BENEMERITA UNIVERSIDAD
AUTONOMA DE PUEBLA

FACULTAD DE CIENCIAS DE LA
COMPUTACION

“PROPUESTA DE SISTEMA DE DETECCIÓN DE INTRUSOS PARA TOPOLOGÍAS DE SEGURIDAD EN REDES DE CÓMPUTO”

TESIS PRESENTADA PARA OBTENER EL TITULO DE LICENCIATURA EN INGIENERIA EN CIENCIAS DE LA COMPUTACION

PRESENTA:

ALAN ISAAC BAUTISTA RUIZ

DIRECTOR DE TESIS:

CARLOS MAURICIO RAMÍREZ ESPITIA

PUEBLA, PUE. AGOSTO 2024

DEDICATORIA

A mis padres que me han apoyado todo el tiempo, incondicionalmente y que siempre me han sacado adelante, motivándome, aconsejándome y ayudándome hasta donde sus posibilidades les permiten.

AGRADECIMIENTOS

Agradezco a las personas que me han ayudado y he conocido durante esta etapa de mi vida en la universidad, mis pocos amigos y profesores que se esmeraron en enseñarme de manera correcta los temas de redes y que gracias a ellos me desenvolví en esta rama de las tecnologías.

A mi asesor que fue constante con las revisiones, el tema y las explicaciones de temas con los cuales a veces tenía confusiones.

A Santiago, mi amigo que durante la carrera me saco de apuros, me ayudaba con dudas y problemas académicos.

Introducción

El rápido desarrollo de la tecnología dentro de las redes en el mundo real ha aumentado el flujo de datos en ellas, lo que ha creado algunos problemas con la seguridad de la información confidencial. A medida que avanza la tecnología, los nuevos atacantes utilizan métodos novedosos para ingresar a las redes corporativas, personales o de cualquier objetivo todos los días. La ciberseguridad es ahora una parte integral de las redes informáticas por ello la seguridad de los sistemas informáticos se ha convertido en un requisito. Esta incluye protocolos, métodos, dispositivos, herramientas y técnicas de ciberseguridad para proteger los datos y mitigar las amenazas tomando en cuenta que debe seguir un estándar de confidencialidad, integridad y disponibilidad de los datos para asegurar todo tipo de información. Dentro de los diversos ataques que los agentes maliciosos implementan solo por mencionar algunos son: Phreaker (Manipulación de la red telefónica para realizar funciones no permitidas), Spammer (Envío de grandes cantidades de mensajes de correo electrónico no solicitados), Phisher (Se usan diversos medios para engañar y que proporcionen información confidencial), Códigos maliciosos (Virus, Gusanos, Caballos de Troya, Roolkit, Puertas traseras, etc.), Ataques de reconocimiento (Queries de información en internet, Barridos de Ping, Exploración de Puertos, Rastreador de paquetes), Ataques de Acceso (Recuperación de información, Obtención de acceso, Escalar privilegios de acceso), Ataque DoS (Ping de la muerte, Ataque pitufo, Ataque de inundación TCP SYN), Ataque DDoS (Tribe Flood Network, Stacheldraht), existen muchos más tipos de ataques y amenazas dentro del mundo de las redes e internet por lo cual es de suma importancia la aplicación de identificadores, herramientas y analizadores de estas redes para un funcionamiento seguro.

En la realización del proyecto se usará Kali Linux como simulador de ataques, Kali Linux es un software de código abierto diseñada para temas de seguridad variados, además tiene muchos programas preinstalados como Nmap, Jhon the Ripper, Metasploit, etc.; con los cuales se pueden explorar vulnerabilidades, escanear redes, puertos, generar ataques de reconocimiento, fuerza bruta,

etc. Principalmente Kali Linux tiene como principal uso: Recopilación de información, Análisis de vulnerabilidades, Análisis de aplicaciones web, Evaluación de bases de datos, Ataques de contraseñas, Ataques Wireless, Ingeniería inversa, Herramientas de explotación, Sniffing y Spoofing, Postexplotacion, Análisis forense, Herramientas de reporte, Herramientas de Ingeniería Social y mucho más, dentro de estos ataques los que se realizaran dentro del proyecto son diversos como ataques de fuerza bruta, ataques de tráfico en la red, escaneo de redes, puertos y aplicaciones y unos cuantos más solo por mencionar algunos, esto con la finalidad de poder usar un IDS, en este caso haciendo énfasis en SNORT.

Un IDS se describe como Sistema de Detección de Intrusos, es un software que supervisa una red en busca de actividades maliciosas o accesos no autorizados y generar informes, para el proyecto SNORT será la herramienta fundamental la cual es de código abierto y sirve para monitorizar la red en tiempo real las principales características son: Monitor de tráfico en tiempo real, Registro de paquetes, Análisis de protocolo, Coincidencia de contenido, Huellas Digitales del Sistema Operativo, Crea registros, Reglas fáciles de implementar. Tiene 3 funcionalidades fundamentales las cuales son: Sniffer (Muestra los paquetes que transitan por la red), Packet Logger (Permite al usuario guardad los paquetes detectados, a partir de estos especificar reglas más concretas para detectar patrones en los paquetes), Network Intrusion Detection System (Permite que se apliquen reglas más específicas refinando los paquetes que se registran) implementar, crear definir algunas reglas con SNORT dando como aportación algunas variantes de reglas propias para detecciones precisas. La finalidad del proyecto es crear un pequeño laboratorio experimental creando una topología básica que integre los elementos que se mencionaron anteriormente, con la cual se hará un sistema de detección de intrusos, tomando en cuenta el uso de las herramientas mencionadas anteriormente, así como de las funcionalidades que ofrece cada una. Todo esto usando un switch, un par de computadoras para levantar el servidor y hacer las pruebas con el IDS y el atacante con Kali Linux, además de una máquina virtual para recrear un escenario real.

Antecedentes del proyecto

Un sistema de detección de intrusos (IDS) es una herramienta importante para garantizar la seguridad de las redes. Hoy en día, el creciente número de ataques informáticos y la sofisticación de las técnicas utilizadas por los atacantes hacen necesario disponer de soluciones cada vez más fuertes y eficaces para detectar y prevenir estas amenazas. El proyecto propuesto surge como respuesta a la necesidad de disponer de un IDS capaz de detectar y prevenir ataques informáticos de forma eficaz. La situación actual demuestra la creciente preocupación por parte de las organizaciones con respecto a la seguridad de sus sistemas y datos, y la necesidad de contar con herramientas que les permitan identificar y mitigar los riesgos potenciales. Los IDS tradicionales tienen una capacidad limitada para detectar nuevas amenazas y adaptarse a patrones de ataque cambiantes. Por tanto, es necesario implementar un IDS que sea capaz de adaptarse dinámicamente a nuevos tipos de ataques y que tenga un alto nivel de precisión en la detección de intrusos. Este proyecto se centrará en el desarrollo e implementación de un sistema donde el IDS se base en la implementación de un sistema robusto en cuanto al análisis de ataques, recolección de paquetes y análisis de estos, lo que permitirá una detección de amenazas más precisa y eficaz. El objetivo es implementar y ejecutar ataques con ayuda de Kali Linux para que con el IDS permita proteger la información crítica y sus sistemas de posibles ataques informáticos además de crear informes y analizar el tipo de ataques lanzados.

Objetivo General

El objetivo de la investigación es implementar el IDS Snort en plataforma Linux/Windows para el monitoreo de datos, paquetes y puertos, también el definir, analizar y crear reglas con SNORT para el análisis de lo antes mencionado y además el realizar diferentes ataques lanzados desde otro dispositivo equipado con Kali Linux. Todo esto se estructura dentro de una red básica de pruebas experimental.

Objetivos específicos

- Proponer un Sistema de Detección de Intrusos para el análisis de anomalías, ataques y envió de paquetes en la red
- Conocer el funcionamiento e implementación de un IDS en Linux
- Conocer el funcionamiento y herramientas que ofrece Kali Linux
- Realizar diferentes tipos de ataque en la red con Kali para que los identifique el IDS
- Identificar y analizar con el IDS cada tipo de ataque y analizar patrones
- Registrar toda la actividad del IDS en el pequeño laboratorio
- Analizar los registros del IDS para establecer y corregir problemas de seguridad o intrusiones en la red
- Para tal propósito se pretende crear una topología básica y simple, donde se tenga un segmento de red LAN (Local Area Network) basándonos en los componentes de los puntos anteriores y los equipos de red requeridos.

Contenido

Índice de Figuras

Índice de tablas

CAPITULO 1

Capítulo 1 Ciberseguridad

1.1 ¿Qué es la ciberseguridad?

Actualmente el uso de la tecnología se ha vuelto una actividad cotidiana además de una herramienta en nuestro día a día y con ello la cantidad de datos e información que viaja por internet se ha disparado en la última década, pero internet es un lugar lleno de vulnerabilidades que pueden atentar con nuestra información. Actualmente se ha vuelto una tarea muy importante el asegurar la información importante que circula por internet, que viaja por dentro y fuera de las grandes empresas y alrededor del mundo pero, es complicado que toda esa información no este expuesta a los ataques malintencionados que se conocen hoy en día, ataques como: Phreaker, phishing, spammer, ransomware, spyware, códigos maliciosos, robo de identidad, secuestro de dispositivos, ataques de reconocimiento, extorsión, robo de información, ataques de acceso, etc. Debido a todo el riesgo que se corre a causa de estas amenazas los especialistas en redes desarrollan e implementan diversas herramientas que utilizan los especialistas en seguridad para que los sistemas sean más seguros y reducir el riesgo de exposición a estos ataques. Para poder entender que significa ciberseguridad es importante definir que no es lo mismo que la seguridad en redes, ya que la ciberseguridad se encarga de la protección de toda información confidencial mediante la mitigación o en otras palabras protección contra amenazas dentro de los sistemas y redes; en cuanto a la protección de redes es cualquier actividad que se aplica para proteger la integridad de la red y sus datos.

La ciberseguridad se encarga de mantener protegidos todos los sistemas dentro de las redes, además de ello mantener los datos digitales seguros contra el acceso y uso no autorizado, a nivel personal se debe proteger nuestra identidad e información personal. A nivel empresarial, se deben proteger los datos tanto de usuarios como de clientes. En pocas palabras la ciberseguridad es el asegurar los recursos tecnológicos, hacer que la información solo esté disponible y se accesible para las personas autorizadas y por el contrario sean inaccesibles a agentes externos. (GSM Latin America; 2018:24)[1].

Para complementar el término de la ciberseguridad hablare de la triada, por sus siglas en ingles CIA (Confidentiality, Integrity, Availability), las cuales significan: Confidencialidad, Integridad y Disponibilidad, la CIA es un concepto fundamental ya que son los principios de la seguridad. Además de que gracias a la CIA es fácil evaluar las amenazas y vulnerabilidades que tienen los datos de una organización.

Figura 1. Triada “CIA”

La confidencialidad (Confidentiality) se refiere a que los datos no deben estar disponibles o deben ser revelados a los individuos que no están autorizados, esto quiere decir que deben mantenerse totalmente privados mediante controles apropiados.

La integridad (Integrity) se refiere a que no debe haber cambios o alteraciones a programas y datos de manera externa, esto quiere decir que esto solo puede ser de manera autorizada, además de que los sistemas deben actuar de manera ordinaria para el fin que fueron hechos y no deben ser modificados.

La disponibilidad (Availability) se refiere a que el sistema debe operar adecuadamente en relación tiempo-respuesta para poder tener y usar la información en todo momento. (Samonas & Coss, 2014) [2]

Retomando lo antes mencionado, hay múltiples soluciones que se pueden implementar dentro del ámbito de la ciberseguridad para poder prevenir o mitigar los múltiples tipos de ataques existentes, dentro de las soluciones a implementar podemos encontrar el uso de Firewalls en sus múltiples plataformas (Hardware y

Software), implementar IDS (Sistema de Detección de Intrusos), implementar sistemas de correlación de logs y eventos, autenticaciones, accesos remotos y VPN's, filtros de contenido, sistemas de protección contra ataques de tipo DoS (Denial of Service/Denegación de Servicio), control de acceso a la red, encriptación de datos, entre muchos otros que se encargan de prevenir, administrar y mitigar los múltiples ataques que conocemos en la actualidad. (Seguridad, 2023) [3]

1.2 Diferentes tipos de amenazas

Dentro de los distintos tipos de amenazas podemos mencionar que existen varios, por ejemplo las amenazas en la red como los hackers, que se dedican a realizar actividades maliciosas utilizando herramientas de software como Kali Linux, Parrot, Linux y variantes de Linux orientadas a la ciberseguridad y ciberataques en el caso de los hackers, también existen amenazas que atentan desde programas como lo son los códigos maliciosos, ataques a servidores o empresas, aprovechar vulnerabilidades en los sistemas para acceder a información clasificada, entre otras técnicas criminales como son el robo de identidad, chantaje o extorción, ENISA Threat Landscape (Agencia para la Ciberseguridad de la Unión Europea) y la INCIBE (Instituto Nacional de Ciberseguridad de España) clasificaron las amenazas en diversos grupos por mencionar. (Threat Landscape, 2016) [4][5]

Dentro de los ataques por malware tenemos diversos tipos de ataques pero los más conocidos e implementados por los agentes maliciosos tenemos

- Malware
- Ransomware
- Virus
- Gusanos
- Caballos de Troya

Dentro de los ataques por ingeniería social y que son los más comunes hoy en día tenemos

- El phishing

- Desinformación
- Spam o correo no deseado
- Fraudes Online

También tenemos ataques muy comunes a las conexiones como los son

- Redes con IP falsa
- Spoofing o suplantación
- Ataques DoS (Denegación de Servicio) y DDoS (Ataque Distribuido de Denegación de Servicios)
- Escaneo de puertos
- Sniffing

Algo que muy comúnmente vemos y de igual manera se ha ido volviendo un problema tanto para instituciones financieras, públicas y gubernamentales además de que actualmente los hackers también implementan de una manera muy cotidiana para hacerse de información son los ataques a contraseñas dentro de los cuales los principales son

- Fuerza bruta
- Ataque por diccionario

MALWARE

Malware es un término general utilizado para describir cualquier software o firmware destinado a realizar un proceso no autorizado que tendrá un impacto adverso en la confidencialidad, integridad y disponibilidad de un sistema. Generalmente se busca causar daño a los dispositivos, obtener información de la víctima o tomar el control, acceso e información de los dispositivos de la víctima. Tradicionalmente los ejemplos de tipos de código malicioso incluyen virus, gusanos, troyanos, spyware, adware y otras entidades basadas en código que infectan un host.

Los virus, gusanos y caballos de Troya se diferencian en muchos puntos, como el vector de infección, la replicación, la distribución y el control de propagación y hasta atacantes. Los componentes de malware utilizados en un ataque dependen del

objetivo del actor de la amenaza. Esto puede ir desde obtener el control sobre sistemas y redes o sobre datos hasta hacer que no estén disponibles por completo. El desarrollo de los componentes que constituyen el malware requiere conocimientos específicos. A medida que las capacidades de detección y de los equipos evolucionan con el tiempo, el código malicioso suele estar en continuo desarrollo para adaptarse a los requisitos cambiantes de los entornos de las víctimas. Este código se vende, se comparte, se roba y se reutiliza, lo que dificulta que los investigadores y las autoridades atribuyen correctamente los actores de amenazas involucrados en una campaña de malware. El código malicioso prevalece, al igual que las nuevas familias y cepas de malware. (Monje & Alexander) [6]

RANSOMWARE

El ransomware es un malware, o software malicioso, que bloquea los datos o el dispositivo de una víctima y amenaza con mantenerlo bloqueado además de que se pueden secuestrar los documentos o base de datos de la victima de manera que todos sus archivos se encriptan, a menos que la víctima pague un rescate al atacante y la mayoría de las veces es imposible la recuperación de los archivos. El ransomware principalmente se propaga a través de correos electrónicos falsos con información engañosa y haciéndose pasar por empresas de prestigio con la finalidad de que la víctima caiga en su trampa accediendo a enlaces, documentos o páginas web que vienen adjuntas al mensaje, también se esconden dentro de actualizaciones de sistemas, aplicaciones piratas o múltiples documentos que las personas pueden descargar de la web. Los delincuentes cibernéticos o hackers suelen secuestrar bases de datos de instituciones gubernamentales, empresariales, bancarias o corporaciones de renombre, donde en su actuar piden un rescate monetario para su liberación. Estos rescates mayormente se piden en el sistema *pseudo anónimo bitcoin*, cuyas transferencias no son reembolsables y por ese motivo; se considera de alta peligrosidad.

Hay tres elementos clave en todo ataque de ransomware: activos, acciones y chantaje. Se utilizan muchos métodos para obligar al objetivo a cumplir con las

peticiones de rescate: publicidad del ataque, filtración parcial o total de datos, ataque de denegación de servicios distribuidos contra la infraestructura objetivo.

Hasta mediados del 2010, el ransomware se centraba en realizar dos acciones: cifrado de la información y bloqueo del dispositivo. Sin embargo, el ransomware ya no está ligado a esas descripciones y su evolución han hecho de esas categorías algo simple.

Actualmente, el ransomware puede ejecutar 4 acciones principales: **bloquear, cifrar, eliminar y robar**. Y nos referimos a estas acciones como ***LEDS (Lock, Encrypt, Delete, Steal).***

Los activos son cualquier cosa de valor para una empresa u organización; y los activos más atacados por el ransomware son las carpetas y archivos con información de clientes o de los mismos trabajadores de la empresa víctima.

En la Figura 2 se muestran las capacidades del ransomware actual tal y como fueron analizadas y está claro que hay margen para la variabilidad a medida que evolucionaba el panorama de las amenazas. (Threat Landscape, 2016) [4]

Figura 2. LEDS

LEDS ACCIONES

- CERRADURA (Lock): La acción de bloquear un bien puede implicar cosas distintas. En teléfonos, puede simplemente cambiar el PIN y bloquear la pantalla. En el caso de una aplicación, puede cambiar las credenciales para acceder a ella.
- CIFRAR (Encrypt): Cifrar se refiere a usar un algoritmo de cifrado para que el contenido de un archivo, carpeta o texto solo esté disponible para los que conocen el algoritmo de cifrado utilizado y posea la clave para descifrarlo.
- BORRAR (Delete): En el caso de las bases de datos en memoria, el borrado consiste en pedir a la base de datos que borre el archivo. El borrado también puede hacerse a máquinas virtuales en entornos de nube utilizando cuadros de mando oficiales, en los que no interviene el sistema operativo.
- ROBAR (Steal): Se refiere a copiar el activo al control del atacante. Puede ser mediante la exfiltración de datos a internet, mediante la copia de datos a una carpeta *local secreta* desconocida para el propietario del activo.

El ciclo de vida del ransomware se mantuvo sin cambios hasta el 2018, cuando este; empezó a añadir más funcionalidad y las técnicas del chantaje maduraron. Podemos identificar 5 etapas de un ataque de ransomware: acceso inicial, ejecución, acción sobre objetivos, chantaje y negociación del rescate. A continuación, se detallan las etapas.

a) ACCESO INICIAL: El ransomware utiliza las mismas técnicas de acceso que emplean otros ataques, como el aprovechamiento de vulnerabilidades de software, el acceso a través de credenciales robadas, el phishing, entre otras. El problema se debe a la falta de notificaciones de incidentes por parte de las organizaciones comprometidas, lo que reduce el intercambio de información y las lecciones aprendidas.
b) EJECUCIÓN: Esta actividad puede durar varias semanas, dependiendo del actor de amenaza, el tamaño y las defensas del objetivo. Este movimiento suele completarse antes de que el ransomware comience a funcionar, aunque cuando el ransomware se inicia también puede moverse dentro de la

red de la víctima. Una vez localizada la información relevante y antes de ejecutar el ransomware, se realiza una *limpieza* (dejar en un estado inoperante el software de seguridad, detener programas como bases de datos que puedan interferir en la escritura, detener funciones de recuperación de sistemas, logs, etc.). El siguiente paso; es el despliegue del ransomware.

c) ACCIÓN SOBRE LOS OBJETIVOS DE: Una vez desplegado, el ransomware ataca la disponibilidad y/o la confidencialidad del objetivo a través de una serie de acciones. Las acciones del ransomware pueden tener lugar semanas después de la infección inicial del sistema, dando a los atacantes tiempo adicional para acceder a más sistemas internos.
d) CHANTAJE: Una vez comprendida la disponibilidad de la información, el actor de la amenaza procede a chantajear al objetivo para obtener un rescate a cambio de la disponibilidad de su información. Los 3 componentes principales del chantaje son la comunicación, la amenaza y la demanda. La comunicación es el acto de informar al objetivo lo que está sucediendo. La amenaza, es la pérdida o daño que se producirá si no se satisface la demanda. Además, anda en lo que el actor de la amenaza espera obtener de la situación.
e) RESCATE-NEGOCIACIÓN: Esta negociación tiene dos resultados: las victimas pagan el rescate o no pagan. Por desgracia, es muy difícil cuantificar quién pagó o no en el rescate y en qué caso se acordó un rescate menor. A menudo hay informes en los que se comunican los ingresos totales de los actores de amenaza, pero no a nivel individual. También hay actores de amenazas que después de un pago exitoso eliminan el nombre del objetivo comprometido de su sitio web público.

Para prevenir ataques de ransomware las recomendaciones deberían ayudar a las organizaciones a prepararse para un ataque de ransomware antes de que ocurra y durante sus secuelas.

- Tener una copia de seguridad de todos los archivos críticos para la empresa y de sus datos personales además de mantenerla actualizada y aislada de la red.
- Aplicar la regla 321 de las copias de seguridad para todos los datos, 3 copias, 2 soportes de almacenamiento diferentes y una copia fuera del sitio.
- Mantener los datos personales cifrados de acuerdo con las disposiciones del GDPR (Reglamento General de Protección de Datos de la Unión Europea) y utilizando controles adecuados basados en el riesgo.
- Ejecute software de seguridad en sus dispositivos que puedan detectar la mayoría del ransomware.
- Mantenga al día su conocimiento en materia de seguridad, la política de seguridad y su política de protección de la privacidad, implementar medidas como la segmentación de la red, parches actualizados, copias de seguridad periódicas y una gestión de identidades, credenciales y acceso adecuado.
- Realice una evaluación periódica de los riesgos y considere la posibilidad de contratar un seguro contra ransomware basado en esta evaluación.
- Restrinjan los privilegios administrativos, tengan cuidado al otorgar privilegios administrativos, ya que la cuenta administrativa tiene acceso a todo, incluso a cambiar configuraciones o eludir ajustes de seguridad críticos.

En caso de que una organización o un particular sean víctimas de ataques de ransomware, se han propuesto varias recomendaciones:

- Póngase en contacto con las autoridades nacionales de ciberseguridad para saber cómo manejar y cómo hacer frente al Ransomware.
- No pague el rescate y no negocie con los actores de amenaza.
- Poner en cuarentena los sistemas afectados, se recomienda desconectar de la red los sistemas afectados para contener la infección y evitar que el ransomware se propague.
- Bloqueé el acceso a los sistemas de copia de seguridad hasta que se elimine la infección.

Además, es muy recomendable compartir información con las autoridades sobre el incidente del ransomware. Esto puede mejorar las lecciones aprendidas para ayudar a otras víctimas potenciales. (Threat Landscape, 2016) [4]

VIRUS

Un virus es un software malicioso que se une a otro programa para ejecutar una función específica no deseada en un equipo. Para comprender mejor este fenómeno, es importante enfatizar que los virus informáticos consisten en un conjunto de instrucciones maliciosas adjuntas a un programa o archivo ejecutable legítimo. El virus debe ser ejecutado por el usuario final para estar activo. Una vez activado, el virus busca otros archivos ejecutables y se propaga, extendiéndose e infectando múltiples categorías de archivos en el sistema de almacenamiento. Los efectos de los virus pueden ser inofensivos sólo causando molestias al usuario o pueden ser altamente destructivos hasta el punto de alterar o eliminar permanentemente los datos contenidos en el disco duro. Actualmente, la mayoría de las infecciones por virus se propagan a través de dispositivos periféricos como unidades USB y CD. Esto también es posible a través de redes públicas, correo electrónico o como mencionamos al inicio por medio de aplicaciones o software legítimo y sitios web. Aunque algunos virus pueden pasar desapercibidos ya hay muchos métodos de detección y defensa como IPS (Sistema de Prevención de Intrusos), Firewall, antivirus, etc, ya que en su mayoría son reconocidos y dejan algún tipo de rastro reconocible en los sistemas infectados. (¿Qué es un virus informático?)[7]

GUSANOS

Un gusano ejecuta el código arbitrario e instala copias de si en la memoria de una computadora infectada hasta infectar a otros hosts. En un contexto más detallado, es importante enfatizar que los gusanos informáticos son una estructura de código hostil altamente dañino que se caracterizan por la capacidad de autorreplicarse sin requerir la intervención activa del usuario. Esta singularidad les brinda la capacidad

de propagarse de forma independiente y rápida a través de sistemas interconectados y explotar vulnerabilidades de seguridad de red y software. El objetivo principal de los gusanos es invadir los sistemas informáticos y explotar las vulnerabilidades descubiertas, lo que puede tener muchos efectos negativos. Estos impactos a menudo incluyen ralentizaciones significativas de dispositivos, redes locales e incluso redes más grandes, lo que afecta negativamente la eficiencia y el rendimiento del desempeño de la red. Además, estos gusanos pueden realizar cambios no autorizados en la configuración del sistema, lo que plantea graves riesgos para la integridad y seguridad de los datos. (Ataques informáticos más comunes en el mundo digitalizado, 2022)[8]

CABALLO DE TROYA

Un caballo de troya es una aplicación pensada para hacerse pasar por otra. Cuando se descarga y se instala ataca al usuario desde dentro. En otras palabras, un caballo de Troya es una forma de malware oculto en un archivo, documento o software que realiza operaciones maliciosas bajo la apariencia de una funcionalidad legítima deseada por el usuario. Este truco permite al troyano explotar los privilegios otorgados al usuario que lo ejecuta y operar en secreto en segundo plano. Es importante tener en cuenta que grandes cantidades de documentos y archivos que descarga de Internet pueden contener caballos de Troya. El troyano se activa tan pronto como se abre o ejecuta un archivo infectado y continúa ejecutándose incluso si el usuario cierra el documento o sale de una aplicación aparentemente inofensiva. Esta actividad maliciosa continua puede tener muchos efectos dañinos. Estos impactos incluyen daño inmediato al sistema, permitiendo a los atacantes acceso remoto no autorizado y la capacidad de realizar acciones de forma remota en el sistema comprometido y extraer información o datos confidenciales y representar una amenaza grave. Tanto para la integridad del sistema como para la confidencialidad de la información. (Ataques informáticos más comunes en el mundo digitalizado, 2022)[8]

Phishing

Es parte de las técnicas de Ingeniería social donde se envían correos fraudulentos donde el atacante se hace pasar por entidades de confianza llámese por ejemplo Amazon, Mercado Libre, Correos de México o hasta instituciones financieras como Banorte, Banamex o inclusive empresas como Microsoft, Apple, etc; donde se les envía un archivo o enlace adjunto malicioso para infectar a la víctima con Malware, para que cuando el objetivo ya abrió el enlace, documento o aplicación adjunta este malware se encargue de obtener por regularidad datos sensibles e inclusive credenciales como por ejemplo contraseñas, números de cuenta, mensajes en específico, etc. (Estado y Evolución de la detección de intrusiones en los Sistemas Industriales 2017)[9]

Desinformación

La desinformación es otro tipo de ataque dentro de la ingeniería social ya que aunado a esto la representación más certera para esto se ve reflejado dentro de las famosas FAKE NEWS o comúnmente conocidas como noticias falsas, esto como tal conlleva a recabar información certera sobre algún tema de importancia o en específico de fuentes confiables o no y alterarlas a gusto y conveniencia de una entidad en específico, a tal grado de hacer redes o cadenas de distribución de esta información alterada, con la finalidad de que la gente crea que la nota es verdadera a tal grado que se empieza a distribuir esta información por diversos medios sin tener certeza de que es verdadera o certera, ni revisar fuentes confiables. Como ejemplo podemos mencionar la manipulación de la información en algunos lugares del mundo donde el régimen gubernamental es estricto y autoritario y no permiten que la información certera llegue a las personas, manipulando de tal modo, que el gobierno distribuye tal información alterada para manipular al pueblo, provocar el pánico o el sometimiento de las personas. (BARTOLOMÉ, 2021) [10]

Spam o correo no deseado

Este tipo de ataque de ingeniería social no son más que una serie de mensajes no solicitados que llegan a nuestra bandeja de correo electrónico, siempre de un usuario o remitente desconocido, generalmente el contenido de este correo contiene publicidad u ofertas con enlaces o documentos con malware que despistan al usuario final para que este acceda y caiga en alguna trampa, la mayoría de estos correos contiene características que los van a distinguir siempre de un correo legítimo, ya que usualmente se usan encabezados de correo llamativos o modificados para parecer reales, con la finalidad de enganchar al usuario pero dentro del contenido hay faltas de ortografía, direcciones de correo electrónico sospechosas, la hora que llevan algunos correos no coinciden con la hora legitima de llegada, entre otros aspectos que al final delatan esta acción fraudulenta. Principalmente están hechos para que cuando el usuario accione el malware contenido en el correo el atacante pueda acceder a datos sensibles e incluso tomar control del dispositivo que llegue a infectar. (Ureña Centeno, Ciberataques, La Mayor Amenaza Actual 2015)

Fraudes Online

Los fraudes online o mejor conocidos como fraudes cibernéticos no son más que la combinación y uso de algunos tipos de ataques que ya hemos mencionado a lo largo de algunas amenazas que ya abordamos anteriormente. Por lo regular los atacantes usan la ingeniería social y algún otro método para implantar malware en algún dispositivo y así apoderarse de la información, cuentas o simplemente chantajear al usuario con alguna de su información personal obtenida de redes sociales, donde pueden surgir desde chantajes sobre algún tema hasta amenazas si es que se llega a publicar información demasiado sensible como el lugar donde vive o la rutina diaria que se sigue y se publica en esas cuentas. La mayoría de las veces hay que tener precaución con lo que se publica dentro de las redes sociales y no subir a la red nuestra información sensible, al igual que si llegamos a tener este tipo de percances en donde quieran cometer un fraude, lo primero es no alterarse, verificar la información del atacante como el tipo de correo, la información que

describe y nunca acceder a enlaces o bajar documentos o archivos adjuntos en sus mensajes, ya que esto pondría en riesgo nuestro dispositivo y nuestra confidencialidad e información.

Redes con IP falsa

También conocidas como redes trampa en realidad son redes de internet (Wi-Fi) falsas, este tipo de redes son un canal de internet abierto a partir de una red legitima, en esta segunda red abierta de internet se configura con parámetros idénticos a la original como el nombre y los servicios, cuando la víctima se conecta creyendo que está navegando por una red segura, el atacante lo que busca es robar y acceder a nuestros datos e inclusive implantar un malware en nuestro dispositivo mientras el usuario navega en la red. (INTEDYA, GUÍA DE CIBERSEGURIDAD)[12]

Spoofing o suplantación

El Spoofing es la implementación de técnicas de hacking para suplantar nuestra identidad, la de una web o alguna institución, su principal objetivo es el de tener acceso a nuestros datos. Se han clasificado cuatro tipos principales de ataques Spoofing.

1. El primero es el IP Spoofing que consiste en que el atacante hace pasar la dirección IP de la víctima por una distinta, para el obtener esa dirección IP perteneciente a la red para hacerse pasar por el usuario y así evadir las restricciones y filtro para hacer llegar al usuario un malware y apoderarse de sus datos sensibles o inclusive el dispositivo.
2. El segundo es el Web Spoofing y consiste en enmascarar un sitio web falso y hacerlo pasar por un sitio web legítimo, usando por ejemplo los mismos diseños, tipografía e inclusive en enlace URL muy similar al original, con esto el atacante hace creer a la víctima que está accediendo al sitio web original y así obtener sus credenciales en múltiples casos contraseñas y datos que llegue a introducir la víctima en ese sitio web falso, además de que el atacante se apoya de la ingeniería social para lograr su cometido.

3. El tercero es el Email Spoofing que consiste en suplantar un correo electrónico de una entidad en específico que ya obtuvo por otros medios o tipos de ataques y al hacerse pasar por dicha entidad suelen enviar o reenviar de manera desproporcionada correos de spam o algún tipo de correo con amenazas o fraudes y así esparcir un malware para hacerse con los datos sensibles de todos los victimarios que lleguen a descargar o abrir el contenido de esos correos.
4. Y por último el DNS Spoofing consiste en usar malware o programas maliciosos por el atacante para aprovechar las vulnerabilidades que lleguen a existir en los sistemas o redes para secuestrar el direccionamiento y acceso a internet y cuando la víctima intente acceder a un sitio web de interés directamente será redirigido a un sito elegido por el atacante que habitualmente suelen ser sitios fraudulentos o que hacen que la víctima descargue algún malware en los dispositivos desde los que se encuentra navegando. (INTEDYA, GUÍA DE CIBERSEGURIDAD)[12]

Ataques DoS y DDoS

El ataque DoS o mejor llamado Denegación de Servicio consiste en incapacitar el hardware, el software o ambos en un dispositivo, con la finalidad de privar el acceso a dicho dispositivo o a la misma red de la víctima.

