UNIVERSIDAD NACIONAL DE SAN CRISTÓBAL DE HUAMANGA

FACULTAD DE INGENIERÍA DE MINAS, GEOLOGÍA Y CIVIL

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS

Sistemas de Detección de Intrusos Suricata Open Source como mecanismo de seguridad corporativa en entornos libres, 2020

Tesis para optar el título profesional de:
INGENIERO DE SISTEMAS

Presentado por:

Bach. Danny Victor TINEO MORALES

Asesor:

Mg. Ing. Hubner JANAMPA PATILLA

Ayacucho - Perú

2020

DEDICATORIA

A nuestro creador Dios que siempre nos cuida y más aún en estos tiempos difíciles de pandemia.

A mis padres, por inculcarme el deseo de superación e incondicional apoyo, pero, sobre todo, por el amor incondicional que me brindan en todo momento.

AGRADECIMIENTOS

En este caso quiero dar mis agradecimientos a mi querida Universidad Nacional de San Cristóbal de Huamanga, y a su estimada plana de docentes de Ingeniería de Sistemas, quienes con sentido crítico y vocación supieron guiarme a lo largo de mi formación profesional.

A mi asesor, el Mg. Ing. Hubner Jananpa Patilla, por el tiempo dedicado y su valiosa guía en la elaboración de este proyecto.

También a mis amigos que me apoyaron con su granito de arena en la elaboración de este proyecto.

CONTENIDO

CAPÍTULO III

METODOLOGÍA DE LA INVESTIGACIÓN

CAPÍTULO IV

IMPLEMENTACIÓN DEL IDS EN LA EMPRESA

CAPITULO V

CONCLUSIONES Y RECOMENDACIONES

RESUMEN

Hoy en día en la actualidad en el Perú los ataques informáticos se han incrementado ya que como la tecnología avanza día a día también van surgiendo nuevas modalidades de ataques en especial a empresas e instituciones que los afecta económicamente. Por ello se han determinado diversas maneras de seguridad informática que protección de datos entre ellos tenemos los sistemas de detección de intrusiones (IDS) que se adecua más a protección empresarial, en la actualidad estos IDS detectan los paquetes de red, monitorea el tráfico de red entrante y saliente e identifica el uso no autorizado mal manejo de las redes informáticas, como son muy sofisticados , detectan ataques difíciles de detectar por el firewall, Fortinet, antivirus u otras soluciones de seguridad informática. Pero pocas empresas e instituciones lo han implementado por varios motivos uno de ellos el costo y la falta de asesoría en seguridad informática.

El objetivo de la investigación fue implementar Suricata Open Source como un software de detección de intrusos en la seguridad en la infraestructura de una red en entornos libres aplicado a empresas e instituciones del Perú mediante técnicas, reglas de seguridad informática. Para alcanzar los objetivos de la investigación, con la ayuda de una máquina virtual se implementó un escenario donde se utilizaron varias herramientas para llevar a cabo los experimentos de investigación e implementación, para luego validar el funcionamiento de Suricata Open Source.

Se implementará un Suricata Open Source, basado en palabras clave de protocolo, perfilado de reglas y mediante algoritmos de comparación de patrones. Se utilizará el Sistema Operativo Ubuntu y Kali Linux y varias herramientas de seguridad informática.

A partir de Suricata Open Source implementado las empresas podrán obtener primeramente la seguridad informática mejorada y luego con el IDS podrán detectar, monitorear ataques informáticos en tiempo real para reportar y dar la solución respectiva.

Palabra clave: Suricata Open Source, Sistema de Detección de intrusiones, Seguridad corporativa, Ataques informáticos.

INTRODUCCIÓN

Suricata es una herramienta gratuita y de monitoreo de seguridad de red de código abierto (NSM) que puede como un sistema de detección y prevención de intrusiones de red (IDS/IPS). Puede capturar y procesar el tráfico en vivo o procesar las capturas de paquetes (PCAP) sin conexión. Suricata admite un lenguaje de reglas extenso que es sintácticamente como el lenguaje de reglas de Snort. Eso también puede registrar metainformación sobre los paquetes y sus diversos protocolos en múltiples formatos de registro, JSON incluido. Es una herramienta para una buena solución de detección que cubre el aspecto de análisis de red del malware y la detección de amenazas (Mohanta y Saldanha, 2020, p. 47).

La cantidad de tráfico malicioso está aumentando y cada día se crean nuevas amenazas. Las amenazas se vuelven cada vez más serias, complejas y sofisticadas. Ya no hay adolescentes jugando, creando virus y gusanos que son el mayor problema para las empresas y organizaciones. Las amenazas internas y los ciberdelincuentes organizados que buscan información confidencial, como el número de seguro social y las cuentas bancarias, son algunos de los mayores problemas en la actualidad. Este tipo de amenazas está empeorando y las empresas necesitan herramientas para evitar que su sistema se vea comprometido (Tafto,2011, p. 6).

Mi motivación para la implementación del Suricata Open Source, se debe a que las empresas corporativas no cuentan con una seguridad informática adecuada y segura frente a posibles ataques informáticos que en su mayoría hoy en día se han vuelvo indetectables, por ello este IDS es una alternativa de solución sofisticado y tiene como función principal proteger la información (confidencialidad, disponibilidad e integridad) de las empresas corporativas.

Los objetivos específicos son: a) Construir el Sistema de Detección de Intrusos Suricata Open Source basada en **palabras clave de protocolo** como mecanismo de Seguridad Corporativa en entornos libres, 2020.b) Construir el Sistema de Detección de Intrusos Suricata Open Source a nivel de **perfilado de reglas** como mecanismo de Seguridad Corporativa en entornos libres, 2020.c) Construir el Sistema de Detección de Intrusos Suricata Open Source mediante **algoritmos de comparación de patrones** como mecanismo de Seguridad Corporativa en entornos libres, 2020.

CAPÍTULO I
PLANTEAMIENTO DEL PROBLEMA

1.1. DIAGNÓSTICO Y ENUNCIADO DEL PROBLEMA

Según el Diario El Peruano (2020), menciona que el estado peruano resulta ser el segundo país que tiene menos protección en ciberseguridad en Sudamérica, posterior a Brasil. Asimismo, a nivel mundial Perú ocupa el puesto 17.

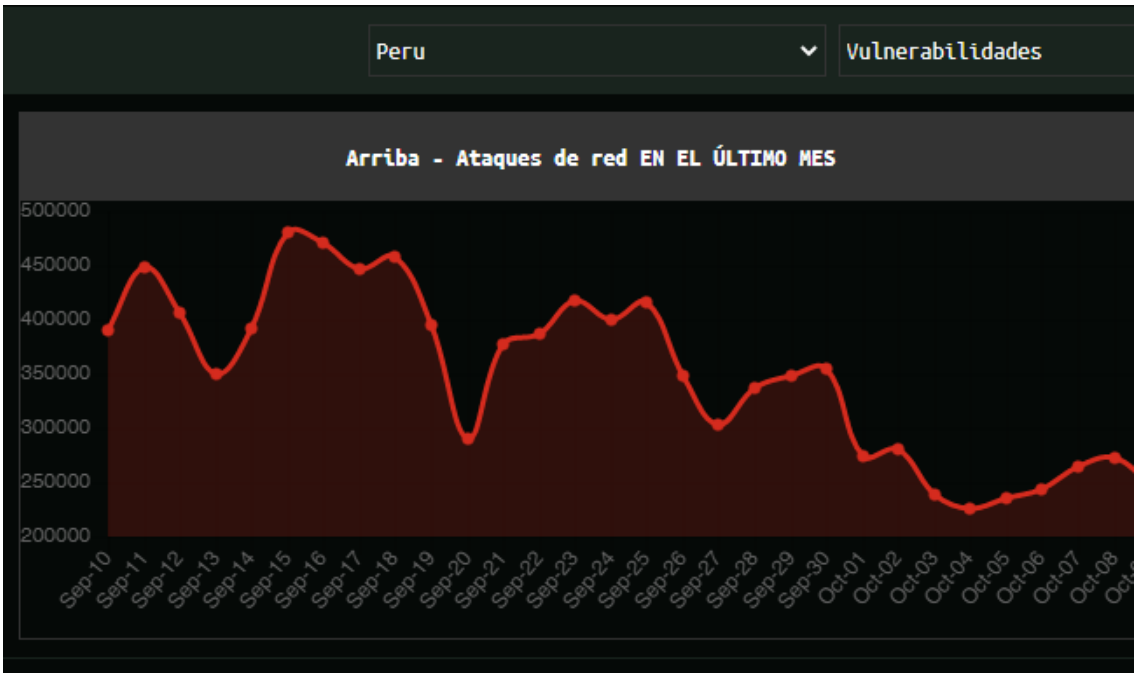
Según Bardales (2014), en una nota publicada en el diario La Gestión, menciona que hay crímenes informáticos que no solo dañan a enormes empresas, sino que también estos crímenes dañan a pequeñas y medianas empresas (Pymes) que en su mayoría digamos de cinco, una de ellas recibe estos ataques.

Para Carpentier (2016), nos dice que las amenazas convierten a una empresa vulnerable son en su mayoría las amenazas ya sean internas o externas, así como también malwares y escaneos de red que filtran información al hacker. En la actualidad la información navega por fronteras fuera de la empresa por ello es importante establecer estrategias de protección.

Para Farro Flores (2019), nos dice que las amenazas de informática en pequeñas y medianas empresas son más constantes, esto se debe a que son diminutas empresas que recién están en desarrollo, y en su mayoría no cuentan con personal de ciberseguridad. Debido a la falta de capacitación por parte de los dueños a los empleados de la empresa sobre el cuidado de la información de las empresas y también, por la falta de implementación de medidas de control para la seguridad informática.

Figura 1

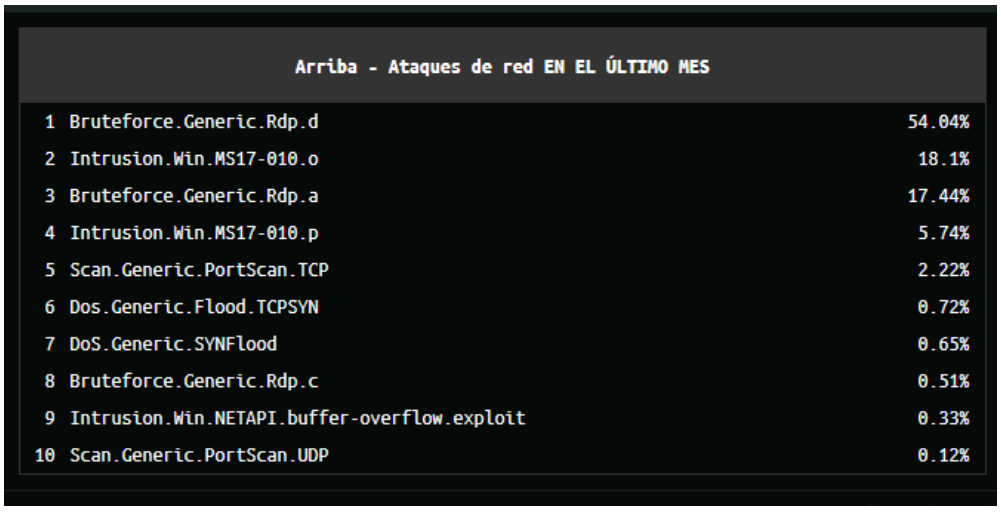
Estadística de los ciberataques informáticos a redes en Perú en el 2020.



Nota. El gráfico indica la cantidad de ataques de red registrados en el Perú entre el 10 de setiembre del 2020 al 9 de octubre del 2020. De donde podemos indicar que se registró mayor incidencia el 15 de setiembre del 2020, llegando a registrar 480025 de ataques en solo ese día. Tomado de (https://cybermap.kaspersky.com/, 2020).

Figura 2

Estadística de los tipos de ataques con mayor frecuencia a redes en Perú en el 2020



Nota. La figura representa los ataques con mayor frecuencia durante el período de 10 setiembre a 9 octubre del 2020 las cuales son de tipo Bruteforce.Generic. Rdp con un 54.04%. Tomado de (https://cybermap.kaspersky.com/, 2020).

1.2. DEFINICIÓN DEL PROBLEMA DE INVESTIGACIÓN

PROBLEMA GENERAL

¿Cómo el Sistema de Detección de Intrusos **Suricata Open Source** es un mecanismo de Seguridad Corporativa en entornos libres, 2020?

PROBLEMAS ESPECÍFICOS

a) ¿Cómo el Sistema de Detección de Intrusos **Suricata Open Source** basada en **palabras clave de protocolo** es un mecanismo de Seguridad Corporativa en entornos libres, 2020?

b) ¿Cómo el Sistema de Detección de Intrusos **Suricata Open Source** a nivel de **perfilado de reglas** es un mecanismo de Seguridad Corporativa en entornos libres, 2020?

c) ¿Cómo el Sistema de Detección de Intrusos Suricata Open Source mediante **algoritmos de comparación de patrones** es un mecanismo de Seguridad Corporativa en entornos libres, 2020?

1.3. OBJETIVO GENERAL

Implementar el **Sistema de Detección de Intrusos Suricata Open Source** como mecanismo de Seguridad Corporativa en entornos libres, 2020.

1.4. OBJETIVOS ESPECÍFICOS

a. Construir el Sistema de Detección de Intrusos Suricata Open Source basada en **palabras clave de protocolo** como mecanismo de Seguridad Corporativa en entornos libres, 2020.

b. Construir el Sistema de Detección de Intrusos Suricata Open Source a nivel de **perfilado de reglas** como mecanismo de Seguridad Corporativa en entornos libres, 2020.

c. Construir el Sistema de Detección de Intrusos Suricata Open Source mediante **algoritmos de comparación de patrones** como mecanismo de Seguridad Corporativa en entornos libres, 2020.

1.5. JUSTIFICACIÓN

Actualmente las empresas corporativas por falta de asesoría y costos no establecen nuevos mecanismos de seguridad en sus empresas, ya que solo se basan en su mayoría en firewall, Fortinet y antivirus pero estos equipos al no estar bien configurados son una puerta de acceso a la información por parte de los hackers. Es por ello que no implementan un sistema de detección de intrusos a nivel de red (NIDS). El presente trabajo pretende brindar un mecanismo de seguridad preventivo adecuado para las empresas que brinde protección a la información que es el activo más valioso de ellas.

1.6. DELIMITACIÓN

La investigación se realizó sobre el sistema de detención de intrusiones Suricata Open Source, abarcando palabras clave de protocolo, métodos de captura, perfilado de reglas y algoritmos de comparación de patrones.

CAPÍTULO II
REVISIÓN DE LA LITERATURA

2.1. ANTECEDENTES DE LA INVESTIGACIÓN

Según Noguera (2019), en su tesis denominada "Implementación de un Sistema de Detección de Intrusos para Venezolana del Vidrio C.A", concluye que; teniendo esta herramienta suricata que es para la detección y alarma de alguna circulación de virus maligno, se aumenta la integridad, disponibilidad y custodia de nuestra información de la empresa, así nos permite dar una mirada más globalizada y así vigilar los "logs de eventos" y sostener medidas significativas. También otorga favores por ejemplo la disminución de precio, porque no es casi forzosa la compra de enmiendas mercantiles, y esto generaría un gran gasto en la compra de equipos de seguridad.

Para Albin (2011), en su tesis sobre "Análisis comparativo de sistemas de detección de intrusiones de Snort y Suricata ", menciona que tanto Suricata como Snort son softwares muy capaces, y tienen sus puntos fuertes y débiles. Suricata tiene la ventaja de que puede crecer para adaptarse a un mayor tráfico de red sin requerir varias instancias. Snort es ligero y rápido, pero tiene una capacidad limitada para escalar más allá de los 200-300 Mbps por instancia. Si bien la sobrecarga de procesamiento de Snort es menor que la de Suricata, la necesidad de múltiples instancias para lograr lo que Suricata puede lograr con su diseño de subprocesos múltiples eleva el costo de operar y administrar un entorno Snort.

Según Yuquilena (2016), en su tesis sobre "Estudio de técnicas y herramientas para la prevención y detección de intrusiones a nivel de aplicación en la red de datos de la UNACH", Universidad Nacional de Chimborazo, menciona que la arquitectura multihilos de Suricata aprovecha las prestaciones hardware de los dispositivos físicos actuales con multiprocesadores y múltiples núcleos, con ello, es posible una mayor velocidad de análisis

Astudillo, Jiménez y Ortiz (2011), en su tesis de investigación "Adaptación del IDS/IPS Suricata para que se pueda convertir en una solución empresarial", menciona que estos son instrumentos esenciales para poder adquirir una defensa completa ante cualquier daño a la infraestructura de los negocios. Ya que es valiosa la información y por el sin número de

amenazas que existen para aprovecharse de la debilidad de varios sistemas, por ello es necesaria su protección con medidas adecuadas.

2.2. MARCO TEÓRICO

2.2.1. MODELO DE REFERENCIA OSI

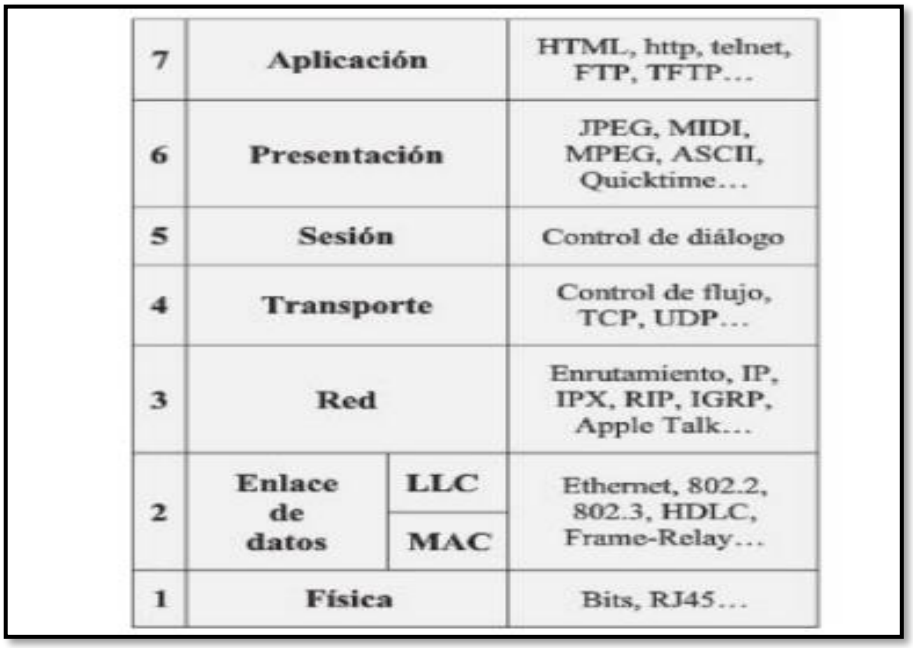
Este modelo de referencia OSI es un modelo conceptual, también conocido como el modelo de siete capas, que fue establecido por la Organización Internacional de Normalización (ISO) y la Unión Internacional de Telecomunicaciones-Sector de Normalización de las Telecomunicaciones (UIT-T), para desarrollar la comunalidad en función e interfaz entre protocolos de comunicación. Es importante tener en cuenta que el modelo OSI no es una regla establecida, sino simplemente una guía de referencia para los proveedores para que sus productos puedan interactuar entre sí. El propósito del modelo es permitir que las redes multiproveedoras interoperen de forma independiente y sólo requieren conocimiento de las interfaces entre capa (Carthern, Wilson, Rivera y Bedwell,2015).

Según Mohammed (2014), menciona que el modelo OSI consta de siete capas. Por lo general, los routers y otros dispositivos de red actúan en las tres capas inferiores y los anfitriones actúan en las siete capas. Cada capa controla los datos de una manera que es diferente de otras capas.

Según Ariganello (2019), menciona que el modelo de referencia, fracciona la red en varias capas con el único fin de que los programadores se desenvuelvan en su área sin la necesidad de la dependencia de otras. En este caso el programador realiza una aplicación sin la necesidad de ver cuáles serán los medios para el transporte de datos, pero un técnico se encargará de dar información y comunicación sin interesarle los datos que circulen.

Figura 3

Las siete capas de representación del modelo de referencia OSI.



7	Aplicación		HTML, http, telnet, FTP, TFTP…
6	Presentación		JPEG, MIDI, MPEG, ASCII, Quicktime…
5	Sesión		Control de diálogo
4	Transporte		Control de flujo, TCP, UDP…
3	Red		Enrutamiento, IP, IPX, RIP, IGRP, Apple Talk…
2	Enlace de datos	LLC	Ethernet, 802.2, 802.3, HDLC, Frame-Relay…
		MAC	
1	Física		Bits, RJ45…

Nota. El grafico representa los sietes capas del modelo de referencia, sus puertos de transmisión y funciones. Tomado de *Guía de estudio para el certificado CCNA* (p.23), por E. Ariganello,2011, Redes Cisco.

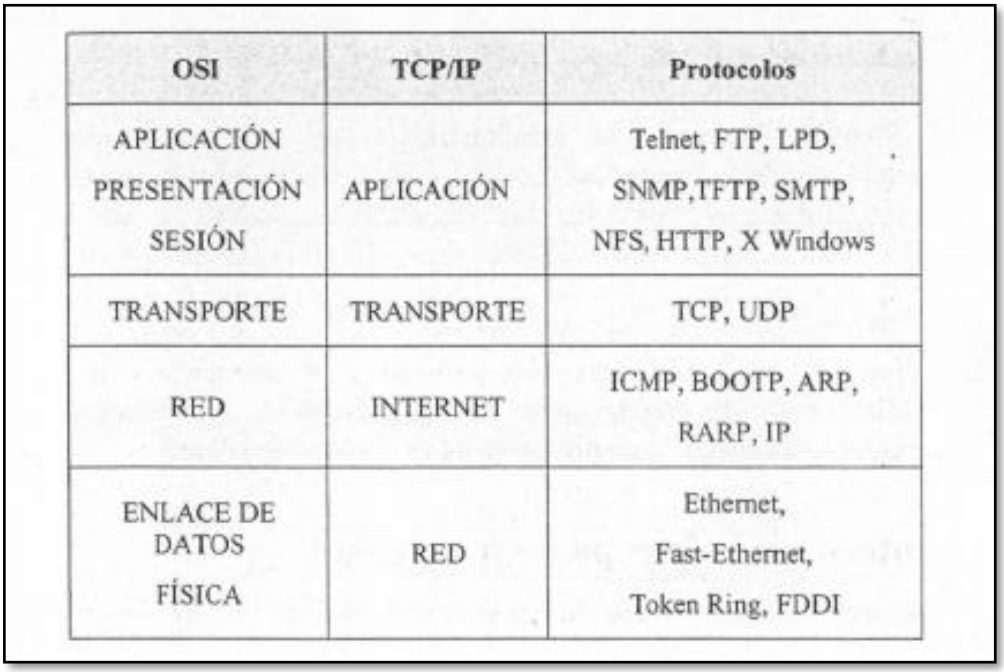
2.2.2. MODELO TCP/IP

Según Mohammed (2014), menciona que el modelo TCP/IP se creó sobre la base de un determinado conjunto de protocolos. A diferencia del Modelo OSI que se creó como un modelo en capas primero con funciones definidas claras. Por lo tanto, una mejor comprensión del TCP/IP será en una mejor comprensión de cómo funcionan los protocolos de su conjunto de protocolos.

Según Ariganello (2019), menciona que el modelo TCP/IP ha llegado a convertirse en un estándar que usa el internet. Este modelo TCP/IP presenta estas capas: primero la capa de transporte, segundo la capa de aplicación, tercero la capa de acceso a la red y por último la capa de internet. En este caso es necesario no tergiversar las funciones de cada modelo ya que cada uno desempeña diferentes formas en cada capa de sus modelos, aunque sean parecidas, pero con una función distinta.

Figura 4

Las cuatro capas de representación del modelo TCP/IP.



OSI	TCP/IP	Protocolos
APLICACIÓN PRESENTACIÓN SESIÓN	APLICACIÓN	Telnet, FTP, LPD, SNMP,TFTP, SMTP, NFS, HTTP, X Windows
TRANSPORTE	TRANSPORTE	TCP, UDP
RED	INTERNET	ICMP, BOOTP, ARP, RARP, IP
ENLACE DE DATOS FÍSICA	RED	Ethernet, Fast-Ethernet, Token Ring, FDDI

Nota. El gráfico representa las cuatro capas del modelo TCP/IP, la comparación con el modelo OSI, protocolos y funciones. Tomado de *Guía de estudio para el certificado CCNA* (p.55), por E. Ariganello,2011, Redes Cisco.

2.2.2.1. PROTOCOLOS DE LA CAPA DE APLICACIÓN

En este caso los programas se intercomunican con esta capa de aplicación TCP/IP. Muchos de sus protocolos se pueden utilizar en esta capa, dependiendo en el programa que se está utilizando. Esta capa también define las especificaciones de la interfaz de usuario. En esta capa se utilizan varios protocolos, entre los que se puede utilizar el Protocolo de transferencia de archivos (FTP) para las transferencias de archivos, Protocolo simple de transporte de correo (SMTP) para datos de correo electrónico y Protocolo de transferencia de hipertexto (HTTP) para el sitio web Tráfico (Mohammed ,2014, p. 34).

Según Ariganello (2011), menciona que estos protocolos representan a este conjunto de normas y acordados que en este caso mandan la manera en que los dispositivos pueden intercambiar informaciones.

2.2.2.2. PROTOCOLOS DE LA CAPA DE TRANSPORTE

En este caso la capa de transporte en el modelo TCP/IP tiene propósitos similares a los del Modelo OSI. Está diseñado para dar a la fuente y el destino la capacidad de tener una conversación end-to-end. En este modelo, vemos que hay protocolos definidos que pueden operar en esta capa; TCP y UDP. Estos dos protocolos proporcionan comunicaciones orientadas a la conexión y sin conexión. Sólo el protocolo TCP proporciona una forma de secuenciación, de tal manera que incluso si los segmentos de datos llegan en una secuencia diferente de la cual fueron enviados, pueden ser reorganizados. Dado que el protocolo IP no es fiable, el uso de TCP proporciona la fiabilidad necesaria para asegurar que los datos lleguen sanos y salvos. Con el uso de UDP la situación es diferente. A veces puede que una sobrecarga causada por la confiabilidad proporcionada por el protocolo TCP compromete la calidad de algunas comunicaciones críticas en el tiempo como la voz sobre IP(VoIP) y el tráfico de videoconferencia. Esto lleva a la conclusión de que no hay como un "mejor protocolo de transporte". Cada uno de los dos protocolos se utiliza como protocolo de transporte para un amplio alcance de aplicaciones y condiciones de Irremplazable (Mohammed,2014, p. 28).

Por otro lado, la capa de transporte es idéntica y paralela a realizar las mismas funciones, en este caso a la capa de transporte en el modelo OSI. Se pueden utilizar dos protocolos en esta capa: TCP y UDP. El primero está orientado a la conexión y este último no tiene conexión, lo que significa que el TCP proporciona fiabilidad y entrega libre de errores. TCP se utiliza para correos electrónicos y datos del sitio web, mientras que UDP se utiliza generalmente para enviar datos de control, incluyendo voz y otra transmisión de datos donde la velocidad es más importante que retransmitir paquetes que se pierden (Carthern, Wilson, Rivera y Bedwell,2015).

Para Gutiérrez (2019), Transmision Control Protocol también conocido como TCP, o el protocolo orientado a conexiones. Esté protocolo está diseñado para mantener una conexión estable y segura, y para esto, lleva a cabo algunas verificaciones por medio de paquetes, esto es llamado el “Three-way handshake”, debido a que se realizan tres verificaciones antes de establecer la conexión y mandar datos. Estás verificaciones se realizan a través de paquetes, y estos se llaman SYN, ACK/SYN, ACK.

a) **SYN:** El cliente manda un paquete para establecer comunicación con el servidor.

b) **ACK/SYN:** El servidor responde con este paquete para reconocer que ha recibido la solicitud de comunicación.
c) **ACK:** El cliente regresa el paquete para finalmente comenzar una comunicación.

El mismo autor menciona que este protocolo es muy importante para casi todo lo que hacemos en la red, ya que confirma que cada paquete que se mando ha sido recibido adecuadamente.

Según Mandl (2018), menciona que el protocolo de transporte TCP es la base de los mecanismos de comunicación de mayor calidad, cuando la orientación de la conexión y transporte fiable. TCP o las instancias TCP implementadas se encargan de la tarea de establecer una conexión y proteger la transmisión de datos.

Para Cox y Gerg (2004), mencionan que TCP está guiado a la conexión diseñado para proporcionar la conexión estable para el intercambio de datos entre dos sistemas. TCP garantiza que todos los paquetes se secuencian y reconocen correctamente, y que se establece una conversación antes de enviar los datos. Esto asegura que ambas máquinas estén listas para tener una conversación y que la información que se mueve de un sistema a otro lo haga sin perder nada. Algunas aplicaciones que utilizan TCP como método de comunicación son:

a) Protocolo de terminal virtual (puerto TELNET, 23).
b) La transferencia de datos (puertos 20 y 21).
c) La transferencia de email (puerto 25 de SMTP).
d) Secure Shell (puertos SSH ,23).

Según Gutiérrez (2019), menciona que en este protocolo de comunicación User Datagram Protocol (UDP), los paquetes UDP están enfocados a velocidad, simplemente se mandan, sin confirmar si han recibido, y es que este protocolo de comunicación se utiliza para aplicaciones que requieren mucha velocidad y que la integridad de cada uno de los paquetes es menos relevante como, por ejemplo, Skype, Google meet, Zoom.

Para Cox y Gerg (2004), el protocolo UDP proporciona un sistema no confiable y sin conexión para entregar paquetes. En lugar de proporcionar mecanismos para garantizar la entrega y la secuencia, UDP permite que las aplicaciones de nivel superior se preocupen por

los datos perdidos o fuera de secuencia. Este protocolo permite que se envíen mensajes (llamados datagramas con UDP) sin la sobrecarga involucrada y el establecimiento de un enlace de comunicaciones. UDP se utiliza principalmente para comunicaciones de difusión o juegos de computadora con reconocimiento de red.

2.2.2.3. PROTOCOLOS DE CAPA DE INTERNET

Esta capa Internet TCP/IP se correlaciona con la capa de network del modelo OSI y es responsable del enrutamiento y Abordar. El protocolo más común utilizado en esta capa es el protocolo de Internet (IP). Esta capa lógicamente direcciones de paquetes con direcciones IP y enruta los paquetes a diferentes redes. La capa de Internet recibe paquetes de la capa de transporte, agrega direcciones IP de destino y origen al paquete, y reenvía esto a capa de interfaz de red para transmitir al remitente. El lógico (direccionamiento virtual), también conocido como dirección IP, permite que el paquete se enrute a su destino. En el camino, los paquetes atraviesan muchas ubicaciones antes de llegar a su fin (Carthern, Wilson, Rivera y Bedwell,2015, p. 13).

Por ellos tenemos estos protocolos que se usan más dentro de la capa de internet del modelo TCP/IP:

A. IP

Fue diseñado para funcionar en sistemas interconectados de redes de comunicación informática conmutadas por paquetes. El deber principal de este protocolo es entregar paquetes de un equipo en una network para otro equipo que se encuentra en la misma red o en otra diferente. Esto se logra añadiendo una encabezado que contiene información de direccionamiento y control. Este encabezado contiene una dirección de origen y una dirección de destino que se definen como direcciones IP. La IP de 32 bits en un formato llamado decimal punteado (por ejemplo 192.168.0.1) El protocolo IP proporciona segmentación y reensamblaje de paquetes largos en paquetes más pequeños. Esto se vuelve muy útil cuando los paquetes pasan a través de las redes que tienen diferentes reglas de longitud máxima de paquetes en el camino al destino Red. El protocolo IP trata cada paquete como una entidad independiente no relacionada con otro paquete. No hay conexiones o circuitos lógicos virtuales o de otro tipo (Mohammed,2014, p .23).

Para Gutiérrez (2019), Internet Protocol o IP, en este contexto, es una "etiqueta" numérica que se le asigna a todo sistema que se comunica bajo el protocolo de internet para identificar en una red. Esta etiqueta, llamada IP, facilita las funciones de ruteo que permiten la transmisión de datos de un sistema a otro, es por así decir, la dirección que te permitirá mandar una carta a alguien. Actualmente existen dos versiones de IP, la IPV4, la cual es la más común actualmente, y la nueva versión, la IPV6.La versión IPV4 contiene 32 bits, y la razón por la que existen dos versiones, es que las direcciones IP, no son ilimitadas.

