ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

ESCUELA DE INGENIERÍA EN ELECTRÓNICA TELECOMUNICACIONES Y REDES

"ANÁLISIS DE VULNERABILIDADES INSIDER CONTRA ATAQUES DE DENEGACIÓN DE SERVICIO (DoS) EN REDES DEFINIDAS POR SOFTWARE."

TRABAJO DE TITULACIÓN:

TIPO: PROPUESTA TECNOLÓGICA

Presentado para optar al grado académico de:

INGENIERO EN ELECTRÓNICA, TELECOMUNICACIONES Y REDES

AUTORES: DANIELA MARLITH TOAINGA URRUTIA

DANIEL ROBERTO PEÑA PÉREZ

TUTOR: ING. MARCO VINICIO RAMOS VALENCIA.

Riobamba - Ecuador

2019

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZZO

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

ESCUELA DE INGENIERÍA ELECTRÓNICA EN TELECOMUNICACIONES Y REDES

El Tribunal del Trabajo de Titulación certifica que: El Trabajo de titulación: **"ANÁLISIS DE VULNERABILIDADES INSIDER CONTRA ATAQUES DE DENEGACIÓN DE SERVICIO (DoS) EN REDES DEFINIDAS POR SOFTWARE",** de responsabilidad de la señorita Daniela Marlith Toainga Urrutia y el señor Daniel Roberto Peña Pérez, ha sido minuciosamente revisado por los Miembros del Tribunal de Trabajo de Titulación quedando autorizado su presentación.

NOMBRE	**FIRMA**	**FECHA**
ING. WASHINGTON LUNA **DECANO DE LA FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**	_____	_____
ING. PATRICIO ROMERO **DIRECTOR DE LA ESCUELA DE INGENIERÍA ELECTRÓNICA, TELECOMUNICACIONES Y REDES**	_____	_____
ING. VINICIO RAMOS VALENCIA **DIRECTOR DEL TRABAJO DE TITULACIÓN**	_____	_____
ING. HUGO MORENO AVILÉS **MIEMBRO DEL TRABAJO DE TITULACIÓN**	_____	_____

Nosotros, Daniela Marlith Toainga Urrutia y Daniel Roberto Peña Pérez, somos responsables de las ideas, doctrinas y resultados expuestos en este Trabajo de Titulación y el patrimonio intelectual de la Trabajo de Titulación, pertenece a la Escuela Superior Politécnica de Chimborazo.

Daniela Marlith Toainga Urrutia

Daniela Roberto Peña Pérez

DEDICATORIA

A Dios por ser mi guía y apoyo en los momentos difíciles; por su infinito amor y bendiciones reflejas a lo largo de mi vida.

A mis padres Narciza y Enrique que han sembrado en mi espíritu y mente valores convirtiéndome en la persona que soy; por ser mi ejemplo de superación y perseverancia he logrado conseguir una meta más; por su amor; por su constante apoyo y siempre confiar en mí. A mis abuelitos Victor y Doraliza, y a mis tías: Deysi, Diana, Adriana y Melani; por su apoyo incondicional; por sus palabras de motivación en todo momento que me han hecho crecer como persona.

A mis amigos que han compartir conmigo gratos momentos, por su comprensión y ayuda desinteresada; y por compartir momentos inolvidables que quedaran guardados en lo más profundo de mi ser.

Daniela

Este trabajo va dedicado a mis padres Beatriz y Hugo que han sido un ejemplo de lucha para poder superarme día tras día siendo mi fortaleza y mi apoyo en los momentos felices y complicados de esta carrera tan hermosa y brindándome su compresión cariño y amor.

A mis hermanos Cristina y Hugo que fueron motivación e inspiración para superarme día tras día, apoyándome incondicionalmente a lo largo de este camino haciendo que este sueño se haga realidad.

Daniel

AGRADECIMIENTO

En este día en particular; quiero agradecer a Dios; por su amor y bondad infinita; por permitirme gozar y disfrutar de una buena salud; por brindarme sapiencia para poder asimilar con madurez los momentos difíciles y haberme dado la oportunidad de superarme para llegar a terminar una etapa más de mi vida.

A mis padres, abuelitos y tías; por ser el pilar fundamental de mi vida; por el amor que siempre me han demostrado; por el apoyo, confianza y fortaleza que me han brindado para enfrentar las dificultades; por su comprensión ante mis debilidades; por compartir conmigo triunfos y fracasos para verme cumplir mis sueños y anhelos.

A mis amigos y compañeros; por compartir sus conocimientos; por su comprensión y ayuda en momentos buenos y malos.

Daniela

Agradezco en primer lugar a Dios, a mis padres, hermanos, familia y amigos por su apoyo, paciencia y preocupación en esta larga y maravillosa trayectoria formativa este logro es en gran parte de ustedes por su incansable apoyo en cada paso de esta maravillosa trayectoria.

Un agradecimiento especial para mi tía Olguita quien con sus palabras me dio un nuevo aire para seguir en este camino por apoyarme y ser un guía más en este interminable proceso.

Daniel

TABLA DE CONTENIDOS

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

ÍNDICE DE GRÁFICOS

ÍNDICE DE ANEXOS

ÍNDICE DE ABREVIATURAS

SDN	Software Defined Networking
DoS	Denial of service
ONF	Open Networking Foundation
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
IP	Internet Protocol
DNS	Domain Name Service
DHCP	Dynamic Host Configuration Protocol
FTP	File Transfer Protocol
VoIP	Voice over IP
HTTP	Hypertext Transfer Protocol
ICMP	Internet Control Message Protocol
ISP	Internet Service Provided
API	Application Programming Interface
SSL	Secure Socket Layer
IPv6	Internet Protocol version 6
OVS	Open vSwitch

ODL	OpenDayLight
OCTAVE	Operationally Critical Threat, Asset and Vulnerability Evaluation
GUI	Graphical User Interface
RAM	Random Access Memory
CAPEX	Capital Expenditure
OPEX	Operational Expenditure
CERT	Computer Emergency Response Team
UPnP	Universal Plug and Play
DDoS	Distributed Denial of Service
SSH	Secure Shell
SMB	Server Message Block
PING	Packet Internet Groper

RESUMEN

El objetivo de este trabajo de investigación consistió en el análisis de vulnerabilidades de ataques de denegación de servicio DoS en redes definidas por software (SDN), para lo cual se implementó una red simulada en el software GNS3 VM, con varios dispositivos: controlador Opendaylight, open vswitches, clientes y servidores DHCP, DNS, HTTP, VoIP y FTP. Para el análisis de vulnerabilidades, se optó por usar la metodología OCTAVE, comprendida en tres etapas; primera: identificación activos y amenazas de la organización, segunda: escaneo de vulnerabilidades y tercera: implementación del plan de contingencia. Para el desarrollo de la fase dos se usa la herramienta Openvas que ayuda a detectar fallas en la red, así como sus características y dispositivos afectados, luego de eso se clasificaron para determinar las que afectan la disponibilidad, como son: versión y tipo del servidor HTTP, acceso con fuerza bruta HTTP, version del demonio del protocolo BIND de DNS y Detección de tiempo de marca ICMP. Una vez completado este paso se ejecutaron los ataques de denegación de servicio: HTTP, DHCP y DNS, para agotar los recursos utilizados por el sistema, y poder evidenciar el comportamiento de la red, basándose en los indicadores: ancho de banda y latencia. Se concluye que en el caso del ancho de banda antes del ataque, el impacto tuvo una efectividad no superior al 34%, pero después incluso arrojaron valores de 71% y 84%, en tanto la latencia antes de la amenaza fueron menores a 1,5 milisegundos y después sube a 4,779 milisegundos, lo que significa que existe sobre-procesamiento. Y el tercer paso es la elaboración de una guía de buenas prácticas en donde se evidenciaron una disminución de las vulnerabilidades. Se recomienda cambiar de credenciales al controlador para evitar ingresos no autorizados y también crear reglas de flujo exactas para cada acción.

PALABRAS CLAVES: <REDES DEFINIDAS POR SOFTWARE>, <ATAQUES DE DENEGACIÓN DE SERVICIO>, <ANÁLISIS DE VULNERABILIDADES>, <LATENCIA>, <ANCHO DE BANDA>, <CONTROLADOR OPENDAYLIGHT>, <OPENFLOW>.

ABSTRACT

The objective of this research work consisted in the analysis of vulnerabilities of DoS denial of service attacks in software-defined networks (SDN), for which a simulated network was implemented in the GNS3 VM software, with several devices: Opendaylight controller, open switches, clients, DHCP and servers. DNS. HTTP. VolP and FTP. For the analysis of vulnerabilities, it was decided to use the OCTAVE methodology, comprised in three stages; first: identification of assets and threats of the organization, second: vulnerability scanning and third: implementation of the contingency plan. For the development of phase two, the Obsolete Open vas was used to help select (alias in the network, as well as its affected features and devices, after that they were classified to determine those that affect availability, such as version and HTTP service type, brute-force login in HTTP with default credentials, determine the ICMP dialing time After completing this step, the Denial of Service Ships were executed: HTTP, DHCP and DNS, to exhaust the resources used by the system, and be able to demonstrate the behavior of the network, based on the indicators: bandwidth and latency. It concludes that in the case of the bandwidth before the attack, the impact was not more than 34% effective, but later they even showed values of 71% and 84%, while the latency before the threat were less than 1.5 milliseconds and then it rises to 4,779 milliseconds, which means that there is over processing. At the same time, a guide of good practices was developed, where a decrease in vulnerabilities was observed. It is recommended to change credentials to the controller to prevent unauthorized entries and also create exact flow rules for each action.

KEY WORDS: <NETWORKS DEFINED BY SOFTWARE>, <DENIAL OF SERVICE ATTACKS>, <VULNERABILITY ANALYSIS>, <LATENCY>, <BANDWIDTH>, <OPENDAYLIGHT CONTROLLER>, <OPENFLOW>.

INTRODUCCIÓN

ANTECEDENTES

Últimamente, la creación de nuevas tendencias tecnológicas, como, la implementación de servicios en la nube "cloud computing", nuevas aplicaciones en tiempo real, más dispositivos enlazados a la red, la big data, entre otras, son algunos factores que influyen en las redes puesto que demandan mayores requerimientos. Por lo tanto, las redes necesitan tener mayor escalabilidad y flexibilidad y no depender de protocolos en muchos casos definidos por los mismos propietarios de los equipos fabricantes, para de esa manera afrontar los inconvenientes presentes en las redes actuales y a su vez lograr comunicaciones más rápidas y eficientes. (Álvarez, 2015, p. 5)

A partir de ello nace una solución conocida como Software Defined Networks (SDN), trayendo consigo un numero de soluciones para los usuarios que constantemente cambian de necesidades, con el objetivo de reaccionar de forma dinámica a los recursos demandados por aplicaciones, la mayoría de ellas en tiempo real, para así evitar interrupciones en sus servicios. Las SDN, además, reducen costos de administración y operación y sobretodo brindan flexibilidad, dinamismo y escalabilidad para implementar aplicaciones que requieren grandes requerimientos.

Las redes definidas por software SDN son un paradigma de red innovador, con una arquitectura, que separa al plano de datos del plano de control: el primero responsable de la conmutación de los paquetes y el segundo de las funciones de gestión de red; lo que permite a las redes ser más programables, flexibles y automatizables. Al separar estos planos, toda la inteligencia de la red se centra en un dispositivo llamado controlador que tiene una visión, encargado de la configuración, control y administración de todo el sistema y para la conectividad con los diferentes planos usa el protocolo estándar OpenFlow, desarrollado por la ONF. El protocolo OpenFlow manipula, identifica y controla todo el tráfico que transita por la red, de acuerdo a reglas previamente predefinidas conforme a los flujos. (Velazquez, 2013, p. 1)

El controlador SDN es considerado como el corazón de esta tecnología, puede ser tanto una solución en software como hardware; aquí se configuran las distintas reglas tráfico de flujos de acuerdo a políticas y también es el encargado de monitorizar toda la red. Este dispositivo usa interfaces programables llamadas APIs, gracias a ellas los administradores de red pueden desarrollar sus aplicaciones fácilmente de acuerdo a sus necesidades, por ejemplo, para brindar seguridad, balancear la carga de aplicaciones, calidad de servicio (QoS), envió de tráfico, etc.

Existen dos interfaces de comunicación: la Northbound API usada para conectar con el plano de aplicaciones y la Southbound. API para el plano de datos. (Rodrigues et al., 2015, p. 1)

Conforme, se siguen con los avances tecnológicos, existen algunas vulnerabilidades o pequeños fallos que pasan por desapercibidas al momento de su desarrollo, siendo blancos fáciles para realizar cualquier fechoría, por ejemplo, cuando se quiera ingresar a perjudicar o realizar alguna maniobra perjudicial a cierta empresa, las personas o miembros aprovecharán estos puntos estratégicos. Las amenazas provienen tanto del exterior como del interior de las organizaciones, pero las más difíciles y más frecuentes son las que se hacen desde el interior, a este tipo se las conoce como amenazas insider, provocadas por personal que está familiarizado con el entorno, es decir, que conocen perfectamente la infraestructura de red

Las Redes Definidas por Software, al igual que todas las redes tradicionales están expuestas a fallas. Entre las vulnerabilidades a las cuales están expuestas las SDN son: fallos en su programabilidad, uso de softwares con licencias libres, o el mismo hecho de que es centralizado; si alguien llegara a ingresar al sistema podría causar graves daños. Según un estudio de Kaspersky Lab, la amenaza que más atenta a este tipo de redes son los ataques de denegación de servicio (DoS) atentando contra la propiedad de la seguridad como es la disponibilidad de los servicios.

La principal función de los ataques de denegación de servicio, es agotar el ancho de banda y recursos de red, interrumpiendo servicios, a través de la generación de peticiones en tiempos cortos generados desde una o múltiples fuentes. Según, estudios de Kasperky Lab, los DoS, ocurren diariamente, con una media aproximada de 500 y un total de 16 millones de paquetes por segundo. Además, este tipo de ataques son muy usados como intermediarios, para camuflar otros tipos de amenazas, como, virus, gusanos, troyanos, entre otros. (Ocampo y et al, 2017, p. 1-2)

En Ecuador, los ataques de denegación de servicio han incrementado causando graves consecuencias en varias organizaciones, un ejemplo reciente es una amenaza, ocurrida en 2017, por parte del grupo Anonymus, a través de las redes sociales lo cual afecto a varias páginas gubernamentales provocando una saturación de las mismas.

FORMULACIÓN DEL PROBLEMA

¿Cómo se pueden analizar las vulnerabilidades de seguridad contra ataques DoS en redes definidas por software?

SISTEMATIZACIÓN DEL PROBLEMA

¿Cuáles son las vulnerabilidades en las Redes SDN?
¿Qué impacto tienen los ataques DoS en las redes SDN?
¿Qué medidas son usadas para disminuir los ataques Denegación de Servicio en una Organización?
¿Cuáles son la consecuencia de un ataque DoS en Redes Definidas por Software?

JUSTIFICACION TEÓRICA

Todas las redes están expuestas a riegos en lo que respeta a su seguridad y las redes SDN no son la excepción, puesto al estar basadas en leguajes de programación, cualquier persona puede escribir reglas de acuerdo a su conveniencia, además el uso softwares libres y una administración centralizada, por parte del controlador, en caso de ingresar a este dispositivo se podrá observar toda la información de la red.

Las redes SDN son aptas para optimizar recursos y brindar un dinamismo en las aplicaciones que requiere el usuario, por lo tanto, están propensas a ataques informáticos como: los ataques DoS, fuerza bruta, ataques de diccionario, entre otros. Pero los más riesgosos y realizados en estas infraestructuras de comunicaciones son los DoS o ataques de denegación de servicio, por su rápida expansión y fácil desarrollo, en el cual el atacante genera una cantidad masiva de peticiones a un servidor, provocando una sobrecarga de los mismo y por consiguiente que los servicios se vean afectados hacia el usuario final. Una de las varias soluciones para contrarrestar los DoS es identificar la dirección IP de la máquina del origen o fuente de la amenaza para bloquearla o a. (Peña, 2017)

Por lo tanto, el presente proyecto pretende estudiar las principales vulnerabilidades internas o insider que presentan las redes definidas por software (SDN), para luego desarrollar un escenario aplicativo con la generación de varios ataques de denegación de servicio (DoS) y finalmente analizar los resultados antes y después de la generación de los mismos con el propósito de implementar medidas que ayuden a mitigar o contrarrestar los ataques, para así brindar una mejor seguridad a la infraestructura de red.

JUSTIFICACIÓN APLICATIVA

Las redes SDN se encuentran en una etapa de desarrollo y su principal característica es la centralización de todas las funciones de red en el controlador, este dispositivo es el que permite

una visión global de todo el entorno de la red, además allí es donde se configuran todas las acciones de control por parte del administrador. Por lo tanto, toda la responsabilidad recaerá sobre el mismo, ya que si este está funcionando correctamente toda la red estará protegida.

Para desarrollar el tema propuesto se lo realizará de manera simulada, usando un software que cumpla con todos los requerimientos del escenario a plantearse. Como se mencionó con anterioridad en el controlador se alojarán todas las configuraciones de red, además de las políticas para analizar el tráfico y a su vez las vulnerabilidades que ocurren con el mismo, con el fin de ofrecer medidas para disminuir los ataques de denegación de servicio (DoS). Para la conectividad de los elementos de red se usa el protocolo OPENFLOW.

OBJETIVOS

OBJETIVO GENERAL

Analizar las vulnerabilidades insider contra ataques DoS en redes definidas por software

OBJETIVOS ESPECIFICOS

- Estudiar las vulnerabilidades insider en Redes Definidas por Software
- Desarrollar un escenario de prueba para la explotación de vulnerabilidades DoS en las redes SDN.
- Evaluar los resultados obtenidos antes y después de realizar los ataques de DoS en el escenario de prueba.
- Desarrollar una guía de buenas prácticas para análisis de vulnerabilidades insider en Redes Definidas por Software.

CAPÍTULO I

1 MARCO TEÓRICO

En el presente capítulo se analizan a las Redes Definidas por Software (SDN), su arquitectura, el protocolo de comunicación OpenFlow, controladores y simuladores, así como también las vulnerabilidades que afectan a la seguridad, para luego centrarse en los ataques de denegación de servicio y su respectiva clasificación. Y por últimos se detallan los mecanismos o estrategias defensivas ante estas amenazas(DoS) para: prevenir, detectar, identificar el origen y mitigar o contrarrestarlas.

1.1 Redes Definidas por Software

La ONF (The Open Networking Foundation) es la organización encargada de desarrollar y estandarizar a las Redes Definidas por Software (SDN), mediante la implementación de soluciones ágiles directamente programables, con una inteligencia centralizada, a través de una configuración mediante programación y con el uso de estándares abiertos lo que permite aprovechar todo el potencial de estas redes. (Open Networking Foundation, 2018)

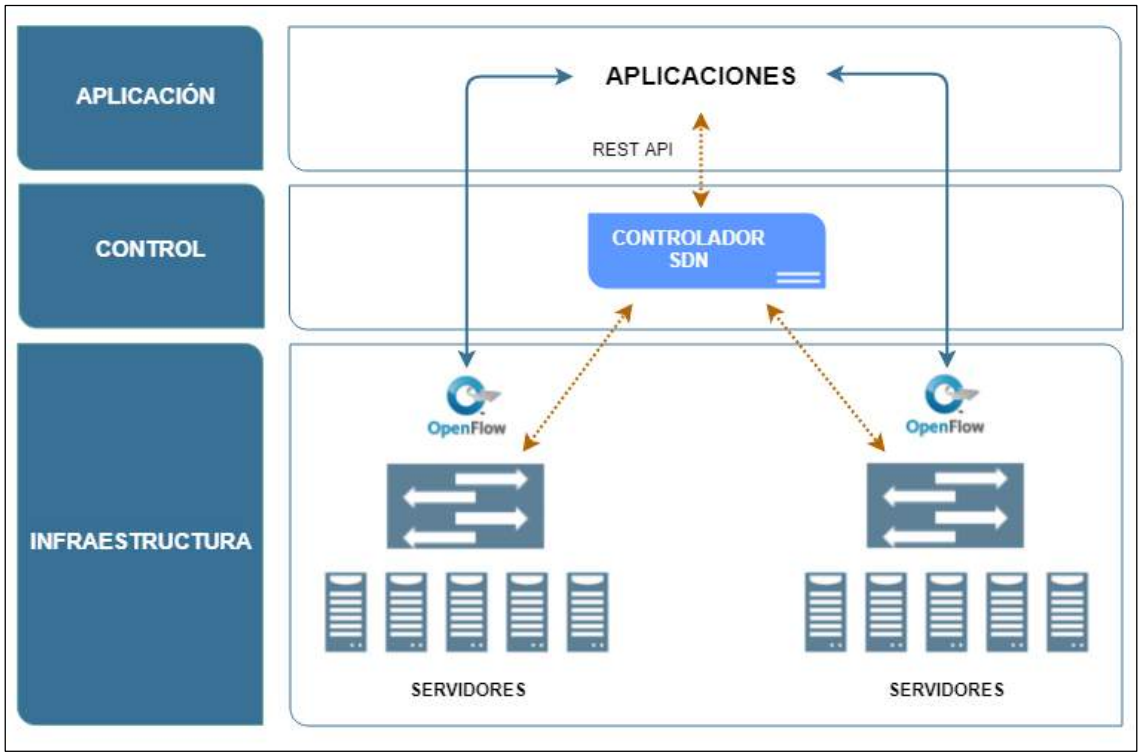


Figura 1-1: Redes Definidas por Software

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019

Las SDN es un nuevo paradigma de red, figura 1-1, que desacopla al plano de datos, especializado del reenvió de los paquetes desde el origen hacia su destino, del plano de control encargado de la gestión de la red. Al establecerse esta separación cualquier dispositivo será independiente del proveedor. En SDN, la inteligencia de todo el sistema radica en los controladores centralizados lo cuales están basados en software y toda la parte de dispositivos se encuentra en el plano de datos, para la comunicación de los mismo se usa el protocolo Openflow. (Azodolmolky, 2013)

Las SDN ofrecen las opciones de: simplificar el control de la red, al eliminar las operaciones de datos, administración e innovación gracias a sus ambientes programables, al ser su arquitectura de naturaleza dinámica, escalable y flexible, son idóneas para implementarse en aplicaciones que requieren de altos requerimientos como gran ancho de banda, seguridad, etc…, adaptándose con facilidad a cualquier necesidad, otra característica es que se reducen drásticamente los costos tanto de operación como de administración mediante la virtualización. (Citrix, 2014, p. 9)

1.1.1 Tipos de redes SDN

Al igual que en cualquier red de computadores, las redes SDN cuenta con dos tipos de redes, cómo son: las redes internas o insiders y redes externas u outsiders, mismas que interconectan a uno o varios sistemas autónomos diferentes mediante el protocolo TCP/IP. La diferencia de una red interna de una externa, radica en que las primeras son consideradas privadas, es decir, usadas al interior de una organización de modo que la información estará restringida para el público, por otra parte, las outsider permiten una comunicación con agentes externos que no forma parte de la institución. (Taqui y Cuadros, 2017, p. 5)

1.1.1.1 Redes Insiders

Son redes privadas donde solo personal y empleados de la organización pueden acceder a la misma puesto que cuentan con servidores locales que conectan a varios dispositivos entre sí, sin acceso externo, para garantizar alta seguridad en la comunicación, Son muy usadas en entornos corporativos, universidades o campus académicos, entidades financieras, etc., como repositorios de información para consultas y tramites. Las redes internas al igual que las demás, también son vulnerables a sufrir cualquier ataque, por tal razón es indispensable implementar mecanismos de defensa como, firewalls, encriptaciones, cortafuegos, entre otros. (Legerén, 2015)

Para construir una red interna se deben considerar los siguientes elementos, hardware y software. Para hardware o correspondiente al soporte físico se incluyen, switches, routers, servidores web, sistema de conexión, por ejemplo, fibra óptica, cable coaxial, etc., y finalmente de dispositivos

que actúen como mecanismos de seguridad. En lo lógico o software se requiere de protocolos de comunicación, sistemas operativos para clientes y servidores como, Linux, Fedora, Windows; aplicaciones y softwares para seguridad y desarrollo. (Redondo y et al, 2015, p. 11, p. 5)

Los beneficios que proporcionan las redes insiders son: escalabilidad y flexibilidad para interconectar múltiples plataformas, comunicaciones seguras y eficientes, administración centralizada de la red, economizar recursos, entre otras. Con todas estas prestaciones fácilmente las organizaciones pueden manipular la información al momento de su almacenamiento, recuperación y gestión. (Redondo y et al, 2015, p. 11, p. 3)

1.1.1.2 Redes outsider

Son redes que no poseen acceso limitado y seguridad, son útiles para comunicar a la red interna de una organización con otras ubicadas en cualquier lugar del mundo, pero para eso en primer lugar deberán estar conectados a los ISP o proveedores de servicio de internet. A diferencia de las internas son de acceso público, gracias a que los servicios están disponibles para todos, por lo tanto si se desea acceder al interior de cualquier empresa se necesitan de permisos de autenticación, es decir identificadores de usuarios y contraseñas.

1.1.2 Arquitectura

Las SDN separan el plano de control y plano de datos, como se muestra en la figura 2-1, creando una infraestructura centralizada, directamente programable por software, lo cual permite a los administradores de red configurar, gestionar, asegurar y optimizar los recursos de red, sin tener un acceso físico a los dispositivos de hardware. (Open Networking Foundation, 2018). En el plano de datos se encuentran las tablas para el tratamiento de paquetes, que trabajan en función de la dirección IP, MAC, entre otras características. Y todo lo concerniente a programación, algoritmos, configuración de tablas de reenvio reside en el plano de control.(Goransson y et al, 2014, p. 58)

Con todas estas prestaciones SDN tiene como fin reducir costos en equipamiento y brindar al usuario un manejo eficaz de los servicios y elementos de red. Además, permiten tener el control mediante el establecimiento de reglas de acuerdo con las necesidades de los administradores y usuarios de la red.

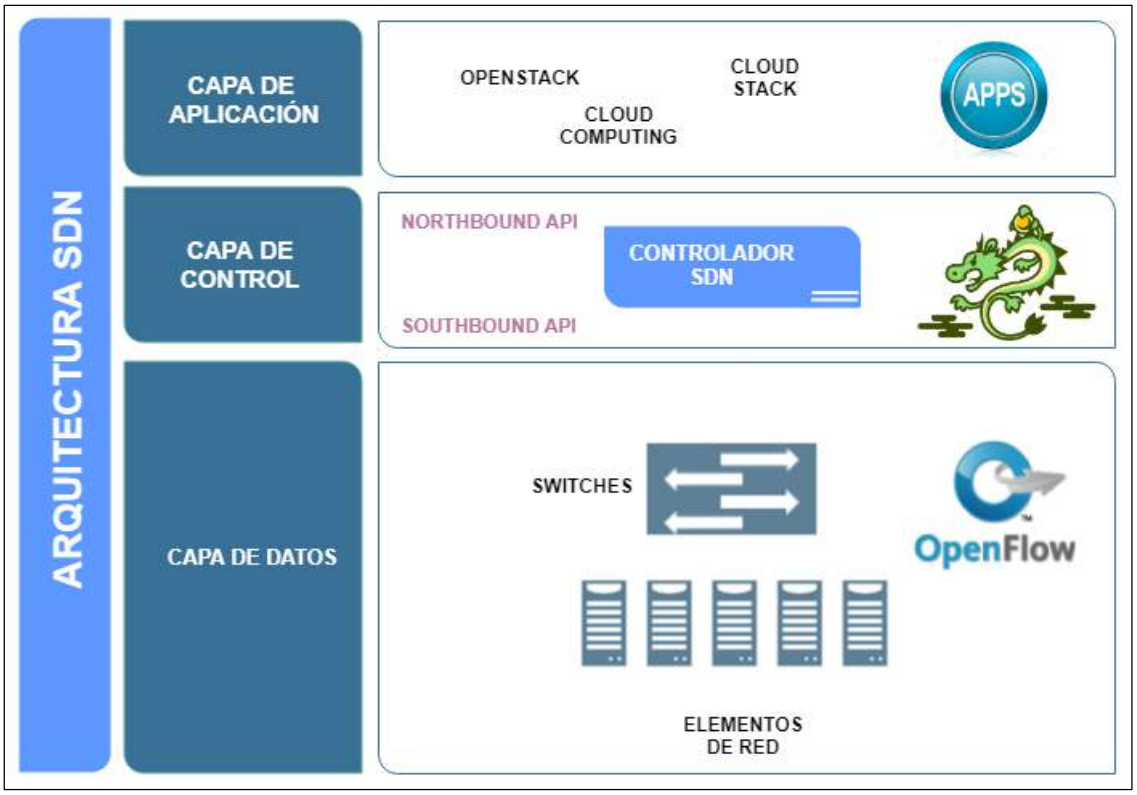


Figura 2-1: Arquitectura SDN

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019

1.1.2.1 Capa de control

El plano de control permite establecer un manejo centralizado de todos los elementos y a la vez controla dinámicamente los demás recursos de red de acuerdo a lo planteado por la capa de aplicación, es decir que el desarrollador podrá crear políticas de acuerdo a sus necesidades. La principal función de este plano es mantener la tabla de reenvió siempre actualizada, para que el plano de datos no tenga inconvenientes y pueda tratar independientemente a cualquier tipo de tráfico. También se encarga de procesar diferentes protocolos de control que puedan influir a la tabla de forwarding, dependiendo del tipo de configuración y conmutador. (Goransson y et al, 2014, p. 7)

Para la comunicación de esta capa con las demás se utilizan interfaces programables de aplicaciones conocidas como (APIs), facilitando la implementación de varios servicios como: balanceo de carga, seguridad, calidad de servicio (QoS), control de acceso, entre otros.

Por ejemplo, para la comunicación con el plano de aplicación utiliza la interfaz NorthBound o Norte y por medio de la interfaz SouthBound o Sur con el plano de datos. A continuación, se detallan cada una de ellas. (Oladunjoye, 2017, p. 20).

- Aplications Programming Interface Northbound

La NBI enlaza a la capa de control con la parte externa de la red o con las aplicaciones. Para tener una seguridad en el sistema el administrador deberá establecer previamente reglas o politicas de acceso o denegación sobre las peticiones que se realicen. Los controladores SDN actualmente cuentan con diferentes APIs. Por ejemplo, API REST, API especializadas ad-hoc, RPC, OSGI y los lenguajes de programación., siendo la más usada la interfaz REST (Representation State Transfer). (Ochoa, 2018, p. 26)

- Aplications Programming Interface Southbound

La interfaz SBI, está situada entre el plano de control y la de infraestructura. Por medio de esta, el controlador puede informar los requerimientos de las aplicaciones, instauración de políticas en cada uno de los dispositivos, mecanismos de envió de tráfico. Facilitando así el control eficiente sobre la red y permitiendo cambios dinámicos de acuerdo con las demandas y necesidades incluso en tiempo real. Garantizando que los clientes tengan mayores facilidades y flexibilidades al implementar redes SDN. Al igual que en la NBI, en la SBI existen APIs, como son: LIST, I2RS, BGP, OVSDB, Net Conf, SNMP, OpenFlex, ForCES, pero el más usado es el que está basado en el protocolo OpenFlow. (Vijay y Vasudevan, 2016, p. 13).

1.1.2.2 Capa de datos

La capa de datos o también conocida como capa de infraestructura, alberga a todos los dispositivos hardware y software de red, encargados del transporte y procesamiento del flujo de datos de acuerdo a la decisión impuesta por el plano de control. Para la comunicación de este plano con el controlador se lo hace a través de la interfaz CDPI, la misma que proporciona informes estadísticos, notificaciones de eventos, controles de las operaciones de reenvió, usando el protocolo Openflow para la comunicación. (Oladunjoye, 2017, p. 19)

Consta de varios puestos para receptar y trasmitir paquetes, así como también de tablas de forwarding, también es responsable del almacenamiento en memoria, programación, modificación de encabezados y reenvió de tramas. (Goransson y et al, 2014, p. 7). Es así que los paquetes ingresan y salen del plano de datos a través de puertos físicos y/o virtuales, una vez ejecutado esta acción, el controlador SDN analiza los datos y envía las reglas a los elementos de red. Este plano por sí solo no tiene la capacidad de tomar decisiones, para ello debe siempre consultar al controlador todas las tareas y este a vez puede ir almacenando las ordenes

preestablecidas para actuar automáticamente a futuro logrando así un procesamiento eficiente en caso de existir fallos en la red. (Moscoso, 2016)

1.1.2.3 Capa de aplicación

También conocido como plano de gestión, aquí, los administradores de la red configuran y supervisan al conmutador, e incluso pueden modificar los flujos en el plano de control como en el de datos. Usa la interfaz NBI para notificar los servicios que el usuario final requiere, ejemplo: QoS, Firewall, proporcionar seguridad, etc., al estar sobre la capa de control, es fundamental que tenga noción de toda la topología, para actuar rápidamente a cada petición, previo al establecimiento de reglas. Esta capa ayuda a simplificar las configuraciones, gestiona nuevos servicios, nuevos accesos y mejoras en la red. (Cosío, 2017, p. 15)

1.2 Protocolo OpenFlow

Openflow fue estandarizado por la ONF, a partir de la investigación realizada en el Programa Clean Slate de la Universidad de Stanford, cuyo principal objetivo es proveer de protocolos experimentales en campus universitarios, con un enfoque a futuro que permita remplazar las funciones de los protocolos de capa 2 y capa 3 de los equipos de enrutamiento y conmutación comerciales. (Big Switch Networks, 2013, p. 4)

Actualmente, este protocolo es el que rige a las Redes Definidas por Software (SDN), permitiendo la automatización de la red a través de un controlador centralizado, mediante una abstracción del plano de control del plano de datos. Openflow utiliza los protocolos TCP y SSL para la comunicación del controlador con el plano de control, permitiendo la configuración de la red de forma rápida, garantizando un manejo inteligente y flexible de las redes, permitiendo manipular, identificar y controlar en tiempo real todo tipo de tráfico que circula por la red, basándose en reglas predefinidas en función a los flujos. (Yaguës, 2015)

OpenFlow se divide en dos grandes grupos: el primero corresponde a los protocolo de conexión, que se encargan de establecer un control de sesión además de definir la estructura de los mensajes para la modificación de los flujos y recolectar estadísticas y el segundo son los protocolos de configuración y administración, que ayudan a asignar los puertos físicos de los dispositivos a un controlador SDN además define el comportamiento de cada uno de los elementos de red en caso de no haber comunicación con el controlador. (Olaya, 2015)

1.2.1 Versiones del protocolo OpenFlow

Se detallan las diferentes versiones del protocolo OpenFlow: v1.0, v1.1, v1.2 y v1.

1.2.1.1 OpenFlow 1.0

Lanzado el 31 de diciembre de 2009, esta versión solo tiene soporte de las capas: física, enlace, red y aplicación La instrucción más importante es la de reenvio del paquete por puertos, otra es la implementación de controles de acceso a la red y modificación de los campos de encabezado del paquete, pero debido a su limitada tabla de flujos realizar una operación a la vez. La tabla de flujos, figura 3-1, consta de tres componentes: encabezado, contadores e instrucciones. (Braun y Menth, 2014, p. 308).

Figura 3-1: Tabla de flujos de OpenFlow 1.0

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019

- ENCABEZADO. - Compuesto por doce campos: puerto de acceso, direcciones IP de origen y destino, VLAN ID, tipo de servicio, prioridad de VLAN, protocolos de red o transporte, etc

- CONTADORES es el encargado del conteo cuidadoso de cada paquete que recibe y envía, mediante actualizaciones permanentes de sus tablas, flujos, puertos.

- INSTRUCCIONES. - donde reside la información de las acciones de reenvio, cada vez que ingresa un paquete, también indica a través de cual puerto se ejecutarán las tareas. Las instrucciones más usadas se las muestra en la tabla1-1:

Tabla 1-1: Instrucciones de las tablas de flujos

Instrucción	Descripción
All	Reenvio a través de todos los puertos
Controller	Encapsulación y reenvio de paquetes al controlador
Drop	Eliminar paquete
Set, add, remove, modify	Establecer dirección: IP, MAC, ID de VLAN, puerto de destino y origen

Fuente: http://flowgrammable.org/sdn/openflow/actions/#tab_ofp_1_0

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

La comunicación entre el controlador y switch se la realiza por un canal seguro, el cual tiene una conexión TLS por medio del protocolo TCP, y si el conmutador relaciona una dirección IP con el controlador SDN, especificará la versión con la que está trabajando, tipo, longitud e identificación del mensaje.(Sandoval, 2018, p. 36)

1.2.1.2 OpenFlow 1.1

Lanzado el 28 de febrero de 2011. Introdujo dos principales mejoras como son: múltiples tablas de flujos, y tablas de grupo. La tabla de grupos permite representar en una sola entidad a varios puertos, también admite funcionalidades de multidifusión y multipath, balancear carga, enlaces agregados, etc. Una característica muy particular de esta versión es el soporte total de VLANs y MPLS.(Open Networking Foundation, 2011, p. 5)

Las tablas de flujo, figura 4-1, consta de varios campos: campos de coincidencia, que abarca a los puertos por el cual ingresa el paquete y además de la cabecera del mismo, contadores e instrucciones, por ejemplo, eliminar, escribir, borrar, agregar, actualizar.

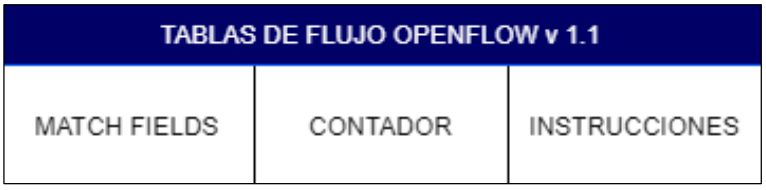


Figura 4-1: Tabla de flujos de OpenFlow 1.0

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019

Las instrucciones más importantes de esta versión son: clear, apply, write action, go to, write metadata y experimenter, en la tabla 2-1, se las puede apreciar con su respectiva descripción.

Tabla 2-1: Instrucciones OpenFlow 1.1

Instrucción	Descripción
Clear	Borrar acciones
Apply	Aplicar acciones
Write action	Escribir acción
Go to	Continuar el proceso en la tabla indicada
Write metadata	Actualizar los metadatos
Experimenter	Instrucciones personalizadas

Fuente: http://flowgrammable.org/sdn/openflow/actions/#tab_ofp_1_0

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

1.2.1.3 OpenFlow 1.2

Publicado el 5 de diciembre de 2011. Incluye múltiples mejores como son: el protocolo de configuración (OF-Config), soporte extendido para IPv6 y una estructura TLV llamada OXM o Extendible Match que determina nuevas entradas de coincidencia extendibles. A pesar de que sigue usando instrucciones de OpenFlow 1.0 y 1.1, en esta nueva versión existe la posibilidad de ingresar en el encabezado cualquier valor.

Además, con esta versión un conmutador puede ser administrado por uno o varios controladores conectados, al tener esta opción se crea un controlador master y esclavos, los esclavos brindan soporte seguro mediante copias de seguridad alojadas en ellos en caso de que existieran fallo en el controlador principal. (Sandoval, 2018, p. 46)

1.2.1.4 OpenFlow 1.3

Publicado el 25 de junio de 2012. Es considerada como una de las últimas versiones, con funcionalidades agregadas para: soportar MPLS, puertos lógicos, Q in Q o dotq1 tunneling, múltiples tablas, túneles, medidores de tráfico. En esta versión los mensajes READ_STATE son reemplazados por MULTIPART_REQUEST y MULTIPART_REPALY. MULTIPART_REQUEST incluye varias estadísticas: flujos, tablas, puertos, métricas, etc. (Natarajan, 2014)

Las tablas de flujo de openflow 1.3, figura 5-1, tienen más campos agregados como: campos de coincidencia, prioridad, contador, instrucciones, tiempo de descarte del flujo y las cookies. El campo PRIORIDAD, indica lo que debe hacer si un paquete ya ha ingresado al sistema, TIEMPO DE DESCARTE DE FLUJOS es el que establece el tiempo de eliminación de un flujo por parte del conmutador y COOKIES delega al controlador la clasificación más eficaz del paquete.(Contreras, 2014, p. 19-21)

TABLAS DE FLUJO OPENFLOW v 1.3					
MATCH FIELDS	PRIORIDAD	CONTADOR	INSTRUCCIONES	TIEMPO DE DESCARTE DE FLUJOS	COOKIES

Figura 5-1: Tabla de flujos de OpenFlow 1.3

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019

1.2.2 Comparativa de versiones OpenFlow

En el siguiente apartado se resume de forma breve las especificaciones de las versiones del protocolo Openflow v1.0, v1.1, v1.2 y v1.3.

En la tabla 3-1, se detalla la información de cada publicación de Openflow, como: el año de lanzamiento al mercado, si soporta la creación de tablas de flujos, grupos y métricas, también si existe la posibilidad de etiquetas de VLANs y MPLS, y si es compatible con el protocolo de internet versión 6 (IPv6).

Tabla 3-1: Versiones del protocolo OpenFlow

Características	OpenFlow v1.0	OpenFlow v1.1	OpenFlow v1.2	OpenFlow v1.3
Año publicado	31, Dic, 2009	28, Feb, 2011	5, Dic, 2011	25, Jun,2012
Tabla de flujos	Simple	Múltiple	Múltiple	Múltiple
Tabla de grupos	No	Si	Si	Si
Tabla de métricas	No	No	No	Si
Etiquetas VLAN y MPLS	No	Si	Si	Si
Soporte IPv6	No	No	Si	Si
Controlador múltiple	No	No	Si	Si

Fuente: http://article.sciencepublishinggroup.com/pdf/10.11648.j.ajsea.20140306.12.pdf

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

1.2.3 Switch OpenFlow

Es un software o hardware que proporciona conectividad a los elementos de la red con el controlador, al usar el protocolo OpenFlow gestiona y controla tablas de flujos, por lo tanto, la data path contiene una serie de campos de las cabeceras de los paquetes y una acción como: reenvió, modificación o eliminación de campo.

Al recibir paquetes el switch openflow va almacenando la información para reducir tiempos de consulta y en caso de que no exista coincidencia con un paquete recibido en la entrada de flujo inmediatamente la instrucción será enviada al controlador para que este decida qué acción tomar, pudiendo ser aceptado o eliminado de las tablas de flujo. En la actualidad existen algunos switches basados en software por ejemplo Open vSwitch, OpenWrt, etc.

El procesamiento de paquetes en un switch openflow, se lo realiza de la siguiente manera: cuando ingresa un paquete a la red hace un proceso opcional del Protocolo Spanning Tree (STP) 802.1d

que permite verificar que no haya lasos entre los diferentes dispositivos de la red, luego pasa a la tabla de flujos para comparar coincidencias con la entrada cero y en caso de encontrarla aplicará las acciones dadas por el controlador y si no existe hasta la N entrada el paquete se enviará al controlador para que este tome la decisión de reenvio o eliminación del paquete, figura 6-1.

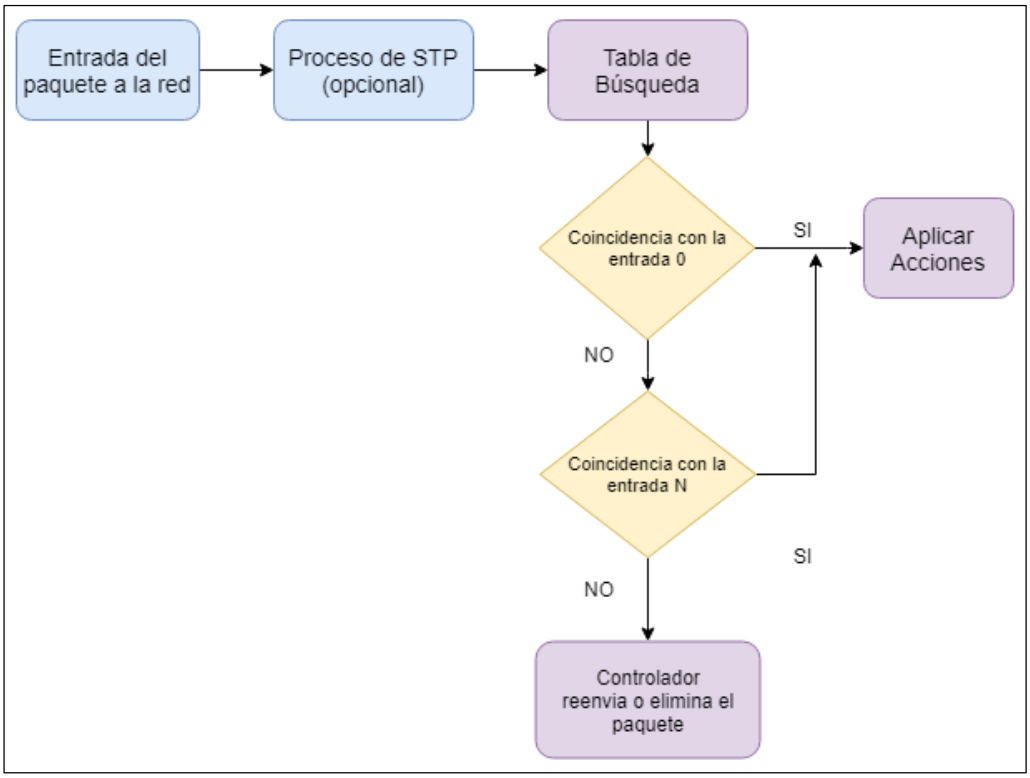


Figura 6-1: Operación del Switch OpenFlow

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019

Según investigaciones los switches openflow más comunes son: simples e híbridos. Cada uno con características específicas. SWITHES SIMPLES no poseen la funcionalidad de toma de decisiones, debido a que solo soportan el protocolo de comunicación Openflow y los SWITCHES HÍBRIDOS, son capaces de realizar procesos como: VLANs, enrutamiento con IPv4 e IPv6, direccionamiento, calidad de servicio. (Nuñez, 2015, p. 26)

1.3 Controlador SDN.

Un controlador es una entidad centralizada que maneja interfaces físicas y virtuales para gestionar, administrar y controlar tareas de acuerdo a instrucciones previamente establecidas de acuerdo al servicio a ofrecer. Además, permite una visualización global de la red y un soporte simplificado al administrador de red. A partir de estos se define como un modelo de datos de alto nivel que tienen relación de los servicios gestionados y las políticas que presta dicho dispositivo. (Pazmiño, 2018, p. 10)

En la actualidad existe una amplia gama de controladores tanto a nivel comerciales como de código abierto. Entre los comerciales se puede mencionar las fabricas: Big Switch Networks, Cisco, Juniper, HP, Brocade, Mikrotik, entre otros; y de código abierto a: Floodligh, POX, Beacom, Opendaylight, Ryu, en la tabla 4-1, se aprecia el dispositivo, la marca fabricante y el tipo (comercial o de acceso libre).

