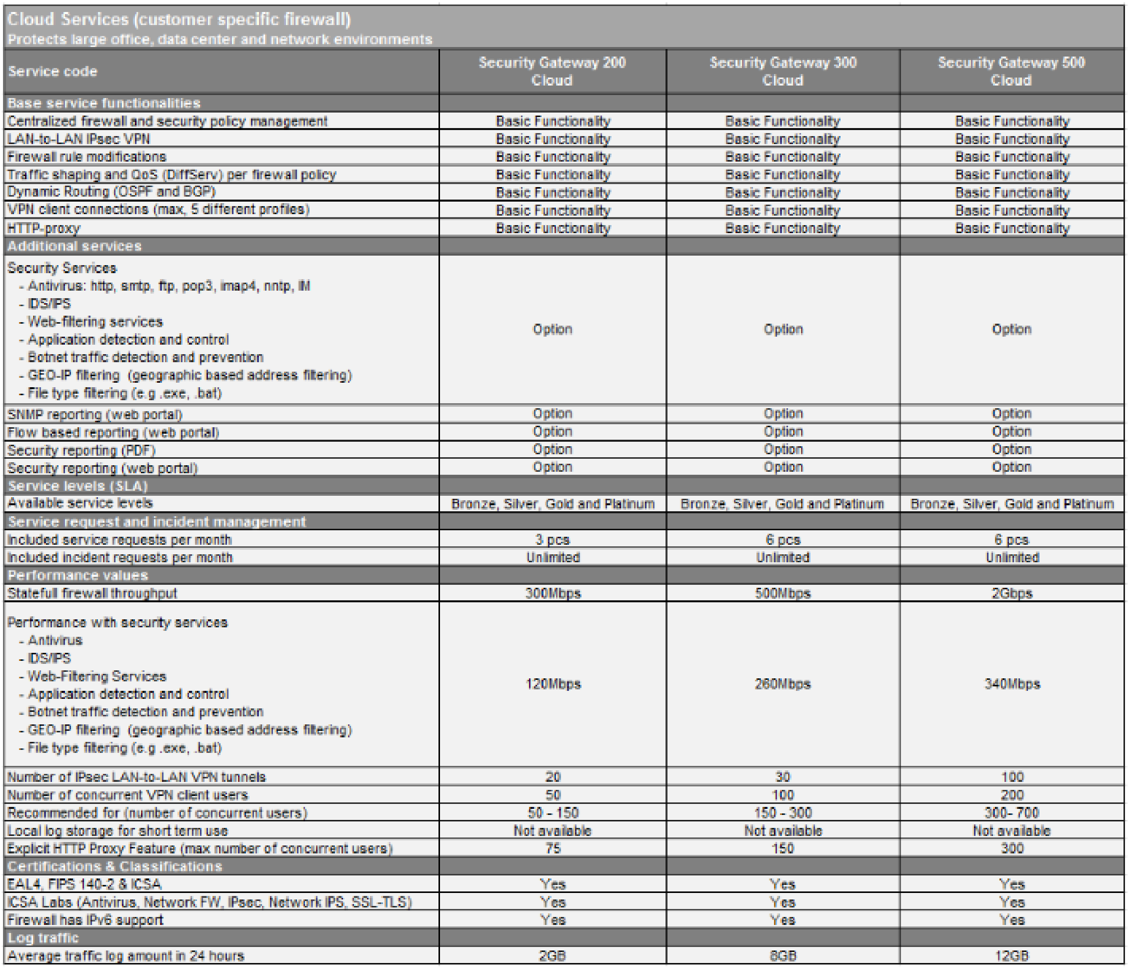
For the purpose of this Thesis, the cloud-based SGW service is interesting as it could, in principle, be partly or completely replaced with the Virtual Firewall based implementation method. The cloud service SGW properties per product code are visible in Tables 4 and 5 in more detail.

Table 4 NSP cloud SGW 10, 20 and 30 properties



Cloud Services (customer specific firewall)			
Protects small and medium size office and data center environments			
Service code	Security Gateway 10 Cloud	Security Gateway 20 Cloud	Security Gateway 30 Cloud
Base service functionalities			
Centralized firewall and security policy management	Basic Functionality	Basic Functionality	Basic Functionality
LAN-to-LAN IPsec VPN	Basic Functionality	Basic Functionality	Basic Functionality
Firewall rule modifications	Basic Functionality	Basic Functionality	Basic Functionality
Traffic shaping and QoS (DiffServ) per firewall policy	Basic Functionality	Basic Functionality	Basic Functionality
Dynamic Routing (OSPF and BGP)	Basic Functionality	Basic Functionality	Basic Functionality
VPN client connections (max, 5 different profiles)	Basic Functionality	Basic Functionality	Basic Functionality
HTTP-proxy	Basic Functionality	Basic Functionality	Basic Functionality
Additional services			
Security Services - Antivirus: http, smtp, ftp, pop3, imap4, nntp, IM - IDS/IPS - Web-filtering services - Application detection and control - Botnet traffic detection and prevention - GEO-IP filtering (geographic based address filtering) - File type filtering (e.g .exe, .bat)	Option	Option	Option
SNMP reporting (web portal)	Option	Option	Option
Flow based reporting (web portal)	Option	Option	Option
Security reporting (PDF)	Option	Option	Option
Security reporting (web portal)	Option	Option	Option
Service levels (SLA)			
Available service levels	Bronze, Silver, Gold and Platinum	Bronze, Silver, Gold and Platinum	Bronze, Silver, Gold and Platinum
Service request and incident management			
Included service requests per month	1 pcs	1 pcs	1 pcs
Included incident requests per month	Unlimited	Unlimited	Unlimited
Performance values			
Statefull firewall throughput	100Mbps	100Mbps	100Mbps
Performance with security services - Antivirus - IDS/IPS - Web-Fitering Services - Application detection and control - Botnet traffic detection and prevention - GEO-IP filtering (geographic based address filtering) - File type filtering (e.g .exe, .bat)	15Mbps	20Mbps	60Mbps
Number of IPsec LAN-to-LAN VPN tunnels	2	5	5
Number of concurrent VPN client users	5	5	10
Recommended for (number of concurrent users)	1-15	10-35	20-70
Local log storage for short term use	Not available	Not available	Not available
Explicit HTTP Proxy Feature (max number of concurrent users)	Not available	Not available	50
Certifications & Classifications			
EAL4, FIPS 140-2 & ICSA	Yes	Yes	Yes
ICSA Labs (Antivirus, Network FW, IPsec, Network IPS, SSL-TLS)	Yes	Yes	Yes
Firewall has IPv6 support	Yes	Yes	Yes
Log traffic			
Average traffic log amount in 24 hours	100MB	150MB	1GB

Table 5 NSP cloud SGW 100, 200 and 300 properties



Cloud Services (customer specific firewall) Protects large office, data center and network environments			
Service code	Security Gateway 200 Cloud	Security Gateway 300 Cloud	Security Gateway 500 Cloud
Base service functionalities			
Centralized firewall and security policy management	Basic Functionality	Basic Functionality	Basic Functionality
LAN-to-LAN IPsec VPN	Basic Functionality	Basic Functionality	Basic Functionality
Firewall rule modifications	Basic Functionality	Basic Functionality	Basic Functionality
Traffic shaping and QoS (DiffServ) per firewall policy	Basic Functionality	Basic Functionality	Basic Functionality
Dynamic Routing (OSPF and BGP)	Basic Functionality	Basic Functionality	Basic Functionality
VPN client connections (max, 5 different profiles)	Basic Functionality	Basic Functionality	Basic Functionality
HTTP-proxy	Basic Functionality	Basic Functionality	Basic Functionality
Additional services			
Security Services - Antivirus: http, smtp, ftp, pop3, imap4, nntp, IM - IDS/IPS - Web-filtering services - Application detection and control - Botnet traffic detection and prevention - GEO-IP filtering (geographic based address filtering) - File type filtering (e.g .exe, .bat)	Option	Option	Option
SNMP reporting (web portal)	Option	Option	Option
Flow based reporting (web portal)	Option	Option	Option
Security reporting (PDF)	Option	Option	Option
Security reporting (web portal)	Option	Option	Option
Service levels (SLA)			
Available service levels	Bronze, Silver, Gold and Platinum	Bronze, Silver, Gold and Platinum	Bronze, Silver, Gold and Platinum
Service request and incident management			
Included service requests per month	3 pcs	6 pcs	6 pcs
Included incident requests per month	Unlimited	Unlimited	Unlimited
Performance values			
Statefull firewall throughput	300Mbps	500Mbps	2Gbps
Performance with security services - Antivirus - IDS/IPS - Web-Filtering Services - Application detection and control - Botnet traffic detection and prevention - GEO-IP filtering (geographic based address filtering) - File type filtering (e.g .exe, .bat)	120Mbps	260Mbps	340Mbps
Number of IPsec LAN-to-LAN VPN tunnels	20	30	100
Number of concurrent VPN client users	50	100	200
Recommended for (number of concurrent users)	50 - 150	150 - 300	300- 700
Local log storage for short term use	Not available	Not available	Not available
Explicit HTTP Proxy Feature (max number of concurrent users)	75	150	300
Certifications & Classifications			
EAL4, FIPS 140-2 & ICSA	Yes	Yes	Yes
ICSA Labs (Antivirus, Network FW, IPsec, Network IPS, SSL-TLS)	Yes	Yes	Yes
Firewall has IPv6 support	Yes	Yes	Yes
Log traffic			
Average traffic log amount in 24 hours	2GB	8GB	12GB

The SGW cloud service description defines the following performance values

- Stateful firewall throughput
- Performance with security services enabled
- Number of IPSec LAN-to-LAN VPN tunnels
- Recommended number of concurrent users
- Maximum number of concurrent HTTP proxy users

Of these performance values, only “stateful firewall throughput” and “performance with security services enabled” are expressed in intuitively measurable values (megabits per second). Other performance values in the service description are expressed with the number of recommended or maximum users. These performance values are difficult to interpret; it is not defined, what kind of network load and traffic mix a single user creates (NSP, 2016).

4.2 NSP Services Implemented with Physical Appliances

When a physical appliance is used to produce a service, it is installed in one of the NSP data centers. The network connectivity for the appliance is implemented by creating a VLAN (or VLANs) between the data center firewall and the device. As an example, the process of implementing a physical appliance based service is described below.

1. Appliance is installed in a rack in the data center.
2. Appliance is connected to a data center switch with Ethernet or fiber cable.
3. VLAN (or VLANs) is configured on the appliance, switch and cloud-SGW.
4. IP addresses are configured on the appliance and on the cloud-SGW.

As the example shows, the "switched service" approach requires configurations on three devices (not accounting for possible configurations on redundant devices). Configuration is needed on the physical appliance, data center switch and Cloud-SGW. Figure 24 illustrates the example network topology in this case.

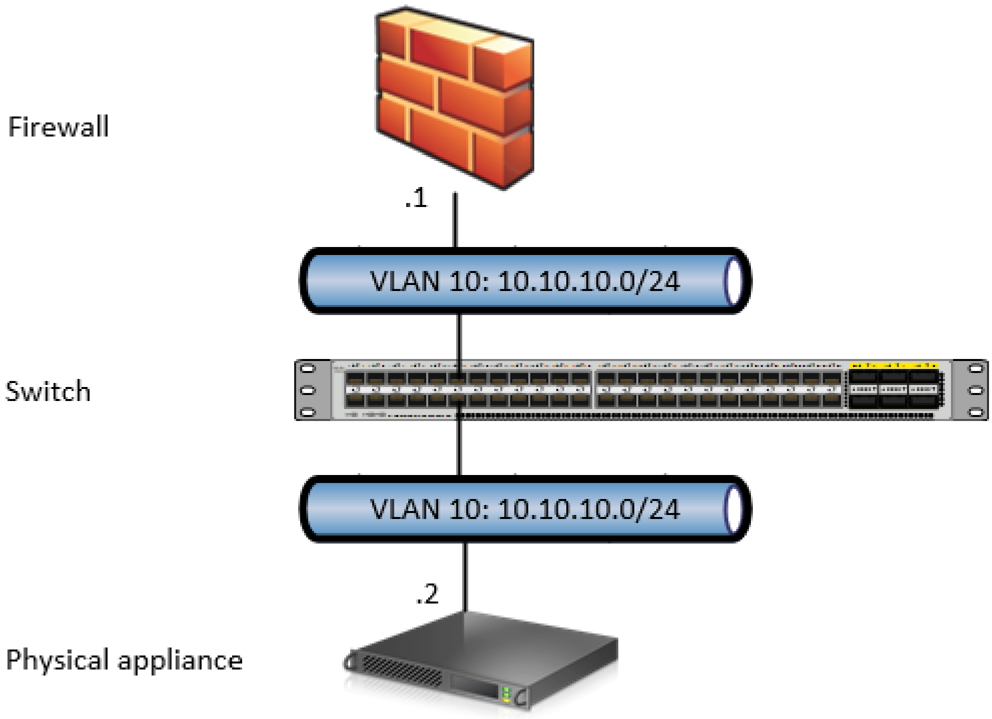


Figure 24 Example network topology with physical appliance

In the example shown in Figure 24:

- The physical appliance is configured with VLAN 10 and IP 10.10.10.2/24.
- Two switch ports are configured with VLAN 10.
- Firewall is configured with VLAN 10 and IP 10.10.10.1/24.

4.3 NSP Services Implemented with Virtual Appliances

Virtual appliances can be located in any data center and they can also migrate transparently from one data center to another. The data centers do not have shared layer 2 infrastructure, meaning that each data center has its own VLAN ranges etc. With this approach, the virtual appliance is configured to send its traffic to the nearest router, which then routes the packets via the NSP core network to a firewall located in the same or in some other data center. This type of service delivery approach offers flexibility, but at the same time requires approximately twice as much configuration work than using only switching to provide connectivity. As an example, the implementation of a new virtual appliance for a customer requires the following steps:

1. Virtual machine is created on the hypervisor host.
2. VLAN(s) are reserved and configured on the hypervisor virtual switch, data center switch and data center router.
3. IP addresses are configured on the VM and the router.
4. Router 1 is configured with a VRF instance
5. Router 2 is configured with a VRF instance.
6. VLANs are reserved and configured on the router 2, switch and firewall.
7. IP addresses are reserved and configured on the destination data center router and cloud-SGW.
8. Firewall policies are configured on the cloud-SGW.

Example network topology is described in Figure 25.

