
# Análisis y selección de un Unified Threat Management (UTM) Open-Source para fortalecer la seguridad de la información en las PYMES

Camacho Rueda, Carlos Steveen

Departamento de Ciencias de la Computación

Carrera de Ingeniería en Tecnologías de la Información

Trabajo de titulación, previo a la obtención del título de Ingeniera en Tecnologías de la Información

Ing. Núñez Agurto, Alberto Daniel, MGS

08 de septiembre de 2021

# Informe de originalidad

NOMBRE DEL CURSO
NRC 6938 MIC - PI

NOMBRE DEL ALUMNO
CARLOS STEVEEN CAMACHO RUEDA

NOMBRE DEL ARCHIVO
CARLOS STEVEEN CAMACHO RUEDA - Tesis_Verificación_Originalidad

SE HA CREADO EL INFORME
7 sept 2021

## Resumen

| | | |
|---|---|---|
| Fragmentos marcados | 15 | 3 % |
| Fragmentos citados o entrecomillados | 7 | 2 % |

Coincidencias de la Web

| | | |
|---|---|---|
| espam.edu.ec | 2 | 0,6 % |
| enovatics.es | 2 | 0,5 % |
| indotel.gob.do | 2 | 0,4 % |
| lareferencia.info | 2 | 0,4 % |
| issuu.com | 1 | 0,3 % |
| simplesite.com | 1 | 0,3 % |
| mintic.gov.co | 2 | 0,3 % |
| uan.mx | 1 | 0,3 % |
| shco.tech | 1 | 0,2 % |
| planificacion.gob.ec | 1 | 0,2 % |
| gobiernoelectronico.gob.ec | 1 | 0,2 % |
| espe.edu.ec | 1 | 0,1 % |
| unlp.edu.ar | 1 | 0,1 % |
| restituciondetierras.gov.co | 1 | 0,1 % |
| slideshare.net | 1 | 0,1 % |
| ups.edu.ec | 1 | 0,1 % |
| redhat.com | 1 | 0,1 % |

1 de 22 fragmentos
Fragmento del alumno MARCADO

...El Data Center De La ESPAM MFL", tuvieron como **objetivo mejorar la seguridad de la información en el Data Center de la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López**



Firmado electrónicamente por:
ALBERTO DANIEL NUNEZ AGURTO

.....

**Agurto Núñez, Daniel Alberto, MGS**

C. C: 1716572548

**DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN**

**CARRERA DE INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN**

## CERTIFICACIÓN

Certifico que el trabajo de titulación, "**Análisis y selección de un Unified Threat Management (UTM) Open-Source para fortalecer la seguridad de la información en las PYMES**" fue realizado por el señor **Camacho Rueda, Carlos Steveen** el cual ha sido revisado y analizado en su totalidad por la herramienta de verificación de similitud de contenido; por lo tanto cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, razón por la cual me permito acreditar y autorizar para que lo sustente públicamente.

Santo Domingo, 08 de septiembre del 2021

Firma:

ALBERTO DANIEL NUÑEZ AGURTO

.....

**Agurto Núñez, Daniel Alberto, MGS**

C. C: 1716572548

**DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN**

**CARRERA DE INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN**

## RESPONSABILIDAD DE AUTORÍA

Yo **Camacho Rueda, Carlos Steveen**, con cédula de ciudadanía n° 2350174096, declaro que el contenido, ideas y criterios del trabajo de titulación**: Análisis y selección de un Unified Threat Management (UTM) Open-Source para fortalecer la seguridad de la información en las PYMES** es de mi autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

**Santo Domingo, 08 de septiembre del 2021**

Firma

**Camacho Rueda, Carlos Steveen**

C.C.: 2350174096

**DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN**

**CARRERA DE INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN**

## AUTORIZACIÓN DE PUBLICACIÓN

Yo **Camacho Rueda, Carlos Steveen,** con cédula de ciudadanía n° 2350174096, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: **Análisis y selección de un Unified Threat Management (UTM) Open-Source para fortalecer la seguridad de la información en las PYMES** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

**Santo Domingo, 08 de septiembre del 2021**

Firma

.....

**Camacho Rueda, Carlos Steveen**

C.C.: 2350174096

**Dedicatoria**

El presente proyecto de titulación lo dedico de todo corazón a Dios,

a mi familia, a los docentes y compañeros de la universidad,

incluyendo a todas las personas que han sido parte de mi vida.

**Agradecimiento**

Agradezco a Dios por darme todo lo vivido.

A mi familia por el apoyo en mis estudios.

Al ingeniero Daniel Núñez por la tutoría, acompañamiento y ayuda en todo momento durante todo este proceso.

A mis compañeros y personas cercanas que han sido de gran ayuda a lo largo de mi vida.

**Índice de contenidos**

**Índice de tablas**

**Índice de figuras**

**Resumen**

En la actualidad las PYMES se encuentran conectadas al Internet y exponen su red, información y aplicaciones a diferentes tipos de ataques, que pueden ser ejecutados a través de malware, spyware, accesos no autorizados y diversas combinaciones de amenazas externas e internas, en su mayoría no implementan sistemas de seguridad por los altos costos en licenciamiento. Una alternativa, para contrarrestar estas amenazas es la implementación de sistemas UTM Open-Source. Sin embargo, existen varias soluciones de UTMs Open-Source, y debido a la variedad y enfoque de los estudios comparativos disponibles, se dificulta determinar de manera objetiva la mejor opción que se ajuste a las necesidades de seguridad de las PYMES. El objetivo principal del presente proyecto, es el de realizar el análisis y selección de un UTM Open-Source, para fortalecer la seguridad de la información en las PYMES. A partir de la revisión sistemática de literatura de la documentación y de proyectos desarrollados sobre la implementación de los UTM Open-Source, se determinó los tres UTMs Open-Source con mejores prestaciones de seguridad. Se desarrolló una matriz comparativa, para analizar el funcionamiento de los servicios en un entorno de pruebas virtual, bajo diferentes tipos de ataques como la descarga de archivos maliciosos, escaneo de puertos y vulnerabilidades. Los resultados obtenidos determinaron que el UTM Open-Source Endian obtuvo los mejores resultados.

- Palabras Clave:
    - **UTM**
    - **FIREWALL**
    - **ANTIVIRUS**
    - **PROXY**
    - **OPEN-SOURCE**

**Abstract**

SMEs are currently connected to the Internet and expose their network, information, and applications to different types of attacks, such as malware, spyware, unauthorized access, and various combinations of external and internal threats, most of them do not implement security systems due to high licensing costs. An alternative to counteract these threats is the implementation of Open-Source UTM systems. However, there are several Open-Source UTM solutions, and due to the variety and focus of the available comparative studies, it is difficult to objectively determine the best option that fits the security needs of SMEs. The main objective of this project is to analyze and select an Open-Source UTM to strengthen information security in SMEs. We made a comparative matrix, based on a systematic literature review of the documentation and projects developed, on the implementation of UTMs Open-Source. From the systematic literature review of the documentation and projects developed on the implementation of Open-Source UTMs, the three Open-Source UTMs with the best security performance were determined. A comparative matrix was developed to analyze the performance of the services in a virtual test environment under different types of attacks such as malicious file downloads, port scanning, and vulnerabilities. The results obtained determined that the Open-Source Endian UTM obtained the best results.

- Keywords:

    - **UTM**
    - **FIREWALL**
    - **ANTIVIRUS**
    - **PROXY**
    - **OPEN-SOURCE**

# Capítulo I

## Introducción

El presente capítulo abarca temas importantes sobre la seguridad de la información y los UTM (Unified Threat Management, Gestión Unificada de Amenazas) Open-Source, de igual forma, se describen los antecedentes, la problemática con la justificación, además, se plantean los objetivos y el alcance del presente proyecto de titulación, de esta manera, se puede conocer la importancia de implementar un UTM Open-Source en una PYME (Pequeña y Mediana Empresa).

## Antecedentes

La seguridad de la información es uno de los aspectos más importantes en una organización, esta garantiza la confidencialidad, la disponibilidad e integridad de la información. Esto se logra mediante la aplicación de un conjunto de controles que son seleccionados a través del proceso de gestión de riesgos y gestionados mediante un sistema de gestión de la seguridad de la información (SGSI) (ISO/IEC 27000, 2018). Sin embargo, constantemente las PYMES suelen ser víctimas de ataques a los sistemas de información. Estos ataques suelen ser realizados a través de malware, spyware, accesos no autorizados, robos de contraseñas y combinaciones de amenazas externas e internas.

Para minimizar el riesgo a estas amenazas existen los UTM, estos UTM hacen referencia a un dispositivo de seguridad como una combinación de hardware, software y tecnologías de red cuya función principal es realizar múltiples funciones de seguridad (Qi et al., 2007). Dentro de los UTM, está la tecnología de seguridad denominada firewall, estos mantienen la seguridad perimetral, para proteger la comunicación entre los dispositivos de la red interna y las conexiones externas. Un firewall tiene la capacidad de inspeccionar los paquetes entrantes o

salientes de la red, a nivel de la capa de aplicación y realizar operaciones, para comprobar, permitir o denegar el tráfico de la red (Senthilkumar & Muthukumar, 2018).

El presente proyecto de titulación tiene como objetivo realizar el análisis y selección de un UTM Open-Source, para fortalecer la seguridad de la información en las PYMES. Por lo tanto, se realizó la selección de tres UTM Open-Source más utilizados, mediante una evaluación basada en sus características, documentación, estudios finalizados y el soporte a sus distribuciones. Los UTM seleccionados serán puestos a prueba en diferentes escenarios de ataques. Este análisis tiene el propósito de determinar cuál UTM Open-Source tiene las características necesarias de seguridad y desempeño, para ser implementado en una PYME.

**Definición de la problemática**

Debido a los avances tecnológicos, las PYMES operan constantemente con una conexión a Internet, además, esto lleva consigo el incremento de amenazas y vulnerabilidades para la red de una empresa. Para contrarrestar estos problemas, existen diversas soluciones UTM Open-Source, cada una con sus respectivas características y funcionalidades que los hacen únicos. Estos UTM Open-Source ofrecen varios servicios de seguridad como antivirus, firewall, IDS, IPS, servidor proxy, servidor DHCP, entre otros, y pueden ser implementados para uso doméstico o entornos de trabajo como las PYMES. Por otra parte, las PYMES cuentan con recursos limitados, debido a esto, se les dificulta la implementación de un UTM comercial, sin embargo, estos son una herramienta de seguridad muy importante e indispensable, ya que permiten controlar el tráfico entrante y saliente en su red interna, para la protección de la información.

Los UTM Open-Source son la mejor alternativa para fortalecer la seguridad de la información en una PYME, sin embargo, existen varias soluciones y diferentes estudios

comparativos entre algunos UTM Open-Source, además, por la variedad y enfoque de estos estudios, se dificulta determinar la mejor alternativa UTM Open-Source para una PYME.

Por esta razón, se debería realizar la selección de los UTM Open-Source más utilizados, mediante una evaluación basada en sus características, documentación, estudios realizados y el soporte a sus distribuciones, para ser puestos a prueba en diferentes escenarios de ataques, con el propósito de determinar cuál UTM Open-Source presenta las mejores características en seguridad y desempeño, para ser implementado en una PYME.

**Justificación**

Por lo general, las PYMES no saben dar importancia a la seguridad de la información, incluso, debido a la limitación de sus recursos, no consideran la idea de implementar sistemas de seguridad, o simplemente suelen pensar que son gastos innecesarios, por esta razón las PYMES están expuestas a diferentes tipos de ataques informáticos, en donde está en riesgo toda la información privada de la empresa.

Los UTM Open-Source son una solución perfecta para las PYMES, ya que no representan grandes gastos económicos en inversión, para la protección de la información, con la implementación de un UTM Open-Source, las PYMES podrían mejorar la protección de su red interna y de las aplicaciones, ante ataques informáticos.

El presente proyecto de titulación está enfocado en determinar el UTM Open-Source más óptimo como la mejor solución, para mitigar los riesgos informáticos y mejorar la seguridad de la información en las PYMES.

## Objetivos

### *Objetivo General*

Realizar el análisis y selección de un Unified Threat Management (UTM) Open-Source, para fortalecer la seguridad de la información en las PYMES.

### *Objetivo Específico*

- Realizar una revisión sistemática de literatura de la documentación y de proyectos desarrollados sobre la implementación de los UTM Open-Source.
- Analizar y seleccionar los UTM Open-Source disponibles.
- Implementar los UTM Open-Source y realizar las respectivas configuraciones.
- Demostrar la potencialidad de la implementación de los UTM Open-Source y sus beneficios.

## Alcance

El presente proyecto de titulación, se espera determinar la mejor solución, para la seguridad de la informática en las PYMES, por medio de un análisis comparativo entre tres UTM Open-source, también se espera realizar una selección de tres UTM Open-Source más utilizados, mediante una matriz comparativa, documentación, estudios finalizados y el soporte a sus distribuciones, después, los UTM seleccionados serán puestos a prueba en diferentes escenarios de ataques, este análisis tiene el propósito de determinar, cuál UTM Open-Source tiene las características necesarias de seguridad y desempeño, para ser implementado en una PYME.

# Capítulo II

## Introducción

En este capítulo se especifican las definiciones sobre la seguridad de la información, seguridad informática, la ciberseguridad, también sobre los controles de seguridad especificados en la norma 27002, además se detallan aspectos importantes sobre los UTM, firewalls, PYMES, servidores y la virtualización. En el contexto del trabajo se especifican los UTM Open-Source que serán puestos a prueba.

