UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

DISEÑO DE ARQUITECTURA DE SEGURIDAD PERIMETRAL DE RED MEDIANTE EL USO DE HERRAMIENTAS DE DETECCIÓN DE INTRUSOS, FIREWALL Y HONEYPOT

HIDALGO SANCHEZ PATRICIA MARILU
INGENIERA DE SISTEMAS

MACHALA
2020

UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

Diseño de arquitectura de seguridad perimetral de red mediante el uso de herramientas de detección de intrusos, firewall y honeypot

HIDALGO SANCHEZ PATRICIA MARILU
INGENIERA DE SISTEMAS

MACHALA
2020

UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

TRABAJO TITULACIÓN
PROPUESTAS TECNOLÓGICAS

Diseño de arquitectura de seguridad perimetral de red mediante el uso de herramientas de detección de intrusos, firewall y honeypot

HIDALGO SANCHEZ PATRICIA MARILU
INGENIERA DE SISTEMAS

VALAREZO PARDO MILTON RAFAEL

MACHALA, 18 DE DICIEMBRE DE 2020

MACHALA
2020

DISEÑO DE ARQUITECTURAS DE SEGURIDAD PERIMETRAL DE RED TRADICIONAL.

INFORME DE ORIGINALIDAD

6%	5%	1%	1%
INDICE DE SIMILITUD	FUENTES DE INTERNET	PUBLICACIONES	TRABAJOS DEL ESTUDIANTE

FUENTES PRIMARIAS

1	red.uao.edu.co Fuente de Internet	1%
2	inba.info Fuente de Internet	<1%
3	ecotec.edu.ec Fuente de Internet	<1%
4	Submitted to Unviersidad de Granada Trabajo del estudiante	<1%
5	ddd.uab.cat Fuente de Internet	<1%
6	Submitted to Universidad Ricardo Palma Trabajo del estudiante	<1%
7	repositorio.usmp.edu.pe Fuente de Internet	<1%
8	documentop.com Fuente de Internet	<1%

CLÁUSULA DE CESIÓN DE DERECHO DE PUBLICACIÓN EN EL REPOSITORIO DIGITAL INSTITUCIONAL

La que suscribe, HIDALGO SANCHEZ PATRICIA MARILU, en calidad de autora del siguiente trabajo escrito titulado Diseño de arquitectura de seguridad perimetral de red mediante el uso de herramientas de detección de intrusos, firewall y honeypot, otorga a la Universidad Técnica de Machala, de forma gratuita y no exclusiva, los derechos de reproducción, distribución y comunicación pública de la obra, que constituye un trabajo de autoría propia, sobre la cual tiene potestad para otorgar los derechos contenidos en esta licencia.

La autora declara que el contenido que se publicará es de carácter académico y se enmarca en las dispociones definidas por la Universidad Técnica de Machala.

Se autoriza a transformar la obra, únicamente cuando sea necesario, y a realizar las adaptaciones pertinentes para permitir su preservación, distribución y publicación en el Repositorio Digital Institucional de la Universidad Técnica de Machala.

La autora como garante de la autoría de la obra y en relación a la misma, declara que la universidad se encuentra libre de todo tipo de responsabilidad sobre el contenido de la obra y que asume la responsabilidad frente a cualquier reclamo o demanda por parte de terceros de manera exclusiva.

Aceptando esta licencia, se cede a la Universidad Técnica de Machala el derecho exclusivo de archivar, reproducir, convertir, comunicar y/o distribuir la obra mundialmente en formato electrónico y digital a través de su Repositorio Digital Institucional, siempre y cuando no se lo haga para obtener beneficio económico.

Machala, 18 de diciembre de 2020

HIDALGO SANCHEZ PATRICIA MARILU
0705418135

DEDICATORIA

Con toda la humildad de mi corazón, dedico este trabajo a Dios, que me ha dado vida, fortaleza y sabiduría para cumplir uno de mis grandes objetivos, culminar mi carrera profesional en ingeniería de sistemas.

A la Sra. Fanny Sánchez, mi amada madre por su amor, consejos, paciencia, oraciones y en reconocimiento a todo el sacrificio puesto para que yo pueda estudiar y cumplir mis metas, de la misma manera a mis hermanos, Raúl, Mayra y Daniel, quienes desde el inicio me brindaron su apoyo incondicional confiando en mi inteligencia y capacidad.

A mi esposo José Miguel, por su amor y paciencia para conmigo, a Rebecca, mi adorada hija quién es mi inspiración en cada momento y con su luz ilumina mi vida.

Hidalgo Sánchez Patricia Marilú

AGRADECIMIENTO

Agradecida con mi padre celestial, por bendecir mi vida y darme las fuerzas necesarias para seguir luchando por mis ideales.

Mi agradecimiento sincero a las personas que me ayudaron al desarrollo de este trabajo investigativo de titulación.

A la facultad de Ingeniería Civil de la Universidad Técnica de Machala, a sus autoridades, por brindarme la oportunidad de profesionalizarme. A los docentes de las asignaturas que nos enseñaron sin egoísmo sus conocimientos actualizados y prácticos para consolidar nuestra formación, al tutor de mi trabajo de grado Ing. Milton Rafael Valarezo Pardo por sus acertadas orientaciones para el desarrollo y culminación de la misma.

Finalmente, a mi familia, quienes estuvieron ahí confiando en mi capacidad para salir adelante, por cuidar de mi hija y compartir mis noches de ansiedad, y porque siempre supieron animarme cuando tuve que enfrentar el desafío de superar obstáculos propios del rigor académico de mi formación profesional, a todos ellos mi agradecimiento imperecedero.

Hidalgo Sánchez Patricia Marilú

RESUMEN

En la actualidad la información digital se considera un recurso importante dentro de toda organización, por lo que esta debe ser tratada y cuidada de manera cautelosa por la constante evolución de diversas amenazas que han ido incrementando, como, por ejemplo: robo o perdida de información, virus informáticos, denegación de servicios y suplantación de identidad. Profesionales especializados en el área de informática recomiendan priorizar la seguridad de la información en todo sistema. Durante la última década, la seguridad perimetral ha sido un elemento fundamental para el diseño de redes porque provee diversas alternativas para identificar, resistir, aislar y bloquear ataques maliciosos además permite distribuir correctamente la red, autorizando o restringiendo el acceso únicamente a usuarios permitidos.

Para la realización del presente trabajo se llevó a cabo una investigación previa basándose en conceptos actualizados para afianzar la fundamentación teórica lo que ayudó a elegir los componentes de seguridad adecuados para salvaguardar el perímetro de la red.

En este trabajo de titulación se plantea el diseño de una arquitectura de seguridad perimetral que incluye herramientas de seguridad con la finalidad de garantizar que la información de una organización no sea vulnerable ante los diversos tipos de ataques, para esto se diseñó una arquitectura que logre proteger todo el perímetro del diseño de red propuesto. El diseño se compone de un sistema de detección de intrusos con reglas locales establecidas para emitir alertas en tiempo real en caso de acciones no autorizadas en la red, así como también de un honeypot que atrae y detecta intrusiones realizadas por un host y crea un log en base a información del atacante, además de una zona desmilitarizada que es un área que aparta al servidor Web Apache y de un firewall que actúa como intermediario en la red para controlar el tránsito dentro de la misma mediante configuraciones y reglas establecidas.

En el diseño de la arquitectura de red se utilizaron los componentes de red que provee el simulador gráfico GNS3 (Graphic Network Simulation) y para las pruebas propuestas fue necesario vincular el software de simulación con la herramienta de virtualización virtualBox, se tomaron en cuenta estas

herramientas por la compatibilidad que existe entre ellas, por la facilidad de uso y por la gratuidad de sus versiones. En la herramienta de virtualización virtualBox se alojaron las máquinas virtuales que contenían el sistema detector de intrusos Snort, el firewall IpFire, el honeypot Kippo, un servidor web Apache, un host de Ubuntu y host Kali Linux que actuaba como atacante. Para comprobar que el diseño propuesto es seguro se ejecutaron pruebas de pentesting o penetración en un entorno virtual para descubrir vulnerabilidades que podrían existir, esto se logró con las herramientas nmap (network maper), metasploit y DirBuster lo que comprobó la funcionalidad de los componentes de seguridad ante las pruebas realizadas.

A la finalización del trabajo se llega a la conclusión que para realizar un buen diseño de red se debe considerar y priorizar aspectos y componentes de seguridad perimetral que permitan mantener la información segura y libre de posibles ataques.

Palabras claves: Arquitectura de red, seguridad perimetral, IpFire, kippo, Snort, zona desmilitarizada.

ABSTRACT

Currently, digital information is considered an important resource within any organization, so it must be treated and cared for in a cautious way due to the constant evolution of various threats that have been increasing, such as, for example: theft or loss of information, computer viruses, denial of services and identity theft. Professionals specialized in the IT area recommend prioritizing information security in every system. During the last decade, perimeter security has been a fundamental element for the design of networks because it provides various alternatives to identify, resist, isolate and block malicious attacks, as well as allowing the correct distribution of the network, authorizing or restricting access only to permitted users.

To carry out this work, a preliminary investigation was carried out based on updated concepts to strengthen the theoretical foundation, which helped to choose the appropriate security components to safeguard the network perimeter. In this degree work, the design of a perimeter security architecture that includes security tools is proposed in order to guarantee that the information of an organization is not vulnerable to the various types of attacks, for this an architecture was designed that manages to protect the entire perimeter of the proposed network design. The design consists of an intrusion detection system with local rules established to issue alerts in real time in the event of unauthorized actions on the network, as well as a honeypot that attracts and detects intrusions made by a host and creates a log based on information from the attacker, as well as a demilitarized zone that is an area that separates the Apache Web server and a firewall that acts as an intermediary in the network to control traffic within it through established rules and configurations. In the design of the network architecture, the network components provided by the GNS3 (Graphic Network Simulation) graphic simulator were used and for the proposed tests it was necessary to link the simulation software with the virtualBox virtualization tool, these tools were taken into account for the compatibility that exists between them, for the ease of use and for the free versions. The virtual machines containing the Snort intrusion detection system, the IpFire firewall, the Kippo honeypot, an Apache web server, an Ubuntu host and a Kali Linux host acting as an attacker were housed in the virtualBox virtualization tool. To verify that the proposed

design is safe, pentesting or penetration tests were carried out in a virtual environment to discover vulnerabilities that could exist, this was achieved with the tools nmap (network maper), metasploit and DirBuster, which verified the functionality of the components of security before the tests carried out. At the end of the work, the conclusion is reached that in order to carry out a good network design, aspects and perimeter security components must be considered and prioritized that allow keeping the information safe and free from possible attacks.
Keywords: Network architecture, perimeter security, IpFire, Kippo, Snort, demilitarized zone.

INTRODUCCIÓN

El diseño de una arquitectura de seguridad de red cada vez resulta más importante y necesario para la protección de la información en un sistema de red moderna. En la actualidad se le debe prestar una especial atención a los componentes tanto físicos como lógicos presentes en una red, debido a que se puede caer en el error de que esta información está protegida sin considerar a los ataques o incidencias de seguridad en redes.

La seguridad perimetral es una estrategia de defensa en una infraestructura de red, la misma que permite proteger la información de una organización cuyos recursos requieren de acceso a internet, haciendo posible la restricción y control de qué datos entran o salen de la red, con la preeminencia al administrador de centralizar los puntos de entrada, sin olvidar el resto de los servidores internos de la red para la protección de intrusos.

En la arquitectura implementada se consideran las herramientas; snort que colabora en la detección de ataques para alertar y enfrentar amenazas que puedan afectar la integridad de la información, ipfire para cumplir las políticas de control y acceso, kippo que actúa como distractor, atrayendo y analizando ataques externos permitiendo adquirir información sobre cyberdelicuentes que puedan comprometer la red, además de la implementación de una zona desmilitarizada que aísla todos los recursos con acceso a internet de la red local.

La presente propuesta tecnológica es elaborada con la finalidad de proponer una topología de red considerando aspectos fundamentales que aseguren la seguridad perimetral de una red, para lo cual se establecen políticas de seguridad en la configuración de red.

En el capítulo 1 Diagnóstico de Necesidades y Requerimientos se define la necesidad de diseñar una arquitectura de seguridad perimetral de red mediante las herramientas snort y kippo, y una zona desmilitarizada, describiendo el enfoque del tema y sus antecedentes, continuando con los requerimientos y justificación en lo que se debe indicar la importancia, la utilidad y la factibilidad del diseño, que corresponda a las necesidades planteadas.

En el capítulo 2 Desarrollo del Prototipo describe la fundamentación teórica, objetivos, diseño y aplicación del diseño de red.

En el capítulo 3 Evaluación del Prototipo, se prueba la funcionalidad de la topología propuesta con las herramientas definidas.

Y finalmente termina con las conclusiones y recomendaciones.

CONTENIDO

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

1. CAPÍTULO I. DIAGNÓSTICO DE NECESIDADES Y REQUERIMIENTOS

1.1. Ámbito de aplicación: descripción del contexto y hechos de interés

Las redes informáticas sobre todo aquellas que con acceso a internet son propensas a ser atacadas por hackers mediante el uso técnicas y mecanismos de ataques que les permiten acceder a la red de datos a través de portales con el propósito de robar o manipular información o realizar ataques al sistema para que este colapse.

En 1988, un programa desarrollado por Robert Morris, dio lugar al primer incidente de seguridad en una red informática provocando la pérdida de millones de dólares, a partir de esta fecha todo lo relacionado a seguridad informática es un factor considerado en sistemas operativos y en el diseño de redes [1]

En consecuencia, de la evolución de la tecnología y el incremento de procedimientos destinados a quebrantar la seguridad de una red es necesario adaptar rigurosos niveles de control en una red de datos. Por lo que se considera importante que en el diseño de una red se establezca un sistema de seguridad perimetral mediante el uso de procedimientos y herramientas que permitan crear una frontera de protección a la infraestructura física y lógica evitando daños a la información [2].

En el trabajo de [3] manifiesta que mediante la seguridad perimetral es posible reducir el riesgo de incidentes a nivel de datos con la implementación de mecanismos de restricción del acceso a personal no autorizado, además [4], afirma que implementar sistemas de seguridad perimetral en una organización es fundamental para minimizar ataques, pérdida o daño de información.

La presente propuesta tecnológica de seguridad perimetral, pretende minimizar el riesgo de sufrir ataques informáticos en el modelo de red propuesta, mediante la identificación de posibles accesos de usuarios no autorizados con la intención de realizar actividades maliciosas.

1.2. Establecimiento de requerimientos

Hoy en día, la información digital de una organización es considerada un elemento valioso, por lo que es primordial tener en cuenta la seguridad en el diseño de una red [5].

Una red debe contar con un sistema de seguridad robusto que permita crear una barrera de protección entre la red externa y la red interna [6], dado que una red con acceso a internet está expuesta a sufrir ataques informáticos que afectan la integridad

de los datos que en su posterior se convierte en información, misma que debe protegerse con rigurosidad.

La arquitectura de red será diseñada en el software GNS3, una herramienta de virtualización y simulación que facilita el diseño gráfico de redes y separa al usuario de la configuración del escenario [7]. Así mismo se requieren de virtualizar 6 máquinas que conllevan las herramientas para realizar las respectivas pruebas.

Para el diseño de seguridad perimetral de la red propuesta se requiere del uso de un firewall, de sistemas de detección de intrusos, de un honeypot y de una zona desmilitarizada.

1.3. Justificación del requerimiento a satisfacer

En los últimos años la confiabilidad y seguridad informática se ha visto en la necesidad de ser fortalecida para asegurar el bien más valioso como es la información, para esto se consideraba que los ordenadores y su sistema operativo deberían estar en constante mantenimiento para que no dejasen de funcionar y también evitar todo tipo de ataque informático y así no perder información, pero ahora con la aparición del internet los equipos tuvieron acceso a la conectividad a nivel global lo mismo que los expuso a un sinnúmero de vulnerabilidades que afectan no solamente a los recursos informáticos sino directamente a la información.

Esta situación dio lugar a diseñar una arquitectura de seguridad perimetral de red en este caso con herramientas de detección de intrusos que monitorean el tráfico de una red para identificar y alertar posibles amenazas que atenten contra la seguridad de la información, y zona desmilitarizada , que es una red local encargada de apartar recursos que requieren de acceso a internet para evitar conexiones directas entre la red interna y externa, todo esto con el fin de controlar la seguridad a nivel periférico tanto de entrada como de salida, con esto también se logra mitigar los riesgos de que exista interconexiones a redes externas.

