UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

ANALISIS Y DISEÑO DE UNA ARQUITECTURA DE RED UTILIZANDO LAS HERRAMIENTAS HONEYPOT, IDS Y FIREWALL

PORRAS SURIAGA CELMIRA MARCELA
INGENIERA DE SISTEMAS

MACHALA
2020

UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

ANALISIS Y DISEÑO DE UNA ARQUITECTURA DE RED UTILIZANDO LAS HERRAMIENTAS HONEYPOT, IDS Y FIREWALL

PORRAS SURIAGA CELMIRA MARCELA
INGENIERA DE SISTEMAS

MACHALA
2020

UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

TRABAJO TITULACIÓN
PROPUESTAS TECNOLÓGICAS

ANALISIS Y DISEÑO DE UNA ARQUITECTURA DE RED UTILIZANDO LAS HERRAMIENTAS HONEYPOT, IDS Y FIREWALL

PORRAS SURIAGA CELMIRA MARCELA
INGENIERA DE SISTEMAS

VALAREZO PARDO MILTON RAFAEL

MACHALA, 18 DE DICIEMBRE DE 2020

MACHALA
2020

ANALISIS Y DISEÑO DE UNA ARQUITECTURA DE RED UTILIZANDO LAS HERRAMIENTAS HONEYPOT, IDS Y FIREWALL

INFORME DE ORIGINALIDAD

7%	7%	2%	3%
INDICE DE SIMILITUD	FUENTES DE INTERNET	PUBLICACIONES	TRABAJOS DEL ESTUDIANTE

FUENTES PRIMARIAS

1	docplayer.es Fuente de Internet	2%
2	btob.com.mx Fuente de Internet	<1%
3	www.buscadoc.org Fuente de Internet	<1%
4	cgi.insecure.org Fuente de Internet	<1%
5	Submitted to Universidad Internacional de la Rioja Trabajo del estudiante	<1%
6	www.anomali.com Fuente de Internet	<1%
7	www.slideshare.net Fuente de Internet	<1%
8	Submitted to Universidad Nacional Abierta y a	

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

La que suscribe, PORRAS SURIAGA CELMIRA MARCELA, en calidad de autora del siguiente trabajo escrito titulado ANALISIS Y DISEÑO DE UNA ARQUITECTURA DE RED UTILIZANDO LAS HERRAMIENTAS HONEYPOT, IDS Y FIREWALL, 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

Marcela Porras

PORRAS SURIAGA CELMIRA MARCELA
0704482694

DEDICATORIA

El presente trabajo de titulación es una representación de todos los años de estudio en la carrera de Ingeniería de Sistemas, por ello quiero dedicárselo a mi abuelito materno; mi Papi Carlos, quien fue para mí un padre y aunque hoy no esté físicamente para celebrar conmigo este triunfo me acompañó toda mi vida y fue siempre ejemplo y fortaleza para no decaer en mis intentos de superación.

A mis padres por ser mi sustento, por confiar en mis sueños, por motivarme a superar mis propios límites. A mis hermanas por enseñarme a ser mejor persona e inspirarme día a día y ver en mí su ejemplo de vida.

Dedico este trabajo a ellos y a todas las personas que estuvieron pendientes de mi proceso de estudio, que confiaron en mis capacidades y auguraron éxitos para mi vida.

AGRADECIMIENTO

Cuando de agradecimientos se trata el primero es para Dios, por ser ese ser de luz, bendecir mi vida y hoy permitirme ver uno de mis anhelos cristalizados.

A mis amados padres, por ser mi sustento económico y moral, de manera especial a mi madre por ser mi amiga y guía durante mi etapa universitaria, por formar mi carácter y motivarme siempre.

A mis abuelitos maternos por apoyarme siempre, aunque hoy uno de ellos ya no esté conmigo, me consuela saber que le hice saber todo lo agradecida que estaba con él por todo y que hoy celebra conmigo este triunfo que tanto anhelaba.

A mi novio por ser mi complemento y acompañarme durante todos estos años, por impulsarme a continuar y no dejar de intentarlo.

A mis hermanas, tíos, primos y demás familiares por estar siempre ahí para mí, aplaudir mis triunfos y ayudarme a levantarme de mis derrotas, han sido seres excepcionales. A mis amigos, compañeros y conocidos, por estar pendientes, apoyarme e incentivarme a alcanzar mi título profesional.

A la Universidad Técnica de Machala y sus autoridades por dirigir y gestionar recursos educativos, permitiéndonos contar con instalaciones y recursos de primer orden haciendo que los conocimientos adquiridos sean los mejores. A los docentes de la carrera de Ingeniería de Sistemas por todas sus enseñanzas, paciencia y apoyo brindado, de manera particular al Ing. Milton Valarezo, quien con sus ideas y conocimientos supo ser guía para la realización del presente trabajo de titulación.

RESUMEN

En la actualidad todas las instituciones cuentan con tecnologías para procesar y comunicar información con propósitos claves como ofertar sus productos o servicios a nuevos y potenciales clientes, accediendo a recursos de internet tales como redes sociales, entre otros; sin embargo, para cumplir con estos propósitos exponen su información y no toman las precauciones necesarias de seguridad de red.

Las formas de ataques de los ciberdelincuentes van en aumento, lo que ha despertado la necesidad de las empresas por implementar y/o reforzar las políticas de privacidad con la que ejecutan sus procesos basando los mismos en características y requisitos de las redes actuales para garantizar la efectividad de los mismos, es por ello que se debe pensar en soluciones dinámicas que proporcionen funcionalidad para evitar y/o prevenir los ataques dando una respuesta acertada con protección perimetral de las redes y puntos externos.

En base a estos indicios, se realizo una investigación minuciosa en base a los conceptos actualizados de la seguridad perimetral de redes de computadoras y de los elementos más importantes que intervienen en este tipo de arquitecturas informáticas, es por esta razón que la arquitectura planteada se fundamenta en estos sistemas ya probados y comprobados de seguridad perimetral, de modo que cuando un usuario malicioso pretende acceder a la red sea detectada.

El desarrollo de esta arquitectura se basa en un cortafuegos y honeypot, que esta representado por la herramienta Kippo que se alojan en el sistema operativo Linux y un IDS representado por la herramienta Snort en el sistema operativo Windows para que sea una red segura, basados en una planificación previa:

Identificando la arquitectura que permitirá la protección de acceso a una red, identificando las herramientas con las que se puede cumplir dicho propósito, estudiando la herramienta de contrafuegos; su funcionamiento y los tipos que existen del mismo, además de estudiar la herramienta honeypot e IDS para detectar las amenazas y cómo funciona el mismo, dando paso a una fase de implementación donde se aplicará cada una de las herramientas antes

mencionadas para lograr un correcto funcionamiento mediante las pruebas de penetrating para evaluar el nivel de seguridad de la red empresarial, es decir verificar cual es el comportamiento de los mecanismos de defensa y detectar vulnerabilidades en los mismos.

El presente trabajo tiene como finalidad desarrollar e implementar una arquitectura de seguridad de red que permita tener acceso determinada red de un modo seguro, de modo que se evidencia la deficiente seguridad que se presenta en la misma, se pretende que el trabajo presentado sirva como referencia para que en lo posterior se manejen un conjunto de estrategias operativas para el desarrollo y puesta en marcha de un modelo general de seguridad y privacidad de la información, que obligue a la adopción de nuevas políticas de seguridad que respondan de forma efectiva a posibles incidencias de seguridad, en respuesta efectiva a un mal funcionamiento de los elementos de la red, donde se permita la detección de vulnerabilidades y ataques, plasmando cambios dinámicos en los procesos.

Palabras Claves: Arquitectura, Red perimetral, Seguridad, Herramientas, Ataques, Vulnerabilidades.

ABSTRACT

Currently all institutions have technologies to process and communicate information for key purposes such as offering their products or services to new and potential clients, accessing Internet resources such as social networks, among others; however, to fulfill these purposes they expose your information and do not take the necessary network security precautions.

The forms of attacks by cybercriminals are on the rise, which has awakened the need for companies to implement and / or enforce the privacy policies with which they execute their processes, basing them on characteristics and requirements of current networks to guarantee the effectiveness of the same, that is why you should think about dynamic solutions that provide functionality to avoid and / or prevent attacks by giving a successful response with perimeter protection of networks and external points.

Based on these indications, a meticulous investigation was carried out based on the updated concepts of perimeter security of computer networks and the most important elements that intervene in this type of computer architecture, it is for this reason that the proposed architecture is based in these already tested and proven perimeter security systems, so that when a malicious user tries to access the network, it is detected.

The development of this architecture is based on a firewall and honeypot, which is represented by the Kippo tool that is hosted on the Linux operating system and an IDS represented by the Snort tool on the Windows operating system to make it a secure network, based on advance planning:

Identifying the architecture that will allow the protection of access to a network, identifying the tools with which this purpose can be fulfilled, studying the firewall tool; its operation and the types that exist, in addition to studying the honeypot and IDS tool to detect threats and how it works, giving way to an implementation phase where each of the aforementioned tools will be applied to achieve correct operation through penetrating tests to evaluate the security level of the business network, that is, to verify the behavior of the defense mechanisms and detect

vulnerabilities in them. The purpose of this work is to develop and implement a network security architecture that allows access to a certain network in a safe way, so that the poor security that is presented in it is evidenced, it is intended that the work presented serves as a reference so that later on, a set of operational strategies are managed for the development and implementation of a general information security and privacy model, which forces the adoption of new security policies that respond effectively to possible incidents of security, in an effective response to a malfunction of network elements, where vulnerabilities and attacks are detected, reflecting dynamic changes in processes.