Tabla 4-1: Controladores comerciales y de código abierto.

Controladores	Fabricante	Tipo
APIC	CISCO	Controlador comercial
XNC	CISCO	Controlador comercial
OnePK	CISCO	Controlador comercial
NorthStart Controller	Juniper	Controlador comercial
Contrail	Juniper	Controlador comercial
VAN SDN Controller	HP	Controlador comercial
Vyatta Controller	Brocade	Controlador comercial
VMWare	NSX	Controlador comercial
Ryu	NT & TC	Controlador de código abierto
Opendayligh	Linux	Controlador de código abierto
Floodlight	Big Switch Networks	Controlador de código abierto
POX	Nicira	Controlador de código abierto
NOX	Nicira	Controlador de código abierto
Beacom	Standford University	Controlador de código abierto

Fuente: http://www.dit.upm.es/~posgrado

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

1.3.1 Plataforma Opendayligh

Opendaylight, figura 7-1, es una plataforma, desarrollada por la organización Linux, basándose en el lenguaje Java y licencia Apache, que ha tenido un largo historial de apoyo y desarrollo a las organizaciones open-source o licencia libre, principalmente al soporte de redes de comunicaciones, facilitando la personalización y automatización de redes de cualquier tamaño y escala gracias a la capacidad de programabilidad.

Además, el controlador ODL desea ir más allá de la implementación de las SDN con OpenFlow y participa con varias empresas a nivel mundial como: Cisco, HP, Microsoft, Brocade, VMware etc. (Salinas, 2017). La arquitectura de ODL es similar a la de SDN, pero con la diferencia de que aquí se integra una nueva capa de abstracción de servicios entre clientes y proveedores, permitiendo a los usuarios desarrollar aplicaciones compatibles con una gran variedad de

hardwares, también contiene complementos dinámicos internos que agregan servicios y varias funcionalidades de red. Para acceder al controlador ODL, se usan las REST INTERFACES. (Ribes, 2015)

Figura 7-1: Arquitectura OpenDayLigh

Fuente: https://www.opendaylight.org/

Dentro de las versiones de Opendaylight se encuentran: Daylight, Hydrogen, Hellium, Lithium, Berylium, Boron, Carbon, Nitrogen, Oxigen y Fluorine, cada uno de ellos a medida que fueron publicadas incluían mejoras. En la tabla 5-1, se observan las versiones y sus fechas de lanzamiento. (OpenDaylight Project, 2018b)

Tabla 5-1: Versiones de Opendaylight

Versión de Opendaylight	Año de publicación
Daylight	2013, Abril
Hydrogen	2014, Febrero
Hellium	2014, Septiembre
Lithium	2015, Junio
Berylium	2016, Febrero
Boron	2016, Septiembre
Carbon	2017, Mayo
Nitrogen	2017, Septiembre
Oxigen	2018, Diciembre
Fluorine	2018, Agosto

Fuente: https://www.opendaylight.org/technical-community/getting-started-for-developers/roadmap

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019

1.3.2 Plataforma Ryu

Ryu, figura 8-1, es un controlador open source, desarrollado con el lenguaje de programación Python por el grupo japonés Nippon Telegraph y Telephone Corporation Labs (NTT Lab's) y al igual que ODL al estar basado en un conjunto de componentes predefinidos permiten modificar,

desarrollar, y crear aplicaciones personalizadas. Es compatible con todas las versiones de Openflow y OpenStack y a su vez proporciona servicio de topología y estadísticas la red.(Cardoso, 2015, p. 28)

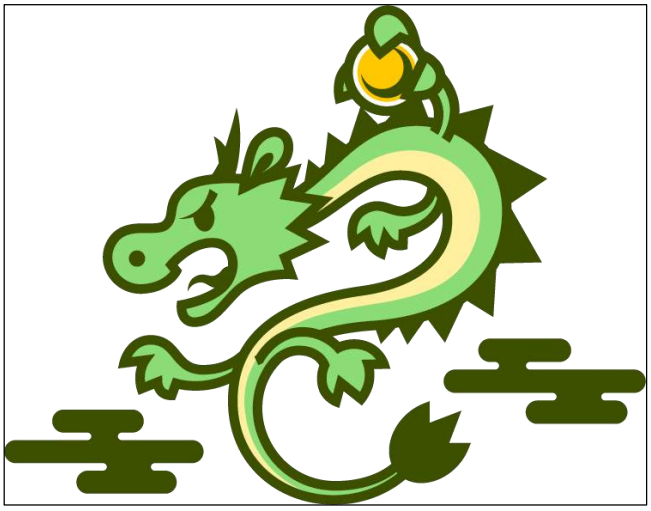


Figura 8-1: Plataforma Ryu

Fuente: https://osrg.github.io/ryu/

1.3.3 Plataforma Pox

La plataforma POX, figura 9-1, fue desarrollado a partir del controlador NOX, usando el lenguaje de programación Pyhon. POX es interoperable con varios sistemas operativos: Windows, Mac OS y Linux. Este controlador es uno de los frameworks con mayor crecimiento en investigación, además, es el usado por defecto en plataformas de simulación de redes como Mininet. Al igual que Opendaylight y Ryu utiliza el protocolo OpenFlow y tambien OVSB.(Yaguës, 2015)

Figura 9-1: Plataforma POX

Fuente: http://networkstatic.net/pox-openflow-controller-installation-screencast/

1.3.4 Plataforma Floodlight

Es otro tipo de controlador para redes definidas por software, pero de clase empresarial, lanzado al mercado en junio de 2013. Está basado en Java y licencia Apache, fue desarrollado por la comunidad Project Floodlight, líderes en redes de código abierto, la cual crea proyectos con soluciones compatibles con desarrolladores mundiales incluido de la Big Switch Networks.

Figura 10-1: Plataforma Floodlight

Fuente: http://www.projectfloodlight.org/floodlight/

Floodlight, figura 10-1, es administrado por la ONF y está diseñado para realizar una variedad de funcionalidades comunes que controlen e investiguen una red Openflow, además orientado para trabajar con el creciente número de conmutadores virtuales y físicos, además de enrutadores y puntos de acceso. (Project Floodlight, 2019)

Este controlador posee una colección de aplicaciones denominadas API REST, para resolver las diferentes necesidades de los usuarios de red. Una de las tantas ventajas de Floodlight es el manejo de redes mixtas, es decir que pueden o no estar basadas en el protocolo Openflow. (Project Floodlight, 2019)

1.3.5 Softwares de simulación y emulación SDN

En la actualidad debido a que las redes y servicios requieren mayores exigencias es necesario el uso de la virtualización, para ahorrar recursos hardware. La ventaja de la virtualización frente a las redes físicas es que permite crear e interactuar diferentes topologías de red en tiempo real. A continuación, se detallan algunos softwares en los cuales se pueden implementar redes SDN.

1.3.6 Mininet

Mininet es un software creado por docentes de la Universidad de Standford, utilizando un núcleo Linux con una API basada en Python, proporcionando la capacidad de probar, simular, desarrollar y crear entornos para redes definidas por software (SDN). Mininet ofrece la ventaja de realizar prototipos con componente virtuales: hosts, conmutadores, controladores, etc., y pruebas de topologías sin la necesidad de ser implementadas en redes físicas, además ofrece opciones multitarea. (Acurcio, 2008, p. 2-5)

Mininet posibilita la opción de simular y configurar remotamente controladores: Opendayligh, Ryu, POX, Floodlight, entre otros. Este software por defecto incluye conmutadores open vSwitch. Además, proporciona una herramienta con interfaz gráfica llamada MiniEdit, la misma que fue creada sobre el lenguaje de programación Python, figura 11-1, que ayuda a crear topologías de manera más fácil.

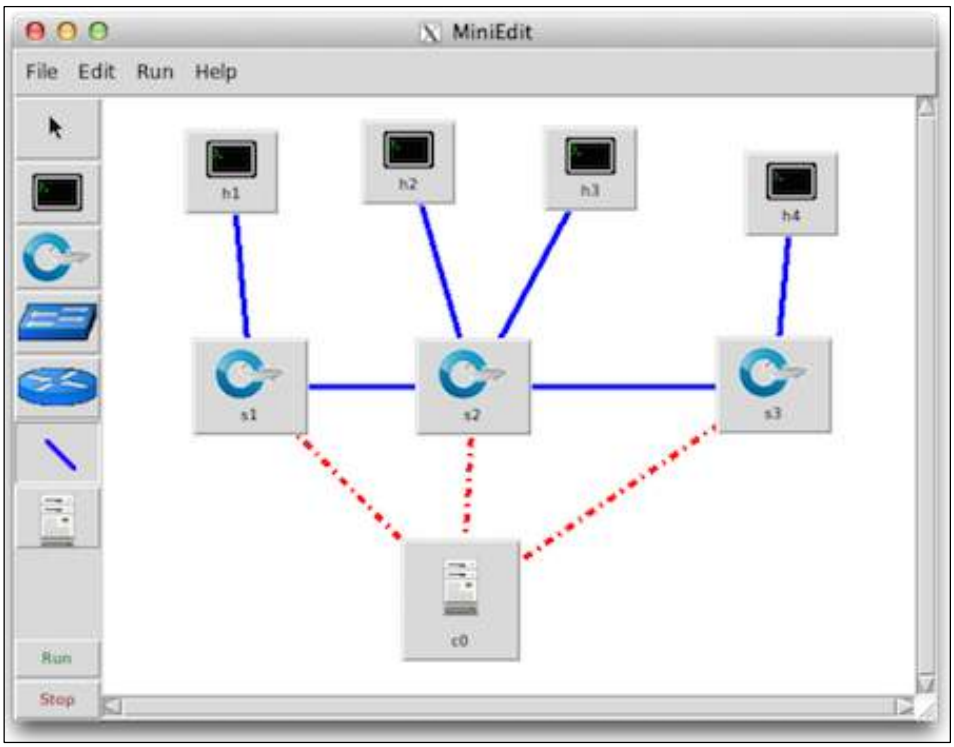


Figura 11-1: Entorno MiniEdit

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

Esta herramienta posee instrumentos para añadir, eliminar, modificar, guardar el proyecto, etc. Por defecto en MiniEdit, vienen incorporados algunos escenarios, pero en caso de que se desee crear uno nuevo, se los realizará mediante el uso de scripts de Python.

1.3.7 EstiNet

Su antecesor fue NCTUns, pero a partir de 2011 se cambió de nombre a EstiNet, este producto pertenece a la empresa EstiNet Technologies Inc, la cual se centra en desarrollar aplicaciones y soluciones para SDN basándose en el protocolo Openflow. En EstiNet, figura 12-1, se pueden simular entornos de red incluyendo las capas: física, enlace de datos, red, transporte y aplicaciones, con sus protocolos respetivos. A su vez también incluye una interfaz gráfica o GUI, para construir redes usando una pantalla visual para observar resultados de la simulación y depuración. (EstiNet, 2019)

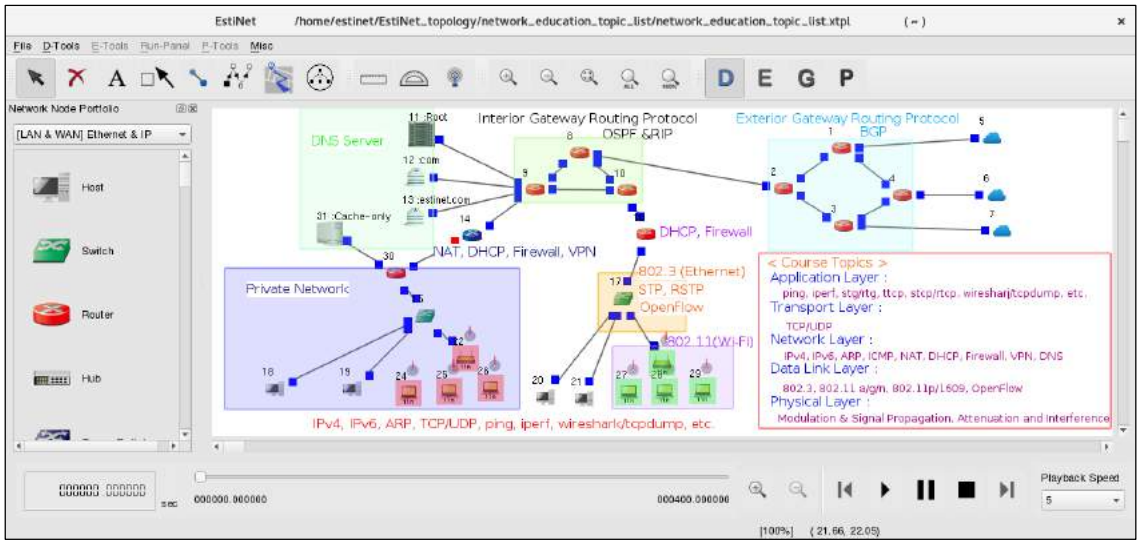


Figura 12-1: Entorno EstiNet

Fuente: http://www.estinet.com/ns/?page_id=21140

EstiNet tiene la funcionalidad de simulador y emulador. Las características de este software son: la ejecución fácil en los hosts de los controladores: NOX, POX, Floodligh, OpenDylight y Ryu.; otra ventaja es el uso de programas auténticos para aplicaciones y controladores, de igual forma la utilización de los protocolos TCP/ IP, con el fin de que el rendimiento de las aplicaciones implementadas sea óptimo.

1.3.8 GNS3

Graphic Network Simulation (GNS3), figura 13-1, es de código libre, lanzado al mercado en 2008, usando la licencia GPLv3. Fue creado por Jeremy Grossmman, usando el programa Python y librerías Dynagen para brindar una apariencia GIU, es decir gráfica, en la cual se realizan tareas como emular y configurar redes, tanto virtuales como reales.

Para crear topologías de red todos los dispositivos (hosts, conmutadores, enrutadores) deben estar alojados en el servidor local o máquina virtual. GNS3 es compatible con múltiples plataformas como: Windows y Linux y equipos de Dynamips, VirtualBox, Wireshark, Dynagen Quemu, entre otros. Los requerimientos mínimos para instalar este software son: memoria RAM de 4 Mb, espacio en el disco de 1 Gb y requiere de un procesador mínimo de 2 núcleos. (GNS3, 2018)

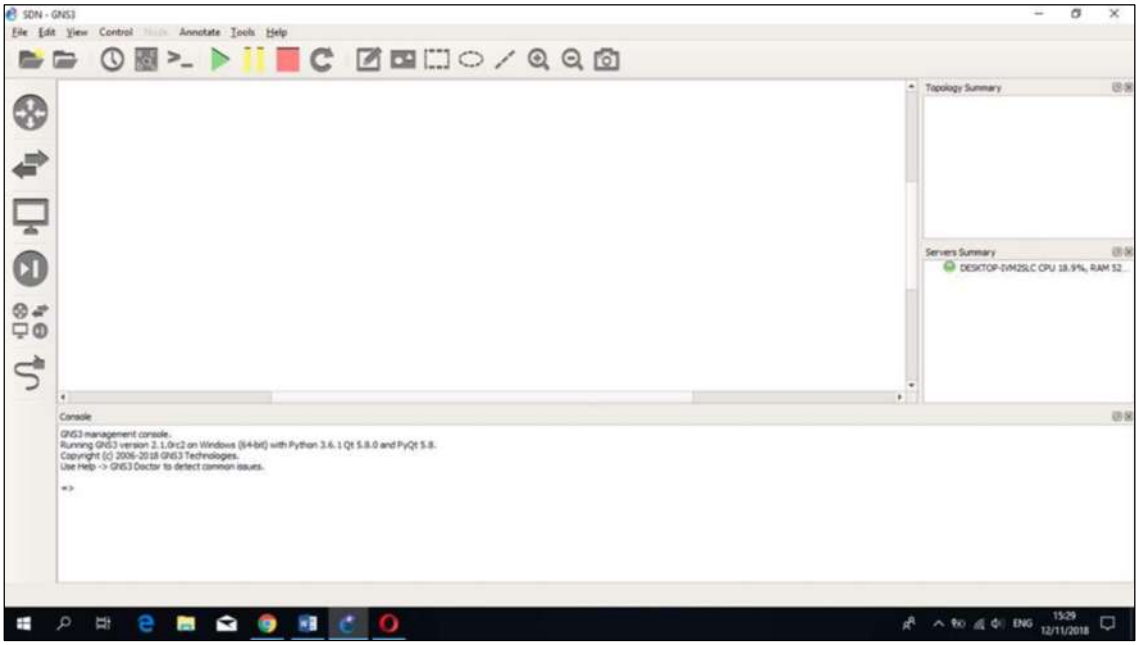


Figura 13-1: Entorno GNS3

Fuente: TOAINGA Daniela, PEÑA Daniel, 2019

GNS3 en sus inicios solo usaba dispositivos Cisco, pero en la actualidad permite interoperabilidad entre equipos de marcar fabricantes: Brocade, HPE, Microtik, Linux, etc, incluso permite el uso de tecnologías de red como: SDN, Linux y NFV. Este software trabaja con más de 20 proveedores diferentes para así crear topologías y laboratorios personalizados. En la tabla 6-1, se encuentra la descripción de cada uno de los simuladores y descripción. (Balsa, 2016, p. 10-11,31).

Tabla 6-1: Colaboradores de GNS3

Simuladores	Descripción
Dynamips	Emulador de IOS cisco
Dynagen	Front-end consola para dynamips
Qemu	Emulador de maquina genérico
Mininet	Simula solo redes SDN
OVS	Conmutador virtual compatible con OpenFlow
Wireshark	Software que captura tráfico de red

Fuente: http://www.adminso.es

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

1.4 Seguridad de las Redes Definidas por Software

Las ventajas de las Redes Definidas por Software es que proveen mayor flexibilidad de la red, aprovisionamiento, rapidez en la prestación de servicios y menores gastos en Capex y Opex, es decir, en gastos administrativos y operativos. Pero al ser centralizadas, el controlador es el punto clave para los atacantes, de manera, que, si se logra un acceder exitosamente, está expuesto a riesgos en la seguridad y disponibilidad de los servicios de manera temporal o definitiva..(Wang, 2016, p. 1-3)

1.4.1 Vulnerabilidades internas

Muchas veces, se tiende a pensar que la mayoría del ataque hacia la seguridad de las redes provienen desde el exterior de ellas, es decir que son realizadas por agentes externos que no tienen nada que ver con la empresa.

Pero por otro lado, se encuentran los victimarios camuflados que fingen ser empleados, antiguos trabajadores, gente de confianza, etc., para acceder fácilmente a la información confidencial sin causar sospechas, a estos agentes se los conoce como amenaza insider y son las que más daño ocasionan a una red, por tener privilegios de administración y acceso a equipos y softwares son más difíciles de detectarlos, por lo que es importante tomar cartas en el asunto, que refuercen la seguridad interna para minimizar riesgos.

Según un informe realizado por el Verizon Data Breach Investigations Report (DBIR), publicado en 2018, revela que el 58% de casos se originaron internamente dentro de las organizaciones, y tan solo el 42 % surgen desde el exterior. Las amenazas internas ocurren de distintas formas, por ejemplo, al otorgar permisos de acceso a cualquier persona, falta de seguridad al almacenar información personal, no tener un sistema de control de actividades de usuarios internos, no cancelación de privilegios para personas que dejen de laborar en la empresa o que hayan sido despedidos, errores de configuración de equipos y servicios.(NetlQ, 2016, p. 6, p. 2-3)

Como toda red tradicional, las redes SDN también están expuestas a varias vulnerabilidades, por ejemplo: en su programabilidad y uso de softwares de licencia libre, gracias a esas fallas cualquier agente puede fácilmente manipular la red o también falsear aplicaciones de seguridad para controlar el tráfico, otra preocupación es el uso excesivo de las interfaces de programación.

Por lo tanto, a partir de las debilidades o fallas en el sistema, un atacante tranquilamente puede realizar una amenaza como, ataques DoS, virus, troyanos, etc que atenten contra la seguridad.

Entre la mayor amenaza que preocupa a las SDN, se encuentran los ataques de denegación de servicio (DoS) que colapsan el sistema, agotando el ancho de banda y los recursos de red.(Wang, 2016, p. 1-3)

Como un antecedente de vulnerabilidad insider, fue el que sucedió el 24 de diciembre de 2012 a las 12:24 pm con una duración de 23 horas y 41 minutos, a la plataforma de streaming Netflix, que, por cuestiones de mantenimiento erróneo, se eliminaron parte de los datos del balanceador de carga elástica o ELB en Amazon Web Service, provocando la interrupción del servicio en Estados Unidos, Canadá y América Latina. Que después de la eliminación de datos es los dispositivos, el plano de control del dispositivo ELB comenzó a experimentar altos índices de latencia, además de errores en las APIs a través de las cuales se administran a los mismos.(NetlQ, 2016, p. 6, p. 2)

Otro ejemplo de ataque, fue el ocurrido el 21 de octubre de 2016, se trató de un ataque DDoS o distribuido de denegación de servicio, interrumpiendo a los sitios web de empresas estadounidenses: Twitter, Box, Netflix, PayPal, Pinterest, Amazon, entre otras. El ataque fue dirigido al servidor DNS de Dyn con sede en New Hampshire, valiéndose de fallas en las páginas web de las organizaciones, para ello los atacantes usaron varios bost o maquinas infectadas para generar solicitudes masivas de correos no deseados, provocando un aumento de SERVFAILs en el servidor, con el fin de colapsar e interrumpir servicios a los consumidores durante varias horas. En la figura 14-1, se muestra el tráfico generado durante el ataque.(Craig Sprosts, 2016)

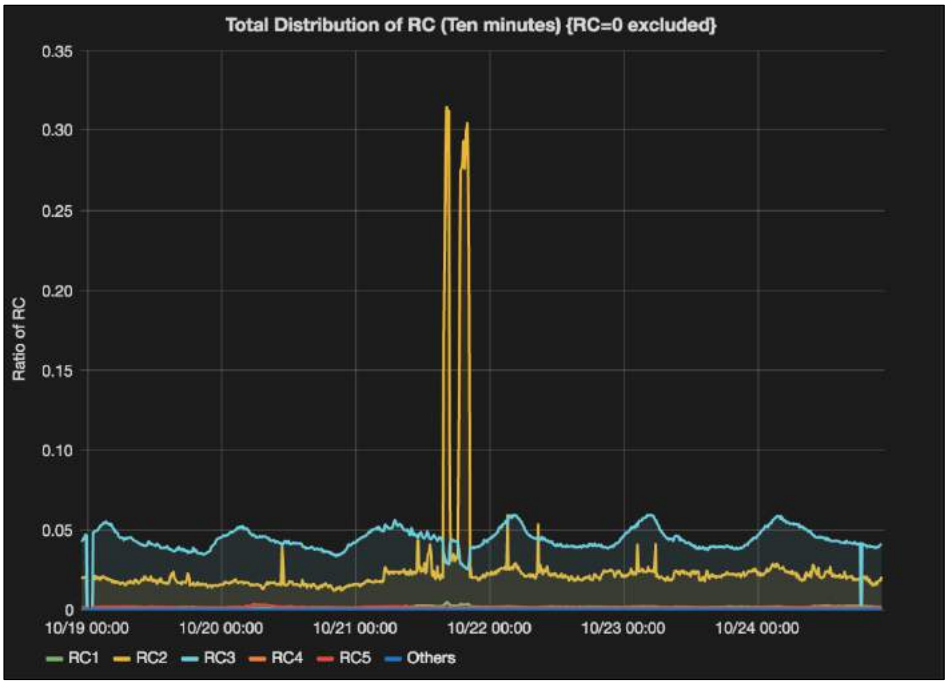


Figura 14-1: Tráfico generado por el ataque DoS a Dyn

Fuente: https://blogs.akamai.com/2016/10/what-csps-can-learn-from-the-latest-ddos-attacks.html

1.4.2 Metodologías de análisis de vulnerabilidades.

Actualmente para realizar análisis de riesgos de la seguridad de redes o sistemas, existen algunas metodologías, tales como: Open Source Security Testing Methodology Manual (OSSTMM), Operationally Critical Threat, Asset, and Vulnerability Evaluation (OCTAVE), y Computer Security Resource Center (CSRC) las mismas que utilizan hacking ético para realizar sus respectivas pruebas.

1.4.2.1 Open Source Security Testing Methodology Manual (OSSTMM).

El Manual de la Metodología Abierta de Comprobación de Seguridad (OSSTMM), es un proyecto desarrollado por el Instituto de Seguridad y Metodologías Abiertas (ISECOM) en el año 2001, siendo catalogado en la actualidad como uno de los más usados por ser un manual que contiene, estándares profesionales completos. Ésta metodología ha sido adoptada para probar la seguridad de las organizaciones de forma externa, dicho de otra manera, en sus: instalaciones físicas, interacciones humanas, y todas las formas de comunicación. (Herzog, 2003)

OSSTMM, realiza testeos de seguridad desde un entorno no privilegiado hacia un entorno privilegiado, para evitar todo tipo de seguridad, técnicas o alarmas. El objetivo de esta metodología es crear un método aceptado para ejecutar un test de seguridad minucioso y cabal. OSSTMM abarca seis secciones y cuatro fases: (Herzog, 2003)

- Seguridad de la información

Consisten en recolectar información y documentos pertinentes, también en revisión de la privacidad desde el punto legal y ético, para los análisis.

- Seguridad de los procesos

Realiza pruebas de test de solicitudes al recurso humano confiable de la empresa para obtener accesos privilegiados fraudulentos que afecten a la organización.

- Seguridad en las tecnologías de internet

Testean a los protocolos y conexiones usados en la comunicación, sondea la red de forma general para identificar los tipos de servicios que existen, así como también la cantidad de puertos

abiertos, comprueban el rendimiento y susceptibilidad de los sistemas de intrusos y lo más importante los test de ataques de denegación de servicio.

- Seguridad en las comunicaciones

Analizan los mecanismos de comunicación que usa el personal de la empresa como: fax, correo de voz sobre IP, escanea módems, etc.

- Seguridad inalámbrica

Verifica todas las tecnologías de comunicación inalámbrica: wifi, bluetooth, RFID, infrarrojos, radiación electromagnética, así como los dispositivos.

- Seguridad física

Se implementa con la finalidad de evaluar la seguridad, descubrir puntos monitoreados, controles de acceso, alarmas, ubicación, entorno del área física de la organización y de sus activos informáticos.

Además, esta metodología cuenta con cuatro fases para evaluar la seguridad de la red desde diferentes perspectivas: físico, humano, redes, telecomunicaciones, comunicaciones etc.

- Fase de inducción.

Para empezar con la inspección de la red, el analista debe saber los requerimientos, alcance y limitaciones de la auditoria.

- Fase de interacción

Es necesario conocer lo que se transmite a cada activo de la red, también la transparencia de las auditorias, accesos, etc.

- Fase de indagación.

En la fase de investigación, se verifican configuraciones y modo de funcionamiento de los activos o dispositivos de la red y también de servicios.

- Fase de intervención

No se puede llevar a cabo si no se cumplen las anteriores fases. La intervención consiste en verificar cuarentenas, proporcionar privilegios a la auditoria, asegurar la continuidad de los servicios, etc.

1.4.2.2 Operationally Critical Threat, Asset, and Vulnerability Evaluation (OCTAVE).

La Evaluación de Amenazas, Activos y Vulnerabilidades Operacionalmente Críticas (OCTAVE), es una metodología desarrollada por la Universidad de Carnegie Mellon, en el año 2001, contiene una serie de herramientas, técnicas y métodos para análisis de redes insider y outsider en busca de riesgos de seguridad, protegiendo a los principios de: confidencialidad, integridad y disponibilidad y rigiéndose al estándar de seguridad ISO 270001. (Estévez, 2014, p. 16)

OCTAVE, fácilmente puede ser construido por el personal interno de la empresa, el cual posea habilidades técnicas y capacidad organizacional. Para la implementación de esta metodología es importante que todos los miembros de la organización, estén de acuerdo en participar y en adquirir nuevos conocimientos para tomar acciones que ayuden a disminuir las afectaciones ocasionadas a los activos de la información. Este método, se implementa en tres fases: (Estévez, 2014, p. 17)

- Construcción de perfiles de amenazas basados en activos.

Consiste en poder identificar a los todos los activos de la organización, como por ejemplo los equipos finales, enrutadores, conmutadores, servidores, y demás dispositivos que conformen una red de datos, todo esto, para poder construir perfiles con las características de software y hardware de cada activo. Asimismo, los problemas detectados por los mecanismos de control y seguridad que usa la organización para protegerse de amenazas.

- Identificación de vulnerabilidades en la información.

Consiste en tener una visión de toda la red, para ello el personal del departamento de tecnologías de la información debe analizar toda la infraestructura, a fin de conocer los activos más críticos y quienes tienen acceso. Esta etapa también ayuda a identificar al personal encargado de la configuración y mantenimiento de la misma. Para identificar vulnerabilidades se recomienda el uso de softwares que escaneen a toda la infraestructura de red.

- Desarrollo de estrategias y planes de seguridad.

Después de haber recolectado información en fases anteriores, esta etapa final es importante para identificar y analizar los riesgos presentes en la empresa, para desarrollar planes y estrategias de seguridad, de acuerdo al nivel de impacto: bajo, mediano y alto de la amenaza, y así poder mitigar o contrarrestar las fallas con el propósito de evitar daños de red a futuro.

1.5 Ataque de denegación de servicio (DoS)

Con el avance de las tecnologías de la investigación y comunicación, la seguridad es de suma importancia, para proteger todos los bienes o recursos de una organización, garantizando la privacidad de la información sin interrumpir los servicios.

Pero a medida que pasa el tiempo aparecen nuevas modalidades delictivas cada vez más sofisticadas e inteligentes llamados ataques, que se aprovechan de la mínima vulnerabilidad o falla en el sistema para causar daños, como: eliminación de información, violación a las propiedades de la seguridad, etc. Entre los ataques de mayor amenaza están la suplantación de identidad, sabotaje, robo de información, denegación de servicio, inserción de códigos maliciosos, entre otros.(Tarazona, 2015, p. 10, p. 138)

Entre los ataques de más fácil ejecución y los más difíciles de mitigar, se encuentran los ataques DoS (Denial of Service), los cuales tienen la finalidad de agotar los recursos del sistema informático (ancho de banda o de procesamiento), logrando así la interrupción temporal o definitiva de los servicios tales, como: bases de datos o páginas web, mediante la generación de varias solicitudes en un instante de tiempo al equipo de destino, comprometiendo a la propiedad de la seguridad denominada disponibilidad. Los protocolos que tienen más prioridad de ser atacados son: SMTP, DNS y NTP.

Durante los años 2013 y 2014 según un informe de la ENISA (Agencia Europea de Seguridad de las Redes de la Información), se detallan que los ataques DoS se incrementaron en un 70%, frente a otras amenazas.

El ejemplo más claro de DoS, fue un ataque ocurrido el 28 de febrero de 2018 a la plataforma GitHub, paralizando los servicios ofertados desde las 5:20 hasta las 5:30 pm. El ataque a GitHub se realizó a través del puerto UDP 11211, dañando a más de 60 000 sistemas autónomos diferentes, al ser un ataque amplificado tuvo un alcance de 1,35 Tbps afectando a 1026,9 paquetes

por segundo, pero afortunadamente pudo ser mitigado por Arbor Networks. En la figura 15-1, se evidencia un monitoreo del impacto que causó el ataque DoS sobre GitHub. (Kottler, 2018)

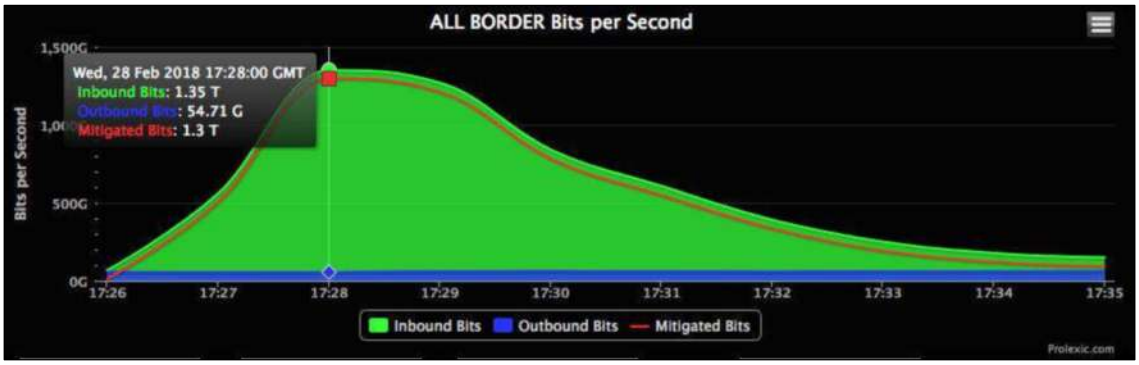


Figura 15-1: Impacto de ataque de denegación a GitHub

Fuente: https://githubengineering.com/ddos-incident-report/

1.5.1 Clasificación de los Ataques de Denegación de Servicio

Según, El Equipo de Respuestas ante Emergencias Informáticas o CERT (Computer Emergency Response Team), creado en 1998 por el Instituto de Ingeniería de Software, por problemas suscitados con una amenaza llamada gusano de Morris. Es un centro que cuenta con expertos en redes para crear soluciones o medidas preventivas y reactivas ante incidentes de seguridad en sistemas informáticos, es decir frente a cualquier ataque en la red. Según este organismo, existen varias formas de clasificar a los ataques de denegación de servicio, las mismas que se detallarán a continuación, en la figura 16-1:

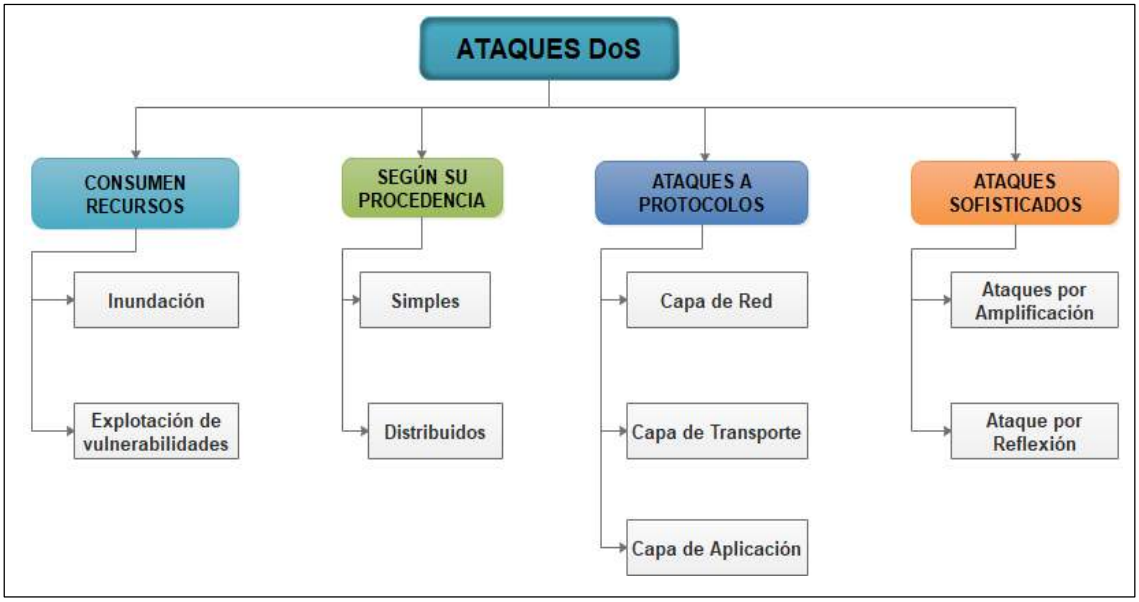


Figura 16-1: Clasificación de los ataques DoS

Realizado por: TOAINGA Daniela, PEÑA Daniel.

1.5.1.1 Ataques destinados al consumo de recursos

- Ataques de denegación de servicio por inundación

Los ataques de denegación por inundación o también conocidos como de fuerza bruta, su principal función, es saturar a los servicios tales como: ancho de banda, memoria, recursos de la CPU, espacio en el disco, etc.., alojado en algún dispositivo, con la generación de varios mensajes. (Chávez, 2011, p. 50).

Las formas de realizar este atacar son varias, por ejemplo, es atacar a los puertos de red del victimario, con la generación de flujos que viajen a velocidades mayores que las que admite la tarjeta de red. Otra es inundar la red local que conecta a Internet con el afectado, sobrecargando de igual forma a todos los nodos del segmento de red, con este caso no solamente se afecta a uno solo sino a todos los que estén usando el servicio. (Maciá, 2007, p. 15).

A continuación, se detalla un ejemplo: cuando se aplica un ataque de denegación por inundación basándose en el protocolo (UDP), la persona que realiza la acción, envía numerosos segmentos UDP de forma aleatoria a diferentes puertos de un equipo, saturando el ancho de banda para que otros hosts no consigan acceder al servicio. (Abliz, 2011, p. 50, p. 3).

- Ataques basados en la explotación de vulnerabilidades

Conocidos como ataques semánticos, se aprovechan de la más mínima vulnerabilidad en una política o errores en el software, para enviar tramas elaboradas de manera mal intencionadas, con el fin de dañar al servicio prestado. (Maciá, 2007, p. 15)

Uno de los más claros ejemplos de este tipo de ataque es hacia los sistemas operativos de computadores y se lo hace mediante la creación de tramas UDP, una vez realizada esa acción se generan bucles infinitos que consumen muchos recursos de memoria, disminuyendo así la velocidad de ejecución de la aplicación.(Chávez, 2011, p. 50)

1.5.1.2 Ataques de denegación de servicio según el origen

La sub clasificación de los ataques DoS según su procedencia son: realizados desde un único origen SDoS y los ataques distribuidos DDoS, generados desde varios bosts o maquinas infectada por alguna amenaza.

- Ataques realizados desde un solo origen

A los ataques realizados desde un solo origen también se los denomina SDoS (Simple Denial of service). En la figura 17-1, se evidencia que el atacante utiliza una única maquina infectada de malware, misma que se encuentra conectada a internet, para realizar tareas ilegales o malignas, provocando así irregularidades en la llegada de tráfico hacia su destino. (Cloudflare, 2019, p. 1)

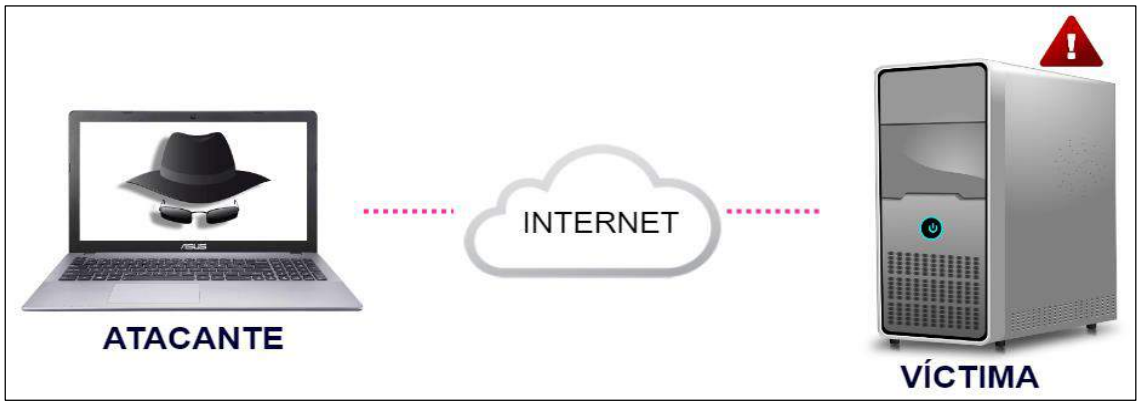


Figura 17-1: Ataque SDoS

Realizado por: TOAINGA Daniela, PEÑA Daniel

- Ataques de denegación distribuidos

Los DDos o distributed denial of service se diferencias de los SDoS, figura 18-1, por usar múltiples bots, reclutados a partir de un escaneo minucioso y una vez explotado las vulnerabilidades, el atacante tendrá un acceso completo a la misma, como si se tratará de un administrador, teniendo privilegios para enviar instrucciones maliciosas sincronizadas, impidiendo la normal circulación de tráfico de un servicio. (Cloudflare, 2019, p. 1).

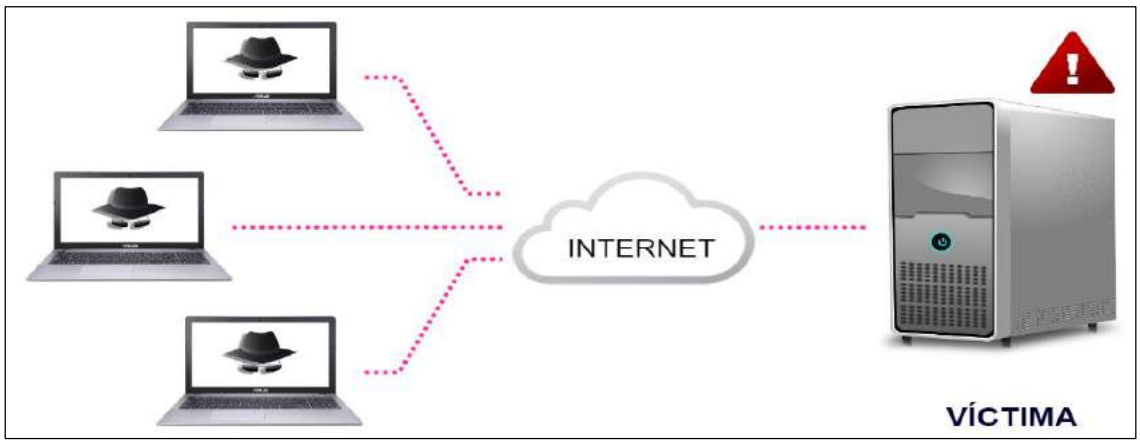


Figura 18-1: Ataque DDoS

Realizado por: TOAINGA Daniela, PEÑA Daniel

Estos tipos de ataques tienen una sub-clasificación de acuerdo a las capas de red, por ejemplo: los ataques DoS se ejecutan a nivel de la capa de aplicación: su finalidad es bloquear a servidores web (Windows, Apache, etc) con la ayuda de inundaciones GET/POST, por otro lado, se encuentran los ataques de protocolo que agotan los recursos de servidores o enrutadores de red y por último los ataques volumétricos los cuales saturan el ancho de banda de su objetivo. (IMPERVA, 2019)

1.5.1.3 Ataques a diferentes protocolos

Tomando en cuenta el modelo TCP/IP, a los ataques se los clasifican de acuerdo a la capa o protocolo al que pertenezcan. En este apartado se detallan los que afectan a las capas de red, transporte y aplicación por ser los más vulnerables:

- Ataques a la capa de red

Su principal es IP o Protocolo de Internet, que provee conectividad entre varios dispositivos mediante la interconexión de diversas redes. Otro es el ICMP o Protocolo de Mensajes de Control de Internet, usado para crear conexiones entre dispositivos y además controlar y notificar errores del protocolo IP. Y ataques a IGMP o Protocolo de Administración de Grupos de Internet usado por un host para notificar que pertenece a cierto grupo de multicast. (Abliz, 2011, p. 50, p. 7)

- Ataques a la capa de transporte

A través de la capa de aplicación se puede realizar la transferencia o enrutamiento de los paquetes, asegurando una comunicación fiable donde los datos que fueron transmitidos sean los mismos que se reciban al final del otro extremo. Los protocolos que corresponden a esta sección son: TCP o Protocolo de Control de Transmisión y UDP o Protocolo de Datagramas de Usuario.(Oracle, 2010)

- Ataques a la capa de aplicación

Se divide en dos categorías: la primera comprende los protocolos que proveen servicios directamente a los usuarios como: FTP, SSH, TELNET, HTTP, entre otros. Y la otra que abarca a los protocolos de soporte de red: DNS, RTP, DHCP, SIP, SMNP, etc…aquí también se encuentran RIP y BGP que son usados en enrutamiento. (Abliz, 2011,p. 8)

Un ejemplo de ataques a la capa de aplicación, es cuando se envían varias solicitudes HTTP con la intención de descargar algún archivo grande alojado en un servidor, pero al ser muy pesado

consumirá mucha memoria, ancho de banda. Un ataque más forzado se produce al combinar solicitudes HTTP para diferentes URL con la finalidad de imitar un tráfico normal. (Oracle, 2010, p. 9)

1.5.1.4 Ataques de denegación sofisticados

Entre los ataques de denegación sofisticados se puede mencionar a: los reflectivos y amplificados.

