



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE CIENCIAS DE LA
COMPUTACIÓN**

CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

**TRABAJO DE TITULACIÓN, PREVIO LA OBTENCIÓN DEL
TÍTULO DE INGENIERO EN SISTEMAS E INFORMÁTICA**

**TEMA: PLATAFORMA EXPERIMENTAL DE
CIBERSEGURIDAD, SOBRE INFRAESTRUCTURA
VIRTUALIZADA PARA MITIGAR LOS ATAQUES DE
DENEGACIÓN DE SERVICIO**

AUTOR: MORALES JERIA, MIGUEL ALEJANDRO

DIRECTOR: ING. FUERTES, WALTER, PhD.

SANGOLQUÍ

ENERO 2016

CERTIFICADO**DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN****CARRERA DE INGENIERÍA EN SISTEMAS****CERTIFICACIÓN**

Certifico que el trabajo de titulación, "**PLATAFORMA EXPERIMENTAL DE CIBERSEGURIDAD SOBRE INFRAESTRUCTURA VIRTUALIZADA PARA MITIGACIÓN DE ATAQUES DE DENEGACIÓN DE SERVICIO**" realizado por el señor **MORALES JERIA MIGUEL ALEJANDRO** ha alcanzado los objetivos específicos de la tesis. Además ha sido revisado en su totalidad y analizado por el software antiplagio, el mismo cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de Fuerzas Armadas ESPE.

Sangolquí, 07 de enero del 2016




Ing. Walter Fuertes, PhD
DIRECTOR

AUTORÍA DE RESPONSABILIDAD**DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN****CARRERA DE INGENIERÍA EN SISTEMAS**

Yo, **MORALES JERIA MIGUEL ALEJANDRO**, con cédula de identidad N° 1719433656, declaro que este trabajo de titulación "**PLATAFORMA EXPERIMENTAL DE CIBERSEGURIDAD SOBRE INFRAESTRUCTURA VIRTUALIZADA PARA MITIGACIÓN DE ATAQUES DE DENEGACIÓN DE SERVICIO**" ha sido desarrollado considerando los métodos de investigación existentes, así como también se ha respetado los derechos intelectuales de terceros considerándose en las citas bibliográficas.

Consecuentemente declaro que este trabajo es de mi autoría, en virtud de ello me declaro responsable del contenido, veracidad y alcance de la investigación mencionada.

Sangolquí, 07 de enero del 2016




Miguel Alejandro Morales Jeria
C.C 1719433656

AUTORIZACIÓN**DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN****CARRERA DE INGENIERÍA EN SISTEMAS****AUTORIZACIÓN**

Yo, **MORALES JERIA MIGUEL ALEJANDRO**, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar en la biblioteca Virtual de la institución el presente trabajo de titulación "**PLATAFORMA EXPERIMENTAL DE CIBERSEGURIDAD SOBRE INFRAESTRUCTURA VIRTUALIZADA PARA MITIGACIÓN DE ATAQUES DE DENEGACIÓN DE SERVICIO**" cuyo contenido, ideas y criterios son de mi autoría y responsabilidad.

Sangolquí, 07 de enero del 2016



Miguel Alejandro Morales Jeria
C.C 1719433656

DEDICATORIA

A Dios y a la Virgen que me han dado la fuerza para seguir adelante con este trabajo.

A mi padre y madre que han sido mi apoyo y mi fuerza, que han sido ejemplo a seguir, que con sus consejos han logrado convertirme en el hombre que soy.

AGRADECIMIENTO

A mi Director de Tesis Dr. Walter Fuertes, que con esfuerzo y paciencia ha guiado acertadamente a la conclusión exitosa del presente trabajo.

Este trabajo va dedicado a mi familia, en la cual encontré amor, comprensión, apoyo y fuerza incondicional en todo momento que lo necesité.

A todos los docentes que formaron parte de mi vida universitaria, los cuales han brindado todo el conocimiento necesario para poder afrontar los diversos retos a los cuales me he enfrentado.

ÍNDICE

CERTIFICADO	ii
AUTORÍA DE RESPONSABILIDAD	iii
AUTORIZACIÓN	iv
DEDICATORIA	v
AGRADECIMIENTO	vi
ÍNDICE	vii
ÍNDICE DE TABLAS	xi
ÍNDICE DE FIGURAS.....	xii
GLOSARIO	xiv
RESUMEN.....	xvi
ABSTRACT.....	xvii
CAPÍTULO I INTRODUCCIÓN	1
1.1 Antecedentes	1
1.2 Planteamiento del problema	1
1.3 Justificación.....	3
1.4 Objetivos	3
1.4.1 Objetivo general	3
1.4.2 Objetivos específicos	3
1.5 Alcance	4
1.6 Herramientas	4
CAPÍTULO II MARCO TEÓRICO	6
2.1 Herramientas para la seguridad de redes	6
2.1.1 Sistema de detección de intrusos (IDS)	6
2.1.2 Sistema de prevención de intrusos (IPS).....	6
2.1.3 Gestión unificada de amenazas (UTM).....	7

2.1.4	Firewall	8
2.1.5	Waf.....	8
2.2	Tipos de ataques de DoS	9
2.2.1	Syn Flood	9
2.2.2	Broadcast.....	9
2.2.3	ICMP Flood.....	10
2.2.4	UDP Flood	10
2.2.5	DNS Flood	10
2.2.6	Ping of death (Ping de la muerte).....	10
2.2.7	Land.....	11
2.2.8	Teardrop	12
2.2.9	Smurf.....	12
2.3	Firewall con IPTables	13
2.4	Herramientas para la ejecución de ataques.....	15
2.4.1	Hping3.....	15
2.4.2	Slowris	16
2.4.3	Ping of Death	16
2.5	Tecnologías de virtualización.....	16
2.5.1	OracleVM Virtual Box.....	16
2.5.2	VMware Workstation.....	17
CAPITULO III PLATAFORMA EXPERIMENTAL DE		
CIBERSEGURIDAD, SOBRE INFRAESTRUCTURA VIRTUALIZADA.....		
3.1	Entorno virtual de red.....	18
3.2	Diseño de la topología	19
3.3	Configuración de escenario virtual.....	20
3.3.1	Configuración de la LAN mediante máquinas virtuales	20

3.3.2	Configuración de DMZ.....	20
3.3.3	Configuración de los Routers-Firewalls.....	20
3.4	Configuración de máquinas virtuales	21
3.4.1	Direccionamiento IP.....	21
3.4.2	Servidor de base de datos	21
3.4.3	Servidor de aplicaciones Web.....	22
3.5	Diseño e implementación de una aplicación tipo escritorio para administrar una plataforma virtual	22
3.5.1	Fase 1: Planificación	23
3.5.2	Fase 2: Diseño	27
3.5.3	Fase 3: Codificación.....	31
3.5.4	Fase 4: Pruebas.....	31
3.6	Diseño e implementación de algoritmos para el despliegue automático del entorno virtual de red.....	31
3.7	Implementación de ataques	33
3.8	Mecanismo para contrarrestar ataques a REDES IP	33
3.8.1	Resolución de anomalías (Línea base).....	33
3.8.2	Resolución de anomalías (Optimizado)	34
3.9	Pruebas	35
3.9.1	Pruebas funcionales.....	36
3.9.2	Pruebas IPP Metrics	39
CAPITULO IV.....		41
EMULACIÓN DE ATAQUES EN ENTORNO VIRTUAL DE RED		41
4.1	Implementación de ataques	41
4.1.1	Slowris	41
4.1.2	Hping3.....	43

	x
4.1.3 Ping of Death	44
4.2 Mitigación de ataques.....	45
CAPITULO V	47
EVALUACIÓN DE RESULTADOS	47
5.1 Evaluación del algoritmo de resolución de anomalías de Firewall: línea base vs optimizado	47
5.1.1 Evaluación de recursos computacionales.....	47
CAPITULO VI.....	62
CONCLUSIONES Y RECOMENDACIONES.....	62
6.1 Conclusiones	62
6.2 Recomendaciones	62
6.3 Trabajos futuros.....	63
REFERENCIAS BIBLIOGRÁFICAS.....	64

ÍNDICE DE TABLAS

Tabla 1: Ataques a la capa de infraestructura	1
Tabla 2: Tipos de cadenas en los IPtables.....	13
Tabla 3: Ordenes básica para reglas de firewall	14
Tabla 4: Parámetros de reglas de firewall	15
Tabla 5: Parámetros de configuración de las máquinas virtuales	21
Tabla 6: Herramientas para realizar ataques DoS.....	33
Tabla 7: Prueba de Unidad	36
Tabla 8: Prueba de Integración	37
Tabla 9: Prueba de Validación.....	37
Tabla 10: Prueba de alto nivel.....	38
Tabla 11: Matriz de definición de variables	39
Tabla 12 Datos – Memoria ataque Slowris	48
Tabla 13 Datos – Memoria ataque HPING3	50
Tabla 14 Datos – Memoria ataque PING OF DEATH	52
Tabla 15 Datos uso procesador ataque SLORIS	53
Tabla 16 Datos uso procesador ataque HPING3	54
Tabla 17 Datos uso procesador ataque Ping of Death	55
Tabla 18 Datos - Latencia Slowris	57
Tabla 19 Datos – Latencia ataque Hping3	58
Tabla 20 Datos – Latencia ataque Ping of Death.....	60

ÍNDICE DE FIGURAS

Figura 1. Diagrama de red con la protección de un firewall.....	8
Figura 2. Formato de cabecera IP	11
Figura 3. Ataque LAND.....	12
Figura 4. Ventana de inicio de Oracle VM VirtualBox	18
Figura 5. Diagrama de Red	19
Figura 6. Archivo /etc/sysctl.conf	20
Figura 7. Fases de la metodología XP	22
Figura 8. Historia de Usuario: Registrar nuevo usuario.....	23
Figura 9. Historia de Usuario: Autenticar usuario	23
Figura 10. Historia de Usuario: Gestión de entorno virtual.....	24
Figura 11. Historia de Usuario: Gestión de ataques	24
Figura 12. Historia de Usuario: Gestión de mitigación	24
Figura 13. Historia de Usuario: Gráfica de datos.....	25
Figura 14. Historia de Usuario: Gestionar reporte	25
Figura 15. Plan de entrega del proyecto.....	25
Figura 16. Tareas establecidas en cada iteración	27
Figura 18. Diagrama de secuencia proyecto	29
Figura 19. Arquitectura de la Aplicación.....	30
Figura 20. Flujograma general del algoritmo de despliegue.....	32
Figura 21. Algoritmo para resolver anomalías.....	33
Figura 22. Algoritmo para resolver anomalías (Repotenciado).....	34
Figura 23. Etapas en la prueba del Software (Pressman ,2005).....	36
Figura 24. Módulo de ataques.....	41
Figura 25. Ataque Slowris	42
Figura 26. Ataque HPING3	43
Figura 27. Ataque Ping Of Death.....	44
Figura 28. Módulo de Servicios.....	45
Figura 29. Uso de memoria Slowris_Old.....	47
Figura 30. Uso de memoria Slowris_NEW	48
Figura 31. Uso de memoria HPING3_OLD	49
Figura 32. Uso de memoria HPING3_NEW	50

Figura 33. Uso de memoria Ping of Death Old.....	51
Figura 34. Uso de memoria Ping of Death New.....	51
Figura 35. Uso de CPU ataque SLOWRIS	53
Figura 36. Uso de CPU ataque HPING3	54
Figura 37. Uso de CPU ataque Ping of Death.....	55
Figura 38. Latencia Slowris_Old	56
Figura 39. Latencia Slowris_New.....	56
Figura 40. Latencia Hping3_Old	58
Figura 41. Latencia Hping3_New	58
Figura 42. Latencia Ping of Death_Old	59
Figura 43. Latencia Ping of Death_New.....	60

GLOSARIO

Código abierto (Open Source): Es un software cuyo código fuente es de dominio público.

Denegación de servicio (DoS): Es un tipo de ataque cuya finalidad es volver inaccesibles los recursos de red de una organización

Dirección IP: Etiqueta numérica que identifica a una interfaz de red que utiliza el protocolo IP.

Dirección de Broadcast: Transmisión de mensaje a través de un nodo a múltiples receptores.

Demilitarized zone (DMZ): Zona desmilitarizada, que se ubica entre la red interna y la externa.

Domain Name System (DNS): Sistema de nombres de dominio, es un sistema de nomenclatura de recursos que están conectados a una red.

Extreme Programming (XP): Programación extrema, es una metodología de desarrollo de software.

Gateway: Puerta de enlace de una red.

GUI (Graphic User Interface): Interfaz gráfica de usuario que permite la comunicación entre el usuario y el programa.

Hypertext Transfer Protocol(HTTP): Es un protocolo utilizado para la distribución de información a través de la Internet.

IPtables: Es una herramienta que permite el filtrado de paquetes, disponible en el núcleo de Linux.

Java: Es un lenguaje de programación orientada a objetos.

JFrame: Es una clase utilizada por la librería SWING, para generar ventanas.

Kermel: Núcleo que constituye la parte fundamental de un Sistema operativo.

Red de área local (LAN): Conjunto de equipos que pertenecen a una organización y están conectados dentro de un área geográfica pequeña.

Red de área empleada (WAN): Conjunto de equipos que abarca varias ubicaciones físicas.

Malware: Es un software que contiene código malicioso que puede dañar el equipo.

Secure Shell (SSH): Es un protocolo que permite la interpretación segura de órdenes, sirve para acceder a máquinas remotas en una red.

Shell Script: Es un programa de computadora que se ejecuta en el shell de Linux.

Sistema de Gestión de Base de Datos (SGBD): Consiste en una colección de datos interrelacionados y un conjunto de programas para acceder a los mismos.

Tabla de Enrutamiento: Tabla que indica cual va a ser el camino o ruta que va a llevar un paquete.

TCP/IP: Descripción de protocolo de transmisión de información,

Transmission Control Protocol (TCP): Protocolo de transporte de la arquitectura de protocolos TCP/IP.

User Datagram Protocol (UDP): Protocolo de transporte de la arquitectura de protocolos TCP/IP.

Virtual environment: Escenario virtual.

RESUMEN

Los ataques de DoS (Denegación de Servicio) tienen como finalidad saturar los recursos de red de una organización por un periodo de tiempo indefinido. Estos tipos de ataques aprovechan tanto las vulnerabilidades del protocolo como los errores de configuración en los diferentes dispositivos de seguridad. El presente trabajo se enfoca en la evaluación de ataques de tipo DoS utilizando como plataforma experimental un entorno virtual con segmentaciones LAN, DMZ y WAN. Las herramientas evaluadas fueron Hping3, Slowris y Ping de la muerte. Las herramientas fueron instaladas sobre Linux. Para validar esta investigación se realizó la repotenciación del firewall a través de IPTables, para que sirva como mecanismo de mitigación de los ataques. Finalmente se evaluó el consumo de memoria, el uso de CPU y el ancho de banda durante los ataques.

Palabras clave:

- **DENEGACIÓN DE SERVICIO**
- **IPTABLES**
- **ENTORNO VIRTUAL.**

ABSTRACT

DoS attacks (Denial of Service) aim to saturate network resources of an organization for an indefinite period of time. These types of attacks exploit vulnerabilities both protocol and configuration errors in the various safety devices. This paper focuses on the evaluation of DoS attacks using experimental platform as a virtual environment with LAN, WAN and DMZ segmentation. The tools were evaluated Hping3, Slowris and Ping of Death. The tools were installed on Linux. To validate this research repowering firewall is made through IPTables, to serve as a mechanism to mitigate the attacks. Finally memory consumption, CPU utilization and network performance was evaluated.

Key words:

- **DENIAL OF SERVICE**
- **IPTABLES**
- **VIRTUAL ENVIRONMENT**

CAPÍTULO I INTRODUCCIÓN

1.1 Antecedentes

Con la aparición de la redes IP, también apareció el problema de los ataques, estos han sido cada vez más sofisticados y siempre esperando encontrar un error en la configuración o diseño de la misma. Los autores de estos ataques dejaron de ser personas con grandes conocimientos sobre informática, ahora con la aparición de varias herramientas no es necesario tener un conocimiento mayor para poder realizar daños en la red.

La Ciberseguridad se puede definir como el conjunto de acciones de carácter preventivo que buscan asegurar las redes para evitar ataques; la Ciberseguridad resulta un elemento importante ya que si no se controla el ciberespacio, desde ahí puede existir una amenaza a la organización.


1.2 Planteamiento del problema

Los ataques a las redes IP con el paso del tiempo han ido aumentando, esto debido a las vulnerabilidades del protocolo TCP/IP y nuevas herramientas para explotar las mismas, cada vez que aparece un tipo de ataque lo hace de manera más agresiva causando con esto pérdidas, en algunos casos incalculables.

La empresa Akamai (Akamai, 2015) elaboró un estudio en el año 2014 el cual dio como resultado que el mayor porcentaje de ataques se concentra en la capa de infraestructura (capa 3 y 4) con un 89.29% mientras que en la capa de aplicación se hace presente con el 10.71%. En la Tabla 1, se presenta una estadística que muestra los ataques más comunes realizados a capa de infraestructura.