Con una arquitectura como la que se proyecta en este trabajo, se intenta minimizar ataques informáticos a la red, la misma que estará en condiciones de lograr una conectividad de manera más segura, con la capacidad de controlar las amenazas y monitoreando el tráfico de red.

La presente propuesta tecnológica tiene como objetivo principal el diseño de arquitectura de seguridad perimetral de red mediante el uso de componentes de seguridad informática, para impedir que se generen ataques informáticos que pongan en riesgo la infraestructura de la red y la información.

2. CAPÍTULO II. DESARROLLO DEL PROTOTIPO

2.1. Definición del prototipo tecnológico

La presente propuesta tecnología consiste en el diseño de una arquitectura de red basada en un modelo de red tradicional con el uso de snort una herramienta de detección de intrusos, ipfire un firewall que controla el acceso en la red , kippo un honeypot que actúa como distractor para posibles atacantes, y una zona desmilitarizada que aparta los recursos con acceso a una red externa, lo que va a permitir vigilar los datos que transitan en la red para evitar la fuga o manipulación de información de uso exclusivo en una organización.

2.2. Fundamentación teórica del prototipo

2.2.1. Seguridad de la información

El término seguridad de la información asemeja a una disciplina responsable de la protección de la información, apoyándose en la seguridad informática y da a conocer mediante análisis de problemas posibles riesgos y amenazas, además exige altos niveles de aseguramiento de procesos y confianza, y es de vital importancia en toda empresa, organización e instituciones ya sean públicas o privadas que haga uso de tecnología para almacenar o trata información [8].

En una organización independientemente de la índole o la actividad a la que se dedique, la información debe estar disponible para todo tipo de usuarios (internos y externos) y puede ser accedida desde cualquier dispositivo, esta situación amerita a considerar la seguridad un elemento clave para garantizar el éxito de la organización en cuanto a información. [9] considera que la seguridad de la información es responsabilidad de todas las personas que integran una organización, mismas que deben procurar protegerla de diversas amenazas porque esta brinda soporte a la actividad de una determinada organización.

2.2.2. Pilares fundamentales de la seguridad

Si bien la seguridad de la información guarda relación con el conjunto de medidas preventivas para proteger la información y regirse a los tres elementos considerados pilares fundamentales de la seguridad (integridad, disponibilidad y confidencialidad) [10].

2.2.2.1. Confidencialidad

[11] hace referencia a la exclusividad de la información, los datos pueden ser accedidos y modificados únicamente por usuarios autorizados.

A continuación, se listan tres formas de garantizar el cumplimiento de este principio:

- Gestión de usuarios

- Gestión de permisos y privilegios
- Cifrado o encriptación de información

2.2.2.2. Integridad

Garantiza que los datos llegan a su destino de forma íntegra, correcta y libre de modificaciones que alteren la veracidad de la información, es decir los datos llegan tal cual fueron creados.

De acuerdo con [12], se consideran las siguientes actividades que permiten garantizar la integridad de la información:

- Controlar el tráfico de red para detectar y prevenir ataques.
- Hacer uso de sistemas de control de cambios.
- Realizar respaldos (backups) constantemente.
- Realizar auditorías en los sistemas.

2.2.2.3. Disponibilidad

El tercer pilar de la seguridad de la información, garantiza que la información puede ser accedida en cualquier lugar y momento por personas autorizadas [13].

2.2.3. Vulnerabilidades

Como expresa [14], una vulnerabilidad es la debilidad que presenta un activo sobre una amenaza.

Basado en el concepto de varios autores sobre las vulnerabilidades, se considera un fallo que puede ser aprovechado por un usuario malintencionado con la finalidad de perjudicar la seguridad del sistema, robar información confidencial o provocar daños directos a una organización.

2.2.4. Amenazas

En informática, este término guarda relación con toda acción realizada para infringir o atentar con la seguridad de la información y son el resultado de una vulnerabilidad correctamente aprovechada. Desde el punto de vista de [15] es un problema importante que afecta a la información y para minimizar su impacto es necesario conocer su origen y cómo proteger la información de la organización en caso de que ocurran .

Sin la implementación de medidas que garanticen la seguridad, una red está expuesta a una cantidad elevada de riesgos, entre ellos [16] y [17] cita los siguientes:

- Pérdida o exposición de información confidencial de una organización.
- Alteración de información.
- Virus informáticos o malware.

- Fraudes por permitir acceso malintencionado
- Suplantación de identidad
- Denegación de servicios
- Sabotaje
- Espionaje
- Sql Injection
- Desastres naturales.

Gran parte de las amenazas provienen de la red externa, sin embargo, a criterio de

2.2.5. Seguridad perimetral de red

Es indispensable que la seguridad esté presente en el diseño e implementación de red con o sin acceso a internet. El término seguridad perimetral hace referencia a la protección necesaria de todo el perímetro de una red con la finalidad de controlar el tráfico de datos dentro y fuera de la red mitigando el riesgo de posibles ataques y protegiendo el hardware, software y los datos que actualmente son considerados activos institucionales.

De acuerdo con [18], no existe una única solución que proteja todo el perímetro de la red de las múltiples amenazas existentes, por lo que se han considerado los siguientes componentes de seguridad:

- Firewall o cortafuego
- Sistemas de detección de intrusos
- Honeypots
- Zona desmilitarizada (DMZ)

2.2.6. Firewall

La palabra firewall traducida al español es cortafuego, en informática es una herramienta de seguridad cuya función es proteger a una red informática de ataques malintencionados y desde hace 25 años han sido considerados en una red como primera línea de defensa ante ataques [19].

Tal como menciona [20], un firewall es un conjunto de sistemas de hardware o software que implementan políticas de control de acceso en el punto de conexión de la red local y la red externa para monitorear la seguridad perimetral y alertar al responsable de la red de posibles ataques a la misma.

Según [21]: un firewall puede ser útil como herramienta para analizar la conducta de sistemas y de una red porque controla la información que transita en la misma, permitiendo el bloqueo de paquetes para aceptar o denegar acceso para el tránsito por la red de dicha información.

A juicio de [22], se atribuye que un firewall es una base fundamental para la prevención y protección de una red, salvaguardando y precautelando la seguridad de los datos en una organización.

Además [23], considera las siguientes ventajas sobre la implementación y/o uso de firewalls:

- Posibilita el refuerzo de políticas de seguridad, permitiendo añadir políticas más robustas.
- Permite monitorear la red, generando alertas cuando se intente un ataque.
- Recopila información sobre eventos realizados tanto en la red interna como en la externa.
- Limita el daño por posibles ataques, puesto que restringe el acceso no autorizado entre la red.

Cabe recalcar que un firewall no garantiza que la red permanezca libre de virus debido a que existen diversos mecanismos que esconden virus, para mitigar este riesgo es necesaria la instalación de software antivirus.

2.2.6.1. IpFire

Es una distribución de Linux empleada para un sistema de seguridad perimetral, actuando como sistema operativo individual y desempeñando las funciones de firewall. Al considerarse un sistema ligero, este puede ser incorporado en equipos con pocas capacidades [24].

Su principal función en la arquitectura propuesta es desempeñar tareas de seguridad, cuando se complementa con la implementación de una DMZ, se divide en varias zonas de la red con diversas zonas de seguridad.

2.2.7. Sistema de detección de intrusos

Por la constante evolución de la tecnología las organizaciones de todo tipo se ven obligadas a implementar sistemas para gestionar su información y redes informáticas que permitan la transmisión de la misma. Sin embargo, la adaptación de nuevas tecnologías atrae usuarios malintencionados que pretenden violentar los sistemas de seguridad y acceder a los sistemas para sustraer, desintegrar y acceder a revisar toda la información posible. Para ello, hoy en día se cuenta con sistemas que detectan el acceso, las actividades de usuarios no autorizados, estos sistemas son denominados sistemas de detección de intrusos y [25] considera que son una de las herramientas más utilizadas en la actualidad por su robustez, facilitando al administrador o responsable mayor visibilidad de lo que circula o sucede en una determinada red.

Un IDS analiza detenidamente el contenido de los paquetes para definir que contiene dentro y que eventos está destinado a realizar, [26] plantea la siguiente lista como las principales funcionalidades de los IDS:

- Detectar ataques y violaciones de seguridad
- Implementa calidad de control en tareas relacionadas con la administración y seguridad.
- Minimiza el riesgo de abusos en la red.

2.2.7.1. Snort

Sistema modular de libre distribución y código abierto bajo la licencia GPL compatible con sistemas operativos como Windows y Linux, permite realizar análisis de protocolos, coincidencias y del tráfico en tiempo real y proporciona un registro de todos los paquetes que transitan por la red, además realiza acciones mediante un conjunto de reglas propias de snort y reglas locales agregadas por el administrador de la red.

A criterio de [27] es un sistema fácil de usar basado en reglas propias del sistema y reglas locales que pueden ser configuradas acorde a las necesidades de la red, esto para gestionar la seguridad de la información que transita en una red.

Snort se considera que es un sistema fiable que aporta actualizaciones constantes mediante boletines de seguridad, además permite detectar y enviar alertas para diversos protocolos. [28]

Este sistema requiere ser configurado mediante reglas empleadas para el reconocimiento de ataques y firmas de difusión. Además, Snort posee su propia sintaxis que abarca detalles a cumplirse para asociar a un paquete, a continuación, se presenta el formato para establecer una regla y se describe cada parámetro en base a [29]:

Figura 1 Reglas de Snort

```
<acción> <protocolo> <IP-origen> <Puerto-origen> <dirección> <IP-destino>
<Puerto-destino> [( <opción-1>; …; <opción-n>; )]
```

Fuente: Elaboración propia

- Acción: Se elige la regla entre las opciones (alert, log, pass, actívate, dynamic, drop, reject, sdrop)
- Protocolo: Se analizan comportamientos sospechosos entre los 4 protocolos TCP, UDP, ICMP e IP
- Direcciones Ip: Referencia la dirección ip de origen y la de destino.

- Números de puerto: Se especifican incluyendo definiciones estáticas de puertos, rangos y por negación.
- Operador de dirección: Indica la orientación del tráfico al que se le aplica la regla.
- Opciones: Se engloban en cuatro categorías (general, payload, non-payload y post-detection)

2.2.8. Honeypot

Es un recurso que simula ser un sistema en producción permitiendo observar el comportamiento de un ataque informático para monitorear, conocer y obtener información del atacante, el ataque y el método implementado [30].

El objetivo de implementar un honeypot en una red es que sea comprometido y atacado por la mayor cantidad de usuarios maliciosos para obtener información relevante de ataques a la red, además [31] aclara que el uso de este distractor no reduce los ataques al sistema.

En opinión de [32] al hacer uso de honeypots es posible:

- Analizar y conocer vulnerabilidades del sistema.
- Persuadir y desviar al atacante obteniendo ventaja en cuanto al tiempo para responder con medidas necesarias.
- Capturar y estudiar nuevas formas de ataques.
- Localizar y conocer atacantes por su dirección IP.
- Obtener información de los atacantes, además de las herramientas, sistemas y aplicaciones usadas para violentar el sistema.

2.2.8.1. HoneyDrive

Distribución propia de Xubunto Desktop que cuenta como una amplia gama de honeypots precargados, entre ellos:

- Dionaea
- Kippo
- Honeyd
- Amun
- Conpot
- HoneydMySQL
- Glastopf

Para la realización de este trabajo se consideró el honeypot kippo y se lo describe a continuación:

2.2.8.1.1. Kippo

Es un honeypot de código abierto diseñado para registrar ataques de fuerza bruta permitiendo detectar fallos a tiempo para conocer nuevas amenazas, estudiarlas e implementar mecanismos de seguridad para mejorar la defensa de la red. [33] lista algunas de las características de este honeypot:

- Basado en Python, un lenguaje de scripting.
- Atrae y distrae posibles atacantes.
- Permite añadir contenido falso de archivos para el atacante.
- Con kippo-graph permite visualizar todas las alertas de forma gráfica.

2.2.9. Zona desmilitarizada

Los avances tecnológicos han llevado a las organizaciones a automatizar sus procesos, por tal motivo hoy en día se ven obligadas al uso de recursos que necesitan de acceso a internet.

La información de una organización que transita por la red mediante correos o almacenamiento en la nube, está expuesta a posibles ataques que comprometen la confidencialidad, disponibilidad e integridad de la información. Por ello existe la zona desmilitarizada (DMZ), que en informática es un área que actúa como filtro, apartando aquellos recursos que requieren de conexiones con alguna red externa, por lo general los hosts de la zona desmilitarizada no se conectan con la red interna, es decir las conexiones cuyo origen van desde la DMZ con destino a la LAN están denegadas, de esta manera se evita comprometer la seguridad de la red interna.

Desde el punto de vista de [34] ,dentro de la DMZ se ubican servidores a los que únicamente se pueden acceder desde la parte externa de la red, como: servidores web, servidores de correo, DNS, HTTP o HTTPS.

2.3. Objetivos del prototipo

2.3.1. Objetivo General

- Diseñar una arquitectura de seguridad perimetral implementando un firewall, IDS, honeypot y zona desmilitarizada asegurando la información que transita en la arquitectura.

2.3.2. Objetivos Específicos

- Establecer fundamentos teóricos que soporten la realización del diseño de la arquitectura.

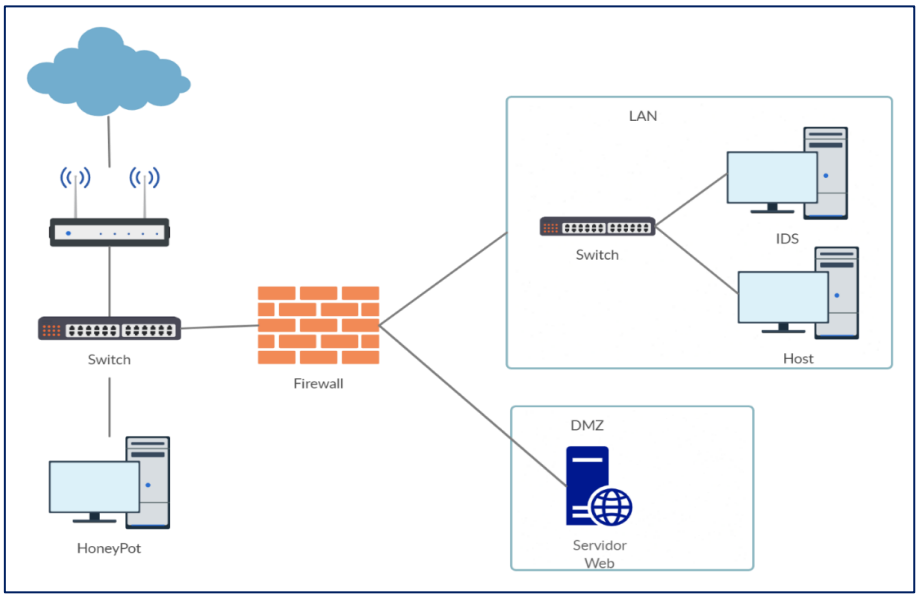
- Implementar un firewall en Linux que permita el control del tráfico en la red mediante políticas de seguridad establecidas.
- Configura el honeypot kippo para atraer ataques en la red.
- Detectar intrusiones en tiempo real con el IDS snort para alertar anomalías en la red.
- Crear una zona desmilitarizada para alojar servicios accesibles en la web.
- Realizar pruebas de pentesting para la verificación de los mecanismos de defensa.

2.4. Diseño del prototipo

El diseño de la arquitectura se ha desarrollado de la siguiente manera, como primero en la línea de defensa se tiene al honeypot, para atraer y detectar al intruso que ha logrado ingresar a la red sin autorización, a su vez este notifica que la red está siendo atacada. Seguido de un firewall como segundo en la línea de defensa, que mediante reglas establecidas controla el tráfico en la red, como tercero se tiene un IDS para detectar intrusos y finalmente una zona desmilitarizada donde se encuentra un servidor web.

Como componentes adicionales se tiene dos switches para conectar los equipos en red.

Figura 2 Diseño del prototipo



Fuente: Elaboración propia

2.4.1. Características de las herramientas utilizadas en la red propuesta.

A continuación, se describen características de las herramientas empleadas para llevar a cabo esta propuesta.

2.4.1.1. Ipfire

Mediante esta herramienta se controla todo el tráfico que transita por la red, restringiendo el paso de acuerdo a reglas establecidas. Este elemento de seguridad interactua con el honeypot, la LAN y la DMZ.