Los ataques DDoS son aquellos que se implementan de manera distribuida en donde se usan varios dispositivos infectados. En este caso por llamarles “zombis”, ya que pueden ser activados remotamente y sin que los usuarios infectados se den cuenta. Una vez activados estos dispositivos y en control del atacante, los usa para atacar servidores o redes y estas colapsen para así vulnerar sistemas, usar puertas traseras y acceder a otros dispositivos, robar información o simplemente secuestrar esos servidores o redes y solicitar un soborno a cambio de ceder el ataque. (Cando Segovia & Medina Chicaiza, Prevención en ciberseguridad enfocada a los procesos de infraestructura tecnológica) [13]

Escaneo de puertos

El escaneo de puertos consiste en la exploración dentro de una red para identificar los servicios que está ofreciendo dicha red o un servidor dentro de esta, se realizan conexiones o múltiples intentos de conexión a los múltiples puertos de enlace (TCP/UDP) esperando a tener una respuesta e identificando que puertos están abiertos, cerrados o si tienen protocolos de seguridad, con la finalidad de así poder infiltrarse a la red y robar datos, información e inclusive apoderarse de algún o algunos dispositivos que se encuentren conectados y vulnerados en ese momento dentro de la red. (Amador, Arboleda, & Bedón, Utilizando Inteligencia Artificial para la detección de Escaneos de Puertos) [14]

Sniffing

El Sniffing no es más que una técnica donde el atacante usa herramientas de hacking con las cuales monitorean el tráfico dentro de la red, una vez dentro, el atacante lo que busca es obtener, interpretar o robar los paquetes que se envían por la red para poder analizarlos o descifrarlos y obtener la información que se envía por esos paquetes. (Kuma & Agarwal, Hacking attacks, methods, techniques and their protection measures) [15]

Fuerza bruta

Los ataques de fuerza bruta consisten en la adivinación por medio de programas y herramientas para poder obtener claves y contraseñas de la víctima mediante el uso de datos personales o algún método de reconocimiento como los ya mencionados antes, por lo cuales se obtiene información de la víctima y se generan múltiples contraseñas o claves a base de ella usando el abecedario, números y símbolos, hasta que logran obtener esa clave y así acceder a su información personal. (V. López, Papel de la Explosión combinacional en ataques de Fuerza Bruta) [16]

Ataque por diccionario

Este tipo de ataque también se considera como ataque de fuerza bruta, aunque en vez de que el atacante genere contraseñas con probabilidades, implementa un programa y un archivo predefinido o con un listado de contraseñas más comunes

que existen en la web, una vez usado esto, el atacante ya no tiene que generar contraseñas manualmente y ya con este conjunto de contraseñas predefinidas puede agregar o quitar a conveniencia con la misma finalidad de acertar y robar la información o hacerse de la información e inclusive los dispositivos. (V. López, Papel de la Explosión combinacional en ataques de Fuerza Bruta) [16]

Dentro de las amenazas que mencionamos, muchas de ellas se pueden evitar desde un comienzo tomando medidas tan simples como el uso e implementación de un Firewall, IDS, o filtros básicos dentro de las redes, para filtrar los ataques antes de que lleguen a entrar dentro de nuestra red, esto con la finalidad de no ser víctimas de alguno de ellos.

1.3 Diferentes tipos de atacante

Sabemos que la información que se almacena en nuestros dispositivos o en la "nube" es y debe ser confidencial y personal, aunque debemos recordar que no siempre está segura donde la guardemos siempre y cuando el dispositivo o plataforma este activo. Las grandes empresas y organizaciones no están protegidas de ser víctimas de Hackers que conocen su trabajo. Todos actúan con motivos específicos y en lo general por dinero. Por eso es necesario conocer los diferentes tipos de piratas informáticos como actúan y sus posibles motivaciones o porque lo hacen.

Para familiarizarnos con el termino de hacker en resumen son personas que se dedican a usar sus conocimientos de informática, programación y uso de plataformas y software dedicado con el cual pueden cometer delitos desde el robo de información, captura y secuestro de dispositivos, secuestro y daño a plataformas y sitios web, hasta dedicarse al delito mayor como venta de sustancias toxicas, narcóticos y múltiples cosas por internet.

ACTORES DE LA AMENZA Y MOTIVACIÓN

En los informes de ENISA Threat Landscape (Agencia para la Ciberseguridad de la Unión Europea) se consideran cuatro categorías principales de actores de amenazas a la ciberseguridad: ciberdelincuentes, hackers a sueldo, hackers

patrocinados por el Estado y hacktivistas. Aunque principalmente y generalizando tenemos tres tipos de hacker dedicados los cuales son los hackers de sombrero blanco, sombrero gris y sombrero blanco.

Hackers maliciosos o de sombrero negro (Black Hat): Este tipo de atacantes siempre están en busca de explotar vulnerabilidades en las redes o sistemas con fines ilegítimos. Este tipo de hacker con sus conocimientos por lo regular vulnera sistemas, dispositivos y redes con la finalidad de robar la información personal de alguna entidad con la finalidad de ganar dinero pidiendo sobornos, rescates de información o algún otro método nada ético.

Hackers éticos o de sombrero blanco (White Hat): Podemos decir que son los chicos buenos ya que son aquellos Hackers que realizan pruebas de seguridad, detectar vulnerabilidades y trabajan para proteger sistemas, redes y dispositivos que sean importantes para la seguridad. Usualmente se está en conflicto con los Hackers de sombrero negro que tratan de vulnerar o acceder a los sitios o empresas que contratan sus servicios de protección.

Hackers de sombrero gris (Gray Hat): Son aquellos que operan en una zona intermedia entre el bien y el mal, a menudo con permiso para probar sistemas pero sin fines maliciosos o poco éticos. Por lo general no realizan los ataques para beneficio personal o con intenciones de realizar el mal, pero pueden llegar a violar leyes o cometer crímenes con la finalidad de completar su cometido.

El principal motivo de los ciberdelincuentes mejor conocidos como Hackers es el beneficio económico, a menudo robando datos o exigiendo rescates, haciendo fraudes o realizando actividades ilícitas en la web. Dentro de los hackers también existen otro tipo de hackers con diferentes motivaciones a las principales ya mencionadas como por ejemplo:

Los hackers de alquiler venden sus servicios a personas que no tienen las habilidades o capacidades para realizar una tarea que conlleve este conocimiento, los hackers de sombrero blanco o gris podrían entrar en esta categoría aunque,

difícilmente, ya que este tipo de hacker solo realiza tareas pequeñas o de bajo perfil para un cierto grupo de público en específico.

Los actores patrocinados por el Estado atacan a las organizaciones para comprometer o robar, cambiar o destruir información con la finalidad de manchar la reputación, generar controversia o desinformar a algún bando o grupo político o gubernamental.

Los hacktivistas tienen motivaciones políticas, sociales o ideológicas y atacan a las víctimas para hacerse publicidad o provocar un cambio, por lo regular este tipo de hackers se dedica a realizar noticias falsas y desinformar a la población o grupos de personas con la finalidad de tener ingresos por parte de sus publicaciones o noticias.

El actor con información privilegiada tanto con intención maliciosa como sin ella. El actor era un empleado actual o antiguo (insider). Este viene siendo un actor interno no malicioso. Aunque al final este tipo de atacantes son los que dejan backdoors o puertas traseras en las redes que al final terminan siendo vulneradas.

Los Script Kiddies son aquellos individuos con habilidades limitadas que utilizan herramientas y scripts preexistentes para llevar a cabo ataques ya que carecen de conocimientos de programación para realizar ciertas tareas. (European Union Agency for Cybersecurity, Enisa Threat Landscape Report) [17] (Flores Quispe, Tipos de hackers) [18]

1.4 ¿Qué es un IDS?

Un IDS o mejor conocido como Sistema de Detección de Intrusos es un sistema ya sea a modo de Hardware o Software encargado de automatizar el proceso de supervisión de tráfico en una red y las actividades que se llevan a cabo dentro de ella, hace todo esto para revisar si existe alguna actividad sospechosa o realizada por actores externos e inclusive internos, de tal forma que con esto se reduce el riesgo de una Intrusion o ataque. (Scarfone & Mell, 2007)[19]

1.5 Clasificación de los IDS

Los IDS se clasifican de dos formas en función de que sistemas son los que vigilan y en función de cómo es que lo hacen.

Para los sistemas que vigilan existen dos tipos, los que analizan una maquina en busca de ataques y los que lo hacen en una subred. Para ello es importante mencionar los IDS basados en Red y los IDS basados en Host.
El IDS basado en Red (NIDS – Network IDS) monitoriza los paquetes que viajan por la red en busca de elementos o patrones que identifiquen un ataque contra algún sistema, este IDS puede colocarse en cualquier host o en una posición que sea factible para analizar todo el tráfico para así analizar el tráfico de diversos dispositivos y no solo uno.

El IDS basado en Maquina (Host IDS) solo se encarga de proteger un sistema y de igual manera busca patrones para detectar una intrusión y alerta o toma medidas cuando detecta los intentos de intrusión.
Para los sistemas que se enfocan en como detectan intrusos también existen dos clasificaciones, las basadas en detectar anomalías, y la que se encarga en detectar el uso indebido del sistema.

1.6 Requisitos de un IDS

Para que un IDS funcione de una manera correcta debe cumplir con las siguientes propiedades. La primera característica y la más importante para un IDS es que debe ejecutarse continuamente y autónomamente, independientemente de que se informe al administrador acerca de un problema o que reaccione de manera automática, su funcionamiento debe ser totalmente autosuficiente.

La segunda que se debe tomar con importancia es el nivel de aceptación de un IDS ya que los mecanismos de detección de intrusos deben ser aceptadas por el personal del entorno. Esto quiere decir que el IDS debe funcionar de manera eficiente sin generar algún problema en la red o generar alertas con falsos positivos constantemente ya que debe haber acción para cada amenaza detectada por el sistema.

Por último un IDS debe tener un alto índice de adaptabilidad, esto significa que debe poder adaptarse a los cambios en el entorno de trabajo. Los IDS deben tener tolerancia a fallos, capacidad de respuesta inesperada y debe ser capaz de reaccionar certeramente a estos últimos.

1.7 TCP/IP

Es importante hablar del modelo TCP/IP debido a que dentro de este modelo se incluye una familia de protocolos que son utilizados dentro de los NIDS (Network Intrusion Detection System) para detectar posibles amenazas, además de usar como fuente de información primordial las características del tráfico de este modelo y la información de los paquetes que circulan por la red.

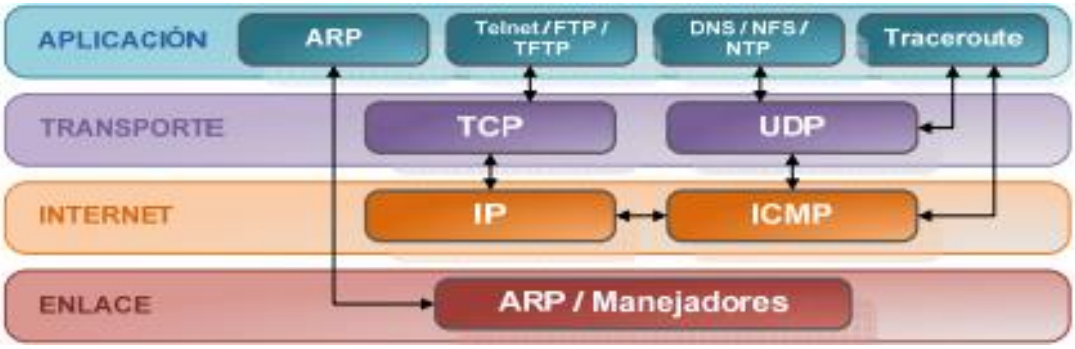


Figura 3. Estructura del stack TCP/IP

TCP/IP se refiere a la pila de protocolos de comunicación, recibe su nombre de los protocolos que le pertenecen, el Protocolo de Control de Transmision (TCP) y el Protocolo de Internet (IP). Como mencionamos anteriormente está constituido por varios protocolos, que principalmente son utilizados y explotados por los actores de amenaza implementando ataques, amenazas y vulnerabilidades en las redes. Los protocolos que lo componen son los siguientes [20] (Lorenzo Fonseca, Método para la detección de Intrusos Mediante redes neuronales basado en la Reducción de Características):

- **Internet Protocol (IP)**. Se encarga de transportar y entregar los paquetes de datos de manera eficiente de una maquina a otra.

- **Internet Control Message Protocol (ICMP).** Se encarga de enviar mensajes de control y error entre dispositivos de red, facilita el enrutamiento y las peticiones de eco.
- **User Datagram Protocol (UDP) y Transmision Control Protocol (TCP).** Envían datos entre aplicaciones usando el protocolo de internet. UDP se encarga de enviar datos de manera rápida y eficiente sin gestión de errores, mientras que TCP garantiza la entrega de datos y a establecer la conexión.
- **Address Resolution Protocol (ARP).** Sen encarga de traducir las direcciones IP a direcciones MAC permitiendo la comunicación entre dispositivos.

El protocolo TCP/IP se encuentra estructurado por cuatro capas:

1. **Capa de enlace.** Responsable de la transmisión de los datos, también se encarga de la encapsulación de datos en tramas para él envió a través de medios físicos.
2. **Capa de red.** Se encarga del envío y recepción de paquetes, se encarga del enrutamiento y el protocolo IP es el principal en esta capa. Aquí también se encuentra el protocolo ARP, ICMP e IGMP.
3. **Capa de transporte.** Se encarga de la comunicación entre aplicaciones finales, en esta capa se regula el flujo de información. Los protocolos principales en esta capa son TCP y UDP.
4. **Capa de aplicación.** Esta capa es la que se utiliza para el acceso a la red, aquí se encuentran protocolos como HTTP, SMTP, FTP, Telnet, SNMP, DNS, entre otros.

La señalización se realiza por medio de los protocolos y por medio de mensajes específicos de las capas que se usan para establecer, mantener y finalizar las conexiones de manera eficiente.

El conjunto de protocolos y servicios del stack de TCP/IP se encarga de dividir la información que debe transmitir en paquetes, a los que les añade información adicional según las capas por las que vaya descendiendo. Finalmente, la última

capa debe componer un paquete que entienda la red física por la que van a ser transmitidos en este nivel, al paquete se le suele denominar trama (frame).

1.7.1 Flags TCP

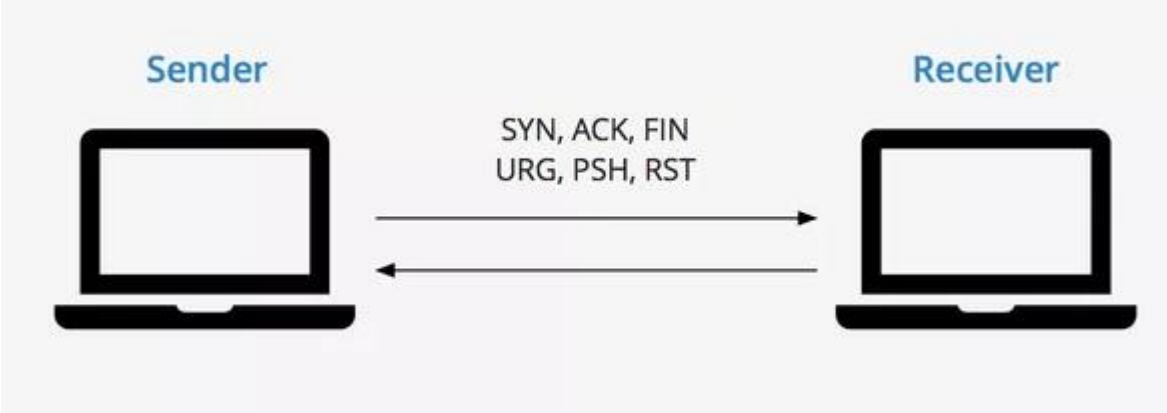


Figura 4. Flags TCP

Los indicadores de conexión TCP se usan dentro de la transferencia de paquetes para indicar los estados de conexión o proporcionar información adicional. Por ello se pueden usar para solucionar problemas de red o controlar las conexiones particularmente. Cada indicador utiliza un bit y la siguiente lista describe el funcionamiento de cada indicador de conexión [21] (TCP flags - keycdn support):

- SYN (Synchronize) El indicador de sincronización se utiliza para establecer un handshake (procedimiento de autenticación) de tres vías. Solo el primer paquete del remitente y del receptor debe tener este indicador establecido.

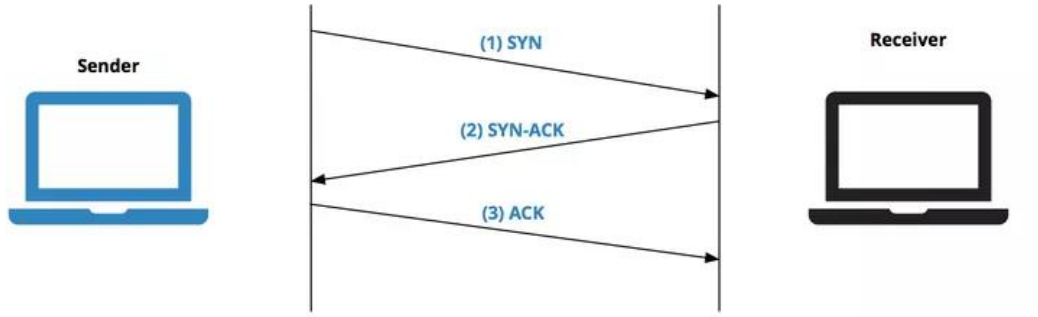


Figura 5. Inicio de sesión de señalización

- ACK (Acknowledgment) Utilizado para la confirmación de recepción de datos. Como podemos ver en el diagrama anterior, el receptor envía un ACK así

como a SYN en el segundo paso para decirle al remitente que recibió su paquete inicial.

- FIN (Finish) Se usa para indicar el final de una conexión. Esto significa que no hay más datos del remitente. Libera los recursos reservados y termina con la conexión.
- RST (Reset) Utilizado para reestablecer las conexiones, este indicador de reinicio se envía desde el receptor al remitente cuando se envía un paquete a un host en particular que no lo esperaba.
- URG (Urgent) Se utiliza para notificar al receptor que debe procesar un paquete en específico antes de procesar todos los demás paquetes.
- PSH (Push) Indica al receptor que debe procesar los segmentos de paquete recibidos y que no debe almacenarlos en un buffer.
- ECE (Explicit Congestion Notification) Esta bandera es responsable de indicar si la conexión TCP permite notificación de congestión explícita para no descartar paquetes.
- CWR (Congestion Windows Reduced) El host emisor utiliza la bandera reducida de la ventana de congestión para indicar que recibió un paquete con el ECE. Reduce a la mitad la ventana de envió, esto con la intención de ralentizar él envió de información.
- NS (Nonce Sum) Sigue siendo una bandera experimental utilizada para ayudar a proteger contra envió de paquetes de forma accidental y maliciosa por parte del remitente.

Los atacantes pueden manipular el comportamiento de las conexiones por medio de estas banderas (flags) y asi explotar vulnerabilidades en los sistemas mediante la manipulación de los bits de control en los encabezados para realizar diferentes tipos de ataques como por ejemplo de denegación de servicio, secuestro de conexiones, reinicio y cierre de conexiones, entre otros. Por lo tanto, es importante que el administrador de red y el equipo de seguridad estén atentos a este tipo de actividad maliciosa, cambios de estado y asi tomen medidas para protegerse contra ellas.

1.7.2 Puertos

Un puerto es un canal de comunicación e identificador de 16 bits, el cual sirve para la identificación en ambos protocolos de transporte (TCP/UDP), su principal función es distinguir diferentes aplicaciones y servicios que se ejecutan en dispositivos finales en una red. Es importante mencionar que dos diferentes servicios no pueden hacer el uso del mismo puerto en simultaneo, si un servicio ha sido iniciado usando un puerto especifico, este no podrá ser usado por otro proceso hasta que el servicio inicial se haya cerrado. Como habíamos mencionado cada puerto está representado por una dirección de 16 bits, existen tres categorías de puertos [22] (Redes y Seguridad, Katz, 2013, p. 116).

- Puerto 0. Reservado
- Puertos 1 a 1023: puertos conocidos (well-known ports). Utilizado para procesos del sistema y servicios comunes de internet.

Puerto	TCP	UDP	Nombre	Descripción
1	✓	✓	tcpmux	Multiplexor TCP
5	✓	✓	rje	Entrada de tarea remota (remote job entry)
7	✓	✓	echo	Protocolo Echo
9	✓	✓	discard	Protocolo Discard (evaluación de conexiones)
11	✓	✓	systat	Información del sistema (enumera los puertos conectados)
13	✓	✓	daytime	Protocolo Daytime: indica fecha y hora
17	✓	✓	qotd	Envía la cita del día (quote of the day)
18	✓	✓	msp	Protocolo de envío de mensajes

Puerto	TCP	UDP	Nombre	Descripción
19	✓	✓	chargen	Protocolo Chargen: envía una cadena infinita de caracteres
20	✓		ftp-data	Transmisión de datos FTP
21	✓	✓	ftp	Conexión FTP
22	✓	✓	ssh	Servicio Secure Shell
23	✓		telnet	Servicio Telnet
25	✓		smtp	Simple Mail Transfer Protocol
37	✓	✓	time	Protocolo de tiempo legible de forma mecanizada
39	✓	✓	rlp	Protocolo de envío de recursos (Resource Location Protocol)
42	✓	✓	nameserver	Servicio de nombres
43	✓		nicname	Servicio de directorio WHOIS
49	✓	✓	tacacs	Terminal Access Controller Access Control System
50	✓	✓	re-mail-ck	Protocolo de verificación de correo remoto (Remote Mail Checking)
53	✓	✓	domain	Resolución de nombres por DNS
67		✓	bootps	Protocolo Bootstrap (servidor)
68		✓	bootpc	Protocolo Bootstrap (cliente)
69		✓	tftp	Protocolo Trivial de Transferencia de Archivos (Trivial File Transfer Protocol)
70	✓		gopher	Búsqueda de documentos

Puerto	TCP	UDP	Nombre	Descripción
71	✓		genius	Protocolo Genius
79	✓		finger	Proporciona información de contacto de usuarios
80	✓		http	Protocolo de Transferencia de HiperTexto (Hypertext Transfer Protocol)
81	✓			Torpark: Onion-Routing (no oficial)
82		✓		Torpark: Control (no oficial)
88	✓	✓	kerberos	Sistema de autenticación de red
101	✓		hostname	Servicios de nombres de host (NIC Host Name)
102	✓		Iso-tsap	Protocolo ISO-TSAP
105	✓	✓	csnet-ns	Servidor de correo
107	✓		rtelnet	Telnet remoto
109	✓		pop2	Post Office Protocol v2 para comunicación de correo electrónico
110	✓		pop3	Post Office Protocol v3 para comunicación de correo electrónico
111	✓	✓	sunrpc	Protocolo RPC para NFS
113		✓	auth	(Antiguo) servicio de autenticación
115	✓		sftp	Protocolo de transferencia de archivos seguros o Simple File Transfer Protocol (versión simplificada de FTP)
117	✓		uucp-path	Transmisión de datos entre sistemas Unix

Puerto	TCP	UDP	Nombre	Descripción
119	✓		nntp	Transmisión se noticias en Newsgroups
123		✓	ntp	Protocolo de sincronización de tiempo
137	✓	✓	netbios-ns	NETBIOS Servicio de nombres
138	✓	✓	netbios-dgm	NETBIOS Servicio de envío de datagramas
139	✓	✓	netbios-ssn	NETBIOS Servicio de sesiones
143	✓	✓	imap	Internet Message Access Protocol para comunicación de correo electrónico
161		✓	snmp	Simple Network Management Protocol
162	✓	✓	snmptrap	Simple Network Management Protocol Trap
177	✓	✓	xdmcp	X Display Manager
179	✓		bgp	Border Gateway Protocol
194	✓	✓	irc	Internet Relay Chat
199	✓	✓	smux	SNMP UNIX Multiplexer
201	✓	✓	at-rtmp	Enrutamiento AppleTalk
209	✓	✓	qmtp	Quick Mail Transfer Protocol
210	✓	✓	z39.50	Sistema de información bibliográfico
213	✓	✓	ipx	Internetwork Packet Exchange
220	✓	✓	imap3	IMAP v3 para comunicación de correo electrónico
369	✓	✓	rpc2portmap	Coda Filesystem Portmapper

Puerto	TCP	UDP	Nombre	Descripción
370	✓	✓	codaauth2	Servicio Coda Filesystem Authentication
389	✓	✓	ldap	Lightweight Directory Access Protocol
427	✓	✓	svrloc	Service Location Protocol
443	✓		https	HTTPS (HTTP a través de SSL/TLS)
444	✓	✓	snpp	Simple Network Paging Protocol
445	✓		microsoft-ds	SMB a través de TCP/IP
464	✓	✓	kpasswd	Modificación de contraseña para Kerberos
500		✓	isakmp	Protocolo de seguridad
512	✓		exec	Remote Process Execution
512		✓	comsat/biff	Mail Client y Mail Server
513	✓		login	Inicio de sesión en ordenador remoto
513		✓	who	Whod User Logging Daemon
514	✓		shell	Remote Shell
514		✓	syslog	Servicio Unix System Logging
515	✓		printer	Servicios de impresión Line Printer Daemon
517		✓	talk	Talk Remote Calling
518		✓	ntalk	Network Talk
520	✓		efs	Extended Filename Server

Puerto	TCP	UDP	Nombre	Descripción
520		✓	router	Routing Information Protocol
521		✓	ripng	Routing Information Protocol para IPv6
525		✓	timed	Servidor de tiempo
530	✓	✓	courier	Courier Remote Procedure Call
531	✓	✓	conference	Chat a través de AIM y IRC
532	✓		netnews	Servicio Netnews Newsgroup
533		✓	netwall	Broadcast de emergencia
540	✓		uucp	Unix-to-Unix Copy Protocol
543	✓		klogin	Kerberos v5 Remote Login
544	✓		kshell	Kerberos v5 Remote Shell
546	✓	✓	dhcpv6-client	DHCP v6 Client
547	✓	✓	dhcpv6-server	DHCP v6 Server
548	✓		afpovertcp	Apple Filing Protocol a través de TCP
554	✓	✓	rtsp	Control de streams
556	✓		remotefs	Remote Filesystem
563	✓	✓	nntps	NNTP a través de SSL/TLS
587	✓		submission	Message Submission Agent
631	✓	✓	ipp	Internet Printing Protocol
631	✓	✓		Common Unix Printing System (no oficial)

Puerto	TCP	UDP	Nombre	Descripción
636	✓	✓	ldaps	LDAP a través de SSL/TLS
674	✓		acap	Application Configuration Access Protocol
694	✓	✓	ha-cluster	Servicio Heartbeat
749	✓	✓	kerberos-adm	Kerberos v5 Administration
750		✓	kerberos-iv	Servicios Kerberos v4
873	✓		rsync	Servicios de transmisión de datos rsync
992	✓	✓	telnets	Telnet a través de SSL/TLS
993	✓		imaps	IMAP a través de SSL/TLS
995	✓		pop3s	POP3 a través de SSL/TLS

Tabla 1. Puertos bien conocidos

- Puertos 1024 a 49151: puertos registrados (registered ports). Puertos registrados por la IANA (Internet Assigned Numbers Authority/Autoridad de Asignación de Números de Internet) de tal manera que permite usar los puertos en internet.

Puerto	TCP	UDP	Nombre	Descripción
1080	✓		socks	SOCKS Proxy
1433	✓		ms-sql-s	Microsoft SQL Server
1434	✓	✓	ms-sql-m	Microsoft SQL Monitor
1494	✓		ica	Citrix ICA Client
1512	✓	✓	wins	Windows Internet Name Service

Puerto	TCP	UDP	Nombre	Descripción
1524	✓	✓	ingreslock	Ingres DBMS
1701		✓	l2tp	Layer 2 Tunneling Protocol/Layer 2 Forwarding
1719		✓	h323gatestat	H.323
1720	✓		h323hostcall	H.323
1812	✓	✓	radius	Autenticación RADIUS
1813	✓	✓	radius-acct	Acceso RADIUS
1985		✓	hsrp	Cisco HSRP
2008	✓			Teamspeak 3 Accounting (no oficial)
2010		✓		Teamspeak 3 Weblist (no oficial)
2049	✓	✓	nfs	Network File System
2102	✓	✓	zephyr-srv	Zephyr Server
2103	✓	✓	zephyr-clt	Zephyr Client
2104	✓	✓	zephyr-hm	Zephyr Host Manager
2401	✓		cvspserver	Concurrent Versions System
2809	✓	✓	corbaloc	Common Object Request Broker Architecture
3306	✓	✓	mysql	Servicio de bases de datos MySQL (también para MariaDB)
4321	✓		rwhois	Remote Whois Service
5999	✓		cvsup	CVSup

Puerto	TCP	UDP	Nombre	Descripción
6000	✓		X11	Servicios X Windows System
11371	✓		pgpkeyserver	Keyserver público para PGP
13720	✓	✓	bprd	Symantec/Veritas NetBackup
13721	✓	✓	bpdbm	Symantec/Veritas Database Manager
13724	✓	✓	vnetd	Symantec/Veritas Network Utility
13782	✓	✓	bpcd	Symantec/Veritas NetBackup
13783	✓	✓	vopied	Symantec/Veritas VOPIE
22273	✓	✓	wnn6	Conversión Kana/Kanji
23399				Skype (no oficial)
25565	✓			Minecraft
26000	✓	✓	quake	Quake y otros juegos multijugador
27017				MongoDB
33434	✓	✓	traceroute	Seguimiento de red

Tabla 2. Puertos registrados

- Puertos 49152 a 65535: puertos dinámicos o privados (ephemeral ports). Estos puertos son utilizados por el sistema operativo cliente al realizar una conexión remota

Dentro de estos puertos es importante recalcar que tenemos puertos de origen y destino los cuales junto con las direcciones IP establecen la conexión y transmiten la información de manera eficiente.

Para entender un poco más este concepto se describe la funcionalidad de los puertos de origen y destino:

- Puerto de origen: Es el puerto utilizado por la aplicación/servicio que inicia la comunicación. Su selección es dinámica por el dispositivo emisor y se incluye en el encabezado del paquete saliente. El puerto de origen permite al dispositivo receptor saber a qué aplicación/servicio enviar la respuesta.
- Puerto de destino: Es el puerto al que se envían los datos y los recibe. El puerto de destino ayuda al dispositivo receptor a dirigir los datos entrantes a la aplicación/servicio correcto.

1.7.3 Fragmentación de paquetes

La fragmentación de paquetes es necesaria cuando cada capa de la arquitectura de red impone un tamaño máximo de bytes (MTU/ unidad de transmisión máxima) a sus paquetes o mensajes para poderlos transmitir, los límites de tamaño máximo van relacionados con el protocolo que se utiliza en cada capa, el tipo de sistema operativo utilizado, el cumplimiento de algún estándar. [23] (Acosta, Red Inalámbrica de Sensores con topología lineal sin capa de red).

Decimos que existe fragmentación de paquetes cuando los routers comprueban el tamaño de cada paquete que reciben y según su tamaño determinado por el siguiente router a recibir el paquete. Si el paquete excede la unidad máxima de medida establecida por el router receptor, el router que envía el paquete se deberá encargar de dividir este en dos o más paquetes y cada uno con su propio encabezado.

Cada nuevo paquete tiene un encabezado copiado del paquete original (para que todos los paquetes tengan las direcciones IP originales de fuente y destino) con algunos cambios importantes. El enrutador edita ciertos campos en el encabezado IP para indicar que los paquetes están fragmentados y deben volver a ensamblarse, cuántos paquetes hay y en qué orden se envían. [24] (¿Qué es MTU (Unidad de Transmisión Máxima)?)

La fragmentación de un paquete se constituye de la siguiente manera. La cabecera, que suele tener un tamaño de 20 bytes y hasta 60 bytes, contendrá la siguiente información [25] (Álvaro. M, 2019):

- Campo Identification (Identificador de fragmento) . Este campo tiene un tamaño de 16 bits y sirve para identificar a los paquetes durante la fragmentación. Se asigna un identificador único a todos los paquetes
- Campo Flags. Tiene un tamaño de 3 bits, donde cada bit representa un flag.
  - El primer bit no se utiliza y siempre tiene un valor de 0.
  - El segundo bit lleva la denominación de Don't Fragment (DF) y se utiliza para indicar que el paquete NO debe ser fragmentado en su trayecto al dispositivo final.
  - El tercer bit lleva la denominación de «More Fragments (MF)», sirve para indicar que existen más paquetes por ser recibidos. Si el valor es 0, hace referencia a que ese paquete es el último a recibir.
- Campo Fragment Offset. Tiene un tamaño de 13 bits y se utiliza para ensamblar nuevamente el paquete.

En la siguiente imagen (Figura 6) podemos observar de una manera más grafica los tres campos principales que ayudan en el proceso de la fragmentación de paquetes.

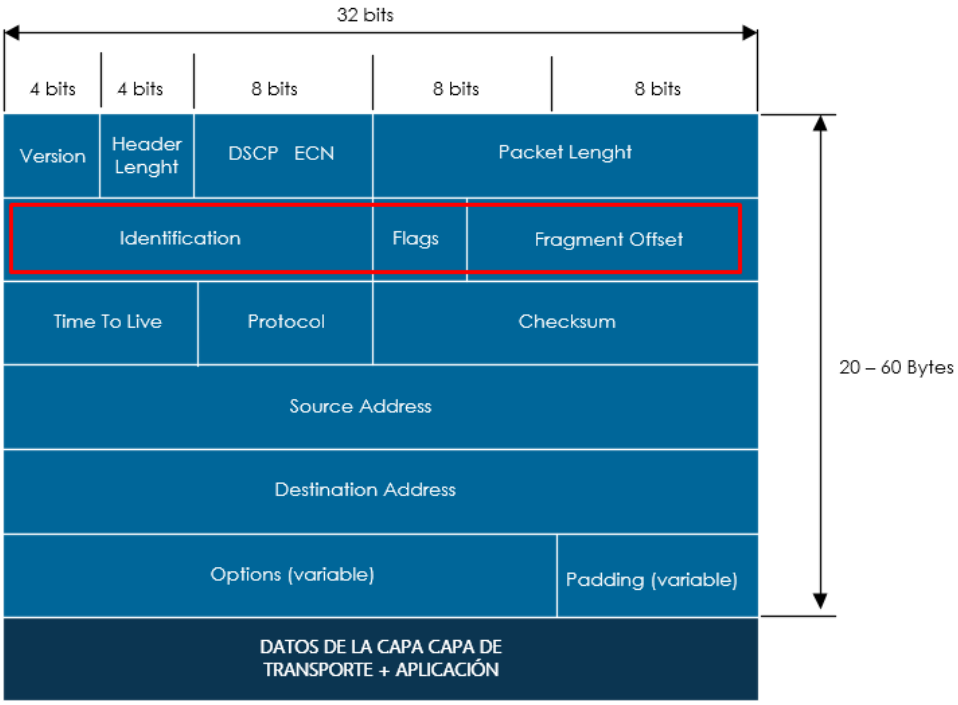


Figura 6. Campos principales de la fragmentación

Como mencionamos anteriormente la fragmentación tiene un límite de unidad de transmisión máxima y este límite depende del protocolo que se utilice para la transmisión de estos datos. En la siguiente imagen podemos observar algunos de los MTU más comunes en cuestión de los protocolos utilizados.