Keywords: Architecture, Perimeter Network, Security, Tools, Attacks, Vulnerabilities.

CONTENIDO

ÍNDICE DE ILUSTRACIONES

INTRODUCCIÓN

En la actualidad, los avances tecnológicos han evidenciado el crecimiento de redes y cantidad de información disponible, a medida que pasa el tiempo los usuarios de las redes van adquiriendo experiencia y el acceso a la información de las vulnerabilidades se encuentra fácilmente, de modo que cada vez llega a más personas, lo que hace que las redes sean más inseguras y por ende demasiado vulnerables a ataques y/o robos de información, ocasionando graves problemas hasta dejar sin servicio a la red.

Se considera insegura a una red, cuando los datos de la misma se encuentran en riesgo de ser interceptados, alterados o robados, dando paso a transferencias de archivos, daños en el sistema operativo o red, etc. En el ámbito empresarial; el aumento de dependencia de un ambiente interconectado de dispositivos en línea, ha elevado en gran medida la dependencia de seguridad de las redes para impedir ataques cibernéticos.

La siguiente propuesta tecnológica tiene como propósito implementar una arquitectura de red tradicional de seguridad perimetral, que permitirá establecer mecanismos de barrera entre la red interna y externa para evitar ataques.

Este documento se encuentra estructurado de la siguiente manera:

Capítulo 1: se describe la necesidad de implementar la plataforma, el análisis previo, sus requerimientos y justificación, destacando la importancia de implementar este tipo de arquitecturas de red.

Capítulo 2: en este se detalla la planificación del prototipo, la fundamentación teórica, objetivos, diseño y ejecución del prototipo.

Capítulo 3: se realizan las pruebas necesarias y se evidencian los resultados y detalle de los mismos, además de las conclusiones y recomendaciones para trabajos futuros.

1. CAPITULO I: DIAGNOSTICO DE NECESIDADES Y REQUERIMIENTOS

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

Hablar de redes es evidenciar los avances tecnológicos que han surgido con el pasar de los años y la influencia de los mismos en el desarrollo de las empresas, es por ello que se destaca que las redes son mas vulnerables e inseguras dando paso a intercepciones, ataques o robos, del mismo modo que son propensas a trasmitir información que dañe la red o el sistema operativo con virus.

Una arquitectura de red, basada en este contexto hace referencia a las tecnologías que admiten la infraestructura, además de servicios y protocolos programados para transferir la información en toda la infraestructura [1]. Cabe mencionar que el Internet evoluciona, al igual que las redes en general, se destacan características básicas que la arquitectura subyacente necesita para cumplir con las expectativas de los usuarios: tolerancia a fallas, escalabilidad, calidad del servicio y seguridad [2].

Con la siguiente propuesta tecnológica se espera reducir riesgos a pequeña y gran escala de sufrir ataques informáticos que ocasionen el acceso de usuarios no autorizados y por ende pérdida de información importante, de modo que este tipo de situaciones en una empresa significarán grandes pérdidas.

1.2. Establecimiento de Requerimientos

En la actualidad, los ataques cibernéticos y amenazas desconocidas requieren soluciones de inteligencia y seguridad estratégica, con el pasar del tiempo los ataques se han vuelto más sofisticados y evolucionan a diario, por ello se debe buscar soluciones dinámicas y óptimas que incluyen análisis de seguridad, respuesta a incidentes e inteligencia de amenazas con protección perimetral de redes y puntos externos.

Se debe establecer mecanismos de control a la gran cantidad de seguridad de flujos no detectables, de aparatos o equipos que permitan penetrar los recursos sin autorización traspasando os límites de seguridad perimetral definidos por la arquitectura a proponer, por ello es necesario que se realicen revisiones periódicas, ajustes y cambios en el sistema para mantener al mismo actualizado.

1.3. Justificación del requerimiento a satisfacer

Las pequeñas o grandes industrias hoy en día, deben tener como prioridad administrar los riesgos de la exposición de sus datos para evitar ataques maliciosos a sus redes, para garantizar y mejorar la continuidad del mismo y por ende la reducción de los costos de operaciones de seguridad de las redes e información.

Al hablar de seguridad perimetral, se hace referencia a la forma de establecer una barrera o frontera lo más inaccesible posible entre las redes, siendo su objetivo restringir y/ o controlar los datos que entran en la organización. La principal ventaja de este tipo de seguridad es permitir al administrador enfocarse en las entradas, sin dejar de laso a los servidores internos de la red.

2. CAPÍTULO II: DESARROLLO DEL PROTOTIPO

2.1. Definición del Prototipo Tecnológico

La siguiente propuesta tecnológica tiene como propósito el diseño e implementación de un modelo de red tradicional con el uso de una herramienta de detección de intrusos, en este caso se destaca snort, por otro lado ipfire es un firewall que controla el accedo a la red y kippo un honeypoot que cumple la función de distractor ante posibles atacantes y un IDS que detecta las amenazas, de modo que al realizar las pruebas de penetrating se pueda evaluar cual es el nivel de seguridad presente en este sistema de red.

2.2. Fundamentación Teórica del Prototipo

2.2.1. ARQUITECTURA DE RED TRADICIONAL DE SEGURIDAD PERIMETRAL

2.2.1.1. ¿Qué es la red tradicional de seguridad perimetral?

Una arquitectura tradicional de seguridad perimetral, es aquella en la cual se establece una segmentación mediante mecanismos de barrera situados en el perímetro que separa la red interna y externa, o bien entre un segmento específico de la red y su exterior; con lo cual permite una oportuna detección de intrusos y el bloqueo de ataques que pongan en riesgo los datos, hardware o software de una organización.

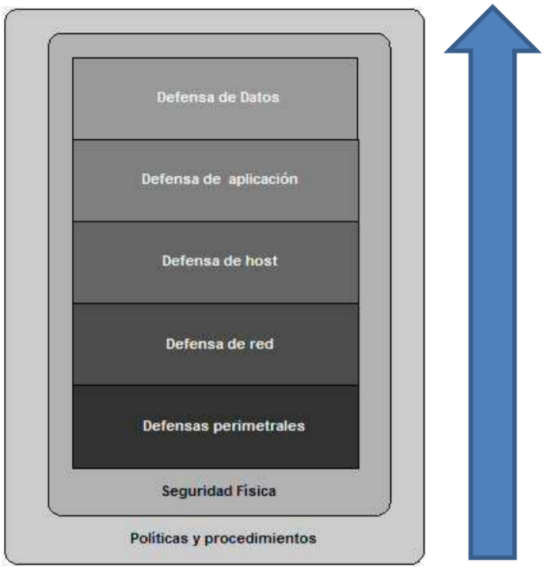


Ilustración 1. Implantación de un sistema de seguridad perimetral [3]

El creciente interés en la computación perimetral está perfilando de forma gradual pero inexorable nuevos modelos arquitectónicos y de uso, que se distinguen por la dispersión geográfica y la heterogeneidad de los dispositivos [4]. La seguridad perimetral típica comienza con una cerca y una puerta y puede incluir múltiples métodos de seguridad (por ejemplo, acceso con tarjeta, cerraduras, sensores, iluminación, CCTV y patrullas) para aumentar la protección [5].

Una de las soluciones que se pueden implementar para proteger el perímetro de la red es el firewall, el cual se considera “la primera línea de defensa de la red de la organización que se espera controle tanto la entrada como la salida del tráfico de la red [6]”.

Además del firewall, una arquitectura de red de seguridad perimetral puede incluir otros dispositivos registradores de eventos, tales como IDS red [6].

“Hay una serie de tecnologías diferentes para hacer frente a los ciberataques, como los sistemas de detección de intrusiones (IDS), los sistemas de prevención de intrusiones (IPS), firewalls,

conmutadores, enrutadores, etc., que están activos las 24 horas [7]". En la figura 1, se muestra una arquitectura de red que incorpora IDS y un Firewall.

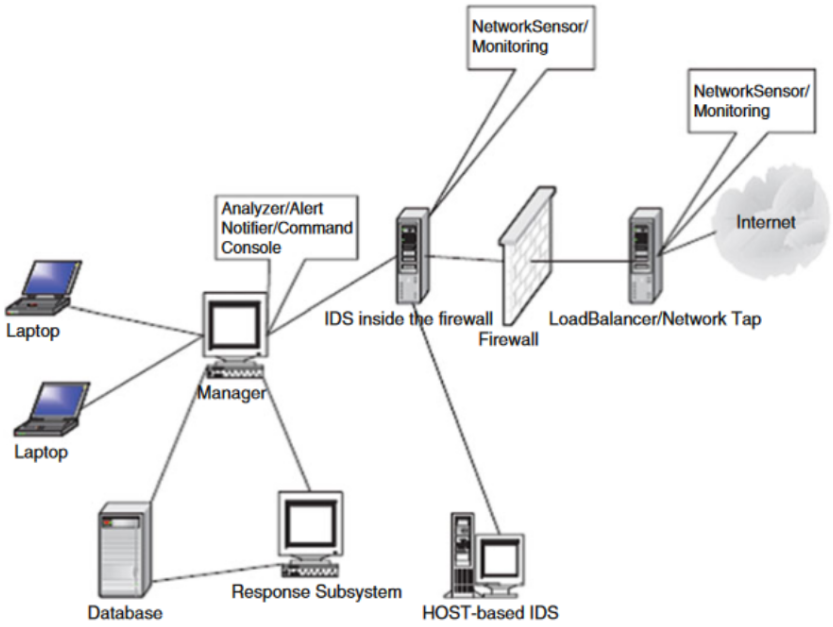


Ilustración 2. Arquitectura de red con seguridad perimetral [6].