Tabla 1 Características máquina Firewall

Tipo de sistema operativo	Ubuntu
Memoria base	1024MB
Versión	64 bit
Categoría	Firewall
Virtualización	VirtualBox
Memoria de video	16MB

Fuente: Elaboración propia

2.4.1.2. HoneyDrive

Mediante esta herramienta es posible captar información relevante de ataques a la red, es posible conocer el perfil y los patrones de los atacantes [35].

Teniendo en cuenta a [36] se colocó el honeypot antes del firewall debido que al encontrarse fuera de la zona protegida por el firewall puede ser atacado sin afectar a otros elementos que integran la red.

Tabla 2 Caracteristícas máquina Honeypot

Tipo de sistema operativo	Ubuntu
Memoria base	1024MB
Versión	32 bit
Categoría	Honeypot
Virtualización	VirtualBox
Memoria de video	12 MB

Fuente: Elaboración propia

2.4.1.3. Snort

Considerando la propuesta de [37], se ha elegido Kali Linux para evaluar la eficacia de esta herramienta mediante la realización de pruebas de vulnerabilidad, además se ubicó a Snort detrás del firewall.

Tabla 3 Características máquina IDS

Tipo de sistema operativo	Ubuntu
Memoria base	2024MB
Versión	64 bit

Virtualización	VirtualBox
Memoria de video	16MB

Fuente: Elaboración propia

2.4.1.4. Kali Linux

Se empleo este sistema operativo para realizar las pruebas a la arquitectura de red propuesta, la elección de esta distribución se debe a que incluye una alta gama de herramientas preinstaladas para realizar pruebas de pentesting.

Tabla 4 Carácteristicas máquina Kali linux

Tipo de sistema operativo	Kali Linux
Memoria base	2048 MB
Versión	64 bit
Categoría	Firewall
Versión	Kali linux 2020.4
Memoria de video	128 MB

Fuente: Elaboración propia

2.4.1.5. Servidor web

Se eligió Apache, un servidor web HTTP y se encuentra instalado en el sistema operativo Ubuntu y se ubica en la zona desmilitarizada. La elección de este servidor fue por la facilidad de uso para la demostración de esta propuesta.

Tabla 5 Característica máquina servidor web

Tipo de sistema operativo	Ubuntu
Memoria base	2024MB
Versión	64 bit
Virtualización	VirtualBox
Memoria de video	16MB

Fuente: Elaboración propia

2.4.1.6. Ubuntu

Sistema operativo de código abierto bajo de distribución de Linux, ubicado como host en la red de área local (LAN)

Tabla 6 Características máquina Ubuntu

Tipo de sistema operativo	Ubuntu
Memoria base	2024MB
Versión	64 bit
Virtualización	VirtualBox
Memoria de video	16MB

Fuente: Elaboración propia

Definida la arquitectura de red se procede a asignar direcciones IP, que en su posterior serán configuradas en cada una de las máquinas virtuales.

2.4.1.7. Tabla de las direcciones IP de cada componente.

Tabla 7 Tabla de direcciones IP

Máquina	Hostname	IP
Mv1	HoneyDrive3-1	10.10.10.2
Mv2	Firewall-1	10.10.10.3 10.10.9.1 10.10.8.1
Mv3	IDS_ubuntu20-1	10.10.9.3
Mv4	Ubuntu20-1	10.10.9.2
Mv5	Ubuntu20Web-1	10.10.8.2
Mv6	Kali-2020.4-vbox-amd64	10.10.x.x

Fuente: Elaboración propia

La dirección ip de Mv6 varía dependiendo el lugar donde se realicen las pruebas.

2.5. Ejecución y/o ensamblaje del prototipo

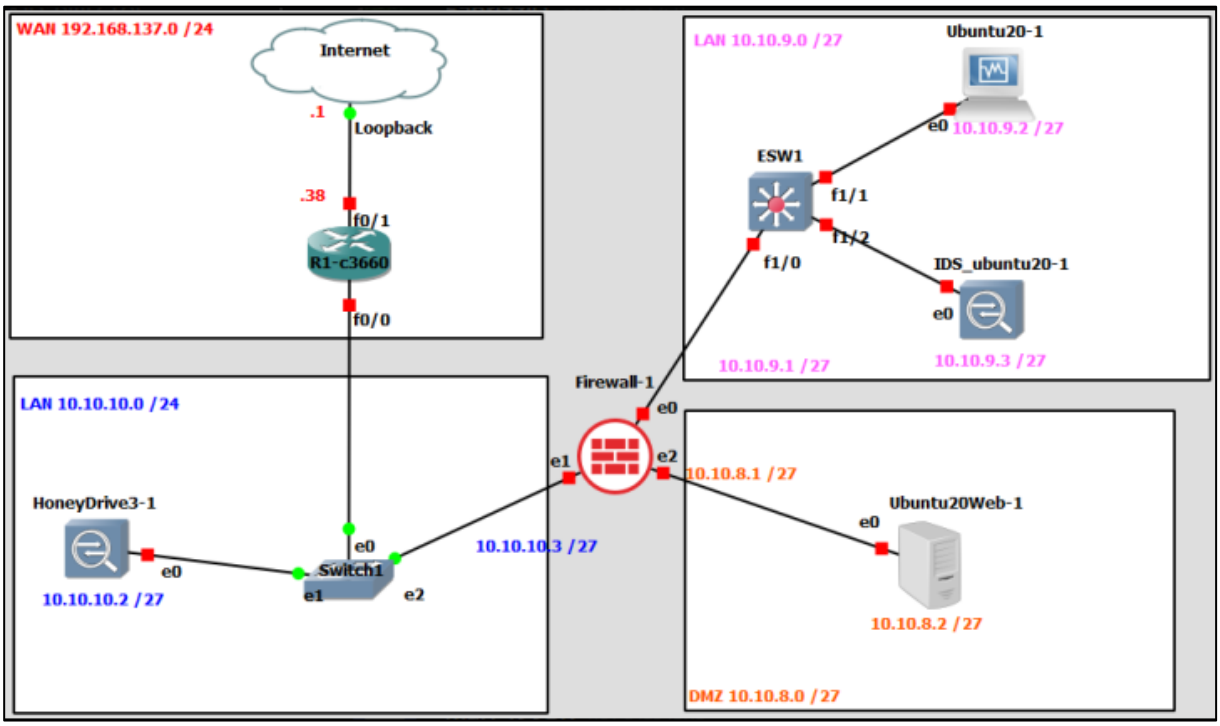
En este ítem se detallan aspectos de la simulación de la arquitectura de red, además de las configuraciones de cada una de las máquinas virtuales.

2.5.1. Instalación de la herramienta GNS3

Tal y como se mencionó en el ítem 1.2, el diseño y simulación de la arquitectura de red se realizará en el programa GNS3.

2.5.2. Simulación de la red

Figura 3 Arquitectura simulada



Fuente: Elaboración propia

Una vez realizada la simulación de la red, se configura el router que se encarga de la conmutación de paquetes de salida a internet, además cuenta con 1 switch para segmentar la red en distintos equipos, era necesario la implementación de otro switch para la red 10.10.9.0 pero un switch común limitaba la funcionalidad del IDS, como alternativa se empleó un router cisco 3640 con una tarjeta NM-16ESW para asemejar el comportamiento de un switch.

Figura 4 Configuración router a switch

```
ESW1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
ESW1(config)#monitor session 1 source interface fastEthernet 1/0
ESW1(config)#monitor session 1 source interface fastEthernet 1/1
ESW1(config)#monitor session 1 source interface fastEthernet 1/3
ESW1(config)#mon
ESW1(config)#monitor see
ESW1(config)#monitor ses
ESW1(config)#monitor session 1 de
ESW1(config)#monitor session 1 destination in
ESW1(config)#monitor session 1 destination interface fas
ESW1(config)#monitor session 1 destination interface fastEthernet 1/2
ESW1(config)#show mon
ESW1(config)#show moni
```

Fuente: Elaboración propia

En cada una de las máquinas virtuales (Mv1, Mv3, Mv4, Mv5 y Mv6) se configura un adaptador, pero para la máquina del Firewall (Mv2) se configuran tres adaptadores, tal como lo describe la siguiente:

Tabla 8 Firewall

RED	WAN	Red externa Zona donde ingresa todo el tráfico
GREEN	LAN	Red interna Se ubican los equipos más sensibles
ORANGE	DMZ	Zona desmilitarizada, es la red que aloja un servidor web y todos los quipos públicos con acceso a internet

Fuente: Elaboración propia

2.5.3. Configuración de Ipfire

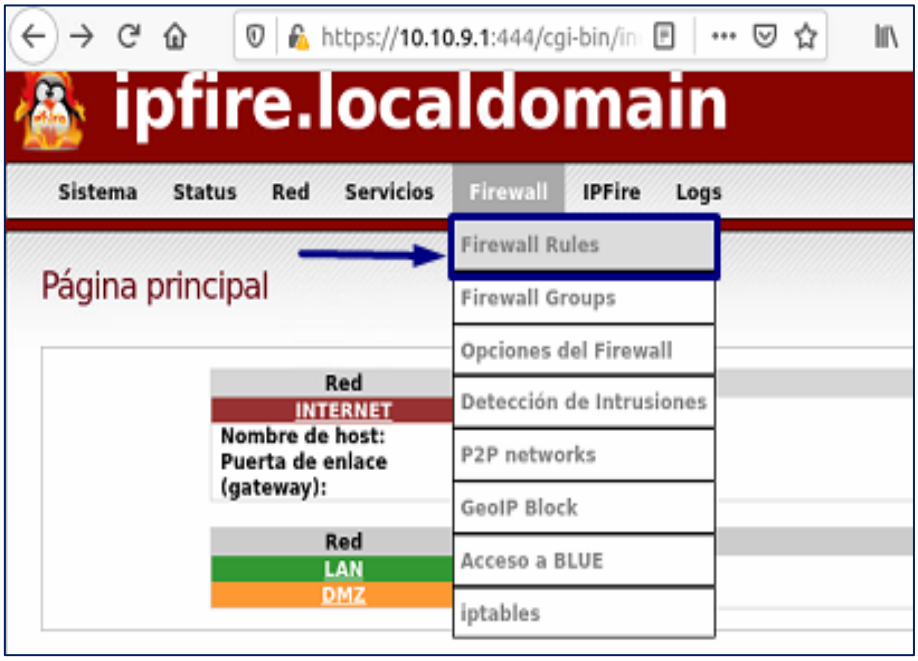
En el **Anexo** 1 se detalla el proceso de instalación ipfire, para la administración se puede hacer uso de comandos y también uso de la web, para este trabajo se realizó la configuración de las reglas de la herramienta vía web por la facilidad de uso.

Para la gestión de esta herramienta es necesario acceder a la plataforma web donde

Es necesario la configuración de reglas que permitan o deniegue el tráfico en la red, una de las actividades a realizar es la asignación o creación de reglas, por ello a continuación se detallan las reglas establecidas para este trabajo.

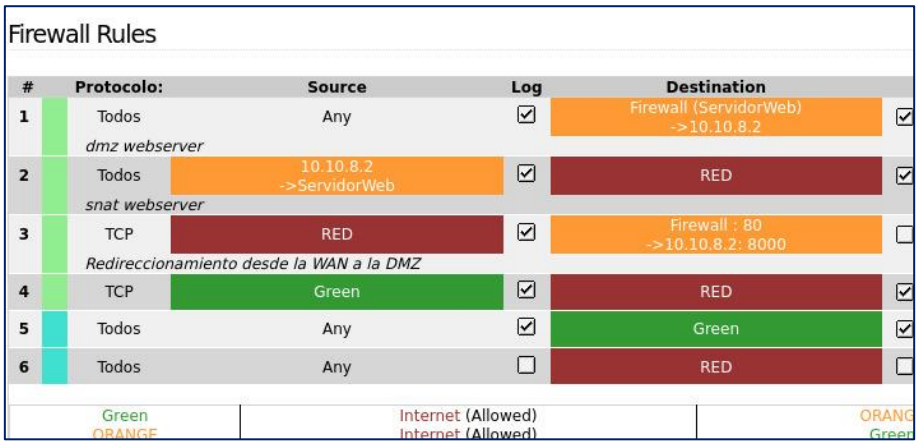
Desde un host en la LAN colocar la puerta de enlace en este caso la IP 10.10.9.1 junto al puerto 444

Figura 5 Ipfire localdomain



Fuente: Elaboración propia

Figura 6 Reglas Ipfire



Fuente: Elaboración propia

1. Permite el acceso de cualquier host a la zona desmilitarizada
2. Permite la salida de la DMZ a una red externa.
3. Redireccionamiento desde la WAN a la DMZ.
4. Desde la red interna (GREEN) cualquier dirección en la zona segura tiene salida a internet

5. Deniega cualquier tráfico de los hosts a la zona segura.
6. No está aplicada.

2.5.4. Configuración de honeydrive

Esta máquina virtual se la empleó para la instalación del honeypot kippo, para la instalación es necesario dirigirse al **Anexo 2**, donde se muestra desde la descarga la implementación de esta herramienta.

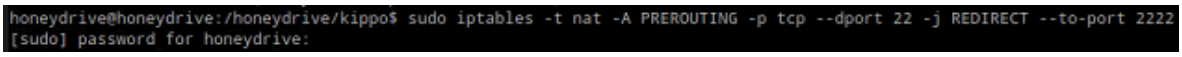
Cuando se inicia la máquina Honeydrive es necesario abrir la terminal para iniciar el servicio de kippo.

Figura 7 Inicialización Kippo

Fuente: Elaboración propia

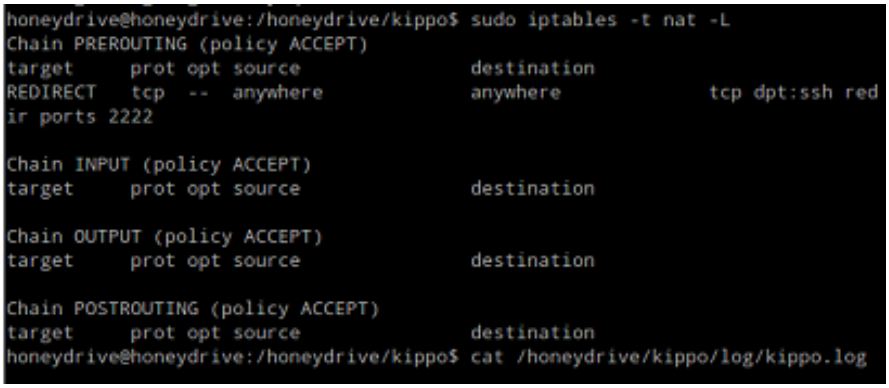
Con iptables se realiza una regla para que toda conexión entrante al puerto 22 lo redireccione al puerto 2222 haciendo creer al atacante que se encuentra en el puerto 22, en la **Figura 8** se encuentra la sintaxis empleada para realizar el redireccionamiento y en la **Figura 9** se verifica la redirección del puerto ha sido realizada.

Figura 8 Redireccionamiento de puertos



Fuente: Elaboración propia

Figura 9 Verificación redirección de puerto



Fuente: Elaboración propia

2.5.5. Configuración de snort

Una vez se ha realizado la instalación del IDS (revisar **Anexo 3** para la instalación) sigue la creación de reglas necesarias para el control de la red y la generación de alertas ante posibles intrusiones no autorizadas.

Todas las configuraciones se almacenan en el archivo snort.conf, donde se pueden añadir, comentar o eliminar reglas ya sea de manera individual o en conjuntos.

Para limitar el tráfico en la red, una de las formas más empleadas por su grado de facilidad es desactivar las reglas mediante la acción de comentar la línea que la contiene.

2.5.6. Creación de la zona desmilitarizada

Esta área es creada con la finalidad de alojar servicios que requieren del acceso a internet. La seguridad y el acceso a esta DMZ ha sido definida y configurada en el Firewall

La red 10.10.8.0 es la red definida para la zona desmilitarizada (área naranja).

Se reserva la dirección 10.10.10.4 la cual a través del firewall se realiza un direccionamiento desde la IP 10.10.8.2 (web server) para que toda solicitud a la 10.10.10.4 sea redireccionada por el firewall al servidor permitiendo el acceso mutuo desde adentro hacia afuera y viceversa.

3. CAPÍTULO III. EVALUACIÓN DEL PROTOTIPO

3.1. Plan de evaluación

Realizado el diseño y la implementación de la arquitectura de red, prosigue la realización de pruebas para evaluar la funcionalidad de la arquitectura propuesta.

La evaluación que se realizará son pruebas de pentesting.