## Antecedentes investigativos

(Pablo & Loor, 2017), en su tesis "Sistema Perimetral Firewall Y Fortalecimiento De La Seguridad En El Data Center De La ESPAM MFL", tuvieron como objetivo mejorar la seguridad de la información en el Data Center de la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López. La implementación de un servidor perimetral firewall en el Data Center fue desarrollada de acuerdo con la metodología de desarrollo de software cascada. Realizaron la recopilación de información haciendo uso de herramientas tales como encuestas, fichas de observación y entrevistas, escogieron tres sistemas de seguridad firewall los cuales fueron IpFire versión 2.17, pfSense versión 2.2.6 y Untangle NG Firewall versión 11.21, y mediante un estudio comparativo aplicado con la herramienta checklist determinaron el software a utilizar, como resultado determinaron que pfSense es el más adecuado para los requisitos de la universidad ESPAM MFL. Implementaron el UTM pfSense, como resultados obtuvieron una mejor protección de los datos y la seguridad en la red, además el control, bloqueo y monitoreo de ataques informáticos a la red interna de la universidad, esto contribuyó a la reducción de costos y la gestión de nuevos planes en niveles de seguridad del Data Center en menor tiempo.

(Zambrano & Sánchez, 2013), en su proyecto “Repotenciación de un sistema de firewall de código abierto basado en funcionalidades de plataforma propietaria”, tuvieron como objetivo realizar un evaluación y potenciación de una plataforma de Firewall Open-Source. Se basaron en las diferentes características de un sistema propietario y verificaron el desempeño y efectividad ante ataques informáticos. Realizaron un análisis de varios Firewalls Open-Source, para lo cual, establecieron como método de selección, la evolución diversos parámetros como el sistema operativo y soporte a las distribuciones, con este análisis, determinaron las herramientas a utilizar la cuales fueron Clearos, Zentyal y Pfsense. Establecieron varios tipos de ataques para la ejecución de pruebas, para obtener resultados con respecto a la evaluación del desempeño, consumo de CPU, memoria, de los sistemas de firewall. Determinaron que ClearOS obtuvo los mejores resultados en optimización de memoria RAM, mejor desempeño ante actividades sospechosas de red.

(Miguez, 2017), “Implementación de un Sistema de Gestión Unificada de Amenazas (UTM) para la Empresa de Créditos Palacio del Hogar”, tuvo como objetivo dar una solución óptima a los problemas de Control de Seguridad de la Información existente dentro de la Infraestructura de Red de la empresa de Créditos “Palacio del Hogar”. La metodología que utilizó fue primero hacer el levantamiento de la información, luego el análisis y diseño, y por último la implementación y pruebas. El software UTM que utilizó es el Endian Firewall Community, considerado que es un software Open-Source y permite como base de configuración clasificar en 5 zonas bien marcadas a toda la red de la empresa, en los resultados mencionó que la empresa obtuvo varios beneficios referentes al control de tráfico de la red entrante como saliente, accesos VPN seguros, control de navegación por Proxy, conexiones más seguras, monitoreo de recursos, entre otros.

(Arunwan et al., 2016), en su paper “Defensive performance comparison of firewall systems”, su objetivo fue realizar una comparación del rendimiento de detección de ataques entre dos famosos sistemas de firewall, Endian y pfSense. Los escenarios de ataque incluyeron escaneo de puertos, ping de la muerte, inundación y ataque de contraseña en diferentes condiciones. Los resultados mostraron que el rendimiento de pfSense fue mejor en general, además menciona que la configuración básica de Endian podría detectar más categorías de ataques, pero con menos ocurrencias. Por último, determina que Endian podría ser el adecuado para pequeñas empresas, también en uso doméstico, con redes y datos menos importantes. Además, menciona que podría valer la pena usar pfSense en una variedad más amplia de organizaciones, sin embargo, con algunas modificaciones de configuración.

(Iriarte Solís et al., 2018), en su trabajo “Evaluación de firewalls basados en software libre”, tuvieron como objetivo evaluar firewalls basados en software libre GNU/LINUX con las características que permitan integrarse a una cama de pruebas en la cual estudiaron, evaluaron y analizaron diversos entornos de red y escenarios de ataque. Las distribuciones que revisaron fueron IPCop, Endian Firewall (EFW), ClearOS y Fedora 2. Los resultados mostraron lo viable que puede ser ClearOS para ser implementado en una cama de pruebas, además obtuvo como resultado una buena respuesta en la defensa de los ataques, con el uso de la herramienta iperf.

(León Casas, 2016), en su trabajo de fin de máster “Estudio de soluciones Unified Threat Management (UTM) de libre acceso”, tuvo como objetivo analizar la seguridad que ofrecen soluciones UTM de libre acceso, analizó las capacidades de protección de las soluciones UTM de libre acceso Endian Firewall Community, Sophos UTM Home Edition, y Untangle NG Firewall, Para ello creó diferentes escenarios simulando las amenazas básicas y avanzadas que un usuario doméstico o una pequeña empresa podrían encontrar para comprobar la eficacia de cada

solución a la hora de detectar y bloquear los distintos ataques, como resultado obtuvo que Sophos fue la que mejor resultados consiguió de las tres soluciones analizadas.

Con la revisión de estos trabajos se determina que pfSense, Endian y ClearOS obtuvieron los mejores resultados en general, pero no hay un análisis en conjunto de estos 3 UTM Open-Source, en el trabajo de fin de master, el mejor UTM fue Sophos, pero este no es Open-Source, por esta razón no es tomado en cuenta para este proyecto.

**Software Open-Source**

El Open-Source software se define como sistemas a los que pueden acceder libremente todos los usuarios sin ningún coste, los usuarios pueden realizar modificaciones o personalizaciones en el código fuente basándose en la licencia pública general de GNU. El enfoque de Open-Source software (OSS) o código abierto ha surgido como un nuevo modelo para crear y distribuir varios sistemas de software. Al adoptar un procedimiento de desarrollo mucho más sencillo que el de la ingeniería de software tradicional, se ha demostrado que el enfoque del OSS produce sistemas de software altamente fiables en un periodo de tiempo mucho más corto. Algunos ejemplos son los servidores web Apache, los sistemas operativos Linux y el navegador Mozilla, que actualmente utilizan miles o millones de usuarios finales. Debido a sus características prometedoras, algunas empresas con fines de lucro conocidas, como IBM, han adoptado el modelo de código abierto y participan activamente en el desarrollo de productos OSS. Su creciente popularidad e importancia indican la necesidad de investigar la evaluación de la fiabilidad del OSS (Razzaq et al., 2018), (Hamid et al., 2016).

## PYME

Las PYMES son el grupo de pequeñas y medianas organizaciones, que según su volumen de ventas, capital social, proporción de trabajadores, y su grado de producción o activos muestran propiedades propias de esta clase de entidades económicas (SRI, 2012).

## Seguridad de la información

La norma ISO/IEC 27002 ofrece directrices para las normas de seguridad de la información de la organización y las prácticas de gestión de la seguridad de la información, también la selección, la implementación y la gestión de los controles teniendo en cuenta el entorno de riesgo de la seguridad de la información de la organización (ISO/IEC 27002, 2013).

### *Seguridad informática*

La seguridad informática se define como un método de protección frente a las amenazas externas, que utiliza tecnologías y herramientas de desarrollo avanzadas, para proteger las infraestructuras de software y hardware conectadas al Internet, estas infraestructuras pueden pertenecer a las empresas, gobiernos o instituciones/organizaciones en las que participan los gobiernos (Efthymiopoulos, 2019)

### *Gestión de la seguridad de las redes*

De acuerdo con (ISO/IEC 27002, 2013), la gestión de la seguridad de las redes tiene como objetivo garantizar la protección de la información en las redes y sus instalaciones de procesamiento.

**Controles de la red.** Las redes se deben gestionar y controlar, para proteger la información de los sistemas y aplicaciones. Estos controles garantizan la protección de los servicios conectados contra el acceso no autorizado. En particular, deben tenerse en cuenta los siguientes elementos:

a) Deben establecerse responsabilidades y procedimientos para la gestión de los equipos de red;
b) la responsabilidad operativa de las redes debe separarse de las operaciones informáticas, cuando proceda;
c) deben establecerse controles especiales para salvaguardar la confidencialidad e integridad de los datos que pasan por las redes públicas o por las redes inalámbricas y para proteger los sistemas y aplicaciones conectados; también pueden ser necesarios controles especiales para mantener la disponibilidad de los servicios de red y de los ordenadores conectados;
d) debe aplicarse un registro y una supervisión adecuados para permitir el registro y la detección de acciones que puedan afectar a la seguridad de la información o que sean relevantes para ella;
e) las actividades de gestión deben estar estrechamente coordinadas tanto para optimizar el servicio a la organización como para garantizar que los controles se aplican de forma coherente en toda la infraestructura de procesamiento de la información;
f) los sistemas de la red deben estar autentificados;
g) la conexión de los sistemas a la red debe ser restringida.

**Seguridad de los servicios de red.** Los mecanismos de seguridad, los niveles de servicio y los requisitos de gestión de todos los servicios de red se deben identificar e incluir en los acuerdos de servicios de red, ya sean servicios internos o externos. Un proveedor de servicios de red, para gestionar los servicios acordados de forma segura debe determinar y supervisar periódicamente los derechos de auditoría. Para algunos servicios como las propiedades de seguridad, los niveles de servicio y los requisitos de administración, se deben establecer

medidas de seguridad esenciales. La organización debe garantizar que los proveedores de servicios de red apliquen estas medidas.

Los servicios de red incluyen el suministro de conexiones, servicios de red privada, redes de valor añadido y soluciones de seguridad de red gestionadas, como firewall o sistemas de detección de intrusiones. Estos servicios pueden ir desde un simple ancho de banda no gestionado hasta complejas ofertas de valor añadido. Las características de seguridad de los servicios de red pueden ser:

a) La tecnología aplicada para la seguridad de los servicios de red, como la autenticación, el cifrado y los controles de conexión a la red.
b) Los parámetros técnicos necesarios para la conexión segura con los servicios de red, de acuerdo con las normas de seguridad y conexión a la red.
c) Los procedimientos de uso de los servicios de red para restringir el acceso a los servicios de red o a las aplicaciones, cuando sea necesario.

**Segregación en las redes**. Los grupos de servicios de información, los usuarios y los sistemas de información deben estar segregados en las redes.

Un método para gestionar la seguridad de las grandes redes es dividirlas en dominios de red separados. Los dominios tienen la posibilidad de elegirse en funcionalidad de los niveles de confianza, por ejemplo: dominio de acceso público, dominio de escritorio o dominio de servidor. El perímetro de cada dominio debe estar bien definido. La entrada entre dominios de red está autorizada, sin embargo, debería controlarse en el perímetro por medio de una pasarela, por ejemplo, un firewall o un router de filtrado. Los criterios de segregación de las redes en dominios, y la entrada autorizada por medio de las pasarelas, tienen que fundamentarse en una evaluación de los requisitos de estabilidad de cada dominio.

### *Gestión técnica de la vulnerabilidad*

De acuerdo con (ISO/IEC 27002, 2013), la gestión técnica de la vulnerabilidad tiene como objetivo evitar la explotación de vulnerabilidades técnicas.

**Gestión de vulnerabilidades técnicas.** La información sobre las vulnerabilidades técnicas de los sistemas de información que se utilizan debe obtenerse de tal forma que se pueda evaluar la exposición de la organización a dichas vulnerabilidades y tomar las medidas adecuadas para abordar el riesgo asociado.

Tienen que tomarse medidas apropiadas y oportunas en contestación a la identificación de probables vulnerabilidades, para establecer un proceso de gestión eficaz de las vulnerabilidades técnicas, deben seguirse la siguiente orientación:

Los parches deben ser probados y evaluados antes de ser instalados para asegurar que son efectivos y no dan lugar a efectos secundarios que no puedan ser tolerados; si no hay ningún parche disponible, se deben considerar otros controles, tales como:

1) Desactivar servicios o capacidades relacionados con la vulnerabilidad;
2) Adaptar o añadir controles de acceso, por ejemplo, firewall, en las fronteras de la red;
3) Aumentar la vigilancia para detectar ataques reales;
4) Aumentar la concienciación sobre la vulnerabilidad;

Existe una idea errónea de que los firewalls protegen a los usuarios del malware en su ordenador, cuando en realidad los firewalls protegen a los usuarios del software defectuoso. Existe la preocupación de que los firewalls den a los usuarios una falsa sensación de seguridad. Los firewalls no son invulnerables y no impedirán la ejecución de malware si el usuario lo permite (Thaler, 2014).

**Unified Threat Management (UTM)**

Los sistemas de gestión unificada de amenazas (UTM) hacen referencia a un dispositivo de seguridad como una combinación de hardware, software y tecnologías de red cuya función principal es realizar múltiples funciones de seguridad (Qi et al., 2007).

Según (Asghari et al., 2016), los UTM reúnen en una sola plataforma las siguientes tecnologías de seguridad de red:

- Firewall.
- Antispam.
- Antivirus.
- Filtrado de URL.
- Red privada virtual (VPN).

***UTM basado en software***

Los UTM basados en software son implementados en servidores de computadora con una configuración basada en la cantidad de usuarios y de las aplicaciones que se ejecutan a la vez. Los UTM basados en software son adaptables, se pueden insertar módulos comprando licencias de software a través de Internet. Actualmente podemos encontrar UTM basados en software de paga y gratuitos o de código abierto para la seguridad de la red (Asghari et al., 2016).

