



REPÚBLICA DEL ECUADOR

**UNIVERSIDAD TECNOLÓGICA
EMPRESARIAL DE GUAYAQUIL**

**TRABAJO DE TITULACIÓN
PARA LA OBTENCIÓN DEL TÍTULO DE:**

**INGENIERO EN GESTIÓN DE TELECOMUNICACIONES
MENCIÓN REDES DE ACCESO Y TELEFONÍA**

TEMA:

**IMPLEMENTACIÓN DE SERVIDORES VIRTUALIZADOS BASADOS EN REDES DEFINIDAS
POR SOFTWARE APLICADOS EN LA EMPRESA MAGOZOLUTIONS**

FRANK JONNATHAN SUÁREZ MAYORGA

2018

GUAYAQUIL – ECUADOR



REPÚBLICA DEL ECUADOR

**UNIVERSIDAD TECNOLÓGICA
EMPRESARIAL DE GUAYAQUIL**

**TRABAJO DE TITULACIÓN
PARA LA OBTENCIÓN DEL TÍTULO DE:**

**INGENIERO EN GESTIÓN DE TELECOMUNICACIONES
MENCIÓN REDES DE ACCESO Y TELEFONÍA**

TEMA:

**IMPLEMENTACIÓN DE SERVIDORES VIRTUALIZADOS BASADOS EN REDES DEFINIDAS
POR SOFTWARE APLICADOS EN LA EMPRESA MAGOZOLUTIONS**

FRANK JONNATHAN SUÁREZ MAYORGA

2018

GUAYAQUIL – ECUADOR

AGRADECIMIENTO

Mi más profundo agradecimiento a Dios por haberme permitido alcanzar esta meta, también quiero agradecer a todas las personas que hicieron posible la realización de este trabajo, a mis padres por sus consejos y darme su apoyo desde el principio de mi vida, a mi esposa por soportarme y comprenderme, sin ella esta meta hubiera tardado más tiempo, gracias por tu apoyo Teresa, también a todos mis profesores, que aportaron a mi desarrollo profesional, a todos muchas gracias.

DEDICATORIA

Quiero dedicar este trabajo a Franklin Suárez y Juanita Mayorga, mis amados padres, a Teresa Lucio mi esposa, y a Rafaela, Camila y Alejandra, mis hijas.

La responsabilidad de este trabajo de investigación, con sus resultados y conclusiones, pertenece exclusivamente al autor.

.....

Frank Jonnathan Suárez Mayorga

IMPLEMENTACION DE SERVIDORES VIRTUALIZADOS BASADOS EN REDES DEFINIDAS POR SOFTWARE APLICADOS EN LA EMPRESA MAGOZOLUTIONS

Frank Jonnathan Suarez Mayorga

jonnathan.suarez@gmail.com

1.- RESUMEN

Con el aumento de tráfico de red en los centros de datos, los administradores de red buscan mecanismos para mejorar los servicios, tales como alojamiento web, correo electrónico, bases de datos entre otros, asimismo optimizar el uso de recursos de hardware, logrando la consolidación de servidores mediante el uso de tecnologías de virtualización, mejorando su administración, que en conjunto, con el uso de redes definidas por software suponen una solución a una serie de problema actuales, entre estos la gestión de equipos, tiempos de aprovisionamiento y costos operativos. Es por eso que en este articulo académico se analizó los beneficios del uso de redes definidas por software aplicadas en servidores virtualizados, se aplicó la lógica del controlador a una red SDN para obtener un balanceo de carga entre servidores. Se hizo uso de un simulador de redes *Mininet*, el cual permitió crear la topología de red compatible con SDN, con el uso de *Apache Traffic Server* se simuló un cacheo de servidores web corriendo bajo *Ubuntu server*, haciendo uso de *OpenDayLight* se obtuvo el controlador que comunicará toda la topología, todo esto bajo un ambiente de virtualización en *Xenserver*, donde se comprobó la viabilidad de la implementación.

Palabras claves: Virtualización, Redes definidas por software, SDN, Mininet, OpenDayLight, Openflow.

2.- INTRODUCCIÓN

En la actualidad es común el uso de un servidor físico por cada servicio que se use en las actividades de una empresa, ocasionando subutilización de recursos y el aumento de espacio necesario y costes de mantenimiento de los centros de datos.

¿Cómo ayudarían a la gestión de centros de datos el uso de redes definidas por software aplicados en servidores virtualizados?

Se analizó la implementación de servidores virtualizados donde se optimizó recursos y se obtuvo tolerancia a fallos al poder migrarlos dentro de un pool de servidores físicos, además con el uso de redes definidas por software se alcanzó mejor dirección, una gestión unificada y estricto registro de red, mediante la creación de reglas de control de tráfico para un balanceo de carga.

3.- MARCO TEÓRICO

3.1.- Virtualización

“Las tecnologías de virtualización pueden ser descritas como un procedimiento lógico que es segmentado, funcionando de manera real dentro de un sistema de cómputo. En las plataformas virtuales creadas el usuario puede observarlas, pero se ejecutan como si fueran computadores físicos” Lugo (2014). “Esta virtualización es lograda instalando software junto con los servicios para gestionar el almacenamiento compartido, los ciclos de computación, además de las aplicaciones”. Casierra, Quiñonez, Herrera y Egas (2018). Es decir, la virtualización de red trata a todos los servidores y servicios en la red como un único grupo de recursos al que pueden acceder sin sopesar sus componentes de hardware. Con esto la virtualización nos permitiría la administración eficiente los recursos, tanto físicos y lógicos, consolidación de equipos, ahorro en costos en el manejo de una infraestructura, mejorando adicionalmente la gestión y seguridad de los ambientes virtualizados, además consideramos varios tipos de virtualización: de redes, de almacenamiento, de estaciones de trabajo.

3.1.1.- Funcionamiento de una máquina virtual

El software genera una capa virtual entre el hardware del equipo anfitrión y el virtualizado, consiguiendo de esta forma emular los componentes de este, como procesadores, memoria RAM, discos duros, interfaces de red (Call, 2015). Al virtualizar un sistema operativo, se puede asignar las características de la máquina virtual y sus respectivas configuraciones. La virtualización completa está diseñada para proporcionar abstracción total del sistema físico fundamental y crea un sistema virtual completo en

que se puede ejecutar el sistema operativo huésped (*guest*), además no se requiere ninguna modificación en el SO o aplicación huésped.