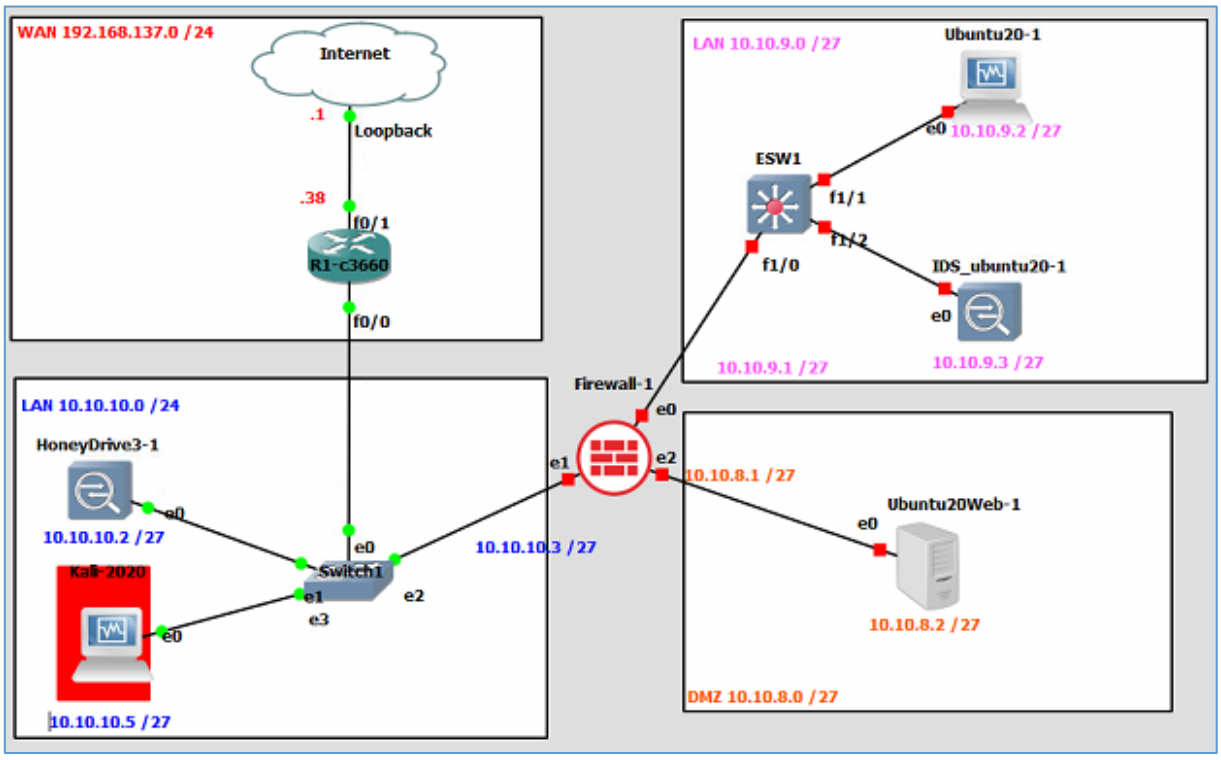
3.1.1. Prueba 1

A criterio de [38], las pruebas de penetración o pruebas de pentesting son un conjunto de técnicas aplicadas para encontrar vulnerabilidades y de esta forma valorar la seguridad informática y minimizar problemas que pueden ser ocasionados por accesos no autorizados con la finalidad de realizar actividades maliciosas

Se realiza pruebas en la red 10.10.10.0 desde Mv6 (Kali 2020), En la **Figura 10**, es posible visualizar que se ha implementado la Mv6 y se asignó la dirección IP 10.10.10.5

Se realizará un ataque de fuerza bruta y el objetivo será Mv1 (10.10.10.2)

Figura 10 Arquitectura figura 1

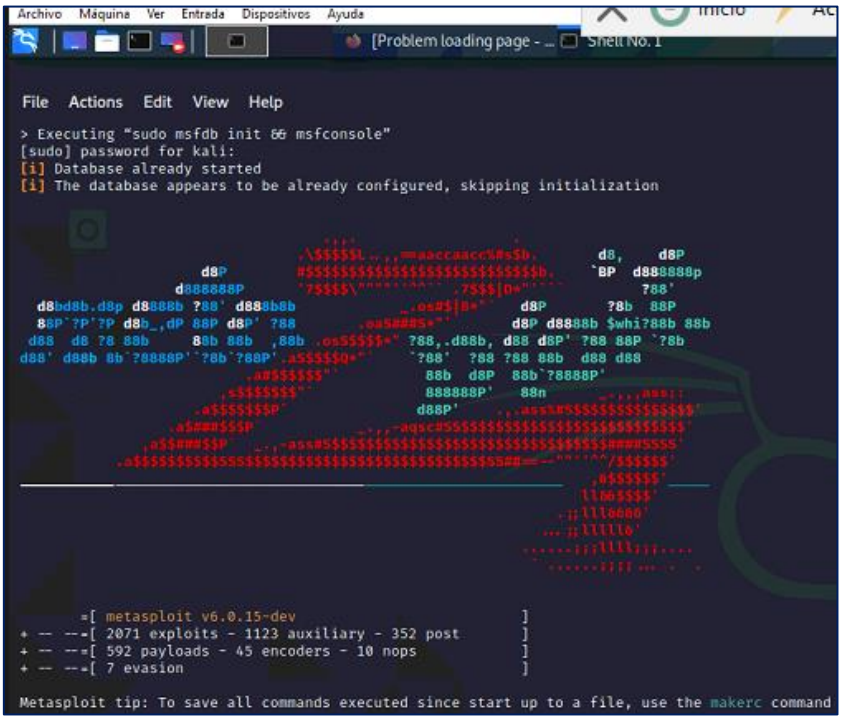


Fuente: Elaboración propia

Para hacer posible esta prueba, se utilizó una de las herramientas propias de Kali Linux para realizar ataques, en este caso fue metasploit, una herramienta de software libre que posibilita identificar amenazas para que el administrador de la red se mantenga al tanto de los ataques que ocurren [39].

En el inicio de Kali se busca la herramienta, y se accede con el password del sistema.

Figura 11 Entorno metasploit



Fuente: Elaboración propia

Desde la consola de metasploit, se realiza un escaneo de servicios con la herramienta nmap, se obtiene que existen 998 puertos cerrados y 3 puertos abiertos en la Mv1 cuya IP es 10.10.10.2

Figura 12 Escaneo Nmap en Metasploit

Fuente: Elaboración propia

Ahora se ejecuta el siguiente comando *use auxiliary/scanner/ssh/ssh_login* para cargar el módulo escáner, cabe destacar *que* ssh_*login* permite realizar de inicio de sesión de fuerza bruta mediante el protocolo ssh.

Figura 13 Scanner SSH login

Fuente: Elaboración propia

Los archivos PASS_FILE y USER_FILE que contienen contraseñas y usuarios en cada linea.

Figura 14 Archivo de contraseña y usuarios

Fuente: Elaboración propia

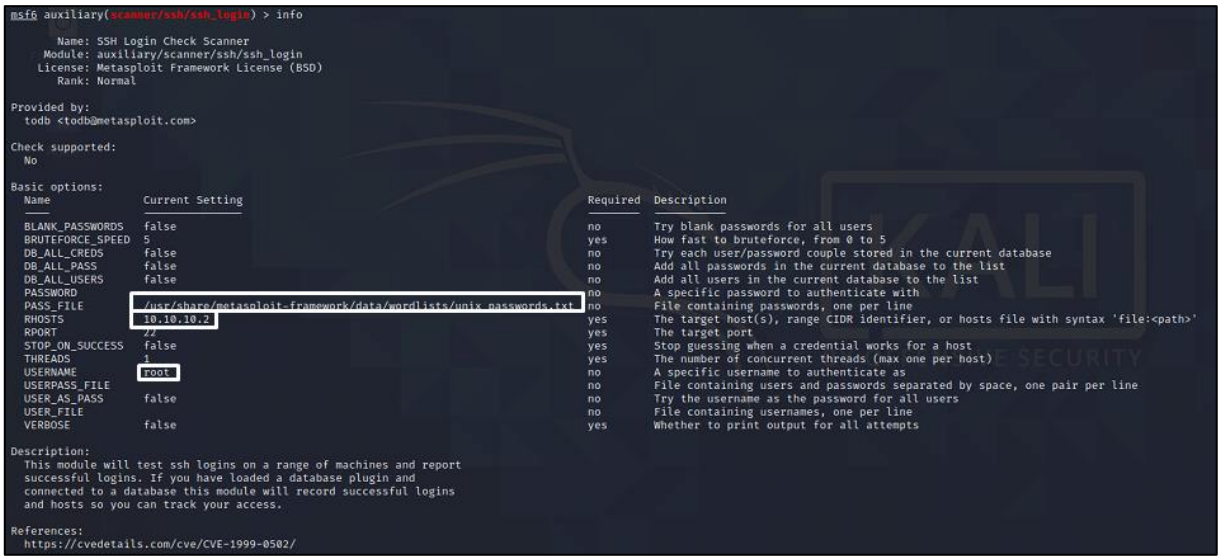
Posteriormente se coloca la el comando *set RHOSTS* junto con la IP del host que actuará como victima, ejecutado el comando anterior sigue, el comando set USERNAME seguido de un nombre de usuario para realizar las pruebas.

Figura 15 Dirección IP y nombre de victima

Fuente: Elaboración propia

Para conocer detalles del ataque a realizar se coloca la palabra *info,* obteniendo información detallada sobre el ataque a realizar.

Figura 16 Información ataque



Fuente: Elaboración propia

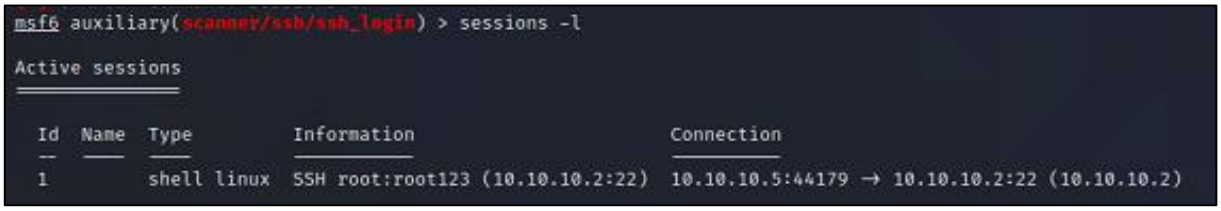
Ahora se procede a ejecutar el módulo auxiliary con la palabra *run.*

Figura 17 ejecucion

Fuente: Elaboración propia

Para conocer la lista de sesiones digitamos el comando *sessions -l,* esto permitirá ver las conexiones remotas obtenidas.

Figura 18 Sesiones

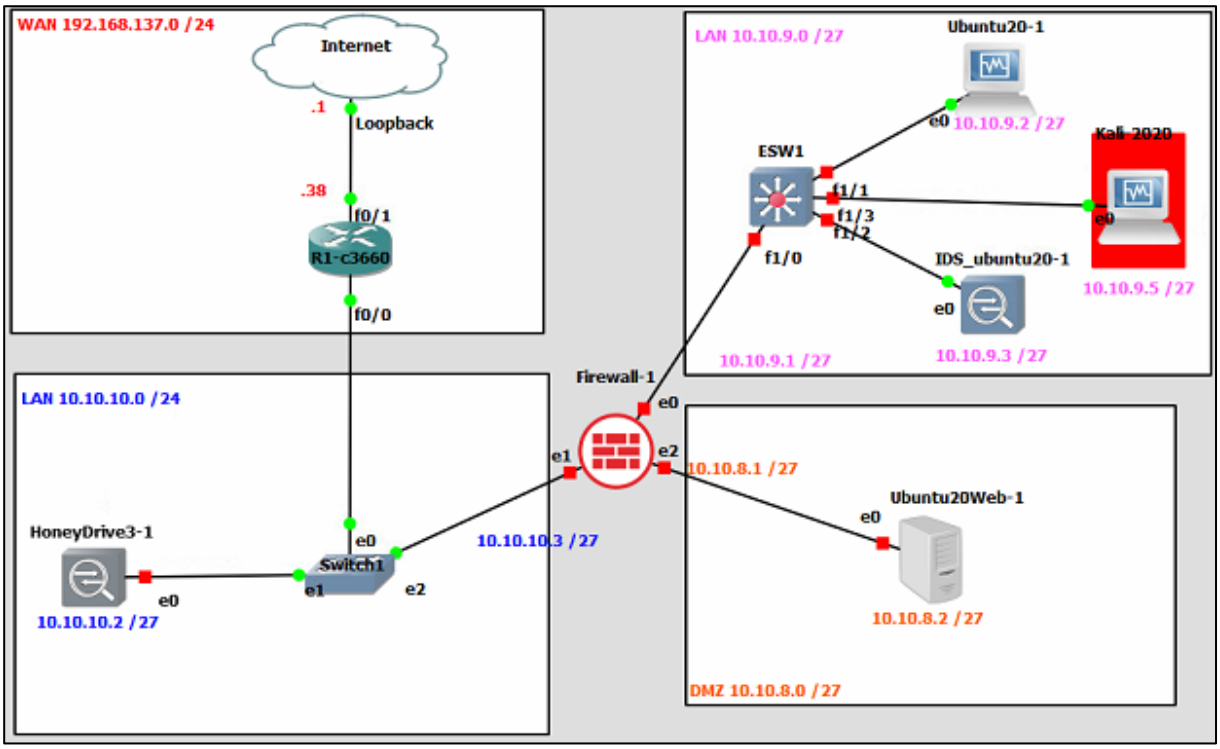


Fuente: Elaboración propia

3.1.2. Prueba 2

Con la realización de esta prueba se busca ejecutar reglas donde Mv3 detecte acciones realizadas por usuarios no autorizados. Para esta prueba Mv6 (Kali-linux 2020) se asignó la dirección IP 10.10.9.5. En la **Figura 19**, se define la arquitectura de red para realizar pruebas donde de la red 10.10.9.0.

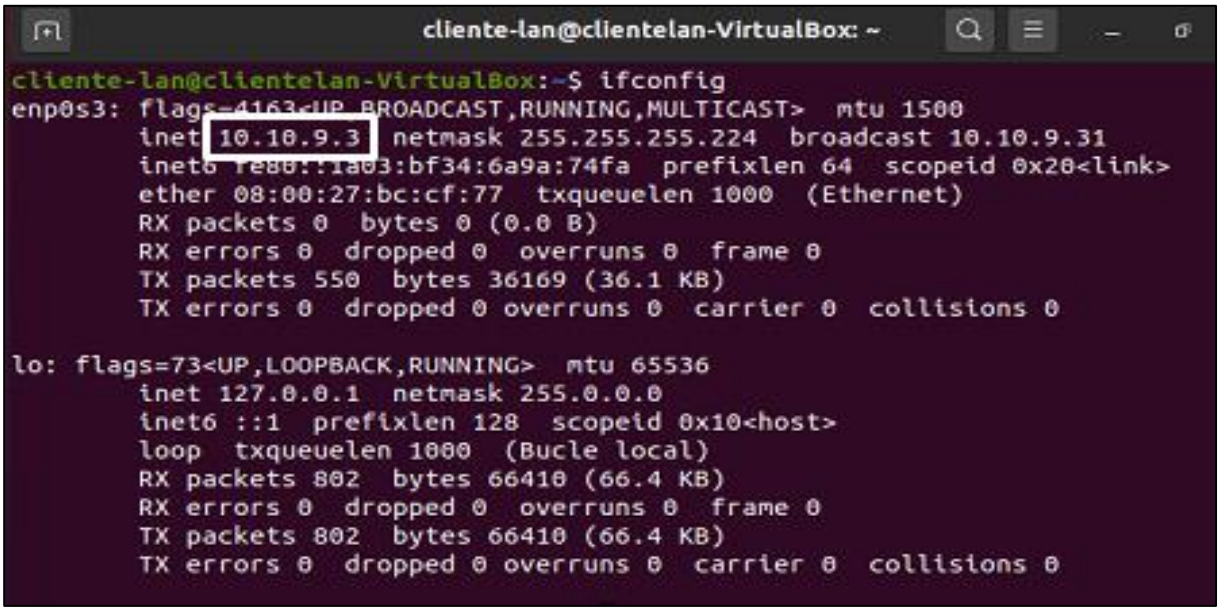
Figura 19 Prueba 2



Fuente: Elaboración propia

Como primer punto se verifica la asignación correcta de la dirección IP a la máquina que contiene el IDS, esto mediante el comando *ifconfig.*

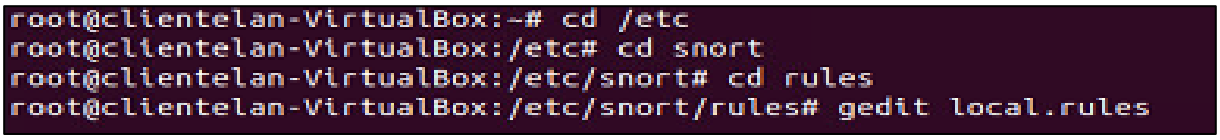
Figura 20 Ifconfig para verificar IP de IDS



Fuente: Elaboración propia

Se crea una nueva regla en el archivo local.rules de la carpeta rules, a la cual es accede tal cual lo indica la **Figura 22**, en esta ubicación se ejecuta el comando *gedit local.rules* para abrir el archivo de reglas locales

Figura 21 Configurar reglas locales



Fuente: Elaboración propia

Una vez sea posible la edición de las reglas locales, se procede a configurar para detectar ataques en red con Snort.