Tabla 1:

Ataques a la capa de infraestructura

Ataque	Porcentaje
SYN Floods	26%
Continua 	

UDP Floods	25%
DNS	8%
ICMP	7%
ACK Floods	5%
CHARGEN	5%
SNMP	3%

Fuente: (Akamai, 2015)

Los tipos de ataques se pueden dividir en activos y pasivos, el primero tiene como finalidad producir daños en la red, mientras que el segundo tiene como finalidad utilizar recursos y tener acceso al sistema.

Los tipos de ataques pueden ser los siguientes: Ataque de denegación de servicio (DoS), ataque de modificación, ataque de interceptación, ataque de intromisión y ataque de suplantación, entre otros.

Los ataques de denegación de servicio (Denial of Service, DoS) consisten en el trabajo de una o más máquinas que tiene como finalidad el lograr que la víctima no haga su trabajo. El afectado puede ser un servidor de red, cliente o router, un enlace de red o una red completa (RFC4732)

Existen también diferentes tipos de intrusos entre los cuales tenemos a: hackers, crackers, sniffers, phreakers, spammer y personal interno, entre otros; todos estos esperando realizar un daño para obtener un beneficio sea este económico o de diferente tipo.

- **Diseño de la investigación**

- **Pregunta principal**

- ¿Cuáles son las plataformas más adecuadas para experimentación de la Ciberseguridad, que permitan reducir los costos de inversión de hardware?

- **Preguntas secundarias**

- ¿Qué tipos de ataques de denegación de servicio existen?

- ¿Cómo diseñar e implementar una topología de experimentación de ataques de DoS a las redes IP?

- ¿Cómo simular ataques de redes IP sobre plataformas virtuales?

¿Qué técnicas existen para neutralizar los ataques de denegación de servicio?

1.3 Justificación

Con la aparición de las aplicaciones Web, surgió un riesgo el cual se hace presente en el entorno que rodea las mismas, se han identificado algunas maneras de ataques que podrían desencadenar en el mal funcionamiento, el cual se puede dar por diferentes razones.

Las vulnerabilidades que están presentes en el protocolo de red TCP/IP, han permitido que personas mal intencionadas atenten contra el buen funcionamiento de diferentes servicios o aplicaciones que se encuentren en la red, por esto la importancia de este proyecto el cual desarrollará e implementará una plataforma donde se podrán probar las aplicaciones web, con el fin de ver si son susceptibles a un tipo de ataque determinado, esto servirá para poder corregir los errores en las respectivas aplicaciones.

1.4 Objetivos

1.4.1 Objetivo general

Diseñar e implementar una plataforma experimental de Ciberseguridad, utilizando herramientas Open Source sobre Infraestructura Virtualizada para detectar, controlar y mitigar los Ataques de Seguridad Informática de tipo DoS.

1.4.2 Objetivos específicos

- i. Investigar los tipos de ataques de denegación de servicio, el marco teórico referencial y el estado del Arte.
- ii. Diseñar e implementar una topología de red para simular ataques a redes utilizando herramientas open source.
- iii. Realizar ataques de Denegación de Servicio y configurar la forma de mitigarlos.
- iv. Evaluar, validar, analizar e interpretar los resultados.

1.5 Alcance

Se realizará ataques de tipo DoS, esto incluirá un listado de las posibles variedades de este tipo de ataques, así como su consecuencia en la red. El plan de tesis de grado incluirá el desarrollo de una plataforma experimental utilizando herramientas Open Source sobre infraestructura virtualizada, en el cual se evaluará los daños causados por ataques..

1.6 Herramientas

A continuación se describe brevemente las herramientas elegidas para el desarrollo del software:

- Ubuntu 14.04 LTS Desktop: Una distribución GNU/Linux la cual consiste en una recopilación de aplicaciones y herramientas junto al núcleo Linux. Se encuentran empaquetadas de una determinada manera y con utilidades extras para facilitar la configuración del sistema (Cuaresma,2015).
- NetBeans 7.4: Es un IDE de desarrollo de código abierto, que permite realizar aplicaciones de tipo Escritorio, móviles y Web. Soporta varios lenguajes como Java, HTML5, PHP y C++ (Oracle,2015).
- Virtual Box versión 5.0.6: Es un software de virtualización multiplataforma, tiene licencia tipo GLP (General Public License) versión 2. Esta herramienta permite la instalación de sistemas operativos adicionales dentro del sistema anfitrión (Oracle, Virtual Box, 2015).
- Power Designer versión 16.1: Herramienta producida por Sybase, es una solución para el modelamiento de procesos de negocio en la industria, también modela arquitecturas de software y empresarial (Designer, 2015).
- MySQL versión 5.5.44: Es un software de gestión de base de datos de código abierto que permite a los usuarios almacenar, organizar y recuperar datos (Sverdlov, 2015).

- Shell Script: El término Shell script o solo script es usado para referirse a un programa escrito en el lenguaje Shell (creado por Bourne Shell), el cual corre el Unix Shell (Seebach, 2008).
- Sar: Es una herramienta que forma parte del paquete sysstat que permite la supervisión del desempeño de los distintos subsistemas de Linux como por ejemplo CPU, memoria en tiempo real (Seebach, 2008)
- Mtr: Es una herramienta de diagnóstico de red de gran alcance que permite a los administradores evaluar y aislar errores. Mtr representa una evolución del comando traceroute el cual proporciona información similar a esta (Linode,2015).

CAPÍTULO II

MARCO TEÓRICO

2.1 Herramientas para la seguridad de redes

2.1.1 Sistema de detección de intrusos (IDS)

Es un mecanismo de seguridad cuya finalidad es detectar intrusos dentro de una red o host (Sartakov, 2015). La manera de trabajar de esta herramienta consiste en escuchar el tráfico dentro de la red para ver si existe actividad sospechosa con la finalidad de reducir el riesgo. Existen dos tipos de IDS:

1) N-IDS

Sistema de detección de intrusos de red, controla el tráfico en la red en busca de actividades sospechosas que podrían tener como finalidad un ataque o actividad no autorizada (Al-Jarrah & Arafat, 2014).

La ubicación del N-IDS en una red puede ser dentro de una red grande para controlar el tráfico o en una red pequeña para controlar el tráfico para un determinado servidor. El N-IDS juega un papel importante de la mano con la seguridad primaria de una red tales como son firewalls, encriptación entre otros métodos de autenticación.

2) H-IDS

Sistema de detección de intrusos que protege un determinado host (Al-Jarrah & Arafat, 2014). Su software es compatible con los diferentes sistemas operativos como Windows, Solaris, Linux, etc. La ubicación estratégica dentro de un host permite que se realice un análisis de la información o registros dentro de la computadora, y también la de capturar paquetes que llegan y salen para verificar si hay algún intruso dentro de la red.

2.1.2 Sistema de prevención de intrusos (IPS)

Es un mecanismo para proteger tanto un host como una red, su ubicación a menudo es detrás del firewall (Kim, J. M. et al 2015). Los IPS suelen considerarse como una expansión de los IDS, aunque en realidad es más que una forma de

control de quien tiene acceso a la red. El IPS controla acceso a una red basado en el contenido de la solicitud de quien quiere acceder.

Un buen sistema de prevención de intrusiones no sólo debe detectar intrusos, sino que también controla el acceso a una red. Existen dos tipos de IPS disponibles para mayor seguridad:

- Host IPS
- Network IPS

1) Host IPS

Es el lugar donde se evitan las intrusiones desde un equipo con una IP específica, estos son solo aplicaciones de software que son muy buenos al momento de detectar entradas no deseadas después de haberse producido el descifrado, el problema de la prevención basada en host es que debe estar instalado en todos los ordenadores de la red.

2) Network IPS

Son plataformas especialmente formadas de hardware y software que están diseñados para detectar, analizar e informar sobre los eventos relacionados con la seguridad. Al igual que un típico firewall, el Network IPS posee al menos dos interfaces de red, una de ellas de manera interna y otra externa. De esta manera el momento que aparecen paquetes tanto de manera interna o externa se activa el sistema de detección como cualquiera IDS que determina si el paquete examinado es considerado como una amenaza para la red.

2.1.3 Gestión unificada de amenazas (UTM)

Apareció en el año 2007 como una buena opción para las PYMES, por tratarse de un dispositivo que concentra diferentes tecnologías integradas que cubren las necesidades básicas dentro de una empresa del tipo antes mencionada(Gerentes,2015).

El UTM integra múltiples funciones de seguridad entre las cuales tenemos:

- Realiza funciones de firewall
- Detección de Intrusos(IDS/IPS)

- Filtra contenido Web
- Funciones antivirus, anti-spyware y anti-span

2.1.4 Firewall

Firewall (cortafuegos), es un sistema de seguridad colocado entre la red privada y el exterior de tal manera que todos los paquetes entrantes y salientes tienen que pasar a través de él (Firewall Design and Analysis, 2011). El firewall es una herramienta que está configurada para revisar el tráfico en la red y permitir en base de ciertas reglas el acceso de personas a la misma.

Un firewall puede ser de dos maneras: física (hardware) o mediante software o también puede resultar como la combinación de las dos, a continuación la Fig.1 muestra un diagrama de red en el cual se utiliza un firewall para proveer de seguridad en el perímetro.

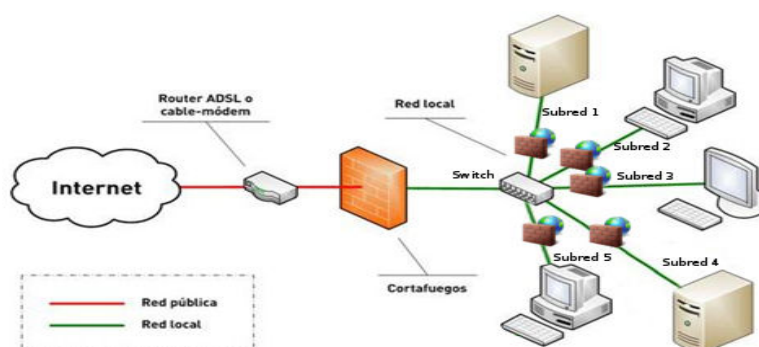


Figura 1. Diagrama de red con la protección de un firewall

Una adecuada configuración del firewall puede contribuir de gran manera a la protección necesaria en una red, pero no es el único dispositivo que se debería utilizar, lo óptimo sería la combinación de diversos dispositivos para lograr así un porcentaje alto de seguridad en la red.

2.1.5 Waf

Web Application Firewall (Firewall de aplicación Web) es una capa indispensable para proteger los sistemas en línea (Applelt,D et al, 2015). Esta opción de seguridad se la puede considerar como una herramienta complementaria para la protección de redes, su funcionamiento consiste en analizar el tráfico de la red con la finalidad de proteger los datos transmitidos, la reglas usadas para la configuración

del firewall contemplan ataques comunes como por ejemplo SQL injections, Cross Site Scripting, Buffer Overflows, etc.

La creación de esta herramienta se basó en la facilidad de acceso a comparación con una aplicación de escritorio, no se debe olvidar que esta opción de seguridad combinada con otras puede dar como resultado un grado de seguridad aceptable.

2.2 Tipos de ataques de DoS

2.2.1 Syn Flood

El método de inundación SYN Flood explota la función de retención de estado del protocolo TCP después de que un paquete SYN ha sido recibido en un puerto el cual ha sido colocado en el estado de LISTEN (Bhat, S et al, 2015). Este ataque es de tipo DoS, su metodología como se lo explico anteriormente consiste en enviar numerosas peticiones de tipo SYN; normalmente en el proceso de comunicación entre un cliente y un servidor hay intercambio de mensajes los cuales se pueden resumir de la siguiente manera.

- El cliente hace una petición de la conexión enviando un mensaje SYN al servidor.
- El servidor acepta esta petición enviando un mensaje SYN-ACK hacia el cliente.
- El cliente responde nuevamente con un ACK, y finalmente se establece la conexión.

2.2.2 Broadcast

Este tipo de ataque consiste en utilizar la dirección de broadcast de la red, como dirección destino de un paquete IP, de esta forma el router ubicado en la red lo reconoce y se ve obligado a enviar ese paquete a todos los computadores de la red, logrando con esto que se consuma el ancho de banda dando como resultado una denegación de servicio.

2.2.3 ICMP Flood

Es una técnica DoS cuya finalidad es saturar el ancho de banda que posee la víctima, su metodología consiste en enviar un número alto de paquetes ICMP de tamaño suficiente como para sobrecarga tanto en la red como en el sistema de la víctima.

Dependiendo el ancho de banda la sobrecarga sobre de la misma varia, llegando a un punto donde no se puede manejar el tráfico que se encuentra en la red, terminando por realizar la esperada denegación de servicio.

2.2.4 UDP Flood

Un ataque UDP Flood es posible cuando un atacante envía paquetes UDP a un puerto randómico de un equipo víctima; cuando este recibe un paquete UDP, determina la aplicación que está esperando en el puerto destino; si no existe ninguna aplicación esperando en el puerto mencionado, entonces genera un paquete ICMP de destino inalcanzable al origen (Fuertes et al,2002).

2.2.5 DNS Flood

Es el envío masivo de consultas de nombres de dominio hacia un servidor DNS en un periodo de tiempo corto, con la finalidad de interrumpir la capacidad del servidor para responder adecuadamente las peticiones de usuarios legítimos (Thornewell, P. M., & Golden, L. M. 2012). Este tipo de ataques puede ser considerado también como DDoS dependiendo de su alcance.

2.2.6 Ping of death (Ping de la muerte)

Ping of death (ping de la muerte) es un tipo de ataques a redes en el cual se envía paquetes de tamaño superior a los cuales los equipos están acostumbrados a trabajar, esto con la finalidad de conseguir una denegación de servicio.

El ping de la muerte se utiliza para causar daños en un sistema de red con él envió de paquetes ping extremadamente grandes al sistema destino a través de una red IPV4. La longitud máxima de un paquete o datagrama IP (IPv4) es 65537 bytes esto significa que un paquete de longitud superior dentro de una red puede causar daños como la suspensión del servicio.

La estructura de un datagrama consiste en una cabecera y un campo de datos, cuando se divide un paquete cada uno de los paquetes de menor tamaño contienen una copia de la cabecera que les permite saber cuál es su origen, su destino y además que número de paquete es para poder así realizar el proceso de reconstrucción del mensaje, en la figura que se muestra a continuación se puede de mejor manera la cabecera de un datagrama.

0-3	4-7	8-15	16-18	19-31
Versión	Tamaño Cabecera	Tipo de Servicio	Longitud Total	
Identificador			Flags	Posición de Fragmento
Time To Live	Protocolo		Suma de Control de Cabecera	
Dirección IP de Origen				
Dirección IP de Destino				
Opciones				Relleno

Figura 2. Formato de cabecera IP

Fuente: (Luz, 2015)

La manera de operación de este tipo de ataques es enviar un datagrama de tamaño superior el cual al momento de fragmentarse y ser enviado a través de la red no da ningún problema, hasta el momento en que llega al destino, aquí se realiza la reconstrucción del mensaje y al dar como resultado un paquete superior causa daño en el destino.

2.2.7 Land

Un ataque de tipo LAND consiste en enviar paquetes los cuales contienen como dirección origen y puerto origen la misma información que el destinatario, consiguiendo con esto una cantidad grande de mensajes enviados y recibidos los cuales pueden desencadenar en un mal funcionamiento o hasta el punto de provocar una denegación de servicio.

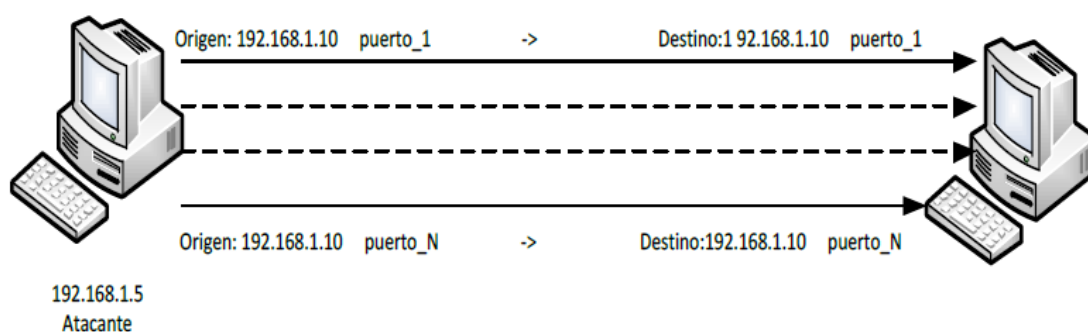


Figura 3. Ataque LAND

Los primeros ataques que se dieron de este tipo fueron alrededor del año 1997 valiéndose de las vulnerabilidades de las diferentes implementaciones del protocolo TCP/IP.

2.2.8 Teardrop

Es un ataque que se realiza en el momento de reconstruir el mensaje que ha sido enviado por fragmentos a través de la red, el atacante crea una secuencia de fragmentos IP con una superposición de campos logrando con esto bloquear algunos sistemas operativos en el momento de ensamblar los fragmentos malformados.

