ESCUELA SUPERIOR POLITECNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

"CREACIÓN Y SIMULACIÓN DE UN AMBIENTE DE PENTESTING SOBRE GNS3 PARA LAS PLATAFORMAS EN ETAPA DE PRE-PRODUCCIÓN DEL BCE"

EXAMEN DE GRADO

Previo a la obtención del título de

MAGÍSTER EN TELECOMUNICACIONES

LUIS FABRICIO SILVA CRUZ

GUAYAQUIL – ECUADOR

AÑO: 2020

AGRADECIMIENTO

Agradezco a Dios, por sobre todas las cosas por su infinito amor y sabiduría, con los cuales puso en mi vida ycarrera profesional la oportunidad de escalar un peldaño más. A mis padres por apoyar desde siempre cada etapa de mi formación personal y profesional y enorgullecerse con la mayor felicidad de cada uno de mis logros.

DEDICATORIA

Dedico este gran esfuerzo, que ha involucrado innumerables sacrificios a mi familia que siempre ha estado a mi lado, quienes me han apoyado y han aplaudido y disfrutado de mis logros y del mismo modo me acompañaron en los momentos más difíciles y complicados. A mis padres, a quienes les debo todo lo que soy por el ejemplo y los valores que me inculcaron, basados en perseverancia, disciplina y sacrificio.

TRIBUNAL DE EVALUACIÓN

M.Sc. Verónica Soto	Ph.D. María Antonieta Álvarez
PROFESOR EVALUADOR	PROFESOR EVALUADOR

DECLARACIÓN EXPRESA

La responsabilidad y la autoría del contenido de este Trabajo de Titulación, me(nos) corresponde exclusivamente; y doy(damos) mi(nuestro) consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual.

SILVA CRUZ LUIS FABRICIO

RESUMEN

La integridad y disponibilidad de una aplicación, plataforma o servicio tecnológico guardan dependencia en los parámetros y consideraciones de seguridad que hayan sido aplicados sobre los elementos activos y pasivos de la infraestructura en las diferentes etapas del ciclo de vida del desarrollo y puesta en producción del servicio o aplicación.

Los ataques cibernéticos que buscan vulnerar los sistemas de información se materializan en gran porcentaje por debilidades en la configuración y el no apropiado análisis de los riesgos y vulnerabilidades activas en los sistemas y sus componentes. El uso de técnicas y procedimientos para evaluar y mitigar las brechas de seguridad sobre cualquier componente tecnológico constituyen en los momentos actuales uno de los métodos más efectivos para reducir el riesgo de ataque y vulneración.

Con el desarrollo del presente proyecto se busca proporcionar un ambiente para la ejecución de pruebas de penetración (pentesting) sobre herramientas opensource, en el cual es factible simular el funcionamiento e interacción de los diferentes elementos que componen la arquitectura de las plataformas de los servicios WEB del BCE, incluyendo aquellos elementos que son excluidos por disponibilidad y criticidad en las pruebas de penetración realizadas previo a la puesta en producción de los sistemas.

ÍNDICE GENERAL

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

CAPÍTULO 1

INTRODUCCIÓN

La implementación de ambientes virtualizados en las plataformas de Tecnología de las compañías a nivel global se ha constituido hoy en día como una de las soluciones más eficientes en las relaciones de costo beneficio tanto en lo económico así como en lo tecnológico a nivel computacional.

¿Qué es la virtualización y porque es tan importante? Una definición podría ser que la virtualización es la abstracción de un recurso de computación desde otro recurso de computación. Estableciendo el concepto anterior en términos de TI, la abstracción de un servidor (un recurso de computación) la realizamos desde un espacio de almacenamiento o storage(otro recurso de computación). [1]

Por otro lado, las pruebas de penetración que evalúan el estado de protección y vulneración de los diferentes sistemas y servicios tecnológicos, brindan precisamente a través de su funcionalidad y resultados, una herramienta más de protección a las compañías y sus sistemas ante los grupos de ataques cibernéticos que buscan obtener algún beneficio a través de la explotación de alguna vulnerabilidad existente en los componentes de los sistemas.

1.1 Descripción del problema

El Banco Central del Ecuador, como entidad rectora de la política monetaria y financiera del Sistema Financiero Nacional, tiene entre sus responsabilidades y atribuciones el brindar los mecanismos y sistemas tecnológicos que garanticen el correcto control y registro de las transacciones financieras que se realizan diariamente entre las diferentes entidades públicas y privadas, para lo cual dispone de sistemas y plataformas WEB que son propensos a cambios en su estructura, configuración y funcionalidad.

Estas nuevas versiones de cada plataforma son construidas siguiendo el procedimiento de ciclo de vida de desarrollo de aplicaciones, en la cual se

contempla la etapa de pruebas de vulnerabilidad y test de penetración (pentesting) sobre los componentes tecnológicos que las conforman.

Para estas pruebas se requiere la participación secuencial de los elementos y administradores de las diferentes capas que conforman el servicio o plataforma, tales como elementos activos y pasivos de red, servidores de infraestructura, balanceadores de aplicaciones, servidores de aplicación y de bases de datos, por mencionar algunos. El escenario descrito anteriormente se vuelve complejo y el alcance del análisis y pruebas no abarca todos los componentes por cuanto muchos de estos elementos forman también parte de las Arquitecturas de Desarrollo y Producción (elementos activos de red como firewall, switches, balanceadores geográficos, etc.) motivo por el cual finalmente son excluidos del análisis de vulnerabilidades y pruebas de penetración en etapas críticas de Pre-Producción.

1.2 Justificación / Propuesta

Se propone la creación y simulación de un ambiente para la ejecución de pruebas de penetración (pentesting) sobre la herramienta opensource GNS3, en el cual sea factible simular el funcionamiento e interacción de los diferentes elementos que componen la arquitectura de las plataformas de los servicios WEB del BCE, incluyendo aquellos elementos que son excluidos por disponibilidad y criticidad en las pruebas de penetración realizadas previo a la puesta en producción de los sistemas antes mencionados.

1.3 Objetivos

1.3.1 Objetivo General

Disponer de un ambiente virtualizado en el cual sea posible poner a prueba los servidores de infraestructura y dispositivos de red que componen las aplicaciones WEB del BCE.

1.3.2 Objetivos Específicos

Crear un ambiente virtualizado con los dispositivos de Red que conforman la infraestructura de comunicaciones en la herramienta GNS3, con la finalidad de simular el comportamiento y funcionalidad en las etapas de Pre-Producción.

Implementar servidores virtuales sobre la herramienta VmWare, los cuales deben poseer las mismas configuraciones, servicios y funciones de los servidores utilizados en el ambiente real de pruebas de las aplicaciones WEB.

Desarrollar un conjunto de pruebas, basado en la metodología del Penetration Testing Execution Standard (PTES) en sus diferentes etapas, que serán aplicadas sobre los servidores y dispositivos de red con la utilización adicional de herramientas de código abierto definidas para la ejecución de pruebas de penetración. [2]

Generar un Informe ejecutivo de las vulnerabilidades y defectos encontrados sobre los elementos evaluados, el cual debe contener sugerencias de las acciones correctivas que deben aplicarse sobre los dispositivos afectados en base a la metodología aplicada.

1.4 Marco Teórico

1.4.1 GNS3

GNS3 es una herramienta Open Source utilizada ampliamente por ingenieros de redes y especialistas de tecnología para emular, configurar y probar infraestructuras de redes reales en entornos virtualizados. [3]

La plataforma GNS3 ofrece un ambiente virtual, que puede ser aislado completamente de la red física, para desplegar y probar configuraciones y topologías de red mucho más complejas sin que las características del ambiente simulado generen alguna afectación sobre el ambiente real.

Al tratarse de un software de tipo opensource, ofrece la compatibilidad de integrar en sus ambientes simulados la gran mayoría de sistemas operativos conocidos en el entorno de TI: Linux, Mac, Windows, etc. En cuanto a emuladores de

equipos de red se destaca la compatibilidad para DYNAMIPS (emulador de imágenes de IOS de equipos CISCO), IOU (es un emulador no nativo para Windows), VMware (plataforma de virtualización y simulación ampliamente utilizada y comercial), etc. [4]

Básicamente su funcionalidad consiste en emular el comportamiento de los elementos de una red física haciendo uso de recursos en software, lo cual permite simular, por ejemplo, la infraestructura de un sistema con todos sus componentes previo a una puesta en Producción.