Para Cox y Gerg (2004), el Protocolo de Internet (IP) se usa para manejar servicios de datagramas entre hosts. Maneja el direccionamiento, el enrutamiento, la fragmentación y el reensamblaje de paquetes.

B. ICMP

Según Carthern, Wilson, Rivera y Bedwell (2015), mencionan que el ICMP es un protocolo importante en la IP que es utilizado por los dispositivos de red para enviar mensajes de error para indicar que un host o router es inalcanzable.

Según Polivio (2011), menciona que el ICMP es el encargado de dar a conocer todos los sucesos en la red e informar los mensajes de falla y de control. ICMP en este caso no se encarga de dar un veredicto al respecto, esta responsabilidad recae en las capas superiores. Y en este caso si la información se pierde o llega a si final con errores no se va crear un nuevo mensaje ICMP ya que simplemente lo descarta.

C. ARP

ARP es un protocolo utilizado para traducir direcciones lógicas de red en hardware físico de capa de enlace Direcciones. En resumen, las direcciones IP se convierten en direcciones MAC y la traducción se coloca en la Tabla ARP.Cuando un dispositivo de red recoje en este caso un paquete con una dirección de destino en una subred que posee, y esta MAC del destino no está en su tabla ARP, este dispositivo transmite un paquete de todas las interfaces a determinar quién es el propietario de esta dirección IP. El host con la dirección IP correspondiente responde con su Mac, y el Switch anota esto en su tabla ARP para una resolución más rápida en el futuro (Carthern, Wilson, Rivera y Bedwell,2015, p. 35).

2.2.2.4. PUERTOS Y FIREWALL

A. PUERTOS

Según Gutiérrez (2019), menciona que los puertos son la entrada y salida de paquetes, es en estos que se corren lo que se llaman servicios, que son aplicaciones que están haciendo uso de puerto para comunicar algo a otro sistema. Los puertos son extremadamente relevantes en el área del hacking, ya que pueden ser utilizados para obtener mucha información de un sistema, y prácticamente todos los ataques informáticos utilizan un puerto para realizar el ataque, o para comunicar información al haberse comprometido un sistema.

Figura 5

Números de puertos y protocolos de las capas de representación del modelo TCP/IP.

Número de puerto	Protocolo	Aplicación	Acrónimo
20	TCP	Protocolo de transferencia de archivos (datos)	FTP
21	TCP	Protocolo de transferencia de archivos (control)	FTP
22	TCP	Shell Seguro	SSH
23	TCP	Telnet	–
25	TCP	otocolo simple de transferencia de correo (Simple Mail Transfer Protocol)	SMTP
53	UDP, TCP	Servicio de nombres de dominios	DNS
67	UDP	Protocolo de configuración dinámica de host (servidor)	DHCP
68	UDP	Protocolo de configuración dinámica de host (cliente)	DHCP
69	UDP	Protocolo de transferencia de archivos trivial	TFTP
80	TCP	Protocolo de transferencia de hipertexto	HTTP
110	TCP	Protocolo de oficina de correos versión 3 (Post Office Protocol version 3)	POP3
143	TCP	Protocolo de acceso a mensajes de Internet (Internet Message Access Protocol)	IMAP
161	UDP	Simple Network Management Protocol	SNMP
443	TCP	Protocolo seguro de transferencia de hipertexto	HTTPS

Nota. La figura representa las características de TCP y UDP, sus números de puertos, protocolos y funciones que cumple la capa de aplicación. Tomado de K. Linares, *Descripción general de TCP y UDP* https://kevin-linares.blogspot.com/2017/05/capa-de-transporte-Protocolos-de-la-capa-de-transporte-Descripcion-general-de-TCP-y-UDP.html,, 2017, CISCO CNA-V6.0

B. FIREWALL

En este caso un cortafuegos es un sistema de protección de seguridad que esencialmente actúa como una frontera en la que se tiene reglas, esta frontera monitorea todos los paquetes, y solo deja entrar los que tengan ciertas características, por ejemplo, lo más común en un firewall empresarial es que no admite ningún paquete que quiera entrar a

la red desde afuera, pero si la conexión se inicia desde adentro, si se admite. Un firewall en sí puede ser configurado en una infinidad de formas, puede poner reglas como “no permitir la entrada de paquetes que contengan x información o que provengan de x dirección”, por lo que es una herramienta muy poderosa de seguridad, si se utiliza correctamente, y esa es la palabra clave, un firewall generalmente hablando es imposible “hackear”, lo que sucede, comúnmente es que los programadores que configuraron el firewall no pusieron la seguridad suficiente o no configuro bien, y por lo tanto se convierte en un punto débil dentro de la infraestructura de seguridad de una organizacion.Los firewall pueden ser instalados en routers,switches,servidores y computadoras, protegiendo de forma perimetral diferentes objetos en una red organizacional (Gutiérrez , 2019, p .50).

Según Arriols (2017), menciona que un cortafuegos (firewall) es un software, hardware o combinación de ambos que generalmente se utiliza para separar una red protegida de una publica no protegida.

Figura 6

Diseño de un firewall dentro de una estructura de red.

Nota. La figura representa el diseño de un firewall dentro de una estructura de red, la protección que nos da a nuestra red LAN ante todo el flujo de información que navega en la red WAN. Tomado de E. Arriols, *Curso completo de hacking ético – Udemy* earriols@redteaming.es, 2017, UDEMY.

Un firewall es un elemento de red que controla el recorrido de los paquetes a través de los límites de una red segura basada en una directiva de seguridad específica. Un cortafuegos de política de seguridad es una lista de reglas de filtrado ordenadas que definen las acciones realizadas en paquetes que cumplen condiciones específicas. Una regla se compone de un

conjunto de campos de filtrado también llamados campos de red como el tipo de protocolo, la dirección IP de origen, la IP de destino puerto de origen y puerto de destino, así como un campo de acción. Los campos de una regla representan los valores posibles de los campos correspondientes en tráfico de red que coincida con esta regla. Cada campo de la red podría ser un valor único o rango de valores. Las acciones de filtrado son para aceptar, lo que permite que el paquete dentro o desde la red segura, o para negar, lo que hace que el paquete sea bloqueado. El paquete es permitido o bloqueado por una regla específica si la información del encabezado del paquete coincide con todos los campos de red de esta regla. De lo contrario, se examina la siguiente regla y el proceso se repite hasta que se encuentra una regla de coincidencia o la acción de política predeterminada se realiza (Enab AL-Shaer,2018, p. 3).

2.2.2.5 INTERFAZ DE RED TCP/IP

Según la organización International Business Machines (IBM), esta capa de interfaz de network resetea a los datagramas IP de esta capa de red en paquetes donde las tecnologías de red detalladas se puedan analizar y enviar.

2.2.3. SEGURIDAD DE LA INFORMACIÓN Y SEGURIDAD INFORMATICA

Según Gutiérrez (2019), menciona que no toda la información está en un sistema informático, sin embargo, no deja de ser información, potencialmente sensible, y con potencial de ser comprometida. Dentro de la seguridad de la información se tiene que considerar todas las áreas en las que se pueda comprometer algún dato sensible.

Figura 7

Los tres ámbitos de la seguridad integral dentro de una organización.

Nota. La figura representa los tres ámbitos de la seguridad integral dentro de una organización (ámbito físico, ámbito digital y ámbito humano). Tomado de E. Arriols, *Curso completo de hacking ético – Udemy* earriols@redteaming.es, 2017, UDEMY.

A. SEGURIDAD FISICA

Esta área es en la que se puede comprometer la seguridad del punto de vista físico, es decir, cualquier cosa tangible, como una puerta o ventana que permita acceder a un intruso, o un papel en donde esté escrito algo confidencial, aunque últimamente se le pone cada vez menos atención, es muy importante tomarlo en cuenta .Algunos ejemplos de cómo mejorar la seguridad de esta área son la seguridad perimetral (cercas, bardas, puerta) y el diseño de seguridad (transparencia, luz, espacios abiertos), entre otras cosas. Cabe mencionar que en esta área se considera particularmente importante proteger la integridad de las personas, también se considera seguridad física proteger a las personas contra incendios, ataques terroristas, terremoto, etc. (Gutiérrez, 2019, p. 59).

B. SEGURIDAD SOCIAL, O INGENIERIA SOCIAL

Según Gutiérrez (2019), menciona que aquí lo importante es proteger la información que las personas saben, ya que también información sensible puede ser filtrada por ese medio. Este tipo de seguridad usualmente es el área más débil de cualquier organización, ya que la gente es fácilmente manipulable. Algunas formas de mitigar esto es con buenas políticas de seguridad y capacitación al personal.

C. SEGURIDAD LÓGICA

Según Gutiérrez (2019), menciona que la seguridad lógica es todo lo que está dentro de un sistema, y es de las formas más comunes de ataque hoy en día, puede ser un ataque a un servidor, una base de datos, computadora, o incluso celular. Algunas de las formas en las que se puede mitigar este tipo de ataques es con soluciones de seguridad como firewall, IDS, IPS y antivirus.

Figura 8

La triada de la seguridad (confidencialidad, integridad y disponibilidad).

Nota. La figura representa el triángulo de los ejes principales de la seguridad de la información (confidencialidad, integridad y disponibilidad) y como se interrelacionan. Tomado de *Hacker's WhiteBook* (p. 63), por P. Gutierrez, 2019.WhiteSuit Hacking.

2.2.4. SISTEMA DE DETECCIÓN DE INTRUSIONES (IDS)

Según Noguera (2019), nos menciona que es un sistema que se encarga de la administración del tránsito del network para que pueda hallar sucesos anormales o dudosas. También el descubrimiento de la actividad anormal y con ello dar a conocer al encargado de la network es la obligación primordial, aunque, varias herramientas de IDS actúan en base a las normas estructuradas cuando se hallan actividades malignas, uno de ellas, restringiendo tráficos anómalos.

Para Santos y Gregg (2019), los IDS se pueden dividir en dos categorías amplias: IDS basados en red (NIDS) e IDS basados en host (HIDS). Ambos tipos de sistemas se pueden configurar para monitorear ataques, rastrear los movimientos de un pirata informático o

alertar a un administrador sobre ataques en curso. La mayoría de los IDS constan de más de una aplicación o dispositivo de hardware. Además, un IDS debe estar capacitado para buscar actividad sospechosa.

Figura 9

Matriz de representación de IDS verdadero/falso.

	VERDADEDO	FALSO
POSITIVO	Verdadero-Positivo Se generó una alarma, y una condición presente debería ser una alarma.	Falso-Positivo Se generó una alarma, y no se presentó ninguna condición para generarla.
	VERDADEDO	**FALSO**
NEGATIVO	Verdadero-Negativo no se generó una alarma, y no hay ninguna condición presente que deba alarmarse.	Falso-Negativo No se generó una alarma, y se presentó una condición que debería alarmarse.

Nota. La figura representa a la comparación entre el positivo y negativo dentro de una Matriz de IDS Verdadero/falso. Tomado de *Certified Ethical Hacker Versión 10 Cert Guide, 3rd Edition.* Por Santos y Gregg, 2019.O´REILLY.

Según Alsmadi, Karabatis y AlEroud (2017), menciona que los sistemas de detección de intrusiones (IDS) realizan análisis exhaustivos del tráfico de red para hacer la detección inteligente de posibles ataques de red. Sus métodos para detectar los ataques de red o el tráfico dañino pueden variar de métodos simples que pueden sacar información directa relacionada con el flujo (por ejemplo, número de puerto, IP, dirección MAC). También pueden realizar métodos inteligentes complejos para llevar a cabo la firma o el patrón de análisis de algunos tipos de ataques complejos

Según Leacock (2018), menciona que los softwares para detectar intrusos (IDS) surgieron primeros, y la función de ellos es de supervisar y hallar procesos o eventos dudosos que pueden ser en el equipo o en la red de manera inmediata, con el tiempo dichos sistemas han evolucionado a sistemas que preveen intrusos (IPS), y estos acogen un punto de vista de precaución y ágil respuesta ante posibles eventos malignos que puedan ocurrir, también de que pueden ser más complejos los análisis que se realicen.

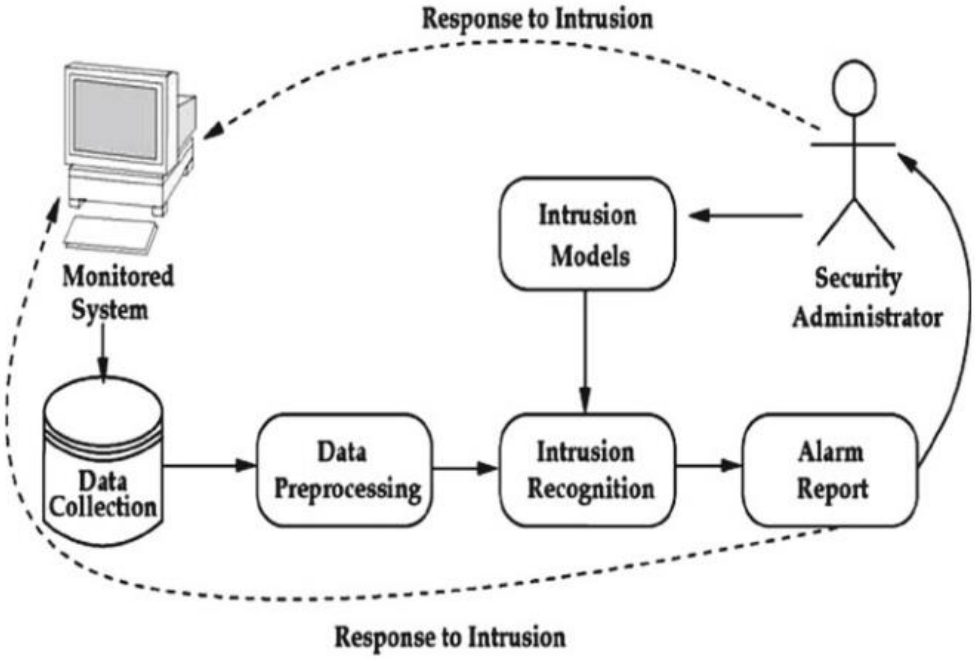
Según Sengupta y Sil (2020), el aspecto clave del desarrollo de IDS se basa en la hipótesis de explotar las susceptibilidades basadas en el uso inusual del sistema. Todos los IDS admiten esta teoría, de una forma u otra. Los sistemas pueden utilizar enfoques estadísticos y máquinas de enfoques de aprendizaje, incluidas las redes neuronales, técnicas de inteligencia de enjambre, algoritmos y programación genéticos.

También las mismas autoras mencionan que un IDS realiza las siguientes tareas:

a) recopilación de datos
b) procesamiento previo de datos
c) reconocimiento de intrusiones
d) realización de medidas correctivas

Figura 10

Componentes del software para la detección de intrusiones.



Nota. Esta figura representa a componentes del sistema de intrusiones y el proceso y secuencia que se realiza por parte de Security Administrador y el Monitoreo del sistema se interrelacionan. Tomado de *Intrusion Detection -A Data Mining Approach* (p. 4), por N. Sengupta y J. Sil, 2020. Cognitive Intelligence and Robotics.

2.2.5. ATAQUES AL SISTEMA DE DETECCIÓN DE INTRUSIONES (IDS)

Para Chakrabarti, Chakraborty y Mukhopadhyay (2010), mencionan que hay muchos tipos diferentes de ataques que pueden dañar un sistema. Por lo general, estos ataques se pueden agrupar en las siguientes categorías:

a) **Confidencialidad:** permite al atacante obtener acceso a la información sin autorización.
b) **Integridad:** permite que un atacante no autorizado afecte el estado del sistema. Esto podría significar afectar el estado del sistema o cualquier dato que resida o pase a través del sistema.
c) **Disponibilidad:** se viola el principio de disponibilidad si el atacante puede evitar que un usuario autorizado acceda a un recurso del sistema.
d) **Control:** el ataque otorga a un atacante no autorizado un privilegio en cuanto a la infracción de política de control de acceso hacia el sistema, el ataque está donde el

principio de control. Este ataque puede proporcionar medios para futuros ataques a la confidencialidad, integridad y / o disponibilidad.

Según Sengupta y Sil (2020), menciona que según el Conjunto de datos NSL-KDD. Estos ataques se clasifican como: Usuario a root, remoto a local, denegación de servicio y sonda.

a) **Denegación de servicio (DoS):** en este ataque, los atacantes han intentado interceptar a los usuarios originales para que hagan uso de cualquier tipo de servicio.
b) **Remoto a local (R2L):** el objetivo de los atacantes es obtener acceso a la máquina sin una cuenta.
c) **Usuario a raíz (U2R):** los atacantes intentan actuar como superusuario accediendo a la computadora localmente.
d) **Sondeo:** los atacantes deben tener la libertad de obtener datos importantes sobre la host de destino.

Las mismas autoras mencionan que la mayoría de los ataques pertenecen a la categoría de denegación de servicio (DoS). Algo de más tipos de ataques se encuentran en el sistema informático, como escuchas, ataques de espionaje, interceptación y denegación de servicio distribuida (DDoS), por nombrar unos pocos:

1. **Escuchar a escondidas**: escuchar en secreto las conversaciones privadas de otros sin su consentimiento.
2. **Snooping:** el Snooping se usa para monitorear la actividad de una computadora o dispositivo de red de forma remota.
3. **Intercepción:** es un tipo de ataque de hombre en el medio que intercepta el mensajes transmitidos entre dos dispositivos y alterarlos.
4. **Denegación de servicio distribuida:** intento malintencionado de interrumpir el tráfico normal de un servidor o red. El servidor de destino y sus alrededores están siendo inundados por Internet tráfico. Como resultado, los usuarios normales no pueden acceder al objetivo afectado y sus alrededores.

2.2.5.1. ATAQUES DE ESCANEO

Los ataques de escaneo se pueden utilizar para asimilar información sobre el sistema atacado. Usando técnicas de escaneo, el atacante puede obtener información de

topología, tipos de tráfico de red permitidos a través de un firewall, hosts activos en una red, SO y kernel de hosts en una red, software de servidor en ejecución, números de versión del software, etc. En formación, el atacante puede lanzar ataques dirigidos a exploits más específicos. Lo anterior se recopiló mediante el lanzamiento de un escaneo SYN sigiloso. Este escaneo se llama sigilo porque en realidad nunca completa las conexiones TCP. Esta técnica a menudo se denomina medio análisis abierto, porque el atacante no abre una conexión TCP completa. El atacante envía un paquete SYN, como si estuviera abriendo una conexión TCP real. Si el atacante recibe un SYN/ACK, esto indica que el puerto está escuchando. Si no se recibe respuesta, el atacante puede asumir que el puerto está no abierto. Este es solo un tipo de técnica de escaneo, hay muchas más disponibles (Chakrabarti, Chakraborty y Mukhopadhyay, 2010, p. 44).

2.2.5.2. ATAQUES DE DENEGACIÓN DE SERVICIO

Según Ramiro (2018), menciona en este caso ante un atentado de denegación de servicio, un delincuente cibernético va intentar camuflar la legalidad de una información cuando los usuarios accedan. Uno de los ataques más comunes es cuanto el delincuente inunda de forma ilegal la información en la red. Si accedemos a una página estamos conectándonos con el servidor web de un dominio y solicitamos acceso para visualizar dicha página. Pero el servidor solo procesa una cantidad adecuada y responde a esas solicitudes y si el delincuente sobrecarga el servidor de solitudes no va a poder responder las solicitudes y habrá una caída del servidor y de esta manera denegar un servicio.

2.2.5.3. ATAQUES DE PENETRACIÓN

Los ataques de penetración contienen todos los ataques que le dan al atacante no autorizado la capacidad de obtener acceso a los recursos, privilegios o datos del sistema. Una forma común de que esto suceda es aprovechando una falla de software. Por ejemplo, en julio de 2002 se encontró un exploit en el código de manejo de respuesta de desafío SSH (Secure Shell) que permitía al atacante ejecutar código arbitrario como el usuario que ejecuta SSH (a menudo root). Este ataque se consideraría un ataque de penetración. Ser capaz de ejecutar código arbitrariamente como root le da al atacante cualquier recurso imaginable del sistema. Además, esto podría permitir al usuario lanzar otros tipos de ataque en este sistema, o incluso atacar otros sistemas desde el sistema comprometido (Chakrabarti, Chakraborty y Mukhopadhyay, 2010, p. 44).

2.2.6. METODOS DE ANALISIS DE SISTEMA DE DETECCIÓN DE INTRUSIONES (IDS)

2.2.6.1. IDS BASADO EN FIRMA

Según Ortiz (2019), menciona que los IDS basados en firmas hace referencia cuando existen ataques y hacen prevenciones a ellos mediante la búsqueda de patrones detallados, que pueden ser malwares. Esta terminología se origina dentro del software antivirus, que se refiere a estos patrones detectados como firmas. En IDS basados en firmas, las firmas son lanzadas por un proveedor para todos sus productos. La actualización a tiempo de los IDS con la firma es un aspecto clave. (Ortiz).

Esta metodología se utiliza para detectar ataques desconocidos que ya están predefinidos en forma de firma y se guardan. Cuando se envían datos a la red, primero van al servidor donde el servidor lo analiza en busca de contenido malicioso. Se compara el paquete de red de una database donde la firma ya está contenida en la red y si algún paquete coincide, entonces descarta el paquete o si no hay coincidencia, envía el paquete a la red (Sharma, Kumar y Tasneem, 2018).

2.2.6.2. IDS BASADO PROTOCOLO

El NIDS (sistema de intrusión basado en red), identifica los elementos del protocolo y los analiza mientras busca una infracción. Algunos sistemas de detección de intrusos examinan campos de protocolo explícitos dentro de los paquetes inspeccionados. Otros requieren técnicas más sofisticadas, como el examen de la longitud de un campo dentro del protocolo o el número de argumentos. Por ejemplo, en SMTP, el dispositivo puede examinar comandos y campos específicos como HELO, MAIL, RCPT, DATA, RSET, NOOP y QUIT. Esta técnica reduce la posibilidad de encontrar falsos positivos si el protocolo que se analiza se define y aplica correctamente (Santos y Gregg, 2019).

2.2.6.3. IDS BASADO EN ANOMALIAS

Según Ortiz (2019, menciona que estos sistemas que detectan intrusiones que se basan en fallas se han introducido con la función de detectar amenazas que son inexplorados, esto basado en la evolución del virus. Por ello hay un enfoque fundamental para utilizar el aprendizaje automático con ello se crea un paradigma de actividad transparente y posteriormente se compara un nuevo proceder con dicho modelo. Ya que estos modelos se pueden preparar de acuerdo con las aplicaciones y las configuraciones de hardware, el

método basado en aprendizaje automático tiene una propiedad mejor generalizada en comparación con los IDS tradicionales basados en firmas. Aunque este enfoque permite la detección de ataques previamente desconocidos, puede sufrir falsos positivos.: la actividad legítima previamente desconocida también puede clasificarse como maliciosa. La mayoría de los IDS existentes sufren el lento proceso de detección que degrada el rendimiento de los IDS. El eficiente algoritmo de selección de características hace que el proceso de clasificación utilizado en la detección sea más confiable.

El modelo se construye considerando la conducta natural para identificar falencias por si los datos proceden a desviarse de los datos comunes. Los dos tipos de métodos para detectar anomalías son: el dinámico y el estático. Los estáticos hacen la suposición de que el patrón de los propósitos que se monitorean sigue siendo los mismos, por ejemplo, secuencias de llamadas al sistema de un servicio Apache. Por ello en el método que detecta las falencias dinámicas, hacen que los patrones se saquen según el comportamiento hábitos generales de los usuarios finales o basados en el historial de redes / hosts. Detección de anomalías los métodos son capaces de identificar intrusiones desconocidas, pero su limitación es obtener conocimiento sobre el comportamiento anormal debido a muestras insuficientes que representan comportamiento anormal durante el entrenamiento (Sengupta y Sil, 2020, p. 5).

2.2.6.4. IDS BASADO EN LA HEURISTICA

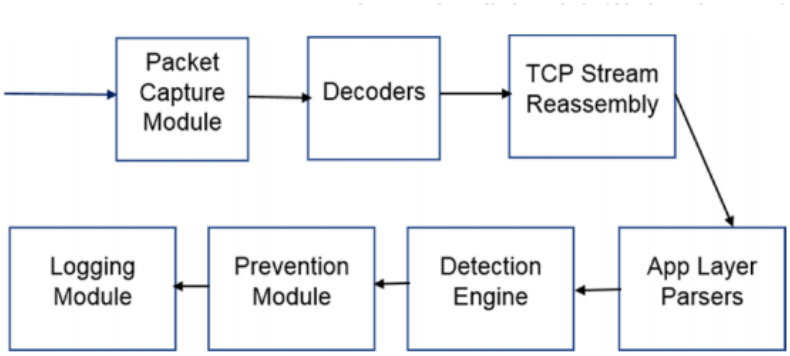
Según Santos y Gregg (2019), el escaneo heurístico utiliza la lógica algorítmica del análisis estadístico del tráfico que pasa a través de la red. Sus tareas consumen mucha CPU y recursos, por lo que es una consideración importante al planificar su implementación. Los algoritmos basados en heurística pueden requerir un ajuste fino para adaptarse al tráfico de la red y minimizar la posibilidad de falsos positivos. Por ejemplo, una firma del sistema puede generar una alarma si se escanea un rango de puertos en un host o red en particular.

2.2.7. COMPONENTES DE UN SISTEMA DE PREVENCION DE DETECCIÓN DE INTRUSIONES (IDS)

Según Mohanta y Saldanha (2020), menciona que un IDPS es una pieza compleja de software que consta de múltiples partes móviles. Más de los componentes en un IDPS son bastante modulares y tienen una tarea o funcionalidad establecida que llevan a cabo en los paquetes entrantes, después de lo cual pasan los paquetes y su salida correspondiente de su procesamiento al siguiente componente, y así sucesivamente.

Figura 11

Componentes de un IDPS y flujo de paquetes.



Nota La figura muestra los principales componentes que componen un IDPS y el flujo de paquetes a través de ellos. Tomado de "*Malware Analysis and Detection Engineering: A Comprehensive Approach to Detect and Analyze Modern Malware"* (p. 829), por A. Mohanta y A. Saldanha, 2020. APRESS®.

2.2.7.1. MODULO DE CAPTURA DE PAQUETES

Cualquier IDPS requiere paquetes de la red que puedan diseccionar y analizar. Hay muchos métodos de captura de paquetes disponibles que son utilizados e implementados por una captura de paquetes IDPS módulo, que escucha los paquetes que ingresan a las interfaces o puertos de red en el IDPS dispositivo. El objetivo principal de cualquier método de captura que utilice el módulo de captura de paquetes es capturar paquetes eliminando la menor cantidad posible y con una sobrecarga de CPU baja. La mayoría de las redes de alta velocidad pueden tener una alimentación de tráfico en el rango de millones de paquetes por segundo y decenas de gigabits de ancho de banda. El método de captura de paquetes debe poder obtener los paquetes que llegan al puerto de red o la tarjeta de interfaz de red (NIC), y con tan bajo sobrecarga posible en el kernel y el sistema y hacer que los paquetes estén disponibles para el resto de los IDPS para procesar. Para ejercer la menor sobrecarga posible, muchos marcos de captura de paquetes implementan una técnica de copia cero que utiliza sus módulos de kernel que les ayuda a copiar o acceder el paquete para el IDPS, omitiendo la mayor cantidad posible de la pila de red del kernel, lo que reduce enormemente la sobrecarga del sistema y del kernel y mantiene el consumo de CPU bajo. Algunos de los marcos de captura de paquetes comunes que IDPS implementa en su paquete (Mohanta y Saldanha 2020, p. 829).

Los mismos autores también menciona que otros métodos personalizados, muchos de los cuales son comerciales y de pago. Aparte de los marcos de paquetes, hay varios otros proveedores comerciales que proporcionar tarjetas de interfaz de red personalizadas que apuntan a redes de gran ancho de banda, promete una sobrecarga de CPU baja mientras se mantiene una baja pérdida de paquetes. Ahora, una vez que el módulo de captura de paquetes en el IDPS obtiene el paquete, podría haber para extraer tanta información sobre el paquete como sea posible de la captura de paquetes marco que utiliza. Muchas veces, el marco de captura de paquetes puede realizar algunos preprocesar en el paquete, extraer cierta metainformación sobre el paquete, antes de entregar el paquete real al IDPS. Es importante para la captura de paquetes módulo para obtener la mayor cantidad de esta metainformación sobre el paquete que recibe del marco de captura de paquetes. Parte de esta información puede ser información de VLAN, paquete marcas de tiempo, hashes de flujo, etc. Esta metainformación es utilizada por varios otros componentes más adelante en el IDPS a medida que procesan el paquete.

2.2.8. TIPOS DE SISTEMA DE DETECCIÓN DE INTRUSIONES (IDS)

2.2.8.1. SISTEMA DE DETECCIÓN DE INTRUSOS BASADO EN RED (NIDS).

En este caso en su mayoría las partes de los sistemas que detectan intrusos se basan en el network. Y estos IDS son los que van a hallar amenazas capturándolos y los paquetes de pasan por el network. Escuchando a la vez en un segmento, por ello un NIDS logra vigilar la circulación que puede aparentar a varios equipos que se interconectan a un network para su protección. Los IDS que se basan en network en varios casos están elaborados por un contiguo de sensores que están localizados en distintos puntos de una red. Dichos sensores cumplen la función de vigilar toda la circulación de red haciendo un análisis local y a la vez informa que estos ataques se realizan en la consola de gestión. Aproximadamente estos sensores son limitados para la ejecución en el software para la detección, ya que fácilmente pueden ser protegidos ante amenazas. Demasiados de estos sensores están elaborados justamente para ejecutarse de forma no visible ya que de esta manera es más imposible para un ciberdelincuente hallar su ubicación y asistencia (Noguera,2019, p. 6).

Figura 12

*Representación de las diferencias entre IPS e I*DS.

	IPS	IDS
	Inline, Bloque Automático	**Mirror, Alertas para analistas**
Estabilidad	Caída del sistema es catastrófica para la red	Caída del sistema quita información al analista de red. No es algo crítico
Desempeño	Requiere mayor capacidad de procesamiento. Puede producir cuellos de botella.	La falta de procesamiento puede ser compensada con buffers de mucha memoria. Nunca producirá cuellos de botella.
Precisión de Falsos Positivos	Produce bloqueos de paquetes. Problema con aplicaciones.	Carga trabajo innecesaria para el analista en busca de falsas alarmas.
Precisión de Falsos Negativos	Paquetes maliciosos entran a la red. No es tan crítico como en el caso de los IDS.	Ataques resultan totalmente **invisibles** y pueden volver a ocurrir. Pérdida de información para el analista.

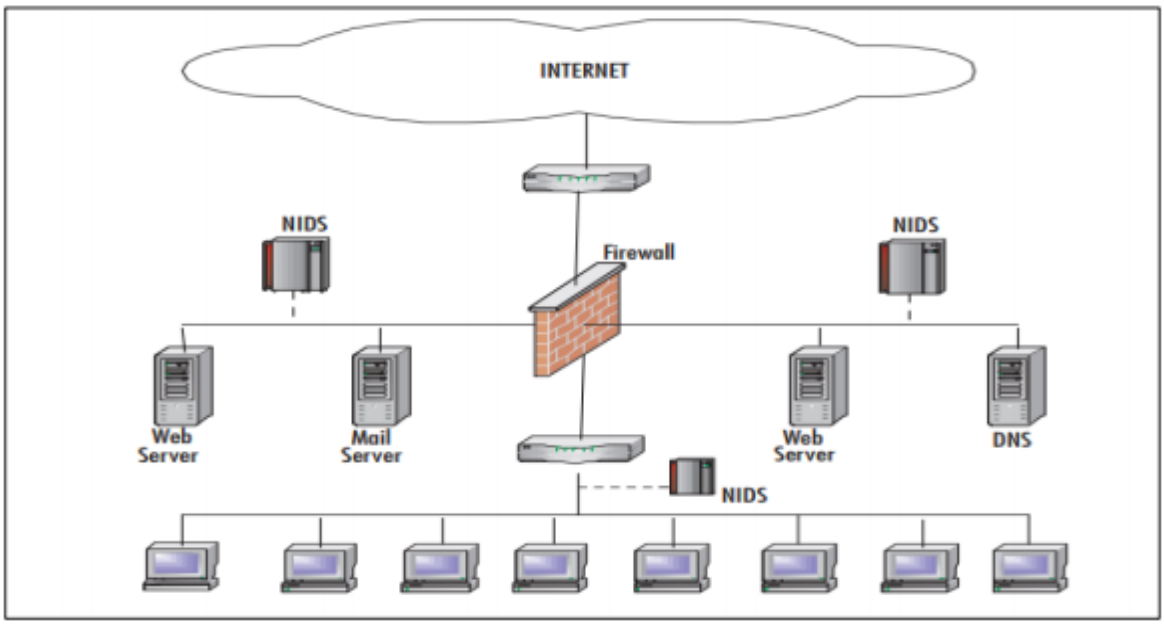
Nota. La figura representa las diferencias entre IDS e IPS entre sus funciones y consecuencias. Tomado de "*Adecuación del IDS/IPS suricata para que se pueda convertir en una solución empresarial"* (p.9), por Astudillo, Jiménez y Ortiz, 2019.