2.2.9 Smurf

Es un tipo de ataque de denegación de servicio que logra su objetivo a través de la explotación de las vulnerabilidades del protocolo IP y del protocolo de control de mensajes de internet (ICMP).

El procedimiento del ataque smurf se muestra a continuación:

1. El malware crea un paquete y lo manda a través de la red con una IP falsa.
2. En el interior del paquete se encuentra un mensaje ICMP preguntando cuales son los nodos de la red para enviar de vuelta una respuesta.
3. Esta respuesta es enviada de nuevo a través de la red logrando con esto que se cree un bucle infinito.

Cuando este ataque se combina con uno de tipo broadcast que envía paquetes maliciosos a todos los nodos de una red puede terminar causando rápidamente una completa denegación de servicio.

2.3 Firewall con IPtables

IPtables es un firewall incluido en el Kermel de Linux desde la versión 2.4 que está incluido en el sistema operativo (Velasco, 2015). El firewall es un conjunto de reglas las cuales se encargan de controlar el tráfico que circula a través de la red. La tabla 1 muestra los tipos de cadenas que se pueden definir en los IPtables.

Tabla 2:

Tipos de cadenas en los IPtables

Cadena	Acción
ACCEPT	Acepta el paquete
DROP	Elimina el paquete sin ningún otro tipo de procesamiento o notificación
LOG	Permite llevar una bitácora del paquete
NAT	Sirve para la traducción de direcciones de red, los paquetes que pasan a través de esta regla terminan con IPs diferentes
DNAT	Enmascara la dirección destino del paquete
SNAT	Enmascara la dirección origen del paquete
MASQUERADE	Similar a SNAT, pero con la variante de que al llegar el paquete se revisa la dirección IP que debe asignarse.
PREROUTING	Acción que se realiza antes de encaminar el paquete
POSTROUTING	Acción que se realiza inmediatamente después de que se encamina el paquete

El filtrado de paquetes se lo puede hacer basado en:

- Direcciones
- Puertos
- Interfaz

Los filtrados se los puede hacer desde distintas partes de la red, a continuación una leve explicación de los comandos que permiten estas acciones:

- **INPUT:** Es dirigido al firewall.
- **OUTPUT:** Es desde el firewall.
- **FORDWARE:** Los paquetes que pasan por el sistema son reenviados a un destino.

La estructura de este tipo de cadenas con las acciones antes mencionadas se muestra a continuación:

```
# iptables -I INPUT, OUTPUT, FORDWARE
```

La tabla 3 muestra las órdenes básicas con las que se pueden construir reglas de firewall para las IPTables

Tabla 3:

Ordenes básica para reglas de firewall

Comando	Función
-A	Agrega una nueva regla a la cadena
-I	Inserta una nueva regla en una cadena especificada
-R	Reemplaza la regla en la cadena especificada
-E	Modifica el nombre de la cadena
-L	Listado de las reglas que se están aplicando
-N	Crea una nueva cadena asociándola a un nombre
-P	Modifica la acción por defecto de la cadena preseleccionada
-D	Elimina una regla específica
-Z	Pone en cero el byte y los contadores de paquetes de una cadena.
-F	Elimina todas las reglas de la cadena
-X	Vacía la configuración

Parámetros

Todas las reglas en IPTables tienen definida su condición por los parámetros que constituye su parte primordial. Algunos de estos parámetros se muestran en la tabla 4.

Tabla 4:
Parámetros de reglas de firewall

Parámetro	Función
-i	Interfaz de entrada (eth0, eth1)
-o	Interfaz de salida (eth0, eth1)
--sport	Puerto origen
--dport	Puerto destino
-p	El protocolo del paquete que se va a verificar
-j	Esto especifica el objetivo de la cadena de las reglas
--line-numbers	Cuando se lista las reglas, agrega el número que ocupa cada regla dentro de la cadena

2.4 Herramientas para la ejecución de ataques

A continuación se describen las herramientas utilizadas en el presente trabajo, cabe mencionar que estas fueron elegidas por la funcionalidad bajo línea de comandos que se pueden ejecutan en la terminal de Linux.

2.4.1 Hping3

Es una herramienta de red capaz de enviar paquetes TCP/IP y UDP personalizados, adicionalmente es muy útil para el testeado de seguridad sobre firewalls y distintas pruebas sobre varios protocolos (die.net, 2015). Es usada especialmente plataformas Linux y Unix, pero también está disponible para sistema operativo Windows.

Es una herramienta modo consola inspirada en el ping de Unix, puede enviar distintos tipos de paquetes y cuenta con un manual de ayuda disponible en internet.

2.4.2 Slowris

Es un script escrito por Robert “RSnake” en lenguaje Perl (Dantas, Y et al.2014), envía una gran cantidad de HTTP Request incompletos, esto quiere decir que al estar mal formados los paquetes el servidor no puede completar el proceso de conexión y los acumula teniendo como consecuencia la saturación de la red y la denegación de servicio para usuarios legítimos.

Los servidores Web que se ven afectados por este tipo de ataques son Apache 1.x y 2.x, en versiones superiores esta debilidad ha sido corregida. Para el desarrollo de la plataforma se usó la versión 2.4.7 de Apache la cual todavía es susceptible a este tipo de ataques.

2.4.3 Ping of Death

Esta herramienta que manda paquetes de longitud mayor 65536 bytes con el objetivo de saturar la red, este tipo de ataques se hizo popular en sistemas operativos en los años 90, poco tiempo después se corrigió el error. A pesar de esto se puede realizar un ataque Ping flood, el cual consiste igual enviar paquetes ICMP a la mayor velocidad posible sin esperar respuesta.

2.5 Tecnologías de virtualización

2.5.1 OracleVM Virtual Box

Es un software de virtualización para arquitecturas x86/amd64, dentro de esta aplicación se pueden instalar diversos sistemas operativos conocidos como sistemas invitados dentro de un sistema anfitrión, cada uno con sus características propias(Oracle, Virtual Box, 2015).

Este software soporta la virtualización de diversos sistemas operativos entre los cuales tenemos GNU/Linux, Microsoft Windows, Solaris/Open Solaris, Macintosh.

Ventajas

- Adaptación entre el sistema operativo anfitrión y huésped
- Acceso a los archivos del sistema anfitrión y huésped
- Permite conexiones de USB y a redes internas y externas

2.5.2 VMware Workstation

Es un software que permite probar, demostrar y desplegar diferentes sistemas operativos basados en x86 de forma simultánea en la misma PC (VMware,2015,a). Esta herramienta es compatible con plataformas Windows, Linux, y MacOS, permite virtualizar sistemas operativos dentro de un mismo hardware de concurrente, consiguiendo con esto el aprovechamiento de los recursos con que cuenta el hardware de la máquina.

Ventajas:

- Plataforma de fácil uso.
- Rendimiento mejorado mediante técnicas avanzadas de virtualización.

Desventajas:

- Existen productos de VMware que se los puede descargar gratuitamente, el inconveniente viene al momento en que queremos tener opciones avanzadas para lograr un trabajo más amplio, en ese momento es necesario comprar un producto de VMware y pagar una licencia.

VMware también ofrece varios productos que tiene que ver con la virtualización pero que se utiliza dependiendo el fin y el medio en el que se va trabajar (VMware,2015,b).

- VMware Player: Es un producto gratuito que permite correr máquinas virtuales creadas con otros productos de VMware, pero no permite crearlas él mismo.
- VMware Server: En un principio era una versión de pago, hace unos meses fue liberada para ser descargada y utilizada de forma gratuita. Esta versión, a diferencia de la anterior, tiene un mejor manejo y administración de recursos.
- VMware Workstation: Esta versión es una aplicación que se instala dentro de un sistema operativo (host) como un programa estándar, de tal forma que las máquinas virtuales corren dentro de esta aplicación, existiendo un aprovechamiento restringido de recursos.

CAPITULO III

PLATAFORMA EXPERIMENTAL DE CIBERSEGURIDAD, SOBRE INFRAESTRUCTURA VIRTUALIZADA

3.1 Entorno virtual de red

Puede ser definido como un conjunto de equipos virtuales (tanto sistemas finales como elementos de red enrutadores y conmutadores) conectados entre sí en una determinada topología desplegada sobre uno o múltiples equipos físicos, el cual emula un sistema equivalente y cuyo entorno deberá ser percibido como si fuera real (Fuertes,2009).

Para implementar el escenario virtual se trabajó sobre plataforma Linux, además se eligió Virtual Box 5.0.6 como software de virtualización. Este software como se mencionó en el capítulo dos soporta arquitecturas x86/amd64, y tiene licencia tipo GLP. A continuación se puede observar en la figura 4 la ventana principal de VirtualBox, en la cual constan las diferentes máquinas que conforman el escenario virtual.



Figura 4. Ventana de inicio de Oracle VM VirtualBox

3.2 Diseño de la topología

La Fig5. muestra la topología de la plataforma experimental base. Como se puede observar existen tres segmentos de red: LAN, WAN y DMZ. Este diseño corresponde a los elementos básicos de una red de producción. A continuación se detalla los diferentes segmentos de la red.

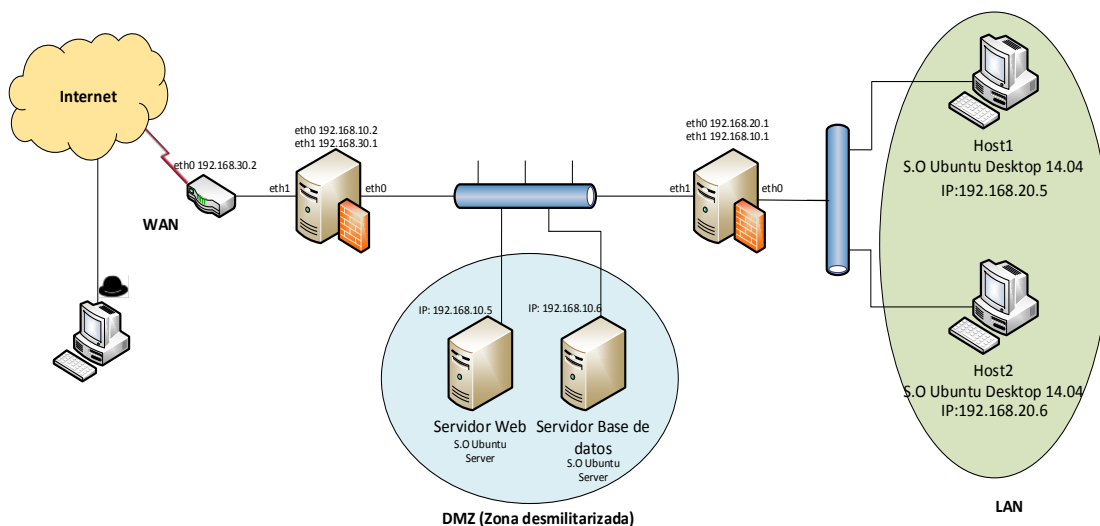


Figura 5. Diagrama de Red

La creación de un segmento LAN tiene como objetivo facilitar la compartición información y recursos de la red, la IP correspondiente a este segmento es 192.168.20.0/24. Adicionalmente una de las máquinas virtuales de la LAN se utilizó como atacante interno.

La creación de la DMZ tiene como objetivo exponer servicios al público como por ejemplo: aplicación web, e-mail, ftp entre otros. Este segmento de red se ubicó entre la LAN y WAN, la presencia adicional de firewall en esta zona es lo que permite un control de las conexiones tanto desde la red interna como externa, esto con el fin de evitar la exposición de algún recurso no autorizado. La IP correspondiente a este segmento es 192.168.10.0/24, estos servidores fueron blanco de ataques de DoS en la presente investigación

Finalmente la creación del segmento de red WAN responde a la necesidad de conectar las diferentes partes que componen el escenario virtual. Este proceso se lo realizó utilizando máquinas virtuales con la funcionalidad de enrutadores.

3.3 Configuración de escenario virtual

3.3.1 Configuración de la LAN mediante máquinas virtuales

Para la creación de las máquinas virtuales que conforman el segmento de red LAN, se utilizó como sistema operativo Ubuntu Desktop 14.04 LTS. Estas máquinas cuentan con servicio SSH para poder realizar operaciones de manejo remoto seguro. Las características de estas máquinas son las siguientes: 2Gb de memoria RAM, 20 GB de disco duro.

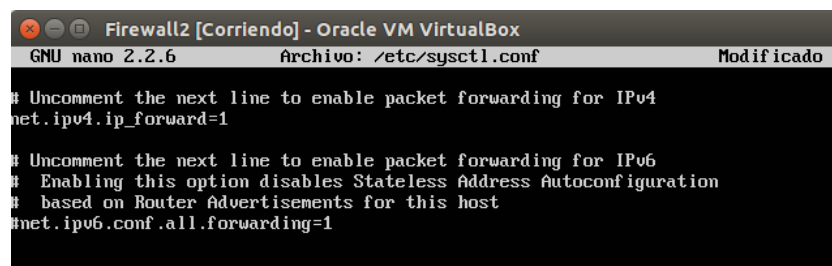
3.3.2 Configuración de DMZ

Para la creación de los servidores que conforman el segmento de red DMZ, se utilizó como sistema operativo Ubuntu Server 14.04. Los servidores que se han creado son el de Aplicaciones Web y el de Base de Datos. En el primer caso se instaló Apache 2.2, y como gestor de base de datos se instaló MySQL Server. Las características de estas máquinas son: 1 Gb de memoria RAM y 8 GB de disco duro.

3.3.3 Configuración de los Routers-Firewalls

Para la conexión de los segmentos de la red se requirió la creación de un servidor con la función de enrutador que permite la retransmisión de paquetes. El mecanismo de mitigación de firewall se lo realizó a través de IPtables en Linux en el servidor antes mencionado.

Para habilitar la función de reenvío de paquetes se procedió a editar el archivo `/etc/sysctl.conf`, y se descomentó la línea `net.ipv4.ip_forward=1`, la Fig 6 muestra el resultado del archivo editado.

A screenshot of a terminal window titled "Firewall2 [Corriendo] - Oracle VM VirtualBox". The terminal shows the nano text editor editing the file `/etc/sysctl.conf`. The visible content includes:

```
GNU nano 2.2.6      archivo: /etc/sysctl.conf      Modificado
# Uncomment the next line to enable packet forwarding for IPv4
net.ipv4.ip_forward=1

# Uncomment the next line to enable packet forwarding for IPv6
# Enabling this option disables Stateless Address Autoconfiguration
# based on Router Advertisements for this host
net.ipv6.conf.all.forwarding=1
```

Figura 6. Archivo `/etc/sysctl.conf`

Este cambio permite que la máquina acepte paquetes ipv4, logrando con esto que trabaje como un enrutador. Para que la misma máquina trabaje adicionalmente como firewall se ha utilizado la herramienta IPtables con reglas que permitirán el filtrado de paquetes.

3.4 Configuración de máquinas virtuales

3.4.1 Direccionamiento IP

Para la configuración de las máquinas virtuales que conforman el escenario se consideró los siguientes parámetros mostrados en la tabla 5.

Tabla 5:

Parámetros de configuración de las máquinas virtuales

Máquina Virtual	Dirección IP	Máscara de red	Puerta de enlace
Host 1	192.168.20.5	255.255.255.0	192.168.20.1
Host 2	192.168.20.6	255.255.255.0	192.168.20.1
Rou-Firewall 2	eth0 192.168.20.1	255.255.255.0	-----
	eth1 192.168.10.1	255.255.255.0	-----
Servidor BDD	192.168.10.6	255.255.255.0	192.168.10.2
Servidor Web	192.168.10.5	255.255.255.0	192.168.10.2
Rou-Firewall 1	eht0 192.168.10.2	255.255.255.0	-----
	eth1 192.168.30.1	255.255.255.0	-----

3.4.2 Servidor de base de datos

El servidor de base de datos se encuentra en la zona desmilitarizada (DMZ), debido a que este servidor estará accesible desde el segmento de red LAN y también podría estar conectado a internet.

3.4.2.1 Instalación de MySQL

Para la instalación del SGBD se debe ejecutar el siguiente comando:

```
#apt-get install mysql-server mysql-client
```

Para el fácil manejo de la base de datos puede instalar phpmyadmin, para hacerlo se debe ejecutar el siguiente comando:

```
#apt-get install phpmyadmin
```

3.4.3 Servidor de aplicaciones Web

El servidor de aplicaciones Web se encuentra en la zona desmilitarizada debido a que puede estar accesible tanto desde el segmento de red LAN como del segmento WAN.

3.4.3.1 Instalación de Apache2

Para instalar el servidor de aplicaciones web se ha elegido a Apache2, para poder instar la presente herramienta se deben realizar los siguientes pasos:

```
#apt-get install apache2
```

Al momento de instalar esta herramienta el Wizard pedirá también la instalación de PHP la cual también podemos aceptar para no tener problemas en el futuro.

3.5 Diseño e implementación de una aplicación tipo escritorio para administrar una plataforma virtual

Para el desarrollo de la aplicación se utilizó la metodología XP, esta se centra en la aplicación de excelentes técnicas de programación, comunicación clara y trabajo en equipo (Extreme Programming Explained, 2005). La figura 7 muestra las diferentes etapas de XP.

