UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI

FACULTAD DE INDUSTRIAS AGROPECUARIAS Y CIENCIAS AMBIENTALES

CARRERA DE INGENIERÍA EN INFORMÁTICA

Tema: "Pruebas de penetración para la seguridad informática al servidor web del laboratorio de ciberseguridad en la Universidad Politécnica Estatal del Carchi"

Trabajo de titulación previa la obtención del
título de Ingeniero en Informática

AUTOR: Castillo Enríquez Alvaro Steebe

TUTOR: Msc. Hidalgo Guijarro Jairo Vladimir

Tulcán, 2021

CERTIFICADO JURADO EXAMINADOR

Certificamos que el estudiante **Castillo Enríquez Alvaro Steebe** con el número de cédula **0402039614** ha elaborado el trabajo de titulación**: "Pruebas de penetración para la seguridad informática al servidor web del laboratorio de ciberseguridad en la Universidad Politécnica Estatal del Carchi".**

Este trabajo se sujeta a las normas y metodología dispuesta en el Reglamento de Titulación, Sustentación e Incorporación de la UPEC, por lo tanto, autorizamos la presentación de la sustentación para la calificación respectiva.

Firmado electrónicamente por:
JAIRO VLADIMIR
HIDALGO
GUIJARRO

f.....

Hidalgo Guijarro Jairo Vladimir

TUTOR

Tulcán, septiembre de 2021

AUTORÍA DE TRABAJO

El presente trabajo de titulación constituye requisito previo para la obtención del título de **Ingeniero** en la Carrera de ingeniería en informática de la Facultad de Industrias Agropecuarias y Ciencias Ambientales

Yo, Castillo Enríquez Alvaro Castillo con cédula de identidad número 0402039614 declaro: que la investigación es absolutamente original, auténtica, personal y los resultados y conclusiones a los que he llegado son de mi absoluta responsabilidad.

f.....

Castillo Enríquez Alvaro Steebe

AUTOR

Tulcán, septiembre de 2021

ACTA DE CESIÓN DE DERECHOS DEL TRABAJO DE TITULACIÓN

Yo, Castillo Enríquez Alvaro Steebe declaro ser autor/a de los criterios emitidos en el trabajo de investigación: “Pruebas de penetración para la seguridad informática al servidor web del laboratorio de ciberseguridad en la Universidad Politécnica Estatal del Carchi” y eximo expresamente a la Universidad Politécnica Estatal del Carchi y a sus representantes legales de posibles reclamos o acciones legales.

f.....

Castillo Enríquez Alvaro Steebe

AUTOR

Tulcán, septiembre de 2021

AGRADECIMIENTO

Agradezco a la vida, que me permitió día a día darme una oportunidad de seguir luchando por mis sueños, a Dios por el cual tuve fe de creer en mí mismo y hacer que las cosas se cumplieran con éxito.

A mi madre quien con su sabiduría, paciencia, aliento, apoyo incondicional y amor supo estar siempre para mí, quien sin duda creyó en mi capacidad de lograrlo todo, motivándome a ser mejor que ayer. A mi padre quien me enseñó a ser fuerte y hacer las cosas correctamente con el fin de ayudar a los demás. A mis hermanos quienes creyeron en mí y estuvieron en mis momentos más complicados.

A mi compañera de vida Ana, quien estuvo presente en todos mis logros, mis tristezas, horas sin dormir, quien me acompaño todas las noches sentada junto a mí a terminar mi proyecto y sobre todo quien me enseñó a valorar cada minuto de mi vida y de mis esfuerzos, a entender que las cosas se dan por alguna razón y que siempre habrá algo por quien luchar y por quien cumplir nuestros sueños.

A mis docentes y universidad que, con su experiencia, dedicación y principalmente amor a su trabajo me han brindado conocimientos necesarios para ser un profesional de calidad y sobre todo tener esa aptitud humana para servir a la comunidad con todo lo aprendido.

A mi tutor quien me acompaño en este proceso de investigación, por su guía constante, experiencia, virtud y conocimientos que día a día impartió para que podamos salir adelante como profesionales.

Finalmente, a mis verdaderos amigos que estuvieron ahí dándome la mano con sus conocimientos, sabiduría, reflexión y sobre todo que me han sacado una sonrisa a lo largo de toda mi etapa universitaria.

DEDICATORIA

A mi madre

Por inculcar en mí el valor de la dedicación, responsabilidad y perseverancia.

A mi padre

Por enseñarme a ser humilde y hacer siempre las cosas con amor.

A mi novia

Por ser mi apoyo incondicional en los momentos más difíciles y no dejar que me rinda. Te amo.

A mis hermanos

Por estar siempre presentes en mis pensamientos y darme ese motivo para seguir adelante.

A mis mascotas

Que me acompañaron toda las noches con su amor incondicional. Los amo Shazam, Luna y Luci.

ÍNDICE

ÍNDICE DE FIGURAS

ÍNDICE DE TABLA

ÍNDICE DE ANEXOS

RESUMEN

La presente investigación denominada “Pruebas de penetración para la seguridad informática al servidor web del laboratorio de ciberseguridad en la Universidad Politécnica Estatal del Carchi”, como objetivo principal del proyecto de investigación fue diagnosticar las vulnerabilidades en los servidores web de Apache sobre Centos 7.0 y Microsoft IIS sobre Windows Server 2016 a través de herramientas de pruebas de penetración (Pentest) utilizando herramientas que se encuentran instaladas y configuradas sobre el sistema operativo Kali-Linux el cual realiza los mejores procesos de auditoría y ciberseguridad con el fin conocer los riesgos, amenazas y su relación con los procesos de seguridad.

Para dar cumplimiento a esta meta se planteó un enfoque cualitativo en conjunto con la investigación de campo y documental. A partir de los resultados obtenidos se elaboró un cuadro comparativo de los servidores con más desempeño, rendimiento y seguridad a la hora de analizar sus características, como también la realización de un test de resultados con las vulnerabilidades producidas en el servidor web y su aplicación, para la realización de la propuesta se empleó la metodología Owasp (“Open web Application Security Project”) donde se definió los instrumentos con más utilidad para el desarrollo del proyecto, además se determinó por medio de un informe al sitio web de orientación profesional upec que puede contar con medios necesarios para adoptar estos análisis de vulnerabilidad a cualquier sitio que sea requerido. Finalmente en el ámbito práctico se estableció varias pruebas con Owasp Zap como herramienta principal para encontrar alertas de amenazas, WPScan como un escáner de vulnerabilidad para el gestor de contenido Wordpress enfocado a la creación de páginas web. En este sentido se estableció una comparativa de los servidores web con un valor alcanzado del 80% para Apache y el 30% para Microsoft IIS, como también una comparación final de las vulnerabilidades del 5,33% para manejo, configuración y desarrollo, 8% manejo de identidad y método http, 7% fuerza bruta y Cross Site Scripting, 5% inyección SQL y DoS y finalmente 4,67% Owasp Zap/directorios. El uso de estas técnicas fusionado con la gestión de las fases de Owasp permitió organizar y orientar de manera rápida y confiable técnicas básicas para proteger contra amenazas comunes e importantes, obteniendo como referencia la documentación generada que puede ser reutilizable para proyectos futuros o en trabajos de implementación.

Palabras clave: vulnerabilidad, seguridad, servidor web, sitios web, Owasp

ABSTRACT

This research named “Penetration testing for computer security to the web server of the cybersecurity laboratory at Universidad Politécnica Estatal del Carchi” has as a main objective to diagnose vulnerabilities in Apache web servers on Centos 7.0 and Microsoft IIS on Windows Server 2016 through penetration testing tools (Pentest), tools that are installed and configured on the Kali-Linux operating system were used, it performs the best auditing and cybersecurity processes to know the risks, threats and their relationship with the security processes. In order to achieve this goal, a qualitative approach, field and documentary research were proposed. A comparative table of the servers with the highest output was drawn up from the results obtained, throwput and safety when their features were analyzed as well as carrying out a test of results with the vulnerabilities produced in the web server and its application. The Owasp methodology (“Open web Application Security Project”) was used to carry out the proposal and the most useful instruments for the development of the project were defined. It was also determined through a report to the professional guidance website UPEC that it may have the necessary means to adopt these vulnerability analyzes to any site that is required. Finally, in the practical area, several tests were established with Owasp Zap as the main tool to find threat alerts, WPScan as a vulnerability scanner for the Wordpress content manager focused on creating web pages. In this sense, a comparison of web servers was established with a value reached of 80% for Apache and 30% for Microsoft IIS, as well as a final comparison of the vulnerabilities of 5.33% for management, configuration and development, 8% identity management and http method, 7% brute force and Cross Site Scripting, 5% SQL injection and DoS and finally 4.67% Owasp Zap / directories. The use of these techniques merged with the management of the Owasp phases allowed to organize and guide basic techniques quickly and reliably to protect against common and important threats having as reference the documentation generated that can be reusable for future projects or implementation work.

Key words: Vulnerability, safety, Web server, websites, Owasp

INTRODUCCION

En la actualidad organizaciones, laboratorios y páginas web que brindan servicios a los usuarios procesan a diario una gran cantidad de amenazas y vulnerabilidades por parte de atacantes informáticos que tratan de robar, alterar y sacar provecho de la información, partiendo de esta necesidad se enfocan cada vez más en el desarrollo de sistemas tecnológicos y la aplicación de técnicas de seguridad para el cuidado de los datos de la organización y así proteger la ejecución de sus procesos que ofrece a sus usuarios.

El laboratorio de ciberseguridad es una de las áreas de la Universidad Politécnica Estatal del Carchi que se encarga de realizar pruebas de investigación por las configuraciones y servicios que se quieran desarrollar, actualmente cuentan con equipamientos e infraestructura que permiten a los estudiantes y docentes realizar pruebas orientadas a la seguridad de los dispositivos, configuraciones de redes, entre otros, en este sentido se permitió la configuración a los servidores que se encontraban con malos procesos de seguridad, además la subutilización de recursos no eran adecuadas provocando vulnerabilidad a los sistemas.

Por lo tanto el estudio tiene como objetivo principal examinar los problemas de seguridad que se desarrollaron en los servidores web y trazar un marco teórico y metodológico que sirva como base para el diagnóstico de las vulnerabilidades presentes en los servidores web aplicando metodologías y técnicas que se adapten a las necesidades de su ejecución de manera organizada.

La importancia de esta investigación se fundamenta en el provecho de conocimiento sobre herramientas utilizadas para el análisis de vulnerabidades y fases que documenten de una mejor manera sus procesos, traducido en la elaboración metodológica Owasp con el fin de identificar y proteger contra debilidades comunes e importantes.

El enfoque cualitativo de investigación permitió analizar y dimensionar las variables de estudio sobre la seguridad al servidor web y la vulnerabilidad al sistema. Se estableció la modalidad de campo, descriptiva y exploratoria para recolectar información y a través de la entrevista al administrador de laboratorio de ciberseguridad se identificó conceptos, procesos y el manejo aplica, con esto se cuantificó los indicadores y se determinó la viabilidad del proyecto.

La construcción de la propuesta está totalmente enfocada al análisis de vulnerabilidad a los servidores web donde fue guiada por metodologías y la información recolectada con los instrumentos de investigación, dando lugar al manejo de técnica y herramientas que se centran en el sistema operativo Kali-Linux para asegurar la coherencia entre los componentes.

I. PROBLEMA

1.1. PLANTEAMIENTO DEL PROBLEMA

En el año 2020 el impacto global fue totalmente diferente a los años anteriores, impregnado el virus en todos los aspectos de nuestras vidas. Los estafadores aprovecharon esta oportunidad para vulnerar y atacar el sistema de organizaciones, así como también a usuarios, Internet Crime Complaint Center (IC3) en el año 2020 recibió más de 28,500 quejas; por lo que afirmó que los cinco principales delitos informáticos fueron encabezados por la técnica de phishing o suplantación de identidad con 241,342 quejas, seguido por delito con 108,869, extorción con 76,741, información personal en incumplimiento y robo de identidad con 45,350 y 43,330 respectivamente (IC3, 2020). Por esta razón organizaciones tanto públicas como privadas se vieron a la necesidad de incrementar su nivel de seguridad informática.

En Sudamérica se realizó un estudio enfocado a los problemas latentes de inseguridad informática y el robo de la información, vulnerabilidad, hackeo, phishing entre otras amenazas en organizaciones financieras y de otra índole. Esta investigación de acuerdo con el Índice Global de Ciberseguridad (IGC) en el Ecuador ocupa el puesto 79 de 127 países respecto a la seguridad en el ranking internacional relacionado a la vulnerabilidad y evaluación de riegos con un indicador del 37% de seguridad y ocupando el puesto 74 para el año 2020 frente a la pandemia del Covid-19 evaluado por (Deep Knowledge Group, 2020). En este sentido existe un elevado índice de inseguridad informática posicionándolo en el sexto lugar de 19 países con un indicador del 31,57% de seguridad que se encuentra por debajo de los países como Perú, Venezuela, Chile, Paraguay, El salvador, Nicaragua y Bolivia de acuerdo con la Unión Internacional de Telecomunicaciones (ITU) (Troein y Acayo, 2020).

Según Enríquez (2015) en Ecuador desde el año 2009 se han ocultado muchas denuncias relacionadas con el robo de contraseñas, clonación de tarjetas, ataques a servidores y páginas web gubernamentales, falsificación o fraude informático, entre otros delitos. En efecto, a finales del año 2014 se aprobó el nuevo Código Orgánico Integral Penal (COIP), el mismo que contiene artículos que sancionan directamente los ilícitos informáticos. El conflicto es que dicha legislación aún no es socializada, aunque es responsabilidad de todas las personas el conocer y acatar las leyes impuestas en el país, de tal manera que hay un bajo interés en conocer tales acuerdos, como también el mal uso de la tecnología informática por parte de usuarios, donde

no es empleada ninguna estrategia de seguridad informática, exponiendo fácilmente a ataques informáticos.

En el Ecuador, existe situaciones similares de vulnerabilidad y amenazas cibernéticas que sufren los servidores. Según datos recopilados por el equipo de respuestas del Centro de Reacción a Incidentes Informáticos (CSIRT) que se encarga a la detección, prevención y gestión de los incidentes de seguridad, el país registra un porcentaje del 46,4% de vulnerabilidad en el uso de la seguridad, siendo muy baja en comparación con el resto de los países, esta vulnerabilidad es provocada por fenómenos externos tales como phishing, bots, hackeo, spam entre otros. En el estudio en el cual se determina que, la vulnerabilidad al sistema informático es una debilidad que pone en peligro la seguridad a la información, afectando a los servidores tecnológicos tanto web, red y de datos de una organización, de esta manera se debe proteger lo más importante que es: confidencialidad, integridad y disponibilidad de una organización o laboratorio, teniendo como consecuencia algún tipo de impacto. Esto dentro de un contexto de la seguridad y en terminos estándares de la Organización Internacional de Normalización 27001 (Mendoza, 2015).

En el laboratorio de ciberseguridad dentro la Universidad Politécnica Estatal del Carchi el uso de herramientas y técnicas de pruebas de penetración aún no se encuentran analizadas, cabe mencionar que, en el servidor web el manejo de la seguridad mediante Pentest no mantiene una evaluación de forma permanente o constante, debido a la subutilización de recursos, por lo que esto podría ocasionar un nivel de seguridad bajo al servidor (Gavira et al., 2015). En este contexto, la presente investigación busca establecer una propuesta que formalice los procesos de seguridad del servidor web del laboratorio de ciberseguridad dentro de la Universidad Politécnica Estatal del Carchi.

1.2. FORMULACIÓN DEL PROBLEMA

El bajo nivel de seguridad en el servidor web del laboratorio de ciberseguridad, se debe a la subutilización de recursos de pruebas de penetración, provocando vulnerabilidad a los sistemas, dentro de la Universidad Politécnica Estatal del Carchi, año 2020.

1.3. JUSTIFICACIÓN

El análisis de esta investigación se la llevará a cabo para desarrollar una prueba de penetración (Pentest) específicamente a los servidores web del laboratorio de ciberseguridad, dentro de la

Universidad Politécnica Estatal del Carchi en la ciudad de Tulcán, con la finalidad de diagnosticar el nivel de vulnerabilidades a los servidores web alojados en el laboratorio contando con un aumento de seguridad, permitiendo cumplir con procesos metodológicos de Owasp, como también cumplir con los estándares de seguridad como son la confidencialidad, integralidad y disponibilidad de la información proporcionada.

Se analizará la adaptación de diversos procesos y método de pruebas de penetración, como es la uso de herramientas y técnicas de análisis de seguridad tales como Maltego, Owasp Zap, Nessus, WPScan, Nmap, Nikto, entre otros, como también la realización de fases y técnicas de (Pentest) que permitan identificar por medio de métricas y herramientas de evaluación el nivel de seguridad a los servidores web, de tal manera brindar controles de protección que serán de utilidad para el laboratorio de ciberseguridad logrando mejorar el funcionamiento (Gavira, Cárdenas y Supelano, 2015).