Se realizan dos reglas, la primera para generar una alerta cuando un host realice un ping a la máquina que contiene el IDS, y la segunda para cuando un usuario acceda a una dirección, en este caso www.facebook.com emita una alerta con un mensaje predeterminado. Como dato importante se debe identificar la regla con un número superior a 1000000, esto para evitar problemas de identificación con reglas preestablecidas propias de snort.

Con estas reglas se busca alertar al administrador de acciones realizadas que no han sido autorizadas.

Figura 22 Reglas locales de Snort

```
8 alert icmp 10.10.9.1/27 any -> any any (msg: "ICMP desde maquina atacante";
  sid: 1000001; rev:3;)
9 alert tcp any any -> any any (msg:"El usuario entro a facebook sin permiso
  del jefe"; content:"www.facebook.com";sid:100001;rev:3;)
```

Fuente: Elaboración propia

Para ejecutar la regla se debe editar el archivo snort.conf con el editor de textos gedit.

Figura 23 Archivo snort.conf

```
root@clientelan-VirtualBox:/etc/snort/rules# cd ..
root@clientelan-VirtualBox:/etc/snort# gedit snort.conf
```

Fuente: Elaboración propia

En este archivo se encuentran parámetros necesarios para la ejecución de las reglas, en este caso la variable HOME_NET se delimita una IP que cubre todas las direcciones IP dentro de la red.

Figura 24 Parametros snort

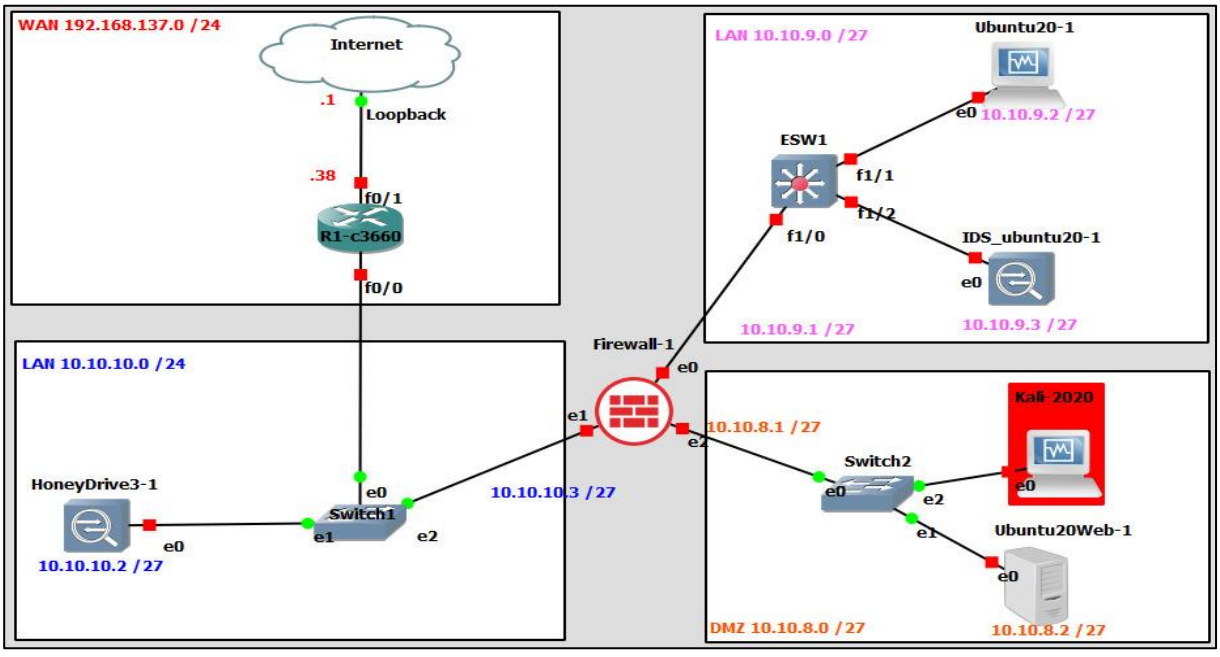
```
51 ipvar HOME_NET 10.10.9.0/27
52 #ipvar 10.10.9.3
```

Fuente: Elaboración propia

3.1.3. Prueba 3

Se realizan pruebas en la red 10.10.8.0 desde Mv6 (Kali 2020), En la **Figura 25**, es posible visualizar que se ha implementado la Mv6 y se asignó la dirección IP 10.10.8.3.

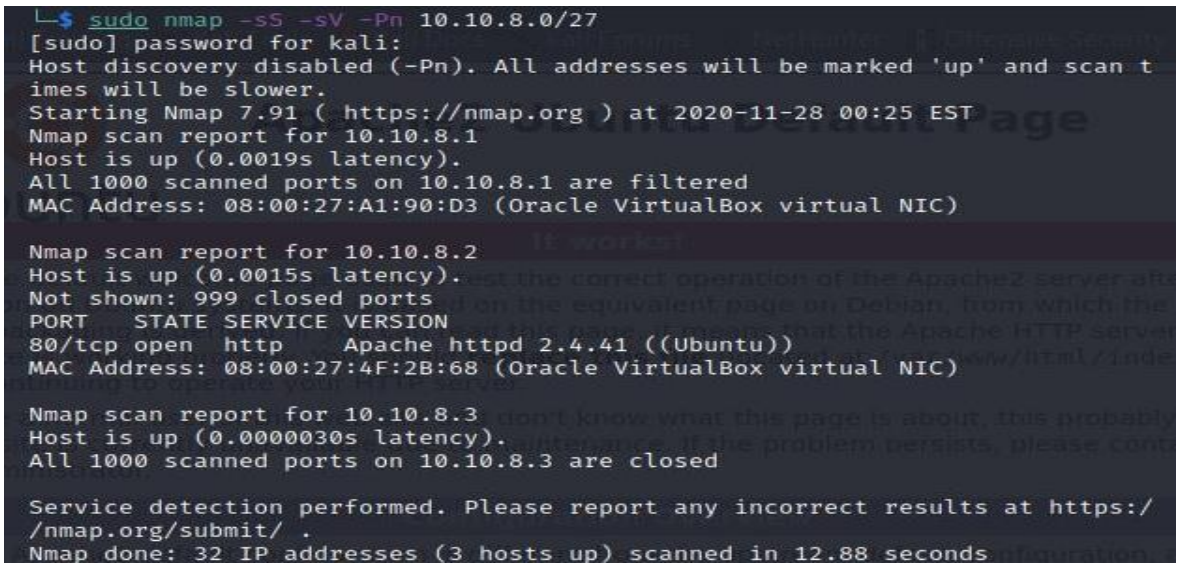
Figura 25 Prueba 3



Fuente: Elaboración propia

La prueba a realizar consiste en escanear los directorios del servidor web por fuerza bruta para encontrar similitudes. Para conocer qué puertos se encuentran accesibles, se utiliza la herramienta nmap. Se hará uso de DirBuster un proyecto de OWASP desarrollado en java con la finalidad de buscar ficheros y directorios alojados en los servidores web para realizar ataque de fuerza bruta [40]. Desde la consola del sistema operativo Kali Linux mediante el comando sudo se ejecuta un escaneo a la IP 10.10.8.0, encontrando libre el puerto 80.

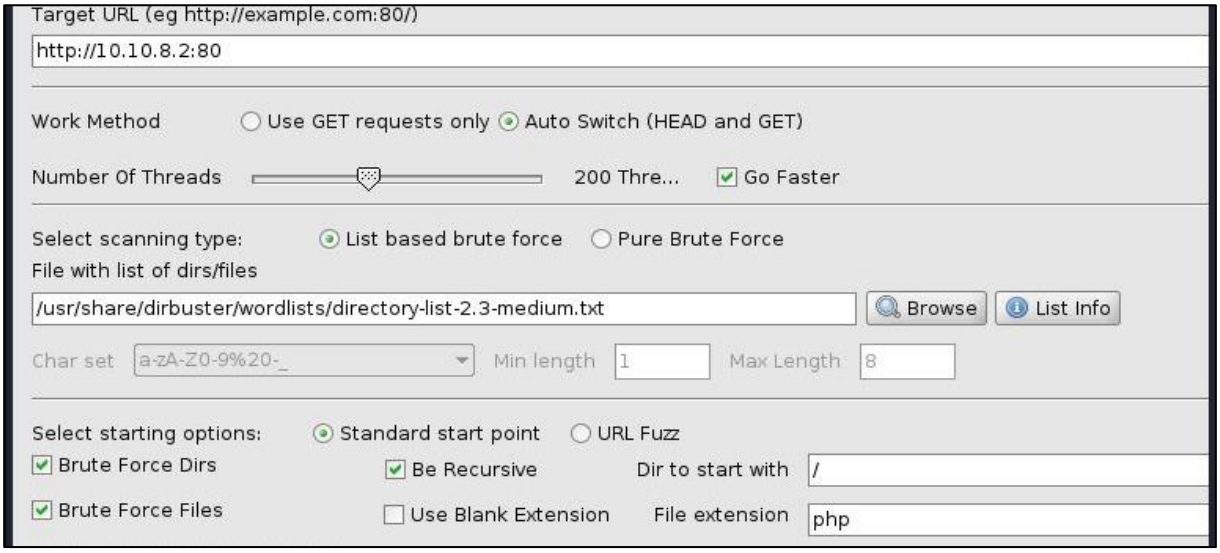
Figura 26 Escaneo Nmap a la red



Fuente: Elaboración propia

Conocido el puerto libre, se usa la herramienta DirBuster donde se proporciona la IP junto al puerto libre para páginas http, se procede hacer clic en el botón start para comenzar el escaneo.

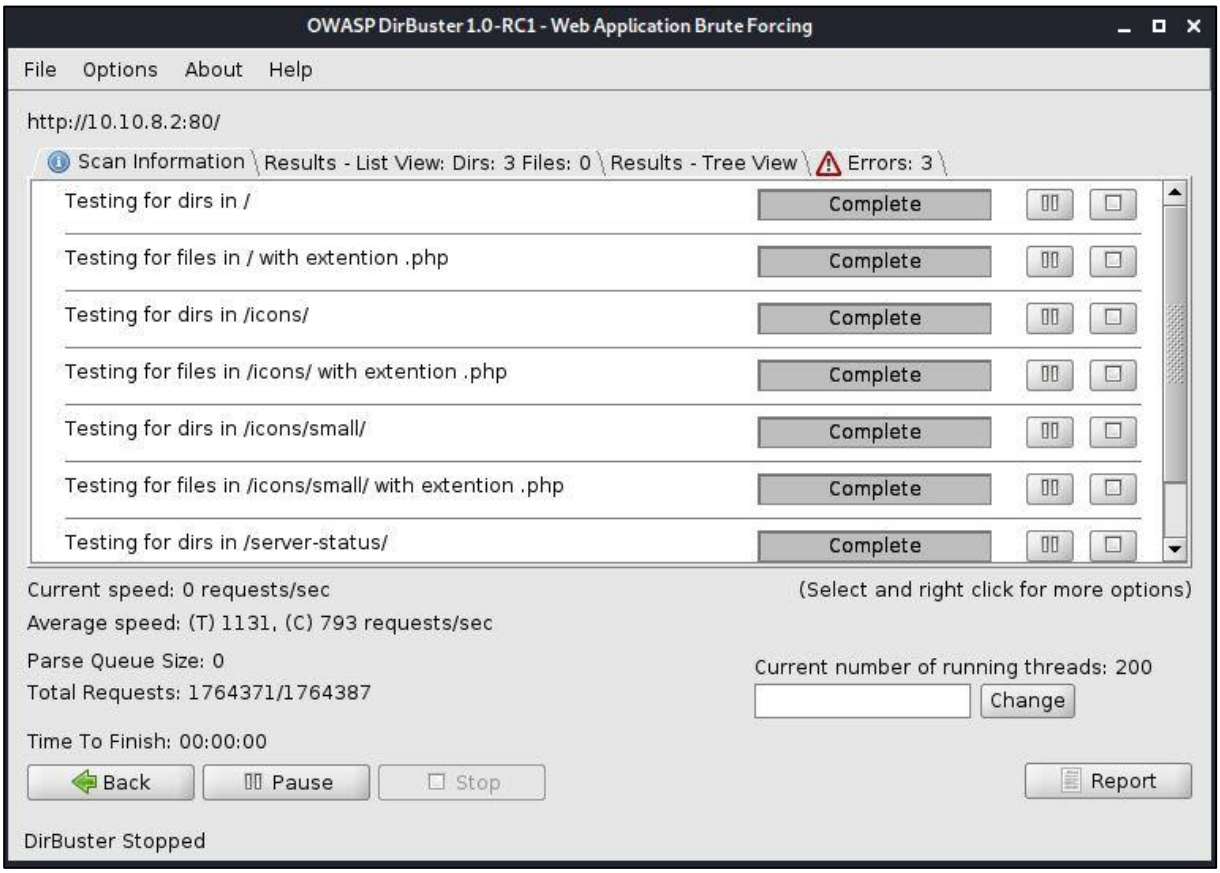
Figura 27 Herramienta DirBuster



Fuente: Elaboración propia

En la **Figura 28** aparecerán los ficheros y directorios encontrados en el proceso.

Figura 28 Escaneo completo de IP



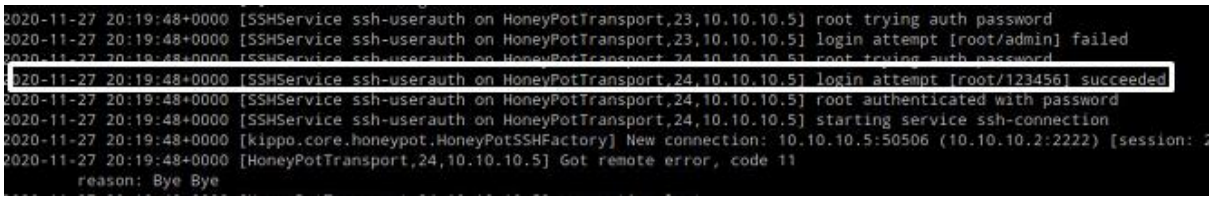
Fuente: Elaboración propia

3.2. Resultados de la evaluación

3.2.1. Resultados de prueba 1

Ejecutando el comando *cat /honeydrive/kippo/log/kippo.log* para observar el archivo log quién está accediendo a Mv1 (10.10.10.2), donde se detalla que Mv6 con la dirección IP 10.10.10.5 mediante servicios SSH logró logearse exitosamente con el usuario: root y el password: 123456

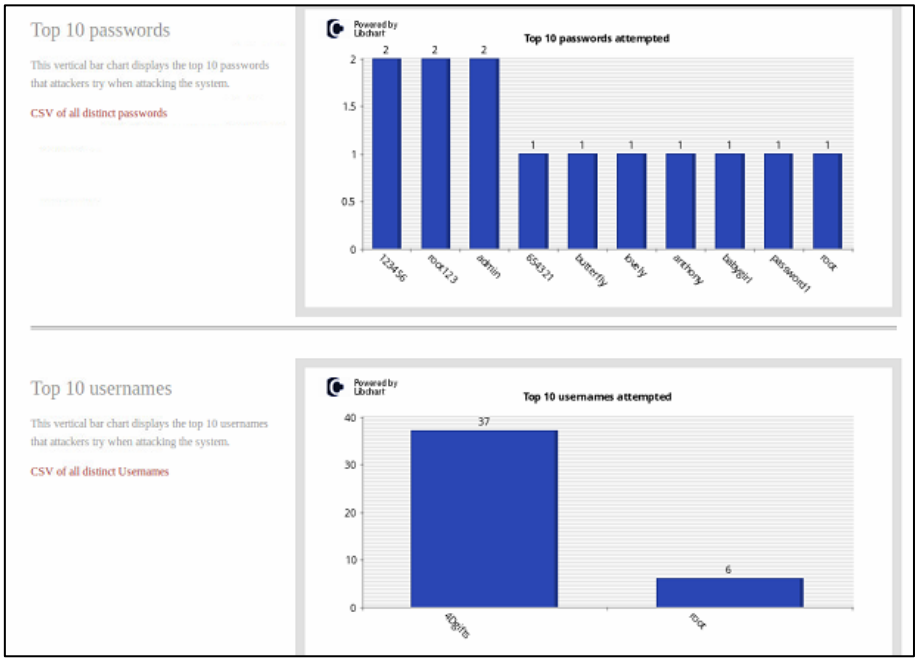
Figura 29 Log de kippo



Fuente: Elaboración propia

Mediante kippo-graph fue posible visualizar de forma gráfica el ataque realizado al host, en la **Figura 30** se visualiza un top 10 de los intentos de usuarios y contraseñas empleados por el atacante.