- Ataques por reflexión

Usan un host reflector para ejecutar lo planeado, desde este dispositivo se envían solicitudes al servidor con la dirección IP de la víctima, usando la dirección de origen como si se trataran de peticiones legitimas. Un reflector puede ser servidores: web, DNS o simplemente enrutadores.(Oracle, 2010, p. 9)

- Ataques por amplificación

Los atacantes se aprovechan de servicios que generan múltiples o grandes flujos de datos para amplificar trafico malicioso dirigido a la víctima. (Abliz, 2011, p. 50). Los servidores DNS son los más vulnerables a sufrir amenazas amplificadas por datagramas con poca información al momento de hacer consultas. (Aguirre, 2015, p. 11)

1.5.2 Estrategias defensivas ante ataques DoS

En cualquier red, se deben implementar mecanismos o estrategias defensivas ante cualquier amenaza, como son la prevención, detección, identificación del origen y mitigación de ataques de denegación de servicio. A continuación, se detalla cada uno:

1.5.2.1 Prevención

Este mecanismo es considerado como una línea de protección, que actúa antes de que el ataque se haya ejecutado, para reducir al máximo los daños ocasionados por el atacante. Pero esta estrategia no elimina por completo la amenaza de un ataque de denegación de servicio, por lo que es necesario introducir modificaciones para reforzar la seguridad en protocolos, aplicaciones y sistemas. (Maciá, 2007, p. 23)

- Monitorización, percepción del sistema y establecimiento de políticas de seguridad

La monitorización de un sistema es un factor clave que ayudará a conocer el estado del sistema y también detectar acontecimientos anormales. Todo administrador de red debe ejecutarla si es posible en tiempo real para tener conocimientos de como es el estado de los recursos físicos y lógicos.

Se puede monitorizar a: información de usuarios, acceso a aplicaciones, puertos, ancho de banda, etc., con la finalidad a un futuro detectar fácilmente ataques. Para tener una red más segura no solo es necesario la implementación de una monitorización o percepción de la misma, sino que también se deben incorporar el uso de antivirus, bloqueo de puertos, actualizaciones automáticas del sistema operativo, cortafuegos, firewalls, etc. (Maciá, 2007, p. 23)

1.5.2.2 Detección

Los ataques de seguridad detectados oportunamente son de vital importancia, para que los componentes defensivos actúen rápidamente. La detección muchas veces es realizada por agente, que actúan como intermediarios, por ejemplos: los firewalls, antivirus, los conocidos IDS (Sistemas de Detección de Intrusos), proxys, creación de scripts que controlen aplicaciones de los usuarios .(Aguirre, 2015, p. 12)

En el caso de los ataques de denegación de servicio, existen dos métodos de identificación: reconocimiento basado en firmas y anomalías.

- El reconocimiento de firmas está basado en patrones de ataques previamente conocidos, los mismos que son almacenados en bases de datos para luego ser comparados con el tráfico de la comunicación con el fin de descubrir ataques DoS. La desventaja de este paradigma es que solo detecta ataques conocidos, mas no nuevas amenazas.

- El reconocimiento basado en anomalías se centra en estudiar o analizar el comportamiento habitual del sistema para de esa manera identificar alguna anomalía. Esta metodóloga tiene diversas técnicas para identificar ataques como, por ejemplo: el sistema NOMAD, D-WARD, la estructura MULTOPS. (Maciá, 2007, p. 28)

1.5.2.3 Identificación del origen

Luego de haber detectado que la red está bajo amenazas de ataques DoS, se debe rastrear su procedencia, con la intención de determinar al autor. Muchas veces este proceso es complicado debido a que los agentes ejecutores del ataque usan técnicas tales como, suplantación de identidad e IP traceback. La identificación es un proceso que pocas veces son logradas con éxito, pero gracias a esto se puede conseguir una localización aproximada con el fin de desplegar métodos defensivos eficaces. (Maciá, 2007, p. 30)

En la actualidad muchas existen softwares y empresas, que ayudan a identificar la fuente de los ataques, en base a mediciones de velocidad de transferencia de información a fin de establecer la cantidad de peticiones y tráfico por parte de los clientes, otro método es clasificar las direcciones IP según el país de origen o simplemente se puede analizar el comportamiento de los usuarios del sistema con la intensión de identificar patrones repetitivos de tráfico. (Maciá, 2007, p. 30)

1.5.2.4 Mitigación

Una vez detectado e identificado la amenaza contra la seguridad del sistema, se debe desplegar rápidamente una serie de acciones que mitiguen o contrarresten el daño ocasionado, para que los servicios comprometidos regresen a funcionar normalmente. La mitigación se la puede ejecutar en los mismos dispositivos o en otros pero que se encuentren dentro de la misma infraestructura de red. Para el caso de denegación de servicio se optaría por crear o actualizar las políticas de cifrado, o aumentar más recursos físicos de red. (Paracuellos, 2016, p. 23)

En el caso de las redes SDN al estar su arquitectura desacoplada, fácilmente se los puede deshabilitar o reemplazar por nuevos equipos, para evitar seguir comprometiendo a más recursos de la infraestructura. La innovación que presenta SDN es su capacidad de reacción en tiempo real para eliminar flujos de datos malintencionados y también es capaz de impedir una comunicación con los atacantes.

CAPÍTULO II

2 MARCO METODOLÓGICO

En el presente capitulo, se detalla la metodología a usar para analizar las vulnerabilidades insider de ataques de denegación de servicio en las redes SDN, se muestra también la topología con sus respectivo direccionamiento y configuración, además de los servicios de red. Para lograr el objetivo planteado se utilizan las herramientas de pentesting Metasploit y Armitage y analizadores de trafico de red Openvas y Wireshare.

2.1 Metodología de la investigación

Para desarrollar el presente trabajo de investigación se aplican una serie de métodos y técnicas con el propósito de recabar, ordenar y analizar correctamente los datos obtenidos. A continuación, se detallarán los procedimientos y técnicas.

2.1.1 Tipo de investigación

El presente trabajo de investigación se estableció como una propuesta tecnológica, fundamentándose en la investigación aplicada, se elige este tipo de investigación, debido a que el presente proyecto tiene como objetivo analizar vulnerabilidades en redes definidas por software, y determinar el comportamiento de la red antes y después de la ejecutar amenazas.

2.1.2 Métodos de investigación

El presente proyecto es una investigación científica, debido a que está orientado a dar una solución a un problema, con el cumplimiento a ciertos objetivos; se opta en usar los siguientes métodos:

2.1.2.1 Método teórico

Se opta por usar el método teórico para tener conocimientos acerca del tema de investigación para buscar la solución, más conveniente.

Método analítico-sintético: se opta por usar este método, para analizar al controlador SDN considerado como el elemento primordial de las redes definidas por software, además, de los servicios implementados, antes y después de estar sometido a ataques de denegación de servicio

Método inductivo: se aplicará luego de que se haya aplicado el método analítico-sintético, es decir al finalizar la investigación, para plantear las conclusiones y recomendaciones de manera general en base a un análisis de datos particulares.

2.1.2.2 Método empírico

Observación científica: consiste en la observación y registros de eventos suscitados en el momento de la implementación del escenario de red, para su posterior análisis con el fin de exponer conclusiones y recomendaciones adecuadas.

2.1.3 Técnicas de la investigación

Una vez definidos los métodos necesarios para el desarrollo del presente proyecto también se deben determinar las técnicas. Las más adecuadas para esta investigación son la documental y la observación.

Técnica documental: se usará esta técnica para recolectar información de publicaciones científicas, tesis, revistas, páginas web, artículos, etc., para especificar parámetros, elegir el software de simulación, controlador SDN, metodología de análisis de vulnerabilidades y la topología a implementarse.

Técnica de la observación: Cuando se haya puesto en marcha la implementación del sistema se usará ésta técnica para analizar el comportamiento de la red cuando se encuentre bajo amenazas de ataques de denegación de servicio.

2.2 Concepción de la arquitectura de la topología de red

Para el desarrollo del presente proyecto se propone la siguiente topología de red, para proceder al análisis de vulnerabilidades en redes definidas por software. En la figura 1-2, se detalla el escenario a implementarse con sus respectivos elementos de red.

El escenario implementado, consta de un controlador, conmutadores open vswitch, entorno GNS3, servidores, clientes y atacantes, en donde cada uno cumple con funciones diferentes, toda la topología será implementada usando máquinas virtuales de Linux y también dispositivos externos. Para la comunicación interna de la red se realiza mediante el protocolo Openflow

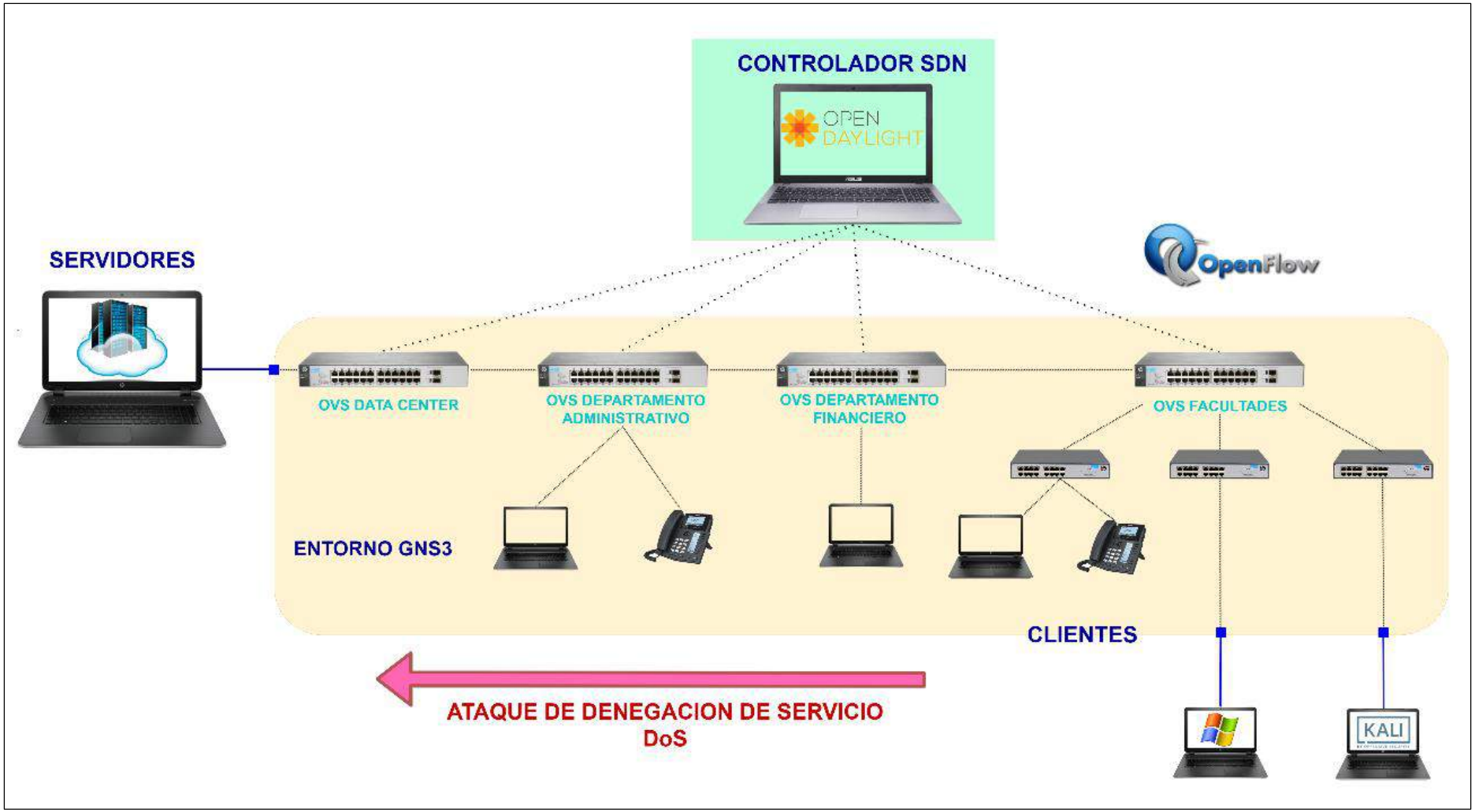


Figura 1-2: Topología de red

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019

2.3 Recursos requeridos por el proyecto

En la siguiente sección, se especifican los recursos necesarios y óptimos para realizar un escenario de prueba para la explotación de vulnerabilidades DoS en las redes definidas por software (SDN).

2.3.1 Análisis del controlador SDN

Para escoger el controlador optimo que mejor se adapte a los requerimientos para la elaboración del proyecto se deben tomar en cuenta ciertos criterios, a continuación, se detallan los más importantes a considerarse:

- Creación dinámica y automática de entradas de flujos
- Interfaz gráfica programable
- Compatibilidad con OpenFlow versiones (1.0, 1.2, 1.3)
- Documentación

Tomando en cuenta todos los parámetros detallados anteriormente, se procede a realizar un estudio de mercado, como resultado se obtuvieron a cuatro controladores con características más cercanas a las requeridas, en la tabla 1-2 se observan los resultados.

Tabla 1-2: Comparación entre controladores SDN

Características	Opendaylight	Ryu	POX	Floodlight
Plataformas	Linux, MAC, Windows	Linux	Linux, MAC, Windows	Linux, MAC, Windows,
Interfaz gráfica	Si	Si	Si	Si
Creación dinámica y automática de flujos	Si	No	No	No
Líneas de código	2 500 000	116 000	20 000	44 000
Lenguaje	Java	Phyton	Python	Java
Virtualización	Mininet y OvS	Mininet y OvS	Mininet y OvS	Mininet y OvS
Open Source	Si	Si	Si	Si
Versiones OpenFlow	1.0, 1.2, 1.3	1.0, hasta 1.5	1.0	1.0
Soporte Openstack	Si	Si	No	Si
REST API	Si	Si (SBD)	No	Si
Código abierto	Si	Si	Si	Si
Fecha de lanzamiento	2013	2012	2012	2013
Documentación	Buena	Media	Baja	Buena

Fuente: http://www.ijcncs.org/published/volume5/issue11/p1_5-11.pdf

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

Una vez que se eligen los criterios de selección se procede a evaluarlos, para determinar al dispositivo óptimo para la elaboración del presente proyecto. El método utilizado para evaluar al más apropiado es cuantitativo, con calificaciones, en escalas que varían desde 1 al 5; donde 1 equivale a pésimo, 2 regular, 3 bueno, 4 aceptable y 5 excelente, tal como se muestra en la tabla 2-2.

Tabla 2-2*:* Método de evaluación cuantitativo-cualitativo.

Peso	Juicio valorativo
1	Pésimo
2	Regular
3	Bueno
4	Aceptable
5	Excelente

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

La evaluación del controlador optimo a utilizar en el proyecto se muestra en la tabla 3-2, con las calificaciones detalladas en la tabla 2-2.

Tabla 3-2: Evaluación cuantitativa de los controladores SDN.

Criterios y ponderación.	Opendaylight	Ryu	POX	Floodlight
Plataformas	5	1	5	5
Interfaz GUI	5	3	1	2
Creación dinámica y automática de flujos	5	4	1	2
Líneas de código	5	4	2	3
Versiones Openflow	4	5	1	1
REST API	5	4	1	5
Documentación	5	3	2	5
TOTAL	**34**	24	13	23

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

La opción que cumple con los requerimientos es el controlador Opendaylight, el cual con 34 puntos resulta ser idóneo para usarse en el presente proyecto. Opendaylight es un controlador de código abierto, incluye interfaces Northbound y Southbound. Y además de admitir el protocolo OpenFlow también permite otros protocolos de licencia libre y también el uso de herramientas: maven, OSGi, karaf, interfaces JAVA y REST. (OpenDaylight Project, 2018a).

Como se evidencia en la tabla 3-2, en el campo plataformas la puntuación para ODL, POX y Floodlight es de 5 se debe a que soportan Windows, Linux y Mac, en GUI Opendaylight tiene 5 debido a que la interfaz es de fácil entendimiento para el usuario para crear, eliminar, modificar de flujos y también visualizar la topología, etc., en comparación a los demás controladores Ryu y Floodlight, en cambio a POX se le asigna el valor de 1 por no poseerla. En la creación dinámica y automática de flujos y las REST API, también tiene 5 por la misma razón de la interfaz GUI. Pero en lo cuanto a Openflow, tiene 4 dado a que no soporta todas las versiones de ese protocolo.

2.3.2 Análisis del software de simulación

Para elegir al software de simulación más óptimo de acuerdo a los requerimientos necesarios para la implementación del sistema, se toma en cuenta las características más relevantes las mismas que se detallan a continuación:

- Compatibilidad con el protocolo OpenFlow.
- Capacidad de nodos activos.
- Interoperabilidad
- Consumo de memoria.
- Escalabilidad.

En base a los parámetros señalados anteriormente se procede a realizar una comparación entre tres softwares, tal como se indica en la tabla 4-2, que posibilitan la implementación de redes SDN.

Tabla 4-2: Comparativa entre softwares de simulación SDN.

Características	Mininet	GNS3	EstiNet
Precio	Ninguno	Ninguno	Alto
Documentación	Media	Alta	Baja
Soporte Windows	No	Si	No
Soporte Linux	Si	Si	Si
Simulador	Si	Si	Si
Emulador	No	Si	Si
Compatible con controladores reales	Todos	Todos	Todos
Escalabilidad	No	Si	Si
Orientación	Solo a SDN	SDN y tradicionales	SDN y tradicionales
Soporte GUI	Adaptable	Si	Si

Fuente: http://journals.pntu.edu.ua/mist/article/view/571/493

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

Para la selección e aplica el mismo método detallado en la tabla 2-2, que es la evaluación cuantitativa, para elegir la mejor opción en cuanto a software. Esto se puede apreciar en la tabla 5-2.

Tabla 5-2: Evaluación cuantitativa del software.

Criterios y ponderación	Mininet	GNS3	EstiNet
Precio	5	5	1
Documentación	3	5	1
Soporte Windows	1	5	1
Soporte Linux	5	5	5
Simulador	5	5	5
Emulador	1	5	5
Compatible con controladores reales	5	5	5
Escalabilidad	2	4	5
Orientación	3	5	5
Soporte GUI	2	4	5
TOTAL	32	**48**	38

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

El software que obtuvo el mejor puntaje de acuerdo a la tabla 5-2, es GNS3 por estar diseñado y orientado a la creación de todo tipo de topología de red con sistemas operativos reales en todos sus dispositivos, facilitando a futuro la implementación rápida con equipos de tipo hardware, además soporta protocolos de conmutación y enrutamiento, así como la posibilidad de NFX y SDN. (Pincay, 2015, p. 79). En el Anexo A, se detalla paso a paso la instalación de GNS3.

2.3.3 Análisis de la metodología para el estudio de vulnerabilidades.

Para elegir la metodología de análisis de vulnerabilidades, se toman en cuenta los siguientes criterios y así lograr establecer la más adecuada. Las características consideradas para evaluarlas son:

- Métodos de análisis (cuantitativo o cualitativo)
- Propiedades de la seguridad (confidencialidad, integridad, disponibilidad)
- Alcance (organizaciones: grandes, pequeñas y medias)
- Se rige a la norma de seguridad ISO 270001
- Documentación

En base a los aspectos mencionados, se procede a comparar dos metodologías, para luego elegir a la que cumpla los requerimientos del proyecto, en la tabla 6-2 se evidencia la información de dos técnicas para analizar vulnerabilidades en redes.

Tabla 6-2: Características de las metodologías de análisis de vulnerabilidades.

Descripción	OSSTMM	OCTAVE
Propiedades de la seguridad	Integridad, confidencialidad y disponibilidad	Integridad, confidencialidad y disponibilidad
Creador	ISECOM	SEI- y CERT
País	Estados Unidos	Estados Unidos
Alcance	Grandes empresas	Todo tipo de empresas
Tipo de análisis	Cualitativo	Cualitativo y cuantitativo
Precio	Gratuito	Gratuito
Se adapta a la ISO 270001	No	Si
Documentación	Media	Alta

Fuente: https://dspace.ups.edu.ec/bitstream/123456789/14631/1/UPS%20-%20ST003221.pdf

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

Para seleccionar la mejor opción, se aplica el mismo método cuantitativo de las secciones 2.3.1 y 2.3.2, con las mismas escalas de calificación. En la tabla 7-2, se muestra el cuadro comparativo entre las metodologías de análisis de vulnerabilidades: OSSTMM y OCTAVE.

Tabla 7-2: Análisis de la metodologías para análisis de vulnerabilidades.

Criterios y ponderación	OSSTMM	OCTAVE
Métodos de análisis	3	5
Propiedades de la seguridad	5	5
Documentación	3	5
Alcance	3	5
Estándar ISO 270001	5	5
TOTAL	19	**25**

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

Con un resultado de 25 puntos, la metodología apta para el desarrollo del presente trabajo es OCTAVE. En el primer criterio que corresponde a los métodos de análisis, la metodología con mayor valoración es OCTAVE, debido a que posee dos formas de evaluación: cuantitativa incluyendo valores numéricos de probabilidades y cualitativa que se basa en la valoración subjetiva y la repetitividad con que suceden los eventos amenazantes, mientras que OSSTMM tiene 3 debido a que solo usa la de tipo cualitativa. En cuanto al alcance también tiene un puntaje de 5 puntos, debido a que está destinada para obtener información de cualquier organización, sea

grande, pequeña o mediana, y también a nivel interno y externo de la corporación a diferencia de OSSTMM que solo está orientada para las grandes empresas y solo para análisis outsider.

En cuanto a las propiedades de la seguridad OSSTMM y OCTAVE tienen la misma puntuación de 5, porque están enfocados a proteger a la integridad, confidencialidad y disponibilidad de la información. En lo que respecta a estándar, las dos se rigen a la norma de seguridad ISO 270001, para gestionar la seguridad de la información de una organización, mediante el uso del hacking ético.

2.4 Implementación de OCTAVE para el análisis de vulnerabilidades.

2.4.1 Primera etapa: Identificación de los activos y amenazas de red.

En la etapa inicial se identifican todos los activos informáticos de la organización, utilizando la técnica de la observación, se realizan perfiles u informes de cada elemento con sus respectivas características de hardware y software, fecha de creación, funcionalidades, en esta sección también se identifican los posibles peligros o amenazas a los que están expuestos.

2.4.2 Segunda etapa: Detección de vulnerabilidades

En base a la información obtenida en la primera fase, se procede a examinar las vulnerabilidades en toda la red y en cada uno de los dispositivos (controlador, open vswitch, servicios y usuarios), para detectar las fallas del sistema se utiliza el software Openvas, la información obtenida en esta sección permite al administrador de red determinar las áreas más críticas de la infraestructura. En el anexo B, se encuentra la instalación de Openvas.

Luego de obtener los resultados del escaneo con Openvas a los diferentes dispositivos, se procede a un análisis, para conocer cuál de esas vulnerabilidades son las que con más probabilidad suceden.

Para predecir el nivel de probabilidad de ocurrencia de las vulnerabilidades, se basa en el informe del CCN-CERT, en donde consideran tres niveles (alto, medio y bajo), en la tabla 8-2 se detallan cada una de las calificaciones dictadas por ese organismo.

Tabla 8-2: Nivel de probabilidad de ocurrencia de vulnerabilidades.

NIVEL DE PROBABILIDAD	DETALLES
Alta	Comprende un gran riesgo para la organización, afectando a toda la seguridad de la red SDN. Por lo que ese problema requiere de una solución inmediata.
Media	El impacto ocasionado no es significativo, pero requiere de un seguimiento constante y también de solución rápida.
Baja	No representa mayores inconvenientes para la seguridad. Se puede solucionar a futuro los inconvenientes ocasionados.

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

Con el objetivo de explotar las vulnerabilidades detectadas, se procede a aplicar amenazas en este caso es la ejecución de ataques de denegación de servicio, y para determinar el rendimiento de la red se utilizan dos indicadores.

- Ancho de banda
- tiempos de respuesta en enviar los paquetes de un punto a otro.

Se eligen solo dos criterios debido a que el presente trabajo de titulación está orientado a evaluar el comportamiento de la infraestructura de red SDN, antes y después de la aplicación de ataques de denegación de servicio, con el propósito de asegurar la disponibilidad de la red y servicios.

Indicador 1: Ancho de banda.

En el caso del indicador 1 que corresponde al impacto del ataque ocasionado al ancho de banda, para definir el nivel de efectividad, , tabla 9-2, de los ataques de denegación de servicio, se guía en un informe emitido por el CCN-CERT, denominado Ciberamenazas y Tendencias 2017, en donde consideran tres niveles (alto, medio y bajo), para calcula este valor se utiliza la fórmula:

$$Efectividad = \frac{número\ de\ paquetes\ detectados}{paquetes\ soportados\ por\ el\ sistema}\ x\ 100$$

El nivel alto se considera cuando la efectividad está comprendida entre los rangos que van desde 70% a 100%, las amenazas medias cuando están entre el 35% al 69% y de baja consecuencia a las que se encuentran entre 10% y 34%.

Tabla 9-2: Efectividad de los ataques DoS.

NIVEL	PORCENTAJE	DETALLES
Alto	70%-100%	Muy riesgoso para la disponibilidad
Medio	35%-69%	Importante
Bajo	10%-34%	No comprende ningún riesgo

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

Luego de haber obtenido los dos valores usando la herramienta wireshark, se puede usar la fórmula para determinar el grado de efectividad de los ataques a la hora de explotar las vulnerabilidades, para determinar el impacto causado, se utiliza la información proporcionada por el CCN-CERT descrita en la tabla 9-2.

- **Indicador 2:** Latencia

En el caso del segundo indicador: impacto ocasionado en los tiempos de respuesta o latencia, para definir los niveles referenciales de latencia en redes SDN, se usa la información proporcionada por el CCN-CERT, donde indica que se puede medir la calidad de conexión en tres escalas, como se observa en la tabla 10-2, en donde cada una de ellas tiene sus propios rangos.

Tabla 10-2: Niveles para medir la latencia en redes SDN.

NIVEL	VALORES	DETALLES
Alto	Menos a 1,5 ms	La conexión de punto a punto es óptima, cuando no tarda más de 1,5 milisegundo
Medio	1,5ms – 5ms	La red está trabajando con sobre-procesamiento
Bajo	Mayores a 5 ms	Indica que la red está a punto de colapsar

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

2.4.3 Tercera etapa: Elaboración de planes de contingencia

La tercera fase es desarrollar un plan de contingencia o guía de buenas prácticas con consejos prácticos, de modo que todo administrador de redes SDN pueda implementarlas a fin de contrarrestar o mitigar vulnerabilidades y prevenir daños a futuro.

2.5 Desarrollo del proyecto

Una vez que se haya definido la metodología de investigación, de igual modo los requisitos tanto de hardware como de software, se procede a la implementación del escenario, todos los pasos de: instalación, configuración de dispositivos de red y pruebas se detallan en este apartado.

2.5.1 Escenario propuesto

Mediante una entrevista realiza al personal técnico del Departamento de Tecnologías de la Información y Comunicación DTIC- ESPOCH, se puedo determinar el tipo de escenario a implementarse. En base a la encuesta se propone una red tipo campus académico, similar a la implementada en la ESPOCH, por el motivo de que es una red compleja que abarcan casi todos los servicios de red existente en cualquier organización. Luego ese escenario va a ser sometido a varios ataques de denegación de servicio (DoS).

La topología mostrada en la figura 2-2. Consta de un controlador, Open vSwitches, switches de capa 2, servidores, clientes. Y solo comprende a un solo ramal de la institución, trabajando con cuatro nodos de interconexión (switches), que dividen en secciones o departamentos a la infraestructura, cada uno de ellos funcionando independientemente del otro, además los conmutadores son encargados de redirigir el tráfico a los otros nodos.

El escenario simulado se especifica como estrella extendida debido a que se centra en el dispositivo llamado controlador y toda configuración se establecen en él. Además de que este modelo es el más desplegada en todo tipo de entidades por su escalabilidad y autonomía de cada nodo, de modo que si cualquiera de ellos es atacado se puede aislarlo con facilidad para no comprometer a toda la infraestructura y también los servicios ofertados hacia los clientes son básicos por las limitaciones que existen a nivel de hardware. La otra característica es que es hibrida por conformase con dispositivos que comprenden o no el protocolo OpenFlow,

Para la puesta en marcha de la red se usan 3 PC externa: en la primera se crea la topología completa en el software GNS3 y en una máquina virtual se aloja al controlador opendaylight , desde el cual se envía acciones a cada uno de los conmutadores open vswitches de acuerdo a las necesidades de red., en el segundo computador se crean los servidores HTTP, DHCP, FTP, DNS y VoIP, y desde el último computador se ejecutan los ataques de denegación de servicio, usando Kali Linux y sus herramientas Armitage y Metasploit. Para la conexión externa entre las maquinas físicas se usa un dispositivo llamado concentrador o ethernet hub. Y finalmente para analizar la red se usa Wireshark y para el escaneo de vulnerabilidades Openvas.

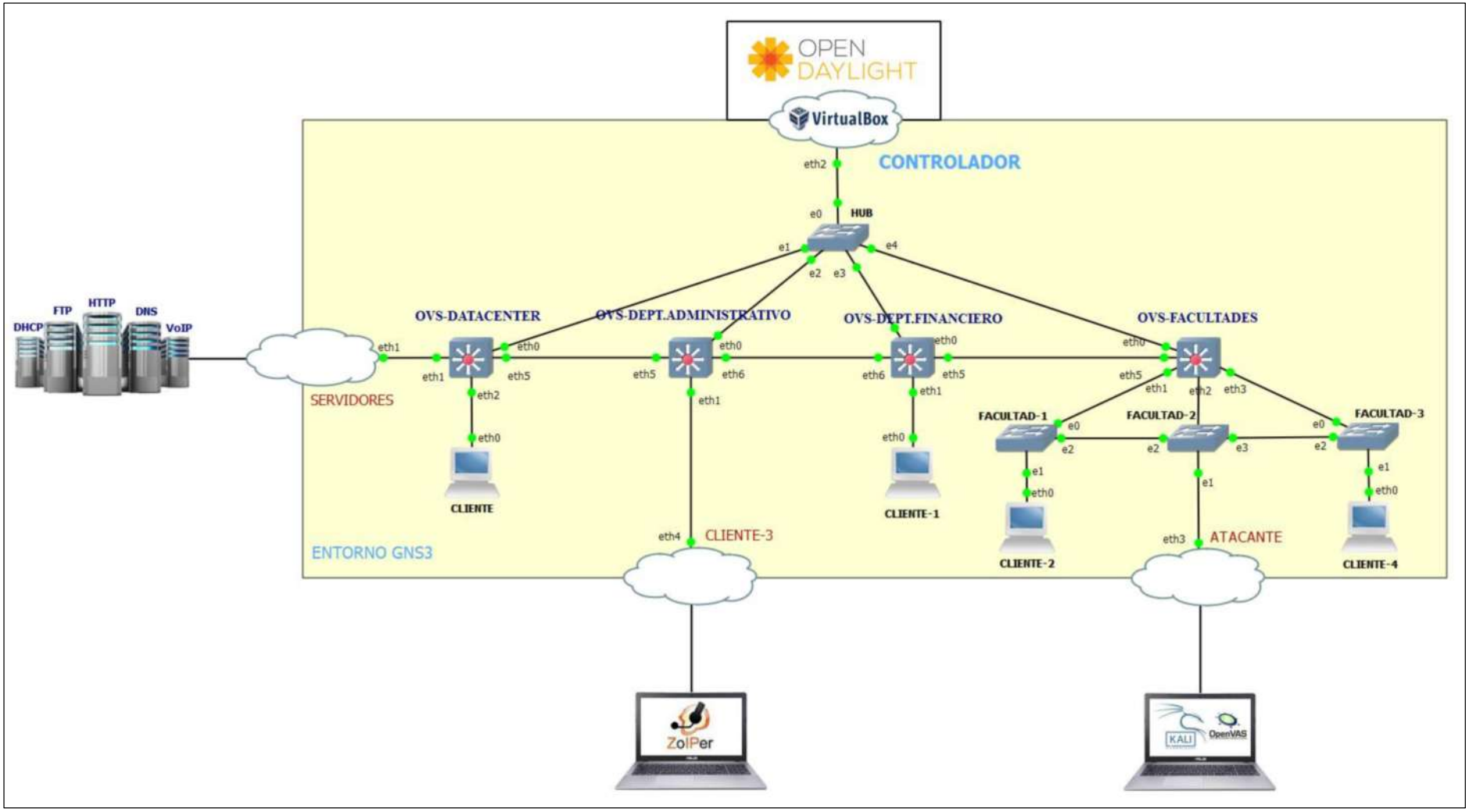


Figura 2-2: Escenario SDN implementado.

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019

2.5.2 Configuración de la topología de red

Luego de simular el escenario de red, se procede a configurar todas las direcciones IP con sus respectivas puertas de enlace de cada dispositivo, todo este proceso se evidencia en la tabla 11-2.

Tabla 11-2: Direccionamiento de los dispositivos de red

Equipos	Sistema operativo	Dirección ip	Dirección MAC
Controlador SDN	Ubuntu 14.04	192.168.1.5	08:00:27:8B:49:CC
Bloque de Servidores	Centos 7	192.168.1.8	00:27:0E:07:43:10
OVS-Data-C		192.168.1.10	22:C5:AA:7C:14:75
OVS-DEPT-ADMIN		192.168.1.11	76:40:52:F6:27:16
OVS-DEPT-FINANCIERO		192.168.1.12	86:0A:FD:D9:B1:06
OVS-DEPT-FACULTADES		192.168.1.13	0E:34:A6:D6:09:ED
Atacante/OpenVas	Kali Linux 2018.3	Cliente DHCP	30:65:EC:2A:1C:0D
Usuarios	Webterm Ubuntu-Docker-Guest	Cliente DHCP Cliente DHCP	8E:3F:6E:83:56:22 B6:4B:78:F2:2A:B5 B6:2F:11:BC:A8:A2 82:1E:D3:68:DB:09
Usuario VoIP	Windows 8.1	Cliente DHCP	3C:07:71:5A:E0:A3

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

Para la comunicación de los conmutadores con el controlador, se debe establecer la interfaz de red con su respectivo puerto y también habilitar el protocolo Spanning Tree para la comunicación entre switches. Mientras que para que los clientes detecten automáticamente las direcciones dadas por el servidor, se debe modificar el archivo /etc/network/interfaces, habilitando las líneas de código de configuración DHCP para la interfaz requerida.

2.5.3 Instalación y configuración del controlador Opendayligh.

Usando VirtualBox, se crea una maquina virtualizada con la distribución de Ubuntu y versión actualizada de JAVA, para posteriormente instalar la cualquier version de Opendaylight. El controlador SDN para ejecutarse sin ningún problema, se necesita de características mínimas detalladas en la tabla 12-2.

Tabla 12-2: Requerimientos para ejecutar el controlador

Requerimientos	Datos
Sistema Operativo	Ubuntu 14.04
Versión de Java	1.8 o superior
RAM	4 GB
CPU	2
Memoria de video	12 MB
Almacenamiento	18 GB

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019

Se sugiere instalar la distribución cuarta de ODL denominada Beryllium SR4, dado que no requiere tantos recursos en hardware para su correcta ejecución. Para descargar cualquier versión dirigir al repositorio oficial y obtener el archivo en formato .zip, Una vez descargado dentro de la máquina virtual de Ubuntu, acceder desde consola al directorio del archivo y ejecutar los comandos de instalación, especificados en el Anexo C. Para habilitar todos los servicios de ODL se usa la opción Karaf, figura 3-2.

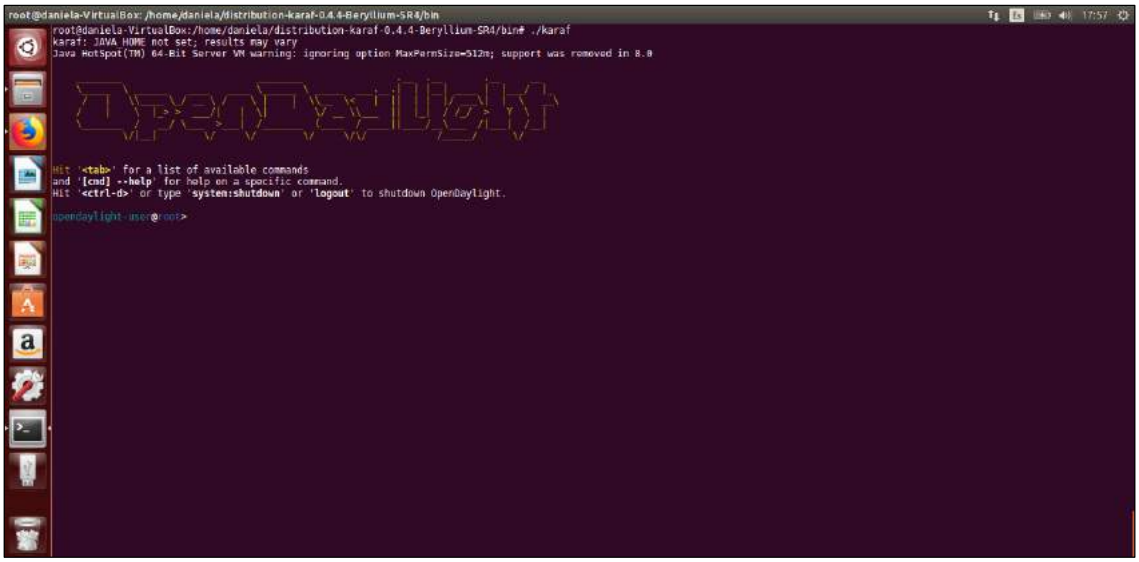


Figura 3-2: Entorno Karaf.

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

Para llevar a cabo las tareas de configuración, administración y otros recursos básicos. Es indispensable que la máquina virtual disponga de conexión a internet. Para enlistar las características disponibles se usa el comando *festure:list* pero si se requiere visualizar los ya instalador se añade *–i* y en caso de ser más específicos en la búsqueda agregar *grep*. Cuando se requiera instalar alguna característica se lo hace a través del comando *feature:install* seguido del o los nombre de la característica.

Para desarrollar del presente proyecto se instalan las siguientes funcionalidades descritas, donde cada uno cumple funciones distintas.

- odl-restconf Soporte de APIs
- odl-l2switch-switch Funciona como switch de capa 2
- odl-dlux-all Interfaz gráfica web
- odl-mdsal-apidocs Acceso a Yang APIs

Para ingresar a la interfaz gráfica, figura 4-2, la dirección IP, corresponde al del adaptador de red de la máquina virtual donde está alojado el controlador Opendaylight. Luego de iniciar sesión en la parte izquierda se despliega un panel con opciones para: visualizar la topología, información de nodos, la interfaz YANG UI, etc.

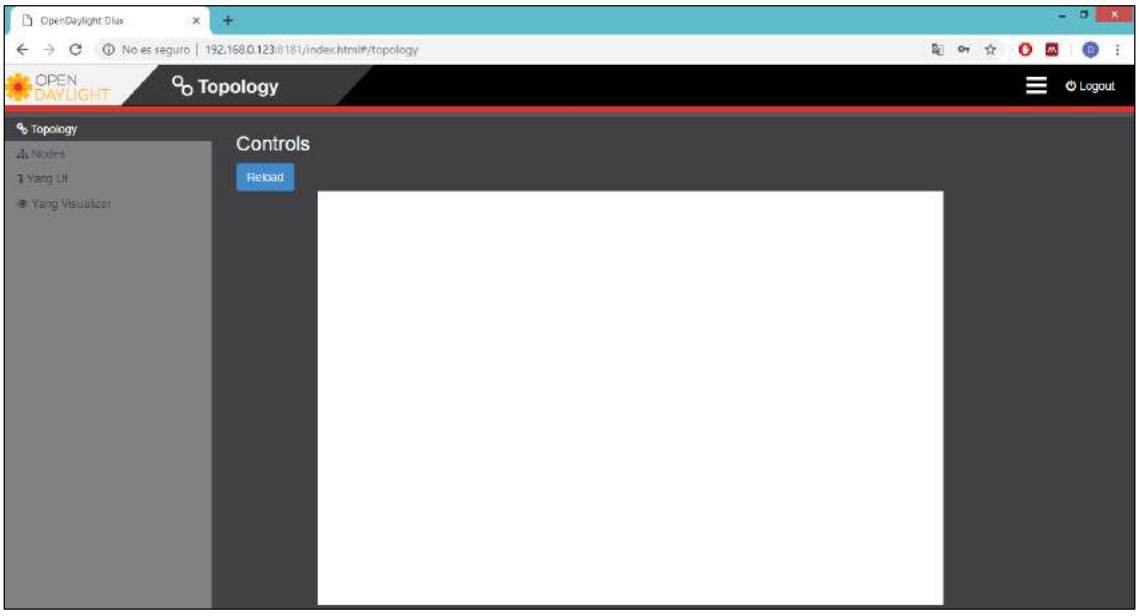


Figura 4-2: Entorno Opendaylight.

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

Después de tener cargado el escenario en Opendaylight se procede a configurar todos los flujos de tráfico de la red, pero para ello es indispensable conocer el identificador de cada elemento, ingresando al menú en la sección *Nodes*.

2.5.4 Desarrollo de los servicios de red

La plataforma en la que se desarrollan los servidores: DHCP, FTP, VoIP, HTTP y DNS, es en el sistema operativo Centos, por ser una distribución gratuita, estable, rápido y confiable en

comparación a otras distribuciones basadas en Linux. Para que los servicios funcionen de manera correcta se necesita una PC, con las características especificadas en la tabla 13-2. En el Anexo D se encuentra la instalación de Centos y en el anexo E están detalladas las todas las configuraciones de cada servicio.

Tabla 13-2: Requerimientos para los servicios.

Requerimientos	Valores
Sistema Operativo	CenTos 7 –Linux
Memoria RAM	2 Gb
Almacenamiento	500 Mb
Procesador gráfico	Intel G41 x86/MMX/SSE2

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

- Servidor DHCP

DHCP o protocolo de configuración dinámica de hosts. El objetivo principal de este servidor es la configuración dinámica de parámetros como: direcciones IP, mascara de subred, gateway, etc., de dispositivos conectados a una misma red, simplificando así la administración de la red de una manera centralizada y automática. DHCP es un protocolo de transporte que usa los puertos UDP 67 para servidores y 68 para clientes. (INTEF, 2012, p. 2) .

- Servidor FTP

El servidor FTP, o protocolo de transferencia de archivos, usa el protocolo TCP/IP para compartir remotamente ficheros entre cliente-servidor. Al momento de transmitir la información lo hace de modo autentificado, es decir, que requiere de un usuario y contraseña para acceder, y de manera bidireccional, pero sin seguridad alguna debido a que no utiliza ningún cifrado. Usa por defecto los puertos TCP 20 o puerto de datos, de acceso a FTP y 21 o puerto de control, donde se especifican parámetros de conexión como puertos, direcciones, etc. (Alvarez y etc, 2019) .

- Servidor DNS.

Los servidores DNS (Sistema de Dominio de Nombres) asocian con un nombre a una dirección IP, por lo tanto, cada dominio es único y fácil de recordar. Este servidor también conserva el modelo cliente/servidor, trabaja con el puerto 53 TCP/UDP para responder a toda consulta. Existen diferentes tipos de dominios dependiendo del tipo de organización, país, etc., por ejemplo: .com, .es, .edu, .org, .ec, entre otros. (INTEF, 2012, p. 3).

Es importante contar con el servicio de internet para descargar todos los archivos. Con la herramienta Bind fácilmente se crean servidores DNS uno con el rol de maestro y otro como esclavo.