TECNOLOGÍA DE CAPA 2	MTU
16 Mbps Token Ring	*17914*
4 Mbps Token Ring	*4464*
ATM (AAL5)	*9180*
FDDI	*4352 (RFC 1188)*
802.11	*2346 (variable)*
Frame Relay (L2TP)	*1564 (RFC 3070)*
Ethernet	*1500*
IEEE 802.3/802.2	*1492*
PPPoE	*1492*
X.25	*576*

Tabla 3. MTU de múltiples tecnologías

Desde el punto de vista de diseño es conveniente usar un servidor de alto rendimiento junto con interfaces de red de alta velocidad como lo son Fast ethernet (100mbps y su MTU es de 1500bytes), gigabit ethernet (1 Gbps y su MTU es de 9000bytes) o 10GB ethernet (10Gbps y su MTU es de 9000bytes), esto dado que los IDS operan con los envíos de altas velocidades que pueden llegar a requerir algunas de estas conexiones antes mencionadas .

En pocas palabras la fragmentación se puede definir como la división de los paquetes de datos en paquetes más pequeños para que puedan ser transmitidos a través de la red.

1.8 IDS basado en red

La mayoría de IDS que hoy en día se encuentran en el mercado son basados en red, esto también son conocidos como NIDS (network based IDS) por lo general están compuestos por un sensor sencillo y se coloca en uno o varios puntos clave dentro de la red, se encargan de monitorizar el tráfico dentro de las redes, analizando paquetes y reportando inconsistencias a la consola central. Es facil resguardar los sensores ya que son limitados a correr el IDS. Gran parte de estos

sensores para que su ejecución sea de manera silenciosa, que sigan un orden y una ubicación, todo esto con la finalidad de que sea dificil para el atacante detectar la presencia de un IDS.

Los NIDS pueden detectar múltiples ataques en una misma red, aunque por lo general solo se ejecuten en un punto de la red. Para lograr su objetivo una de las interfaces trabaja en modo promiscuo con la finalidad de ir capturando y analizando todos los paquetes que se envían por la red en busca de patrones que delaten un ataque.

Uno de los lugares más comunes para implementar un IDS es cerca del firewall. Dependiendo del tráfico a monitorizar, se coloca antes o despues de un firewall para analizar el tráfico sospechoso, originado desde dentro o desde fuera de la red. Cuando se coloca en el interior, el IDS debe estar cerca de la DMZ (Zona Desmilitarizada). Aunque la mejor opción en cuanto a seguridad es utilizar una defensa en capas mediante la implementación de un IDS delante del firewall y otro detrás del firewall en la red. La siguiente imagen muestra la colocación de IDS en capas, de modo que se encuentra un IDS antes y despues del firewall.

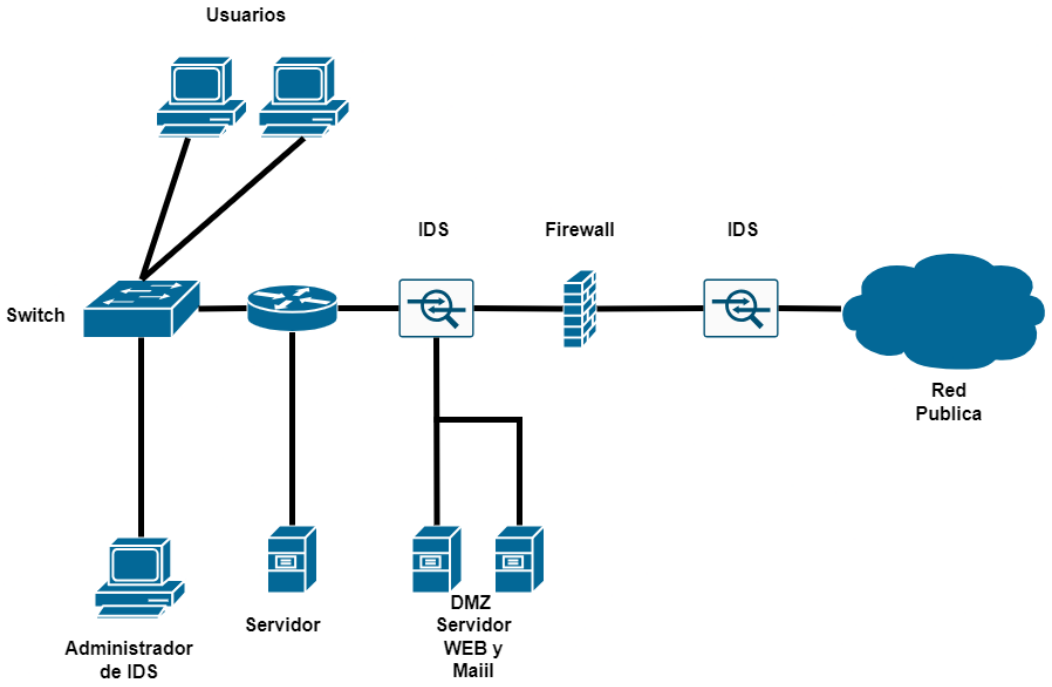


Figura 7. Defensa en capas (IDS)

Hablando de patrones nos referimos a que puede ser cualquier fragmento de datos que se analiza dentro de los paquetes de información que viajan por los protocolos TCP/IP y cada uno puede contener información que puede representar un ataque o actividad inusual, los casos más habituales representativos de ataques por patrones son:

- Campos de fragmentación, los cuales contienen 16 bits dentro de la cabecera IP los cuales están reservados para información sobre el nivel de fragmentación del paquete. Uno de estos bits no tiene uso y 13 muestran el movimiento de los paquetes que han sido fragmentados. Los otros dos bits indican que el paquete no está fragmentado por routers intermedios o que el paquete está fragmentado y no es el último paquete por recibir. Comúnmente se han utilizado valores incorrectos para los parámetros de fragmentación de paquetes que causan problemas graves. También se utiliza para evitar la denegación de servicio a un sistema y, más recientemente, para obtener la versión del sistema operativo que se ejecuta en un host en particular.
- Dirección origen y destino de los paquetes, la dirección de la máquina que envía el paquete y la dirección de la máquina que recibe el paquete también son campos de interés para detectar intrusiones en un sistema o red. Lo que se debe considerar es que muy probablemente el tráfico que se origina en la DMZ (Zona Desmilitarizada) y tiene como destino la red protegida y estos paquetes sean un intento de violación a las políticas de seguridad.
- Puerto origen y destino, los puertos de origen y destino de los paquetes son buenos indicadores de actividad sospechosa en la red. Estos puertos identifican los puntos de conexión de las comunicaciones entre dispositivos. Actividades sospechosas pueden ser detectadas mediante el seguimiento de patrones por medio de estos puertos. La violación de las políticas de seguridad y el tráfico generado en la red de cierto tipo generado por amenazas como por ejemplo la presencia de caballos de Troya, ciertos tipos de escaneo de puertos o la presencia de servidores no autorizados dentro de nuestra red, así como intentos de obtener acceso no autorizado a servicios

en los sistemas pueden indicar vulnerabilidades o brechas que deben ser atendidas de inmediato.

- Flags TCP, uno de los campos de análisis en el encabezado TCP contiene 6 bits (urg, ack, psh, rst, syn, fin), cada uno con un propósito diferente (por ejemplo, el bit syn se usa para establecer una nueva conexión, el bit final se usa para establecer la conexión). Obviamente, cada uno de estos bits tiene un valor de 0 o 1, pero eso por sí solo generalmente no dice mucho (para bien o para mal) sobre el remitente. Sin embargo, ciertas combinaciones de valores tienden a resultar muy sospechosas. Por ejemplo, una trama con los dos bits que mencioné anteriormente (syn y end) activos al mismo tiempo indica un intento de abrir y cerrar una conexión al mismo tiempo. Para entender la importancia de estos bits de control no debemos olvidar entender y analizar estas Flags para evitar uno de los problemas de seguridad más conocidos.
- Campo de datos, de hecho, los campos de datos de los paquetes que circulan a través de una red son el lugar más probable para identificar ataques a un sistema. Esto se debe a que es probable que un firewall detenga un datagrama o paquete con un encabezado "sospechoso", pero es poco probable que un firewall lo detenga para analizar el contenido de los datos del datagrama. Por ello es esencial que los sistemas de seguridad analicen el contenido de los paquetes para identificar las amenazas

Podríamos decir que los NIDS solamente funcionan con detección de patrones, pero en realidad no es asi, ya que estos IDS está basado en la detección de anomalías. Aunque la intrusión genere comportamientos anormales susceptibles a que sean detectados y mitigados, estos sistemas no detectaran la intrusión hasta que esta se encuentre ya activa. Este problema hace que este tipo de IDS funcione siguiendo modelos de detección de intrusos indebidos.

Dentro de los NIDS se encuentran los famosos Honeypots (redes de atracción para atacantes). Este tipo de redes están diseñadas para ser atacadas, formadas por

sistemas que una vez comprometidos permite analizar y capturar las acciones del atacante e identificar sus técnicas y sus objetivos.

Dentro de las “Honeypots” hay dos aspectos fundamentales, el primero es el control de flujo de los datos, este es vital para garantizar la seguridad una vez que el sistema haya sido penetrado y no se use como plataforma para atacar otras máquinas, esto quiere decir que la honeypot debe mantenerse controlada y aislada del resto de la red. El segundo aspecto por tomar con mayor importancia es la captura de datos, es imprescindible monitorear todas las actividades que realiza el atacante dentro de esta red.

El objetivo principal de estos sistemas es conocer los movimientos y los ataques que realizan los hackers para poder para poder extrapolarlos en sistemas reales, además del correcto funcionamiento de esta red es necesario una correcta recolección de datos que genere el atacante, de manera que cada movimiento sea anotado y sin que se dé cuenta el atacante. Además estos datos que lleguen a recabarse deberán ser almacenados externamente a esta red para evitar posible destrucción o comprometer este tipo de pruebas. [26] (Villalon Huerta, *SEGURIDAD EN UNIX Y REDES*)

Los NIDS actualmente son los que más se usan en sistemas de defensa y prevención, pero como casi cualquier herramienta de seguridad estos sistemas no llegan a ser todo para proteger una red por ello su aplicación debe ser complementada con algún firewall o algún otro tipo de herramienta o filtro, ya sea físico o digital.

Ventajas de los NIDS:

- Los NIDS que llegan a estar bien configurados pueden llegar a monitorizar un amplio campo dentro de la red.
- Los NIDS son dispositivos que normalmente escuchan a una red sin interferir con su operación cotidiana. De esta manera hay posibilidad de readaptar la red de ser necesario y con un mínimo esfuerzo.

- Los NIDS en su mayoría son muy seguros contra los ataques e invisibles a los atacantes.

Desventajas de los NIDS:

- A veces los NIDS tienen complicaciones al procesar una gran cantidad de paquetes dentro de una red muy grande u ocupada, por lo que puede fallar al detectar un ataque en periodos con un tráfico de datos muy alto.
- Los NIDS tienen que ser ubicados adecuadamente en los puntos de la red despues de pasar por una VPN ya que no pueden analizar información cifrada, por lo que es dificil usarlos en organizaciones que utilizan VPN y colocarlo de manera incorrecta dentro de la red hará que falle su eficacia.
- Muchos NIDS no pueden interpretar que un ataque haya sido exitoso, solo lo hacen cuando uno ha sido iniciado, por lo que después de haber sido detectado el administrados deberá investigar cada ataque al host para verificar si ha sido comprometido o no.
- Algunos NIDS tienen problemas con ataques basados en red ya que si estos involucran paquetes fragmentados, estos harán que el IDS no reaccione o no se active y no funcione con éxito. [27] (Scarfone & Mell, Intrusion Detection Systems (IDS))

1.9 Herramientas de IDS

Para analizar el tráfico en una red y poder monitorizar esta misma es necesario implementar los rastreadores de paquetes, dentro de los cuales se pueden detectar los diferentes tipos de problemas que pueden presentarse, todo esto implementando los famosos cuellos de botella, para poder capturar toda la información que viaja por la red. Debido a la gran importancia que existe actualmente cabe mencionar algunos de los más famosos y utilizados en la actualidad.

SNORT

Es utilizado como un NIDS. Implementa un motor de detección de ataques y barrido de puertos que registra, alerta y responde ante las anomalías con un análisis de

patrones que corresponden a ataques, barridos o aprovechamiento de vulnerabilidades en tiempo real. Es un software de código abierto y acceso libre que funciona en plataformas Windows y UNIX/Linux. Ofrece una amplia gama de filtros y patrones que la comunidad va a portando asi como una actualización constante para las amenazas previamente mencionadas que se detectan a través de boletines de seguridad. Snort es la parte medular de este proyecto ya que será implementado, además de que se hablará de una manera más específica en un próximo capitulo y hablaremos de su funcionalidad, implementación y relevancia en este trabajo.

Security Onion

Es distribuido por Linux, su objetivo monitorear la red y la administración de registros . Está basado en Ubuntu y se compone de muchas herramientas IDS como Snort, Suricata, Bro, Sguil, Squert, Snorby, ELSA, Xplico, NetworkMiner y muchas otras. Proporciona alta visibilidad y contexto al tráfico de red, alertas y actividades malintencionadas. Requiere de una buena gestión por parte del administrador de sistemas al verificar alertas, supervisar la red y actualizar las reglas, su principal función es capturar paquetes, IDS basado en host y red, además de una herramienta de análisis.

Open WIPS-NG

Al igual que Snort, es un sistema de software libre de detección de intrusos y prevención de intrusiones inalámbricas basado en sensores, servidores e interfaces. Utiliza funciones y servicios integrados en Aircrack-NG para escanear, detectar y prevenir intrusiones.

Suricata

También es un IDS de libre acceso rápido y robusto. El motor de Suricata es capaz de detectar intrusos en tiempo real, prevenir intrusiones en línea y monitorear la seguridad de la red. Consta con modulos como captura, recopilación, decodificación y salida. Su funcionalidad radica en capturar el tráfico dentro de la red antes de la decodificación dentro de la misma. A diferencia de Snort configura los flujos separados de captura y especifica como se separa el flujo.

BroIDS

Es un analizador de tráfico pasivo usado para la recopilación de mediciones de red, realizar investigaciones forenses y otras funciones. Incluye archivos de registro para identificar las actividades de red. Además analiza y detecta amenazas como malware, vulnerabilidades de software, ataques de fuerza bruta y validación de certificados.

OSSEC

Este es un IDS centralizado que realiza tareas como análisis de registro, auditoría de estado, monitoreo de registro de Windows, detección de rootkits, alertas basadas en tiempo y respuestas activas, lo que permite a los administradores monitorear de cerca múltiples sistemas. Tiene una arquitectura multiplataforma.

Open Source Tripwire

Su propósito es detectar cambios en el sistema de archivos. Sigue las instrucciones del administrador para escanear el sistema de archivos y guardar información sobre cada archivo en una base de datos. Si el archivo cambia durante escaneos futuros, los resultados se compararán con los valores guardados y se informará cualquier cambio.

AIDE

Es considerado como una de las herramientas más poderosas para monitorear cambios en sistemas UNIX o Linux. AIDE crea una base de datos a través de reglas de expresión regular que encuentra en los archivos de configuración . Al inicializar la base de datos, se verifica la integridad de los archivos.

CAPITULO 2

Capítulo 2 Snort

2.1 ¿Qué es Snort?

Snort es un IDS basado en red, o mejor dicho, es un NIDS de los más comunes y utilizados actualmente en diversas redes debido a que Snort es clasificado como un Sniffer de paquetes eficiente y con un gran repositorio de reglas para detectar intrusiones dentro de las redes. Snort es famoso por implementar un escaneo e identificación de puertos que permite alertar, registrar y tomar acciones ante xalguna amenaza en la red o uso indebido y manejando un tráfico de datos sumamente moderado. [28] (Whitman & Mattord, Principles of Information Security 2009). Estos usos indebidos o sospechosos, se ven reflejados en una base de datos que contiene patrones de ataques, los análisis de puertos, intentos de explotación de vulnerabilidades, análisis de protocolos, entre muchos otros, esta base de datos también se encuentra disponible desde la página web oficial de SNORT (https://www.snort.org) , además se pueden generar bases de datos con patrones enfocados a los diversos ataques que existen (por ejemplo, ataques a servidores, ataques DoS y DDoS, exploits, backdoors, Spoofing, etc).

El archivo de base de datos utilizado dentro del entorno de SNORT será base para la correcta funcionalidad del IDS, SNORT es de facil configuración, puede analizar el tráfico en la red en tiempo real, asi como registrar los paquetes que viajan por la red y es un software gratuito (OPEN SOURCE) lo que hace que SNORT sea una de las mejores elecciones en cuanto a IDS, además de que está avalado e implementado por CISCO. Además Snort es un IDS multiplataforma por lo cual puede ser implementado bajo sistemas operativos funcionales como Linux o Windows que son los más usados dentro de los dispositivos pertenecientes a una red, además de que está basado en las librerías PCAP e implementadas para Linux como libpcab y para Windows como winpcap, es indispensable la instalación de estas librerías en el dispositivo en el cual se vaya a instalar Snort, esta librería está escrita en C y en general es la encargada de capturar los paquetes que viajan por la red, además captura interfaces, se puede configurar en modo normal o modo promiscuo la tarjeta, tiene la capacidad de reestructurar paquetes que han sido

fragmentados para obtener la información del paquete original, también se encarga de filtrar el tráfico en función del protocolo o configuración y también cuenta con funciones para el manejo de archivos [29] (SISTEMAS DETECTORES DE INTRUSOS Y ANALISIS DE FUNCIONAMIENTO DEL PROYECTO DE CODIGO ABIERTO SNORT - 2011).

Una vez instaladas las librerías y el programa ya pueden ponerse en funcionamiento, pero hay que tener cuidado ya que una mala configuración inicial puede hacer que no exista una detección de intrusos. A veces se suele tener una pequeña idea errónea la cual es que entre más patrones tenga la base de datos de patrones el sistema proporcionara mejores resultados, pero esto es algo equivoco ya que no todos los ataques que el sistema Snort llega a detectar pueden producirse en el segmento de red donde se encuentra ubicado el sistema y con mayor razón si se encuentra en una DMZ (Zona desmilitarizada) donde solamente se ofrece servicio web. También es necesario estudiar los patrones que se llegan a detectar al momento de capturar los paquetes, ya que en múltiples ocasiones, el sistema puede detectar falsos positivos o falsos negativos, una solución a este problema es la reconfiguración de la BD para eliminar los patrones que generan estas alarmas erróneas.

Para comprender los falsos positivos y falsos negativos que nos podria mostrar snort es necesario explicarlos de la siguiente manera [43] (SNORT User ’s Manual 2.9.16):

- Falsos positivos: Ocurren cuando el sistema identifica una actividad legitima como maliciosa, es decir detecta un paquete como una amenaza cuando no lo es. Esto puede desencadenar una sobrecarga o saturación de alertas lo que puede ocasionar que se ignoren o no se identifiquen las amenazas reales. Lo que ocasiona regularmente los falsos positivos son:
    - Reglas insuficientes o mal diseñadas
    - Trafico que es legítimo dentro de la red pero viaja de manera inusual
    - Actualizaciones o creación de nuevas de reglas
- Falsos negativos: Ocurren cuando una verdadera amenaza no es detectada por Snort y en consecuencia no genera ninguna alerta debido a que no esta

detectando amenaza alguna, esto es peligroso ya que múltiples ataques pueden pasar desapercibidos. Lo que ocasiona regularmente un falso negativo es:

- Reglas mal diseñadas o desactualizadas
- Ataques más sofisticados que pasen desapercibidos como la fragmentación de los ataques en partes más pequeñas
- Falta de actualización de las reglas
- Ignorar el cambio, actualización o reemplazo de alguna regla ante nuevas amenazas

En pocas palabras hay que adaptar bien el IDS al entorno de trabajo y ubicarlo en una posición estratégica dentro de la red, además de establecer una buena base de datos con patrones eficientes para detectar los posibles ataques, aunque conlleve tiempo, lo mejor en la práctica es dedicarle tiempo a esta base de datos para evitar una detección falsa y tener una eficacia en la detección de cualquier tipo de ataque.

Snort genera archivos en los cuales encontraremos un archivo denominado alert con las actividades que se registren durante la detección, la extensión de este archivo es .log y se ubica por default en el directorio /var/log/Snort o si se configura, se encontrara en el directorio especificado, también existe el “Packet login” que son subdirectorios en los cuales los nombres son las direcciones IP de los dispositivos los cuales se detecta alguna actividad, además es configurable.

Si dejamos que el IDS analice el tráfico antes de ser filtrado en el firewall, se detectan todos los paquetes que se lanzan a la red y esto sería un riesgo debido a que no hay filtro para detener las actividades de un atacante, es parte del sistema la detección de estos intentos de ataque, aunque también cabe recalcar la importancia de detectar el tráfico sospechoso que el cortafuegos permite pasar y que puede comprometer la red o los dispositivos dentro de ella. Por ello es recomendable emplear el IDS en una zona protegida, aunque los ataques que no lleguen al mismo serán registrados en los archivos .log e inclusive pueden ser neutralizados por el mismo sistema. [30] (Intrusion Detection System buyer’s guide, 1999)

2.2 Componentes de Snort

Snort está creado de manera modular en donde cada componente tiene aplicaciones concretas y todas ellas permiten realizar el análisis de trafico de manera efectiva en tiempo real además de ser flexible y adaptable al tráfico de la red donde se está implementando, está constituido por los siguientes componentes [33] (Snort. preprocesadores (i) parte. 2009):

- Sensor/Decodificador de paquetes

  Es el encargado de capturar la información que viaja por la red o host para su posterior análisis. Se conforma por una serie de decodificadores que se encargan de clasificar el tráfico que se captura e identifica según el protocolo por el que viaja para facilitar el análisis al preprocesador. La identificación de los paquetes incluye los datos de la cabecera y los de su contenido.
- Preprocesador

  Recibe los paquetes tal cual los manda el decodificador. Se encarga de manipular los paquetes recibidos de una manera eficiente para su posterior análisis en el motor de análisis y detección, haciendo que el tráfico quede ordenado y se apliquen con éxito las reglas para identificar un ataque en específico.
- Motor de análisis y detección

  Verifica la información que se obtiene del preprocesador para su análisis y comparación con la información de las reglas definidas para asi poder crear una alerta de ser necesario. Esta es la parte más importante de Snort ya que es el módulo en donde se detectan las amenazas y vulnerabilidades.
- Reglas

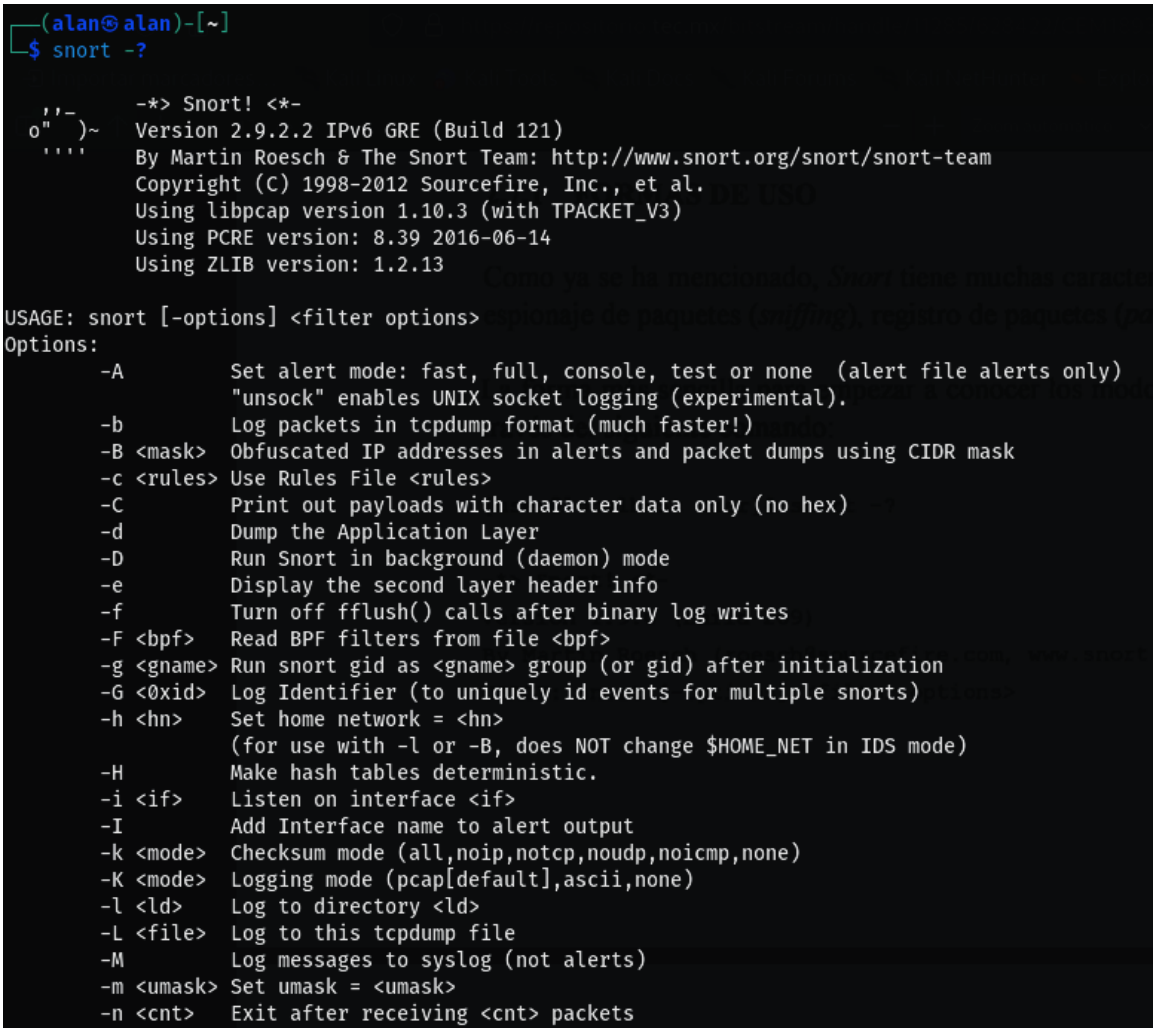
  Las reglas como ya vimos se conforman por un grupo de firmas con las cuales se puede identificar y categorizar los ataques y anomalías. Estas reglas son las que se comparan con cada paquete enviado a través de la red para realizar el tratado correspondiente de los paquetes y alertar al usuario o en dado caso descartar el paquete.
- Sistema de notificaciones

Como ya vimos Snort nos deja guardar la información en archivos o en un gestor de la base de datos para asi poder almacenar las alertas. Hay herramientas hechas por otras empresas que ayudan a interpretar la información otorgada por Snort.

2.3 Formas de Uso

Anteriormente hemos mencionado que Snort posee características sólidas y bastante poderosas como lo son sus funcionalidades como IDS (sistema de detección de intrusos), como Sniffer (análisis de paquetes) como Packet Logger (registro de paquetes) [31] (Sistemas de detección de Intrusos y snort. 2008).

La forma más común y facil de conocer los comandos con lo que trabaja Snort y las opciones de trabajo que ofrece son a través del comando **# snort -?**



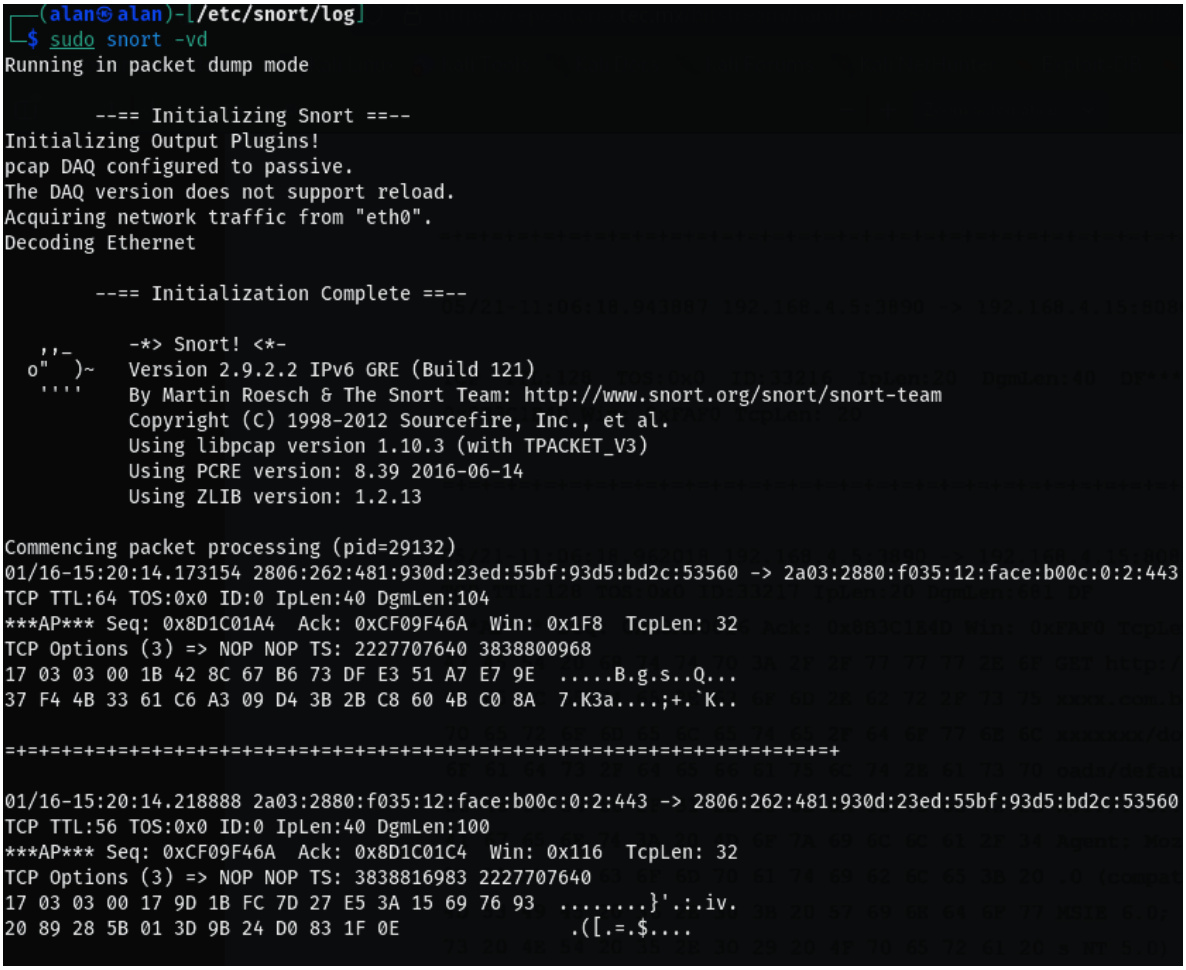
```
-n <cnt>   Exit after receiving <cnt> packets
-N         Turn off logging (alerts still work)
-O         Obfuscate the logged IP addresses
-p         Disable promiscuous mode sniffing
-P <snap>  Set explicit snaplen of packet (default: 1514)
-q         Quiet. Don't show banner and status report
-Q         Enable inline mode operation.
-r <tf>    Read and process tcpdump file <tf>
-R <id>    Include 'id' in snort_intf<id>.pid file name
-s         Log alert messages to syslog
-S <n=v>   Set rules file variable n equal to value v
-t <dir>   Chroots process to <dir> after initialization
-T         Test and report on the current Snort configuration
-u <uname> Run snort uid as <uname> user (or uid) after initialization
-U         Use UTC for timestamps
-v         Be verbose
-V         Show version number
-X         Dump the raw packet data starting at the link layer
-x         Exit if Snort configuration problems occur
-y         Include year in timestamp in the alert and log files
-Z <file>  Set the performonitor preprocessor file path and name
-?         Show this information
```

Este comando muestra las opciones con los cuales podemos trabajar Snort y usarlo en sus diferentes modos, la mayoría de las opciones se usan en conjunto. La captura y registro de paquetes funcionan en conjunto por lo que el formato más común para implementar este modo es con el comando siguiente: **sudo snort -v**.

```
┌──(alan㉿alan)-[/etc/snort/log]
└─$ sudo snort -v
Running in packet dump mode

        --== Initializing Snort ==--
Initializing Output Plugins!
pcap DAQ configured to passive.
The DAQ version does not support reload.
Acquiring network traffic from "eth0".
Decoding Ethernet

        --== Initialization Complete ==--

   ,,_     -*> Snort! <*-
  o"  )~   Version 2.9.2.2 IPv6 GRE (Build 121)
   ''''    By Martin Roesch & The Snort Team: http://www.snort.org/snort/snort-team
           Copyright (C) 1998-2012 Sourcefire, Inc., et al.
           Using libpcap version 1.10.3 (with TPACKET_V3)
           Using PCRE version: 8.39 2016-06-14
           Using ZLIB version: 1.2.13

Commencing packet processing (pid=29028)
01/16-15:12:21.100570 192.168.100.35:55760 -> 31.13.89.15:443
TCP TTL:64 TOS:0x0 ID:22391 IpLen:20 DgmLen:81 DF
***AP*** Seq: 0xEF938AAF  Ack: 0x17119AD5  Win: 0x1F5  TcpLen: 32
TCP Options (3) => NOP NOP TS: 1439248341 3769688921
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

01/16-15:12:21.108818 31.13.89.15:443 -> 192.168.100.35:55760
TCP TTL:85 TOS:0x0 ID:62878 IpLen:20 DgmLen:52 DF
***A**** Seq: 0x17119AD5  Ack: 0xEF938ACC  Win: 0x1BF  TcpLen: 32
TCP Options (3) => NOP NOP TS: 3769699874 1439248341
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

01/16-15:12:21.167915 31.13.89.15:443 -> 192.168.100.35:55760
TCP TTL:85 TOS:0x0 ID:62879 IpLen:20 DgmLen:77 DF
***AP*** Seq: 0x17119AD5  Ack: 0xEF938ACC  Win: 0x1BF  TcpLen: 32
```

2.3.1 Modo Sniffer

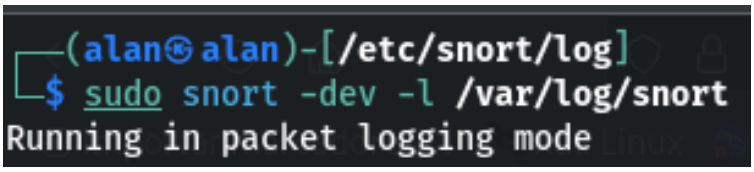
Siempre iniciaremos Snort con nuestro comando anterior **-v** y a continuación se presentarán en pantalla las cabeceras de los paquetes que se vayan obteniendo de la red, en la mayoría de los casos estas cabeceras suelen ser TCP/IP en el modo Sniffer. Esta opción como podemos observar es el modo verbose y en todo caso mostrara las cabeceras de los protocolos más comunes y utilizados para transportar paquetes los cuales son las cabeceras IP, TCP, UDP e ICMP. También si deseamos visualizar la información que pasa por la interfaz de la red añadiremos el comando **-d** a nuestra ejecución.