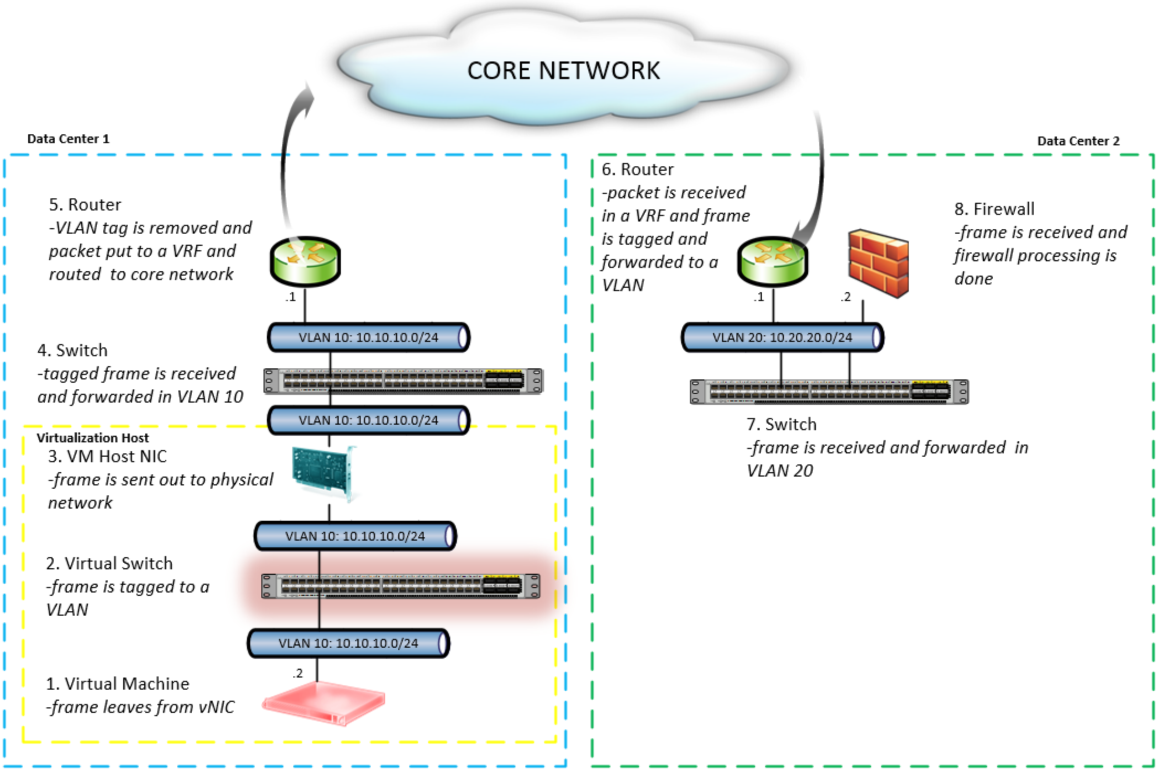

CORE NETWORK
Data Center 1
Data Center 2
5. Router
-VLAN tag is removed and packet put to a VRF and routed to core network
.1
VLAN 10: 10.10.10.0/24
4. Switch
-tagged frame is received and forwarded in VLAN 10
VLAN 10: 10.10.10.0/24
Virtualization Host
3. VM Host NIC
-frame is sent out to physical network
VLAN 10: 10.10.10.0/24
2. Virtual Switch
-frame is tagged to a VLAN
VLAN 10: 10.10.10.0/24
.2
1. Virtual Machine
-frame leaves from vNIC
6. Router
-packet is received in a VRF and frame is tagged and forwarded to a VLAN
.1
.2
8. Firewall
-frame is received and firewall processing is done
VLAN 20: 10.20.20.0/24
7. Switch
-frame is received and forwarded in VLAN 20

Figure 25 Example network topology with virtual appliance

In the example depicted in Figure 25, data center 1 devices are configured as follows:

- The virtual appliance is configured with VLAN 10 and IP 10.10.10.2/24.
- Two virtual switch ports are configured with VLAN 10.
- Two physical switch ports are configured with VLAN 10.
- Router is configured with VLAN 10 and IP 10.10.10.1/24.
- Router is configured with a new VRF where VLAN 10/network 10.10.10.0/24 is placed.

Data center 2 devices are configured as follows:

- Router is configured with corresponding VRF where network 10.10.10.0/24 is placed.
- Router is configured with VLAN 20 and IP 10.20.20.1/24.
- Two physical switch ports are configured with VLAN 20.
- Firewall is configured with VLAN 20 and IP 10.20.20.1/24.
- Dynamic routing is configured between firewall and router.

As the example shows, there are at minimum ten different configuration steps needed to deploy the network connectivity for a VM in this case. The amount of needed configurations is often greater, as in most cases the configuration needs to be duplicated in redundant networking devices.

4.4 Public Cloud Usage Related to NSP Services

There is increasing need to integrate the NSPs Security Gateway services with public cloud solutions, such as Microsoft Azure and Amazon AWS. Customers who are leveraging public cloud solutions are often creating hybrid cloud solutions by connecting their private networks to the public cloud. The connections to public cloud are created either with VPN tunnels or with dedicated circuits provided by the cloud provider.

VPN tunnel connection to cloud provider infrastructure is a common method to provide secure connection to public. When the customer is using NSP Security Gateway service the VPN tunnel is usually created from Security Gateway to the cloud provider in question (i.e. Azure, AWS, Google or Rackspace).

In some cases, the customers require a more reliable connection to the cloud provider than a VPN tunnel. This is the case if the customer has significant investment in public cloud infrastructure and has decided that connectivity to public cloud is mission critical. In that case, the customer can purchase a dedicated circuit from the provider. With Security Gateway Cloud - service it is possible to create a dedicated circuit to

- Amazon Web Services (Direct Connect).
- Microsoft Azure (Express Route).

4.5 Customer Demand for Virtual Firewalls

In addition to VPN tunnel and dedicated circuit services there is growing interest from customers to start using managed firewall services to protect private and public cloud deployments. In answer to the customer demand, it has been decided to start productizing Security Gateway service implemented with Virtual Appliances.

4.6 Conclusions

Based on the current state analysis it was found that in the NSP data centers the network connectivity between virtual appliances and cloud-SGWs is implemented with routing. The traffic is routed to a physical network firewall that can reside in the same or in another data center as the originating device. Routing approach to connectivity requires a large amount of configuration work in routers, switches and firewalls.

In theory, implementing firewall services on a virtual platform would reduce the number of required device configurations to make the service operational. This would speed up the service implementation time and improve service delivery times. Reducing delivery times is important, because with shorter delivery times the billing from customer could be started sooner. The customers are already leveraging hybrid cloud solutions and there is growing demand to start providing Security Gateway services inside public cloud infrastructures. The virtual firewall service productization has started and the goal is to have a ready service by the end of year 2016. The timetable of the product development in relation to this Master's thesis is shown in Figure 26.

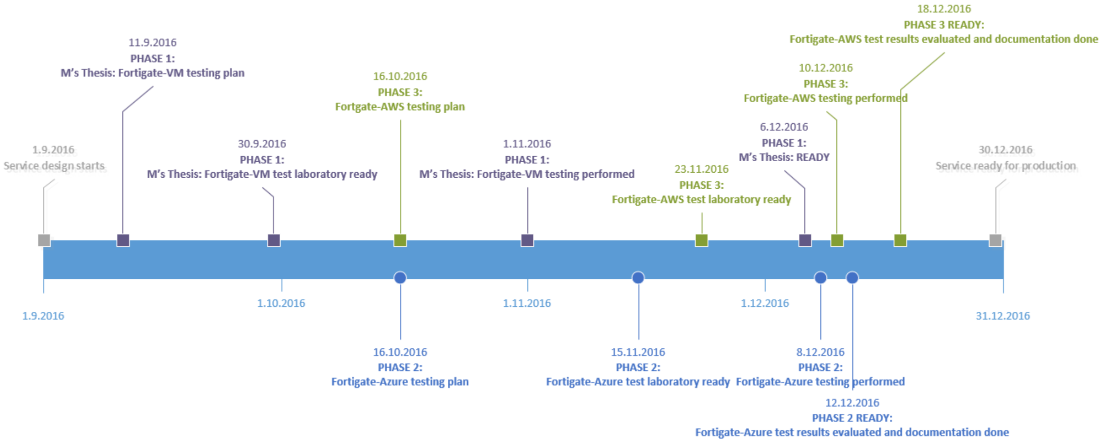

18.12.2016
PHASE 3 READY:
Fortigate-AWS test results evaluated and documentation done
11.9.2016
PHASE 1:
M's Thesis: Fortigate-VM testing plan
16.10.2016
PHASE 3:
Fortgate-AWS testing plan
10.12.2016
PHASE 3:
Fortigate-AWS testing performed
6.12.2016
PHASE 1:
M's Thesis: READY
1.9.2016
Service design starts
30.9.2016
PHASE 1:
M's Thesis: Fortigate-VM test laboratory ready
1.11.2016
PHASE 1:
M's Thesis: Fortigate-VM testing performed
23.11.2016
PHASE 3:
Fortigate-AWS test laboratory ready
30.12.2016
Service ready for production
1.9.2016
1.10.2016
1.11.2016
1.12.2016
31.12.2016
16.10.2016
PHASE 2:
Fortigate-Azure testing plan
15.11.2016
PHASE 2:
Fortigate-Azure test laboratory ready
8.12.2016
PHASE 2:
Fortigate-Azure testing performed
12.12.2016
PHASE 2 READY:
Fortigate-Azure test results evaluated and documentation done

Figure 26 Service development timetable in relation with Master's thesis

In phase one of productizing the virtual firewall based service, the technical feasibility (performance) of virtual firewall technology in a private cloud environment is evaluated. This evaluation will form be the scope of this Master's thesis. In the next productization phases two and three, similar virtual firewall testing environments are created in Microsoft Azure and in Amazon Web Services public clouds. The goal is to have the performance tests developed in phase 1 to be then reproduced in public cloud deployments.

The obtained results will then be used to create the service specifications in service design phase.

5 Solution Needs Analysis

The solution has the requirement to measure virtual firewall network performance. To evaluate the performance level of virtual firewall technology, the study was conducted in the following order:

1. Network performance testing metrics are decided.
2. Testing methodology is decided.
3. Network topology is designed.
4. Test laboratory is procured and built.
5. Testing tools are decided.
6. Network performance testing is performed
7. Results are reported and analyzed.

5.1 Network Performance Test Metrics

As demonstrated in Chapter *3.5 Network Performance Testing,* there are various methods for testing network and firewall performance. To compare VF and cloud-SGW performance we must firstly define performance values that are evaluated. The service description for the cloud SGW describes the following performance values (see Tables 4 and 5):

1. Stateful firewall throughput: 100 Mbps to 2 Gbps.
2. Stateful firewall throughput with security services enabled: 15 Mbps to 340 Mbps.
3. Number of IPsec LAN-to-LAN VPN tunnels: 2 to 100.
4. Number of concurrent VPN client users: 5 to 200.
5. Number of concurrent users: 1 to 700.
6. Number of concurrent HTTP proxy users: 0 to 300.

Of these values, only the firewall throughput values (1 and 2) are concrete and easily measurable. User based metrics are difficult to define because all users have their unique behaviour and network usage varies greatly between users. In addition, it is time consuming to create tests that would mimic the connections from hundreds of concurrent users. To keep the thesis scope manageable it was decided to test four performance values shown in Table 6 below.

Table 6 Testing metrics

Service Description Performance Value	Test Metric
Stateful firewall throughput	1. Firewall throughput
Firewall throughput /w sec. services	2. Firewall throughput with firewall security services enabled
# of IPsec LAN-to-LAN VPN tunnels	3. IPsec VPN throughput
# of concurrent VPN client users	4. SSL VPN throughput

These throughput tests provide concrete data that can be presented to customers in an understandable way.

5.2 Testing Tools

There are several tools for testing network throughput. A popular test method is downloading a file from server to a client using i.e. FTP. This method gives indication of the network performance, but has some issues because there are often unknown variables affecting the results. These variables are, for example, hard disk delays and OS queueing mechanisms. More accurate way of measuring throughput is using tools designed for the task (Firewall.cx, 2013). Examples of such tools are HTTPing, IxChariot and IPerf.

After researching the testing tools, it was decided to use a freeware tool called **iPerf3** for throughput measurements. Iperf3 is network performance testing tool developed by ESnet and Lawrence Berkeley National Laboratory. IPerf3 can be used to determine the maximum bandwidth that can be utilized in IP networks. With the tool it is possible to measure TCP, SCTP and UDP bandwidth and it allows several testing parameters to be modified (such as timers, buffers and protocols) (Dugan, et al., 2016).

To test firewall throughput with security services enabled, it was decided that test traffic needs to be “real” rather than iPerf generated. Therefore, it was decided to test firewall in this scenario by performing **FTP** file transfers several times and calculating the average from these transfers.

5.3 Test Methodology

To establish a baseline performance level the throughput between physical servers will first be measured. In the next phase, baseline throughput is measured for VM to VM traffic. After baseline performance level is established, the Virtual Firewall is introduced into the test network and tests are run according to test metrics list.
The tests were run in the following scenarios:

Baseline measurements:

1. Physical server to physical server throughput.
2. VM client to VM server throughput.

Virtual firewall measurements:

1. VM to VM stateful firewall throughput through VF.
2. VM to VM stateful firewall throughput through VF, security services enabled.
3. VM to VM IPsec throughput through VF.
4. VM to VM SSL VPN throughput through VF.

NSP Cloud-SGW service description does not mention frame size, NAT or size of the firewall rule set. To deal with this fact, the following restrictions were decided on:

- As VM to VM traffic does not necessarily go through physical interfaces it was decided to test with only default Ethernet frame size (1518 bytes).
- Virtual firewall measurement #1 will be run with three firewall rule set sizes, 1 rule, 100 rules and 1000 rules.
- Virtual firewall measurement #1 will be run first with NAT enabled and then with NAT disabled.
- Virtual firewall measurement #2 will be performed with FTP transfer, rather than with iPerf 3. This decision was made to create real application traffic the UTM engine can inspect.
- Virtual firewall IPSec VPN throughput will be measured with three encryption mechanisms, AES 256, AES 128 and 3 DES.
- Virtual firewall measurements 3 and 4 will be run with minimal firewall rule set and NAT disabled.

These restrictions were made to narrow the scope of this thesis. Using broader scope was deemed not possible due to timetable limitations.

5.4 Network Topology

The key requirement in designing the laboratory was keeping the costs to a minimum. Therefore, no new hardware could be obtained and used software and licensing had to be designed with free options when possible.

According to the test methodology, the traffic flows to be tested were:

1. Between two physical servers.
2. Between two virtual machines.
3. Between two virtual machines and virtual firewall.
4. Between two virtual machines and two virtual firewalls (IP-sec VPN throughput).