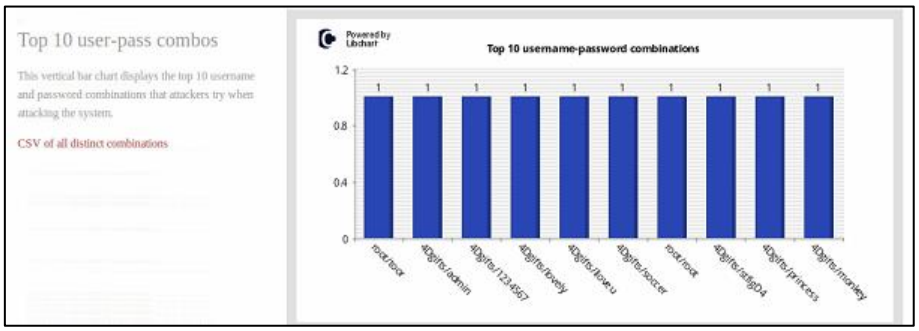
Figura 30 Intento de contraseñas y usuarios



Fuente: Elaboración propia

Además, muestra mediante un gráfico de barras las 10 principales combinaciones tanto de nombre de usuario como de contraseñas que emplean los atacantes.

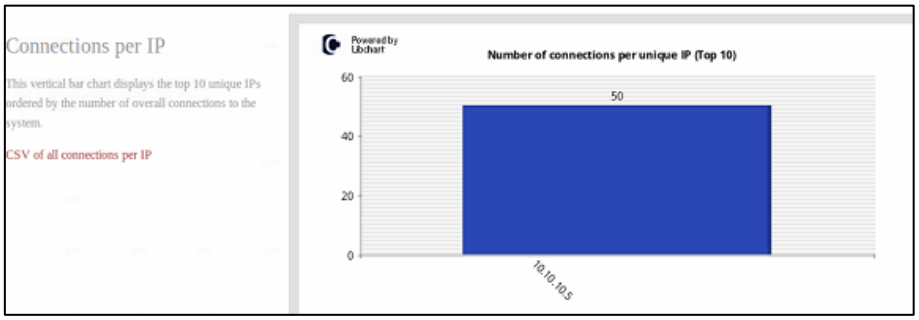
Figura 31 Combinaciones de nombre de usuario y contraseña



Fuente: Elaboración propia

En el gráfico de la **Figura 29** se tiene el número de conexiones por IP que tenía el host del atacante, en este caso fue Mv6 cuya IP asignara es 10.10.10.5

Figura 32 Número de conexiones por IP única



Fuente: Elaboración propia

3.2.2. Resultados de prueba 2

Para ejecutar snort empleamos el comando descrito en la **Figura 33**, donde se envía la orden de leer las reglas por consola en una interfaz determinada a escanear, como resultado se obtiene que el IDS se mantiene pendiente del tráfico de la red.

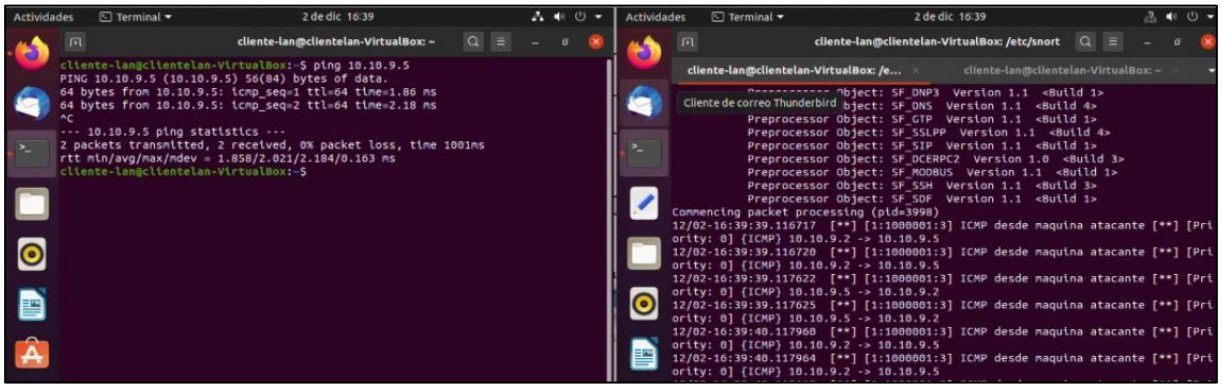
Figura 33 Ejecución snort

```
root@clientelan-VirtualBox:/etc/snort# snort -c snort.conf -A console -i enp0s3
```

Fuente: Elaboración propia

Desde MV4(10.10.9.2) se comprueba conectividad con Mv6(10.10.9.5) mediante el comando ping. En la **Figura 34** se observa que snort envía la alerta en tiempo real enviando el mensaje predeterminado, y muestra la IP del host de destino y del host de origen, cumpliendo la regla de ICMP.

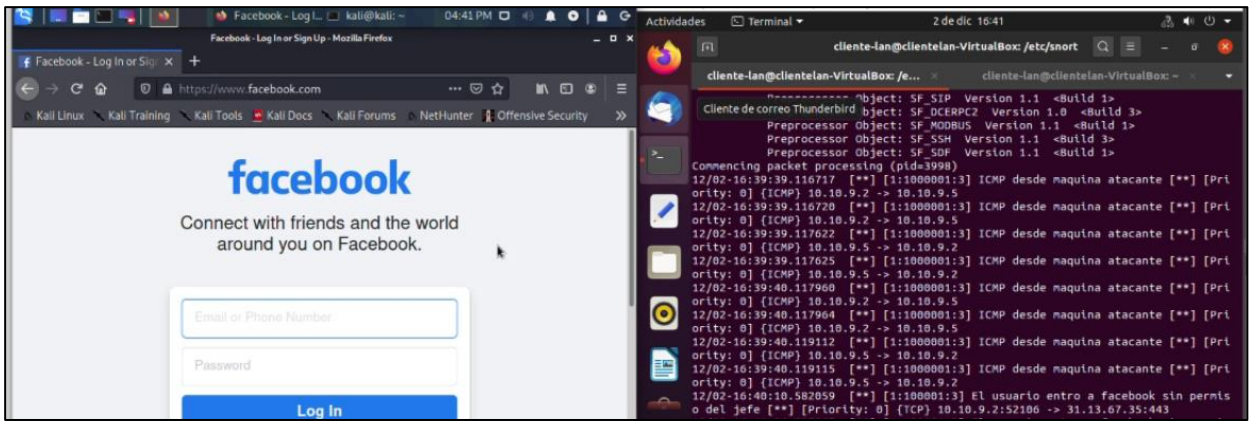
Figura 34 Alerta regla ICMP



Fuente: Elaboración propia

Desde MV4(10.10.9.2) se ingresa la dirección especificada en la regla local (**Figura 32**), y se puede observar que en el IDS se genera la alerta con el mensaje predefinido en la regla establecida. En la alerta enviada por el IDS se observa un mensaje, el protocolo, la IP del equipo que accedió a facebook y el número de puerto.

Figura 35 Alerta regla TCP

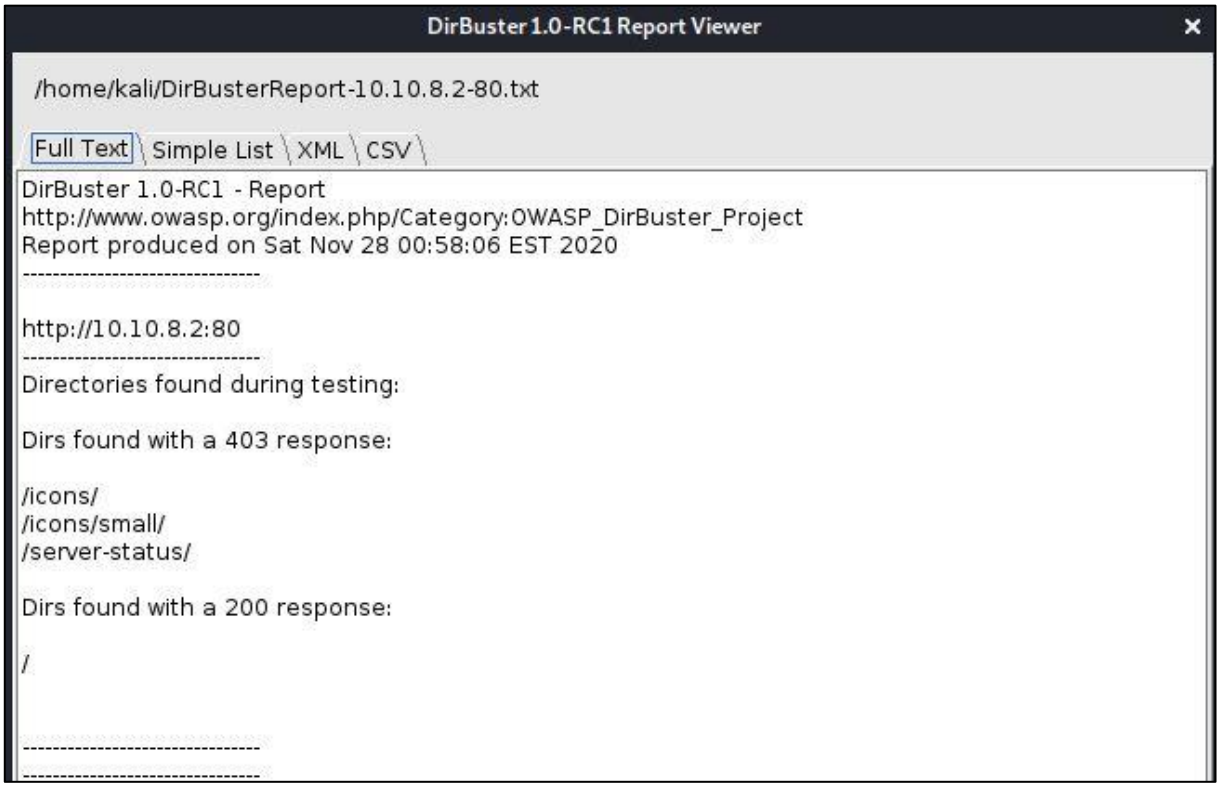


Fuente: Elaboración propia

3.2.3. Resultados de prueba 3

Con la herramienta DirBuster se logró identificar directorios y archivos en el servidor web, información que en sistemas vulnerables resultaría en ataque de SQL injection, suplantación de archivos y virus. En la **Figura 36** de la demostración práctica el software de servidor web APACHE no presenta vulnerabilidades, pero se presencian algunas coincidencias en directorios que pueden ser aprovechadas por un atacante.

Figura 36 Resultados prueba DirBuster



Fuente: Elaboración propia

3.3. Conclusiones

Al término del presente trabajo investigativo se llegan a las siguientes conclusiones:

- Que toda red de trabajo de cualquier organización es constantemente amenazada por ataques informáticos que pueden generar pérdida de la información, por consiguiente, se ve en la necesidad de proteger la información mediante políticas de seguridad aplicadas en la red de trabajo.
- Que, a pesar de no contar con los recursos necesarios de una red en producción, se pudo diseñar una arquitectura de red que permite detectar y alertar frente a intrusiones no autorizadas por el administrador, garantizando la seguridad de la información.

- Que, el uso de un firewall brinda protección a la red, sin embargo, no garantiza la seguridad de todo el perímetro de la red por tal motivo es necesario la combinación de diversos elementos de seguridad que ayuden a la protección de la información ante posibles ataques.
- En la arquitectura propuesta, el honeypot kippo actuó de distractor permitiendo registrar los ataques y mediante kippo graph se logró facilitar de manera visual el registro de los accesos no autorizados dentro de la red.
- Que, la ejecución de las pruebas se comprobó que el IDS Snort fue capaz de detectar y enviar una alerta en el momento que se perpetró una intrusión por protocolo ICMP y TCP, lo que ayudó a corroborar la teoría de la eficacia de esta herramienta basada en reglas.
- Que, mediante el testing con los diccionarios de datos de la herramienta DirBuster se realizó el escaneo al servidor web, identificando coincidencias en los directorios que pueden ser sensibles ante ataques informáticos.

3.4. Recomendaciones

Al término de este trabajo se sugieren las siguientes recomendaciones:

- Se recomienda que toda organización debe contar con una arquitectura de red segura que abarque todo el perímetro para salvaguardar la información, esta debe ser validada mediante pruebas realizadas con herramientas para detectar vulnerabilidades.
- Cuando se diseña una red se debe contar con equipos necesarios que soporten la arquitectura y la implementación de sus componentes para evitar la saturación y daños de los equipos.
- Se recomienda el uso del honeypot kippo en el diseño de una red porque permite atraer y distraer posibles atacantes a la red para dar ventaja al administrador y tomar acciones correctivas.
- Para mantener informado al administrador de la red de las actividades que ocurren en la misma, el IDS Snort es eficaz en la detección de intrusiones y el envío de alertas generadas por las reglas preestablecidas por tanto debe ser considerado en la implementación de una red.
- Se sugiere realizar análisis de seguridad en los servidores web para descartar vulnerabilidades a tiempo y minimizar el riesgo de sufrir ataques, garantizando el correcto funcionamiento y la seguridad del sitio web y de la red.

BIBLIOGRAFÍA

[1] C. Aguilar, "Soluciones open source para seguridad perimetral de empresas PYMES," *Universidad y cambio,* vol. 2, no. 2, pp. 1-14, 2017.

[2] J. Marin, A. Patiño and J. Acevedo, "Implementación de un sistema de seguridad perimetral informático usando VPN, Firewall e IDS," *Revista Universidad Católica de Oriente,* vol. 31, no. 45, pp. 84-99, 2020.

[3] M. Bohorquez and L. Paez, "Diseño de un sistema de seguridad perimetral en las instalaciones del consorcio expansion PTAR salitre, sede Bogotá d.c," Universidad Católica de Colombia, Bogotá, 2017.

[4] F. Morales, S. Toapanta and R. Toasa, "Implementación de un sistema de seguridad perimetral como estrategia de seguridad de la información," *Revista Iberica de sistemas e tecnologías de informacao,* vol. E, no. 27, pp. 553-565, 2020.

[5] C. Galarza, "Diseño e implementación de una red de datos segura para la Pontificia Universidad Catolica del Ecuador," *Ciencias de la computación,* vol. 4, no. 2, pp. 123-137, 2018.

[6] R. Sisalima, Murillo and Hugo, "Rediseño de la infraestructura de una red considerando seguridad perimetral para una institución de educación superior privada de la ciudad de Guayaquil," Universidad de Guayaquil. Facultad de Ciencias Matemáticas y Físicas. Carrera de Ingeniería en Networking y Telecomunicaciones, Guayaquil, 2019.

[7] L. Dayanand, B. Ghorbani and Z. Vaghri, "A survey on the use of GNS3 for virtualizing computer networks," *International Journal of Computer Science and Engineering,* vol. 5, no. 1, pp. 49-58, 2016.

[8] V. Gil and J. Gil, "Seguridad informática organizacional: un modelo de simulación basado en dinámica de sistemas," *Scientia et tecnica,* vol. 22, no. 2, pp. 193-197, 2017.

[9] D. Parada, A. Florez and U. Gómez, "Análisis de los Componentes de la Seguridad desde una perspectiva Sistémica de la Dinámica de Sistemas," *Información Tecnológica ,* vol. 29, no. 1, pp. 27-38, 2018.