El NIDS monitorea un segmento de red completo. Normalmente, en este caso la tarjeta de la interfaz de red de computadora (NIC) funciona en modo no promiscuo. En este modo de operación, solo los paquetes destinados a la dirección de control de acceso a medios (MAC) específica de la NIC (o paquetes de difusión) son reenviado la pila para su análisis. Los NIDS deben de andar en un modo atento que pueda vigilar todo el tráfico que circula en la red que no está direccionado a la propia MAC. Además, el NIDS debe estar conectado a un puerto SPAN (Switched Port Analyzer) en su conmutador local, o una red toca duplicar el

tráfico en el enlace que desea monitorear. Operación de la NIC del NIDS en modo promiscuo es necesaria para proteger su red (Baker y Esler ,2005).

Figura 13

El Sistema para detectar intrusos que se basan en network NIDS.



Nota. La figura representa el diseño físico de una Red NIDS dentro de una organización y su ubicación adecuada. Tomado de "*Certified Ethical Hacker Versión 10 Cert Guide", 3rd Edition.* Por Santos y Gregg, 2019.O´REILLY.

2.2.8.2. SISTEMA DE DETECCIÓN DE INTRUSOS BASADO EN HOST (HIDS).

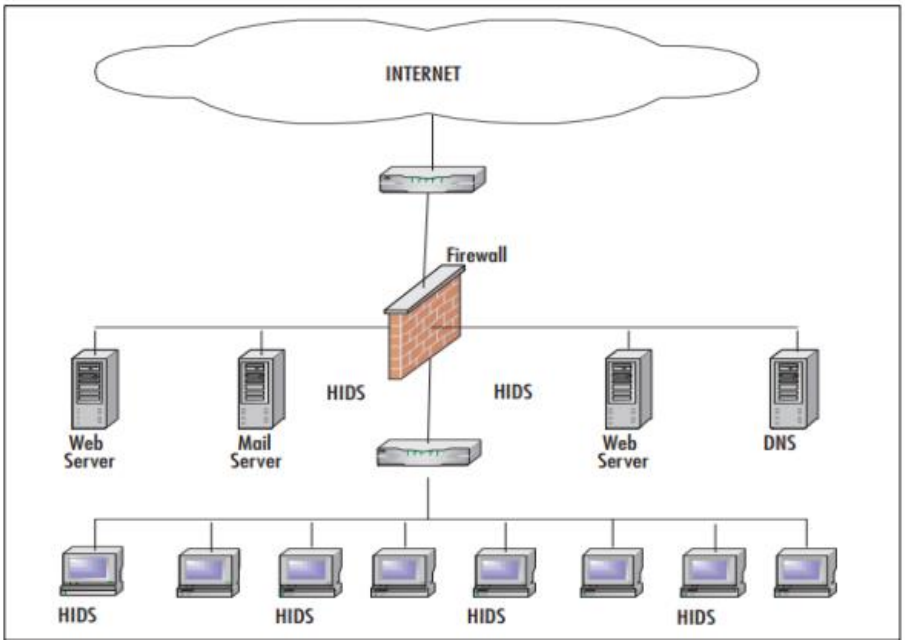
Estos HIDS llevaron desde un inicio a ser los primeros tipos de IDS que han sido elaborados y desarrollados. Ya que realizan su ejecución dentro de informaciones halladas de un equipo. Todo ello concede a que el IDS examine todos los trabajos, para de esta manera decidir con exactitud esos procesos e individuos están asociados en una amenaza individua que embarque a un sistema. (Noguera, 2019, p. 7).

Según Baker y Esler (2005), menciona que el HIDS protege solo el sistema host en el que reside y su tarjeta de red funciona de manera predeterminada en modo no confidencial. El modo de operación no promiscuo puede ser una ventaja en algunos casos, porque no todas las NIC son capaces de modo promiscuo. Además, el modo promiscuo puede ser CPU intensiva para una máquina host lenta. Debido a su ubicación en el host para ser

monitoreados, los HIDS tienen acceso a todo tipo de información local adicional con seguridad, otra ventaja de HIDS es la capacidad de adaptar el conjunto de reglas muy finamente para cada host individual.

Figura 14

Sistema de detección de intrusos basados en host HIDS.



Nota. La figura representa el diseño físico de una Red HIDS dentro de una organización y su ubicación adecuada. Tomado de *Certified Ethical Hacker Versión 10 Cert Guide, 3rd Edition.* Por Santos y Gregg, 2019.O´REILLY.

Para Bace y Mell (2015), mencionan que los IDS basados en host funcionan con información recopilada dentro de un sistema informático individual. Este punto de vista permite a los IDS basados en host que puedan ser capaces de dirigir las tareas con mucha credibilidad y exactitud, también determinan de forma precisa que avances y sujetos están interrelacionados ante una acometida individual hacia el sistema operativo. Además, estos se diferencian de los IDS basados en network, los IDS basados en host logran "ver" el producto de un intento de acometida, si esto pasa tendrían acceso directamente para vigilar los ficheros de datos y el sistema de avances que suelen ser objeto de ataques.

Para Parisi (2019), la tarea de Host Intrusion Detection Systems (HIDS) es detectar posibles intrusiones que afectan a las máquinas host dentro de una organización, especialmente las

máquinas que se consideran críticas. Con este fin, monitorea algunas de las métricas del sistema que se supone que son significativas para identificar posibles ataques.

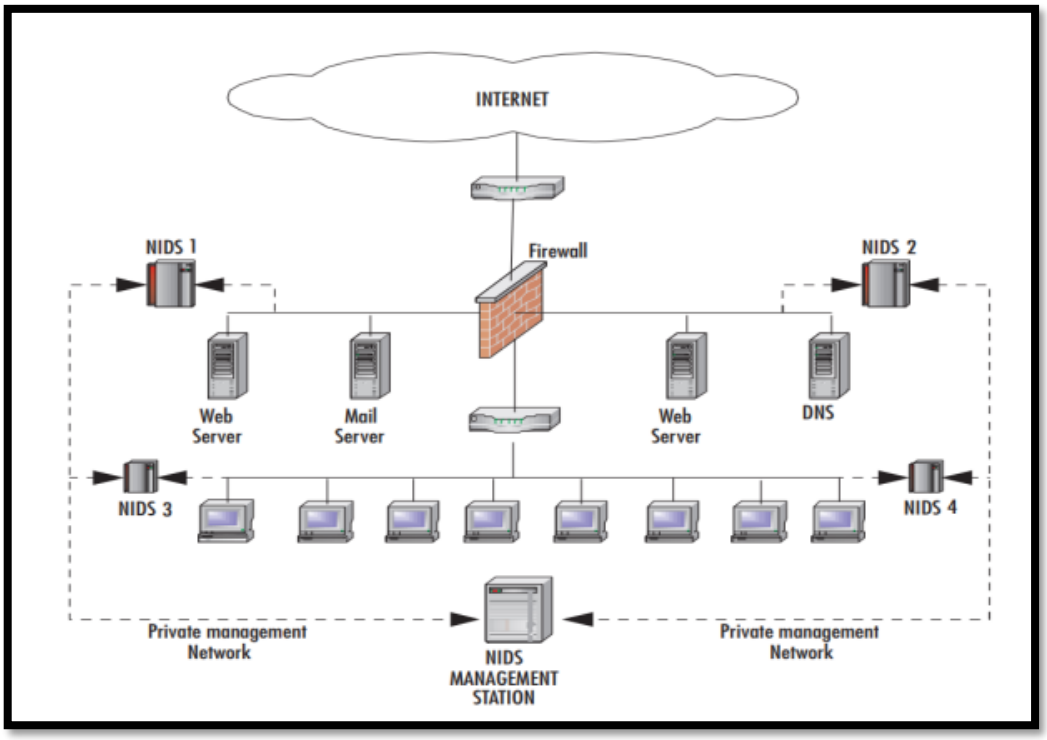
2.2.8.3. SISTEMA DE DETECCIÓN DE INTRUSOS DISTRIBUIDO (DIDS).

Según Baker y Esler (2005), menciona que el DIDS estándar funciona en una arquitectura Manager/Probe. Los sensores de detección de NIDS se ubican de forma remota e informan a una estación de administración centralizada. se cargan periódicamente a la estación de administración y se pueden almacenar dentro de una central de database; se pueden descargar firmas nuevas de ataque a los sensores según sea necesario.

La Figura 15 muestra un DIDS este mezclado por cuatro sensores y una estación para la gerencia centralizada. Este sensor NIDS 1 y NIDS 2 funcionan en cautela promiscuo y están protegiendo los domésticos públicos. Este sensor NIDS 3 y NIDS 4 protegen a los sistemas de equipo en una central de informática confiable.

Figura 15

Sistema de detección de intrusos distribuido DIDS.



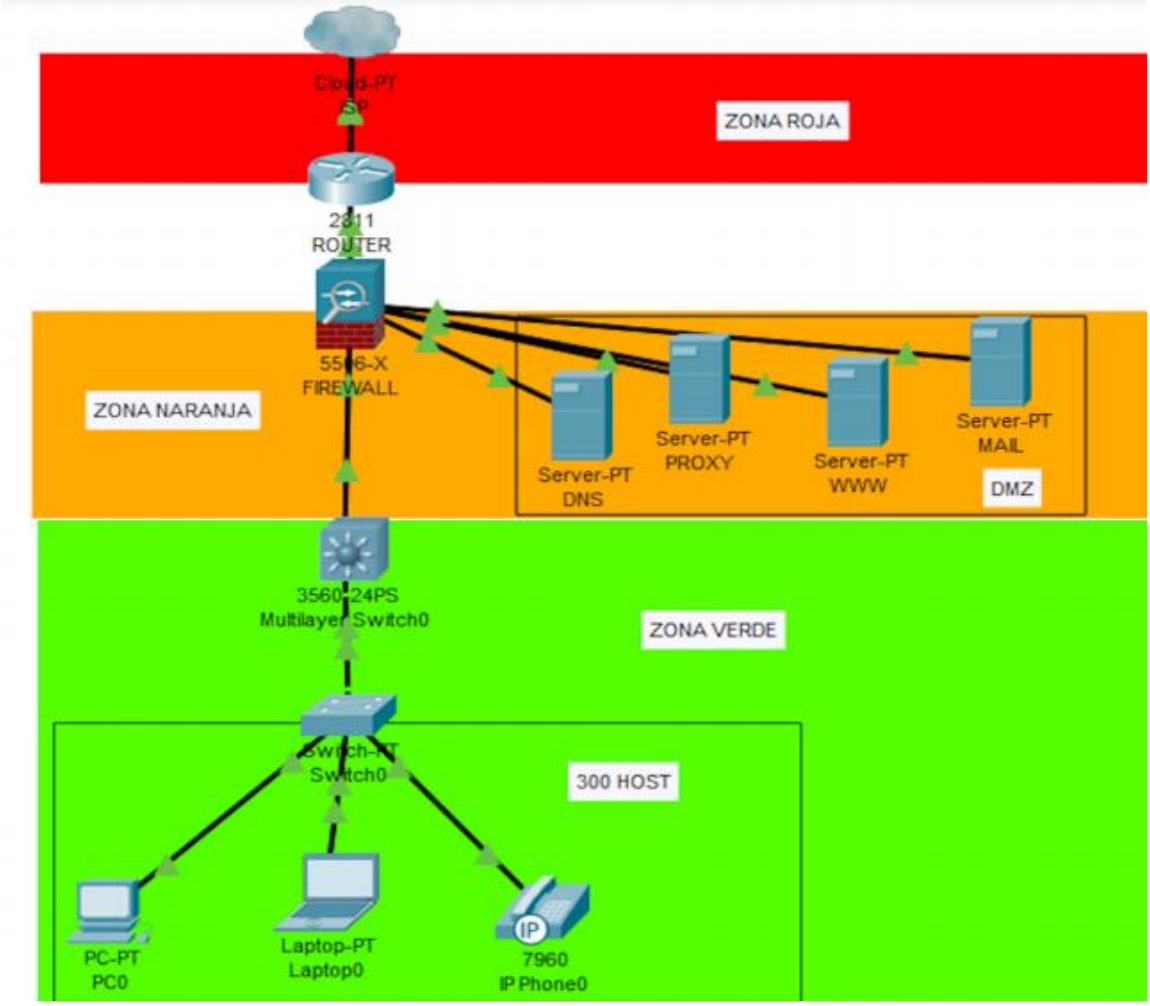
Nota. La figura representa el diseño físico de una Red NIDS dentro de una organización y su ubicación adecuada. Tomado de *Certified Ethical Hacker Versión 10 Cert Guide, 3rd Edition.* Por Santos y Gregg, 2019.O´REILLY.

2.2.8.4. LOCALIZACION DE UN IDS DENTRO DE UNA RED EMPRESARIAL

Según Noguera (2019), menciona que se tiene que hallar muchas causas para la ubicación la ubicación entre los cuales se tiene: la parte física, el sistema, y por su puesto el recurso individual. Ahora ya teniendo en cuenta todo ello se procede a determinar las zonas que son tres para colocar el IDS.

Figura 16

Ubicación de una IDS dentro de una red corporativa.



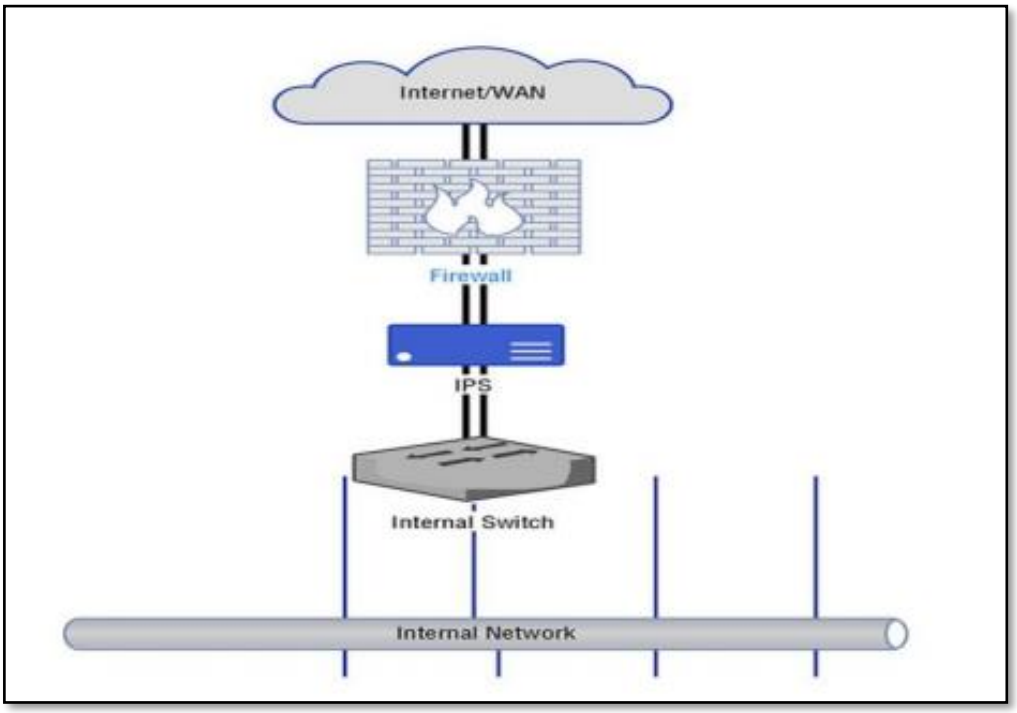
Nota. La figura representa tres zonas posibles donde colocar un IDS dentro de una red corporativa. Tomado de "*Implementación de un Sistema de Detección de Intrusos para venezolana del vidrio C.A"* (p.34), por A. Noguera, 2019.

Para Messier (2019), el lugar de colocación de un IDS de red es importante, puesto que vigila todo el transito que va a pasar por la interfaz de network. El mejor enfoque es colocar los IDS conectados en paralelo en lugar de en serie en el perímetro de la red para que se pueda

observar todo el tráfico de red que ingresa, como se muestra en la Figura 17, donde muestra el IDS detrás del firewall, pero no directamente en el flujo de tráfico. El tráfico se desvía al IDS para que pueda detectarlo sin interponerse ni causar un fallo de red.

Figura 17

Diagrama de red que muestra la ubicación del IDS.



Nota. La figura representa el diagrama de red de la ubicación de un IDS que en este caso adecuado es antes del firewall para detectar anomalías antes de la red LAN, por R. Messier, *Certified Ethical Hacker Study Guide*,2019.(https://www.oreilly.com/library/view/ceh-v10-certified/9781119533191/ftoc.xhtml.)

2.2.9. SURICATA

Esto es un instrumento gratuito y de monitoreo de la seguridad de red de código abierto (NSM) que puede como un sistema de detección y prevención de intrusiones de red (IDS/IPS). Puede capturar y procesar el tráfico en vivo o procesar las capturas de paquetes (PCAP) sin conexión. Suricata admite un lenguaje de reglas extenso que es sintácticamente como el lenguaje de reglas de Snort. Eso también puede registrar metainformación sobre los paquetes y sus diversos protocolos en múltiples formatos de registro, JSON incluido. Es una herramienta para una buena solución de detección que cubre el aspecto de análisis de red del malware y la detección de amenazas (Mohanta y Saldanha, 2020, p. 47).

“Suricata es un motor IPS/IDS de código abierto bajo licencia GPLv2, desarrollado por la comunidad de OISF (Open Infomation Security Foundation), con muy buenas características siendo la más importante su arquitectura multihilos. Los principales beneficios de un diseño de múltiples hilos es que ofrece mayor velocidad y eficiencia en el análisis de trafico de red y también puede ayudar a dividir la carga de trabajo de IDS / IPS en función de las necesidades de procesamiento. Además de la aceleración de hardware (con limitaciones de hardware y de tarjeta de red), el motor está diseñado para utilizar la mayor potencia de procesamiento ofrecida por los últimos chips de CPU multinúcleo” (Noguera,2019, p. 36).

Como otros sistemas de detección de intrusiones en la red, Suricata monitorea el tráfico de la red y crea registros de alarmas o alertas cuando que detecta el tráfico malicioso. Suricata está diseñado para ser compatible con otros componentes de seguridad y ofrece características como funcionalidad de salida unificada y es posible aceptar llamadas de otras aplicaciones a través de sus bibliotecas conectables. El motor admite y proporciona funcionalidades como la última Snort VRT, registro de Snort, opciones de lenguaje de reglas, subprocesos múltiples, aceleración de hardware, salida unificada que permite la interacción con sistemas de gestión de registros externos e IPv6. Además, admite y proporciona funciones como la reputación de IP basada en reglas, la capacidad de conexión de la biblioteca para la interacción con otras aplicaciones, la salida de estadísticas y un manual de usuario de inicio simple y eficaz (Tafto ,2011, p. 15).

2.2.9.1. CARACTERISTICAS DE SURICATA

Según Parrado (2019), menciona que también es motor de network de clave abierto y a la vez multiplataforme con rendición alta IDS, IPS y por tanto la certeza en la red, desarrollado por la comunidad OISF (Open Information Security Foundation).

Según Astudillo, Jiménez y Ortiz (2011), menciona las siguientes características:

A. **MULTIPROCESO.** Esto hace referencia a una función fuerte de este IDS suricata ya que las mejoras de sistemas libres recientes y de sus desarrolladores son unithreaded.

B. **ESTADÍSTICA DE RENDIMEINTO.**En este caso este módulo es el encargado de contabilizar los elementos de mejoras de nuevas tramas y secuencias, tiempo,

etc. y la vez recepciona información para realizar su presentación estadísticamente hacia el encargado puede ser de manera web, por mensajes, vía logs.

C. **LA DETECCIÓN DE PROTOCOLOS EN AUTOMÁTICO.** En este caso el motor de suricata posee varias palabras fundamentales que hacen referencia a los protocolos que pueden ser: TCP, UDP, ICMP, HTTP, IP, FTP,SMN Y TLS.Nos da a conocer que la detección es hacia una ocurrencia dentro de un stream de datos, pero no importa el embarcadero donde suceda.

D. **LA DESCOMPRESIÓN GZIP.**En este caso el favor en la librería HTP hace que sea factible la descomprensión de un fichero en GZIP de esta manera se examina para la detección de factores para hallar el patrón de amenaza.

E. **LA INDEPENDENCIA DE LA LIBRERÍA HTP.**En este caso la librería HTP resulta ser un plan autónomo a suricata y a la vez unido de forma efectiva al suricata. Este es usado por varias aplicaciones que pueden ser filtros y también módulos de seguridad de apache.

F. **LOS MÉTODOS DE ENTRADA ESTÁNDAR.** Aquí vemos soporte para LibPcap , IPFRing y NFQueu generales que permiten capturar el tráfico.

G. **UNIFIELD2 OUTPUT.** Este es un soporte de métodos y utilidades que pueden ser de salida estándar Unifield2. Ante este tipo de salida dual lo que se busca es reducir la carga. Ahora suricata cuando menciona a la información que sale da el trabajo a soluciones que pueden ser exteriores tal cual es el Barnyard que sujeta dos duales y los razona y deposita en base a la configuración del encargado.

H. **COINCIDENCIA RÁPIDA DE IP.** En este caso el motor de este IDS suricata usa de forma automática un llamado preprocesador que es especial para que se pueda validar de forma rápida las bases y reglas que puedan hacer coincidencia de manera única de “IP”, ejemplo de ello tenemos el “RBN” o listas de “IP” de “Emerging Threats”

I. **MÓDULO DE REGISTRO HTTP.** En este caso las mediciones de “HTTP” requieren retomar una respuesta adecuada con el formato de log y de apache para que se pueda monitorear y registrar las actividades.

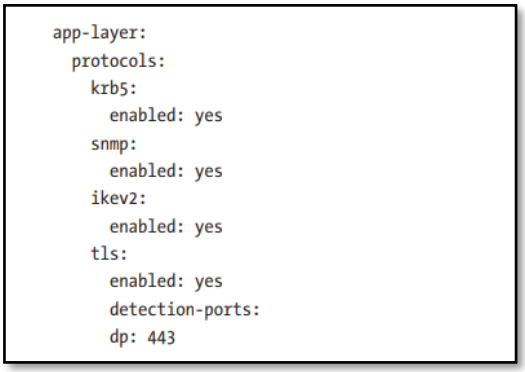
J. **REPUTACIÓN DE IP.** En este caso se comparte informaciones que tienes dirección “IP” con una inadecuada imagen ante otros orgnismos y se plantea las soluciones de una seguridad adecuada, para de esta manera se pueda eliminar a los falsos positivo. Cabe recalcar también que dicho modulo se va ha encargar de juntar, depositar, renovar, y redistribuir las enseñanzas de la imagen de una dirección “IP”.

2.2.9.2. FORMATO DE REGLAS DE SURICATA

Según Mohanta y Saldanha (2020), mencionan que Suricata, como la mayoría de los demás programas, necesita varias opciones de configuración para ejecutarse, y lo hace por medio de un archivo de configuración **yaml**, que puede pasar el Suricata en la línea de comandos. Un IDPS admite el análisis de protocolos de capa de aplicaciones, y Suricata no es diferente. El archivo de configuración de Suricata proporciona una forma de habilitar y deshabilitar selectivamente una aplicación específica.

Figura 18

Archivo de configuración de Suricata Yaml.

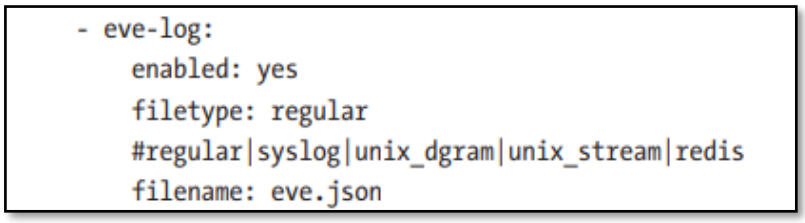


Nota. La figura representa la sección en la configuración de Suricata Yaml que le permite seleccionar Habilitar / deshabilitar analizadores de capa de aplicaciones dentro de Suricata. Tomado de “*Malware Analysis and Detection Engineering: A Comprehensive Approach to Detect and Analyze Modern Malware”* (p. 836), por A. Mohanta y A. Saldanha, 2020. APRESS®.

Los mismos autores también mencionan que Suricata también le permite registrar diversa metainformación sobre los paquetes y los campos extraídos de los datos de la capa de la aplicación de los paquetes. Suricata puede registrar esta metainformación en varios formatos y en varios mecanismos de salida. El método ampliamente utilizado por los usuarios de Suricata es **eve-log**, que genera toda la metainformación sobre los paquetes en formato **json**.

Figura 19

Sección de registro de salida Eve-Log.



Nota. La figura representa la sección de registro de salida Eve-Log. Eso muestra que Eve Logging está habilitado y genera metainformación en Json. Tomado de "*Malware Analysis and Detection Engineering: A Comprehensive Approach to Detect and Analyze Modern Malware"* (p. 836), por A. Mohanta y A. Saldanha, 2020. APRESS®.

Suricata, como la mayoría de los IDPS, admite un lenguaje de reglas enriquecido que escribe reglas. Cuando se alimenta a Suricata, estas reglas son consumidas y utilizadas por su módulo de detección para inspeccionar contra los paquetes y la metainformación extraída del paquete y su carga útil. Si alguna de las reglas coincide con los paquetes y cualquier dato asociado, Suricata toma la acción apropiada contra el empaquetado según lo definido por la acción en la regla. Ahora, la sintaxis del lenguaje de reglas utilizada por Suricata se toma prestada en gran medida del notable Snort IDS lenguaje de reglas. Pero, aunque deriva su sintaxis y la mayoría de sus palabras clave del lenguaje de reglas de Snort, la semántica del lenguaje y las palabras clave pueden variar. También las horas extraordinarias Suricata ha evolucionado con la adición de nuevas palabras clave y sintácticas actualizaciones que no están disponibles y difieren del lenguaje de reglas de Snort. Para muchos de las palabras clave y características del lenguaje de reglas presentes, las reglas escritas para Suricata deberían funcionar para Snort y viceversa, siempre que no utilice una sintaxis de palabra clave o regla que sea específico para los IDPS (Mohanta y Saldanha,2020, p. 838).

Figura 20

Estructura básica del lenguaje de reglas de Suricata.

```
ACTION PROTOCOL SRC_IP SRC_PORT DIRECTION DEST_IP DEST_PORT (keywords
semicolon and space separated…)
```

Nota. La figura representa los primeros siete campos del listado que son necesarios para cada regla de Suricata. Aparte de estos siete campos, una regla Suricata también debe contener una palabra clave llamada **sid**. Tomado de "*Malware Analysis and Detection Engineering: A Comprehensive Approach to Detect and Analyze Modern Malware"* (p. 839), por A. Mohanta y A. Saldanha, 2020. APRESS®.

Según la organización Open Information Security (OISF), menciona que las firmas juegan un papel muy importante en Suricata. En la mayoría de las ocasiones, las personas utilizan conjuntos de reglas existentes. Este documento de Reglas de Suricata explica todo acerca de las firmas; cómo leerlos, ajustarlos y crearlos.

La misma organización menciona que una regla/firma consta de lo siguiente:

a) **La acción,** que determina lo que sucede cuando la firma coincide
b) **El encabezado,** definiendo el protocolo, las direcciones IP, los puertos y la dirección de la regla.
c) **Las opciones de regla,** definiendo los detalles de la regla.

Figura 21

Formato de Reglas de Suricata.

```
drop tcp $HOME_NET any -> $EXTERNAL_NET any (msg:"ET TROJAN Probablemente Bot Nick en IRC (USA
+..)"; flow:established,to_server; flowbits:isset,is_proto_irc; content:"NICK "; pcre:"/NICK
.*USA.*[0-9] . reference:url,doc.emergingthreats.net/2008124; classtype:trojan-activity;
sid:2008124; rev:2;)
```

Nota. La figura representa la documentación para el formato de las reglas del IDS Suricata, donde están representados por colores rojo, verde y azul cada uno cumpliendo una función. Tomado por Suricata 5.0.2, rules, documentación OISF, 2019, ("https://suricata.readthedocs.io/en/suricata-5.0.2/rules/intro.html".)

A. ACCIÓN

Según Mohanta y Saldanha (2020), mención que Suricata admite siete ACCIONES que se pueden utilizar en sus reglas, y sus significados son:

Tabla 01

Reglas de acción de Suricata.

ACION	DESCRIPCION
alert	Registra una alerta para una regla si coincide.
pass	Si una regla con esta acción coincide, Suricata no alerta en el paquete de ninguna reglas que coinciden con él hasta ahora y también omite la coincidencia con cualquier otra regla cargada para ese paquete.
drop	Deja caer el paquete. Se utiliza cuando Suricata se ejecuta como IPS. También registra una alerta para la regla.
reject	Todas las acciones de rechazo son una función IPS, donde cuando una regla con esta acción coincidencias, Suricata envía un rechazo activo del paquete. Si el paquete en que la regla coincide es un paquete TCP, Suricata envía un paquete TCP RST a el remitente del paquete en el que coincidió. Para todos los demás tipos de paquetes, envía un paquete de error ICMP.
rejectsrc	Lo mismo que rechazar la acción.
rejectdst	Funciona igual que el rechazo, excepto que el paquete de rechazo activo se envía a el destino del paquete en el que coincidió la regla.
rejectboth	Funciona igual que el rechazo, excepto que el paquete de rechazo activo se envía a tanto el origen como el destino del paquete en el que coincidió la regla.

Nota. La tabla representa las diversas ACCIONES disponibles por Suricata Rule Language, y la funcionalidad que cumplen. Tomado de "*Malware Analysis and Detection Engineering:*

A Comprehensive Approach to Detect and Analyze Modern Malware" (p. 839-840), por A. Mohanta y A. Saldanha, 2020. APRESS®.

B. PROTOCOLO

Según la organización Open Information Security (OISF), menciona que esta palabra clave en una firma indica a Suricata qué protocolo se refiere. Puede elegir entre cuatro protocolos básicos:

a) TCP (para el tráfico TCP)
b) UDP
c) ICMP
d) IP

La misma organización menciona que la disponibilidad de estos protocolos depende de si el protocolo está habilitado en el archivo de configuración suricata. yaml. Si usted tiene una firma como por ejemplo un protocolo http, Suricata se asegura de que la firma pueda hacer juego solamente si se refiere al tráfico http.

Figura 22

Regla de protocolo de Suricata.

```
drop tcp $HOME_NET any -> $EXTERNAL_NET any (msg:"ET TROJAN Probablemente Bot Nick en IRC (USA
+..)"; flow:established,to_server; flowbits:isset,is_proto_irc; content:"NICK "; pcre:"/NICK
.*USA.*[0-9] . reference:url,doc.emergingthreats.net/2008124; classtype:trojan-activity;
sid:2008124; rev:2;)
```

Nota. La figura representa el formato de la regla de protocolo del IDS Suricata, en donde el color rojo hace su representación. Tomado por Suricata 5.0.2, rules, documentación OISF, 2019, (" https://suricata.readthedocs.io/en/suricata-5.0.2/rules/intro.html").

Según Mohanta y Saldanha (2020), menciona que este campo contiene el protocolo del paquete que debe coincidir con la regla. Si un paquete es con cualquier otro protocolo que no sea el especificado en la regla, la regla no coincidirá en eso. Los valores de protocolo que podemos usar aquí pueden pertenecer a la Capa 3, Capa 4 o incluso Capa 7.