The simplest way all traffic flows can be tested, is a setup where two physical servers are connected to a network switch (as seen in Figure 27). In the RFC 3511 test methodology this topology is called “Dual-Homed”. (Hickman, et al., 2003, p. 4).

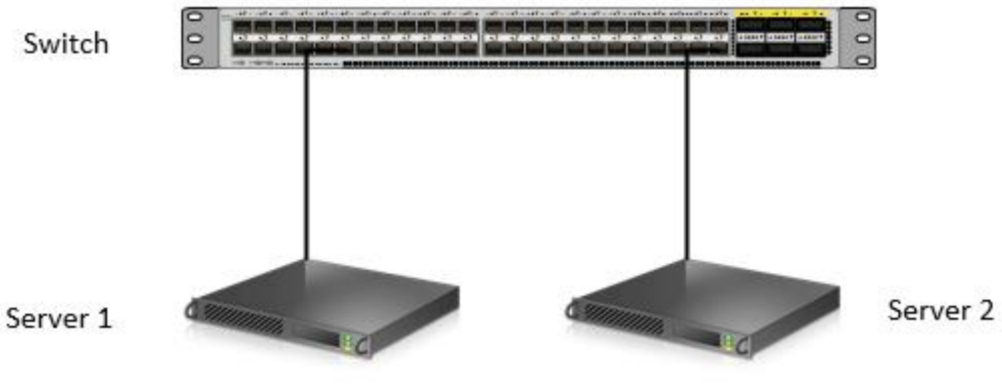


Figure 27 Laboratory network

It was decided that the servers and switch in the laboratory network are to be connected with gigabit Ethernet interfaces. In effect, this decision limits the maximum theoretical throughput between physical servers to 1000 megabits per second.

5.5 Test Laboratory

After laboratory topology and hardware requirements were decided on, the project started by the acquisition of the hardware. After research, it was found that there were suitable HP ProLiant blade servers available that could be used. The servers were already connected to a switch and were remotely manageable. Hardware details are illustrated in Table 7.

Table 7 Laboratory hardware details

	Server 1	**Server 2**	**Switch**
Hardware	HP ProLiant BL460c G7	HP ProLiant BL460c G7	Cisco WS-CBS3120X-S
CPU	Intel Xeon E5620 @ 2.4 GHz	Intel Xeon E5620 @ 2.4 GHz	PowerPC405
RAM	24 GB	24 GB	256 MB
Disk	136 GB	136 GB	64 MB
NIC	1 Gb/s	1 Gb/s	2 x 10 Gb/s + 26 x 1 Gb/s
OS	MS Hyper-V Server 2012 R2	MS Windows Server 2012 R2 Standard	IOS 12.2

As Table 7 shows, each server had rather limited disk space (136 GB). This puts a limitation to the amount of virtual machines that can be installed, as modern operating systems often require a minimum of 10 GB of disk space. As an example, Windows 8.1 OS requires 20 GB of disk space (64-bit version). (Microsoft Inc, 2016).

5.5.1 Virtualization Platform and Physical Server OS

Arguably, there are two viable choices for an enterprise virtualization platform, VMWare vSphere or Microsoft Hyper-V (Gartner, Inc., 2016). The virtualization platform for this study was chosen to be **Microsoft Hyper-V**. The reason was that Hyper-V was easily obtainable with no costs. Server 1 was installed with Hyper-V Server 2012 and server 2 with Windows Server 2012.

5.5.2 Virtual Firewall

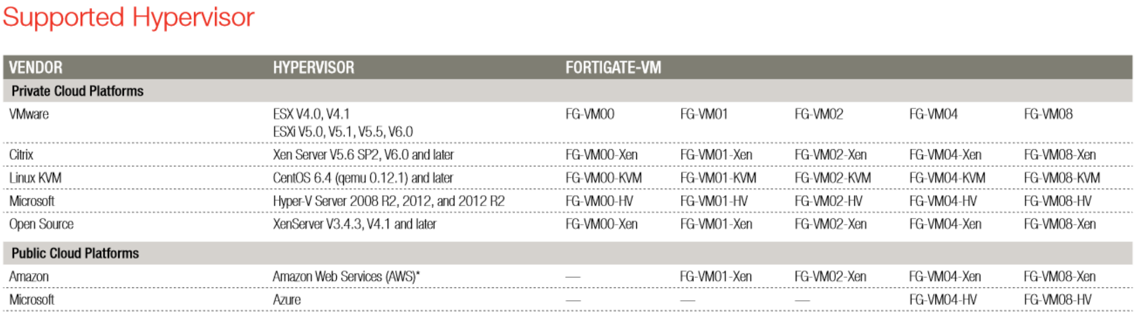
Several vendors have virtual firewall products. Some of these are:

- Cisco ASAv.
- Juniper Networks vGW.
- VMWare vShield.

- Vyatta Network OS.
- Fortinet Fortigate-VM.
- Palo Alto VM-Series.
- Checkpoint vSEC.

A majority of firewalls maintained by the NSP are Fortinet Fortigates. For this reason the virtual firewall brand was decided to be Fortinet's **Fortigate-VM**. The Fortigate-VM is available for most popular private cloud hypervisors and is also available in Microsoft Azure and Amazon AWS public clouds. Table 8 displays the available Fortigate-VM options.

Table 8 Fortigate-VM supported hypervisors (Fortinet Inc., 2015)



Supported Hypervisor

VENDOR	HYPERVISOR	FORTIGATE-VM				
Private Cloud Platforms						
VMware	ESX V4.0, V4.1 ESXi V5.0, V5.1, V5.5, V6.0	FG-VM00	FG-VM01	FG-VM02	FG-VM04	FG-VM08
Citrix	Xen Server V5.6 SP2, V6.0 and later	FG-VM00-Xen	FG-VM01-Xen	FG-VM02-Xen	FG-VM04-Xen	FG-VM08-Xen
Linux KVM	CentOS 6.4 (qemu 0.12.1) and later	FG-VM00-KVM	FG-VM01-KVM	FG-VM02-KVM	FG-VM04-KVM	FG-VM08-KVM
Microsoft	Hyper-V Server 2008 R2, 2012, and 2012 R2	FG-VM00-HV	FG-VM01-HV	FG-VM02-HV	FG-VM04-HV	FG-VM08-HV
Open Source	XenServer V3.4.3, V4.1 and later	FG-VM00-Xen	FG-VM01-Xen	FG-VM02-Xen	FG-VM04-Xen	FG-VM08-Xen
Public Cloud Platforms						
Amazon	Amazon Web Services (AWS)*	—	FG-VM01-Xen	FG-VM02-Xen	FG-VM04-Xen	FG-VM08-Xen
Microsoft	Azure	—	—	—	FG-VM04-HV	FG-VM08-HV

The Fortigate-VM demo download is available for Fortinet partners. The demo version however, has considerable restrictions compared to actual product. The demo license is valid for only 15 days since installation, supports only low encryption and maximum of 1 CPU and 1 GB of memory (Fortinet, Inc., 2015, pp. 7-8).

To overcome these restrictions Fortinet was contacted and six unlimited 2-month Fortigate-VM licenses were obtained for the study. It was decided that virtual firewalls in this study would be configured a single virtual CPU and with two gigabytes of memory.

5.5.3 VM client / server OS

The Hyper-V server was planned to run with three virtual machines, VM client, VM server and the Fortigate-VM. The physical servers had limited disk space (135 GB on each server) and therefore it was decided to use Linux as the VM OS, as Windows in most cases requires a significantly larger amount of disk space. The Linux distribution chosen was **CentOS 7,** as it is redeemed in the community as one of the best free distributions available (Bhartiya, 2016).

6 Performance Test Measurements

Network baseline performance was established by running throughput tests in the following scenarios:

1. Physical server to physical server.
2. Virtual machine to virtual machine.

The tests were run with the following iPerf3 options set:

1. Duration: 900 seconds
2. Traffic type: TCP
3. Parallel streams: 10
4. Reporting format: Mbits/sec
5. Reporting interval: every 10 seconds
6. Frame size: 1518 bytes

Option 1, test duration of 15 minutes was chosen as it is one of the defined test durations in the ITU-T Y1564 service activation test recommendation. (ITU-T, 2016, p. 17). Option 2, traffic type, was chosen to be TCP because firewall testing methodology described in the RFC 3511 document employs HTTP, which is a TCP based protocol (Hickman, et al., 2003, p. 4). As option 3, parallel streams, it was decided to use ten parallel TCP streams after testing with several stream numbers that performance did not improve in laboratory network after more than ten streams were used. Reporting options (4 and 5) were chosen in order to keep the report files readable and the reports decent sized. Finally, option 6, frame size used for testing was chosen to be 1518 bytes. This was done in order to keep amount of needed test runs manageable.

6.1 Baseline 1: Server to Server Throughput

The first baseline for the network performance is the throughput between physical servers. The servers are connected directly to a network switch and there are no other components in between (see Figure 28).

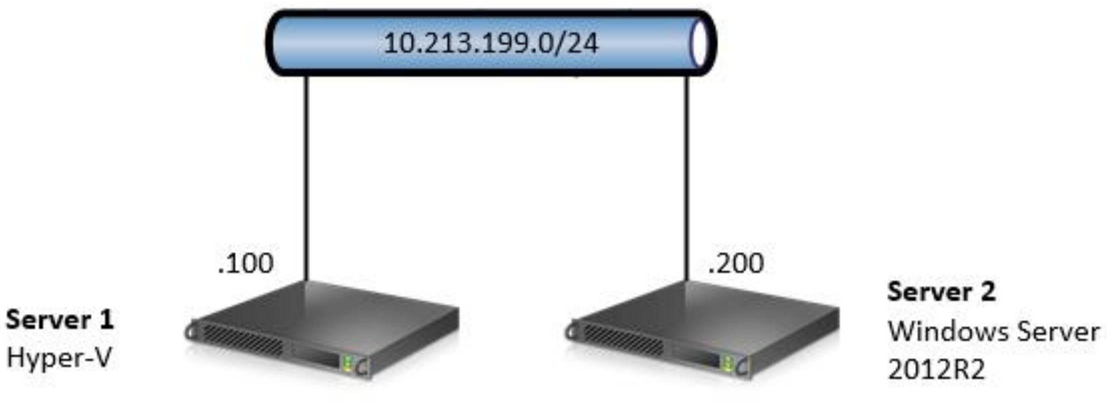
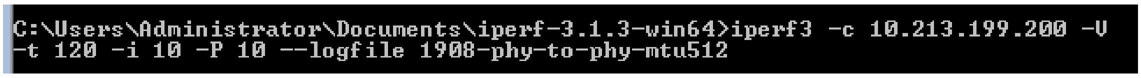


Figure 28 Laboratory network topology

Because the needed physical components (two servers and one network switch) were already installed, the laboratory implementation began with installing the operating systems on the servers. Server 1 was installed with Microsoft Hyper-V hypervisor OS and server 2 with Windows Server 2012 R2. Then, to have network connectivity between servers, both servers were configured with IP addresses from network 10.213.199.0/24.

To run the throughput tests iPerf3 was installed on both servers. Server 1 was running iPerf3 in server mode and then on server 2 iPerf3 test was run with the following command “iperf3 -c IP -V -t 900 -i 10 -P 10 –logfile logexample”. Example of the command is visible in Figure 29.



```
C:\Users\Administrator\Documents\iperf-3.1.3-win64>iperf3 -c 10.213.199.200 -V
-t 120 -i 10 -P 10 --logfile 1908-phy-to-phy-mtu512
```

Figure 29 iPerf 3 test command example

The results of the test were recorded in Table 9.

Table 9 Baseline 1, server to server throughput

Physical to physical	Throughput Mbit/s
Server 2012 – Server 2012	**949**

It can be seen from the achieved result that baseline throughput is close to the theoretical maximum speed of the link (1 Gb/s).

6.2 Baseline 2: VM to VM Throughput

VM to VM throughput was measured by connecting client and server VMs to the same vSwitch and by running iPerf. Results of VM to VM throughput measurements are seen in Table 10.

Table 10 Baseline 2, VM to VM throughput

VM to VM	Throughput Mbit/s
CentOS1 – CentOS2	**1930**

The obtained result show the throughput between virtual machines running on the same host is over two times more than throughput over physical network.

6.3 Test 1: Stateful Firewall Throughput

To test the firewall throughput, the network topology needed to be such that traffic between endpoints traverses the virtual firewall. To achieve this, three virtual machine instances were needed, two VMs and one virtual firewall. In addition to virtual machine instances, an internal virtual switch was needed to segment the traffic and an external switch was needed to have a management connection to the VF. The two VM: s were placed in separate VLANs and traffic from each VLAN was terminated on virtual firewall. The test topology is depicted in Figure 30.