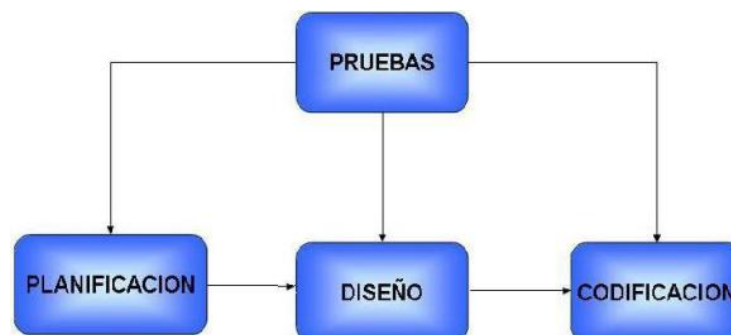


Figura 7. Fases de la metodología XP

3.5.1 Fase 1: Planificación

Es la primera etapa de la metodología XP, en esta se comienza a interactuar con el cliente para descubrir cuáles son los requerimientos del sistema. Uno de los entregables en esta fase son las historias de usuario las cuales contienen las tareas que va a realizar la aplicación.

A continuación se describen los diferentes artefactos producidos por la metodología XP.

- Historias de usuarios

Historia de Usuario	
Número: 1	Usuario: Usuario
Nombre de Historia: Registrar nuevo usuario	
Prioridad de Negocio: Media	Iteración asignada: 1
Riesgo en Desarrollo: Media	
Descripción: Permite el registro de nuevos usuarios para poder usar la aplicación	
Observaciones: Ninguna	

Figura 8. Historia de Usuario: Registrar nuevo usuario

Historia de Usuario	
Número: 2	Usuario: Usuario
Nombre de Historia: Autenticar usuario	
Prioridad de Negocio: Media	Iteración asignada: 1
Riesgo en Desarrollo: Media	
Descripción: Permite ingresar a la aplicación una vez registrado correctamente.	
Observaciones: Ninguna	

Figura 9. Historia de Usuario: Autenticar usuario

Historia de Usuario	
Número: 3	Usuario: Usuario
Nombre de Historia: Gestión de entorno virtual	
Prioridad de Negocio: Alta	Iteración asignada: 2
Riesgo en Desarrollo: Alta	
Descripción: Permite el encender, apagar y ver el diagrama del entorno virtual.	
Observaciones: Ninguna	

Figura 10. Historia de Usuario: Gestión de entorno virtual

Historia de Usuario	
Número: 4	Usuario: Usuario
Nombre de Historia: Gestión de ataques	
Prioridad de Negocio: Alta	Iteración asignada: 2
Riesgo en Desarrollo: Alta	
Descripción: Permite realizar ataques a las máquinas virtuales que se encuentran en el segmento de red DMZ.	
Observaciones: Ninguna	

Figura 11. Historia de Usuario: Gestión de ataques

Historia de Usuario	
Número: 5	Usuario: Usuario
Nombre de Historia: Gestión de mitigación	
Prioridad de Negocio: Alta	Iteración asignada: 3
Riesgo en Desarrollo: Alta	
Descripción: Permite realizar la mitigación mediante la aplicación del firewall tanto el de línea base como el repotenciado.	
Observaciones: Ninguna	

Figura 12. Historia de Usuario: Gestión de mitigación

Historia de Usuario	
Número: 6	Usuario: Usuario
Nombre de Historia: Gráfica de datos	
Prioridad de Negocio: Media	Iteración asignada: 4
Riesgo en Desarrollo: Baja	
Descripción: Permite realizar la gráfica de los datos que se produjeron del ataque realizado	
Observaciones: Ninguna	

Figura 13. Historia de Usuario: Gráfica de datos

Historia de Usuario	
Número: 7	Usuario: Usuario
Nombre de Historia: Gestionar reporte	
Prioridad de Negocio: Media	Iteración asignada: 4
Riesgo en Desarrollo: Baja	
Descripción: Reporte de los ataques realizados y la hora de conexión de los usuarios.	
Observaciones: Ninguna	

Figura 14. Historia de Usuario: Gestionar reporte

- Plan de entrega


Iteraciones	Orden de las historias de usuario	Duración de las iteraciones
1era	1. Registrar nuevo usuario 2. Autenticar usuario	1 Semana
2da	3. Gestión de entorno virtual 4. Gestión de ataques	4 Semanas
3era	5. Gestión de mitigación	2 Semana
4ta	6. Gráfica de datos 7. Gestionar reporte	1 Semana

Figura 15. Plan de entrega del proyecto

- Iteraciones
 - 1era Iteración: Esta iteración tendrá como objetivo darle cumplimiento a las historias de usuario 1 y 2 que adicionalmente tiene prioridad media.
 - 2da Iteración: En esta iteración se le dará cumplimiento a las historias de usuario 3 y 4, las cuales tienen prioridad alta.
 - 3era Iteración: Esta iteración le dará cumplimiento a la historia de usuario 5 la cual tiene prioridad alta.
 - 4ta Iteración: Esta iteración finalmente le dará cumplimiento a las historias de usuario 6 y 7, la cual tiene prioridad media.
- Tareas

Toda iteración se la puede dividir en tareas, cada una de ellas es asignada a un programador y adicionalmente se puede agregar un tiempo para su cumplimiento.

Iteración	Historia de Usuario	Tarea
1era	Registrar nuevo usuario	Diseño de la interfaz de registro de nuevo usuario
		Inserción nuevo usuario
	Autenticar usuario	Diseño de interfaz de autenticación
		Autenticación de usuario
2da	Gestión de entorno virtual	Diseño de la interfaz de manejo de entorno virtual
		Gestión de ataques
	Lanzamiento de ataque Slowris	Lanzamiento de ataque Hping 3
		Lanzamiento de ataque Ping de la muerte

Continua 

3era	Gestión de mitigación	Aplicación de Firewall (Línea base)
		Aplicación de Firewall (Repotenciado)
4ta	Gráfica de datos	Diseño de la interfaz para la presentación de datos
		Representación de datos
	Gestionar reporte	Mostrar reporte

Figura 16. Tareas establecidas en cada iteración

3.5.2 Fase 2: Diseño

- **Diagrama de Clases**

Se modela el presente diagrama para describir las clases, operaciones y relaciones entre las mismas (ver figura 17).

- **Diagrama de Secuencia**

Este diagrama se lo realiza con la finalidad de modelar la interacción entre objetos del sistema. Tal como muestra la Figura 18 se inicia un ataque:

- El usuario decide qué tipo de ataque va a realizar.
- Se realiza la interpretación de la instrucción que realizó el usuario.
- Se ejecuta el procedimiento asociado a la instrucción interpretada.
- Se informa que se acabó de ejecutar el procedimiento.
- Con la finalización del procedimiento también acaba la interpretación de la instrucción.
- Después de haber acabado el ataque se realiza el registro.
- El registro pide una conexión con la base de datos
- Se retorna una conexión, la cual es usada para registrar el ataque
- Retorno de mensaje que indica que se ha registrado el ataque en la base de datos.
- Retorno de mensaje de confirmación de ingreso de registro
- Finalización de ataque



Figura 17. Diagrama de clases de la aplicación

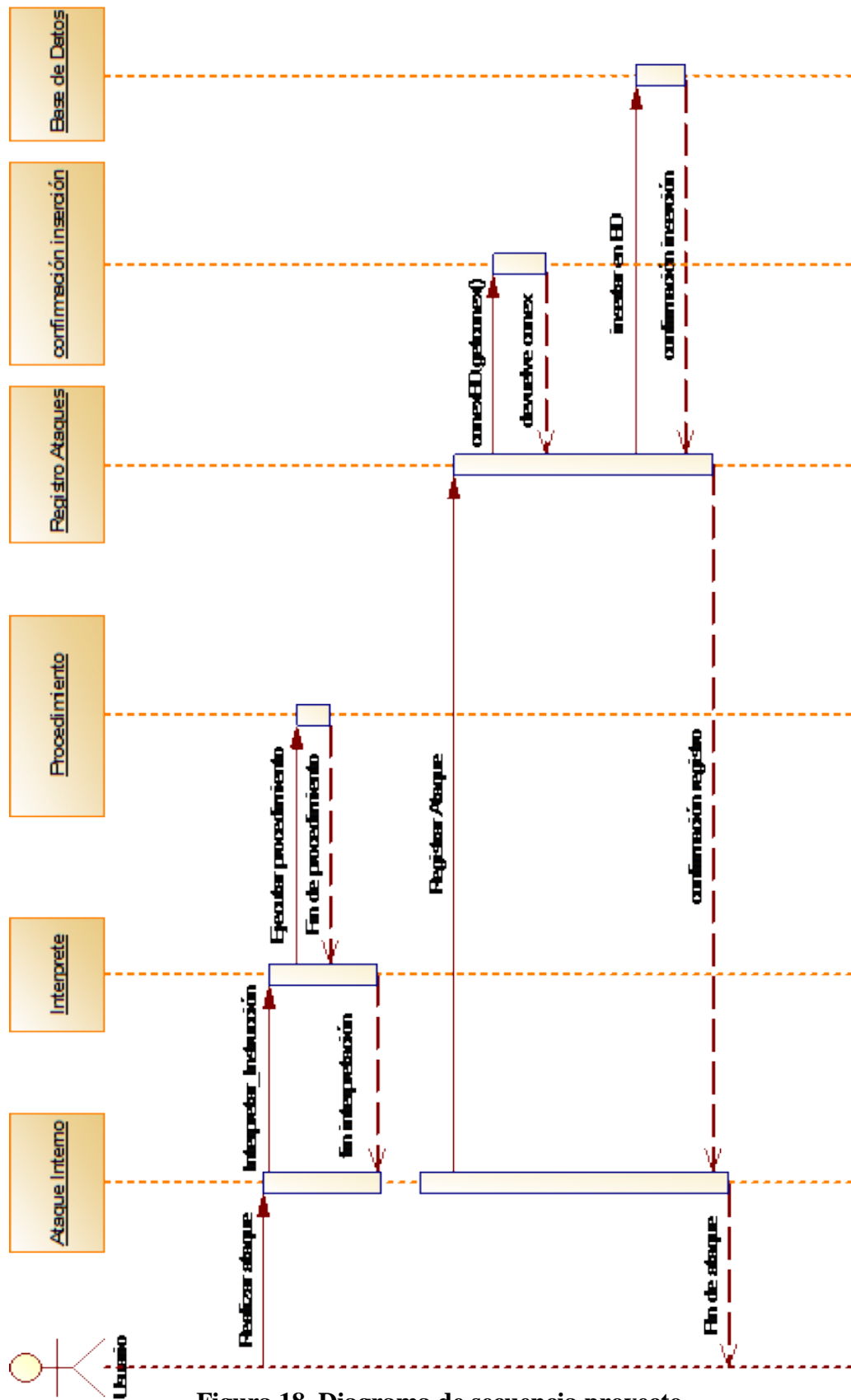


Figura 18. Diagrama de secuencia proyecto

- **Arquitectura del Sistema**

La arquitectura que se eligió para desarrollar la aplicación fue la de n capas, esto se lo hizo con el fin de separar la lógica del negocio de la lógica del diseño. La arquitectura se puede ver en la figura 19.

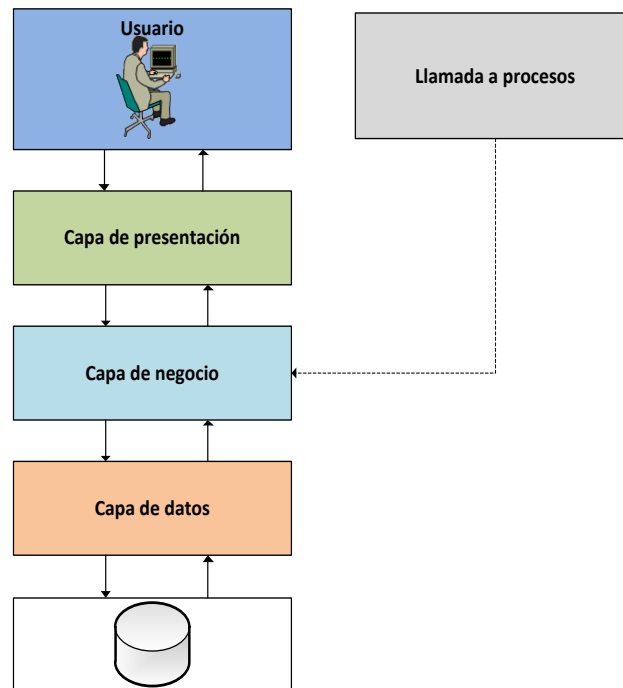


Figura 19. Arquitectura de la Aplicación

A continuación se explicará cada capa de la aplicación:

- **Capa de presentación:** Esta es la encargada de capturar toda la información del usuario, aquí además se realiza un filtrado para comprobar que no existan errores en el formato (Lopez,2015). En esta capa se encuentran los diferentes JFrame que permiten el manejo del escenario virtual.
- **Capa de Negocios:** Aquí se encuentra toda la lógica del negocio, esta capa se comunica con la de presentación para realizar las operaciones que el usuario desea ejecutar. En esta capa se ubican todas las clases que contienen las diferentes instrucciones que deben realizar las máquinas virtuales.

- **Capa de datos:** La capa de datos está formada por los servicios que proporcionan los datos persistentes utilizados por la lógica de negocios. Los datos pueden ser datos de aplicaciones almacenados en un sistema de administración de bases de datos (Oracle, Sun Java Enterprise System, 2005). En la presente aplicación se utilizó como SGBD a MySQL Server por ser de open source.

Esta aplicación consiste en un menú que contiene cuatro opciones: (1) Inicio: En esta opción se podrá encender, apagar, ver diagrama de plataforma virtual y salir de la aplicación; (2) Ataque: Esta opción permitirá al usuario realizar diversos ataques tanto al servidor Web como al de base de datos; (3) Servidores: Aquí se podrá realizar las técnicas de defensa aplicando los respectivos firewalls así como ver estadísticas de los ataques recibidos; (4) Registro: Esta opción permitirá almacenar la información de los usuarios así como los tipos de ataques que se realizaron, y además la hora de conexión.

3.5.3 Fase 3: Codificación

La codificación es un proceso que se lo realiza de forma paralela al diseño y el cual está sujeto a varias observaciones por parte de XP (Lara, 2015). Esta tarea se la realizó con herramientas Open Source, como lenguaje de programación se utilizó Java, el IDE Netbeans 7.4 y para la capa de datos se utilizó MySQL.

3.5.4 Fase 4: Pruebas

Los resultados de las pruebas funcionales de la aplicación se muestran a detalle en la sección 3.9.1.

3.6 Diseño e implementación de algoritmos para el despliegue automático del entorno virtual de red

Para el diseño e implementación de algoritmos de despliegue se utilizó el lenguaje Shell Script, esto permite la comunicación directa con las máquinas virtuales. El proceso de ejecución de los algoritmos se muestra en el siguiente flujograma de la figura 20.

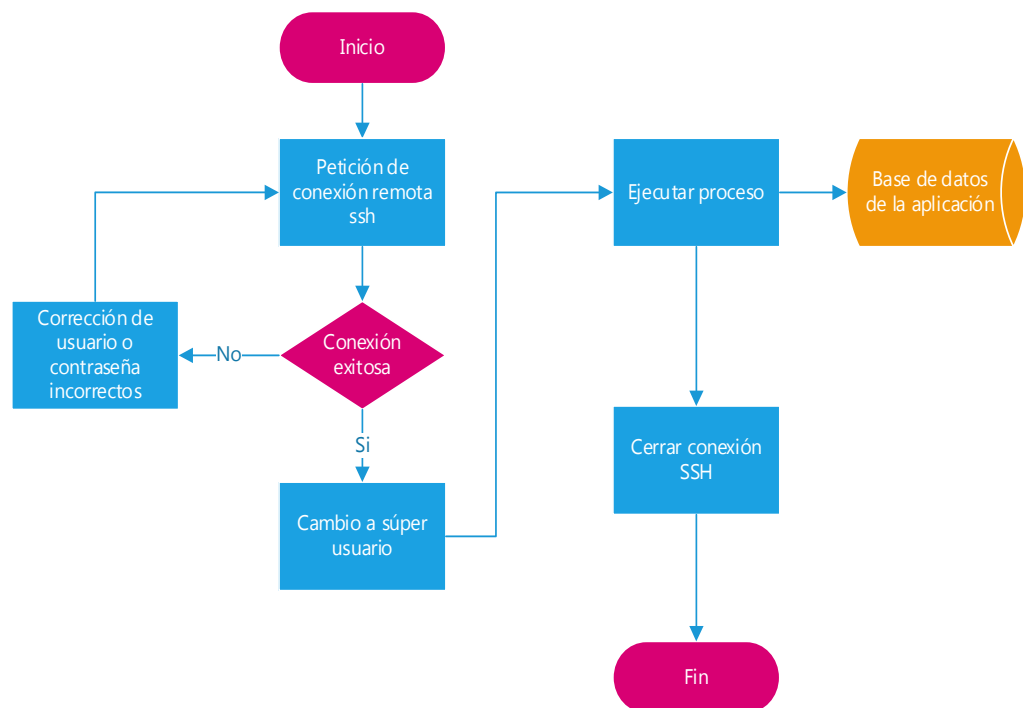


Figura 20. Flujograma general del algoritmo de despliegue

El despliegue del escenario virtual se lo hace mediante la comunicación de la capa de negocios con los diferentes procesos, una breve explicación de las actividades que se realizarán en esta operación se realiza a continuación:

- El proceso realiza una petición de conexión remota a través de ssh, esto lo hace con el usuario y contraseña de la máquina.
- Si el usuario y la contraseña están correctos realizan una petición de cambio de usuario normal por el de súper usuario para poder realizar operaciones complejas.
- Al momento de encontrarse bajo el perfil de súper usuario se ejecuta la instrucción la cual tendrá un tiempo determinado, después de que esta operación se realizó, esta actividad queda registrada en la base de datos de la aplicación.
- Después de registrar la actividad del usuario en la base de datos se cierra la conexión ssh.