Figura 23

Los valores de protocolo de Capa 3 y Capa 4 que admite el lenguaje de reglas.

tcp	tcp-pkt	tcp-stream	udp	icmpv4	ip	
icmpv6	icmp	sctp	ip	ipv4	ipv6	ip6

Nota. La figura representa los diversos protocolos de capa 3 y capa 4 que se pueden especificar en el campo del PROTOCOLO de una regla de Suricata. Tomado de "*Malware Analysis and Detection Engineering: A Comprehensive Approach to Detect and Analyze Modern Malware"* (p. 840), por A. Mohanta y A. Saldanha, 2020. APRESS®.

Los mismos autores también menciona que Aparte de los protocolos, también puede especificar los protocolos de Capa 7, cuya lista se puede obtener ejecutando el siguiente comando:

Figura 24

Comando de Suricata para obtener varios protocolos de capa aplicación.

```
# suricata --list-app-layer-protos
```

Nota. La figura representa el comando Suricata para ejecutar y obtener la lista de varios protocolos de capa de aplicación que puede usar en el campo PROTOCOLO de una regla. Tomado de "*Malware Analysis and Detection Engineering: A Comprehensive Approach to Detect and Analyze Modern Malware"* (p. 841), por A. Mohanta y A. Saldanha, 2020. APRESS®.

Figura 25

Diversos protocolos de la capa 7.

http	ftp	smtp	tls	ssh	imap	smb
dns	enip	dnp3	nfs	ntp	dcerpc	ftp-data
tftp	ikev2	krb5	dhcp	snmp	modbus	

Nota. La figura representa los diversos protocolos de capa de aplicación de Capa 7 obtenidos al ejecutar el comando de la figura 23, que se pueden usar en el campo PROTOCOLO de

una regla Suricata. Tomado de "*Malware Analysis and Detection Engineering: A Comprehensive Approach to Detect and Analyze Modern Malware"* (p. 841), por A. Mohanta y A. Saldanha, 2020. APRESS®.

C. FUENTE Y DESTINO

Según la organización Open Information Security (OISF), menciona que la primera parte enfatizada es la fuente, la segunda es el destino (tenga en cuenta la dirección de la flecha direccional). Con el origen y el destino, usted especifica el origen del tráfico y el destino del tráfico, respectivamente. Puede asignar direcciones IP (se admiten IPv4 e IPv6) e intervalos IP. Estos se pueden combinar con operadores:

Figura 26

Regla de fuente y destino de Suricata.

```
drop tcp $HOME_NET any -> $EXTERNAL_NET any (msg:"ET TROJAN Probablemente Bot Nick en IRC (USA
+..)"; flow:established,to_server; flowbits:isset,is_proto_irc; content:"NICK "; pcre:"/NICK
.*USA.*[0-9] . reference:url,doc.emergingthreats.net/2008124; classtype:trojan-activity;
sid:2008124; rev:2;)
```

Nota. La figura representa el formato de la regla de fuente y destino del IDS Suricata, en donde el color rojo hace su representación. Tomado por *Suricata 5.0.2, rules*, documentación OISF, 2019, (" https://suricata.readthedocs.io/en/suricata-5.0.2/rules/intro.html").

Figura 27

Operadores de la Regla de fuente y destino de Suricata.

Operador	Descripción
.. /..	Intervalos IP (notación CIDR)
!	excepción/negación
[.., ..]	Agrupación

Nota. La figura representa los operadores de la regla de fuente y destino del IDS Suricata, en donde hace también la descripción de ellas. Tomado por *Suricata 5.0.2, rules*, documentación OISF, 2019, ("https://suricata.readthedocs.io/en/suricata-5.0.2/rules/intro.html".).

La misma organización también menciona que normalmente, también haría uso de variables, como **y**. El archivo de configuración especifica las direcciones IP que se refieren, y esta configuración se utilizará en lugar de las variables en las reglas.

Figura 28

Uso de variables del IDS de Suricata.

Ejemplo	Significado
! 1.1.1.1	Cada dirección IP excepto 1.1.1.1
! [1.1.1.1, 1.1.1.2]	Cada dirección IP excepto 1.1.1.1 y 1.1.1.2
$HOME_NET	Su configuración de HOME_NET en yaml
[$EXTERNAL_NET, !$HOME_NET]	EXTERNAL_NET y no HOME_NET
[10.0.0.0/24, !10.0.0.5]	10.0.0.0/24 excepto por 10.0.0.5
[..., [....]]	
[..., ! [.....]]	

Nota. La figura representa los usos de las variables del IDS Suricata, en donde hace también la descripción de ellas. Tomado por *Suricata 5.0.2, rules*, documentación OISF, 2019, (“ https://suricata.readthedocs.io/en/suricata-5.0.2/rules/intro.html”).

El campo SRC_IP contiene la dirección IP que debería hacer coincidir con la dirección IP de origen del paquete, mientras que el campo DST_IP corresponde a la dirección IP de su destino del paquete. Estos campos le proporcionan una forma expresiva de especificar direcciones IP. No solo permite especificar direcciones IP únicas, también direcciones IP múltiples. También puede especificar rangos de IP negados, y puede hacer múltiples combinaciones para especificar expresamente IP de direcciones que la regla debe coincidir. Una ventaja adicional de estos campos es que también le permite especificar rangos de subred usando la notación CIDR (Mohanta y Saldanha ,2020, p. 841).

Figura 29

Regla con variable para los campos SRC_IP y DST_IP.

```
vars:
  address-groups:
      HOME_NET: "[192.168.0.0/16,10.0.0.0/8,172.16.0.0/12]"
      EXTERNAL_NET: "!$HOME_NET"
```

Nota. La figura representa los Variables que puede definir en el archivo Suricata.yaml para que pueda especificar en una regla para los campos SRC_IP y DST_IP. Tomado de "*Malware Analysis and Detection Engineering: A Comprehensive Approach to Detect and Analyze Modern Malware"* (p. 841), por A. Mohanta y A. Saldanha, 2020. APRESS®.

Figura 30

Listado de algunos valores SRC_IP y DST_IP.

```
10.8.0.1
[10.8.0.1,10.8.0.2]
[10.8.0.0/16]
[!10.8.0.0/16]
[!10.8.0.0/16, 10.8.25.1]
HOME_NET
!HOME_NET
```

Nota. La figura representa algunos valores SRC_IP y DST_IP. Tomado de "*Malware Analysis and Detection Engineering: A Comprehensive Approach to Detect and Analyze Modern Malware"* (p. 841), por A. Mohanta y A. Saldanha, 2020. APRESS®.

D. PUERTOS

Según la organización Open Information Security (OISF), menciona que el tráfico entra y sale a través de los puertos. Diferentes puertos tienen diferentes números de puerto. Por ejemplo, el puerto predeterminado para HTTP es 80 mientras que 443 suele ser el puerto para HTTPS. Tenga en cuenta, sin embargo, que el puerto no dicta qué protocolo se utiliza en la comunicación. En su lugar, determina qué aplicación está recibiendo los datos.

Figura 31

Trafico dentro de los puertos del IDS de Suricata.

```
drop tcp $HOME_NET any -> $EXTERNAL_NET any (msg:"ET TROJAN Probablemente Bot Nick en IRC (USA
+..)"; flow:established,to_server; flowbits:isset,is_proto_irc; content:"NICK "; pcre:"/NICK
.*USA.*[0-9] . reference:url,doc.emergingthreats.net/2008124; classtype:trojan-activity;
sid:2008124; rev:2;)
```

Nota. La figura representa el tráfico que entra y sale a través de los puertos del IDS Suricata. Tomado por *Suricata 5.0.2, rules,* documentación OISF, 2019, ("https://suricata.readthedocs.io/en/suricata-5.0.2/rules/intro.html").

La misma organización menciona que los puertos mencionados anteriormente son típicamente los puertos de destino. Los puertos de origen, es decir, la aplicación que envió el paquete normalmente recibe asignado un puerto aleatorio por el sistema operativo. Al escribir una regla para su propio servicio HTTP, normalmente escribiría, ya que eso significaría que cualquier paquete de cualquier puerto de origen a la aplicación HTTP (que se ejecuta en el puerto 80) coincide.**any -> 80.**

Figura 32

Puertos del IDS de Suricata.

Ejemplo	Significado
[80, 81, 82]	puertos 80, 81 y 82
[80: 82]	Rango de 80 a 82
[1024:]	Desde 1024 hasta el número de puerto más alto
!80	Todos los puertos menos 80
[80:100,!99]	Rango de 80 a 100 pero 99 excluidos
[1:80,! [2,4]]	Rango de 1-80, excepto los puertos 2 y 4
[.., [.,.,.]]	

Nota. La figura representa los puertos de origen y destino del IDS Suricata. Tomado por *Suricata 5.0.2, rules,* documentación OISF, 2019, ("https://suricata.readthedocs.io/en/suricata-5.0.2/rules/intro.html").

Muy similar a SRC_IP y DST_IP, estos campos le permiten especificar valores de puerto que debe coincidir con estas cualidades de varios puertos de inicio y destino de los paquetes. La sintaxis y la expresividad de estos campos en una regla sigue el mismo formato que SRC_IP y DST_IP campos, incluida la disponibilidad de la especificación de **vars** en el archivo de configuración **yaml,** que puede luego especifique como valores para estos campos en la regla. Tenga en cuenta que los números de puerto son una característica que está presente en ciertos protocolos como TCP y UDP, y las reglas que escribe con números de puerto específicos deben apuntar a paquetes que llevan encabezados de Capa 4 que admiten números de puerto (Mohanta y Saldanha ,2020, p. 842).

E. DIRECCIÓN

Según la organización Open Information Security (OISF), menciona que la dirección indica en qué dirección debe coincidir la firma. Casi todas las firmas tienen una flecha a la derecha (). Esto significa que solamente los paquetes con la misma dirección pueden hacer juego. Sin embargo, también es posible que una regla coincida en ambos sentidos **() :-><>**

Figura 33

Configuración de la dirección en el IDS de Suricata.

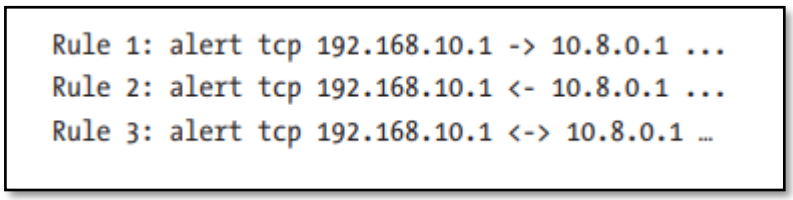
```
source -> destination
source <> destination  (both directions)
```

Nota. La figura representa la configuración y comandos de la dirección en el IDS de Suricata. Tomado por *Suricata 5.0.2, rules*, documentación OISF, 2019, ("https://suricata.readthedocs.io/en/suricata-5.0.2/rules/intro.html").

Según Mohanta y Saldanha (2020), menciona que este campo toma uno de los 3 valores: ->, <- o <->. Este campo especifica la dirección del paquete para los campos SRC_ * y DST_ * con los que la regla debe coincidir.

Figura 34

Reglas de muestra con diferentes valores de DIRECTION.



Nota. La figura representa algunos ejercicios de reglas de muestra con diferentes valores de DIRECTION. Tomado de "*Malware Analysis and Detection Engineering: A Comprehensive Approach to Detect and Analyze Modern Malware"* (p. 842), por A. Mohanta y A. Saldanha, 2020. APRESS®.

a) REGLAS DE SOLO IP

Es incorrecto si dijéramos que, si escribimos una regla con solo los siete campos de la sección, la regla es inútil. Muchas reglas se escriben con solo estos siete campos establecidos y un campo en la sección de palabras clave llamado SID. Sin otras palabras clave. Es ACCIÓN, SRC_IP, SRC_PORT, DIRECTION, DEST_IP, DEST_PORT y la palabra clave SID. Estas reglas se denominan reglas de solo IP. En el mundo del malware donde tiene servidores maliciosos que surgen y se apagan cada día, las empresas de seguridad escriben reglas de solo IP que contienen direcciones IP de servidores maliciosos, para que puedan detectar cualquier comunicación que se produzca desde los hosts de la red a estos servidores maliciosos, lo que sugiere una infección de malware en el host. Además, una ventaja adicional de las reglas de solo IP es que el motor de detección de Suricata los maneja de manera eficiente, ya que solo se emparejan en el primer paquete de un flujo, reduciendo así la sobrecarga de inspección de reglas para los paquetes posteriores, lo que lo convierte en la opción de escribir reglas que apunten a detectar puramente comunicaciones basadas en direcciones IP (Mohanta y Saldanha ,2020, p. 843).

b) PALABRAS CLAVE

El jugo real de un paquete está en las partes internas de un paquete con varios detalles repartidos por sus diversos campos. Todos estos campos están expuestos a través del lenguaje de reglas de Suricata a través de varias palabras clave, y esto es lo que utilizan la mayoría de las reglas desarrolladores de contenido para escribir reglas expresivas para que coincidan en estos paquetes que fluyen sobre la red. Ahora, las palabras clave que desea usar

en la regla van todas en los dos corchetes **()** de una regla. Las palabras clave están separadas por punto y coma y espacios, lo que le permite especificar múltiples palabras clave. No todas las palabras clave necesitan un valor, pero si necesita un valor, se suministra con la ayuda de dos puntos que separan la palabra clave y su valor (Mohanta y Saldanha ,2020, p. 843).

Figura 35

Estructura para especificar palabras clave en una regla Suricata.

Nota. La figura representa la Estructura para especificar palabras clave en una regla Suricata Tomado de "*Malware Analysis and Detection Engineering: A Comprehensive Approach to Detect and Analyze Modern Malware"* (p. 844), por A. Mohanta y A. Saldanha, 2020. APRESS®.

c) SID EN PALABRAS CLAVE

SID es la palabra clave que se necesita en todas las reglas de Suricata. Necesita un valor, y el valor contiene la identificación de la firma, que es un valor numérico sin firmar y de 4 bytes de longitud, que identifica de forma única la regla. Cada regla cargada en Suricata debe tener un **sid** cuyo valor sea único en la lista de reglas cargadas. Si otra firma usa un valor sid que ya está siendo usado por otra regla que Suricata ha cargado, Suricata descarta la nueva regla que contiene el valor **sid** duplicado (Mohanta y Saldanha ,2020, p. 844).

F. OPCIONES DE REGLAS

Según la organización Open Information Security (OISF), menciona que el resto de la regla consta de opciones. Estos están entre paréntesis y separados por punto y coma. Algunas opciones tienen ajustes (como), que se especifican mediante la palabra clave de la opción, seguido de dos puntos, seguido de la configuración. Otros no tienen ajustes, y son simplemente la palabra clave.

Figura 36

Opciones de reglas en el IDS de Suricata.

```
<keyword>: <settings>;
<keyword>;
```

Nota. La figura representa las opciones de reglas en el IDS de Suricata. Tomado por *Suricata 5.0.2, rules*, documentación OISF, 2019, ("https://suricata.readthedocs.io/en/suricata-5.0.2/rules/intro.html").

Suricata admite varias palabras clave que se pueden utilizar para hacer coincidir todos los aspectos de un paquete. Con el apoyo de analizadores de capa de aplicaciones y los datos expuestos por ellos a través de varias palabras clave de lenguaje de reglas, puede escribir reglas precisas y de alto rendimiento para coincidir con casi cualquier tipo de contenido de carga útil de paquetes que fluya a través de su red. Puede obtener más información sobre el lenguaje de reglas de Suricata que se describe en la Guía del usuario de Suricata. También puede consultar el Manual de usuario de SNORT, que también cubre el idioma de las reglas de Snort en detalle. Tenga en cuenta que existen algunas variaciones sintácticas y semánticas entre Snort y Suricata gobiernan los lenguajes, pero por lo demás, son en gran medida similares. Suricata también proporciona una opción de línea de comandos que enumera todas las palabras clave que expone a través de su lenguaje de reglas (Mohanta y Saldanha ,2020, p. 849).

Figura 37

Comando para enumerar todas las palabras.

```
# suricata --list-keywords
```

Nota. La figura representa el Comando Suricata para enumerar todas las palabras clave expuestas por el lenguaje de reglas SuricataTomado de "*Malware Analysis and Detection Engineering: A Comprehensive Approach to Detect and Analyze Modern Malware"* (p. 849), por A. Mohanta y A. Saldanha, 2020. APRESS®.

Los mismos autores mencionan que lo importante de trabajar con cualquier IDPS o Suricata es que necesita conocer todas las funciones proporcionadas por el IDPS y sus lenguajes de reglas para hacer sus funciones y poder. Como analistas de malware, el aspecto de red de

una amenaza es de suma importancia, y la mayoría de las infecciones se pueden detectar mediante una inspección minuciosa de los paquetes de red. Lo mismo ocurre con la escritura de reglas. Intente escribir tantas reglas como sea posible, recogiendo diferentes archivos PCAP con diferentes protocolos. La práctica hace la perfección. Suricata también ofrece varias opciones para perfilar las reglas de rendimiento, de modo que puedes escribir reglas eficientes. Escribir reglas eficientes es de suma importancia. Si una regla IDPS es ineficaz, puede conducir a malas rendimiento, lo que pueden llevarnos a la escasez de los paquetes, lo que puede conducir a detecciones perdidas, que finalmente conduce a infecciones no detectadas en la red e incluso a evitar la prevención silo están ejecutando en modo IPS.

2.2.9.3. FUNCIONAMIENTO DE SURICATA

Según la organización Open Information Security (OISF), Los modos de funcionamiento de Suricata son los mismos que los de Snort. Se puede utilizar como sistema IDS o IPS. No existen diferencias al conectar Suricata a la red. Suricata incluso tiene básicamente la misma sintaxis de reglas que Snort (aunque no al 100%), lo que significa que ambos sistemas pueden usar más o menos las mismas reglas. El flujo de datos general a través de Suricata es similar a Snort. Los paquetes se capturan, decodifican, procesan y analizan. Sin embargo, cuando se trata de los componentes internos del motor Suricata, las diferencias se hacen evidentes.

Suricata es un sistema multiproceso que es más eficaz y escalable que Snort que es de un solo hilo. Inspecciona los datos entrantes hasta que finalice la conexión a Internet, los paquetes se inspeccionan utilizando el concepto de ventana deslizante. Suricata recibe el primer segmento e inmediatamente lo inspecciona. Entonces recibe el segundo segmento, lo junta con el primero y lo inspecciona. Al final, atrapa el tercer segmento, corta el primero, junta el segundo segmento con el tercer segmento, y lo inspecciona. Una vez que se inspecciona el paquete completo y no hay malware se encontró o se detectó una intrusión, el paquete se envía al receptor. Además, Suricata tiene dos modos que actúan sobre la intrusión cuando se detecta, es decir, un modo de caída y o un modo de rechazo (Moloja,2018, p. 115).

El mismo autor menciona los modos que actúa Suricata sobre la intrusión:

a) Gota (modo IPS):

-Si una firma que contiene una acción de soltar coincide con un paquete, se descarta inmediatamente y no se enviará más lejos.

- El receptor no recibe el mensaje, lo que resulta en un tiempo muerto.
- Todos los siguientes paquetes del mismo remitente son caído.
- Se genera una alerta para un paquete

b) Rechazar (modo IDS e IPS):

- Este es un rechazo activo del paquete; ambos el receptor y el remitente reciben un paquete rechazado.

2.2.10. Kali Linux

Según (Gonzales, Sánchez y Soriano,2013) mencionan que en este caso el "Kali Linux" nos puede ayudar a poder hacer la auditoria de los sistemas que nos informan. Ello debido a la recolección de la información, ante el análisis de las vulnerabilidades y la exploración de ellas mismas, estas son las ramificaciones de dicha seguridad informática que "Kali Linux" ha profundizado con buen augurio.

2.2.11. OPEN SOURCE

Según Easttom (2014), menciona que el código abierto es una forma de licenciar software. Significa que el software se puede distribuir libremente y contiene el código fuente. Esto significa que los usuarios pueden hacer copias, dárselas a amigos e incluso obtener una copia del código fuente. La idea detrás del código abierto es animar a los usuarios a examinar el código fuente de un producto y, si es posible, mejorarlo. La creencia es que, a través de la revisión y las mejoras realizadas por tantas personas, un producto alcanzará un mayor nivel de calidad más rápido que los comerciales.

Según Gonzáles-Barahona (2011), menciona que el concepto de open source es esencial: es un sistema en donde se pueden hacer varios tipos de cosas, ya que el encargado da la autorización y para eso tiene un enfoque bastante distinto sobre los sujetos que deberían poder realizar.

2.2.12. VMware

Según Robles (sf), "VMware es un sistema de virtualización por software ósea es un programa que simula un sistema físico con unas características de hardware determinadas. Cuando se ejecuta el programa proporciona un ambiente de ejecución similar a todos los

efectos a un computador físico, con CPU, BIOS, tarjeta gráfica, memoria RAM, tarjeta de red, sistema de sonido, conexión USB, disco duro, etc".

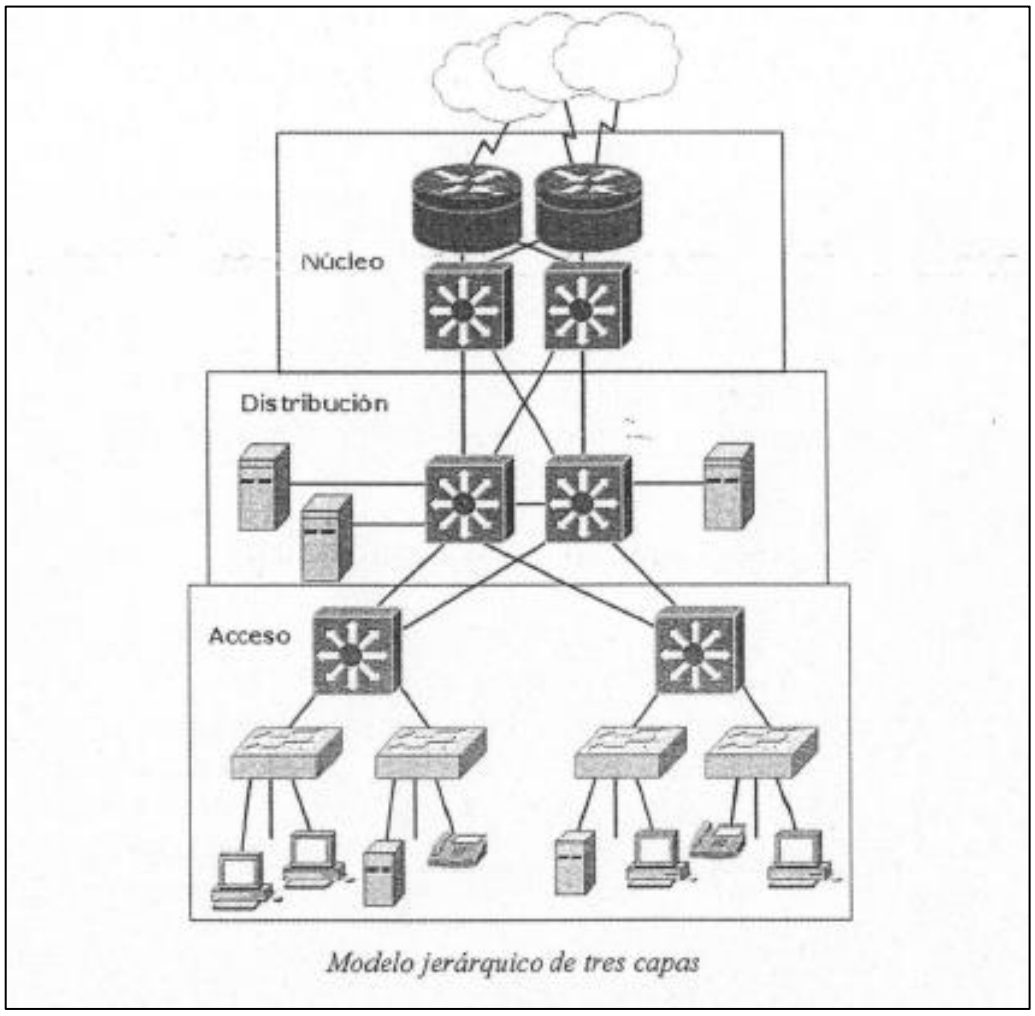
2.2.13. MODELO JERARQUICO DE TRES CAPAS

"Con el fin de simplificar el diseño, implementación y administración de las redes, Cisco utiliza un modelo jerárquico para describir la red. Aunque la práctica de este método suele estar asociada con el proceso de diseño de una red, es importante comprender el modelo para poder determinar el equipo y características que van a necesitar en la red. Un modelo jerárquico acelera la convergencia, mantiene posibles problemas aislados por capas y reduce la sobrecarga en los dispositivos". (Ariganello ,2011, p. 51).

"En un modelo jerárquico implica dividir la red en capas independientes. Cada capa (o nivel) en la jerarquía proporciona funciones específicas que definen su función dentro de la red general. Esto ayuda al diseñador y al arquitecto de red a optimizar y seleccionar las características, el hardware y el software de red adecuados para llevar a cabo las funciones específicas de esa capa de red. Los modelos jerárquicos se aplican a diseños de LAN y WAN" (Wikipedia, s.f)

Figura 38

Modelo jerárquico de tres capas.



Nota. El grafico representa el modelo jerárquico de tres capas (núcleo, distribución, acceso). Tomado de "*Guía de estudio para la certificación CCNA 640-8028"* (p.51), por E. Ariganello,2011, Redes Cisco.

A. CAPA DE ACCESO

"La capa de acceso de la red es el punto en el que cada usuario se conecta a la red. Esta es la razón por la cual la capa de acceso se denomina a veces capa de puesto de trabajo, capa de escritorio o de usuario. Los usuarios, así como los recursos a los que estos necesitan acceder con más frecuencia están disponibles a nivel local. El tráfico hacia y desde recursos locales está confinado entre los recursos, switches y usuarios finales" (Ariganello,2011).

B. CAPA DE DISTRIBUCIÓN

"La capa de distribución puede resumirse como la capa que proporciona una conectividad basada en una determinada política, dado que determina cuándo y cómo los paquetes pueden acceder a los servicios principales de la red. La capa de distribución determina la forma más rápida para la petición de un usuario (como acceso al servidor de archivos) puede ser remitida al servidor. Una vez que la capa de distribución ha elegido la ruta, envía la petición a la capa de núcleo. La capa de núcleo podrá entonces transportar la petición al servicio apropiado" (Ariganello,2011).

C. CAPA DE NUCLEO

"La capa de núcleo, principal o core se encarga de desviar el tráfico lo más rápidamente posible hacia los servicios apropiados. Normalmente, el tráfico transportado se dirige o proviene de servicios comunes a todos los usuarios. Estos servicios se conocen como servicios globales o corporativos. Algunos de ellos pueden ser e-mail, el acceso a Internet o videoconferencia. Cuando un usuario necesita acceder a un servicio corporativo, la petición se procesa al nivel de la capa de distribución. El dispositivo de la capa de distribución envía la petición del usuario al núcleo" (Ariganello,2011).

Tabla 02

Modelo jerárquico de tres capas.

Capa	Caracteristicas	Hadware
Núcleo	Concierne el tráfico y lo deriva hacia un servicio requerido, y es esta comunicación rápida y segura.	**Los routers, switch multicapas**
Distribución	En este caso se da el direccionamiento, el filtrado, el permiso "WAN", la protección basada en políticas, también los servicios corporativos, el enrutamiento de "VLANs",	**El "Router"**

	el concepto de broadcast y multicast.	
Accesibilidad	En este caso se define los dominios que hay de colisión, las estaciones terminales, la localización de usuarios, los servicios de varios grupos de trabajo.	**"Hub, switch"**

Nota. La Tabla representa las funciones del modelo jerárquico de tres capas (núcleo, distribución, acceso). Tomado de "*Guía de estudio para la certificación CCNA 640-8028"* (p.51), por E. Ariganello,2011, Redes Cisco.

2.2.14. SEGURIDAD CORPORATIVA

"La Seguridad Corporativa es un concepto consolidado. Aunque no existe una definición concreta, entre otras razones por la versatilidad del modelo, entendemos como Seguridad Corporativa el conjunto de políticas, procedimientos y recursos humanos, organizativos y técnicos destinados a proteger a las personas, a los activos tangibles e intangibles y a la reputación de una organización. También se puede identificar la Seguridad Corporativa como la función que identifica, gestiona y mitiga eficazmente, en una fase temprana, cualquier situación que pueda amenazar la resiliencia y la capacidad de supervivencia de una organización" (Muñoz, 2016).

Según Sarría (sf), menciona que la seguridad corporativa, debería de estar siempre presente en cada uno de los procesos de una empresa, y a la vez en cada área que se ejerce en base a la amenaza y su funcionamiento de ellos.

2.2.15. POBLACIÓN

Para Chávez (2007), la población "es el universo de estudio de la investigación, sobre el cual se pretende generalizar los resultados, constituida por características o estratos que le permiten distinguir los sujetos, unos de otros".

Según Tamayo y Tamayo (1997), la población se define como la totalidad del fenómeno a estudiar donde las unidades de población poseen una característica común la cual se estudia y da origen a los datos de la investigación.

Lepkowski (Como se citó en Hernández Sampieri, Fernández y Baptista, 2014) menciona que una población es el conjunto de todos los casos que concuerdan con una serie de especificaciones.

2.2.16. MUESTRA

"No siempre, pero en la mayoría de las situaciones sí realizamos el estudio en una muestra. Sólo cuando queremos efectuar un censo debemos incluir todos los casos (personas, animales, plantas, objetos) del universo o la población" (Hernández Sampieri et al., 2014).

CAPITULO III
METODOLOGÍA DE LA INVESTIGACIÓN

3.1. TIPO Y NIVEL DE INVESTIGACIÓN

A. TIPO DE INVESTIGACIÓN

"Investigación aplicada se distingue por tener propósitos prácticos inmediatos definidos, es decir, se investiga para actuar, transformar, modificar o producir cambios en un determinado sector de la realidad. Para realizar investigaciones aplicadas es muy importante contar con el aporte de las teorías científicas, que son producidas por la investigación básica y sustantiva" (p.44). Por esta consideración el tipo de **investigación es aplicada.**

B. NIVEL DE INVESTIGACIÓN

De acuerdo con el autor Hernández Sampieri Et. Al (2010), las investigaciones descriptivas vienen a ser "los estudios descriptivos buscan especificar las propiedades, las características y los perfiles importantes de personas, grupos, comunidades o cualquier otro fenómeno que se someta a análisis." De acuerdo con el autor (2006), “también menciona que dichas funciones principales de la investigación descriptiva es la capacidad para seleccionar las características fundamentales del objeto de estudio y su descripción detallada de las partes, categorías o clases de dicho objeto”; y agrega “La investigación descriptiva es uno de los tipos o procedimientos investigativos más populares y utilizados por los principiantes en la actividad investigativa. Los trabajos de grado, en los pregrados y en muchas maestrías, son estudios de carácter eminentemente descriptivo. En tales estudios se muestran, narran, reseñan o identifican hechos, situaciones, rasgos característicos de un objeto de estudio, o se diseñan productos, modelos, prototipos, guías, etcétera” (p.112).

De acuerdo con el autor Carrasco (2006), “señala que, la investigación descriptiva se soporta principalmente en técnicas como la encuesta, la entrevista, la observación y revisión documental. Este tipo de investigación estudia, analiza, describe y especifica situaciones y propiedades de personas, grupos, comunidades o cualquier otro fenómeno u objeto que sea sometido al análisis”. Por esta consideración el nivel de **investigación es descriptivo.**

3.2. DISEÑO DE INVESTIGACIÓN

De acuerdo con el autor Hernández Sampieri Et. Al (2010), menciona que podemos definir la investigación no experimental, "como aquella investigación que se realiza sin manipular deliberadamente las variables, se trata de estudios donde no hacemos variar en forma intencional las variables independientes para ver su efecto sobre otras variables. Lo que hacemos en la investigación no experimental es observar fenómenos tal como se dan en su contexto natural, para posteriormente analizarlos" (p.191).