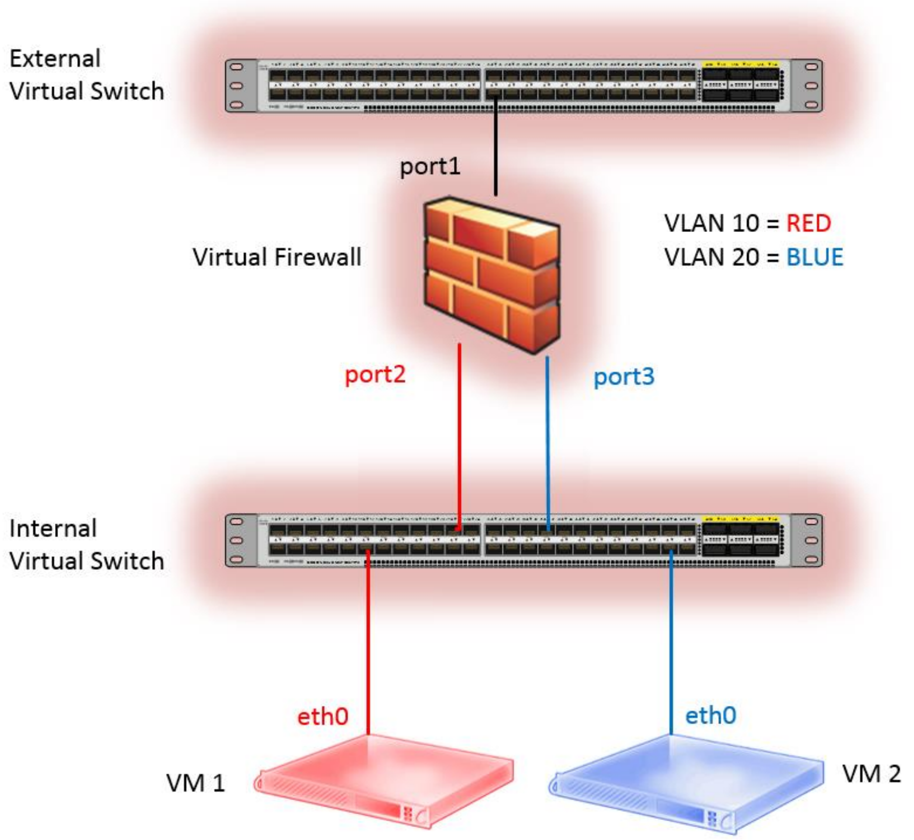
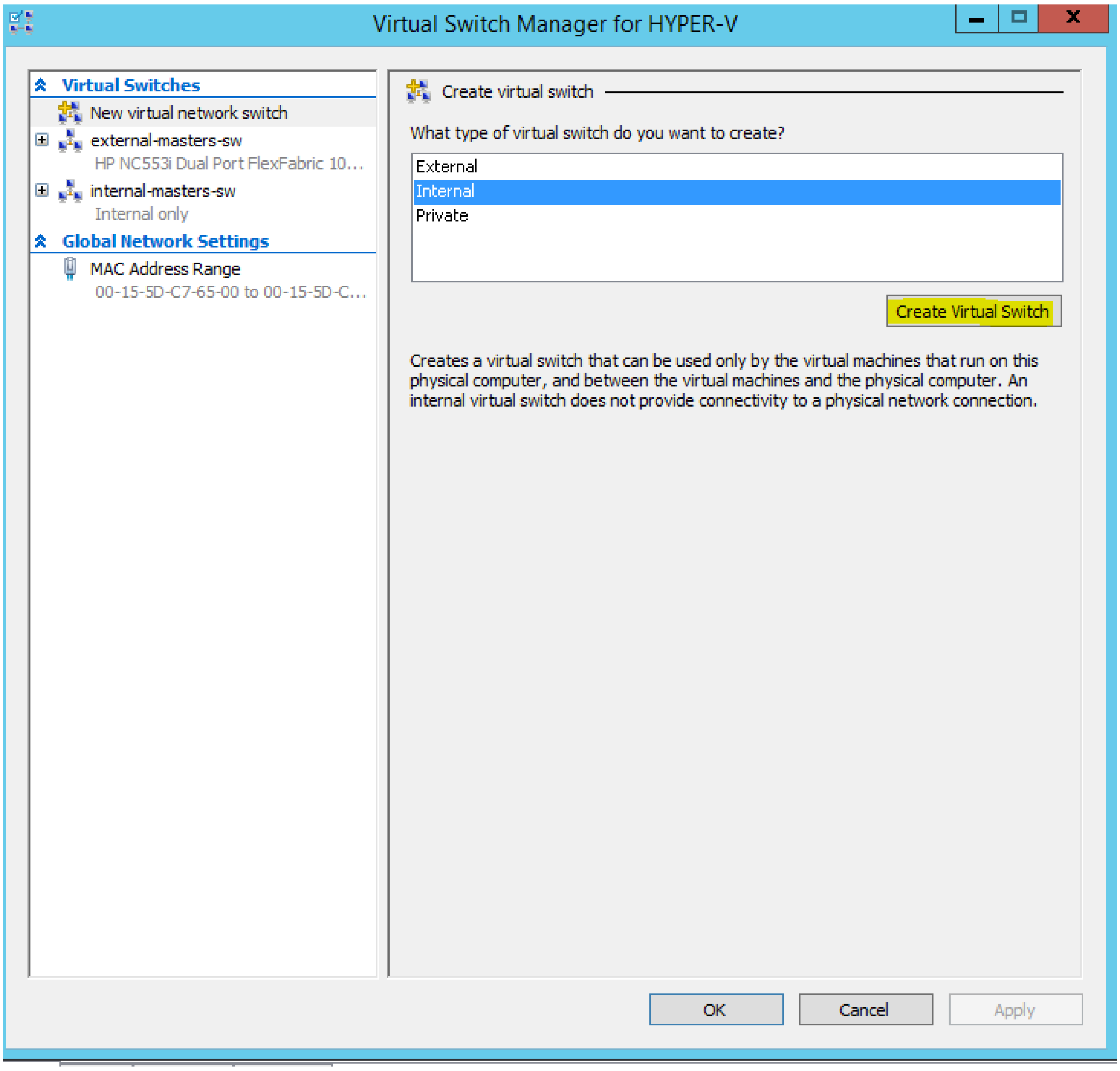


Figure 30 Firewall throughput test topology

To create the test network, firstly, an external virtual switch was created on Hyper-V host. The external switch was named as “external-masters-sw”. Then, a second virtual switch was created and the switch type was set as “Internal”. The virtual switch was named as “internal-masters-sw”. Figure 31 shows the creation of the internal virtual switch.

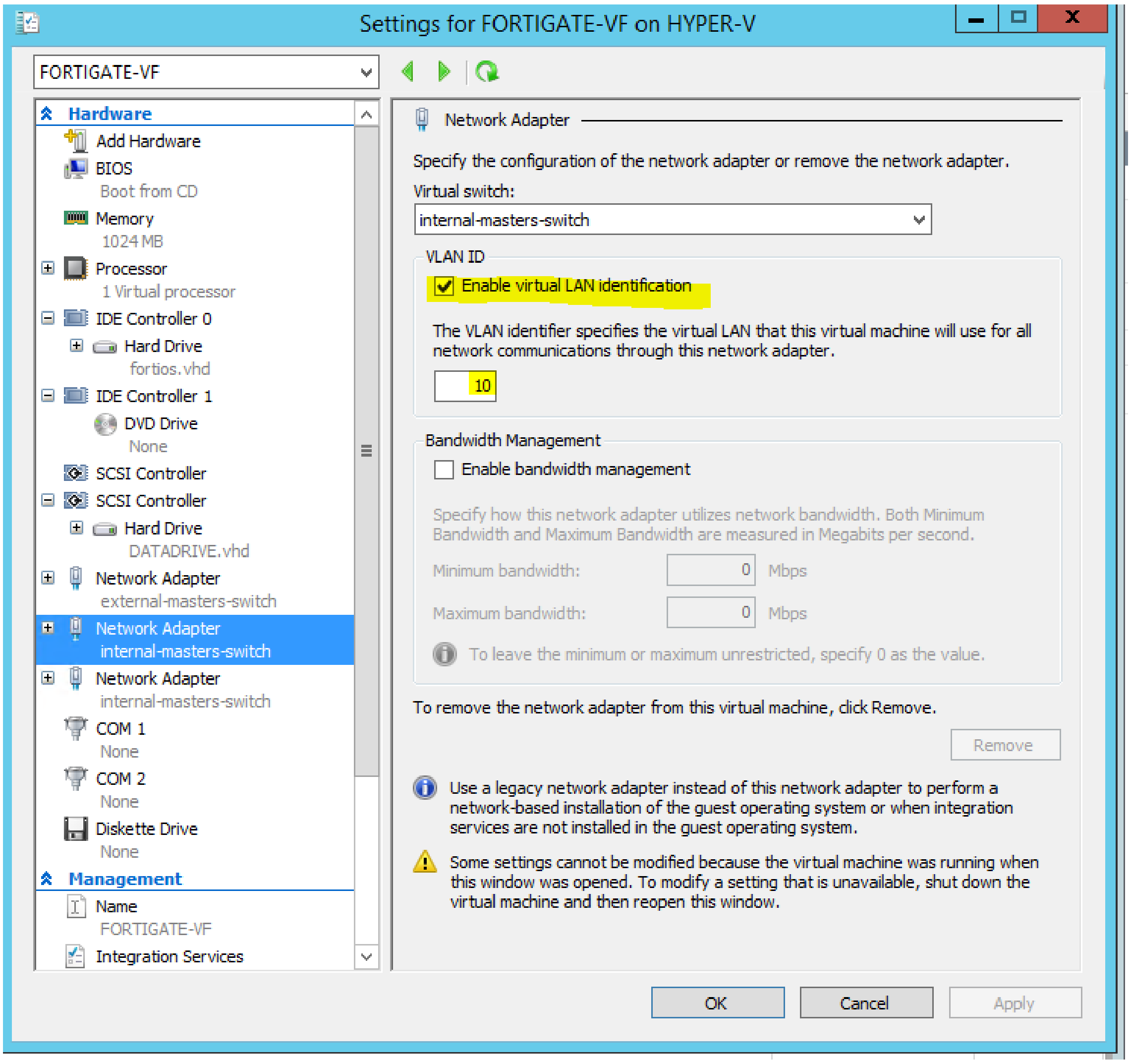


What type of virtual switch do you want to create?

Creates a virtual switch that can be used only by the virtual machines that run on this physical computer, and between the virtual machines and the physical computer. An internal virtual switch does not provide connectivity to a physical network connection.

Figure 31 Creating a new virtual switch

Next, the virtual firewall was deployed and its interface “port 1” was connected to “external-masters-sw”. Interfaces “port 2” and “port 3” were connected to “internal-masters-sw” and each interface was consequently tagged with a VLAN. Port 2 was tagged in VLAN 10 and port 3 in VLAN 20. Figure 32 illustrates the virtual firewall port configuration.



Network Adapter

Specify the configuration of the network adapter or remove the network adapter.

Virtual switch: internal-masters-switch

VLAN ID

☑ Enable virtual LAN identification

The VLAN identifier specifies the virtual LAN that this virtual machine will use for all network communications through this network adapter.

10

Bandwidth Management

□ Enable bandwidth management

Specify how this network adapter utilizes network bandwidth. Both Minimum Bandwidth and Maximum Bandwidth are measured in Megabits per second.

Minimum bandwidth: 0 Mbps

Maximum bandwidth: 0 Mbps

To leave the minimum or maximum unrestricted, specify 0 as the value.

To remove the network adapter from this virtual machine, click Remove.

Use a legacy network adapter instead of this network adapter to perform a network-based installation of the guest operating system or when integration services are not installed in the guest operating system.

Some settings cannot be modified because the virtual machine was running when this window was opened. To modify a setting that is unavailable, shut down the virtual machine and then reopen this window.

Figure 32 Virtual firewall port configuration

After virtual firewall was deployed, two CentOS Linux virtual machines were created and their interfaces were connected to the same “internal-masters-switch” as the firewall. First VMs network adapter “eth0” was tagged in VLAN 10 and second VM’s network adapter “eth0” was tagged in VLAN 20.

In order to route the network traffic via virtual firewall, the VMs were configured with IP addresses on different subnets: 192.168.199.50/24 in VM 1 and 192.168.200.50/24 in VM 2. Firewall ports were configured with IP address .1 in each subnet. Figure 32 shows the network topology in layer 3.

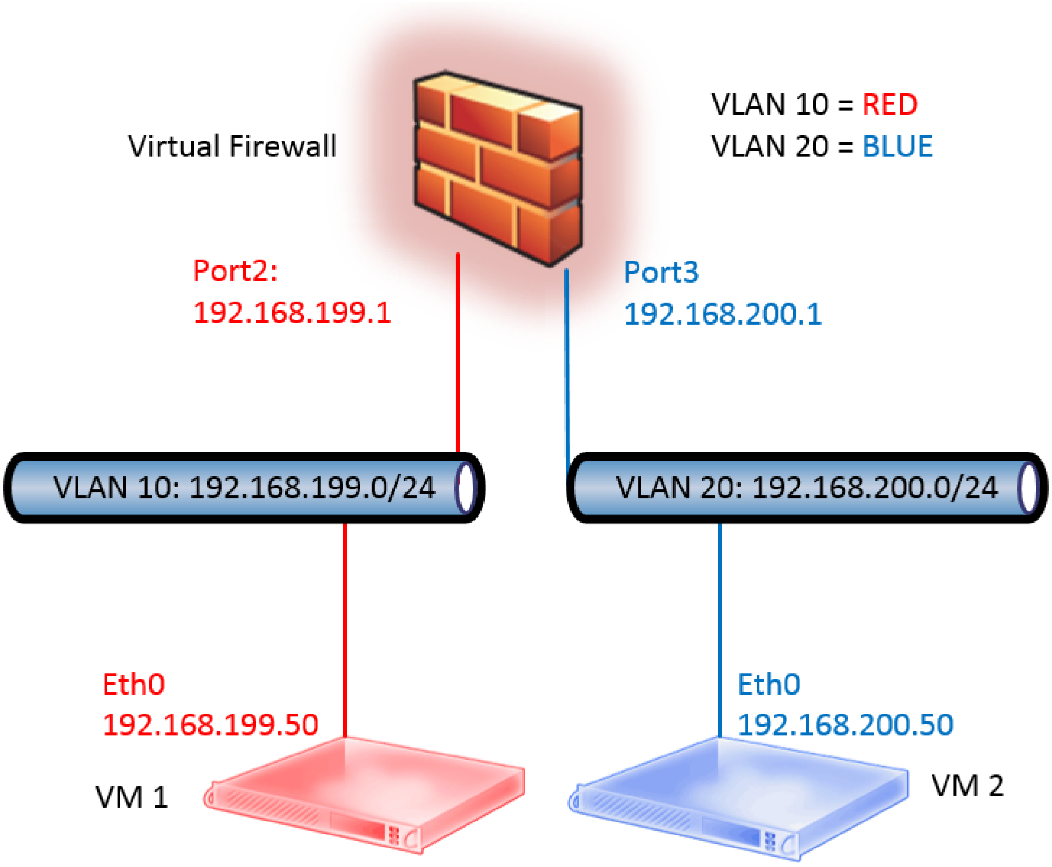
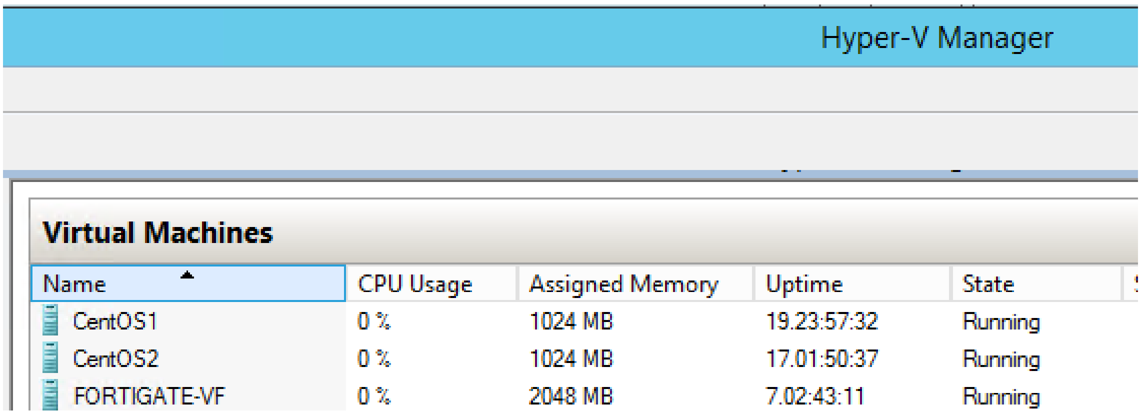


Figure 33 Firewall throughput test topology in layer 3

On Hyper-V platform there were now three virtual machines running as shown in Figure 34.