***Firewalls***

Un firewall es un sistema de seguridad que detecta y gestiona continuamente los paquetes salientes según las reglas de acceso al firewall. Según las reglas de acceso de los firewalls predefinidos, la red pública y las redes privadas se pueden separar. La seguridad del sistema depende de las reglas de la configuración de los firewalls, ya que de lo contrario el

tráfico de paquetes no deseados puede pasar o bloquear los paquetes deseados. La función principal del firewall es controlar la política de seguridad y proteger la red de la organización del tráfico no legítimo. También proporciona una alta seguridad flexible a los usuarios remotos. Los firewalls pueden alcanzar sus objetivos mediante pruebas de todo el tráfico de red restringido y no restringido de acuerdo con las reglas predefinidas (Senthilkumar & Muthukumar, 2018).

**Características de los Firewalls.** Según (Freed, 2000) los firewalls actúan como un punto final de protocolo, por ejemplo, un cliente / servidor SMTP o un agente proxy web, como un filtro de paquetes o una combinación de ambos.

Cuando un firewall actúa como un punto final de protocolo, puede

1) implementar un subconjunto "seguro" del protocolo;
2) realizar comprobaciones exhaustivas de la validez del protocolo;
3) utilizar una metodología de implementación diseñada para minimizar la probabilidad de errores;
4) correr en un ambiente aislado o "seguro";
5) utilizar alguna combinación de estas técnicas en conjunto.

Los firewalls que actúan como filtros de paquetes no son visibles como puntos finales de protocolo. El firewall examina cada paquete y luego

1) pasa el paquete al otro lado sin cambios;
2) descarta el paquete por completo;
3) maneja el paquete en sí de alguna manera;

Los firewalls suelen basar algunas de sus decisiones en direcciones IP de origen y destino y números de puerto. Por ejemplo, los firewalls pueden

1) bloquear paquetes del lado de Internet que reclaman una dirección de origen de un sistema en la red interna;
2) bloquear las conexiones TELNET o RLOGIN de Internet a la red interna;
3) bloquear las conexiones SMTP y FTP a Internet desde sistemas internos no autorizados para enviar correos electrónicos o mover archivos;
4) actuar como un servidor intermedio en el manejo de conexiones SMTP y HTTP en cualquier dirección, o
5) requieren el uso de un protocolo de encapsulación y negociación de acceso como SOCKS para obtener acceso a Internet, a la red interna o ambos.

**Reglas de acceso a Firewalls.** De acuerdo con (Senthilkumar & Muthukumar, 2018), indican que las reglas de acceso son reglas de seguridad de la red que pueden ser establecidas por el administrador de la red, para permitir el tráfico a sus respectivos servidores de alojamiento web, archivos FTP y servidores demonio, dando así a los propietarios de las computadoras un control inmenso sobre el tráfico que fluye dentro y fuera de sus sistemas. o redes. En los firewalls distribuidos, no debe haber dos firewalls que tengan las mismas reglas y regulaciones de acceso.

Las reglas de acceso de los firewalls permiten mantener alejados a los usuarios malignos y también ampliar el control sobre los usuarios riesgosos inherentes dentro de su empresa. Una regla de acceso es darse cuenta de la información y los servicios disponibles, presentes inherentes al deterioro y si ya existe alguna seguridad para inhibir el uso indebido.

Las políticas son reglas de tráfico, la política de red permite al administrador del sistema coordinar los elementos de la red, para ofrecer servicios a un conjunto de usuarios. Cada

sistema donde se permita conectarse con todos los demás sistemas vecinos sin ninguna limitación, entonces no habría reglas de acceso para la red.

**Enrutamiento de las reglas de acceso al firewall.** Según (Senthilkumar & Muthukumar, 2018), el enrutamiento del firewall se utiliza para entregar el paquete desde el origen al destino y enviarlo a través de un entorno de red de dominio a otro entorno de dominio. La política de enrutamiento le permite administrar el enrutamiento en secuencia entre las propiedades de enrutamiento y las tablas de enrutamiento.

Este sistema de firewall admite las siguientes áreas de enrutamiento:

- Filtro de paquetes con estado.
- Traducción de direcciones de red (NAT).
- Filtrado basado en la regla de acceso del firewall.
- Coincidencia de paquetes.
- Limpieza de paquetes.

***Firewall de nueva generación***

Los firewalls de nueva generación hacen frente a las nuevas amenazas que comprometen los sistemas de red. La próxima generación de firewalls ofrece más accesibilidad al tráfico de red, operatividad a través de las capas OSI y características avanzadas, para proteger la infraestructura de red contra las amenazas emergentes. El Firewall de nueva generación analiza el tráfico de las aplicaciones e informa de las posibles amenazas detectando las aplicaciones maliciosas que se filtran dentro de las aplicaciones legítimas. También son capaces de detectar los protocolos de control y protocolos de comando utilizados por los bots. Además, pueden inspeccionar el tráfico cifrado en busca de malware descifrando el flujo de paquetes (Neupane et al., 2018).

**VPN.** La red privada virtual (VPN) son redes construidas como una superposición sobre la infraestructura pública de uno o más proveedores, para permitir el acceso entre un conjunto definido de dispositivos. Una VPN puede considerarse simplemente como un túnel autenticado y encriptado que sirve de línea alquilada virtual sobre una infraestructura pública compartida, las VPN no tienen que garantizar el cifrado de los datos. La VPN segura es una conexión encriptada y autenticada por el usuario entre dos segmentos de la misma red privada, de un ordenador a una red privada, o entre dos ordenadores. Debido a que la VPN se realiza entre usuarios/dispositivos autorizados, se necesita un fuerte control de acceso es esencial para una VPN segura (Alshalan et al., 2016).

**Sistema de detección de intrusos.** El sistema de detección de intrusos (IDS) según (Kumar & Singh, 2020), se utiliza para proteger la infraestructura de red y los ordenadores de actividades maliciosas y usos no autorizados. Detecta los diferentes tipos de amenazas. Ayuda a los usuarios y a los administradores de la red a tomar medidas preventivas. Los IDS desempeñan un papel importante en la protección de la infraestructura informática. Los sistemas de detección de intrusiones capturan y analizan el tráfico de la red para detectar actividades sospechosas.

Los IDS funcionan principalmente con dos enfoques diferentes:

- **Detección de anomalías:** En esta técnica, el tráfico de la red o el comportamiento del sistema operativo del host se analiza en base a varios parámetros y se compara con el comportamiento normal. Si el sistema detecta cualquier desviación del comportamiento normal, lanza una alarma.
- **Detección de uso indebido/firmas:** Esta técnica busca un patrón específico de comportamiento que ya se conoce como un ataque. Todos los patrones y

comportamientos maliciosos que se identifican como ataques se almacenan en la base de datos de firmas del IDS. Estas bases de datos de firmas se actualizan continuamente y se utilizan para la detección de ataques. La limitación de esta técnica es que no podrá detectar un ataque nuevo, ya que no habrá una firma disponible.

Los sistemas de detección de intrusos se clasifican a grandes rasgos en dos categorías:

- **Sistema de Detección de Intrusos en la Red (NIDS):** Captura los paquetes del tráfico de la red. La cabecera de los paquetes capturados se analiza en función de varios parámetros para detectar actividades maliciosas. Puede instalarse en la red troncal, el servidor, switches y gateway.
- **Sistema de detección de intrusiones en el host (HIDS):** Se instala en el sistema individual para detectar la intrusión o el uso indebido. El HIDS analiza los archivos clave del sistema, los comportamientos de los procesos, utilización inusual de recursos, acceso no autorizado, etc.

**Sistema de prevención de intrusión.** El Sistema de Prevención de Intrusiones (IPS) según (Pratama et al., 2018), es un sistema que puede detectar automáticamente cualquier actividad sospechosa que sea potencialmente maliciosa en una red. El IPS busca manualmente los datos de los paquetes anómalos y los recoge en un archivo de registro e inspecciona si el paquete debe o no estar en la red. Este proceso no es eficiente y consumiría muchos recursos, por lo que se sugiere un sistema que pueda leer el registro automáticamente.

Hay dos métodos para determinar si un paquete es malicioso o no (Pratama et al., 2018):

- **Basado en firmas:** IPS con esta técnica funciona analizando cada paquete que entra y sale de la red y lo compara con una base de datos que contiene firmas o reglas para los atributos de los paquetes maliciosos. Esta técnica tiene un nivel de adaptación bajo, lo que significa que puede sólo puede detectar tipos de ataques conocidos que se encuentran en la base de datos.
- **Basado en anomalías:** El IPS basado en anomalías compara cada paquete con la estación base en vigor. La estación base definirá todos los parámetros utilizados, empezando por el puerto, el protocolo y el ancho de banda. Si hay tráfico diferente al definido, entonces el IPS alertará al administrador.

**Servidor**

En los modelos cliente-servidor, cada terminal o proceso informático de la red puede ser un cliente o un servidor, un cliente es un programa o un terminal informático que permite a los usuarios acceder a sus interfaces, cada cliente se comunica con cada uno a través del servidor, es decir, el cliente es el que inicia la comunicación y el servidor es el que espera pasivamente para responder a la solicitud del cliente (Maata et al., 2018).

Las empresas tienen la opción de comprar licencias y utilizar el software comercial, por ejemplo: Microsoft Windows como sistema operativo, Microsoft SQL Server como servidor de base de datos, Microsoft Internet Information Server como servidor web, etc., o utilizar software gratuito o de código abierto, por ejemplo: Linux como sistema operativo, MySQL como servidor de base de datos, Apache como servidor web, etc (Soloviev, 2008).

### *Servidor Linux*

Un servidor Linux es un sistema de alto rendimiento para dirigir grandes cantidades de información a través de una conexión de red. La configuración y el mantenimiento de un servidor Linux requiere conocer los inconvenientes del sistema operativo Linux y su catálogo de utilidades de apoyo, así como muchas capas de software de aplicaciones, el servidor Linux es una máquina de alto rendimiento que proporciona acceso a la información de manera rápida y eficientemente como sea posible. El servidor Linux obtiene la información de algún tipo de almacenamiento como el sistema de archivos, una base de datos, o algún otro lugar en la red y entrega esa información a través de la red a quien la haya solicitado, ya sea un ser usuario conectado a un servidor web, un usuario sentado en un shell, o a través de un puerto a otro servidor (Flickenger, 2009).

### Virtualización

(Ma et al., 2012), define la virtualización como al uso virtual de los recursos informáticos tales como las unidades de proceso, el almacenamiento, etc. Esto significa que se combinan o se dividen en los recursos informáticos. El proceso de virtualización permite hacer que un solo recurso físico, como un servidor, un dispositivo de almacenamiento o un sistema operativo, se utilice como múltiples recursos virtuales, y que varios recursos se utilicen como un único recurso virtual. También se pueden integrar varios recursos físicos en muchos recursos virtuales, por ejemplo, la partición o el ensamblaje de hardware o software, la simulación, el uso compartido a tiempo parcial o consolidación de máquinas. La virtualización se divide en diferentes niveles como el nivel de instrucción, nivel abstracto de hardware, nivel de sistema operativo, biblioteca de clases, nivel de programación, máquina de virtualización ligera.

La virtualización puede clasificarse en cuatro tipos:

- **Virtualización de hardware:** El proceso de ejecución real se demuestra construyendo y simulando un traje de instrucciones diferentes. En general, debe traducir todas las instrucciones emitidas por otros ordenadores en instrucciones locales, y ejecutarlas en la máquina real.
- **Virtualización del sistema operativo:** Añade un nivel único entre el sistema operativo y el usuario del sistema para proporcionar un entorno de aplicación múltiple separado.
- **Virtualización del lenguaje de programación:** El lenguaje de programación abarca la plataforma puede aplicarse a la aplicación de la gran empresa, la pequeña aplicación y otras.
- **Virtualización de la biblioteca de programación:** Un mecanismo se utiliza para mejorar el porcentaje de programación disponible para disminuir la dificultad de desarrollo.

***Monitor de máquinas virtuales***

Según (Santoso et al., 2014), un monitor de máquina virtual (VMM) o hipervisor es un componente de software que se encarga de la creación, gestión y ejecución de una máquina virtual (VM), un hipervisor puede ejecutar múltiples VMs simultáneamente en un solo hardware. Las VMs se ejecutan con o sin darse cuenta de que están en un hardware virtualizado.

Hay dos tipos de hipervisores: Tipo-1 y Tipo-2. El hipervisor de tipo 1 se ejecuta exactamente encima del hardware. Normalmente, este tipo de hipervisor trabaja en conjunto con un sistema operativo anfitrión (Host OS). Accede al hardware directamente y virtualiza el hardware para las VM. Cuando las VMs quieren acceder al hardware, su acceso es atrapado por

el hipervisor y luego el hipervisor virtualiza el acceso. El hipervisor se ejecuta junto con un sistema operativo (SO) host para utilizar los controladores de E/S del SO, de modo que el VMM no está inflado con código de controladores. Ejemplos de este tipo de VMM son Xen, Microsoft Hyper-V, ViMo, etc. Por otro lado, el VMM de tipo 2 se ejecuta sobre un SO anfitrión. Los hipervisores de este tipo serán considerados sólo como un proceso por su SO anfitrión. Para el Tipo 2 tenemos como ejemplos Vimo Type-2, y VMware Workstation.