3.7 Implementación de ataques

La tabla 6 muestra las herramientas que fueron necesarias para realizar el presente trabajo. La explicación de la elección de cada una de ellas se mencionó en el Capítulo II.

Tabla 6:

Herramientas para realizar ataques DoS

N° ataque	Descripción	Sistema Operativo	Software de ataque
1	Ataque SYN Flood	Ubuntu	HPING3
2	Saturación de ancho de banda	Ubuntu	Slowris
3	Ping of death	Ubuntu	-----

3.8 Mecanismo para contrarrestar ataques a REDES IP

3.8.1 Resolución de anomalías (Línea base)

En este proyecto se utilizó como línea base el trabajo propuesto por los autores Abedin, Muhammad, et al quienes establecieron el siguiente procedimiento que se muestra en la figura 21.

Algorithm. RESOLVE-ANOMALIES: Resolve anomalies in firewall rules file

```

1. old_rules_list ← read rules from config file
2. new_rules_list ← empty list
3. for all r ∈ old_rules_list do
4.     INSERT(r, new_rules_list)
5.     for all r ∈ new_rules_list do
6.         for all s ∈ new_rules_list after r do
7.             if r < s then
8.                 if r.action=s.action then
9.                     Remove r from new_rules_list
10.                    break

```

Figura 21. Algoritmo para resolver anomalías

Para la resolución de las anomalías, los autores antes citados utilizan un algoritmo el mismo que considera a cada regla de firewall un conjunto de atributos, a los cuales se puede exponer a diferentes funciones, las mismas

que están encargadas de ver si existe problemas en las reglas como por ejemplo: reglas disjuntas, reglas iguales, reglas inclusivas o correlacionadas.

Después del paso por las diferentes funciones que tiene este algoritmo se tendrá como resultado una lista de reglas sin anomalías.

Ventajas

- Se dividió al programa en módulos
- Se analiza todas las posibles anomalías entre las reglas de firewall.
- La lista que contiene las nuevas reglas es analizada varias veces para depurar alguna anomalía que haya quedado.

Desventajas

- Al momento de comparar dos reglas no disjuntas, las IPs de las mismas son divididas y sometidas a funciones para detectar cual es el octeto máximo y mínimo para poder con esto crear nuevas reglas que no tengan anomalías entre ellas.
- Si se produce un error en un módulo el programa falla.
- Al ser una programación de tipo modular requiere más memoria y tiempo de ejecución.

3.8.2 Resolución de anomalías (Optimizado)

La figura 22 muestra una propuesta para resolver las anomalías entre reglas de firewall. Se parte del principio que se puede separar a la dirección IP tanto origen como destino en octetos, este procedimiento permitirá resolver anomalías con respecto a este parámetro.

```

1. Resolver_Anomalias(r, lista_nueva)
2. array_aux1[]
3. array_aux2[]=r.attributes.split(',')
4. for all s ∈ lista_nueva
5.     array_aux3[]=s.attributes.split(',')
6.     if(array_aux2[ip_source]=array_aux3[ip_source]) then
7.         if(array_aux1[ip_source]='*')
8.             array_aux1[]=array_aux2[rest_of_attribute]
9.             array_aux1[]=array_aux3[rest_of_attribute]
```

Figura 22. Algoritmo para resolver anomalías (Repotenciado)

- (1) La función resolver recibe dos parámetros, uno de ellos es una regla de la lista antigua y como segundo argumento recibe la lista nueva, esta nueva contiene solo una regla insertada.
- (2) Se crea un arreglo auxiliar array_aux1[] el cual contendrá el resto de parámetros en caso de que el source_ip de la regla r sea igual al de la regla de la lista nueva.
- (3) Los parámetros de la nueva regla son guardados cada uno de ellos en un arreglo auxiliar array_aux2[].
- (4) Se recorre los elementos de la nueva lista.
- (5) Los parámetros de la regla de la nueva lista son guardados en un arreglo auxiliar array_aux3[].
- (6) Los parámetros source_ip y destination_ip de las dos reglas son comparadas, esto se lo hace a partir de los octetos de cada una de ellas.
- (7) En caso de que los tres primeros octetos sean iguales y el último sea diferente, se prestará atención a la presencia del carácter '*' en el octeto.
- (8) y (9) El resto de atributos son guardados en el arreglo auxiliar array_aux1[], para posterior uso en la creación de las nuevas reglas.

Diferencia entre algoritmo Línea base y Repotenciado

La principal diferencia entre el código línea base y el optimizado radica en el tipo de programación utilizado, mientras el primero utiliza modular el segundo utiliza orientado a objetos.

Ventajas

- La comparación de las IPs se las realiza a través de sus octetos partiendo de la izquierda hacia la derecha.
- Se crea un arreglo auxiliar que contiene los octetos que hayan resultado iguales al momento de la comparación de dos reglas, esto se lo realiza con la finalidad de poder crear reglas que no contengan ninguna anomalía.
- Comparación especial al momento de encontrar los siguientes parámetros dentro de las IPs: (ANY, *).

Desventajas

- Mayor cantidad de código

3.9 Pruebas

Las pruebas que se han realizado se han dividido en dos:

1. Pruebas funcionales: La Plataforma Experimental fue sometida a este tipo de pruebas con la finalidad de saber si satisfacen los requisitos del cliente.

2. Pruebas basadas en IPP Metrics: El algoritmo de resolución de anomalías entre políticas del firewall fue sometido a este tipo de pruebas para saber si existió la repotenciación del mismo.

3.9.1 Pruebas funcionales

Según Pressman (Pressman,2005) las pruebas permiten validar y verificar el software, entendiendo como validación del software el proceso externo al equipo de desarrollo, que determina si el software satisface los requisitos, y verificación como el proceso interno que determina si los productos de una fase satisfacen las condiciones de dicha fase. El proceso a seguir para la realización de las pruebas se muestra en la Figura 23.

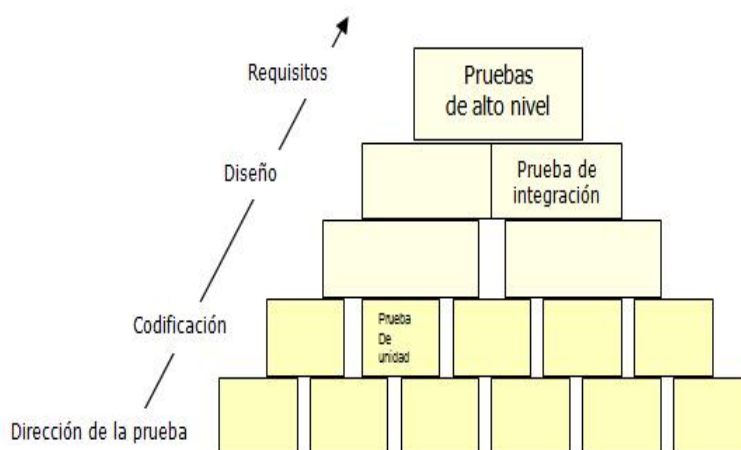



Figura 23. Etapas en la prueba del Software (Pressman ,2005)

3.9.1.1 Pruebas de unidad

Se ha realizado este tipo de prueba con la finalidad de comprobar el correcto funcionamiento de un módulo. En la tabla 7 se indica los errores encontrados con su respectiva solución.

Tabla 7:

Prueba de Unidad

Módulo	Error	Solución
Ataques	No se encontró	
Firewall	No se encontró	
		Continua 

Registro	Si se ingresa al módulo de ataques y no se realiza ninguno hay un error en la base de datos al querer guardar el registro de actividades	Se crea una opción en el caso que no se llega a realizar ningún ataque
-----------------	--	--

3.9.1.2 Pruebas de integración

Se realizó este tipo de pruebas para descubrir errores asociados con la interfaz, por lo tanto en la Tabla 8 se toma como estrategia la integración descendente la cual comienza desde el módulo principal.

Tabla 8:

Prueba de Integración

Módulo	Error	Solución
Registro	No se encontró	
Firewall	No se encontró	
Ataque	Los resultados de los ataques se pueden ver solamente en las estadísticas	Mostrar en tiempo real el comportamiento del servidor ante el ataque

3.9.1.3 Pruebas de validación

Estas pruebas se realizaron con la finalidad de asegurar que el software que se creó se ajusta a los requerimientos y necesidades del cliente. En la tabla 9 se muestran los resultados de las pruebas de validación realizadas sobre la aplicación de escritorio que maneja el escenario virtual.

Tabla 9:

Prueba de Validación

Módulo	Error	Solución
Portabilidad	La aplicación funciona solamente	Implementar el escenario virtual en un servidor, al cual se podrá

	con el escenario virtual en el mismo host.	acceder a través de la IP para poder manejarlo desde la GUI.
Compatibilidad	Con diferente software de virtualización hay problemas en el manejo de las máquinas	Instalar la misma versión del software de virtualización
Facilidad de mantenimiento	Los procedimientos están guardados en la máquina anfitrión	Ubicar los procedimientos en un servidor externo al anfitrión.


3.9.1.4 Pruebas de alto nivel

Después de haber realizado las pruebas de integración de software, se procedió a verificar la correcta interacción entre hardware, software y base de datos. Los errores encontrados en esta prueba se muestran en la Tabla 10.

Tabla 10:

Prueba de alto nivel

	Error	Solución
Base de datos	Existen varios tipos de datos para obtener la fecha del sistema, lo que puede llevar a un error al momento de almacenar los registros de actividades	Utilizar la variable tipo DATE_FORMAT, para luego darle un conveniente formato.
Hardware	Los ordenadores de procesador menor a i5 y baja memoria Ram no podrán ejecutar de manera correcta la plataforma	Ejecutar la plataforma en máquinas i5 en adelante con memoria Ram de 6 Gb recomendable.

Continua 

Software	No se debe activar más de una vez las instrucciones que manejan procedimientos de las máquinas virtuales	Desactivar el botón hasta acabar de ejecutar el proceso.
-----------------	--	--

3.9.2 Pruebas IPP Metrics

Para el desarrollo de las pruebas se trabajó con el estándar definido en IP Performance Metrics (Emile, 2015), estos parámetros permiten medir el rendimiento de una red IP. Para realizar las pruebas se sometió al firewall producido tanto por el algoritmo línea base como el optimizado a ataques DoS.

Las herramientas que permitieron obtener los datos del rendimiento de la red fueron SAR (System Activity Report) y MTR (My Traceroute), las variables evaluadas se muestran en la siguiente tabla:

Tabla 11:

Matriz de definición de variables

Rendimiento de red				
Variable		Definición operacional	Indicador	Herramienta
Latencia	Last	Latencia del último paquete enviado	Milisegundo(ms)	
	Avg	Latencia promedio de todos los paquetes	Milisegundo(ms)	mtr
	Best	El mejor tiempo de ida y vuelta de un paquete	Milisegundo(ms)	
	Wrst	El peor tiempo de ida y vuelta de un paquete	Milisegundo(ms)	
	StDev	Proporciona la desviación estándar de las latencias en cada host	Milisegundo(ms)	

Continua ➡

Consumo Recursos Computacionales			
Variable		Definición Operacional	Herramienta
Consumo del procesador	%user	Porcentaje del CPU utilizado por aplicaciones nivel usuario	Sar
	%nice	Porcentaje del CPU utilizado por aplicaciones con prioridad	
	%system	Porcentaje del CPU utilizado por aplicaciones nivel sistema/kermel.	
	%iowait	Porcentaje del tiempo del CPU en espera a que terminen operaciones I/O.	
	%steal	Porcentaje de tiempo utilizado por el CPU involuntariamente mientras espera que el hipervisor mantenga otro proceso virtual.	
	%idel	Porcentaje de tiempo en el que el CPU se encuentra inactivo.	

CAPITULO IV

EMULACIÓN DE ATAQUES EN ENTORNO VIRTUAL DE RED

4.1 Implementación de ataques

En esta sección se describe la ejecución de los tres ataques de estudio: HPING3, SLOWRIS y PING OF DEATH, todos estos haciendo uso de herramientas de libre distribución.

Los ataques se realizaron a través de la GUI, esta interfaz cuenta con un módulo que permite la ejecución automática de los mismos. La figura 24 muestra los tres tipos de ataques implementados. Una breve descripción se presenta a continuación.



Figura 24. Módulo de ataques

4.1.1 Slowris

Este tipo de ataque se inicia ejecutando un script desarrollado en Perl, el cual genera una denegación de servicio sobre un servidor Apache. La manera de operar es crear “request HTTP” con cabeceras falsas y enviarlos al servidor con la finalidad de mantener abierta las conexiones.

la cuales consisten en el cambio de tiempo de espera de envío de paquetes, el número de paquetes enviados en un tiempo determinado

4.1.2 Hping3

Como se mencionó en el capítulo HPING3 es una herramienta de línea de comandos que permite crear paquetes TCP/IP y ensamblarlos. Además permite el escaneo de puertos y como capacidad especial el hecho de que puede crear un ataque de DoS mediante el envío masivo de paquetes con el fin de saturar los recursos de la máquina.

El ataque se inicia (ver figura 15) cuando la máquina atacante utilizando la herramienta HPING3 ejecuta el siguiente comando

```
#hping3 -p 80 -S --flood IP
```

Donde:

hping3: Herramienta de envío de paquetes TCP/IP.

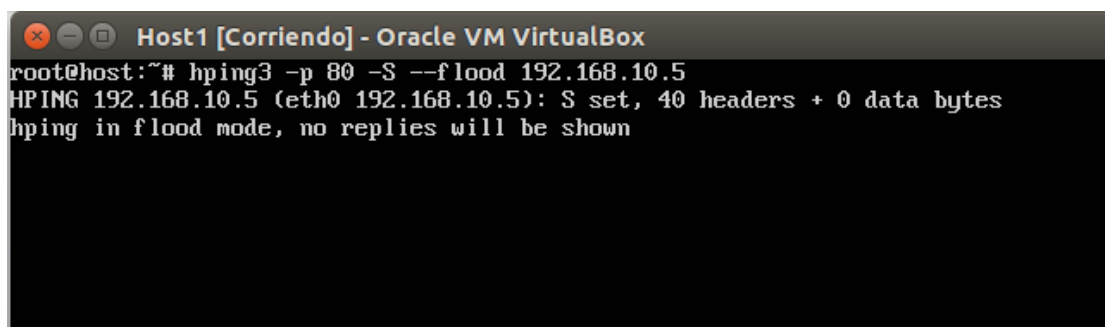
-p 80: Puerto que va a ser atacado en este caso el número 80 (HTTP).

-S: Activa el flag SYN.

-- flood: Indica al hping3 que envíe los paquetes a la máxima velocidad posible.

-IP: Dirección IP a la cual se atacara

Resultados Obtenidos

A screenshot of a terminal window titled "Host1 [Corriendo] - Oracle VM VirtualBox". The terminal shows the command "root@host:~# hping3 -p 80 -S --flood 192.168.10.5" being executed. The output is "HPING 192.168.10.5 (eth0 192.168.10.5): S set, 40 headers + 0 data bytes" followed by "hping in flood mode, no replies will be shown".

```
Host1 [Corriendo] - Oracle VM VirtualBox
root@host:~# hping3 -p 80 -S --flood 192.168.10.5
HPING 192.168.10.5 (eth0 192.168.10.5): S set, 40 headers + 0 data bytes
hping in flood mode, no replies will be shown
```

Figura 26. Ataque HPING3

En la figura 26, se puede observar el ataque al equipo 192.168.10.5 que corresponde al servidor de aplicaciones Web que se encuentra en la zona DMZ. Cabe mencionar que el ataque se realiza por un tiempo estimado de 10 segundos que es lo que demora el algoritmo en ejecutar los comandos en las diferentes máquinas virtuales que conforman el escenario.

4.1.3 Ping of Death

Este tipo de ataque tiene como finalidad enviar paquetes ICMP de longitud superior a 65536 bytes, el resultado de este ataque es la saturación del sistema. Esto se da porque algunas computadoras no pueden manejar ping de longitud mayor a la de un paquete ICMP.