El sistema operativo o aplicación cliente no distingue el entorno virtualizado, de modo que se puede ejecutar en la máquina virtual (VM) como si ella fuera un sistema físico. (Segovia, Núñez, Velastegui, Navas, Vásquez, 2018). Esto nos da la posibilidad de minimizar los tiempos de interrupción durante una eventualidad, ya sea programada como por ejemplo un mantenimiento, o fortuito como en el caso de daño en el hardware, donde se puede realizar la migración de la máquina virtual de manera paralela a la principal minimizando el impacto en la empresa y garantizar la operatividad de los sistemas.

En otros casos la virtualización de escritorios brinda una infraestructura, plataforma y software como servicio, ofreciendo ventajas como control de accesos a usuarios, seguridad de la información, gestión unificada, reducción de costos, disponibilidad para la administración en todo momento.

“La parte encargada de realizar la virtualización en el host se denomina hipervisor” (Gallego, 2014). Hay dos tipos:

- Nativo: el hipervisor es el host y se ejecuta directamente sobre el hardware del equipo.
- Alojado: el hipervisor se ejecuta como una aplicación el sistema operativo del host, es decir toda interacción de los recursos del equipo se realiza a través del sistema operativo.

3.1.2.- Ventajas de la Virtualización

La consolidación mediante la virtualización de servidores genera una reducción en el costo de energía eléctrica, uso de espacio físico y tiempo de implementación de nuevos servicios o en la aplicación de políticas tales como control de acceso o actualizaciones en los proyectos en producción, además de permitir la centralización de los registros de actividades en la red. La virtualización de la infraestructura de redes ya sea conmutadores, enlaces o enrutadores mejora la calidad de servicio en las aplicaciones en la nube, mitiga los riesgos en la seguridad física y de lógica, facilita la migración de las aplicaciones existentes en los clientes, incrementa el soporte en la red, entre otros.

3.2.- Redes Definidas por Software

El fundamento de la arquitectura SDN *-Software Defined Networking-* o Redes definidas por software según la ONF, “Es una arquitectura de red dinámica, gestionable, adaptable, de costo eficiente. Lo cual la hace ideal para las altas demandas de ancho de banda y la naturaleza dinámica de las aplicaciones actuales”. (Dueñas, Marín, Cruz, 2017). “SDN permite que el control de la red sea programable logrando que las aplicaciones y servicios de red se separen de la infraestructura” (*Open Networking Foundation*, 2018). En contraste con las redes tradicionales, que por naturaleza son estáticas lo cual es el principal obstáculo para los propósitos de las actuales implementaciones que deben crecer para interconectar cientos de nuevos dispositivos a diario, tales como teléfonos y televisores inteligentes, tabletas, computadores portátiles

y de escritorio, dispositivos de vigilancia o sistemas inteligentes para el hogar, entre otros, los cuales tienen diferentes necesidades de rendimiento y de servicio. Donde, por ejemplo, para realizar cambios de enrutamiento o implementar un nuevo servicio, implicaría programar rutinas para intervenir en cada uno de los dispositivos y terminales que intervienen en el servicio. Con el uso de SDN las redes pueden evolucionar para responder rápidamente a los cambios de necesidades de la empresa, o de los usuarios, gestionando recursos en tiempo real, como ancho de banda, calidad de servicio, ingeniería de tráfico, rutas de protección, etc.

3.2.1.- Funcionamiento de las redes definidas por software

“En una red virtualizada toda la funcionalidad reside en los programas que ejecutan las diferentes funciones y no en el hardware en sí. En este tipo de redes el plano de control está centralizado por un software que controla todo” (García, Rodríguez, Anias, Camilo, 2014). De esta forma un mismo dispositivo se pueda utilizar para distintos propósitos en función de la aplicación que tenga instalada, de modo que es posible agregar varias funcionalidades de red en una o múltiples máquinas virtuales y decidir en cada momento en qué servidor va a ejecutarse. Usando SDN se pueden administrar y gestionar las distintas interconexiones de la red desde un punto centralizado. Se separa el plano de control del plano de datos, y formadas por un controlador SDN, una *Application Programming Interface* (API por sus siglas en inglés) o interfaz de programación de aplicaciones hacia el sur (*southbound API*) y una API hacia el norte (*northbound API*).

3.2.2.- Arquitectura básica de las redes definidas por software

“The goal of SDN is to provide open, user-controlled management of the forwarding hardware in a network. SDN exploits the ability to split the data plane from the control plane in routers and switches” Fei Hu y Ke Bao (2014). En comparación con las redes tradicionales, las redes definidas por software poseen 3 componentes que definen su arquitectura.

1) *Infraestructura o plano de datos*: se trata de la logística de red, routers, switches, hosts, sin incluir ningún protocolo y de cualquier fabricante, solo deben contar con el soporte del método que los conectada al controlador en este caso, el protocolo Openflow.

2) *Capa de control*: las interfaces de programación de aplicaciones representan la interfaz de software entre los módulos de la plataforma de control y las aplicaciones SDN que se ejecutan encima de la plataforma de red, aquí se mantiene la inteligencia de la red, los controladores se comunicarían con los hardware de la capa de infraestructura mediante protocolo Openflow y brindarían una API abierta para que puedan ser programados desde la capa de aplicación.

3) *Capa de aplicación*: es la de más alto nivel y desde la cual se controla el comportamiento de la red. Para ello se utiliza la API del controlador, que puede ser utilizada por usuarios o por otros Software de orquestación o aplicaciones de infraestructura.