**Contexto del trabajo**

Los Firewalls UTM Open-Source, son alternativa para mitigar la seguridad de las empresas de tipo PYMES, por lo tanto, es importante seleccionar el más adecuado. En el presente trabajo se plantea encontrar el UTM Open-Source más óptimo para la fortalecer seguridad de la información en las PYMES.

Existen varias opciones UTM basados software de código abierto como:

- pfSense.
- Endian.
- OPNSense.
- ClearOS.
- Untangle.
- IPFire.
- Smoothwall.
- UFW.
- CSF.
- Sophos.

Basándose en los antecedentes investigativos se han seleccionado estos tres firewalls para ser implementados y puestos a prueba en diferentes escenarios de ataques.

### *pfSense*

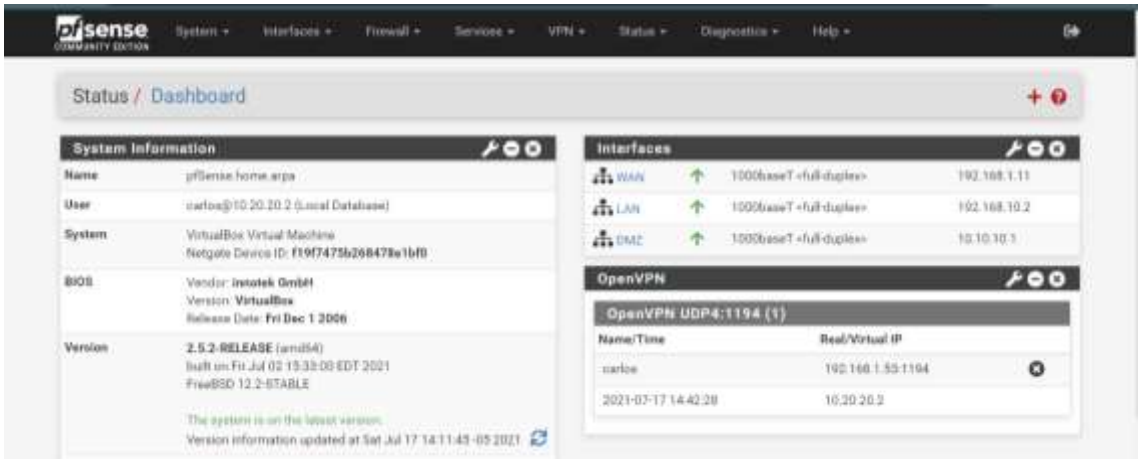
Según (pfSense, 2021), pfSense es una repartición de firewall gratuita, basada en el sistema operativo FreeBSD e incluye paquetes de programa gratuitos de terceros para una función adicional. El programa pfSense, con el apoyo del sistema de paquetes, puede dar la misma funcionalidad o más de los firewalls comerciales habituales. Además, mencionan que pfSense ha reemplazado con éxito todos los firewalls comerciales de renombre como Check Point, Cisco PIX, Cisco ASA, Juniper, Sonicwall, Netgear, Watchguard, Astaro y más. El software pfSense incluye una interfaz web, mostrada en la Figura 1, esta sirve para la configuración de todos los componentes incluidos. No es necesario tener conocimientos de UNIX, no es necesario utilizar la línea de comandos para nada y no es necesario editar manualmente ningún conjunto de reglas.

Los requisitos mínimos de hardware para pfSense Community Edition versión 2.5.2 son:

- CPU compatible con amd64 (x86-64) de 64 bits.
- 1 GB o más de RAM.
- Unidad de disco de 8 GB o más (SSD, HDD, etc.).
- Una o más tarjetas de interfaz de red compatibles.
- Unidad USB de arranque o unidad óptica de alta capacidad (DVD o BD) para la instalación inicial.

**Figura 1**

*Interfaz web de pfSense Community Edition*




***Endian***

De acuerdo con (Endian, 2021), Endian Firewall Community (EFW) es un producto de software de seguridad basado en Linux llave en mano diseñado para el hogar que puede transformar cualquier dispositivo de hardware no utilizado en una solución de Gestión Unificada de Amenazas (UTM) con todas las funciones, además, cuenta con una interfaz web, presentada en la Figura 2, esta sirve para la configuración de los servicios que incluye este UTM. Endian Community está pensado para simplificar la estabilidad y contribuir a defender las redes domésticas usando el poder del código abierto.

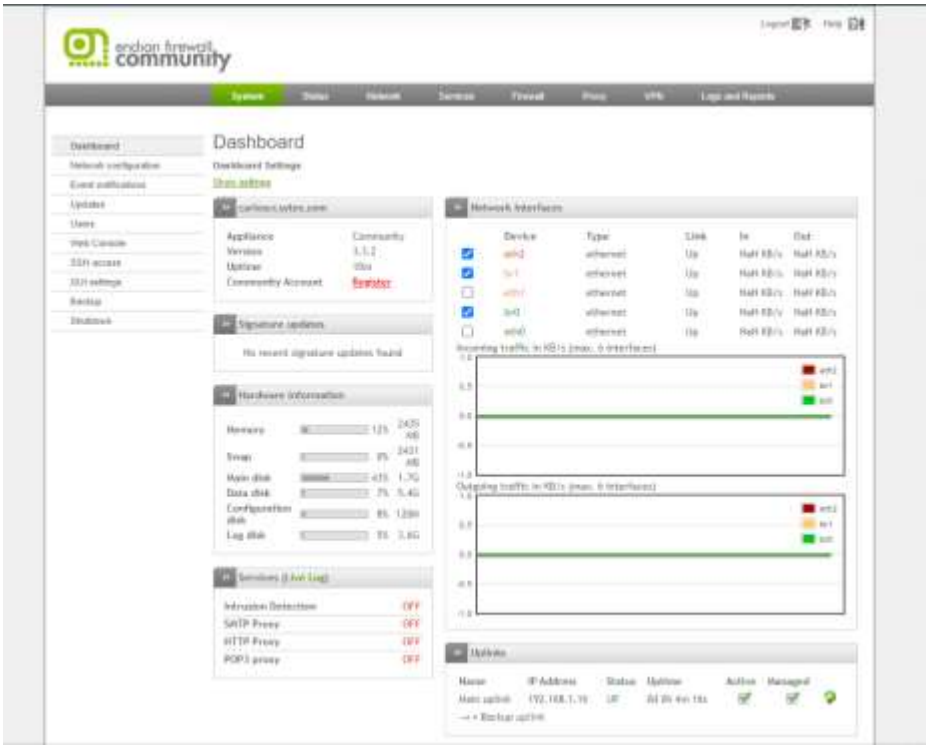
Los requisitos mínimos de hardware para Endian Firewall Community versión 3.3.2 son:

- Compatible con Intel x86_64 /1 GHz como mínimo, se recomiendan 2 GHz de doble núcleo.
- 2 GB mínimo de RAM, se recomiendan 4 GB.
- Se requiere un disco SCSI, SATA, SAS o IDE, se recomienda un mínimo de 8 GB y 20 GB.

- Se admiten las tarjetas de interfaz de red más comunes, incluidas las NIC de fibra y Gigabit.
- Se requiere una unidad de CDROM IDE, SCSI o USB para la instalación, no se requiere después de la instalación.

**Figura 2**

*Interfaz* web *de Endian Firewall Community*




***ClearOS***

(ClearOS, 2021), describe que ClearOS 7 Community Edition es un sistema operativo de servidor Linux de código abierto. Esta edición está diseñada para expertos en Linux y aficionados que disfrutan del código de vanguardia y contribuyen a una comunidad de usuarios globales con sugerencias y soporte de foros. Todas las actualizaciones, correcciones de errores, parches y correcciones de seguridad se proporcionan de forma gratuita, aunque no probada, desde fuentes originales. Las características abarcan más de 75 funciones de TI desde control de

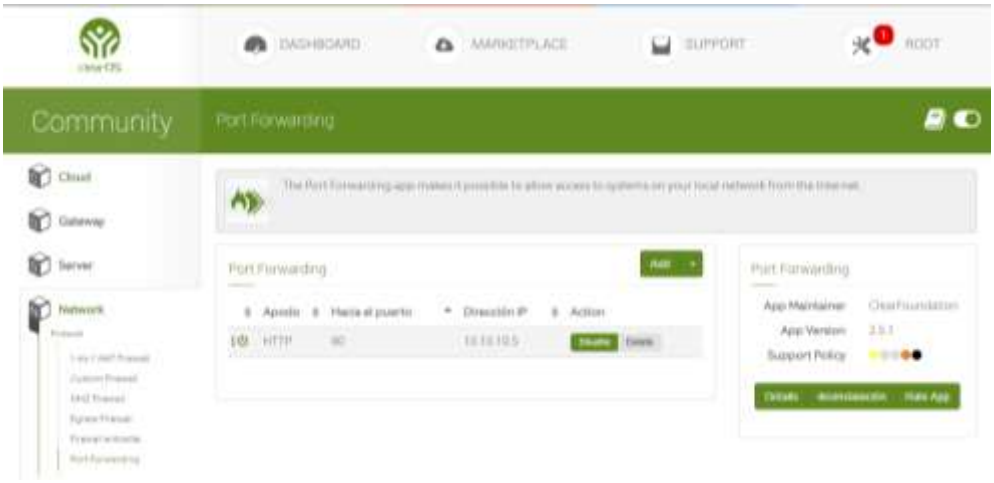
dominio, control de red y ancho de banda, mensajería y más, para la configuración de los servicios, cuenta con una interfaz web, mostrada en la Figura 3.

Los requisitos mínimos de hardware para ClearOS 7 Community Edition (7.2.0) son:

- CPU de 32 bits o 64 bits.
- Se recomienda al menos 1 GB de RAM.
- Se recomiendan al menos 10 GB de disco duro.
- La red puede ser Ethernet, cable, DSL

**Figura 3**

*Interfaz web de ClearOS 7 Community Edition*



# Capítulo III

## Introducción

En el presente capítulo se expone una matriz comparativa entre los tres UTM Open-Source seleccionados, además, se muestra el entorno de pruebas con la configuración de las máquinas virtuales, los servicios configurados en los UTM Open-Source, la topología de red y los ataques realizados en el entorno de pruebas.

## Matriz comparativa entre los UTM opens source

La Tabla 1 presenta una comparación entre los servicios que ofrecen pfSense Community Edition, Endian Firewall Community y ClearOS 7 Community Edition.

**Tabla 1**

*Matriz comparativa entre los UTM opens source*

| **Servicios** | **pfSense Community Edition (versión 2.5.2)** | **Endian Firewall Community (versión 3.3.2)** | **ClearOS 7 Community Edition (versión 7.2.0)** |
|---|---|---|---|
| Servidor DNS | X | X | X |
| Servidor DHCP | X | X | X |
| VPN (Ipsec y OpenVPN) | X | X | X |
| Balanceo de carga NAT | X | X | X |
| Tabla de estado | X | X | X |
| Proxy | X | X | X |
| Enrutamiento | X | X | |
| IP virtuales | X | | |
| Portal cautivo | X | | |
| Filtrado web | X | X | X |
| IPS | X | X | X |
| IDS | X | X | X |
| Antispam | X | X | X |
| Antivirus | X | X | X |
| Antiphishing | | | X |
| Servidor PPPoE | X | | |
| Control de usuarios | X | | X |
| Servidor SNMP | | X | |
| Spyware | X | | X |

*Nota.* La tabla representa una matriz comparativa sobre los servicios que ofrece cada UTM.

**Entorno de pruebas**

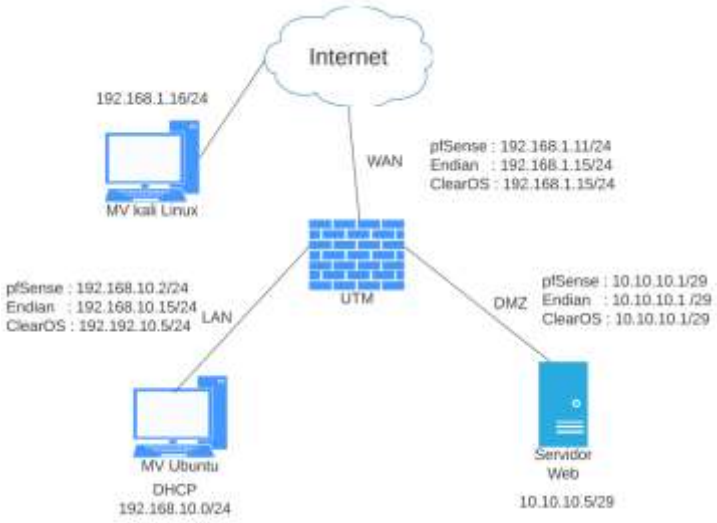
Para la creación de las máquinas virtuales se utilizó el software de virtualización VirtualBox, se utilizaron seis máquinas virtuales, donde cada una desempeña su rol dentro del entorno de pruebas.

***Topología de la red***

En la Figura 4, se observa la tipología de red para desarrollar el entorno de pruebas, la topología está basada en la red básica de una PYME, los UTM Open-Source están configurados con una red WAN, LAN y DMZ, en la red LAN se encuentra una máquina virtual con el sistema operativo Ubuntu y en la red DMZ se implementó un servidor web, todo tráfico que venga desde la WAN hacia las redes internas pasa por el UTM, la máquina virtual con Kali Linux realiza los escaneos de vulnerabilidades hacia el UTM, en cambio, la máquina de la red LAN realiza la descarga de archivos maliciosos.

**Figura 4**

*Topología de la red*




*Nota.* La figura representa la topología de red para el entorno de pruebas, esta topología es utilizada para todos los ataques, la abreviación MV significa máquina virtual.