Según Carrasco (2008), el diseño de investigación transversal descriptivo se empela para analizar y conocer las características, cualidades internas y externas, propiedades y rasgos esenciales de los hechos y fenómenos de un hecho realizado a momento determinado del tiempo. De acuerdo con Hernández et al 2010, una investigación de diseño transversal es "cuando se recolectan datos en un solo momento o en un tiempo único y su propósito es describir variables y analizar los hechos tal como se dan". Los instrumentos de recolección de datos, son usados durante el proceso de forma única. En esta investigación se tenía que analizar procesos, conocer características, estudiar rasgos y entender funcionalidades para encontrar datos necesarios para la construcción del aplicativo de inteligencia de negocios; por lo tanto, el diseño de investigación es **no experimental de tipo transversal descriptivo.**

3.3. HIPÓTESIS DE LA INVESTIGACIÓN

"No en todas las investigaciones cuantitativas se planean hipótesis. El hecho de formulemos o no hipótesis depende de un factor esencial: el alcance inicial del estudio. Las investigaciones cuantitativas que formulan hipótesis son aquellas cuyo planteamiento define que su alcance será correlacional o explicativo, o en las que tienen un alcance descriptivo, pero que intentan pronosticar una cifra o un hecho" (Hernández, baptista y Collado, 2014, p.104).

"Las investigaciones de tipo descriptivo no requieren formular hipótesis; es suficiente plantear algunas preguntas de investigación que, como ya se anotó, surgen del planteamiento del problema, de los objetivos y, por supuesto, del marco teórico que soporta el estudio" (Bernal, 2010, P.136). **La investigación que se desarrollará es de tipo descriptivo, por lo que no se pretende pronosticar hallar o verificar lo planteado en los objetivos, se optó por no plantear hipótesis.**

3.4. POBLACIÓN Y MUESTRA

A. POBLACIÓN

Todos los Sistemas de Detección de Intrusos (IDS)**.**

B. MUESTRA

Sistema de Detección de Intrusos Suricata Open Source.

3.5. DEFINICIÓN CONCEPTUAL DE LAS VARIABLES

VARIABLE DE ESTUDIO 1

a. Sistema de Detección de Intrusos Suricata Open Source. Viene a ser una herramienta de monitoreo y detección de ataques, anomalías en la red, mediante firmas, reglas previamente definidas. Este IDS da respuesta, alerta y responde ante cualquier anomalía.

DIMENSIONES

a) Sistema de Detección de Intrusos Suricata Open Source basada en palabras clave de protocolo. Es una configuración al IDS que trabaja en el tráfico de red mediante una palabra clave en una firma para indicar a que protocolo se refiere entre ellos el UDP, TCP, ICMP, IP, HTTP, FTP mediante la disponibilidad habilitada de configuración en suricata. yaml.

b. Sistema de Detección de Intrusos Suricata Open Source a nivel de perfilado de reglas. Es una configuración al IDS para que trabaja en base a reglas que generan la acción a cumplir por parte de las firmas con propiedades diferentes para cada suceso en la red previamente analizadas.

c. **Intrusos Suricata Open Source mediante algoritmos de comparación de patrones.** Es la creación o reutilización de patrones en común de protección de ataques en la red mediante algoritmos de comparación entre muestras de procesamiento de información de ataques cibernéticos de red.

VARIABLES DE ESTUDIO 2

a. **Seguridad Corporativa.** La seguridad Corporativa es aquella que identifica, administra y aplica de manera eficaz, ante una etapa temprana, ante cualquier suceso que podría amenazar a la resistencia y a la capacidad de sobrevivir de una empresa.

DIMENSIONES

b. **Entornos libres.** Son los softwares libres basados en Linux que utilizan las empresas corporativas para implementar la seguridad informática para la protección de datos.

3.6. DEFINICIÓN OPERACIONAL DE LAS VARIABLES

VARIABLE DE ESTUDIO 1

a. Sistema de Detección de Intrusos Suricata Open Source

DIMENSIONES

a. Basada en palabras clave de protocolo
b. Nivel de perfilado de reglas
c. Mediante algoritmos de comparación de patrones

VARIABLE DE ESTUDIO 2

a. Seguridad Corporativa

DIMENSIONES

a. Entornos libres

3.7. TÉCNICAS E INSTRUMENTOS

A. TÉCNICAS

Análisis documentario

CAPITULO VI

IMPLEMENTACIÓN DEL SISTEMA DE DETECCIÓN DE INSTRUSOS (IDS) EN LA EMPRESA

4.1. ANTECEDENTES DE LA EMPRESA

La empresa DYTELSUR SRL inicia su operación hace más de 4 años en el departamento de Ica desde el 2018 e inicia brindando su servicio corporativo de venta de servicios de fibra óptica, soporte, instalaciones de fibra óptica para empresas. Es una Sociedad de Responsabilidad Limitada, formada por dos socios fundadores. Posee un moderno equipamiento de redes, cuenta con técnicos altamente capacitados en diferentes materias relacionadas a telecomunicaciones.

4.1.1. ESTRUCTURA ORGANIZACIONAL

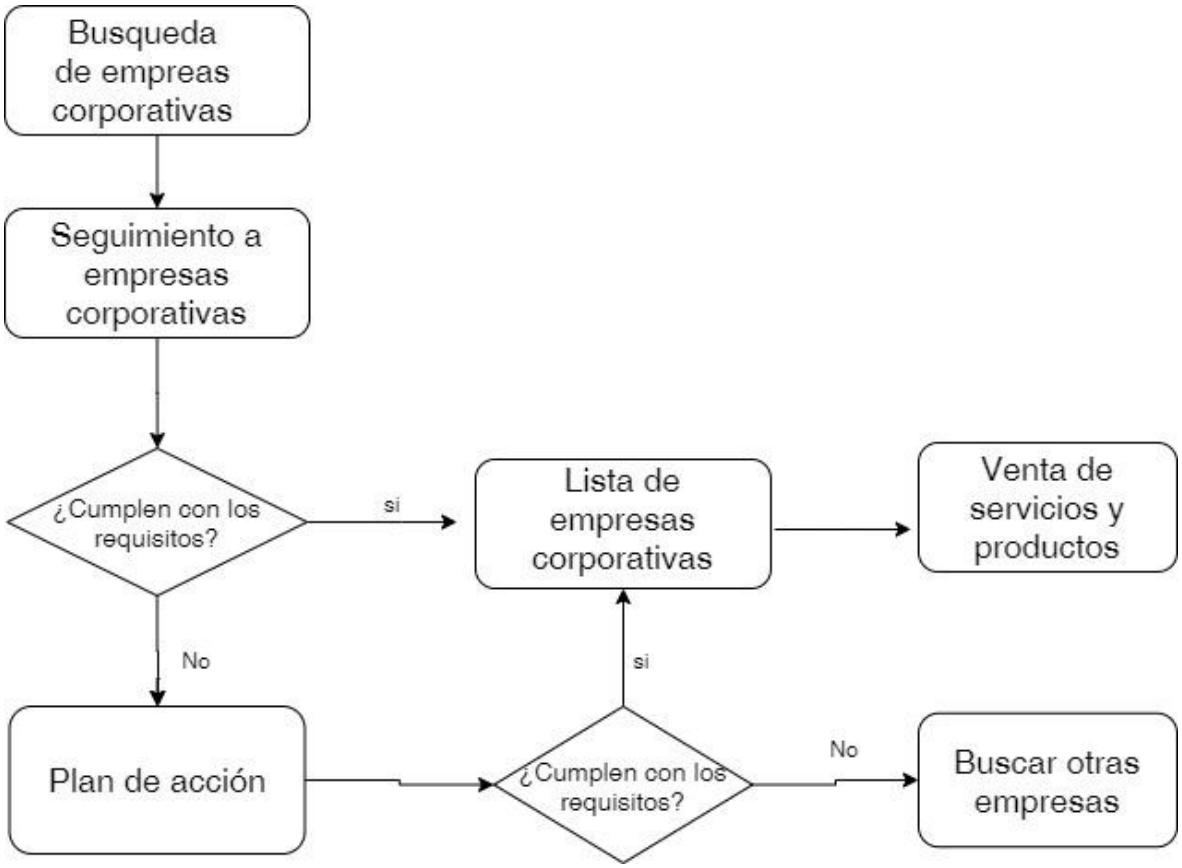
La empresa DYTELSUR SRL, está conformada por 5 áreas principales: Gerencia general, ventas, operaciones, contabilidad.

a) Gerencia General: Se denominó un gerente general, el cual es el encargado de la gerencia. La persona que ocupa este cargo es responsable de velar por el correcto funcionamiento de la empresa y lograr los objetivos propuestos.

b) Ventas: Esta área está encargada de las funciones comerciales, ya que como se indicó también se dedica al rubro de venta del servicio de fibra óptica a nivel corporativo y equipos de telecomunicaciones. Las principales actividades que se realizan en esta área es la de buscar empresas corporativas de cualquier rubro que requieran servicios de internet, mantenimiento, posteriormente se realiza seguimiento y se le ofrece con prospectos o proformas de costos de servicio. Una vez el cliente acceso a un servicio se otorga 7 días hábiles en caso de instalación para la instalación, en caso de averías de internet es en un plazo máximo de 12 horas.

Figura 39

Proceso de búsqueda de empresas corporativas, DYTELSUR *SRL.*



Nota. La figura representa el proceso de negocio de la empresa Dytelsur para el servicio de fibra óptica en las empresas corporativas. Elaboración propia.

c). Almacén: En esta se almacena los materiales de telecomunicaciones que utilizan los técnicos y también los equipos para su posterior venta.

d) Operaciones: La principal función que se realiza en esta área se encarga de las instalaciones de internet, HFC y mantenimiento en redes.

e). Contabilidad: Esta área es muy importante en toda la empresa ya que se realizan todos los procedimientos relacionados con información financiera que debe registrar su exactitud. Registrando y controlando los gastos e ingresos y las diversas operaciones económicas que realiza la empresa en sus diferentes actividades.

Figura 40

Organigrama de la empresa DYTELSUR *SRL*.

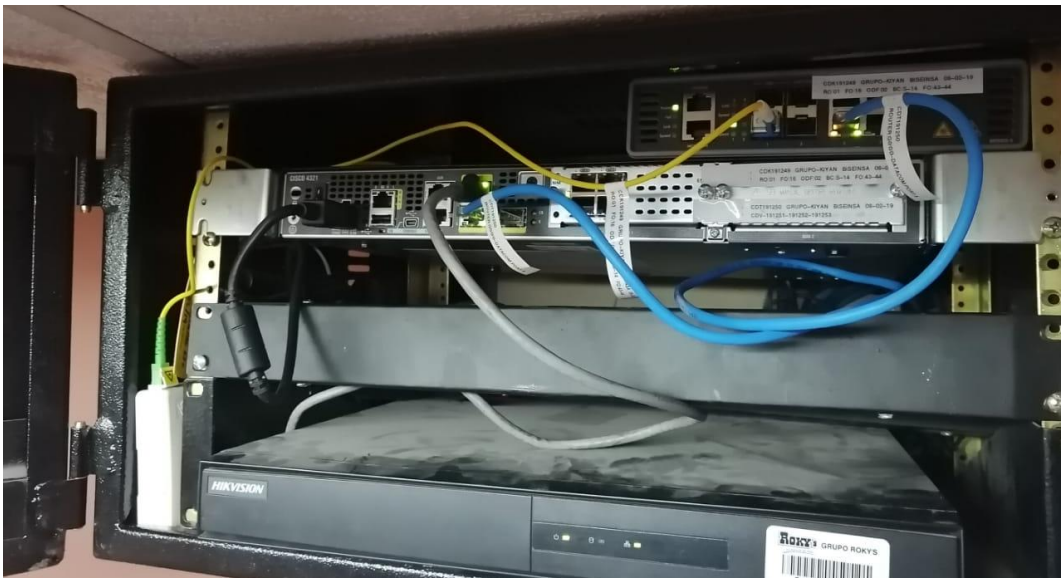
Nota. La figura representa el organigrama de la empresa Dytelsur donde se hace mención de sus áreas y su interrelación. Elaboración propia.

4.1.2. SITUACION ACTUAL DE INFRAESTRUCTURA DE RED

La empresa DYTELSUR cuenta con los siguientes dispositivos:

Figura 41

Equipos de red principal de la empresa DYTELSUR *SRL*.



Nota. La figura representa los equipos de red actuales que tiene la empresa Dytelsur y su conexión por medio de fibra óptica. Elaboración propia.

Tabla 03

Dispositivos de red y sus características de la empresa DYTELSUR SRL.

DISPOSITIVO	
ROUTER CISCO 4321	En este caso los servicios integrados de los routers Cisco 4000 vienen a ofrecer una potencia de reenviar Gigabit. Por ello es que se inician un amplio conjunto relacionados a servicios de red y a las aplicaciones que están dentro de una única plataforma.
SWITCH CISCO CATALIST 2800	• 2 o 24 puertos 10BaseT ofrecen 10 Mbps dedicados de ancho de banda • Funcionamiento dúplex completo en todos los puertos • Dos ranuras de expansión de alta velocidad maximizan la configuración de alta velocidad y la flexibilidad de la columna vertebral • Sin limitaciones de direcciones de control de acceso a medios (MAC) por puerto: proporciona flexibilidad para conectarse a usuarios individuales o concentradores compartidos
PC-01(contabilidad).	• Marca: HP 4ta Generación. • Modelo: 1000 Notebook PC. • Procesador: Intel Core i3. • Disco duro: 500 GB. • RAM 4 GB. • Sistema Operativo: Windows 10.
PC-2(Gerencia).	• Marca: Lenovo • Modelo: Lenovo E590 - Notebook - 15.6 • Procesador: Intel Core i5 • Disco duro: 500 GB

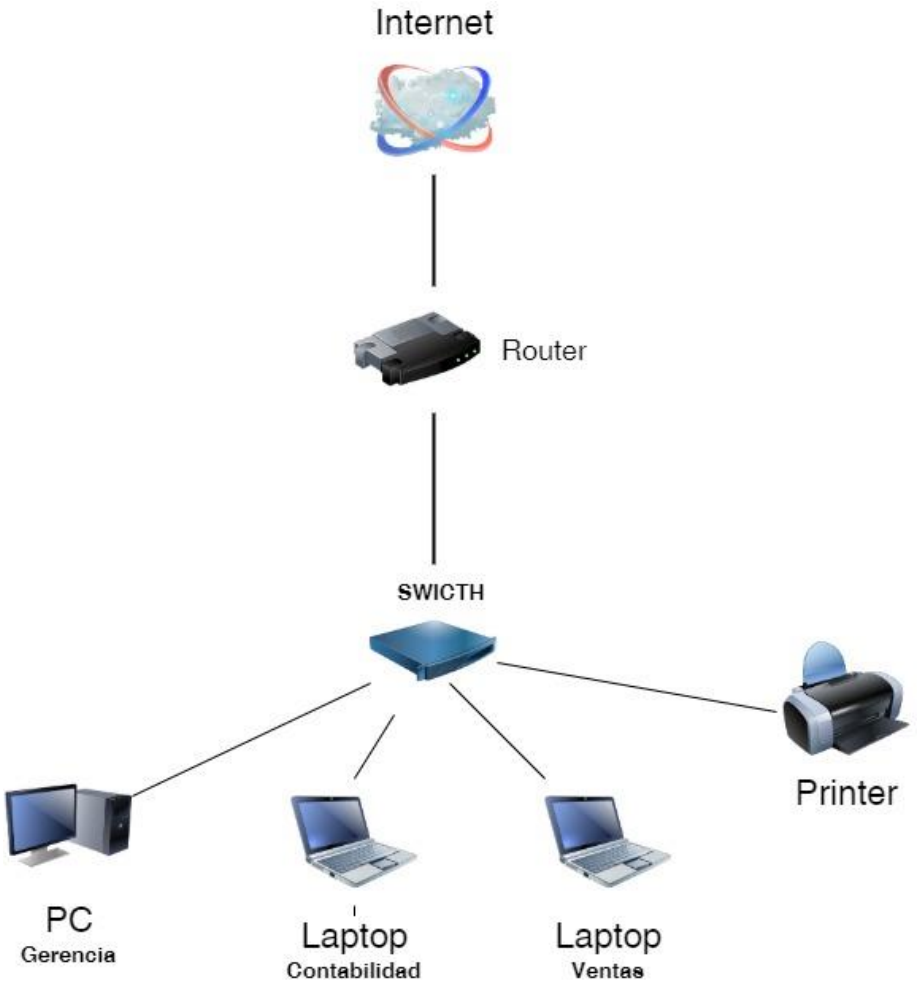
	• RAM 4 GB Sistema Operativo: Windows 10
PC-3(Ventas).	• Marca: HP • Modelo: HP 1000 - Notebook PC • Procesador: Intel Core i3R • Disco duro: 500 GB • RAM 4 GB Sistema Operativo: Windows 10
Impresora	• Modelo L4150. • Conexión Wifi. Escanea y fotocopiadora doble cara. Resolución de escáner 48 bits y 1200 x 2400 dpi.

Nota: La tabla hace mención de todos los dispositivos de red con la cual cuenta la empresa Dytelsur.Elaboración propia.

La estructura de red de esta empresa DYTELSUR SRL, como se detalla en la Figura 40, el internet llega a través del router por medio de fibra óptica, este se conecta al router de ahí al switch a través de cableado UTP CAT 6, y posteriormente llega hacia todos los dispositivos del network interno.

Figura 42

Topología de red de la empresa DYTELSUR SRL.



Nota. La figura representa la topología de red anillo en este caso de la empresa Dytelsur y los equipos interconectados. Elaboración propia.

4.2. DISEÑO DE LA ESTRUCTURA DE RED BASADA EN SISTEMA DE DETECCIÓN DE INTRUSOS A NIVEL DE RED (NIDS)

De acuerdo al capítulo II en la sección 2.2.13, la estructura jerárquica de la red está dividida en 3 capas:

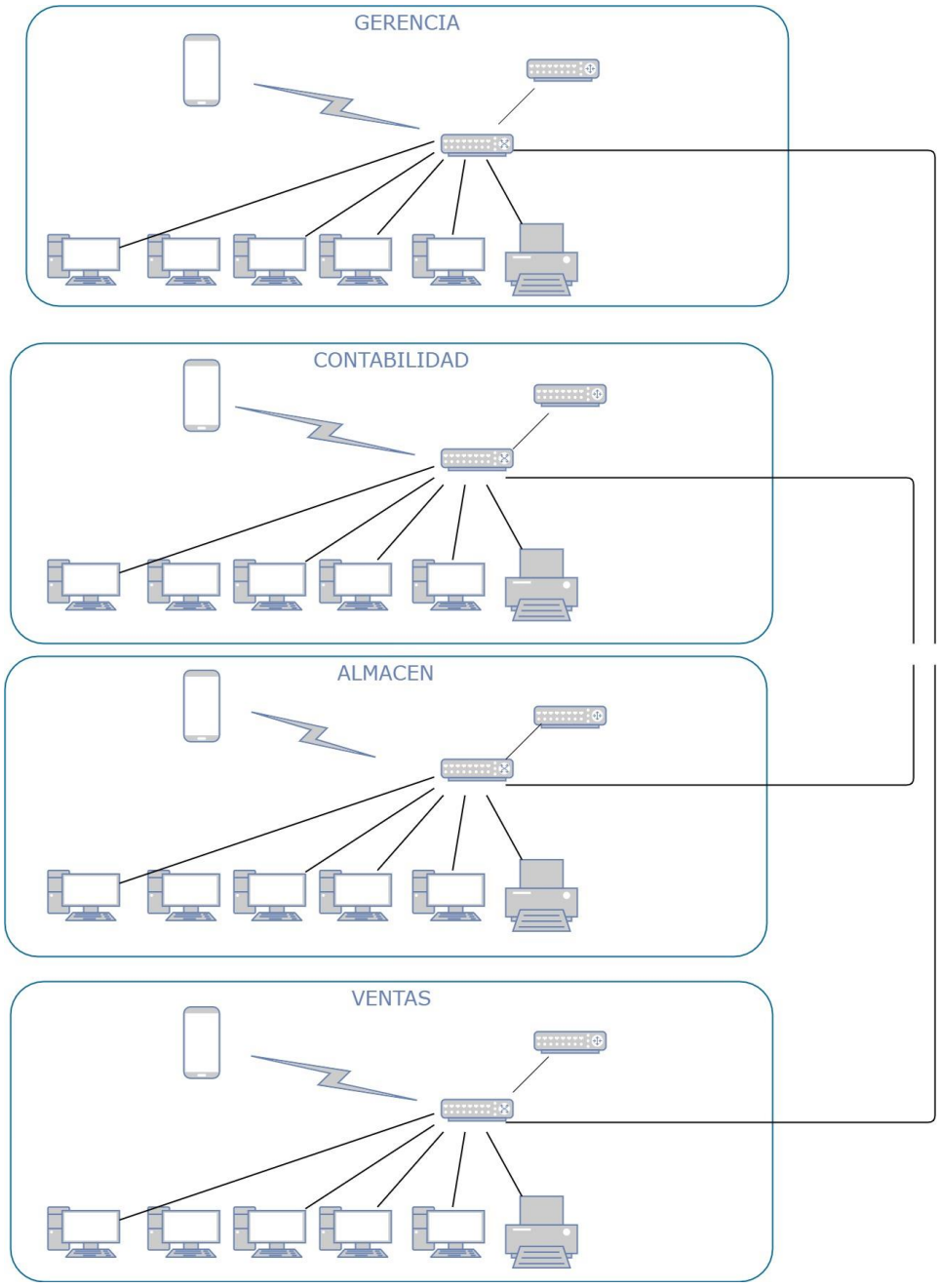
A. CAPA DE ACCESO

La red esta creada de acuerdo a las diferentes áreas de la empresa: Gerencia, contabilidad, almacén y ventas. Las cuales forman parte de esta capa de acceso, como se detalla en la figura 4.4. Todas las áreas de la empresa cuentan con la misma topología y estructura de red. La topología que muestra en cada una de las áreas es la topología estrella,

donde la conexión entre los diferentes dispositivos finales y el switch es de punto a punto. El access point puede tener conexión con los dispositivos por medios no guiados, para interconectar los 05 ordenadores con el switch se utiliza el cableado UTP y conectores RJ-45, mientras que para los móviles la conexión es través del wifi y el access point.

Figura 43

Estructura de red dentro del nivel de la capa de acceso para la empresa DYTELSUR SRL.



Nota. La figura representa la capa de acceso planteado dentro del nivel jerárquico de red para la empresa Dytelsur y la interrelación para cada área de trabajo. Elaboración propia.

Tabla 04

Dispositivos de red dentro de la capa de acceso para la empresa DYTELSUR SRL.

DISPOSITIVO DE ENTRADA	**MEDIO**	**DISPOSITIVO DE SALIDA**
PC-01	Cable UTP Cat 6 (guiado)	Switch
PC-02	Cable UTP Cat 6 (guiado)	Switch
PC-03	Cable UTP Cat 6 (guiado)	Switch
PC-04	Cable UTP Cat 6 (guiado)	Switch
Impresora	Cable UTP Cat 6 (guiado)	Switch
Móvil	Wifi (no guiado)	Access Point

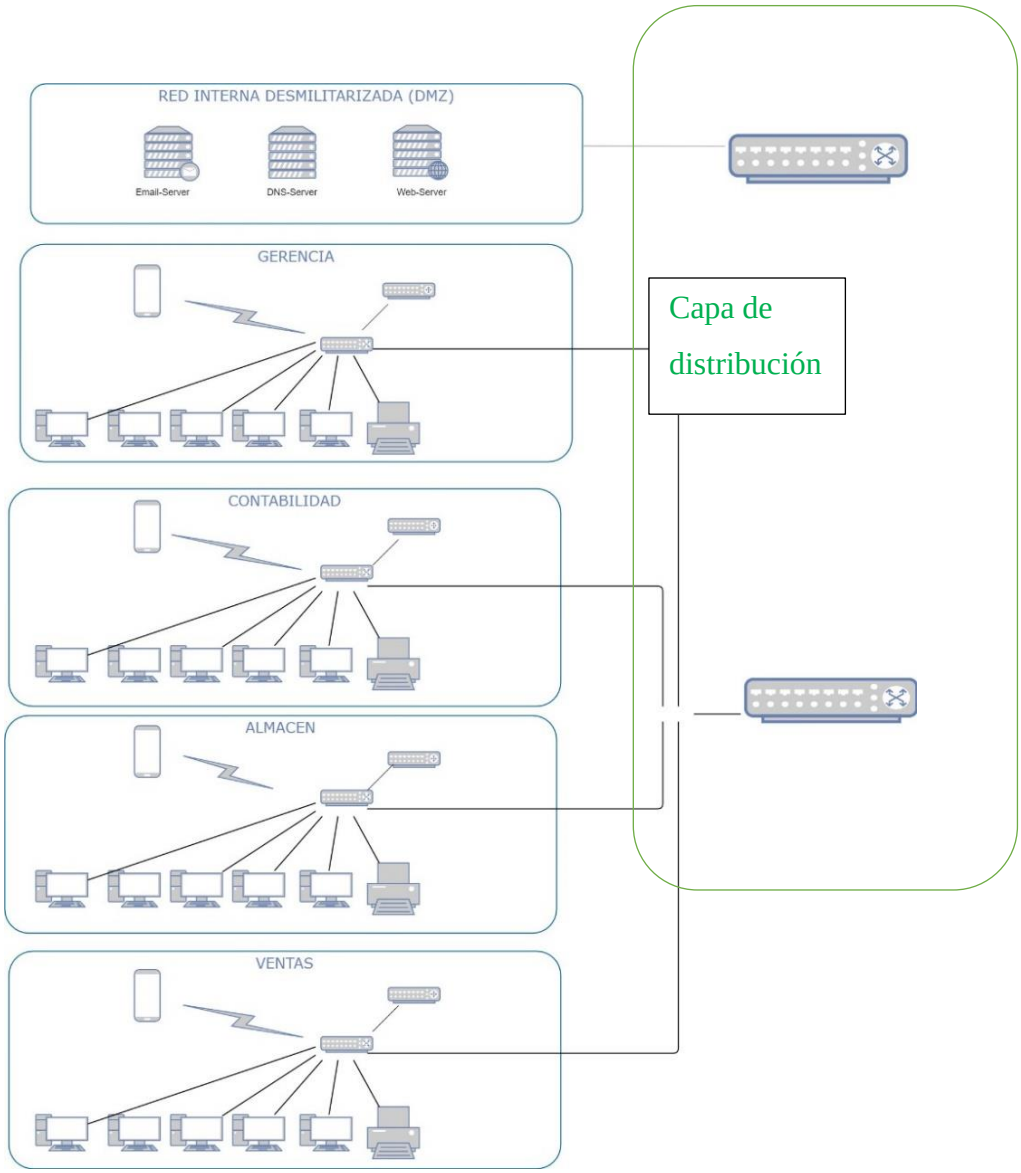
Nota. La figura representa los dispositivos de entrada y salida y medios guiados de red que interconectan dentro de la capa de acceso. Elaboración propia.

B. CAPA DE DISTRIBUCION

Dentro de dicha capa se conecta la capa acceso con la capa núcleo. La capa de distribución está conformada por 2 switch, como se muestra en la figura 43, lo que realiza es agregar los datos recibidos de la capa de acceso antes de que se transmitan a la capa núcleo para el enrutamiento hacia su destino final, se utiliza el cableado UTP cat 6 y conectores RJ-45.

Figura 43

Estructura de red dentro de la capa de distribución para la empresa DYTELSUR SRL.



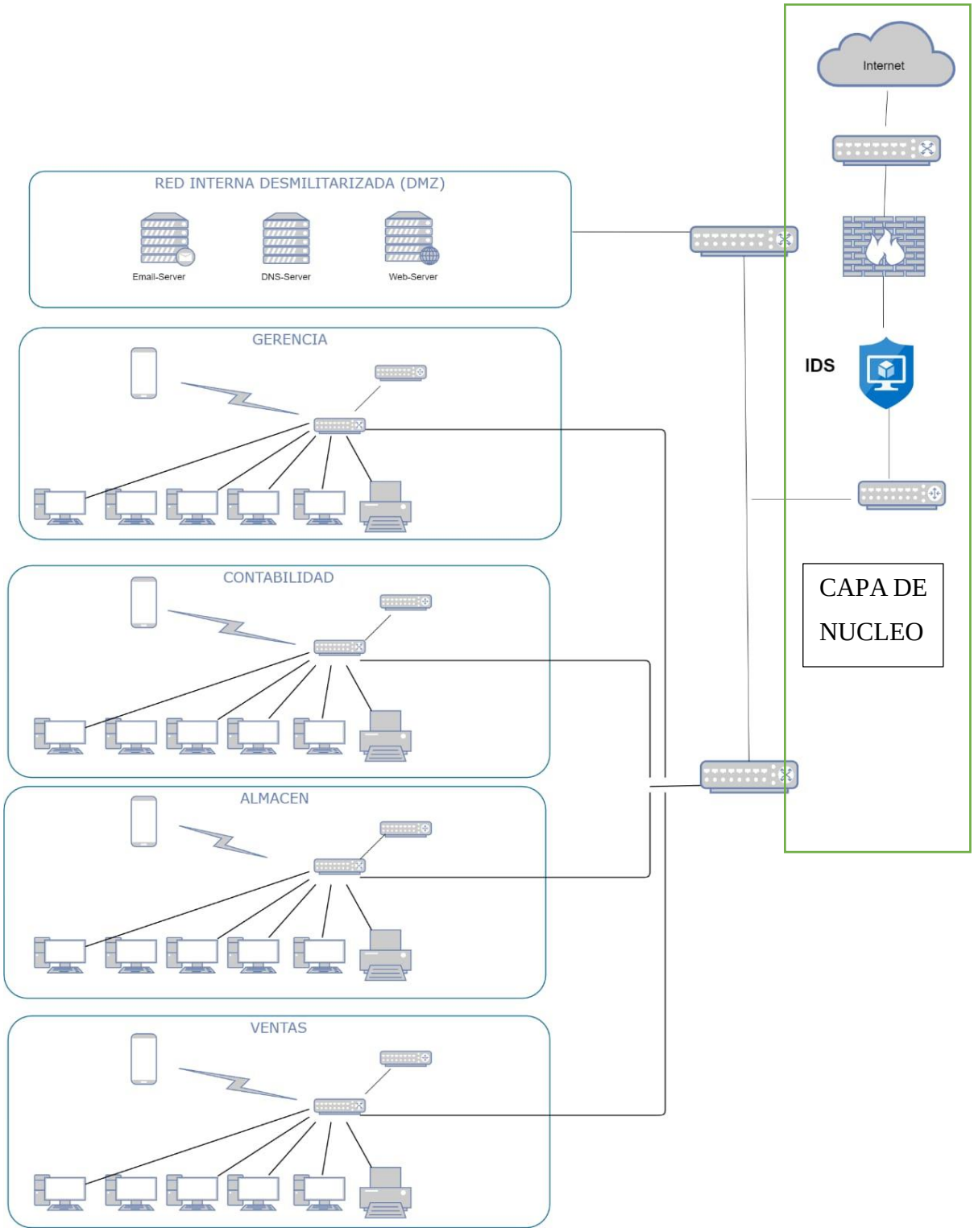
Nota. La figura representa la capa de distribución para la empresa Dytelsur, donde la capa de acceso se conecta con la capa núcleo. Elaboración propia.

C. CAPA DE NUCLEO

La capa núcleo está conformada por 1 switch, como se muestra en la figura 44 esta capa es fundamental para la interconectividad entre los elementos de la capa de distribución y la capa núcleo. La capa núcleo complementa el tráfico de todos los dispositivos de la capa de distribución, y reenvía grandes cantidades de datos rápidamente, para el enrutamiento hacia su destino se utiliza el cableado UTP y conectores RJ-45.

Figura 44

Estructura de red a nivel de la capa de núcleo en la empresa DYTELSUR SRL.