2.2.1.2. Importancia de la seguridad perimetral

La importancia de la seguridad perimetral radica en que la información es uno de los activos más importantes para cualquier organización, la cual debe protegerse garantizando la privacidad de las personas y de la organización misma. Debido a la amplia variedad de dispositivos utilizados en los sistemas de redes informáticas, la seguridad juega un papel importante en la protección y mejora del rendimiento de la red o sistema. Aunque esta temática ha recibido un gran interés mundial en los últimos años, sigue siendo un espacio de investigación abierto [8].

El perímetro de la red tradicional ha cambiado con el pasar del tiempo, más sin embargo "sigue siendo la piedra angular de la defensa del sistema cibernético" [6]. Por tanto, se asume que todas aquellas

cosas que se deben proteger estarán encerradas dentro del perímetro de la red.

El perímetro, por lo tanto, separa la "mala Internet" exterior de la red protegida. Los cortafuegos se han construido para cumplir con este propósito. Sin embargo, todavía la seguridad perfecta dentro de las redes protegidas es una utopía, pero el ideal es lograr diseñar una defensa perimetral a prueba de penetraciones [6].

La seguridad de la red generalmente se refleja en los datos relevantes generados, originados o extraídos del sistema de red. Al estudiar los datos relacionados con los eventos de seguridad de la red, se puede cuantificar y medir la seguridad del sistema de red [9].

2.2.2. FIREWALL

2.2.2.1. ¿Qué es Firewall?

"Un firewall es un hardware, un software, o una combinación de ambos que monitorea y filtra los paquetes de tráfico que intente entrar o salir de la red privada protegida [6]."

"Un firewall es un dispositivo importante de seguridad de la red, y el núcleo es una política de cortafuegos, que consiste en un conjunto de reglas para administrar y controlar el acceso a los recursos de la red [10]." Un firewall está configurado para mantener un conjunto de reglas orientadas a preservar la integridad de la red [11].

De acuerdo a la revisión bibliográfica se puede resumir que hay un consenso respecto a considerar que el firewall es la primera línea de defensa contra los ataques cibernéticos [6], [12], [11].

2.2.2.2. ¿Para qué sirve?

"Los firewalls se utilizan para proteger las redes esenciales de los ataques al aire libre para guiar el acceso a la red según las reglas de acceso del firewall [13]."

Un firewall es una herramienta fundamental que separa una red o subred protegida, de una red desprotegida, como Internet [6].

Si bien los firewalls están asociados con el aislamiento de una red interna de una red externa, con frecuencia Internet, también se debe considerar para aislar segmentos dentro de una red interna. Si un adversario ingresa a una red, utilizará sistemas confiables para comprometer otras partes de la red. Un Firewall, también evita que personas internas malintencionadas exploten sistemas fuera de su acceso inmediato [14].

2.2.2.3. ¿Cómo funciona?

"Un firewall funciona como un escudo en la seguridad de redes corporativas [11]", en donde es de vital importancia proteger la información confidencial.

"La función de un firewall es examinar cada paquete que pasa a través de él y decidir si desea dejarlos pasar o detenerlos según reglas preconfiguradas y políticas [15]."

De manera simplificada se puede decir que, los firewalls son dispositivos de seguridad perimetral que limitan las conexiones para la entrada y salida de datos. Principalmente basados en los servicios de red, los firewalls pueden resultar muy útiles para aliviar la carga de seguridad de los sistemas individuales [14].

2.2.2.4. Tipos de Firewall

Los firewalls pueden clasificarse de acuerdo a la función en la cual se especializan, de esta forma tenemos los siguientes cinco tipos que se detallan a continuación:

Firewall de filtrado de paquetes: También llamados cortafuegos de capa de red o cortafuegos sin estado. Un firewall sin estado trata cada paquete o trama de red individualmente. La técnica utilizada en

este tipo de firewalls mira cada paquete que entra o sale de una red, aceptando o rechazando en función de reglas establecidas, si un paquete coincide con el conjunto de reglas, el paquete puede reenviarse a su destino o descartarse [15].

Circuit-level gateways: Supervisa el protocolo de enlace de TCP entre los hosts locales y remotos para determinar si la sesión que se inicia es legítima o si efectivamente el sistema remoto se considera "confiable". Sin embargo, no inspeccionan los paquetes ellos mismos. Aplica mecanismos de seguridad cuando se establece una conexión TCP o UDP. Una vez que se ha realizado la conexión, los paquetes pueden fluir entre los hosts sin más verificacion [15].

Filtros con estado: Mantiene registros de todas las conexiones que lo atraviesan y puede determinar si un paquete es el inicio de una nueva conexión, una parte de una conexión existente o es un paquete inválido. Para hacer esto, el firewall mantiene una entrada, en una caché, para cada flujo abierto. Cuando el primer paquete de un nuevo flujo es visto por el firewall (este es el llamado paquete SYN en un flujo TCP que significa sincronizar, y ACK que es el reconocimiento de que ya se ha establecido una conexión entre hosts, etc.), el firewall lo compara con la base de reglas [15].

Firewall de capa de aplicación: Este tipo de firewall también se denomina como Aplicación firewall proxy; y es un sistema de seguridad de red que protege los recursos de red mediante el filtrado de mensajes en la capa de aplicación [15].

El proxy de la aplicación comprende el protocolo y los datos de la aplicación, e intercepta cualquier información destinada a esa aplicación, considerando la base de la cantidad de información disponible para tomar decisiones; de esta forma El puede autenticar a los usuarios y juzgar si los datos podría representar una amenaza [15].

Firewall de inspección multicapa: Los cortafuegos multicapa funcionan conservando el estado (estado) asignado a un paquete por cada componente del cortafuegos a través del cual pasa en el camino hacia la pila de protocolos. Esto da el control máximo del usuario sobre qué paquetes pueden llegar a su destino final, pero nuevamente afecta el rendimiento de la red, aunque generalmente no tan dramáticamente como lo hacen los proxies [15].

2.2.3. HONEYPOT

2.2.3.1. ¿Qué es un Honeypot?

"Un honeypot es un sistema que permite observar las acciones del atacante en diferentes fases de un ciberataque [16]."

Un Honeypot es un concepto avanzado de seguridad de redes [17], que consiste en un sistema diseñado para parecerse a algo que un intruso puede intentar piratear, están construidos para muchos propósitos, pero el principal es engañar a los atacantes y aprender acerca de sus herramientas y métodos [6].

"Los Honeypots son recursos configurados para atrapar a los atacantes mediante la ejecución de servicios que tienen vulnerabilidades y luego observar sus actividades en un entorno controlado [18]." Al implementar honeypots, es obligatorio considerar los riesgos que pueden imponer a la red y los sistemas [18], dado que el objetivo de un honeypot es engañar a los intrusos y aprender de ellos sin comprometer la seguridad de la red [6].

2.2.3.2. ¿Dónde ubicar los Honeypots?

La mejor ubicación para para un honeypot es en la DMZ o detrás del firewall si la red privada no tiene un DMZ [6].

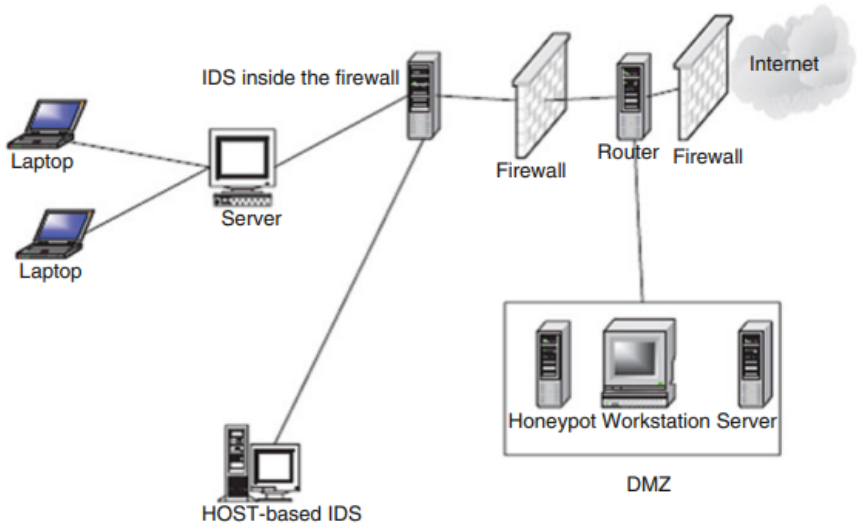


Ilustración 3. Posicionamiento de un honeypot [6].

2.2.3.3. Clasificación de los Honeypots

Los honeypots se clasifican mediante aspectos como nivel de interacción, entorno de implementación, tipo de recurso, servicios, adaptabilidad e implementación.

El nivel de interacción: Va desde honeypots de baja interacción, que emulan solo la pila de comunicación, hasta honeypots de alta interacción, que ejecutan un sistema operativo real [17].

Entorno de implementación: Hay dos casos de uso generales. El primero es el despliegue como honeypots de investigación, se usa ampliamente como sensor para actualizaciones de bases de datos antivirus [17], estos honeypots son complejos de implementar y mantener [19]. La segunda implementación se conoce como honeypots de producción, la cual se realiza detrás del perímetro dentro de una red empresarial o industrial. En esta implementación, cualquier interacción con el honeypot indica una brecha de seguridad [17].

Tipo de recurso: Se establecen varios conceptos de recursos engañosos. Los honeypots del lado del servidor son sistemas con servicios en modo de escucha que esperan conexiones entrantes.

Los honeypots del lado del cliente se están conectando activamente a sistemas potencialmente peligrosos para investigar sus intentos de intrusión [17].