Este ataque es común generarlo desde Windows, pero también hay la opción de hacerlo desde sistema operativo Linux, el comando que se ejecuta para realizar este tipo de ataques se muestra a continuación:

```
#sudo ping IP -l 65536
```

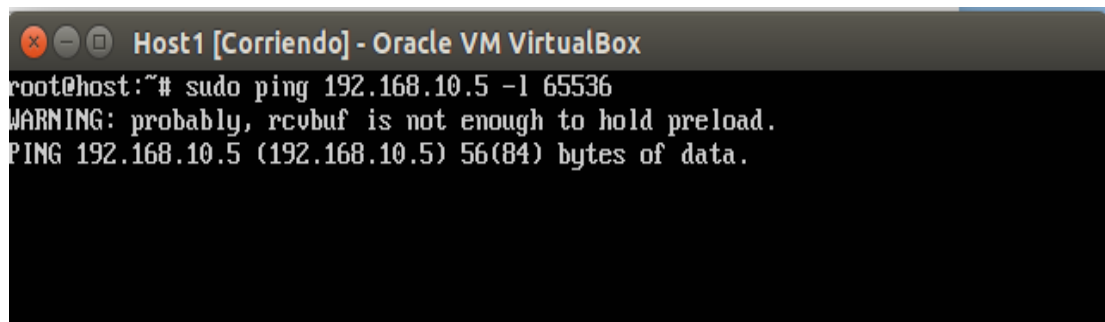
Donde:

sudo: Comando que permite ejecutar instrucciones como súper usuario

ping: Programa que permite verificar si un equipo acepta peticiones

-l: tamaño del buffer

Resultados Obtenidos



```
Host1 [Corriendo] - Oracle VM VirtualBox
root@host:~# sudo ping 192.168.10.5 -l 65536
WARNING: probably, rcvbuf is not enough to hold preload.
PING 192.168.10.5 (192.168.10.5) 56(84) bytes of data.
```

Figura 27. Ataque Ping Of Death

Como se observa en la figura 27, el atacante envía un paquete de longitud mayor superior a 65535 bytes, incluso aparece una advertencia informando sobre que el tamaño del buffer no es lo suficientemente grande para albergar estos paquetes.

4.2 Mitigación de ataques

El mecanismo de mitigación se ha implementado a través de IPtables en Linux, la aplicación cuenta con un módulo en el cual se puede aplicar un firewall sea este el de línea base o repotenciado, además cuenta con la opción de graficar los resultados de esta aplicación, en este caso el uso de memoria, CPU y red (ver figura 28).

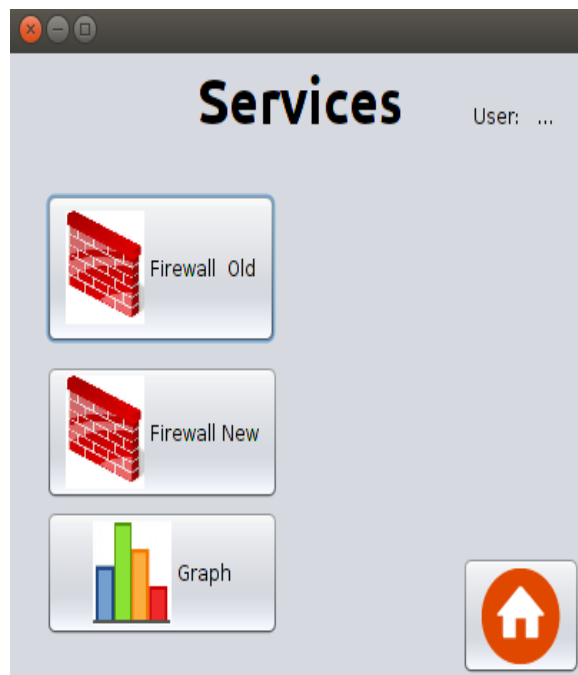


Figura 28. Módulo de Servicios

La implementación de cada uno de los botones se describe a continuación:

- Firewall Old y Firewall New: Cada uno de estos dos botones realiza dos operaciones las cuales son explicadas a continuación:
 - Aplicación del firewall correspondiente gracias a la ejecución de un proceso previamente establecido.

- Ejecución de algoritmo de recolección de datos en el equipo que va a ser víctima del ataque
- Graph: La operación de este botón se divide en dos partes las cuales van a ser explicadas a continuación:
 - Detener las operaciones de recabar datos acerca del ataque que se están ejecutando en la máquina virtual.
 - Procesamiento de datos, esto lo hace gracias a la ayuda de la librería JfreeChart². Para la representación de los valores fue necesario guardar los datos en un arreglo el cual va ser la fuente para esta herramienta gráfica.

² Disponible en <http://sourceforge.net/projects/jfreechart/files/>

CAPITULO V

EVALUACIÓN DE RESULTADOS

5.1 Evaluación del algoritmo de resolución de anomalías de Firewall: línea base vs optimizado

Para la evaluación del algoritmo de resolución de anomalías de Firewall se plantearon 3 escenarios, que consistió en realizar ataques DoS de tipo Hping3, Slowris y Ping of Death contra los servidores que se encuentran en la zona DMZ. Los Firewall implementados a través de IPtables fueron ubicados en los routers comprendidos entre la zona LAN y WAN.

5.1.1 Evaluación de recursos computacionales

5.1.1.1 Uso de memoria

Para la evaluación de esta variable, se toma el parámetro *libre* (cantidad de memoria libre) producido al ejecutar la instrucción *free*, con unidad de almacenamiento igual a KiloByte abreviado como KB.

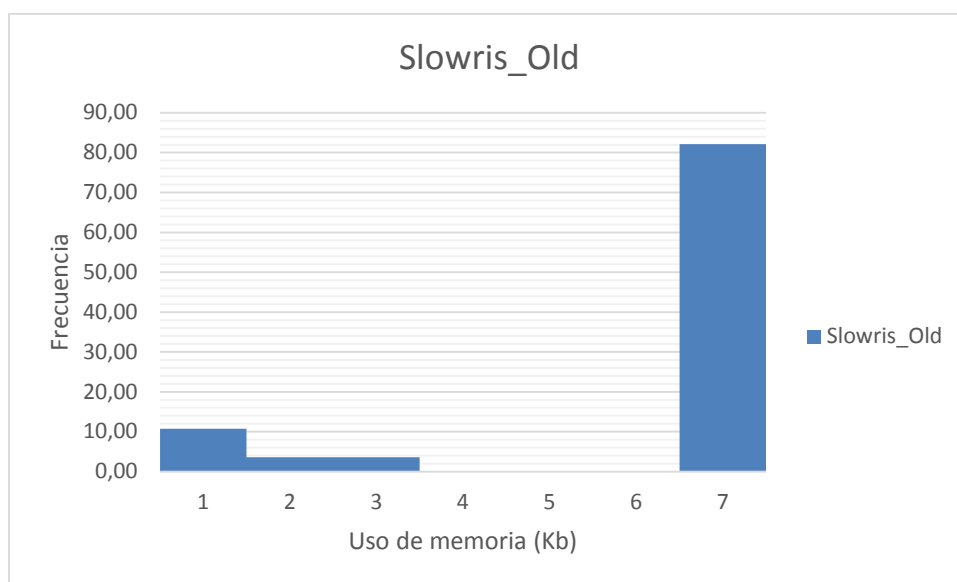


Figura 29. Uso de memoria Slowris_Old

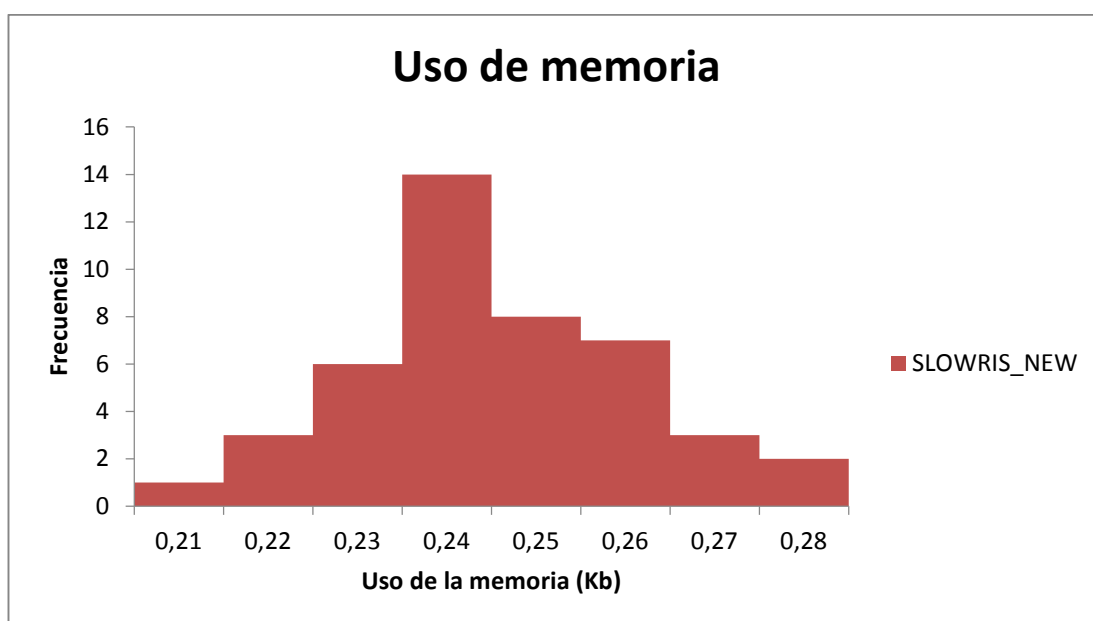


Figura 30. Uso de memoria Slowris_NEW

Los resultados de la línea base y optimizado (Figuras 29 y 30) ilustran el histograma de densidad.

Tabla 12

Datos – Memoria ataque Slowris

Memoria Línea Base			Memoria Optimizado		
Mínimo	17,23	90,67%	Mínimo	16,58	99,59%
Primer cuartil	18,861	99,24%	Primer cuartil	16,6023	99,69%
Mediana	18,913	99,52%	Mediana	16,6076	99,72%
Promedio	18,80	98,92%	Promedio	16,61	99,74%
Tercer cuartil	18,97	99,79%	Tercer cuartil	16,6134	99,76%
Máximo	19,01	100,00%	Máximo	16,65	100,00%

La figura 29, ilustra una diferencia entre el segundo cuartil y el primer cuartil del 0,28% este resultado es menor al 25% de las observaciones y desde el tercer cuartil a la mediana se tiene 0,27% menor al 25%. Esto quiere decir que la distribución de la memoria es asimétrica a la izquierda.

Por otro lado en la figura 30, se tiene que la diferencia entre el segundo cuartil y el primer cuartil es de 0,03%, y la diferencia del tercer y segundo cuartil es 0,04%. Esto quiere decir que las observaciones tienden hacia la izquierda con valores atípicos por abajo del primer cuartil.

Para la evaluación de estos diagramas es necesario comparar las medianas (porcentaje) donde el 99,72% corresponde a la memoria optimizada y el 99,52% a la memoria línea base. Se observa que la memoria optimizada es mayor a la de memoria de línea base ($99,72\% > 99,52\%$), con una diferencia del 0,2%, es decir que el 99,80% de las observaciones indica que el consumo de memoria con el algoritmo optimizado es favorable respecto a la línea base.

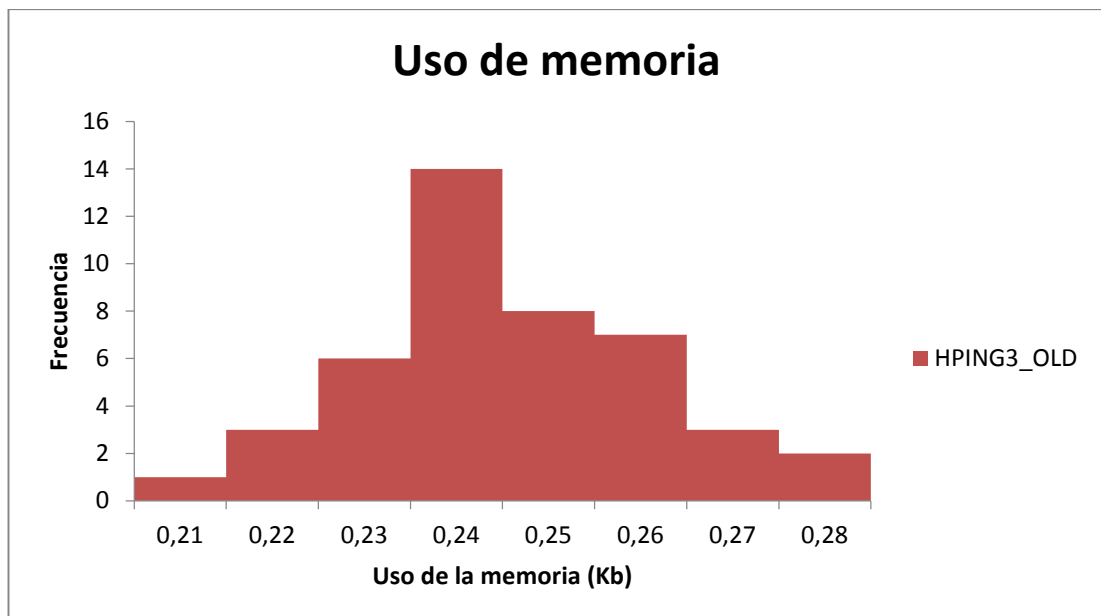


Figura 31. Uso de memoria HPING3_OLD

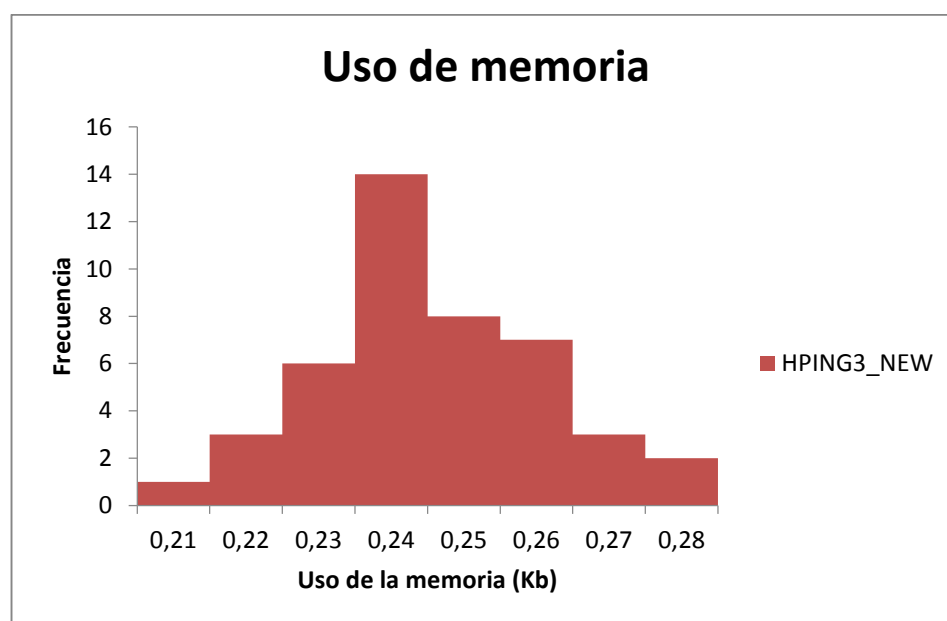


Figura 32. Uso de memoria HPING3_NEW

Tabla 13

Datos – Memoria ataque HPING3

Memoria Línea Base			Memoria Optimizado		
Mínimo	16,60	99,91%	Mínimo	16,34	99,18%
Primer cuartil	16,59	99,87%	Primer cuartil	16,43	99,74%
Mediana	16,59	99,88%	Mediana	16,44	99,80%
Promedio	16,59	99,87%	Promedio	16,43	99,78%
Tercer cuartil	16,60	99,90%	Tercer cuartil	16,44	99,81%
Máximo	16,61	100,00%	Máximo	16,47	100,00%

La figura 31, ilustra una diferencia entre el segundo cuartil y el primer cuartil del 0,01% este resultado es menor al 25% de las observaciones y desde el tercer cuartil a la mediana se tiene 0,02% menor al 25%. Esto quiere decir que la distribución de la memoria es asimétrica a la izquierda.

Por otro lado en la figura 32, se tiene que la diferencia entre el segundo cuartil y el primer cuartil es de 0,06% y la diferencia del tercer cuartil y segundo

cuartil es 0.01%. Esto quiere decir que las observaciones tienden hacia la izquierda con valores atípicos por abajo del primer cuartil.

Para la evaluación de estos diagramas es necesario comparar las medianas (porcentaje) donde el 99,80% corresponde a la memoria optimizada y el 99,88% a la memoria línea base. Se observa que la memoria optimizada es mayor a la de memoria de línea base ((99,88%>99,80%), con una diferencia del 0,08%, es decir que el 99,88% de las observaciones indica que el consumo de memoria con el algoritmo optimizado es favorable respecto a la línea base.