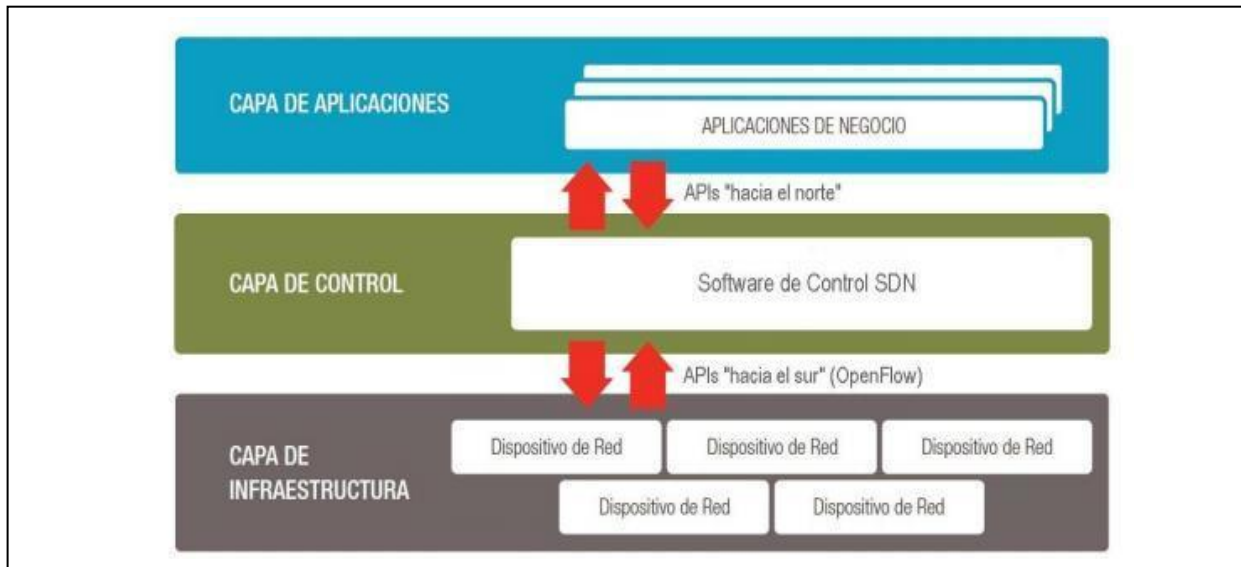


Gráfico 1. Arquitectura SDN Fuente: Citrix, 2014 Autor: Citrix

3.2.3.- Openflow

El Openflow para su funcionamiento se apoya en tres componentes: ONF (2016)

- Tabla de flujo: instaladas en todos los *routers*, que le indican a cada dispositivo que realizar con los paquetes.
- El controlador: es el cerebro, encargado de comunicar a los demás encaminadores la información que soliciten.
- Dispositivos: con soporte para Openflow, los cuales se guían por las tablas de flujo para hacer el reenvío de paquetes, si se presenta una situación en la que las tablas no la puedan ayudar, preguntan al controlador y este les dirá que realizar, por eso cada equipo debe tener incorporado la interfaz de Openflow.

El funcionamiento de este protocolo está dado porque al separar el plano de datos del plano de control, se puede tener un mejor dominio de la red, y, por tanto, mayor eficiencia. (Blandon, 2013). Destacamos que Openflow permite a las redes definidas por software resolver problemas que presentan las grandes empresas de telecomunicaciones o de servicios de internet, donde necesitan variar constantemente la configuración de dispositivos, ya sean físicos o virtuales, pero con la limitante de que los dispositivos actuales que al usar su propio “*firmware*”, cada uno define como deben ser tratados los paquetes, y al ser propietarios, son difíciles de integrar con otros. Mediante el uso de Openflow los proveedores de servicio tendrían la habilidad de procesar todo el tráfico de manera dinámica y usar una programación que pueda actuar en caso de un daño, por ejemplo, realizando un desvío de rutas, el funcionamiento de Openflow contempla esta necesidad, además de generar valor agregado en las redes, Calidad de Servicios (QoS), Tipos de Servicio (ToS), Ingeniería de Trafico, entre otros.

3.3.- Herramientas Utilizadas

Para cumplir las políticas de licenciamiento de la empresa Magozolutions SA., se escogió las siguientes herramientas, ya que las mismas no tienen costo de licenciamiento. OpenDayLight es Open Source bajo la licencia Eclipse Public License 1.0. Mininet también es Open Source bajo una licencia BSD. ATS utiliza la licencia Apache license v2. Xenserver tiene una versión gratuita bajo GNU GPL v2 y Ubuntu Server es un Sistema operativo de código abierto bajo licencia GPL.

3.3.1.- Xenserver 7.5

Xenserver es una plataforma de virtualización de servidores administrada, completa e integrada en el proyecto Open-source Hypervisor Xen. Es desarrollado y gestionado por Linux Foundation. Este hypervisor es utilizado por Xenserver como el núcleo de su arquitectura, a la que Citrix añade otros componentes propios y Xencenter, la consola de administración. (Systems, 2018).

3.3.2.- Ubuntu Server 16.04

Ubuntu Server ofrece escalabilidad económica y técnica. Ya sea que desee implementar una nube OpenStack, o un clúster, además posee certificación como invitado en AWS, Microsoft Azure, Joyent, IBM, Google Cloud Platform y Rackspace, Se ejecuta en todas las arquitecturas principales: x86, x86-64, ARM v7, ARM64, POWER8, POWER9 e IBM s390x (LinuxONE). (Ubuntu.Org, 2018).

3.3.3.- Apache Traffic Server

Apache Traffic Server (ATS) es un proxy de cacheo que permite mejorar la eficiencia y el rendimiento de la red. ATS está diseñado para mejorar la distribución de contenidos para pequeñas y grandes empresas. Principalmente suele usarse para implementar un cacheo de la información frecuentemente accedida, especialmente en los bordes de la red, de esta forma se consigue acercar el contenido a los usuarios, reduciendo el uso de ancho de banda total de la red y consiguiendo una distribución del contenido más rápida. (Apache Software Foundation, 2018).

3.3.4.- OpenDayLight

OpenDayLight actúa como controlador SDN en nuestro escenario. Es considerado por muchos el estándar de facto en la industria. Cuentan con el apoyo de la Linux Foundation, con el objetivo de avanzar en el avance de tecnologías SDN y NFV colaborando en una plataforma común *open source* como es OpenDayLight. Otros miembros encargados de apoyar ODL son Cisco, Ericsson e Intel, entre otros. Por lo que OpenDayLight tiene un gran apoyo por parte de la industria (OpenDayLight, 2018).

4.- METODOLOGÍA

En este documento académico se aplicó el método científico, ya que se observó la problemática en el cuarto de servidores de la empresa Magozolutions SA. debido al número de equipos físicos y la preocupación por la administración de los mismos, se realizó una investigación de nivel exploratorio, se usó una técnica de análisis de datos documentales para poder determinar la viabilidad de la implementación y de cómo ayudarían a la gestión del centros de datos el uso de redes definidas por software aplicados en servidores virtualizados. Magozolutions SA es una empresa dedicada al *outsourcing* de servicios de infraestructura, instalación de servidores, centrales telefónicas y servicios de asesoría informática, al momento de iniciar esta implementación contaban con ocho servidores físicos, destinados a sus diferentes clientes, equipos donde se comprobó una subutilización de recursos, tanto de memoria RAM, uso de procesadores y almacenamiento, se tomó como muestra uno de los servidores Linux, usado para alojamiento web, donde se verifica un uso de procesador

de menos del 2% y una ocupación de memoria física menor al 6% del total instalado en dicho equipo, como se evidencia en el gráfico 2.