[10] F. Solarte, E. Enriquez and M. Benavides, "Metodología de análisis y evaluación de riesgos aplicados a la seguridad informática y de información bajo la norma ISO/IEC 27001," *Revista tecnológica ESPOL,* vol. 28, no. 5, pp. 492-507, 2015.

[11] M. Yandún, E. Cando and D. Mora, "Fuga de información confidencial en las instituciones financieras y uso de data Loss Prevention," *Visión empresarial,* no. 8, pp. 42-49, 2016.

[12] M. Romero, G. Figueroa, D. Vera, J. Álava, G. Parrales, C. Álava, Á. Murillo and M. Castillo, Introducción a la seguridad informática y el análisis de vulnerabilidades, Alicante: 3 Ciencias, 2018.

[13] M. Corda, M. Viñas and M. Coria, "Gestión del riesgo tecnológico y bibliotecas: una mirada transdisciplinar para su abordaje," *Palabra clave (La plata),* vol. 7, no. 1, pp. 1-19, 2017.

[14] K. Pintado and C. Hurtado, "Diagnóstico de las vulnerabilidades informáticas de los sistemas de información para proponer soluciones de seguridad a la rectificadora Gabriel Mosquera S.A.," Universidad Politécmoca Salesiana, Guayaquil, 2015.

[15] K. Kovalenko, N. Kovalenko and J. Gonzalez, "A variety of information security threats," *Universidad y sociedad,* vol. 11, no. 5, pp. 256-261, 2019.

[16] J. Figueroa, R. Rodriguez, C. Bone and J. Saltos, "La seguridad informática y la seguridad de la información," *Polo del conocimiento,* vol. 2, no. 12, pp. 145-155, 2017.

[17] J. Rivera, V. Herrera, X. Naranjo and C. Narváez, "Gestión de Riesgos de TIC en hospitales públicos," *Revista Ibérica de Sistemas e Tecnologias de Informação,* no. E20, pp. 280-291, 2019.

[18] N. Villacrés, "Esquema de seguridad perimetral para el monitoreo de eventos en el instituto tecnológico Bolivariano," Universidad Regional autónoma de los Andes, Ambato, 2018.

[19] M. Lescay, L. E. L. Montoya, G. Torre and L. Barrera, "Estrategia de superación para la utilización de proxmox y pfSense en las instituciones de salud," *Revista cubana de informática Médica,* vol. 11, no. 2, pp. 100-114, 2019.

[20] J. Bolaños, "Diseño de la arquitectura de seguridad perimetral de la red informática en la industria de licores del valle," Universidad Autónoma de Occidente, Santiago de Cali, 2018.

[21] H. Ramilez and J. Mejia, "Propuesta de infraestructura técnica de seguridad para un Equipo de Respuesta ante Incidentes de seguridad (CSIRT)," *Revista electrónica de Computación, Informática Biomédica y Electrónica,* vol. 4, no. 1, 2015.

[22] L. Juan, "Estudio y propuesta de diseño para la arquitectura de seguridad perimetral de campus, caso de estudio data center para el muncipio del distrito metropolitano de Quito," Pontificia Universidad Católica del Ecuador, Quito, 2016.

[23] O. Callegari, "Firewall / Cortafuegos," *Revistas negocios de seguridad,* pp. 180-184, 2008.

[24] P. Delgado and L. Loor, "Seguridad perimetral firewall y fortalecimiento de la seguridad en el data center de la ESPAM MFL," Escuela superior politécnica agropecuaria de Manabí Manuel Féliz López, Calceta, 2017.

[25] C. Ocampo, V. Castro and G. Solarte, "Sistema de detección de intrusos en redes corportativas," *Scientia Et Technica,* vol. 22, no. 1, pp. 60-68, 2017.

[26] J. Mejia and H. Ramirez, "Establecimiento de controles y perímetro de seguridad para una página web de un CSIRT," *Revista ibérica de sistemas y tecnologías de la información,* vol. 3, no. 17, pp. 1-15, 2016.

[27] F. Reyes, W. Fuertes and C. Guzman, "Application of business intelligence For analyzing vulnerabilities to increase the security level in an academic CSIRT," *Revista Facultad de Ingeniería,* vol. 27, no. 47, pp. 21-29, 2018.

[28] S. Quiroz, J. Zapata and V. Héctor, "Predicción de ciberataques en sistemas industriales SCADA a través de la implementación del filtro Kalman," *TecnoLógicas,* vol. 23, no. 48, 2020.

[29] F. de Haro, "Detección de intrusiones con Snort," Universitat Oberta de Catalunya, 2015.

[30] M. Traver, "Honeypots. L'art de la Guerra," Universitat Autònoma de Barcelona., 2020.

[31] C. Gonzáles, "Detectando honeypots: Estudio, análisis y mejora de la ofuscación de honeypots.," Universidad Autónoma de Madrid, Madrid, 2018.

[32] M. Leguizamón, M. Bonilla and C. León, "Analysis of computer attacks through Honeypots in the District University Francisco José de Caldas," *Ingenieria y competitividad,* vol. 22, no. 2, pp. 1-13, 2019.

[33] M. Solomon and P. Avadhani, "Honeypot System for Attacks on SSH Protocol," *Computer Network and Information Security,* vol. 8, no. 9, pp. 19-26, 2016.

[34] C. Batista, Z. Lujo, L. Cedeño, A. Perez and R. Pantaleon, "Propuesta e implementación de la arquitectura de la red LAN empresa ACINOX Las Tunas," *Revista de Investigación en Tecnologías de la Información,* vol. 6, no. 11, pp. 1-6, 2018.

[35] M. Palmay, "Propuesta de mejores prácticas para el establecimiento de políticas de seguridad informática basado en honeynet Virtuales," Escuela Superior Politécnica de Chimborazo, Riobamba, 2017.

[36] R. Gonzales, E. Leaños, F. Rodríguez and R. Saca, "Honeypots para la detección de ataques informáticos realizados a instituciones financieras caso de estudio: Normativa nacional - ASF," *Revista UTEPSA,* no. 4, pp. 15-37, 2019.

[37] R. Gaddam and M. Nandhini, "An Analysis of Various Snort Based Techniques to Detect and Prevent Intrusions in Networks," in *International Conference on Inventive Communication and Computational Technologies*, Coimbatore, 2017.

[38] H. González, "Capacidades de las metodologías de pruebas de penetración para detectar vulnerabilidades frecuentes en aplicaciones web," *Revista cubana de ciencias informáticas,* vol. 12, no. 4, pp. 52-65, 2018.

[39] A. Rodríguez, "Herramientas fundamentales para el hacking ético," *Revista Cubana de Informática Médica,* vol. 12, no. 1, pp. 116-131, 2020.

[40] L. Quirola, "Análisis de Vulnerabilidades de Seguridad Informática, del Sistema de Gestión Médica SISMEDICALEC, de la empresa Incomsis.," Universidad Técnica de Ambato, Ambato, 2019.

ANEXOS

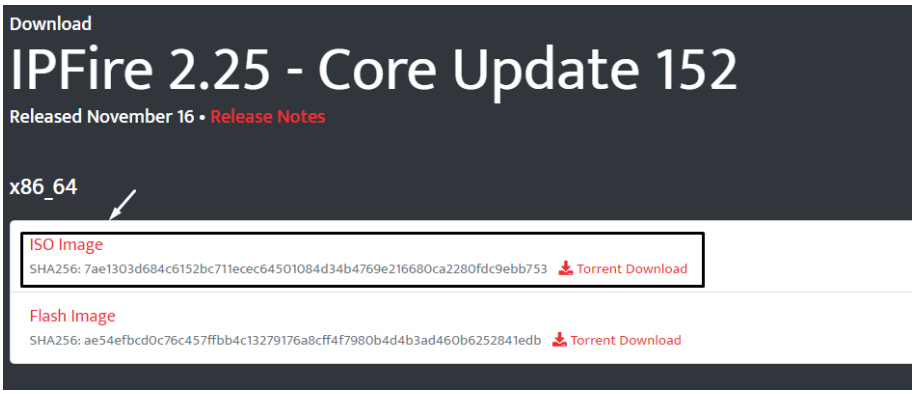
Anexo 1

Descarga e instalación de Ipfire

Link de descarga: [www.ipfire.org - IPFire 2.25 - Coe Update 152]

Ingresando al link proporcionado para descargar esta herramienta, aparecerá la opción de obtener la imagen ISO, dependiendo de la versión del sistema operativo se elegirá la imagen a descargar.

Figura 37 Imagen ISO ipfire



Fuente: Elaboración propia

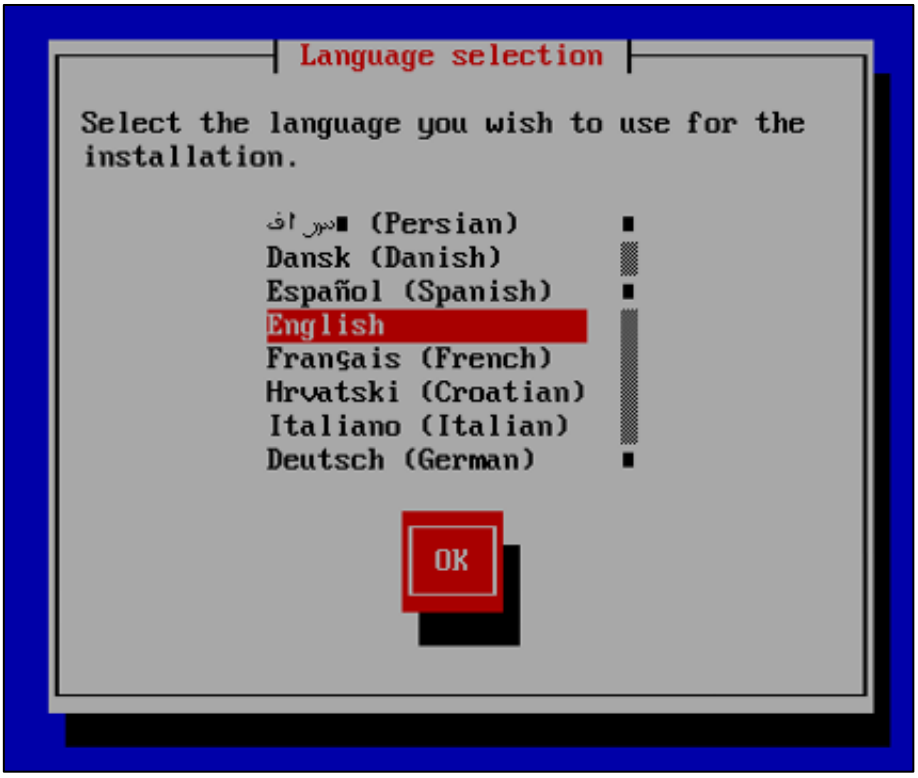
Descargada la ISO, se crea una máquina virtual cuyas características están descritas en la **Tabla 2,** seguido se carga la imagen en la máquina creada.

Figura 38 Instalación IPFire

Fuente: Elaboración propia

Se realiza la selección del Idioma para el firewall, para esta ocasión se define español como idioma de la máquina virtual.

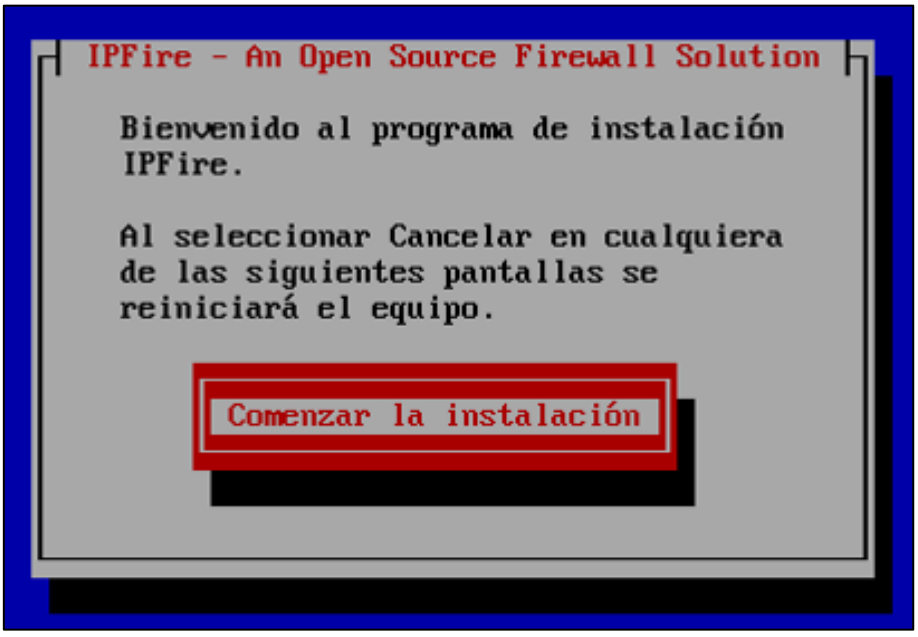
Figura 39 Selección idioma



Fuente: Elaboración propia

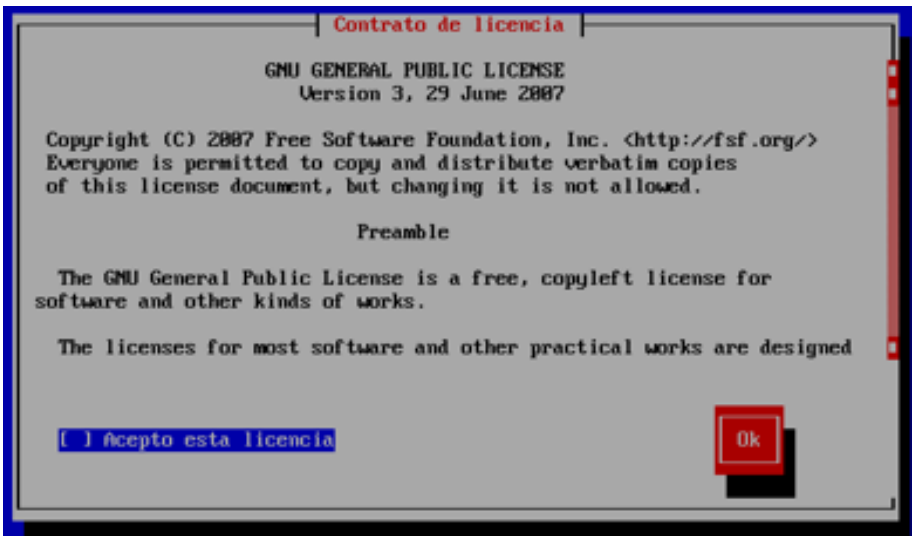
En la ventana presentada en la **Figura 40,** se autoriza el inicio de la instalación de la herramienta IPFire, y en su posterior (**Figura 41**) se acepta el contrato de licencia de la herramienta.

Figura 40 Inicio instalación IPFire



Fuente: Elaboración propia

Figura 41 Aceptar licencia IPFire



Fuente: Elaboración propia

Es necesario eliminar todos los datos que contiene el disco duro para crear un sistema de archivos donde se aloja el sistema.