Si queremos la información más detallada de los paquetes que se están monitoreando añadiremos el comando **-e** a la ejecución y asi podremos observar cómo se despliega la información de la cabecera a nivel de enlace.

2.3.2 Modo Registro de paquetes

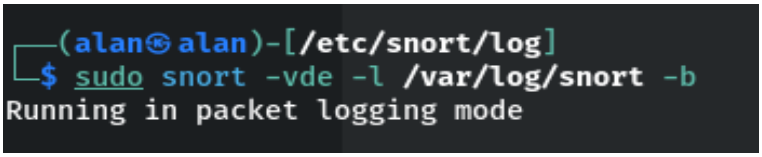
Para el registro de paquetes es necesario considerar el tráfico de la red y el número de dispositivos conectados al dispositivo que este analizando la red ya que pasara una gran cantidad de información que Snort ira filtrando, con esto se guardaran y registraran todos estos datos y paquetes obtenidos para su futuro análisis.



El comando **-l** en Snort se usa para guardar los logs en un directorio determinado, en este caso el directoria asignado es /var/log/snort, en este directorio (carpeta) se creará una estructura de directorios en los cuales se irán archivando los registros, en este modo podemos implementar además otros comandos como:

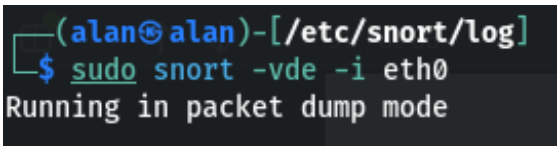
- **-h** Comando para indicar la IP de la red a registrar

- **-b** Comando para hacer que los logs se almacenen en formato binario, con ello TCPDump lo puede analizar para poder implementar los potentes filtros que utiliza Snort, y la salida de los logs ya no será una estructura de directorios sino un solo archivo.



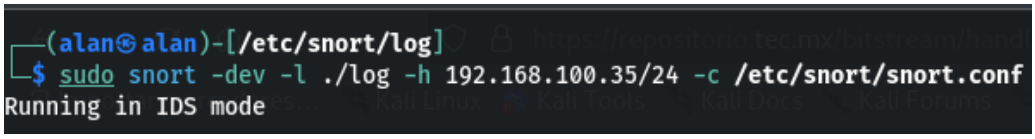
Las ventajas de usar esta opción son que no habrá falta indicar la IP de la red, guardará toda la información en un solo archivo y recogerá los datos de toda la red, además no será necesario usar algún comando para leer el archivo binario, solo es necesario introducir la opción **-r nombrearchivo.log**.

Otra opción es usar el comando **-i** para poder indicarle a Snort la interfaz a utilizar si es que hay más de una.



2.3.3 Modo NIDS

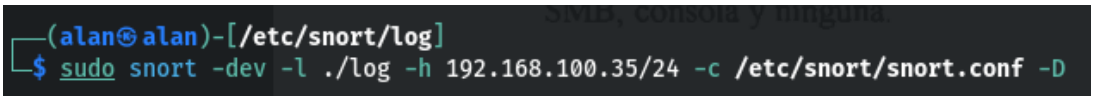
El modo NIDS (Detección de Intrusos) se activará usando la línea de comando **-c snort.conf**.



La configuración completa se guardará en el archivo de snort.conf y lo que guardará será:

- Las reglas
- Preprocesadores
- Configuraciones necesarias para el funcionamiento del modo NIDS

El comando -D hará que snort funciones y se ejecute como servicio.



Alertas generadas

Además de crear la estructura de directorios, Snort creara un archivo con el nombre alert.ids donde se almacenarán las alertas generadas. Para ello Snort cuenta con siete modos de alertas las cuales son : completo, rápido, socket, syslog, SMB, consola y ninguna.

- Fast. En modo de alerta rápida mostrara información acerca del tiempo, mensaje de alerta, clasificaciones, prioridad de la alerta, IP, además de los puertos de origen y destino.

```
09/19-19:06:37.421286 [**] [1:620:2] SCAN Proxy (8080) attempt [**]
[Classification: Attempted Information Leak] [Priority: 2] ...
... {TCP} 192.168.4.3:1382 -> 192.168.4.15:8080
```

- Full. En el modo de alerta completa mostrara el tiempo, mensaje de alerta, clasificación, prioridad de la alerta, IP y puerto de origen/destino, también la información completa de las cabeceras de cada paquete que se ha registrado.
- Socket. Se mandan las alertas a través de un canal para que otra aplicación las escuche. El comando que se utiliza es:

```
┌──(alan㉿alan)-[/etc/snort/log]
└─$ sudo snort -A unsock -c /ect/snort/snort.conf
Running in IDS mode
```

- Console. Se imprimen las alarmas en pantalla.

```
┌──(alan㉿alan)-[/etc/snort/log]
└─$ sudo snort -A console -dev -l /ect/snort/log -h 192.168.100.81/24 -c /etc/snort/snort.conf
Running in IDS mode
```

- None. Desactiva las alarmas.

```
┌──(alan㉿alan)-[/etc/snort/log]
└─$ sudo snort -A none -c /etc/snort/snort.conf
Running in IDS mode
```

- SMB. Hace que snort haga llamadas al cliente de SMB y asi envía mensajes de alerta a hosts de Windows (WinPopUp). Para poder activar este modo se debe compilar Snort con el switch de habilitar alertas SMB (enable -smbalerts).
- Syslog. El programa enviara alarmas al manejador de logs del sistema.
- Eventlog. En este modo se registran las alertas para poder visualizarlas con el visor de sucesos de un sistema Windows. Se activa con -E y solo funciona con win32.

2.4 Estructura de las reglas

Las reglas se pueden catalogar en dos secciones lógicas: la primera la cabecera de la regla y la segunda las opciones.

La cabecera contiene la acción de la regla, los protocolos con los cuales viaja el paquete, las direcciones IP junto con sus máscaras de red, los puertos de origen/destino y el destino del paquete o dirección.

La sección de opciones contiene los mensajes e información para la decisión a tomar por parte de la alerta en forma de opciones [32] (Network Intrusion Detection & Prevention System).

Para poder comprender la forma en que se escriben las reglas en Snort además de su uso veamos el siguiente ejemplo: Es una regla escrita para detectar malware, en específico el Ransomware.

```
alert tcp $EXTERNAL_NET $FILE_DATA_PORTS -> $HOME_NET any
(msg:"MALWARE-OTHER Win.Ransomware.Agent payload download attempt"; flow:to_client,established
file_data; content:"secret_encryption_key"; metadata:service ftp-data, service http;
classtype:trojan-activity; sid:1;)
```

Separando la regla, el contenido de la cabecera se conforma de la siguiente manera:

- Acción de la regla: alert
- Protocolo: tcp (aquí puede varear entre TCP, UDP, IP o ICMP)
- Dirección IP origen: $EXTERNAL_NET (se indica que el análisis es dentro de toda la red)
- Puerto IP origen: $FILE_DATA_PORTS, también puede contener any (indica que puede venir de cualquiera)
- Dirección IP destino: $HOME_NET (indica que será en toda nuestra red)
- Puerto IP destino: any (indica que puede ir a cualquier puerto)
- Dirección de la operación: -> (también puede ser ->, <-, <>)

La información de la sección de opciones es:

Mensaje: msg

Opciones: flow: to client, established file data; content: "secret_encryption_key"; metadata: service ftp-data, service http; classtype: trojan-activity; sid: 1;)

Cada opción tiene un significado y de cada una es la siguiente:

- flow: to client , established file data - Flow revisa las características de sesión de un paquete como dirección cliente/servidor, si forma parte de una conexión TCP, si el paquete esta reensamblado o fragmentado; el argumento to client hace coincidir las respuestas con el servidor y el argumento established que coincida solo con conexiones TCP establecidas.
- content: "secret_encryption_key" – Hace coincidir el contenido de patrones en los paquetes para detectar actividad maliciosa.
- metadata: service ftp-data, service http – Es un complemento de información adicional a la regla para detectar actividades maliciosas que viajan por el protocolo FTP además de detectar en el protocolo HTTP y en ambos analizar actividad maliciosa.
- classtype: trojan-activity – Asigna una clasificación para indicar un ataque asociado a un evento, en este caso a detectar patrones relacionadas con los troyanos.
- sid: 1 – Número de identificación único para esta regla.

Como podemos ver las reglas de Snort no son fáciles de escribir ya que se debe tener buen conocimiento del tipo de amenaza de la cual se quiere proteger y detectar, además de estar actualizados con ellas y las posibles vulnerabilidades.

CAPITULO 3

Capítulo 3 Kali Linux

3.1 ¿Qué es Kali Linux?

Kali Linux es una distribución de Linux Open Source, que posee múltiples herramientas que sirven para realizar pruebas de penetración de los sistemas y redes. Cuenta con múltiples herramientas con las cuales podemos realizar múltiples tareas de prueba, las herramientas las podemos catalogar según sus funciones como:

- Recolección de datos (dominios y direcciones IP)
- Análisis de vulnerabilidades de sitios web
- Ataques a contraseñas
- Análisis de seguridad y vulnerabilidad de redes
- Análisis de tráfico en la red

Además de que Kali Linux tiene un ahorro considerable de recursos al momento de ejecutar algunos ataques de prueba, cualquier persona puede contribuir con la mejora y actualización de la aplicación ya que es de código abierto, también destaca por los pocos requisitos que se solicitan al instalar y sus herramientas cuentan con interfaces graficas que por ello es mucho más rápido el ejecutar algunos ataques a las víctimas. [34] (How to obtain keys in WLAN / WPS networks using Wifislax and Denial of Services with Kali Linux 2019).

3.2 Características de Kali Linux

1. Mas de 300 herramientas de prueba de penetración: Como en cada actualización y despues de la revisión de algunas herramientas, nos podemos dar cuenta de que se han eliminado algunas que no funcionaban correctamente o se han mejorado otras que además de tener una interfaz visual que facilita el trabajo, han sido mejoradas para el facil funcionamiento de estas.
2. GIT- árbol de código abierto: Cuenta con un sistema de versiones distribuido de código abierto, que están disponibles para todos, permite la colaboración de la comunidad publicando los paquetes o repositorios y teniendo las

fuentes disponibles para aquellos que desean modificar o reconstruir paquetes.

3. FHS: Filesystem Hierarchy Standard es el estándar de jerarquía del sistema de archivos, que facilita y especifica la ubicación de directorios y archivos en el sistema de archivos, con esto quiere decir que se pueden ubicar fácilmente archivos binarios, archivos de soporte, bibliotecas, etc.
4. Amplio apoyo orientado a dispositivos inalámbricos: Es una característica de las más destacadas ya que está diseñado para pruebas de penetración y auditoria, lo que incluye pruebas a redes inalámbricas, esto incluye soporte para adaptadores de red, tarjetas inalámbricas, bluetooth, etc.
5. Kernel personalizado con parches de inyección: Como auditores de penetración, el equipo de desarrollo debe realizar evaluaciones inalámbricas para que el kernel tenga los últimos parches de inyección incluidos y realizar de manera exitosa las pruebas de penetración.
6. Entorno de desarrollo seguro: Tiene un entorno configurado y gestionado con la finalidad de minimizar los riegos de seguridad referentes con el software y las pruebas de penetración, con ello haciendo uso múltiples protocolos seguros.
7. Paquetes firmados con PGP y repositorios: Cada paquete está firmado utilizando (Pretty Good Privacy) asi como otros métodos de cifrado esto para verificar la autenticidad y seguridad de cada paquete. Los repositorios de igual manera firman los paquetes para protegerlos de manipulación maliciosa.
8. Multilenguaje: Aunque Kali y sus herramientas están creadas e implementadas con el idioma Inglés, cuenta con un soporte multilenguaje y con esto permite a la mayoría de los usuarios utilizar el sistema junto con sus herramientas en su idioma de preferencia o nativo.
9. Totalmente personalizable: Significa que los usuarios más aventurados y que conocen el ambiente de Linux pueden ajustar y modificar la mayoría de los aspectos para adaptarlos a las necesidades, esto va desde selección de paquetes, configuración del sistema, personalización del entorno grafico

(escritorio, temas, apariencia), desarrollo de scripts o herramientas personalizadas, hasta una imagen completa del sistema totalmente personalizada.

10. Soporte ARMEL y ARMHF: Dado los sistemas de placa-única como Raspberry Pi y Beagle Bone Black, entre otros, se están convirtiendo en sistemas más frecuentes y económicos, se conocía el soporte ARM de Kali Linux debería ser tan robusto como se pudiese gestionar, con instalaciones totalmente funcionales para sistemas ARMEL y ARMHF. Kali Linux está disponible sobre una amplia diversidad de dispositivos ARM, y tiene repositorios ARM integrados con una distribución principal, por lo cual herramientas para ARM son actualizadas en conjunción con el resto de la distribución. [35] (Penetration Testing with Kali Linux, Offensive Security Ltd, 2018)

3.3 Principales herramientas de Kali Linux

NMAP

Nmap (“Network Mapper”) es un software OPEN SOURCE (libre y de código abierto) que se utiliza para el descubrimiento de la red y para realizar auditorías de seguridad.

Generalmente se usa por los administradores de red, mayormente para realizar tareas de inventario de la red, obtener horarios de actualización de servicios , administrar y/o monitorear un host o algún servicio del tiempo de actividad. Utiliza paquetes IP crudos con la finalidad de poder determinar que host están disponibles en la red, los servicios que están ofreciendo (nombre de la aplicación y versión), el sistema operativo que se ejecuta al igual que su versión, que filtro de paquetes está en uso, etc. Está diseñado para el escaneo instantáneo de grandes redes, aunque en función de ejércitos individuales es eficiente. NMAP se ejecuta en múltiples sistemas operativos por lo cual es multiplataforma.

Con la ejecución de NMAP la salida del análisis de la aplicación es un listado con los objetivos analizados aunque dependiendo de las funciones utilizadas puede o

no mostrar información adicional. La información primordial es la tabla de puertos de nuestro interés para monitorear, esta tabla lista el número de puerto junto con el protocolo, el nombre más común de del servicio, y el estado del dispositivo. El estado puede ser de las siguientes maneras:

- Open (abierto). Significa que el puerto se encuentra abierto y esperando conexiones entrantes.
- Closed (cerrado). El puerto está cerrado y no escucha conexiones entrantes, aunque el puerto puede abrirse en cualquier momento.
- Filtered (filtrado). Indica que un firewall, filtro de red o alguna otra medida de seguridad está bloqueando el acceso a ese puerto por lo que NMAP no puede determinar si se encuentra abierto o cerrado.
- Unfiltered (no filtrado). Responden al sondeo de NMAP pero no puede determinar si el puerto está abierto o cerrado

La tabla de puertos puede o no incluir detalles de la versión de las aplicaciones, además NMAP ofrece información de los protocolos IP soportados, en vez de puertos abiertos cuando se solicita un análisis de protocolo IP.

También NMAP puede dar información adicional de los objetivos como el nombre de DNS según la resolución inversa de la IP, un listado de posibles sistemas operativos, el tipo de dispositivo y direcciones MAC. [36] (Nmap network scanning guide - Gordon Lyon, 2013)

WIRESHARK

Es un analizador de protocolos de red que permite ver lo que está sucediendo dentro de ella a un nivel muy detallado de paquetes. Es el estándar más usado en industrias e instituciones educativas. El desarrollo de Wireshark prospera gracias a las aportaciones de expertos de todo el mundo en red. Es la continuación de un proyecto que se inició en 1988. [37] (Combs, About Wireshark)

Con Wireshark podemos analizar los paquetes que entren y salgan de cualquier interfaz de red de nuestro dispositivo (tarjetas/puertos ethernet o Wi-Fi). Toda la información se puede obtener en tiempo real y de igual manera puede ser filtrada

en tiempo real. De igual manera es multiplataforma por lo cual lo hace accesible a múltiples usuarios.

La funcionalidad es muy similar a la de tcpdump con la diferencia de que se incluye una interfaz gráfica, además de muchas funciones de organización y filtrado de información. También permite visualizar todo el tráfico que pasa a través de una red e incluye una versión llamada tshark que está basada en texto. Posee un decodificador de protocolos que incluyen desde los más comunes hasta los menos conocidos.

Permite examinar los datos de una red activa o de los de un archivo de captura almacenado en algún dispositivo. Se puede analizar la información capturada, a través de los detalles y resúmenes de cada paquete. Incluye un amplio lenguaje en cuanto el filtrado de paquetes que queremos ver y posee una gran habilidad al momento de mostrar el flujo reconstruido de una sesión TCP.

BURP SUITE

Es una plataforma para la realización de las pruebas de seguridad de aplicaciones web. Sus herramientas funcionan perfectamente en conjunto para el proceso de prueba, cartografía y análisis de la superficie de ataque de una aplicación, todo ello a través de la búsqueda y explotación de vulnerabilidades de seguridad. Dentro de la aplicación usa un proxy el cual intercepta las solicitudes de manera que inspecciona el tráfico en búsqueda de amenazas y protege a los usuarios de datos maliciosos.

Esta gran herramienta cuenta con los siguientes componentes clave [38] (Seguridad Informática, 2012):

- Un Proxy que se encarga de interceptar los paquetes, lo que le permite inspeccionar y modificar el tráfico entre el navegador y la aplicación de destino.
- Un Spider Web con reconocimiento de aplicaciones, para el rastreo de contenido y funcionalidad de un sitio web.

- Un escáner de aplicaciones web avanzadas para automatizar la detección de numerosos tipos de vulnerabilidades.
- Una herramienta de intrusión, para realizar poderosos ataques personalizados de encontrar y explotar vulnerabilidades inusuales.
- Una herramienta de repetición de solicitudes, para manipular y volver a enviar peticiones individuales y verificar la consistencia de los controles de seguridad.
- Una herramienta secuenciador, para probar la aleatoriedad de las credenciales de sesión.
- La capacidad de guardar su trabajo y reanudar el trabajo más tarde.
- Extensibilidad, lo que le permite escribir fácilmente tus propios plugins, para realizar tareas complejas y personalizadas muy dentro de Burp.

AIRCRACK-NG

Aircrack-ng es un programa que puede recuperar claves a través de la captura de paquetes de datos. Consiste en un analizador de paquetes de redes, el cual recupera contraseñas WEP y WPA/WPA2-PSK y otro conjunto de herramientas de auditoría inalámbrica. WPA2 (Wi-Fi Protected Access 2), es un sistema para proteger las redes inalámbricas (Wi-Fi); creado para corregir las deficiencias del sistema del estándar 802.11. De hecho es un conjunto de herramienta dedicada a la auditoria de redes inalámbricas. [39] (Paspuel,2018, Hack de Redes Wireless con Aircrack-ng)

Las herramientas que más se usan en cuanto a auditoria inalámbrica son [40] (Aircrack-ng, 2007):

- Aircrack-ng. Descifra las claves de redes Wi-Fi usando ataques de fuerza bruta o diccionario a claves WEP y WPA/WPA2
- Airodump-ng. Escanea las redes para capturar paquetes de datos y realizar un escaneo pasivo de redes cercanas, además de mostrar información detallada de las redes (nombre de la red, dirección MAC, canal de transmisión, potencia de la señal y dispositivos conectados).

- Aireplay-ng. Inyecta tráfico en la red para recuperar claves WEP o probar la seguridad de la red con ataques de autenticación.
- Airmon-ng. Permite crear interfaces virtuales de redes inalámbricas en modo punto a punto, punto a multipunto y esto sirve para hacer pruebas de seguridad en entornos controlados.
- Airbase-ng: Puede usarse para crear un punto de acceso falso (fake AP) y puede ser funcional para realizar ataques de intercepción de paquetes (man in the middle).

Hay diferencia entre crackear WPA/WPA2 y WEP. Ya que en las WEP se usan métodos estáticos de inyección y para WPA/WPA2 solo podemos implementar ataques de fuerza bruta debido a que la clave no es estática y no conseguiremos de manera facil la clave.

METASPLOIT FRAMEWORK

Metasploit comenzó como un juego de red , pero su potencial fue descubierto cuando se convirtió totalmente en una herramienta de exploración. Contiene un conjunto de herramientas que contienen múltiples funciones para diferentes propósitos pero es probablemente más conocido por su marco de exploración.

Podemos definirlo como una herramienta GNU desarrollada en Perl y escrita con múltiples lenguajes como C, Python, ASM, etc, para el desarrollo, prueba, mejora y penetración de diversos sistemas.

Trabaja con una base de datos en la cual se encuentra una gran lista de exploits y vulnerabilidades, lo único que hay que indicarle es la vulnerabilidad que se va a utilizar, el sistema víctima, el tipo de ataque y datos diversos que se utilizaran para atacar al host.

Su nombre se debe a que en realidad es todo un entorno de pruebas para diversas plataformas, la cual trabaja a base de bibliotecas, bases de datos, diversidad de programas, Shell codes, etc, además de que puede aprovechar múltiples cargas útiles como la creación de nuevos usuarios, abrir puertas traseras, hasta instalar

software dentro de un host. Por tal motivo deja de ser un software y se considera un framework. [41] (Pritchett & De Smet, 2013)

JHON THE RIPPER

Es un programa que permite la recuperación de contraseñas a partir de datos que existen en nuestro sistema. El propósito principal es la detección de contraseñas que son débiles por parte del administrador del sistema.

Este sistema no solamente funciona por fuerza bruta, dispone de varios modos de funcionamiento que permite la búsqueda inteligente de las contraseñas más inseguras. El sistema usa ataques por diccionario, es decir tiene un archivo de texto el cual puede incluir contraseñas típicas, fáciles (password,12345,###$$$) y las va probando una a una. El sistema prueba con variaciones de estas palabras de modo que les añade números, signos, mayúsculas, minúsculas, cambia letras, combina palabras, etc.

Además ofrece el típico sistema de fuerza bruta, donde se prueban las combinaciones posibles sin importar si son palabras o no. Este es el sistema más lento y solo usado en casos específicos ya que hay ataques con diccionario que permiten descubrir de manera más rápida los ataques de diccionario.

Tiene la capacidad de detectar el tipo de cifrado de entre los más comunes como lo son: DES, MD5, Blowfish, Kerberos, AFS, LM, etc. Además de que puede ser personalizable y configurado de varias maneras para agilizar la velocidad de respuesta, con ello nos referimos a que podemos crear e implementar nuestro propio diccionario para ataques personalizados. [36] (Benito, Laboratorio de Seguridad Informática con Kali Linux 2014)

HYDRA

Es una herramienta que usa el método de ataque por fuerza bruta, es decir, realiza una gran cantidad de intentos de registros en contra de diferentes protocolos hasta lograr el acceso, esta herramienta es ideal para ataques de sistema de correo electrónico, puede apuntar a atacar a una dirección IP en específico y/o protocolo,

hydra soporta los siguientes protocolos [42] (Benito, Laboratorio de Seguridad Informática con Kali Linux 2014):

- Asterisk: Protocolo de comunicaciones de voz sobre IP (telefonía)
- AFP: Protocolo para compartir archivos e impresoras en redes Macintosh
- Cisco AAA: Protocolo de autenticación, autorización y contabilidad
- Cisco auth: Protocolo de autenticación
- Cisco enable: Protocolo de habilitación
- CVS: Protocolo de Control de Versiones
- Firebird: Protocolo de base de datos relacional
- FTP: Protocolo de transferencia de archivos
- HTTP-FORM-GET /HTTP-FORM-POST /HTTP-GET /HTTP-HEAD /HTTP-PROXY /HTTPS-FORM-GET /HTTPS-FORM-POST /HTTPS-GET /HTTPS-HEAD /HTTP-Proxy: Protocolos de transferencia de hipertexto (navegación web y transferencia de datos)
- ICQ: Protocolo de mensajería instantánea
- IMAP: Protocolo de Acceso a Mensaje de Internet (gestión de correos electrónicos)
- IRC: Protocolo de Comunicación de Internet Relay Chat (comunicación en tiempo real)
- LDAP: Protocolo de Acceso Ligero a Directorios (acceder/mantener la información de un directorio)
- MS-SQL /MYSQL /Oracle Listener /Oracle SID /Oracle / POSTGRES: Protocolos de bases de datos relacionales
- NCP: Protocolo de Conexión NetWare
- NNTP: Protocolo de Transferencia de Noticias de Red (lectura y publicación de mensajes de noticias)
- PC-Anywhere: Protocolo de acceso remoto
- PCNFS: Protocolo de Sistema de Archivos de Red de Computadora
- POP3: Protocolo de Oficina de Correo (recuperación de correo electrónico)
- RDP: Protocolo de Escritorio Remoto (acceso remoto sistemas windows)

- Rexec /Rlogin /Rsh: Protocolos de Acceso Remoto (sistemas unix)
- S7-300: Protocolo de programación para PLC
- SAP/R3: Comunicación con sistemas SAP
- SIP: Protocolo de Inicio de Sesión (señalización y control de sistemas multimedia)
- SMB: Protocolo de Mensajería de Servidor (compartir archivos e impresoras en redes de Microsoft)
- SMTP: Protocolo de Transferencia de Correo Simple (envió de correo electrónico)
- SMTP Enum: Protocolo para enumerar direcciones de correo electrónico validas en un dominio
- SNMP: Protocolo de Administración de Red Simple (supervisar y gestionar dispositivos de red)
- SOCKS5: Protocolo de retransmisión de sockets (enrutar conexiones de res)
- SSH/SSH2: Protocolo de Shell Seguro (acceso remote y transferencia de daros segura)
- Subversion: Protocolo de control de versiones
- Teamspeak (TS2): Protocolo utilizado en la comunicación por voz en tiempo real (juegos en línea y aplicaciones colaborativas)
- Telnet: Protocolo de Emulación de Terminal (acceso remoto a sistemas)
- Vmware-Auth: Autenticación en sistemas Vmware
- VNC: Protocolo de Computación Virtual de Red (acceso remoto y control de ordenadores)
- XMPP: Protocolo Extensible de Mensajería y Presencia (mensajería instantánea y colaboración en tiempo real)

Para llevar a cabo sus ataques hydra utiliza los datos como nombre de usuario que sean validos que se encuentran mediante un reconocimiento previo de la red y aplica los diccionarios de posibles contraseñas validas hasta encontrar la correcta.

CAPITULO 4

Capítulo 4 Desarrollo del proyecto

4.1 Planteamiento y desarrollo de la topología experimental

El entorno de seguridad de la universidad se encuentra controlado, con esto se quiere decir que los paquetes que viajan por la red ya se encuentran filtrados y son seguros, por lo cual una prueba dentro del entorno de la red de la universidad o la facultad no sería optimo. Sin embargo hay que tener conciencia de que ninguna red es totalmente inmune a las amenazas de red. Una vez mencionado esto la propuesta realizada es sobre la implementación de un Sistema de Detección de Intrusos (IDS).

Para esto se utilizaron dos computadoras y un router para crear un entorno de red controlado y asi no afectar ningún entorno perteneciente a la institución.

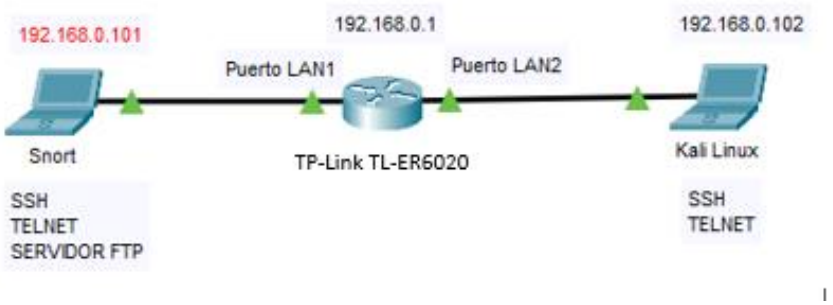


Figura 8 Prototipo de red experimental

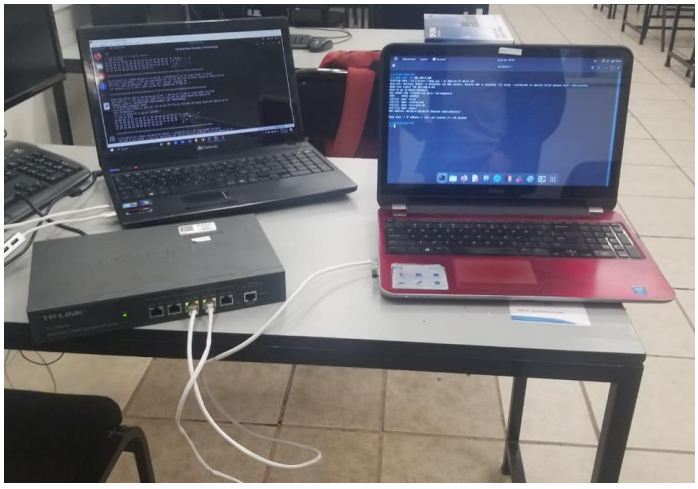
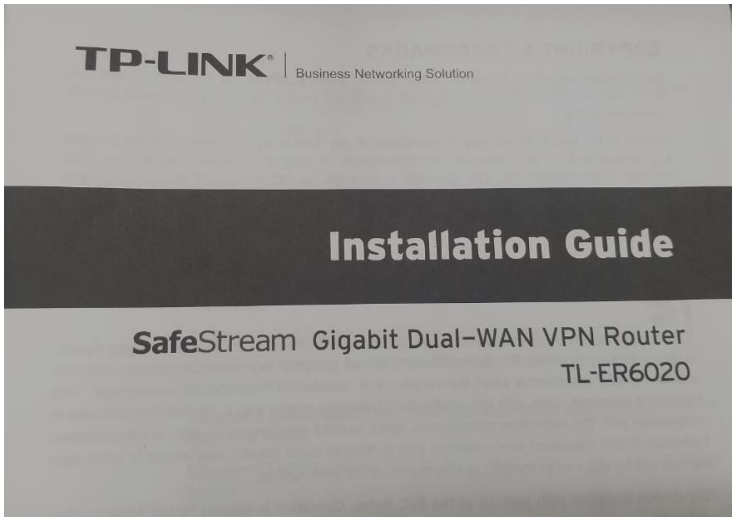
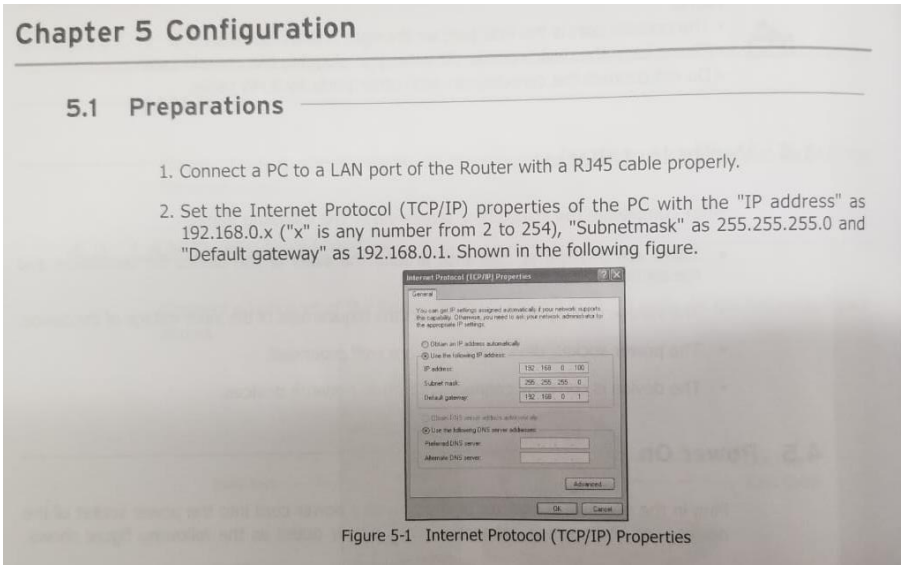


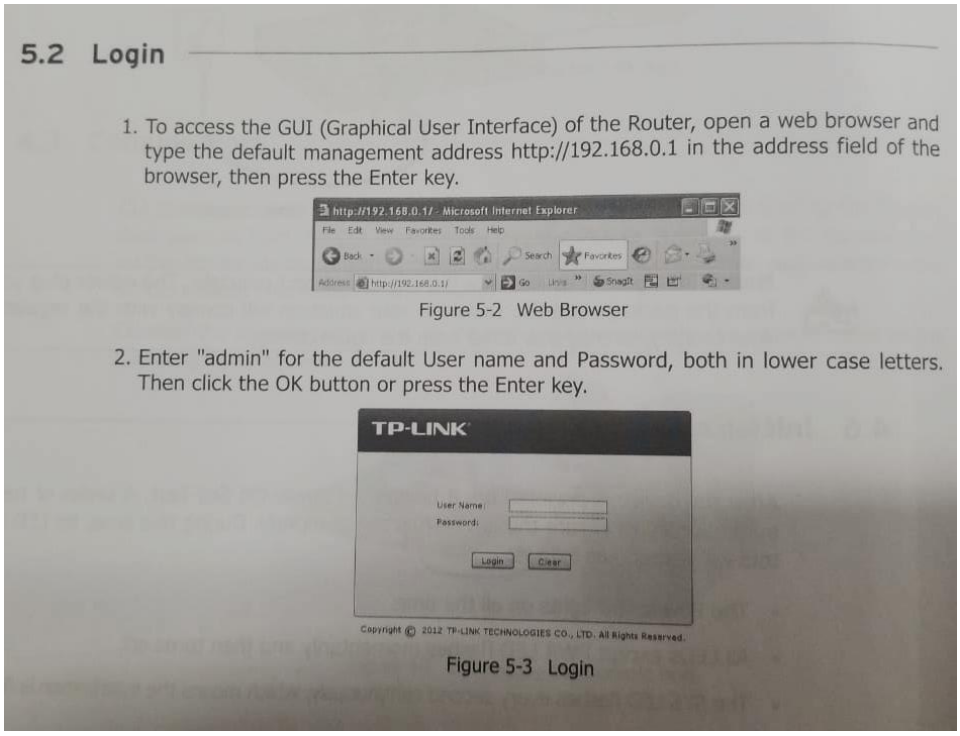
Figura 9 Topología física

Descripción de los componentes de la topología

- **Router TP-Link TL-ER6020**: Este router se utiliza para dirigir el tráfico entre las dos computadoras y simular el tráfico de red en un entorno controlado. El router también puede configurarse para redirigir ciertos tipos de tráfico hacia el host del IDS para su análisis, asi como dar direcciones por medio de DHCP, una VPN, una red con DMZ, quitar o agregar protocolos de red, etc. Todo lo anterior puede ser administrado desde el navegador por medio de la dirección de red predeterminada 192.168.0.1 del router según el manual de usuario, además de que hay que asignar la máscara de red 255.255.255.0.