Hyper-V Manager

Virtual Machines

Name	CPU Usage	Assigned Memory	Uptime	State
CentOS1	0 %	1024 MB	19.23:57:32	Running
CentOS2	0 %	1024 MB	17.01:50:37	Running
FORTIGATE-VF	0 %	2048 MB	7.02:43:11	Running

Figure 34 Hyper-V virtual machines for test 1

The firewall throughput was tested by running iPerf3 server on VM2 and by running iPerf3 client on VM1. The throughput was measured with three policy set sizes, 1, 100 and 1000 policies. All tests were also ran with NAT enabled and NAT disabled. The results are recorded in Table 12:

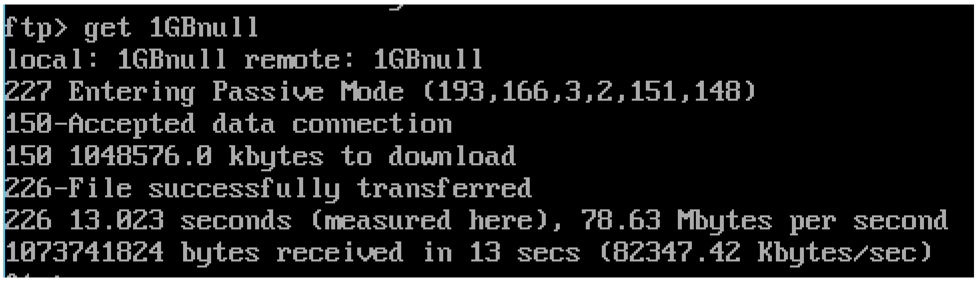
Table 11 Firewall throughput with 1, 100 and 1000 policies

Test	**no NAT (Mb/s)**	**NAT (Mb/s)**
Stateful firewall throughput 1 policy	**1900**	**1890**
Stateful firewall throughput 100 policy	**1890**	**1890**
Stateful firewall throughput 1000 policy	**1890**	**1890**

It should be noted that despite the fact that throughput stayed on a high level in this testing phase, the virtual firewall GUI slowed down significantly and was nearly unusable when there were 100 or more policies in the policy set. GUI performance was also tested with a virtual firewall having twice the resources (two vCPUs, 4 GBs of memory), but there was no improvement. This issue is suspected to be related to the firewall software version used (5.4.1), but it was not tested with other software version.

6.4 Test 2: Stateful Firewall Throughput with Security Services Enabled

In the second test, FTP download was tested with security services (UTM) enabled and disabled. The UTM options used were Antivirus, Web Filter, Application Control and IPS. All UTM profiles were running with default settings (see Appendix 1, UTM Options). Testing was performed with single firewall policy and with NAT enabled. Network topology in the second test was identical to the first test. The test was performed from VM1 by connecting with FTP to ftp.funet.fi and by downloading a file with the size of 1 Gigabyte. Example of a download is visible in Figure 35.



```
ftp> get 1GBnull
local: 1GBnull remote: 1GBnull
227 Entering Passive Mode (193,166,3,2,151,148)
150-Accepted data connection
150 1048576.0 kbytes to download
226-File successfully transferred
226 13.023 seconds (measured here), 78.63 Mbytes per second
1073741824 bytes received in 13 secs (82347.42 Kbytes/sec)
```

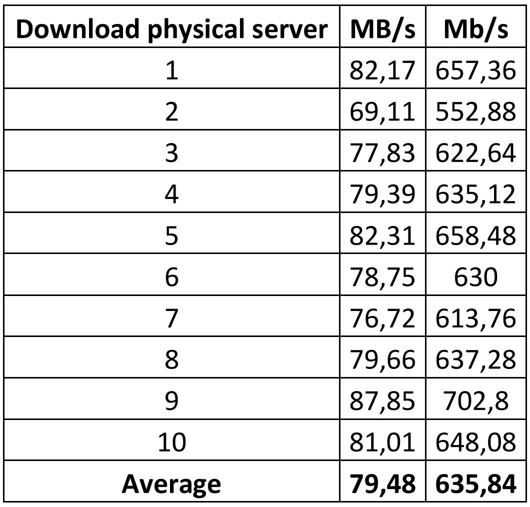
Figure 35 FTP download

The download was performed ten times. As seen in Figure 36, the download speed was reported by the FTP client. The download speed average was calculated in three scenarios:

- Physical server download speed.
- VM download speed through VF.
- VM download speed through VF with security services (UTM) enabled.

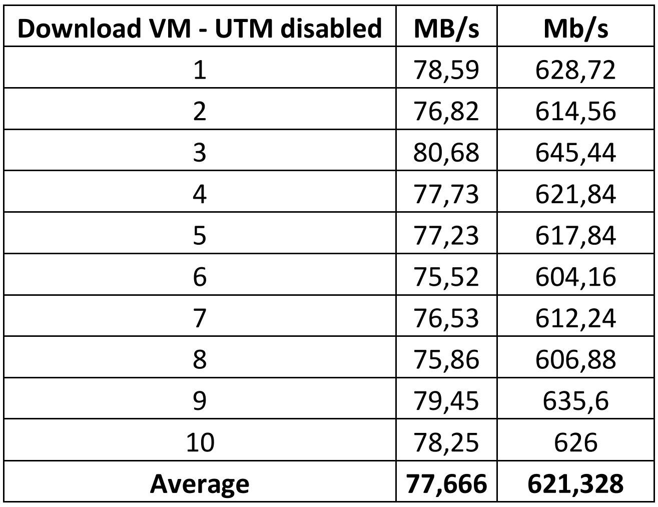
The results were recorded in Tables 12-14 and the average download speeds in each case were calculated.

Table 12 Baseline for FTP transfer



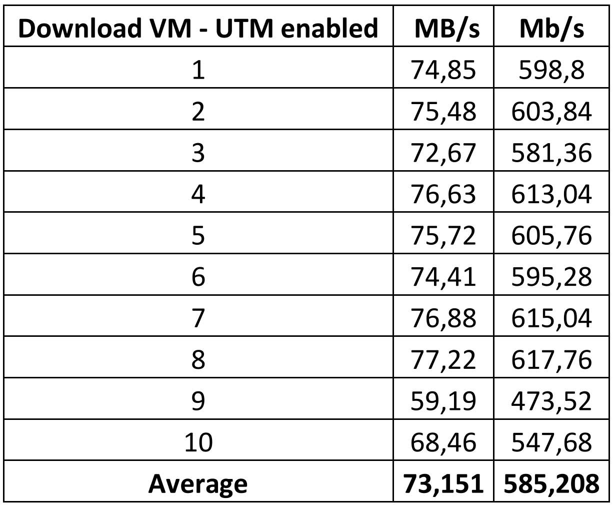
Download physical server	MB/s	Mb/s
1	82,17	657,36
2	69,11	552,88
3	77,83	622,64
4	79,39	635,12
5	82,31	658,48
6	78,75	630
7	76,72	613,76
8	79,66	637,28
9	87,85	702,8
10	81,01	648,08
Average	**79,48**	**635,84**

Table 13 VM FTP transfer throughput



Download VM - UTM disabled	MB/s	Mb/s
1	78,59	628,72
2	76,82	614,56
3	80,68	645,44
4	77,73	621,84
5	77,23	617,84
6	75,52	604,16
7	76,53	612,24
8	75,86	606,88
9	79,45	635,6
10	78,25	626
Average	**77,666**	**621,328**

Table 14 VM FTP transfer throughput with security services enabled



Download VM - UTM enabled	MB/s	Mb/s
1	74,85	598,8
2	75,48	603,84
3	72,67	581,36
4	76,63	613,04
5	75,72	605,76
6	74,41	595,28
7	76,88	615,04
8	77,22	617,76
9	59,19	473,52
10	68,46	547,68
Average	**73,151**	**585,208**

The results indicate expectedly, that for a virtual host the throughput is less than for a physical host, and that the throughput decreases even more when UTM options are enabled.

6.5 Test 3: IPSec VPN Throughput

To test the IPSec VPN throughput, a second virtual firewall was installed on the Hyper-V host. The configuration was identical with first VF, except for IP addressing and VLAN numbering. IPSec VPN test topology in physical layer is described in Figure 36.

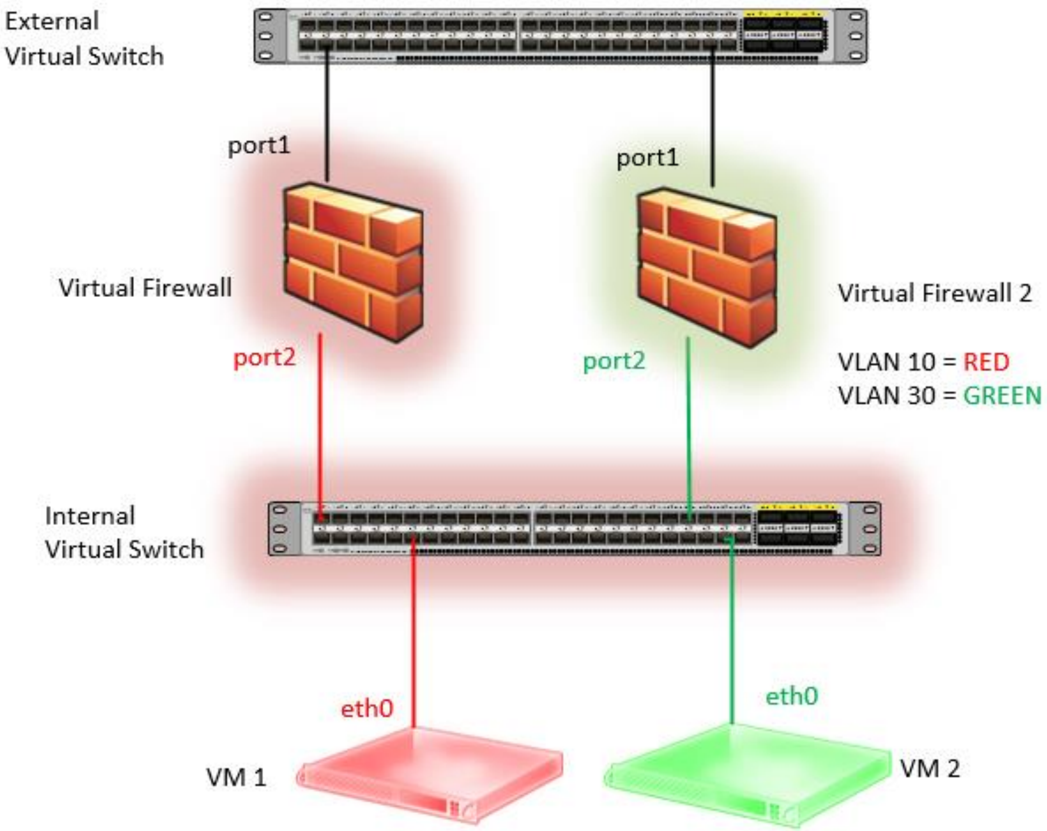


Figure 36 IPSec VPN test topology, physical layer

A logical level description of the topology is seen in Figure 37.

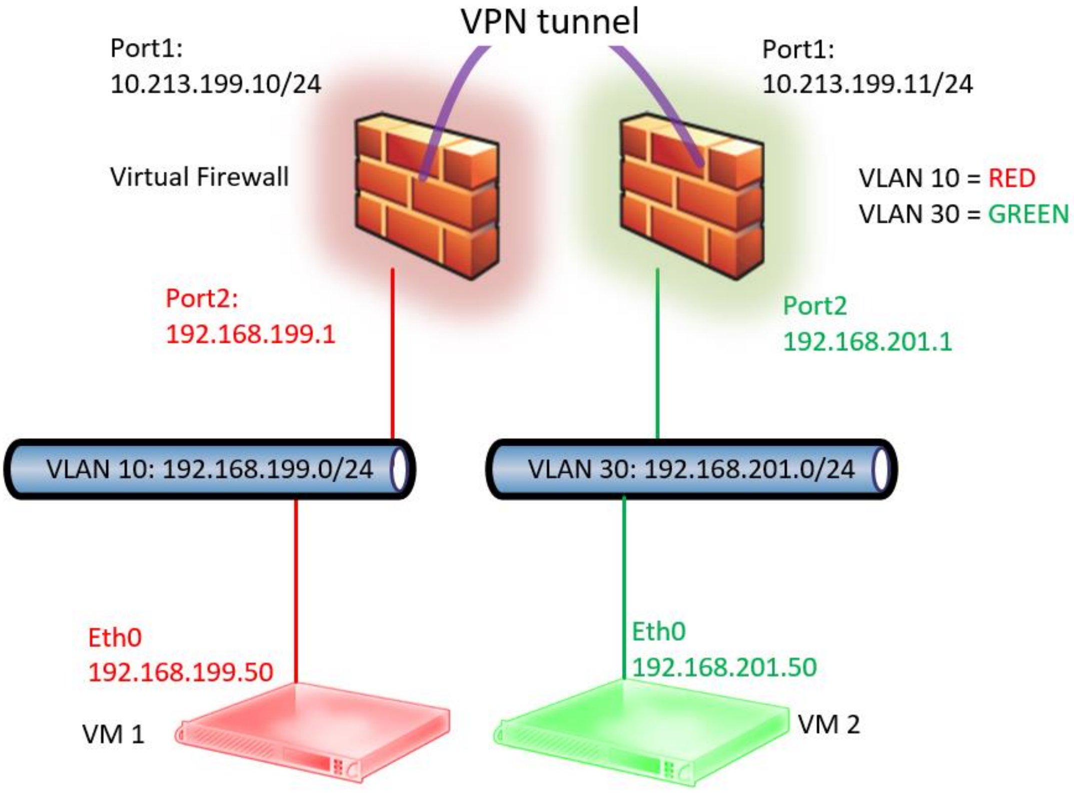


Figure 37 IPSec VPN test topology

In order to move VM 2 behind VF 2 the following configuration changes were made:

- VM 2 interface “Eth0” was reconfigured with IP address 192.168.201.50/24.
- VM 2 interface was tagged to VLAN 30 on “internal-masters-sw”.