- Servidor HTTP

Los servidores HTTP o web son sitios diseñados exclusivamente para transferir datos de hipertexto (páginas web, archivos, aplicaciones) por lo que deben disponer de acceso al internet. Trabajan con el modelo cliente-servidor, es decir el cliente hace la petición y el servidor atiende a la solicitud mediante los puertos TCP 80 (por defecto) y 443 para HTTPS. Estos servidores requieren de otros recursos como: DNS, FTP, etc.

La aplicación más usada a nivel mundial en este tipo de servidores HTTP es APACHE por ser de acceso libre, y también por poseer componentes de fácil personalización y configuración, además permite alojar varios sitios web sin limitaciones. La ventaja de APACHE frente a la seguridad es su modo de accedo autenticado. (Mifsuf, 2017)

- Servidor VoIP

VoIP es una tecnología de comunicación en tiempo real que utiliza el protocolo IP para transmitir voz. La telefonía IP incluye a dos protocolos SIP (UDP/TCP 5060) y RTP: el primero encargado de todos los detalles de la comunicación y el otro para transmisión de datos, por lo tanto, son aptos para ofrecer varios servicios como: llamadas a cualquier lugar del mundo, videoconferencias con una o varios clientes al mismo tiempo, etc., a costos bajos en comparación con la telefonía tradicional. (Soler, 2015).

Para implementarse el servicio de VOIP se requiere de una centralita encargados de operar todas las bases de datos, el software más usado en este servicio es Asterisk.

2.5.5 Desarrollo de los ataques de denegación de servicio

Se usa la plataforma Kali Linux, la cual contiene un conjunto de herramientas muy útiles para realizar pruebas de penetración y hacking ético, y junto a este software las herramientas Metasploit y Armitage.

Metasploit es una herramienta de auditoría enfocada al análisis de vulnerabilidades de los sistemas informáticos, contiene varias aplicaciones llamadas exploit, enfocados en analizar: sistemas

informáticos, servicios web entre otros, con un excelente grado de desarrollo de explotación, además ayuda a los administradores de red a llevar un conjunto de armas para estar un paso adelante de los atacantes. Mientras que Armitage sirve para visualizar a los objetivos a atacar.

Para la ejecución correcta de los ataques de denegación de servicio, necesita una PC, con las características especificadas en la tabla 14-2. En el Anexo E, están detalla la forma de instalación de Kali Linux.

Tabla 14-2: Requerimientos para Kali Linux.

Requerimientos	Valores
Sistema Operativo	Kali Linux
Herramientas	Metasploit y Armitage
Memoria RAM	4 Gb
Almacenamiento	500 Mb
Procesador gráfico	Intel G41 x86/MMX/SSE2

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

CAPÍTULO III

3 MARCO DE RESULTADOS

En el presente capítulo se analizan los resultados obtenidos de la implementación de ataques de denegación de servicio (DoS) en una infraestructura de redes definidas por software SDN, utilizando la metodología OCTAVE para un mejor estudio de los riesgos que puedan afectar al normal funcionamiento de la red misma que se desarrolla en tres fases: identificación de activos y amenazas, detección de vulnerabilidades de la red y por último el plan de contingencia que abarca una guía con consejos prácticos para mitigar o contrarrestar cualquier incidente que comprometa a la disponibilidad.

3.1 Análisis de la simulación: conectividad

En este apartado se verificó si existe conectividad en la red, para lo cual se utilizó el protocolo ARP, con la ayuda de este protocolo el controlador localizó a todos los dispositivos de red y con la herramienta ping se comprobó si el nodo de destino es alcanzable desde el origen.

3.1.1 Integración de elementos de red

Luego de haber completado las configuraciones de las secciones que conforman la topología SDN, el siguiente consistió en acoplar las secciones y formar la red, todo este procedimiento de integración se encuentra especificado en el Anexo G. Luego se procedió a realizar pruebas, usando la herramienta ping se verificó la conectividad existente entre los clientes y servidores que conforman la red.

3.1.2 Visualización de la topología en ODL

Después de verificar conectividad entre todos los nodos, se pudo visualizar y monitorear la topología de red desde la interfaz gráfica web del controlador Opendaylight, en la opción *topology*.

En la figura 1-3, se evidencia el escenario SDN implementado con sus respectivos nodos e identificadores. Y para conocer información más detallada de los nodos se desplegó la opción *Nodes*, ubicada al lado izquierdo de la interfaz.

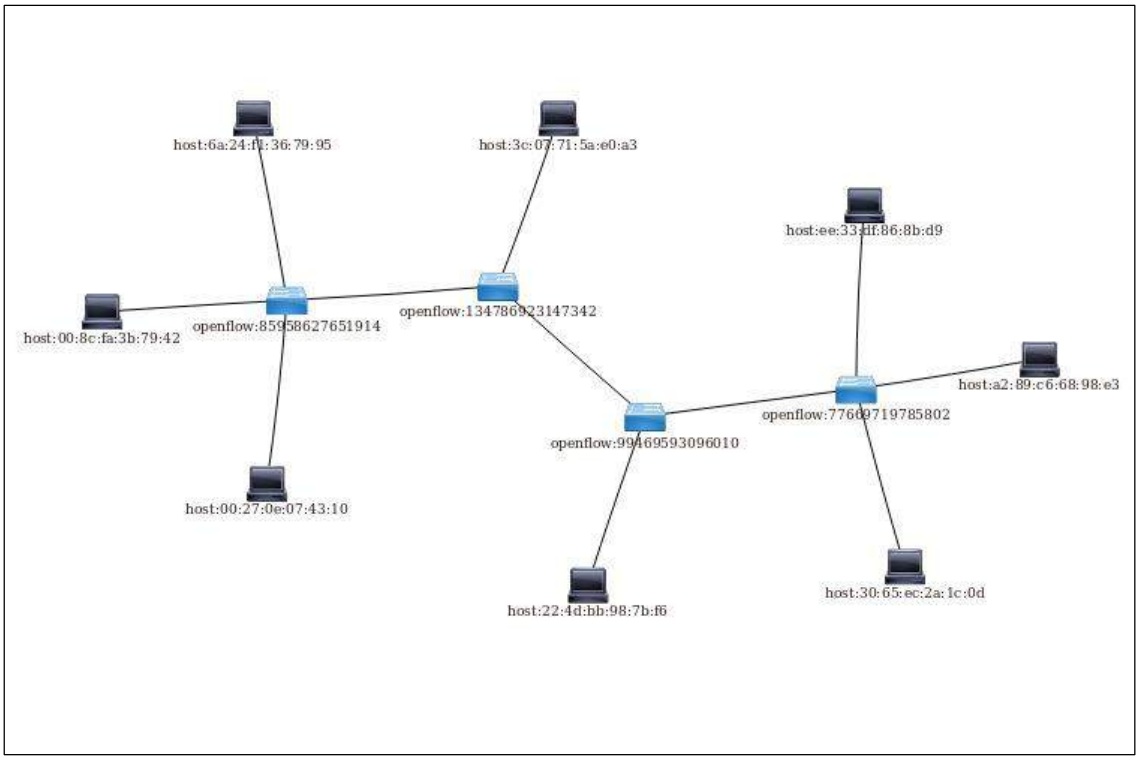


Figura 1-3: Topología visualizada en ODL.

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

En la figura 2-3, se aprecia la información de los cuatro switches generados. La administración de flujos o políticas de la red se realizó en el modulo YANG UI, en el siguiente bloque: *opendaylight-inventory / config / nodes / node {ID} / table {ID} / flow {ID}*, donde se llenaron todos llenar los campos.

Node Id	Node Name	Node Connectors	Statistics
openflow:134786923147342	None	16	Flows | Node Connectors
openflow:85958627651914	None	16	Flows | Node Connectors
openflow:77669719785802	None	16	Flows | Node Connectors
openflow:99469593096010	None	16	Flows | Node Connectors

Figura 2-3: Información de los switches de la red SDN.

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

En la figura 3-3, se observa una captura de tráfico, de la comunicación entre cliente-servidor, para ello se envió ping echo request desde la dirección IP 192.168.1.69 hasta los servicios con dirección IP 192.168.1.8. y estos respondieron con una solicitud de echo reply. Para este análisis se utilizó el software Wireshare en el servidor.

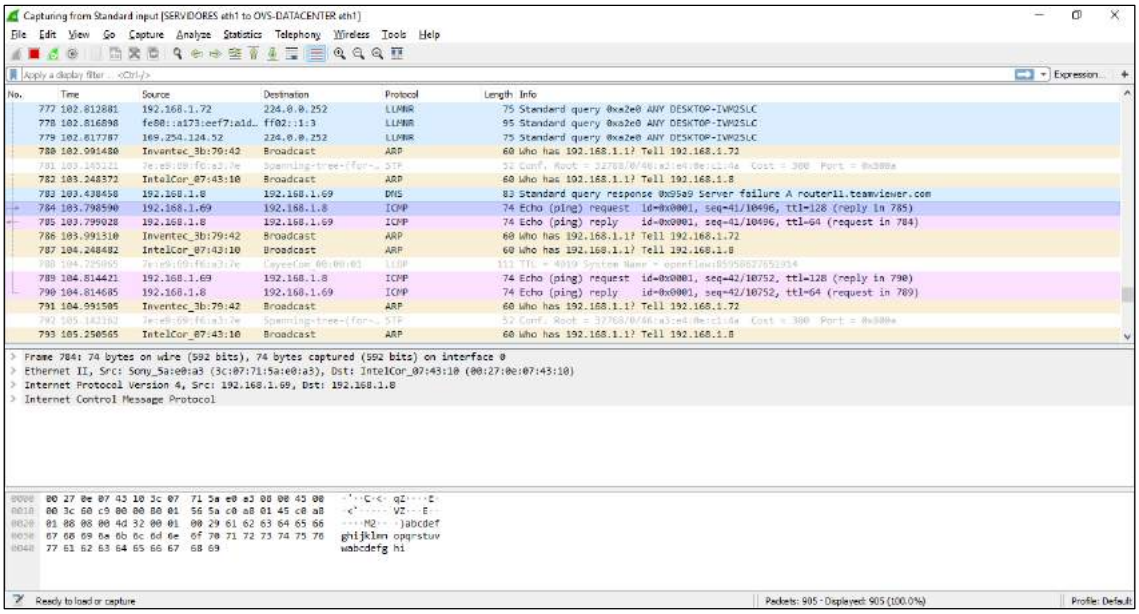


Figura 3-3: Comunicación clientes-servidores

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019

3.2 Resultados de la implementación de OCTAVE.

Para detectar las vulnerabilidades existentes en la red SDN implementada, se utilizó la metodología OCTAVE, desarrollada en tres fases.

3.2.1 Identificación de los activos y amenazas de red.

Como primer paso se evaluaron riesgos, donde se debe identificar a todos los activos informáticos de la organización (controlador, open vswitch, servidores y clientes), para conocer la información de cada uno, dirigir al Anexo H, en esta etapa también se detallaron todas sus posibles amenazas a las que están expuestas.

En la tabla 1-3, se muestra un ejemplo para llenar el perfil, donde debe ingresar información del activo a analizar con su respectiva versión de trabajo, los requerimientos que necesita en hardware y software, el modo de funcionamiento, responsable que está a cargo, requerimientos de seguridad, en cuanto a la disponibilidad, confidencialidad e integridad y las posibles amenazas a las que están expuestos, basándose en información de fabricantes o desarrolladores.

Tabla 1-3: Ejemplo de perfil del activo de la información.

HOJA DE TRABAJO DE LA METODOLOGÍA OCTAVE		
PERFIL DE ACTIVOS DE LA INFORMACIÓN		
ACTIVO CRÍTICO	Nombre del activo	
VERSION	En caso de existir	
REQUERIMIENTOS	**HARDWARE**	**SOFTWARE**
DESCRIPCIÓN	Funcionalidad del activo	
TITULAR DEL ACTIVO	Responsable	
REQUERIMIENTOS DE SEGURIDAD	**DISPONIBILIDAD**	
	INTEGRIDAD	
	CONFIDENCIALIDAD	
AMENAZAS	Detallar todas las posibles amenazas	
FECHA		
REALIZADO POR		

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

3.2.2 Detección de vulnerabilidades.

Se utilizó Openvas, por ser un software potente que además de detectar fallas la red proporciona información de los escaneos de manera estadística y detallada de las vulnerabilidades encontradas en toda la red SDN, basándose en la dirección IP, los puertos abiertos de cada activo informático, figura 4-3. En el anexo I, se muestran todas las fallas del: controlador, servidores, open vswitch y usuarios.

Cuando se escaneó al controlador en un tiempo de 13 minutos y 40 segundos, con dirección IP 192.168.1.5, se localizaron vulnerabilidades a nivel medio, con un 80% de calidad en la detección, que afectan al puerto 8181(comunicación HTTP de ODL). El efecto causado en este dispositivo es que, si se logra ingresar, el atacante puede obtener información sensible de toda la red SDN. También se encontraron debilidades en los DIRB (NASL wrapper), las cuales son herramientas basadas en diccionarios, que buscan fallas existentes u ocultas, para acceder al servidor web, usando ataques de fuerza bruta.

En cambio, cuando se analizó al dispositivo open vswitch, con dirección IP 192.168.1.10, en un periodo de 8 minutos y 25 segundos, se encontraron fallas en ICMP y CPE inventory, con un 80% de QoD o calidad de detección, ese valor indica se debe por ser productos patentados. Por otro lado, en los servicios de red, se observó una incidencia hacia al puerto 80 es decir al servidor HTTP con 95% seguido del puerto 22 o SSH con 95%, 21 FTP con 80%, BIND e ICMP. Y en los usuarios con un 98% se observan amenazas: DIRB, SMB, SSL/TLS y con 80% en ICMP.

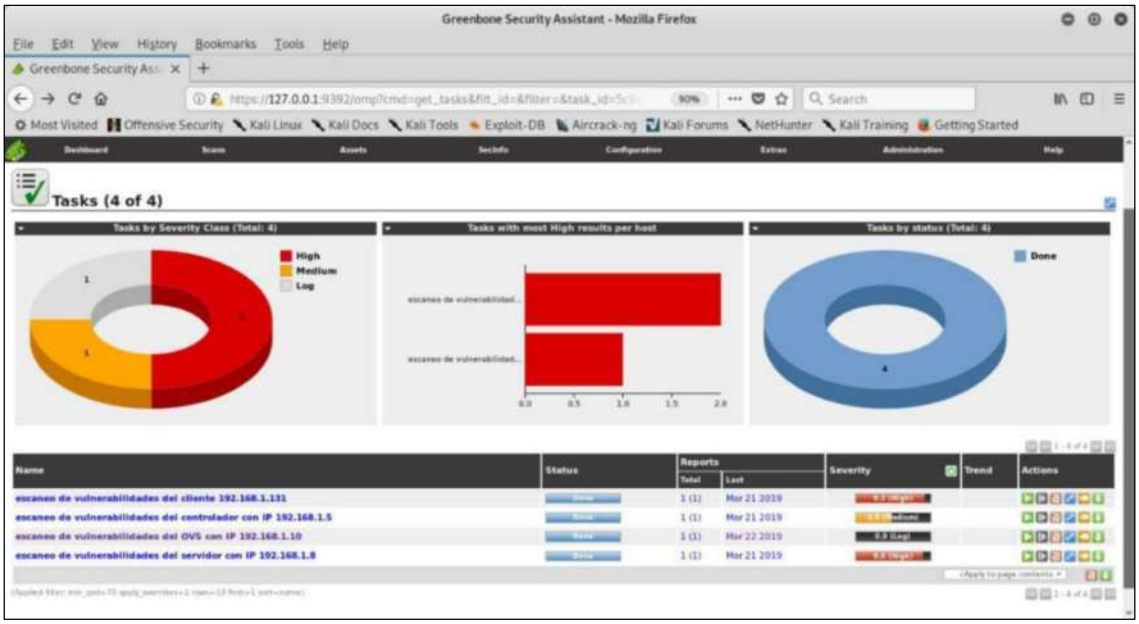


Figura 4-3: Escaneo de vulnerabilidades SDN con OpenVas.

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

En la tabla 2-3 se detallan, las principales vulnerabilidades encontradas en toda la infraestructura de red SDN, sus características y al dispositivo que afectan.

Tabla 2-3: Vulnerabilidades encontradas por Openvas.

VULNERABILIDADES DETECTADAS	CARACTERISTICAS	DISPOSITIVOS AFECTADO
HTTP Server type and version	Detecta el tipo y versión del servidor HTTP.	Controlador
DIRB (NASL Wraper)	Utiliza DIRB para encontrar directorios y archivos en la página web.	Controlador, servidores y clientes
ICMP Timestamp Detection	Esta vulnerabilidad permite sondear equipos en tiempo real un acceso remoto	Controlador, switch, servidores, y clientes
SSH Protocol Versions Supported	Descifra y alterar el tráfico.	Servidores
FTP Banner Detection	Averiguar el sistema instalado en el servidor FTP.	Servidores
Determine which version of BIND name daemon is running	El servidor DNS tiene algunas falencias en sus versiones.	Servidores
HTTP brute force logins with default Credentials reporting	Sirve para iniciar sesión remota con credenciales por defecto.	Servidores
SSL/TLS Certificate- Self-Signed Certificate Detection	Las comunicaciones no tienen cifrado de datos.	Clientes
SMB NativeLanMan	El atacante puede ejecutar un DoS, usando el protocolo SMB.	Clientes
CPE inventory	Ayudan a descubrir la versión del software, hardware y sistema operativo del dispositivo.	Switch

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

Para determinar el grado de ocurrencia de las vulnerabilidades detectadas por el software Openvas a la red SDN, se utilizó la tabla 10-2 de la sección 2.4.2. Como se observa en el gráfico 1-3, las vulnerabilidades con mayor incidencia son hacia los protocolos SMB en un 67% Y HTTP en un 33%, lo que indica que se debe de asegurar las credenciales del protocolo HTTP.

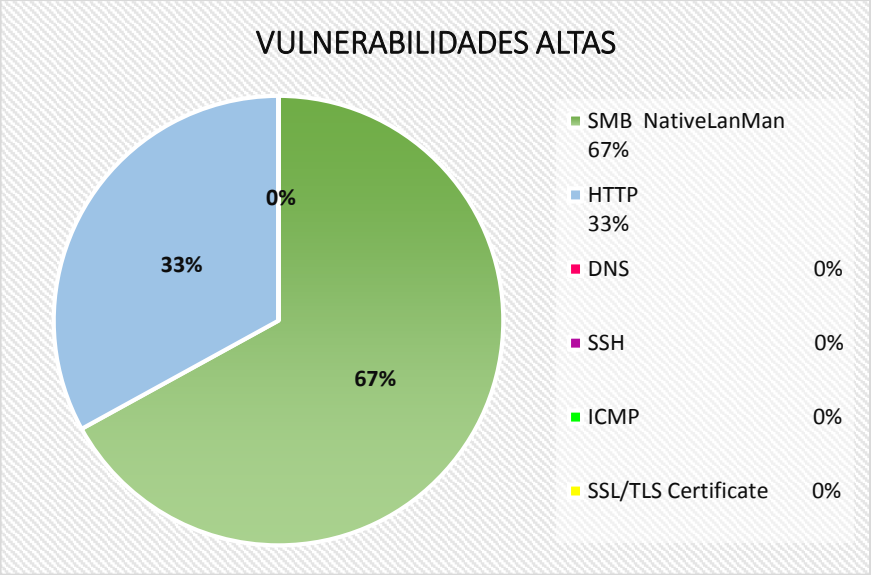


Gráfico 1-3: Vulnerabilidades altas detectadas con OpenVas.

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

En las vulnerabilidades de medio impacto, gráfico 2-3, el protocolo más afectado resultó ser HTTP con un 31%, seguido con un 23% de SSL, mientras que SSH y FTP arrojaron resultados de 15%, también se visualizaron amenazas a TCP y DCE en un 8%.

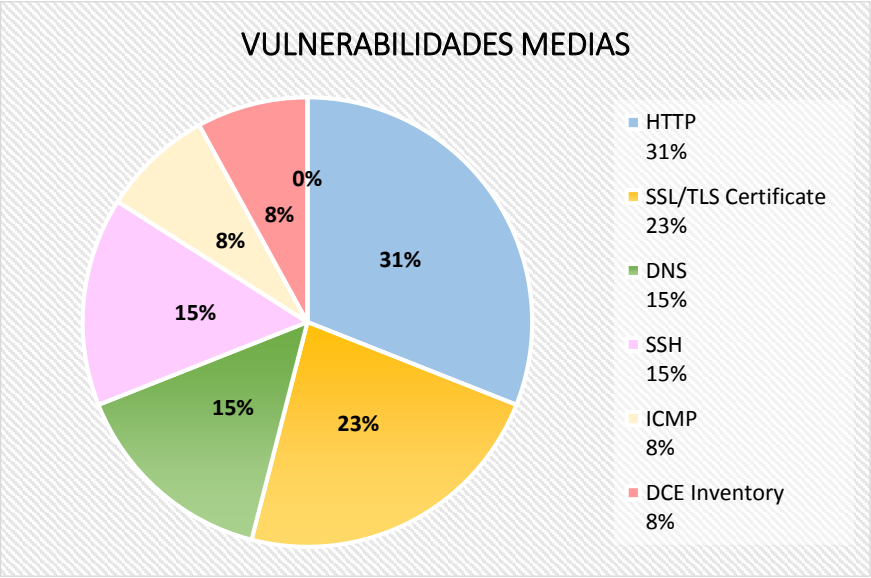


Gráfico 2-3: Vulnerabilidades de nivel medio detectadas con OpenVas.

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

En el gráfico 3-3, se observan las vulnerabilidades bajas, que afectan al protocolo ICMP en un 100%, con esas fallas el atacante de red puede ingresar a cualquier servicio y robarse credenciales, información, etc.

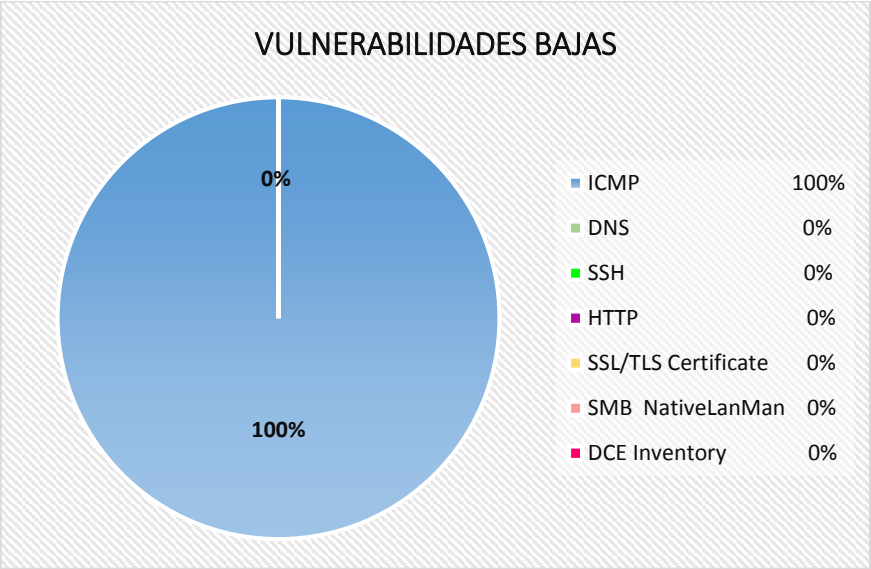


Gráfico 3-3: Vulnerabilidades bajas detectadas con OpenVas.

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

Después de que se realizó ese análisis, en la tabla 4-3, se clasificaron solamente a las fallas que afectan a la disponibilidad, además se evidencia una lista de los principales ataques de denegación de servicio que pueden ejecutarse para cada vulnerabilidad.

Tabla 3-3: Vulnerabilidades detectadas para la ejecución de ataques DoS.

VULNERABILIDAD	ACTIVO	ATAQUE
HTTP server type and versión	Controlador	Ataque HTTP
HTTP brute force logins with default credentials reporting	Servidores	Ataque HTTP
Determine which version of BIND name daemon is running	Servidores	Ataque DNS
ICMP Timestamp Detection	Controlador, switch, servidores, y clientes	Ataque DHCP

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

3.2.3 Ataques a la infraestructura SDN.

Con el objetivo de explotar las vulnerabilidades detectadas se ejecutaron a ataques de denegación de servicio hacia la infraestructura de red SDN, mismos que fueron generados desde Kali Linux, usando las herramientas Armitage y Metasploit. Las amenazas DoS fueron seleccionados en base a una encuesta realizada al departamento DTIC-ESPOCH, donde indicaron que los principales ataques son hacia los servicios DHCP, HTTP y DNS. En los anexos siguientes, se evidencia una guía con todos los pasos de la ejecución de los ataques.

Anexo I: Ataque HTTP
Anexo I: ataque DHCP
Anexo I: Ataque DNS

También se analizó el comportamiento de la red frente a este tipo de amenazas utilizando dos indicadores de disponibilidad: ancho de banda y latencia.

3.2.3.1 Indicador I: impacto ocasionado en el ancho de banda.

El ancho de banda es la cantidad de información que se puede enviar a través de una conexión de red, durante cierto tiempo (generalmente medido en 1 segundo), por lo que se deduce que a mayor número de paquetes a transmitir se necesitará de un mayor ancho de banda para evitar que el sistema colapse.

En base a lo descrito anteriormente, sea en redes tradicionales o redes definidas por software, para saturar el ancho de banda, se envían grandes cantidades de trafico mediante los ataques de denegación de servicio, provocando que la red no tenga suficiente capacidad de responder rápidamente a las solicitudes. Para el desarrollo de la esta sección, se consideró la información detallada en el apartado 2.4.2 concerniente al indicador 1.

En la tabla 4-3, se muestran las vulnerabilidades explotadas. Para medir el impacto causado al ancho de banda, se utiliza la fórmula de la efectividad, descrita en el apartado 2.4.2, en la cual se debe establecer la cantidad de paquetes antes y después de ejecutarse las amenazas, sobre la cantidad total de tramas máximas soportados por el sistema, para lo cual se utilizó la herramienta Wireshark. Para determinar el total de paquetes soportados por el sistema, se realizaron varias pruebas de laboratorio, es decir a la red se inundó con suficiente tráfico que colapsó la misma, donde se puedo comprobar que el sistema puede soportar un máximo de 1200 paquetes por segundo.

Tabla 4-3: Impacto de DoS en el ancho de banda.

ATAQUE	TRAMAS DETECTADAS		TOTAL TRAMAS	EFECTIVIDAD	IMPACTO
	ANTES	DESPUÈS			
Ataque HTTP controlador	281	716	1200	23% - 59%	Bajo - Medio
Ataque HTTP servidores	408	859	1200	34% - 71%	Bajo - Alto
Ataque DNS	62	80	1200	0,5% - 6%	Bajo - Baja
Ataque DHCP	411	995	1200	34% - 82%	Bajo - Alto

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

Conforme a la tabla 4-3, se verifica que antes de aplicar los ataques DoS a la infraestructura SDN, las tramas detectadas. En el controlador y servidores el impacto es bajo, con valores de efectividad del 23%, 34%, 0.5%, y 34% respectivamente.

Mientras que después de ejecutar los ataques denegación de servicio se evidencia que el consumo de ancho de banda se duplica para todos los casos. En el controlador el impacto fue medio con un 59% de efectividad, en servidores fue alto con 71%, en el servidor DNS fue bajo con 6% y en el servidor DHCP fue alto con 82% de efectividad.

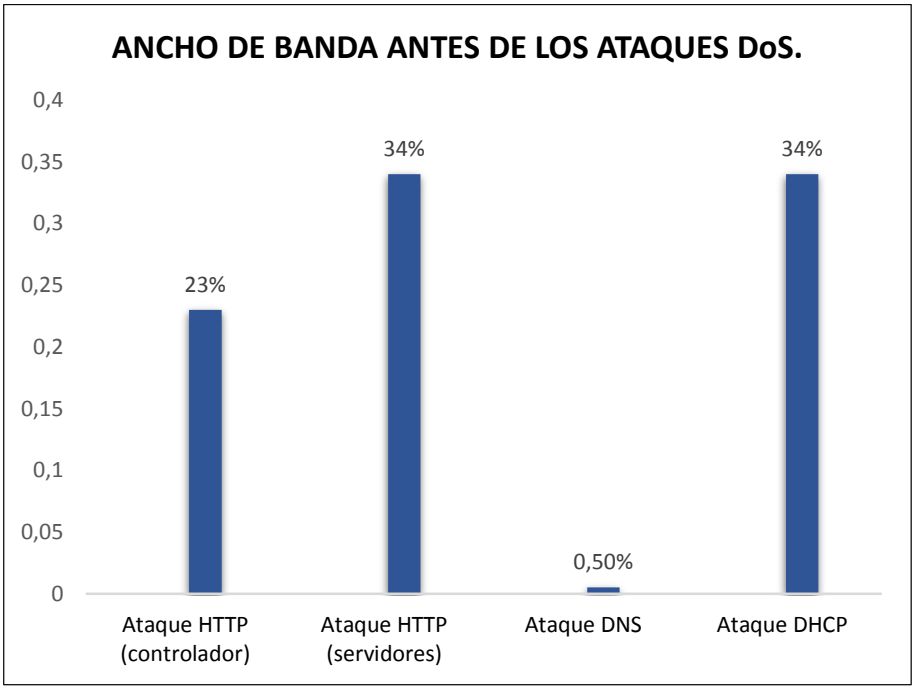


Gráfico 4-3: Mediciones de ancho de banda antes de los ataques DoS.

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

El grafico 4-3 muestra que los valores no superan el 34% de efectividad antes de la ejecución de los ataques de denegación de servicio, por lo que según especificaciones del CERT, la red no está siendo blanco de amenazas que comprometan la disponibilidad. Mientras que en el grafico 5-3, se visualiza un consumo excesivo de ancho de banda que superan el 70%, de efectividad, y basándose en información del CERT la red está en alto riesgo.

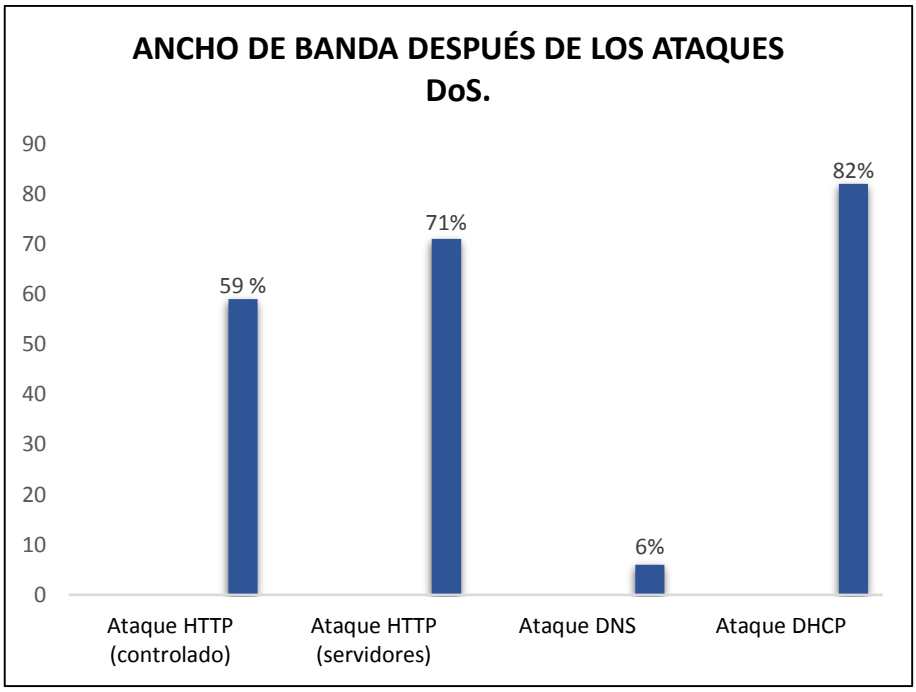


Gráfico 5-3: Mediciones de ancho de banda después de los ataques DoS.

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

3.2.3.2 Indicado II: impacto ocasionado en los tiempos de respuesta.

La latencia o tiempos de respuesta es considerado como el lapso en que se tardan los paquetes en llegar a su destino, sea en redes tradicionales, donde estos intervalos son mayores, o en redes SDN con tiempos de respuesta muchos más bajos debido a su arquitectura, generalmente medidos en milisegundos.

En esta sección, este indicador se calcula en dos instantes, el primero, cuando la red se encuentra trabajando con normalidad, y el segundo al aplicar los ataques de denegación de servicio, para poder extraer resultados de las mediciones se utilizó la herramienta PING.

En la tabla 5-3, se analiza el comportamiento de la red SDN con respecto a los tiempos de respuesta o latencia existente entre puntos de conexión, para ello se enviaron diez solicitudes

ICMP, luego en el Anexo K, se tabularon los datos obtenidos, para calcular un valor promedio de latencia antes y después de generar cada uno de los ataques DoS, en base a la tabla 12-3.

Tabla 5-3: Mediciones de latencia en la red SDN.

ATAQUE	LATENCIA		CALIDAD DE CONEXIÓN
	ANTES	DESPUES	
Ataque HTTP controlador	0,687ms	4,009 ms	Alto - medio
Ataque HTTP servidores	0,646 ms	4,779 ms	Alto – medio
Ataque DNS	0,692 ms	2,064 ms	Alto – bajo
Ataque DHCP	0,687 ms	3, 592 ms	Alto – medio

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

En el grafico 5-3, se observa que los valores de latencia obtenidos antes de la ejecución de los ataques y de acuerdo a la tabla 7-3, se deduce que toda la red está trabajando de manera óptima, puesto que los valores de latencia son menores a 1,5 milisegundos.

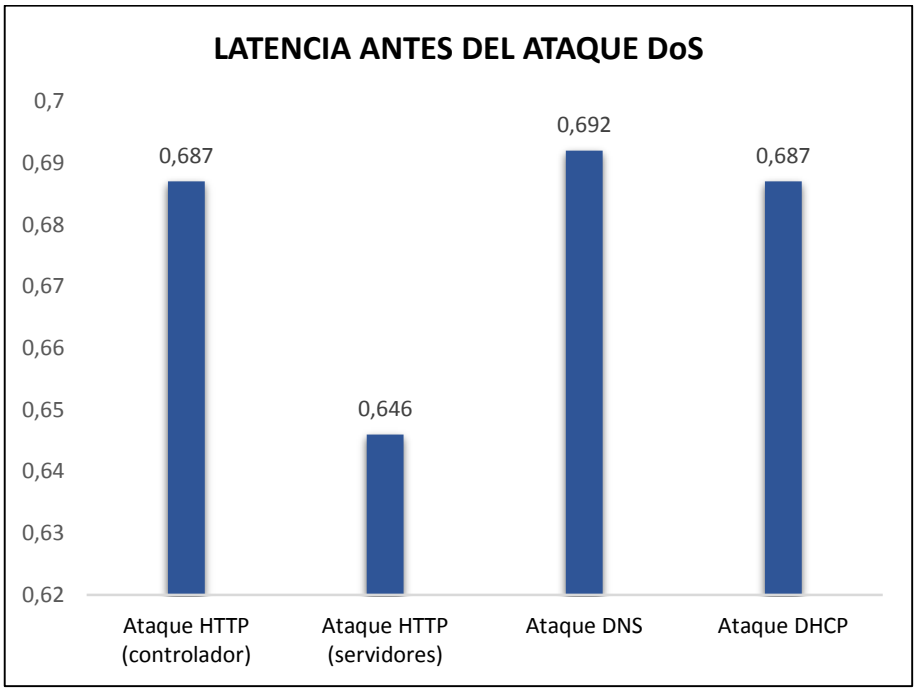


Gráfico 6-3: Mediciones de latencia antes de los ataque DoS.

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

Por otra parte, en el grafico 6-3, al aplicar los diferentes ataques, la latencia aumenta con relación al caso anterior donde no existe ningún ataque. Al aplicar un ataque HTTP al controlador, el tiempo en establecerse la comunicación sube a 4,779 milisegundos. Se debe destacar que según

información del CERT si los tiempos de respuesta están en el rango de 1,5 milisegundos y 5 milisegundos significa que está trabajando con sobre-procesamiento.

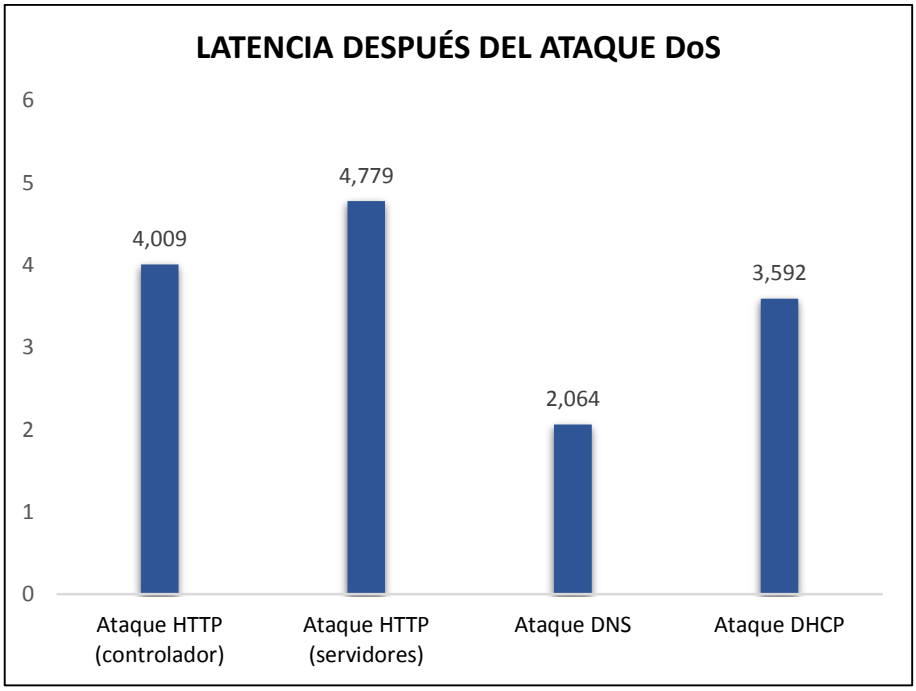


Gráfico 7-3: Mediciones de latencia después de los ataque DoS.

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

3.2.3.3 Análisis de tráfico ante ataques DoS.

En esta sección se realizó el análisis de la red SDN, cuando estaba sometida a ataques de denegación de servicio, con la ayuda de la herramienta Wireshark, se puedo visualizar la cantidad de tráfico que circula por la infraestructura de red.

La figura 5-3, corresponde a las mediciones captadas en el controlador, donde en el eje horizontal se muestra una línea de tiempo de 500 segundos de análisis antes de aplicar ataques DoS, mientras que en el eje vertical indica la cantidad de paquetes generados en cierto tiempo. Allí se comprobó que, entre los 0 segundos y 400 segundos, la red trabaja con total normalidad, ya que la red envía de 5 a 35 paquetes por segundo, y a partir de los 400 segundos a 500 segundos la red empezó con un poco de carga, evidenciándose un envió de 65 paquetes por segundo.

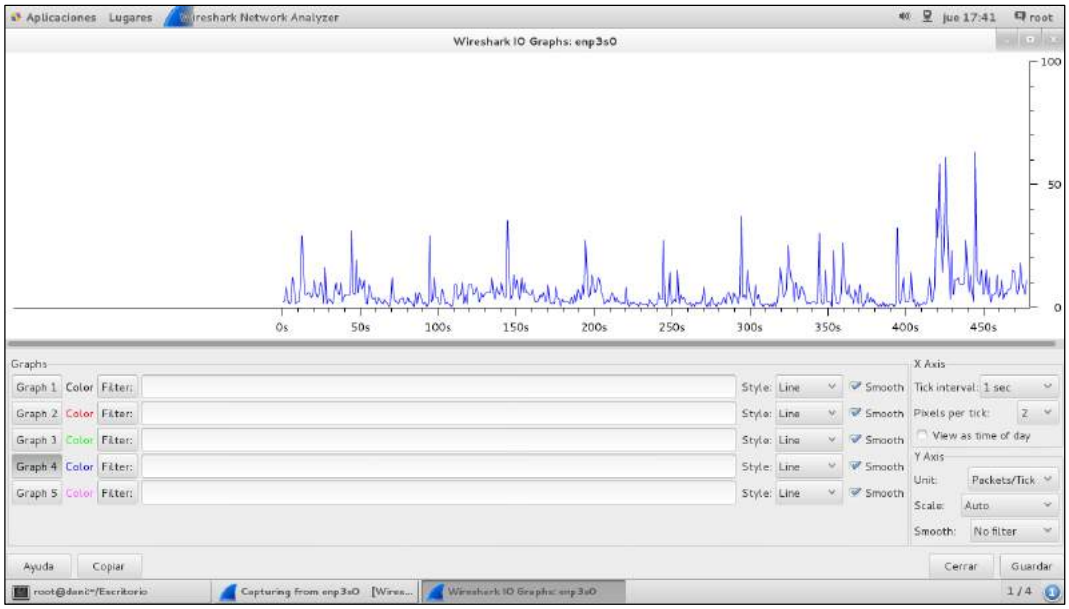


Figura 5-3: Análisis con Wireshare antes de realizar ataques DoS.

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

En tanto a la figura 6-3, corresponde a mediciones realizadas cuando ya se aplicaron las amenazas, donde se observó un incremento drástico de tráfico al enviar paquetes. En el transcurso de los 550 segundos hasta los 710 segundos, es decir alrededor de 300 segundos, se enviaban hasta 900 paquetes por segundos.

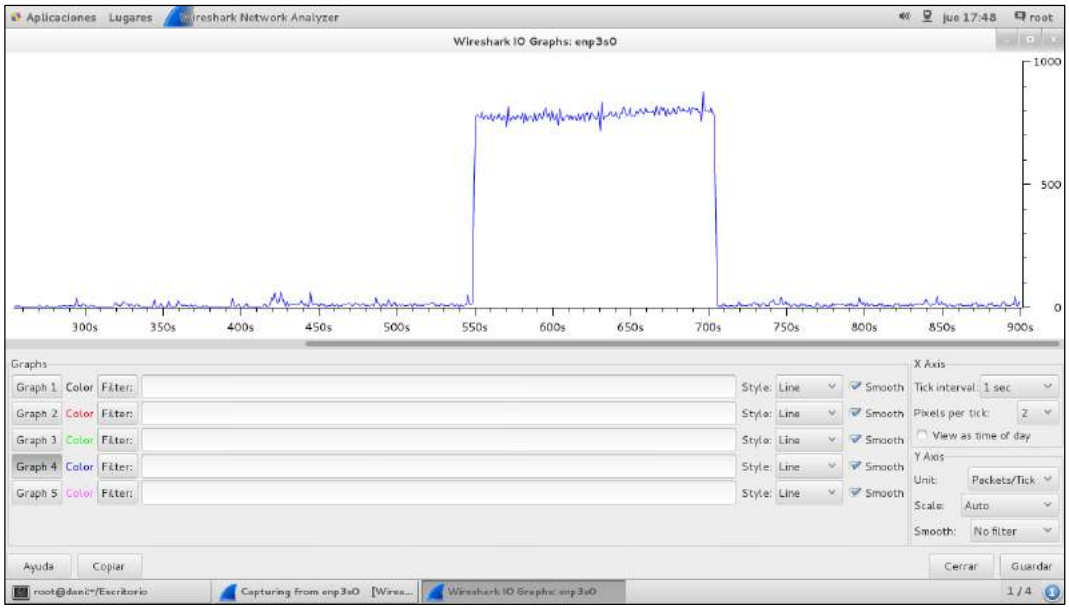


Figura 6-3: Análisis con Wireshark después de realizar ataques DoS.

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

En el Anexo L, se muestran dos tablas, que contiene el número de paquetes generados en 60 segundos. La primera tabla, corresponde a un análisis antes de la generación de ataques de

denegación de servicio, en el intervalo de tiempo de 0 a 60 segundos, se evidenció un envió de 678 paquetes, es decir que en promedio se transmitió aproximadamente 12 paquetes cada segundo, mientras que, la segunda tabla, fue tomada cuando se ejecutó la amenaza entre los 550 y 610 segundos se detectó un incremento a 4668 paquetes, equivalente a 778 por segundo.

3.2.3.4 Prueba de hipótesis de la red

Para comprobar que el tráfico de red, aumenta después de los ataques de denegación de servicio se realizó una prueba de hipótesis, para lo cual se trabajó con un nivel de significancia del 5%. Para determinar si se va a realizar una prueba paramétrica o no paramétrica, se empezó con un estudio de normalidad, entre los resultados obtenidos, bajo el mismo escenario y las mismas condiciones.

La prueba se desarrolló utilizando, el programa estadístico SPSS 21, debido a que el número de datos es mayor a 30, se aplicó el estadístico de Kolmogorov Smirnov, obteniendo una probabilidad de 0,00018 (para el antes) y de 0,015 (para el después), dichas probabilidades son, menores al nivel de significancia, lo que indica que los datos que se obtuvieron no siguieron una distribución normal, lo que conlleva a realizar una prueba no paramétrica. En el Anexo M, se evidencia la prueba mencionada.