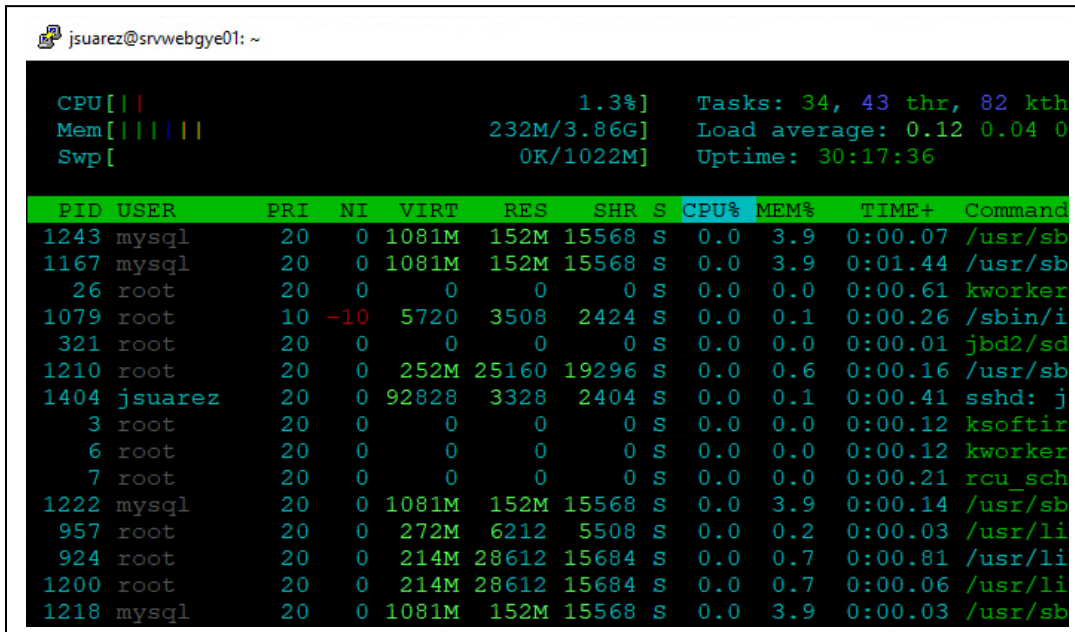


Gráfico 2. Uso de recursos. Fuente: servidor web Magozolutions, Elaborado por: Autor

En el siguiente diagrama de Gantt se resume la planificación de los procesos con su tiempo.

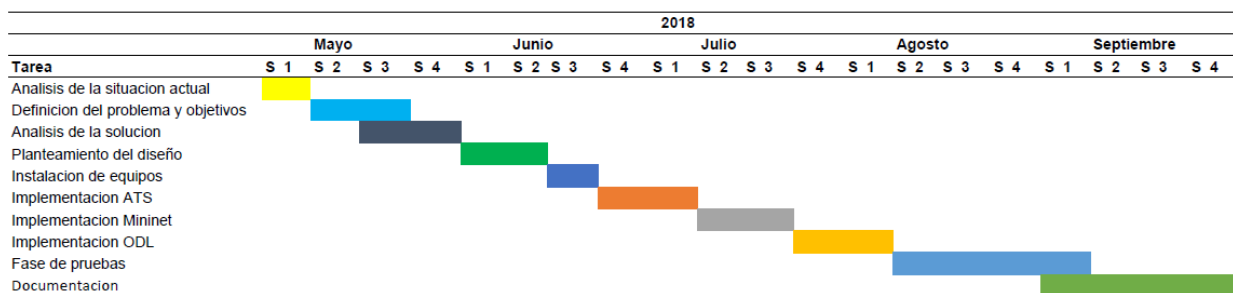


Gráfico 3. Diagrama de Gantt de la planificación del proyecto. Fuente: Autor

4.1 Análisis de la situación actual

La idea es tener un claro entendimiento del entorno de actuación y las posibles mejoras que se pudieran adoptar, adicionalmente se tuvo una visión más clara analizando proyectos con características similares a este artículo académico donde se obtuvo el pautas necesarias y actualizadas sobre virtualización y las redes definidas por software.

4.2 Planteamiento de diseño y análisis de la solución

Luego de varias reuniones en conjunto con la gerencia de Magozolutions y varios clientes donde se analizaron las diferentes opciones sobre la estructura y funcionamiento de los probables escenarios, partiendo de las características brindadas por las distintas aplicaciones, se seleccionó uno de los diseños. se revisaron las posibles limitaciones y fortalezas del escenario, en este caso se eligió la opción de Xenserver por la compatibilidad con el requerimiento de un cliente que implementara adicionalmente una solución de virtualización de escritorios, en específico Citrix Virtual Apps and Desktops.

Feature	Hyper-V 2016	vSphere 6.5	XenServer
RAM/Host	24TB	12 TB	5TB
RAM/VM	12 TB for generation 2 1 TB for generation 1	6 TB	1.5TB
CPUs/VM	240 for generation 2 64 for generation 1	128	32
VM Disk	64 TB for VHDX format 2040 GB for VHD format	62TB	2TB
VM Live Migration	Si	Si	Si
VM Replication supports	Si	Si	Si
Overcommit resources	No	No	No
Disk I/O Throttling	Si	Si	Si
Hot plug of virtual resources	Si	Si	Si

Grafico 4. Comparativa entre Hipervisores. Elaborado por: Autor

4.3 Instalación de Equipos

Para la implementación preliminarmente se realizó la puesta en marcha de un servidor de pruebas con Xenserver en su versión 7.5, donde luego se instalaron las máquinas virtuales para crear una topología sencilla, uno de esos servidores será implementado con Ubuntu Server 16.04 en el cual se agregara el servidor Apache para la gestión de las paginas web, sin usar aun de SDN, luego se implementó servidores proxy usando *Apache Traffic Server* para simular a pequeña escala balanceo de carga y tráfico sobre los servidores, finalmente se llevó a cabo un escenario sobre SDN para estudiar el comportamiento de la red SDN, una vez montada la topología se añadió la funcionalidad de balanceo de carga, haciendo uso del controlador OpenDayLight, mediante peticiones al switch donde luego se obtendría información del tráfico enviado, y en función de este se prioriza el tráfico entre un equipo u otro.