A continuación, se describirá a breves rasgos los elementos que componen su interfaz gráfica:

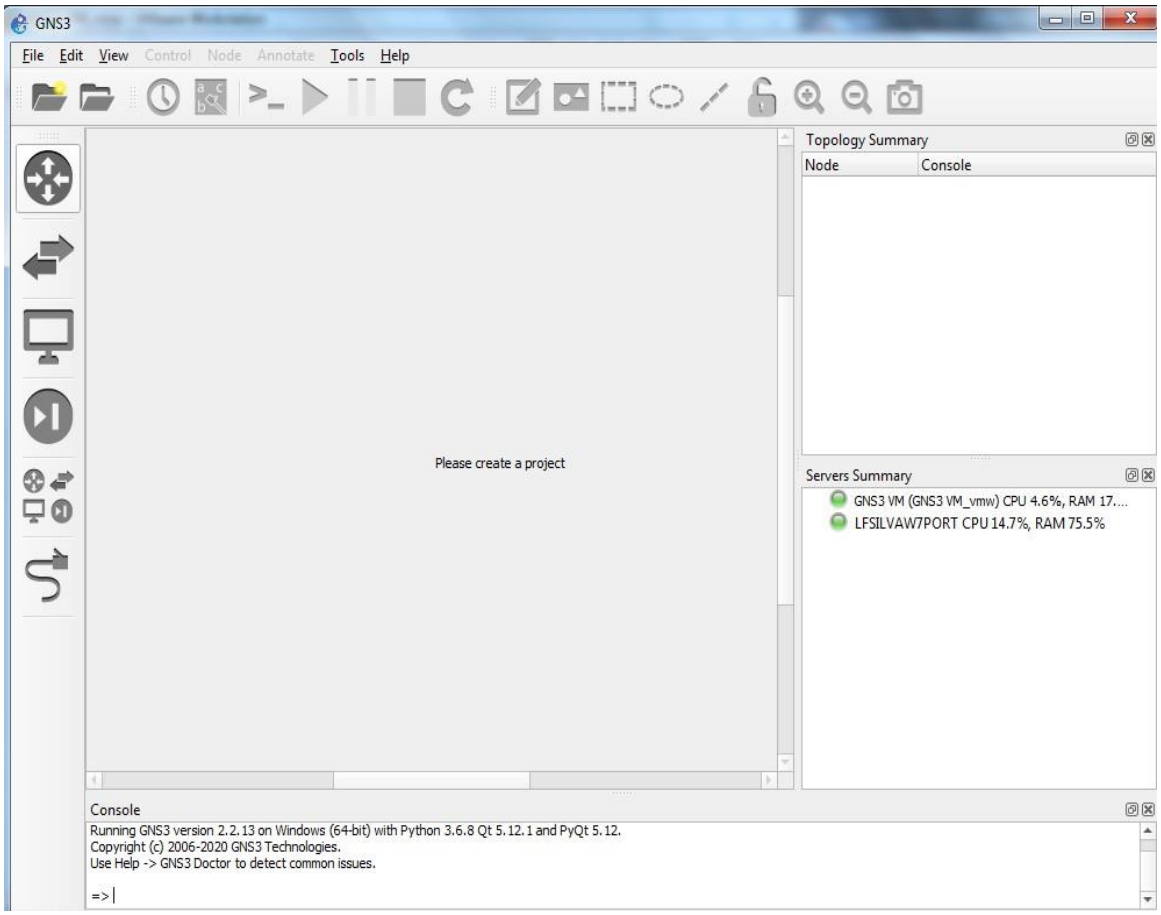


Figura 1.1 Interfaz de Software GNS3

En la Figura 1.1 se muestra la interfaz de GNS3 donde podemos identificar en la barra superior las opciones que permiten ejecutar y editar el contenido de los proyectos. En la barra lateral izquierda se encuentran los elementos de red y dispositivos que forman parte de la simulación. En la sección derecha se ubican los resúmenes de la topología que se está implementando y el estatus del servidor y el equipo que soportan la simulación de los proyectos.

1.4.2 Pentesting

La prueba de penetración, o Pentesting se define como la simulación real de ataques para identificar los riesgos asociados con brechas potenciales de seguridad [5] sobre entornos de redes y sistemas. En otras palabras, podemos definirlo como una prueba de intrusión que pretende evaluar las características de seguridad de un sistema o plataforma informática.

Para el presente proyecto se ha tomado como referencia la metodología del *Penetration Testing Execution Standard [6],* el cual se divide en cada una de las fases detalladas en la Tabla 1.

	Fase	Descripción
Penetration Testing Execution Standard	Pre-engagement Interactions	Establecer las normas que se aplicarán en la ejecución del Pentesting
	Intelligence Gathering	Extraer la mayor cantidad de información posible de la infraestructura a analizar
	Threat Modeling	Modelado de amenazas que buscan identificarse
	Vulnerability Analysis	Análisis y clasificación de vulnerabilidades
	Exploitation	Aprovechar vulnerabilidades para afectar al sistema
	Post Exploitation	Actividades que se realizan sobre el elemento vulnerado
	Reporting	Análisis de la información obtenida para la generación de un informe ejecutivo de la prueba

Tabla 1. Fases de una prueba Pentesting

CAPÍTULO 2

DISEÑO E IMPLEMENTACIÓN

En esta etapa del proyecto, se procurará describir y especificar con los detalles que corresponden el escenario que se ha propuesto así como los elementos que serán analizados y como fueron estos configurados e integrados en la plataforma de virtualización GNS3.

2.1 Definición del Escenario y Alcance

El escenario sobre el cual se pretende simular la aplicación del entorno de simulación corresponde a la forma en que se encuentra estructurada una de las plataformas de servicio WEB que el BCE gestiona y ofrece a las Entidades del Sistema Financiero Nacional y a Instituciones Públicas que utilizan estos servicios.

El alcance de la implementación y simulación abarca a elementos de red activos y pasivos (Firewall, Routers, Switches) así como a servidores de servicios y aplicaciones que se detallan en el diseño funcional representado en la Figura 2.1.

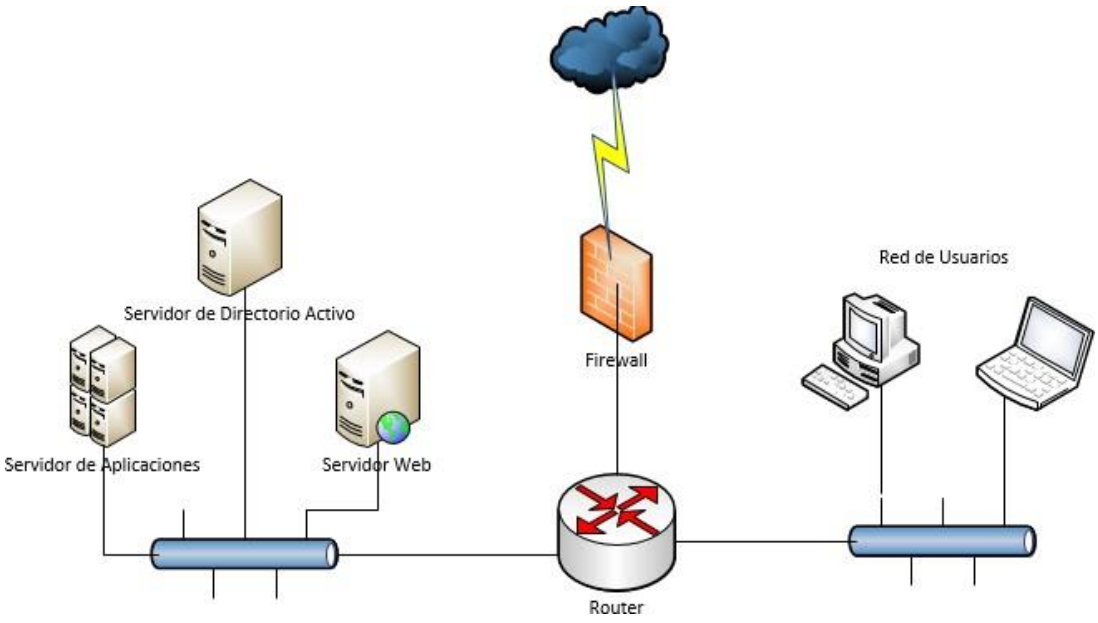


Figura 2.1 Diseño Funcional del escenario propuesto

Para la implementación y simulación de los elementos antes descritos, se dispondrán de los siguientes recursos a nivel de software:

- **Herramientas de virtualización**, que permitan disponer de recursos de software para simular el funcionamiento de los componentes de la red y la plataforma propuesta para el análisis, para esto utilizaremos GNS3 y VMware.
- **Servidores Virtuales,** sobre los cuales se emularán los Sistemas Operativos característicos de las plataformas de producción en servicios WEB de alta disponibilidad.
- **Firewall o Cortafuego,** que permitirá simular la gestión del tráfico recibido en la red del sistema desde y hacia el internet o redes WAN.
- **Router,** que nos ayudará a establecer la comunicación entre la red de servidores, internet y la red interna.
- **Switch,** elemento con el que se interconectarán a nivel lógico los servidores y elementos de red.

2.2 Elementos de Análisis

Para efectuar el análisis y prueba de Pentesting en el ambiente simulado propuesto, se hará uso de los siguientes recursos de TI, disponibles en plataformas opensource y comerciales:

- **Kali Linux.-** Es una de las herramientas sobre sistema operativo basada en la distribución DEBIAN de Linux utilizada ampliamente en el campo de las auditorias y pruebas de vulnerabilidad y seguridad. Se caracteriza por estar integrada de cientos de herramientas orientadas a diferentes tareas de seguridad, sobre todo en el ámbito del Pentesting. [7] Una vista general de la interfaz y sus componentes es mostrada en la Figura 2.2.