La prueba no paramétrica que se utilizó fue la de Wilcoxon, que es equivalente a la prueba paramétrica t-student, para muestras relacionadas. En la figura 7-3, se evidencia que la probabilidad obtenida en la prueba fue de 9,23E-43, que es menor al valor de significancia, lo que permite concluir que existen diferencias entre las medianas de los resultados de paquetes generados, antes y después de los ataques de denegación de servicio.

Estadísticos de contraste[a]

	Después - Antes
Z	-13,707[b]
Sig. asintót. (bilateral)	9,23E-43

a. Prueba de los rangos con signo de Wilcoxon

b. Basado en los rangos negativos.

Figura 7-3: Prueba estadística de los ataques DoS.

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

3.2.4 Elaboración del plan de contingencia.

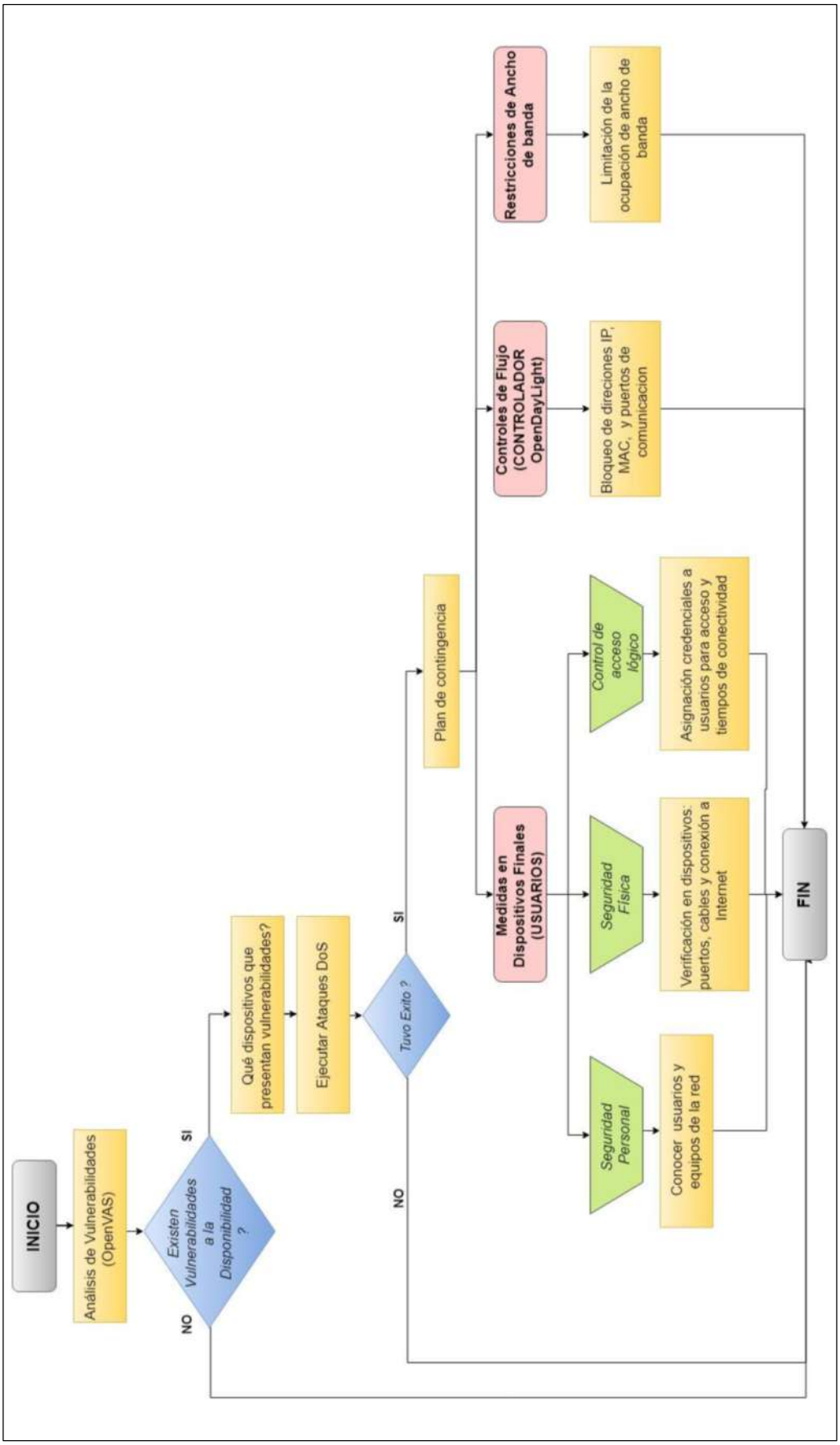


Figura 8-3: Diagrama de flujos del plan de contingencia del proyecto.

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

Los planes de contingencia son muy necesario e importante, tanto en redes tradicionales o redes definidas por software, para contrarrestar o mitigar cualquier vulnerabilidad detectada y así evitar futuros daños y perjuicios a futuro.

Para el desarrollo del presente trabajo investigativo, se elaboró una guía de buenas prácticas, con referencias y especificaciones del estándar ISO 270001 adaptables a las SDN, con una serie de métodos que pueden ser aplicados por cualquier administrador de red, para mantener a salvo su infraestructura de cualesquier ataques. En el Anexo L, se desarrolló, la etapa tres de la metodología OCTAVE. En la figura 7-3, se muestra una breve descripción del plan de contingencia ante vulnerabilidades de ataques de DoS, elaborado en un diagrama de flujo del proyecto.

3.2.4.1 Propuesta 1: Medidas en los dispositivos finales.

Los dispositivos finales o host funcionan como interfaz de comunicación entre usuarios y servicios de la red, transmitiendo datos de origen a destino, por citar algunos ejemplos: portátiles, computadoras de escritorios, teléfonos inteligentes, teléfonos IP, impresoras, entre otros, por ello es necesario tomar precauciones para garantizar la seguridad y por supuesto su correcto funcionamiento y disponibilidad.

Para proteger a los dispositivos finales es necesario, aplicar ciertas medidas preventivas como, actualizaciones del software instalado, el firmware, etc., con la finalidad de proteger al sistema de vulnerabilidades que puedan afectar a su seguridad.

- Seguridad Personal:

Un aspecto indispensable dentro de alguna empresa o compañía tiene que ver con los usuarios que utilizan un bien o servicio informáticos, como primer punto se debe conocer a cada uno de los involucrados que tengan acceso al equipo, luego determinar que funcionalidades tienen referente a dicho dispositivo.

- Seguridad Física:

Luego de verificar la seguridad personal, se verificó las conexiones de dispositivos, como, puertos USBs, cables de conexión, todo esto con el fin de precautelar la fuga de información y también la conexión internet.

- Controles de Accesos Lógicos:

Se optó por asignar credenciales exclusivas para cada usuario, así como de una contraseña este paso es esencial para acceder a la información o acceder a la infraestructura de red.

3.2.4.2 Propuesta 2: Generación de Flujos.

Las redes SDN están basadas en flujos, que son reglas generadas por el administrador de la infraestructura, con el objetivo de establecer medidas de seguridad o mejorar la capacidad del sistema basándose en políticas. Todos estos flujos son establecidos de manera centralizada en el controlador y los switches simplemente se encargan de reenviar esas órdenes a los demás dispositivos.

En la elaboración del presente proyecto, para la generación de flujos en el controlador Opendaylight, se utilizó la interfaz gráfica YANG, una herramienta fácil y rápida que ayuda a crear reglas de acuerdo a la restricción deseada que permita, rechace o a su vez bloquee las conexiones de ciertos objetivos, mediante direcciones MAC, IP, puerto de comunicación, puertos de enlace.

En el apartado 3.1.2, se detallan los pasos de la creación de políticas o reglas de flujo en el controlador.

- Flujos de bloqueo de puertos de comunicación.

Los puertos de comunicación se establecen en la capa de aplicación, ya sea del modelo OSI o TCP/IP, los mismos que permiten establecer conexión con otros dispositivos de la red para algún fin específico, existen un gran número de puertos para identificar las conexiones, entre los más conocidos están: HTTP (80), HTTPS (443), FTP (21), SSH (22), DNS (53), etc., estos servicios se encuentran prácticamente activos en cualquier tipo de red de datos por lo que en una red SDN no es la excepción.

Se detectaron puertos abiertos en el escenario implementado, utilizando la herramienta Openvas, y después se procedió a cerrar o bloquear la conexión que representaban peligro para la red. En el caso del controlador se descubrió, el puerto abierto número 8181 en TCP usado para configuración y operación del mismo.

- Flujo de bloqueo de direcciones MAC

Las direcciones MAC son códigos únicos de 48 bits, que identifican a un dispositivo conectado a la red. En el Anexo K, se detalla brevemente la configuración de filtrado MAC, desde la dirección de origen 8E:3F:6E:83:56:22 con dirección de destino B6:4B:78:F2:2A:B5, de modo que al establecerse esta política no se pudo existió comunicación.

- Flujo de bloqueo direcciones IP.

Se bloqueó ciertas direcciones IP de usuarios, para que no accedan a servicios o departamentos de la institución.

3.2.4.3 Propuesta 3: Restricciones de ancho de banda

Este recurso muy valioso en la red, para establecer una comunicación fiable sin interrupciones.se creó una regla para establecer el tamaño del ancho de banda de 10 Mbps de velocidad.

3.2.5 Evidencia de la mitigación de vulnerabilidades.

Luego de haber implementado el plan de contingencia en la red SDN, en la figura 8-3 se evidenció que las vulnerabilidades disminuyeron.

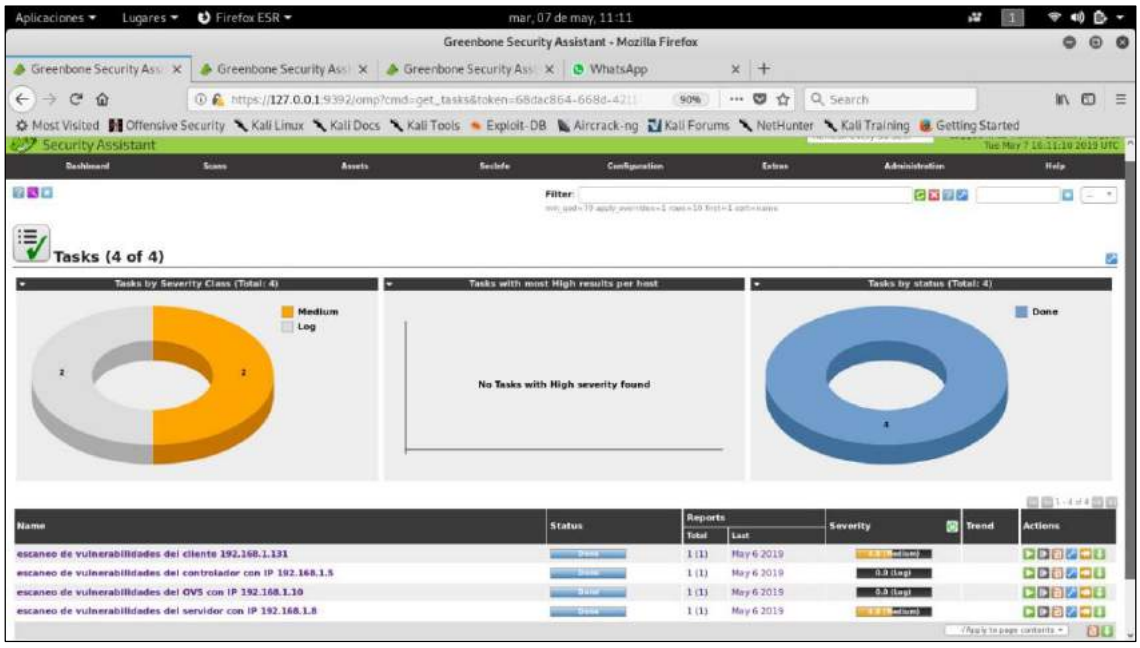


Figura 9-3: Vulnerabilidades detectadas después del plan de contingencia.

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

En la figura 10-3, los cliente y servidores se evidenció una gravedad de nivel medio con valores de 4.8, a comparación de la figura 4-3 que indicaba una severidad alta con valores comprendidos de 9.0 hasta 9.3 de afectación.

En la tabla 6-3, se presenta de manera resumida el nivel de gravedad e impacto de las vulnerabilidades antes y después de la ejecución de la guía de buenas prácticas a cada uno de los activos informáticos.

Tabla 6-3: Impacto de vulnerabilidades en la infraestructura de red

Activo	**Gravedad**		**Impacto**	
	Antes	**Después**	**Antes**	**Después**
Controlador	4,8	0	Medio	Bajo
Open vswitch	0	0	Bajo	Bajo
Servicios	9,0	4,8	Alto	Medio
Cliente	9,3	4,8	Alto	Medio

Realizado por: TOAINGA Daniela, PEÑA Daniel, 2019.

CONCLUSIONES

- El estudio realizado de las vulnerabilidades en Redes definidas por Software muestra varios perjuicios tanto a nivel del protocolo de comunicación OPENFLOW, debido a que carece de seguridad, como en el controlador, en el despliegue de flujos o reglas.

- Para el análisis de vulnerabilidades se utilizó la metodología OCTAVE y para el escaneo de la red Openvas, donde se comprobó, que controlador y servidores son vulnerables por protocolo HTTP (puerto 8181y 80 respectivamente), con un 33% de ocurrencia, Otra falla es la versión de BIND de DNS con 15% y por último en los tiempos de detección de solicitudes de marca de tiempo ICMP con un 100% de riesgo para la seguridad de la red.

- El escenario implementado para las pruebas de los ataques DoS muestra una gran eficiencia debido a que muestra una topología tipo estrella extendido en el cual, el administrador red puede separar puntos de falla en caso de afectación en algún nodo.

- El impacto causado al ancho de banda antes de los ataques no superaba el 34% de efectividad, lo que significa que no existe ningún riesgo, pero después de la amenaza superaban el 70%, representando un alto peligro. En el caso de análisis de latencia los valores antes de la ejecución de ataques son menores a 1,5 milisegundos, mientras que después de la amenaza, tarda un tiempo de 4,779 milisegundos lo cual nos da un indicador que el servicio tiene sobre procesamiento.

- La prueba no paramétrica que se utilizó fue la de Wilcoxon, que es equivalente a la prueba paramétrica t-student, para muestras relacionadas. La probabilidad obtenida en la prueba fue de 9,23E-43, que es menor al valor de significancia, lo que permite concluir que existen diferencias entre las medianas de los resultados de paquetes generados, antes y después de los ataques de denegación de servicio.

- Al implementarse el plan de contingencia en el escenario se evidenció una disminución a la mitad de las vulnerabilidades en todos los activos de red. En servicios y clientes, de valores de 9,0 y 9,3 de gravedad, se redujeron a 4,8 que indica un impacto medio y en el controlador se comprobó cero vulnerabilidades.

RECOMENDACIONES

- Para prevenir ataques directamente al controlador Opendaylight, se recomienda activar la opción defens4 que por defecto viene con la herramienta.

- Se recomienda implementar un escenario tipo estrella extendida debido a su gran escalamiento y seguridad, ya que permiten separar puntos de fallas y la red puede seguir trabajando con normalidad

- Se deben mantener siempre actualizados al controlador y los servidores, para obtener parches de seguridad y así prevenir cualquier vulnerabilidad y un posible ataque.

- Se recomienda utilizar herramientas de análisis de tráfico y softwares de escaneo de vulnerabilidades, para verificar el estado de la red y de dispositivos.

- También se sugiere estudiar a fondo al controlador Opendaylight ya que tiene un gran número de aplicaciones tanto de seguridad como de despliegue de tráfico para el ámbito de las telecomunicaciones y redes.

BIBLIOGRAFÍA

ABLIZ, M., *Internet Denial of Service Attacks and Defense Mechanisms* [en línea]. 2011. S.l.: Department of Computer Science. [Consulta: 17 noviembre 2018]. 2011. ISBN 0710077483. Disponible en: https://people.cs.pitt.edu/~mehmud/docs/abliz11-TR-11-178.pdf.

ACURCIO, C., *The Comparison of Network Simulators for Sdn* [en línea]. 2008. S.l.: POLTAVA NATIONAL TECHNICAL YURI KONDRATYUK UNIVERSITY. [Consulta: 6 diciembre 2018]. 2008. Disponible en: http://journals.pntu.edu.ua/mist/article/view/571.

AGUIRRE, J., *Sistema de detección de ataques DDoS en Tor* [en línea]. Tesis Pregrado. S.l.: UNIVERSIDAD COMPLUTENSE DE MADRID. 2015. [Consulta: 19 noviembre 2018]. Disponible en: http://eprints.sim.ucm.es/33138/.

ALVAREZ, M. et al, *Tutorial de FTP* [en línea]. 2019. S.l.: Desarrollador Web. [Consulta: 24 marzo 2019]. 2019. Disponible en: http://www.geocities.ws/hpotau/manual-tutorial-ftp.pdf.

ÁLVAREZ, R., *Estudio de las Redes Definidas por Software mediante el desarrollo de escenarios virtuales basados e el controlador OpenDaylight* [en línea]. Tesis Postgrado. S.l.: UNIVERSIDAD POLITÉCNICA DE MADRID. 2015. [Consulta: 16 diciembre 2018]. Disponible en: http://oa.upm.es/42968/1/TFM_RAUL_ALVAREZ_PINILLA.pdf.

AZODOLMOLKY, S., *Software Defined Networking with OpenFlow* [en línea]. Primera. S.l.: Packt Publishing Ltd. 2013. [Consulta: 15 noviembre 2018]. ISBN 0816519226. Disponible en: http://medcontent.metapress.com/index/A65RM03P4874243N.pdf.

BALSA, C., *Emulación de Redes Cisco con GNS3* [en línea]. Tesis Postgrado. S.l.: UNIVERSIDAD DE ALMERÍA. 2016. [Consulta: 26 febrero 2019]. Disponible en: http://www.adminso.es/recursos/Proyectos/PFM/2013_14/PFM_Aprende_GNS/Proyecto_Aprende_a_emular_redes_cisco_con_GNS3.pdf.

BIG SWITCH NETWORKS, *The Open SDN Architecture* [en línea]. 2013. S.l.: Big Switch Networks. [Consulta: 3 mayo 2018]. 2013. Disponible en: www.bigswitch.com.

BRAUN, W. y MENTH, M., *Software-Defined Networking Using OpenFlow*: *Protocols, Applications and Architectural Design Choices. Future Internet* [en línea], 2014. p. 35. [Consulta:

21 noviembre 2018]. ISSN 1999-5903. DOI 10.3390/fi6020302. Disponible en: http://www.mdpi.com/1999-5903/6/2/302/.

CARDOSO, L., *Despliegue de un testbed de redes definidas por software para la gestión de recursos de red en un CPD* [en línea]. Tesis Postgrado. S.l.: UNIVERSIDAD DE EXTREMADURA. 2015. [Consulta: 12 noviembre 2018]. Disponible en: http://dehesa.unex.es/bitstream/handle/10662/4417/TFMUEX_2016_Amarilla_Cardoso.pdf?sequence=1&isAllowed=y.

CHÁVEZ, J., *Simulación y análisis de mecanismos de defensa ante los ataques de denegación de servicio (DoS) en redes de área local convergentes* [en línea]. Tesis Pregrado. S.l.: ESCUELA SUPERIOR POLITÉCNICA NACIONAL. 2011. [Consulta: 17 noviembre 2018]. Disponible en: http://bibdigital.epn.edu.ec/bitstream/15000/10439/3/CD-6187.pdf.

CITRIX, *SDN 101 : Introducción a Software Defined Networking* [en línea]. 2014. S.l.: Citrix. [Consulta: 15 noviembre 2018]. 2014. Disponible en: https://www.citrix.com/content/dam/citrix/en_us/documents/oth/sdn-101-an-introduction-to-software-defined-networking-es.pdf.

CLOUDFLARE, *What is a DDoS Botnet? Cloudflars, Inc* [en línea]. 2019. [Consulta: 18 noviembre 2018]. Disponible en: https://www.cloudflare.com/learning/ddos/what-is-a-ddos-botnet/.

CONTRERAS, C., *Implementación de un Openflow controller para el manejo de Openflow switches* [en línea]. Tesis Pregrado. S.l.: PONTIFICIA UNIVERSIDAD JAVERIANA. 2014. Disponible en: https://www.cloudflare.com/learning/ddos/what-is-a-ddos-botnet/.

COSÍO, E., *Modelado de una arquitectura de red definida por software (SDN) para el aprovisionamiento de recursos utilizando Cross- Layer Design (CLD)* [en línea]. Tesis Postgrado. S.l.: CENTRO DE INVESTIGACIÓN CIENTÍFICA Y DE EDUCACIÓN SUPERIOR DE ENSENADA, BAJA CALIFORNIA. 2017. [Consulta: 3 mayo 2018]. Disponible en: https://cicese.repositorioinstitucional.mx/jspui/bitstream/1007/898/1/Formato Tesis - Ernesto Cosio 09-02-2017 biblioteca.pdf.

CRAIG SPROSTS, *What CSPs Can Learn from the Latest DDoS Attacks - The Akamai Blog. Akamai* [en línea]. 2016. [Consulta: 4 enero 2019]. Disponible en: https://blogs.akamai.com/2016/10/what-csps-can-learn-from-the-latest-ddos-attacks.html.

ESTÉVEZ, D., *Estudio Para El Desarrollo De Un Modelo De Gestión De Riesgos Y Seguridad De La Información Para Instituciones Militares* [en línea]. Tesis Postgrado. S.l.: ESCUELA POLITÉCNICA NACIONAL. 2014. Disponible en: http://bibdigital.epn.edu.ec/bitstream/15000/8642/1/CD-5812.pdf.

ESTINET, EstiNet. *EstiNet Technologies Inc* [en línea]. 2019. [Consulta: 12 noviembre 2018]. Disponible en: http://www.estinet.com/ns/?page_id=21140.

GNS3, *Getting Started with GNS3. GNS3* [en línea]. 2018. [Consulta: 12 noviembre 2018]. Disponible en: https://docs.gns3.com/1PvtRW5eAb8RJZ11maEYD9_aLY8kkdhgaMB0wPCz8a38/index.html?fbclid=IwAR3DdIJwgpU3ZiNvWisRhWIg9NxZZviuAy6Urso6irVeCCU_QZ51mPPUs2I.

GORANSSON, P. et al, *Software Defined Networks* [en línea]. San Francisco- Estados Unidos: ELSEVIER SCIENCE & TECHNOLOGY. 2014. [Consulta: 20 noviembre 2018]. ISBN 012416675X. Disponible en: http://bibliotecas.espoch.edu.ec/bdatos.html.

GUTIERREZ, C., *Tutorial de Wireshark* [en línea]. 2012. S.l.: Facultad de Ciencias. [Consulta: 26 marzo 2019]. 2012. Disponible en: https://es.scribd.com/doc/125347967/Tutoriales-WireShark.

HERZOG, P., *Institute for Security and Open Methodologies* [en línea]. 2003. 2019-03-26: ISECOM. 2003. Disponible en: http://fcbi.unillanos.edu.co/segurinfo.unillanos/archivos/materialApoyo/OSSTMM.es.2.1.pdf.

IMPERVA, *DDoS Attacks*. *Source* [en línea], 2019. [Consulta: 18 noviembre 2018]. ISSN 13891286. DOI 10.1109/ISSPIT.2003.1341092. Disponible en: https://www.incapsula.com/ddos/ddos-attacks.html.

INTEF, *Servidor DHCP y Servidor DNS* [en línea]. 2012. España: Instituto Nacional de Tecnologías Educativas y de Formación del Profesorado. 2012. Disponible en: http://fluidos.eia.edu.co/hidraulica/articuloses/maquinashidraulicas/turbinas_gas/page4.html.

KOTTLER, S., *February 28th DDoS Incident Report. GitHub* [en línea]. 2018. [Consulta: 13 noviembre 2018]. Disponible en: https://githubengineering.com/ddos-incident-report/.

LEGERÉN, E., *Diseño de un sistema de información mediante una intranet corporativa :*

propuesta de implementación en una empresa constructora de la provincia de Granada. [en línea], 2015. [Consulta: 7 diciembre 2018]. Disponible en: http://eprints.rclis.org/20357.

MACIÁ, G., *Ataques de denegación de servicio a baja tasa contra servidores* [en línea]. Tesis Postgrado. S.l.: UNIVERSIDAD DE GRANADA. 2007. [Consulta: 17 noviembre 2018]. Disponible en: http://digibug.ugr.es/bitstream/handle/10481/1543/16714763.pdf?sequence=1&isAllowed=y.

MIFSUF, E., *Introducción a Apache* [en línea]. 2017. España: Ministerio de Educacion, Cultura y Deporte - España. 2017. Disponible en: http://descargas.pntic.mec.es/mentor/visitas/Apache.pdf.

MOSCOSO, E., *Desarrollo de una aplicación para la implementación de calidad de servicio por priorización de tráfico sobre una red definida por software (SDN)* [en línea]. Tesis Pregrado. S.l.: ESCUELA POLITÉCNICA NACIONAL. 2016. [Consulta: 31 octubre 2018]. Disponible en: http://bibdigital.epn.edu.ec/handle/15000/15202.

NATARAJAN, S., *OpenFlow version 1.3 tutorial | SDN Hub. SDN Hub* [en línea]. 2014. [Consulta: 21 noviembre 2018]. Disponible en: http://sdnhub.org/tutorials/openflow-1-3/.

NETLQ, *Amenazas internas: Otros objetivos que ponen en peligro a su empresa* [en línea]. 2016. S.l.: NEtIQ. [Consulta: 6 diciembre 2018]. 2016. Disponible en: https://www.microfocus.com/es-es/media/flash-point-paper/insider_threats_moving_targets_that_put_your_organization_at_risk_fpp_es.pdf.

NUÑEZ, A., *Red Definida por Software (SDN) en base a una infraestructura de software de libre distribución* [en línea]. Tesis Pregrado. S.l.: UNIVERSIDAD TÉCNICA DE AMBATO. 2015. [Consulta: 23 marzo 2018]. Disponible en: http://repositorio.uta.edu.ec/handle/123456789/10587.

OCAMPO, C. et al, *Sistema de detección de intrusos en redes corporativas Intrusion.* [en línea]. S.l.: 2017. [Consulta: 4 enero 2019]. 0122-170. Disponible en: http://revistas.utp.edu.co/index.php/revistaciencia/article/view/9105/10161.

OCHOA, J., *Características de las Redes Definidas por Software (SDN) para su Implementación en el Ecuador* [en línea]. Tesis Postgrado. S.l.: UNIVERSIDAD CATÓLICA DE SANTIAGO DE GUAYAQUIL. 2018. [Consulta: 3 mayo 2018]. Disponible en:

http://repositorio.ucsg.edu.ec/bitstream/3317/9748/1/T-UCSG-POS-MTEL-88.pdf.

OLADUNJOYE, O., *SOftware Defined Networking– The Emerging Paradigm To Computer Networking* [en línea]. Tesis Pregrado. S.l.: UNIVERSIDAD DE TURKIA DE CIENCIAS APLICADAS. 2017. [Consulta: 15 noviembre 2018]. Disponible en: https://www.theseus.fi/bitstream/handle/10024/125665/Oladunjoye_Olanrewaju.pdf?sequence=1&isAllowed=y.

OLAYA, M., *Diseño e implementación de una aplicación para balanceo de carga para una Red Definida por Software (SDN)* [en línea]. Tesis Pregrado. S.l.: ESCUELA POLITÉCNICA NACIONAL. 2015. [Consulta: 26 marzo 2018]. Disponible en: http://oa.upm.es/42968/1/TFM_RAUL_ALVAREZ_PINILLA.pdf.

OPEN NETWORKING FOUNDATION, *OpenFlow Switch Specification.* [en línea]. S.l.: 2011. Disponible en: https://www.opennetworking.org.

OPEN NETWORKING FOUNDATION, *Reference Designs.* [en línea]. 2018. [Consulta: 30 octubre 2018]. Disponible en: https://www.opennetworking.org/reference-designs/.

OPENDAYLIGHT PROJECT, *OpenDaylight Controller Overview.* [en línea]. 2018a. [Consulta: 25 febrero 2019]. Disponible en: https://docs.opendaylight.org/en/stable-fluorine/user-guide/opendaylight-controller-overview.html.

OPENDAYLIGHT PROJECT, ROADMAP & DEVELOPMENT PROCESS. *OpenDaylight Project a Series of LF Projects* [en línea]. 2018b. [Consulta: 25 febrero 2019]. Disponible en: https://www.opendaylight.org/technical-community/getting-started-for-developers/roadmap.

OPENVAS, *OpenVAS - Open Vulnerability Assessment System*. *Greenbone Networks GmbH* [en línea]. 2019. [Consulta: 13 marzo 2019]. Disponible en: http://www.openvas.org/index-de.html.

ORACLE, *Introducción al conjunto de protocolos TCP/IP*. [en línea]. 2010. [Consulta: 18 noviembre 2018]. Disponible en: https://docs.oracle.com/cd/E24842_01/html/820-2981/ipov-6.html.

PARACUELLOS, J., *Defensa proactiva y reactiva ante ataques DDoS en un entorno simulado de redes definidas por software* [en línea]. Tesis Pregrado. S.l.: UNIVERSIDAD DE

ZARAGOZA. 2016. [Consulta: 3 diciembre 2018]. Disponible en: http://webdiis.unizar.es/~ricardo/files/PFCs-TFGs/Defensa-Proactiva-Reactiva-DDoS-SDN/Memoria_TFG_DefensaProactivaReactivaDDoSSDN.pdf.

PAZMIÑO, E., *Implementación de la tecnología SDN para control de acceso y calidad de servicio en redes domésticas* [en línea]. Tesis Pregrado. S.l.: UNIVERSIDAD DE LAS FUERZAS ARMADAS-ESPE. 2018. [Consulta: 7 noviembre 2018]. Disponible en: https://repositorio.espe.edu.ec/bitstream/21000/13847/1/T-ESPE-057535.pdf.

PEÑA, Ó., *Tendencias digitales 2016/17* [en línea]. 2017. España: Grey Group. [Consulta: 13 abril 2018]. 2017. Disponible en: http://grey.com/emea/spain/noticias/oscar/tendencias-digitales-2016-2017/tendencias_digitales_2016_2017.pdf.

PINCAY, E., *Análisis comparativo de la calida de servicio entre las redes actuales y redes de próxima generación* [en línea]. Tesis Pregrado. S.l.: ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO. 2015. [Consulta: 26 febrero 2019]. Disponible en: http://dspace.espoch.edu.ec/bitstream/123456789/3776/1/18T00581.pdf.

REDONDO, M. et al, *Intranet: Soporte para entornos de aprendizaje.* [en línea]. 2015. S.l.: UNIVERSIDAD DE JAÉN. [Consulta: 9 diciembre 2018]. 2015. Disponible en: https://www.researchgate.net/publication/39148968_Intranet_soporte_para_entorno_de_aprendizaje.

RIBES, B., *OpenDaylight SDN controller platform* [en línea]. Tesis Pregrado. S.l.: UNIVERSITAT POLITÈCNICA DE CATALUNYA. 2015. [Consulta: 8 noviembre 2018]. Disponible en: https://upcommons.upc.edu/bitstream/handle/2117/79422/odl-completo.pdf.

RODRIGUES, C. et al, *Avaliação de Balanceamento de Carga Web em Redes Definidas por Software* [en línea]. 2015. S.l.: UNIVERSIDADE FEDERAL DE JUIZ DE FORA. [Consulta: 23 marzo 2018]. 2015. Disponible en: http://sbrc2015.ufes.br/wp-content/uploads/138783.1.pdf.

SAKELLAROPOULOU, D., *A Qualitative Study of SDN Controllers* [en línea]. Tesis Pregrado. S.l.: UNIVERSIDAD DE ATENAS DE ECONOMÍA Y NEGOCIOS. 2017. [Consulta: 12 noviembre 2018]. Disponible en: https://mm.aueb.gr/master_theses/xylomenos/Sakellaropoulou_2017.pdf.

SALINAS, G., *Estudio de redes definidas por software e implementación de escenarios virtuales*

de prueba [en línea]. Tesis Postgrado. S.l.: UNIVERSIDAD POLITÉCNICA DE MADRID. 2017. [Consulta: 8 noviembre 2018]. Disponible en: https://www.dit.upm.es/~posgrado/doc/TFM/TFMs2016-2017/TFM_Gabriela_Salinas_Jardon_2017.pdf.

SANDOVAL, C., *Implementación de un clúster-controlador de SDN basado en un framework de software libre para la infraestructura cloud de la Facultad de Ingeniería en Ciencias Aplicadas.* [en línea]. Tesis Pregrado. S.l.: UNIVERSIDAD TÉCNICA DEL NORTE. 2018. [Consulta: 23 marzo 2018]. Disponible en: http://repositorio.utn.edu.ec/handle/123456789/7986.

SOLER, E., *Diseño e implementacion de una solucion de VOIP* [en línea]. Tesis Pregrado. S.l.: UNIVERSITAT POLITÉCNICA DE CATALUNYA. 2015. [Consulta: 24 marzo 2019]. Disponible en: https://upcommons.upc.edu/bitstream/handle/2099.1/8373/Memoria PFC - Erika Soler.pdf.

TAQUI, S. y CUADROS, C., *Implementación de una extranet para la gestión académica en el instituto de emprendedores de la Universidad San Ignacio de Loyola* [en línea]. Tesis Pregrado. S.l.: UNIVERSIDAD DE SAN MARTÍN DE PORRES. 2017. [Consulta: 29 diciembre 2018]. Disponible en: http://www.repositorioacademico.usmp.edu.pe/bitstream/usmp/3980/1/tarqui_cuadros.pdf.

TARAZONA, C., *Amenazas Informáticas y seguridad de la informacion* [en línea]. 2015. S.l.: s.n. [Consulta: 13 noviembre 2018]. 2015. Disponible en: https://revistas.uexternado.edu.co/index.php/derpen/article/view/965.

VELÁSQUEZ, W., *Emulación de una red definida por software utilizando MiniNet.* [en línea]. S.l.: 2013. [Consulta: 23 marzo 2018]. Disponible en: https://www.academia.edu/5730624/Emulación_de_una_red_definida_por_software_utilizando_MiniNet.

VIJAY, P. y VASUDEVAN, D., *The Northbound APIs of software defined networks.* [en línea]. 2016. S.l.: INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY. [Consulta: 1 mayo 2018]. 2016. Disponible en: http://www.ijesrt.com/issues pdf file/Archive-2016/October-2016/61.pdf.

WANG, T., *Benefits and the Security Risk of Software-defined Networking feature. Isaca Journal* [en línea], 2016. p. 1-3. [Consulta: 2 diciembre 2018]. Disponible en:

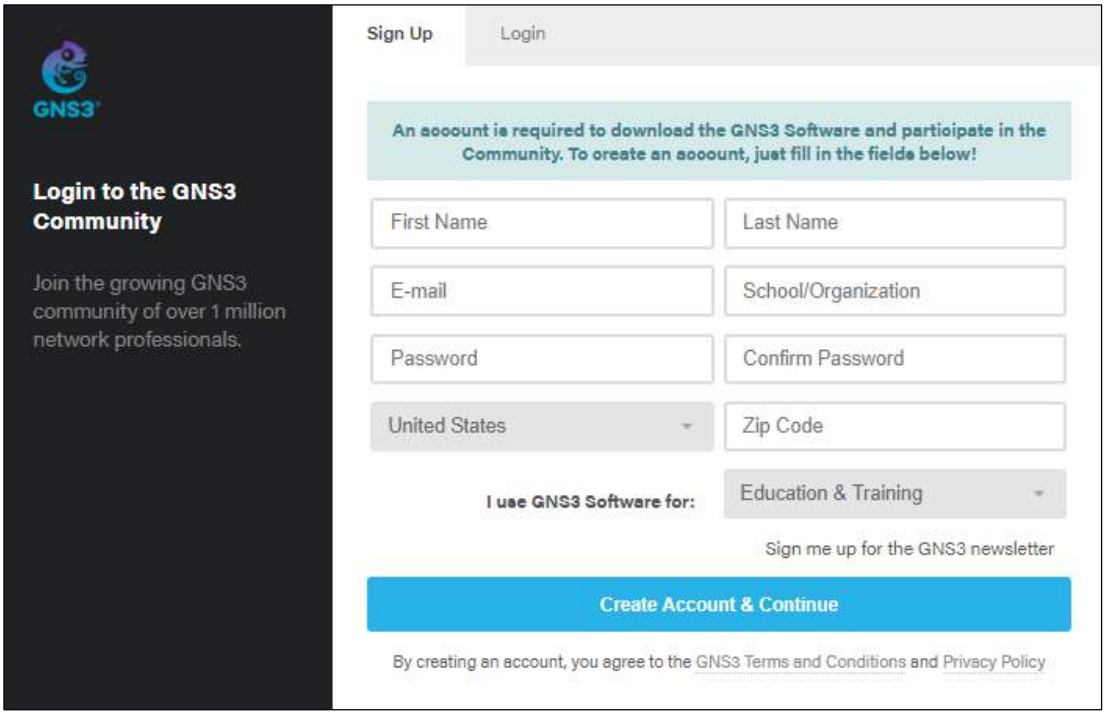
https://www.isaca.org/Journal/archives/2016/volume-4/Documents/Benefits-and-the-Security-Risk-of-Software-defined-Networking_joa_Eng_0716.pdf.

YAGUËS, P., *Programación de redes SDN mediante el controlador POX* [en línea]. Tesis Pregrado. S.l.: UNIVERSIDAD POLITÉCNICA DE CARTAGENA. 2015. [Consulta: 6 noviembre 2018]. Disponible en: http://repositorio.upct.es/bitstream/handle/10317/5254/tfg729.pdf?sequence=1.

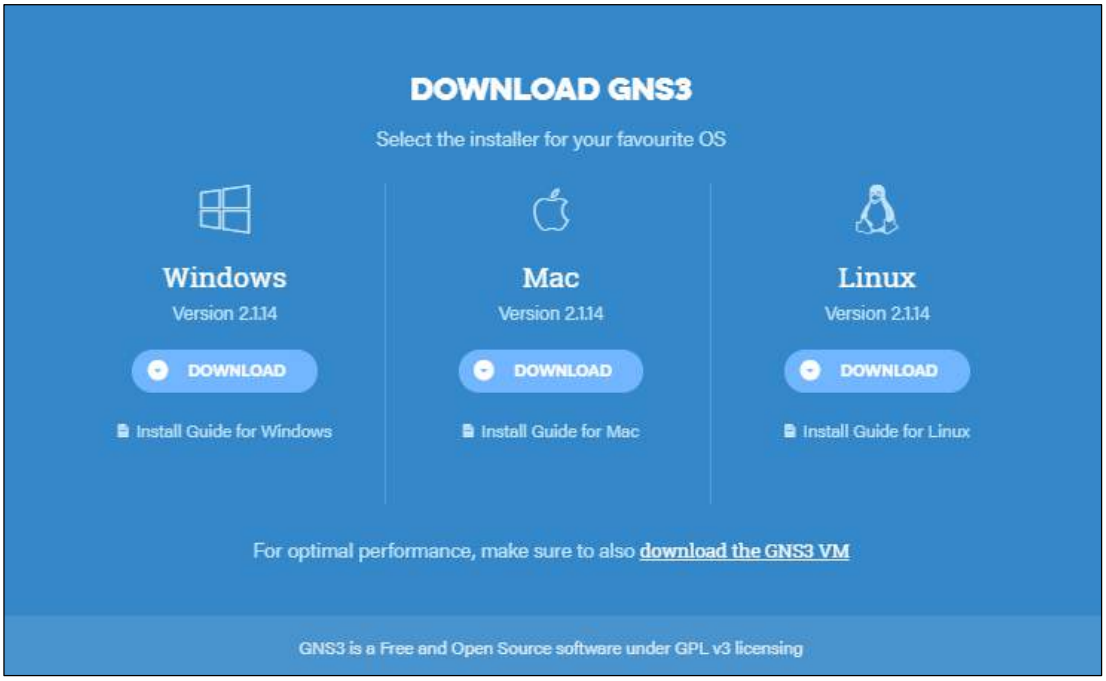
ANEXOS

ANEXO A: Instalación de GNS3.

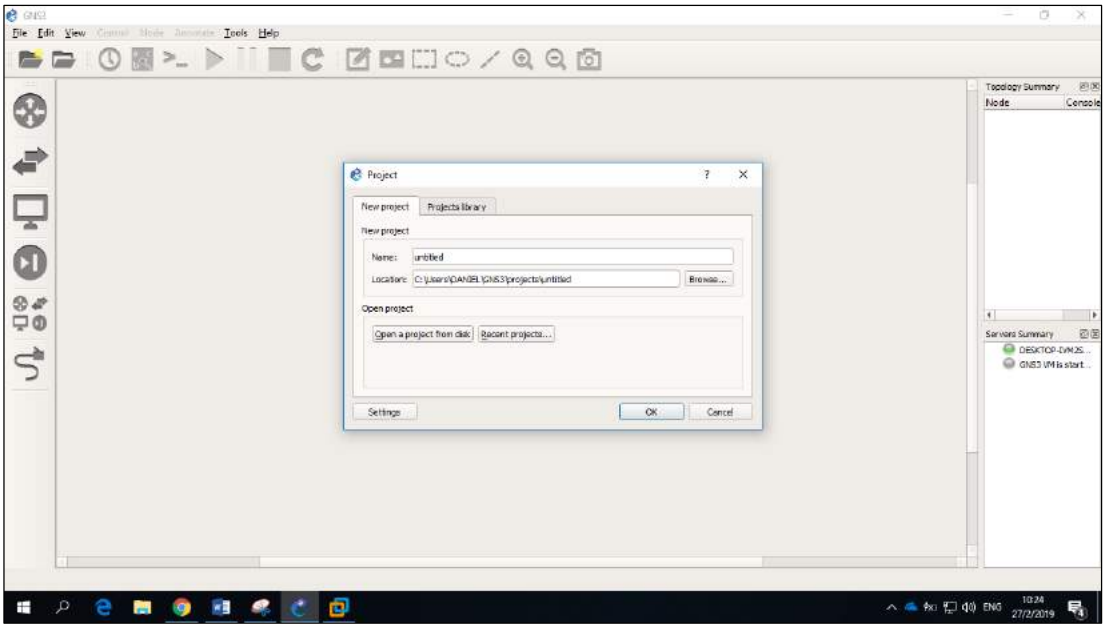
Dirigir al sitio oficial de GNS3 https://gns3.com/software/download y registrarse como usuario.



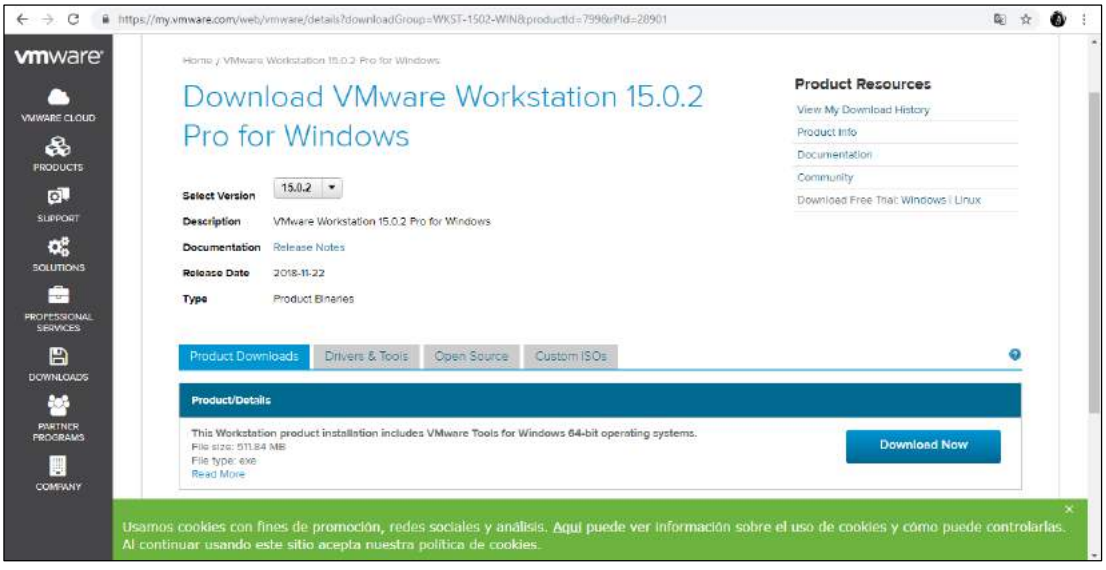
Elegir para el tipo de S.O requerido.