Por otro lado, es de mucha importancia aplicar una metodología de buenas prácticas que guie en el desarrollo de las pruebas de penetración en un entorno virtual, por medio de indicadores que evalué el nivel de seguridad. Es por ello, que la técnica propuesta además de identificar los riegos y vulnerabilidades permitirán establecer recomendaciones que serán de mucho provecho para la institución a la hora de diagnosticar los problemas de seguridad al servidor web y a la toma de decisiones en la elaboración de cada uno de sus procesos.

Teniendo como base metodológica Owasp donde se considera el nivel de riego que se manejara como calificador de las vulnerabilidades, la recolección de la información, el pruebas de manejo, configuración y desarrollo, pruebas de manejo de identidad, pruebas de autenticación, pruebas de autorización, pruebas de uso de sesiones, manejo de fallas y criptografía, con el fin de reducir las vulnerabilidades que generen errores, la ejecución sobre los servidores web más de una vez con distintas herramientas con el fin de comparar y optimizar los resultados generados, la documentación para las respectivas evidencias que se identifiquen en el objetivo de prueba, entre otros aspectos analizar dentro de la metodología Owasp (Sánchez, 2017).
De igual manera métodos de Pentest como analítico-sistemático como es el caso del sistema operativo Kali-Linux con el cual se trabajará para encontrar los problemas de seguridad más comunes, como también la utilización de escaneo de vulnerabilidades como Nmap, WPScan y herramientas de explotación como fail2ban y slowhttptest, con el fin de identificar los elementos útiles generando reportes automáticos, pero la mayoría de estas herramientas son libres, gratuitas y procesan de manera eficiente y eficaz las vulnerabilidades descubiertas, lo

que esto conlleva a generar mayor esfuerzo para llevar a cabo estas actividades. De esta manera se debe tomar en cuenta que al ser programas gratuitos sus licencias presentan restricciones o en otros casos licencias de pruebas de corto tiempo, es por ello que algunas organizaciones de pequeño y mediano optan por usar herramientas gratuitas debido a que no cuentan con un presupuesto razonable para la adquisición de estas herramientas. Teniendo en cuenta lo anteriormente mencionado, el uso de pentest utilizando herramientas libres, gratuitas y buenas prácticas permite así ahorrar significativamente tiempo y costos dentro del laboratorio de ciberseguridad identificado para nuestra investigación, lo que pretenderá abarcar distintos enfoques informáticos (Muñoz y Pérez, 2017).

El enfoque del plan está encaminado hacia el desarrollo de soluciones tecnológicas para el servidor web del laboratorio de ciberseguridad que permitan manejar de manera eficiente los procesos de seguridad, identificando el uso de métodos y técnicas actuales de Pentest, para satisfacer las necesidades y objetivos estratégicos del laboratorio. Finalmente, la razón por la cual se realiza la investigación es para usar y compartir los conocimientos brindados por los docentes, para que sea un apoyo en la formación profesional y de esta manera llegar a dar una solución a los problemas detectados en el servidor web en el laboratorio de ciberseguridad dentro de la Universidad Politécnica Estatal del Carchi de la ciudad de Tulcán.

1.4. OBJETIVOS Y PREGUNTAS DE INVESTIGACIÓN

1.4.1. Objetivo General

Diagnosticar los problemas de seguridad al servidor web del laboratorio de ciberseguridad a través de pruebas de penetración, disminuyendo la vulnerabilidad al sistema en la Universidad Politécnica Estatal del Carchi.

1.4.2. Objetivos Específicos

- Fundamentar bibliográficamente las variables de estudio para la sustentación de la propuesta.
- Identificar el nivel de seguridad a los servidores web mediante el uso de metodologías para la detección de riesgos y amenazas en los sistemas.
- Determinar mediante pruebas de penetración las vulnerabilidades presentes en el servidor web para comparación y evaluación de los riesgos más críticos.

- Establecer una propuesta para formalizar los procesos de seguridad al servidor web del laboratorio de ciberseguridad dentro de la Universidad Politécnica Estatal del Carchi.

1.4.3. Preguntas de Investigación

- ¿Cómo la fundamentación bibliográfica ayuda a profundizar el conocimiento de la seguridad a los servidores web y la vulnerabilidad a los sistemas?
- ¿Cuál es el nivel de seguridad al servidor web en el laboratorio de ciberseguridad?
- ¿El adecuado uso de las pruebas de penetración (Pentest) pueden determinar las vulnerabilidades y problemas de seguridad en el servidor web en el laboratorio de ciberseguridad de la Universidad Politécnica Estatal del Carchi?
- ¿El uso de herramientas ayudarán a disminuir la vulnerabilidad al servidor web?

II. FUNDAMENTACIÓN TEÓRICA

2.1. ANTECEDENTES INVESTIGATIVOS

Para la presente investigación se han recopilado varios antecedentes que sirven como punto de refuerzo de las variables planteadas, algunos de estos trabajos fueron rescatados de repositorios digitales y revistas indexadas de artículos científicos relacionadas al tema de estudio.

En el artículo científico presentado por Sánchez y Santander (2016) con el tema. "Herramientas DNP3 Pentesting para redes de infraestructura crítica".

El cual consiste en evaluar cualquier tipo de amenazas y debilidades, desde la perspectiva de un atacante real, se desarrolla para analizar los controles de seguridad en el interior de la red con el fin de impedir que una orden falsificada pueda llegar a cualquier examinador, analizando los resultados y corroborar si existe un control positivo con la realización de este tipo de herramientas y convertirlo en un elemento principal en la protección de la sociedad, la economía y el funcionamiento general de los países. Con los resultados obtenidos a partir de la prueba de fallo y error fueron positivos, se muestra que la aplicación de un Pentest mejora el nivel de seguridad y disminuye la vulnerabilidad informática la: confidencialidad, integridad y disponibilidad.

Este artículo de Sánchez y Santander para el proceso de mejorar el nivel crítico de la infraestructura, tuvieron que identificar los problemas que hacen que incremente la vulnerabilidad, realizaron un proceso de pentesting en ambientes de control industrial finalmente el uso de herramientas de pentesting para la ejecución permitiendo suplantar de manera correcta comandos enviados por la estación administradora.

Al terminar el artículo de "Herramientas DNP3 Pentesting para redes de infraestructura crítica" se obtuvo las siguientes conclusiones:

El proceso de pentesting es importante, permitiendo una interacción mejor y eficaz en la infraestructura crítica porque permite que los responsables de ciberseguridad puedan realizar verificaciones de las configuraciones en los dispositivos de seguridad, con el propósito de disminuir la probabilidad de ocurrencia, factibilidad en la suplantación y estación administrativa, lo cual permitirá constatar que los atacantes no podrán cambiar los controladores en la infraestructura (Sánchez y Santander, 2016).

El aporte de este trabajo para con la investigación es fundamental debido a que ayuda de forma directa para probar que la ejecución del manejo de Pentest optimiza el nivel de seguridad de una empresa, organización o este caso al laboratorio.

En la tesis realizada por García (2014) extraído del repositorio digital de la "Universidad de Catilla", realizado por el autor con el tema "Herramientas de Pentest orientado y automático".

En función del antecedente se puede concretar que uno de los principales problemas a nivel organización y a sus equipos informáticos es el robo de la información y las amenazas tanto externos como internos que se generan diariamente, como solución a esta incógnita el autor presenta el desarrollo de una aplicación de forma automática que realice pruebas de penetración (Pentest) para evitar el ingreso de amenazas y mejorar paulatinamente la seguridad en el área de proyectos, además de disminuir la vulnerabilidad obteniendo reportes exactos y de manera inmediata, que sirva como soporte para contar con una decisión clara. Otro puntos clave del antecedente se encuentra en las etapas de las pruebas de penetración, porque sirven como lugar de referencia para el cumplimiento del objetivo general en la investigación, además de ofrecer un panorama de las herramientas que se pueden utilizar para el funcionamiento de este tipo de aplicativos informáticos.

El proyecto realizado por García permitirá tener claro conceptos importantes y específicos acerca de la seguridad de información, pentesting y sobre todo a las diferentes vulnerabilidades que pueden dañar a un sistema informático como también la aplicación de pruebas de penetración exponiendo los distintos niveles de privilegios, logrando brindar seguridad al entorno virtual asignado.
Finalizado el proyecto de investigación, el autor obtuvo como resultado la identificación de las brechas de seguridad, la distribución de Backtack, herramientas que ayudan a realizar pruebas de penetración de forma eficiente y la explotación de la máquina virtual cumpliendo los requisitos previstos.

En la tesis realizada por Pérez y Quiñones (2017), que forma parte del repositorio de la Universidad de Guayaquil con el tema "Uso de herramientas de Pentesting para el análisis de vulnerabilidad de las operadoras ubicadas en la ciudad de Guayaquil".
Esta investigación trata del diagnóstico de las vulnerabilidades que se presentan a los sistemas de comunicaciones móviles, dando a conocer los ataques relacionados con cada vulnerabilidad con el objetivo de proponer una solución que avale identificar cualquier tipo de amenazas que

se encuentra aplicado en la localidad. La contribución de este estudio con respecto a la investigación se enfoca en la elaboración de análisis de los requerimientos, los procesos y técnicas que completan las pruebas de penetración brindando una apariencia más profunda con relación al uso de estas herramientas. Igualmente, demuestra la extensa aplicabilidad de estos métodos de pentest en diferentes áreas como en este caso, a la ciudad de Guayaquil.

El proyecto realizado por Pérez y Quiñones (2017) permitirá cumplir con el proceso de evaluación de las vulnerabilidades presentes en los sistemas de comunicaciones móviles mediante un test de intrusión, ayudando a identificar el nivel de seguridad en la infraestructura y ver el nivel de riego y amenaza al cual se está expuesto, de tal manera que un cracker realice un ataque cibernético y violente con la integridad, confidencialidad, y disponibilidad a la información, y finalmente evaluar vulnerabilidades a los sistemas de comunicaciones móviles con todas las técnicas de protección para cubrir fallos de seguridad detectados.

Finalizado el proyecto de investigación, el autor obtuvo como resultado una tabla de aceptación donde para mitigar las posibles vulnerabilidades estableció herramientas tecnológicas donde permitan detectar falencias en los sistemas de seguridad, la implementación de un control adecuado bajo el modelo Magerit, la realización completa de establecer conclusiones para una solución acorde a las necesidades que se presenten en las operadoras, la simulación de analizar y finalmente la simulación que realizan los ataques informaciones y de qué manera podría llegar afectar la infraestructura.

Otro de los antecedentes investigativos forma parte del repositorio de la Universidad de Guayaquil, elaborado por Briones y Hernández (2018), que trata acerca de "Auditoria de seguridad del servidor web de la empresa Publinext S.A. Utilizando mecanismo basados en OWASP".

Esta investigación menciona los varios tipos de servicios que se han ganado popularidad en los mercados tecnológicos y han conllevado a que la información proporcionada sea robada y alterada sino se utilizan medidas de seguridad necesarias para un buen manejo en los servicios que han sido incorporados en los últimos años gracias a la tecnología, como es el caso de sitios de comercio electrónico, servicios web, bancos entre otras más. Por lo que, con el pasar de los días las amenazas han sido ejecutadas por piratas informáticos que ponen en riegos a los sistemas informáticos, de la misma manera se han creado procesos o metodologías como es el caso de OWASP, OSSTMM, buenas prácticas, entre otras, que se han encargado de crear métodos y técnicas que evalúen los riegos y analicen todo tipo de vulnerabilidades que pueden

llegar a tener cualquier servicio tecnológico, para proteger y sobreguardar cualquier tipo de amenaza.

Otro de los antecedentes investigativos forma parte del repositorio de la Universidad Politécnica Salesiana Sede Cuenca, elaborado por Jaramillo y Riofrío (2015), que trata de "Metodología para realizar la evaluación, detección de riegos, vulnerabilidades y contramedidas en el diseño e implementación de la infraestructura de la red de la editorial Don Bosco, mediante un test de intrusión de Caja Blanca".

Esta investigación refiere a los diferentes tipos de metodologías que se encuentran presentes a la hora de realizar una evaluación, detección de riegos y vulnerabilidades que se encuentran presentes en la infraestructura, utilizando pruebas de penetración de tipo caja blanca. De esta manera se involucran etapas que ayuden a mejorar el nivel de seguridad fortaleciendo cualquier tipo de servicio informáticos, así como evaluar los riegos de los que está expuesto el sistema en su totalidad. Implementando metodologías como es el caso de OSSTMM, OWASP y buenas prácticas que se deben tomar en cuenta a la hora de desarrollar prueba de penetración dentro de una institución o empresa que deberán cumplir con premisas de seguridad que determinen y evalúen cada uno de los puntos a seguir en el proceso de buenas prácticas.

Una aportación importante para el mejoramiento de seguridad en las infraestructuras, servidores, laboratorios de ciberseguridad de una organización.

Para futuras investigaciones se puede realizar estudios con otras variables de comparación, ya sea cuestionarios aplicados a las organizaciones finales, entrevistas a pentester, entre otros instrumentos, para los servidores web que se adapten a metodologías orientadas a las buenas prácticas y un correcto manejo de seguridad.

2.2. MARCO TEÓRICO

2.2.1. Seguridad Informática

> La seguridad informática permite garantizar la privacidad de los datos, mediante la aplicación de proceso de gestión de riesgo, brindando confianza a las partes interesadas de una organización y la importancia de contar con herramientas que aseguren el buen manejo de los recursos y la seguridad de la información conduciendo a la prevención de la confidencial, integridad y disponibilidad (Carvajal, Vega y García, 2021).

De acuerdo con Garcés (2015), considera que los aspectos más importantes en la seguridad son:

Confidencialidad: Son servicios de seguridad que comprende a la información para no sea descubierta por procesos no autorizados.
Disponibilidad: Hace referencia a un sistema donde la información, hardware y software se mantenga seguro para los usuarios todo el tiempo.
Integridad: Hace referencia a que toda la información que se crea modifica y se borra debe realizarse sólo por el personal autorizado, garantizando una mejor seguridad dentro de una organización, institución, entidad, etc.

Según Díaz (2018) menciona que una vez conocido las definiciones de cada uno de los términos tanto de seguridad como informática podríamos decir que la seguridad informática radica en proteger los recursos de una organización de tal manera que el ingreso a la información ya sea a eliminar, editar o crear sólo sea realizado por el personal que se encuentra acreditado y autorizado dentro de cada organización.

La seguridad informática es indispensable en cada organización con el fin de proteger los servicios y archivos, de esta manera las amenazas para la seguridad se las divide de la siguiente manera (Ochoa, 2018).

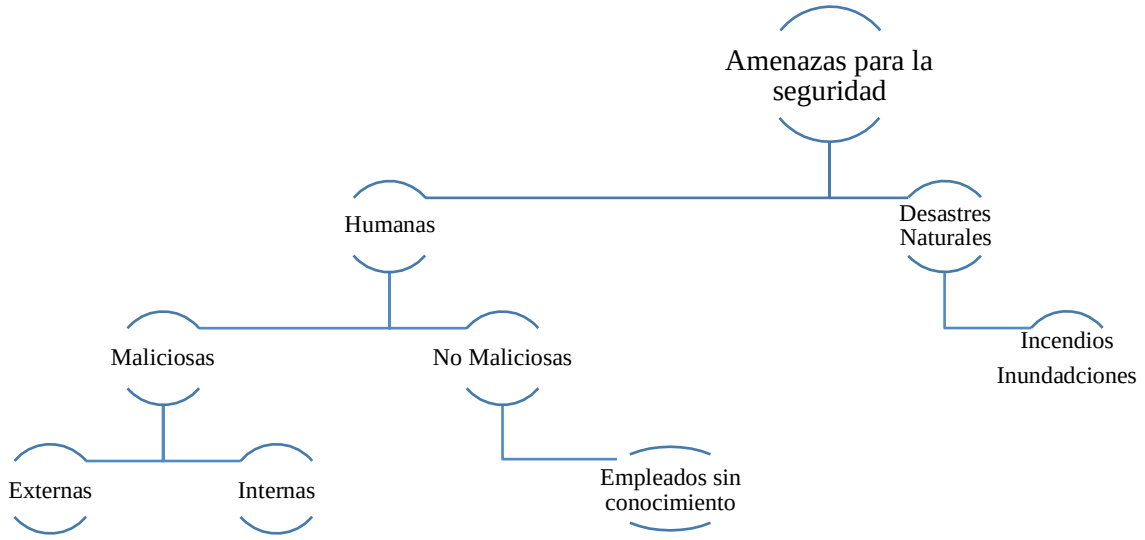


Figura 1. Amenaza para la seguridad

Fuente: Seguridad en las bases de datos mediante metodologías de pentest (Ochoa, 2018)

Amenazas para la seguridad: Consiste en actos malintencionados que pueden darse de diferentes maneras, ya sea mediante la realización humana o por un eventual caso de desastre natural. A continuación las amenazas para la seguridad son:

Humanas: Como amenaza en el aspecto humano se refiere a actos cometidos por un individuo cuyo intención es lastimar o hacer el daño a sistemas que puedan contener información sensible dentro de cualquier organización, entidad o área en donde la seguridad deba ser protegida. En este sentido no todos los individuos con el afán de hacer el daño puedan realizarlo de manera rigurosa, es por ello que se clasifican en:

Maliciosas: Cuyo individuo se proponga a realizar cualquier tipo de atentando mal intencionado ya sea externamente como internamente.