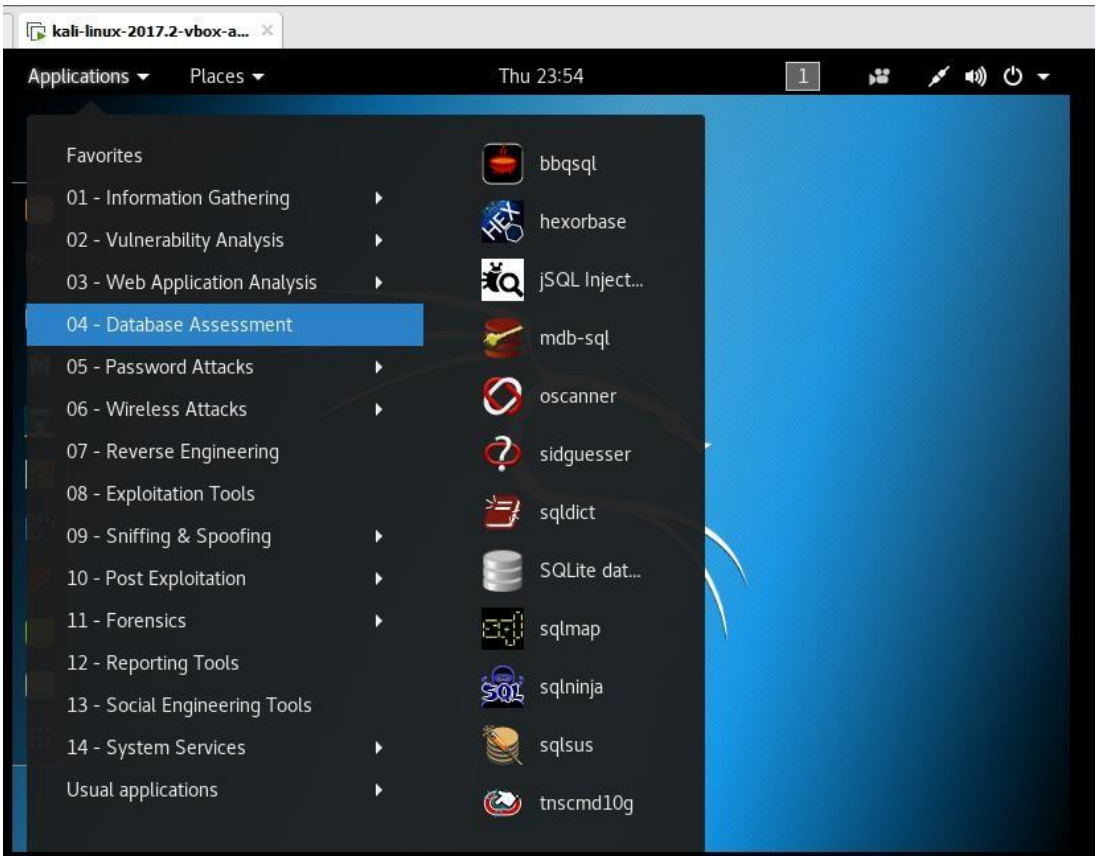


Figura 2.2. Interfaz de Kali Linux y herramientas de Pentesting

- **Nessus Profesional.** Es una herramienta de Software utilizado para la evaluación de vulnerabilidades que integra características automatizadas que permiten identificar y corregir con rapidez las vulnerabilidades incluidos parches faltantes, defectos de software, malware y configuraciones erróneas, en diversos sistemas operativos, dispositivos y aplicaciones.[8]

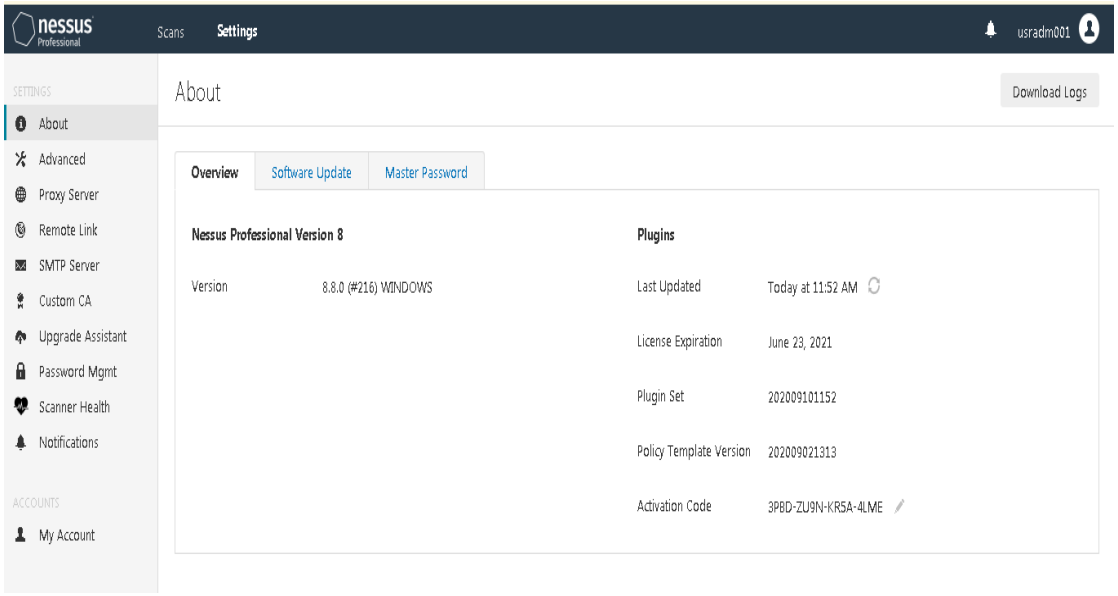


Figura 2.3. Vista General de Nessus Professional

La Figura 2.3, presenta una vista de la interfaz de la plataforma Nessus Professional utilizada como complemento en el proceso de análisis de vulnerabilidades para el presente proyecto.

2.3 Definición de PoC

La prueba de concepto abarca todos los elementos detallados en el CAPÍTULO 2.1 y consiste en la integración de los mismos en un escenario virtualizado dentro de las plataformas GNS3 y VMware de acuerdo al esquema en la Figura 2.4.

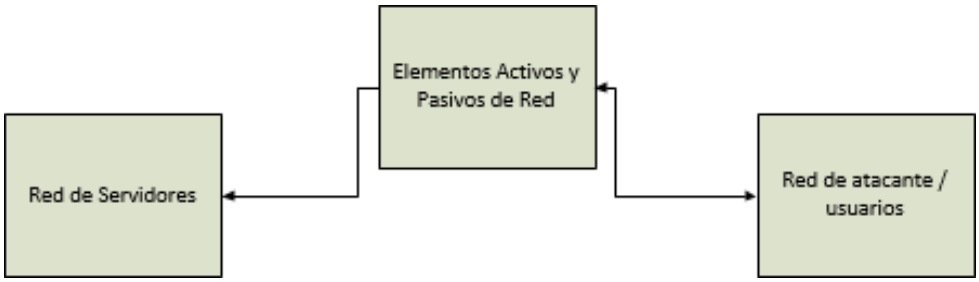


Figura 2.4. Diagrama esquemático de la prueba

Como recursos de Pentesting, se utilizará para la prueba descrita las herramientas de KALI Linux que se enuncian a continuación:

- **Nmap.** Es una herramienta utilizada para la identificación de puertos de comunicación basados en protocolos que se encuentren activos en los servidores o elementos atacados. Es también capaz de identificar y relacionar estos puertos con los servicios activos.
- **Metasploit.** Es uno de los elementos de Kali Linux con mayor orientación al pentesting, sobre todo por la variedad de módulos de los que dispone con los que es posible por ejemplo realizar un análisis de DLLs en los sistemas y generar programas Shell que pueden ser embebidos como ejecutables en Sistemas Operativos Windows y UNIX.
- **Routersploit.** Entre sus características se destaca el módulo que es utilizado para intentar vulnerar credenciales de dispositivos de red, en este contexto examina todas aquellas brechas presentes en este tipo de dispositivos.

2.4 Diagrama de la red

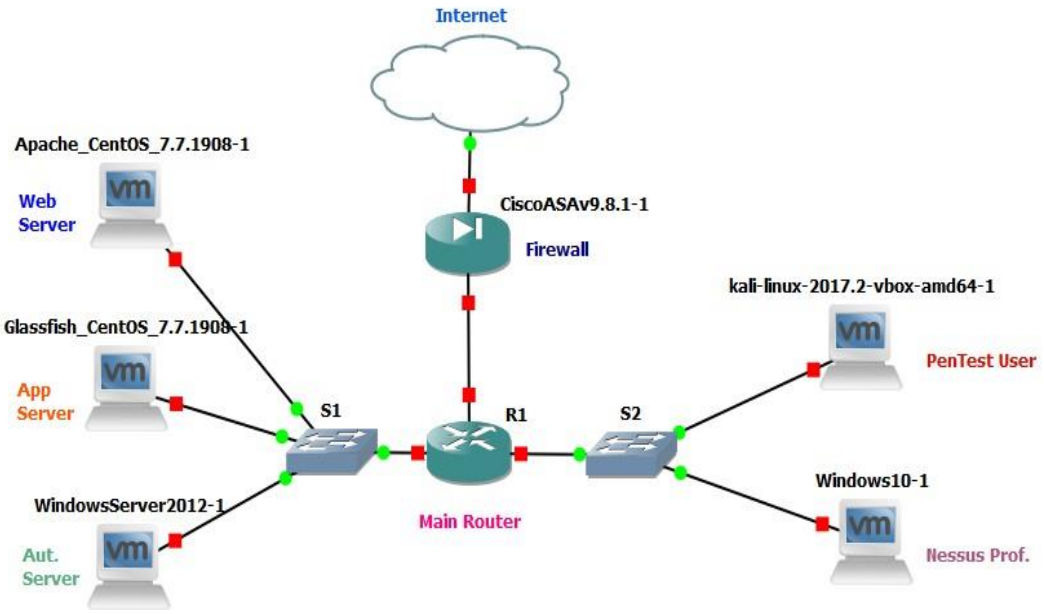


Figura 2.5. Diagrama de Topología de Red

En la Figura 2.5 es posible observar e identificar en un esquema de topología de red, la estructura que tiene el ambiente de simulación creado sobre la plataforma GNS3.

En éste se ubican por un lado, los servidores virtuales que serán objetivo de la prueba de penetración en un segmento de red independiente. Por el otro lado se representan los elementos de análisis tales como el servidor con Kali Linux y un dispositivo de usuario final con Sistema Operativo Windows 10, sobre el cual se está ejecutando el software Nessus Professional.