Servicios: Existe un conjunto de servicios que utiliza el sistema honeypot. Para dificultar la identificación de sistemas honeypot, el conjunto de servicios debe elegirse con cuidado, con respecto a los servicios esperados en el entorno desplegado [17].

Adaptabilidad: Históricamente, los honeypots son estáticos, lo que significa que la configuración, la implementación y el mantenimiento son tareas manuales. Los honeypots que presentan adaptabilidad se denominan honeypots dinámicos [17].

Implementación: Los honeypots que emplean hardware dedicado se clasifican como honeypots reales. Por otro lado, los honeypots de hardware compartidos se denominan honeypots virtuales [17].

2.2.3.4. Kippo

2.2.3.4.1. ¿Qué es Kippo?

Kippo es un honeypot de interacción media construido para estudiar los ataques SSH. Tiene la capacidad de registrar todos los intentos de nombre de usuario y contraseña de ataques de fuerza bruta y de diccionario [18].

Kippo permite que una entidad atacante intente iniciar sesión en el sistema, creyendo que está entrando en una sesión SSH legítima con el servidor. En donde el atacante intenta adivinar la contraseña y, una vez que la adivina correctamente, el atacante se traslada a un sistema falso con el que puede interactuar [20].

2.2.3.4.2. ¿Cómo ejecutarlo?

Kippo SSH honeypot se coloca antes de cualquier sistema administrativo para que el atacante inicie sesión en él asumiendo que

es un sistema legítimo. En este sistema falso, todas las interacciones con el shell son monitoreadas y registradas. El sistema también permite el uso de wget y otros comandos comúnmente usados para buscar o descargar archivos. El objetivo principal de la implementación es darle al atacante la impresión de estar navegando por el sistema real de la organización [20].

Después de un inicio de sesión exitoso en el servidor SSH, registra todas las interacciones de shell realizadas por los atacantes. En una sesión SSH típica, el cliente primero establece una conexión TCP con el servidor SSH y luego intercambia información de autenticación. Después de la etapa de autenticación de la negociación de algoritmos de seguridad, el cliente envía una solicitud de inicio de sesión SSH. El servidor SSH verificará la combinación de nombre de usuario y contraseña para decidir si el cliente está autorizado o no. Cuando llegamos a Kippo SSH honeypot, todos los pasos anteriores son los mismos, excepto que el cliente es ahora el atacante. Aquí los nombres de usuario y las contraseñas ingresadas por los atacantes se comparan con las listas de nombres de usuario y contraseñas preconfiguradas que han sido almacenadas en el archivo userdb. Cuando los atacantes adivinan correctamente el nombre de usuario y la contraseña, se les permite iniciar sesión y ejecutar algunos comandos en el servidor de Kippo honeypot [18].

Kippo honeypot permite ejecutar algunos comandos como ls y wget. Dado que el honeypot no se da cuenta de todos los comandos reales de Linux, los atacantes pueden averiguar fácilmente si están dentro de un honeypot o en un sistema real [18].

2.2.4. Sistema de Detección de Intrusos

2.2.4.1. ¿Qué es un IDS?

"Un sistema de detección de intrusos (IDS) es un sistema que se utiliza para detectar intrusiones en sistemas y redes informáticos [6]."

Un IDS es particularmente útil para dar el elemento de identificación de ataques, perturbaciones de seguridad y en la documentación de advertencias de intrusión para la organización [21].

Los escenarios más utilizados de recopilación de datos relacionados con la seguridad de la red son IDS y otros sistemas de seguridad de red o dispositivos de seguridad que detectan ataques e intrusiones en la red. El módulo de recopilación de datos de un IDS es responsable de monitorear el estado del host, los datos de la red y el comportamiento del usuario. Los datos de red aquí incluyen los parámetros de las actividades de la red, el número de conexiones de red, el número de paquetes, el contenido de los paquetes, etc [9].

Una de las aplicaciones más importantes de un sistema de detección de intrusos (IDS) es resolver problemas de seguridad en una Red Definida por Software (SDN) [22], pero pueden no ser adecuados para subestaciones digitales, porque estas tienen componentes críticos con estrictos requisitos de tiempo y se ha podido comprobar que, aunque algunos ataques están cubiertos por los IDS implementados actualmente, es necesario avanzar más para hacer frente a los ataques enmascarados [23].

Un IDS estándar puede no ser muy efectivo o incluso inadecuado para los requisitos de una organización o un individuo [24], por lo cual existen gran cantidad de estudios académicos respecto a la optimización de los IDS, incluyendo la aplicación de técnicas de aprendizaje supervisado [25] o machine learning [26] [27], redes neuronales convolucionales [28], máquina de aprendizaje extrema

[29], entre muchos otros estudios orientados a mejorar la eficiencia de los IDS.

2.2.4.2. Tipos de Sistemas de Detección de intrusos

Algunos sistemas de detección de intrusiones (IDS) basados en aprendizaje automático monitorean el tráfico y el flujo de datos en el sistema operativo (sistema de detección de intrusiones en el host - HIDS) o en el nivel de la red (sistema de detección de intrusiones en la red - NIDS) para detectar ataques dirigidos al host o la red [30].

Como una de las tecnologías más confiables, el sistema de detección de intrusiones en la red (NIDS) permite el monitoreo del tráfico entrante y saliente para identificar el uso no autorizado y el mal manejo de los atacantes en los sistemas de redes informáticas [31].

2.2.4.3. ¿Cómo detectan tráfico malicioso?

Los IDS están equipados con varias partes, en particular, los clasificadores neuro-difusos se utilizan para organizar la información de tráfico del sistema en el tipo de información típica e información entrometida y producir actividad a partir de un estado de información [21].

2.2.4.4. IDS más conocidos

2.2.4.5. Snort

2.2.4.5.1. ¿Qué es Snort?

“Snort es un sistema de detección de intrusiones de red en tiempo real basado en software desarrollado por Martin Roesch. Es un buen IDS que se puede utilizar para notificar a un administrador de un posible intento de intrusión [6].”

SNORT como sistema de detección de intrusos detecta ataques DoS y DDoS [32]. Al ser un IDS de código abierto, Snort se puede configurar e

implementar fácilmente en cualquier entorno [33]. De acuerdo con una comparación de rendimiento de sistemas de detección de intrusos, Snort cuenta con una buena precisión de detección, pero tiene un alto recursos informáticos [34].

Permite el análisis de paquetes a nivel de carga útil para determinar la causa de la alerta, el motivo de la alerta y si se deben tomar medidas. Estas características de Snort (flexibilidad, fácil configuración, análisis de paquetes sin procesar) lo convierten en un potente dispositivo de detección de intrusos [35].

2.2.4.5.2. Reglas Snort

Snort es altamente configurable, lo que permite a los usuarios, después de la instalación, crear sus propias reglas, a partir de un lenguaje flexible basado en reglas para describir el tráfico para recopilar o transmitir, utilizando un motor de detección modular y reconfigurar su funcionalidad básica utilizando su interfaz de complemento [6]. Las reglas requeridas son fáciles de escribir, flexibles y se pueden ingresar fácilmente en la base de datos de reglas. En caso de que se encuentre un nuevo ataque o explotación, se puede agregar una nueva regla a la base de datos para el mismo en poco tiempo [35].

Las reglas de Snort se pueden escribir en cualquier idioma, su estructura también es buena, se puede leer fácilmente y las reglas también se pueden modificar. Siempre que un paquete ingresa a la red, Snort comprueba el comportamiento de la red si el rendimiento de la red se degrada, luego Snort detiene el procesamiento del paquete, descarta el paquete y almacena su detalle en la base de datos de firmas [36].

Sin embargo, los métodos basados en reglas tienen la desventaja de una alta complejidad de reglas, debido a que pueden entrar en conflicto entre sí. Como resultado, se han hecho muchos esfuerzos

para resolver los conflictos, como disparar las reglas por orden de preeminencia, disparar las reglas por orden de pila o cola, incluso disparar las reglas al azar [9].

2.3. Objetivos del Prototipo

Objetivo Principal:

Implementar una arquitectura de seguridad de red que permita proteger el acceso a la red de una organización, utilizando la arquitectura de red tradicional de seguridad perimetral.

Objetivos Específicos:

- Realizar un firewall en el Sistema Operativo Linux para que ayude a controlar el tráfico de red que entra y sale entre una red interna confiable y una externa no confiable (Internet).
- Crear un IDS (Sistema de Detección de Intrusos) para detectar alguna actividad maliciosa.
- Implementar un honeypot que ayudará a detectar, desviar o neutralizar los intentos de uso no autorizados.
- Realizar pruebas de penetrating para la verificación del comportamiento de los mecanismos de defensa, detectando de esta manera las vulnerabilidades en los mismos.

2.4. Diseño del Prototipo

El diseño de la arquitectura está desarrollada de la siguiente manera, como red interna el servidor de Oracle y el host, delante de los mismo están ubicados el firewall como segundo en la línea de defensa, el honeypoot para atraer y detectar al intruso que ingresa a la red sin autorización y el IDS para detectar intrusos y el otro host es en el que se realizan las pruebas de virtualización.