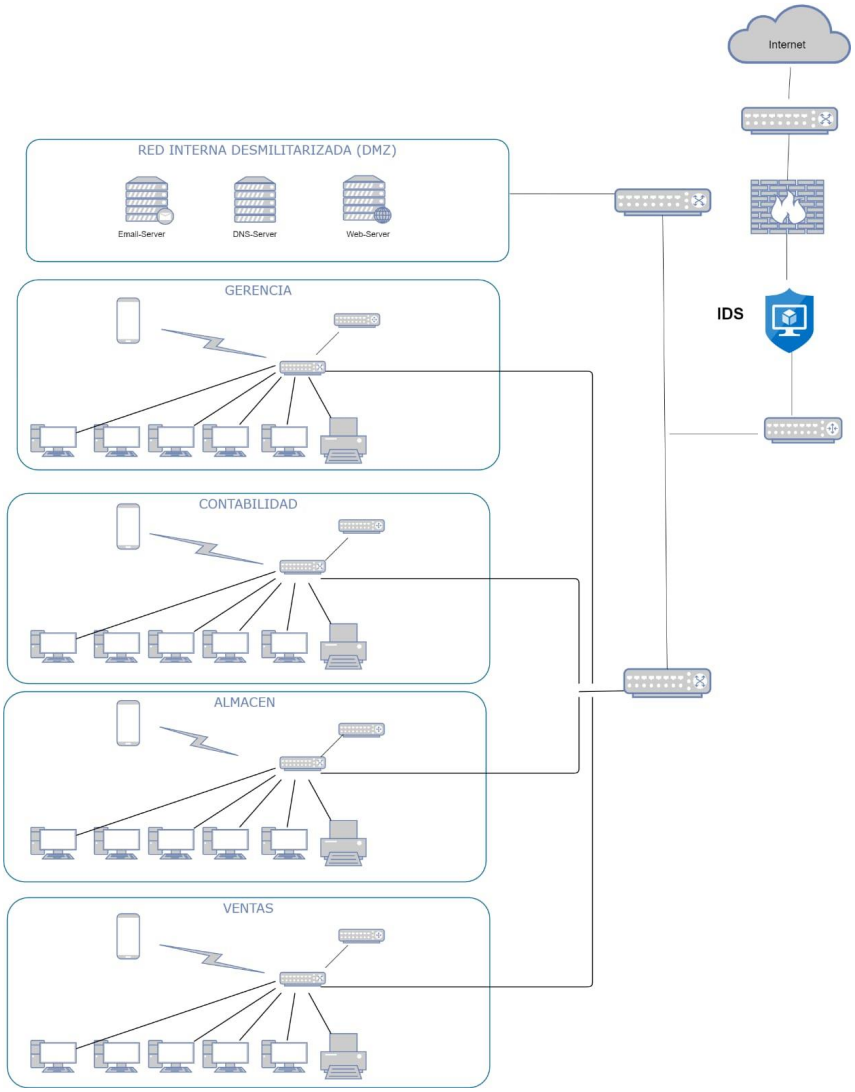
Nota. La figura representa la capa de núcleo para la empresa Dytelsur, esta capa es elemental ya que interrelaciona los elementos de la "capa de distribución" y la capa núcleo. Elaboración propia.

4.3. UBICACIÓN DEL SISTEMA DE DETECCIÓN DE INTRUSOS EN LA ESTRUCTURA DE RED

En este punto, se muestra en la figura 43 la estructura jerárquica de red LAN, basada en un Sistema de Detección a nivel de red para empresas corporativas. De acuerdo al capítulo II en la sección 2.2.8.4, se optó por colocar el Sistema de Detección de Intrusiones (IDS) detrás del firewall, debido a que desde este punto estratégico para monitorizar las amenazas externas que lograron atravesar el firewall.

Figura 45

Estructura de red basada en "sistema de detección de intrusiones" a nivel de red.



Nota. La figura representa la ubicación del Sistema de Detección de Intrusiones (IDS) detrás del firewall por ser un punto estratégico dentro de la red. Elaboración propia.

4.4. CONFIGURACIÓN DEL SISTEMA DE INTRUSOS (IDS) SURICATA

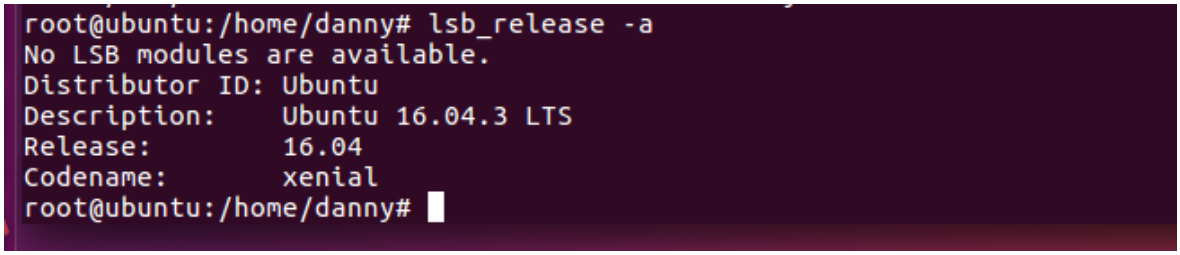
Tanto la Instalación como la Configuración de Suricata, están realizadas en el sistema operativo Ubuntu basado en Debian, versión 16.4.

4.4.1 INSTALACIÓN DE LOS REQUISITOS PREVIOS DE UBUNTU

Para ver la versión de Ubuntu que tenemos ejecutamos el siguiente comando:

Figura 46

Verificación de la versión de mi Ubuntu.



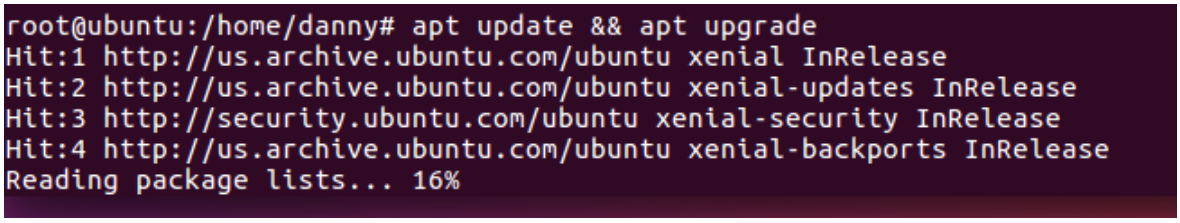
Nota. La figura representa el comando **lsb_release -a**. que se utiliza para ver la versión de mi sistema operativo Ubuntu. Elaboración propia.

Antes de configurar Suricata en el sistema, ejecutamos el siguiente comando para asegurarnos que tenemos todo lo necesario para la instalación como las ultimas actualizaciones del sistema operativo. Seguidamente se reinicia para que se apliquen todas las actualizaciones.

```
sudo apt update && sudo apt upgrade -y
```

Figura 47

Actualización de mi sistema operativo Ubuntu.



Nota. La figura representa la aplicación del comando **apt update && apt upgrade** que se utilizan previa a la instalación de Suricata para actualizar los paquetes disponibles del sistema operativo Ubuntu. Elaboración propia.

4.4.2. INSTALACIÓN Y CONFIGURACIÓN DE SURICATA

4.4.2.1. INSTALACION DE DEPENDENCIAS REQUERIDAS

Para una adecuada compilación de Suricata desde el origen debemos instalar algunas dependencias necesarias para el funcionamiento de Suricata.

Tabla 05

Dependencias para la instalación de Suricata en Ubuntu.

PCRE	(libpcre3-dev)	Disponible desde el repositorio de Linux
PCAP	(libcap3-dev)	Disponible desde el repositorio de Linux
Libdnet	(libdumbnet-dev)	Disponible desde el repositorio de Linux
Libnetfilter	(libnetfilter-queue-dev)	Disponible desde el repositorio de Linux

Nota. La tabla representa las dependencias que se tiene que instalar en el sistema operativo Ubuntu previa a la instalación de Suricata Elaboración propia.

Ahora ejecutamos los comandos para descargar suricata:

```
wget https://www.openinfosecfoundation.org/download/suricata-4.0.5.tar.gz
tar -xvzf suricata-4.0.5.tar.gz
```

Figura 48

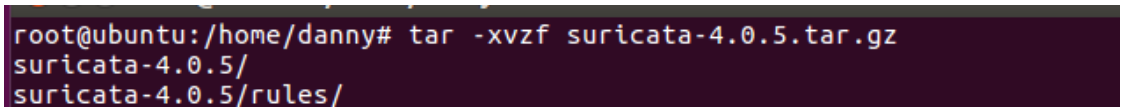
Descargando el IDS Suricata.

Nota. La figura representa la ubicación dentro del internet de Suricata 4.0.5 y el comando **wget** que se utiliza para su descarga. Elaboración propia.

Luego descomprimimos el archivo con el comando ***tar -xvzf***

Figura 49

Descomprimiendo el archivo suricata-4.0.5.tar.tz del IDS Suricata.

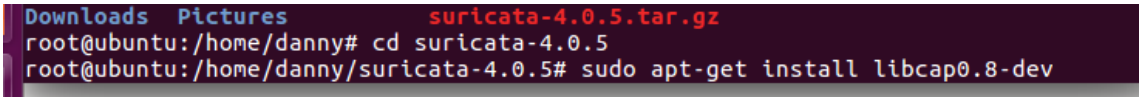


Nota. La figura representa el uso de comando **tar -xvzf** para descomprimir el archivo descargado **suricata-4.0.5.tar.gz** dentro de nuestro sistema**.** Elaboración propia.

Y una vez descomprimido nos ubicamos en el directorio para instalar las dependencias.

Figura 50

Localizando la ubicación de Suricata para la configuración e instalación de dependencias.



Nota. La figura representa el uso de comando **cd suricata-4.0.5** para la localización **dentro** de nuestro sistema del Suricata y desde ahí la ejecución para la instalación de sus dependencias. Elaboración propia.

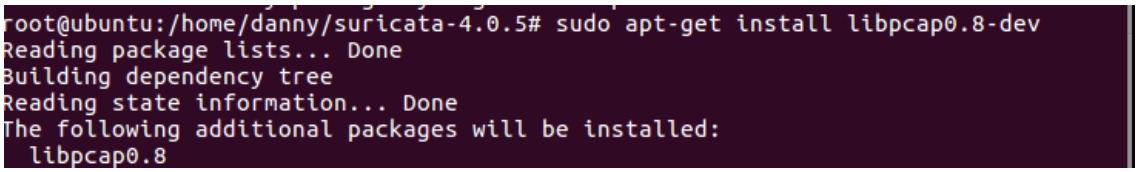
Ahora procedemos a instalar las dependencias de suricata para su funcionamiento adecuado:

```
sudo apt-get install libcap0.8-dev
```

Esta dependencia nos permitirá capturar el tráfico de datos de la red.

Figura 51

Instalando la dependencia PCAP.



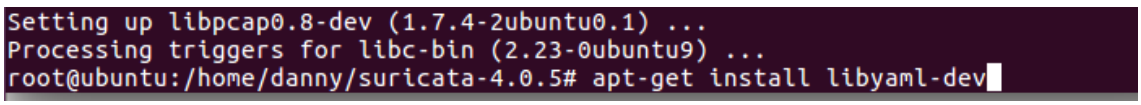
Nota. La figura representa instalación de la dependencia PCAP que permitirá la captura de tráfico de red utilizando Suricata. Elaboración propia.

Luego instalamos el analizador y biblioteca YAML

```
sudo apt-get install libyaml-dev
```

Figura 52

Instalando el analizador y biblioteca YAML.



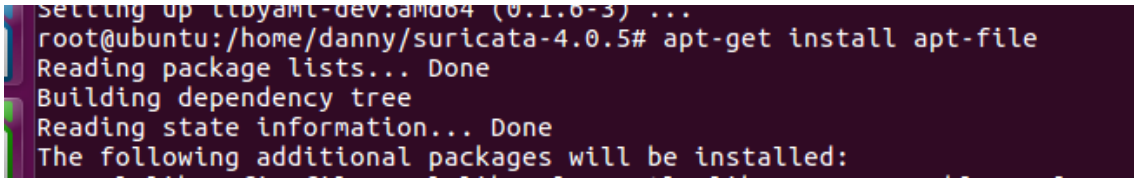
Nota. La figura representa la instalación del analizador y biblioteca YAML en donde podremos hacer el llamado a las dependencias de Suricata. Elaboración propia.

El comando **apt-file** busca en los paquetes disponibles un archivo o archivos específicos.

```
sudo apt-get install apt-file
```

Figura 53

Instalando el buscador de archivos específicos.



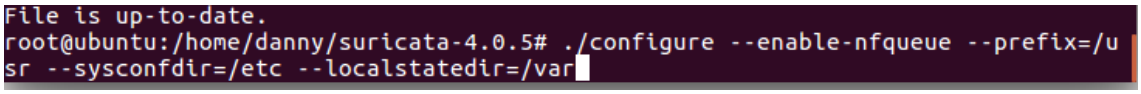
Nota. La figura representa la instalación del buscador de archivos específicos dentro de Suricata para su uso cuando se requiera. Elaboración propia.

Especificamos el directorio del archivo extraído.

```
“./configure --enable-nfqueue --prefix=/usr --sysconfdir=/etc --localst
atedir=/var”
```

Figura 54

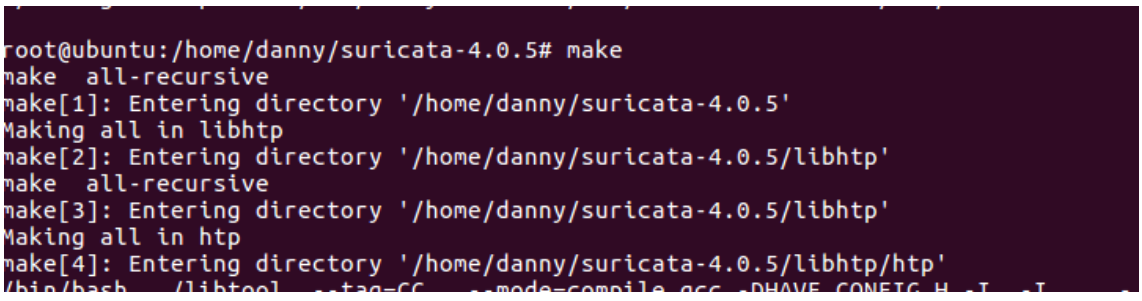
Especificando el directorio del archivo extraído de suricata.



Nota. La figura representa el directorio de Suricata para su funcionamiento desde esa dirección. Elaboración propia.

Figura 55

Instalando últimas versiones de herramientas y dependencias requeridas.



Instalamos el **make install-config** para crear un archivo llamado makefile y el makefile constituye la base de compilación.

Figura 56

Instalando el makefile para la base de nuestra compilación.

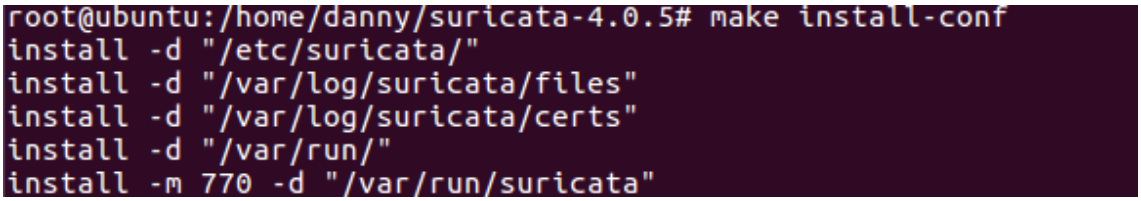
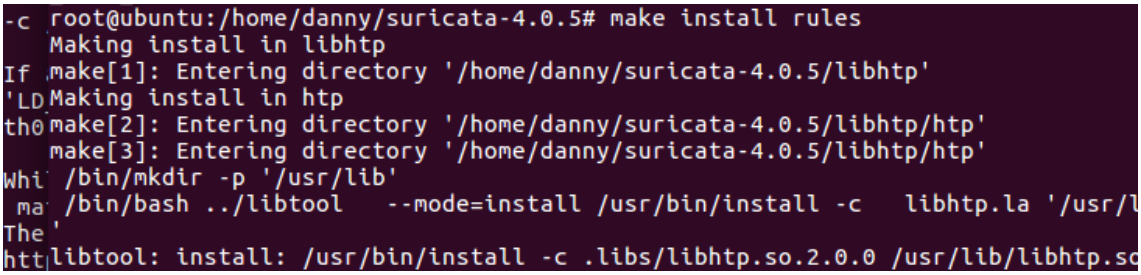


Figura 57

Instalando las reglas de suricata.

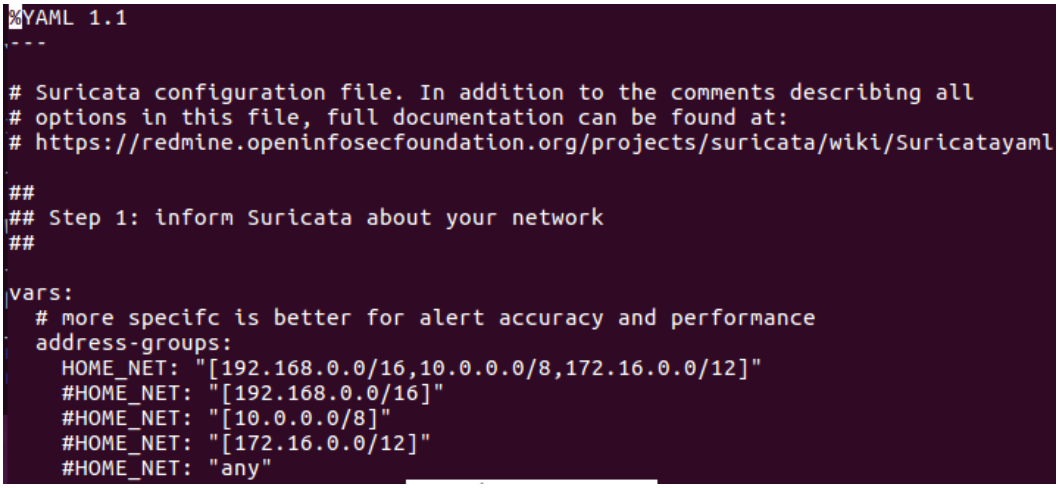


Y listo ya está instalado suricata y abrimos el archivo principal con el código:

```
“nano /etc/suricata/suricata.yaml”
```

Figura 58

Abriendo el archivo principal suricata. yaml.



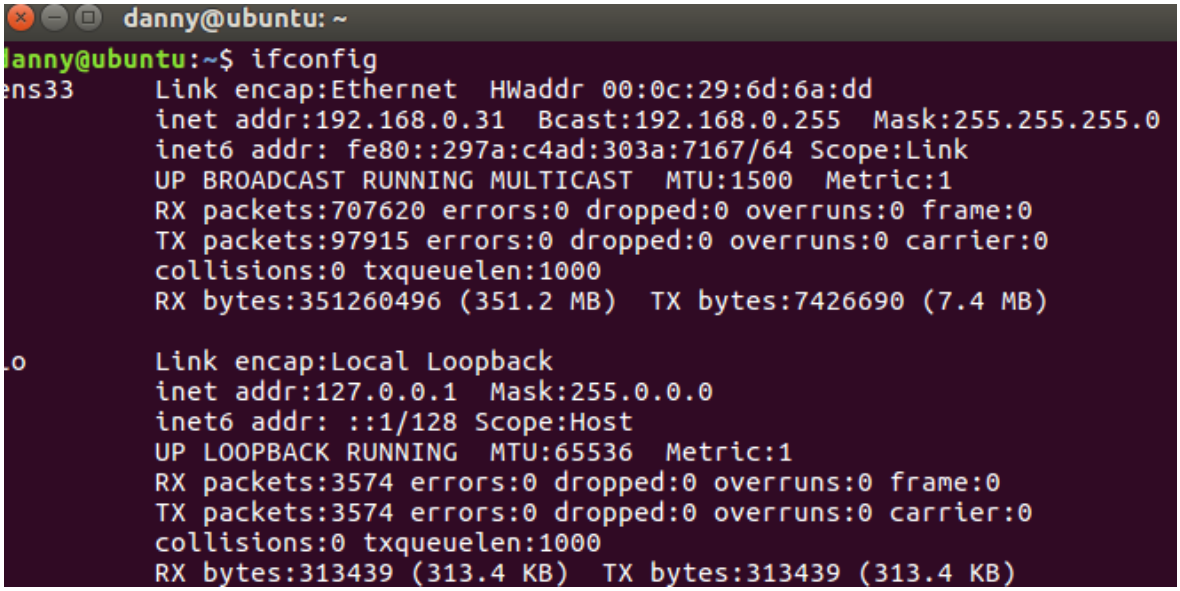
Nota. La figura representa el archivo principal **suricata. yaml** en donde podremos hacer nuestra configuración de red y reglas. Elaboración propia.

4.4.2.2. CONFIGURACION DE LA RED

Verificamos con el comando **ifconfig** el nombre de la red, en este caso el nombre es **ens33**.Vamos a configurar nuestra red.

Figura 59

Verificando el nombre de nuestra red e IP asignada por el sistema.

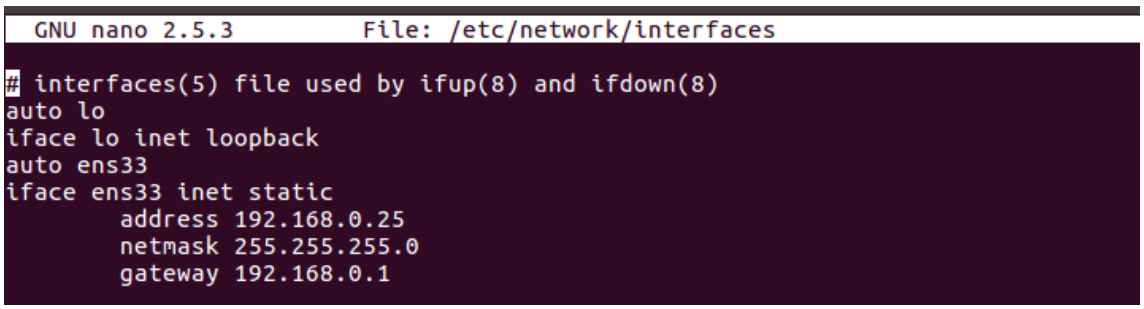


Nota. La figura representa nuestra configuración de red y la **IP** que tenemos asignada inicialmente **192.168.0.31** dentro de nuestra red. Elaboración propia.

Cambiaremos las siguientes líneas para cumplir con su entorno: HOME_NET debe coincidir con la red interna. En este caso es HOME_NET es 192.168.0.25/24 con una máscara de subred de 24 bits 255.255.255.0, cambiamos **EXTERNAL_NET a $HOME_NET**.

Figura 60

Creando nuestra red propia interna 192.168.0.25/24.

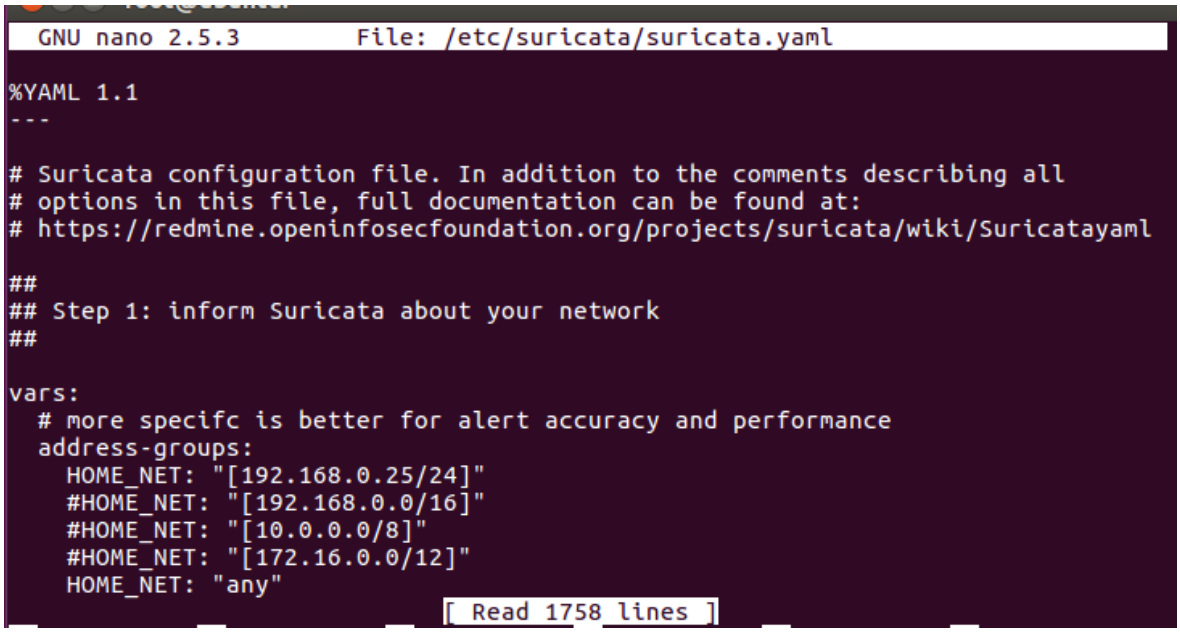


Nota. La figura representa la configuración dentro de **/etc/network/interfaces** de nuestra propia red y la **IP** que tenemos asignado **192.168.0.25/24**. Elaboración propia.

Ahora procederemos a ingresar al archivo principal de Suricata que es *suricata. yaml* y cambiaremos por nuestra red interna creada.

Figura 61

Configurando nuestra red propia interna 192.168.0.25/24.



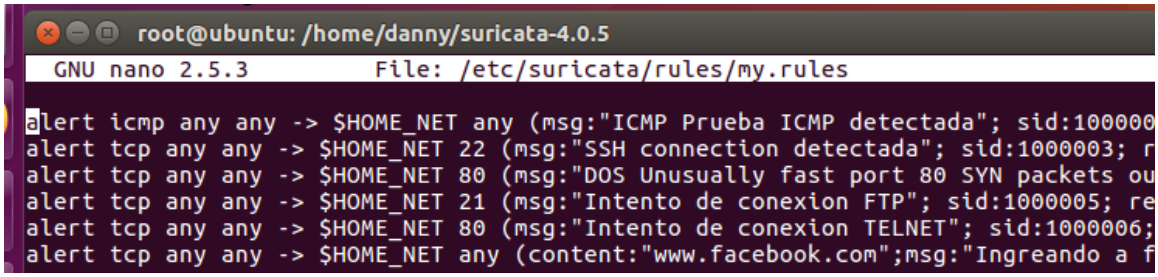
Nota. La figura representa la configuración de nuestra propia red dentro del archivo principal **suricata. yaml**. Elaboración propia.

4.4.2.3. CONFIGURACION DE LAS REGLAS EN SURICATA Y TEST

De acuerdo al capítulo II en la sección 2.2.9.2, de donde podemos decir que, la variable HOME_NET define las direcciones de red a monitorear. Asimismo, la variable EXTERNAL_NET define qué hosts externos monitorear. El valor predeterminado de any permite a monitorear todas las direcciones de red local. En esta etapa, se define las reglas en el motor de detección. Estas reglas serán colocadas en el directorio de **my.rules.**

Figura 62

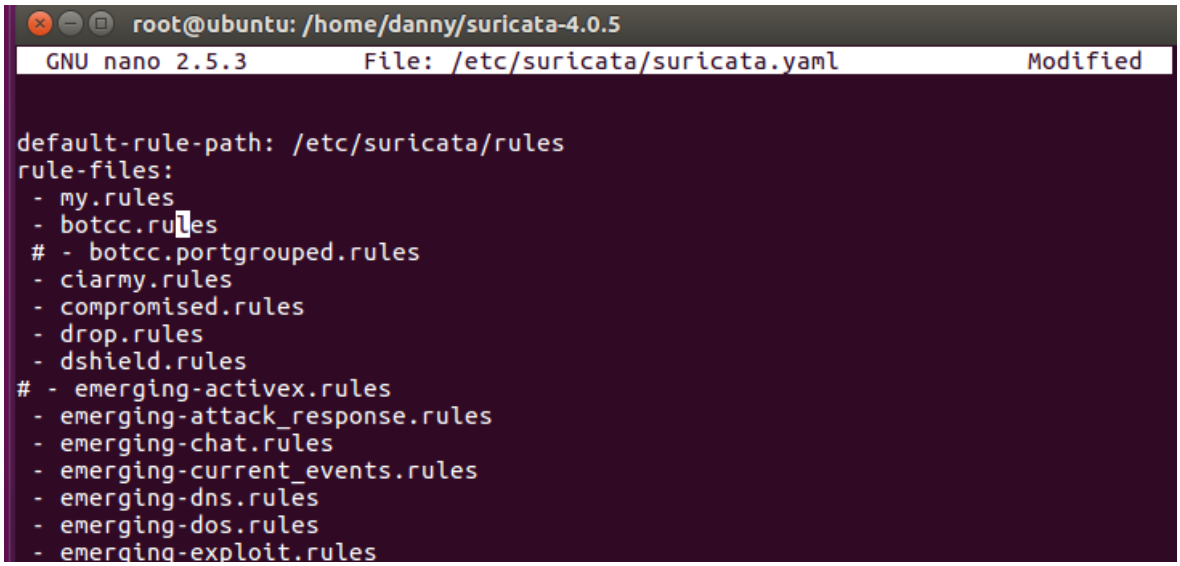
Creando nuestras propias reglas con el nombre de my.rules.



Luego de ello copiamos nuestra regla creada en el archivo de configuración de suricata. Yaml para que pueda funcionar.

Figura 63

*Copiando **my.rules** en el archivo principal de suricata para su funcionamiento.*



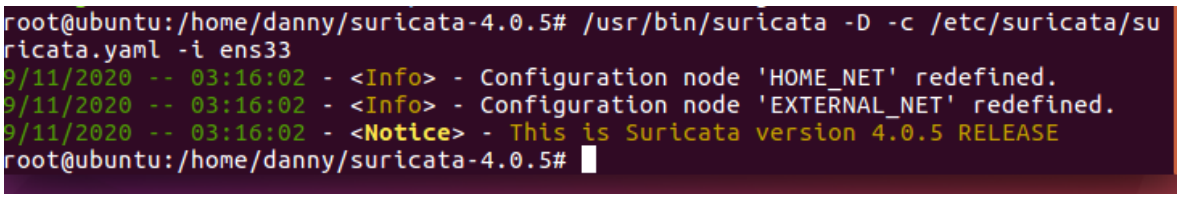
Nota. La figura representa la configuración de nuestra propia regla creada que es **my.rules** dentro del archivo principal **suricata. Yaml** para su ejecución. Elaboración propia.

Y finalmente hacemos correr el suricata, antes de ello reiniciamos el equipo para efectuar todos los cambios.

```
“/usr/bin/suricata -D -c /etc/suricata/suricata.yaml -i ens33”
```

Figura 64

Iniciando el IDS suricata con toda la configuración preestablecida.



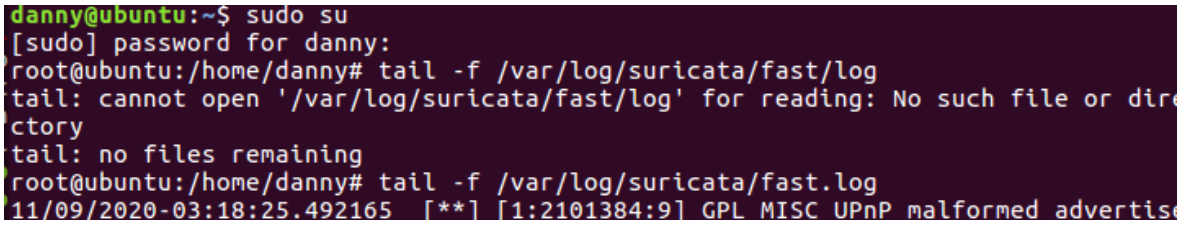
Nota. La figura representa la ejecución del IDS suricata con nuestras reglas también predefinidas. Elaboración propia.

Luego de ello ejecutamos el archivo log para ver que las reglas estén corriendo.

```
tail -f /var/log/suricata/fast.log
```

Figura 65

Iniciando el fast.log para ver todas las reglas my.rules corriendo.