Para efectos de establecer correctamente la comunicación entre los servidores virtualizados en VMware y la plataforma GNS3, se definieron las subredes internas que se detallan en la Figura 2.6.

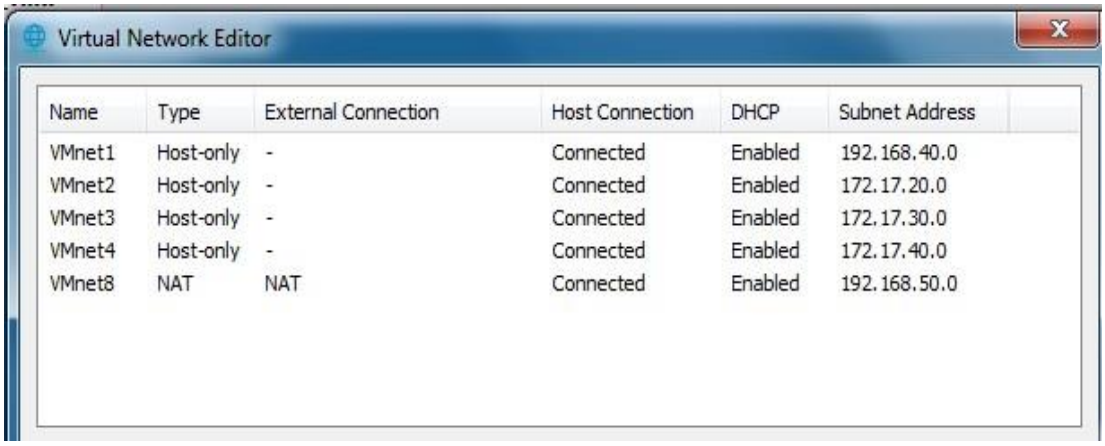


Figura 2.6. Definición de segmentos de red en VMware

Así, la red identificada con el nombre **VMnet2** será el segmento de red asignado a los servidores. La red **VMnet3** se reservará para el segmento de usuarios/pentester y la red **VMnet4** será utilizada para establecer la comunicación entre los dispositivos de red (Firewall – Router).

2.5 Configuración de servidores virtuales

Luego de Creadas o importadas las máquinas Virtuales en la plataforma VMware se procede con la asignación de recursos a nivel de hardware y la instalación de los componentes de software adicionales requeridos para que puedan cumplir su rol dentro del ambiente de Pre-producción simulado.

A continuación, se detallan las características configuradas sobre cada servidor virtual:

- **Servidor WEB**
    - Sistema operativo: Centos 7.7
    - Memoria virtual asignada: 2048 MB
    - Procesadores asignados: 2
    - Software adicional: Apache/2.4.6
    - Dirección IP: 172.17.20.21

- **Servidor de Aplicaciones**
    - Sistema operativo: Centos 7.7
    - Memoria virtual asignada: 2048 MB
    - Procesadores asignados: 2
    - Software adicional: Glassfish 5.0
    - Dirección IP: 172.17.20.22
- **Servidor de Autenticación**
    - Sistema operativo: Windows Server 2012 R2
    - Memoria virtual asignada: 3072 MB
    - Procesadores asignados: 4
    - Software adicional: Domain Controler Rol
    - Dirección IP: 172.17.20.23
- **Kali Linux Server**
    - Sistema operativo: Kali Linux 2017.2
    - Memoria virtual asignada: 2048 MB
    - Procesadores asignados: 2
    - Software adicional: Ninguno
    - Dirección IP: 172.17.30.15

En la Figura 2.7 se muestra el resumen de los servidores virtuales implementados sobre VMware.

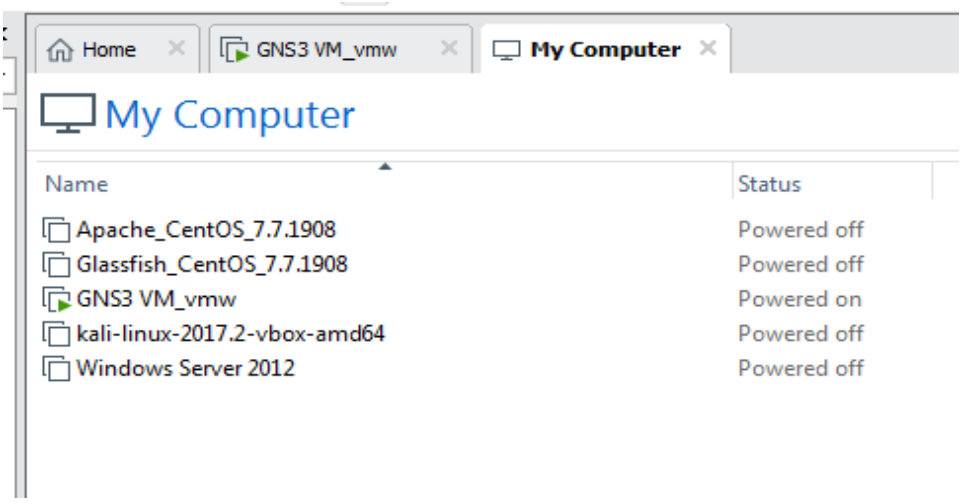


Figura 2.7. Resumen de servidores en WMware

2.6 Configuración de elementos de red virtuales

El Router R1, que es el router principal de nuestra topología se encuentra emulado sobre un router c3600 Cisco, para el cual realizaremos la configuración de los parámetros de red y enrutamiento a través de la consola de acuerdo a la siguiente información:

Router R1	
Interfaz	Dirección IP
e0/0	172.17.40.10
e0/1	172.17.20.10
e0/2	172.17.30.10

Tabla 2. Descripción de interfaces en Router R1

Como se puede observar en la Tabla 2, cada segmento de red ha sido asignado de acuerdo a la ubicación o subred en la que se conectará de acuerdo a la topología, así la interfaz e0/0 servirá de Gateway entre el firewall y la red interna, la interfaz e0/1 se encontrará ubicada en la subred de servidores y la interfaz e0/2 en el segmento de los equipos de análisis / atacantes.

La configuración lógica y asignación de interfaces de red en el router R1 se muestran en la consola de GNS3 en la Figura 2.8.

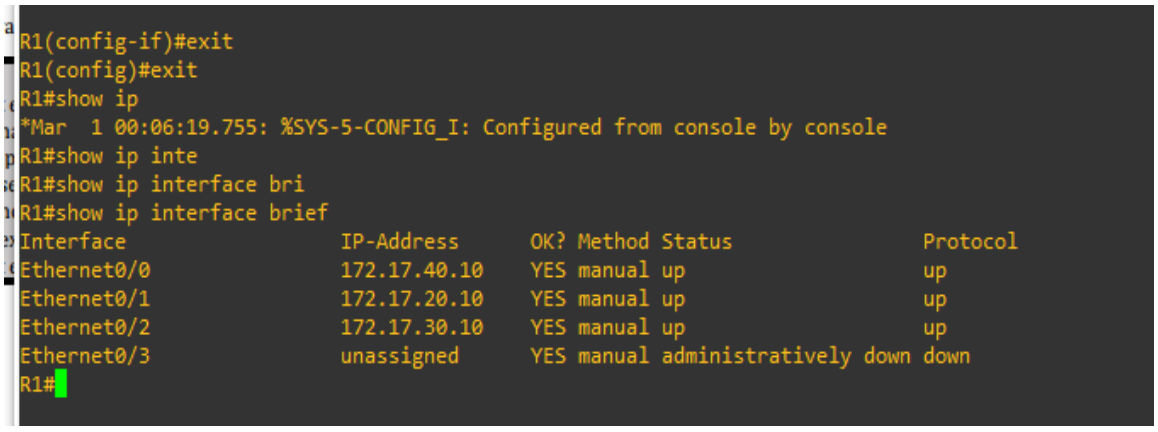


Figura 2.8. Configuración de interfaces de red en Router R1

Para establecer el enrutamiento dinámico entre los diferentes segmentos de red, se aplicará el protocolo RIP, la configuración se muestra en la Figura 2.9.