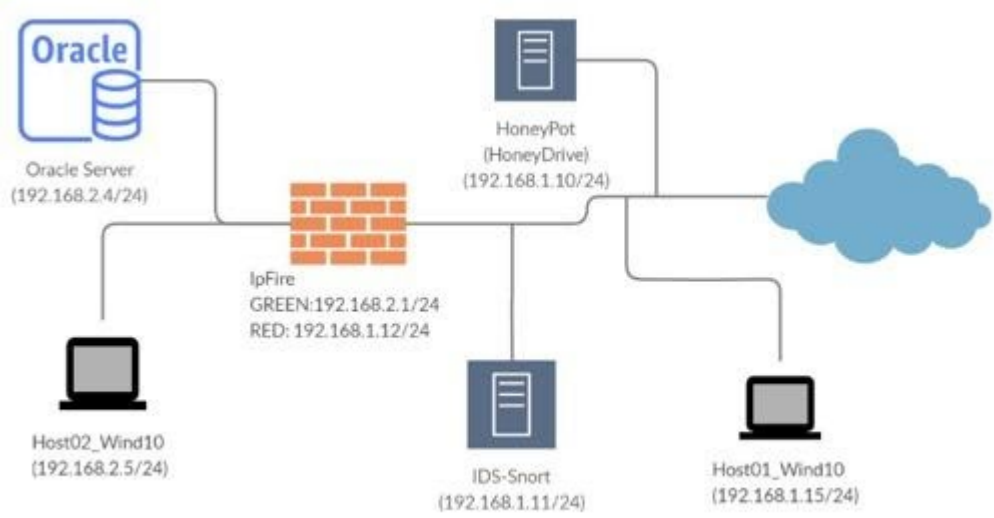


Ilustración 4: Diseño del Prototipo

2.5. Ejecución y/o Ensamblaje del Prototipo

En esta sección se describen los pasos a seguir para la configuración de la arquitectura de red, destacando los aspectos mas relevantes de cada una de las máquinas virtuales.

2.5.1. Instalación de GNS3

- Descargar el archivo desde su página oficial http://www.gns3.com/
- Ejecutar el instalador y dar clic en Next.

Ilustración 5: Instalación de GNS3

- Escoger las herramientas de las opciones a instalar.

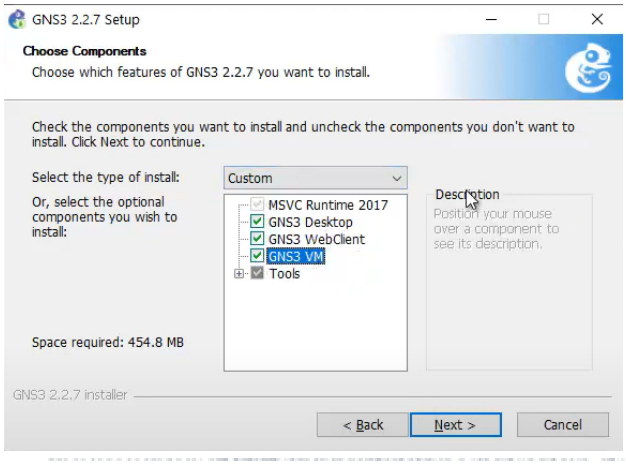


Ilustración 6: Instalación de GNS3

- De manera inmediata inicia el proceso de instalación de GNS3

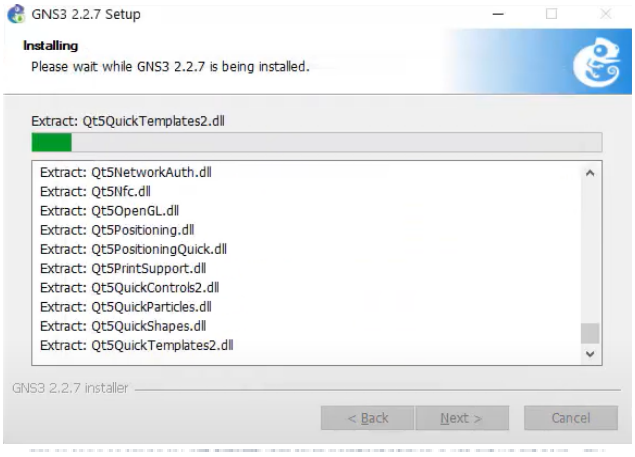


Ilustración 7: Instalación de GNS3

- Automáticamente también empezará la descarga de Wireshark, el cual es un analizador de protocolos utilizado para realizar análisis y solucionar problemas en redes de comunicaciones

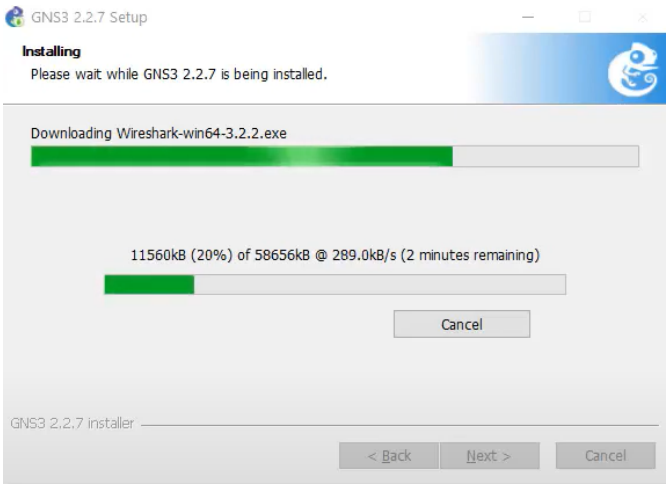


Ilustración 8: Instalación de GNS3

- Después de unos minutos finalmente la instalación de GNS3 habrá terminado

Ilustración 9: Instalación de GNS3

2.5.2. Importar una máquina virtual de Virtual Box

- Una vez iniciado GNS3 nos dirigimos a la opción de preferencias en el menú editar.

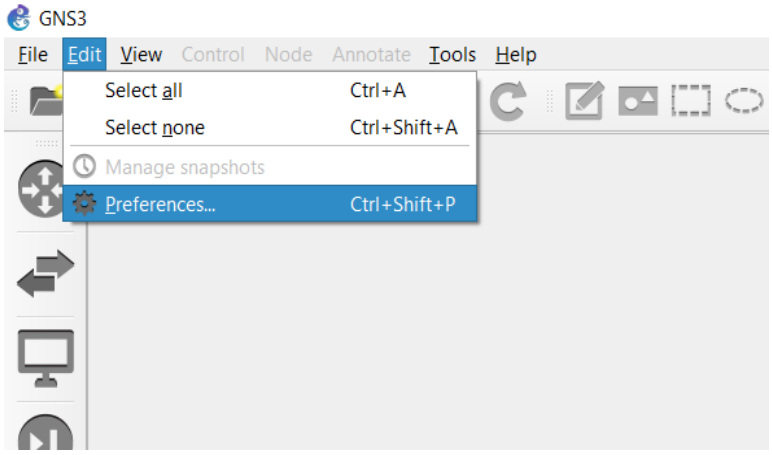


Ilustración 10: Importación de Máquina Virtual

- Dentro del menú, hay varias opciones en los ajustes, nos dirigimos a la opción VirtualBox VMs y desde ahí se busca la máquina virtual.

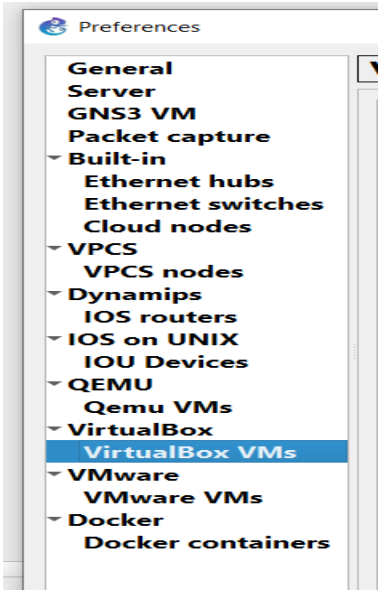


Ilustración 11: Importación de Máquina Virtual

- En la ventana escogemos la opción New o Nuevo

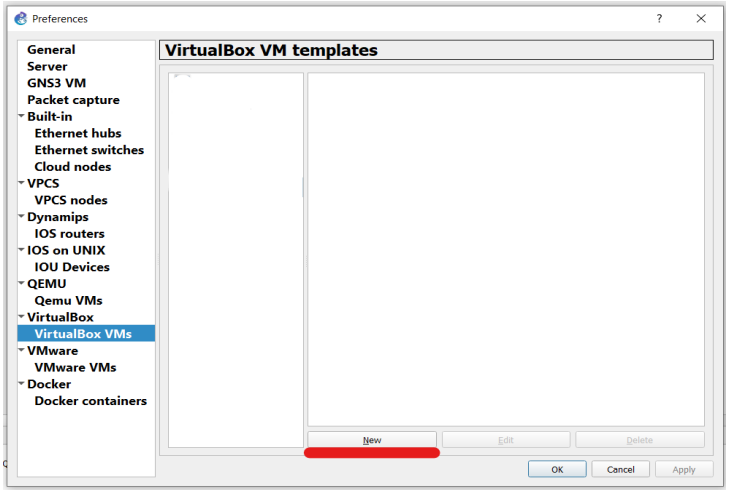


Ilustración 12:Importación de Máquina Virtual

- Se desplegará una nueva ventana con todas las máquinas virtuales que tengamos instaladas en VirtualBox y escogeremos una por una las que deseemos importar y escogemos Finalizar.