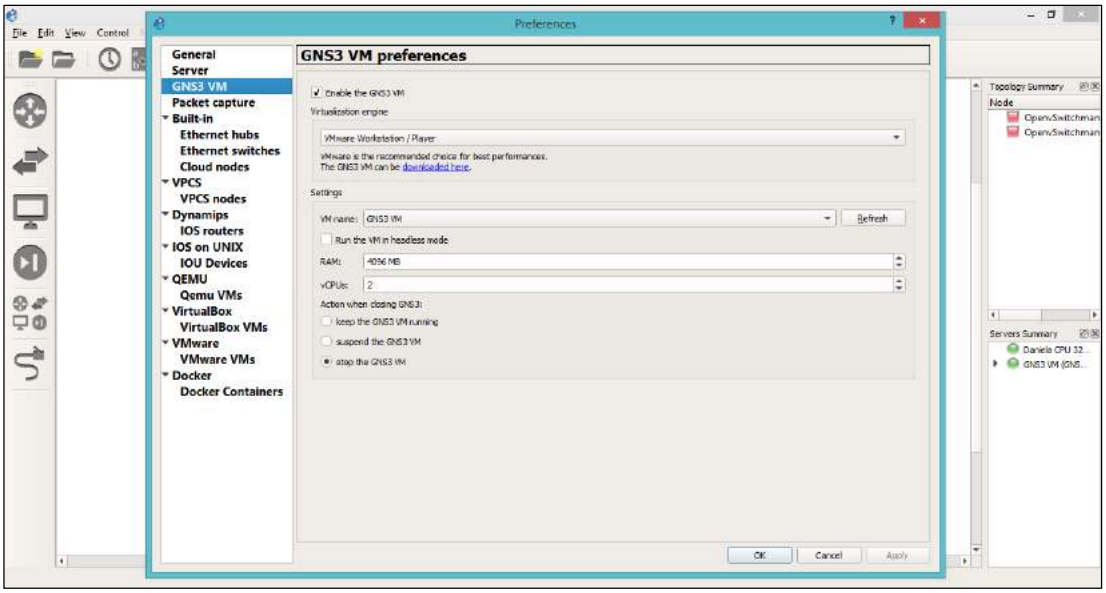
Para instalar se debe ejecutar el archivo de descarga y seguir todos los pasos. Y finalmente se podrá acceder al software.



Después vincular GNS3 con una máquina virtual, e este caso se utiliza VMWare Workstation PRO, obtenida desde el sitio oficial:

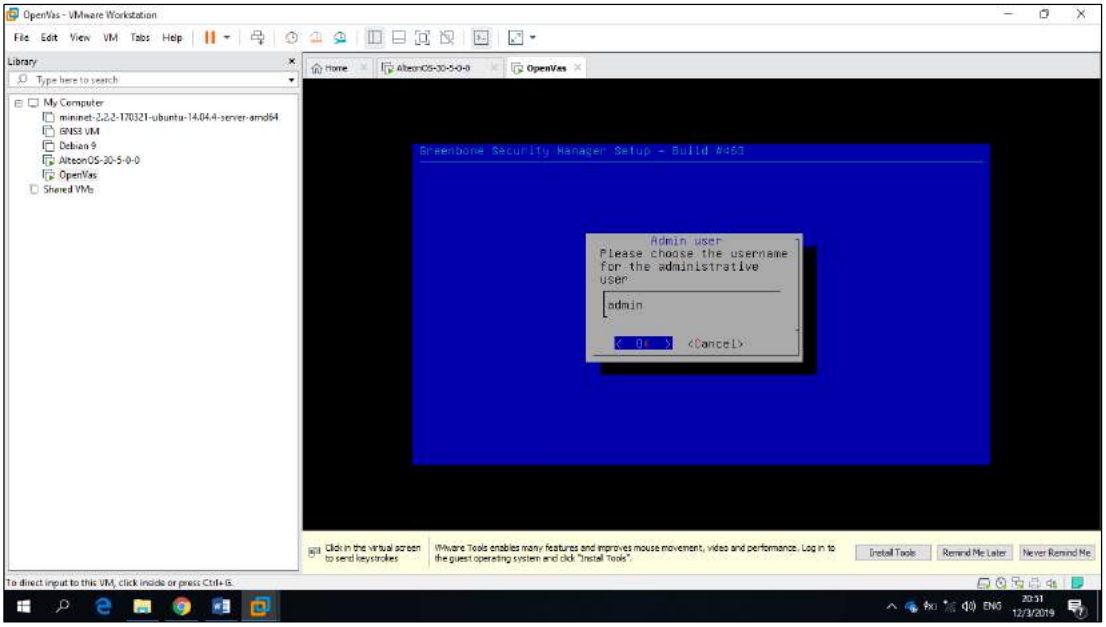


Y para vincularlo con la máquina creada en VMware, colocar la dirección IP y puerto de comunicación con la que está trabajando el GNS3.

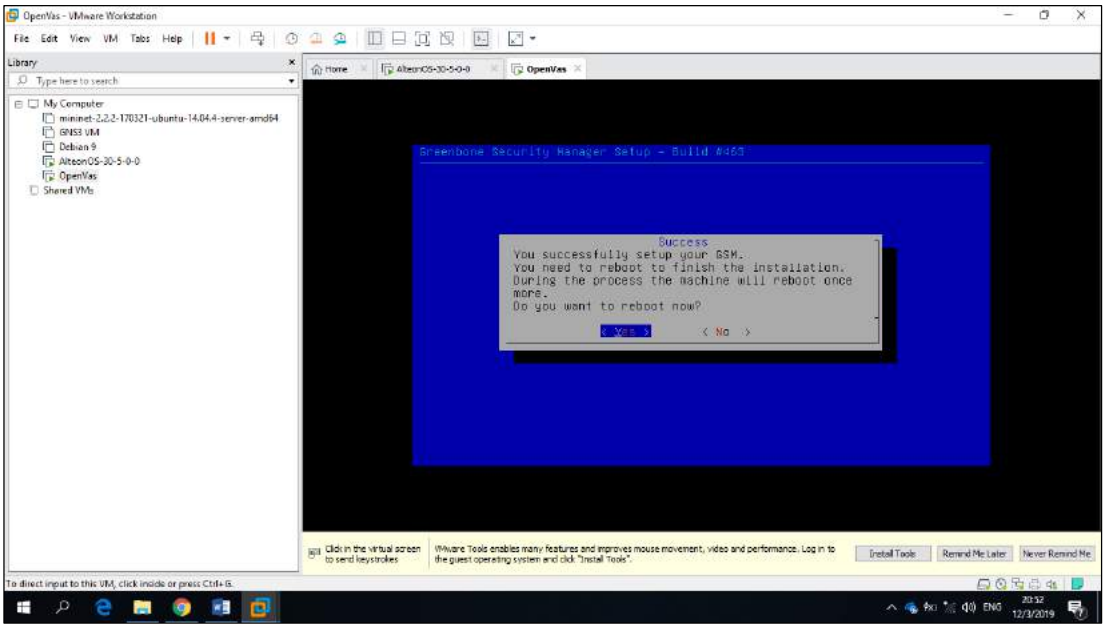

File Edit View Control
Preferences
General
Server
GNS3 VM
Packet capture
Built-in
Ethernet hubs
Ethernet switches
Cloud nodes
VPCS
VPCS nodes
Dynamips
IOS routers
IOS on UNIX
IOU Devices
QEMU
Qemu VMs
VirtualBox
VirtualBox VMs
VMware
VMware VMs
Docker
Docker Containers
GNS3 VM preferences
Enable the GNS3 VM
Virtualization engine
VMware Workstation / Player
VMware is the recommended choice for best performances.
The GNS3 VM can be downloaded here.
Settings
VM name:
GNS3 VM
Refresh
Run the VM in headless mode
RAM:
4096 MB
vCPUs:
2
Action when closing GNS3:
keep the GNS3 VM running
suspend the GNS3 VM
stop the GNS3 VM
OK
Cancel
Apply
Topology Summary
Node
Servers Summary

ANEXO B: Instalación de OpenVas.

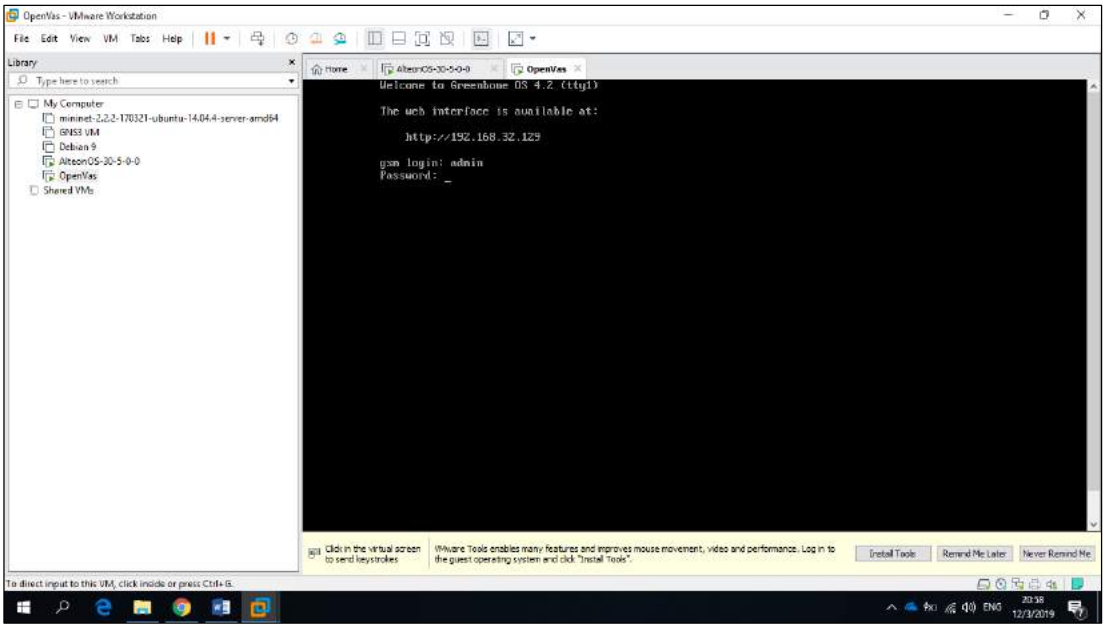
Seleccionar la imagen ISO de OpenVas y ejecutar en una máquina virtual. Crear un nombre de usuario (admin) y su respectiva contraseña.



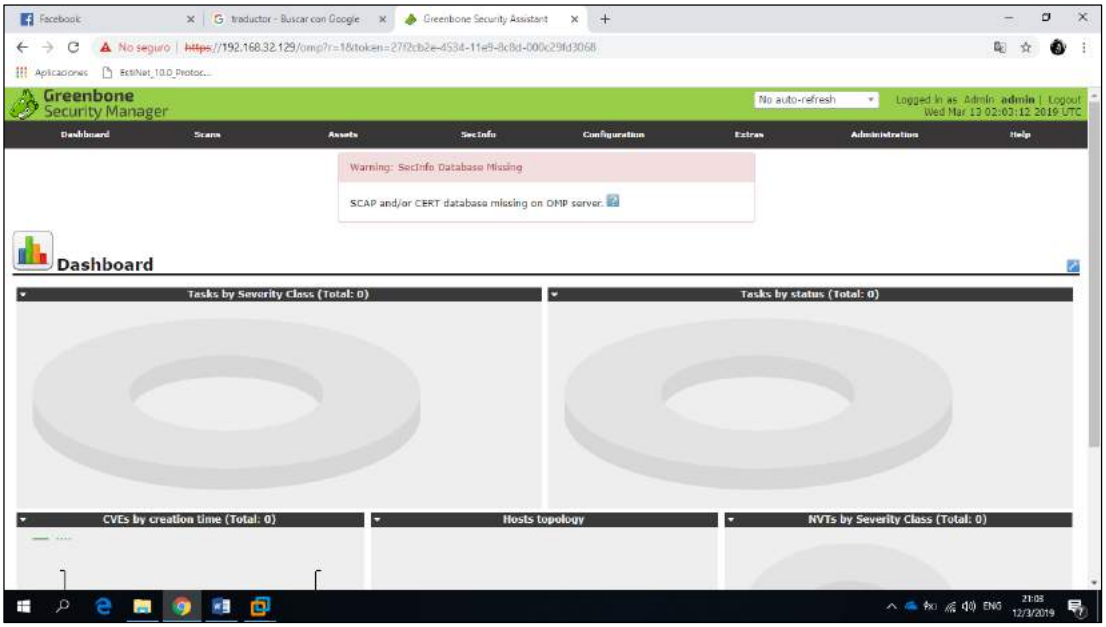
Configurar el GSM y reiniciar el sistema.



Reiniciar y acceder con el usuario y contraseña anteriormente.



Para completar la configuración de GSM, crear un administrador web. Y acceder ingresando la dirección IP de la interfaz de la MV.



ANEXO C: Instalación de Opendaylight

Descargar todas las actualizaciones de los repositorios y del sistema.

```
root@daniela-VirtualBox:/home/daniela# apt-get update
root@daniela-VirtualBox:/home/daniela# apt-get upgrade
```

Agregar Java 8

```
root@daniela-VirtualBox:/home/daniela#      sudo      add-apt-repository
ppa:webupd8team/java
root@daniela-VirtualBox:/home/daniela# apt-get updade
root@daniela-VirtualBox:/home/daniela# sudo apt-get install oracle-java8-
installer
root@daniela-VirtualBox:/home/daniela# java -version
```

Instalación de ODL Berylium.

```
root@daniela-VirtualBox:/home/daniela# ls
root@daniela-VirtualBox:/home/daniela#  unzip  distribution-karaf-0.4.4-
Beryllium-SR4.zip
root@daniela-VirtualBox:/home/daniela#    cd    distribution-karaf-0.4.4-
Beryllium-SR4
root@daniela-VirtualBox:/home/daniela# cd bin
roo t@daniela-VirtualBox:/home/daniela# ./karaf
```

Instalación de características de ODL.

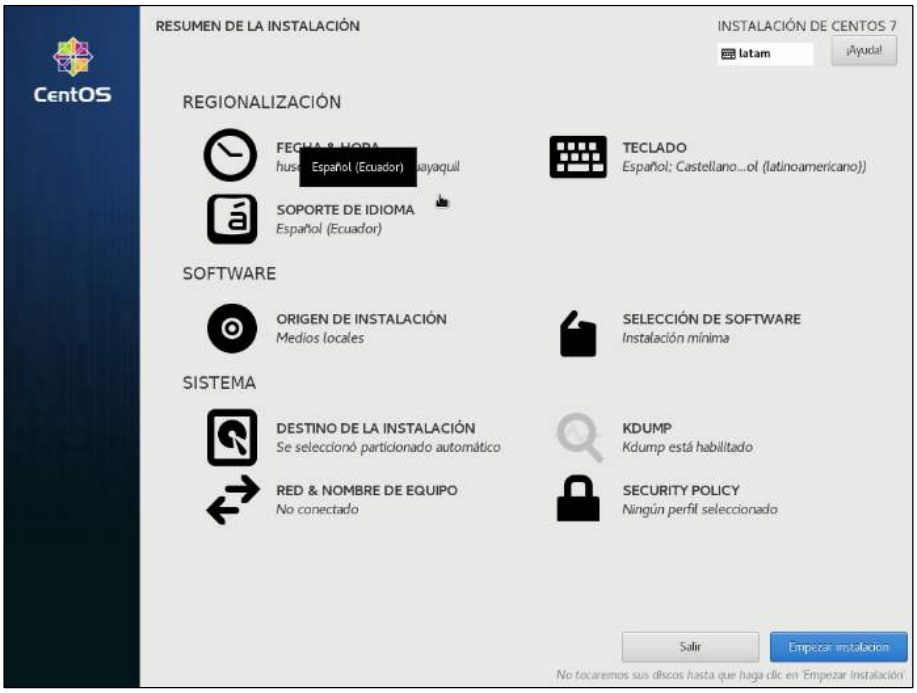
```
feature:install <característica 1>
feature:install <característica1><característica2><característica n> …
```

Verificar los módulos instalados con el siguiente comando.

```
opendaylight-user@root>feature:list --installed
```

ANEXO D: Instalación de Centos

Importar la imagen ISO en una nueva máquina virtual y modificar los parámetros de acuerdo a preferencias del usuario y presionar empezar instalación.



Cambiar el nombre de usuario y contraseña. Al final reiniciar e ingresar con el usuario y contraseña.

ANEXO E: Desarrollo de los servicios de red.

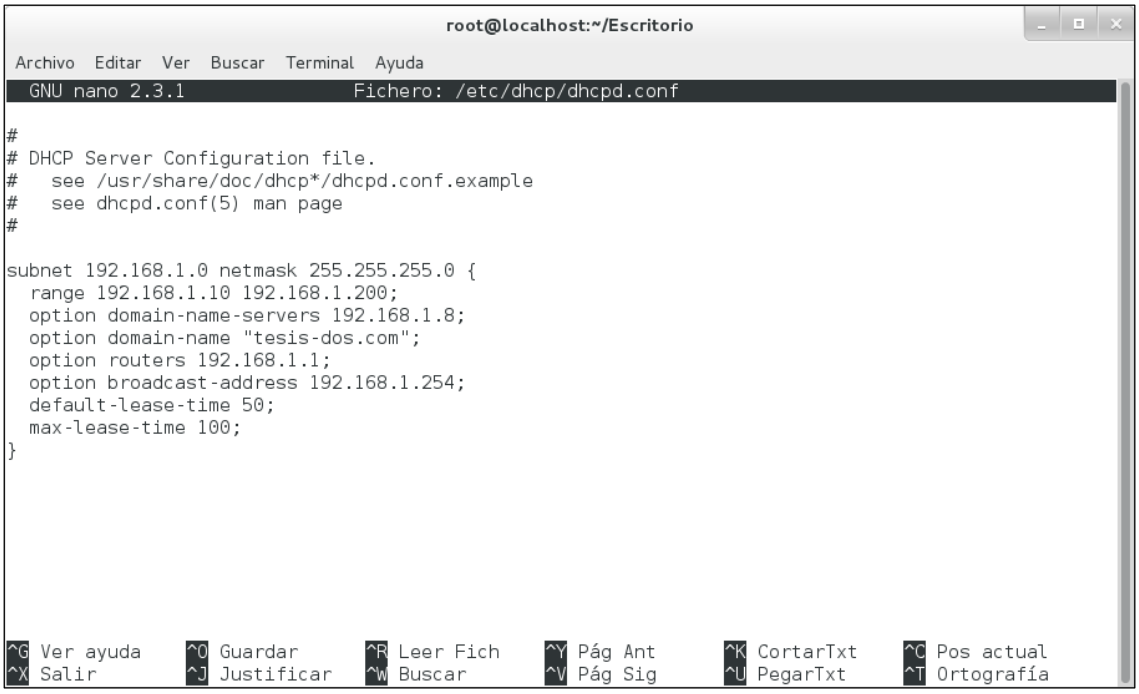
SERVIDOR DHCP

Descargar todos los paquetes desde el repositorio.

```
yum install dhcp
```

Editar el archivo dhcpd.conf.

```
nano /etc/dhcp/dhcpd.conf
```



Iniciar, activar y comprobar el estado del servicio DHCP.

```
systemctl start dhcpd
service named start
```

Deshabilitar y detener Firewall y Selinux en el archivo /etc/sysconfig/selinux la opción SELINUX=disabled,

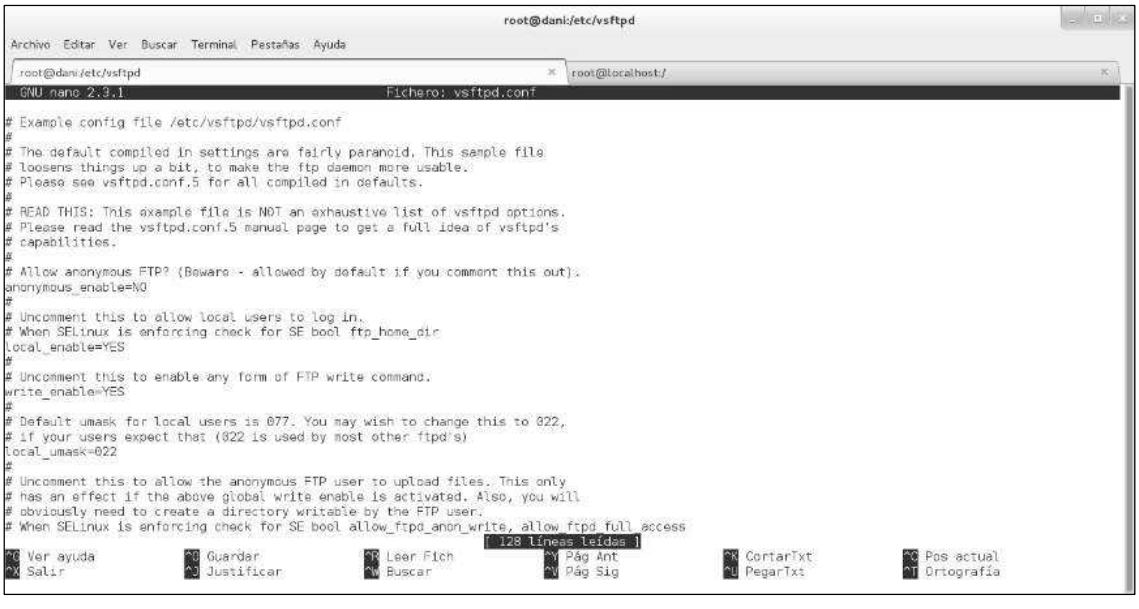
```
systemctl disable firewalld
systemctl stop firewalld
systemctl restart firewalld
```

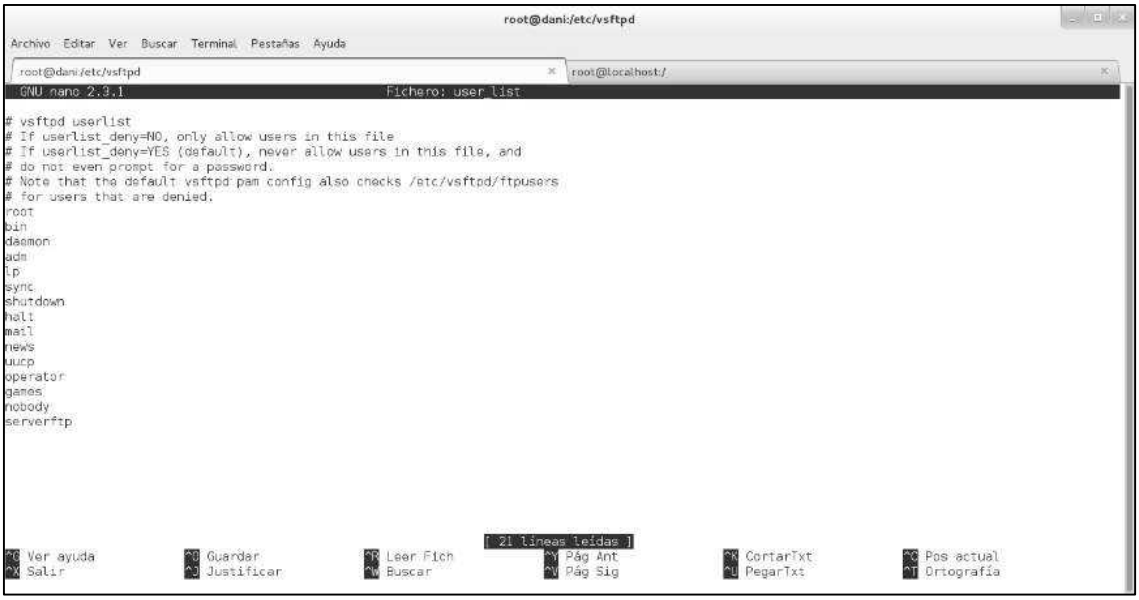
SERVIDOR FTP

Descargar los archivos de los repositorios.

yum install vsftpd

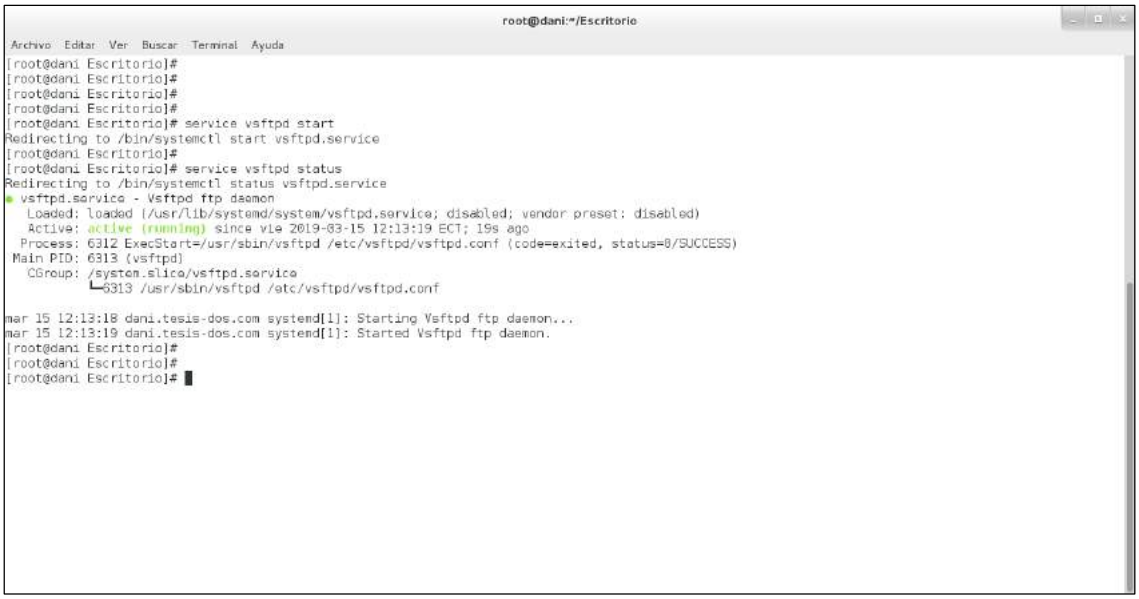
Configurar los archivos vsftpd.conf y *user_list*..





Ejecutar el servicio FTP.

```
systemctl start vsftpd
```



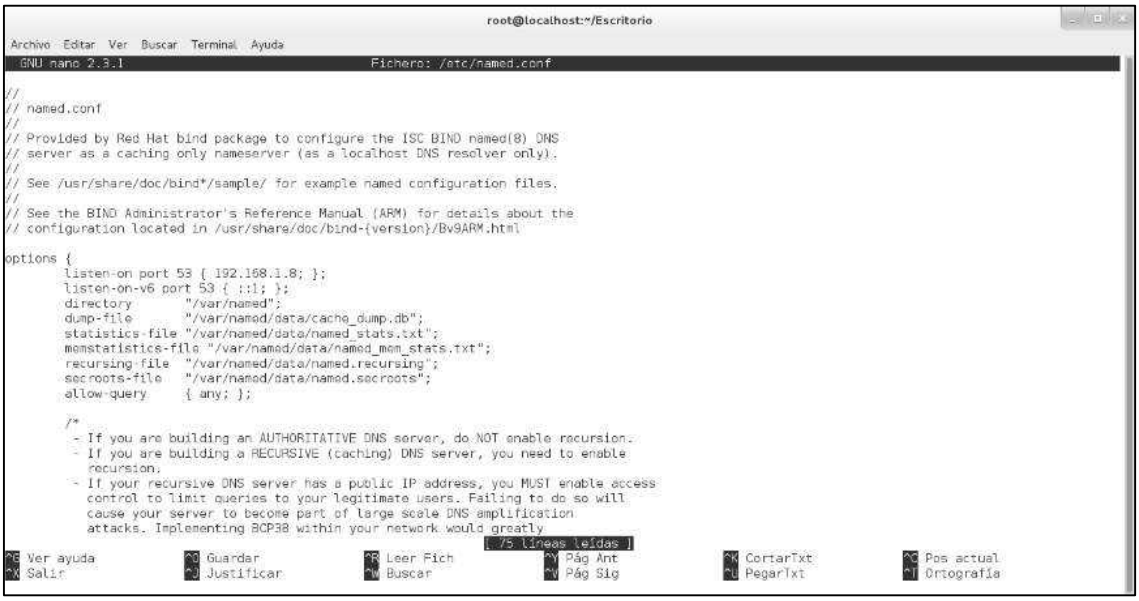
SERVIDOR DNS

Para empezar, instalar los ficheros BIND.

```
yum install bind
```

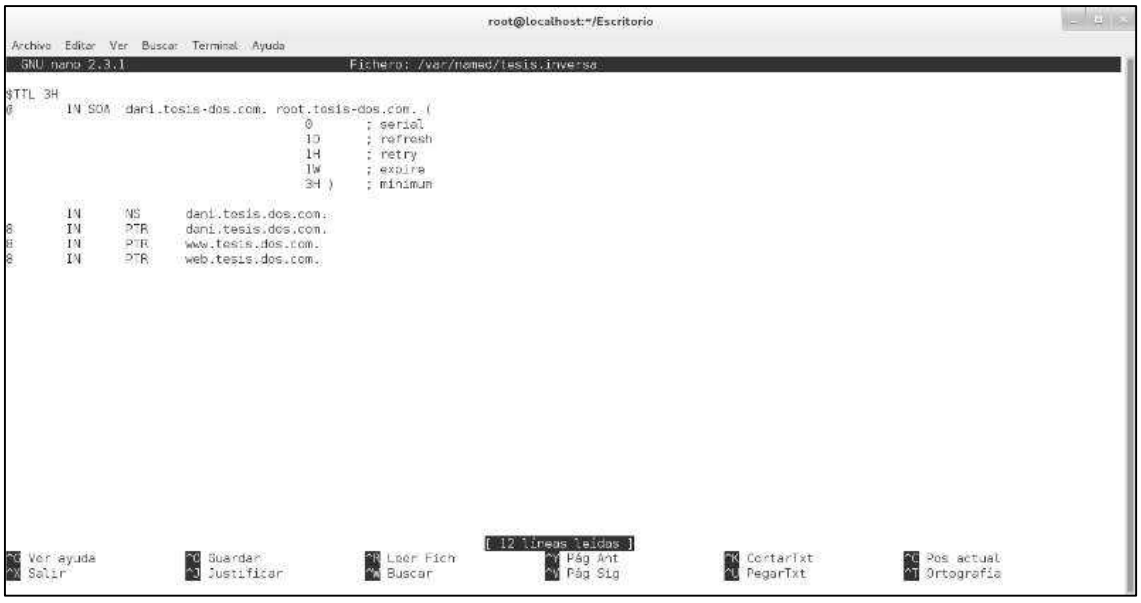
Desactivar firewall y Selinux. Modificar el archivo de configuración.

```
nano /etc/named.conf
```

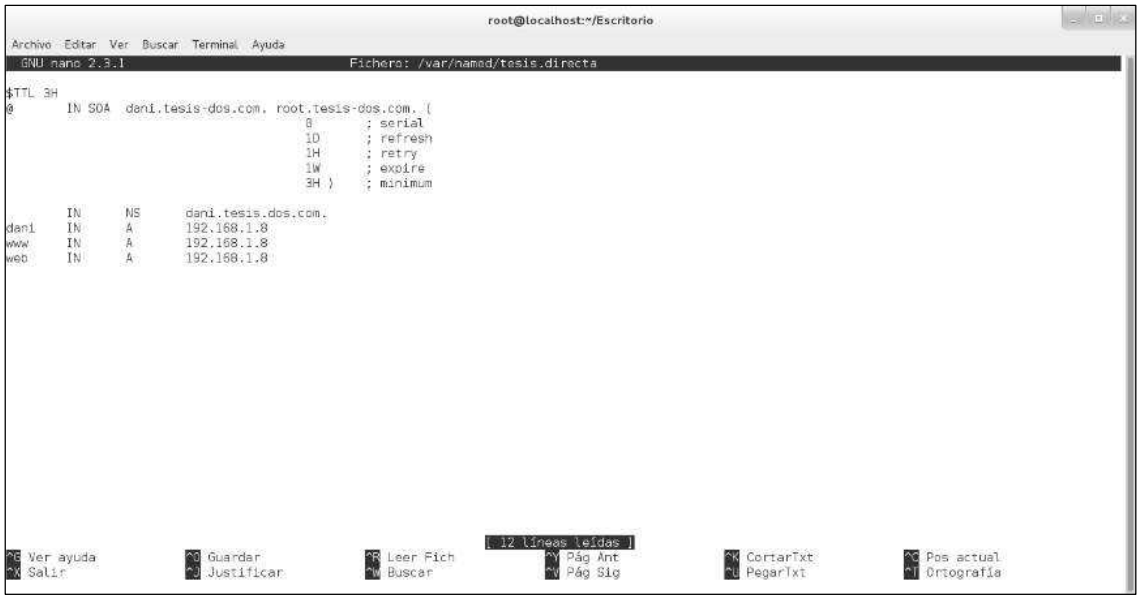


Crear dos archivos de zonas DNS: directa e inversa.

nano tesis.inversa



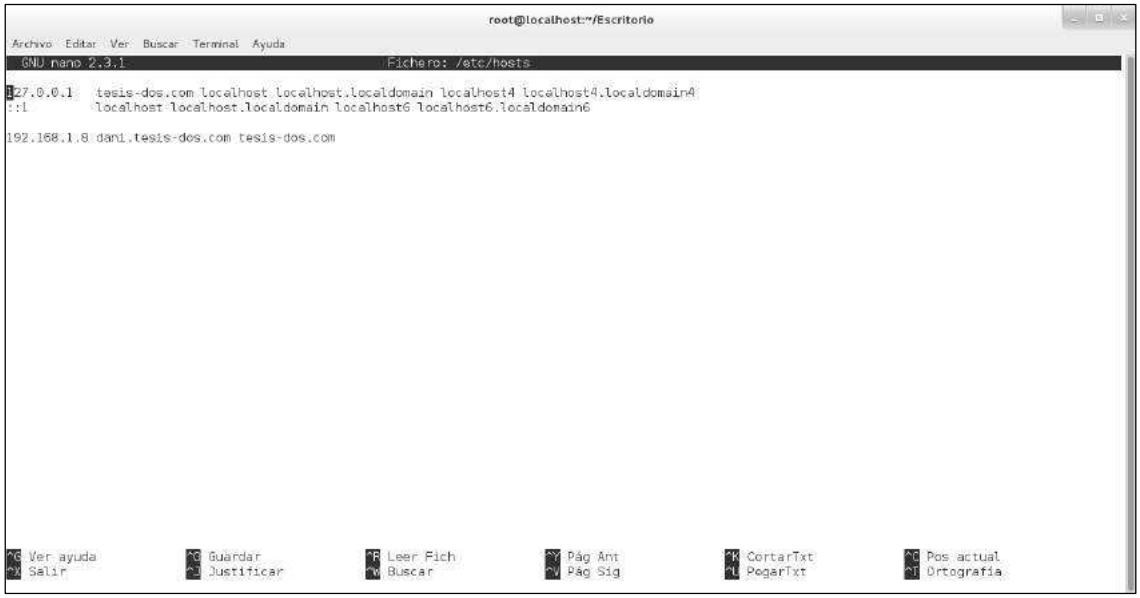
nano tesis.directa

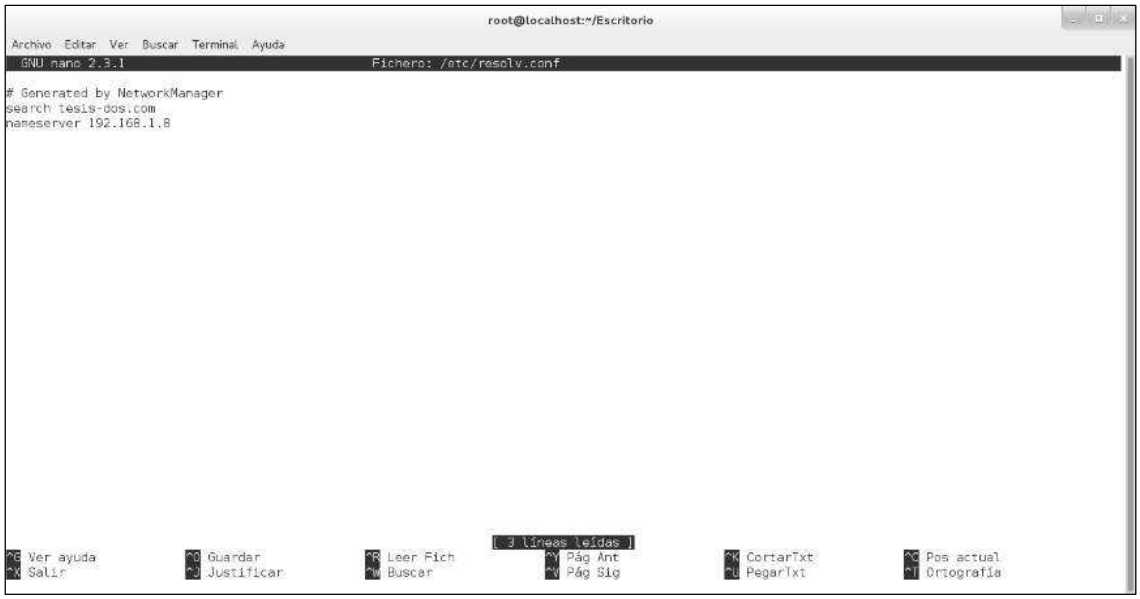


Registrar el nombre del servidor en el archivo.

nano /etc/hosts

/etc/resolv.conf





Modificar permisos de los ficheros. Y ejecutar el servicio DNS.

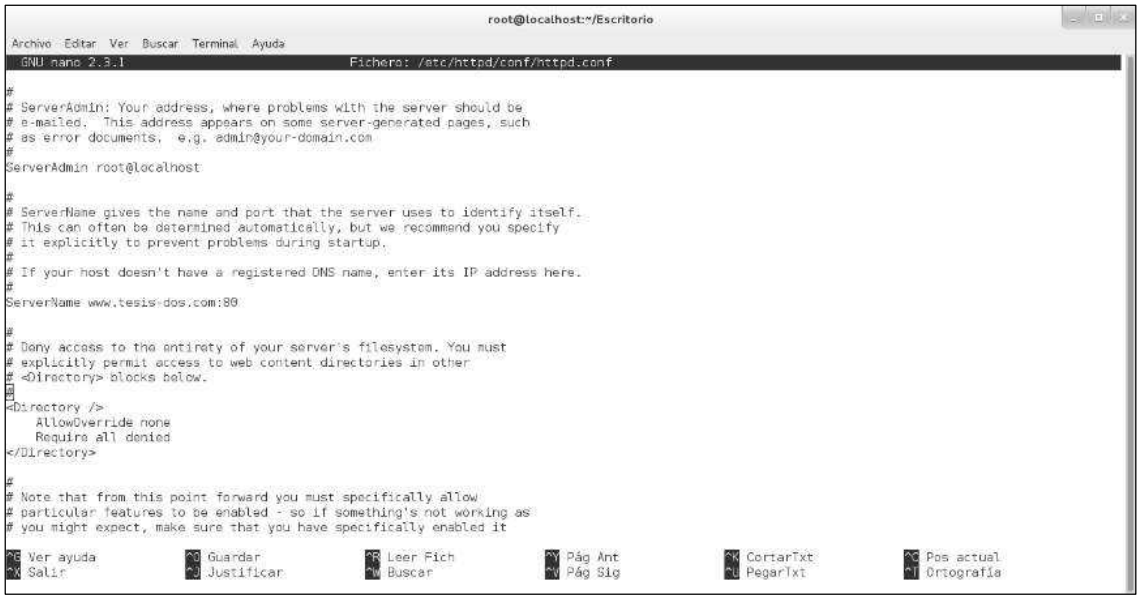
```
chgrp named prueba.directa
chgrp named prueba.inversa
systemctl start named
service named start
```

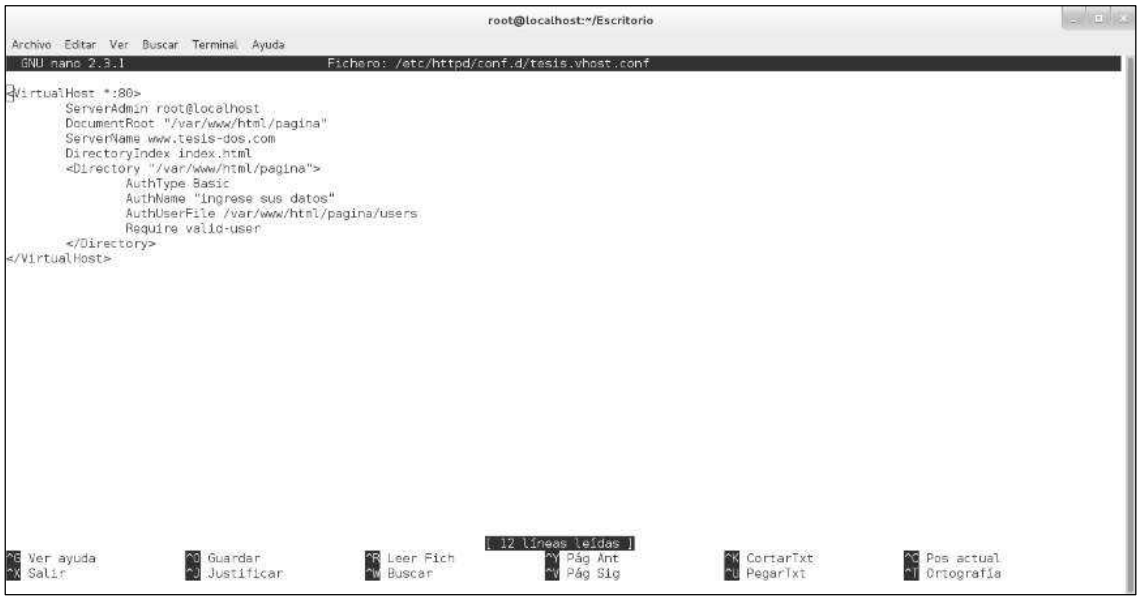
SERVIDOR HTTP

Empezar instalando el software APACHE.

```
yum install httpd
```

Desactivar firewall y Selinux y configurar el archivo httpd.conf.





Generar un nuevo fichero con los siguientes parámetros.

```
cd confi.d
touch tesis.vhost.conf
nano tesis.vhost.conf
```

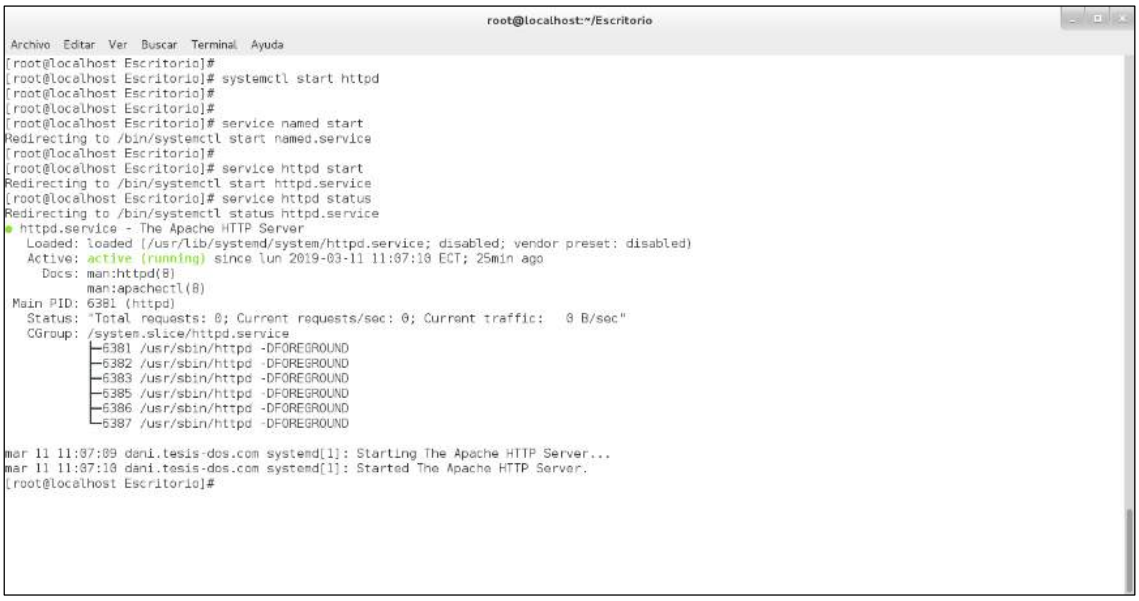
Ingresar la dirección de la página web, guardar toda la configuración y ejecutar el servicio HTTP.

```
cd /var/www/html
cd pagina
```

nano index.html

systemctl httpd status

service httpd start



SERVIDOR VoIP

Instalar algunas dependencias básicas para Asterisk.

yum install make wget openssl-devel ncurses-devel newt-devel libxml2-devel kernel-devel gcc gcc-g++ sqlite-devel

Acceder al directorio *cd /usr/src* y después descargar los códigos fuente.

wgetdownloads.asterisk.org/pub/telephony/asterisk/asterisk-15-current.tar.gz

wgetldownloads.asterisk.org/pub/telephony/dahdi-linux-complete/dahdi-linux-complete-current.tar.gz

wget downloads.asterisk.org/pub/telephony/libpri/libpri-current.tar.gz

wget http://www.digip.org/jansson/releases/jansson-2.5.tar.gz

Extraer los archivos descargados a sus directorios correspondientes.

tar zxfv dahdi-linux-complete-current.tar.gz

tar zxfv libpri-current.tar.g

tar -zxf jansson-2.5.tar.gz

tar zxfv asterisk-15-current.tar.gz

DAHDI

```
cd dahdi-linux-complete-3.0.0+3.0.0/
make
make install
make install-config

LibPRI
yum install uuid-devel libuuid-devel bison sub version git-core

Jansson
cd jansson-2.5/
./configure
./configure -prefix=/usr/
make clean
make make install
ldconfig
```

Instalar Aterisk dependiendo de la versión de Centos.

```
cd ../asterisk-15.7.2/
./configure --libdir=/usr/lib64
make menuselect
make
make install
make samples
```

Configuración de parámetros para el servicio VoIP

```
We will be editing the following files:
? sip.conf
? extensions.conf
? voicemail.conf
These files are located in:
? /etc/asterisk
Making a backup of sip.conf
Type the following to move the original sip.conf
sudo mv /etc/asterisk/sip.conf /etc/asterisk/sip.conf.orig
Creating a new sip.conf and configuring it
Type the following to create a new sip.conf
sudo vi /etc/asterisk/sip.conf
[general]
context=internal
allowguest=no
allowoverlap=no
bindport=5060
bindaddr=0.0.0.0
```

```
srvlookup=no
disallow=all
allow=ulaw
alwaysauthreject=yes
canreinvite=no
nat=yes
session-timers=refuse
localnet=192.168.1.0/255.255.255.0
[7001]
type=friend
host=dynamic
secret=123
context=internalGG
[7002]
type=friend
host=dynamic
secret=456
context=internal
```

Image below for reference.