Externas: Esta amenaza maliciosa externa es provocada mediante factores en la que el individuo trata de realizar su acto de malicia ya sea robando archivos físicos de la empresa, contraseñas, perder información, entre otros.

Internas: Esta amenaza maliciosa interna se centra en los ataques mediante malware, comando infecciosos, aplicaciones que trates de dañas, eliminar cualquier información sensible.

No maliciosas: Cuyo individuo no trata de dañar la integridad, disponibilidad y confidencialidad de una empresa.

Empleados sin conocimiento: No son considerados agentes maliciosos, ya sea por el desconocimiento que tienen al querer hacer algún tipo de ataque o simplemente por ética profesional y respeto a la organización.

Desastres naturales: Dentro de una amenaza para la seguridad, un desastre natural es provocados por fenómenos ya sea un apagón, incendio, terremoto, entre otros, que interceptan o dañan algún tipo de cableado o dispositivo que almacenan dicha información.

De esta manera puntualizando el concepto de seguridad informática utilizada en mi proyecto de investigación se toma en cuenta los datos, información y credenciales de acceso a la hora de desarrollar y configurar un servidor web que se encontrará alojada en una aplicación web que cuente con información valiosa, en este sentido se tomará la seguridad respectiva en cada una de las funcionalidades.

2.2.2. Servidor Web

Para la ejecución de esta investigación se hará uso de servidores web con el fin de realizar las pruebas necesarias de vulnerabilidad para detectar en cada uno de los sistemas, además de definir mediante un cuadro comparativo que servidores cuentan con mayor rendimiento, factibilidad, seguridad, entre otros, para de esta manera analizar cada uno de ellos y valorar su alcance.

Un servidor Web es un sistema que nos proporciona el control de aplicaciones que se encuentran en el servidor dando respuesta a las peticiones que realiza un usuario, ejecutando los recursos que requieren a través del protocolo HTTP o HTTPS (protocolo más seguro, cifra los datos de una manera autentica y segura) respondiendo mediante peticiones GET, que corresponden a la capa de aplicación del modelo OSI mediante TCP asignado sobre el puerto 80 y 443, el servidor web está constituido por un servidor FTP (Protocolo de transferencia de archivos), HTTP (Protocolo de transferencia de hipertexto) y SMPT (protocolo simple de transferencia de correo) (Mateu, s.f.).

De manera técnica cuando el usuario solicita un archivo a través del navegador a un archivo alojado en un servidor web, el navegador lo procesa por medio del protocolo HTTP, de esta manera cuando la solicitud alcanza el servidor adecuado tanto (Hardware y Software) y es

aceptada la solicitud, lo encuentra el archivo solicitado y lo envía de regreso al navegador, pero si el navegador no encuentra el dato solicitado procede a enviar el error 404 no fount (Cañola, 2020).

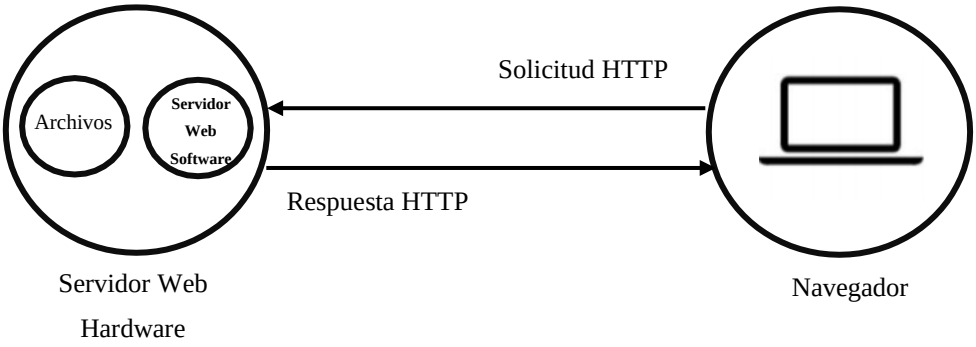


Figura 2. Proceso de solicitud a un servidor web

Fuente: Sistema preventivo contra ataques de denegación de servicio web (Cañola, 2020).

El servidor web está constituido en cuatro pasos:

1. Cliente estable conexión
2. El servidor web resuelva la petición
3. El cliente interpreta la respuesta
4. Cierra la conexión

Entre cliente y servidor lo que hay es una transacción, en primer lugar lo que ocurre es que se establece una conexión con el cliente, el cual inicia esta conexión con el servidor y pedirá a la página mediante la url por ejemplo la 172.20.24.53, lo que se está haciendo es realizar una petición al servidor web de 172.20.24.53. El servidor como está permanentemente pendiente lo que hace es resolver esta petición del cliente a través del puerto indicado y la resuelve sirviendo los ficheros de texto o binarios necesarios para resolverlo, sin embargo cuando existen peticiones que su repuesta da un error pues no se envía nada y simplemente no procesa ningún fichero, pero si ejecuta un código de error, para que el cliente note que esa petición ha dado lugar a una falla. De esta manera cuando el cliente recibe la respuesta del servidor tiene que interpretar los ficheros que han sido procesados, generalmente lo más común es que reciba como base en HTML y así gracias a esta interpretación de estos ficheros se podrá observar la página en sí. Finalmente, como último paso es que presenta un cierra de conexión entre cliente y servidor, es decir no hay un argumento que permanezca entre conexiones sino que cada conexión es independiente, no obstante se puede darse el uso de algunas variables de sesión y cookies.

Al momento de usar el servidor web para la ejecución de las pruebas de penetración el más destacado en servidores web y más utilizado desde 1996 es el software libre Apache, y su famosa configuración LAMP que forma parte de la configuración de los servicios Linux, que son: Apache, MariaDB y PHP considerado la configuración más estable para la creación de un servidor web, porque cuenta con servicios que dan un buen soporte a la hora de mantener alojado un servidor, una base de datos y privilegios funcionales a la hora de mantener segura la información. Según López y Romero (2010) hay que entender y tenerlo muy en claro que las capacidades del servidor web se denominan servidor de aplicaciones web y las páginas que se ejecutan hacia el servidor de aplicación se las denomina paginas activas.

Según Burbano (2019) menciona que existen diferentes servidores web que ayudan a la elaboración y procesos para la ejecución de pruebas de penetración, como es el caso de Apache donde su uso es más para el sistema operativo Linux y Microsoft IIS para el sistema operativo Windows, esta también Java System Web Server que permite convertir la máquina en un servidor web cumpliendo con un alto rendimiento escalable y seguro siendo uno de los servicios que brindan una mejor calidad al momento de variabilidad, configuración y seguridad.
En este sentido para que esta configuración no tenga errores se recomienda configurar el sitio web donde permita controlar las actualizaciones y de esta manera poder ejecutar al servidor remoto para evitar que afecten a los servicios que ofrece el sitio web, finalmente es de sus importancia y tener claro que estos procesos de configuración debe tener una copia de seguridad y Backups para facilitar el proceso a la hora de cargar el hosting.

2.2.3. Tipo de servidores web

Como parte investigativa del proyecto y parte del marco teórico se definirá algunos de los servidores más utilizados en el mercado global, a partir de ello se manejará dos servidores que cuentan con ventajas significativas en termino de métricas que otros servidores, como también su acoplamiento a las observaciones de vulnerabilidad que se desea analizar. Como primer servidor tenemos Apache con versiones 7.9 y 8,4 Core y segundo el servidor Microsoft IIS con la versión 2016 Standard Evolution 1607 Windows Server. En este sentido se analizará su instalación, configuración y pruebas de vulnerabilidad que se realizaran a cada uno de los sistemas.
Como los más importantes a la hora de la creación de un servidor web están los siguientes:

2.2.3.1. Servidor Apache

Según Miranda (2016) apache es un servidor HTTP (HyperText Transfer Protocol), es un protocolo para las transacciones web, siendo uno de los más utilizados por ser código abierto y licencia libre, se adapta a sistemas operativos como Linux, Unix y Windows NT con todas sus variantes.
Una de las características que presenta Apache es la gran variedad de configuración de hardware obteniendo rendimientos máximos a partir de dichos recursos tanto de hardware y software. Apache es una aplicación altamente modular donde se pueden cargar solo los módulos necesarios de entre muchos que existen, permitiendo compilar de forma estática o dinámica según convenga. En este sentido Apache tiene como principal uso el enviar páginas web tanto estáticas como dinámicas en la Word Wide Web conocida como www, siendo muchas de las aplicaciones web diseñadas como ambiente de implantación de Apache, o que se serán utilizadas características propias de este servidor web (Morales, Sánchez y Barrera, s.f.).

De acuerdo con Morales, Sánchez y Barrera (s.f.) manifiesta que, con respecto a la seguridad, Apache es considerado como un servidor web seguro, aunque se debe tomar en cuenta que Apache es instalado con las opciones por defecto **(out-the-box)**, resultando vulnerable a cualquier amenaza. En este sentido se debe tomar medidas preventivas para incrementar la seguridad. Entre ellos están:

- **Mod_ss.** Encripta el canal de comunicaciones utilizado entre el servidor y el cliente, verificando las identidades del servidor y el cliente.
- **Mod_redwrite.** Permite controlar los intentos de acceso por parte de los clientes a los sitios web.
- **Mod_log_forensic.** Crea una bitácora permitiendo identificar la fuente de los ataques.
- **Mod_Dosevasive.** Analiza vulnerabilidades tipo DoS, Fuerza Bruta para tomar alternativas para contrarrestarlo.
- **Mod_security.** Es uno de los mejores módulos que aplicar en temas de seguridad, capaz de interceptar e inspeccionar solicitudes de los clientes como las respuestas del servidor para identificar datos anormales o maliciosos.

Según Morales, Sánchez y Barrera (s,f.) apache es un aplicación altamente modular, es decir que dispone más de 80 componentes en donde se puede hacer uso de configuraciones para la seguridad en el servidor web, cada módulo representa una forma específica de tratar e

interpretar con el servidor, de acuerdo con la (figura 3) la estructura de apache esta representa de la siguiente manera. La parte del servidor el cual se el cual da paso a la funcionalidades de los módulos, cada módulo esta interconectado con el servidor para dar funcionamiento a cada uno de ello. Dentro del servidor se encuentran los diferentes directorios que están constituidos de las siguiente manera:

- **El apache conf:** Fichero el cual se encarga del comportamiento general del servidor web.
- **Php conf:** El encargado de almacenar los datos como los detalles de conexión de una base de datos.
- **System binaries:** Constituye a todos los archivos del sistema que han sido instalados.
- **System files:** Como mecanismo para el intercambio de datos entre sistemas o conexiones entre sí.

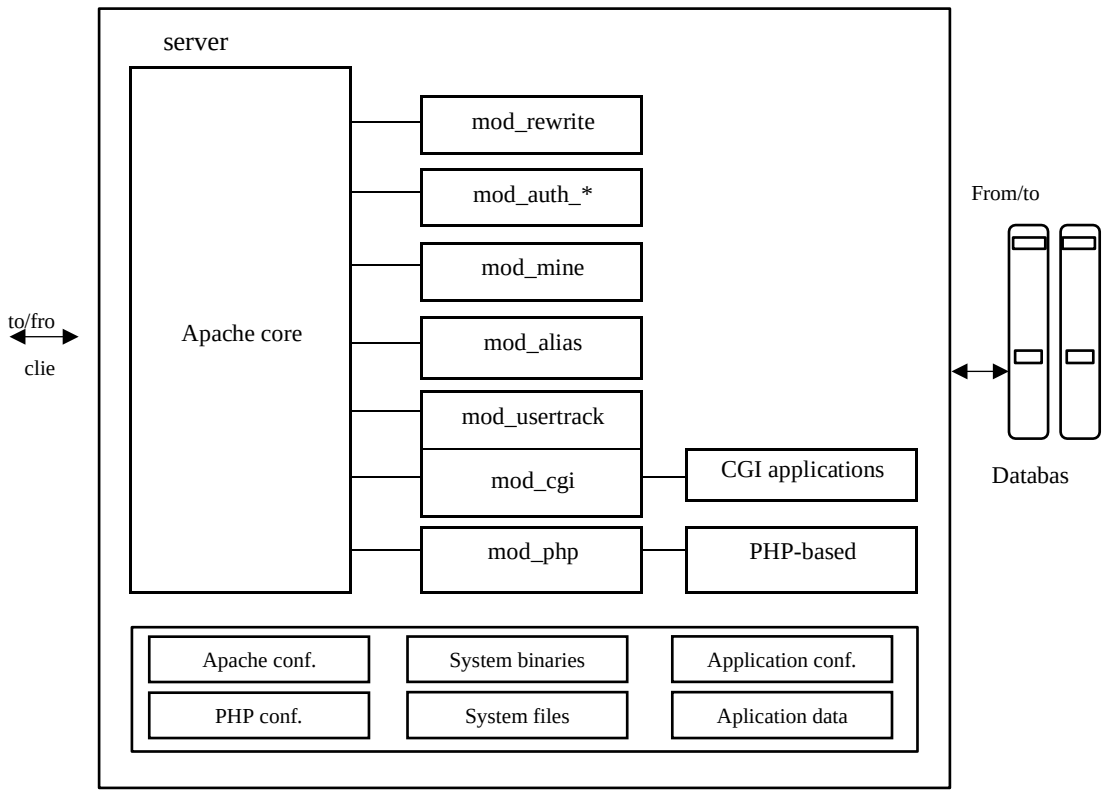


Figura 3. Componentes de Apache

Fuente: Análisis comparativo de servidores web: Apache vs Microsoft IIS (Morales, Sánchez y Barrera, s.f.).

Finalmente, una de las características más importantes del servidor Apache el valor de costo, licenciamiento, de mantenimiento y de escalabilidad, en todos los casos es de $0,00, haciendo que cualquier persona pueda descargar gratuitamente y utilizarlo de manera privada o comercial sin restricción alguna.

2.2.3.2. Servidor Nginx

Según Esteban, (2018) Nginx es un sistema de código abierto gratuito que se definide por su alto rendimiento, además ejecuta funciones tales como proxy reverso HTTP, balanceador como POP3 y IMAP. El servidor puede alojarse y funcionar de una manera correcta en varios sistemas operativos como Windows, Linux y Unix. Sus ventajas podemos decir que su configuración es muy simple pero muy poderosa, permitiendo su configuración para adaptarse casi con cualquier tipo de tecnología y lenguajes de programación sofisticados.
Dentro de las características que presenta Nginx son las siguientes:

- Servidor de archivos estáticos
- Proxys inversos
- Balanceos
- Tolerancia de fallos
- Soporte en Http-Http2 encima de certificaciones SSL
- Soporte para autenticación
- Compatible IPv6
- Soporta hasta más de 10000 conexiones simultáneas (García, 2018)

De esta manera como parte física este servidor, cuenta con un 12Gb de memoria Ram, doble procesador Intel Xeon X5660 estos trabajan a 2,67Ghz, 2 discos duros de 500Gb HDD es decir que uno falla por a o b motivo el otro se levanta automáticamente, inclusive está distribuido en un array 1+0 y de la misma manera su trabajo es de 7200 rpm, y finalmente cuenta con una tarjeta de red con 4 puerto gigabit Ethernet. Como parte lógica el servidor cuenta con un sistema GNU/Linux el cual es Centos 7 el mismo que da cabida y paso al servidor Apache, su instalación es con el consumo mínimo de paquetería como uso exclusivo del servidor, cuenta con una tarjeta de red con 4 entradas gigabit Ethernet (García, 2018).

2.2.3.3. Servidor LiteSpeed

Según Borges (2018) este servidor es una versión open source comercial donde incluyen varios tipos de licencia, también resiste cantidades grandes de conexiones simultaneas con un gasto muy bajo incluso con aplicaciones que son muy demandantes como las que se utiliza en PHP. Como parte de sus desventajas es una versión bastante comercial, donde requiere de un presupuesto alto para poder acceder a todos sus beneficios.

Está compuesto por rendimiento con Ruby on Ralis, su procesamiento es dinámico, es decir que en lugar de iniciar un proceso este crea una llamado conocida como "fork" al sistema que cambia el id del proceso que se le estaba asignando, además soporta protocolo QUIC que es descendiente de HTTP/2, finalmente el uso de caches disponibles en módulos para todo tipo de gestores de contenidos (León, 2019).