5.2 Login

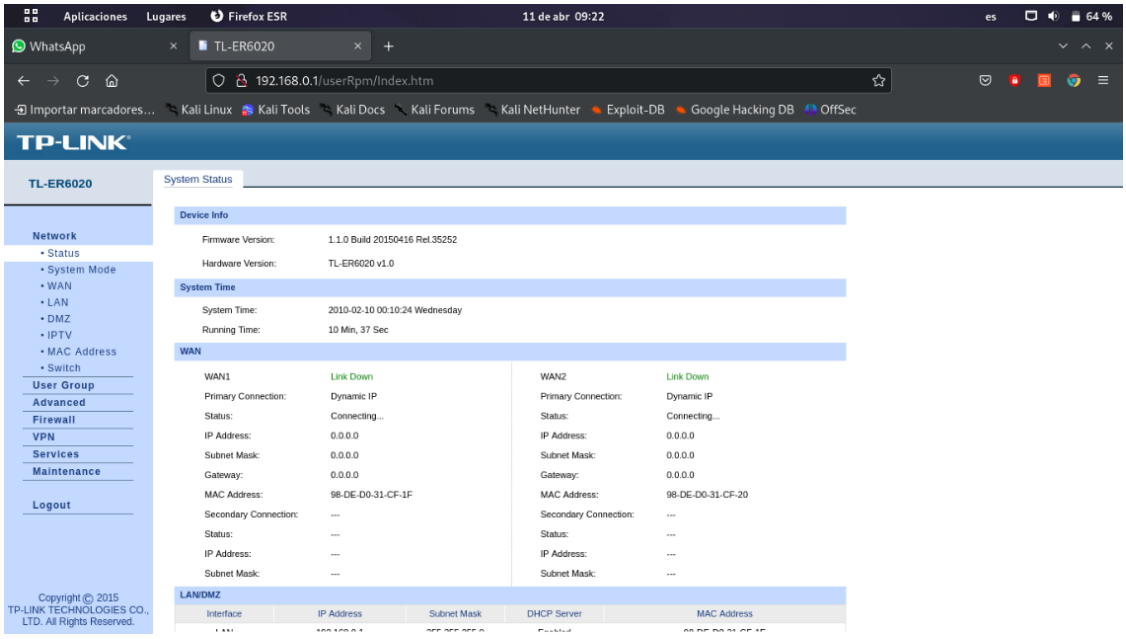
1. To access the GUI (Graphical User Interface) of the Router, open a web browser and type the default management address http://192.168.0.1 in the address field of the browser, then press the Enter key.

Figure 5-2 Web Browser

2. Enter "admin" for the default User name and Password, both in lower case letters. Then click the OK button or press the Enter key.

Figure 5-3 Login

Figura 10 Manual de Usuario TP-Link



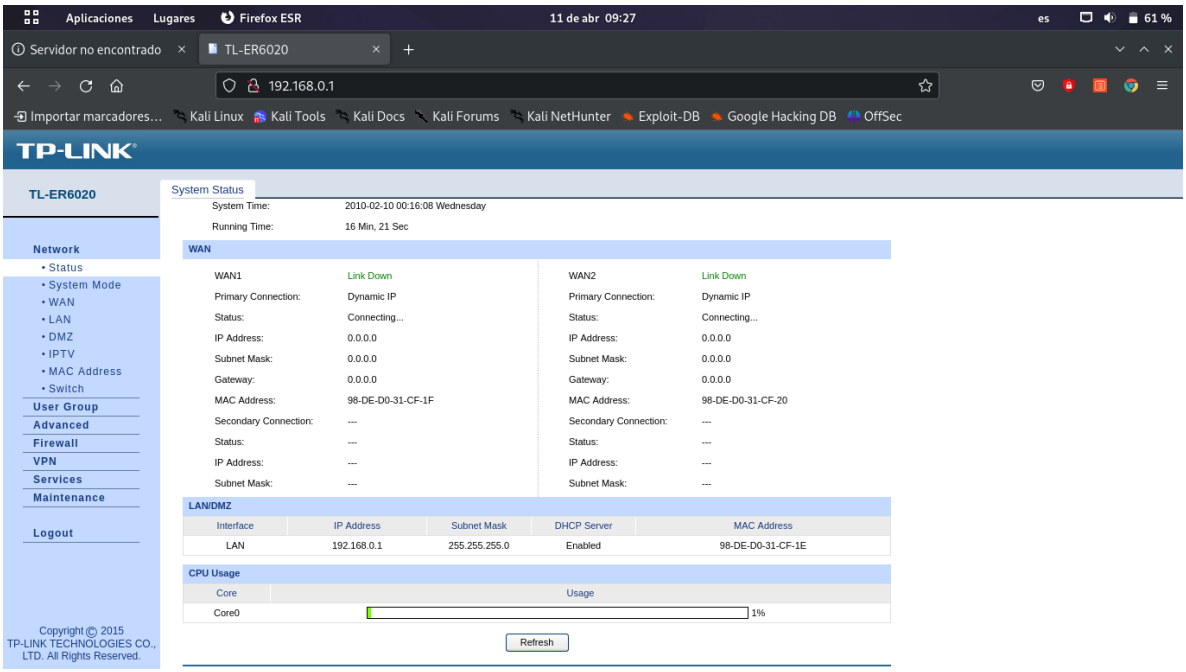


Figura 11 Status del Router

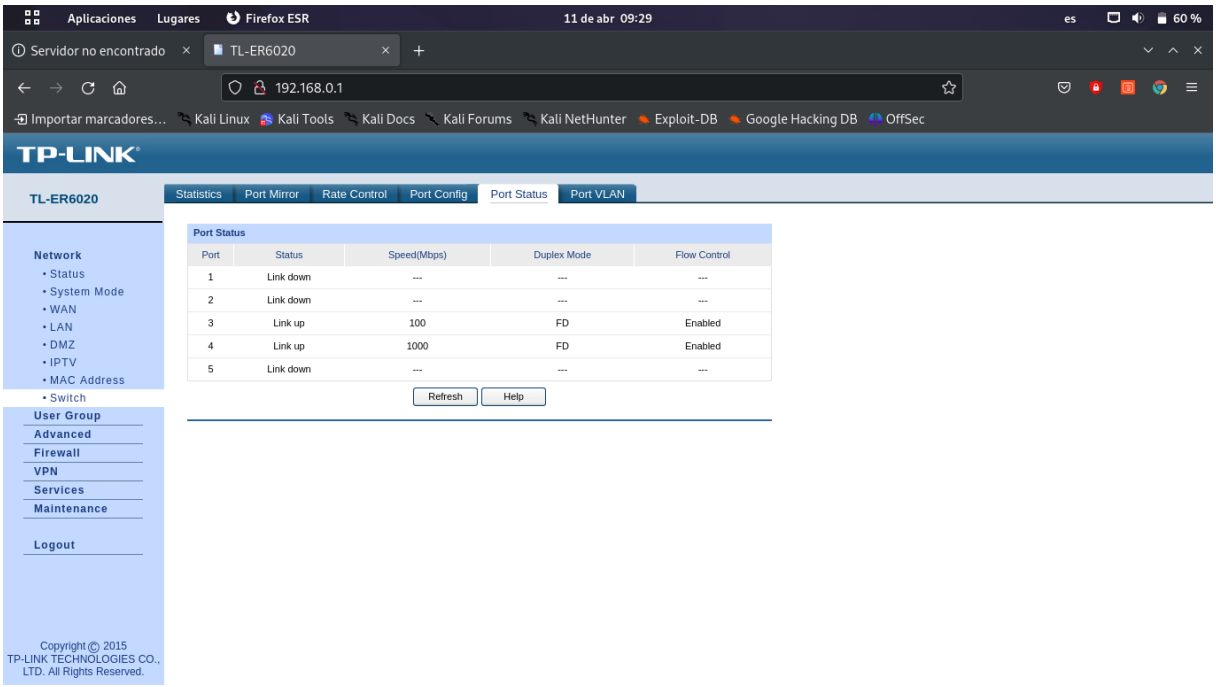


Figura 12 Status de los puertos del router

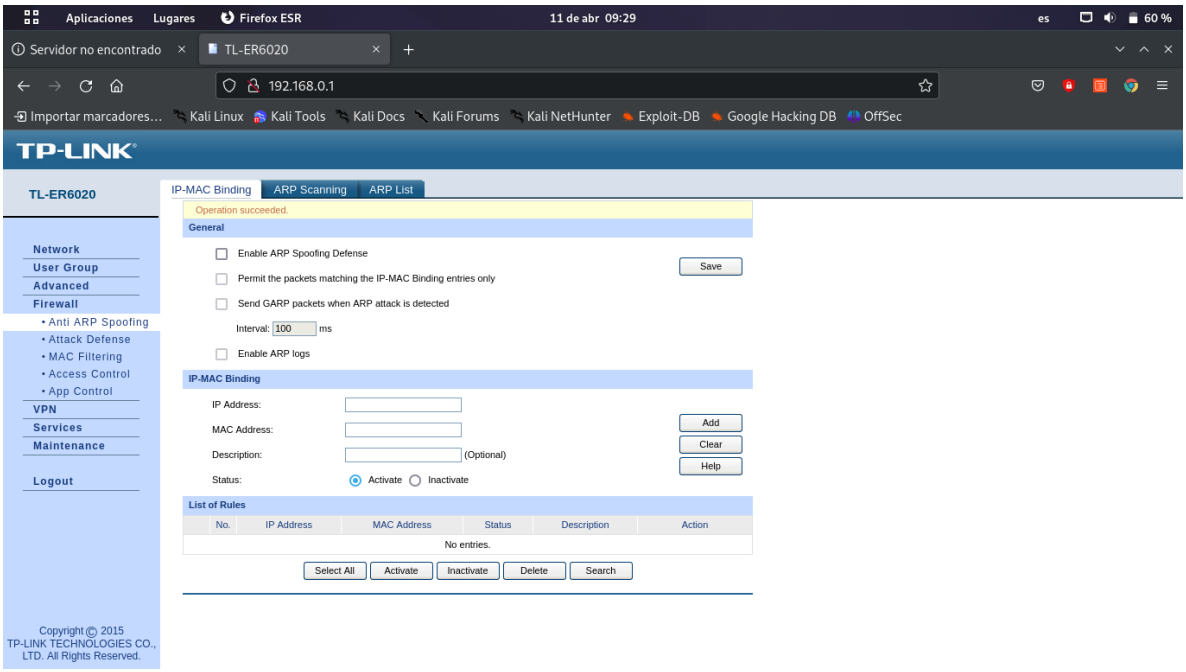
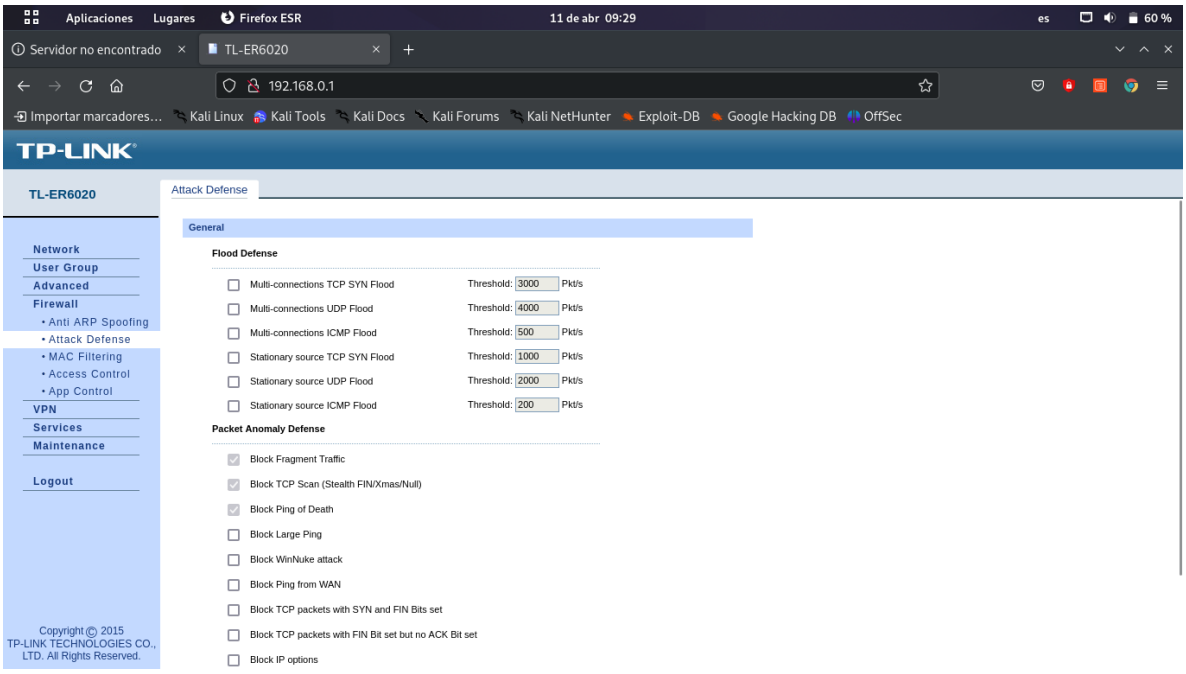


Figura 13 Defensa de Spoofing del router



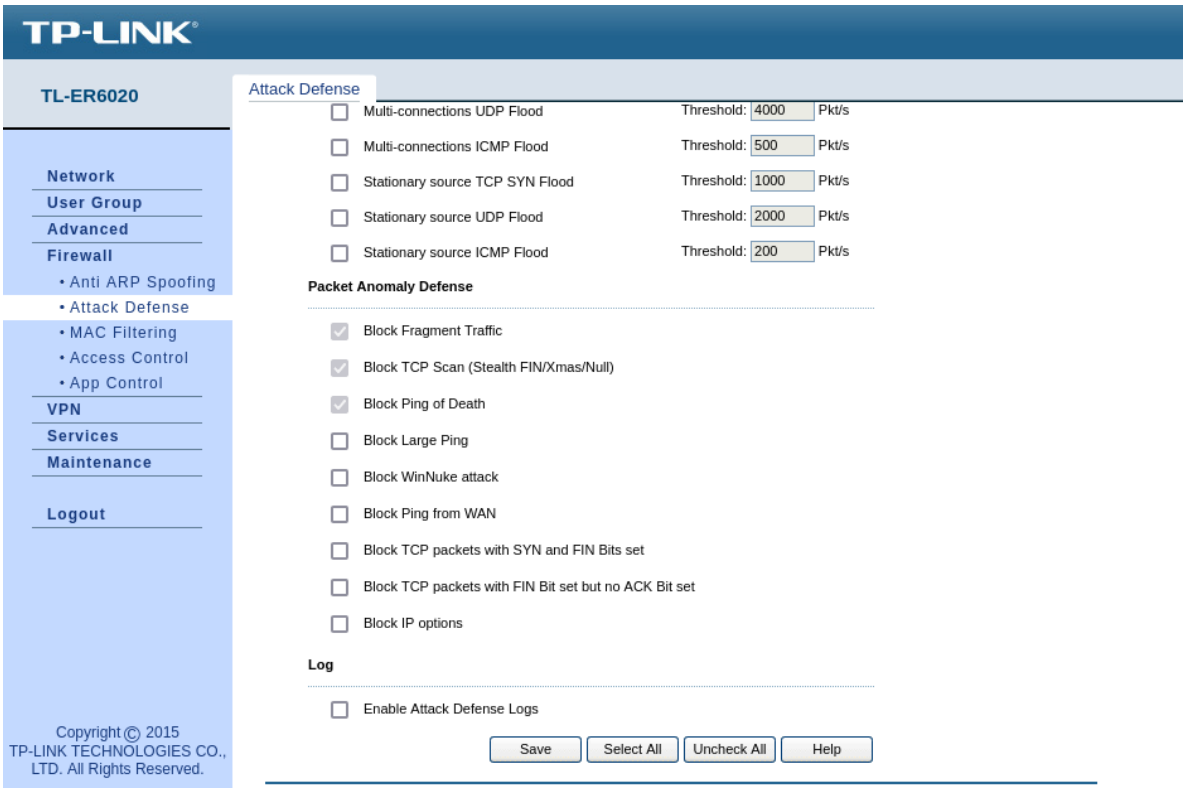


Figura 14 Defensas generales de ataques, puertos, paquetes y protocolos del router

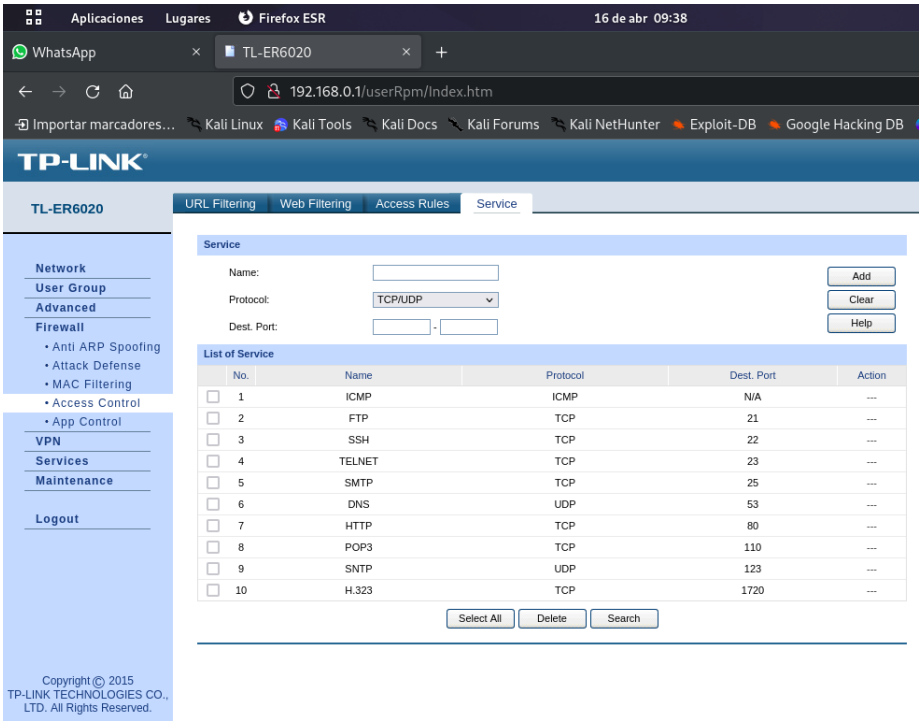


Figura 15 Lista de puertos, protocolos y puertos

Este router contiene cierto nivel de seguridad ya que viene configurado de fábrica de manera que bloquee el tráfico fragmentado, el escaneo de paquetes TCP y el ping de la muerte, ya que estos son elementos altamente usados por las amenazas y los atacantes, para hacer funcionar las reglas se desactivaron todas las defensas posibles del router. Por lo cual algunas pruebas se vieron afectadas y solo para comprobar algunas reglas se hicieron pruebas directas entre las computadoras.

- **Computadora con Kali Linux:** Esta computadora actúa como la fuente de tráfico malicioso o sospechoso. Se seleccionó Kali Linux debido a su amplia gama de herramientas de seguridad y su capacidad para generar tráfico de red, además de múltiples ataques a diferentes puertos usando diversos protocolos y debido a su facilidad de uso para ello. Esta computadora se configuro con la IP 192.168.0.102/24 para pertenecer a la red.

Figura 16 Configuración IP Kali

- **Computadora con Snort:** Esta computadora actúa como el host del IDS, donde se implementa y configura Snort para detectar posibles intrusiones o amenazas en el tráfico de red. Además de la configuración estándar de Snort, se han instalado y configurado servicios adicionales, como SSH, Telnet y un servidor FTP, para ampliar la funcionalidad del entorno experimental y poder observar diferentes tipos de ataques simulados. Esta computadora se configuro con la IP 192.168.0.101/24 para pertenecer a la red.

Figura 17 Configuración IP Snort

Una vez realizada la conexión y configuración de los dispositivos se procedió a realizar varios pings en donde podemos confirmar la conexión entre ellos.

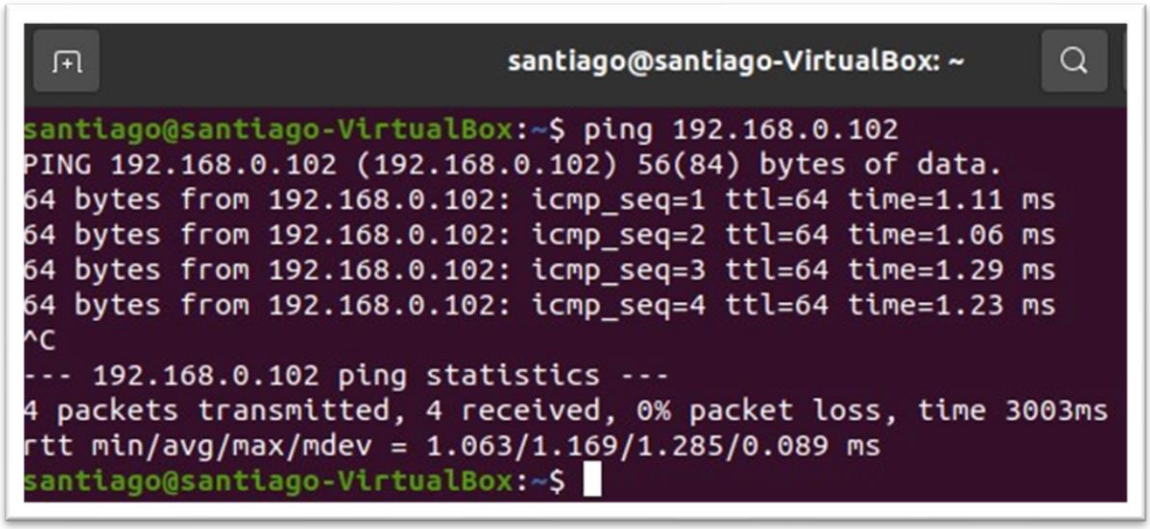


Figura 18 Ping de configuración hacia pc Kali

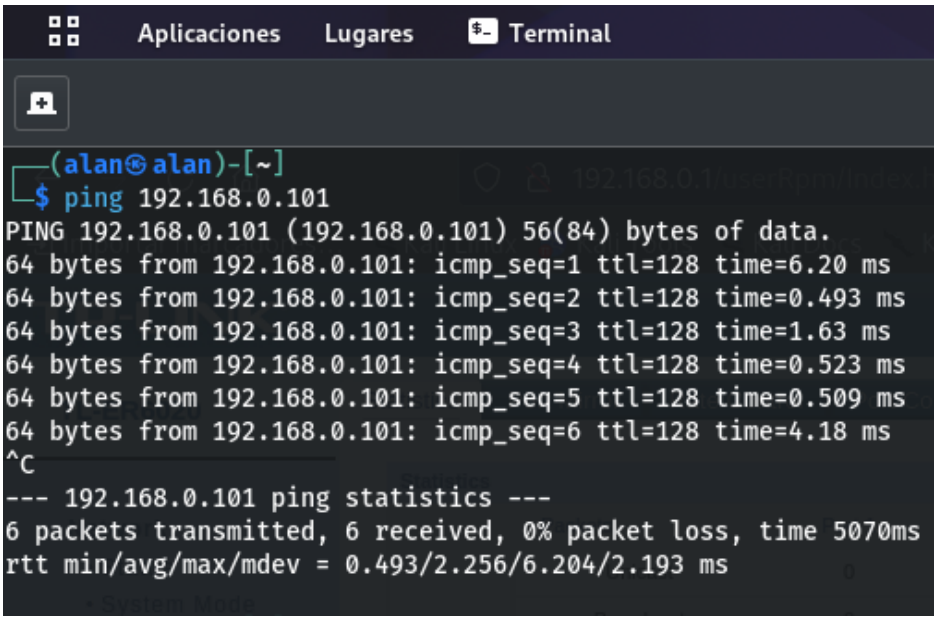


Figura 19 Ping de configuración hacia Snort

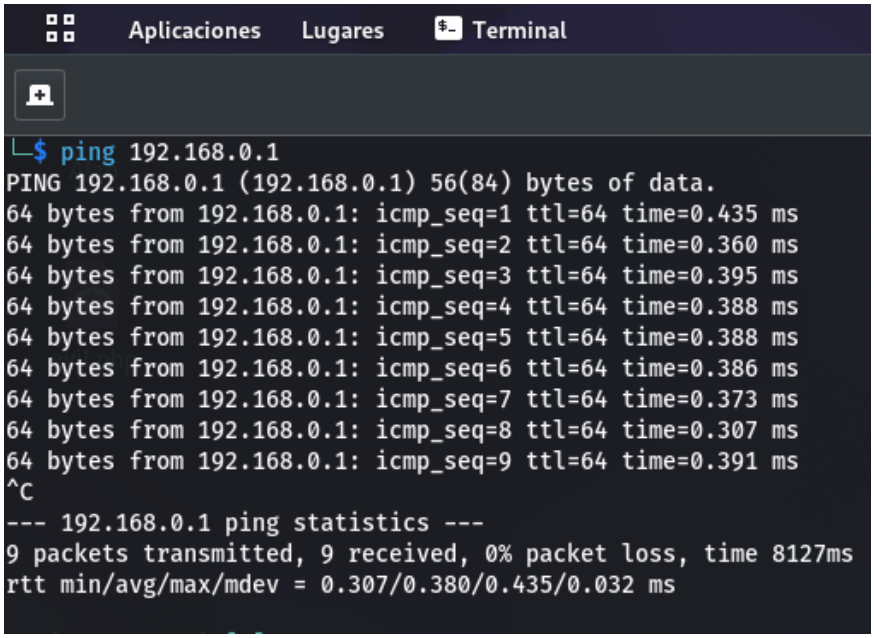


Figura 20 Ping de configuración hacia Router

Descripción general del experimento:

- **Configuración del entorno de red**: Como punto crucial se estableció una conexión física entre las dos computadoras y el router por los puertos de ethernet de cada dispositivo correspondiente. Se configuraron dichas interfaces de cada dispositivo y se aseguró la conectividad entre ellos.

- **Instalación y configuración de Snort:** En la computadora designada como host del IDS, se instaló y configuró Snort según las especificaciones del fabricante. Además de la configuración estándar de Snort, se implementaron herramientas adicionales y se configuraron servicios como SSH, Telnet y un servidor FTP para ampliar la funcionalidad del entorno experimental y el uso del puerto 21 y 80.
- **Generación de tráfico de red:** Utilizando herramientas disponibles en Kali Linux como Nmap, Hydra, hping 3, Metasploit se generó tráfico de red simulado con el fin de emular actividades maliciosas o sospechosas. Este tráfico se dirigió hacia el router y luego hacia el host del IDS para su análisis.
- **Monitoreo y análisis de alertas:** Se monitorearon las alertas generadas por Snort en respuesta al tráfico de red analizado. Se evaluaron estas alertas para determinar su validez y gravedad, así como para identificar posibles mejoras en la configuración del IDS y el conocimiento de los ataques que se realizaron.

4.2 Justificación de la topología

La configuración de la topología experimental surge como respuesta a las limitaciones específicas del entorno de red en nuestra universidad, donde los paquetes de red y servicios ya están filtrados. Esta justificación se fundamenta en los siguientes aspectos:

- **Superación de Limitaciones Actuales:** En nuestra universidad el recibir paquetes filtrados genera una falsa sensación de seguridad, ya que se asume que la red está protegida contra amenazas. Sin embargo ninguna red está completamente inmune. Por lo tanto, la topología experimental busca simular un entorno donde estas restricciones no estén presentes, permitiendo así una evaluación más precisa de la efectividad de un IDS en condiciones realistas.
- **Enfoque en la Evaluación de un IDS:** La implementación de un Sistema de Detección de Intrusos (IDS) como Snort en la topología experimental nos brinda la oportunidad de evaluar su capacidad para detectar y responder a

posibles amenazas en un entorno controlado, además de saber a qué amenazas se está enfrentando y la posible fuente del ataque, determinando si es una amenaza externa o interna. Al simular un escenario donde los filtros de red no están presentes y los paquetes viajan sin ser previamente analizados, podemos examinar cómo el IDS responde a diferentes tipos de tráfico y actividades potencialmente maliciosas dentro de la red.

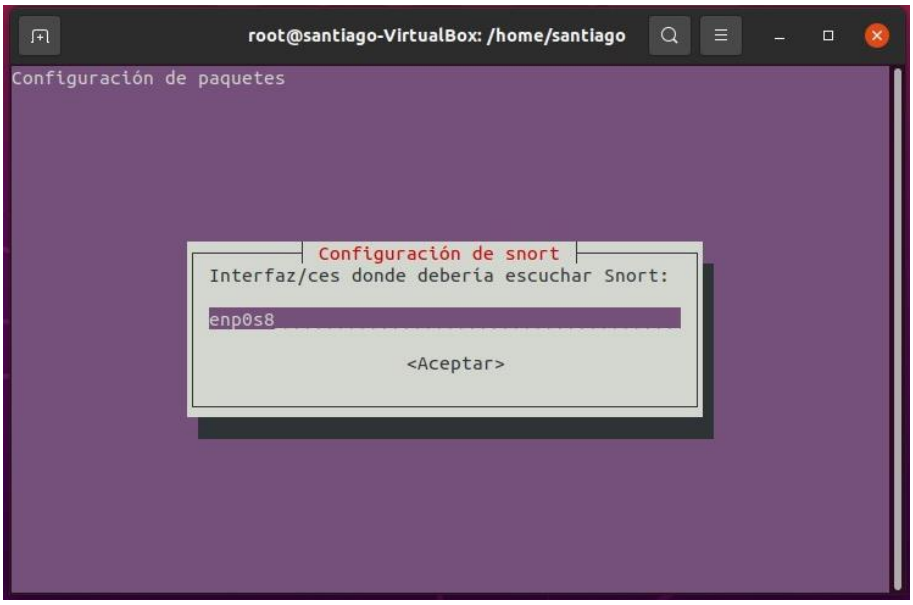
- **Identificación de Amenazas**: A pesar de las medidas de seguridad existentes en la red de la universidad, es importante reconocer que siempre existen posibles vulnerabilidades y amenazas que podrían pasar desapercibidas. La topología experimental nos permite explorar estas posibles brechas de seguridad y evaluar cómo un IDS como Snort puede contribuir a una detección más temprana y efectiva de estas amenazas, con esto tener una garantía de seguridad en una red y una administración optima de la misma.
- **Relevancia para la Seguridad**: Esta investigación no solo tiene implicaciones para la seguridad de la red en el contexto de una simulación, sino que también puede proporcionar información valiosa sobre las estrategias de seguridad en entornos reales. Los hallazgos obtenidos pueden ser aplicables a instituciones educativas, entornos empresariales o pequeñas redes que no requieren una administración continua, ya que enfrentan desafíos similares en términos de filtrar los paquetes de red y contener las posibles amenazas.

4.3 Configuración de Snort

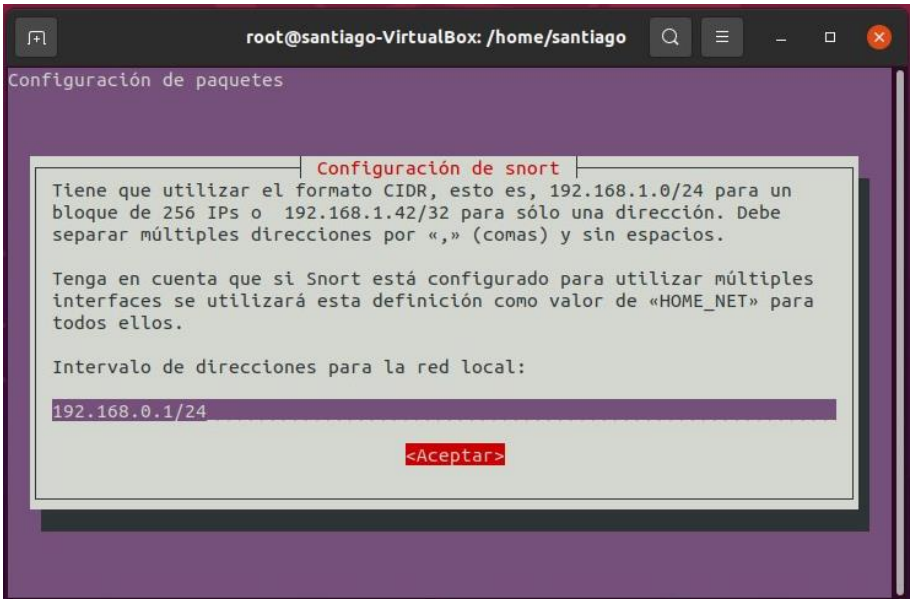
La instalación de Snort es muy sencilla y se puede realizar desde la terminal de Linux con el comando:

- $ apt-get install snort

Una vez instalado Snort nos abrirá una ventana donde nos indicará que debemos ingresar la interfaz de red que usaremos para analizar el tráfico.



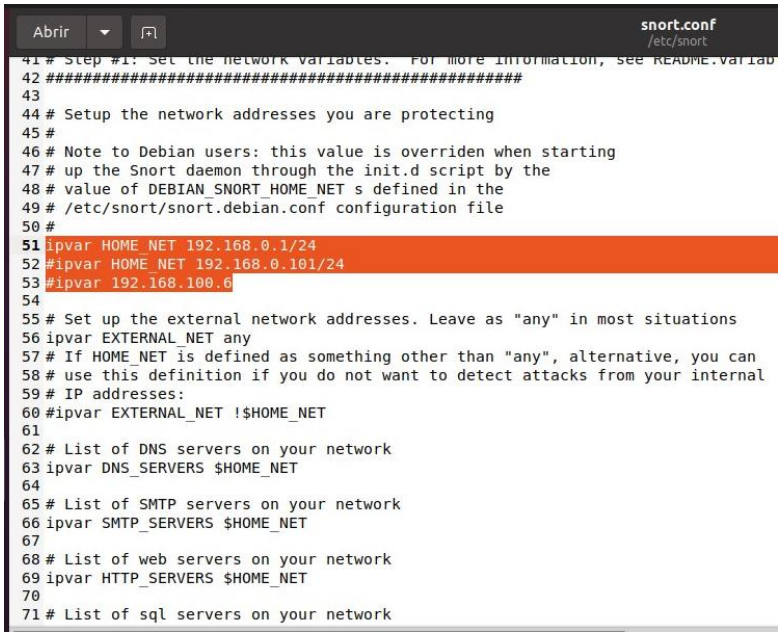
Y despues nos pedirá que ingresemos la dirección IP del host o dispositivos del que analizara la red.