***UTM Open-Source***

Los UTM Open-Source se instalan en una máquina virtual por separado. Para realizar una comparación equitativa, se configuraron las tres máquinas virtuales con las mismas características de hardware, y se configuraron los mismos servicios.

**Servicios configurados.** En los UTM Open-Source se configuraron los siguientes servicios:

- Antivirus.
- Servidor DHCP.
- IDS/IPS.
- Servidor Proxy.

En las figuras 5, 6 y 7, se presentan los parámetros configurados del antivirus, para los UTM pfSense, Endian y ClearOS respectivamente, los antivirus que utilizan estos UTM, están basados en un antivirus Open-Source llamado ClamAV.

**Figura 5**

*Configuración del antivirus en pfSense*

*Nota.* Parámetros para la configuración del antivirus en pfSense.

**Figura 6**

*Configuración del antivirus en Endian*


*Nota.* Parámetros para la configuración del antivirus en Endian.

**Figura 7**

*Configuración del antivirus en Endian*


*Nota.* Parámetros para la configuración del antivirus en ClearOS.

En las figuras 8, 10 y 11, se presentan los parámetros de configuración, para la instalación del servidor DHCP en pfSense, Endian y ClearOS respectivamente.

**Figura 8**

*Configuración del servidor DHCP en pfSense*


*Nota.* Parámetros para la configuración del servidor DHCP en pfSense.

**Figura 9**

*Configuración del servidor DHCP en Endian*


*Nota.* Parámetros para la configuración del servidor DHCP en Endian.

**Figura 10**

*Configuración del servidor DHCP en ClearOS*


*Nota.* Parámetros para la configuración del servidor DHCP en ClearOS.

En las figuras 11, 12 y 13, se presentan los parámetros de configuración, para la instalación del IPS/IDS en pfSense, Endian y ClearOS respectivamente.

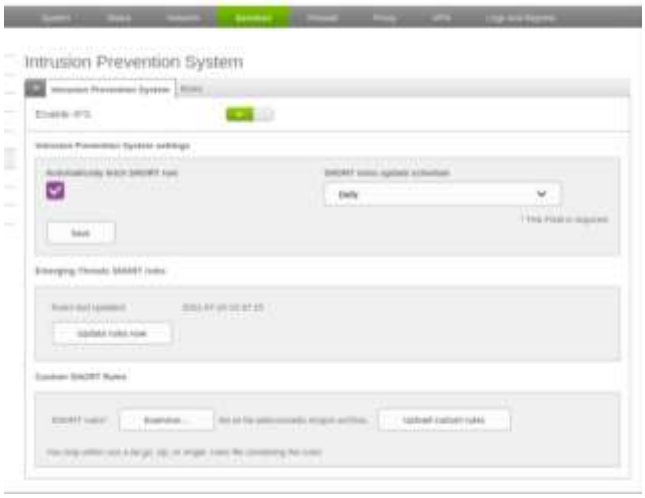
**Figura 11**

*Configuración del IPS/IDS en pfSense*


*Nota.* Parámetros para la configuración del IPS/IDS en pfSense.

**Figura 12**

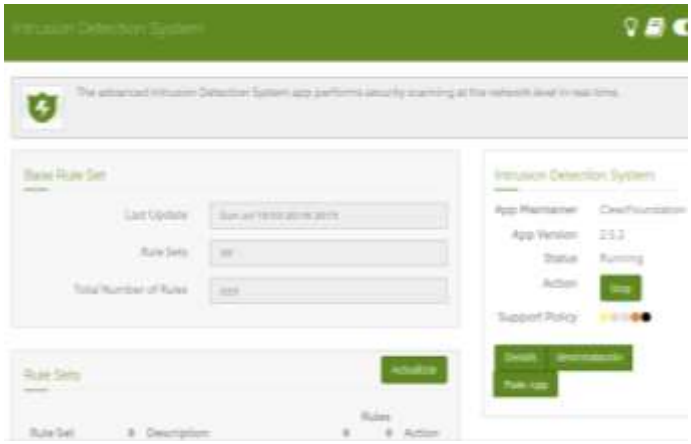
*Configuración del IPS/IDS en pfSense*




*Nota.* Parámetros para la configuración del IPS/IDS en Endian.

**Figura 13**

*Configuración del IPS/IDS en ClearOS*



*Nota.* Parámetros para la configuración del IPS/IDS en ClearOS.

En las figuras 14, 15 y 16, se presentan los parámetros de configuración, para la instalación del servidor Proxy en pfSense, Endian y ClearOS respectivamente.

**Figura 14**

*Configuración del servidor Proxy en pfSense*

*Nota.* Parámetros para la configuración del servidor Proxy en pfSense.

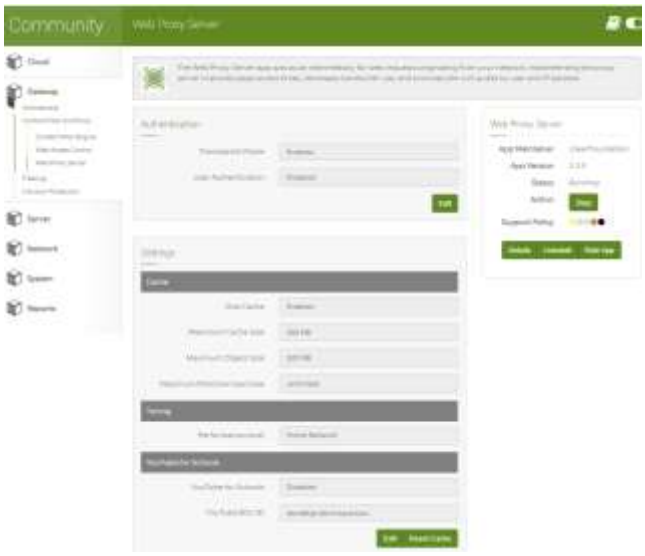
**Figura 15**

*Configuración del servidor Proxy en Endian*


*Nota.* Parámetros para la configuración del servidor Proxy en Endian.

**Figura 16**

*Configuración del servidor Proxy en ClearOS*



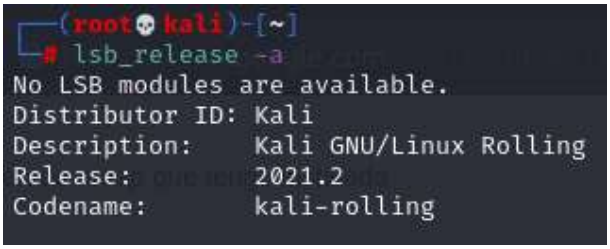

*Nota.* Parámetros para la configuración del servidor Proxy en ClearOS.

***Kali Lunix***

En esta máquina virtual se instaló el sistema operativa Kali Linux versión 2021.2, como se muestra en la Figura 17, esta máquina desarrolla el rol de atacante, se utilizó la herramienta nmap para escanear los puertos abiertos, para el escaneo de vulnerabilidades al host del UTM Open-Source y servidor web se utilizó la herramienta Nessus.

**Figura 17**

*Versión instalada de Kali Linux*



***Ubuntu Server***

Para esta máquina virtual se instaló el sistema operativo Ubuntu Server versión 20.04.1, como se presenta en la Figura 18 LTS, en este Ubuntu Server se instaló un servidor Apache versión 2.4.41, mostrada en la figura 19. Este servidor se configuró en la red DMZ.

**Figura 18**

*Versión instalada de Ubuntu Server*

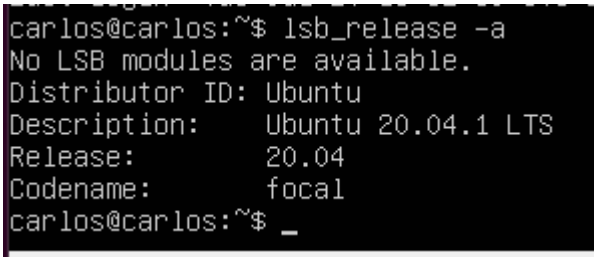



**Figura 19**

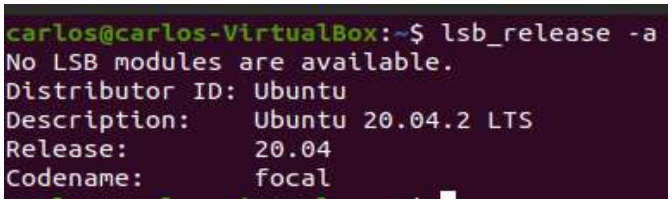
*Versión instalada del servidor Apache*


***Ubuntu Desktop***

En esta máquina virtual se instaló el sistema operativo Ubuntu versión 20.04.2, como se presenta en la Figura 20, la máquina se configuró en una red LAN, también se realizó pruebas sobre descarga de archivos maliciosos, para esto se utilizó EIACAR y HTTP Evader.

**Figura 20**

*Versión instalada de Ubuntu Desktop*



***Configuración de las máquinas virtuales***

En la Tabla 2, se muestra la configuración de las máquinas virtuales para la implementación del entorno de pruebas.

**Tabla 2**

*Configuración de las MVs para los UTM Open-Source*

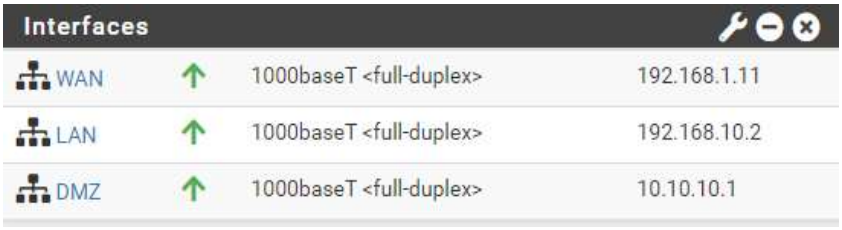
| **Máquinas Virtuales** | **Configuración** |
|---|---|
| pfSense Community Edition (versión 2.5.2)<br>Endian Firewall Community (versión 3.3.2)<br>ClearOS 7 Community Edition (versión 7.2.0) | 3 GB de Memoria RAM.<br>2 procesadores.<br>3 particiones, cada una de 16 GB.<br>3 adaptadores de red, una en adaptador puente (WAN) y los otros en red interna (DMZ, LAN). |
| Kali Linux versión 2021.2 | 2 GB de Memoria RAM.<br>1 procesador.<br>1 partición de 16 GB.<br>1 adaptador puente (WAN). |
| Ubuntu Server versión 20.04.1 LTS | 1 GB de Memoria RAM.<br>1 procesador.<br>1 partición de 10 GB.<br>1 adaptador de red en red interna (DMZ). |
| Ubuntu Desktop versión 20.04.2 | 1,5 GB de Memoria RAM.<br>1 procesador.<br>1 partición de 10 GB.<br>1 adaptador de red en red interna (LAN). |

***Interfaces de red***

En la Figura 21, se muestra la configuración de las redes WAN, LAN y DMZ en pfSense.

**Figura 21**

*Configuración de las interfaces en pfSense*

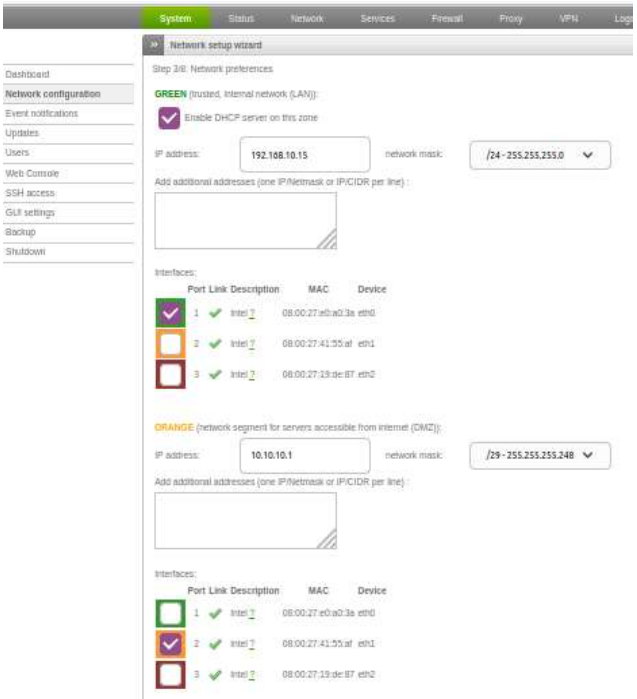



*Nota.* Configuración de las interfaces WAN, LAN y DMZ en pfSense.

En la Figura 22, se presenta la configuración de las interfaces LAN y DMZ, y en la Figura 23, se muestra la configuración de la interfaz WAN en Endian.

**Figura 22**

*Configuración de las interfaces en Endian*




*Nota.* Configuración de las interfaces LAN y DMZ en Endian.

**Figura 23**

*Configuración de las interfaces en Endian*


*Nota.* Configuración de la interfaz WAN en Endian.

En la Figura 24, se presenta la configuración de las interfaces WAN, LAN y DMZ en ClearOS.

**Figura 24**

*Configuración de las interfaces en ClearOS*


*Nota.* Configuración de las interfaces WAN, LAN y DMZ en ClearOS.

**Definición de ataques**

En este apartado, se explican los tipos de ataques que se realizan en el entorno de pruebas, se realizó un escaneo de vulnerabilidades y de puertos, también, se utilizó la página web de EICAR y HTTP Evader para descargar archivos maliciosos para probar la funcionalidad del antivirus.