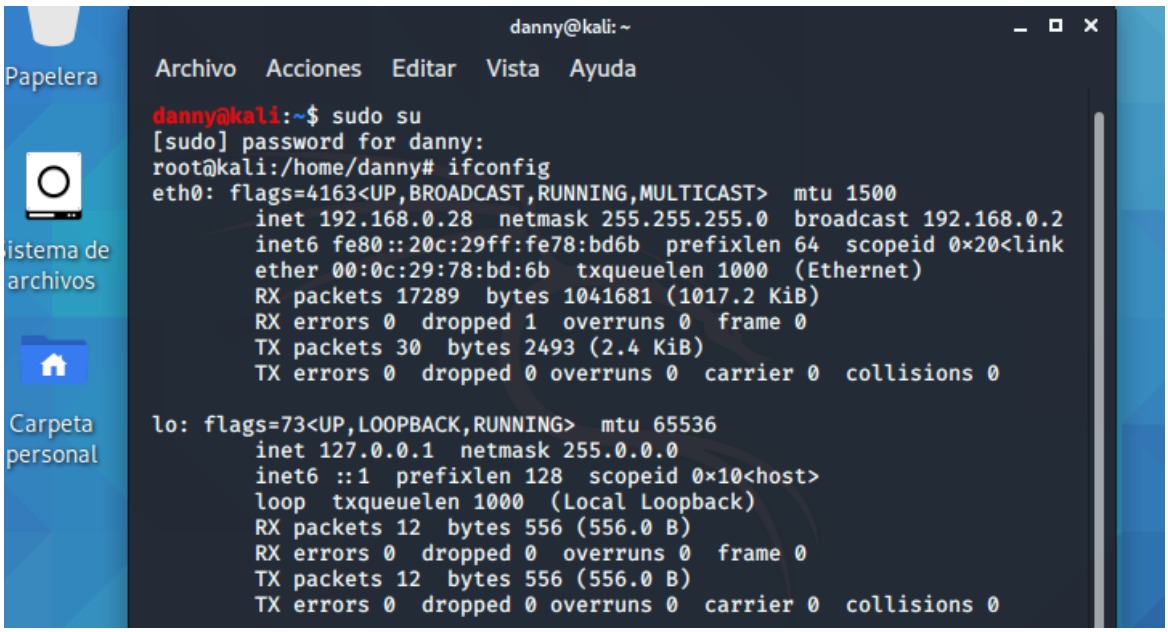
1. ICMP detectada

Esta regla genera una alerta, con un comportamiento sospechoso en el protocolo ICMP desde cualquier dirección IP de origen y puerto. Teniendo como dirección de IP destino $HOME_NET ingresando por cualquiera de sus puertos. El mensaje que emite la alerta es “Prueba ICMP detectada”.

```
Alert icmp any any -> $ HOME_NET any (msg:”Prueba ICMP detectada”; sid:
1000001;rev:1;
```

Figura 66

Viendo la IP de la supuesta máquina que va realizar los ataques, que es 192.168.0.28



Nota. La figura representa la configuración de red que tiene la máquina del atacante en este caso es 192.168.0.28 y usa Kali Linux como sistema operativo. Elaboración propia.

En la figura 67 nos indica desde que dirección IP se realizó el ping, fecha y hora de la ejecución de vulnerabilidad ICMP.

Figura 67

Realizando el ping al 192.168.0.25 de nuestra máquina Kali Linux

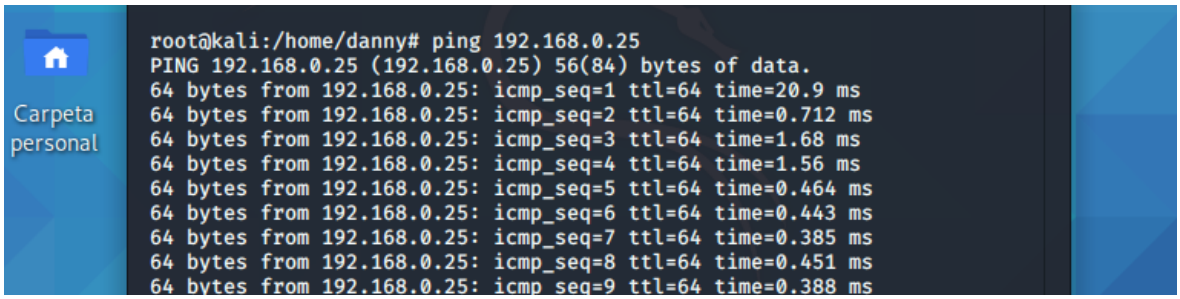
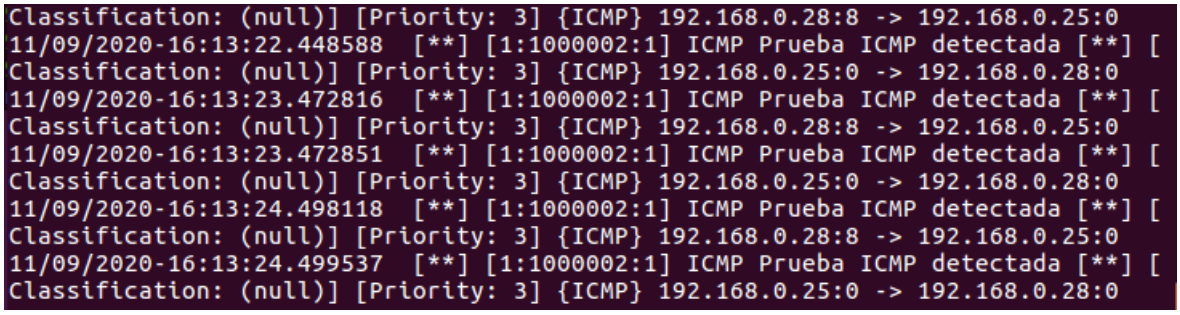


Figura 68

Mensaje de alerta por medio de la regla preestablecida de ICMP en suricata.



Nota. La figura representa el ICMP detectado dentro de la ejecución de nuestro suricata en el sistema operativo Ubuntu, donde se observa la dirección IP del atacante. Elaboración propia.

Verificamos si al realizar ping desde cualquiera de los hosts, hacia cualquiera de los puertos nos emite una alerta detectada.

2. SSH detectada

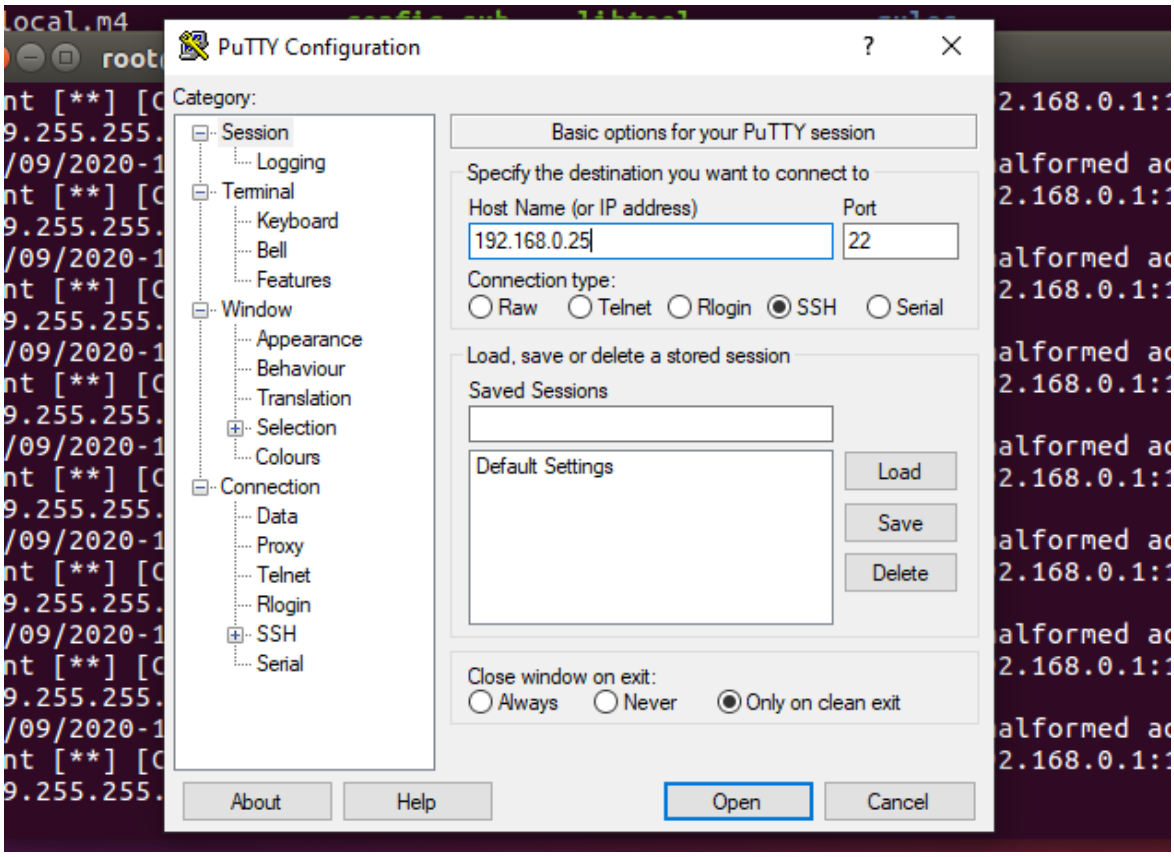
La regla genera una alerta de una conexión no permitida de SSH, esto hace que las personas se conecten a una computadora local y remota. Esta regla tiene un comportamiento sospechoso en el protocolo TCP desde cualquier dirección IP de origen y puerto. Tiene como dirección de IP destino 192.168.0.25 y puerto de destino 22. El mensaje que emite la alerta es “SSH connection detectada”.

```
alerta tcp any any -> $ HOME_NET 22 (msg:"SSHH connectionn detectada";
sid:1000002;rev:1;
```

Para eso abrimos el PuTTY que es un programa que nos permite realizar conexiones SSH

Figura 69

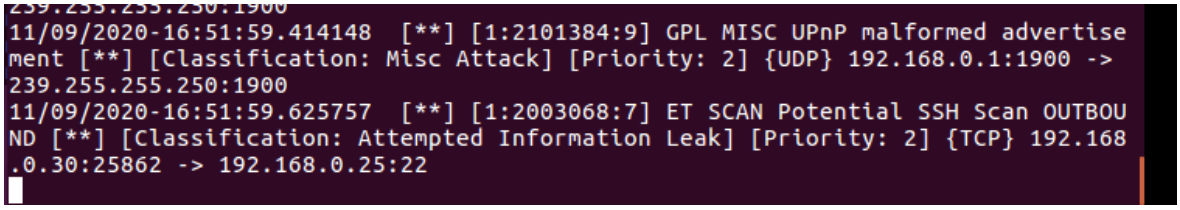
Realizando la prueba SSH desde el programa PuTTY al 192.168.0.25.



Nota. La figura representa la conexión SSH que se quiere realizar a nuestra red, el cual debe detectar nuestro IDS Suricata. Elaboración propia.

Figura 70

Mensaje de alerta ante posible conexión SSH en nuestro IDS suricata.



Nota. La figura representa la "Alerta de conexión SSH" en nuestro IDS Suricata por la regla preestablecida. Elaboración propia.

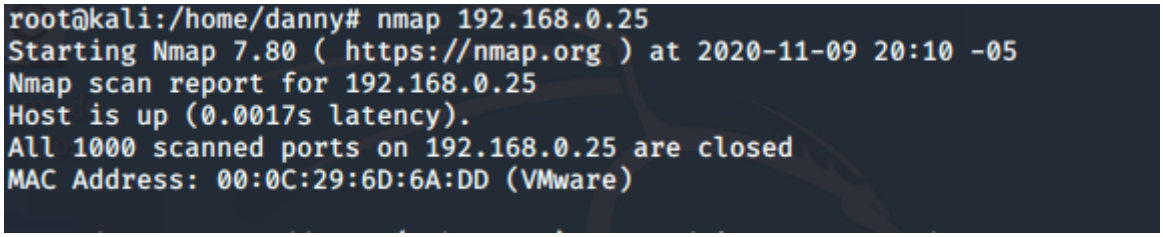
3. Escaneo de puertos TCP

La regla genera una alerta, con un comportamiento sospechoso en el protocolo TCP desde cualquier dirección IP de origen y puerto. Teniendo como dirección de IP destino $HOME_NET, ingresando por cualquiera de los puertos. El mensaje que emite la alerta es “Escaneo de puertos TCP detectada”.

```
alert tcp any any -> $ HOME_NET any (msg:” Escaneo de puertos TCP detec
tato”; sid:1000003; rev:1;
```

Figura 71

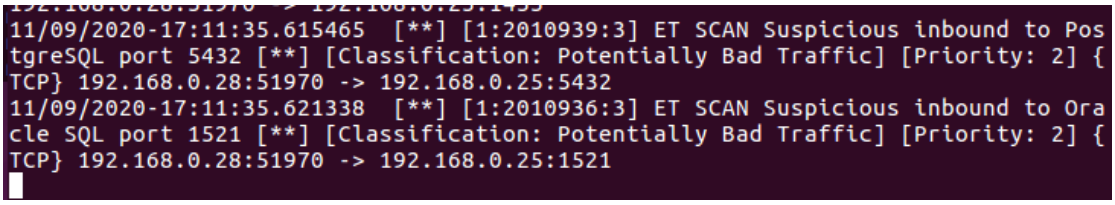
Realización de ataque por medio de Kali linux.



Nota. La figura representa el ataque mediante el uso de **nmap** a nuestra red. Elaboración propia.

Figura 72

Mensaje de alerta en el IDS suricata.



Nota. La figura representa el msj de alerta por la regla preestablecida dentro de nuestro IDS suricata. Elaboración propia.

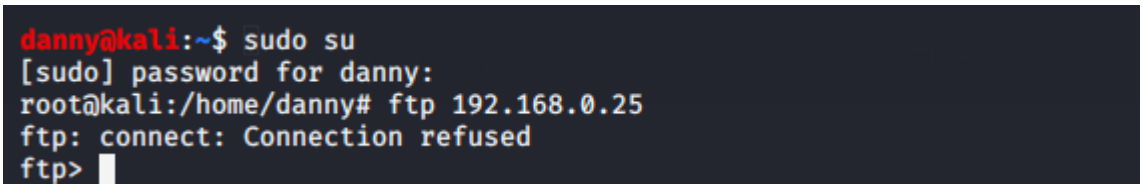
4. Intento de conexión FTP

La regla genera una alerta, con un comportamiento sospechoso en el protocolo TCP desde cualquier “dirección IP de origen y puerto”. Tiene como dirección de IP destino $HOME_NET y puerto de destino 21. El mensaje que emite la alerta es “Intento de conexión FTP”.

```
alert tcp any any -> $ HOME_NET any (msg:” Escaneo de puertos TCP detec
tato”; sid:1000003;rev:1;
```

Figura 73

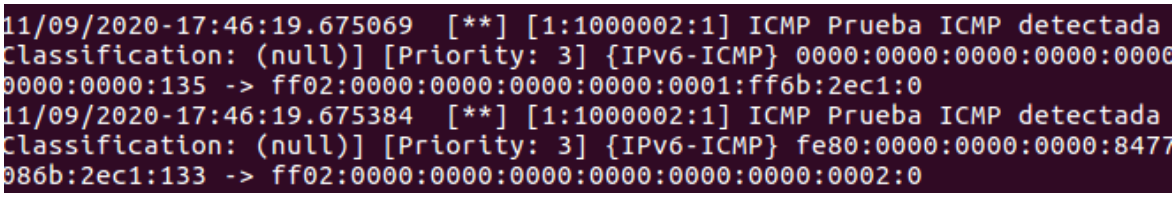
Realización de ataque ftp.



Nota. La figura representa el ataque mediante el intento de conexión **ftp** a nuestra red. Elaboración propia.

Figura 74

Mensaje de alerta en el IDS suricata.



Nota. La figura representa el mensaje de alerta de conexión **ftp** a nuestra red. Elaboración propia.

2. OTRAS REGLA ALTERNAS

a. Alerta de ingreso a Facebook

La regla genera una alerta de una conexión a Facebook, esta regla tiene un comportamiento sospechoso en el protocolo TCP desde cualquier "dirección IP de origen y puerto". Tiene como dirección de IP destino $HOME_NET y puerto de destino any. El mensaje que emite la alerta es "Alguien se encuentra ingresando a Facebook".

```
alert tcp any any -> $HOME_NET any (content: "www.facebook.com"; msg:" Alguien se encuentra ingresando a Facebook"; sid:10000007; rev:001;)
```

b. Alerta de ingreso a Youtube

La regla genera una alerta de una conexión a YouTube, esta regla tiene un comportamiento sospechoso en el protocolo TCP desde cualquier dirección de una IP de origen y puerto. Tiene como dirección de IP destino $HOME_NET y puerto de destino any. El mensaje que emite la alerta es "Alguien se encuentra ingresando a youtube".

Alert tcp any any -> $HOME_NET any (content: “www.facebook.com”; msg:” Alguien se encuentra ingresando a Facebook”; sid:10000007; rev:001;)

c. Posibles ataques DoS TCP

La regla genera una alerta amenaza ante una restricción de servicio distribuido, con un comportamiento sospechoso en el protocolo TCP desde cualquier dirección IP de origen y puerto. Tiene como dirección de IP destino $HOME_NET y puerto de destino 80.

alert tcp any any -> $HOME_NET 80 (flags: S; msg"Posible TCP DoS;flow:stateless;";

Tabla 06

Reglas de motor de detección

PROTOCOLO	DIRECCION DE ORIGEN	DIRECCION DESTINO	PUERTO ORIGEN	PUERTO DESTINO	ALERTA
IMCP	any	HOME_NET	any	any	ICMP detectada
TCP	any	HOME_NET	any	22	SHH detectada
TCP	any	HOME_NET	any	80	Ataque DoS TCP
TCP	any	HOME_NET	any	21	Conexión FTP
TCP	any	HOME_NET	any	any	Ingreso Facebook
TCP	any	HOME_NET	any	any	Ingreso youtube

Nota: La table representa las reglas configuradas en nuestro Suricata en base a los puertos mencionados. Elaboración propia.

CAPITULO V
CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

En base a todo lo realizado en el Capítulo IV, a lo largo de toda la investigación podemos afirmar que se logró implementar, configurar y mostrar la utilidad del IDS suricata en una infraestructura corporativa a nivel de red, donde se realizaron las pruebas respectivas necesaria para verificar las alertas de seguridad del IDS suricata.

De acuerdo al numeral 4.4.2.3, del Capítulo IV, se logró realizar la implementación de Suricata con palabras claves de protocolos, para que al momento en que circula un paquete por la red, este es capturado y analizado por el IDS Suricata por el protocolo mencionado.

De acuerdo al numeral 4.4.2.3, del Capítulo IV, se logró implementar el perfilado de reglas en el motor de reglas de Suricata, y se realizó el filtrado de los eventos y vulnerabilidades, donde estas reglas alertan los posibles intentos ataques como ICMP, SSH, FTP, TCP. Se logró verificar que dichas reglas implementadas cumplen con su propósito puesto que se realizaron las pruebas necesarias para garantizar que la misma cuenta con la funcionalidad necesaria y su funcionamiento es el esperado.

De acuerdo al numeral 4.4.2.3, del Capítulo IV, se logró hacer la comparación de patrones de trafico de red en donde se vio los tráficos más frecuentes y posibles tráficos de amenazas para establecer reglas en el motor de suricata para su protección.

De acuerdo a lo mencionado en el numeral 4.3, del Capítulo IV, se logró implementar el diseño de estructuración de red basada en Sistema de Detección de Intrusos (IDS), donde se identificó el lugar más ideal o adecuado para la ubicación del IDS desde se logró identificar los ataques por medio de tráfico de red a la empresa corporativa para de esta manera monitorear la seguridad para estar alertas ante ataques cibernéticos.

5.2 RECOMENDACIONES

Se recomienda implementar un motor de reglas actualizados ya que cada día surgen nuevas formas de ataques, el Sistema de Detección de Instrusos (IDS) Suricata, detecta anomalías en base al motor de reglas que uno implementa, si no existe esa regla que detecte una anomalía simplemente nuestra infraestructura estará expuesta al ataque cibernético.

Se recomienda utilizar la propuesta planteada sobre la ubicación del Sistema de "Detección de Intrusos (IDS)" Suricata en una infraestructura de red "open source". Si la infraestructura de red cuenta con otro dispositivo de filtrado como el firewall, la ubicación idónea es colocar el Sistema de Detección de Intrusiones Suricata detrás del firewall obteniendo mayor seguridad por el primer filtro. Sin embargo, si no tuviese ningún otro dispositivo de seguridad o filtrado se coloca detrás del dispositivo router

Tener cuidado al momento de la instalación específicamente en las versiones del sistema operativo Linux y Suricata, porque en algunos casos no es necesario instalar algunos complementos ya que lo tienen por defecto, sin embargo, en otras versiones es necesario la instalación de los prerrequisitos en el sistema operativo. Tener en cuenta que esta investigación se realizó en Kali Linux 20.02, Ubuntu 16,04 y Suricata 4.0.5.

Es necesario identificar los procedimientos antes de realizar alguna actualización del database del "Sistema de Detección de Intrusos" (IDS), dado que servirán como guía para acciones futuras de la misma índole. Asimismo, si la implementación del Sistema de Detección de Intrusos (IDS) se realiza por primera vez se recomienda capacitar al personal encargado de monitorizar o administrar el Sistema de Detección de Intrusiones. Se recomienda mantener las reglas actualizadas, debido a que podrían surgir nuevos tipos de ataques que existen en las redes.

Se recomienda implementar reglas actualizadas para realizar pruebas a las reglas antes de implementarlas puesto que en ocasiones generan bastantes archivos de amenazas que no son amenazas verdaderas, esto debido a que se suele saturar debido a que cada paquete debe ser comparado con cada requerimiento existente, lo que ocasiona que se requiera cada vez más capacidad de procesamiento de maquina en donde se ejecute el "Sistema de Detección de Intrusiones" Suricata.

En el ámbito de Suricata como Sistema de detección de Intrusos se recomienda la mejora continua de Suricata, donde se debería habilitar la comunicación entre un motor de detección y otro para que de esta manera el conocimiento obtenido por los motores de detección se comparta entre todos estos.

Se recomienda fomentar el uso de herramientas Open Source en la medianas y pequeñas corporaciones que no cuentan con capital para la inversión en la implementación de en un Sistema de Detección de intrusiones (IDS) a software. Frente a esto encontramos a Suricata Open Source que es beneficioso a nivel económico, además, al tener la característica de código libre hace que el desarrollo sea rápido y cubre de manera eficiente dicha necesidad de seguridad en las Pymes.

Se recomienda instalar las versiones recientes de Suricata ya que debido a sus actualizaciones la nueva versión es más eficiente, proporciona un mayor rendimiento, escalabilidad, usabilidad y permite una gran extensibilidad, pero tener en cuenta que dichas actualizaciones dependen de varias dependencias que se tendrá que instalar con sumo cuidado para no generar errores de instalación.

Se recomienda usar Kali Linux para la identificación de instrusos en nuestra red, con ese sistema operativo podremos analizar a más a fondo sobre el intruso como saber de dónde proviene la red, desde que dispositivo, de esta manera podremos detectar al sospechoso.

BIBLIOGRAFÍA

Albin, E. (2017). *Análisis comparativo de los sistemas de detección de intrusiones de snort y suricata.* Tesis de Posgrado. Naval Postgrado Colegio. Monterrey, California.EEUU.

Ariganello, E. (2011) *Redes Cisco. Guía de estudio para la certificación CCNA 640-802.*Alfaomega Grupo Editor. 2da edición. México.

Ariganello, E. (2019) *Modelo de referencia OSI*. Recuperado el 11 de octubre del 2020 de https://aprenderedes.com/2019/05/modelo-de-referencia-osi/

Arteaga, J. E. (2019). *Evaluation of the functionalities of the intrusion detection systems based on the network of open source platforms using the anomaly detection technique*. LatinAmerican Journal of Computing (LAJC), 7(1), pp.53. Recuperado de http://dspace.espoch.edu.ec/handle/123456789/8748.

Astudillo, Ortiz, Jiménez y Aranda. (sf). *Adaptación del IDS/IPS Suricata para que se pueda convertir en una solución empresarial*. Articulo. Escuela Superior Politécnica del Litoral. Guayaquil. Ecuador.

Bace, R. y Mell, P. (2015). NIST Special Publication on Intrusion Detection System. Intrusion Detection Systems, 3, pp.1-51. Recuperado de http://www.iwar.org.uk/comsec/resources/ids/sp800-31.pdf.

Baker, A.R., Esler, J. (2005). Snort® IDS and IPS Toolkit. Burlington: SYNGRESS.

Bardales, E. (25 septiembre, 2014). Diario Gestión. *Una de cada cinco pymes es víctima de delitos cibernéticos,* según Microsoft. Recuperado de https://gestion.pe/tecnologia/cinco-pymes-victima-delitos-ciberneticos-microsoft-73780-noticia/?ref=gesr.

Bernal, A. (2010). *Metodología de la investigación.* México: Pearson Educación.

Cox, K. y Gerg, C. (2004). *Managing Security with Snort & IDS Tools*: Recuperado de.https://www.oreilly.com/library/view/managing security%20with/0596006616/pr02.html.

Carthern, C., Wilson, W., Rivera, N y Bedwell R. (2015). *Cisco Networks: Engineers' Handbook of Routing, Switching, and Security with IOS, NX-OS, and ASA.* Springer Science Business Media New York.EEUU.

Charkrabarti, S., Chakraborty, M. y Mukhopadhyay, I. (2010). *International Conference and Workshop on Emerging Trends in Technology. Study of Snort-BasedIDS.1,pp.434.https://www.researchgate.net/publication/220902217_Study_of_snort-based_IDS.*

Chávez. (2011) *Simulación y análisis de mecanismos de defensa ante los ataques de denegación de servicios (DoS) en redes de área local convergente.* Tesis de Pregrado. Escuela Politécnica Nacional. Quito. Ecuador.

Diario El Peruano (09 marzo, 2020). Diario El Peruano. Perú entre países más inseguros en línea en América Latina. Recuperado de https://elperuano.pe/noticia-peru-entre-paises-masinseguros-linea-america-latina-90806.aspx.

Easttom, C. (2014). Network Defense and Countermeasures: Principles and Practices, Second Edition. Estados Unidos: Pearson IT Certification.

E. Arriols (2017), *Curso completo de hacking ético – Udemy* earriols@redteaming.es, , UDEMY.

Farro Flores, C. (2019). *"Uno de los activos más importantes del negocio es la información".* Ciberdelincuencia: Amenaza Latente, 38, pp.11. Recuperado de https://www.bascperu.org/pdf/principales/REVISTA-38_opt.pdf.

Francois Carpentier, J. (2016). *La seguridad informática en la PYME: Situación actual y mejores prácticas.* Barcelona: Ediciones ENI.

González Barahona, J.M. (2011). El concepto de software libre: Revista Tradumática Tecnologies de la Traducción, 9, pp. 5-11. Recuperado de https://www.oreilly.com/openbook/osfreesoft/book/index.html.

Gonzales, Sánchez y Soriano (2013) *Pentesting con Kali.* Recuperado de file:///C:/Users/LARED/Downloads/0xword-Pentesting-Con-Kali-Linux_v6.pdf.

Gutiérrez, P. (2019*). Hacker's WhiteBook*. Edicion While Suit Hacking. Monterrey. México.

Hernández Sampieri, R., Fernández, C. y Baptista, P. (2014) *Metodología de la investigación* (6ta Ed.). México, D.F., México: McGraw Hill Interamericana.

Hernández, B. (2001). Técnicas estadísticas de investigación social. España: Diaz de Santos S.A.

Kaspersky (2020). Ciber Amenaza Mapa En Tiempo Real. Estadística histórica mundial. Recuperado de https://cybermap.kaspersky.com/es/stats/#country=4&type=ids&period=w.

Leacock, S. (2018) Que es un sistema de detección y prevención de intrusos (IDS). Articulo. Recuperado de https://backtrackacademy.com/articulo/que-es-un-sistemas-de-deteccion-y-prevencion-de-intrusos-ids.

Messier, R. (2019). CEH v10 Certified Ethical Hacker Study Guide. Blog. Recuperado de https://learning.oreilly.com/library/view/ceh-v10certified/9781119533191/ftoc.xhtml.

Gutiérrez, P. (2019*). Hacker's WhiteBook*. Edicion While Suit Hacking. Monterrey. México.

Mohammed M, Alani (2014). *Guide to OSI and TCP/IP Models. (4ta Edicion).* Springer Cham Heidelberg New York Dordrecht London.

Mohanta, A y Saldanha, A. (2020). *Malware Analysis and Detection Engineering: A Comprehensive Approach to Detect and Analyze Modern Malware*. Edición Science Business Media New York.EEUU.

Muñoz, J (2016) Seguridad Organizacional. Recuperado de https://www.seguritecnia.es/tecnologias-y-servicios/seguridad-organizacional_20160910.html.

Noguera, A. (2019). *Implementación de un sistema de detección de intrusos para venezolana del vidrio C.A.* Tesis de pregrado. Universidad Central de Venezuela. Caracas. Venezuela.
OPENBIZ. (11 septiembre, 2009). OPEN SOURCE [Documento informativo]. Recuperado de http://www.openbiz.com.ar/Open%20Source.pdf.

Ortiz, E. (2019) *Sistema de detección de intrusos, IDS, Intrusión detection system.* Blog Recuperado de https://pcweb.info/sistema-de-deteccion-de-intrusos-ids-intrusion-detection-system-que-es/.

Parisi, A. (2019). Hands-On Artificial Intelligence for Cybersecurity. Reino Unido: Packt Publishing.

Prakash, O. y Kumar, V. (2012). International Journal of Computer Applications & Information Technology. Signature Based Intrusion Detection System Using SNORT, Vol. I, pp. 35-41. Recuperado de https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.462.1508&rep=rep1&type=pdf.

Ramiro, R. (2018) Tipos de ataques informáticos y como prevenirlos. Artículo Recuperado de https://ciberseguridad.blog/25-tipos-de-ataques-informaticos-y-como-prevenirlos/.

Robles, M.(sf) *Virtualización de servidores con VMware.* Recuperado de https://www.usmp.edu.pe/vision2017/pdf/materiales/VIRTUALIZACION_DE_SERVIDORES_CON_VMWARE.pdf.

Santos, O y Gregg, M. (2019). *Certified Ethical Hacker (CEH)* Version 10 Cert Guide, 3rd Edition. Estados Unidos: Pearson IT Certification.

Sarria, C.(sf) *Seguridad Corporativa: El equilibrio entre aumentar las ganancias y controlar las perdidas*. Recuperado de http://www.gestiondelriesgo.com/artic/segcorp/7201.htm.

Tafto, J. (2011), *Comparación de Intrusión en redes de código abierto-Sistemas de detección*. Tesis de pregrado. Universidad de Oslo. Noruega.

Tamayo y Tamayo, M. (1997) *El Proceso de la Investigación científica*. México D.F., México: Editorial Limusa S.A.

Tomas, J. (2009). Fundamentos de bioestadística y análisis de datos para enfermería. España: Bellaterra.

Vacca, J. R. (2012). *Computer and Information Security Handbook*, 2nd Edition. Recuperado de https://learning.oreilly.com/library/view/computer-and-information/9780123943972/xhtml/Title_page.html.

Villagómez.C. (2017) *El protocolo ARP*.Recuperado de https://es.ccm.net/contents/260-el-protocolo-arp.

Yuquilema, N. (2016). *Estudio de técnicas y herramientas para la prevención y detección de intrusiones a nivel de aplicación en la red de datos de la UNACH*. Tesis de Pregrado. Universidad Nacional de Chimborazo. Riobamba. Ecuador.