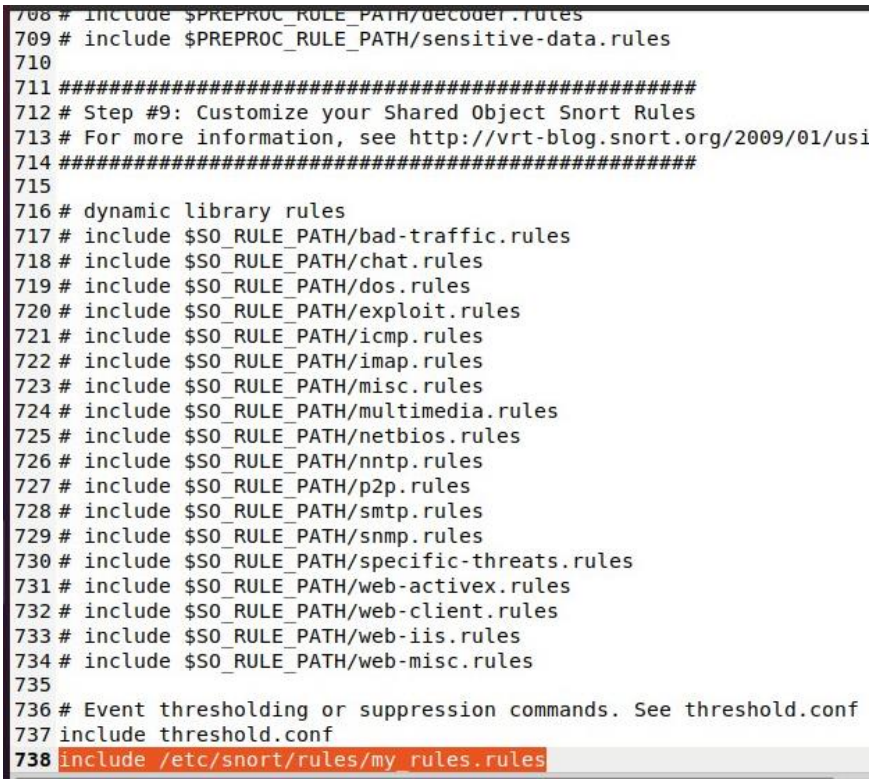
También nos da la opción de que nos envíe de manera periódica un resumen diario de los registros de snort a una dirección de correo electrónico.

Despues de que la instalación de Snort haya sido realizada podremos cambiar los parámetros necesarios como la interfaz de red si es que hay más de una o se debe cambiar, definir la IP del host o dispositivo que se vaya a analizar. En este caso el archivo de configuración de snort al cual accederemos con el siguiente comando.

- $ gedit /etc/snort/snort.conf



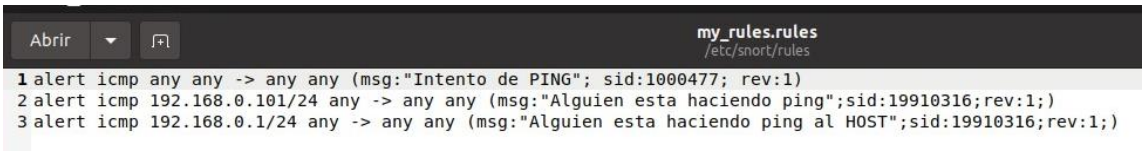
En este archivo se editará la ruta de acceso para que ejecute las reglas que creamos y pueda generar las alertas de aviso de nuestro laboratorio experimental. En este caso la ruta que se agrego fue /my_rules.rules y la dirección IP del host.



Una vez configurada la IP de análisis de red y la ruta de las reglas que agregaremos a Snort creamos la carpeta de reglas con la ruta /etc/snort/rules/my_rules.rules y el comando:

- gedit /etc/snort/rules/my_rules.rules&

En donde se agregaron las primeras reglas de Ping que se hicieron para comprobar que había conexión entre ambas computadoras y el Router.



Las reglas que se ocuparon para comprobar las conexiones fueron:

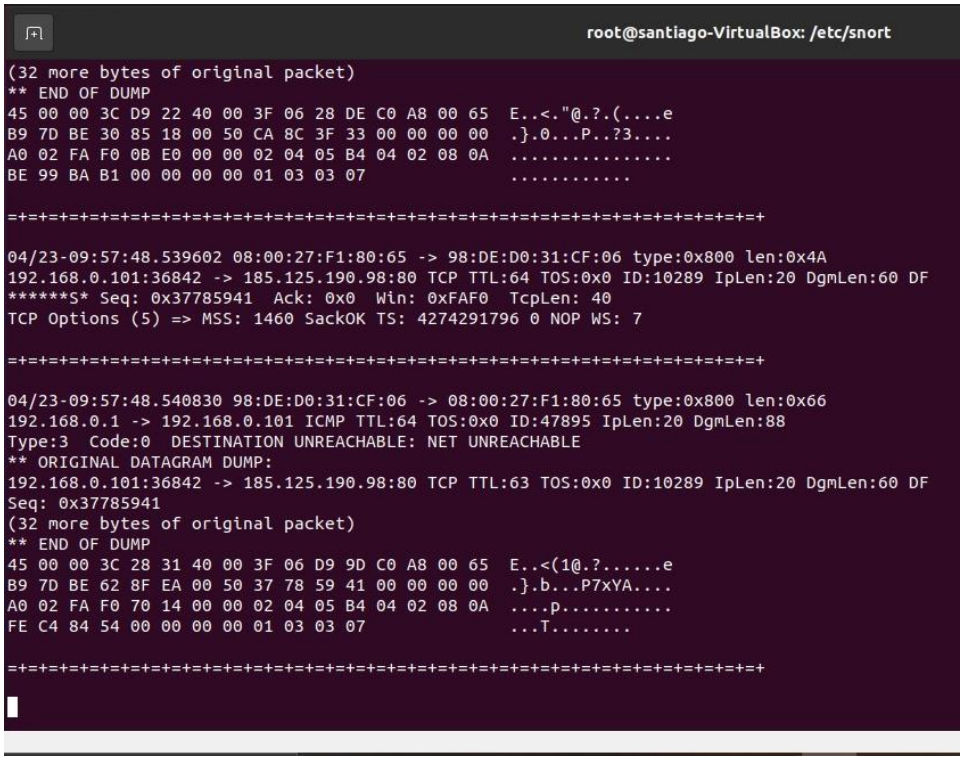
- alert icmp any any -> any any (msg: ”Intento de ping”; sid: 1000477; rev: 1)

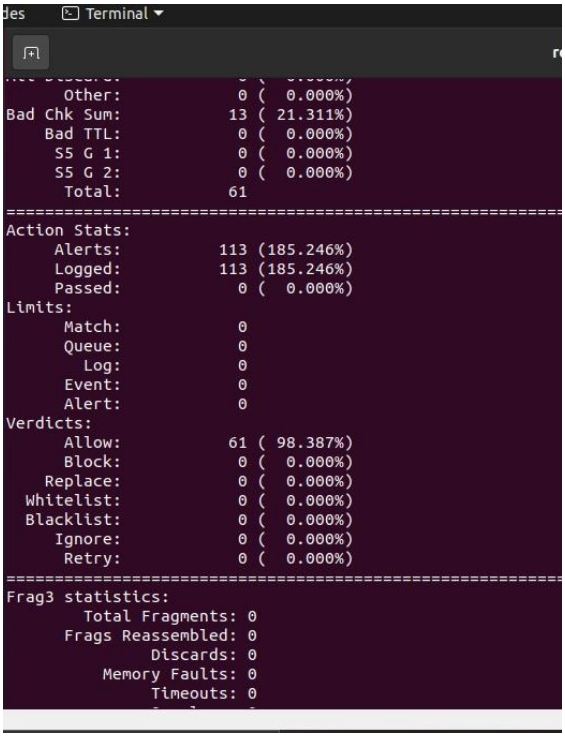
- alert icmp 192.168.0.101/24 any -> any any (msg: ”Alguien está haciendo ping”; sid: 19910316; rev: 1)
- alert icmp 192.168.0.1/24 any -> any any (msg: ”Alguien esta haciendo ping al HOST”; sid: 19910316; rev: 1)

En donde la primera regla usamos el protocolo icmp que es el común para los pings, designamos que sea de cualquier red y cualquier puerto, en el destino de igual manera que vaya a cualquier red o cualquier destino, el mensaje de la alerta, el sid único y la revisión de la regla, este sirve para futuras actualizaciones de la misma regla. En la segunda y tercera se aplicó la misma estructura el único cambio aquí fue que realizamos una regla para que reconociera el ping hacia la ip del host que este caso es nuestro router y hacia la ip de la computadora que tiene snort.
Una vez ingresadas a nuestras reglas procedemos a ejecutar Snort con el comando:

- snort -dev –c /etc/snort/snort.conf –l /var/log/snort/ -i enp0s8 –A full

Una vez ejecutado snort nos empezara a enviar alertas de las reglas y paquetes que se han recibido, junto con la información de los paquetes, un contador de paquetes y posiblemente las alertas que se van generando en tiempo real.

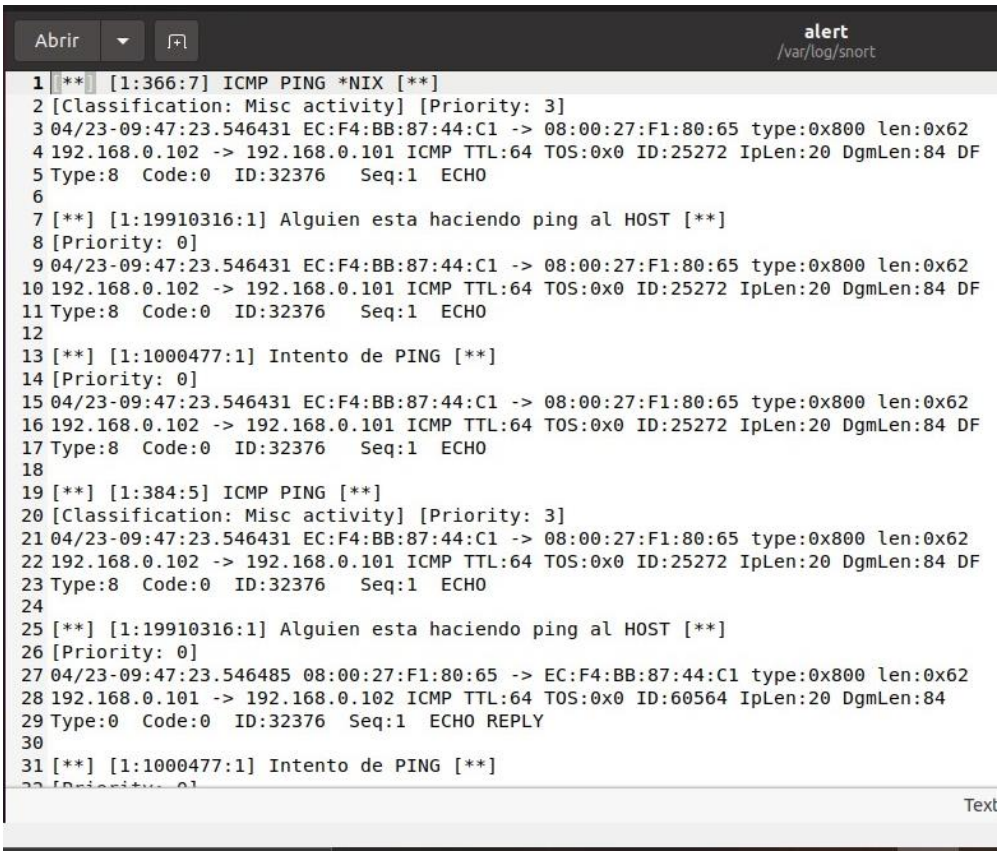




Para detener la ejecución de Snort solo usamos la combinación de teclas Ctrl + C, una vez detenido, para acceder a la lista de alertas que se han generado exitosamente usamos el comando:

- gedit /var/log/snort/alert

El cual nos abrirá el archivo con las alertas generadas por Snort.



4.4 Definición e implementación de Reglas

4.4.1 Reglas para detectar mapeo de redes y detección de puertos vulnerables

- alert tcp any any -> $HOME_NET any (msg:"Posible escaneo de puerto TCP detectado" ; flags:S; threshold: type threshold, track by_src , count 5, seconds 60; sid:1000001;)
    - **tcp:** especificamos que el protocolo a analizar se trata del protocolo TCP.
    - **any any:** El primer any se refiere a cualquier dirección IP origen. El segundo se refiere a cualquier puerto de origen.
    - **$HOME_NET any:** es una variable que representa la red interna protegida, any se refiere a cualquier puerto de destino en la red.

    - **flags:S :**especifica que la regla debe coincidir con paquetes TCP que tienen el flag SYN que se usan a menudo en escaneos de puertos para identificar puertos abiertos.
- alert udp any any -> $HOME_NET any (msg:"Posible escaneo de puerto UDP detectado" ; threshold: type threshold, track by_src , count 5, seconds 60; sid:1000002;)
- alert tcp any any -> $HOME_NET any (msg:"Potential Port Scan Detected"; flags:S; detection_filter:track by_src, count 10, seconds 60; sid:1000003; rev:1;)

4.4.2 Reglas para detectar posibles ataques DoS (Denegación de Servicio)

- alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"Potencial ataque de inundacion del enlace"; threshold: type threshold, track by_dst,count 30, seconds 120; classtype:attempted-dos; sid:10002;rev:2;)
    - **icmp:** Utilizan el protocolo ICMP (Internet Control Message Protocol), que se usa para enviar mensajes de control y diagnóstico.
    - **$EXTERNAL_NET any:** es una variable que representa cualquier red externa.
    - **classtype:attempted-dos:** classtype clasifica el tipo de ataque. attempted-dos significa que es un intento de ataque de denegación de servicio (DoS).
- alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg:"Detectado ataque de SYN Flood"; flags:S; threshold: type threshold, track by_dst, count 20, seconds 10; classtype:attempted-dos; sid:10005; rev:1;)
- alert udp $EXTERNAL_NET any -> $HOME_NET any (msg:"Detectado ataque de UDP Flood"; threshold: type threshold, track by_src, count 50, seconds 30; classtype:attempted-dos; sid:10006; rev:1;)

4.4.3 Reglas para detectar ataques de fuerza bruta

- alert tcp $EXTERNAL_NET any -> $HOME_NET 22 (msg:"Intento de ataque de fuerza bruta SSH"; flags:S; threshold:type threshold, track by_src, count 6, seconds 120; flowbits:set, ssh.brute.attempt; classtype:attempted-admin; sid:2001219; rev:1;)
    - **$HOME_NET 22**: 22 es el puerto de destino, que es el puerto utilizado para conexiones SSH.
    - **flowbits:set, ssh.brute.attempt; :** flowbits se utiliza para marcar flujos de tráfico que coinciden con ciertas condiciones y se pueden usar en reglas posteriores. ssh.brute.attempt marca el flujo actual como un intento de fuerza bruta SSH.
    - **classtype:attempted-admin:** classtype clasifica el tipo de ataque. attempted-admin significa que es un intento de obtener acceso administrativo no autorizado.
- alert tcp any any -> $HOME_NET 22 (msg:”Error de autenticacion SSH”; sid: 1000009; priority:1;)
- alert tcp any any -> $HOME_NET 23 (msg:"Error de autenticacion Telnet"; sid:1000009; priority:1;)
- alert tcp any any -> $HOME_NET 21 (msg:"Error de autenticacion FTP"; sid:1000011; priority:1;)
- alert tcp any any -> $HOME_NET 80 (msg:"Error de autenticacion HTTP"; sid:1000013; priority:1;)

4.5 Ataques con Kali Linux y varias herramientas

4.5.1 Mapeo de redes y detección de vulnerabilidades con Nmap y Metasploit

El mapeo de redes no es un ataque como tal, pero si es importante mencionarlo y realizar pruebas con ello ya que los atacantes toman este aspecto como el primer paso, ya que buscan redes cercanas y escanean los posibles puertos,

vulnerabilidades y accesos que pueden explotar para poder realizar sus ataques a las redes con dichas deficiencias.

Por lo general al usar Nmap se lanza un ping contra el objetivo, para comprobar que está disponible, despues de ello se envían una serie de paquetes. En función de cómo sean estos paquetes se distinguen dos tipos de escaneo:

- Non stealth: Se trata de un tipo de escaneo de fácil detección, pues emplea métodos de conexión TCP, ya sea realizando el típico three-way handshake completo o sin el tercer paquete.
- Stealth: Se envían paquetes con unas determinadas banderas (flags) activados de manera que pueda determinarse el estado de los puertos de manera eficaz. Algunos de las combinaciones de banderas (flags) que pueden utilizarse son F (Fin), Syn/ack, xmas tree2 o Null, sin flag. En todos estos casos un puerto cerrado responde con un reset, RST, mientras que uno abierto no responde. Debido a que no se establece una conexión TCP convencional, este escaneo resulta más difícil de detectar, por lo que es el que emplearía un atacante. Para realizar este escaneo, resulta necesario contar con permisos de administrador para poder generar este tipo de paquetes.

4.5.1.1 Pruebas con Nmap

El primer comando que se usó en Nmap fue para verificar las direcciones IP de los dispositivos que se encontraban cerca de la maquina atacante.

```
┌──(alan㉿alan)-[~]
└─$ nmap -n -sn 192.168.0.102/24
Starting Nmap 7.93 ( https://nmap.org ) at 2024-04-25 09:24 CST
Nmap scan report for 192.168.0.1
Host is up (0.00074s latency).
Nmap scan report for 192.168.0.101
Host is up (0.0012s latency).
Nmap scan report for 192.168.0.102
Host is up (0.015s latency).
Nmap done: 256 IP addresses (3 hosts up) scanned in 2.31 seconds
┌──(alan㉿alan)-[~]
└─$ nmap -n -sn 192.168.0.102/24 -oG - | awk '/Up$/{print $2}'
192.168.0.1
192.168.0.101
192.168.0.102
```

-n: La opción -n le dice a Nmap que no realice la resolución de nombres DNS.

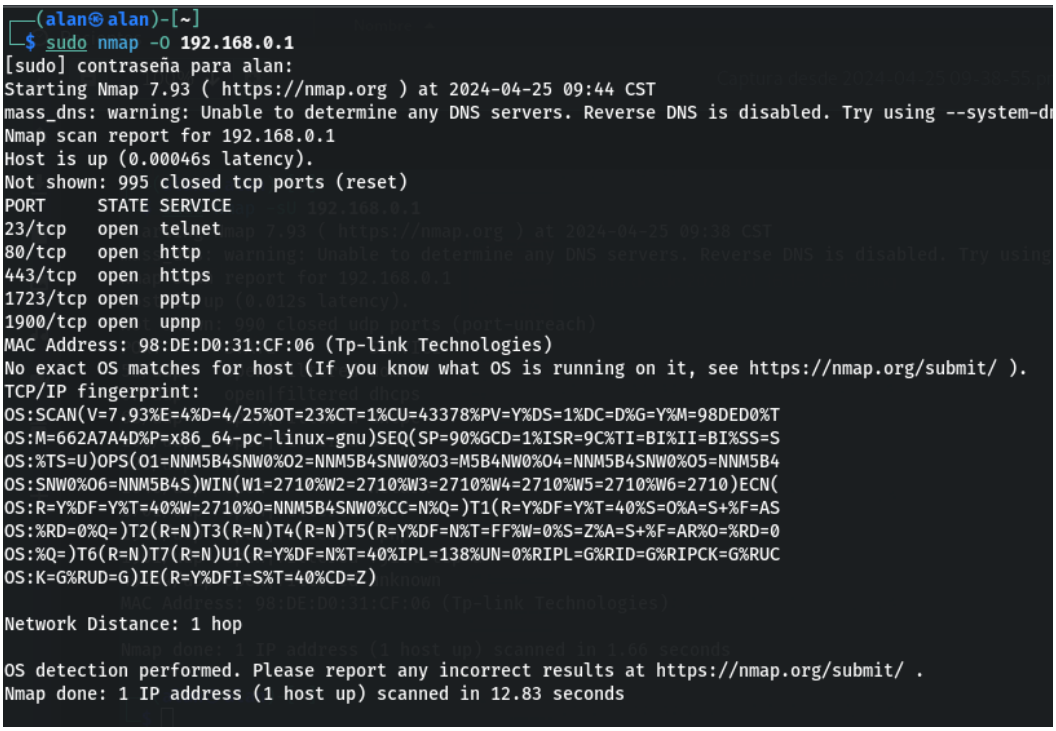
-sn: La opción -sn le indica a Nmap que realice un "ping scan". Esto significa que solo comprobará si los hosts en la red están activos y no realizará un escaneo de puertos.

Una vez analizadas las redes que se encontraban activas, podemos observar que están las direcciones IP del host y de la PC con Snort.

Despues usamos el comando -O en Nmap para detectar todos los componentes de la PC y del Host, en este caso nuestro Router.

```
┌──(alan㉿alan)-[~]
└─$ sudo nmap -O 192.168.0.101
Starting Nmap 7.93 ( https://nmap.org ) at 2024-04-25 09:45 CST
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled. Try using --syste
Nmap scan report for 192.168.0.101
Host is up (0.0013s latency).
Not shown: 996 filtered tcp ports (no-response)
PORT     STATE SERVICE
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
5357/tcp open  wsdapi
MAC Address: 00:E0:4C:68:05:49 (Realtek Semiconductor)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed p
Device type: general purpose
Running (JUST GUESSING): Microsoft Windows XP|2008|7 (89%)
OS CPE: cpe:/o:microsoft:windows_xp::sp3 cpe:/o:microsoft:windows_server_2008::sp1 cpe:/o:microsof
Aggressive OS guesses: Microsoft Windows XP SP3 (89%), Microsoft Windows Server 2008 SP1 or Windo
ows 7 (85%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.14 seconds
```

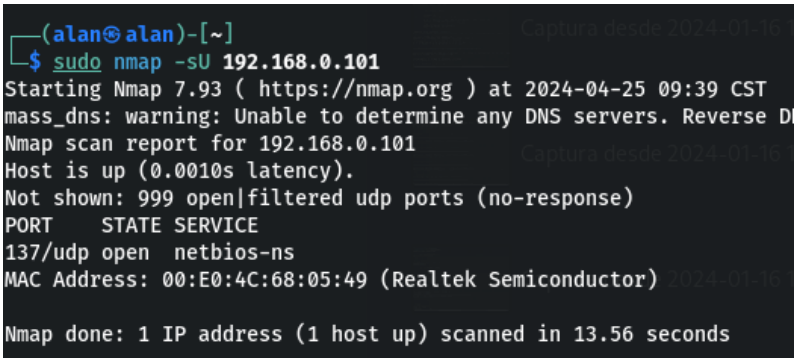
Como podemos observar en la imagen la información obtenida de nuestra PC con Snort nos muestra un pequeño reporte con los puertos con el protocolo TCP que mantiene abiertos el sistema en ese momento, además de proporcionarnos la dirección MAC e información del dispositivo como el sistema operativo que contiene o que ha tenido trabajando el dispositivo.



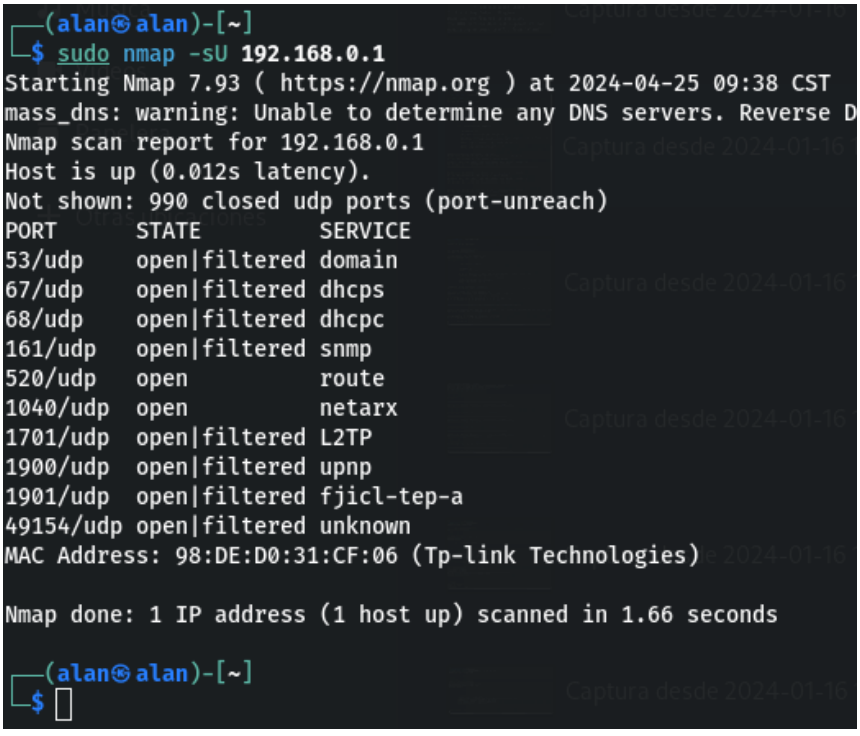
En el caso del Host, muestra los puertos que están abiertos y habilitados en el Router, asi como su dirección MAC junto con la marca del dispositivo y en este caso como es un Router con métodos de seguridad habilitados de fábrica, no nos deja ver la información del sistema operativo.

Una vez realizadas estas pruebas mientras se ejecutaba Snort, pudimos obtener en el archivo de alertas, nuestra alerta con la fecha, hora, el mensaje que designamos, asi como nos muestra la IP de origen, la IP de destino, el protocolo que se usó para él envió de paquetes

También se realizó una prueba para escanear los puertos abiertos con el protocolo UDP con el comando -sU de Nmap.



Como podemos observar la computadora con Snort solo tiene un puerto UDP abierto en ese momento.



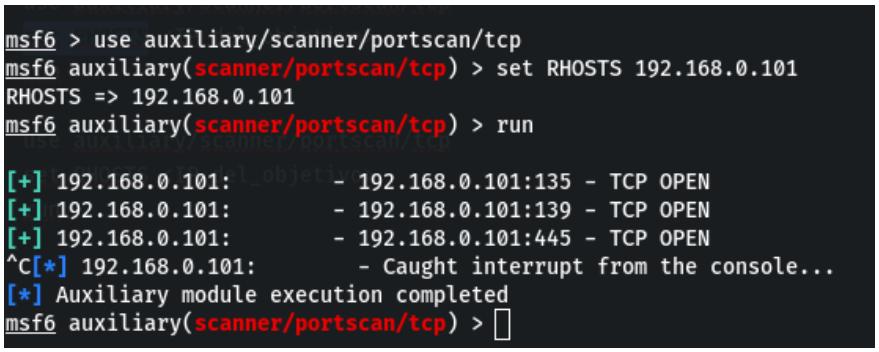
Y el Router tiene todos esos puertos UDP abiertos en ese momento. La mayoría con filtro debido a la configuración.

De igual manera Snort nos ha creado la alerta del escaneo de puertos UDP como en el caso de los puertos TCP.

4.5.1.2 Pruebas con Metasploit

En este caso se usó el módulo auxiliar que ofrece Metasploit para analizar los puertos vulnerables de TCP.

- set RHOSTS: se usó para definir la IP de la cual se obtendrían los puertos vulnerables, en este caso solo los TCP.
- Para hacer ejecución del módulo a usar se usa el comando run o exploit.



Como podemos observar realiza un muestreo completo de todos los puertos abiertos y vulnerables en ese momento.

```
[**] [1:1000003:1] Potential Port Scan Detected [**]
[Priority: 0]
05/16-10:17:38.805846 EC:F4:BB:87:44:C1 -> 08:00:27:F1:80:65 type:0x800 len:0x4A
192.168.0.102:42497 -> 192.168.0.101:345 TCP TTL:64 TOS:0x0 ID:41357 IpLen:20 DgmLen:60 DF
******S* Seq: 0x43954142  Ack: 0x0  Win: 0xFAF0  TcpLen: 40
TCP Options (5) => MSS: 1460 SackOK TS: 1636195278 0 NOP WS: 7
```

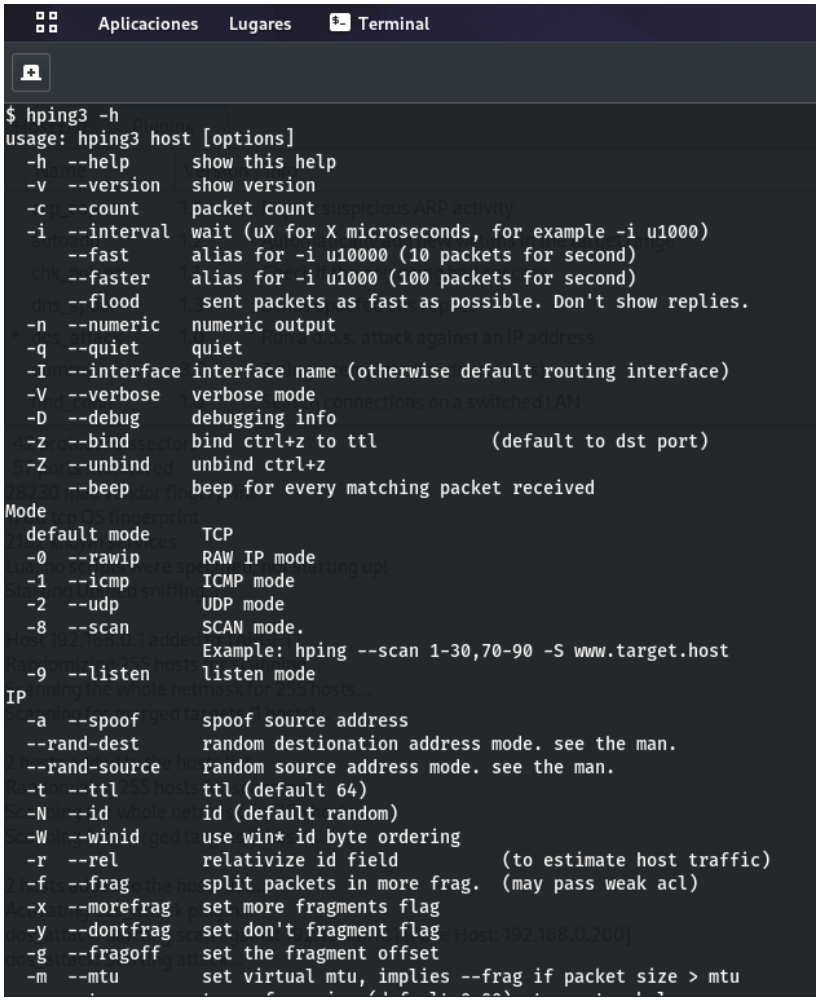
Como podemos observar, Snort una vez más nos registró la alerta de que se estaba realizando el escaneo de puertos en ese momento.

Nmap sirve para analizar puertos no solo con esto protocolos, sino también los puertos con protocolos ICMP, SCTP, IP y ARP, por lo cual es una buena herramienta para el escaneo de puertos y vulnerabilidades.

4.5.2 Pruebas de ataques DoS con Hping3

Hping3 es una herramienta de línea de comandos que se usa para la manipulación y el análisis de paquetes de red en la capa de transporte (TCP/UDP). Tiene diversas

funcionalidades y se utiliza en una variedad de casos, pero para esta situación se usó para generar tráfico de manera personalizada para simular ataques DoS.



4.5.2.1 Pruebas de ataque inundación de enlace

En este caso se usó para generar un ataque de inundacion ICMP. Con la opción -1 –icmp especificamos que el ataque será al protocolo antes mencionado.

```
┌──(alan㉿alan)-[~]
└─$ sudo hping3 -1 --icmp 192.168.0.101
HPING 192.168.0.101 (eth0 192.168.0.101): icmp mode set, 28 headers + 0 data bytes
len=46 ip=192.168.0.101 ttl=64 id=57747 icmp_seq=0 rtt=7.7 ms
len=46 ip=192.168.0.101 ttl=64 id=57982 icmp_seq=1 rtt=3.6 ms
len=46 ip=192.168.0.101 ttl=64 id=58217 icmp_seq=2 rtt=7.6 ms
len=46 ip=192.168.0.101 ttl=64 id=58309 icmp_seq=3 rtt=7.5 ms
len=46 ip=192.168.0.101 ttl=64 id=58338 icmp_seq=4 rtt=3.4 ms
len=46 ip=192.168.0.101 ttl=64 id=58409 icmp_seq=5 rtt=3.4 ms
len=46 ip=192.168.0.101 ttl=64 id=58601 icmp_seq=6 rtt=3.3 ms
len=46 ip=192.168.0.101 ttl=64 id=58827 icmp_seq=7 rtt=3.2 ms
len=46 ip=192.168.0.101 ttl=64 id=59032 icmp_seq=8 rtt=3.1 ms
^C
--- 192.168.0.101 hping statistic ---
9 packets transmitted, 9 packets received, 0% packet loss
round-trip min/avg/max = 3.1/4.8/7.7 ms
```

Como podemos observar el comando ejecutado comienza a mandar una serie de paquetes ping dirigidos a ICMP hasta que el atacante desee detener la ejecución.

```
[**] [1:10002:2] Potencial ataque de inundacion del enlace [**]
[Classification: Attempted Denial of Service] [Priority: 2]
05/02-09:33:24.206185 EC:F4:BB:87:44:C1 -> 08:00:27:F1:80:65 type:0x800 len:0x3C
192.168.0.102 -> 192.168.0.101 ICMP TTL:64 TOS:0x0 ID:43149 IpLen:20 DgmLen:28
Type:8  Code:0  ID:37387   Seq:1792  ECHO
```

Como resultado Snort generara la alerta con el mensaje designado, además de la clasificación que nosotros designamos que fue ataque DoS, también muestra las direcciones MAC de origen y destino, asi como las respectivas direcciones IP, también muestra la información de los paquetes, como el tiempo de vida, el id, entre otros, en este caso nos muestra la alerta que está recibiendo paquetes ECHO.