2.2.3.4. Servidor Microsoft IIS

A partir de la investigación se hará uso del servidor Microsoft IIS donde se tomará en cuenta para su utilización los componentes de seguridad tales como sus filtros de solicitudes para bloquear la conducta de los protocolos y del contenido, la autenticación básica, es decir sus certificaciones SSL, credenciales de acceso y demás, finalmente la autenticación de Windows que permitirá crear credenciales entregadas para el sistema operativo.

Según Borges (2018) fue elaborado por Microsoft netamente a sistemas operativos Windows, brindado un procesamiento hacia páginas desarrolladas en ASP/ASP.NET, de esta manera Microsoft IIS no solamente es un servidor sino un suite de servicios para la página web debido a que proporciona servicios de SMTP y FTP.

De acuerdo con Morales, Sánchez y Barrera (s.f.) IIS es un poco portable al ser atado a una versión especifica de sistema operativo resultado un limitante, sin embargo, logra aprovechar al máximo sus funcionalidades y lograr un buen rendimiento. Sus módulos también llamados extensiones, pueden ser agregado o removidos individualmente de acuerdo con las necesidades del usuario, mejorando su escalabilidad y rendimiento. De esta manera respecto a la seguridad presenta un módulo de seguridad para el manejo de las autenticaciones y autorización del servidor web.

Según el Centro Criptológico Nacional (2018) el servidor Microsoft IIS está compuesto por:

Modularidad: Es decir que son administrativas que pueden ser fácilmente añadidos, eliminados o remplazados, desarrollando varias mejoras como la asegurar la reducción de una superficie de ataque y la reducción del consumo de memoria.
Extensibilidad: Permite crear nuevos módulos, ya que permite a terceros el desarrollo de sus propios módulos, automatizando o mejorándolos para cualquier funcionalidad.
Integración ASP.NET: Brinda la oportunidad de adquirir características de ASP.ET ya sea autenticación de formularios, estados de sesión, entre otras.
Fácil transporte de la configuración: Permitiendo al servidor se manejable y capaz de transportar las configuración de una rápida y sencilla sin posibilidad de errores.

2.2.3.5. Servidor Google web

Según Borges (2018) abreviado como GWS, es un servidor web realizado en lenguaje de programación C++ por la compañía de Google, en la actualidad es utilizada para la mayoría de su infraestructura web y no se encuentra disponible para el público.
Estos servidores de Google están compuesto con diferentes categorías: las reparticiones de carga, que aprueban la petición del cliente y son enviados a los servidores de Google por medio de Proxys Squid, estos proxys permiten que sean enviados mediante la cáche local o en el caso de no ser renviados se proceden a realizar la petición al servidor web. estas ejecuciones son enviadas por todos los usuarios utilizando lenguaje HTML para que puedan ser vista en el interfaz web, este servidor GWS requiere de menos espacio de disco, pero si aguantan cargar altas de procesamiento, finalmente este servicio al ser parte de Google permite gran cantidad de documentos y almacenamiento de acuerdo con la configuración que se desee requerir.

De acuerdo con Netcraft (2021) mediante una encuesta se recibió respuesta de 1,204,252,411 sitios en 23,042,054 dominios únicos y 10,766,066 ordenadores con acceso a la web, reflejando una ganancia de 6,270,052 sitios, 92.829 dominios y 116.789 computadoras. Nginx está posicionándose en la cima de las listas que respecta al recuento total de sitios, así como el número de dominio únicos y sistemas con acceso a la internet. En este sentido el 34,5% de todos los sitios se ejecutan en Nginx, el 30.4% de los dominios y el 35.0% de las computadoras web, como segundo lugar esta Apache con una participación de mercado del 26.3%, una participación muy similar de dominio del 26.4% y un 32.7% de computadoras orientadas a la

web. Aunque Nginx lidera el mercado en general, Apache tiene pequeñas ventajas en los que respecta al millón de sitios más ocupados, con una participación de mercado del 25.6% teniendo 2.4 puntos por delante de Nginx. De este modo Apache favorece a los sitios con contenido único sirviendo al 25.5% de los sitios activos en términos de métricas, mientras que Nginx sirve el 19.8%. Finalmente, Google que representa una parte considerablemente grande de 9.9% de los sitios activos, debido a su popular servicio Blogger y Microsoft IIS sigue disminuyendo debido a caída significativa en 2020 a favor de OpenResty contando con un 6,5% del mercado de sitios y el 6,0% de dominios en febrero del 2021.

De todos los servidores anteriormente mencionados aprovechan de una fama tanto por su experiencia, ya sea por su rendimiento y tecnologías que pueden llegar a resistir. Según el reporte oficial de (Netcraft, 2018) mediante una gráfica y tabla de datos indica los servicios con más usabilidad y qué porcentaje tienen en el mercado global, como se mira en la siguiente imagen (figura 4).

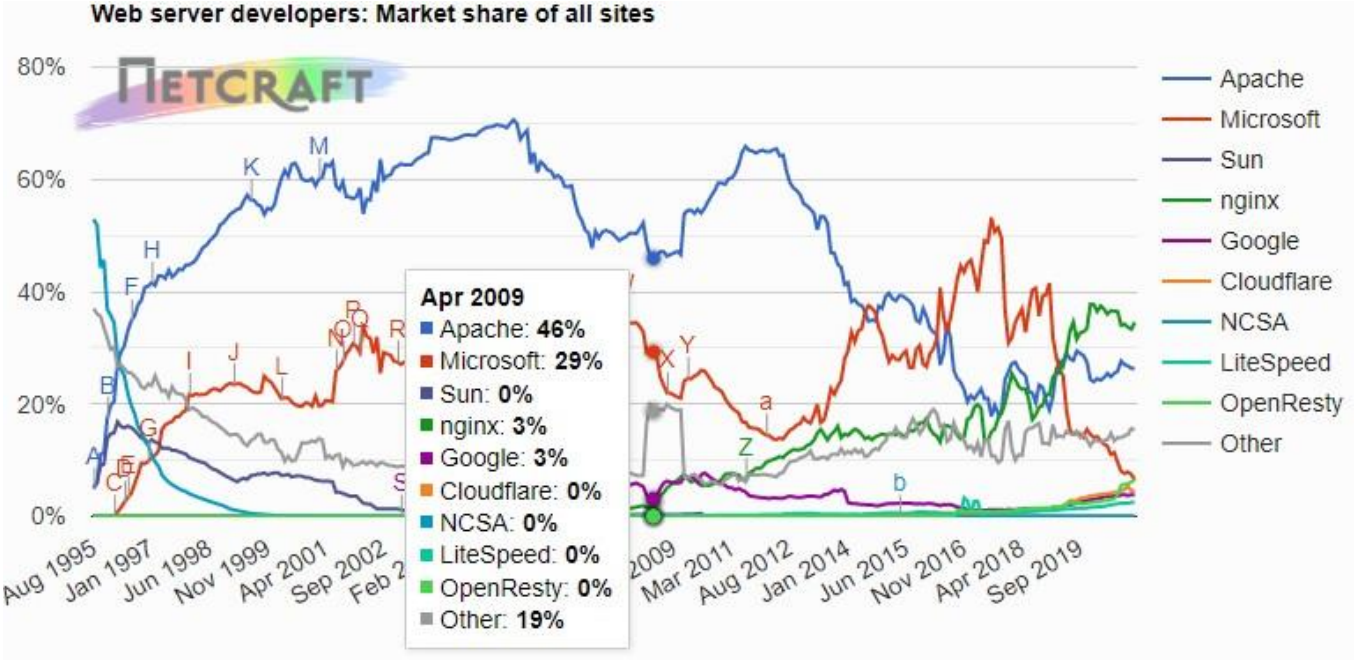


Figura 4. Desarrolladores del servidor

Fuente: Encuesta sobre servidores web (Netcraft, 2021).

Tabla 1. Comparativo de servidores web

Nombre	Definición	Características
Apache	• Potente y flexible • Funciona en varias plataformas de entorno • Mas 80 de módulos	• Es de código abierto • No tiene costo alguno • Corre en cualquier plataforma • Produce aplicaciones de calidad
Microsoft IIS	• Fácil programación en ASP (Páginas de servidor activo) • Componentes programables a cada uno de sus módulos para un cargo determinada	• Solo para versionamiento Windows Nt, Server 2000,2006 y 2016 • Versiones como 4.0 5.0 y 5.1
Nginx	• Es Http y proxy inverso sin costo • Proxy para IMAP y POP3 • Código abierto • Alto rendimiento	• Es estable • Sencillo de configurar • Utiliza pocos recurso • Envía solicitudes de uno o varios procesos Merb • Seguro, rápido, compatible y flexible
LiteSpeed	• Para Unix/Linux y Microsoft • Ligero	• Óptimo para entonarnos con velocidades criticas • Memoria muy pequeña • Adecuado para servidores con mucha carga

Fuente: Cuadro comparativo a servidores web (Idiaquez, 2019).

Tabla 2. Datos obtenidos a servidores web

Desarrollador	Enero 2021	Por ciento	Febrero 2021	Por ciento	Cambiar
Nginx	399,330,927	33,3%	415,900,479	34,54%	1,20
Apache	316,046,149	26,38	316,992,638	26,32%	-0,06
Microsoft	89,781,136	7,49%	78,331,379	6,50&%	-0,99
OpenResty	74,385,487	6,21%	76,623,440	6,36%	0,15

Fuente: Encuesta sobre servidores web (Netcraft, 2021).

2.2.3.6. Servidores web con código libre

Según (Mateu, s.f.) manifiesta que las características a servidores de código libre son las siguientes:

a. **"Spelling de Apache".** Permite detallar una página error, como recursos no encontrados, en donde asignan al usuario algunos nombres al que solicita para el caso hubiera cometido un error al escribir.
b. **"Status de Apache".** Proporciona el estado de la página web que es formada por el servidor donde indica como se encuentra su funcionamiento, el nivel de respuesta, entre otras.
c. **"RXML Tags (Roxen)".** Añade HTML algunos tags mejorados para programación y generación de contenidos dinámicos.
d. **"SQL Tags (Roxen)".** Proporciona al lenguaje HTML comandos con el fin de tener acceso a la base de datos SQL desde paginas netamente HTML.
e. **"Graphics (Roxen)".** Es añadido al lenguaje HTML, es decir funciones para desarrollo de graficos, títulos ya que no requieren un compromiso de diseño gráfico.
f. **"Bfnsgd (AOLServer), mod_gd de Apache".** Proporciona ejecutar gráficos mediante textos y fuentes True Type.
g. **"Mod_mp3 de Apache, ICECAST, MPEG (Roxen)".** Ejecuta al servidor como un servidor de música (Streaming).
h. **"Throttle (Roxen), mod_throttle de Apache".** Ejecuta medios que limitan la velocidad de servicio de HTTP, ya sea por usuario, servidor virtual, etc.
i. **"Nsxml (AOLServer), Tdom (AOLServer), mod_xslt de Apache".** Permite convertir fiches XML a través de XSL.

2.2.2.7. Seguridad y Autenticación

Parte de los servidores web ofrecen controlar aspectos vinculados con seguridad y autenticación de los usuarios, de manera que la forma más simple para controlar es facilitado por el manejo del fichero .htaccess siendo un método de seguridad que proviene del uso a servidores web, donde radica en colocar el fichero con la asignación del nombre .htaccess en cualquier parte del directorio del contenido web que se vaya utilizar, mostrando que ese fichero tiene acceso a todos los directorios de los ficheros y directorios de donde se encuentran ubicados (Mateu, s.f.).

2.2.2.8. Tipos de ataques al servidor web

Los ataques informáticos que hoy en día se someten a divulgarse por todo tipo de servicio tecnológico que está presente en cada organización, es un problema que se lo ha ido tratando desde sus primeros inicios, cuando comenzaron a envenenar a cada uno de ellos, ya sea a servidores de correos, servidores web, aplicaciones, entre otras. De acuerdo con Akamai (s.f.) existe 79,509 ataques informáticos que se realizan por hora, de tal manera que nos enfocamos netamente en sus tipos de ataques al servidor web que han sido latentes a la hora de poner en funcionamiento, enfrentándose con riegos si no se aplica procesos o técnicas para mantener seguro estos servicios.

En este sentido todo lo anteriormente mencionado, el servidor web debe cumplir con ciertos parámetros de seguridad y configuraciones internas que puedan mejor el rendimiento y calidad de esta, cumpliendo con el propósito que es el identificar las posibles amenazas y vulnerarlas antes de que se filtren al servidor y puedan ocasionar daños más graves a una organización, de esta manera se debe conocer y prever todas las posibles acciones que puede ser provocados por el intruso. Tomando en cuenta este analices podemos decir que los riegos generados se pueden establecer mediante esta siguiente ecuación (Cautín, 2019).

$$\text{Riesgo} = \frac{\text{Amenaza x Vulnerabilidad}}{\text{Contramedida}}$$

De acuerdo con (Cañola, 2020) menciona que el termino servidor web se manifiesta en dos diferentes componentes el hardware y software, para el hardware un es un computador que almacena software de servidor web y archivos de sitios web como HTML, imágenes, CSS, JavaScript, entre otros, de tal manera que dispone de la red y gestiona el intercambio físico de los datos conjuntamente con dispositivos conectados a la red, para el software contiene toda la

lógica que controla como los usuarios acceden a los archivos que se suministran en el servidor web, con el fin de procesar las peticiones realizadas por otros dispositivos a través de dirección web como URL realizando transferencias de información mediante protocolo de HTTP. En este sentido se toma en consideración ataques que llegan afectar el funcionamiento del servidor web ya sea en los dos componentes anteriormente mencionados. A continuación, los tipos de ataques a un servidor web más comunes en el mundo del cibercrimen:

- **Infección de Malware.** Comúnmente conocido como programa malicioso, este software hace que todos los códigos maliciosos y programas ejecutados tengan como propósito dañar el sistema, aprovechando sus vulnerabilidades provocando daños fuertes a los sistemas operativos con el fin de causar un mal funcionamiento Rivero (citada en Alvear, 2019).

Tomando en cuenta lo anteriormente mencionado existen aplicaciones, que se encuentran en plataformas conocidas, que contienen malware que han sido ocultas durante 3 años, esta aplicación tiene más de 3 millos de descargas, y su nombre es System Update, de esta manera ha hecho que usuarios inexpertos descarguen con el fin de actualizar su sistema operativo, lo que ocurre al momento de ejecutar el programa, genera un error eliminando el programa de la pantalla de inicio colocándolo en segundo plano, convirtiéndose así en un Spyware (software malicioso espía) en tiempo real, consiguiendo recopilar información de sus víctimas. En este sentido la seguridad proporcionada a este tipo de programas maliciosos debe ser profunda con el objetivo de cuidar a los usuarios expertos e inexpertos a la hora ejecutar un software desconocido, ya sea en plataformas conocidos como en sitios web igualmente conocidos, mejorando el nivel de seguridad y experiencia de usuario a los clientes u organizaciones que se encuentren afectadas con este tipo de malware Rivero (citada en Alvear, 2019).

- **Virus.** Los virus que se presentan en los servidores web tienen como objetivo duplicarse y dañar archivos del sistema operativo que se encuentran visibles para el atacante, con el fin de causar daño y mal funcionamiento tanto en el sitio web como el servidor. Estos virus se encuentran dentro de software ejecutables, o anuncios que rebotan al dar clic sobre él, duplicándose dentro del sistema operativo dañando archivos importantes del sistema, infectando componentes de almacenamiento con el propósito de expandirse de tal manera que se encuentre interconectados entre sí Rivero (citada en Alvear, 2019).

En este sentido uno de los virus que se generó a los inicios del internet y hasta la actualidad es el más conocido y famoso “Melissa” creado por David. L Smith el cual llegaba mediante vía correo electrónico como un documento Word que decía “Que el documento que es enviado, no sea mostrado a nadie más”. De esta manera desactivada las opciones de escritura y cambiaba la extensión de los documentos, este virus era incontrolable, se reenviaba a los primeros 50 contactos de los usuarios que han sido ejecutados.

De esta manera se pudo analizar que se requería de procesos, metodología y técnicas que verifiquen, analicen y procesen todo ese tipo de anomalías, con el fin de establecer niveles de seguridad óptimos para su correcto funcionamiento Tercera (Alvear, 2019). Los ataques que se verán a continuación son netamente expresados dentro del servidor web como problemas latentes que se han venido resolviendo de una manera rigurosa y compleja.

De acuerdo con Cabezas (2020) lo tipos de ataques al servidor web son:

- **Ataques de Inyección SQL.** Este tipo de ataque sucede cuando se establece en sentencias de SQL dinámicas, es decir que permite la entrada de un formato SQL al sitio web considerándolo uno de los más peligrosos y comunes ataques al servidor web, ya que de esta manera ingresará a la información de un formulario web accediendo a las cuentas provocando su modificación, robo y eliminación de datos.