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#router rip
R1(config-router)#ne
R1(config-router)#net
R1(config-router)#network 172.17.20.0
R1(config-router)#network 172.17.30.0
R1(config-router)#network 172.17.40.0
R1(config-router)#no au
R1(config-router)#no auto-summary
R1(config-router)#exit
R1(config)#
```

Figura 2.9. Configuración de protocolo RIP

El Firewall escogido para la simulación de la presente prueba, el Cisco ASAV contienen la emulación del sistema operativo ASAV9.8.1, para el cual es posible de manera adicional establecer el parámetro del recurso de memoria asignado para su correcto funcionamiento de acuerdo a la complejidad de la topología que se haya implementado.

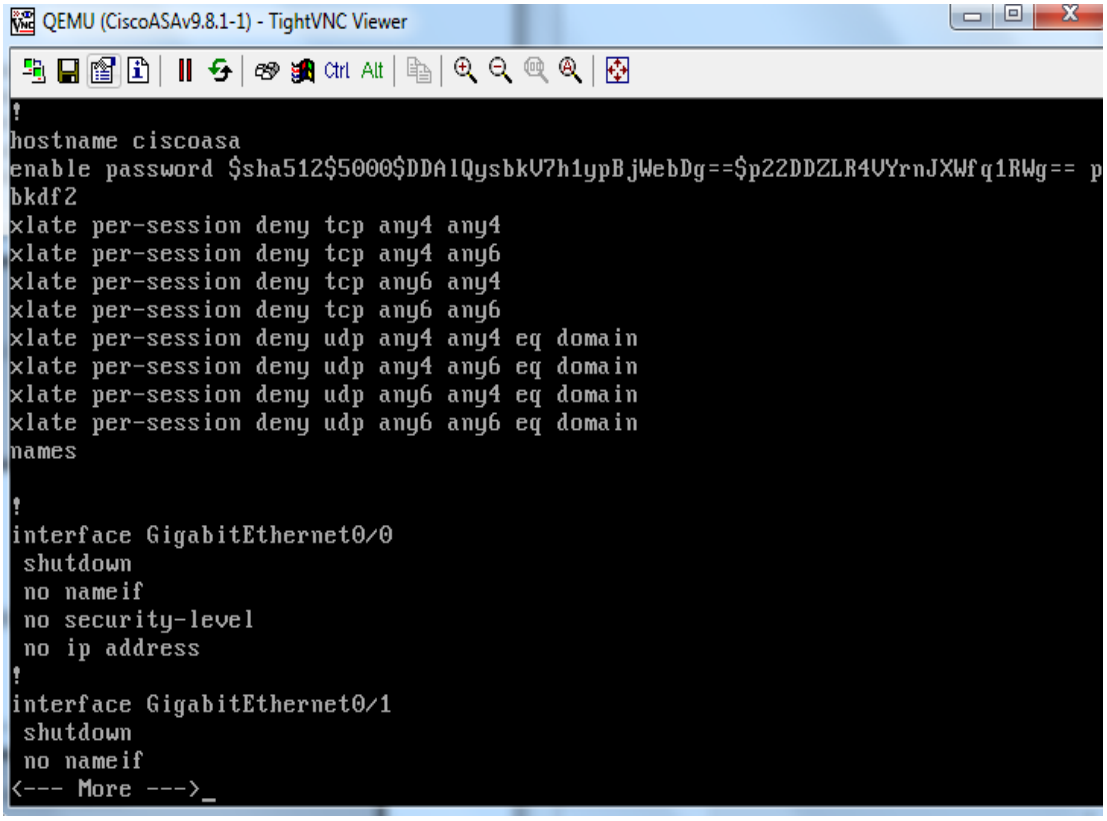


Figura 2.10. Terminal de configuración en VNC de Firewall Cisco ASA

En la figura 2.10 se establecen los parámetros por defecto del firewall lógico implementado en nuestra simulación, para efectos de la prueba se han configurado políticas por defecto de filtrado por cuanto el ambiente no será expuesto a internet.

CAPÍTULO 3

EJECUCIÓN DE PRUEBAS

Para la simulación de pentesting se han establecido las pruebas que se detallan a continuación, con las cuales se buscará verificar la interacción y comunicación entre los elementos simulados y virtualizados.

3.1 Descripción de pruebas

3.1.1 Prueba de Pentesting sobre elemento de Red

Se realizó una prueba de pentesting sobre el router principal de la plataforma simulada, desde el atacante establecido en la máquina virtual Kali Linux. El objetivo de la prueba consiste en obtener el control de la administración del router a través de la herramienta Routersploit.

Tomando como punto de partida las fases del pentesting, en la fase de definición del Pre-engagement interactions (alcance y aspectos de la prueba), lo que buscamos es acceder al router sin ningún tipo de restricción para ejecutar o modificar sus configuraciones.

Para la ejecución de la segunda fase de pentesting, necesitamos recolectar la mayor cantidad de información que el dispositivo nos puede brindar acerca de su estado.

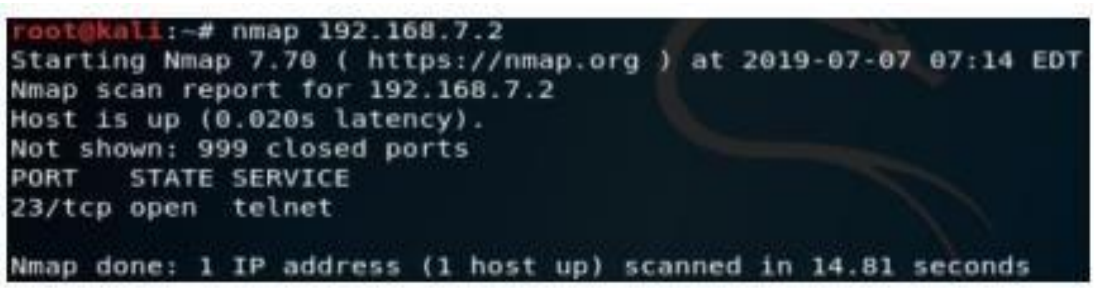


Figura 3.1. Nmap en R1

De acuerdo al resultado del comando Nmap en el router R1 en la Figura 3.1, se identifica que el puerto 23 se encuentra abierto y esto podría dar lugar a una vulneración del dispositivo por ésta vía.

A continuación, luego de lo evidenciado como información inicial obtenida del Router, procedemos con la herramienta de Kali Linux, Routersploit realizando un escaneo general de los exploits posibles con la opción *"scanners/autopwn",* estableciendo a través de comandos el parámetro "*set target".* El resultado del comando se muestra en la Figura 3.2.

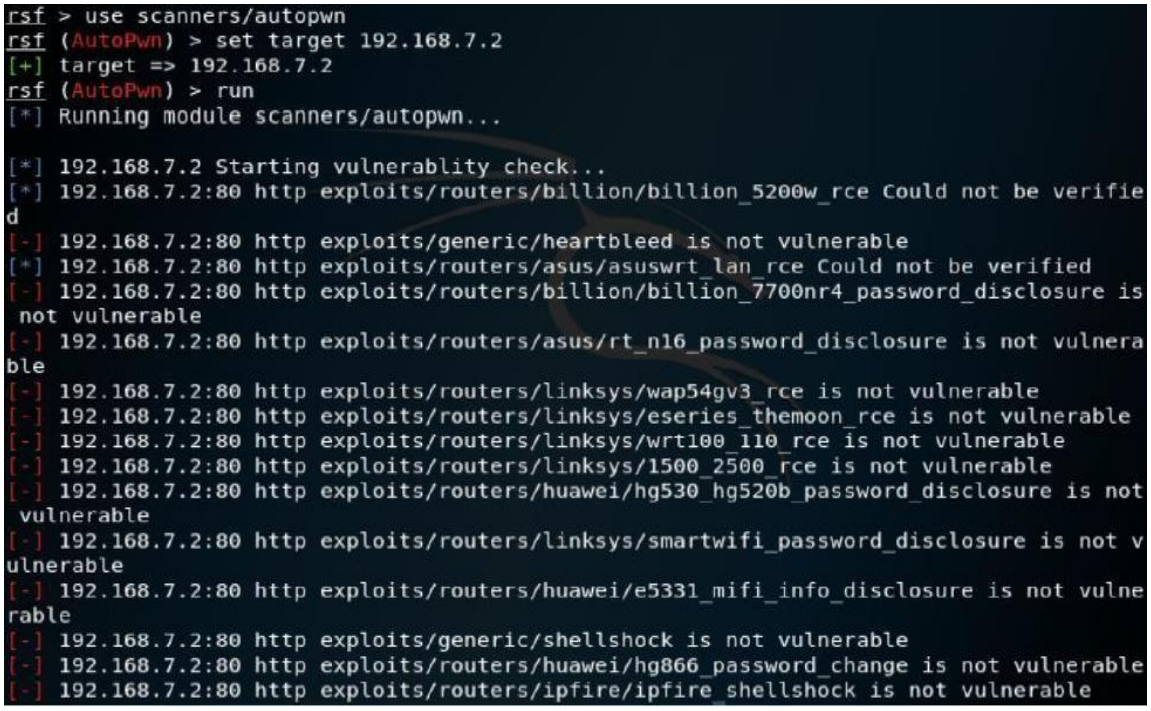


Figura 3.2. Ejecución de herramienta RouterSploit

De acuerdo al resultado obtenido, no se lograron identificar vulnerabilidades de acuerdo a los exploit evaluados. Sin embargo y con el objetivo de profundizar con un paso más en la presente prueba, se realizó la denominada búsqueda de patrones que permitan detectar configuraciones de credenciales por defecto en el dispositivo a través de la búsqueda del patrón cisco telnet: *"search cisco telnet",*