4.3.1. Instalación Xenserver

Tomando en cuenta la guía de instalación (Citrix, 2016), los requerimientos hardware de Xenserver 7.5 son:

- Uno o más procesadores 64-bit x86, se recomienda de 2 GHz, con soporte Intel VT o AMD-V 64-bit
- Se recomienda 4GB en RAM
- Almacenamiento en disco local: 70GB recomendado.
- SAN vía HBA, si se utiliza multipath boot desde SAN.

- Interfaz de red de 100 Mbits mínimo. Para la importación y exportación y migración de máquinas virtuales en caliente se recomienda usar varias interfaces de 1000 Mbits.

El ISO de instalación se debe descargar desde <http://www.xenserver.org/overview-xenserver-open-source-virtualization/download.html>. Luego de seleccionar la distribución del teclado, en la pantalla de bienvenida o “*Welcome to XenServer Setup*” OK (si es necesario cargar algún driver, se debe oprimir F9). Se procedió a leer y aceptar el Contrato de licencia de usuario final o EULA por sus siglas en ingles.

En la pantalla “*Virtual Machine Storage*” elegimos el o los discos que se usaran en el almacenamiento local. Los mismo que conformarán un sólo repositorio. (Bustamante, 2018). Seleccionamos el origen de la instalación, en nuestro caso, “Local media”, para agregar funciones añadidas se habilita según sea el caso *Supplemental Packs*. Luego se configura la contraseña para el usuario *root* del servidor. En la pantalla de networking elegimos que interfaz se usara para la gestión y configuramos la dirección IP; máscara de red y puerta de enlace, posteriormente se especificamos el *hostname* de nuestro servidor que en nuestra implementación denominamos UTEG-SRV01, así como las direcciones de los servidores DNS. Adicionalmente se especificó los servidores NTP a usar, en nuestro caso, “inocar.ntp.ec”. Confirmamos la instalación de XenServer 7.5, luego de terminado el proceso se reinició el servidor.

El software de XenCenter se obtiene desde <http://www.xenserver.org/overview-xenserver-open-source-virtualization/download.html> o en la imagen de de XenServer, en la carpeta “client_install”.

Luego añadimos los servidores a administrar en la consola de Xencenter usando las credenciales que se configuraron durante la instalación.

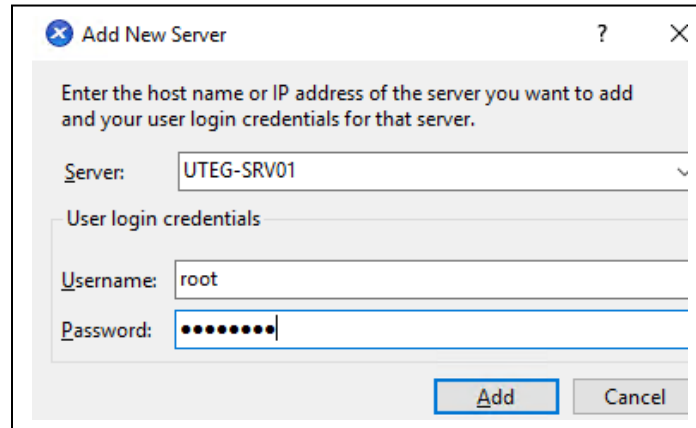


Gráfico 4. Fuente Xencenter. Elaborado por: Autor

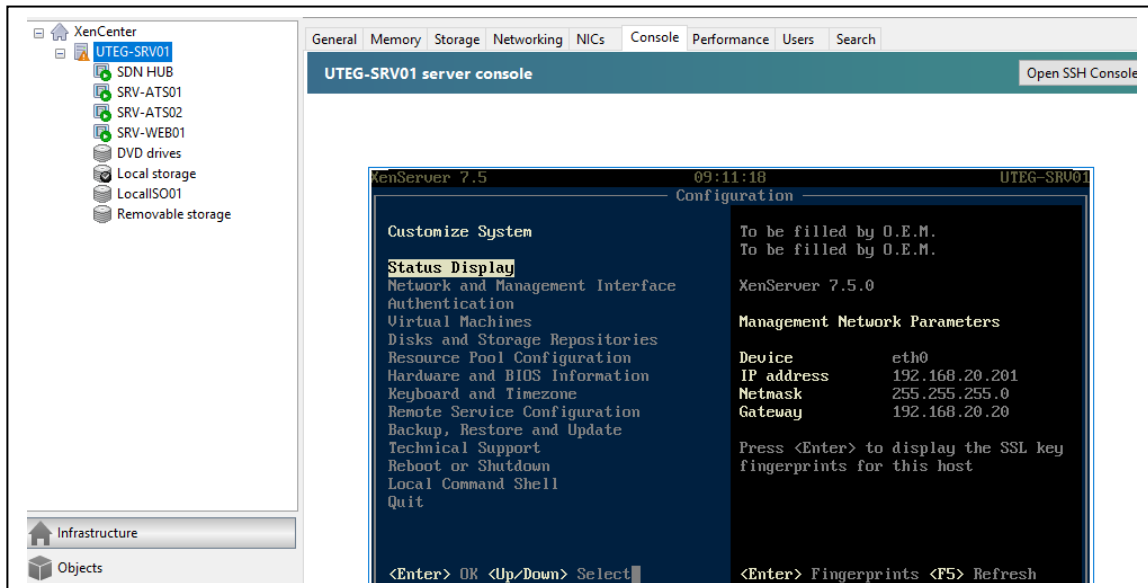


Gráfico 5. Fuente Xencenter. Elaborado por Autor

4.3.2 Instalación servidor web

Se copia la imagen ISO de Ubuntu server 16.04 en el repositorio ISO, en Xencenter se hace clic derecho sobre uno de los servidores y se elige "*New VM*" y se inicia el asistente de creación de máquinas virtuales, elegimos la plantilla, luego indicamos el nombre y descripción de la máquina virtual que se está creando, en este caso SRV-WEB, posteriormente se procede a elegir el origen de instalación, luego seleccionamos el "home server" o servidor en que iniciara la VM por defecto, en la sección "*Allocate processor and memory resources*" asignaremos la cantidad de CPUs virtuales, así como la memoria RAM, en "*Configure Storage*" donde colocaremos el tamaño del o los discos VDI de nuestra VM. En la ventana "*Configure Networking*" configuraremos las interfaces de red, finalmente repasaremos la configuración asignada. Posterior a esto Xencenter aprovisiona la nueva maquina virtual, una vez creada, desde la pestaña de consola podremos seguir con la instalación habitual del sistema operativo, en este caso Ubuntu Server 16.04 donde agregaremos Apache para la gestión del servicio web.