Figura 42 Disco de Instalación

Fuente: Elaboración propia

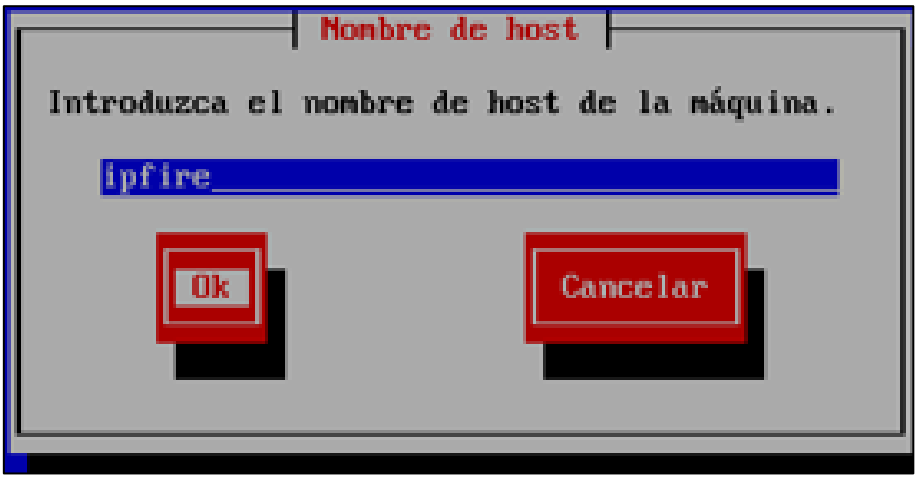
Finalmente aparecerá la ventana que indica que IPFire fue instalado con éxito. Lo que sigue es elegir el sistema de archivos, para este caso se selecciona el sistema de archivos ext4

Figura 43 Selección de sistema de archivos

Fuente: Elaboración propia

Luego se debe configurar el mapa de teclado y la zona horaria de acuerdo al lugar de instalación. Posterior se introduce el nombre del host (**Figura 44**) y el nombre del dominio (**Figura 45**)

Figura 44 Nombre de host IPFire



Fuente: Elaboración propia

Figura 45 Nombre del dominio

Fuente: Elaboración propia

El siguiente paso es necesario asignar un password de administrador

Figura 46 Contraseña IPFire

Fuente: Elaboración propia

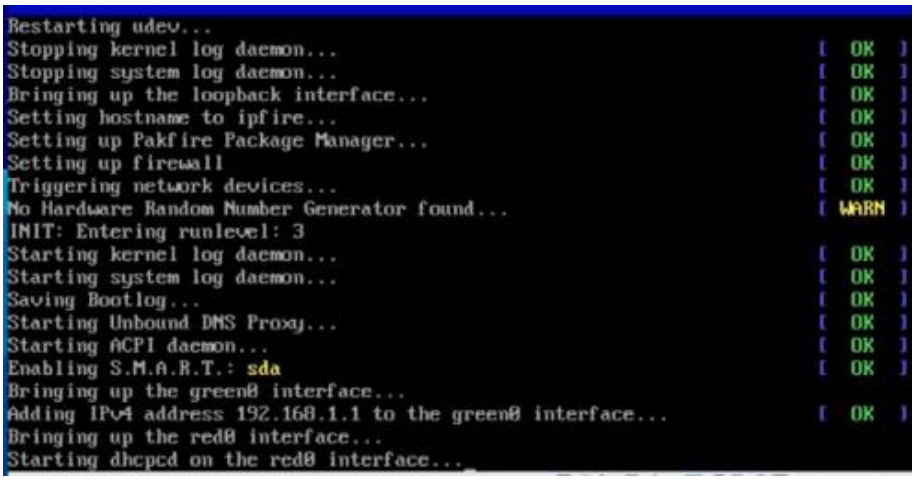
Finalmente se configuran los parámetros de red, para este caso se elige el tipo de configuración GREEN+RED+ORANGE, en la **Tabla 8** se describe su uso

Figura 47 Tipo de configuración de red

Fuente: Elaboración propia

Realizados los pasos anteriores el sistema se reiniciará, y al arrancar se observa la suite de seguridad.

Figura 48 IPFire después de su instalación



Fuente: Elaboración propia

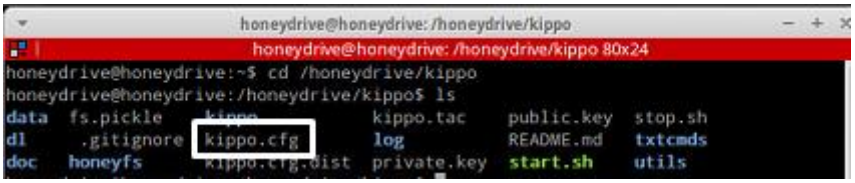
Anexo 2

Descarga e instalación de honeydrive

Link de descarga: https://sourceforge.net/projects/honeydrive/.

Desde el terminal no dirigimos a la carpeta kippo para acceder al archivo de configuración kippo.cfg

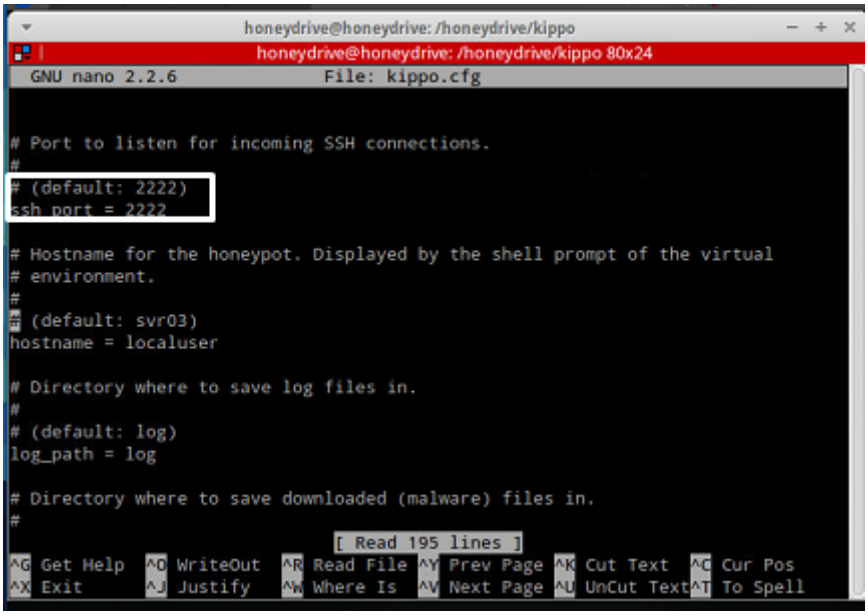
Figura 49 Archivo de configuración



Fuente: Elaboración propia

Se abre el archivo con el comando *nano* y se verifica que se encuentre en el puerto 22, que es el puerto por defecto para el servicio SSH.

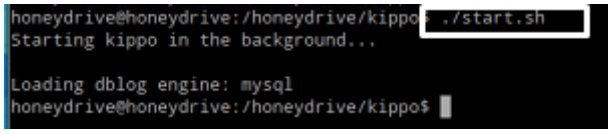
Figura 50 Verificación puerto 22



Fuente: Elaboración propia

Verificado que se encuentre en el puerto 22, se procede a realizar la inicialización de kippo y el servicio SSH a la espera de usuarios no autorizados en la red.

Figura 51 Iniciar Kippo

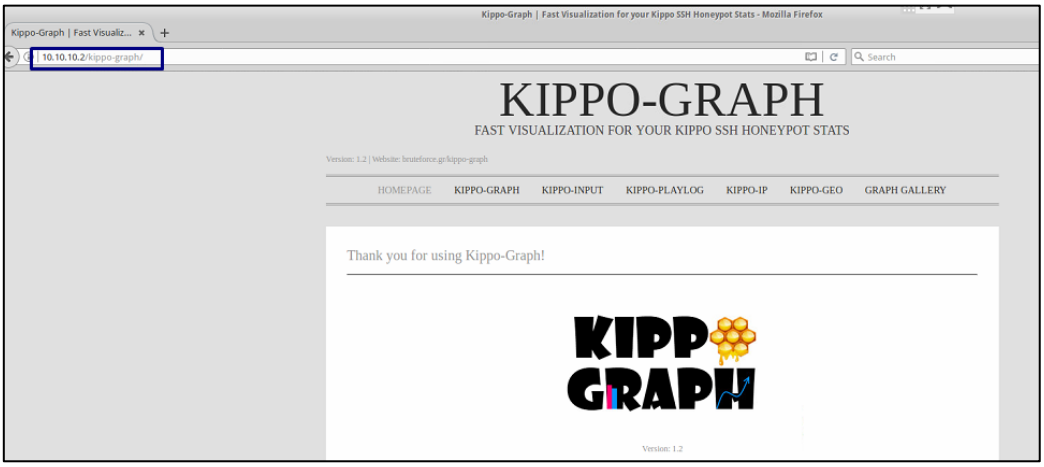


Fuente: Elaboración propia

Esta herramienta cuenta con un entorno gráfico donde es posible visualizar mediante cuadros estadísticos que incluyen las visitas no autorizadas y la información de aquellos atacantes que estuvieron en la red.

Para acceder nos dirigimos al navegador e insertamos la dirección: http://10.10.10.2/kippo-graph/

Figura 52 Kippo-Graph



Fuente: Elaboración propia

Anexo 3

Descarga e instalación de snort

Para instalar esta herramienta se requiere tener instalada una distribución de Linux, se ha elegido Ubuntu.

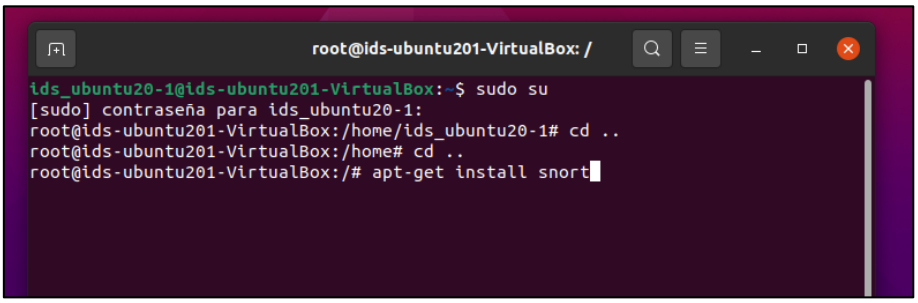
Ingresar los siguientes comandos para actualizar y optimizar el sistema *sudo apt-get update && upgrade*

Ahora, es necesario descargar 9 dependencias que utilizara el IDS Snort

Una vez el sistema cuente con las dependencias mencionadas, se requiere reiniciar la máquina, esto es posible con *reboot*

Desde la consola, se inicia sesión con el usuario root y se procede a instalar el sistema detector de intrusos mediante el comando *apt-get install snort*

Figura 53 Sudo su



Fuente: Elaboración propia

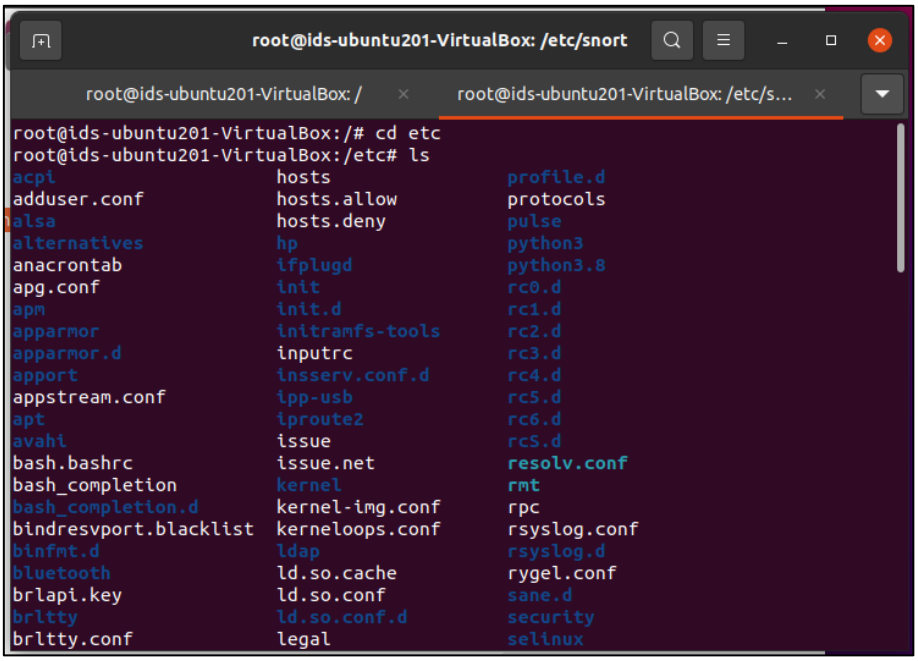
Se presentan instrucciones para la configuración de paquetes del IDS snort, donde se asigna la interfaz que está usando la máquina virtual y por defecto asignará la dirección IP.

Figura 54 Configuración Snort

Fuente: Elaboración propia

Revisamos los archivos que existen en la carpeta etc para verificar la carpeta snort.

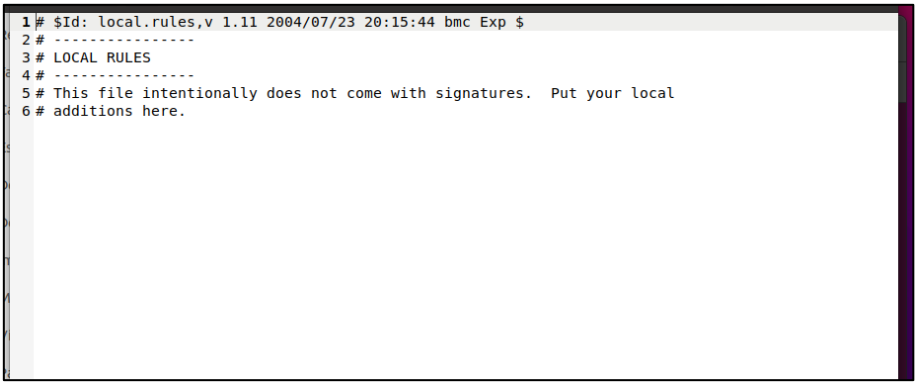
Figura 55 Carpeta etc



Fuente: Elaboración propia

Se procede a revisar el archivo de reglas locales, donde para la práctica se hará uso de él, la revisión de este archivo es con la finalidad de comprobar que no vienen reglas definidas dentro de este archivo por lo que pueden ser creadas, lo mismo en la biblioteca de reglas dinámicas snort,conf.

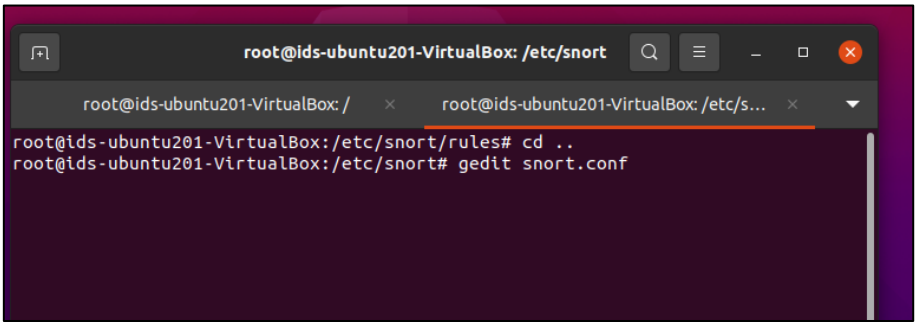
Figura 56 Reglas locales de Snort



Fuente: Elaboración propia

Ahora se debe buscar el archivo de configuración de snort, mediante el comando *gedit snort.conf*, donde se asignan los archivos de reglas nuevas junto con los archivos de reglas existentes.

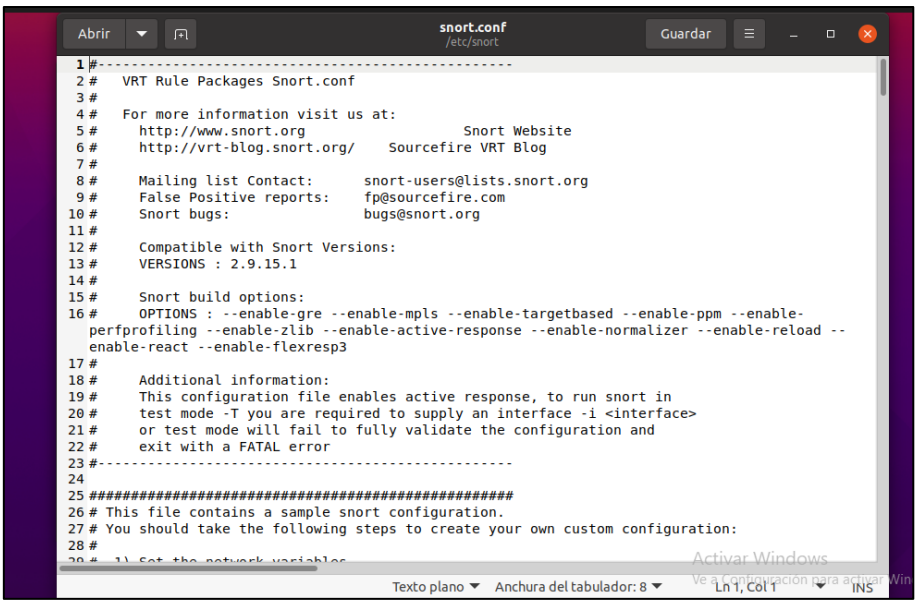
Figura 57 Snort.conf



Fuente: Elaboración propia

Con el comando editor de texto gedit, podemos observar e ingresar configuraciones de la herramienta, finalmente se digita ctrl+x para guardar cambios.

Figura 58 Ruta a la biblioteca de reglas dinámicas



Fuente: Elaboración propia