Según Zabala (2016) los problemas que pueden se afectados mediante este ataque son:

Confidencialidad: Ya que la base de datos mantiene información sensible, porque es muy común la perdida de la confidencialidad a sitios web vulnerables.
Disponibilidad: Debido a que los atacante pueden conocer las contraseñas de usuario y poder acceder ante ellas.
Integridad: El mismo que si se consigue vulnerar una base de datos, podría tener como poder el borrar los datos de tablas, categorías, entre otros.

- **Ataques DoS**. También conocido como denegación de servicio, lo que hace es proporcionar una gran cantidad de peticiones al servidor desde un ordenador externo, comenzando a rechazar todas las peticiones después de haber consumido recursos que brinda el servidor hasta el punto de dejar sin capacidad de respuesta. La denegación de servicio se genera a través de un ataque de seguridad que logra desgastar los recursos informáticos de un host o red, haciendo inaccesibles el acceso de los usuarios, en este

sentido el ataque DoS se clasifica en el aumento de paquetes y la cifra de atacantes. El primero método es Software exploits donde el atacante vulnera las fallas de un programa que provoca a los servidores una falla en sus servicios con el objetivo que el rendimiento y funcionalidad disminuyan, como segundo método es flooding que interrumpe los servicios del sistema como son la memoria o la red, enviando grandes cantidades de peticiones falsas. A continuación, se muestra los métodos que ejecuta el ataque de negación de servicios (Cañola, 2020).

Según Bermejo (2017) los ataques que se pueden llevar a cabo de diferentes maneras tales como:

Truncamiento de la Aplicación: Buffer Overflows

Este tipo de ataque es ocasionado cuando se copia secuencias de caracteres buffer a otra, provocando así ataques DoS con el sitio web, en este sentido hay que estar muy pendientes de funciones que se encuentran tales como: strcpy(), strcat(), gest().

Modificación y daño de datos

Este tipo de ataque ocasiona un mal funcionamiento a los servicios, llegando al punto de que puedan dejar de manejarse, ya sea mediante fallos de SQL o código en script mal ejecutados.

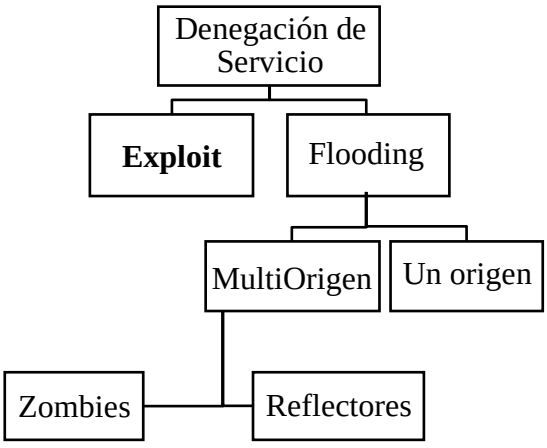


Figura 5. Clasificación de tipos de denegación de servicio

Fuente: Sistema preventivo contra ataques de denegación de servicio web (Cañola, 2020).

- **Ataques DDoS.** También conocido como denegación de servido distribuido, es tipo de ataque a diferencia del DoS empieza con las interrupciones de los servicios que ofrece a los usuarios, este tipo de ataque malicioso es el más fuerte causando perdidas para grandes organizaciones como también a instituciones gubernamentales. Por lo que se debe ser detectado a tiempo, de caso contrario imposibilita que el sistema provea servicio a sus usuarios.

En este sentido los ataques DDoS se generan mediante él envió de una gran cantidad de datos, haciendo de esta acción menos avanzada dependiendo de la cantidad de nodos infectados para su ejecución (Gado, 2015).

De acuerdo con Bautista (2019, como se citó en Pandya, 2015) menciona que, este tipo ataque DDoS se clasifican en:

- **Ataque directo:** Estas peticiones son enviadas directamente hacia un host, pero sin ocultar las IPs atacantes.

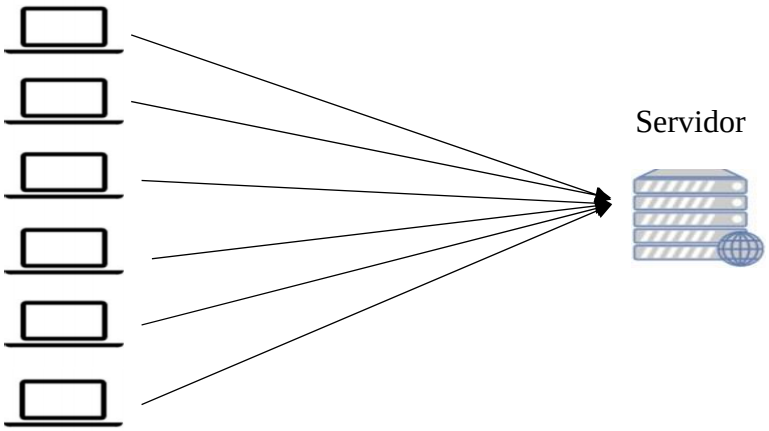


Figura 6. Ataques DDoS Directo

Fuente: Ataques DDoS, análisis y prevención de riesgos (Bautista,2019).

- **Ataque Indirecto:** En este ataque se distribuye el tráfico de las solicitudes mediante intermediarios antes de atentar con el host objetivo, con el fin de ocultar la IPs atacantes que son más complejas de localizar, en este sentido permite que los mismos dispositivos actúen como intermediarias que multiplican los paquetes, aumentando así la potencia del ataque.

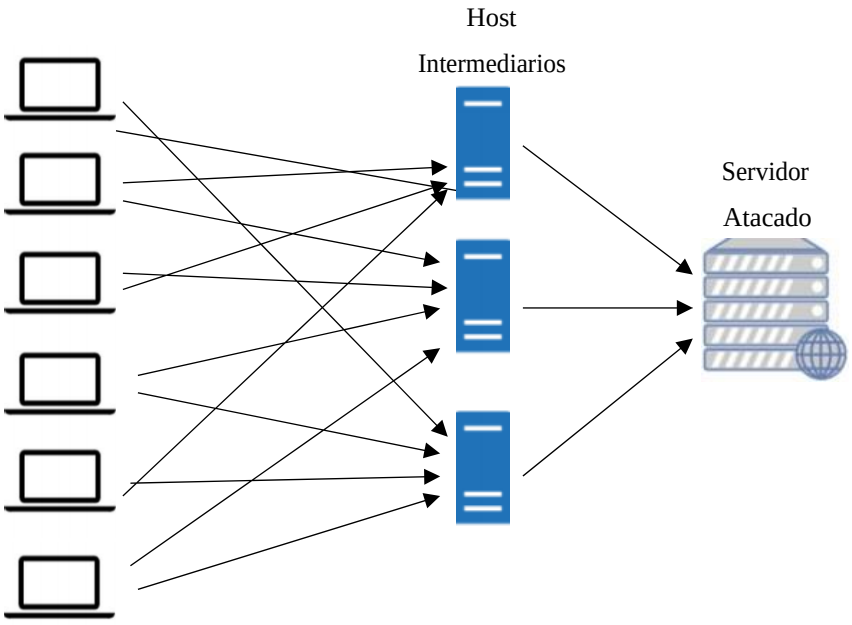


Figura 7. Ataques DDos indirecto

Fuente: Ataque DDoS con IoT, análisis y prevención de riegos (Bautista, 2019).

- **Ataques según el recurso atacado.** Estos ataques actúan de acuerdo con cada diferente recurso del host objetivo. Los ataques que son dirigidos a la red tienen como objetivo impedir el tráfico hacia la red, ocasionando el mal funcionamiento del servicio, en este sentido se pueden interpretar de varias maneras como:

    - UPD (User Datagram Protocol): El atacante pretende de invadir el host procesando una cierta cantidad de paquetes **UPD** (protocolo que permite la transmisión sin conexión) a puertos aleatorios de manera que la cantidad de peticiones provocara la saturación del servidor (Bautista, 2019, como se citó en Cloudflare, 2019).
    - ICMP (Internet Control Message Protocol): Este tipo de ataque utiliza paquetes **ICMP** protocolo responsable de reportar errores, también conocidos como “Ping” (Bautista, 2019).
    - LOIC (Low Orbit Ion Cannon): Forma parte de una herramienta realizada en C# como lenguaje de programación, que ejecuta un proceso DoS sobre una IP o URL de destino, de manera que al ser configurada, crea un envió intensivo de paquetes al destino realizado por el atacante (Bautista, 2019).

- **Ataques Fuerza Bruta.** Este tipo ataque es el más desafiante para los encargados de seguridad informática, y hacking ético ya que rompe todas las combinaciones posibles para obtener nombres de usuarios y contraseñas en una página web, incluso este tipo de ataque por fuerza bruta busca las contraseñas que sean más vulnerables para que puedan ser descifrables y permitan el libre acceso al servidor web.
- **Cross Site Scripting.** Según Zabala (2016) También conocido como XSS es manejado por los atacantes para inyectar scripts (documento que contiene código de programación) maliciosos al sitio web. Este script malicioso es enviado hacia cualquier usuario desprevenido o inexperto que esté haciendo uso del navegador, y lo ejecutara sin conocer que este script contiene funciones que hará que su servidor colapse en cuestión de segundos, como obtener información sensible ya sea robo de contraseña, tener permisos, acceso, credenciales, entre otros. Se manifiesta el ataque de diferentes formas:

XXS almacenado: Se presenta cuando en una página web se almacena scripts para el ataque, que cuando el usuario ingresa a la página, se ejecuta esos scripts adquiriendo así información del usuario visitante.

XXS reflejado: Esta ataque es cuando el usuario ingresa a la página mediante un correo electrónico o enlace desde un sitio comprometió, es decir de una página alterada obteniendo así información valiosa del usuario.

XSS basado en DOM (Documento Object Model) del navegador: Mediante este ataque se trata de obtener cookies o sesiones, es muy difícil de detectarlo, para solucionar este ataque es necesario la actualización del software.

Como parte del proyecto de investigación se hará evidencia de algunos de estos ataques a los servidores web tales como:

Tabla 3. Ataques al servidor web

Inyección SQL	Se demostrará mediante la herramienta SQLmap para identificar la existencias de líneas de comando script maliciosas, además la utilización de Owasp Zap para comprobar la vulnerabilidad en el sitio web
Denegación de Servicios DoS	Se analizará mediante el software de Loic y la utilización de comando slowhttptest para demostrar el truncamiento de la aplicación web
Ataques de fuerza bruta	Se analiza a través de la herramienta fai2lab para comprobar mediante generadores de contraseñas encontrar el acceso a la base de datos del sitio web.
Cross Site Scripting	Se analizará mediante la herramienta de Owasp Zap y un script malicioso para identificar la existencia de XXS

2.2.2.9. Tipos de ataques en las redes

Los ataques en redes tienen más procedencia en ser atacados por fuentes maliciosas, ya sea por falla en el software, hardware e incluso en el personal que se encuentra en el área informático, con el objetivo de causar daño en la información, archivos de la víctima. Estos ataques se los divide en dos categorías: “Pasivos”, cuando un intruso obstruye los datos de red que viajan a través de ello y se limitan a registrar el uso de los recursos del sistema y “Activo” cuando el intruso utiliza sus técnicas de comandos para alterar el funcionamiento de sus recursos del sistema (Rodríguez y Santibáñez, 2018).

De acuerdo con Valderrama (2017) menciona que, podemos destacar los ataques más comunes que se encuentran presentes en la red, a la hora de aplicar pruebas de penetración y analizarlas, de tal manera que se hayan enlistadas de esta forma:

- **Acces Point Spoofing.** Es una forma de ataque en la que se hace pasar por un Access point (dispositivos de red en donde permite conectarse a una red de cables o punto de acceso) y el usuario atacado piensa que se encuentra conectando a la red WLAN, provocando así la filtración del atacante generando el mal funcionamiento de la red en el ordenador , esta vulnerabilidad es muy común en la redes "ad-hoc" (conjunto de redes donde todos los nodos son libre de unirse con cualquier otro dispositivo de red).
- **Mac Spoofing.** Este ataque es causado cuando se vulnera una dirección MAC de una red, donde el atacante se hace pasar por personal autorizado, permitiendo así el cambio de la MAC, provocando gravemente fuertes entradas de amenazas o fallas a la red.
- **ARP Poisoning.** Este tipo de ataque forma parte de una técnica donde el atacante se infiltra a la red y busca la mayor cantidad de datos que pasan por la LAN, alterando el tráfico que pasa por cada una de las redes provocando colapsos en los envíos, hasta incluso detenerlos. En este sentido el atacante obtiene información valiosa y puede llegar a recopilar nombres de usuarios, contraseñas, cookies y hasta conversaciones ya sea por correo o mensajería instantánea.
- **WLAN escáner.** Valderrama (2017) afirma "Este tipo de ataque consiste en vigilar y descubrir redes WLAN activas, con el fin de poderlas escanear y analizar posibles amenazas".
- **Wardriving y warchalking.** Wardriving es una actividad donde busca puntos de acceso en redes inalámbricas en todos los puntos de la ciudad, colocando en un ordenador portátil una placa de red Wireless con el fin de detectar las señales y Warchalking se le dice cuando una red es vulnerada y permite a que otros ataques puedan ingresar a la misma red.

Tabla 4. Tipos de ataques

Técnica	Definición	Objetivo del ataque	Consecuencia
Scanning	Escanea información básica de sobre la red	Establece conexión con algunos de los puertos del sistema vulnerable	- Avisos de firewall - Aprovechamiento de puertos abiertos
Enumeración	Recoge información	Recoge la mayor cantidad necesaria	- Acceder a recursos

	necesaria de sistemas y servicios	de información valiosa para posibles ataques.	- Ataques de validación
Sniffing	Permite robar la información que está siendo enviada a otro sistema mediante tráfico de datos	Recoge la total la información que pasa por la red, sin importar que se encuentren con seguridad estricta.	- Revela archivos confidenciales: ya sea contraseñas, tarjetas de crédito, usuarios, entre otros.
Fuerza Bruta	Proceso que accede mediante generar claves	Mediante herramientas genera claves para verificar en Login, autentificaciones o archivos de acceso.	- Obtención de todos los permisos del sistema
Spoofing	Reemplaza una identidad de un equipo de red para conseguir información restringida	Objetivo es el falsificar, adquirir, simular datos	- Phishing
Hijacking	Impide y roba sesiones de usuario con fin de adueñarse de ese servicio	Congela la conexión entre el servidor y cliente para que él envió de paquetes se redirija al atacante	- Intercepción entre el usuario y el servidor
Ingeniería Social	Se basa en el engaño para conseguir información útil	Envíos mediante emails, phishing	- Vulnerabilidades a la red, servidores, usuarios, entre otros

DoS	Impide el flujo de la conexión para que exista una comunicaciones entre el cliente y servidor	Envía pedidos falsos saturando el recurso del sistema	- Colapso en el servidores - Disminución del rendimiento por la cantidad de tráfico que está siendo procesado.

Fuente: Cuadro comparativo de técnicas de hacking (Silva et al., 2011).

2.2.3 Escaneo de puertos

De acuerdo con Mendaño y Hurtado (2016) menciona que, este tipo de escaneo es un proceso donde se revela puertos tanto UDP (User Datagram Protocol), aquel protocolo sin conexión que se ejecuta sobre la IP (número que es asignado a cualquier dispositivo de forma única dentro de la red de) y TCP (protocolo que permite la comunicación con la internet. De esta manera hacen que estos dos servicios corran a través de la red, el cual tienden hacer entradas ocultas de ataque. En este sentido, al momento de la realización de nuestro pentest o pruebas de penetración se hará uso de la herramienta Nmap, que permitirá realizar varios tipos de escaneo de acuerdo con la información que se desee conseguir, con el fin de que el nivel de seguridad sea mayor y tener en cuenta el tipo de escaneo que se requiere ejecutar de la siguiente manera:

- sT: Permite ejecutar un escaneo de los puertos TCP, con el fin de comprobar si sen encuentran archivos activos o puertos abiertos.
- sU: La mayor parte de estos servicios utilizan puertos TCP, pero también es recomendable analizar puertos UPD que podrían ser utilizados para futuros ataques.
- n: No necesita hacer ningún tipo valor a los DNS.
- Pn: Se realizan métodos para conocer si el host se encuentra en buen estado, ya que se conoce por naturalizar que está en funcionamiento.

Además de eso es importante aplicar y reconocer los conceptos de estado de los puertos:

- Abierto: Da referencia puertos que se encuentran libres de conexiones TCP y UDP. ya que son usados como orígenes de ataque.
- Cerrado: No tiene ninguna entrada de alguna aplicación o servicio, aunque pueden ser respondidos a pruebas de Nmap.