Snort también funciona con reglas predefinidas por la comunidad de usuarios, empresas y organizaciones que lo utilizan (grupos, foros, asociaciones), en este caso, el ataque genero también alertas diversas como en la siguiente imagen, que está denotando un posible escaneo ICMP de Nmap e intentos de ping ICMP cuando en este caso no es asi.

```
[**] [1:469:3] ICMP PING NMAP [**]
[Classification: Attempted Information Leak] [Priority: 2]
05/02-09:33:24.206185 EC:F4:BB:87:44:C1 -> 08:00:27:F1:80:65 type:0x800 len:0x3C
192.168.0.102 -> 192.168.0.101 ICMP TTL:64 TOS:0x0 ID:43149 IpLen:20 DgmLen:28
Type:8  Code:0  ID:37387   Seq:1792  ECHO
[Xref => http://www.whitehats.com/info/IDS162]

[**] [1:384:5] ICMP PING [**]
[Classification: Misc activity] [Priority: 3]
05/02-09:33:24.206185 EC:F4:BB:87:44:C1 -> 08:00:27:F1:80:65 type:0x800 len:0x3C
192.168.0.102 -> 192.168.0.101 ICMP TTL:64 TOS:0x0 ID:43149 IpLen:20 DgmLen:28
Type:8  Code:0  ID:37387   Seq:1792  ECHO
```

Las alertas muestran la clasificación de Misc activity debido a que está denotando que hay actividad sospechosa con el protocolo ICMP, además de que afirma la información de la alerta que se definió para el ataque DoS como las IP 's, los paquetes ECHO, además de la fecha y hora.

4.5.2.2 Pruebas de ataque SYN Flood

En el segundo caso se generó un ataque SYN Flood en el cual se envía una gran cantidad de paquetes SYN haciendo que se mantengan las conexiones TCP abiertas, esperando la respuesta ACK que nunca llega. En este caso se usó el comando -S –syn para enviar paquetes a la bandera SYN.

```
┌──(alan㉿alan)-[~]
└─$ sudo hping3 -S --syn 192.168.0.101
HPING 192.168.0.101 (eth0 192.168.0.101): S set, 40 headers + 0 data bytes
len=46 ip=192.168.0.101 ttl=64 DF id=0 sport=0 flags=RA seq=0 win=0 rtt=3.7 ms
len=46 ip=192.168.0.101 ttl=64 DF id=0 sport=0 flags=RA seq=1 win=0 rtt=3.7 ms
len=46 ip=192.168.0.101 ttl=64 DF id=0 sport=0 flags=RA seq=2 win=0 rtt=3.6 ms
len=46 ip=192.168.0.101 ttl=64 DF id=0 sport=0 flags=RA seq=3 win=0 rtt=3.6 ms
len=46 ip=192.168.0.101 ttl=64 DF id=0 sport=0 flags=RA seq=4 win=0 rtt=3.5 ms
len=46 ip=192.168.0.101 ttl=64 DF id=0 sport=0 flags=RA seq=5 win=0 rtt=3.4 ms
len=46 ip=192.168.0.101 ttl=64 DF id=0 sport=0 flags=RA seq=6 win=0 rtt=3.4 ms
len=46 ip=192.168.0.101 ttl=64 DF id=0 sport=0 flags=RA seq=7 win=0 rtt=7.3 ms
len=46 ip=192.168.0.101 ttl=64 DF id=0 sport=0 flags=RA seq=8 win=0 rtt=3.2 ms
len=46 ip=192.168.0.101 ttl=64 DF id=0 sport=0 flags=RA seq=9 win=0 rtt=7.2 ms
len=46 ip=192.168.0.101 ttl=64 DF id=0 sport=0 flags=RA seq=10 win=0 rtt=7.1 ms
len=46 ip=192.168.0.101 ttl=64 DF id=0 sport=0 flags=RA seq=11 win=0 rtt=7.0 ms
len=46 ip=192.168.0.101 ttl=64 DF id=0 sport=0 flags=RA seq=12 win=0 rtt=3.0 ms
len=46 ip=192.168.0.101 ttl=64 DF id=0 sport=0 flags=RA seq=13 win=0 rtt=6.9 ms
^C
--- 192.168.0.101 hping statistic ---
14 packets transmitted, 14 packets received, 0% packet loss
round-trip min/avg/max = 3.0/4.8/7.3 ms
```

Como podemos observar de igual manera se envían paquetes hasta que el atacante lo detenga.

```
[**] [1:10005:1] Detectado ataque de SYN Flood [**]
[Classification: Attempted Denial of Service] [Priority: 2]
05/02-09:28:11.981478 EC:F4:BB:87:44:C1 -> 08:00:27:F1:80:65 type:0x800 len:0x3C
192.168.0.200:47591 -> 192.168.0.101:60 TCP TTL:64 TOS:0x0 ID:32487 IpLen:20 DgmLen:40
******S* Seq: 0x19C1ADAB  Ack: 0x0  Win: 0x7FFF  TcpLen: 20
```

Como podemos ver Snort genero la alerta exitosamente. De igual manera nos muestra el ID, el mensaje que asignamos, la clasificación de ataque DoS y la prioridad/ importancia que hay que darle a este tipo de ataques.

4.5.2.3 Pruebas de ataque UDP Flood

En la tercera prueba se generó un ataque UDP Flood en donde se envían paquetes UDP falsificados o no solicitados, con lo cual se genera una carga que puede provocar saturación de la red. En este caso se usó el comando -1 --udp en donde definimos que el ataque se realice a protocolo UDP.

```
┌──(alan㉿alan)-[~]
└─$ sudo hping3 -1 --udp 192.168.0.101
HPING 192.168.0.101 (eth0 192.168.0.101): icmp mode set, 28 headers + 0 data bytes
len=46 ip=192.168.0.101 ttl=64 id=54256 icmp_seq=0 rtt=3.6 ms
len=46 ip=192.168.0.101 ttl=64 id=54450 icmp_seq=1 rtt=3.5 ms
len=46 ip=192.168.0.101 ttl=64 id=54532 icmp_seq=2 rtt=3.5 ms
len=46 ip=192.168.0.101 ttl=64 id=54648 icmp_seq=3 rtt=7.4 ms
len=46 ip=192.168.0.101 ttl=64 id=54692 icmp_seq=4 rtt=11.3 ms
len=46 ip=192.168.0.101 ttl=64 id=54701 icmp_seq=5 rtt=7.2 ms
len=46 ip=192.168.0.101 ttl=64 id=54882 icmp_seq=6 rtt=7.2 ms
len=46 ip=192.168.0.101 ttl=64 id=55058 icmp_seq=7 rtt=7.1 ms
len=46 ip=192.168.0.101 ttl=64 id=55249 icmp_seq=8 rtt=3.0 ms
len=46 ip=192.168.0.101 ttl=64 id=55476 icmp_seq=9 rtt=3.0 ms
len=46 ip=192.168.0.101 ttl=64 id=55601 icmp_seq=10 rtt=6.9 ms
len=46 ip=192.168.0.101 ttl=64 id=55850 icmp_seq=11 rtt=10.9 ms
len=46 ip=192.168.0.101 ttl=64 id=55887 icmp_seq=12 rtt=6.8 ms
len=46 ip=192.168.0.101 ttl=64 id=56013 icmp_seq=13 rtt=2.7 ms
len=46 ip=192.168.0.101 ttl=64 id=56194 icmp_seq=14 rtt=2.7 ms
len=46 ip=192.168.0.101 ttl=64 id=56330 icmp_seq=15 rtt=2.6 ms
len=46 ip=192.168.0.101 ttl=64 id=56559 icmp_seq=16 rtt=6.6 ms
len=46 ip=192.168.0.101 ttl=64 id=56571 icmp_seq=17 rtt=2.5 ms
len=46 ip=192.168.0.101 ttl=64 id=56665 icmp_seq=18 rtt=6.4 ms
len=46 ip=192.168.0.101 ttl=64 id=56899 icmp_seq=19 rtt=6.3 ms
len=46 ip=192.168.0.101 ttl=64 id=57096 icmp_seq=20 rtt=6.3 ms
len=46 ip=192.168.0.101 ttl=64 id=57201 icmp_seq=21 rtt=6.2 ms
^C
--- 192.168.0.101 hping statistic ---
22 packets transmitted, 22 packets received, 0% packet loss
round-trip min/avg/max = 2.5/5.6/11.3 ms
```

Como podemos observar se mandan múltiples paquetes hasta que se detenga el ataque.

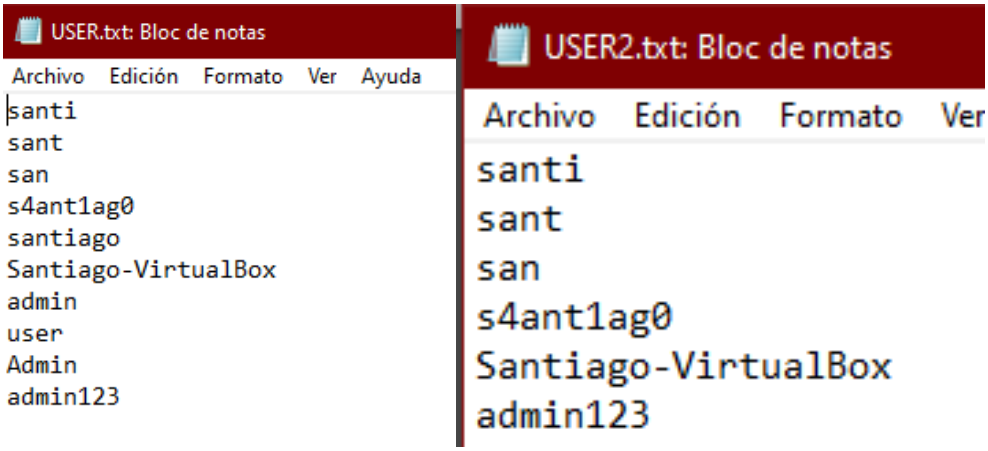
```
[**] [1:10006:1] Detectado ataque de UDP Flood [**]
[Classification: Attempted Denial of Service] [Priority: 2]
05/02-10:40:54.114343 EC:F4:BB:87:44:C1 -> 08:00:27:F1:80:65 type:0x800 len:0x3C
192.168.0.102:60859 -> 192.168.0.101:965 UDP TTL:41 TOS:0x0 ID:51121 IpLen:20 DgmLen:28
Len: 0
```

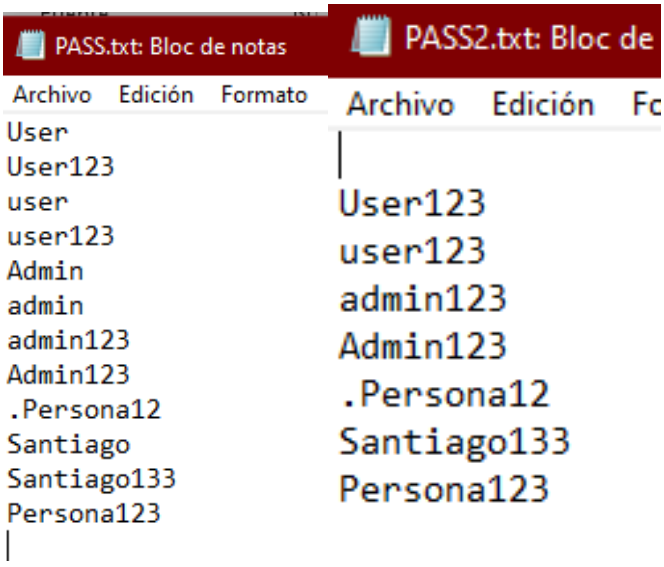
Finalmente Snort genera la alerta como en los casos anteriores y siendo exitosa nuestra regla.

Es importante determinar que Snort puede tener falsos positivos o falsos negativos dentro de un ambiente más amplio, dado que no es el caso las alertas generadas han sido exitosas, es importante el uso de las reglas de este tipo para poder identificar y responder de manera eficiente a los ataques DoS para asi disminuir el impacto de disponibilidad de la red.

4.5.3 Ataque de fuerza bruta con HYDRA

Hydra como ya habíamos visto realiza ataques de fuerza bruta con la ayuda de diccionarios ya sean públicos o generados por uno mismo, aprovechando los protocolos compatibles o puertos en específico, en este caso se realizaron ataques a los protocolos SSH puerto 22, Telnet puerto 23, FTP puerto 20/21, HTTP puerto 80. Usando 2 diccionarios en archivo de texto plano (.txt) con las contraseñas de administrador del sistema, telnet además del servidor FTP y dos diccionarios con los usuarios de estos. En este caso para usuario fueron USER.txt y USER2.txt; y para las contraseñas fueron PASS.txt y PASS2.txt.



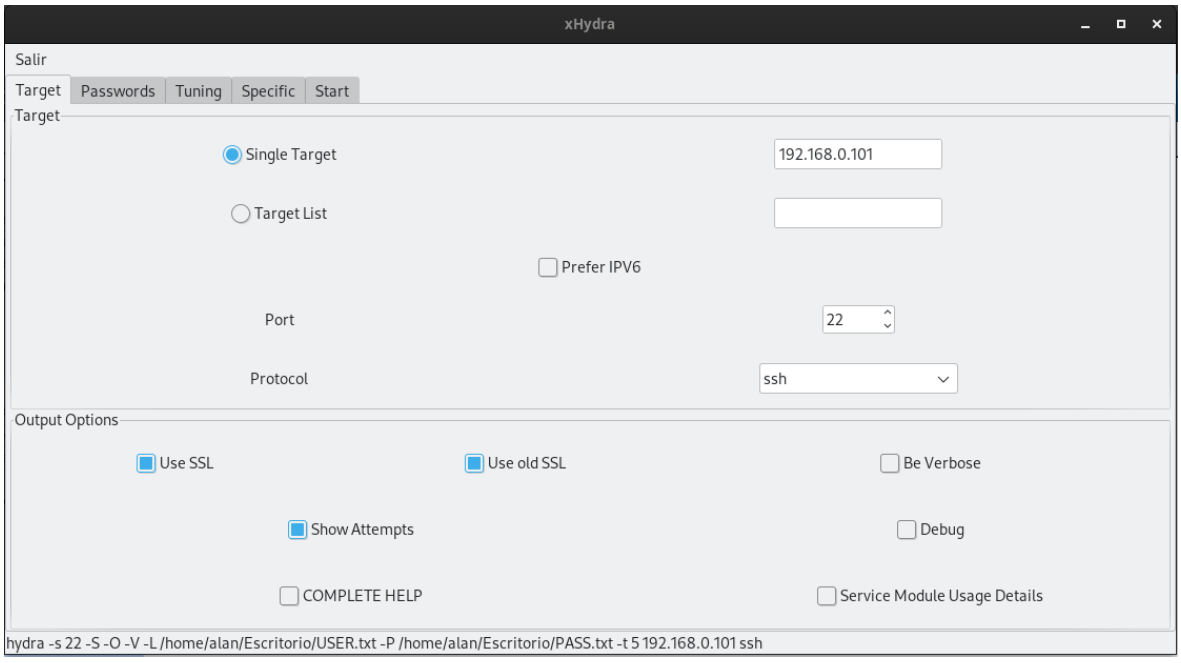


En los primeros archivos de usuario y contraseña se usaron para tratar de hacer acertados y efectivos los ataques de cada protocolo ya que contiene las contraseñas

además de los nombres de usuario correctos, los segundos fueron generados para crear ataques fallidos de dichos protocolos y se cree una alerta de acceso fallido. Hydra actualmente tiene dos maneras de realizar los ataques por fuerza bruta, una es por medio de la terminal y la otra se trata de una interfaz gráfica con la misma funcionalidad. En este caso se decidió realizar los ataques con la interfaz gráfica para verificar la información de manera acertada, además de que al final de cada ejecución nos muestra el comando por si se desea ejecutar desde la terminal.

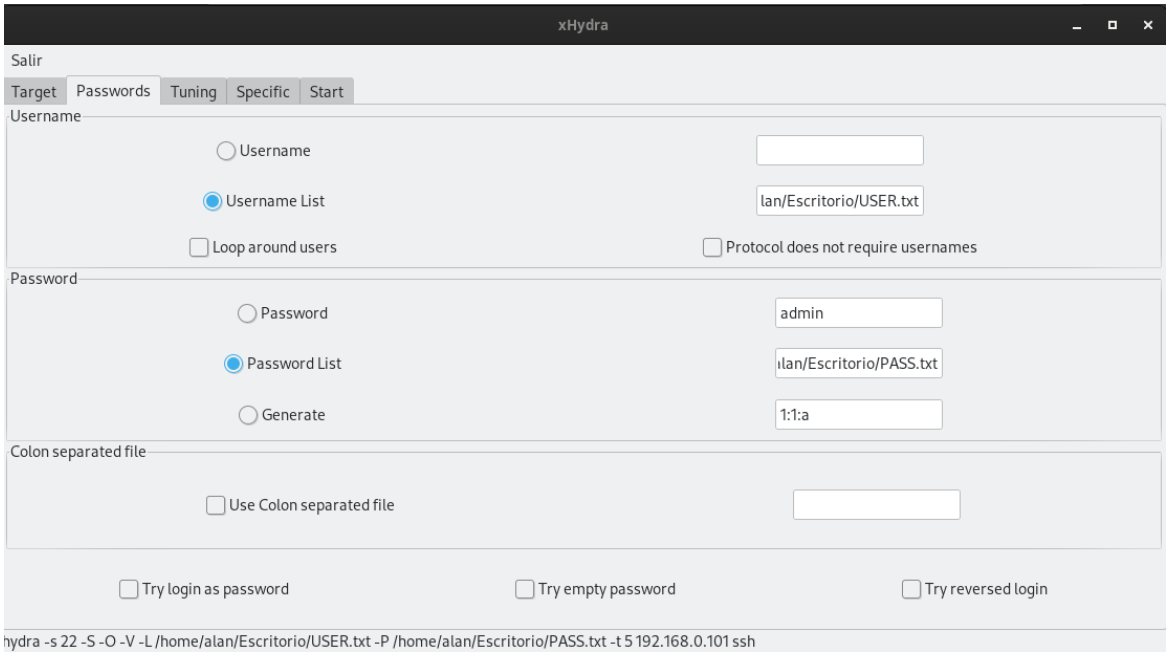
4.5.3.1 Ataque a SSH

Al abrir la interfaz gráfica de hydra lo primero que nos mostrara es un apartado para ingresar la información del objetivo, acepta la opción de un objetivo o múltiples objetivos por medio de un archivo de texto, en este caso fue un objetivo, la IP de la pc con Snort que es la 192.168.0.101, también nos indica que debemos ingresar el puerto por el cual queremos realizar el ataque y seleccionamos el protocolo, en este caso SSH por el puerto 22, también habilitamos la opción de SSL y que muestre los intentos en la salida.

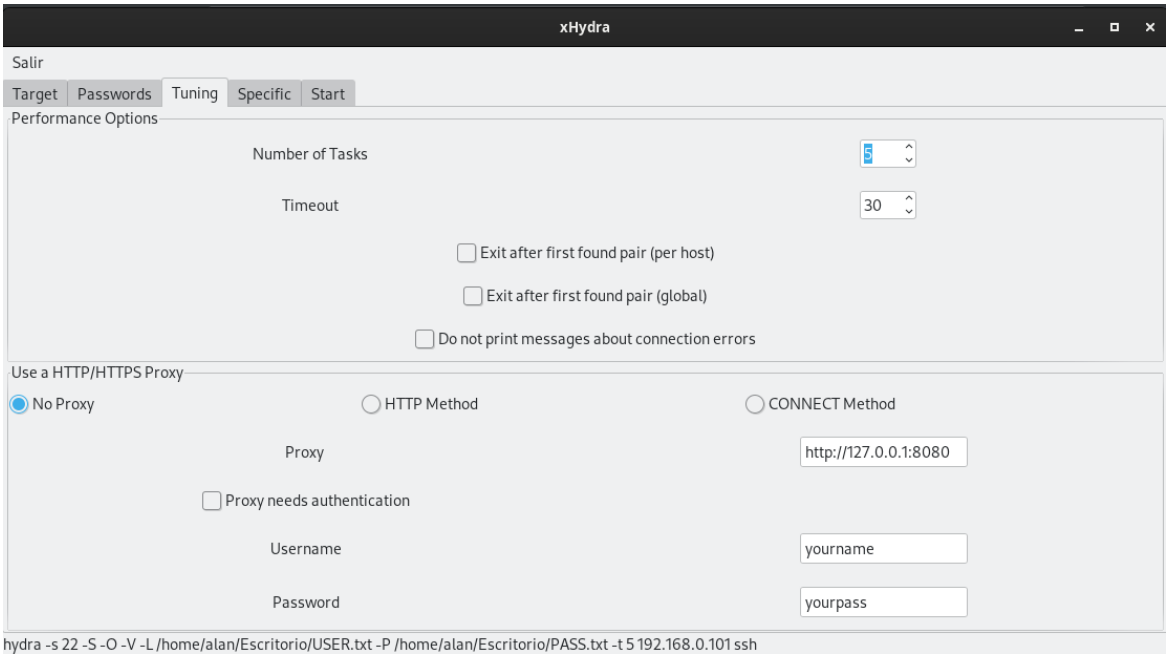


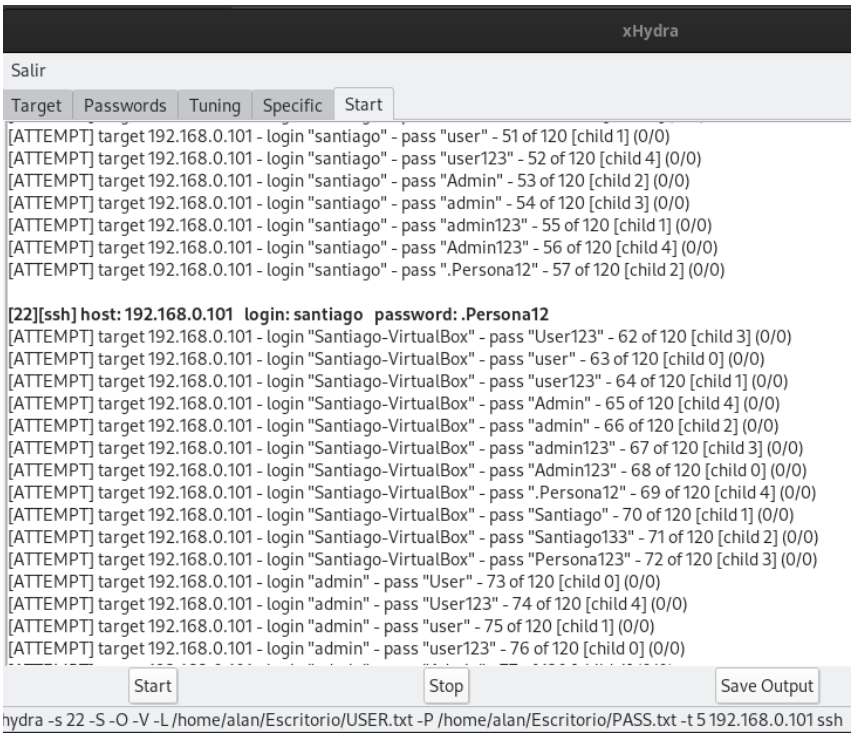
Despues en la siguiente pestaña nos pedirá que ingresemos el usuario y la contraseña en caso de ser solo uno en cada caso, o que seleccionemos el archivo

que contiene nuestra lista de usuarios y contraseñas que en este caso fue nuestra prueba con el archivo USER.txt y PASS.txt



Despues en la siguiente pestaña nos solicita ingresar el número de intentos y el tiempo de estos, para ello seleccionamos uno razonable que era de 5 intentos para que no generar alguna sospecha en el ataque. Además se puede configurar el proxy, dado que es un solo objetivo no fue necesario asi que no se habilito.





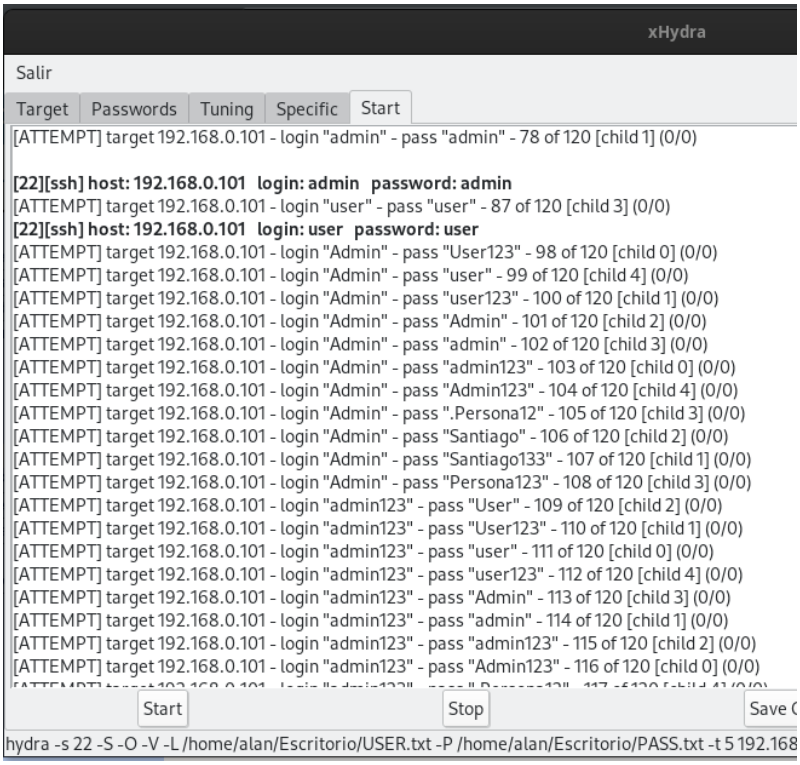
Una vez configurado podemos observar que va realizando las pruebas de las contraseñas, además como podemos ver al final de la interfaz nos muestra el comando como si se estuviera ingresando por medio de la terminal, si se quisiera hacer este ataque desde la terminal seria con el comando:

- hydra -s 22 -S -O -V -L /home/alan/Escritorio/USER.txt -P /home/alan/Escritorio/PASS.txt -t 5 192.168.0.101 ssh

En donde:

- hydra: es el comando principal de la herramienta
- -s 22: especifica el puerto por el cual se realiza el ataque
- -S: indica que se debe usar SSL
- -O: indica que hydra ignore contraseñas en blanco
- -V: indica que hydra mostrara la información durante la ejecución
- -L /home/alan/Escritorio/USER.txt: indica la ruta del archivo que contiene los nombres de usuario
- -P /home/alan/Escritorio/PASS.txt: indica la ruta del archivo que contiene las contraseñas

- -t 5: indica el número de conexiones simultaneas para realizar el ataque
- 192.168.0.101: es la dirección IP de la victima
- Ssh: es el protocolo al cual se quiere acceder



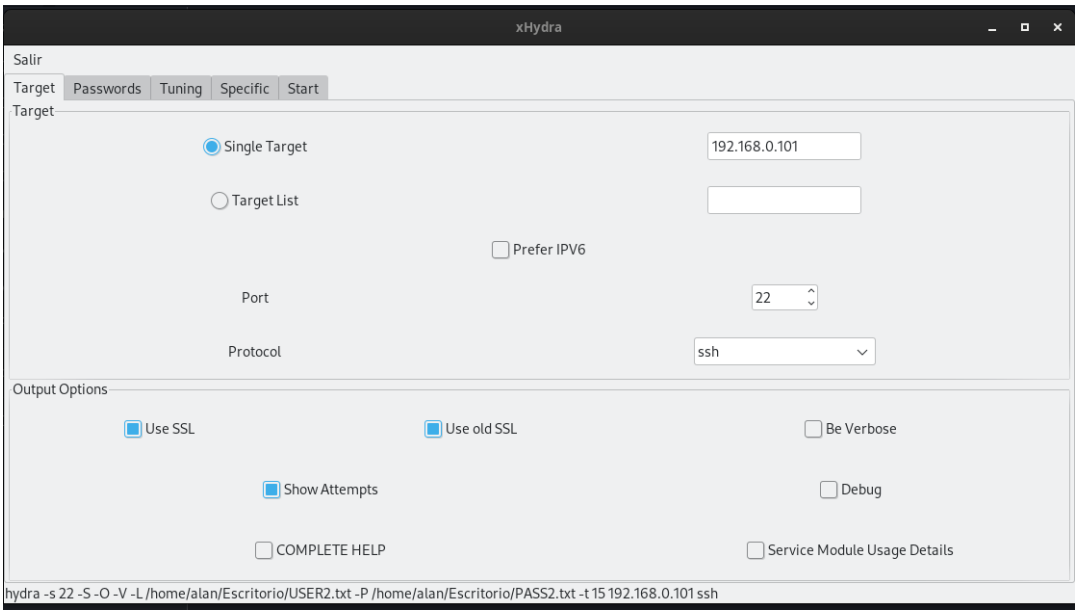
Durante los intentos, cuando hydra descubre usuarios y contraseñas de manera exitosa estas serán resaltadas de manera que se pueden identificar del resto de los intentos.

```
[**] [1:2001219:1] Intento de ataque de fuerza bruta SSH [**]
[Classification: Attempted Administrator Privilege Gain] [Priority: 1]
05/09-10:43:18.633064 EC:F4:BB:87:44:C1 -> 08:00:27:F1:80:65 type:0x800 len:0x4A
192.168.0.102:44560 -> 192.168.0.101:22 TCP TTL:64 TOS:0x0 ID:31405 IpLen:20 DgmLen:60 DF
******S* Seq: 0x8E720FF  Ack: 0x0  Win: 0xFAF0  TcpLen: 40
TCP Options (5) => MSS: 1460 SackOK TS: 3620179443 0 NOP WS: 7
```

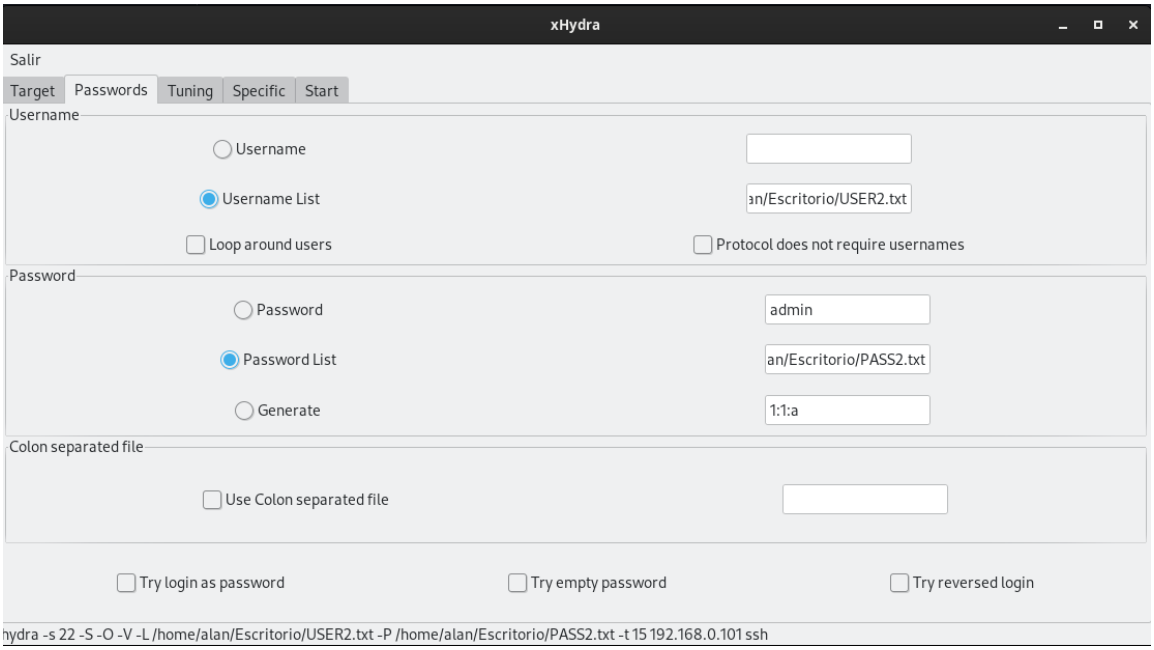
Como podemos observar la regla funciono exitosamente mostrando el mensaje designado, con la clasificación de intento de obtención de privilegios de administrador.

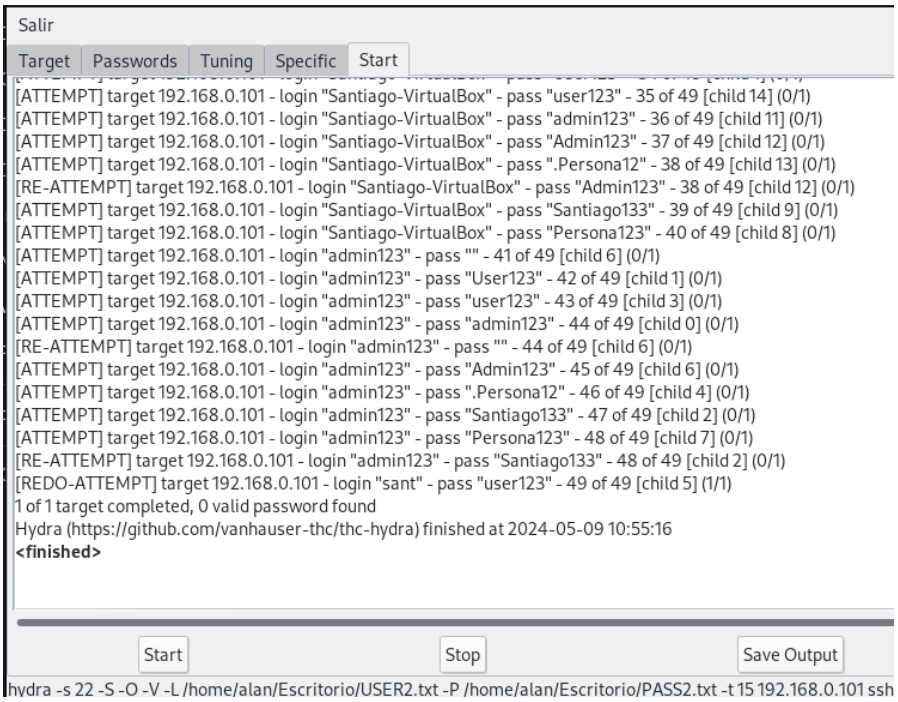
4.5.3.2 Segundo ataque a SSH

Para el segundo intento el objetivo, protocolo y puerto se dejaron con los mismos datos al igual que la opción de salida de SSL y que muestre los intentos.