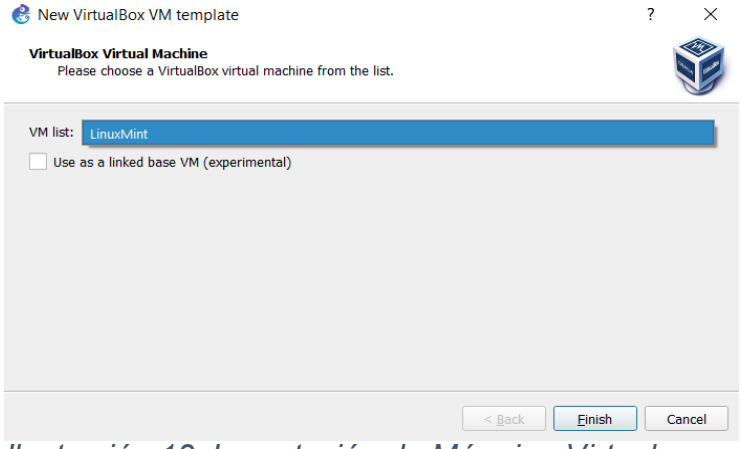


Ilustración 13: Importación de Máquina Virtual

- Luego de este proceso, se tiene todas las maquinas necesarias para montar nuestra topología en GNS3. En esta misma ventana está la opción de editar las configuraciones de una maquina o eliminarla.

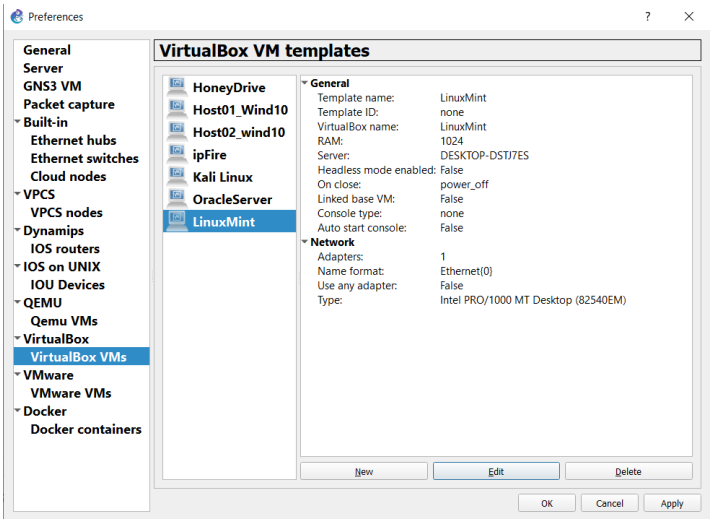


Ilustración 14: Importación de Máquina Virtual

- Luego de realizar esa importación, se tiene las maquinas en la opción de dispositivos finales y procederemos a montar nuestra topología de red.

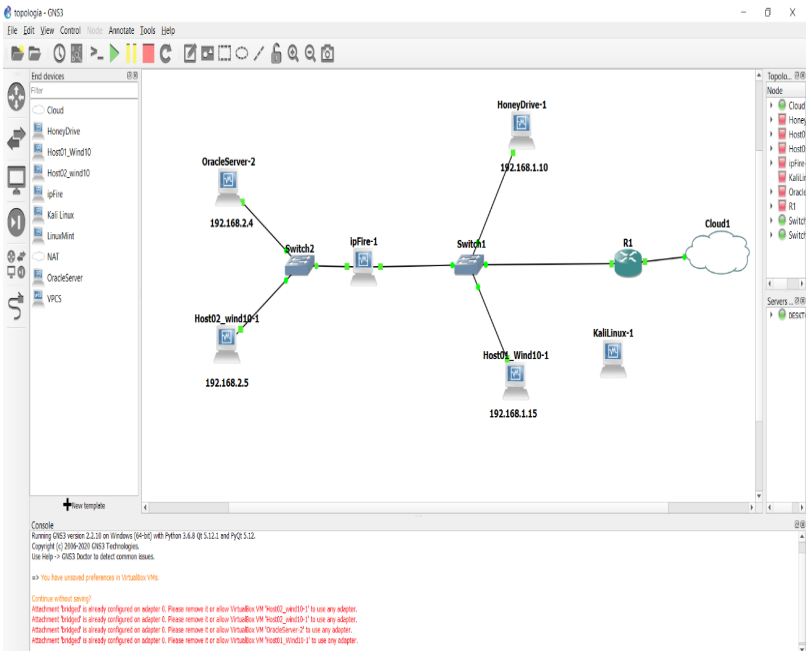


Ilustración 15: Importación de Máquina Virtual

- Al hacer clic derecho sobre un dispositivo se observan las opciones de Iniciar, suspender y detener. Esto nos permitirá manipular la máquina virtual desde GNS3 sin necesidad de abrir VirtualBox.

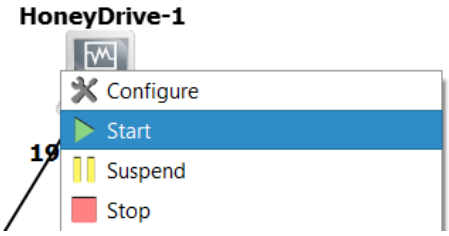


Ilustración 16: Importación de Máquina Virtual

- Al hacer clic en Start se iniciará automáticamente la máquina virtual seleccionada. En este caso Honey Drive.

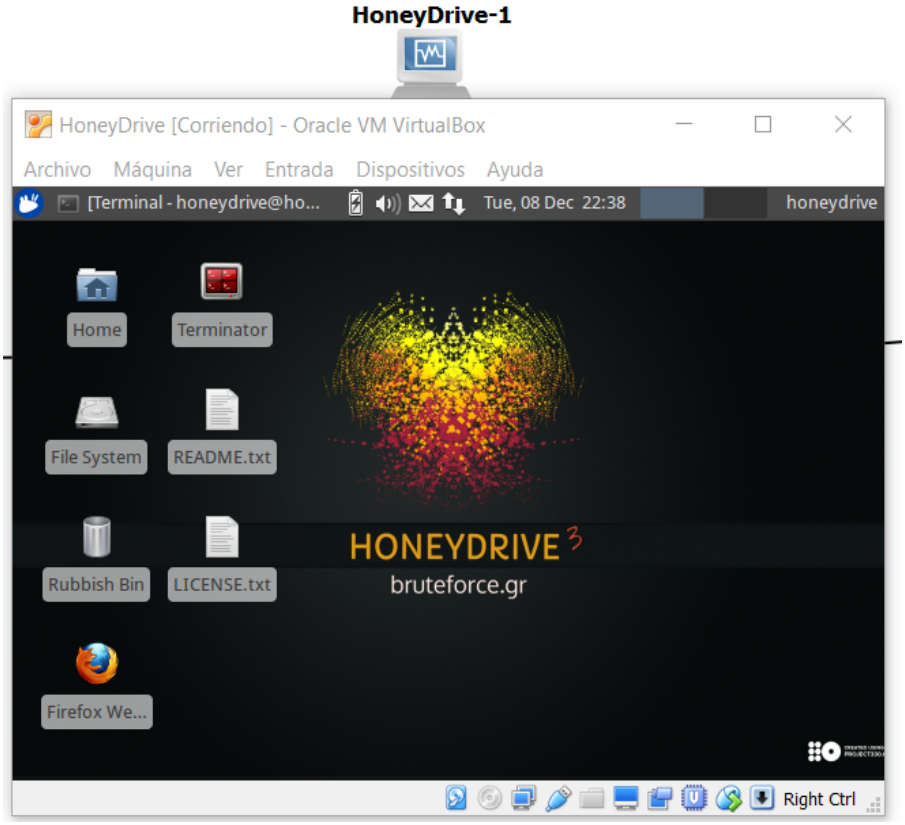


Ilustración 17: Importación de Máquina Virtual

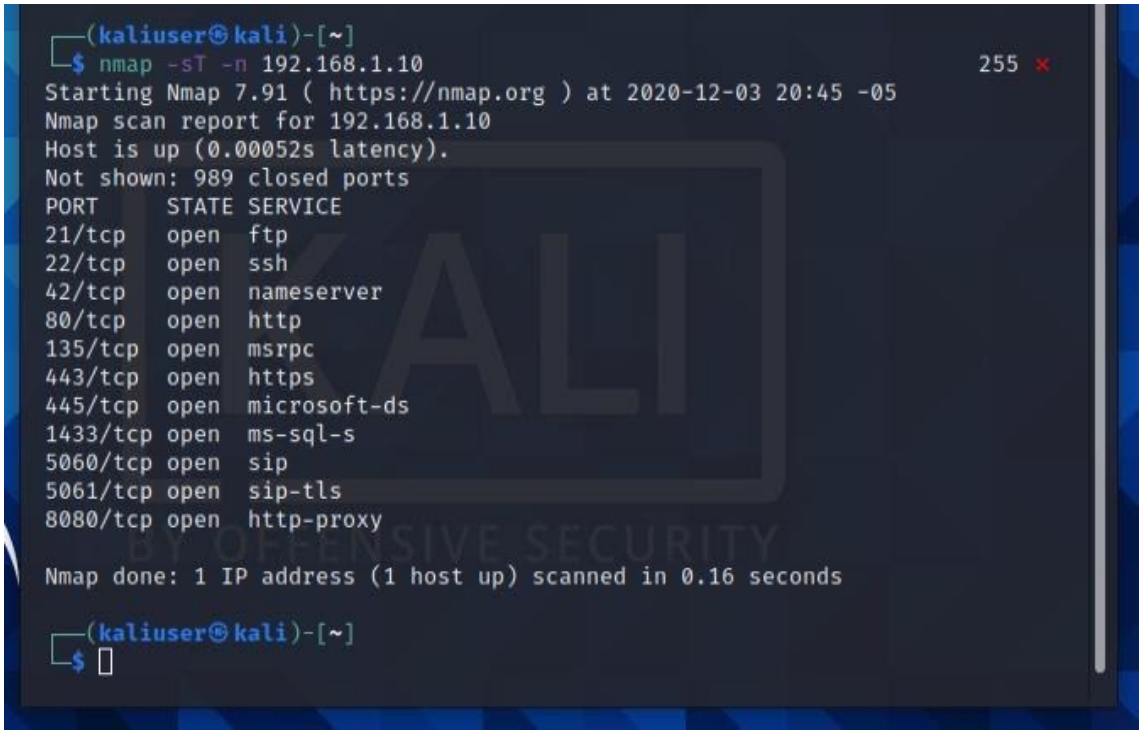
3. CAPÍTULO III: EVALUACIÓN DEL PROTOTIPO