***Escaneo de Vulnerabilidades***

Para el escaneo de vulnerabilidades se utilizaron las siguientes herramientas:

**Nessus Essentials.** La herramienta Nessus Essentials permite escanear hasta 16 direcciones IP por escáner, con la misma velocidad, evaluaciones y comodidad de escaneo que las versiones de paga, cuando termina un escaneo, esta herramienta separa las vulnerabilidades en los siguientes tipos: crítico, alto, medio, bajo e informativo, además, muestra los detalles de cada vulnerabilidad, en la máquina virtual Kali Linux se instaló Nessus Essentials versión 8.15 como se muestra en la Figura 25.

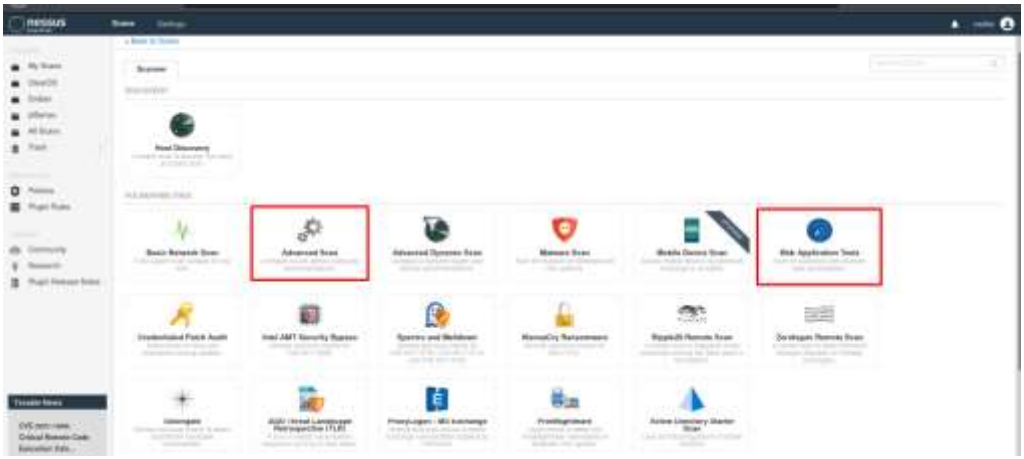
**Figura 25**

*Versión instalada de Nessus*

En la Figura 26, se presentan los diferentes tipos de escaneos. Desde la máquina virtual Kali Linux, mostrada anteriormente en la Figura 4, se realizó un escaneo avanzado y un escaneo web hacia los hosts de los UTM.

**Figura 26**

*Tipos de escaneos de Nessus Essentials*



*Nota.* Tipos de escaneos disponibles en la versión de Nessus Essentials, los dos tipos de escaneos utilizados están marcados en un cuadro rojo.

***Escaneo de puertos***

**Nmap.** Es una herramienta que permite hacer un escanear y obtener información sobre los puertos de un destino en específico, desde la máquina virtual de Kali Linux como se observa en la Figura 4, se realizó un escaneo de puertos hacia el host UTM.

***Descarga de archivos maliciosos.***

Para la descarga de archivos maliciosos se utilizaron las siguientes herramientas:

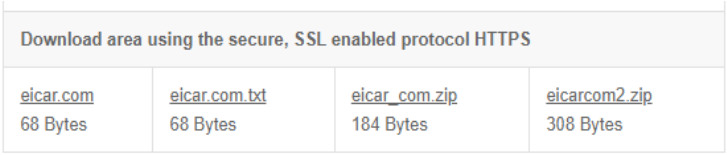
**EICAR Test.** Esta herramienta web permite probar la funcionalidad del antivirus, en la página web de EICAR hay cuatro archivos que podemos descargar, las extensiones de los archivos son:

- .com
- .com.txt
- .zip

Con la máquina virtual de Ubuntu conectada a la red LAN, como se observa en la Figura 4, se ingresa a esta página web y se intenta descargar los cuatros archivos mostrados en la Figura 27.

**Figura 27**

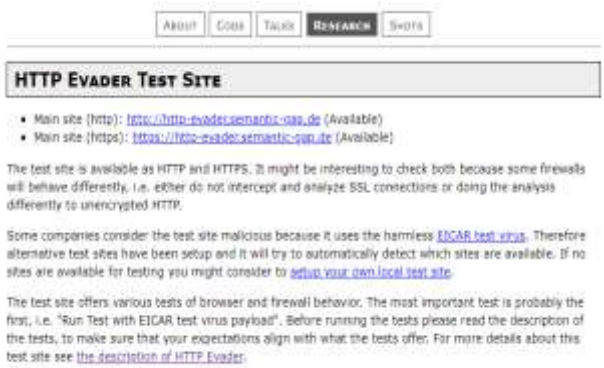
*Archivos maliciosos de EICAR test*




**HTTP Evader.** Esta herramienta web permite probar la vulnerabilidad del UTM, en la página web de HTTP Evader se puede realizar la descarga automática de malware, al terminar este proceso, la misma página nos devuelve los resultados sobre el funcionamiento del UTM, como se muestra en la figura 28, en la máquina virtual de Ubuntu conectada a la red LAN, se ingresa a la página web de HTTP Evader y se realiza la prueba de vulnerabilidad del UTM.

**Figura 28**

*Sitios web para la prueba de evasiones con http y https*



# Capítulo IV

## Introducción

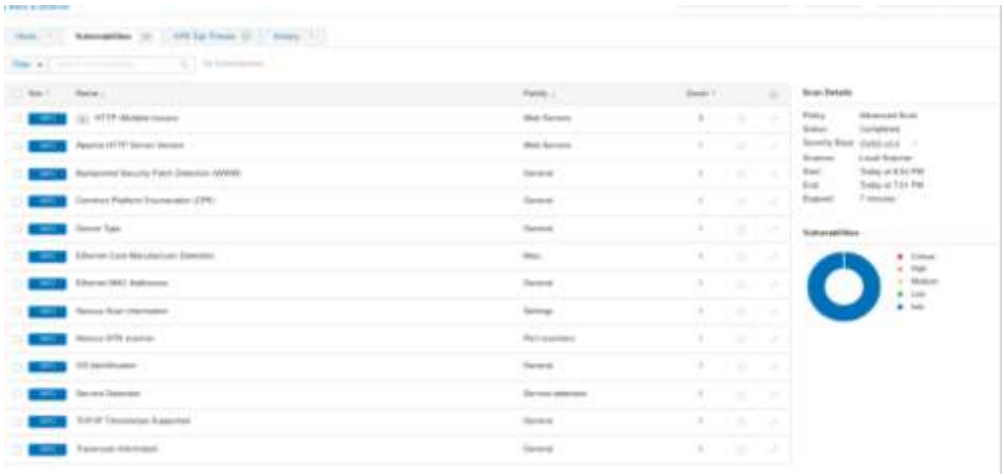
Al terminar toda la configuración del entorno de pruebas y definir los ataques, se pone en ejecución todo lo mencionado en el capítulo anterior, en el presente capítulo se describen los resultados obtenidos, con el objetivo de comparar el desempeño de los UTM Open-Source y determinar el que obtuvo mejores resultados.

## Uso de la herramienta Nessus

Primero se realizó un escaneo avanzado hacia los hosts de los UTM, en las figuras 29, 30 y 31 se observa los resultados de los escaneos avanzados hacia los UTM pfSense, Endian y ClearOS respectivamente.

**Figura 29**

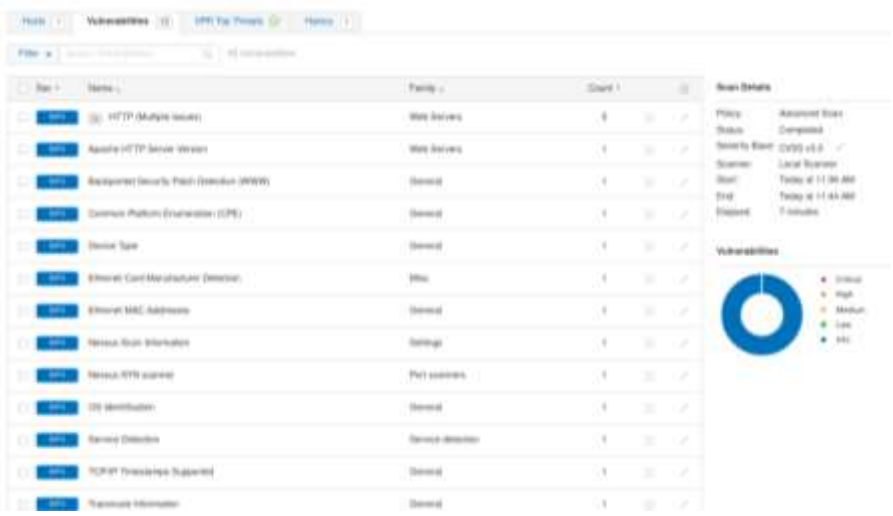
*Escaneo avanzado a pfSense*



*Nota.* Detalles de las vulnerabilidades encontradas con el escaneo avanzado hacia el host de pfSense.

**Figura 30**

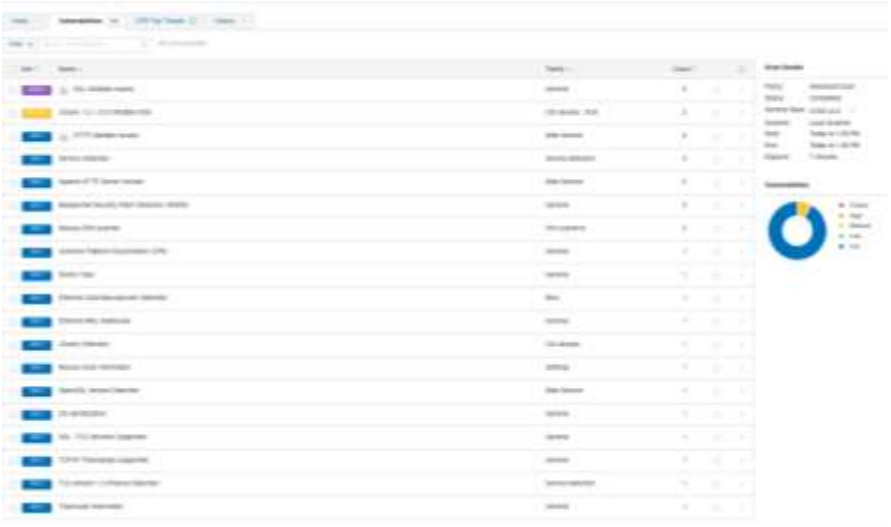
*Escaneo avanzado a Endian*



*Nota.* Detalles de las vulnerabilidades encontradas con el escaneo avanzado hacia el host de Endian.

**Figura 31**

*Escaneo avanzado a ClearOS*



*Nota.* Detalles de las vulnerabilidades encontradas con el escaneo avanzado hacia el host de ClearOS.

En resumen, los resultados obtenidos tras el escaneo avanzado con la herramienta Nessus están mostrados en la Tabla 3.

**Tabla 3**

*Escaneo de vulnerabilidades por host*

| **UTM** | **Vulnerabilidades** |
|---|---|
| pfSense Community Edition (versión 2.5.2) | 13 del tipo informativo. |
| Endian Firewall Community (versión 3.3.2) | 13 del tipo informativo. |
| ClearOS 7 Community Edition (versión 7.2.0) | 18 del tipo informativo.<br>1 del tipo media. |

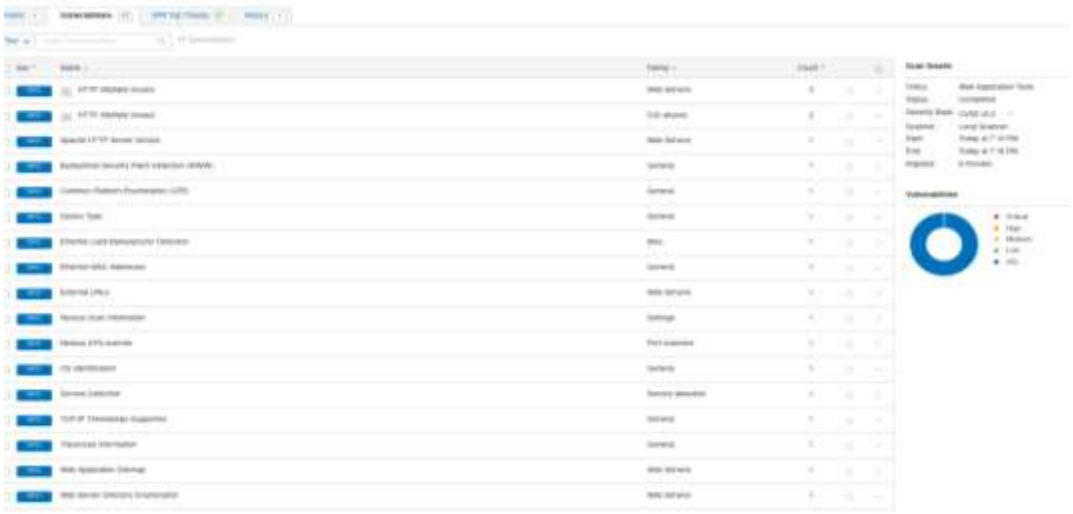
*Nota.* Resultados del escaneo web de vulnerabilidades hacia los hosts de los UTM.

Como se puede ver en la Tabla 3, en los UTM pfSense y Endian se detectaron 13 vulnerabilidades de tipo informativo, en el UTM ClearOS se detectaron 18 vulnerabilidades de tipo informativo y 1 vulnerabilidad de tipo media, en esta prueba los UTM pfSense y Endian obtuvieron los mejores resultados.