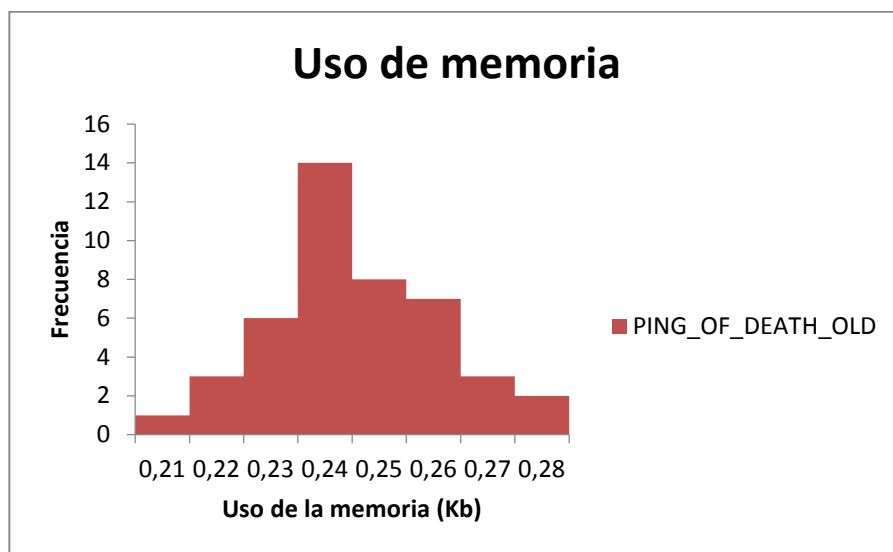


Figura 33. Uso de memoria Ping of Death Old

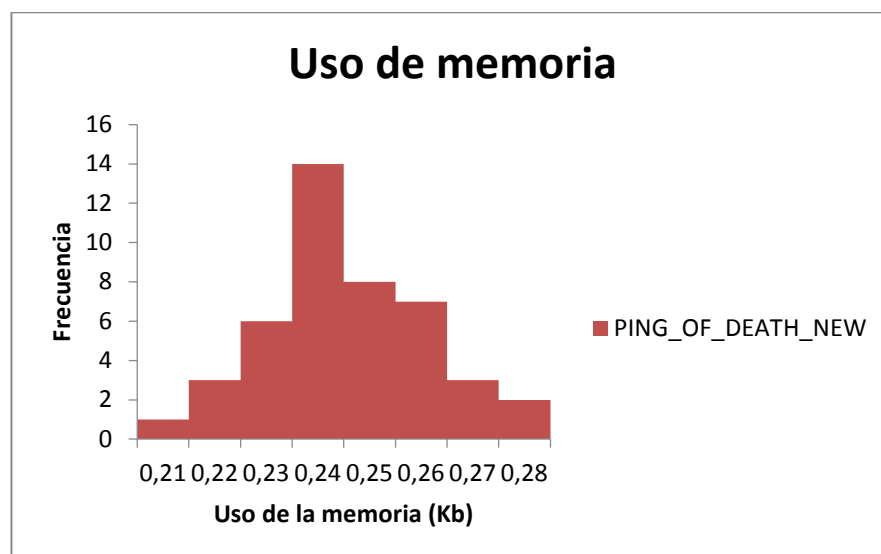


Figura 34. Uso de memoria Ping of Death New

Tabla 14

Datos – Memoria ataque PING OF DEATH

Memoria Línea Base			Memoria Optimizado		
Mínimo	17,62	99,76%	Mínimo	16,79	98,92%
Primer cuartil	17,64	99,85%	Primer cuartil	16,93	99,75%
Mediana	17,64	99,88%	Mediana	16,95	99,90%
Promedio	17,64	99,91%	Promedio	16,94	99,82%
Tercer cuartil	17,66	99,97%	Tercer cuartil	16,96	99,92%
Máximo	17,66	100,00%	Máximo	16,97	100,00%

La figura 33, ilustra una diferencia entre el segundo cuartil y el primer cuartil del 0.03% este resultado es menor al 25% de las observaciones y desde el tercer cuartil a la mediana se tiene 0.09% menor al 25%. Esto quiere decir que la distribución de la memoria es asimétrica a la izquierda.

Por otro lado en la figura 34, se tiene que la diferencia entre el segundo cuartil y el primer cuartil es de 0.15% y la diferencia del tercer cuartil y segundo cuartil es 0.02%. Esto quiere decir que las observaciones tienden hacia la izquierda con valores atípicos por abajo del primer cuartil.

Para la evaluación de estos diagramas es necesario comparar las medianas (porcentaje) donde el 99,90% corresponde a la memoria optimizada y el 99,88% a la memoria línea base. Se observa que la memoria optimizada es mayor a la de memoria de línea base ((99,90%>99,88%), con una diferencia del 0,02%, es decir que el 99,90% de las observaciones indica que el consumo de memoria con el algoritmo optimizado es favorable respecto a la línea base.

5.1.1.2 Uso del procesador

Para la evaluación de esta variable, se tomó el parámetro `%system`, el cual muestra el porcentaje de tiempo de CPU utilizado por aplicaciones/procesos a nivel de sistema/kernel. Este resultado se muestra después de ejecutar el comando `sar`.

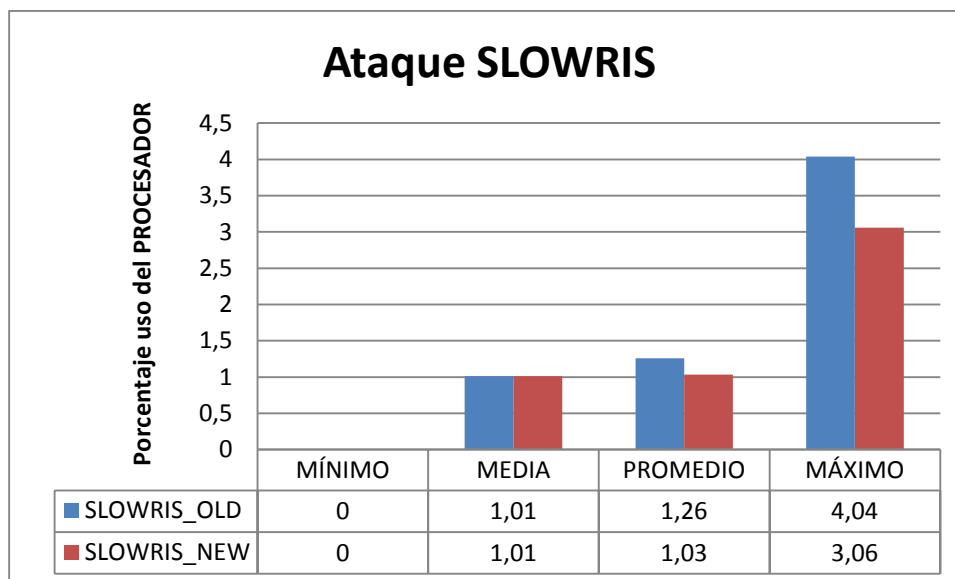


Figura 35. Uso de CPU ataque SLOWRIS

Los resultados de la línea base y la propuesta optimizada (Figura 35), ilustran un gráfico de barras con los respectivos valores, esta estadística proporciona los datos de los valores mínimos, media, promedio y máximo en función del porcentaje de procesador que se utiliza. La tabla 15 muestra un resumen de los datos procesados con Excel.

Tabla 15

Datos uso procesador ataque SLORIS

Procesador Línea Base			Procesador Optimizado		
Mínimo	0	0,00%	Mínimo	0	0,00%
Mediana	1,01	25,00%	Mediana	1,01	33,01%
Promedio	1,26	31,11%	Promedio	1,03	33,79%
Máximo	4,04	100,00%	Máximo	3,06	100,00%

Para la evaluación de este tipo de gráficos es necesario comparar las medianas donde 1.01% corresponde al uso del procesador tanto para el algoritmo línea base como por el optimizado. Se observa también que el valor máximo de procesador utilizado de parte del algoritmo optimizado es menor al de la línea base ($3.06\% < 4.04\%$), con una diferencia de 0.98%, esto indica que el consumo del procesador gracias al algoritmo optimizado es favorable respecto a la línea base.

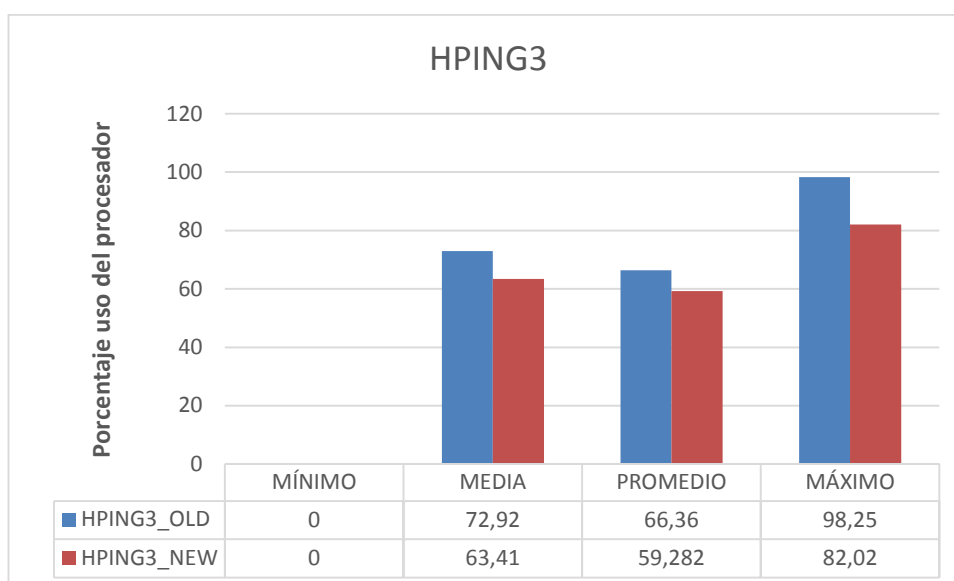


Figura 36. Uso de CPU ataque HPING3

Los resultados de la línea base y la propuesta optimizada (Figura 36), ilustran un gráfico de barras con los respectivos valores, esta estadística proporciona los datos de los valores mínimos, media, promedio y máximo en función del porcentaje de procesador que se utiliza. La tabla 16 muestra un resumen de los datos procesados con Excel.

Tabla 16

Datos uso procesador ataque HPING3

Procesador Línea Base			Procesador Optimizado		
Mínimo	0	0,00%	Mínimo	0	0,00%
Mediana	72,92	74,22%	Mediana	63,41	77,31%
Promedio	66,36	67,54%	Promedio	59,282	72,28%
Máximo	98,25	100%	Máximo	82,02	100%

Para la evaluación de este tipo de gráficos es necesario comparar las medianas donde 72.92% corresponde al uso del procesador utilizando el algoritmo de línea base y el 63.41% a el uso del algoritmo optimizado. Se observa que el porcentaje de procesador usando el algoritmo optimizado es menor al que usa la línea base

(63.41% < 72.92%), esto indica que el consumo de CPU con el algoritmo optimizado es favorable respecto a la línea base.

Se observa también que el valor máximo del porcentaje de procesador utilizando el algoritmo optimizado es menor al de la línea base (82.02% < 98.25%), con una diferencia de 16.23%, esto indica que el consumo del procesador gracias al algoritmo optimizado es favorable respecto a la línea base.

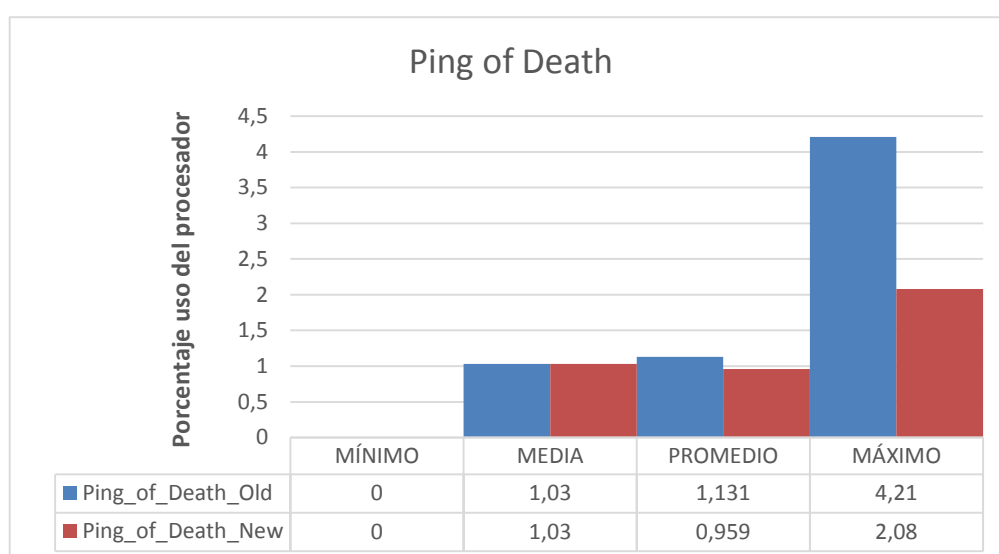


Figura 37. Uso de CPU ataque Ping of Death

Los resultados de la línea base y la propuesta optimizada (Figura 37), ilustran un gráfico de barras con los respectivos valores, esta estadística proporciona los datos de los valores mínimos, media, promedio y máximo en función del porcentaje de procesador que se utiliza. La tabla 17 muestra un resumen de los datos procesados con Excel.

Tabla 17

Datos uso procesador ataque Ping of Death

Procesador Línea Base			Procesador Optimizado		
Mínimo	0	0%	Mínimo	0	0%
Mediana	1,03	24,47%	Mediana	1,03	49,52%
Promedio	1,131	26,86%	Promedio	0,959	46,11%
Máximo	4,21	100%	Máximo	2,08	100%

Para la evaluación de este tipo de gráficos es necesario comparar las medianas donde 1.03% correspondiente al uso del procesador tanto para el algoritmo línea base como para el optimizado. Se observa también que el valor máximo del porcentaje de procesador utilizando el algoritmo optimizado es menor al de la línea base ($2,08\% < 4,21\%$), con una diferencia de 2,03%, esto indica que el consumo del procesador gracias al algoritmo optimizado es favorable respecto a la línea base.

5.1.1.3 Evaluación de la Red

Para la evaluación de esta variable se tomó el parámetro *Last* (latencia del último paquete enviado) producido al ejecutar la instrucción *mtr*, con unidad de milisegundos.

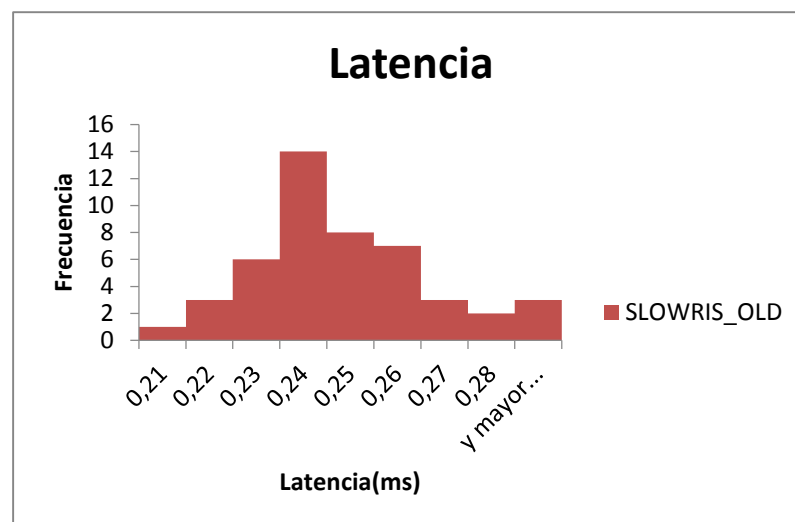


Figura 38. Latencia Slowris_Old

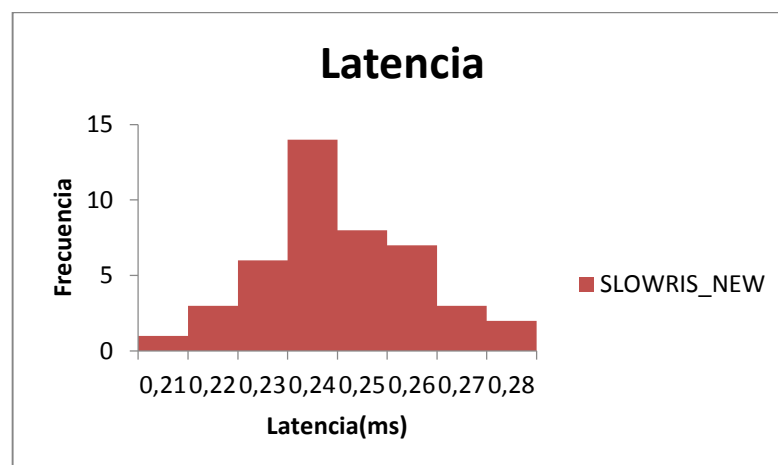


Figura 39. Latencia Slowris_New

Los resultados de la línea base y optimizado (Figura 38 y 39) ilustran el histograma de densidad

Tabla 18

Datos - Latencia Slowris

Latencia Línea Base			Latencia Optimizado		
Mínimo	0	0,00%	Mínimo	0	0,00%
Primer cuartil	0,21	20,39%	Primer cuartil	0,21	40,38%
Mediana	0,22	21,36%	Mediana	0,27	51,92%
Promedio	0,3745	36,36%	Promedio	0,317	60,96%
Tercer cuartil	0,29	28,16%	Tercer cuartil	0,351	67,50%
Máximo	1,03	100,00%	Máximo	0,52	100,00%

La figura 38, ilustra una diferencia entre el segundo cuartil y el primer cuartil del 0,97% este resultado es menor al 25% de las observaciones y desde el tercer cuartil a la mediana se tiene 7,77% menor al 25%. Esto quiere decir que la distribución de la memoria es asimétrica a la izquierda.