ANEXO A

Una vez descargada desde la página oficial de Kali Linux, se procede a instalar nuestro sistema operativo y seleccionamos los siguientes pasos:

Figura 75

Instalación de Kali Linux

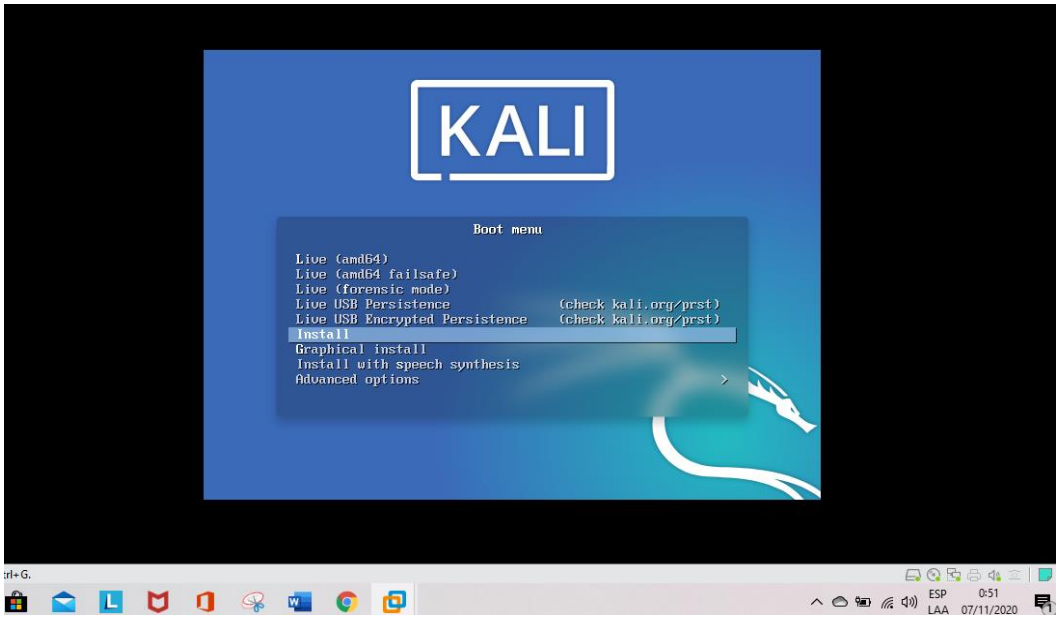


Figura 76

Configuracion del idioma del sistema Spanish

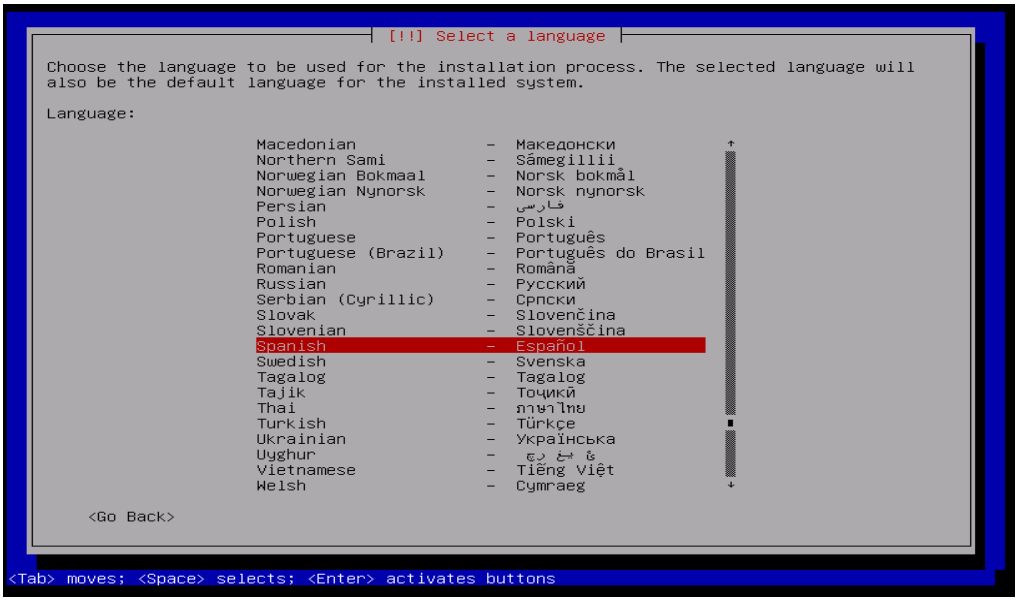


Figura 77

Configuración de la ubicación o zona horaria, Perú.

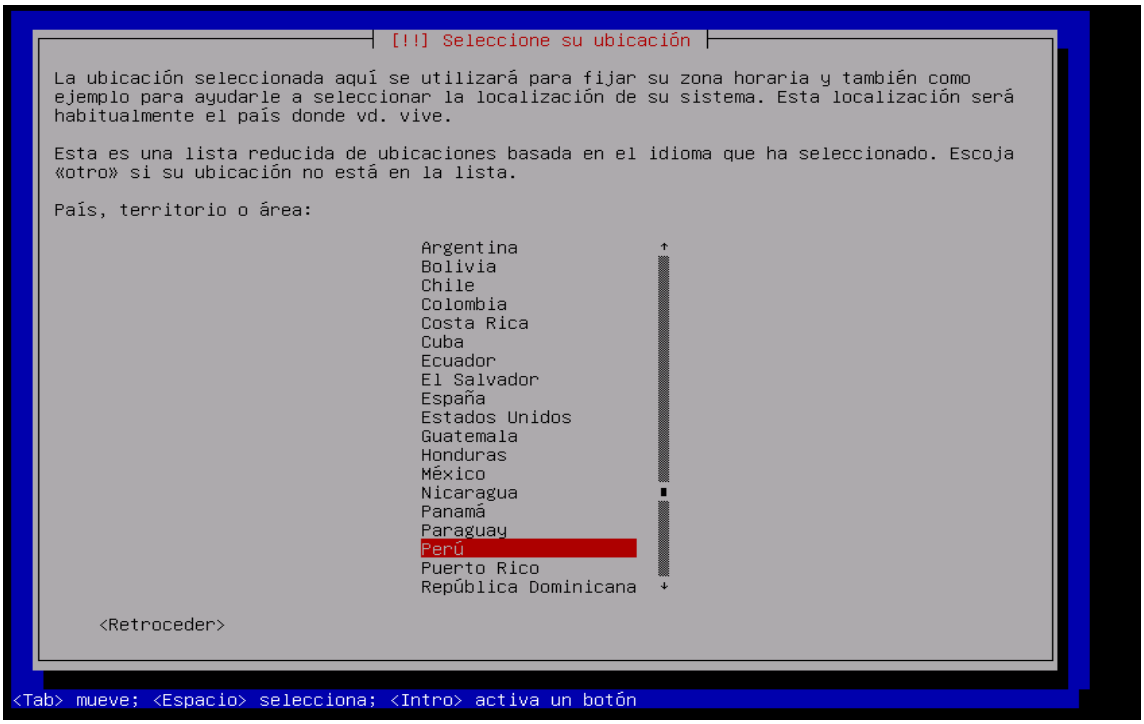


Figura 78

Configuración el método de ingreso del teclado

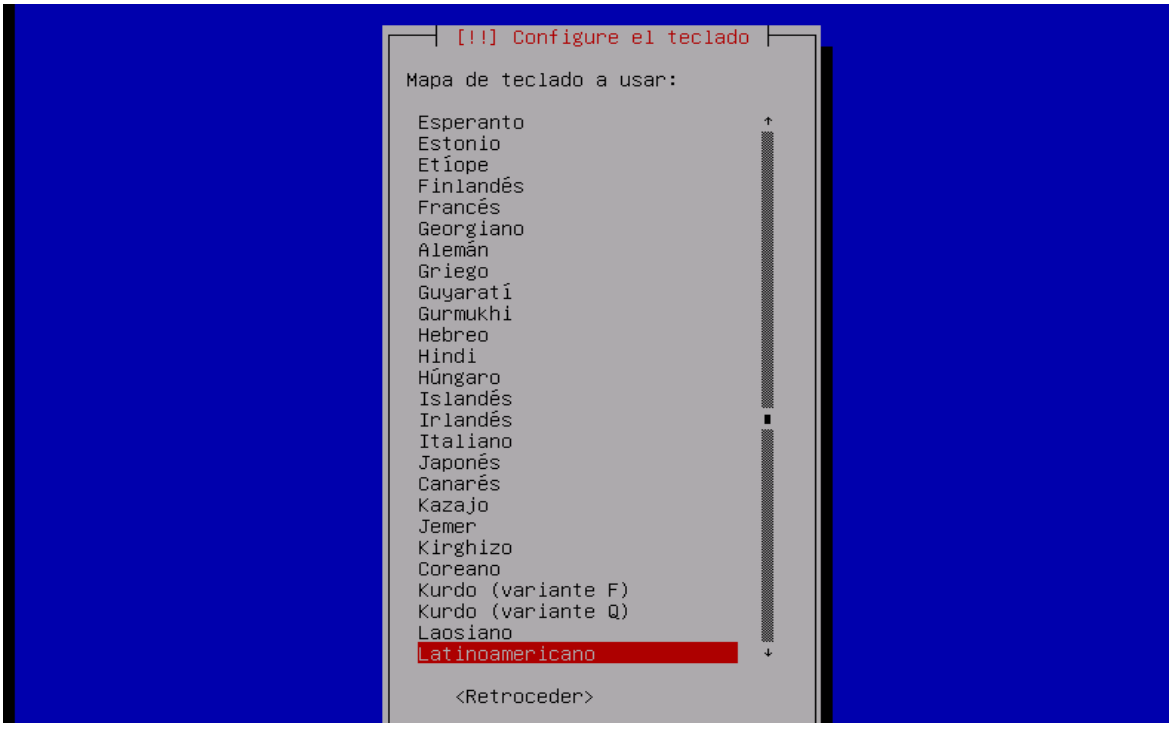


Figura 79

Configuración el nombre del equipo

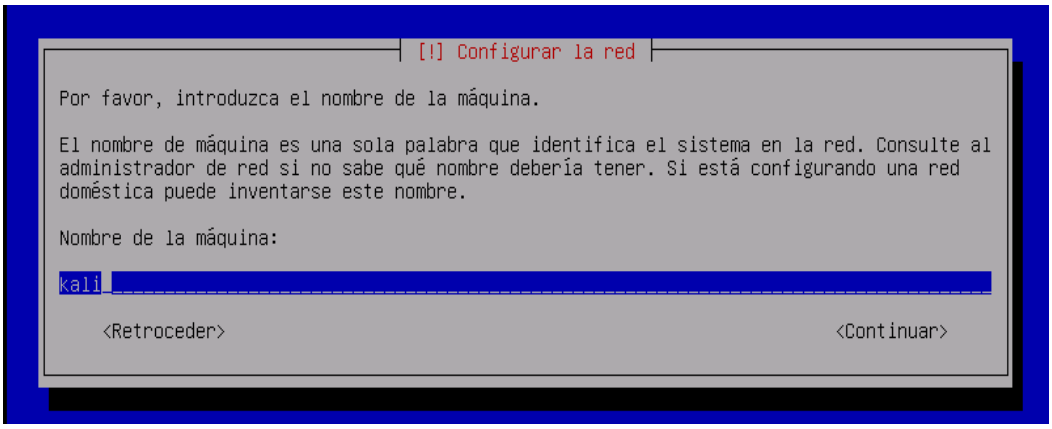


Figura 80

Configuración del nombre de dominio

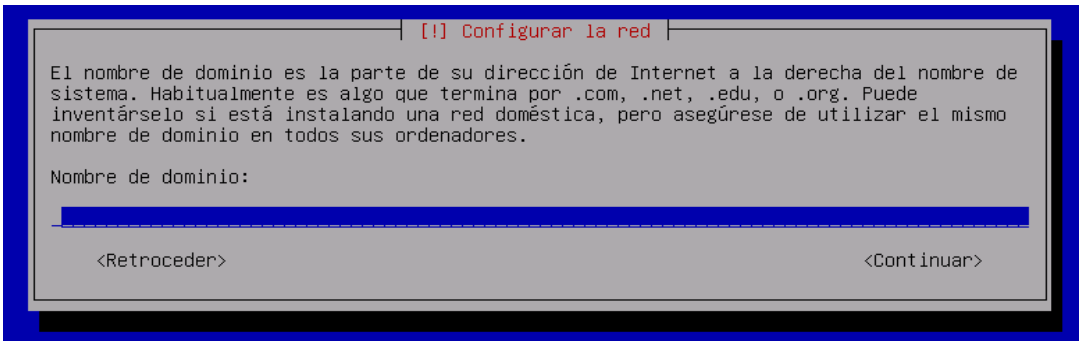


Figura 81

Configuramos los usuarios y contraseña

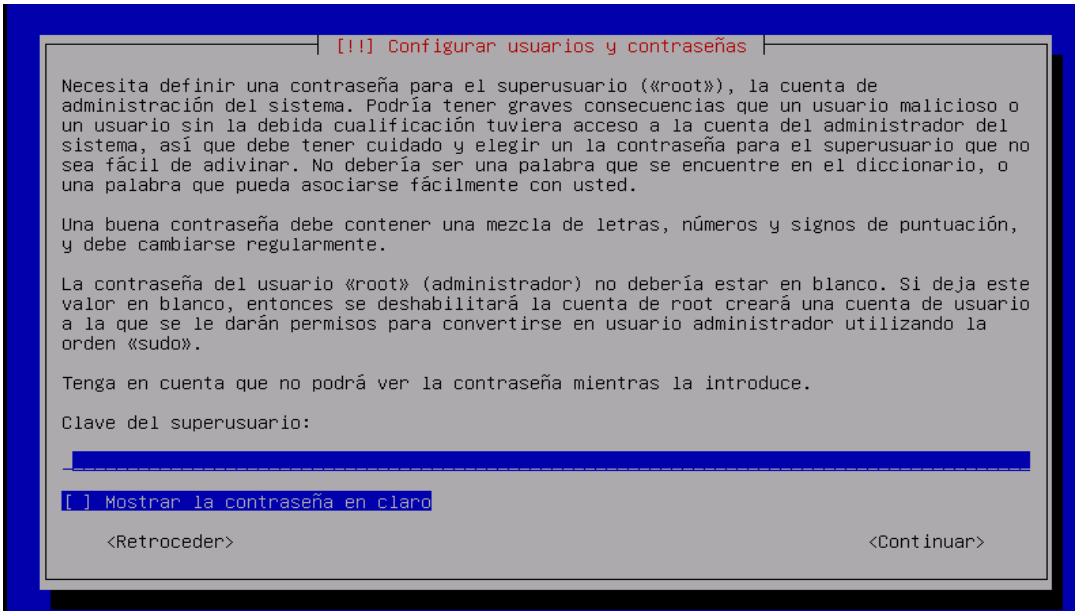


Figura 82

Configuramos la contraseña

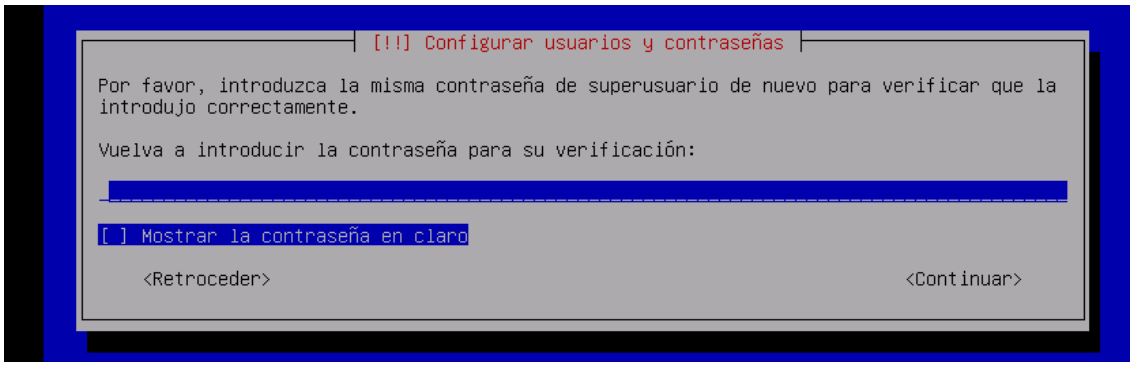


Figura 83

Seleccionamos la partición del disco

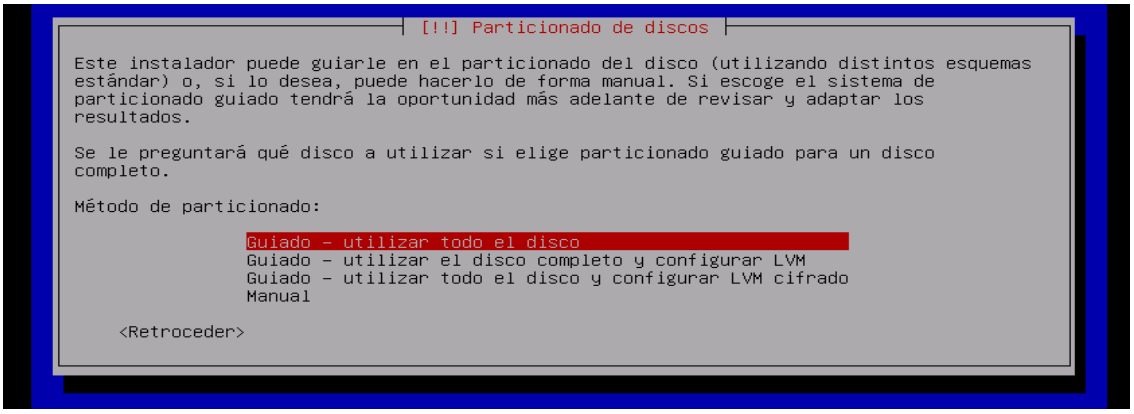


Figura 84

Elegimos el disco a particionar

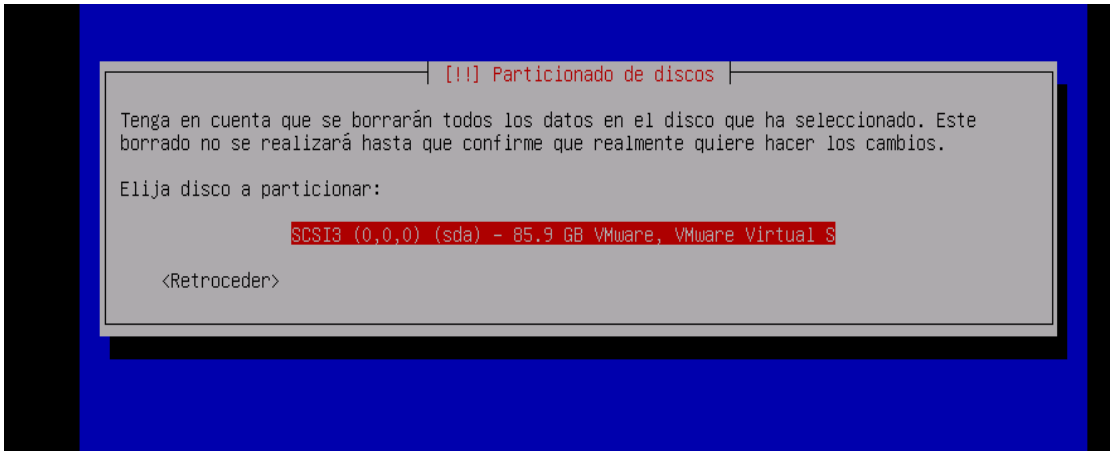


Figura 85

Elegimos el Esquema del particionado

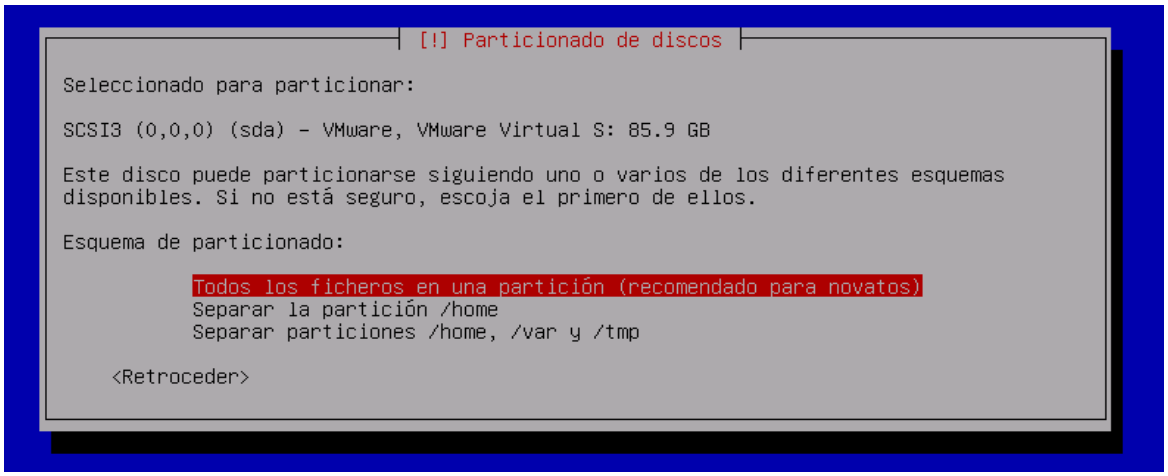


Figura 86

Finalización del particionado de disco

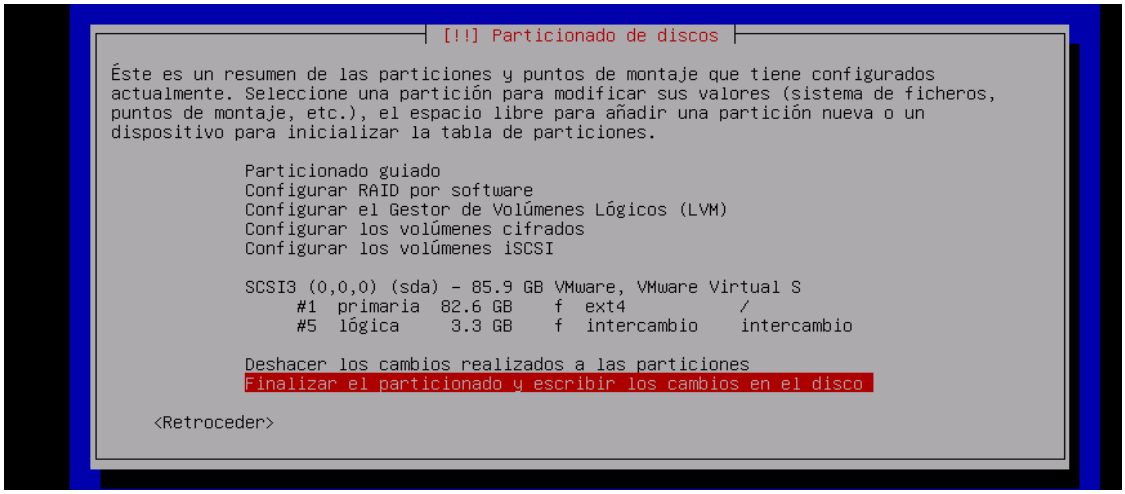


Figura 87

Formateamos el disco seleccionado

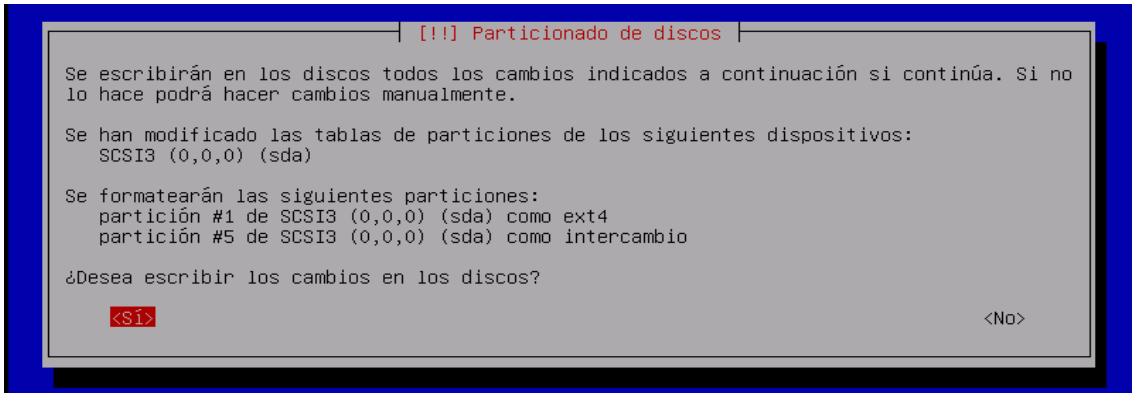


Figura 88

Instalando el sistema

Figura 89

Configuración del proxy, lo dejamos en blanco.

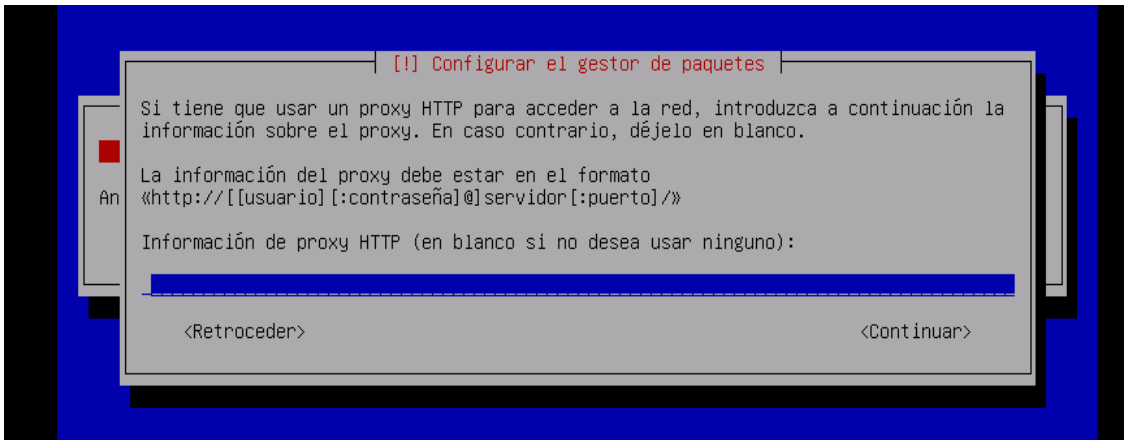


Figura 90

Finalización de la instalación de Kali Linux, iniciamos sesión.

ANEXO B

Figura 91

Ficha Ruc de la empresa DYTELSUR SRL RUC 20602825711

CONSULTA RUC: 20602825711 - DYTELSUR S.R.L.			
Número de RUC:	20602825711 - DYTELSUR S.R.L.		
Tipo Contribuyente:	SOC.COM.RESPONS. LTDA		
Nombre Comercial:	-		
Fecha de Inscripción:	23/01/2018	**Fecha Inicio de Actividades:**	23/01/2018
Estado del Contribuyente:	ACTIVO		
Condición del Contribuyente:	HABIDO		
Dirección del Domicilio Fiscal:	MZA. I1 LOTE. 8 A.H. ALTO EL MOLINO (POR LA LADRILLERA) ICA - PISCO - PISCO		
Sistema de Emisión de Comprobante:	MANUAL	**Actividad de Comercio Exterior:**	IMPORTADOR/EXPORTADOR
Sistema de Contabilidad:	MANUAL		
Actividad(es) Económica(s):	Principal - 6120 - ACTIVIDADES DE TELECOMUNICACIONES INALÁMBRICAS Secundaria 1 - 6110 - ACTIVIDADES DE TELECOMUNICACIONES ALAMBICAS		
Comprobantes de Pago c/aut. de impresión (F. 806 u 816):	NINGUNO		
Sistema de Emisión Electrónica:	FACTURA PORTAL DESDE 04/06/2018		
Afiliado al PLE desde:	-		
Padrones :	NINGUNO		

Imprimir

UNSCH FACULTAD DE INGENIERÍA DE MINAS, GEOLOGÍA Y CIVIL

"Año de la Universalización de la Salud"

ACTA DE SUSTENTACIÓN DE TESIS N° 060-2020-FIMGC

En la ciudad de Ayacucho, en cumplimiento a la **Resolución Decanal N° 403-2020-FIMGC-D,** siendo los nueve días del mes de diciembre del 2020, a horas 10.00 a.m.; se reunieron los jurados del acto de sustentación, en el Auditórium virtual google meet del Campus Universitario de la Universidad Nacional de San Cristóbal de Huamanga.

Siendo el Jurado de la sustentación de tesis compuesto por el Presidente el, **Dr. Ing. Manuel Avelino LAGOS BARZOLA,** Jurado la **Ing. Elinar CARRILLO RIVEROS,** Jurado – Asesor el **Mg. Ing. Hubner JANAMPA PATILLA,** y Secretario del proceso **Mg. Ing. Christian LEZAMA CUELLAR,** con el objetivo de recepcionar la sustentación de la tesis denominada **"SISTEMA DE DETECCIÓN DE INTRUSOS SURICATA OPEN SOURCE COMO MECANISMO DE SEGURIDAD CORPORATIVA EN ENTORNOS LIBRES, 2020",** sustentado por el Bach**. Danny Víctor TINEO MORALES,** bachiller **en Ingeniería de Sistemas.**

El Jurado luego de haber recepcionado la sustentación de la tesis y realizado las preguntas, el sustentante al haber dado respuesta a las preguntas, y el Jurado haber deliberado; califica con la nota aprobatoria de **16 (dieciséis).**

En fe de lo cual, se firma la presente acta, por los miembros integrantes del proceso de sustentación.

Dr. Ing. Manuel Avelino LAGOS BARZOLA
Presidente

Ing. Elinar CARRILLO RIVEROS
Jurado

Mg. Ing. Hubner JANAMPA PATILLA
Jurado – Asesor

Mg. Ing. Christian LEZAMA CUELLAR
Secretario del Proceso

c.c.:
Bach. Danny Víctor TINEO MORALES
Jurados (4)
Archivo

FACULTAD DE INGENIERÍA
DE MINAS, GEOLOGÍA Y CIVIL
Av. Independencia S/N
Ciudad Universitaria
Central Tel 066 312510
Anexo 151

UNSCH FACULTAD DE INGENIERÍA DE MINAS, GEOLOGÍA Y CIVIL

CONSTANCIA DE ORIGINALIDAD DE TRABAJO DE INVESTIGACIÓN

El que suscribe; responsable verificador de originalidad de trabajos de tesis de pregrado en segunda instancia para las Escuelas Profesionales de la Facultad de Ingeniería de Minas, Geología y Civil; en cumplimiento a la Resolución de Consejo Universitario N° 039-2021-UNSCH-CU, Reglamento de Originalidad de Trabajos de Investigación de la UNSCH y Resolución Decanal N° 158-2021-FIMGC-UNSCH-D, deja constancia que:

- Apellidos y Nombres del Bach. : Tineo Morales Danny Víctor
- Escuela Profesional : Ingeniería De Sistemas
- Título de la Tesis : Sistema de Detección de Intrusos Suricata Open Source como mecanismo de seguridad corporativa en entornos libres, 2020.
- Evaluación de la originalidad : 4 % de similitud

Por tanto, según los artículos 12, 13 y 17 del Reglamento de Originalidad de Trabajos de Investigación, ***es procedente otorgar la constancia de originalidad*** para los fines que crea conveniente.

Ayacucho, 02 de junio del 2021

CCatamayo

Firmado digitalmente por Mg. Ing. Johnny Henrry Ccatamayo Barrios
Fecha: 2021.06.02 10:15:46 -05'00'

Mg. Ing. Ccatamayo Barrios Johnny Henrry
Verificador de originalidad de trabajos de tesis de pregrado de la FIMGC

Numero de constancia: 039-2021-FIMGC.

FACULTAD DE INGENIERÍA
DE MINAS, GEOLOGÍA Y CIVIL
Av. Independencia S/N
Ciudad Universitaria
Central Tel 066 312510
Anexo 151

Sistema de Detección de Intrusos Suricata Open Source como mecanismo de seguridad corporativa en entornos libres, 2020

por Danny Víctor Tineo Morales

Fecha de entrega: 02-jun-2021 09:43a.m. (UTC-0500)
Identificador de la entrega: 1599065872
Nombre del archivo: TESIS_Danny_Victor_Tineo_Morales_EPIS.docx (5.55M)
Total de palabras: 21557
Total de caracteres: 116572

Sistema de Detección de Intrusos Suricata Open Source como mecanismo de seguridad corporativa en entornos libres, 2020

INFORME DE ORIGINALIDAD

4%	4%	0%	1%
INDICE DE SIMILITUD	FUENTES DE INTERNET	PUBLICACIONES	TRABAJOS DEL ESTUDIANTE

FUENTES PRIMARIAS

1	Submitted to Universidad Nacional de San Cristóbal de Huamanga Trabajo del estudiante	1%
2	lajc.epn.edu.ec Fuente de Internet	1%
3	idoc.pub Fuente de Internet	<1%
4	doku.pub Fuente de Internet	<1%
5	qdoc.tips Fuente de Internet	<1%
6	es.scribd.com Fuente de Internet	<1%
7	documentop.com Fuente de Internet	<1%
8	rodin.uca.es Fuente de Internet	<1%

9	riunet.upv.es Fuente de Internet	<1 %
10	www.pearson.com.au Fuente de Internet	<1 %
11	stadium.unad.edu.co Fuente de Internet	<1 %
12	www.authorstream.com Fuente de Internet	<1 %
13	repositorio.unh.edu.pe Fuente de Internet	<1 %
14	alibaba-cloud.medium.com Fuente de Internet	<1 %

Excluir citas	Activo	Excluir coincidencias	< 30 words
Excluir bibliografía	Activo		