4.3.3 Instalación Apache Traffic Server

Luego se procedió con la instalación de las maquinas con *Apache Traffic Server*, denominadas SRV-ATS01y SRV-ATS02, las mismas que formando un clúster realizaran el cacheo del contenido, recibiendo las peticiones y en caso de no tener la información en su cache, consultaran al servidor web, manteniendo actualizada dicha información en el clúster. Estos equipos hicieron de proxy y cache, para esto fue necesario modificar el archivo de asignación *remap.config*, con el objetivo de que las peticiones que vayan a los equipos 192.168.20.221 y 192.168.20.222 sean dirigidos al equipo 192.168.20.211 para esto se añadió la siguiente línea en el archivo *remap.config*:

```
map 192.168.20.221:80 192.168.20.211:80
```

4.3.4 Instalación OpenDayLight

Para la ejecución de OpenDayLight se usó de una máquina virtual denominada SDNHub, que contiene todas las aplicaciones para emular el escenario SDN. (SDN Hub, 2018), misma que descargamos desde <http://sdnhub.org/tutorials/sdn-tutorial-vm/>. En esta implementación, se necesita que el controlador sea capaz de modificar la tabla de flujo del conmutador para que realice correctamente el proceso de *routing* cuando se realiza una petición de un equipo a otro. Instalando el paquete de "*learning switch*", se consigue esta capacidad. Se habilito mediante el comando:

```
feature:install sdnhub-tutorial-learning-switch
```

4.3.5 Instalación de Mininet

Mininet es un emulador de redes capaz de establecer redes virtuales. Los hosts virtuales ejecutan Linux estándar, y los switches tienen soporte Openflow, de manera que puede crearse una topología SDN con los equipos emulados en Mininet.

Se comenzaría añadiendo un controlador, llamado "c0" con Openflow:

```
c0 = net.addController( name='c0', controller=RemoteController,  
                        protocols='OpenFlow13', ip='127.0.0.1', port=6633)
```

Posteriormente, se agregará un switch con la MAC '00:00:00:00:00:10':

```
s1 = net.addSwitch('s1', cls=OVSSwitch,  
mac='00:00:00:00:00:10', protocols='OpenFlow13')
```

Este switch es creado con 3 interfaces para poder conectar los dos equipos ATS y el servidor web. De forma similar a la siguiente:

```
intfName= 'eth1'  
Intf( intfName, node=s1 )
```

Finalmente se añadió el host con una MAC "00:00:00:00:00:01" y una IP "192.168.20.100". Este actuaría de cliente, para llevar a cabo las peticiones al servidor:

```
h1 = net.addHost('h1', mac='00:00:00:00:00:01',  
ip='192.168.20.100/24', defaultRoute='via 192.168.20.20')
```

Para el balanceo de carga sería necesario hacer uso de una IP y una Media Access Control (MAC) virtual de manera estática, de forma que a esta dirección IP será a la que el cliente haga la petición HTTP. Posteriormente, el switch envía el paquete al controlador, y será este el que decida a que dirección IP se redirigirá la petición enviada a la dirección virtual:

```
h1 arp -s 192.168.20.100 00:00:00:00:00:10
```


En el switch se va a especificar una política que hará que las peticiones que van destinadas a la IP virtual sean redirigidas a una tabla de grupo de tipo *select*. La regla es la siguiente:

```
sudo ovs-ofctl --protocols=OpenFlow13 add-flow s1 ip,nw  
dst=192.168.20.100,priority=32769,actions=group:1
```

Una vez configurados todos los componentes necesarios llevaremos a cabo la ejecución.

1. El usuario lleva a cabo una petición HTTP a una IP virtual, la IP 192.168.20.100.
2. La petición llega al *switch*, el cual tiene la regla que dirige las peticiones a la IP virtual a la tabla de tipo *select*.
3. La llegada de un paquete al controlador activa la aplicación encargada de comprobar los paquetes transmitidos por cada puerto, mediante una petición "GET" al *switch*. En caso de superar el umbral del puerto actualmente utilizado, la tabla sera modificada mediante un "PUT" a la tabla de grupo del *switch*, de forma que asignara mayor prioridad al otro puerto. En caso de no superar el umbral, las prioridades de la tabla no serán modificadas.
4. La petición es enviada al equipo correspondiente del *bucket* elegido de la tabla.
5. El equipo con *Traffic Server* sirve la información en caso de tenerla en cache. En caso de no tenerla lo consulta al servidor web, y posteriormente la guardaría en cache y lo enviaría al cliente.
6. El usuario recibe el contenido solicitado.

5.- Resultados

Luego de la implementación verificamos que hemos cumplido con nuestros objetivos, tales como reducir costos operacionales, al migrar servidores físicos hacia una plataforma de virtualización, reducimos el consumo de energía eléctrica tanto de los servidores como del sistema de climatización, ya que, al disminuir el número de equipos encendidos, se genera menos calor en el cuarto de equipo.

	Cantidad	Potencia (W)	BTU x Equipo	KWH al Mes por equipo	Costo por KWH	Costo al Mes por equipo	Costo total al mes
HP ML350 G6	8	320W	1794	230 KWH	\$0.14	\$32.2	\$257.6
Dell R730	2	450W	1547.1	326 KWH	\$0.14	\$45.64	\$91.28

Gráfico 6. Comparativo consumo energía eléctrica de servidores. Elaborado por: Autor

También optimizamos los recursos físicos de nuestro servidor de virtualización, como verificamos en el gráfico 7, tenemos asignado en los 4 servidores virtualizados, alrededor del 70% de memoria RAM, la misma que puede ser reasignada aumentando o disminuyendo la cantidad según se necesite, así mismo con la asignación de los núcleos de procesador.