After reconfigurations the throughput was measured with iPerf3 between VM 1 (client) and VM 2 (server). The test was run three times, each time the VPN tunnel was configured with different encryption/authentication combinations. The combinations used were: AES256/SHA256, AES128/SHA256 and 3DES/SHA1. Test results for each test are recorded in Table 13:

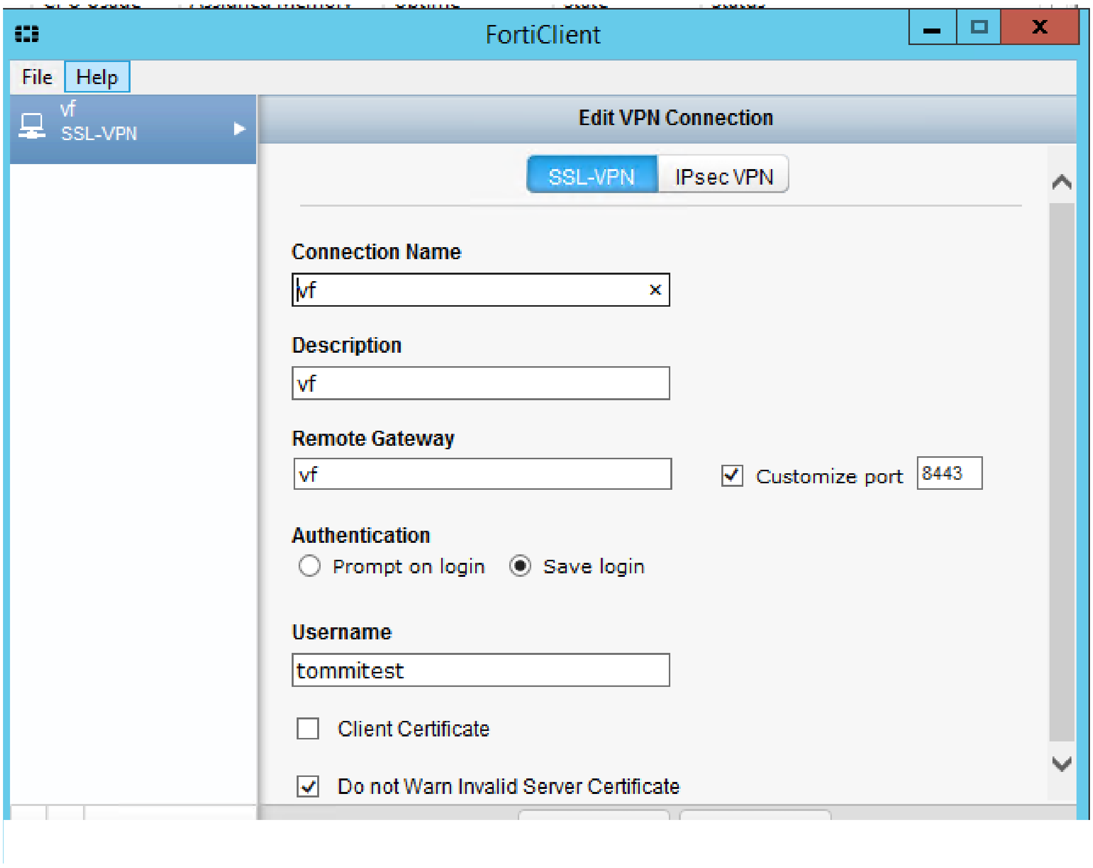
Table 15 IPSec VPN throughput results

IPSec encryption/authentication	Throughput (Mb/s)
AES256/SHA256/DH 14	**318**
AES128/SHA256/DH 5	**443**
3DES/SHA1/DH 2	**121**

The results show that throughput is best with AES 128 encryption (443 Mb/s), somewhat less with AES 256 encryption (218 Mb/s) and significantly less with 3 DES encryption (121 Mb/s).

6.6 Test 4: SSL VPN throughput

SSL VPN testing was performed so that (physical) server 2 was acting as the SSL VPN client and the iPerf client, virtual firewall was acting as the SSL VPN server and virtual machine VM 1 was acting as an iPerf server. SSL VPN client “FortiClient 5.4.1.0840” was first installed on the Server 2. Screenshot of the FortiClient software is seen in Figure 38.



Connection Name
vf

Description
vf

Remote Gateway
vf

☑ Customize port 8443

Authentication
○ Prompt on login ● Save login

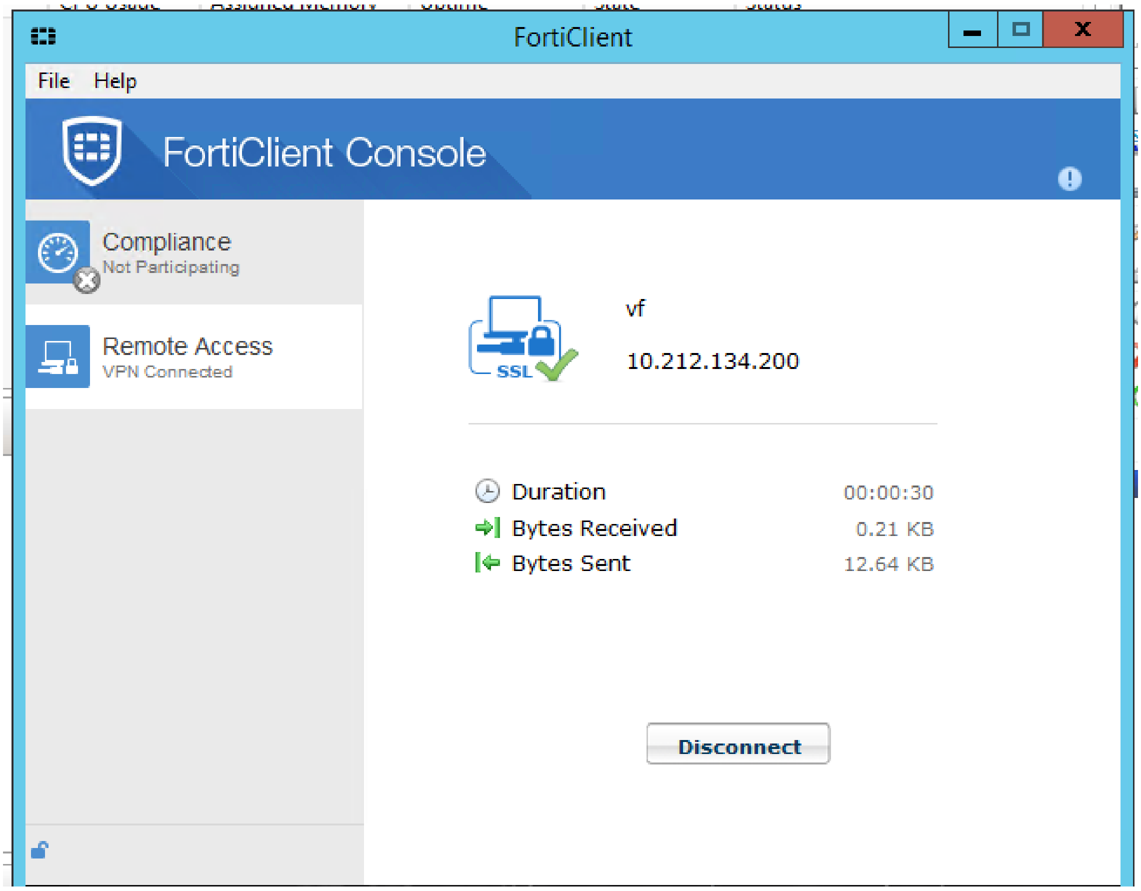
Username
tommitest

□ Client Certificate

☑ Do not Warn Invalid Server Certificate

Figure 38 Screenshot of SSL VPN client software

After SSL VPN client was installed, the connection to virtual firewall was started as shown in Figure 39:



vf

10.212.134.200

Duration 00:00:30

Bytes Received 0.21 KB

Bytes Sent 12.64 KB

Figure 39 Active FortiClient SSL VPN connection to virtual firewall

After the connection was established, the throughput between SSL VPN client and VM 1 was measured by running the iPerf3 test. Figure 40 portrays the SSL VPN traffic flow from Server 2 to virtual firewall to VM 1.

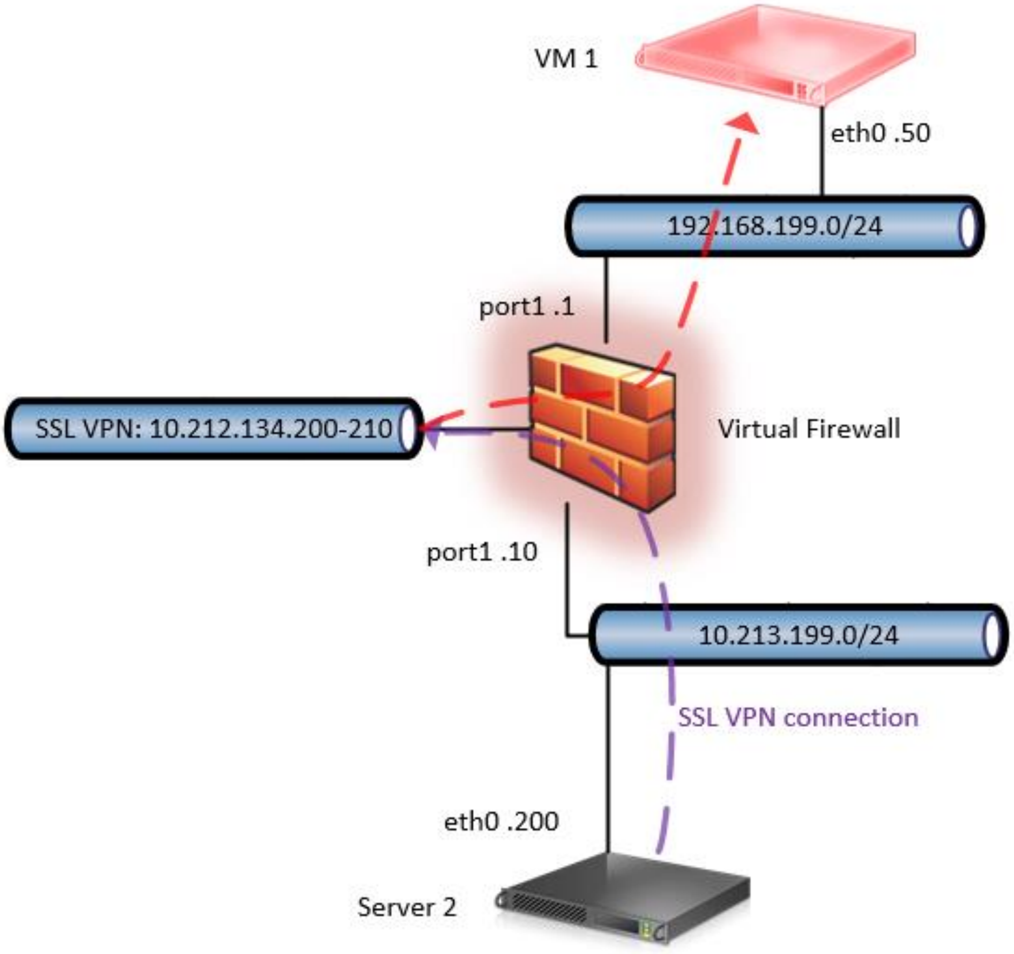


Figure 40 SSL VPN test traffic flow

In this test, SSL VPN connection between Server 2 and virtual firewall was established and Server 2 received virtual IP from range 10.212.134.200-210. Then, iPerf3 test was performed between Server 2 and VM1 (192.168.199.50).

Results of the SSL VPN throughput test are visible in Table 14.

Table 16 SSL VPN throughput

SSL VPN	Throughput (Mb/s)
	109

As seen from the result, the throughput of 109 Mb/s via SSL VPN connection is almost ten times less than theoretical throughput of 1 Gb/s.

7 Measurement Analysis

In the first part of the testing, the throughput between two VMs was measured. Then the virtual firewall was introduced between the VMs and throughput testing was again performed with three policy set sizes (1, 100 and 1000 policies) and with both NAT enabled and disabled. The results recorded in Figure 41 show that NAT or policy set size appears to have no impact on the firewall throughput. There is only a two-percentage drop ((1-(1890/1930 Mb/s))*100 = 2 %) in throughput when virtual firewall is between the VMs.

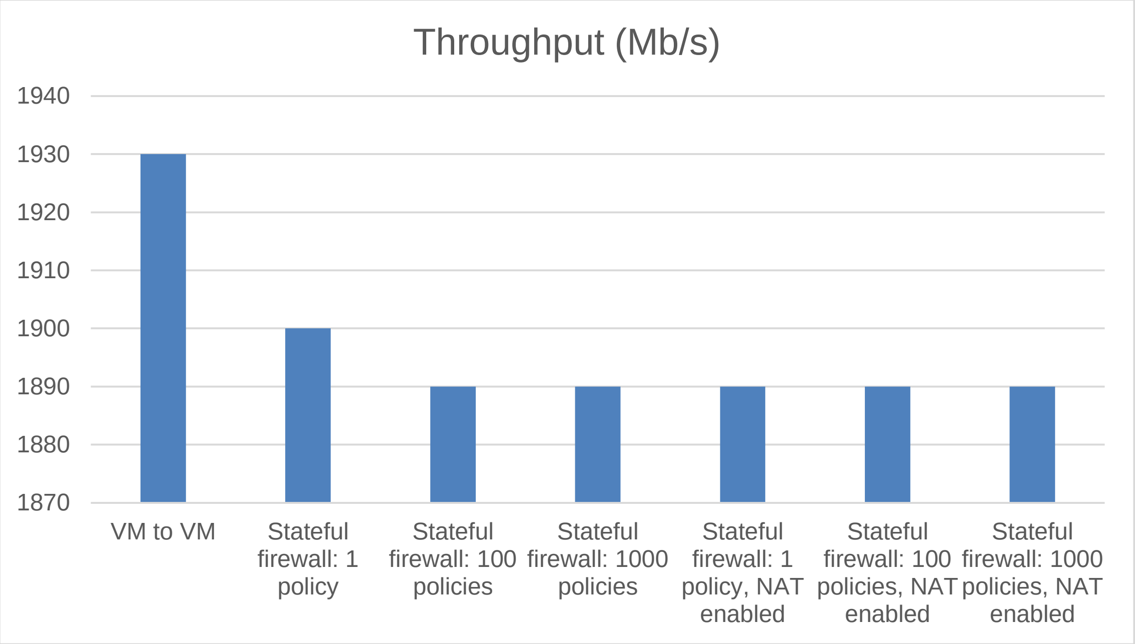


Figure 41 VM to VM throughput

In the next testing phase the throughput of was tested when the virtual firewall was running UTM security services. The test was run with FTP protocol and with only a single simultaneous download at a time. Each download was performed ten times and the average transfer speed was calculated. The results in Figure 42 show that there is a performance drop of approximately six percentage ((1-(585/621 Mb/s))*100 = 6 %) when UTM is enabled.