desde donde se devolverán a través de la ejecución del comando las ubicaciones de las credenciales por defecto que se relacionen con el servicio telnet.

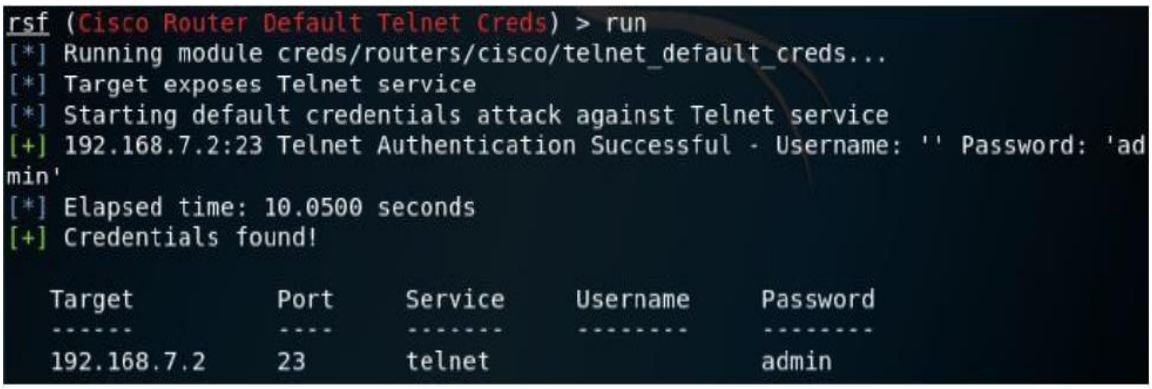


Figura 3.3. Exploit de ataque sobre credenciales de router

En la Figura 3.3 se visualiza la ejecución del comando, luego de lo cual se realiza la obtención de las credenciales por defecto identificadas en los campos username y password, identificando la posibilidad de realizar una conexión a través de protocolo telnet al router con estas sencillas credenciales encontradas.

3.1.2 Prueba de Pentesting sobre servidor de Aplicaciones

A continuación, se ha definido un conjunto de pruebas utilizando la herramienta Metasploit Framework para intentar vulnerar y tomar el control del Servidor de Aplicaciones de nuestro entorno virtualizado, utilizando la elevación de privilegios no autorizada.

En el contexto de las fases del pentesting, definiremos también para esta prueba que el objetivo y alcance de la misma será obtener el acceso con privilegios elevados sobre el servidor Centos que corre la plataforma de aplicaciones glassfish.

Como preámbulo, en la fase de evaluación y recolección de información, obtenemos a través del comando la información de las interfaces de red o dirección IP del dispositivo a atacar, a continuación a través de la ejecución del

comando NMap con dos de sus my bien conocidas opciones (-A .sV) obtendremos características generales del dispositivo atacado como son el sistema operativo que ejecuta y versión del mismo, lo cual puede ser visualizado en la Figura 3.4.

Figura 3.4. Resultado de NMap –A –sV sobre servidor de aplicaciones

Como es posible observar en la Figura 3.4, el comando ejecutado anteriormente también genera un detalle de los puertos y servicios asociados entre los que se destacan, por ejemplo, el puerto 21 que aloja un servicio VSFTP.

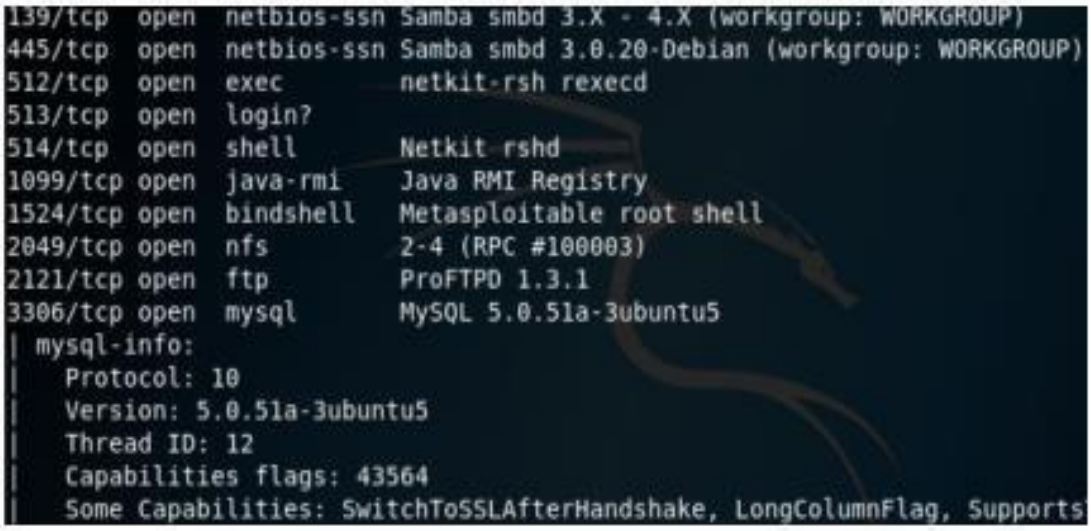


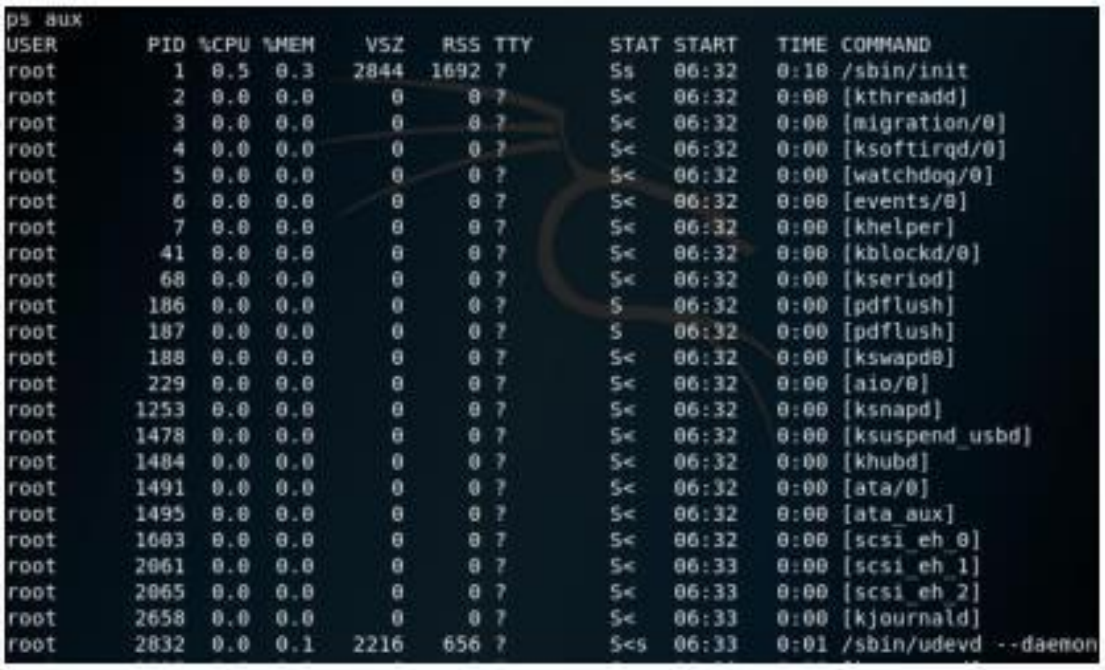
Figura 3.5. Identificación de puertos con NMap

En la Figura 3.5, que corresponde a la parte final del comando Nmap ejecutado anteriormente, se identifican además el puerto 2121 que corresponde a un servicio ftp, el cual puede ser vulnerado a través de un ataque de fuerza bruta.

Pasando a la fase de explotación, vamos a utilizar la herramienta Metasploit, a través del modulo *“use exploit/unix/misc/distcc_exec”* donde se establecen los parámetros como por ejemplo el equipo remoto de destino. La carga y ejecución de éste comando se muestra en la Figura 3.6.

Figura 3.6. Ejecución de Metasploit Distcc

Luego de realizada la carga del módulo anterior de Metasploit, lo siguiente será intentar realizar una escalación de privilegios para lo cual debemos previamente identificar los procesos que se están ejecutando en el sistema a través del comando ps-aux mostrado en la Figura 3.7.