De igual forma, se realizó un escaneo web hacia los hosts de los UTM, en las figuras 32, 33 y 34 se observa los resultados de los escaneos web hacia los UTM pfSense, Endian y ClearOS respectivamente.

**Figura 32**

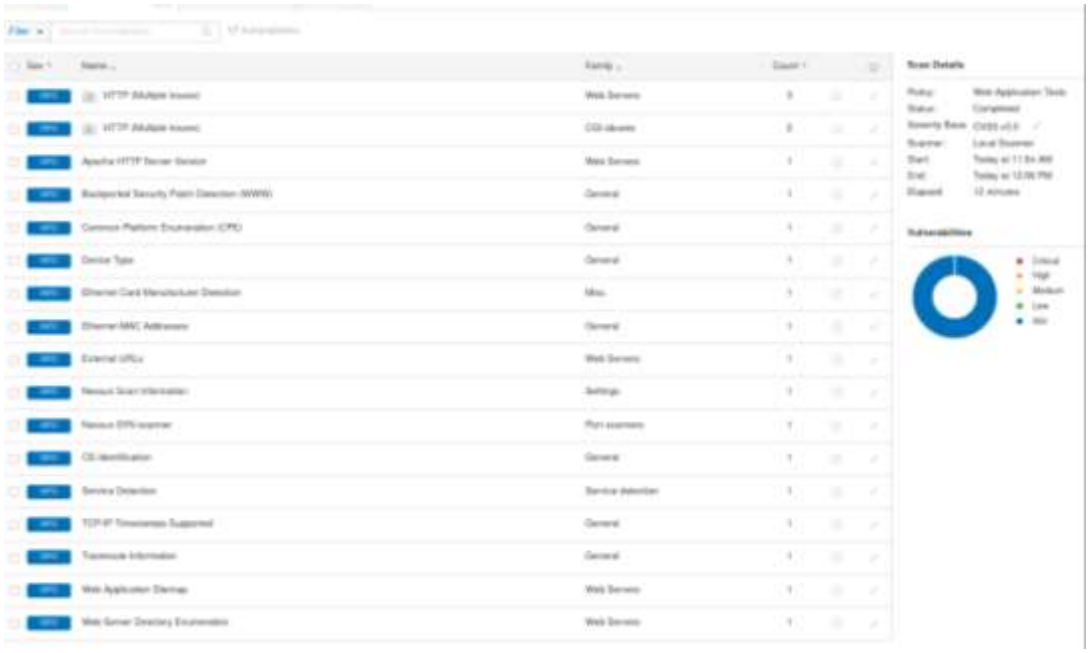
*Escaneo web a pfSense*



*Nota.* Detalles de las vulnerabilidades encontradas con el escaneo web hacia el host de pfSense.

**Figura 33**

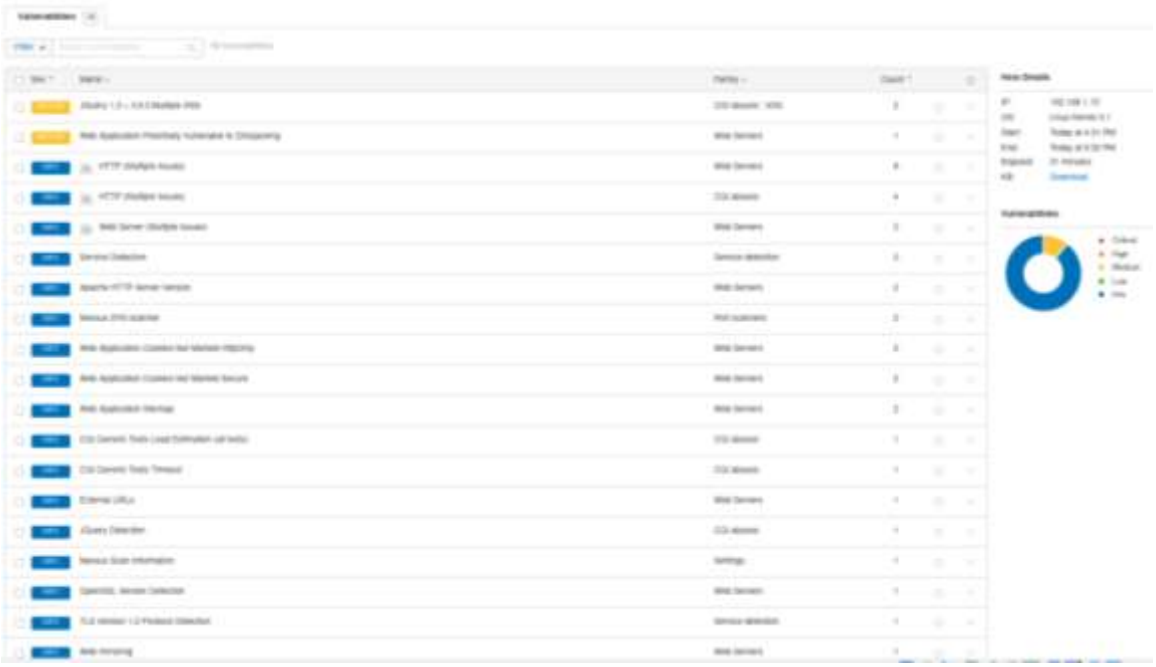
*Escaneo web a Endian*



*Nota.* Detalles de las vulnerabilidades encontradas con el escaneo web hacia el host de Endian.

**Figura 34**

*Escaneo web a ClearOS*



*Nota.* Detalles de las vulnerabilidades encontradas con el escaneo web hacia el host de ClearOS.

En definitiva, los resultados obtenidos tras el escaneo web están presentados en la Tabla 4.

**Tabla 4**

*Escaneo web de vulnerabilidades por host*

| **UTM** | **Vulnerabilidades** |
|---|---|
| pfSense Community Edition (versión 2.5.2) | 17 del tipo informativo. |
| Endian Firewall Community (versión 3.3.2) | 17 del tipo informativo. |
| ClearOS 7 Community Edition (versión 7.2.0) | 17 del tipo informativo.<br>2 de tipo media. |

*Nota.* Resultados del escaneo web de vulnerabilidades hacia los hosts de los UTM.

Como se observa en la Tabla 4, en los UTM pfSense y Endian se detectaron 17 vulnerabilidades de tipo informativo, en el UTM ClearOS se detectaron 17 vulnerabilidades de tipo informativo y 2 vulnerabilidades de tipo media, en esta prueba los UTM pfSense y Endian obtuvieron los mejores resultados.

**Escaneo de puertos nmap**

En esta prueba, se realizó un escaneo de puertos con nmap hacia los hosts de los UTM, en las figuras 35, 36 y 37 se observan los resultados de los escaneos de puertos hacia los UTM pfSense, Endian y ClearOS respectivamente.

**Figura 35**

*Escaneo de puertos a pfSense*


*Nota.* Detalles de los resultados con el escaneo de puertos hacia el host de pfSense.

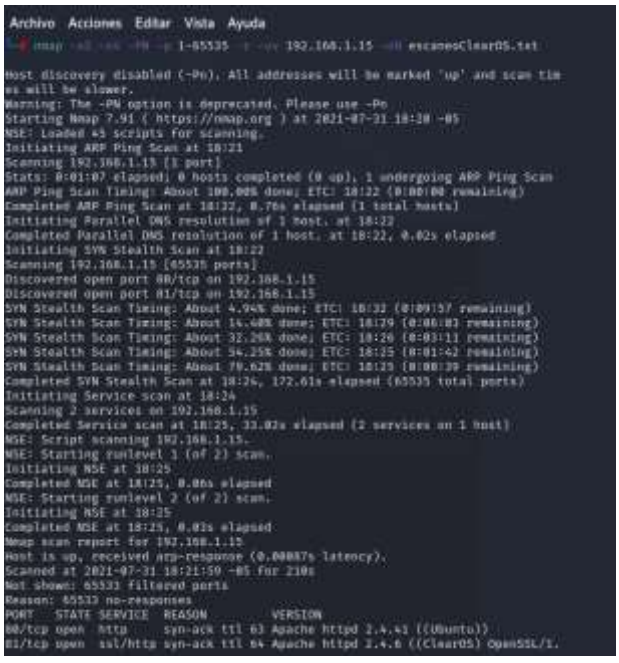
**Figura 36**

*Escaneo de puertos a Endian*


*Nota.* Detalles de los resultados con el escaneo de puertos hacia el host de Endian.

**Figura 37**

*Escaneo de puertos a ClearOS*




*Nota.* Detalles de los resultados con el escaneo de puertos hacia el host de ClearOS.

En resumen, los resultados obtenidos con nmap están presentados en la Tabla 5

**Tabla 5**

*Escaneo de puertos con nmap*

| UTM | Puerto | Estado | Servicio | Versión |
|---|---|---|---|---|
| pfSense Community Edition (versión 2.5.2) | 80 | Abierto. | http | Apache httpd 2.4.41 ((Ubuntu)). |
| Endian Firewall Community (versión 3.3.2) | 80 | Abierto. | http | Apache httpd 2.4.41 ((Ubuntu)). |
| ClearOS 7 Community Edition (versión 7.2.0) | 80 | Abierto. | http | Apache httpd 2.4.41 ((Ubuntu)). |
| | 81 | Abierto. | ssl/http | Apache httpd 2.4.6 ((ClearOS) OpenSSL/1. |

*Nota.* Resultados del escaneo de puertos con nmap.

Como se muestra en la Tabla 5, en los UTM pfSense y Endian se detectó el puerto 80 en estado abierto con el servicio http en la versión Apache httpd 2.4.41 ((Ubuntu)), en el UTM ClearOS se detectaron dos puertos abiertos, el puerto 80 con servicio http en la versión Apache httpd 2.4.41 ((Ubuntu)), y el puerto 81 con servicio ssl/http con la versión Apache httpd 2.4.6 ((ClearOS) OpenSSL/1, en esta prueba los UTM pfSense y Endian obtuvieron los mejores resultados.

**Descarga de archivos maliciosos con EICAR test**

Para esta prueba, se realizó la descarga de archivos maliciosos con EICAR test, en las Figuras 38, 39, 40 y 41 se muestra los resultados al descargar los archivos malasios con la protección de pfSense.

**Figura 38**

*Descarga de archivo malicioso .com con uso de pfSense*


*Nota.* Resultado de la descarga del archivo malicioso llamado eicar.com con la protección de pfSense.

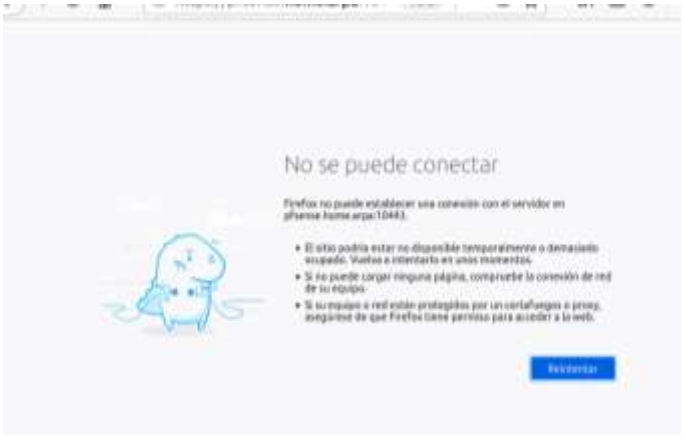
**Figura 39**

*Descarga de archivo malicioso .txt con uso de pfSense*


*Nota.* Resultado de la descarga del archivo malicioso llamado eicar.com.txt con la protección de pfSense.

**Figura 40**

*Descarga de archivo malicioso .zip con uso de pfSense*




*Nota.* Resultado de la descarga del archivo malicioso llamado eicar_com.zip con la protección de pfSense.

**Figura 41**

*Descarga de archivo malicioso .zip con uso de pfSense*


*Nota.* Resultado de la descarga del archivo malicioso llamado eicarcom2.zip con la protección de pfSense.

En las Figuras 42, 43, 44 y 45 se presenta los resultados al descargar los archivos maliciosos con la protección de Endian.

**Figura 42**

*Descarga de archivo malicioso .com con el uso de Endian*


*Nota.* Resultado de la descarga del archivo malicioso llamado eicar.com con la protección de Endian.

**Figura 43**

*Descarga de archivo malicioso .txt con el uso de Endian*


*Nota.* Resultado de la descarga del archivo malicioso llamado eicar.com.txt con la protección de Endian.

**Figura 44**

*Descarga de archivo malicioso .zip con el uso de Endian*


*Nota.* Resultado de la descarga del archivo malicioso llamado eicar_com.zip con la protección de Endian.

**Figura 45**

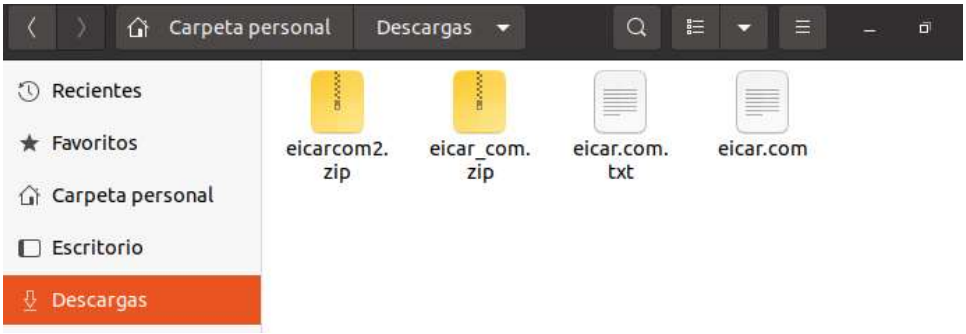
*Descarga de archivo malicioso .zip con el uso de Endian*


*Nota.* Resultado de la descarga del archivo malicioso llamado eicarcom2.zip con la protección de Endian.