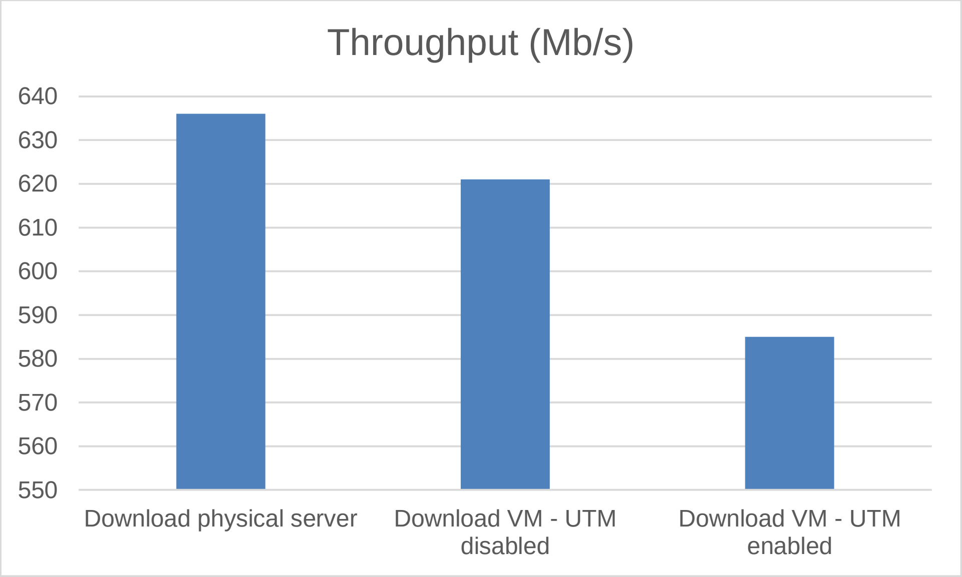


Figure 42 Throughput with and without UTM security services

In the third testing phase a second virtual firewall was installed on the hypervisor and a VPN tunnel was created between firewalls. VM2 was reconfigured so that it was “behind” the second VF (see Figure 37). The VPN tunnel between VFs was configured with three separate encryption settings, with AES 256, AES 128 and 3DES. Then, iPerf3 testing was performed between client (VM1) and server (VM2) with each tunnel configuration and results were recorded in Figure 43.

The results show that AES 256 throughput is approximately 30 % less than AES128 throughput ((1 - (318 / 443 Mb/s))*100 = 28 %). Albeit being simpler algorithm than AES, the 3DES throughput is significantly lower compared to AES 128 or AES 256 (121 vs. 318 or 443 Mb/s) throughput).

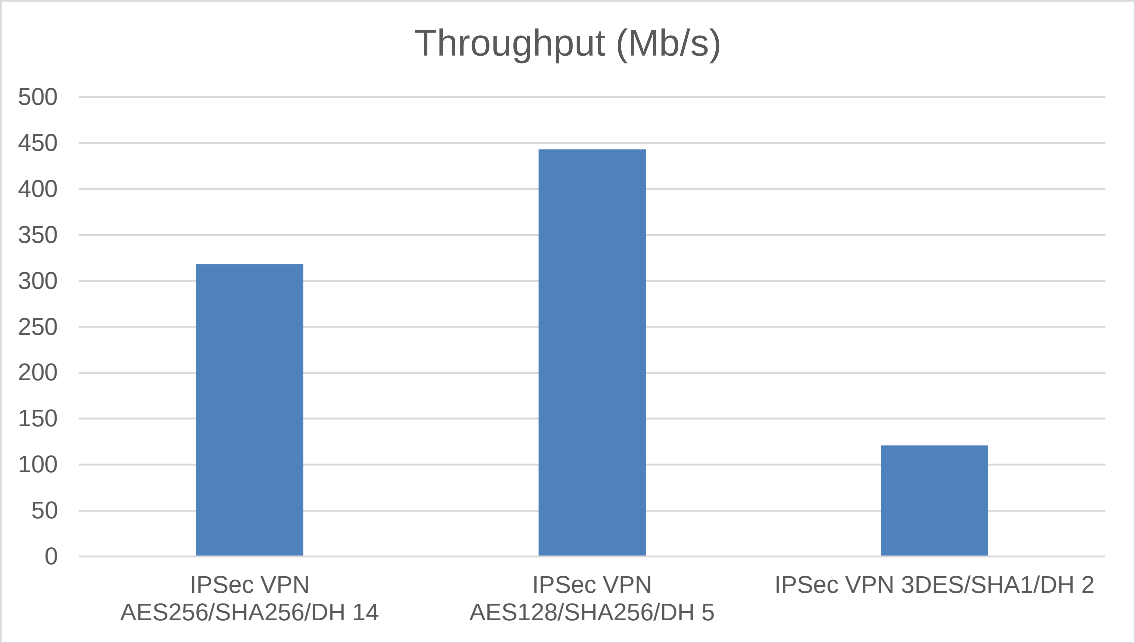


Figure 43 IPSec VPN throughput

In the fourth testing phase SSL VPN performance was tested. On virtual firewall the SSL VPN feature was configured and SSL VPN plugin was installed on a physical server Then, SSL VPN connection was started and iPerf testing was performed (see Figures 38, 39 and 40). SSL VPN throughput is on the same level as 3DES IPSec VPN tunnel throughput (109 vs 121 Mb/s). The throughput is nearly ten times smaller than the baseline as seen in Figure 44 (109 Mb/s vs. 949 Mb/s).

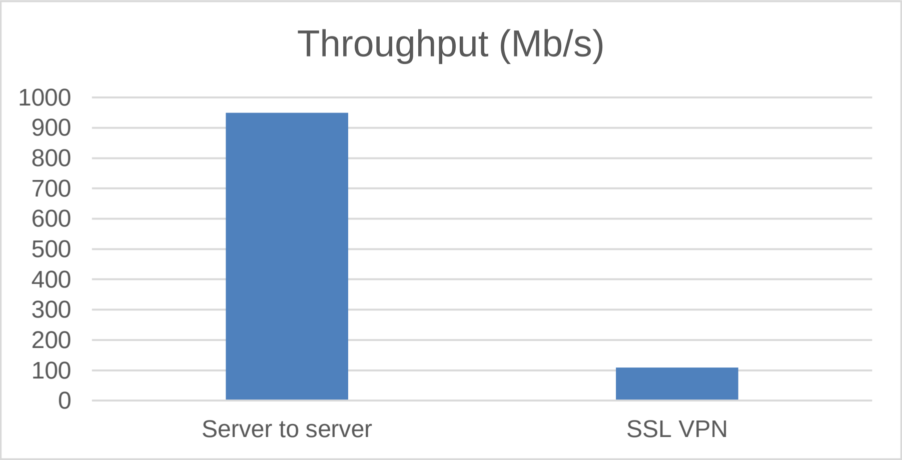


Figure 44 SSL VPN throughput

The achieved results demonstrate that firewall throughput diminishes when complexity of used firewall features increases. This is expected and shows used testing methodology is valid.

8 Discussion and Conclusions

The use of virtualization technologies is rapidly becoming more popular in organizations and enterprises. In many organizations over 75 % of servers are already virtualized (Gartner, Inc., 2016). In addition, it can be said that network virtualization techniques (such as VLANs and VRFs) are used in nearly all service provider and data center networks. The improvement and popularity of virtualization technologies has enabled cloud computing.

A large part of organizations have already deployed private clouds and there is a growing need to create hybrid cloud deployments. In hybrid clouds, private and public cloud deployments are connected, effectively connecting the organization’s existing private network directly to a public cloud network infrastructure. Virtual and cloud networking environments need to be protected with NGFWs like traditional networks. In many cases the use of physical NGFWs for this task becomes increasingly complex and expensive. For this reason, there is an emerging need and demand for virtual firewalls both in private and public cloud deployments. The objective of this thesis was to study virtual firewall technology in the perspective of a network service provider. The main focus of the study was on evaluating the maturity of virtual firewall technology in a private cloud. This evaluation was largely done by performing network throughput benchmarking in various network configurations. The throughput testing scenarios covered four virtual firewall functions: stateful firewall, UTM services, IPSec VPN and SSL VPN. The results of these tests is discussed next.

Firstly, the stateful firewall throughput was measured. The results indicate that directing VM traffic through a virtual firewall has only a small impact on network throughput. The firewall throughput in each test of this test phase was close to 1.9 Gb/s. NAT or the number of firewall policies up to 1000 policies had almost no effect on the performance. However, the firewall GUI became almost unusable when 100 or more policies were configured on the firewall. In CLI, the policy configuration was not affected by the number of policies. Therefore, the issue is not critical but rather annoying. The GUI issue needs to be investigated more closely with the vendor. In this test, the test traffic was artificially generated with iPerf. Test could be developed further by generating traffic with a protocol mix that emulates real life situations. This could be done for example by commercial product IxChariot from Ixia Plc.

Secondly, basic UTM throughput of the firewall was tested. Security functions of the virtual firewall were not the focus of this study, but this testing was done as security is an inherent part of firewall functionality. The testing was performed by simply enabling default scan settings in and initiating FTP downloads from the Internet. The results in this test phase show there was only six percentage drop in network throughput when UTM options were enabled. This result was somewhat surprising, given that in a similar network firewall test performed by an independent testing Miercom, all tested firewalls experienced a significant performance drop when UTM options were enabled (Miercom, Inc., 2015, p. 9). However, it should be noted that throughput was tested only with an FTP application and only with one simultaneous download. Therefore, this result is only an indication of the performance and as such does not necessarily relate to real world usage scenarios. In the future, it would be interesting to study the UTM performance more closely with real world traffic. This could be done as part of a proof of concept (PoC) deployment for an interested customer.

Thirdly, IPSec VPN throughput of the firewall was evaluated. IPSec VPN tunneling is a popular method of securely connecting trusted networks together over untrusted ones (i.e. the Internet). VPN tunnel testing was performed by configuring two virtual firewalls and creating a tunnel between them. Then, network throughput was measured by running iPerf traffic via both firewalls and the tunnel. Results show that IPSec VPN throughput is comparable to many lower end Fortigate models when AES 256 or AES 128 encryption is used (see Appendix 2). Interestingly, it was found that 3DES encrypted IPSec VPN performance was significantly lower. After research it was found that 3DES is deemed to be a legacy, not so secure encryption method (Cisco, Inc., 2012). Therefore, it is recommended to avoid using 3DES encryption in virtual firewall VPN tunnel deployments and use AES encryption instead.

Fourthly, SSL VPN throughput of the virtual firewall was tested. SSL VPN is a technology that provides secured remote access to network resources. The traffic from remote user is tunneled over untrusted network (Internet) and tunnel is secured with SSL or TLS encryption. In the test, an SSL VPN connection was created between a client machine and virtual firewall. Then, iPerf test was performed via the SSL VPN tunnel. As a result in this test, the average throughput recorded was 109 Mb/s. Again, this result is comparable to many low-end Fortigate models (see Appendix 2). Both IPSec and SSL VPN tests could be further enhanced by evaluating the performance when several tunnels are simultaneously active, and when there is more load on the firewall.

In conclusion, the results show that virtual firewall throughput performance is close to same vendors' (Fortinet) low-end network firewalls. In addition, it was found that virtual firewall deployment and configuration was practical and straightforward. No major issues were encountered in any part of the deployment. It can be said that virtual firewall network performance is feasible and the technology is ready for production use. The developed test method can be replicated in other deployment scenarios, e.g. when measuring performance of virtual firewalls deployed in Microsoft Azure or in Amazon Web Services public clouds.

For further research, it would be interesting to measure the throughput with more resources allocated on the firewall. Now the testing was performed with only one resource configuration. In this study, only network throughput measurement was included in the scope. The performance evaluation would have been more comprehensive if other benchmarks were measured as well. Also, it would be interesting to compare virtual against a physical firewall appliance. Now the virtual firewall performance was only compared to the theoretical performance values reported by the vendor.

References

Amit, N. et al., 2015. *Virtual CPU validation.* New York, NY, USA, ACM.

Apple Inc., 2016. *iOS Developer Library.* [Online] Available at: https://developer.apple.com/library/ios/documentation/IDEs/Conceptual/iOS_Simulator_Guide/Introduction/Introduction.html
[Accessed 13 June 2016]

Bhartiya, S., 2016. *The Best Linux Distros of 2016.* [Online] Available at: https://www.linux.com/NEWS/BEST-LINUX-DISTROS-2016
[Accessed 4 August 2016]

Boley, L., 2014. *Emulation and Virtualization: What's the difference?.* [Online] Available at: https://powermore.dell.com/technology/emulation-virtualization-whats-difference/
[Accessed 13 June 2016]

Bradner, S., Dubray, K., McQuaid, J. & Morton, A., 2012. *Applicability Statement for RFC 2544: Use on Production Networks Considered Harmful,* s.l.: Internet Engineering Task Force (IETF)

Bradner, S. & McQuaid, J., 1999. *Benchmarking Methodology for Network Interconnect Devices,* s.l.: Network Working Group (IETF)

Brocade Inc., 2005. *Multi-VRF overview.* [Online] Available at: http://www.brocade.com/content/html/en/configuration-guide/nos-701-l3guide/GUID-DCD4B821-E1CE-47E1-A2E3-CADA63792CF7.html
[Accessed 26 July 2016]

Brodkin, J., 2009. *With long history of virtualization behind it, IBM looks to the future.* [Online]
Available at: http://www.networkworld.com/article/2254433/virtualization/with-long-history-of-virtualization-behind-it--ibm-looks-to-the-future.html
[Accessed 2 March 2016]

Cisco Systems, Inc., 2015. *Inter-Switch Link and IEEE 802.1Q Frame Format - Cisco.* [Online]
Available at: http://www.cisco.com/c/en/us/support/docs/lan-switching/8021q/17056-741-4.html#topic2
[Accessed 25 July 2016]

Cisco Systems, Inc., 2015. *Network Virtualization--Path Isolation Design Guide - Cisco.* [Online]
Available at: http://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Network_Virtualization/PathIsol.html#wp80043
[Accessed 26 July 2016]

Cisco Systems, Inc, 2013. *Catalyst 4500 Series Switch Cisco IOS Software Configuration Guide, 12.2(25)EW - Understanding and Configuring VLANs.* [Online]
Available at: http://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst4500/12-2/25ew/configuration/guide/conf/vlans.html
[Accessed 25 July 2016]

Cisco Systems, Inc, 2014. *Cisco Active Network Abstraction Reference Guide, 3.7 - Virtual Routing and Forwarding.* [Online]
Available at: http://www.cisco.com/c/en/us/td/docs/net_mgmt/active_network_abstraction/3-7/reference/guide/ANARefGuide37/vrf.html
[Accessed 26 July 2016]

Cisco Systems, Inc, 2015. *Borderless Campus Network Virtualization—Path Isolation Design Fundamentals - Cisco.* [Online]
Available at: http://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Campus/Network_Virtualization/sccsolover.html#wp416616
[Accessed 26 July 2016]

Dugan, J. et al., 2016. *iPerf - The network bandwidth measurement tool.* [Online]
Available at: https://iperf.fr
[Accessed 19 August 2016]
EXFO Electro-Optical Engineering Inc., 2008. *RFC 2544: HOW IT HELPS QUALIFY A CARRIER ETHERNET NETWORK,* Quebec: EXFO Inc..