Making a backup of extensions.conf

Type the following to move the original extensions.conf

```
sudo mv /etc/asterisk/extensions.conf /etc/asterisk/extensions.conf.orig
```

Creating a new extensions.conf and configuring it

Type the following to create a new extensions.conf

```
sudo vi /etc/asterisk/extensions.conf
[internal]
exten => 7001,1,Answer()
exten => 7001,2,Dial(SIP/7001,60)
exten => 7001,3,Playback(vm-nobodyavail)
exten => 7001,4,VoiceMail(7001@main)
exten => 7001,5,Hangup()
exten => 7002,1,Answer()
exten => 7002,2,Dial(SIP/7002,60)
exten => 7002,3,Playback(vm-nobodyavail)
exten => 7002,4,VoiceMail(7002@main)
exten => 7002,5,Hangup()
exten => 8001,1,VoicemailMain(7001@main)
exten => 8001,2,Hangup()
exten => 8002,1,VoicemailMain(7002@main)
exten => 8002,2,Hangup()
```

Image below for reference

Type the following to move the original voicemail.conf

```
sudo mv /etc/asterisk/voicemail.conf /etc/asterisk/voicemail.conf.orig
```

Creating a new voicemail.conf and configuring it

Type the following to create a new voicemail.conf

sudo vi /etc/asterisk/voicemail.conf

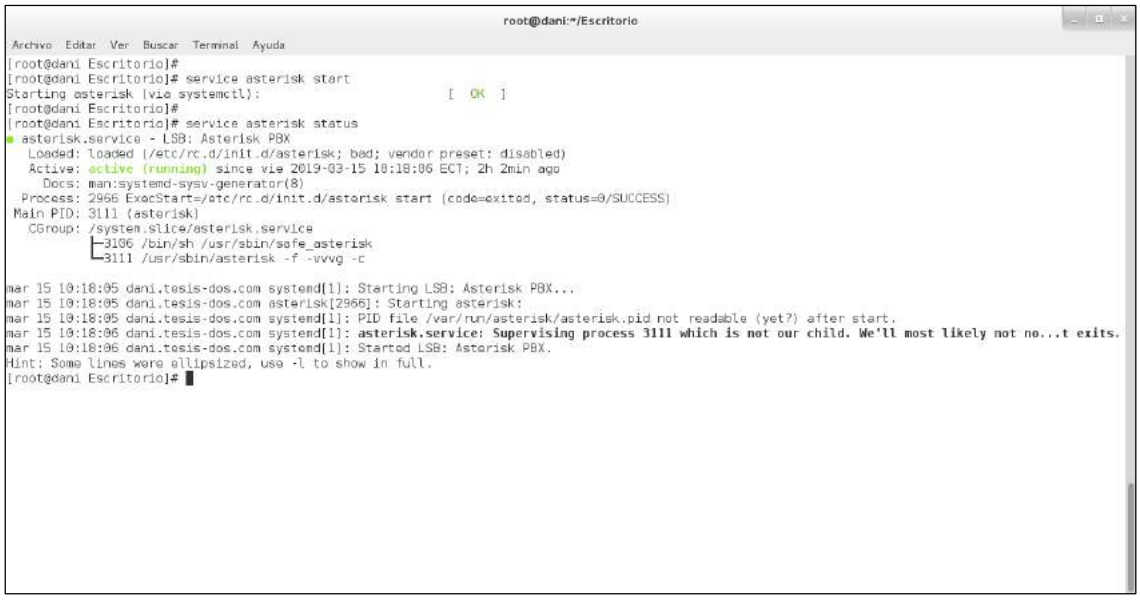
[main]

7001 => 123

7002 => 456

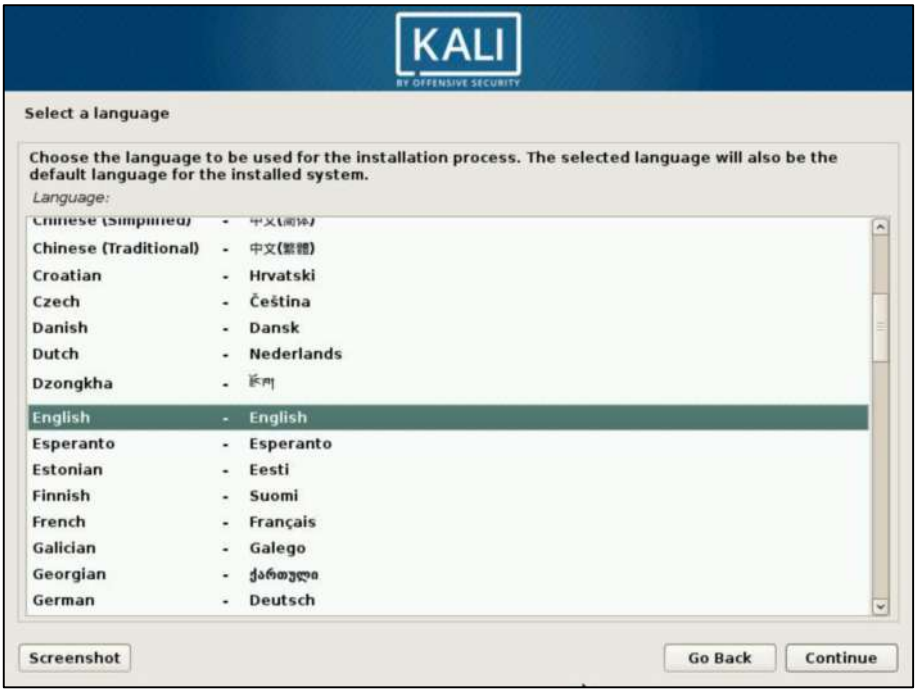
Ejecutar el servicio de Asterisk.

service asterick start



ANEXO F: Instalación de Kali Linux

Elegir la instalación en modo gráfico. Luego seleccionar el lenguaje del sistema y zona horaria.



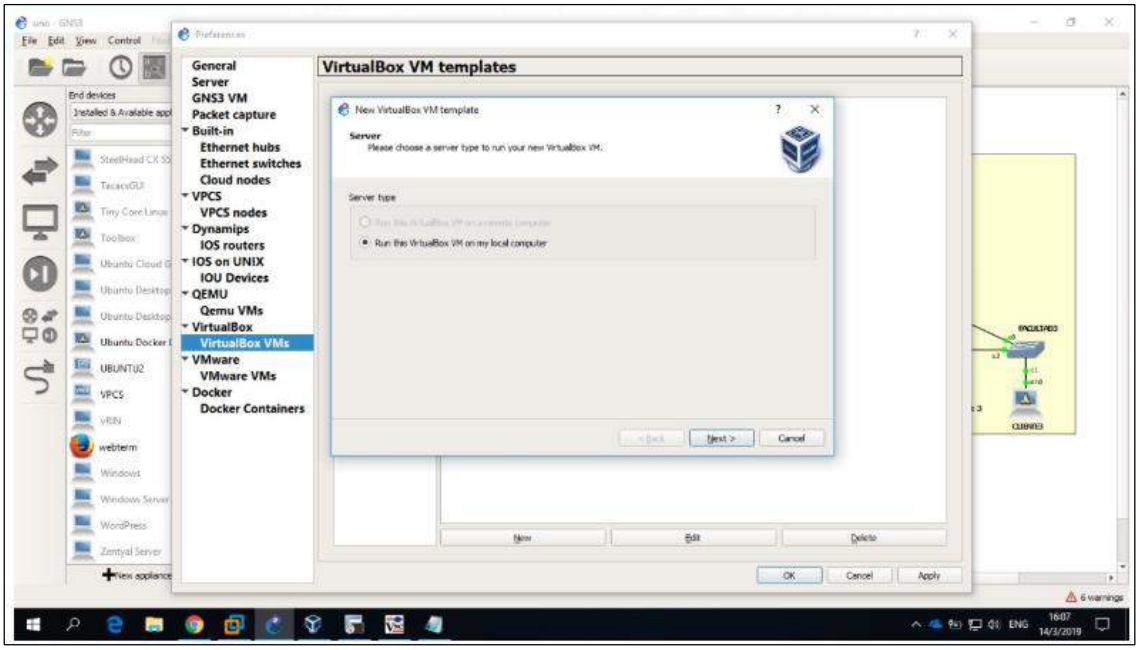
Definir usuario y contraseña para tener más seguridad en el sistema. Y particionar el disco utilizando toda su capacidad y guardar cambios.

Instalar el cargador de arranque de GRUB y reiniciar Kali Linux.

ANEXO G: Integración de elementos de red.

INTEGRACIÓN DEL CONTROLADOR

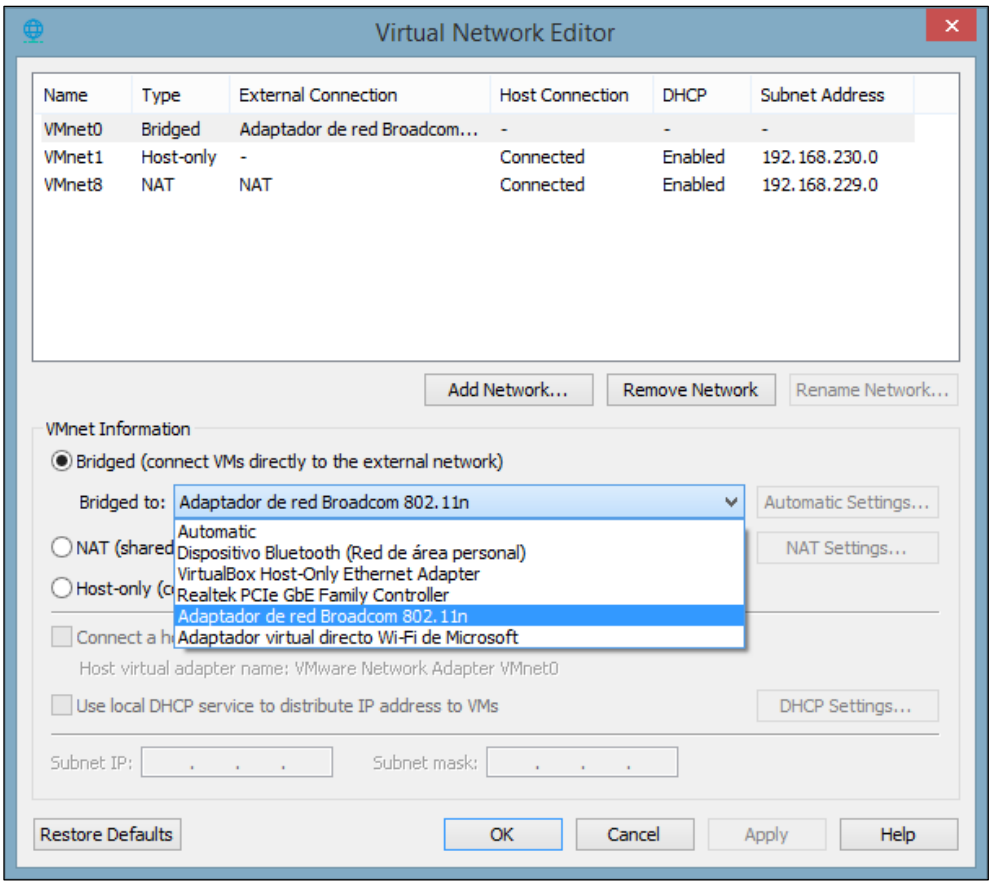
Luego que el controlador se encuentre en funcionamiento se procede a integrarlo al GNS3, para ello se debe importar la imagen del virtualbox al entorno GNS3. En la opción edit, preference, virtualbox VMS como se observa la siguiente figura.



INTEGRACIÓN DE LOS SERVIDORES A GNS3

Para la integración de los servicios, se crea un nuevo adaptador de red en VMware Workstation, donde se configura una nueva interfaz de red en modo puente [bridge], en la sección EDIT –VIRTUAL NETWORK EDITOR una vez ubicados es ese sitio aparece la opción change setting, escoger cualquier adaptador, dependiendo del tipo de conexión (cableada o wifi) y seleccionarlo como bridge para que se conecte directamente con el servidor.

Luego de realizar la configuración de los adaptadores de red en el entorno GNS3, colocar una cloud conectado a la máquina virtual del mismo, la funcionalidad que hará es conectarse los Open vSwitch.



INTEGRACIÓN DE KALI LINUX AL GNS3

Acoplar la PC instalada con Kali Linux junto con la herramienta OpenVas, puesto que de esta herramienta se realizan los ataques. Para ello en la topología creada en GNS3 se debe generar una nube conectada al desktop o maquina física, luego con el uso del cable ethernet de categoría 5 conectar la interfaz física (***eth3***) hacia la máquina que contiene el sistema operativo Kali, cabe recalcar que debe detectar la IP automáticamente por DHCP

INTEGRACIÓN DE LOS CLIENTES DE VOIP AL GNS3

Al igual que el paso anterior se procede a ubicar una nube en la pantalla de trabajo del GNS3 y se configura la interfaz física (eth4), para la comunicación con los clientes de VoIP los cuales se instalan con el cliente Zoiper en las PC, en este caso se configura dicho cliente es el sistema operativo Windows 10 y se procede a obtener una dirección IP automática del servidor DHCP

ANEXO H: Implementación de la metodología OCTAVE.

En este Anexo se detallan todas las especificaciones de los activos de la red SDN: controlador, openvswitch, servicios de red y clientes, y a las supuestas amenazas a los que están propensos.

FASE I:

IDENTIFICACION DE ACTIVOS Y AMENAZAS DE RED

<table>
<tr><th colspan="3">HOJA DE TRABAJO DE LA METODOLOGIA OCTAVE ALLEGRO</th></tr>
<tr><th colspan="3">PERFIL DE ACTIVOS DE LA INFORMACION</th></tr>
<tr><td>ACTIVO CRITICO</td><td colspan="2">Controlador Opendaylight</td></tr>
<tr><td>VERSION</td><td colspan="2">Berillium SR3</td></tr>
<tr><td rowspan="2">REQUERIMIENTOS</td><td>HARDWARE</td><td>SOFTWARE</td></tr>
<tr><td>4 GB de RAM
CPU de 2 núcleos
Memoria de video 12 Mb
Almacenamiento de 18 Gb</td><td>Distribución Ubuntu 14.04
Java 1.8 o superior</td></tr>
<tr><td>DESCRIPCION</td><td colspan="2">Centraliza toda la arquitectura de red y encargado de enviar flujos a los demás dispositivos.</td></tr>
<tr><td>TITULAR DEL ACTIVO</td><td colspan="2">Administrador de red (Daniel)</td></tr>
<tr><td rowspan="6">REQUERIMIENTOS DE SEGURIDAD</td><td colspan="2">DISPONIBILIDAD</td></tr>
<tr><td colspan="2">El controlador debe estar siempre activo y funcional para controlar a toda la red.</td></tr>
<tr><td colspan="2">INTEGRIDAD</td></tr>
<tr><td colspan="2">Solo el administrador es el encargado de manipularlo o configurarlo</td></tr>
<tr><td colspan="2">CONFIDENCIALIDAD</td></tr>
<tr><td colspan="2">Solo el administrador conoce la forma de configuración y administración de la red</td></tr>
<tr><td>AMENAZAS</td><td colspan="2">Puertos abiertos
Canal de comunicación (controlador-switch)no encriptado
Establecimiento y configuración de flujos incorrecta.
Ataques de denegación de servicio.</td></tr>
<tr><td>FECHA</td><td colspan="2">27 de febrero de 2019</td></tr>
<tr><td>REALIZADO POR</td><td colspan="2">Departamento de TI</td></tr>
</table>

HOJA DE TRABAJO DE LA METODOLOGIA OCTAVE ALLEGRO		
PERFIL DE ACTIVOS DE LA INFORMACION		
ACTIVO CRITICO	Switch OVS	
VERSION	Open vSwitch Manager	
REQUERIMIENTOS	**HARDWARE**	**SOFTWARE**
	4 Gb de RAM CPU de 2 núcleos Memoria de video 12 Mb Almacenamiento de 18 Gb	Hipervisor (Secure CRT) GNS3
DESCRIPCION	Conmutador que reenviar los paquetes por la ruta especificada.	
TITULAR DEL ACTIVO	Administrador de red (Daniel)	
REQUERIMIENTOS DE SEGURIDAD	**DISPONIBILIDAD**	
	Debe estar siempre activo y funcional para controlar a toda la red.	
	INTEGRIDAD	
	Solo el administrador es encargado de manipularlo y configurarlo	
	CONFIDENCIALIDAD	
	Solo el administrador conoce la forma de configuración y administración de la red	
AMENAZAS	Programación mal realizada Configuración de flujos de manera incorrecta Ataques de denegación de servicio	
FECHA	27 de febrero de 2019	
REALIZADO POR	Departamento de TI	

HOJA DE TRABAJO DE LA METODOLOGIA OCTAVE ALLEGRO		
PERFIL DE ACTIVOS DE LA INFORMACION		
ACTIVO CRITICO	Cliente Windows	
VERSION	Windows 10	
REQUERIMIENTOS	**HARDWARE**	**SOFTWARE**
	8 Gb de RAM CPU de 4 núcleos Memoria de video 2 Gb Almacenamiento de 512 Gb	S.O Windows 10
DESCRIPCION	Accede a todos los servicios	
TITULAR DEL ACTIVO	Clientes	
AMENAZAS	Virus Troyanos Ataques de denegación de servicio	
FECHA	27 de febrero de 2019	
REALIZADO POR	Departamento de TI	

HOJA DE TRABAJO DE LA METODOLOGIA OCTAVE ALLEGRO		
PERFIL DE ACTIVOS DE LA INFORMACION		
ACTIVO CRITICO	Servidor HTTP	
VERSION	HTTPD 2.4.6	
REQUERIMIENTOS	**HARDWARE**	**SOFTWARE**
	2 Gb de RAM CPU de 2 núcleos Memoria de video Intel G41 Almacenamiento de 500 Gb	Centos 7 Linux Apache
DESCRIPCION	Transfieren datos de hipertexto (páginas web, archivos, aplicaciones). Trabajan bajo el modelo cliente-servidor.	
TITULAR DEL ACTIVO	Administrador de red (Daniel)	
REQUERIMIENTOS DE SEGURIDAD	**DISPONIBILIDAD**	
	La información que se encuentra en el servidor web, debe estar siempre disponible, cuando sea requerida.	
	INTEGRIDAD	
	La información del servidor debe ser confiable y segura.	
	CONFIDENCIALIDAD	
	Todos los usuarios pueden acceder a la información.	
AMENAZAS	Ataques de denegación de servicio Falsas actualizaciones Defectos de software	
FECHA	27 de febrero de 2019	
REALIZADO POR	Departamento de TI	

HOJA DE TRABAJO DE LA METODOLOGIA OCTAVE ALLEGRO		
PERFIL DE ACTIVOS DE LA INFORMACION		
ACTIVO CRITICO	Servidor FTP	
VERSION	VSFTPD 3.0.2-25	
REQUERIMIENTOS	**HARDWARE**	**SOFTWARE**
	2 Gb de RAM CPU de 2 núcleos Memoria de video Intel G41 Almacenamiento de 500 Gb	Centos 7 Linux
DESCRIPCION	Transfiere archivos y documentos remotamente entre cliente y servidor.	
TITULAR DEL ACTIVO	Administrador de red (Daniel)	
REQUERIMIENTOS DE SEGURIDAD	**DISPONIBILIDAD**	
	La información que se encuentra en el servidor web debe estar siempre disponible cuando sea requerida por los clientes.	
	INTEGRIDAD	
	La información del servidor debe ser confiable y segura.	

	CONFIDENCIALIDAD
	Todos los usuarios pueden acceder a la información.
AMENAZAS	Ataques de denegación de servicio Falsas actualizaciones Defectos de software
FECHA	27 de febrero de 2019
REALIZADO POR	Departamento de TI

HOJA DE TRABAJO DE LA METODOLOGIA OCTAVE ALLEGRO		
PERFIL DE ACTIVOS DE LA INFORMACION		
ACTIVO CRITICO	Servidor DNS	
VERSION	NAMED 9.9.4-73	
REQUERIMIENTOS	**HARDWARE**	**SOFTWARE**
	2 Gb de RAM CPU de 2 núcleos Memoria de video Intel G41 Almacenamiento de 500 Gb	Centos 7 Linux BIND 9
DESCRIPCION	Asocian con un nombre de dominio a una dirección IP.	
TITULAR DEL ACTIVO	Administrador de red (Daniel).	
REQUERIMIENTOS DE SEGURIDAD	**DISPONIBILIDAD**	
	La información que se encuentra en el servidor web, debe estar siempre disponible cuando sea requerida por clientes y administradores.	
	INTEGRIDAD	
	La información del servidor debe ser confiable y segura.	
	CONFIDENCIALIDAD	
	Todos los usuarios pueden acceder a la información.	
AMENAZAS	Ataques de denegación de servicio Falsas actualizaciones Defectos de software	
FECHA	27 de febrero de 2019	
REALIZADO POR	Departamento de TI	

HOJA DE TRABAJO DE LA METODOLOGIA OCTAVE ALLEGRO		
PERFIL DE ACTIVOS DE LA INFORMACION		
ACTIVO CRITICO	Servidor DHCP	
VERSION	DHCP 4.2.5-68	
REQUERIMIENTOS	**HARDWARE**	**SOFTWARE**
	2 Gb de RAM CPU de 2 núcleos Memoria de video Intel G41	Centos 7 Linux
DESCRIPCION	Configura dinámicamente las direcciones IP, máscaras de subred y puertas de enlaces de cada dispositivo.	

TITULAR DEL ACTIVO	Administrador de red (Daniel)
REQUERIMIENTOS DE SEGURIDAD	**DISPONIBILIDAD**
	La información que se encuentra en el servidor web, debe estar siempre disponible cuando sea requerida.
	INTEGRIDAD
	La información del servidor debe ser confiable y segura.
	CONFIDENCIALIDAD
	Todos los usuarios pueden acceder a la información.
AMENAZAS	Ataques de denegación de servicio Falsas actualizaciones Defectos de software
FECHA	27 de febrero de 2019
REALIZADO POR	Departamento de TI

HOJA DE TRABAJO DE LA METODOLOGIA OCTAVE ALLEGRO		
PERFIL DE ACTIVOS DE LA INFORMACION		
ACTIVO CRITICO	Servidor VoIP	
VERSION	Asterisk 15.7.2	
REQUERIMIENTOS	**HARDWARE**	**SOFTWARE**
	2 Gb de RAM CPU de 2 núcleos Memoria de video Intel G41 Almacenamiento de 500 Gb	Centos 7 Linux Asterisk
DESCRIPCION	Utiliza el protocolo IP (INTERNET) para transmitir voz en tiempo real.	
TITULAR DEL ACTIVO	Administrador de red (Daniel)	
REQUERIMIENTOS DE SEGURIDAD	**DISPONIBILIDAD**	
	La información que se encuentra en el servidor web, debe estar siempre disponible cuando sea requerida por usuarios y administradores.	
	INTEGRIDAD	
	La información del servidor debe ser confiable y segura.	
	CONFIDENCIALIDAD	
	Todos los usuarios pueden acceder a la información.	
AMENAZAS	Ataques de denegación de servicio Falsas actualizaciones	
FECHA	27 de febrero de 2019	
REALIZADO POR	Departamento de TI	

ANEXO I: Escaneo de vulnerabilidades con OpenVAS

INFORME CONTROLADOR

Scan Report

March 22, 2019

Summary

This document reports on the results of an automatic security scan. All dates are displayed using the timezone "Coordinated Universal Time", which is abbreviated "UTC". The task was "escaneo de vulnerabilidades del controlador con IP 192.168.1.5". The scan started at Thu Mar 21 23:46:29 2019 UTC and ended at Fri Mar 22 00:00:09 2019 UTC. The report first summarises the results found. Then, for each host, the report describes every issue found. Please consider the advice given in each description, in order to rectify the issue.

Contents

1 Result Overview

Host	High	Medium	Low	Log	False Positive
192.168.1.5	0	2	1	0	0
Total: 1	0	2	1	0	0

Vendor security updates are not trusted.
Overrides are on. When a result has an override, this report uses the threat of the override.
Information on overrides is included in the report.
Notes are included in the report.
This report might not show details of all issues that were found.
It only lists hosts that produced issues.
Issues with the threat level "Log" are not shown.
Issues with the threat level "Debug" are not shown.
Issues with the threat level "False Positive" are not shown.
Only results with a minimum QoD of 70 are shown.

This report contains all 3 results selected by the filtering described above. Before filtering there were 29 results.

2 Results per Host

2.1 192.168.1.5

Host scan start	Thu Mar 21 23:46:59 2019 UTC
Host scan end	Fri Mar 22 00:00:07 2019 UTC

Service (Port)	Threat Level
8181/tcp	Medium
8080/tcp	Medium
general/tcp	Low

2.1.1 Medium 8181/tcp

Medium (CVSS: 4.8)
NVT: Cleartext Transmission of Sensitive Information via HTTP

Summary
The host / application transmits sensitive information (username, passwords) in cleartext via HTTP.

Vulnerability Detection Result
`The following URLs requires Basic Authentication (URL:realm name):`

... continues on next page ...

... continued from previous page ...

http://192.168.1.5:8181/auth:"application"

Impact
An attacker could use this situation to compromise or eavesdrop on the HTTP communication between the client and the server using a man-in-the-middle attack to get access to sensitive data like usernames or passwords.

Solution
Solution type: Workaround
Enforce the transmission of sensitive data via an encrypted SSL/TLS connection. Additionally make sure the host / application is redirecting all users to the secured SSL/TLS connection before allowing to input sensitive data into the mentioned functions.

Affected Software/OS
Hosts / applications which doesn't enforce the transmission of sensitive data via an encrypted SSL/TLS connection.

Vulnerability Detection Method
Evaluate previous collected information and check if the host / application is not enforcing the transmission of sensitive data via an encrypted SSL/TLS connection.
The script is currently checking the following:
- HTTP Basic Authentication (Basic Auth)
- HTTP Forms (e.g. Login) with input field of type 'password'
Details: Cleartext Transmission of Sensitive Information via HTTP
OID:1.3.6.1.4.1.25623.1.0.108440
Version used: $Revision: 10726 $

References
Other:
URL:https://www.owasp.org/index.php/Top_10_2013-A2-Broken_Authentication_and_S
↪ession_Management
URL:https://www.owasp.org/index.php/Top_10_2013-A6-Sensitive_Data_Exposure
URL:https://cwe.mitre.org/data/definitions/319.html

[return to 192.168.1.5]

2.1.2 Medium 8080/tcp

Medium (CVSS: 4.8)
NVT: Cleartext Transmission of Sensitive Information via HTTP

Summary
The host / application transmits sensitive information (username, passwords) in cleartext via HTTP.

Vulnerability Detection Result

... continues on next page ...

... continued from previous page ...

The following URLs requires Basic Authentication (URL:realm name):
http://192.168.1.5:8080/auth:"application"

Impact
An attacker could use this situation to compromise or eavesdrop on the HTTP communication between the client and the server using a man-in-the-middle attack to get access to sensitive data like usernames or passwords.

Solution
Solution type: Workaround
Enforce the transmission of sensitive data via an encrypted SSL/TLS connection. Additionally make sure the host / application is redirecting all users to the secured SSL/TLS connection before allowing to input sensitive data into the mentioned functions.

Affected Software/OS
Hosts / applications which doesn't enforce the transmission of sensitive data via an encrypted SSL/TLS connection.

Vulnerability Detection Method
Evaluate previous collected information and check if the host / application is not enforcing the transmission of sensitive data via an encrypted SSL/TLS connection.
The script is currently checking the following:
- HTTP Basic Authentication (Basic Auth)
- HTTP Forms (e.g. Login) with input field of type 'password'
Details: Cleartext Transmission of Sensitive Information via HTTP
OID:1.3.6.1.4.1.25623.1.0.108440
Version used: $Revision: 10726 $

References
Other:
URL:https://www.owasp.org/index.php/Top_10_2013-A2-Broken_Authentication_and_S
↪ession_Management
URL:https://www.owasp.org/index.php/Top_10_2013-A6-Sensitive_Data_Exposure
URL:https://cwe.mitre.org/data/definitions/319.html

[return to 192.168.1.5]

2.1.3 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

Summary
The remote host implements TCP timestamps and therefore allows to compute the uptime.

Vulnerability Detection Result

... continues on next page ...

... continued from previous page ...

It was detected that the host implements RFC1323.
The following timestamps were retrieved with a delay of 1 seconds in-between:
Packet 1: 2650759
Packet 2: 2651035

Impact
A side effect of this feature is that the uptime of the remote host can sometimes be computed.

Solution
Solution type: Mitigation
To disable TCP timestamps on linux add the line 'net.ipv4.tcp_timestamps = 0' to /etc/sysctl.conf. Execute 'sysctl -p' to apply the settings at runtime.
To disable TCP timestamps on Windows execute 'netsh int tcp set global timestamps=disabled'
Starting with Windows Server 2008 and Vista, the timestamp can not be completely disabled.
The default behavior of the TCP/IP stack on this Systems is to not use the Timestamp options when initiating TCP connections, but use them if the TCP peer that is initiating communication includes them in their synchronize (SYN) segment.
See also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

Affected Software/OS
TCP/IPv4 implementations that implement RFC1323.

Vulnerability Insight
The remote host implements TCP timestamps, as defined by RFC1323.

Vulnerability Detection Method
Special IP packets are forged and sent with a little delay in between to the target IP. The responses are searched for a timestamps. If found, the timestamps are reported.
Details: TCP timestamps
OID:1.3.6.1.4.1.25623.1.0.80091
Version used: $Revision: 10411 $

References
Other:
URL:http://www.ietf.org/rfc/rfc1323.txt

INFORME DEL OVS

Scan Report

March 22, 2019

Summary

This document reports on the results of an automatic security scan. All dates are displayed using the timezone "Coordinated Universal Time", which is abbreviated "UTC". The task was "escaneo de vulnerabilidades del OVS con IP 192.168.1.10". The scan started at Fri Mar 22 01:08:39 2019 UTC and ended at Fri Mar 22 01:17:04 2019 UTC. The report first summarises the results found. Then, for each host, the report describes every issue found. Please consider the advice given in each description, in order to rectify the issue.

Contents

1 Result Overview

Host	High	Medium	Low	Log	False Positive
Total: 0	0	0	0	0	0

Vendor security updates are not trusted.
Overrides are on. When a result has an override, this report uses the threat of the override.
Information on overrides is included in the report.
Notes are included in the report.
This report might not show details of all issues that were found.
It only lists hosts that produced issues.
Issues with the threat level "Log" are not shown.
Issues with the threat level "Debug" are not shown.
Issues with the threat level "False Positive" are not shown.
Only results with a minimum QoD of 70 are shown.

This report contains 0 results. Before filtering there were 4 results.

2 Results per Host

INFORME DE LOS USUARIOS

Scan Report

March 22, 2019

Summary

This document reports on the results of an automatic security scan. All dates are displayed using the timezone "Coordinated Universal Time", which is abbreviated "UTC". The task was "escaneo de vulnerabilidades del cliente 192.168.1.131". The scan started at Thu Mar 21 01:27:33 2019 UTC and ended at Thu Mar 21 01:49:17 2019 UTC. The report first summarises the results found. Then, for each host, the report describes every issue found. Please consider the advice given in each description, in order to rectify the issue.

Contents

1 Result Overview

Host	High	Medium	Low	Log	False Positive
192.168.1.131	2	4	1	0	0
Total: 1	2	4	1	0	0

Vendor security updates are not trusted.
Overrides are on. When a result has an override, this report uses the threat of the override.
Information on overrides is included in the report.
Notes are included in the report.
This report might not show details of all issues that were found.
It only lists hosts that produced issues.
Issues with the threat level "Log" are not shown.
Issues with the threat level "Debug" are not shown.
Issues with the threat level "False Positive" are not shown.
Only results with a minimum QoD of 70 are shown.

This report contains all 7 results selected by the filtering described above. Before filtering there were 46 results.

1.1 Host Authentications

Host	Protocol	Result	Port/User
192.168.1.131	SMB	Success	Protocol SMB, Port 445, User

2 Results per Host

2.1 192.168.1.131

Host scan start	Thu Mar 21 01:28:00 2019 UTC
Host scan end	Thu Mar 21 01:49:17 2019 UTC

Service (Port)	Threat Level
445/tcp	High
3389/tcp	Medium
135/tcp	Medium
general/tcp	Low

2.1.1 High 445/tcp

... continues on next page ...

... continued from previous page ...

High (CVSS: 9.3)
NVT: Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)

Summary
This host is missing a critical security update according to Microsoft Bulletin MS17-010.

Vulnerability Detection Result
Vulnerability was detected according to the Vulnerability Detection Method.

Impact
Successful exploitation will allow remote attackers to gain the ability to execute code on the target server, also could lead to information disclosure from the server.

Solution
Solution type: VendorFix
Run Windows Update and update the listed hotfixes or download and update mentioned hotfixes in the advisory

Affected Software/OS
Microsoft Windows 10 x32/x64 Edition Microsoft Windows Server 2012 Edition Microsoft Windows Server 2016 Microsoft Windows 8.1 x32/x64 Edition Microsoft Windows Server 2012 R2 Edition Microsoft Windows 7 x32/x64 Edition Service Pack 1 Microsoft Windows Vista x32/x64 Edition Service Pack 2 Microsoft Windows Server 2008 R2 x64 Edition Service Pack 1 Microsoft Windows Server 2008 x32/x64 Edition Service Pack 2

Vulnerability Insight
Multiple flaws exist due to the way that the Microsoft Server Message Block 1.0 (SMBv1) server handles certain requests.

Vulnerability Detection Method
Send the crafted SMB transaction request with fid = 0 and check the response to confirm the vulnerability.
Details: `Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)`
OID:1.3.6.1.4.1.25623.1.0.810676
Version used: `$Revision: 11874 $`

References
```
CVE: CVE-2017-0143, CVE-2017-0144, CVE-2017-0145, CVE-2017-0146, CVE-2017-0147,
↪CVE-2017-0148
BID:96703, 96704, 96705, 96707, 96709, 96706
Other:
  URL:https://support.microsoft.com/en-in/kb/4013078
   URL:https://technet.microsoft.com/library/security/MS17-010
   URL:https://github.com/rapid7/metasploit-framework/pull/8167/files
```

High (CVSS: 7.5)
NVT: Microsoft Windows SMB/NETBIOS NULL Session Authentication Bypass Vulnerability

Summary
The host is running SMB/NETBIOS and prone to an authentication bypass vulnerability

Vulnerability Detection Result
`It was possible to login at the share 'IPC$' with an empty login and password.`

Impact
Successful exploitation could allow attackers to use shares to cause the system to crash.

Solution
Solution type: WillNotFix
No known solution was made available for at least one year since the disclosure of this vulnerability. Likely none will be provided anymore. General solution options are to upgrade to a newer release, disable respective features, remove the product or replace the product by another one.
A workaround is to,
- Disable null session login.
- Remove the share.
- Enable passwords on the share.

Affected Software/OS
Microsoft Windows 95,
Microsoft Windows 98,
Microsoft Windows NT.
Other Windows implementations / versions might be affected as well.

Vulnerability Insight
The flaw is due to an SMB share, allows full access to Guest users. If the Guest account is enabled, anyone can access the computer without a valid user account or password.

Vulnerability Detection Method
Details: `Microsoft Windows SMB/NETBIOS NULL Session Authentication Bypass Vulnerability`
OID:1.3.6.1.4.1.25623.1.0.801991
Version used: `$Revision: 11997 $`

References
`CVE: CVE-1999-0519`
`Other:`
`URL:http://xforce.iss.net/xforce/xfdb/2`
`URL:http://seclab.cs.ucdavis.edu/projects/testing/vulner/38.html`

[return to 192.168.1.131]

2.1.2 Medium 3389/tcp

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

Summary
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

Vulnerability Detection Result
```
'Weak' cipher suites accepted by this service via the TLSv1.0 protocol:
TLS_RSA_WITH_RC4_128_MD5
TLS_RSA_WITH_RC4_128_SHA
'Weak' cipher suites accepted by this service via the TLSv1.1 protocol:
TLS_RSA_WITH_RC4_128_MD5
TLS_RSA_WITH_RC4_128_SHA
'Weak' cipher suites accepted by this service via the TLSv1.2 protocol:
TLS_RSA_WITH_RC4_128_MD5
TLS_RSA_WITH_RC4_128_SHA
```

Solution
Solution type: Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

Vulnerability Insight
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

Vulnerability Detection Method
Details: SSL/TLS: Report Weak Cipher Suites
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: $Revision: 11135 $

References
```
CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000
Other:
  URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-
↪1465_update_6.html
   URL:https://bettercrypto.org/
   URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/
```

Medium (CVSS: 4.0)
NVT: SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerability

Summary
The SSL/TLS service uses Diffie-Hellman groups with insufficient strength (key size < 2048).

Vulnerability Detection Result
`Server Temporary Key Size: 1024 bits`

Impact
An attacker might be able to decrypt the SSL/TLS communication offline.

Solution
Solution type: Workaround
Deploy (Ephemeral) Elliptic-Curve Diffie-Hellman (ECDHE) or use a 2048-bit or stronger Diffie-Hellman group (see the references).
For Apache Web Servers: Beginning with version 2.4.7, mod_ssl will use DH parameters which include primes with lengths of more than 1024 bits.

Vulnerability Insight
The Diffie-Hellman group are some big numbers that are used as base for the DH computations. They can be, and often are, fixed. The security of the final secret depends on the size of these parameters. It was found that 512 and 768 bits to be weak, 1024 bits to be breakable by really powerful attackers like governments.

Vulnerability Detection Method
Checks the DHE temporary public key size.
Details: `SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerabili.`
`↪..`
OID:1.3.6.1.4.1.25623.1.0.106223
Version used: `$Revision: 12865 $`

References
```
Other:
  URL:https://weakdh.org/
   URL:https://weakdh.org/sysadmin.html
```

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

Summary
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

Vulnerability Detection Result
`The following certificates are part of the certificate chain but using insecure`
`↪signature algorithms:`

... continues on next page ...

... continued from previous page ...

Subject:	CN=Daniela
Signature Algorithm:	sha1WithRSAEncryption

Solution
Solution type: Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

Vulnerability Insight
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

Vulnerability Detection Method
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm
OID:1.3.6.1.4.1.25623.1.0.105880
Version used: $Revision: 8810 $

References
Other:
URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with-sha-1-based-signature-algorithms/

[return to 192.168.1.131]

2.1.3 Medium 135/tcp

Medium (CVSS: 5.0)
NVT: DCE/RPC and MSRPC Services Enumeration Reporting

Summary

... continues on next page ...

... continued from previous page ...

Distributed Computing Environment / Remote Procedure Calls (DCE/RPC) or MSRPC services running on the remote host can be enumerated by connecting on port 135 and doing the appropriate queries.

Vulnerability Detection Result

```
Here is the list of DCE/RPC or MSRPC services running on this host via the TCP p
↪rotocol:
Port: 49152/tcp
     UUID: d95afe70-a6d5-4259-822e-2c84da1ddb0d, version 1
     Endpoint: ncacn_ip_tcp:192.168.1.131[49152]
Port: 49153/tcp
     UUID: 06bba54a-be05-49f9-b0a0-30f790261023, version 1
     Endpoint: ncacn_ip_tcp:192.168.1.131[49153]
     Annotation: Security Center
     UUID: 30adc50c-5cbc-46ce-9a0e-91914789e23c, version 1
     Endpoint: ncacn_ip_tcp:192.168.1.131[49153]
     Annotation: NRP server endpoint
     UUID: 3c4728c5-f0ab-448b-bda1-6ce01eb0a6d5, version 1
     Endpoint: ncacn_ip_tcp:192.168.1.131[49153]
     Annotation: DHCP Client LRPC Endpoint
     UUID: 3c4728c5-f0ab-448b-bda1-6ce01eb0a6d6, version 1
     Endpoint: ncacn_ip_tcp:192.168.1.131[49153]
     Annotation: DHCPv6 Client LRPC Endpoint
     UUID: abfb6ca3-0c5e-4734-9285-0aee72fe8d1c, version 1
     Endpoint: ncacn_ip_tcp:192.168.1.131[49153]
     Annotation: Wcm Service
     UUID: f6beaff7-1e19-4fbb-9f8f-b89e2018337c, version 1
     Endpoint: ncacn_ip_tcp:192.168.1.131[49153]
     Annotation: Event log TCPIP
Port: 49154/tcp
     UUID: 1a0d010f-1c33-432c-b0f5-8cf4e8053099, version 1
     Endpoint: ncacn_ip_tcp:192.168.1.131[49154]
     Annotation: IdSegSrv service
     UUID: 201ef99a-7fa0-444c-9399-19ba84f12a1a, version 1
     Endpoint: ncacn_ip_tcp:192.168.1.131[49154]
     Annotation: AppInfo
     UUID: 2e6035b2-e8f1-41a7-a044-656b439c4c34, version 1
     Endpoint: ncacn_ip_tcp:192.168.1.131[49154]
     Annotation: Proxy Manager provider server endpoint
     UUID: 30b044a5-a225-43f0-b3a4-e060df91f9c1, version 1
     Endpoint: ncacn_ip_tcp:192.168.1.131[49154]
     UUID: 3a9ef155-691d-4449-8d05-09ad57031823, version 1
     Endpoint: ncacn_ip_tcp:192.168.1.131[49154]
     UUID: 552d076a-cb29-4e44-8b6a-d15e59e2c0af, version 1
     Endpoint: ncacn_ip_tcp:192.168.1.131[49154]
     Annotation: IP Transition Configuration endpoint
     UUID: 58e604e8-9adb-4d2e-a464-3b0683fb1480, version 1
```

... continues on next page ...