3.1. PLAN DE EVALUACIÓN

Una vez implementada la arquitectura de red perimetral, da paso a la realización de pruebas que permitan evaluar el funcionamiento óptimo de la arquitectura propuesta, cabe mencionar que las pruebas que se realizan se basan en los criterios de penetrating:

3.1.1. Pruebas desde Kali Linux

Desde Kali Linux hacemos un análisis de puertos hacia un host, se realiza mediante la IP asignada. El registro demuestra que el puerto 445 de Microsoft-ds está libre por lo que para el atacante se concluye que es un equipo con sistema operativo Windows

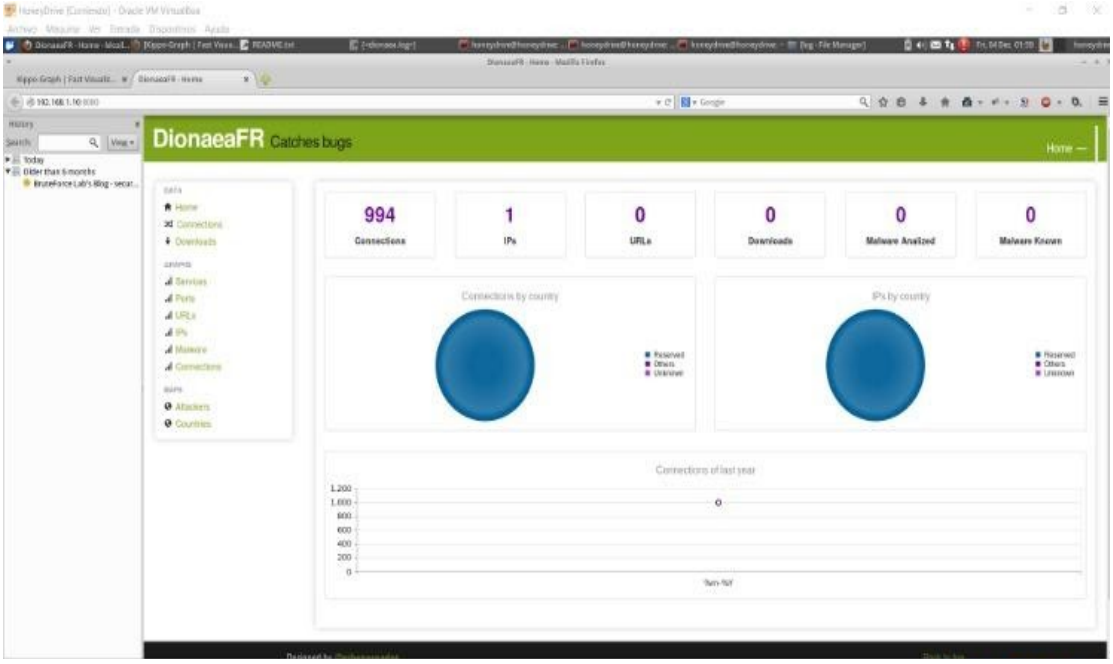


Siguiente en la interfaz de kali utilizando la herramienta Metasploit se realiza un ataque del supuesto puerto de Windows a esa dirección IP en donde se intentó iniciar una sesión, pero no se concretó.

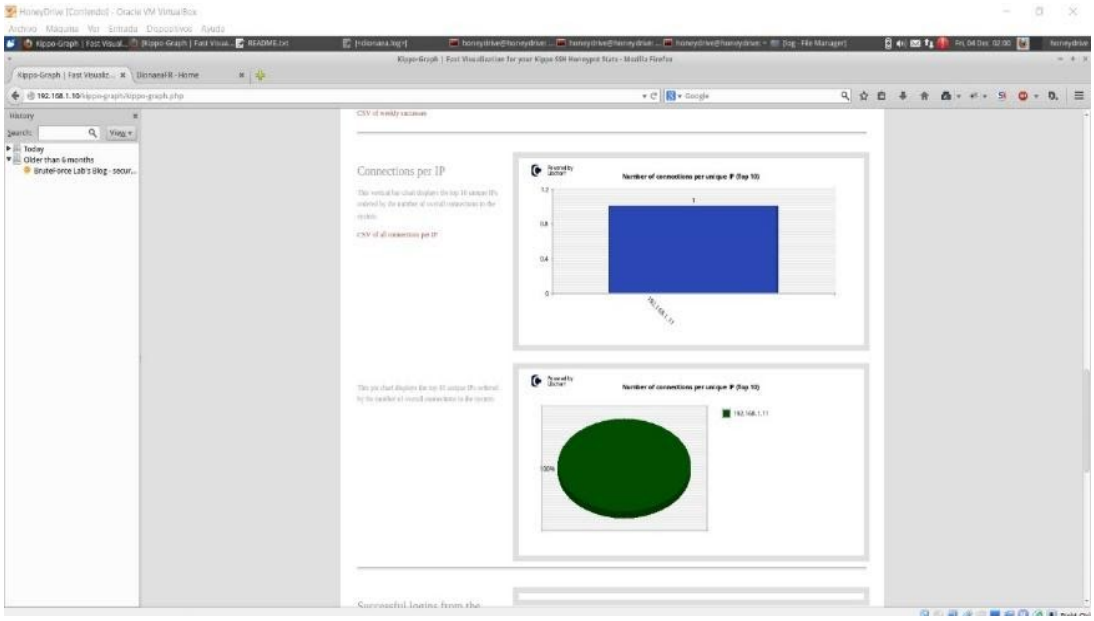
```
msf6 > use exploit/windows/smb/ms06_040_netapi
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms06_040_netapi) > set payload windows/shell/bind_
tcp
payload ⇒ windows/shell/bind_tcp
msf6 exploit(windows/smb/ms06_040_netapi) > set rhost 192.168.1.10
rhost ⇒ 192.168.1.10
msf6 exploit(windows/smb/ms06_040_netapi) > exploit

[*] 192.168.1.10:445 - Detected a Windows XP SP0/SP1 target
[*] Started bind TCP handler against 192.168.1.10:4444
[*] Exploit completed, but no session was created.
msf6 exploit(windows/smb/ms06_040_netapi) >
```

3.1.2 Pruebas desde Honey Drive

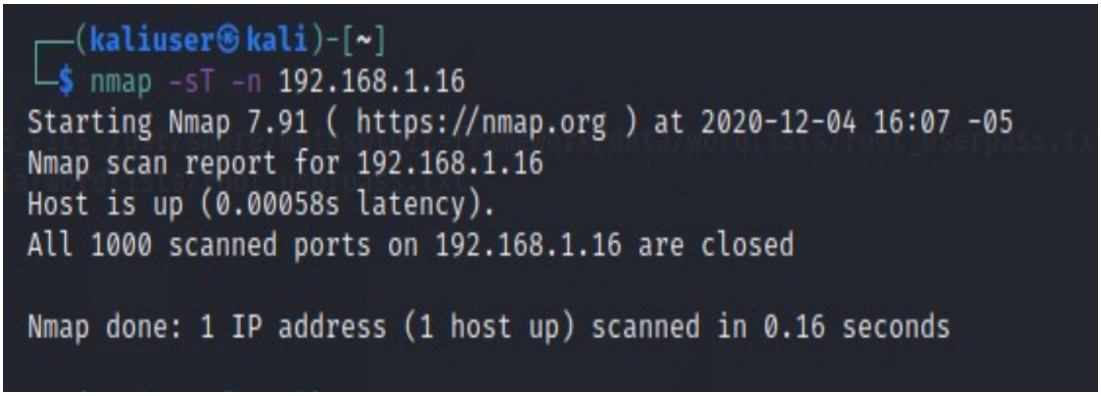
Desde Honey Drive utilizando la herramienta Kippo se evidencia el registro del intento de ataque y el número de direcciones IP's que participaron en dicho ataque para nada exitosa.



Con Kippo Grap también se ve el registro y las estadísticas del ataque con la dirección IP.



Aquí en Kali se realizó el escaneo de puertos a otro equipo con IDS Snort, este caso sin éxitos de encontrar un puerto libre.



A pesar de que los resultados anteriores para Kali fueron negativos, en el equipo con IDS Snort se evidencia este escaneo y además se registra desde que dirección IP se lo realizó.

3.2. RESULTADOS DE LA EVALUACIÓN

Mediante la realización de pruebas desde las diferentes herramientas propuesta para las pruebas de la topología propuesta se evidencia que la misma cumple con los requerimientos establecidos de seguridad perimetral ante posibles ataques, dando paso a posibles mejoras y adaptaciones basado en investigaciones futuras.

Es importante destacar que para futuras investigaciones o aportes de este tipo de topologías se debe tener en cuenta los recursos tecnológicos para realizar las pruebas y evitar fallos en la misma.

3.3. CONCLUSIONES

- El uso de Firewall como bloqueo es de gran utilidad ya que no el ingreso de intrusos a la red interna de la organización, reduciendo así problemas de vulnerabilidad de la información.
- ·El uso del sistema de detección de intrusos ayuda a controlar que usuarios ajenos a la red realicen alguna actividad que ponga en riesgo toda la infraestructura de la red y en consecuencia el despliegue de información interna.
- ·Utilizar una herramienta como honeyDrive permite que al presentarse un ataque, se despliegue la seguridad necesaria para evitar el ingreso de intrusos a la red, por ende el sistema es el que recibe todos estos ataques y evidenciarse como un host débil.
- Al realizar las pruebas de detección de intrusos en una red interna se pueden tomar mejores decisiones para mejorar las vulnerabilidades que esta presenta.