Fei Guo, VMWare, Inc, 2011. *Understanding Memory Resource Management in VMware vSphere 5.0.* [Online] Available at: https://www.vmware.com/files/pdf/mem_mgmt_perf_vsphere5.pdf
[Accessed 18 May 2016]

Firewall.cx, 2013. *Firewall.cx.* [Online] Available at: http://www.firewall.cx/networking-topics/general-networking/970-network-performance-testing.html
[Accessed 15 November 2016]

Fortinet Inc., 2015. *Fortigate Virtual Appliances.* [Online] Available at: https://www.fortinet.com/content/dam/fortinet/assets/data-sheets/Forti-Gate_VM.pdf
[Accessed 6 10 2016]

Fortinet, Inc., 2015. *FortiGate VM Installation Guide - Fortinet Document Library.* [Online] Available at: http://docs.fortinet.com/uploaded/files/2324/fortigate-vm-install-52.pdf
[Accessed 4 August 2016]

Gartner Inc., 2016. *Magic Quadrant for Cloud Infrastructure as a Service, Worldwide.* [Online]
Available at: https://www.gartner.com/doc/reprints?id=1-2G2O5FC&ct=150519
[Accessed 1 11 2016]

Gartner, Inc., 2016. *Gartner Says Worldwide Server Virtualization Market Is Reaching Its Peak.* [Online]
Available at: http://www.gartner.com/newsroom/id/3315817
[Accessed 10 November 2016]

Hickman, B., Newman, D., Tadjudin, S. & Martin, T., 2003. *Request for Comments: 3511.* [Online]
Available at: https://www.ietf.org/rfc/rfc3511.txt
[Accessed 2 August 2016]

Hossain, M., 2014. *Trends in Data Center Security: Part 1 – Traffic Trends.* [Online] Available at: http://blogs.cisco.com/security/trends-in-data-center-security-part-1-traffic-

trends
[Accessed 10 November 2016]

IEEE Standards Association, 2014. *Bridges and Bridged Networks.* New York: The Institute of Electrical and Electronics Engineers, Inc..

ITU-T, 2016. *Ethernet service activation test methodology,* Geneva, Switzerland: ITU-T.
Jithin, R. & Chandran, P., 2014. Virtual Machine Isolation. In: *Recent Trends in Computer Networks and Distributed Systems Security.* Kerala, India: Springer Berlin Heidelberg, pp. 91-102

Josyula, V., Orr, M. & Page, G., 2011. *Cloud Computing: Automating the Virtualized Data Center.* s.l.:Cisco Press

Liebowitz, M., Kusek, C. & Spies, R., 2014. *VMware vSphere Performance.* s.l.:Sybex.

Mell, P. & Grance, T., 2011. *The NIST Definition of Cloud Computing,* Gaithersburg, MD: National Institute of Standards and Technology: U.S. Department of Commerce.

Omnitron Systems Technology, Inc, 2016. *Service Activation Testing.* [Online] Available at: http://www.omnitron-systems.com/products/service-activation-testing.php
[Accessed 31 July 2016]

Oracle, 2016. *VirtualBox manual.* [Online] Available at: https://www.virtualbox.org/manual/ch01.html#hostossupport
[Accessed 10 June 2016]

Popek, G. J. & Goldberg, R. P., 1974. Formal requirements for virtualizable third generation architectures. *Communications of the ACM,* 1974(7), pp. 412-421.

Portnoy, M., 2012. *Essentials: Virtualization Essentials.* s.l.:Sybex.

Santana, G. A. A., 2014. *Data Center Virtualization Fundamentals.* Indianapolis, IN 26240 USA: Cisco Press.

Srinivasan, S., 2014. *Cloud Computing Basics.* Houston, Texas: Springer New York.

Stuart Whitehead, Anritsu Corporation, 2011. *ITU Y.1564 Ethernet Testing,* Japan: s.n.

VMWare Inc., 2016. *VMware vSphere 4 - ESX and vCenter Server.* [Online] Available at: https://pubs.vmware.com/vsphere-4-esx-vcenter/index.jsp?topic=/com.vmware.vsphere.server_configclassic.doc_40/esx_server_config/security_for_esx_systems/c_security_and_virtual_machines.html
[Accessed 12 June 2016]

VMWare, Inc, 2007. *Understanding Full Virtualization, Paravirtualization and Hardware Assist.* [Online]
Available at: http://www.vmware.com/files/pdf/VMware_paravirtualization.pdf
[Accessed 2 March 2016]

Xen.org, 2015. *Paravirtualization (PV) - Xen.* [Online]
Available at: http://wiki.xen.org/wiki/Paravirtualization_%28PV%29
[Accessed 8 June 2016]

Xing, C., Zhang, G. & Chen, M., 2007. *Research on Universal Network Performance Testing Model.* s.l., IEEE.

Appendix 1: UTM Options configuration

Fortigate VM64-HV, software version 5.4.1

```
VF (root) # show antivirus profile
config antivirus profile
    edit "default"
        set comment "Scan files and block viruses."
        set inspection-mode proxy
        config http
            set options scan
        end
        config ftp
            set options scan
        end
        config imap
            set options scan
            set executables virus
        end
        config pop3
            set options scan
            set executables virus
        end
        config smtp
            set options scan
            set executables virus
        end
    next
end
```

```
VF (root) # show webfilter profile default
config webfilter profile
    edit "default"
        set comment "Default web filtering."
        config ftgd-wf
```

```
unset options
config filters
    edit 1
        set category 2
        set action warning
    next
    edit 2
        set category 7
        set action warning
    next
    edit 3
        set category 8
        set action warning
    next
    edit 4
        set category 9
        set action warning
    next
    edit 5
        set category 11
        set action warning
    next
    edit 6
        set category 12
        set action warning
    next
    edit 7
        set category 13
        set action warning
    next
    edit 8
        set category 14
        set action warning
    next
    edit 9
        set category 15
        set action warning
    next
    edit 10
        set category 16
        set action warning
    next
    edit 11
        set action warning
    next
    edit 12
        set category 57
        set action warning
    next
    edit 13
        set category 63
        set action warning
    next
    edit 14
        set category 64
        set action warning
    next
    edit 15
        set category 65
        set action warning
    next
    edit 16
        set category 66
        set action warning
    next
    edit 17
        set category 67
        set action warning
    next
    edit 18
```

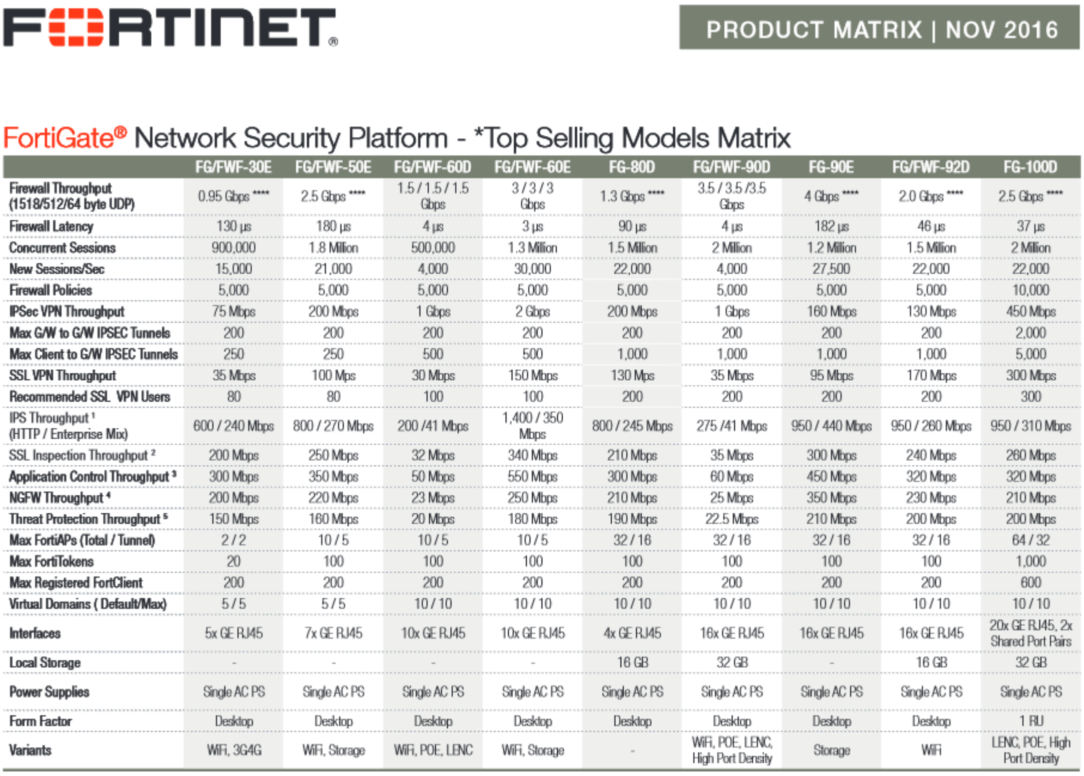
```
                    set category 26
                    set action block
                next
                edit 19
                    set category 61
                    set action block
                next
                edit 20
                    set category 86
                    set action block
                next
                edit 21
                    set category 88
                    set action block
                next
            end
        end
    next
end


VF (root) #  show application list default
config application list
    edit "default"
        set comment "Monitor all applications."
        config entries
            edit 1
                set action pass
            next
        end
    next
end


VF (root) # show ips sensor default
```

```
config ips sensor
    edit "default"
        set comment "Prevent critical attacks."
        config entries
            edit 1
                set severity medium high critical
            next
        end
    next
end
```

Appendix 2: Fortigate Network Firewall Performance Values



PRODUCT MATRIX | NOV 2016

FortiGate® Network Security Platform - *Top Selling Models Matrix

	FG/FWF-30E	FG/FWF-50E	FG/FWF-60D	FG/FWF-60E	FG-80D	FG/FWF-90D	FG-90E	FG/FWF-92D	FG-100D
Firewall Throughput (1518/512/64 byte UDP)	0.95 Gbps ****	2.5 Gbps ****	1.5 / 1.5 / 1.5 Gbps	3 / 3 / 3 Gbps	1.3 Gbps ****	3.5 / 3.5 /3.5 Gbps	4 Gbps ****	2.0 Gbps ****	2.5 Gbps ****
Firewall Latency	130 µs	180 µs	4 µs	3 µs	90 µs	4 µs	182 µs	46 µs	37 µs
Concurrent Sessions	900,000	1.8 Million	500,000	1.3 Million	1.5 Million	2 Million	1.2 Million	1.5 Million	2 Million
New Sessions/Sec	15,000	21,000	4,000	30,000	22,000	4,000	27,500	22,000	22,000
Firewall Policies	5,000	5,000	5,000	5,000	5,000	5,000	5,000	5,000	10,000
IPSec VPN Throughput	75 Mbps	200 Mbps	1 Gbps	2 Gbps	200 Mbps	1 Gbps	160 Mbps	130 Mbps	450 Mbps
Max G/W to G/W IPSEC Tunnels	200	200	200	200	200	200	200	200	2,000
Max Client to G/W IPSEC Tunnels	250	250	500	500	1,000	1,000	1,000	1,000	5,000
SSL VPN Throughput	35 Mbps	100 Mps	30 Mbps	150 Mbps	130 Mps	35 Mbps	95 Mbps	170 Mbps	300 Mbps
Recommended SSL VPN Users	80	80	100	100	200	200	200	200	300
IPS Throughput [1] (HTTP / Enterprise Mix)	600 / 240 Mbps	800 / 270 Mbps	200 /41 Mbps	1,400 / 350 Mbps	800 / 245 Mbps	275 /41 Mbps	950 / 440 Mbps	950 / 260 Mbps	950 / 310 Mbps
SSL Inspection Throughput [2]	200 Mbps	250 Mbps	32 Mbps	340 Mbps	210 Mbps	35 Mbps	300 Mbps	240 Mbps	260 Mbps
Application Control Throughput [3]	300 Mbps	350 Mbps	50 Mbps	550 Mbps	300 Mbps	60 Mbps	450 Mbps	320 Mbps	320 Mbps
NGFW Throughput [4]	200 Mbps	220 Mbps	23 Mbps	250 Mbps	210 Mbps	25 Mbps	350 Mbps	230 Mbps	210 Mbps
Threat Protection Throughput [5]	150 Mbps	160 Mbps	20 Mbps	180 Mbps	190 Mbps	22.5 Mbps	210 Mbps	200 Mbps	200 Mbps
Max FortiAPs (Total / Tunnel)	2 / 2	10 / 5	10 / 5	10 / 5	32 / 16	32 / 16	32 / 16	32 / 16	64 / 32
Max FortiTokens	20	100	100	100	100	100	100	100	1,000
Max Registered FortClient	200	200	200	200	200	200	200	200	600
Virtual Domains (Default/Max)	5 / 5	5 / 5	10 / 10	10 / 10	10 / 10	10 / 10	10 / 10	10 / 10	10 / 10
Interfaces	5x GE RJ45	7x GE RJ45	10x GE RJ45	10x GE RJ45	4x GE RJ45	16x GE RJ45	16x GE RJ45	16x GE RJ45	20x GE RJ45, 2x Shared Port Pairs
Local Storage	-	-	-	-	16 GB	32 GB	-	16 GB	32 GB
Power Supplies	Single AC PS	Single AC PS	Single AC PS	Single AC PS	Single AC PS	Single AC PS	Single AC PS	Single AC PS	Single AC PS
Form Factor	Desktop	Desktop	Desktop	Desktop	Desktop	Desktop	Desktop	Desktop	1 RU
Variants	WiFi, 3G4G	WiFi, Storage	WiFi, POE, LENC	WiFi, Storage	-	WiFi, POE, LENC, High Port Density	Storage	WiFi	LENC, POE, High Port Density