- Filtrado: El estado filtrado se le dificulta reconocer si se encuentra abierto o cerrado debido a la filtración de datos, estos puedes ser firewall (su función de proteger a una red), tanto para el enrutado como por el mismo ordenador.
- No-filtrado: En este filtrado se obtiene más acceso, sin embargo no identifica si está abierto o cerrado.
- Abierto-Filtrado: Indica los puertos perteneciente cuando no se puede evidenciar si se encuentra abierto o filtrado.
- Cerrado-Filtrado: Indica a los puertos cuando no se puede evidenciar si está cerrado o filtrado.

Para el escaneo de puertos dentro la investigación se hará uso de la herramienta Nmap como herramienta gratuita que se encuentra alojada en el sistema operativo Kali-Linux, mediante el comando **nmap -PN -sT -Sv dirección sitio web**, se verificara los puertos que se encuentran abiertos y cerrados independientemente de cada servicio que presenta la aplicación web.

2.2.4. Estructura del servidor web

De acuerdo con Chavarría y Gudiño (2017) el funcionamiento de su arquitectura entre cliente servidor web es de la siguiente manera:

- El usuario detalla la URL de la página que se desea consultar.
- El cliente crea una conexión juntamente con el servidor web por medio de internet solicitando la página deseada.
- El servidor encuentra la página en los archivos del sistema, si el sistema lo localiza la traslada al sitio del servidor sino envía un código de error.
- El cliente interpreta el código HTML e indica la página realizada al usuario.
- Se cierra la conexión.
- La conexión se libera al terminar de cargar la página.

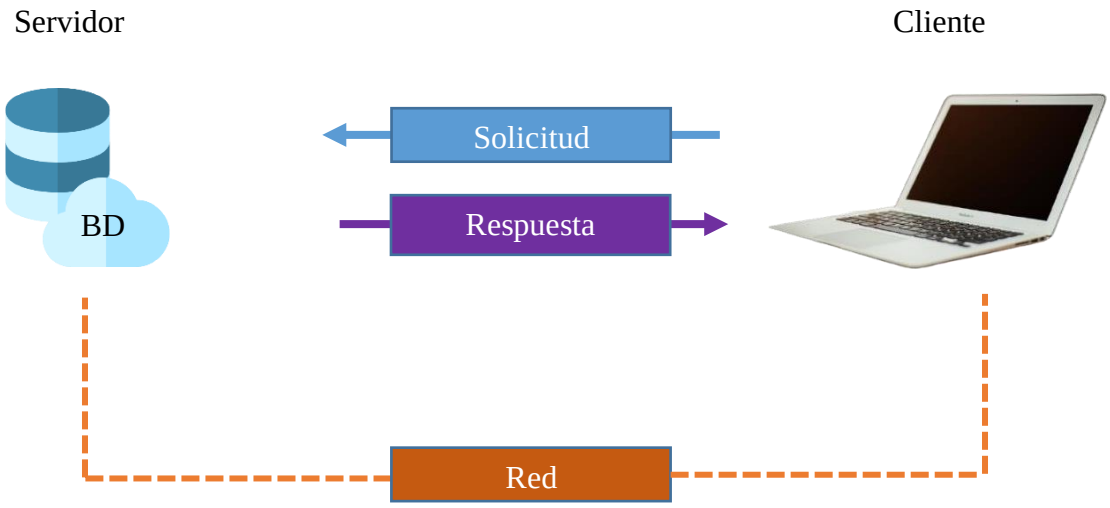


Figura 8. Estructura del Servidor Web

Fuente: Implementación de un servidor web y un diseño de una página utilizando herramientas de software libre (Chavarría y Gudiño, 2017)

2.2.5. Pruebas de penetración (Pentesting)

Como parte de la investigación principal, nos debemos enfocar en las pruebas de penetración o también conocidas como Pentest, donde analizaremos cada punto, con el fin de retroalimentar y conocer todo lo relacionado al Pentest y sus componentes.

De acuerdo con Metso (2019, como se citó en Weidman, 2013) menciona que las pruebas de penetración o pentest son maneras de simular ataques para que el pentester pueda evaluar las amenazas de filtraciones de seguridad. Hay que tomar en cuenta que un pentesting es muy diferente a un evaluador de vulnerabilidad, porque un evaluador solo descubre las vulnerabilidades que pueden ser utilizada por los atacantes, mientras que un pentester manifiesta las amenazas, vulnerabilidades y las explota cuando es necesario.

2.2.5.1. Tipos de pentesting

De acuerdo con (Metso, 2019 como se citó en Weidman, 2013) existen dos tipos de penetración: las pruebas de penetración interna o caja blanca (también conocida como pruebas de caja de vidrio) y las pruebas de penetración externa que se clasifica en la caja negra y caja gris.

- Una persona encargada de **pruebas de penetración de caja blanca** tiene ingreso a la información interna de una entidad, infraestructura y todos los documentos que la organización pueda tener en sus sistemas. Cuando el pentester tiene acceso a los sistemas internos, el proceso que realiza es más profundo y completo, donde maximiza el tiempo de prueba y utiliza más recursos para la fase de pruebas.
- A diferencia de la prueba de **penetración externa caja negra**, se enfoca más en encontrar las vulnerabilidades del servicio de acceso del internet que tiene la empresa, creando un escenario como un atacante llevándolo más tiempo en algunas fases de penetración.

Para la realización y comienzo de aplicar las pruebas de penetración se tomó en cuenta la aplicación de pruebas internas caja blanca con el propósito de realizar un proceso más realista del entorno en el que se está trabajando.

2.2.5.2. Importancia de un pentesting

De acuerdo con Zetina (2014) para una organización, empresa, laboratorio, entidad o entorno que almacene información valiosa como para no ser robado, las pruebas de penetración son muy importantes donde debe ser priorizado desde los archivos más insignificantes hasta lo más importante de una organización. Los pentest permiten priorizar riegos y proponer soluciones, considerando los impactos de ataques, vulnerabilidades y valor en los recursos, de esta manera aplicando las fases de planeación, implementación y administración, sin embargo, otra de las importancias de un pentest es el motivar el cambio hacia un incremento de nivel más seguro, generando conciencia y sentido de protección de los activos y recursos de cada una de sus organizaciones. En este sentido se toma en cuenta cada uno de los pasos, procesos y fases que tiene el pentest con el fin de valorar y definir los análisis adquiridos por esta técnica de seguridad más completa.

2.2.5.3. Ventajas y desventajas de pruebas de penetración

Para poder definir cada punto tanto positivo como negativo de las pruebas de penetración se ha realizado un cuadro comparativo de los tipos de pentest, en este sentido se tomará en cuenta las ventajas más principales que tiene las pruebas de penetración a la hora de aplicar esta técnica de mejoramiento de seguridad. De acuerdo con Tech (2018) menciona que las ventajas de pentest se manifiesta de esta manera:

- Ayudan a establecer medidas de protección más eficaces.
- Favorecen el seguimiento de estándares y certificaciones.
- Garantizan una mejora continua y competitiva del negocio, mejoran la seguridad en las organizaciones.
- Aplica actividades de organización juntamente con herramienta de seguridad digital.

De esta manera procedemos a realizar un cuadro comparativo en donde se identifiquen sus ventajas y desventajas de los siguientes tipos de pentest.

Tabla 5. Tipos de pruebas de penetración

BLACK BOX	
VENTAJAS	**DESVENTAJAS**
El consultor no recibe ninguna información ni permiso autorizado a los sistemas o servicios web	Tiene la probabilidad más baja de detectar vulnerabilidades
Es considerado el más rápido y barato de los servicios, algunos de los casos se pueden generar automáticamente	El principal punto es exclusivamente la infraestructura, dejando atrás los aplicativos
	Simula el mejor caso de la organización por lo que no genera un buen nivel de seguridad
WHITE BOX	
Tiene el mayor alcance de detectar vulnerabilidad	Tiene una mayor duración y costo
Simula y crea escenarios del peor caso para la organización lo cual brinda un nivel alto de seguridad	Requiere un nivel de confianza con el consultor que va a adquirir toda la información y los detalles de la instalación

Fuente: Penetration Testing. ISACA (Guirado, 2017).

Una vez puntualizando cada uno de los conceptos, ventajas y desventajas se analizó la utilidad de las pruebas de penetración para la investigación con el fin de establecer un mejora continua de acuerdo con las actividades y sus procesos que realiza pentest, además la aportación de herramientas de penetración servirán para identificar las vulnerabilidades al servidore web.

2.2.6. Etapas de prueba de penetración

De acuerdo con Imperva (2021) las etapas de pruebas de penetración se dividen en cinco etapas.

Figura 9. Etapas de prueba de penetración

Fuente: Pruebas de penetración (Imperva, 2021).

a. Planificación y reconocimiento

- Definir el alcance, los objetivos de una prueba y los métodos de prueba que se van a utilizar.
- Recopilación de inteligencia, es decir, nombres del dominio y de red, servidor de correo, con el fin de conocer cómo funciona el objetivo y sus vulnerabilidades permisibles.

b. Escaneo

- **Análisis estático:** Reconocimiento del código de una aplicación para comprobar cómo se comporta mientras se ejecuta. Esta función puede escáner la totalidad del código.
- **Análisis dinámico:** Reconoce el código de una aplicación en estado de ejecución. Esta función es la forma más práctica de escanear, ya que brinda una vista en tiempo real del rendimiento.

c. Obtener acceso

En esta etapa utilizas ataques a aplicaciones web, como comando de inyección SQL y puertas traseras para descubrir la vulnerabilidad. Posteriormente a eso se intenta

explotar esas vulnerabilidades, generalmente robando datos, interceptando el tráfico y demás, con el fin de comprender el daño que pueden causar.

d. Mantener acceso

En esta etapa es observar si la vulnerabilidad se puede utilizar para lograr ver una persistencia en el sistema que se explota, la idea es imitar las amenazas y ataques que a menudo permanecen en un sistema durante mucho tiempo para robar datos más confidenciales en una organización.

e. Análisis

Los resultados obtenidos se compilan en un informe que detalla:

- Vulnerabilidades específicas que fueron explotadas.
- Datos sensibles a los que se accedió.
- La cantidad de tiempo que el atacante pudo permaneces en el sitio sin ser detectado.

2.2.6.1. Análisis de riegos

Una vez reconocidas cada una de las vulnerabilidades y amenazas, se realiza el análisis de cada riesgo, el mismo que determinará su impacto. Por lo que, se debe de considerar las variables: probabilidad, impacto e importancia.

Tabla 6. Nivel de probabilidad de salida de una amenaza

Valor	*Nivel*	*Definición*
3	Muy probable	Sucede 1 vez al año y sucedido varias veces
2	Probable	Ocurre varias vez o solo una vez
1	Improbable	Nunca ocurrido, sin embargo es posible que ocurra en algún momento

Fuente: Propuesta de una metodología de pruebas de penetración orientada a riesgos (Álvarez, 2018).

Sin embargo, para localizar la segunda variable del impacto, se debe utilizar (tabla 7).

Tabla 7. Nivel del impacto de materialización de un riesgo

Valor	*Nivel*	*Definición*
3	Alto	Los resultados atacarán los objetivos de la organización.
2	Medio	Los resultados realizaran cambios en la organización.
1	Bajo	Los resultados pueden solucionarse ya sea con normativas, leyes y políticas

Fuente: Propuesta de una metodología de pruebas de penetración a riesgos (Álvarez, 2018)

Finalmente, para identificar la tercera variable, se analiza el nivel de relevancia que tiene cada riesgo (tabla 8).

Tabla 8. Nivel de relevancia de un riesgo

Valor	*Escala*	*Definición*
10	Alta	Elemento de conflicto notable o muy importante para la organización.
5	Media	Elemento de riego de relevancia media.
1	Baja	Elemento de riesgo no relevante.

Fuente: Propuesta de una metodología de pruebas de penetración a riesgos (Álvarez, 2018).

2.2.6.2. Evaluación de riegos

Una vez evaluado los riegos, se ordena de acuerdo con los resultados que se obtuvieron en las variables definidas de análisis de riego. En este sentido se maneja una escala de riesgo que se muestra (tabla 9).

Tabla 9. Nivel de evaluación del riesgo

Clasificación Final	**Riego**	**Color**
De 1 a 10	Bajo	Verde
De 11 a 30	Aceptable	Naranja
De 31 a 90	Alto	Rojo

Fuente: Propuesta de una metodología de pruebas de penetración a riesgos (Álvarez, 2018).

2.2.7. Herramientas de pruebas de penetración

Para el manejo de herramientas en las pruebas de penetración se hará uso en la investigación con las más significativas que demuestren información óptima para identificar cada una de las vulnerabilidades y amenazas en el sistema y su aplicación.

Tabla 10. Herramientas para Pentesting

Herramienta	Descripción
SQLmap	Permite explotar vulnerabilidades de código script, mediante inyecciones de SQL con el fin de conseguir información de la base de datos.
Nessus	Permite escanear vulnerabilidades como son alertas dentro del servidor, como también soluciones de las amenazas encontradas.
OWASP Zap	Monitorea y analiza las vulnerabilidades detectadas en la aplicación web, estableciendo patrones y configuraciones de seguridad.

Fuente: Pentesting para web (Pilar, 2019)

- **Metasploit.** De acuerdo con Díaz (2018) este potente framework maneja una cantidad alta de datos que ejecuta ataques que son acumulados en el sistema operativo, en este sentido gestiona todos los módulos, auxiliares y exploits (programa informático que se aprovecha de un error o vulnerabilidad que provoca una reacción imprevista en el software, hardware o cualquier dispositivo), utilizando base de datos PostgreSQL.

Zaragoza (s.f.) afirma que, Metasploit cuentan con dos particularidades esta puede ser realizadas en todas las plataformas de manera fácil y cómodo, solo es cuestión de elegir el gusto del desarrollador.

Modo Web: Esta modalidad cuenta con interfaces web, se debe aplicar las opciones correspondientes y finalmente presionar en exploit para iniciar con el ataque, además existe la característica de ataque mediante el comando shell, es decir seleccionar el archivo **msfweb.bat**, lo cual permitirá que se despliegue un mensaje como el siguiente:

Metasploit Framework Web Interface (xxx.x.x.x:xxxxx)

Una vez identificado, nos dirigimos al navegador y acceder a la dirección proporcionada por el proxy de Metasploit, y desde esa página realizar los ataques propuestos por el framework.

Modo Consola: Esta modalidad funciona por medio de comando, es mucho más fácil y mejor para ejecutarlo, mediante el archivo **msfconsole.bat** de la carpeta del framework se puede empezar a trabajar. Al abrir la consola Metasploit alguno de los comandos que se utilizara serán los siguientes:

Tabla 11. Comando por consola para Metasploit

Comandos	¿Qué hace?
show exploit	Muestra una lista de **exploits** (una secuencia de comando que aprovecha de un error para provocar una conducta no intencionada) de los cuales selecciona uno de acuerdo con el sistema que desee atacar.
use [exploit]	Se utiliza este comando cuando ya se haya encontrado un **exploit** adecuado.
use msrpe_dcom_ms03_026	Como este comando asigna el exploit Microsoft RPCDCOM dependiendo del sistema que se requiera utilizar
show targets	Una vez ejecutado el anterior comando, se necesitará seleccionar un sistema afectado, es por ello que se utilizara ese comando que indicara todo los sistemas operativos vulnerables.
set	Se utiliza este comando para seleccionar la opción.
set [Variable] [Valor]	Estructura para seleccionar una opción.
set TARGET 0	Con este comando dice que la variable de **target** (objetivo) es igual a 0, es necesario que la variable sea en mayúsculas ya que si no se coloca así podría dar algunos errores.

show payloads	Una vez teniendo identificado los **exploit** y sistema que se va a atacar, solo faltaría indicar que tipo de ataque se utilizar, es por ello que se utiliza este comando.
Existen variedad de tipos de ataques en la lista, los más comunes son los siguientes:	
win32_adduser	Añade usuarios con autorizaciones altas al sistema que tratemos de vulnerar
win32_bind_vncinject	Aparecerá una pantalla para el usuario con la que se podrá manipular el equipo remotamente
win32_downloadexee	Descargara archivos mediante un servidor web o **ftp** y lo procesara, es decir pueden venir **troyanos**, **malware** entre otros.
win32_exec	Ejecutará un comando a través del **CMD** (herramienta interprete de comandos para configuraciones del sistema).
win32_reverse	Nos brindara una **shell inversa** (crea un túnel entre maquina local y una maquina remota, para poder ejecutar comando a través de ello) por si tiene **firewall** (dispositivo de seguridad donde decide si permite o bloquea el tráfico de red) y no se podrá abrir puertos.
win32_reverse_vncinject	Otorgará un **VNC inverso,** es decir el ordenador a controlar se conecta a ti por si tiene firewall y no se pude abrir puertos.
SET [Variable] [Valor]	Una vez que se ha identificado que tipo de ataque se quiere utilizar se aplicara esta estructura.
set PAYLOAD win32_reverse_vncinject	Mediante este comando Metasploit hace uso de **meotod win32_reverse_vncinject**, hay que tomar en cuenta que las variables este en mayúsculas.