3.4. RECOMENDACIONES

- ·Para implementar una topología de seguridad de red, es indispensable considerar todas las posibles vulnerabilidades que puedan presentarse, porque no todas las redes son para las mismas organizaciones.
- ·En la implementación de firewall utilizar herramientas que estén bien documentadas, que tengan una rápida respuesta antes problemas de ingresos de intrusos o ataque a la red y sobre todo adquirir los recursos de hardware y software necesarios para evitar fallos.
- ·En caso de utilizar herramientas de código libre se debe verificar las vulnerabilidades que esta haya presentado ante la comunidad y si su implementación significa un aporte de seguridad o por el contrario puede conllevar un mayor riesgo a la infraestructura.

BIBLIOGRAFÍA

[1] L. A. Orellana Benavides y R. C. Hernández Vásquez, «Seguridad en Redes de Datos Universidad Don Bosco, Soyapango, 2015.

[2] A. Singh, D. Singh, A. K. Singh, H. Pandey y P. C. Vashist, «Security through Optimizatior Techniques of Firewall Rule Sets,» de *2020 International Conference on Computatic Automation and Knowledge Management (ICCAKM)*, Dubai, 2020.

[3] J. Bolaños, «Diseño de la arquitectura de seguridad perimetral de la red,» 2018. [En líne Available: http://red.uao.edu.co/bitstream/10614/10248/4/T07892.pdf. [Último acceso: 1 Diciembre 2020].

[4] R. Rapuzzi y M. Repetto, «Building situational awareness for network threats in fog/ec computing: Emerging paradigms beyond the security perimeter model,» *Future Generati Computer Systems,* vol. 85, pp. 235-249, 2018.

[5] P. Purpura, «Chapter 9 - External Threats and Countermeasures,» de *Effective Physical Secur (Fifth Edition)*, ScienceDirect, 2017, pp. 219-248.

[6] J. Kizza, Guide to Computer Network Security, Chattanooga: Springer International Publish AG, 2017, pp. i-xxiv.

[7] H. Almohannadi, I. Awan, J. A. Hamar, A. Cullen y J. P. Disso, «Cyber Threat Intelligence fro Honeypot Data Using Elasticsearch,» de *2018 IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA)*, Krakow.

[8] D. Puthal, S. P. Mohanty, P. Nanda y U. Choppali, «Building Security Perimeters to Prote Network Systems Against Cyber Threats [Future Directions],» *IEEE Consumer Electroni Magazine,* vol. 6, nº 4, pp. 24-27, 2017.

[9] H. Lin, Z. Yan, Y. Chen y L. Zhang, «A Survey on Network Security-Related Data Collectio Technologies,» *IEEE Access ,* vol. 6, pp. 18345 - 18365, 2018.

[10] X. Liang, C. Xia, J. Jiao, J. Hu y X. Li, «Modeling and global conflict analysis of firewall polic de *China Communications*.

[11] A. Singh, D. Singh, A. Kumar , H. Pandey y P. Vashist, «Security through Optimizatior Techniques of Firewall Rule Sets,» de *2020 International Conference on Computatic Automation and Knowledge Management (ICCAKM)*, Dubai, 2020.

[12] H. Sheng, L. Wei, C. Zhang y X. Zhang, «Privacy-Preserving Cloud-Based Firewall fc IaaS-based Enterprise,» de *2016 International Conference on Networking and Netwo Applications (NaNA)*, Hakodate, 2016.

[13] P. SenthilKumar y M. Muthukumar, «A Study on Firewall System, Scheduling and Routin using pfsense Scheme,» de *2018 International Conference on Intelligent Computing ar Communication for Smart World (I2C2SW)*, 2018.

[14] I. Winkler y A. T. Gomes, «Chapter 10 - Countermeasures,» de *A Cyberwarfare Approach Implementing Adaptive Enterprise Protection, Detection, and Reaction Strategies*, ScienceDirect, 2017, pp. 105-130.

[15] S.-d. Krit y E. Haimoud, «Overview of firewalls: Types and policies: Managing windov embedded firewall programmatically,» de *2017 International Conference on Engineering MIS (ICEMIS)*, Monastir, 2018.

[16] S. Kemppainen y T. Kovanen, «Honeypot Utilization for Network Intrusion Detection,» vol. 9 Springer, 2018, pp. 249-270.

[17] D. Fraunholz, M. Zimmermann y H. D. Schotten, «An adaptive honeypot configuration deployment and maintenance strategy,» de *2017 19th International Conference on Advanc Communication Technology (ICACT)*, Bongpyeong, 2017.

[18] S. Melese y P. Avadhani, «Honeypot System for Attacks on SSH Protocol,» *Internationa Journal of Computer Network and Information Security(IJCNIS),* vol. 8, nº 9, pp. 19-26, 2016.

[19] P. D. Ali y T. G. Kumar, «Malware capturing and detection in dionaea honeypot,» de *201 Innovations in Power and Advanced Computing Technologies (i-PACT)*, Vellore, 2017.

[20] A. Yaday, S. Raisurana, H. Balaji, P. Lalitha, R. Caytiles y N. Iyengar, «Information Security ir Healthcare Organizations using Low-interaction Honeypot Intrusion Detection System *International Journal of Security and Its Applications,* vol. 11, nº 9, pp. 95-108, 2017.

[21] B. Kumar, «Combining the OGA with IDS to improve the detection rate,» *Materials Tod Proceedings,* 2020.

[22] A. Yazdinejadna, R. Parizi, A. Dehfhantanha y M. Khan, «A kangaroo-based intrusio detection system on software-defined networks,» *Computer Networks,* vol. 184, 2018.

[23] S. Quincozes, C. Alburqueque, D. Passos y D. Mossé, «A survey on intrusion detection ar prevention systems in digital substations,» *Computer Networks,* nº 184, 2020.

[24] N. Naik, R. Diao y Q. Shen, «Application of dynamic fuzzy rule interpolation for intrusi detection: D-FRI-Snort,» *2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE),* 2016.

[25] O. Mebawondu, O. Alowolodu, J. Mebawondu y A. Adetunmbic, «Network intrusi detection system using supervised learning paradigm,» *Scientific African,* vol. 9, 2020.

[26] T. Saranya, S. Sridevi, C. Deisy, T. D. Chung y M. Khan, «Performance Analysis of Machine Learning Algorithms in Intrusion Detection System: A Review,» *Procedia Computer Scien* vol. 171, pp. 1251-1260, 2020.

[27] M. Pawlicki, M. Choraś y R. Kozik, «Defending network intrusion detection systems agair adversarial evasion attacks,» *Future Generation Computer Systems,* vol. 110, pp. 148-154, 2020.

[28] M. Nguyen y K. Kim, «Genetic convolutional neural network for intrusion detection system *Future Generation Computer Systems,* vol. 112, pp. 418-427, 2020.

[29] L. Lv, W. Wang, Z. Zhang y X. Liu, «A novel intrusion detection system based on an optima hybrid kernel extreme learning machine,» *Knowledge-Based Systems,* vol. 195, 2020.

[30] A. Ayodeji, Y.-k. Liu, N. Chao y L.-q. Yang, «A new perspective towards the development robust data-driven intrusion detection for industrial control systems,» *Nuclear Engineering Technology,* vol. 52, pp. 2687-2698, 2020.

[31] I. Karim, «A Comparative Experimental Design and Performance Analysis of Snort-Bas Intrusion Detection System in Practical Computer Networks,» *Computers,* 2017.

[32] Z. Hassan, Shahzeb, R. Odarchenko, S. Gnatyuk, A. Zaman y M. Shah, «Detection of Distributed Denial of Service Attacks Using Snort Rules in Cloud Computing & Remote Cor Systems,» de *2018 IEEE 5th International Conference on Methods and Systems of Navigatio and Motion Control (MSNMC)*, Kiev, 2018.

[33] R. Gaddam y M. Nandhini, «An analysis of various snort based techniques to detect a prevent intrusions in networks proposal with code refactoring snort tool in Kali environment,» de *2017 International Conference on Inventive Communication an Computational Technologies (ICICCT)*, Coimbatore, 2017.

[34] S. A. R. Shah y B. Issac, «Performance comparison of intrusion detection systems an application of machine learning to Snort system,» *Future Generation Computer Systems,* v 80, pp. 157-170, 2018.

[35] R. F. Olanrewaju, B. U. I. Khan, A. R. Najeeb y K. N. A. Ku, «Snort-Based Smart and Swif Intrusion Detection System,» *Indian Journal of Science and Technology,* vol. 11, nº 4, 2018.

[36] S. Patel y A. Sonker, «Rule-Based Network Intrusion Detection System for Port,» *Internatio Journal of Future Generation Communication and Networking,* vol. 9, nº 6, pp. 339-350, 2016

[37] X. Liang, C. Xia, J. Jiao, J. Hu y X. Li, «Modeling and global conflict analysis of firewall polic de *China Communications* .

[38] M. Nguyen y K. Kim, «Genetic convolutional neural network for intrusion detection system *Future Generation Computer Systems,* vol. 112, pp. 418-427, 2020.

[39] R. F. Olanrewaju, B. U. I. Khan, A. R. Najeeb y K. N. A. Ku, «Snort-Based Smart and Swif Intrusion Detection System,» *Indian Journal of Science and Technology,* vol. 11, nº 4, 2018.