```
ps aux
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.5  0.3   2844  1692 ?        Ss   06:32   0:10 /sbin/init
root         2  0.0  0.0      0     0 ?        S<   06:32   0:00 [kthreadd]
root         3  0.0  0.0      0     0 ?        S<   06:32   0:00 [migration/0]
root         4  0.0  0.0      0     0 ?        S<   06:32   0:00 [ksoftirqd/0]
root         5  0.0  0.0      0     0 ?        S<   06:32   0:00 [watchdog/0]
root         6  0.0  0.0      0     0 ?        S<   06:32   0:00 [events/0]
root         7  0.0  0.0      0     0 ?        S<   06:32   0:00 [khelper]
root        41  0.0  0.0      0     0 ?        S<   06:32   0:00 [kblockd/0]
root        68  0.0  0.0      0     0 ?        S<   06:32   0:00 [kseriod]
root       186  0.0  0.0      0     0 ?        S    06:32   0:00 [pdflush]
root       187  0.0  0.0      0     0 ?        S    06:32   0:00 [pdflush]
root       188  0.0  0.0      0     0 ?        S<   06:32   0:00 [kswapd0]
root       229  0.0  0.0      0     0 ?        S<   06:32   0:00 [aio/0]
root      1253  0.0  0.0      0     0 ?        S<   06:32   0:00 [ksnapd]
root      1478  0.0  0.0      0     0 ?        S<   06:32   0:00 [ksuspend_usbd]
root      1484  0.0  0.0      0     0 ?        S<   06:32   0:00 [khubd]
root      1491  0.0  0.0      0     0 ?        S<   06:32   0:00 [ata/0]
root      1495  0.0  0.0      0     0 ?        S<   06:32   0:00 [ata_aux]
root      1603  0.0  0.0      0     0 ?        S<   06:32   0:00 [scsi_eh_0]
root      2061  0.0  0.0      0     0 ?        S<   06:33   0:00 [scsi_eh_1]
root      2065  0.0  0.0      0     0 ?        S<   06:33   0:00 [scsi_eh_2]
root      2658  0.0  0.0      0     0 ?        S<   06:33   0:00 [kjournald]
root      2832  0.0  0.1   2216   656 ?        S<s  06:33   0:01 /sbin/udevd --daemon
```

Figura 3.7. Resultado de procesos ps-aux

En la Figura 3.7 se logra identificar el proceso *“udev/sbin”udevd – Daemon”* identificado con el PID 2832. Luego de buscar información del proceso anterior con el comando “*dpkg – l| grep “udev”,* se logra identificar la versión del software que se ejecuta bajo el proceso: 117-8.

A continuación con esta información, se realiza una búsqueda de posiblesexploits del software identificado utilizando el comando “searchsploit …” en el cual se ingresa como parámetro el nombre del servicio. La Figura 3.8 muestra elresultado de la ejecución de este comando.

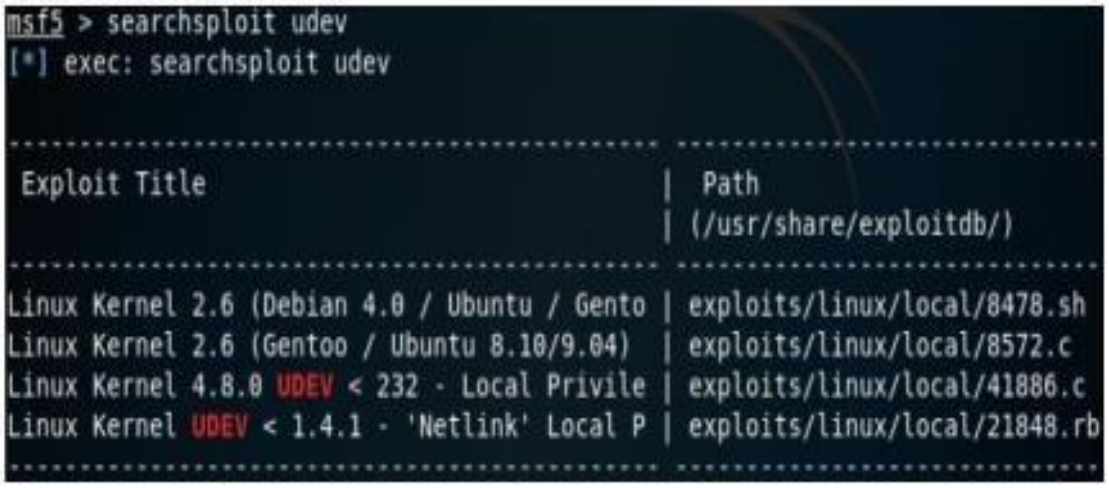


Figura 3.8. Resultado de comando de Metasploit UDEV

El objetivo de la prueba se basa en obtener acceso completo al servidor objetivo del ataque, sin embargo, el exploit evaluado e identificado en el servidor de aplicaciones tiene la característica de vulnerar el kernel del sistema operativo y acceder a los recursos de hardware, los cuales no han sido planificados como elementos de análisis y pruebas en la presente prueba de Pentesting.

3.2 Pruebas de Vulnerabilidad

A través de la herramienta Nessus Professional, fue posible realizar un Escaneo a nivel de todos los elementos básicos de Red de dos servidores de nuestro ambiente virtualizado. Nessus, para el propósito del análisis a ejecutar, ayudará a identificar y solucionar las vulnerabilidades, incluidos parches faltantes, defectos de software, malware y configuraciones erróneas en cada uno de los objetos evaluados indistintamente de su infraestructura o Sistema Operativo.

Un resumen de las vulnerabilidades y el nivel de criticidad, obtenido de la herramienta se muestran a continuación en la Figura 3.9, para el Servidor Virtualizado con Windows Server 2012 y el Servidor de Aplicaciones con Glassfish sobre Linux:

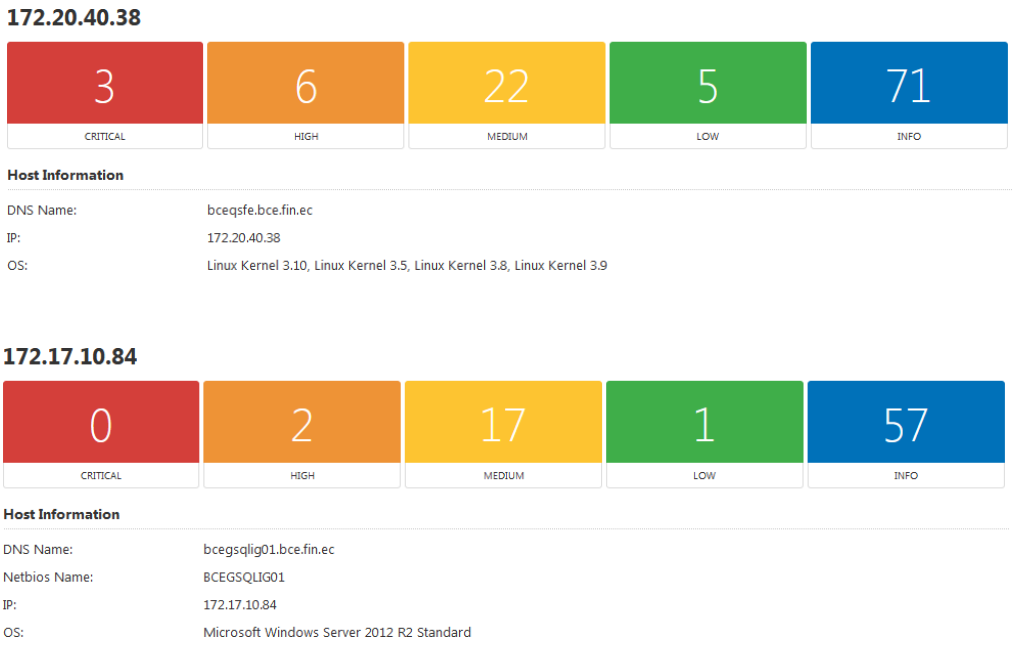


Figura 3.9. Resultado de Análisis de Vulnerabilidades en Nessus

El resultado de estas pruebas será documentado en el siguiente capítulo, en conjunto con las acciones correctivas para cada vulnerabilidad encontrada.

CAPÍTULO 4

ANÁLISIS DE RESULTADOS

4.1 Resumen de Vulnerabilidades y Hallazgos

Se procedió a ejecutar el análisis de vulnerabilidades utilizando el perfil de análisis "*Basic Network Scan*" que realiza un Escaneo completo sobre cada servidor sin acciones invasivas o que afecten su rendimiento y disponibilidad.

Para el análisis técnico de vulnerabilidades, se considerarán y reportaron con mayor detalle las vulnerabilidades catalogadas como críticas y altas, basado en la metodología propia de la herramienta de apoyo para el proceso de escaneo de vulnerabilidades, *Nessus® Professional.*

Información de Host	172.20.40.38			
Tipo de Servidor	APLICACIÓN			
Vulnerabilidades encontradas	Critica	Alto	Medio	Bajo
	1	2	10	3

DETALLE DE VULNERABILIDADES SEVERIDAD ALTA Y CRITICA:

Se han identificado múltiples Vulnerabilidades relacionadas con el mismo producto o componente:

Severidad	**Nombre**
Critica	Oracle Glassfish Server 3.1.2.x < 3.1.2.15 Multiple Vulnerabilities (July 2016 CPU)
Alto	Oracle Glassfish Server 2.1.1.x < 2.1.1.30 / 3.0.1.x < 3.0.1.15 / 3.1.2.x < 3.1.2.16 Multiple Vulnerabilities (January 2017 CPU)
Alto	Oracle Glassfish Server 3.0.1.x < 3.0.1.17 / 3.1.2.x < 3.1.2.18 (October 2017 CPU)

Descripción: De acuerdo a la documentación del producto Oracle GlassFish Server en sus versiones 3.1.2.15/ 2.1.1.30/ 3.1.2.18 , la versión 3.1.2.x instalada y funcionando en el servidor evaluado se encuentra afectada con múltiples vulnerabilidades que incluyen ataques de Denegación de Servicios y acceso no autorizado a información sensible, se resumen algunas:

- Una vulnerabilidad de divulgación de información existente en la versión bundledde la librería de transferencia *libcurl* en la función smb_request_state(), debido a la utilización de valores que son asumidos como válidos. Se puede explotar

Información de Host	172.20.40.38			
Tipo de Servidor	APLICACIÓN			
Vulnerabilidades encontradas	Critica	Alto	Medio	Bajo
	1	2	10	3

esta vulnerabilidad a través de un servidor SMB malicioso y revelar de manera arbitraria información contenida en la memoria.
- Existe una falla en el subcomponente *Web Container* que permite a un atacante remoto ejecutar comandos sin autenticarse.
- Falla en un componente de seguridad que permitiría la ejecución de código remoto sin autorización.
- Una falla en un componente de Administración que permitiría la divulgación de información sensible.
- Existe una falla en uno de los componentes de Core que permiten la ejecución de sentencias tales como: update, insert, delete; a través del protocolo SMTP.
- Falla reportada en uno de los componentes de Seguridad que permitirían la ejecución de sentencias update, insert, delete, a través de LDAP, con un alto riesgo de denegación en el servicio de la aplicación.
- Falla en un componente de seguridad que permite la ejecución de sentencias update, insert, delete, a través de HTTP, causando eventualmente una denegación en el servicio de la aplicación.

Referencias:
(CVE-2015-3237)
(CVE-2016-3607)
(CVE-2016-5528)
(CVE-2017-3239)
(CVE-2017-3247)
(CVE-2017-3249)
(CVE-2017-3250)
https://www.oracle.com/security-alerts/cpujan2017.html
https://www.oracle.com/security-alerts/cpujul2016.html
https://www.oracle.com/security-alerts/cpuoct2017.html
https://www.oracle.com/ocom/groups/public/@otn/documents/webcontent/3937099.xml

Solución: Actualizar el producto Oracle GlassFish Server a una de las versiones más recientes, tal como se indica en la referencia del sitio de Oracle.

RESUMEN DE VULNERABILIDADES SEVERIDAD MEDIO Y BAJO:

Severidad	Nombre
Medio	Oracle Glassfish Embedded Server Vulnerabilities (January 2016 CPU)
Medio	Oracle Glassfish Server 3.1.2.x < 3.1.2.19 (October 2018 CPU)
Medio	Oracle Glassfish Server 2.1.1.x < 2.1.1.29 / 3.0.1.x < 3.0.1.14 / 3.1.2.x < 3.1.2.15 Java Server Faces RCE (October 2016 CPU)

Información de Host	172.20.40.38			
Tipo de Servidor	APLICACIÓN			
Vulnerabilidades encontradas	Critica	Alto	Medio	Bajo
	1	2	10	3

Medio	SSL Certificate Cannot Be Trusted
Medio	SSH Weak Algorithms Supported
Medio	SSL Self-Signed Certificate
Medio	TLS Version 1.0 Protocol Detection
Medio	TLS Version 1.1 Protocol Detection
Medio	SSL Medium Strength Cipher Suites Supported (SWEET32)
Medio	SSL RC4 Cipher Suites Supported (Bar Mitzvah)
Bajo	Oracle Glassfish Server 3.1.2.x < 3.1.2.17 Java Server Faces Information Disclosure (April 2017 CPU)