En usuarios y contraseñas se cambiaron los archivos por USER2.txt y PASS2.txt





En la salida nos podemos dar cuenta de que ninguna contraseña o usuario coincidieron, esto dando como un ataque no realizado y nuestra regla funcionando.

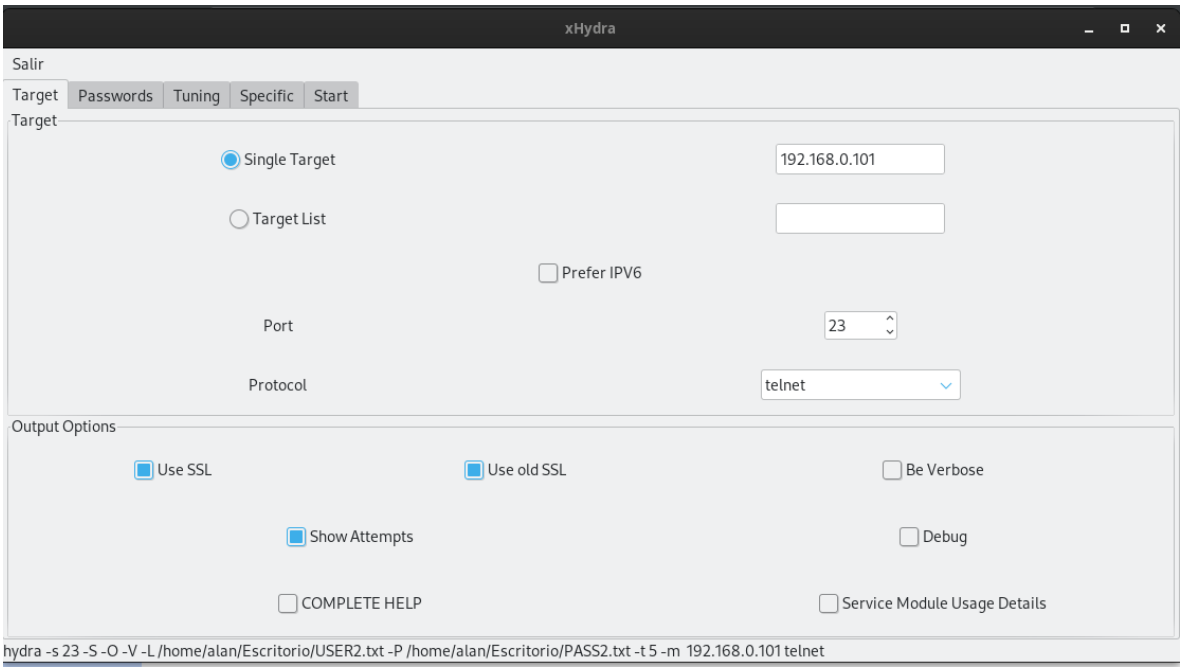
```
[**] [1:1000009:1] "Error de autenticacion SSH" [**]
[Priority: 0]
05/09-10:55:06.520067 EC:F4:BB:87:44:C1 -> 08:00:27:F1:80:65 type:0x800 len:0x4A
192.168.0.102:60768 -> 192.168.0.101:22 TCP TTL:64 TOS:0x0 ID:22025 IpLen:20 DgmLen:60 DF
******S* Seq: 0x3433F465  Ack: 0x0  Win: 0xFAF0  TcpLen: 40
TCP Options (5) => MSS: 1460 SackOK TS: 3620887374 0 NOP WS: 7
```

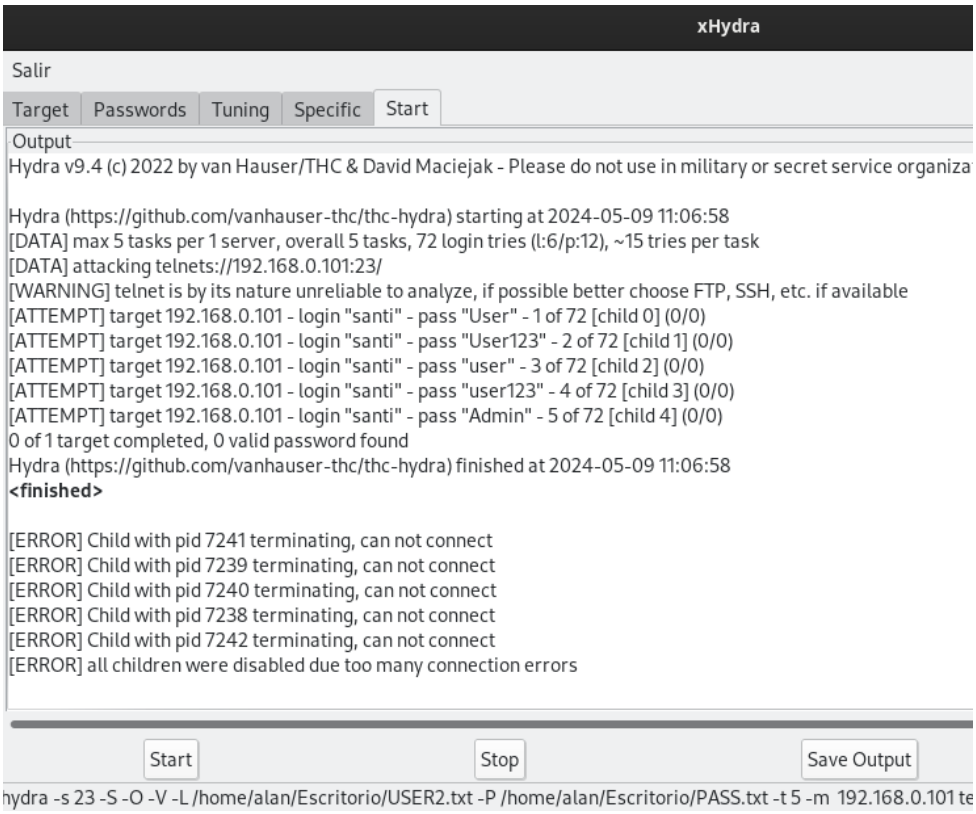
En este caso podemos observar que el ataque no fue exitoso, pero la regla funciono, ya que nos está avisando que hubo un error de autenticacion con el protocolo SSH.

4.5.3.3 Ataque a Telnet

Para el ataque a telnet el objetivo sigue siento el mismo, en este caso se cambiaron el puerto y el protocolo por el 23 y Telnet respectivamente, el comando cambio viéndose:

- hydra -s 23 -S -O -V -L /home/alan/Escritorio/USER2.txt -P /home/alan/Escritorio/PASS2.txt -t 5 -m 192.168.0.101 telnet





En este caso el router como tiene seguridad por defecto cortaba la conexión para realizar la prueba de telnet, pero la prueba para Snort fue exitosa.

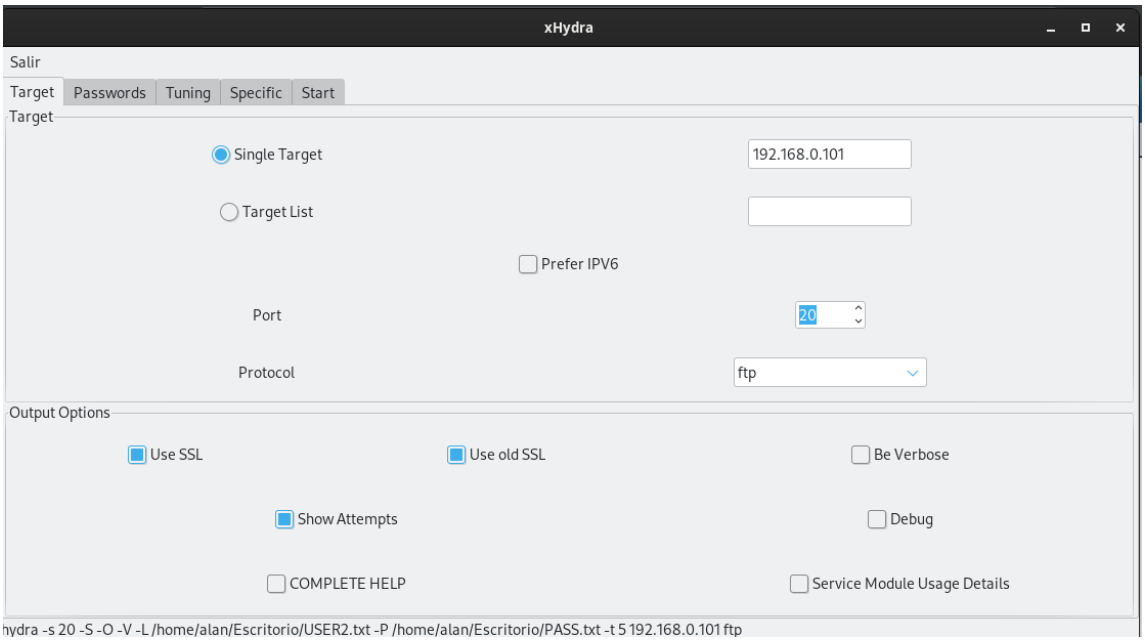
```
[**] [1:1000009:1] Error de autenticación Telnet [**]
[Priority: 0]
05/09-11:06:58.296050 EC:F4:BB:87:44:C1 -> 08:00:27:F1:80:65 type:0x800 len:0x4A
192.168.0.102:56274 -> 192.168.0.101:23 TCP TTL:64 TOS:0x0 ID:56487 IpLen:20 DgmLen:60 DF
******S* Seq: 0x97EA5EF  Ack: 0x0  Win: 0xFAF0  TcpLen: 40
TCP Options (5) => MSS: 1460 SackOK TS: 3621599195 0 NOP WS: 7
```

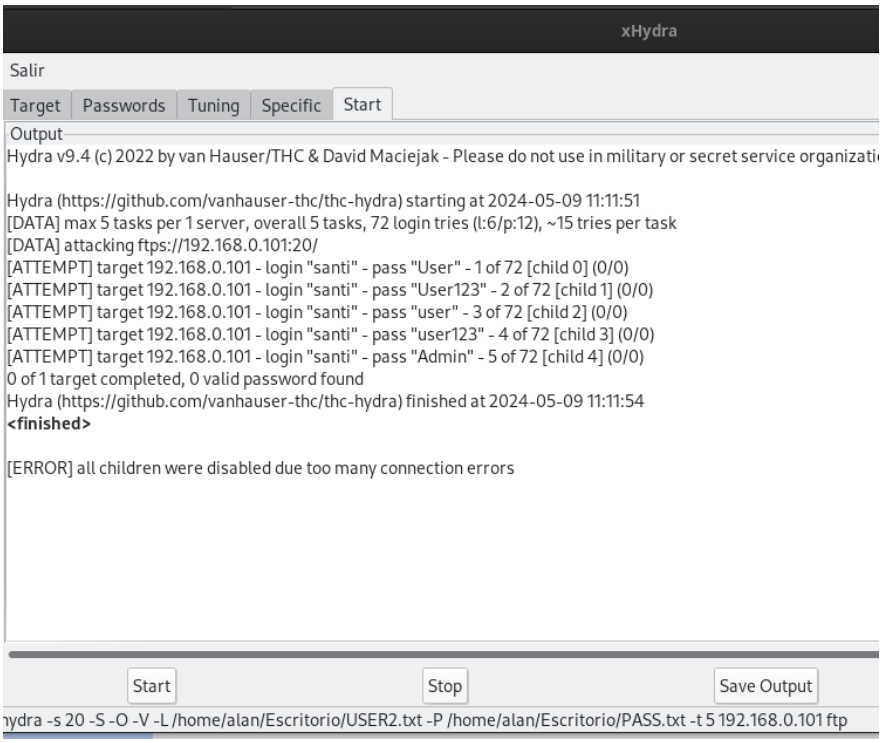
En este caso aunque telnet cerraba la conexión, Snort fue capaz de analizar el tráfico y lanzar la alerta, esto debido a que adaptamos la regla para que si detectaba paquetes por el puerto 23, nos alertara sobre la actividad en el puerto.

4.5.3.4 Ataque a FTP

Para el ataque a FTP el objetivo sigue siento el mismo, en este caso se cambiaron el puerto y el protocolo por el 20 y FTP respectivamente, el comando cambio viéndose:

- hydra -s 20 -S -O -V -L /home/alan/Escritorio/USER.txt -P /home/alan/Escritorio/PASS.txt -t 5 192.168.0.101 ftp





Como se ha comentado el router tiene seguridad por defecto cortaba la conexión para realizar la prueba de ftp, pero la prueba para Snort fue exitosa.

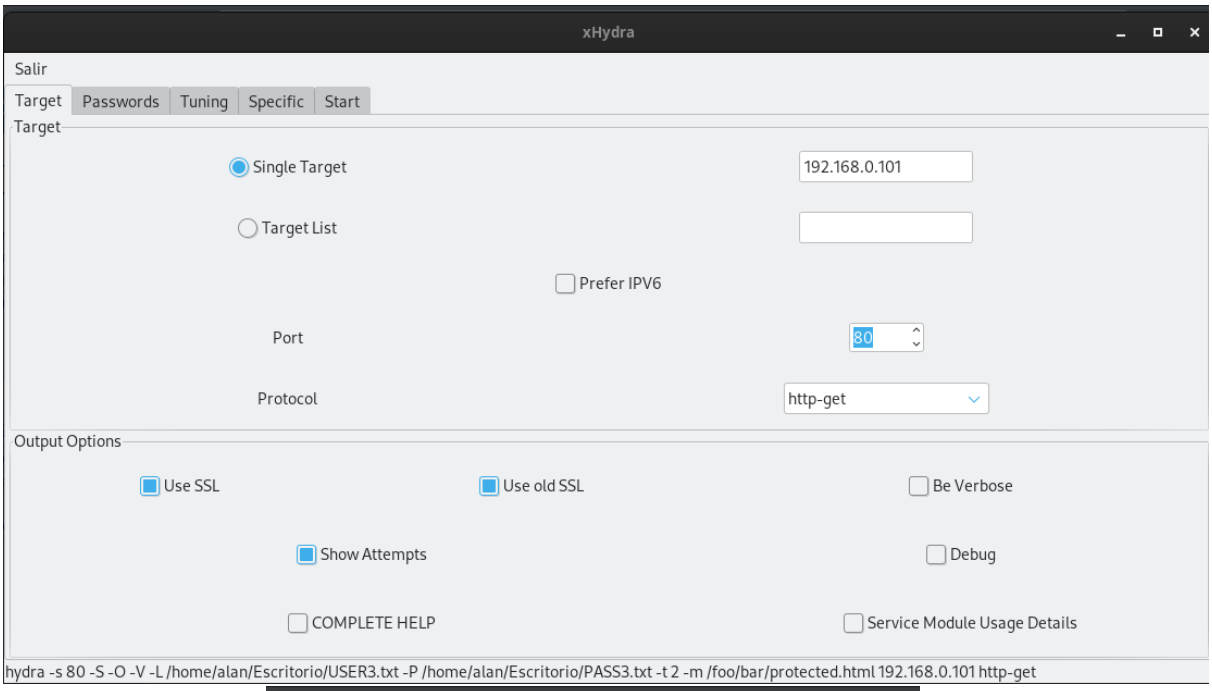
```
[**] [1:1000011:1] Error de autenticación FTP [**]
[Priority: 0]
05/09-11:13:42.159421 EC:F4:BB:87:44:C1 -> 08:00:27:F1:80:65 type:0x800 len:0x4A
192.168.0.102:39252 -> 192.168.0.101:21 TCP TTL:64 TOS:0x0 ID:14652 IpLen:20 DgmLen:60 DF
******S* Seq: 0xE3F0A43E  Ack: 0x0  Win: 0xFAF0  TcpLen: 40
TCP Options (5) => MSS: 1460 SackOK TS: 3622003083 0 NOP WS: 7
```

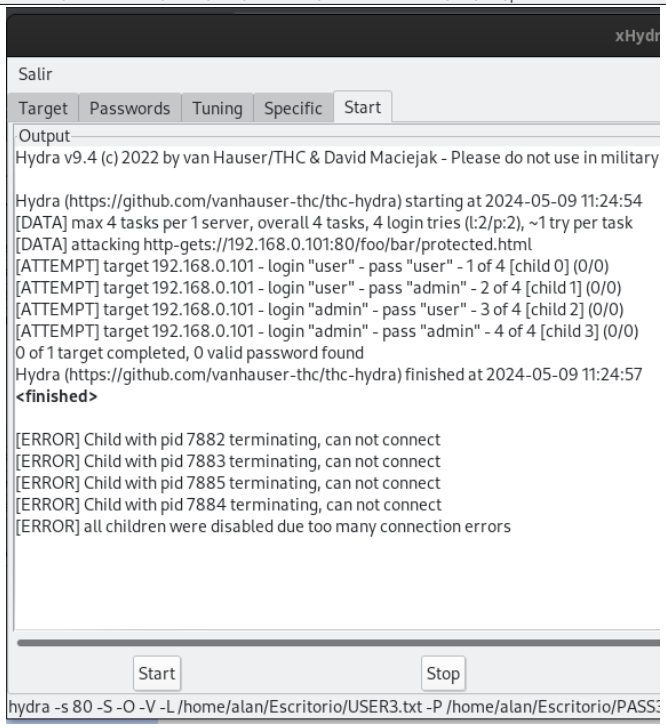
Como en el caso de Telnet la alerta fue un éxito debido a que se configuro para detectar tráfico perteneciente al puerto 20.

4.5.3.5 Ataque a HTTP-GET

Para el ataque a telnet el objetivo sigue siento el mismo, en este caso se cambiaron el puerto y el protocolo por el 80 y HTTP-GET respectivamente, el comando cambio viéndose:

- hydra -s 80 -S -O -V -L /home/alan/Escritorio/USER2.txt -P /home/alan/Escritorio/PASS2.txt -t 2 -m /foo/bar/protected.html 192.168.0.101 http-get






```
[**] [1:1000013:1] Error de autenticación HTTP [**]
[Priority: 0]
05/09-11:24:54.593544 EC:F4:BB:87:44:C1 -> 08:00:27:F1:80:65 type:0x800 len:0x4A
192.168.0.102:35292 -> 192.168.0.101:80 TCP TTL:64 TOS:0x0 ID:65302 IpLen:20 DgmLen:60 DF
******S* Seq: 0x561DF683  Ack: 0x0  Win: 0xFAF0  TcpLen: 40
TCP Options (5) => MSS: 1460 SackOK TS: 3622675559 0 NOP WS: 7
```

La alerta de Snort fue exitosa, debido a que la alerta se configuro para detectar tráfico que se transmitiera por el puerto 80.

CONCLUSION

En el proyecto fue interesante la implementación de la herramienta Snort, además de que se desarrollaron una serie de reglas para detectar intrusiones en la red, además de que es fundamental que la instalación de Snort este bien hecha ya que es fundamental para el uso correcto del mismo. Para ello se hizo un entorno controlado para estudiar el tráfico y los tipos de ataques más comunes y reales.

Durante la investigación se obtuvo como resultado que existen múltiples herramientas para la protección y seguridad de una red, pero que también existen infinidad de herramientas, vulnerabilidades y métodos para obtener información de los sistemas y obtener beneficio de ello.

Snort es una herramienta que ayuda a mejorar la seguridad de cualquier entorno de redes, ya que al analizar el tráfico en tiempo real puede alertar al administrador sobre posibles amenazas, intentos de ataques. Pero snort es más eficaz cuando se combina junto con otras herramientas de protección como Firewalls, IPS, antivirus, etc.

Desafortunadamente no se ha podido utilizar e implementar al 100% el uso de snort, debido a que como se hizo mención al inicio, usamos un entorno controlado en el cual el tráfico solo pudo ser simulado y no fue generado por múltiples dispositivos, con lo cual no se pudo explotar a su máxima capacidad las bondades de snort, además de que no se podria analizar la existencia de los posibles falsos positivos o falsos negativos que pueden llegar a existir dentro las alertas generadas por el tráfico de la red.

Snort es un IDS bueno ya que hay soporte de una amplia comunidad, actualizaciones constantes y como pudimos ver en un caso, tiene reglas predefinidas por la comunidad que nos pueden ayudar a detectar e identificar otras múltiples amenazas o riesgos dentro de la red.

Referencias bibliográficas

[1] Seguridad y privacidad en las redes móviles. Gsma.com. https://www.gsma.com/latinamerica/wp-content/uploads/2018/04/Seguridadyprivacidad.pdf

[2] Samonas, S., & Coss, D.L. (2014). The CIA Strikes Back: Redefining Confidentiality, Integrity and Availability in Security.

[3] Seguridad. (2023, agosto 30). Cisco. https://www.cisco.com/site/mx/es/products/security/index.html

[4] ENISA [Agencia de Ciberseguridad de la Unión Europea]. (2022). Panorama de amenazas. ENISA Agencia de Ciberseguridad de la Unión Europea. https://www.enisa.europa.eu/topics/cyber-threats/threats-and-trends

[5] Incibe.es. https://www.incibe.es/sites/default/files/docs/guia-ciberataques/osi-guia-ciberataques.pdf

[6] Monje, G., & Alexander, R. SEGURIDAD INFORMÁTICA Y EL MALWARE – Universidad Piloto de Colombia. http://repository.unipiloto.edu.co/bitstream/handle/20.500.12277/2641/00004128.pdf?sequence=1

[7] *¿Qué es un virus informático?* Norton.com. https://mx.norton.com/blog/malware/what-is-a-computer-virus

[8] Guaña Moya, J., Sánchez Zumba, A., Chérrez Vintimilla, P., Chulde Obando, L., Jaramillo Flores, P., & Pillajo Rea, C. Ataques informáticos más comunes en el mundo digitalizado(2022). https://search.proquest.com/openview/9db6b9a61f9a3f127ab3f13f39566c26/1?pq-origsite=gscholar&cbl=196259

[9] C. Alcaraz, J. Rodríguez, R. Román, & J. E. Rubio. (2017). Estado y Evolución de la detección de intrusiones en los Sistemas Industriales. https://www.nics.uma.es/pub/papers/1653.pdf

[10] BARTOLOMÉ, M.C. (2021) *Redes sociales, desinformación, cibersoberanía y vigilancia digital: una visión desde la ciberseguridad*. Available at: https://dialnet.unirioja.es/descarga/articulo/8306043.pdf

[11] Ureña Centeno, F. J. (2015, January 16). Ciberataques, La Mayor Amenaza Actual. https://dialnet.unirioja.es/servlet/articulo?codigo=7684551

[12] GUÍA DE CIBERSEGURIDAD. Intedya. Consultoría, Auditoría y Formación. (n.d.). https://www.intedya.com/internacional/3449/noticia-publicada-la-guia-de-ciberseguridad-intedya-para-empresas-como-herramienta-de-concienciacion-para-las-personas.html

[13] Cando Segovia, M. R., & Medina Chicaiza, R. P. (2021). Prevención en ciberseguridad enfocada a los procesos de infraestructura tecnológica. https://dialnet.unirioja.es/descarga/articulo/7888164.pdf

[14] Amador, Arboleda, & Bedón. (n.d.). Utilizando Inteligencia Artificial para la detección de Escaneos de Puertos . https://acis.org.co/portal/sites/all/themes/argo/assets/img/Pagina/ArticuloIAPortScan_VIJNSI.pdf

[15] Kumar, Sunil. (2018). Hacking Attacks, Methods, Techniques And Their Protection Measures. International Journal of Advance Research in Computer Science and Management. 4. 2353-2358.

[16] López, V. (2013). Papel de la Explosión Combinacional en Ataques de Fuerza Bruta. *Investigación E Innovación En Ingenierías*, *1*(1). https://doi.org/10.17081/invinno.1.1.2069

[17] ENISA [EUROPAN UNION AGENCY FOR CYBERSECURITY]. (2023). PANORAMA DE AMENAZAS DE ENISA: SECTOR SANITARIO [INTERNET]. ENISA, 2019/881. https://doi.org/10.2824/163953

[18] Flores Quispe, C. A. (n.d.). Tipos de hackers. Studocu. https://www.studocu.com/pe/document/universidad-tecnologica-del-peru/ingenieria-de-sistemas/06-tipos-de-hackers-articulo-autor-carlos-alberto-flores-quispe/64312729

[19] Scarfone, K. A., & Mell, P. M. (2007). Guide to intrusion detection and prevention systems (IDPS). National Institute of Standards and Technology.

[20] Lorenzo Fonseca, I., Maciá Pérez, F., Lau Fernández, R., Mora Gimeno, F. J., & Martínez Abarca, J. A. G. (2022, January 14). Método para la detección de Intrusos Mediante redes neuronales basado en la Reducción de Características. Academia.edu. https://www.academia.edu/68161440/M%C3%A9todo_para_la_detecci%C3%B3n_de_intrusos_mediante_redes_neuronales_basado_en_la_reducci%C3%B3n_de_caracter%C3%ADsticas

[21] TCP flags - keycdn support. KeyCDN. https://www.keycdn.com/support/tcp-flags

[22] Katz, M. (2013). Redes y seguridad. Alpha Editorial.

[23] Acosta, C. E., Castiñeira, F. G., & Montenegro, E. C. (1970, January 1). Red Inalámbrica de Sensores con topología lineal sin capa de red. Dialnet. https://dialnet.unirioja.es/servlet/articulo?codigo=7741838

[24] Cloudfire. ¿Qué es mtu (Unidad de Transmisión Máxima)? https://www.cloudflare.com/es-es/learning/network-layer/what-is-mtu/

[25] Álvaro M. (2019, febrero 18). FRAGMENTACIÓN IPv4. Networkgeeks. https://netwgeeks.com/fragmentacion-ipv4/

[26] Villalon Huerta, A. (2002, Julio). SEGURIDAD EN UNIX Y REDES. https://www.rediris.es/cert/doc/unixsec/unixsec.pdf

[27] Scarfone, K., & Mell, P. (n.d.). Intrusion Detection Systems (IDS). https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-30r1.pdf

[28] Whitman, M., & Mattord, H. (2009). Principles of Information Security. https://almuhammadi.com/sultan/sec_books/Whitman.pdf

[29] Rojas, J. A., y Manta, H. C. (2011). SISTEMAS DETECTORES DE INTRUSOS Y ANALISIS DE FUNCIONAMIENTO DEL PROYECTO DE CODIGO ABIERTO SNORT. Redes de Ingeniería. https://doi.org/10.14483/2248762X.7187

[30] Intrusion Detection System Consortium. “Intrusion Detection System buyer’s guide” Technical report, ICSA.NET, 1999.

[31] “Sistemas de Detección de intrusos y Snort”, Maestros del Web. http://www.maestrosdelweb.com/snort/

[32] Network Intrusion Detection & Prevention System. Snort. https://www.snort.org/

[33] Snort. preprocesadores (i) parte. Seguridad y Redes. https://seguridadyredes.wordpress.com/2009/03/03/snortpreprocesadores-i-parte/

[34] Gabriela, V., Teresa, G., Christian, M., & Geovanni , Q. N. (2019, February). How to obtain keys in WLAN / WPS networks using Wifislax and Denial of Services with Kali Linux. https://www.proquest.com/openview/f4b193b46ccb16a2fa7f1e3c633e8dd1/1?pq-origsite=gscholar&cbl=1006393

[35] Penetration Testing with Kali Linux. Anarcho-copy.org. https://edu.anarcho-copy.org/Against%20Security%20-%20Self%20Security/Offensive%20Security%20-%20Pentesting%20with%20Kali%20(PWK).pdf

[36] lyon, gordon. (2013, July 25). Nmap network scanning guide - gordon lyon. UserManual.wiki. https://usermanual.wiki/Document/Nmap20Network20Scanning20Guide2020Gordon20Lyon.369448098/help

[37] Combs, G. About Wireshark. Wireshark. https://www.wireshark.org/about.html

[38] Esteban. (2012, November 24). Seguridad Informática [ethical hacking, Pen-Test, anti-script-kiddies]. Seguridad Informática [Ethical Hacking ,Pen-test, Anti-Script-kiddies].
https://antisec-security.blogspot.com/2012/11/burp-suite-professional-burp-suite-es.html

[39] Paspuel, M. (2018). Hack de Redes Wireless con Aircrack-ng. NEXOS CIENTÍFICOS - ISSN 2773-7489, 2(2), 16–20.
https://nexoscientificos.vidanueva.edu.ec/index.php/ojs/article/view/20

[40] Aircrack-ng. es: newbie_guide [Aircrack-ng].
https://www.aircrack-ng.org/doku.php?id=es%3Anewbie_guide

[41] Pritchett, W. L., & De Smet, D. (2013). Kali Linux Cookbook. Packt Publishing.

[42] Benito, F. G. (2014). Laboratorio de Seguridad Informática con Kali Linux.
https://uvadoc.uva.es/bitstream/handle/10324/5141/PFC-B.14.pdf?sequence=1

[43] SNORT User's Manual 2.9.16.
http://manual-snort-org.s3-website-us-east-1.amazonaws.com/

IBM. (2021). ¿Qué es el ransomware?
https://www.ibm.com/mx-es/topics/ransomware#:~:text=El%20ransomware%20es%20un%20tipo,pague%20un%20rescate%20al%20atacante.

General rule options - snort 3 rule writing guide. (s/f). Snort.org. Recuperado el 24 de mayo de 2024, de https://docs.snort.org/rules/options/general/

Anexo

Composición de reglas de Snort

Las acciones de las reglas le dicen a Snort cómo manejar los paquetes coincidentes. Hay cinco acciones básicas:

- alert-> generar una alerta en el paquete actual
- block-> bloquear el paquete actual y todos los paquetes posteriores en este flujo
- drop -> descartar el paquete actual
- log -> registrar el paquete actual
- pass -> marcar el paquete actual como pasado

También existen las que se conocen como “respuestas activas” que realizan alguna acción en respuesta al paquete que se está detectando:

- react -> enviar respuesta al cliente y finalizar la sesión.
- reject -> finalizar sesión con restablecimiento de TCP o ICMP inalcanzable
- rewrite -> permite sobrescribir el contenido del paquete basándose en una opción de "reemplazar" en las reglas

La acción deseada para una regla determinada es lo primero que se declara en una regla.

msg	msg establece el mensaje que se imprimirá cuando una regla coincida
referencia	reference se utiliza para proporcionar un contexto adicional a las reglas en forma de enlaces a sistemas de identificación de ataques relevantes
gid	gid identifica el componente Snort específico que genera un evento dado

sid	sid identifica el número de firma único asignado a una regla de Snort dada
rev	rev identifica el número de revisión particular de una regla de Snort dada
classtype	classtype asigna una clasificación a la regla para indicar el tipo de ataque asociado con un evento
prioridad	priority establece un nivel de gravedad para priorizar eventos apropiados
metadatos	metadata agrega información adicional y arbitraria a una regla en forma de pares nombre-valor
servicio	service establece la lista de servicios que se asociarán con una regla dada
rem	rem se utiliza para transmitir un comentario arbitrario en el cuerpo de reglas
archivo_meta	file_meta se utiliza para establecer los metadatos del archivo para una regla de identificación de archivo dada

Instalar SSH en Ubuntu

Con el comando “install” de Ubuntu para el servicio SSH, ahora puedes instalar OpenSSH en la herramienta de línea de comandos iniciada. El comando es el siguiente:

- sudo apt install openssh-server

Después de finalizar el proceso de instalación, puedes comprobar con el siguiente comando si el Daemon SSH ya funciona de la manera prevista:

- sudo systemctl status ssh

Si el servidor SSH sigue inactivo y al reiniciarlo no está activado el inicio automático, puedes cambiarlo introduciendo otros dos comandos:

- sudo systemctl enable ssh
- sudo systemctl start ssh

Para que puedas conectarte a tu sistema Ubuntu desde cualquier lugar a través de SSH, debes liberar también el puerto del protocolo de red (por defecto: puerto TCP 22).

- sudo ufw allow ssh

El archivo de configuración central del paquete de Ubuntu SSH se llama sshd_config. Para cualquier tipo de cambio, abre este archivo con el editor de texto que desees de la siguiente manera:

- sudo gedit /etc/ssh/sshd_config

Ajusta el contenido del Config a tus necesidades y guarda los cambios antes de cerrar el archivo.

Instalar telnet en Ubuntu

Por defecto, el paquete del servidor Telnet está disponible en el repositorio por defecto de Ubuntu 20.04. Puedes instalarlo simplemente ejecutando el siguiente comando:

- sudo apt install telnetd -y

Una vez completada la instalación, puedes comprobar el estado del servicio Telnet con el siguiente comando:

- sudo systemctl status inetd

Instalar servidor ftp en Ubuntu

Instalar Vsftpd

- sudo apt-get install vsftpd

Una vez completada la instalación, haz una copia de seguridad del archivo original para que podamos comenzar nuestro trabajo con un archivo de configuración en blanco:

- sudo cp /etc/vsftpd.conf /etc/vsftpd.conf.original

Permitir el tráfico FTP desde el firewall

- sudo apt-get install ufw
- sudo ufw enable

Si ya está activo, todavía tienes que asegurarte de que el tráfico FTP está permitido. Para ello, ejecuta los siguientes comandos uno a uno:

- sudo ufw allow 20/tcp
- sudo ufw allow 21/tcp
- sudo ufw allow 990/tcp
- sudo ufw allow 40000:50000/tcp

Crear el directorio de usuarios

- sudo adduser admin

Usa el siguiente comando para crear la carpeta FTP:

- sudo mkdir /home/admin/ftp

Establece la propiedad usando:

- sudo chown nobody:nogroup /home/admin/ftp

Finalmente, elimina los permisos de escritura:

- sudo chmod a-w /home/admin/ftp

Ahora, usa el siguiente comando para verificar los permisos:

- sudo ls -la /home/admin/ftp

Como paso siguiente, crearemos el directorio contenedor de archivos y asignaremos la propiedad:

- sudo mkdir /home/admin/ftp/files
- sudo chown admin:admin /home/admin/ftp/files

Finalmente, agrega un archivo de prueba al directorio el cual se usará cuando probemos todo más adelante:

- echo "vsftpd sample file" | sudo tee /home/admin/ftp/files/sample.txt