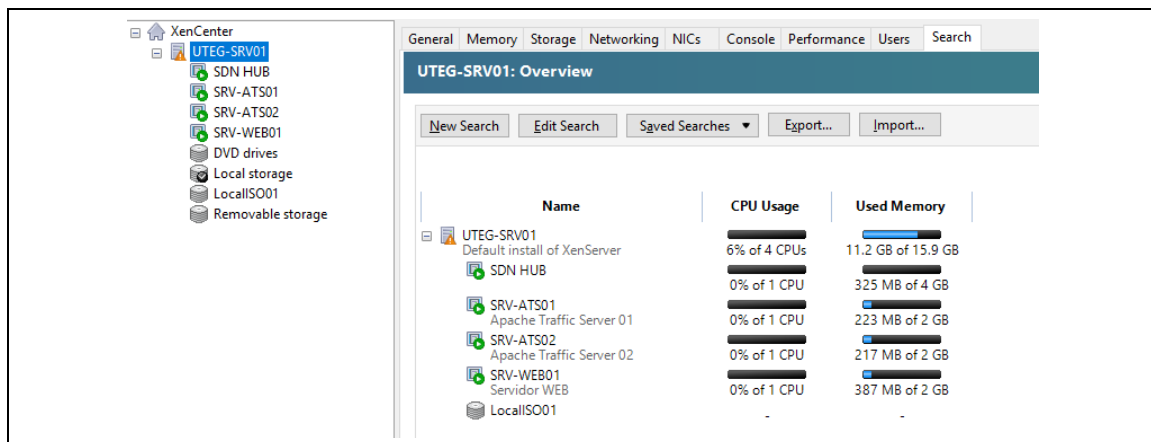


Gráfico 7. Asignación de recursos. Fuente Xencenter. Elaborado por: Autor

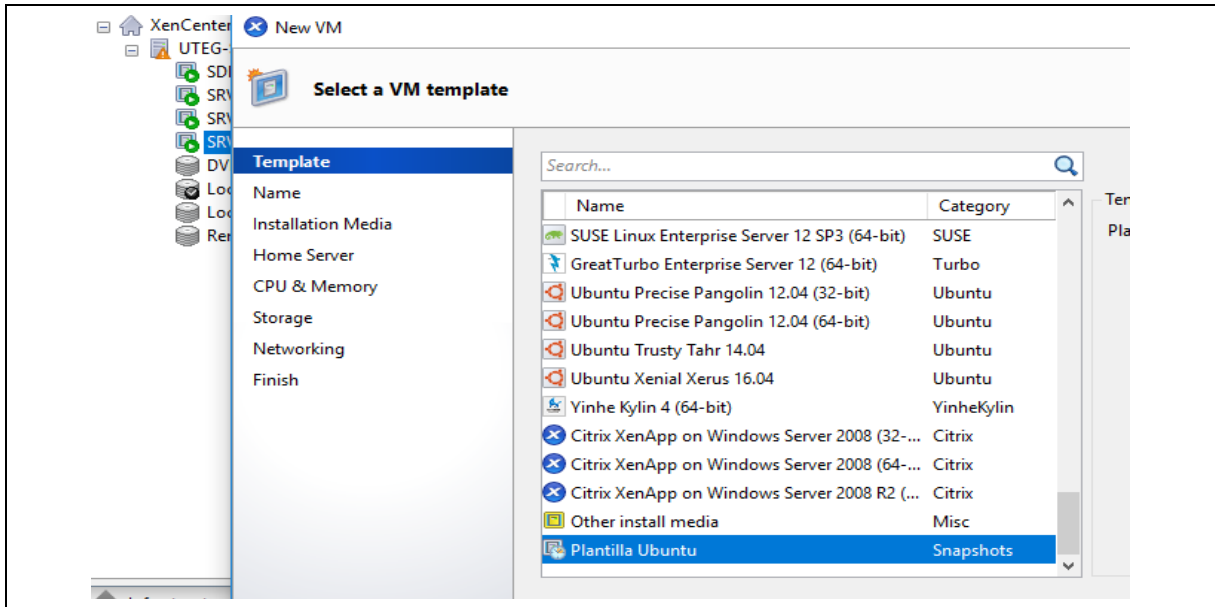


Gráfico 8. Uso de snapshots. Fuente Xencenter. Elaborado por: Autor

Se redujo el tiempo para aprovisionar nuevos servicios a tan solo minutos, se uso una plantilla o *snapshot* de un servidor base, para realizar nuevas implementaciones.

Detallado en los gráficos 8 y 9

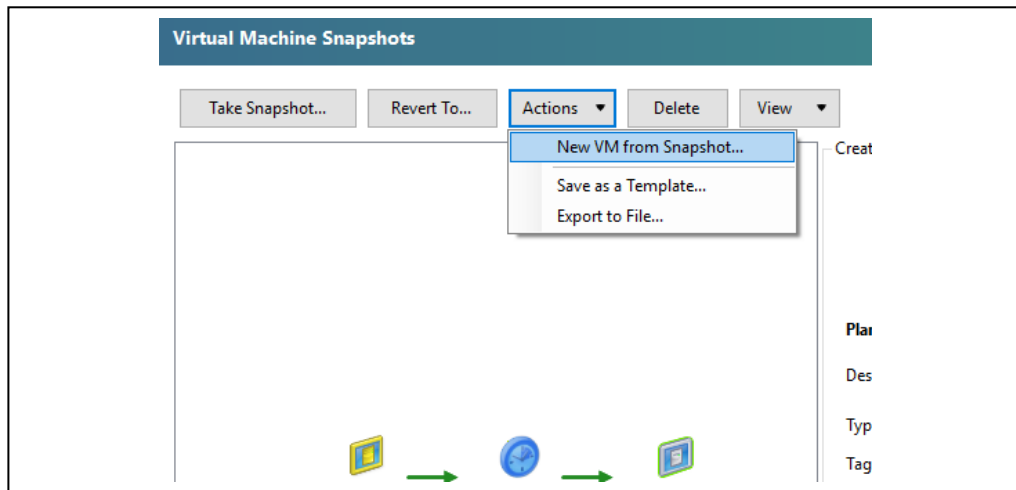


Gráfico 9. Uso de Snapshot. Fuente Xencenter. Elaborado por: Autor

Adicionalmente se redujo la complejidad en la administración, al tener el control de todos los servidores, desde la aplicación Xcenter.

Se verificó la factibilidad de aplicar la lógica del controlador a una red definida por software para ayudar a los administradores de red a obtener un balanceo de carga entre sus equipos, incluso usando mayor cantidad de parámetros de la red como el número de errores en un puerto, paquetes descartados, disponibilidad de enlaces entre otros, para así optimizar el balanceo de carga.

6.- CONCLUSIONES

El objetivo principal consistía en verificar como ayudarían a la gestión de centros de datos el uso de redes definidas por software aplicados en servidores virtualizados, el mismo que se ha alcanzado con esta implementación, donde se optimizó recursos, se logró tener tolerancia a fallos.