Bajo	SSH Server CBC Mode Ciphers Enabled
Bajo	SSH Weak MAC Algorithms Enabled

Tabla 3. Formulario resumen Vulnerabilidades servidor de aplicación

En la Tabla 3 se realiza un resumen ejecutivo de las vulnerabilidades encontradas con la herramienta Nessus en la ejecución del análisis sobre el servidor de aplicaciones, entre la que destacan brechas por versiones desactualizadas de la plataforma de aplicaciones Glassfish.

Información de Host	172.17.10.84			
Tipo de Servidor	DIRECTORIO ACTIVO			
Vulnerabilidades encontradas	Critica	Alto	Medio	Bajo
	0	2	10	1

DETALLE DE VULNERABILIDADES SEVERIDAD ALTA Y CRITICA:

Severidad	**Nombre**
Alto	BMC SNMP Agent Default Community Name (public)

Descripción: El servicio SNMP server, activado en el puerto 8161 como parte de la aplicación BMC Patrol, registra el parámetro *community name* con el valor *´public´*. Esta información podría ser utilizada de manera maliciosa inclusive para realizar cambios en la configuración del servidor.

Información de Host	172.17.10.84			
Tipo de Servidor	DIRECTORIO ACTIVO			
Vulnerabilidades encontradas	Critica	Alto	Medio	Bajo
	0	2	10	1

Solución: Deshabilitar el servicio SNMP a través de la aplicación BMC Patrol, o en su defecto realizar el cambio del valor en el parámetro *community name*

Severidad	**Nombre**
Alto	SSL Version 2 and 3 Protocol Detection

Descripción: El servicio SSL en el servidor realiza la encriptación de tráfico utilizando protocolos con debilidades conocidas, aceptando conexiones en versiones SSL 2.0 o SSL 3.0, las cuales se encuentran afectadas por una falla en la sesión de renegociación y cuando ocurren reanudación de las sesiones. Una falla de este tipo puede ser aprovechada para generar ataques de tipo MITM (Man-in-the-Middle) o en su defecto que la comunicación entre el cliente y el servidor sea desencriptada.
A la fecha ninguna versión de SSL cumple con los requisitos mínimos del estándar de seguridad PCI relacionados a “criptografía robusta”.

Referencias:
https://www.schneier.com/academic/paperfiles/paper-ssl.pdf

https://www.openssl.org/~bodo/ssl-poodle.pdf

https://www.imperialviolet.org/2014/10/14/poodle.html

https://tools.ietf.org/html/rfc7507

https://tools.ietf.org/html/rfc7568

Solución: Es recomendable realizar la deshabilitación de estos protocolos en su totalidad, para lo cual se debe consultar con la documentación relacionada al servidor de aplicaciones. Además se sugiere utilizar TLS 1.1 o superior.

RESUMEN DE VULNERABILIDADES SEVERIDAD MEDIO Y BAJO:

~~**Severidad**~~	~~**Nombre**~~
~~Medio~~	~~SSL Certificate Cannot Be Trusted~~
~~Medio~~	~~SSL Self-Signed Certificate~~
~~Medio~~	~~TLS Version 1.0 Protocol Detection~~
~~Medio~~	~~TLS Version 1.1 Protocol Detection~~
~~Medio~~	~~SMB Signing not required~~
~~Medio~~	~~SSL Certificate Signed Using Weak Hashing Algorithm~~
~~Medio~~	~~SSL Certificate with Wrong Hostname~~
~~Medio~~	~~SSL Medium Strength Cipher Suites Supported (SWEET32)~~
~~Medio~~	~~SSL RC4 Cipher Suites Supported (Bar Mitzvah)~~

Información de Host	172.17.10.84			
Tipo de Servidor	DIRECTORIO ACTIVO			
Vulnerabilidades encontradas	Critica	Alto	Medio	Bajo
	0	2	10	1
Medio	SSLv3 Padding Oracle On Downgraded Legacy Encryption Vulnerability (POODLE)			
Bajo	SSL Certificate Chain Contains RSA Keys Less Than 2048 bits			

Tabla 4. Formulario resumen Vulnerabilidades servidor de directorio activo

En la Tabla 4, se entrega del mismo modo un resumen ejecutivo del estado de las vulnerabilidades encontradas durante el análisis del servidor de directorio activo, el cual en la simulación de nuestro ambiente corresponde al servicio de autenticación.

4.2 Acciones de Remediación

Las acciones de remediación, en base a las brechas de seguridad identificadas en el proceso de la prueba de pentesting y el análisis de vulnerabilidades se detallan en la Tabla 6 para cada elemento de la topología analizado:

ELEMENTO	VULNERABILIDAD	MITIGACIÓN
Router R1	Se Identificaron vulnerabilidades relacionadas con la configuración de seguridad del dispositivo, exponiendo a que sea administrado remotamente de manera maliciosa.	Aplicar metodología y técnicas de Hardening sobre éste y todos los dispositivos activos y pasivos de la red.
Servidor de Aplicaciones	Versión de plataforma de aplicaciones Glassfish instalada posee distintas vulnerabilidades a nivel de software y SSL.	Actualizar el producto Oracle GlassFish Server a una de las versiones más recientes, tal comose indica en la referencia del sitio de Oracle.
Servidor de Directorio Activo	El servicio SNMP server, activado en el puerto 8161. El servicio SSL en el servidor realiza la encriptación de tráfico utilizando protocolos con debilidades conocidas, aceptando conexiones en versiones SSL 2.0 o SSL 3.0,	Deshabilitar el servicio SNMP, o en su defecto realizar el cambio del valor en el parámetro *community name.* Es recomendable realizar la deshabilitación de estos protocolos en su totalidad, para lo cual se debe consultar con la documentación relacionada al servidor de aplicaciones. Además se sugiere utilizar TLS 1.1 o superior.

Tabla 5. Acciones de Remediación

CONCLUSIONES

La utilización de ambientes virtualizados sobre plataformas que utilizan recursos de software para emular elementos de hardware, se constituyen en la actualidad como una herramienta de mucha utilidad que puede ayudar a Instituciones Comerciales y Financieras a realizar pruebas de seguridad sobre sus sistemas considerando la criticidad de los mismos y la afectación económica y reputacional de la que podrían ser víctima ante eventos de ataques cibernéticos.

El desarrollo de pruebas de pentesting, aplicando la metodología PTES, involucra un alto conocimiento en el campo del Networking y la Administración de la Infraestructura, así como conocimientos profundos en el ámbito de la seguridad informática.

La generación de resultados de las pruebas de Pentesting y de vulnerabilidades se deben ajustar al producto esperado por los administradores y propietarios de la plataforma, con el objetivo de que pueda ser entendible y otorgar en su contenido una visión general y en los casos específicos que aplique del estado de la infraestructura de tecnología y el comportamiento de los elementos que pueden ser vulnerados y los riesgos que deben ser evaluados en conjunto con las entidades que los administran y procesan.

RECOMENDACIONES

La construcción de ambientes virtualizados para brindar una gran funcionalidad y utilidad debe ser diseñado con los recursos necesarios para el comportamiento y performance adecuado de los dispositivos y elementos que se integran.

Existen limitaciones en la plataforma GNS3 en cuanto a los productos y appliance que dependen de licencias de los fabricantes. Por lo general y en los mejores casos es posible obtener una versión de prueba del sistema operativo de una plataforma, por lo cual esta recomendación debe ser tomada en consideración en el momento del diseño de los elementos de red que se integrarán en los ambientes virtuales.

El campo y objetivo de las pruebas de Pentesting, sobre el presente proyecto, puede ser ampliado específicamente a las aplicaciones que son publicadas en plataformas WEB o cliente servidor, siempre que se cuente con los recursos de software y los derechos de propiedad intelectual asociados a su utilización.

BIBLIOGRAFÍA

[1] S. Lowe. *Mastering VMware vsphere 5*. Edition 1. John Wiley & Sons Incorporated, 2011

[2] [Online] The Pentest Execution Standard. Disponible en http://www.pentest-standard.org/index.php/Main_Page

[3] [Online] Getting Started with GNS3. Disponible en https://docs.gns3.com/docs/

[4] C. Welsh. *Gns3 network simulation guide*. Edition 1. Packt Publishing Limited, 2013.

[5] G. Weidman. *Penetration testing: A hands-on introduction to hacking*. Edition 1. No Starch Press Incorporated, 2013

[6] [Online] The Pentest Execution Standard. Disponible en http://www.pentest-standard.org/index.php/Main_Page

[7] [Online] What is Kali Linux? Disponible en https://www.kali.org/docs/introduction/what-is-kali-linux/

[8] [Online] Nessus Professional: hoja de Datos. Disponible en https://static.tenable.com/marketing/datasheets/Datasheet-Nessus_Professional_es-la.pdf