Por otro lado en la figura 39, se tiene que la diferencia entre el segundo cuartil y el primer cuartil es de 11,54%, y la diferencia del tercer y segundo cuartil es 27,12%. Esto quiere decir que las observaciones tienden hacia la derecha con valores atípicos por abajo del primer cuartil.

Para la evaluación de estos diagramas es necesarios comparar las medianas (porcentaje) donde el 0,27% corresponde a la latencia aplicando el algoritmo optimizado y el 0,22% aplicando el algoritmo línea base. Se observa que la algoritmo optimizado es mayor al de la línea base ($0,27\% > 0,22\%$), con una diferencia del 0,05%, esto indica que la latencia aplicando el algoritmo optimizado es menor.

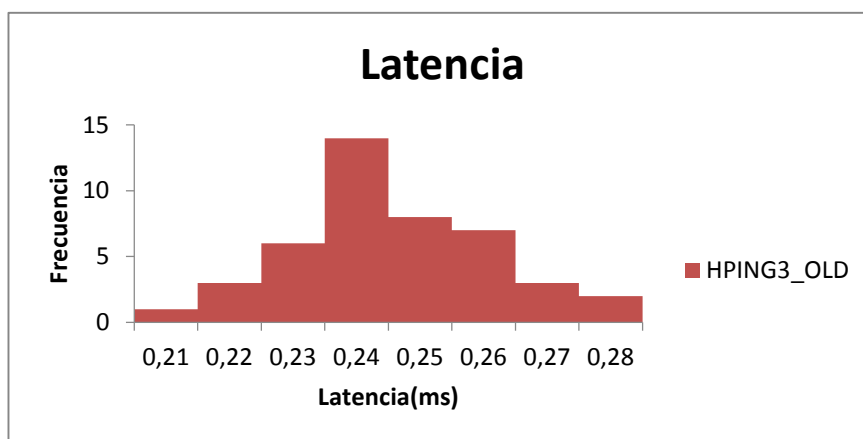


Figura 40. Latencia Hping3_Old

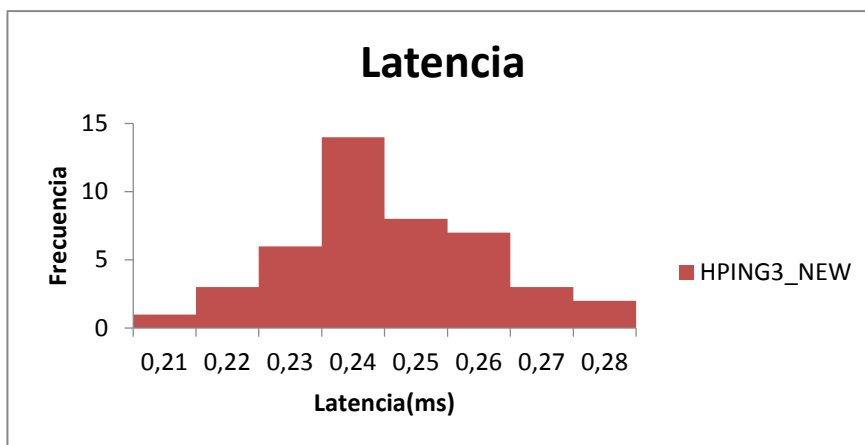


Figura 41. Latencia Hping3_New

Los resultados de la línea base y optimizado (Figura 40 y 41) ilustran el histograma de densidad

Tabla 19

Datos – Latencia ataque Hping3

Latencia Línea Base			Latencia Optimizado		
Mínimo	0	0,00%	Mínimo	0	0,00%
Primer cuartil	0,31	57,41%	Primer cuartil	0,21	43,75%
Mediana	0,36	66,67%	Mediana	0,26	54,17%
Promedio	0,417	77,22%	Promedio	0,379	78,96%
Tercer cuartil	0,39	72,22%	Tercer cuartil	0,391	81,46%
Máximo	0,54	100,00%	Máximo	0,48	100,00%

La figura 40, ilustra una diferencia entre el segundo cuartil y el primer cuartil del 9,21% este resultado es menor al 25% de las observaciones y desde el tercer cuartil a la mediana se tiene 5,55% menor al 25%. Esto quiere decir que la distribución de los datos es asimétrica a la izquierda.

Por otro lado en la figura 41, se tiene que la diferencia entre el segundo cuartil y el primer cuartil es de 10,42%, y la diferencia del tercer y segundo cuartil es 27,29%. Esto quiere decir que las observaciones tienden hacia la derecha con valores atípicos por abajo del primer cuartil.

Para la evaluación de estos diagramas es necesario comparar las medianas (porcentaje) donde el 0,26% corresponde a la latencia aplicando el algoritmo optimizado y el 0,36% aplicando el algoritmo línea base. Se observa que la algoritmo optimizado es mayor al de la línea base ($0,36\% > 0,26\%$), con una diferencia del 0,10%, esto indica que la latencia aplicando el algoritmo optimizado es menor.

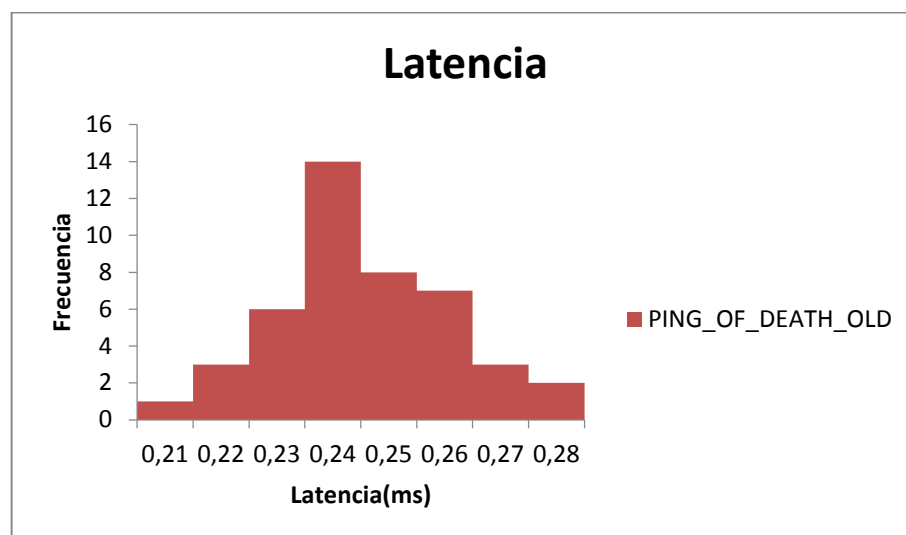


Figura 42. Latencia Ping of Death_Old

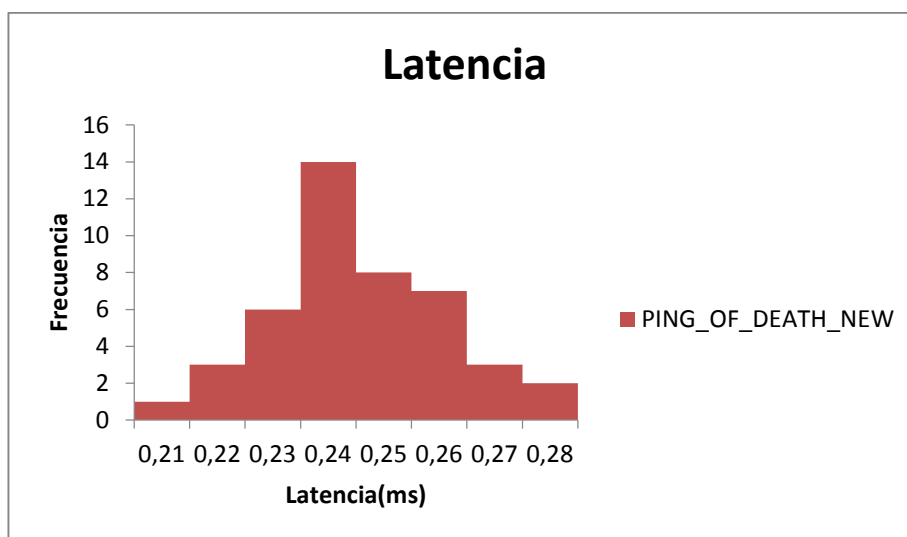


Figura 43. Latencia Ping of Death_New

Los resultados de la línea base y optimizado (Figura 42 y 43) ilustran el histograma de densidad

Tabla 20

Datos – Latencia ataque Ping of Death

Latencia Línea Base			Latencia Optimizado		
Mínimo	0	0,00%	Mínimo	0	0,00%
Primer cuartil	0,25	59,52%	Primer cuartil	0,21	53,85%
Mediana	0,29	69,05%	Mediana	0,252	64,62%
Promedio	0,331	78,81%	Promedio	0,282	72,31%
Tercer cuartil	0,31	73,81%	Tercer cuartil	0,31	79,49%
Máximo	0,42	100,00%	Máximo	0,39	100,00%

La figura 42, ilustra una diferencia entre el segundo cuartil y el primer cuartil del 9,53% este resultado es menor al 25% de las observaciones y desde el tercer cuartil a la mediana se tiene 4,76% menor al 25%. Esto quiere decir que la distribución es asimétrica a la izquierda.

Por otro lado en la figura 43, se tiene que la diferencia entre el segundo cuartil y el primer cuartil es de 10,77%, y la diferencia del tercer y segundo cuartil es 14,87%. Esto quiere decir que las observaciones tienden hacia la izquierda con valores atípicos por abajo del primer cuartil.

Para la evaluación de estos diagramas es necesario comparar las medianas (porcentaje) donde el 64,62% corresponde a la latencia aplicando el algoritmo optimizado y el 69,05% aplicando el algoritmo línea base. Se observa que la algoritmo optimizado es mayor al de la línea base ($69,05\% > 64,02\%$), con una diferencia del 5,03%, esto indica que la latencia aplicando el algoritmo optimizado es menor.

CAPITULO VI

CONCLUSIONES Y RECOMENDACIONES

6.1 Conclusiones

- Se repotenció el algoritmo de resolución de anomalías entre políticas de firewall, logrando con esto reducir el uso de recursos computacionales y de red.
- La plataforma de virtualización VirtualBox permitió el diseño e implementación del entorno virtual de red, además sirvió para probar el algoritmo de resolución de anomalías de políticas de firewall tanto la línea base como el optimizado.
- Se desarrolló e implementó una aplicación de tipo escritorio, la cual permite el manejo del escenario virtual antes mencionado.
- Se desarrolló e implementó un algoritmo para el despliegue automático del entorno de red.
- El trabajar con Linux como sistema operativo anfitrión permitió la fácil comunicación con las máquinas virtuales a través de script.
- El firewall a través de IPtables es una herramienta sumamente potente si las reglas se las escribe correctamente, logrando con esta una mejor seguridad del perímetro de la red.

6.2 Recomendaciones

- Al formar parte de una red, un equipo está expuesto a ataques maliciosos con diversos fines, por lo cual se sugiere que se revise que puertos van a figurar como abiertos frente a los demás.
- Colocar las suficientes reglas para poder controlar tanto el tráfico interno como externo que existe en la red.
- En la actualidad existen diversos ataques de DoS que van cambiando día a día, lo que se recomienda en este caso es tener fijado políticas de seguridad con respecto a la información que se va a manejar, para con esto evitar futuros problemas.

- El uso de escenarios virtuales para simular posibles ataques y encontrar soluciones permite ahorrar dinero y además no pone en exposición equipos físicos.

6.3 Trabajos futuros

- Para reforzar este proyecto se planea el cambio de enfoque de estático a dinámico, logrando con esto una mayor interacción con el usuario.
- Se planea cambiar de tipo de aplicación, pasar de una tipo escritorio a una aplicación Web.

REFERENCIAS BIBLIOGRÁFICAS

- Akamai. (07 de Diciembre de 2015). Akamai. Obtenido de <https://www.stateoftheinternet.com/downloads/pdfs/2014-state-of-the-internet-web-security-global-ddos-attack-report-2014-q2.pdf>
- Alex X.Lui (2011). Firewall Design and Analysis. Vol 4to.USA. Ediciones: World Scientific
- Al-Jarrah, O., & Arafat, A. (2014, April). *Network Intrusion Detection System using attack behavior classification. In Information and Communication Systems (ICICS), 2014 5th International Conference on (pp. 1-6). IEEE.*
- Appelt, D., Nguyen, C. D., & Briand, L. (2015, April). *Behind an Application Firewall, Are We Safe from SQL Injection Attacks?. In Software Testing, Verification and Validation (ICST), 2015 IEEE 8th International Conference on(pp. 1-10). IEEE.*
- Bhat, S., Kanitkar, G., Kannan, P., Nair, S., Dehus, M., & Hogg, S. (2015). Can SDN controller based NSCs help improve user experience of online games?.
- Cuaresma, S. B. (10 de Diciembre de 2015). Manual básico Ubuntu GNU/Linux. Obtenido de <http://www.uls.edu.sv/pdf/ubuntu.pdf>
- Dantas, Y. G., Nigam, V., & Fonseca, I. E. (2014, September). A Selective Defense for Application Layer DDoS Attacks. In Intelligence and Security Informatics Conference (JISIC), 2014 IEEE Joint (pp. 75-82). IEEE.
- Designer, P. (10 de Diciembre de 2015). Power Designer. Obtenido de <http://www.powerdesigner.de/en/>
- Emile, S. (15 de Diciembre de 2015). IETF. Obtenido de <https://tools.ietf.org/html/rfc4148>

- Extreme Programming Explained. (2005). En C. A. Kent Beck. Boston: Addison-Wesley.
- Fuertes, W., Rodas, F., & Toscano, D. (2012). Evaluación de ataques UDP Flood utilizando escenarios virtuales como plataforma experimental. Facultad de Ingeniería, 20(31), 37-53.
- Fuertes, W. (2009). An emulation of VoD services using virtual network environments. Electronic Communications of the EASST, 17.
- Gerentes, T. p. (11 de Diciembre de 2015). Obtenido de <http://www.telecomunicacionesparagerentes.com/gestion-unificada-de-amenazas-utm-proteccion-desde-dentro-y-desde-fuera/>
- Kim, J. M., Kim, A. Y., Yuk, J. S., & Jung, H. K. (2015). *A Study on Wireless Intrusion Prevention System based on Snort. International Journal of Software Engineering and Its Applications*, 9(2), 1-12.
- Linode. (10 de Diciembre de 2015). Linode. Obtenido de <https://www.linode.com/docs/networking/diagnostics/diagnosing-network-issues-with-mtr>
- Lopez, E. (20 de Diciembre de 2015). Academia. Obtenido de https://www.academia.edu/10102692/Arquitectura_de_n_capas
- Luz, S. D. (20 de Diciembre de 2015). Redes Zone. Obtenido de <http://www.redeszone.net/2011/08/04/la-capa-de-red-volumen-iv-ipv4/>
- Oracle. (10 de Diciembre de 2015). Netbeans.Developing Applications with NetBeans IDE Release 7.4. Obtenido de https://docs.oracle.com/cd/E40938_01/doc.74/e40142.pdf
- Oracle. (10 de Diciembre de 2015). Virtual Box. Obtenido de <https://www.virtualbox.org/>

R.Pressman. “Ingeniería del Software, un enfoque práctico”, 6a ed. McGraw-Hioll Interamericana de México. ISBN 9789701054734

Sartakov, V. A (2015). *Ontological representation of networks for IDS in cyber-physical systems*.

Seebach, P. (2008). *Beginning Portable Shell Scripting* . New York.

Sverdlov, E. (10 de Diciembre de 2015). Digital Ocean. Obtenido de <https://www.digitalocean.com/community/tutorials/a-basic-mysql-tutorial>

Thornewell, P. M., & Golden, L. M. (2012). U.S. Patent No. 8,261,351. Washington, DC: U.S. Patent and Trademark Office.

Velasco, R. (10 de Diciembre de 2015). RedesZone.net. Obtenido de <http://www.redeszone.net/gnu-linux/iptables-configuracion-del-firewall-en-linux-con-iptables/>.

VMware a. (15 de Diciembre de 2015). VMware. Obtenido de <http://www.vmware.com/products/workstation/>

VMware b. (15 de Diciembre de 2015). VMware. Obtenido de <http://www.vmware.com/products>