Así mismo, se verificó que es factible reducir el tiempo para aprovisionar nuevos servidores con el uso de las herramientas, en este caso el Xcenter nos permite realizar una instalación base y guardarla como plantilla para posteriores implementaciones.

A su vez, a través del desarrollo de este artículo se evidenció que es posible reducir costos operativos no solo en el área de tecnología de la empresa sino en general, al disminuir el número de servidores físicos en producción, se redujo el uso de espacio físico y del sistema de climatización del cuarto de equipos, generando un ahorro de energía y reduciendo la huella de carbono, disminuyendo a si el impacto al medio ambiente.

Además, mediante el uso de redes definidas por software, se verificó la factibilidad de obtener balance de carga entre servidores y una gestión unificada con el uso de reglas de control de tráfico.

7. Recomendaciones

En cuanto a la implementación de futuros sistemas se recomienda seguir con el uso de herramientas *Open Source*, que generarían ahorro en licencias, disminuyendo los costos operativos.

Como en toda implementación de servidores, se recomienda las buenas prácticas el aprovisionamiento de los mismos, y se aconseja mantener un monitoreo tanto de los servidores como del hipervisor, además de implementar una política de respaldos de configuraciones y de archivos, así mismo se sugiere mantener actualizados el software para evitar posibles vulnerabilidades.

Una de las principales recomendaciones refiere a mantener la hoja de ruta trazada con esta implementación no solo por la optimización de recursos que benefician a la empresa sino por la responsabilidad social beneficiando al medio ambiente por la disminución de la huella de carbono resultado del uso de prácticas conocidas como Green Computing.

Se recomienda ampliar la investigación sobre SDN, existen muchas opciones debido a la flexibilidad de configuración. En especial un futuro uso en conjunto con IPV6 cuyo uso permitiría la implementación de más servicios en particular con el despliegue del Internet de las Cosas.

8.- REFERENCIAS

Lugo, N. (2014). Tecnologías de virtualización en los sistemas informáticos de las organizaciones empresariales del Estado Zulia. Revista Electrónica de Estudios Telemáticos, 13, 49-67. Recuperado de <https://dialnet.unirioja.es/descarga/articulo/5157975.pdf>

Casierra Cavada, J., Quiñonez Ku, X., Herrera Izquierdo, L., Egas Acosta, C. (2018). *Virtualización de Redes y Servidores Emulando Infraestructuras Tecnológicas*. Revista Científica Hallazgos21, 3(Suplemento Especial). Recuperado de <http://revistas.pucese.edu.ec/hallazgos21>

Call , J. *Que es una máquina virtual, usos y ventajas que nos proporciona*. Recuperado de Geekland Blog de Tecnología: <https://geekland.eu/que-es-una-maquina-virtual-usos-y-ventajas-que-nos-proporciona/>

Jácome Segovia, D., Núñez Núñez, J., Velasteguí López, E., Navas Moya, M., & Vásquez Carrera, P. (2018). LA VIRTUALIZACIÓN DE SERVIDORES COMO UNA HERRAMIENTA PARA LA OPTIMIZACIÓN DE RECURSOS. Ciencia Digital, 2(2), 277-301. Recuperado a partir de <http://www.cienciadigital.org/revistascienciadigital/index.php/CienciaDigital/article/view/93>

Gallego, J. (2014). Operaciones Auxiliares para la configuración y la Explotación. Madrid: Editex.

Open Networking Foundation, «Software-Defined Networking (SDN) Definition», 2018. Recuperado de: <https://www.opennetworking.org/sdn-definition/>

Dueñas Santos, Carlos & Muro, Yanko Antonio & Cruz-Enriquez, Héctor. (2017). Analysis of the performance of an SDN network over a traditional network in LAN environments. Recuperado de:

https://www.researchgate.net/publication/321197856_Analysis_of_the_performance_of_an_SDN_network_over_a_traditional_network_in_LAN_environments

García Centeno, Alejandro & Rodríguez Vergel, Carlos Manuel & Anías Calderón, Caridad & Camilo, Frank & Bondarenko, Casmartiño & , Uci. (2014). Controladores SDN, elementos para su selección y evaluación. Telemática. Recuperado de :

<https://docplayer.es/45541382-Controladores-sdn-elementos-para-su-seleccion-y-evaluacion.html>

Manar Jammal, Taranpreet Singh, Abdallah Shami, RasoolAsal, y Yiming Li, «Software Defined Networking: State of the Art and Research Challenges», 2016

Fei Hu y Ke Bao, «A Survey on Software Defined Network and OpenFlow: From Concept to Implementation», recuperado de <https://ieeexplore.ieee.org/abstract/document/6819788> 2014

Open Networking Foundation, ONF SDN Evolution, 2016, Recuperado de: http://3vf60mmveq1g8vzn48q2o71a-wpengine.netdna-ssl.com/wp-content/uploads/2013/05/TR-535_ONF_SDN_Evolution.pdf

Citrix, 2014, sdn-101-an-introduction-to-software-defined-networking-es. Recuperado de: https://www.citrix.com/content/dam/citrix/en_us/documents/oth/sdn-101-an-introduction-to-software-defined-networking-es.pdf

Blandon Gomez,D "Openflow El protocolo del futuro," 2013. : Recuperado de:
<https://dialnet.unirioja.es/servlet/articulo?codigo=4897871>.

Systems, C. (2018). *Revisión del Servidor de virtualización con código abierto XenServer*.
Obtenido de <http://xenserver.org/overview-xenserver-opensource-virtualization.html>

Ubuntu. (2018). Ubuntu.com. Obtenido de <http://www.ubuntu.com/desktop>

Apache Software Foundation. (2018)
<https://docs.trafficserver.apache.org/en/latest//index.en.html#what-is-apache-traffic-server>

OpenDaylight: Open Source SDN Platform. <https://www.opendaylight.org>.

Citrix, 2016. Citrix XenServer ® 7.0 Quick Start Guide. Recuperado de:
<https://docs.citrix.com/en-us/xenserver/7-0/downloads/quick-start-guide.pdf>

Eladio Bustamante Herrera ,Miniacademia (2018). Recuperado de:
<http://www.miniacademia.es/xenserver-instalacion-de-xenserver-y-xencenter/>

SDN Hub, 2018, All-in-one SDN App Development Starter VM, Recuperado de
<http://sdnhub.org/tutorials/sdn-tutorial-vm/>