Las otras alternativas se las puede añadir intuitivamente dependiendo del exploit o ataque que se ha seleccionado	
set RHOST IPVIctima set RPORT PuertoDeLaVictima	El comando **RPORT** tiene el valor por default, no cambia al menos de que sea necesario, es decir que la víctima tengo el servicio otro puerto.

Fuente. Adaptado de (Zaragoza, s.f.). Manual Básico de Metasploit. hackpr.net

Finalmente, una vez aplicada todas las variables el último paso es colocar **exploit** y esperar la respuesta de Metasploit.

- **Honeypot.** Este sistema sirve para analizar cómo un hacker aplica sus ataques para intentar ingresar al sistema, alterar, copiar, dañar los datos o la información. De esta manera ayuda a detener al atacante o aprender del comportamiento sin necesidad de que se enteren que están siendo vigilados. Honeypot generalmente es utilizado en sistemas de vigilancia, así como mecanismo de alerta, en este caso simula un ambiente tentador que capte la atención del atacante como por ejemplo un servidor de bases de datos o de front, servicios donde son más propensos a ser atacados y genere un reto para al atacante al momento de ejecutarlo (Arenas y López, s.f.).

De acuerdo con Ochoa (2018) afirma que, se debe de tomar en consideración a la hora de realizar esta implementación donde se debe tener controlados los intentos de ataques tanto internos como externos, el inconveniente prioritario se centra en que debe tener una configuración especial de parte del Firewall con el objetivo de tener acceso al Honeypot, al no ser bien aplicado podría ocasionar brechas de seguridad en la filtración del tráfico, alguno de estos diferentes Honeypot sirven para varios usos a la hora de su aplicación, como son:

- **Sepecter**: Es un Honeypot de baja interacción, brinda procesos como PHP, SMTP, FTP, POP3, HTTP Y TELNET que atraen con facilidad intrusos, sin embargo son servicios con trampa para recolectar la información.
- **Honeyd:** De igual manera es un Honeypot de baja interacción, donde crea hosts virtuales en la red y llegar a ser configurados para elaborados en distintos procesos.
- **Kippo:** Es un sistema de interacción media que es emulado por un servicio conocido como SSH y vigila toda la interacción y procesos realizados por el atacante, esta

herramienta se encuentra ya descontinuada es por ello que crearon una más nueva llamada sucesor cowrie, donde provee más funciones.

- **Kali Linux.** Kali Linux es un sistema gratuito que es financiado por seguridad ofensiva (Offensive Security), este sistema es una distribución que se enfoca principalmente para la elaboración de pruebas y auditorias de seguridad. Kali Linux es un sistema operativo que incluye más 600 herramientas de pruebas de penetración que están dentro del sistema, haciendo más factible para un pentester, ya que poseen todas las herramientas y técnicas necesarias para la realización de un pentesting, sin la necesidad de contar con otros procedimientos (Metso, 2019).

De acuerdo con la OffSec Services Limited (2021) menciona que, Kali Linux está diseñado propiamente para las necesidades de profesionales dedicados a la realización de las pruebas de penetración, las características más destacadas que brinda Kali Linux son:

- **Personalización completa de las ISO de Kali:** Permitiendo una flexibilidad para personalizar y adaptar cada aspecto del sistema operativo, ya sea creando una imagen de Kali de puente de red, auto instalación, VPN inversa y conexión automática.
- **Arranque USB en vivo:** Permitiendo colocar Kali en un dispositivo y arrancarlo sin tocar el sistema operativo host, de esta manera permitiendo que los archivos se guarden entre sesiones, en este sentido también existe la opción nuclear **LUKS**, que controla rápidamente la destrucción de datos.
- **Kali en cubierto:** Es adecuado para no destacar al combinarse con un sistema operativo familiar.
- **Kali NetHunter:** Es especialmente para teléfono Android donde contiene elementos como una superposición ROM para múltiples dispositivos, creando una experiencia de usuario completa para el profesional de pentest.
- **Kali ARM:** Ofrece imágenes pre generadas, listas para ser utilizadas, así como scripts de compilación.
- **Estandar de la industria:** Es la plataforma más precisa para el desarrollo de pruebas de penetración de código abierto que existe hasta la actualidad.

¿Por qué usar Kali Linux?

Kali Linux cumple con requisitos de pruebas de penetración profesional y auditoria de seguridad, para lograr que el sistema operativo cumpla con las expectativas reflejadas deben ejecutar estas necesidades OffSec Services Limited (2021).

- **Servicio de red deshabitados de forma predeterminada:** Permitiendo instalar varios servicios, permaneciendo segura de forma predeterminada, sin importar que paquetes estén instalados.
- **Kernel Linux personalizado:** Kali Linux usa un Kernel ascendente (función no negativa, con valores reales.
- **Un conjunto mínimo y confiable de repositorios:** Mantiene la integridad del sistema en clave, manteniendo seguro cualquier tipo de riegos hacia el sistema operativo.

La utilización de Kali Linux en nuestra investigación se debe al conocimiento de los siguientes puntos.

- Si de seguridad se habla kali Linux se convierte en una opción preferida por los usuarios.
- Su estabilidad en su funcionamiento.
- Dispone de un entorno de más de 600 aplicaciones de hacking ético y seguridad.
- El objetivo de kali Linux es contar con herramientas potentes para la seguridad informática.
- Maneja análisis forense, ingeniería inversa y evaluación de vulnerabilidades.
- Los paquetes disponibles de Kali Linux se basan de la distribución de Debian conocida por su calidad y estabilidad.
- Cuenta con 400 mil descargas al mes.
- Soporte para ARM y Android.
- Kali Linux cuando con el análisis forense más popular dentro del mercado hasta la actualidad.

- **Nmap.** De acuerdo con Arenas y López (s.f.) menciona que, Nmap es un sistema que se encuentra dentro de Kali Linux, se encarga de escanear fácilmente host, servicios, filtros de paquetes, corta fuegos entre otras características. Al ser una herramienta muy poderosa cuentan con desarrolladores y usuarios especializados para gestionar y brindar soporte.

Nmap proporciona paquetes de IP sin procesar que determinen que hosts están abiertos en nuestra red, Nmap está diseñado para escanear rápidamente redes grandes funcionando bien con hosts únicos, este paquete está disponible para Linux, Windows y MacOS, tomando en cuenta que esta herramienta fue nombrada producto de seguridad más destacado del año por las compañías de Linux, Info World, Linux.Org y Codetalker Digest. A demás de formar parte de doce películas como Matrix, Die Hard The Bourne Ultimatum, entre otros (OffSec Services Limited, 2021).

De acuerdo con Nmap.org menciona que para poder instalar Nmap en Linux se requiere de una versión con formato .tar.bz, posteriormente colocar el siguiente comando:

1. **bzip2 -cd Nmap-** <VERSION>.tar.bz2 | **tar xvf –**
2. Una vez creado se cambia el directorio recién creado: cd Nmap-<VERSION>
3. Se configura el sistema de compilación: **./configure**

Si la configuración cumple con todos los procesos, aparece un dragón de arte ASCII indicando que la configuración se ha logrado con éxito, como también advirtiéndole que tenga cuidado.

Nmap utiliza biblioteca Libpcap donde captura paquetes IP, se debe considerar los paquetes binarios de Nmap ya que deben estar disponible en las plataformas y deben ser fáciles de instalar, la desventaja es que no pueden estar actualizados y hará que pierda flexibilidad en la compilación automática.

Nmap funciona desde el sistema operativo Kali Linux, una vez realizada la respectiva instalación, si se encuentra con los módulos y paquetes actualizados no es necesario su instalación caso contrario requerirá su instalación. A continuación se coloca el comando Nmap siguiendo de apostrofes que tienes diferente significado, posteriormente la url de la página que se quiere escanear y finalmente Nmap hará su proceso de escaneo.

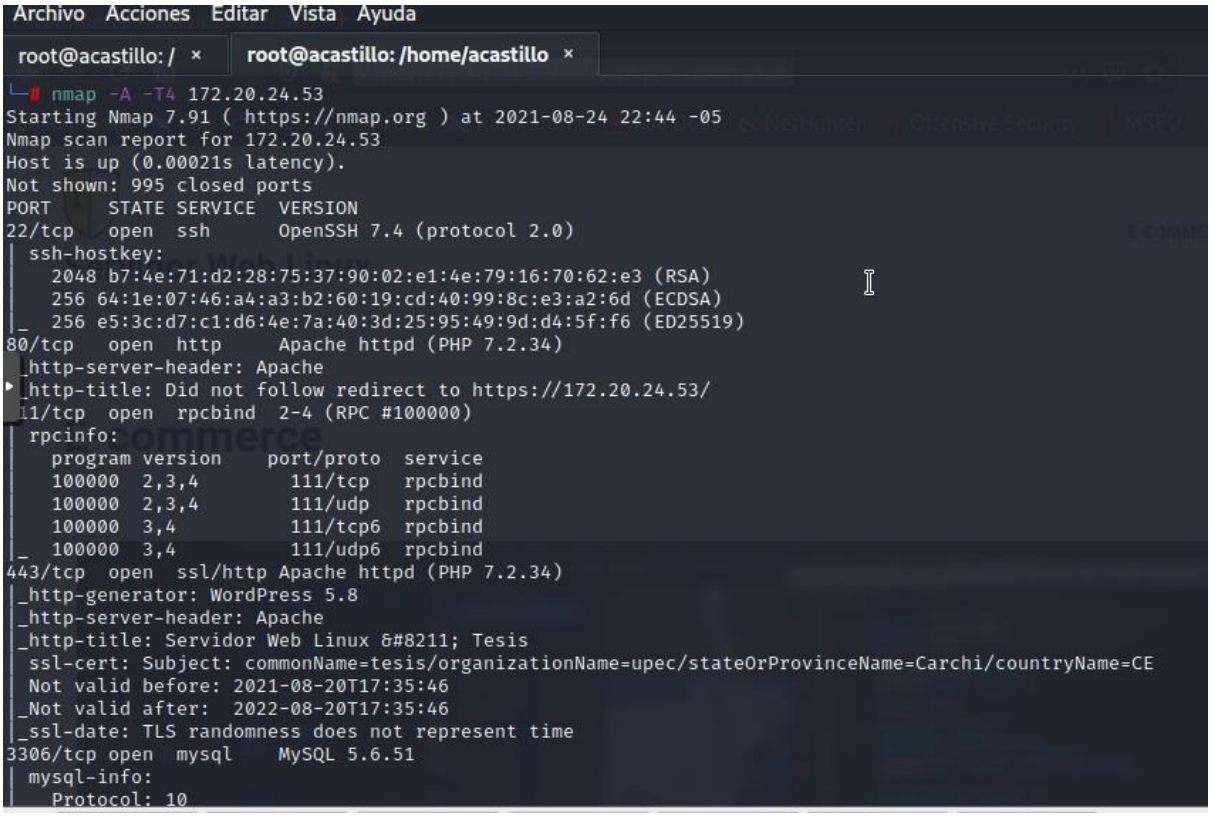


Figura 10. Nmap

- **Vulscan.** Es un módulo que se utiliza para mejorar las capacidades al momento de escanear vulnerabilidades dentro de una red. Estos scripts donde forman parte de la herramienta Nmap utilizan ayudan a encontrar las amenazas en un objetivo único o una red. Estos scripts se centran en la detección de servicios, con el fin de evaluar las vulnerabilidades dependiendo de la maquina o red de destino (Mohammed y Mahmud, 2018).
- **Nesuus.** Es un programa que escanea vulnerabilidades, donde el cliente indica el informe sobre el estado del escaneo, como también de manera programada que se la puede realizar a través de una consola. Alguna de sus características principales e importantes es la actualización permanente, el reporte de riegos con sus respectivas características, el escaneo simultaneo de varias máquinas, permitiendo así integrarse con otras herramientas como Nmap y Metasploit (Mendaño y Hurtado, 2016).

De acuerdo con García (s.f.) Nessus es una herramienta que se basa en el modelo cliente-servidor contando con un propio protocolo de comunicación, su función es el de explotar y probar ataques realizados por el servidor Nessus (nessusd), de esta manera las funciones de control, generación de informes y presentación de los datos son procesadas por el cliente (nessus), brindando una exploración proactiva a los sistemas de la red en busca de servicios que

están siendo atacados, finalmente Nessus avisara deficiencias relacionadas como las que se presentan a continuación:

- Utilización de servidores no actualizados, con escases de seguridad conocidas como sendmail, finger, wu-fitpd, etc.
- Fallas de seguridad relacionas a las malas configuraciones de los servidores como permisos de escritura para usuarios.
- Fallas de seguridad mediante implementación TCP/IP hacia el equipo remoto.
- Manejo de aplicaciones CGI hacia servidores web con malas configuraciones o codificaciones que limitan una brecha de seguridad con el sistema que los aloja.
- Instalación de troyanos, malwares, denegación de servicios DDoS u otros servicios extraños.

- **Owasp Zap.** Es un programa de software libre con código abierto que es manejado como un sistema de seguridad específico para la elaboración de pruebas de penetración, al ser una herramienta específica para realizar pruebas de vulnerabilidad en servidores web, se lleva a cabo de la siguiente manera, una vez descargado Owasp Zap, se accede al siguiente directorio (Sarmiento y Rodríguez, 2019).

/usr/share/zaproxy

En donde aparecerá librerías, filtro y los lenguajes a utilizar, de todos los archivos salientes escogeremos la que dice **"zap.sh"** con el fin de poderla ejecutar owasp desde cualquier plataforma **Linux y zap-2.7, 0.jar** para poder ser ejecutado desde cualquier sistema operativo. Existen dos maneras para que arranque la herramienta desde jar que sería java **-jar zap-2.7.0.jar** y desde el script **./zap.sh**.

Una vez realizado las configuraciones y comandos antes ejecutados, solamente se coloca el url que se desea escanear y listo, sin embargo, hay que tomar en cuenta que, se debe especificar el protocolo es decir si la página sera por http o https, caso contrario saldrá error.

Owasp Zap para mi investigación fue de total importancia ya que al tener una conexión con la metodología Owasp me permitió profundizar el manejo de esta herramienta juntamente con la metodología aplicada, y a partir de ahí realizar los procesos y fases que describe Owasp. Por otra parte se seleccionó esta herramienta por las siguientes características más relevantes que son:

- Es multiplataforma de código abierto.
- Existe demasiada guías de ayuda para manejar la herramienta.
- Fácil de usar.
- Esta traducida en 20 idiomas.
- Es basado en una comunidad sin fines de lucro por organizadores dedicados a la mejora de la seguridad.

- **Kismet.** De acuerdo con Yacchirema et al. (s.f.) menciona que, Kismet es un detector de redes inalámbricas, puede detectar ataques desde la capa de enlace, datos y red, de esta manera trabaja con tarjetas inalámbricas que soporten el monitoreo de tráfico.

Kismet no envía paquetes detectables, lo que logra detectar la presencia de algunos puntos de acceso y usuarios inalámbricos (Narváez, 2019).

De acuerdo con Narváez, 2019, como se citó en McCloure, Scambray y Kurtz (2010) menciona que, para empezar usar Kismet se debe instalar los controladores personalizados, necesarios para ejecutar de modo supervisor o modo del sistema, esto dependiendo de la tarjeta inalámbrica, sus características que tiene esta herramienta son:

- Permite exportar el tráfico captura en Wireshark
- Disponible en Linux y soporte limitado en Windows
- Brinda conexión GPS, útil para búsquedas de redes desde un vehículo en movimiento).

- **InSSIDer.** InSSIDer nos brinda un análisis más profundo donde se puede cambiar el tipo de protocolo de 802.11b que es el actual a un 802.11n, con el propósito de optimizar exponencialmente el rendimiento de una red y de la misma manera poderla optimizar en nuestras redes que intervienen (Gonzales, 2018).
- **Armitage.** Es una interfaz de usuario más amigable que la de Metasploit, al ser una herramienta gráfica del frameworks Metasploit permite explotar vulnerabilidades a cualquier equipo que esté en una red a la que se tenga acceso, en este sentido Armitage facilita el acceso a los procesos de los exploits, así como también se revelara toda la información de su víctima (Fernández, 2019)

- **WPScan.** De acuerdo con León (2013) WPScan es un sistema para realizar escaneo de vulnerabilidades donde realiza su inspección en sitios que son desarrollados en WordPress, brindando seguridad al evaluar y analizar el grado de se encuentran las aplicaciones que se desarrollan a través de gestor de contenido indicado, obteniendo datos importantes como es: versionamiento actualizados, plugins instalados, nombres de los usuarios y vulnerabilidades comunes, en este sentido se maneja componentes principales que se describen a continuación.