... continued from previous page ...

```
      Endpoint: ncacn_ip_tcp:192.168.1.131[49154]
      Annotation: AppInfo
      UUID: 5f54ce7d-5b79-4175-8584-cb65313a0e98, version 1
      Endpoint: ncacn_ip_tcp:192.168.1.131[49154]
      Annotation: AppInfo
      UUID: 86d35949-83c9-4044-b424-db363231fd0c, version 1
      Endpoint: ncacn_ip_tcp:192.168.1.131[49154]
      UUID: 98716d03-89ac-44c7-bb8c-285824e51c4a, version 1
      Endpoint: ncacn_ip_tcp:192.168.1.131[49154]
      Annotation: XactSrv service
      UUID: a398e520-d59a-4bdd-aa7a-3c1e0303a511, version 1
      Endpoint: ncacn_ip_tcp:192.168.1.131[49154]
      Annotation: IKE/Authip API
      UUID: c36be077-e14b-4fe9-8abc-e856ef4f048b, version 1
      Endpoint: ncacn_ip_tcp:192.168.1.131[49154]
      Annotation: Proxy Manager client server endpoint
      UUID: c49a5a70-8a7f-4e70-ba16-1e8f1f193ef1, version 1
      Endpoint: ncacn_ip_tcp:192.168.1.131[49154]
      Annotation: Adh APIs
      UUID: c9ac6db5-82b7-4e55-ae8a-e464ed7b4277, version 1
      Endpoint: ncacn_ip_tcp:192.168.1.131[49154]
      Annotation: Impl friendly name
      UUID: fd7a0523-dc70-43dd-9b2e-9c5ed48225b1, version 1
      Endpoint: ncacn_ip_tcp:192.168.1.131[49154]
      Annotation: AppInfo
Port: 49155/tcp
      UUID: 0b6edbfa-4a24-4fc6-8a23-942b1eca65d1, version 1
      Endpoint: ncacn_ip_tcp:192.168.1.131[49155]
      UUID: 12345678-1234-abcd-ef00-0123456789ab, version 1
      Endpoint: ncacn_ip_tcp:192.168.1.131[49155]
      Named pipe : spoolss
      Win32 service or process : spoolsv.exe
      Description : Spooler service
      UUID: 4a452661-8290-4b36-8fbe-7f4093a94978, version 1
      Endpoint: ncacn_ip_tcp:192.168.1.131[49155]
      UUID: 76f03f96-cdfd-44fc-a22c-64950a001209, version 1
      Endpoint: ncacn_ip_tcp:192.168.1.131[49155]
      UUID: ae33069b-a2a8-46ee-a235-ddfd339be281, version 1
      Endpoint: ncacn_ip_tcp:192.168.1.131[49155]
Port: 49156/tcp
      UUID: 12345778-1234-abcd-ef00-0123456789ac, version 1
      Endpoint: ncacn_ip_tcp:192.168.1.131[49156]
      Named pipe : lsass
      Win32 service or process : lsass.exe
      Description : SAM access
      UUID: b25a52bf-e5dd-4f4a-aea6-8ca7272a0e86, version 2
      Endpoint: ncacn_ip_tcp:192.168.1.131[49156]
```

... continues on next page ...

... continued from previous page ...

```
     Annotation: KeyIso
Port: 49157/tcp
     UUID: 367abb81-9844-35f1-ad32-98f038001003, version 2
     Endpoint: ncacn_ip_tcp:192.168.1.131[49157]
Port: 49158/tcp
     UUID: 6b5bdd1e-528c-422c-af8c-a4079be4fe48, version 1
     Endpoint: ncacn_ip_tcp:192.168.1.131[49158]
     Annotation: Remote Fw APIs
Note: DCE/RPC or MSRPC services running on this host locally were identified. Re
↪porting this list is not enabled by default due to the possible large size of
↪this list. See the script preferences to enable this reporting.
```

Impact
An attacker may use this fact to gain more knowledge about the remote host.

Solution
Solution type: Mitigation
Filter incoming traffic to this ports.

Vulnerability Detection Method
Details: DCE/RPC and MSRPC Services Enumeration Reporting
OID:1.3.6.1.4.1.25623.1.0.10736
Version used: $Revision: 6319 $

[return to 192.168.1.131]

2.1.4 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

Summary
The remote host implements TCP timestamps and therefore allows to compute the uptime.

Vulnerability Detection Result

```
It was detected that the host implements RFC1323.
The following timestamps were retrieved with a delay of 1 seconds in-between:
Packet 1: 16519984
Packet 2: 16520094
```

Impact
A side effect of this feature is that the uptime of the remote host can sometimes be computed.

Solution
Solution type: Mitigation

... continues on next page ...

... continued from previous page ...

To disable TCP timestamps on linux add the line 'net.ipv4.tcp_timestamps = 0' to /etc/sysctl.conf. Execute 'sysctl -p' to apply the settings at runtime.
To disable TCP timestamps on Windows execute 'netsh int tcp set global timestamps=disabled' Starting with Windows Server 2008 and Vista, the timestamp can not be completely disabled.
The default behavior of the TCP/IP stack on this Systems is to not use the Timestamp options when initiating TCP connections, but use them if the TCP peer that is initiating communication includes them in their synchronize (SYN) segment.
See also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

Affected Software/OS
TCP/IPv4 implementations that implement RFC1323.

Vulnerability Insight
The remote host implements TCP timestamps, as defined by RFC1323.

Vulnerability Detection Method
Special IP packets are forged and sent with a little delay in between to the target IP. The responses are searched for a timestamps. If found, the timestamps are reported.
Details: TCP timestamps
OID:1.3.6.1.4.1.25623.1.0.80091
Version used: $Revision: 10411 $

References
Other:
URL:http://www.ietf.org/rfc/rfc1323.txt

| [illegible] |

INFORME DE LOS SERVICIOS

Scan Report

March 22, 2019

Summary

This document reports on the results of an automatic security scan. All dates are displayed using the timezone "Coordinated Universal Time", which is abbreviated "UTC". The task was "escaneo de vulnerabilidades del servidor con IP 192.168.1.8". The scan started at Thu Mar 21 01:49:53 2019 UTC and ended at Thu Mar 21 02:10:09 2019 UTC. The report first summarises the results found. Then, for each host, the report describes every issue found. Please consider the advice given in each description, in order to rectify the issue.

Contents

1 Result Overview

Host	High	Medium	Low	Log	False Positive
192.168.1.8 web.tesis.dos.com	1	4	1	0	0
Total: 1	1	4	1	0	0

Vendor security updates are not trusted.
Overrides are on. When a result has an override, this report uses the threat of the override.
Information on overrides is included in the report.
Notes are included in the report.
This report might not show details of all issues that were found.
It only lists hosts that produced issues.
Issues with the threat level "Log" are not shown.
Issues with the threat level "Debug" are not shown.
Issues with the threat level "False Positive" are not shown.
Only results with a minimum QoD of 70 are shown.

This report contains all 6 results selected by the filtering described above. Before filtering there were 72 results.

2 Results per Host

2.1 192.168.1.8

Host scan start	Thu Mar 21 01:50:20 2019 UTC
Host scan end	Thu Mar 21 02:10:08 2019 UTC

Service (Port)	Threat Level
80/tcp	High
22/tcp	Medium
80/tcp	Medium
21/tcp	Medium
general/tcp	Low

2.1.1 High 80/tcp

High (CVSS: 9.0)
NVT: HTTP Brute Force Logins With Default Credentials Reporting

Summary
It was possible to login into the remote Web Application using default credentials.

... continues on next page ...

... continued from previous page ...

As the NVT 'HTTP Brute Force Logins with default Credentials' (OID: 1.3.6.1.4.1.25623.1.0.108041) might run into a timeout the actual reporting of this vulnerability takes place in this NVT instead. The script preference 'Report timeout' allows you to configure if such an timeout is reported.

Vulnerability Detection Result

```
It was possible to login with the following credentials <Url>:<User>:<Password>:
↪<HTTP status code>
http://web.tesis.dos.com/:admin:admin:HTTP/1.1 200 OK
```

Solution

Solution type: Mitigation

Change the password as soon as possible.

Vulnerability Detection Method

Try to login with a number of known default credentials via HTTP Basic Auth.

Details: `HTTP Brute Force Logins With Default Credentials Reporting`

OID:1.3.6.1.4.1.25623.1.0.103240

Version used: `$Revision: 11663 $`

[return to 192.168.1.8]

2.1.2 Medium 22/tcp

Medium (CVSS: 4.3)

NVT: SSH Weak Encryption Algorithms Supported

Summary

The remote SSH server is configured to allow weak encryption algorithms.

Vulnerability Detection Result

```
The following weak client-to-server encryption algorithms are supported by the r
↪emote service:
3des-cbc
aes128-cbc
aes192-cbc
aes256-cbc
blowfish-cbc
cast128-cbc
The following weak server-to-client encryption algorithms are supported by the r
↪emote service:
3des-cbc
aes128-cbc
aes192-cbc
aes256-cbc
blowfish-cbc
```

... continues on next page ...

... continued from previous page ...

cast128-cbc

Solution
Solution type: Mitigation
Disable the weak encryption algorithms.

Vulnerability Insight
The 'arcfour' cipher is the Arcfour stream cipher with 128-bit keys. The Arcfour cipher is believed to be compatible with the RC4 cipher [SCHNEIER]. Arcfour (and RC4) has problems with weak keys, and should not be used anymore.
The 'none' algorithm specifies that no encryption is to be done. Note that this method provides no confidentiality protection, and it is NOT RECOMMENDED to use it.
A vulnerability exists in SSH messages that employ CBC mode that may allow an attacker to recover plaintext from a block of ciphertext.

Vulnerability Detection Method
Check if remote ssh service supports Arcfour, none or CBC ciphers.
Details: SSH Weak Encryption Algorithms Supported
OID:1.3.6.1.4.1.25623.1.0.105611
Version used: $Revision: 13581 $

References
Other:
URL:https://tools.ietf.org/html/rfc4253#section-6.3
URL:https://www.kb.cert.org/vuls/id/958563

[return to 192.168.1.8]

2.1.3 Medium 80/tcp

Medium (CVSS: 5.8)
NVT: HTTP Debugging Methods (TRACE/TRACK) Enabled

Summary
Debugging functions are enabled on the remote web server.
The remote web server supports the TRACE and/or TRACK methods. TRACE and TRACK are HTTP methods which are used to debug web server connections.

Vulnerability Detection Result
The web server has the following HTTP methods enabled: TRACE

Impact
An attacker may use this flaw to trick your legitimate web users to give him their credentials.

Solution
Solution type: Mitigation

... continues on next page ...

... continued from previous page ...

Disable the TRACE and TRACK methods in your web server configuration.
Please see the manual of your web server or the references for more information.

Affected Software/OS
Web servers with enabled TRACE and/or TRACK methods.

Vulnerability Insight
It has been shown that web servers supporting this methods are subject to cross-site-scripting attacks, dubbed XST for Cross-Site-Tracing, when used in conjunction with various weaknesses in browsers.

Vulnerability Detection Method
Details: `HTTP Debugging Methods (TRACE/TRACK) Enabled`
OID:1.3.6.1.4.1.25623.1.0.11213
Version used: `$Revision: 10828 $`

References
```
CVE: CVE-2003-1567, CVE-2004-2320, CVE-2004-2763, CVE-2005-3398, CVE-2006-4683,
↪CVE-2007-3008, CVE-2008-7253, CVE-2009-2823, CVE-2010-0386, CVE-2012-2223, CVE
↪-2014-7883
BID:9506, 9561, 11604, 15222, 19915, 24456, 33374, 36956, 36990, 37995
Other:
  URL:http://www.kb.cert.org/vuls/id/288308
   URL:http://www.kb.cert.org/vuls/id/867593
   URL:http://httpd.apache.org/docs/current/de/mod/core.html#traceenable
   URL:https://www.owasp.org/index.php/Cross_Site_Tracing
```

Medium (CVSS: 4.8)
NVT: Cleartext Transmission of Sensitive Information via HTTP

Summary
The host / application transmits sensitive information (username, passwords) in cleartext via HTTP.

Vulnerability Detection Result
```
The following URLs requires Basic Authentication (URL:realm name):
http://web.tesis.dos.com/:"ingrese sus datos"
```

Impact
An attacker could use this situation to compromise or eavesdrop on the HTTP communication between the client and the server using a man-in-the-middle attack to get access to sensitive data like usernames or passwords.

Solution
Solution type: Workaround

... continues on next page ...

... continued from previous page ...

Enforce the transmission of sensitive data via an encrypted SSL/TLS connection. Additionally make sure the host / application is redirecting all users to the secured SSL/TLS connection before allowing to input sensitive data into the mentioned functions.

Affected Software/OS
Hosts / applications which doesn't enforce the transmission of sensitive data via an encrypted SSL/TLS connection.

Vulnerability Detection Method
Evaluate previous collected information and check if the host / application is not enforcing the transmission of sensitive data via an encrypted SSL/TLS connection.
The script is currently checking the following:
- HTTP Basic Authentication (Basic Auth)
- HTTP Forms (e.g. Login) with input field of type 'password'
Details: `Cleartext Transmission of Sensitive Information via HTTP`
OID:1.3.6.1.4.1.25623.1.0.108440
Version used: `$Revision: 10726 $`

References
```
Other:
  URL:https://www.owasp.org/index.php/Top_10_2013-A2-Broken_Authentication_and_S
↪ession_Management
   URL:https://www.owasp.org/index.php/Top_10_2013-A6-Sensitive_Data_Exposure
   URL:https://cwe.mitre.org/data/definitions/319.html
```

[return to 192.168.1.8]

2.1.4 Medium 21/tcp

Medium (CVSS: 4.8)
NVT: FTP Unencrypted Cleartext Login

Summary
The remote host is running a FTP service that allows cleartext logins over unencrypted connections.

Vulnerability Detection Result
```
The remote FTP service accepts logins without a previous sent 'AUTH TLS' command
↪. Response(s):
Anonymous sessions:     331 Please specify the password.
Non-anonymous sessions: 331 Please specify the password.
```

Impact
An attacker can uncover login names and passwords by sniffing traffic to the FTP service.

Solution

... continues on next page ...

... continued from previous page ...

Solution type: Mitigation
Enable FTPS or enforce the connection via the 'AUTH TLS' command. Please see the manual of the FTP service for more information.

Vulnerability Detection Method
Tries to login to a non FTPS enabled FTP service without sending a 'AUTH TLS' command first and checks if the service is accepting the login without enforcing the use of the 'AUTH TLS' command.
Details: `FTP Unencrypted Cleartext Login`
OID:1.3.6.1.4.1.25623.1.0.108528
Version used: `$Revision: 13611 $`

[return to 192.168.1.8]

2.1.5 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

Summary
The remote host implements TCP timestamps and therefore allows to compute the uptime.

Vulnerability Detection Result
```
It was detected that the host implements RFC1323.
The following timestamps were retrieved with a delay of 1 seconds in-between:
Packet 1: 6215825
Packet 2: 6216942
```

Impact
A side effect of this feature is that the uptime of the remote host can sometimes be computed.

Solution
Solution type: Mitigation
To disable TCP timestamps on linux add the line 'net.ipv4.tcp_timestamps = 0' to /etc/sysctl.conf. Execute 'sysctl -p' to apply the settings at runtime.
To disable TCP timestamps on Windows execute 'netsh int tcp set global timestamps=disabled' Starting with Windows Server 2008 and Vista, the timestamp can not be completely disabled.
The default behavior of the TCP/IP stack on this Systems is to not use the Timestamp options when initiating TCP connections, but use them if the TCP peer that is initiating communication includes them in their synchronize (SYN) segment.
See also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

Affected Software/OS
TCP/IPv4 implementations that implement RFC1323.

Vulnerability Insight

... continues on next page ...

... continued from previous page ...

The remote host implements TCP timestamps, as defined by RFC1323.

Vulnerability Detection Method
Special IP packets are forged and sent with a little delay in between to the target IP. The responses are searched for a timestamps. If found, the timestamps are reported.
Details: `TCP timestamps`
OID:1.3.6.1.4.1.25623.1.0.80091
Version used: `$Revision: 10411 $`

References
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[return to 192.168.1.8]

FASE II:

DETECCION DE VULNERABILIDADES

HOJA DE TRABAJO DE LA METODOLOGIA OCTAVE ALLEGRO	
VULNERABILIDADES DETECTADAS DE LOS ACTIVOS DE INFORMACIÓN	
ACTIVO CRITICO	CONTROLADOR
AREA DE PREOCUPACIÓN	Exponga la información de configuración Acceso no autorizado
ACTOR	Personal interno
MEDIO DE ACCESO	Ingreso al controlador con acceso de credenciales y autenticación de los administradores.
MOTIVOS	Intereses personales Revelar información administrativa
RESULTADO	Interrumpen el servicio o incluso modificar la información.
VULNERABILIDADES DETECTADAS	HTTP (puerto 8080 y 8181) DIRB (NSAL wrapper) ICMP Timestamp Detection Services

HOJA DE TRABAJO DE LA METODOLOGIA OCTAVE ALLEGRO	
VULNERABILIDADES DETECTADAS DE LOS ACTIVOS DE INFORMACIÓN	
ACTIVO CRITICO	OPEN VSWITCH
AREA DE PREOCUPACIÓN	Exponga la información de configuración Acceso no autorizado
ACTOR	Personal interno
MEDIO DE ACCESO	Ingreso al switch con acceso de credenciales y autenticación de los administradores.
MOTIVOS	Intereses personales Revelar información administrativa
RESULTADO	Interrumpen el servicio o incluso modificar la información.
VULNERABILIDADES DETECTADAS	ICMP Timestamp Detection Protocolo TCP

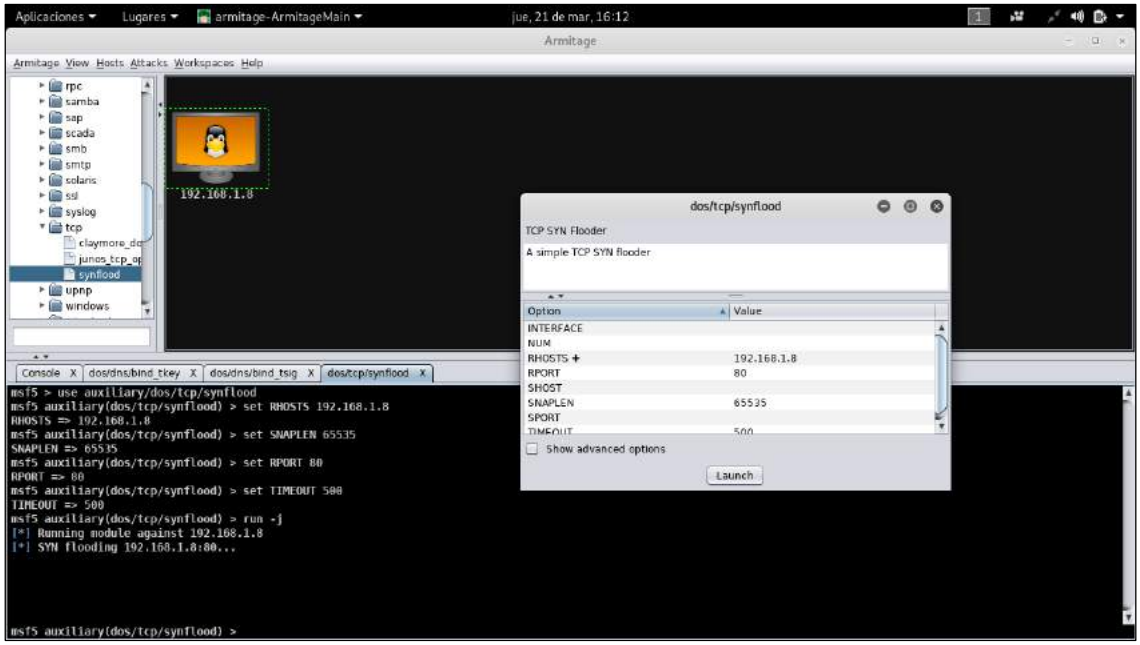
HOJA DE TRABAJO DE LA METODOLOGIA OCTAVE ALLEGRO	
VULNERABILIDADES DETECTADAS DE LOS ACTIVOS DE INFORMACIÓN	
ACTIVO CRITICO	SERVIDORES (DHCP, HTTP, FTP, DNS y VoIP)
AREA DE PREOCUPACIÓN	Interrupción del servicio. Falla en actualizaciones

ACTOR	Personal interno
MEDIO DE ACCESO	Ingreso a los servidores con credenciales y autenticación
MOTIVOS	Intereses personales Problemas en la comunicación
RESULTADO	Interrumpen el servicio o incluso modificar la información.
VULNERABILIDADES DETECTADAS	Protocolo SSH Problemas con el servidor DNS Problemas con el servidor FTP Problemas con el servidor HTTP ICMP Timestamp Detection

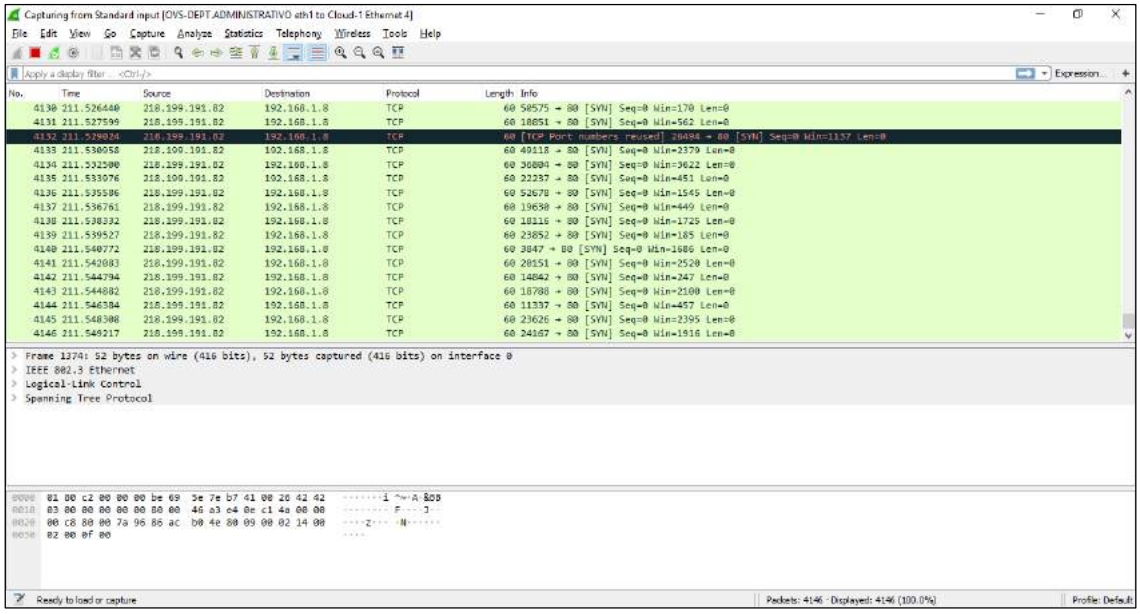
ANEXO J: Ejecución de ataques de denegación de servicio.

Ataque TCP SYN

Para ejecutar el ataque TCP SYN, se deben configurar los siguientes parámetros, detallados en la figura.



La figura siguiente muestra ataques al puerto 80 de la maquina con IP 192.168.1.8, desde la dirección 218.199.191.82.

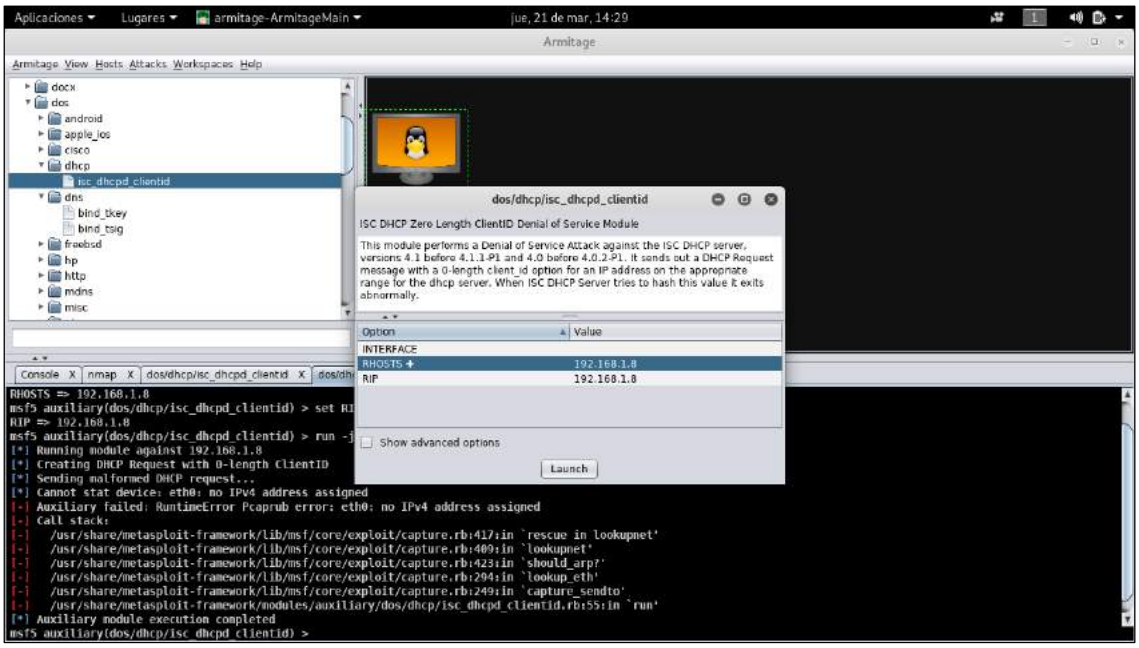


Enmascarando su dirección original para no exponer su origen y evitar ser rastreados. Cuando se realiza este ataque, el servidor deja de trabajar de manera normal o incluso puede dejar de funcionar debido, a que al llegar más cantidad de paquetes el procesamiento debe ser más rápido.

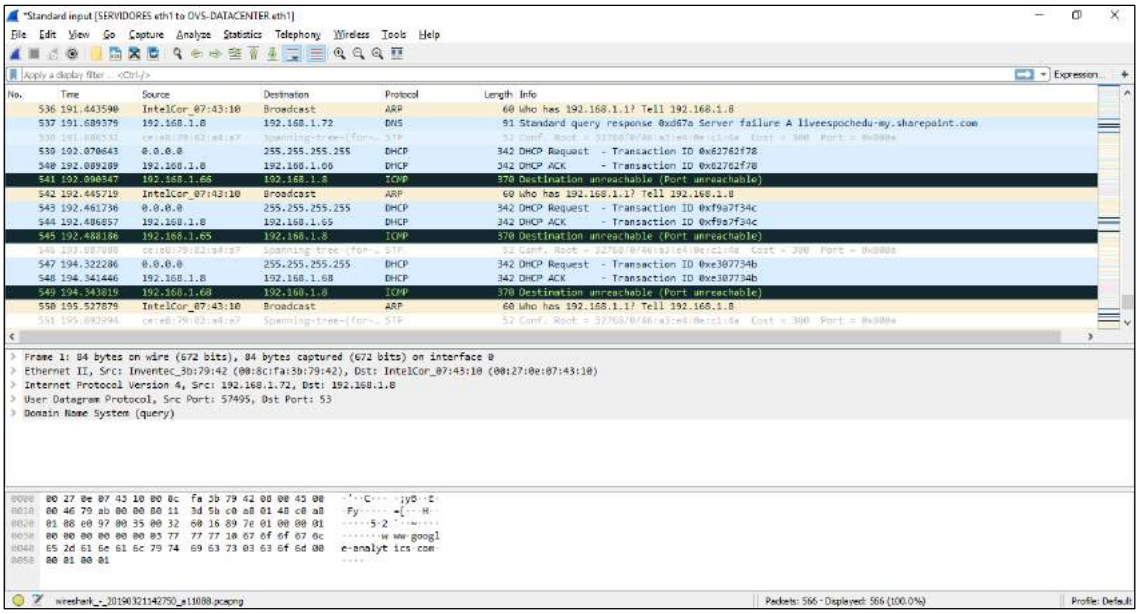
Ataque DHCP

Para iniciar con el ataque DHCP, configurar los parámetros.

El atacante simula ser un servidor DHCP, para ello instala un DHCP falso, con el propósito de que los clientes de la red se redirijan a este. Como se puede ver en la figura, la solicitud 541, 545 y 549 son usuarios que intentan conectarse al servidor original, pero como ya se inyecto el ataque DHCP, la conexión se rechaza se manera automática.

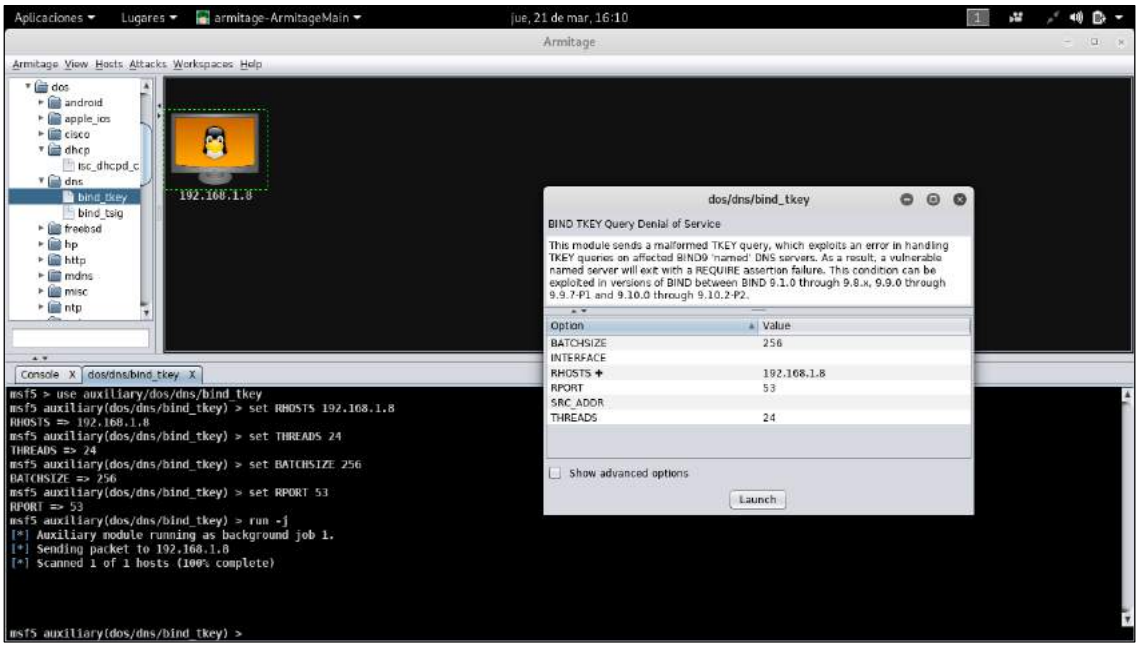


De modo que cuando el cliente desee conectarse pedirá la autorización para acceder, enviando un mensaje DHCP Request, tal como se muestra en las solicitudes 539, 543 y 547, a la dirección de broadcast y el servidor responderá con otro llamado DHCP Ack, líneas 540, 544 y 548.

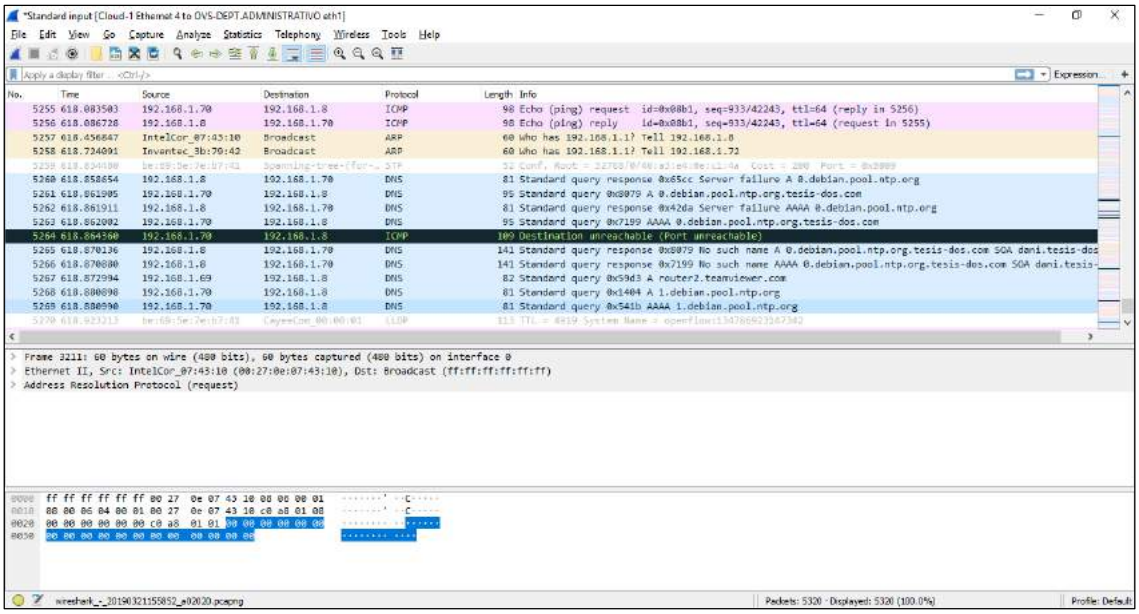


Ataque DNS

Como punto de partida, en el software Armitage, configurar el archivo bind_tkey, conforme a la figura.



De acuerdo a la figura, el atacante de red con dirección 192.168.1.70, envía solicitudes pequeñas, a la dirección 192.168.1.8, y las solicitudes de regreso resultan ser de mayor tamaño, para congestionar el servicio, pero gracias a las actualizaciones de BIND 7 a BIND 9 este ataque ya no constituye una gran amenaza.


*Standard input [Cloud-1 Ethernet 4 to OVS-DEPT.ADMINISTRATIVO eth1]
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
No. Time Source Destination Protocol Length Info
5255 618.083503 192.168.1.70 192.168.1.8 ICMP 98 Echo (ping) request id=0x08b1, seq=933/42243, ttl=64 (reply in 5256)
5256 618.086728 192.168.1.8 192.168.1.70 ICMP 98 Echo (ping) reply id=0x08b1, seq=933/42243, ttl=64 (request in 5255)
5257 618.456847 IntelCor_07:43:10 Broadcast ARP 60 Who has 192.168.1.1? Tell 192.168.1.8
5258 618.724091 Invantec_3b:70:42 Broadcast ARP 60 Who has 192.168.1.1? Tell 192.168.1.72
5259 618.824480 be:69:5e:7e:b7:41 Spanning-tree-(for-... STP 52 Conf. Root = 32768/0/46:a2:e4:8e:c1:4a Cost = 200 Port = 0x8009
5260 618.858654 192.168.1.8 192.168.1.70 DNS 81 Standard query response 0x85cc Server failure A 0.debian.pool.ntp.org
5261 618.861905 192.168.1.70 192.168.1.8 DNS 95 Standard query 0x8079 A 0.debian.pool.ntp.org.tesis-dos.com
5262 618.861911 192.168.1.8 192.168.1.70 DNS 81 Standard query response 0x42da Server failure AAAA 0.debian.pool.ntp.org
5263 618.862002 192.168.1.70 192.168.1.8 DNS 95 Standard query 0x7199 AAAA 0.debian.pool.ntp.org.tesis-dos.com
5264 618.864360 192.168.1.70 192.168.1.8 ICMP 109 Destination unreachable (Port unreachable)
5265 618.870136 192.168.1.8 192.168.1.70 DNS 141 Standard query response 0x8079 No such name A 0.debian.pool.ntp.org.tesis-dos.com SOA dani.tesis-dos
5266 618.870880 192.168.1.8 192.168.1.70 DNS 141 Standard query response 0x7199 No such name AAAA 0.debian.pool.ntp.org.tesis-dos.com SOA dani.tesis-
5267 618.872994 192.168.1.69 192.168.1.8 DNS 82 Standard query 0x59d3 A router2.teamviewer.com
5268 618.880898 192.168.1.70 192.168.1.8 DNS 81 Standard query 0x1404 A 1.debian.pool.ntp.org
5269 618.880990 192.168.1.70 192.168.1.8 DNS 81 Standard query 0x541b AAAA 1.debian.pool.ntp.org
5270 618.923213 be:69:5e:7e:b7:41 CmyeeCom_00:00:01 LLDP 113 TTL = 4919 System Name = openflow:134766923567342
Frame 3211: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
Ethernet II, Src: IntelCor_07:43:10 (00:27:0e:07:43:10), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
Address Resolution Protocol (request)
0000 ff ff ff ff ff ff 00 27 0e 07 43 10 08 06 00 01 ....' ..C.....
0010 08 00 06 04 00 01 00 27 0e 07 43 10 c0 a8 01 08 ....' ..C.....
0020 00 00 00 00 00 00 c0 a8 01 01 00 00 00 00 00 00 ....
0030 00 00 00 00 00 00 00 00 00 00 00 00 ....
wireshark_-_20190321155852_a02020.pcapng
Packets: 5320 · Displayed: 5320 (100.0%)
Profile: Default

ANEXO K: Muestreo de datos ante ataques de DoS.

ANTES	
1	6
2	6
3	10
4	10
5	4
6	5
7	12
8	5
9	4
10	4
11	21
12	31
13	29
14	21
15	14
16	9
17	7
18	5
19	5
20	12
21	10
22	10
23	8
24	9
25	11
26	10
27	5
28	16
29	15
30	4
31	4
32	4
33	8
34	12
35	11
36	9
37	11
38	10
39	6

DESPUES	
1	300
2	600
3	788
4	765
5	760
6	800
7	800
8	788
9	770
10	780
11	800
12	780
13	790
14	780
15	800
16	780
17	780
18	770
19	750
20	810
21	820
22	790
23	780
24	770
25	790
26	790
27	780
28	800
29	780
30	760
31	750
32	800
33	800
34	790
35	790
36	800
37	790
38	780
39	790

40	6
41	12
42	7
43	8
44	9
45	20
46	33
47	31
48	22
49	21
50	21
51	11
52	12
53	11
54	9
55	11
56	11
57	11
58	10
59	5
60	4
	678

40	800
41	800
42	800
43	790
44	810
45	810
46	800
47	800
48	805
49	805
50	790
51	760
52	770
53	805
54	805
55	805
56	795
57	790
58	800
59	800
60	805
	46686

ANEXO L: Prueba estadística de datos recolectados antes y después del ataque DoS.

Para realizar la prueba de normalidad, se consideró la cantidad de 250 datos, antes y después de realizarse el ataque de denegación de servicio.

Pruebas de normalidad

	Kolmogorov-Smirnov[a]		
	Estadístico	gl	Sig.
Antes	,084	250	,000
Después	,064	250	,015

a. Corrección de la significación de Lilliefors

ANEXO M: Plan de contingencia o guía de buenas practicas

Propuesta 2: Generación de Flujos.

Flujos de bloqueo de puertos de comunicación.

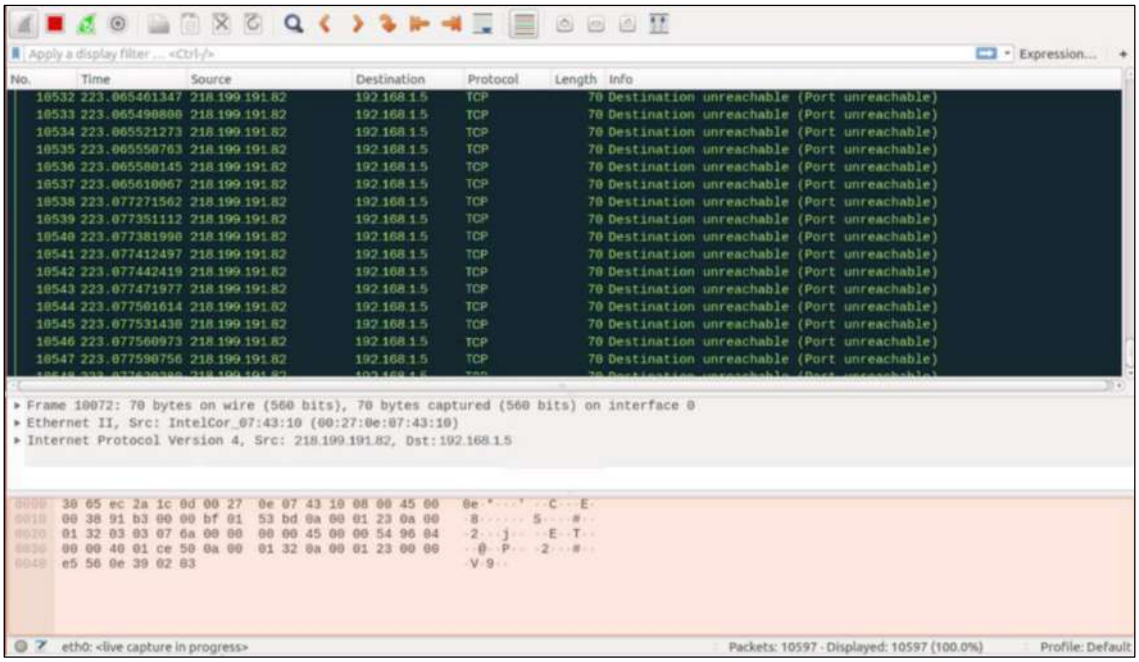
En este anexo se puede observar la creación de flujos, indicando tanto el número de tabla, como el número de flujo creado, también se menciona la prioridad de dicho flujo así también al puerto que va hacer enviado con las indicaciones o acciones a tomar desde el controlador OpenFlow.

```
http://192.168.1.5:8181/restconf/config/opendaylight-inventory:nodes/node/openflow:77669719785802/flow-node-inventory:table/1/flow/1

{
    "flow": [
        {
            "id": "1",
            "match": {
                "tcp-destination-port": "8181"
                     },
            "instructions": {
                "instruction": [
                    {
                        "apply-actions": {
                            "action": [
                                {
                                    "drop-action": {}
                                }
                                      ]
                                         }
                    }
                                ]
                               },
            "flow-name": "flujo-bloqueo-puertos",
            "priority": "1"
         }
             ]
}
```

En el código, se puede observar que se crea una regla que con la acción de bloqueo al puerto 8181 del controlador, con numero de tabla 1 y prioridad 1 con el propósito de protegerlo de alguna amenaza externa.

En la figura que se muestra a continuación se verifica que no haya conectividad al aplicar un ataque de denegación de servicio.



Flujo de bloqueo de direcciones MAC

En el siguiente código se puede observar la creación de un flujo que bloquea las direcciones MAC. Se puede apreciar que se crea un flujo llamado flujo-bloqueo-MAC con identificador 2 para realizar la acción de eliminar todo tipo de paquete procedente desde la dirección 8E:3F:6E:83:56:22 hasta la B6:4B:78:F2:2A:B5

```
http://192.168.1.5:8181/restconf/config/opendaylight-inventory:nodes/node/openflow:77669719785802/flow-node-inventory:table/1/flow/2

{
    "flow": [
        {
            "id": "2",
            "match": {
                "ethernet-match": {
                    "ethernet-source": {
                        "address": "8E:3F:6E:83:56:22"
                    },
                    "ethernet-destination": {
                        "address": "B6:4B:78:F2:2A:B5"
                    }
```

```
                }
            },
            "instructions": {
                "instruction": [
                    {
                        "apply-actions": {
                            "action": [
                                {
                                    "drop-action": {}
                                }
                            ]
                        }
                    }
                ]
            },
            "flow-name": "flujo-bloqueo-MAC",
            "priority": "1"
        }
    ]
}
```

Flujos de bloqueo de direcciones IP

Para proceder al bloqueo de direcciones IP, se debe seguir los mismos pasos que se realizaron con las direcciones MAC, pero en este caso indicar la ruta de IPV4.

El código mostrado a continuación se establece que en la tabla 1 el flujo numero 3 no permita que se realiza una solicitud de la dirección IP 192.168.1.72 hacia la dirección IP 192.168.1.8 que es la dirección lógica del controlador ya que probablemente haya una falla de seguridad.

```
http://192.168.1.5:8181/restconf/config/opendaylight-inventory:nodes/node/openflow:77669719785802/flow-node-inventory:table/1/flow/3

{
    "flow": [
        {
            "id": "3",
            "match": {
                "ipv4-source": "192.168.1.72/24",
                "ipv4-destination": "192.168.1.8/24"
            },
            "instructions": {
```

```
                "instruction": [
                    {
                        "apply-actions": {
                            "action": [
                                {
                                    "drop-action": {}
                                }
                            ]
                        }
                    }
                ]
            },
            "flow-name": "flujo-bloqueo-IP",
            "priority": "1"
        }
    ]
}
```

Restricción de ancho de banda

Se establece este criterio, debido a que cuando la red está siendo atacada la cantidad de paquetes generados son muy elevados, por lo tanto, consumen mucho ancho de banda y así asegurar una buena calidad de servicio en el transporte de datos.

En este punto se limita el ancho de banda usado en la comunicación, con la creación de una política, en los servidores cuya dirección MAC es 00:27:0E:07:43:10, para impedir que los paquetes lleguen a esa dirección, luego se establece el tamaño del ancho de banda o velocidad máxima en el campo band-rate y band-rate en este caso, se propone de 10 Mbps, el primero corresponde a la velocidad requerida por el cliente mientras que el segundo es a la que el sistema trabaja.

En tanto a los a band-burst-sizze y drop-burst-size se deben llenarse con el valor de 0 por recomendación para que el switch vaya descartando paquetes con normalidad.

```
http://192.168.1.5:8181/restconf/config/opendaylight-inventory:nodes/node/77669719785802/flow-node-inventory:meter/2

{
    "meter": [
        {
```

```
            "flags": "meter-kbps",
            "meter-id": "2",
            "meter-name": "ancho-de-banda",
            "container-name": "ancho-de-banda",
            "meter-band-headers": {
                "meter-band-header": [
                    {
                        "band-id": "0",
                        "band-rate": "10000",
                        "band-burst-size": "0",
                        "drop-rate": "10000",
                        "drop-burst-size": "0"
                    }
                ]
            }
        }
    ]
}
```

Para verificar que la política haya sido establecida con éxito, se realiza la prueba de conectividad hacia la dirección IP donde se encuentran alojados los servicios de red.