En la Figura 46, se muestra la descarga exitosa de los archivos maliciosos. El UTM ClearOS no bloqueó las descargas.

**Figura 46**

*Descarga de archivos maliciosos con el uso de ClearOS*




*Nota.* Resultado de las descargas de los archivos maliciosos con la protección de ClearOS.

En resumen, los resultados obtenidos de la descarga de archivos maliciosos con EICAR test están presentados en la Tabla 6.

**Tabla 6**

*Descarga de archivos con EICAR test*

| **Archivos de prueba** | **pfSense Community Edition (versión 2.5.2)** | **Endian Firewall Community (versión 3.3.2)** | **ClearOS 7 Community Edition (versión 7.2.0)** |
|---|---|---|---|
| eicar.com | Bloqueado. | Bloqueado. | No bloqueado. |
| eicar.com.txt | Bloqueado. | Bloqueado. | No bloqueado. |
| eicar_com.zip | Bloqueado. | Bloqueado. | No bloqueado. |
| eicarcom2.zip | Bloqueado. | Bloqueado. | No bloqueado. |

*Nota.* Resultados de las descargas de archivos maliciosos con EICAR test.

En la Tabla 6, Los UTM pfSense y Endian bloquearon la descarga de todos los archivos maliciosos, pero el UTM ClearOS no bloqueó las descargas de los archivos maliciosos, pfSense y Endian obtuvieron los mejores resultados.

**Pruebas de evasión de firewall con HTTP Evader**

En esta prueba, se utilizó el sitio web HTTP Evader para detectar evasiones de archivos maliciosos, en las figuras 47 y 48 se muestra los resultados de las evasiones http y https con pfSense.

**Figura 47**

*Evasiones http con pfSense*


*Nota.* Resultados de las evasiones http con la protección de pfSense.

**Figura 48**

*Evasiones https con pfSense*


*Nota.* Resultados de las evasiones https con la protección de pfSense.

En las figuras 49 y 50 se muestran los resultados de las evasiones http y https con Endian.

**Figura 49**

*Evasiones http con Endian*


*Nota.* Resultados de las evasiones http con la protección de Endian.

**Figura 50**

*Evasiones https con Endian*


*Nota.* Resultados de las evasiones https con la protección de Endian.

En las figuras 51 y 52 se muestran los resultados de las evasiones http y https con ClearOS.

**Figura 51**

*Evasiones http con ClearOS*


*Nota.* Resultados de las evasiones http con la protección de Endian.

**Figura 52**

*Evasiones https con ClearOS*


*Nota.* Resultados de las evasiones https con la protección de ClearOS.

Los resultados obtenidos con HTTP Evader están presentados en la Tabla 7.

**Tabla 7**

*Evasiones con HTTP Evader*

| **UTM** | **Evasiones HTTP** | **Evasiones HTTPS** |
|---|---|---|
| pfSense Community Edition (versión 2.5.2) | 55 | 55 |
| Endian Firewall Community (versión 3.3.2) | 21 | 21 |
| ClearOS 7 Community Edition (versión 7.2.0) | 23 | - |

*Nota.* Resultados de las evasiones con el sitio web HTTP Evader.

Como se observa en la Tabla 7, el UTM pfSense obtuvo 55 evasiones con el protocolo HTTP y HTTPS, el UTM Endian obtuvo 21 evasiones con el protocolo HTTP y HTTPS, el UTM ClearOS obtuvo 23 evasiones con HTTP, pero con el HTTPS no se obtuvo resultados porque el UTM ClearOS bloquea el sitio web donde se realiza la prueba de evasiones, en esta prueba Endian obtuvo los mejores resultados.

# Capítulo V

## Conclusiones

Se realizó el análisis y selección de un UTM Open-Source, para fortalecer la seguridad de la información en las PYMES, en el análisis de los tres UTM Open-Source, se determinó que el UTM Open-Source con los mejores resultados fue Endian. En la fase de pruebas, los resultados demostraron que pfSense y Endian son los más efectivos para la protección de una red, pero Endian sobresalió en una prueba, la cual consistió en la descarga de archivos maliciosos.

Con la revisión sistemática de literatura de la documentación y de proyectos desarrollados sobre la implementación de los UTM Open-Source, se determinó las tres mejores soluciones. Además, se realizó una matriz comparativa entre los tres UTM Open-source, para ser analizados en diferentes tipos de ataques.

Se implementó tres UTM Open-Source y se configuró los mismos servicios e interfaces de red. No se presentaron problemas en el desarrollo de las pruebas realizadas hacia la red de pfSense y Endian. Sin embargo, en la prueba con HTTP Evader, ClearOS bloqueó las direcciones web, que se utilizan para realizar esta prueba, pero el sitio web de HTTP Evader proporcionó una dirección web adicional con el protocolo http, debido a este inconveniente, no se obtuvieron resultados de este ataque con el protocolo https en ClearOS.

Se demostró la potencialidad de implementar un sistema UTM Open-Source y sus beneficios. Con las pruebas ejecutadas pfSense y Endian obtuvieron los mejores resultados, pero en la prueba de HTTP Evader con protocolo http y https, Endian consiguió los mejores resultados.

**Recomendaciones**

Tomar como referencia proyectos relacionados con los UTM Open-Source, para conocer las soluciones más eficaces y que mejor desempeño obtienen en diferentes implementaciones, sobre todo, conocer los servicios y la usabilidad en las configuraciones para cada UTM Open-Source.

Las configuraciones de los UTM Open-Source se deben realizar de acuerdo a las necesidades de cada PYME, los UTM deben ser implementadas en una máquina con las mejores características en lo que respecta al hardware, para que estas soluciones obtengan un mejor desempeño en la protección de la red.

Implementar un UTM Open-Source, para fortalecer la seguridad de las PYMES. Basándose en los resultados del presente proyecto, Endian y pfSense son las mejores soluciones para la protección de la red en una PYME.

**Bibliografía**

Alshalan, A., Pisharody, S., & Huang, D. (2016). A Survey of Mobile VPN Technologies. *IEEE Communications Surveys and Tutorials*, *18*(2), 1177–1196. https://doi.org/10.1109/COMST.2015.2496624

Arunwan, M., Laong, T., & Atthayuwat, K. (2016). Defensive performance comparison of firewall systems. *2016 Management and Innovation Technology International Conference, MITiCON 2016*, MIT221–MIT224. https://doi.org/10.1109/MITICON.2016.8025212

Asghari, V., Amiri, S., & Amiri, S. (2016). Implementing UTM based on PfSense platform. *Conference Proceedings of 2015 2nd International Conference on Knowledge-Based Engineering and Innovation, KBEI 2015*, 1150–1152. https://doi.org/10.1109/KBEI.2015.7436210

ClearOS. (2021). *Comunidad de ClearOS Server - ClearUnited*. Www.Clearos.Com. https://www.clear.store/products/z-clearos-7-community

Efthymiopoulos, M. P. (2019). A cyber-security framework for development, defense and innovation at NATO. *Journal of Innovation and Entrepreneurship*, *8*(1), 1–26. https://doi.org/10.1186/s13731-019-0105-z

Endian. (2021). *UTM de código abierto y firewalls | Comunidad de Firewall de Endian*. Www.Endian.Com. https://www.endian.com/community/

Flickenger, R. (2009). *Linux Server Hacks: 100 Industrial-Strength Tips and Tools* (I. O'Reilly Media (ed.)). http://www.amazon.com/Linux-Server-Hacks-Industrial-Strength-Tools-ebook/dp/B0043D2EA4

Freed, N. (2000). *Behaviour of and requirements for Firewalls*. 1–7. https://www.rfc-editor.org/pdfrfc/rfc2979.txt.pdf

Hamid, A., Abdullah, N. L., & Idrus, R. (2016). Framework for successful Open-Source Software implementation in the Malaysian Public Sector. *4th IGNITE Conference and 2016 International Conference on Advanced Informatics: Concepts, Theory and Application, ICAICTA 2016*. https://doi.org/10.1109/ICAICTA.2016.7803143

Iriarte Solís, A., Velarde Alvarado, P., Aguirre Villaseñor, A., Mena Camaré, L. J., Martínez Peláez, R., & Ochoa Brust, A. M. (2018). *Evaluación De Firewalls Basados En Software Libre*. *40*(130), 625–637.

ISO/IEC 27000. (2018). International Standard ISO / IEC Information technology — Security techniques — Information security management systems — Overview and vocabulary. *ACM Workshop on Formal Methods in Security Engineering.Washington, DC, USA*, *34*(19), 45–55. http://www.worldcat.org/title/service-operation/oclc/254028066&referer=brief_results%0Ahttps://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-53r4.pdf%0Ahttps://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-37r2.pdf%0Ahttp://k504.kh

ISO/IEC 27002. (2013). ISO/IEC 27002:2013. *Iec*, *2013*, 90. www.iso.org

Kumar, M., & Singh, A. K. (2020). Distributed Intrusion Detection System using Blockchain and Cloud Computing Infrastructure. *Proceedings of the 4th International Conference on Trends in Electronics and Informatics, ICOEI 2020*, 248–252. https://doi.org/10.1109/ICOEI48184.2020.9142954

Ma, L., Chen, Y., Sun, Y., & Wu, Q. (2012). Virtualization maturity reference model for green software. *Proceedings - 2012 International Conference on Control Engineering and Communication Technology, ICCECT 2012*, 573–576. https://doi.org/10.1109/ICCECT.2012.230

Maata, R. L., Cordova, R., Sudramurthy, B., & Halibas, A. (2018). Design and Implementation of Client-Server Based Application Using Socket Programming in a Distributed Computing Environment. *2017 IEEE International Conference on Computational Intelligence and Computing Research, ICCIC 2017*, 1–4. https://doi.org/10.1109/ICCIC.2017.8524573

Miguez, G. (2017). *Implementación de un Sistema de Gestión Unificada de Amenazas (UTM) para la Empresa de Créditos Palacio del Hogar*. 185.

Neupane, K., Haddad, R., & Chen, L. (2018). Next Generation Firewall for Network Security: A Survey. *Conference Proceedings - IEEE SOUTHEASTCON*, *2018-April*, 1–6. https://doi.org/10.1109/SECON.2018.8478973

Pablo, D., & Loor, L. (2017). *Sistema Perimetral Firewall Y Fortalecimiento De La Seguridad En El Data Center De La Espam MFL*. 18–27.

pfSense. (2021). *Introducción al software pfSense*. Www.Pfsense.Org. https://www.pfsense.org/getting-started/

Pratama, R. F., Suwastika, N. A., & Nugroho, M. A. (2018). Design and implementation adaptive Intrusion Prevention System (IPS) for attack prevention in software-defined network (SDN) architecture. *2018 6th International Conference on Information and Communication Technology, ICoICT 2018*, *c*, 299–304. https://doi.org/10.1109/ICoICT.2018.8528735

Qi, Y., Yang, B., Xu, B., & Li, J. (2007). Towards system-level optimization for high performance unified threat management. *3rd International Conference on Networking and Services,ICNS 2007*, 2–7. https://doi.org/10.1109/ICNS.2007.126

Razzaq, S., Li, Y. F., Lin, C. T., & Xie, M. (2018). A Study of the Extraction of Bug Judgment and Correction Times from Open Source Software Bug Logs. *Proceedings - 2018 IEEE 18th International Conference on Software Quality, Reliability, and Security Companion, QRS-C 2018*, 229–234. https://doi.org/10.1109/QRS-C.2018.00050

Santoso, G. Z., Jung, Y. W., & Kim, H. Y. (2014). Analysis of Virtual Machine Monitor as Trusted Dependable Systems. *Proceedings - 2014 IEEE International Conference on Ubiquitous Intelligence and Computing, 2014 IEEE International Conference on Autonomic and Trusted Computing, 2014 IEEE International Conference on Scalable Computing and Communications and Associated Sy*, 603–608. https://doi.org/10.1109/UIC-ATC-ScalCom.2014.32

Senthilkumar, P., & Muthukumar, M. (2018). A study on firewall system, scheduling and routing using pfsense scheme. *Proceedings of IEEE International Conference on Intelligent Computing and Communication for Smart World, I2C2SW 2018*, 14–17.

https://doi.org/10.1109/I2C2SW45816.2018.8997167

Soloviev, V. I. (2008). Mathematical modeling of strategic commitments and piracy in windows/linux competition. *2008 International Conference on Management Science and Engineering 15th Annual Conference Proceedings, ICMSE*, 10–12. https://doi.org/10.1109/ICMSE.2008.4668886

SRI. (2012). *Servicio de Rentas Internas*. Http://Www.Sri.Gob.Ec. https://www.sri.gob.ec/de/home

Thaler, D. (2014). *Reflections on Host Firewalls*. 1–13. https://www.rfc-editor.org/pdfrfc/rfc7288.txt.pdf

Zambrano, P., & Sánchez, M. (2013). *REPOTENCIACION DE UN SISTEMA DE FIREWALL DE CODIGO ABIERTO BASADO EN FUNCIONALIDADES DE PLATAFORMA PROPIETARIA*. 1–122. http://repositorio.espe.edu.ec/bitstream/21000/7039/1/T-ESPE-047113.pdf