- URL: Permite ingresar la página web.
- Enumérate: Permite realizar la enumeración
    - u: Usuarios
    - vp: Plugins vulnerables
    - vt: Temas vulnerables
- Proxy: Permite acceso de un proxy
- Wordlist: Lista de palabras que realiza ataque de fuerza bruta sobre los usuarios encontrados
- Verbose: Permite ver detalladamente el completo proceso en tiempo de ejecución

Dentro de la investigación se utilizará WPScan para escanear los sitios web que se encuentran instalados en el gestor de contenido Wordpress, con el fin de realizar pruebas de penetración y conocer su versionamiento, como también la utilización de Wordlist para la ejecución de fuerza bruta.

- **OpenVas.** Es un sistema que provee de varios servicios y herramientas adquiridas de Nessus (escáner de vulnerabilidades), funcionales tanto en Linux como en Windows. Permite escanear y analizar vulnerabidades en servidores desde un ordenador cliente (Burbano, 2019) .

Este framework puede ser utilizado individual o como función de las herramientas de seguridad GPL (ficha de seguridad) de OSSIM (Open Source Security Information Management) herramienta cuyo valor es la detección y prevención de intrusos (Carrasco, 2020).

2.2.8. Metodologías

2.2.8.1 Metodología "OWASP (Open web Application Security Project)"

El Proyecto Abierto de Seguridad en Aplicaciones Web (OWASP por sus siglas en inglés) es una comunidad dedicada a brindar a las organizaciones, a que adquieran y conserven aplicaciones y APIs en las que confiar (OWASP Foundation, 2017).

Esta metodología se basa en la contribución de datos por empresas que son netamente especializados en la seguridad de aplicaciones; a través de ranking de debilidades hacia sitos web que sucede con mayor frecuencia en internet, es una de las tantas colecciones de datos sobre vulnerabilidades más grandes que se haya conseguido coleccionar de manera pública. Estas vulnerabilidades son recogidos por cientos de organizaciones, así como también más de cien mil aplicaciones y APIs del mundo en la actualidad. Las principales categorías son escogidas y priorizadas mediante datos de prevalencia, con consecuencias consensuadas de explotabilidad, detectabilidad e impacto; con el fin de educar a los desarrolladores, diseñadores, arquitectos, gerentes y organizaciones sobre técnicas básicas para protegerse contra debilidades comunes e importantes, así como también de problemas de riesgo alto ofreciendo orientación para continuar con su aplicación.

A través de su aplicación los atacantes pueden utilizar diversas rutas para perjudicar su organización. Cada una de estos camino (Figura 11) presenta un peligro que puede o no ser significativo. Es importante conocer los riesgos para una organización porque las consecuencias pueden ser muy graves, como quedarse en quiebra por ejemplo (Mora, 2017). El riesgo general se determina mediante la evaluación conjunta de la probabilidad de cada amenaza, vector de ataque, debilidad de seguridad y concertar con una evaluación del impacto (OWASP Fundation, 2017).

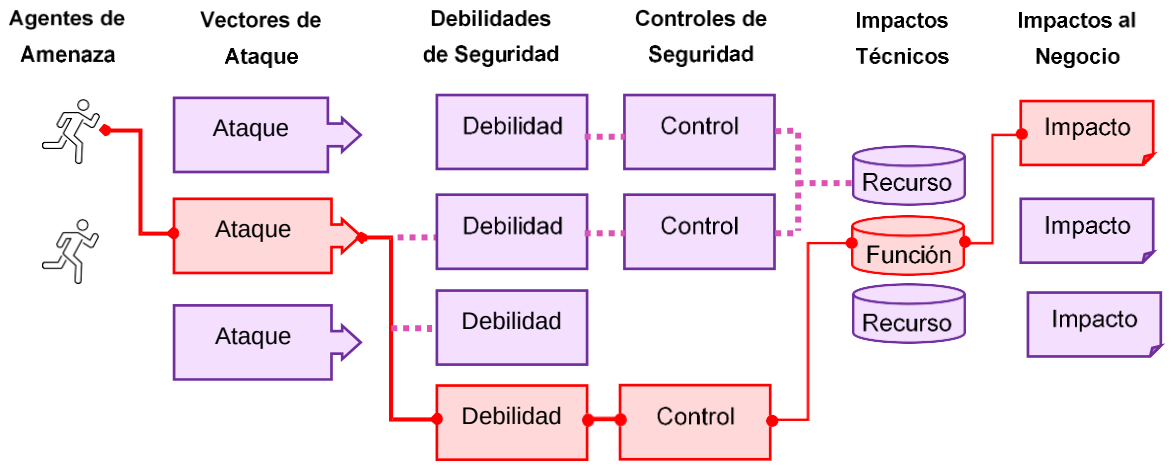


Figura 11. Criterios de riesgos

Fuente: Metodología Owasp 2017

De esta manera la metodología Owasp sera parte fundamental para el desarrollo de nuestra investigación ya que se la selecciono por sus procesos fáciles de organización y detección de vulnerabilidades con el fin de mejorar y aumentar el nivel de seguridad al sistema informático. Por otra parte Owasp es la única metodológica que se encuentra bien documentada a la hora de seguir las fases tales como: recolección de información, test de manejo de configuración y desarrollo, test de manejo de identidad, test de validación de entradas, test para proteger la seguridad al servidor web y finalmente el test de resultado donde arroja mediante porcentajes las vulnerabilidades con más riesgos en el sistema que es analizado.

2.2.8.2. Diagrama de flujo de la metodología.

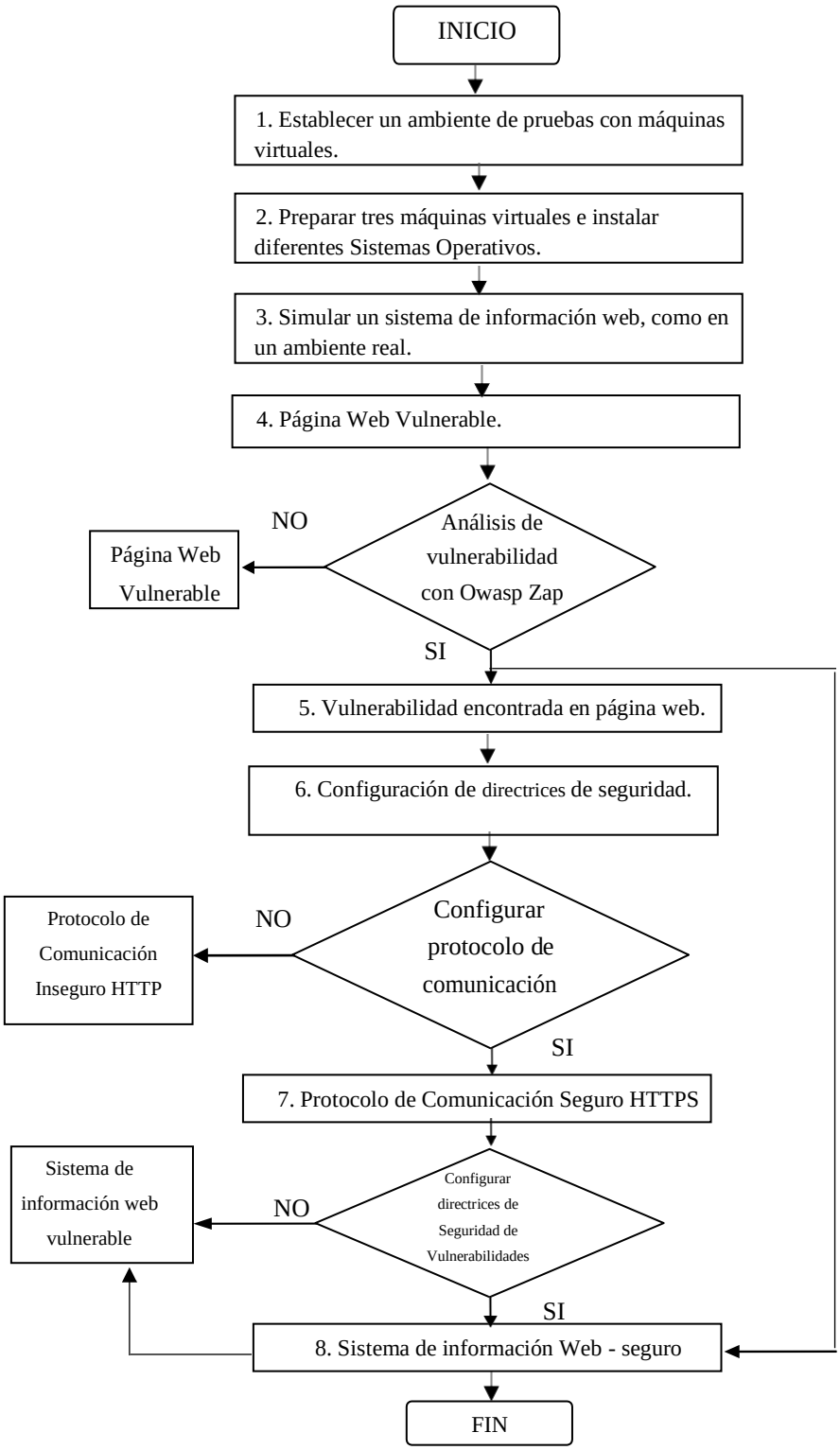


Figura 12. Diagrama de flujo de la metodología

Fuente: Metodología Owasp (Chiquito, 2016)

Cada organización es diferente, por consiguiente, estas amenazas, los objetivos y el impacto también son únicos, por lo tanto, es imprescindible comprender el riesgo para el laboratorio de ciberseguridad de la UPEC. En este sentido, OWASP se enfoca en diagnosticar e identificar los riesgos más críticos de manera que provee información valiosa sobre probabilidad e impacto, utilizando criterios de evaluación (Tabla 12).

Tabla 12. Criterios de evaluación de riesgo

Agente de Amenaza	Explotabilidad	Prevalencia de Vulnerabilidad	Detección de Vulnerabilidad	Impacto Técnico	Impacto de Negocio
Específico del Sitio web	Fácil 3	Difundido 3	Fácil 3	Severo	Específico del Laboratorio
	Promedio 2	Común 2	Promedio 2	Moderado 2	
	Difícil	Poco común 1	Difícil	Mínimo 1	

Fuente: Open web Application Security Project (2017)

2.2.9.1. Metodología de Buenas Prácticas aplicado a los servidores web

Como parte metodológica se enfocará en la elaboración del manejo de bases, diseños que orienten y evalúen los riegos que se observan a la hora de proceder con las técnicas de pentest para el mejoramiento de la seguridad, en este sentido se tomará los demás tipos de metodologías que se utiliza a la hora de implementar la técnica de pruebas de penetración; los principales y más importantes dentro del entorno de seguridad son: OSSTMM, **OWASP**, PTES, ISSAF, CVSS y la realización de las buenas prácticas en el ámbito de la aplicación de pentest en una organización, una vez descrito cada uno de las metodologías se empleara la metodología de las OWASP con el fin de proponer registro de indicaciones que serán seguidos al momento de mejorar el nivel de seguridad evitando la entrada de amenazas en un porcentaje considerable, tomando en cuenta seis aspectos importantes que se deberá considerar (Álvarez, 2018).

- Ámbito y enfoque: Organización que la utilizara.
- Alcance: Todas las tareas que pude abarcar la metodología.
- Profundidad: El detalle con la que trabaja la metodología.
- Usabilidad: Y fácil uso y manejo de la metodología.
- Métricas: Mediar las vulnerabilidades encontradas.
- Evaluación de riegos: Mide el nivel de riegos que se encuentra y como poderlo mitigar el impacto.

2.2.9.2. Buenas prácticas

Son aquellos procesos que cumplen premisas o procesos de evaluación obteniendo los resultados bien definidos y de esta manera tener en cuenta a la hora de evaluar las fortalezas de la seguridad de una organización, en este sentido se aplica las buenas prácticas con el fin de brindar una evaluación y estructurada definida donde ayudara exponencialmente a fortalecer las defensas y eliminar cualquier tipo de amenaza (Díaz, 2018).

2.2.9.3. Proceso de buenas prácticas

De acuerdo con (Jaramillo y Riofrío, 2015) menciona que, el analista de pentesting debe cumplir las siguientes premisas de seguridad.

- Identificar con el encargado de la organización, como se encuentra el nivel de riego puede ser: alto, medio y bajo, el cual se hará uso para la calificación de las vulnerabilidades.
- No debe existir la ejecución por más de una herramienta por objetivo de prueba, con el fin disminuir la explotación de vulnerabilidades que generen errores o fallas en los servidores y dispositivos de conexión.
- La misma prueba se puede realizar más de una vez sobre los servidores, con distintas herramientas, para intenciones de comparación y optimización de resultados.
- Realizar un registro de las vulnerabilidades identificadas por cada objetivo de prueba.
- Las pruebas para la detección de vulnerabilidades deben ser realizados con cuidado para que no haya la existencia de caídas en los servidores, ciclos inactivos u otros problemas que pueden surgir de manera imprudente. Por lo cual el analista deberá previamente establecer que se requiere para ejecutar la prueba en un ambiente controlado.
- En ninguna ocasión se autoriza al pentest a divulgar información que conozca de la empresa en el desarrollo de la ejecución.
- Responder el cumplimiento de las políticas de la organización, sus procedimientos, reglas y manuales.
- El analista deberá estar directamente vinculado con la ejecución del proyecto, durante la realización de cada escenario de las pruebas.

2.3.1. Vulnerabilidad al sistema

Según Díaz (2018) menciona que, una debilidad al sistema informático y tecnológico, podría ser manejada para ocasionar algún tipo de daño. Estas debilidades mencionadas tienden a encontrarse en elementos tanto en el hardware, software o sistemas operativos. Estas vulnerabilidades pueden ser aprovechadas por usuarios malintencionados que puede ser capaces de afectar el funcionamiento del sistema, en algunos de los casos tienden la capacidad de alcanzar un cierto control esquivando algunas de las seguridades comúnmente definidas. Si el programa adquiere privilegios mal intencionados de vulnerabilidades constituiría en un fallo grave de seguridad para la organización.

2.3.2. Hackeo ético

De acuerdo con Verdesoto (2007) menciona que, los hackers éticos son personas que se encargan totalmente a evaluar las amenazas de una organización, que se encuentran comprometidos día a día, de esta manera estos tipos de hackers poseen manejan muy bien el tema, con habilidades técnica e informáticas muy confiables a la hora de realizar cualquier tipo de análisis.

2.3.3. Clasificación de los hackers

- **Hackers de sombrero negro.** De acuerdo con Cabezas (2020) menciona que, Black Hat tienen una amplia técnica a la hora de ingresar a las redes informáticas y vulnerar los niveles de seguridad, utilizando malware con el fin de obtener acceso a los sistemas. El propósito de estos hackers es obtener beneficios personales o económicos como también pueden estar involucrados en espionajes cibernéticos sin autorización ni permiso del propietario.
- **Hackers de sombrero blanco.** De acuerdo con (Cabezas, 2020) menciona que, White Hat son más conocidos como hackers éticos, se especializan en seguridad, el cual usan sus habilidades para buscar los problemas de seguridad por medio de piratería, emplean la misma técnica que la de los sombreros negros, con la única diferencia que lo hacen con la autorización del propietario del sistema, teniendo en cuenta el proceso legal. En este sentido una de las características que brindan los sombreros negros es el notificar a la organización su correcta corrección y buen funcionamiento.
- **Hackers de sombrero gris.** De acuerdo con (García, 2014) menciona que el hacker sombrero gris recibe información parcial sobre la red de la entidad, el cual simula una

cuenta de acceso a la red interna contando con privilegios limitados, permitiendo así evaluar las amenazas internas de los empleados hacia la organización.

2.4.1. Clasificación de las vulnerabilidades

Tabla 13. Categorización de riesgos

Categorización Riesgo de vulnerabilidades			
Alta	Media	Baja	Info

Fuente: Ejecución de una prueba a la página web a los servidores aplicando metodología (Sarmiento y Rodríguez, 2019).

De acuerdo con Sarmiento y Rodríguez (2019) las vulnerabilidades se las menciona de la siguiente manera:

Tabla 14. Nivel para medir las vulnerabilidades

Vulnerabilidad alta.	Logra poner en riesgo la vulnerabilidad, integridad y disponibilidad de los datos, de manera que el impacto puede ser alto o catastrófico.
Vulnerabilidad media.	Este tipo de riesgo puede ser solucionable con configuraciones, auditorias o metodologías.
Vulnerabilidad baja.	Este riesgo es muy fácil de detectar y solucionar, el impacto es mínimo y no afecta a gran mayoría al usuario.
Vulnerabilidad Info.	Este riesgo arroja información de impacto mínimo, sin embargo la información puede ser valiosa para completar algunas de las vulnerabilidades de amenazas críticas.

Se hará usabilidad de los criterios de riesgo en la investigación con el fin de evaluar los riesgos encontrados en los servidores web mediante la herramienta Owasp.

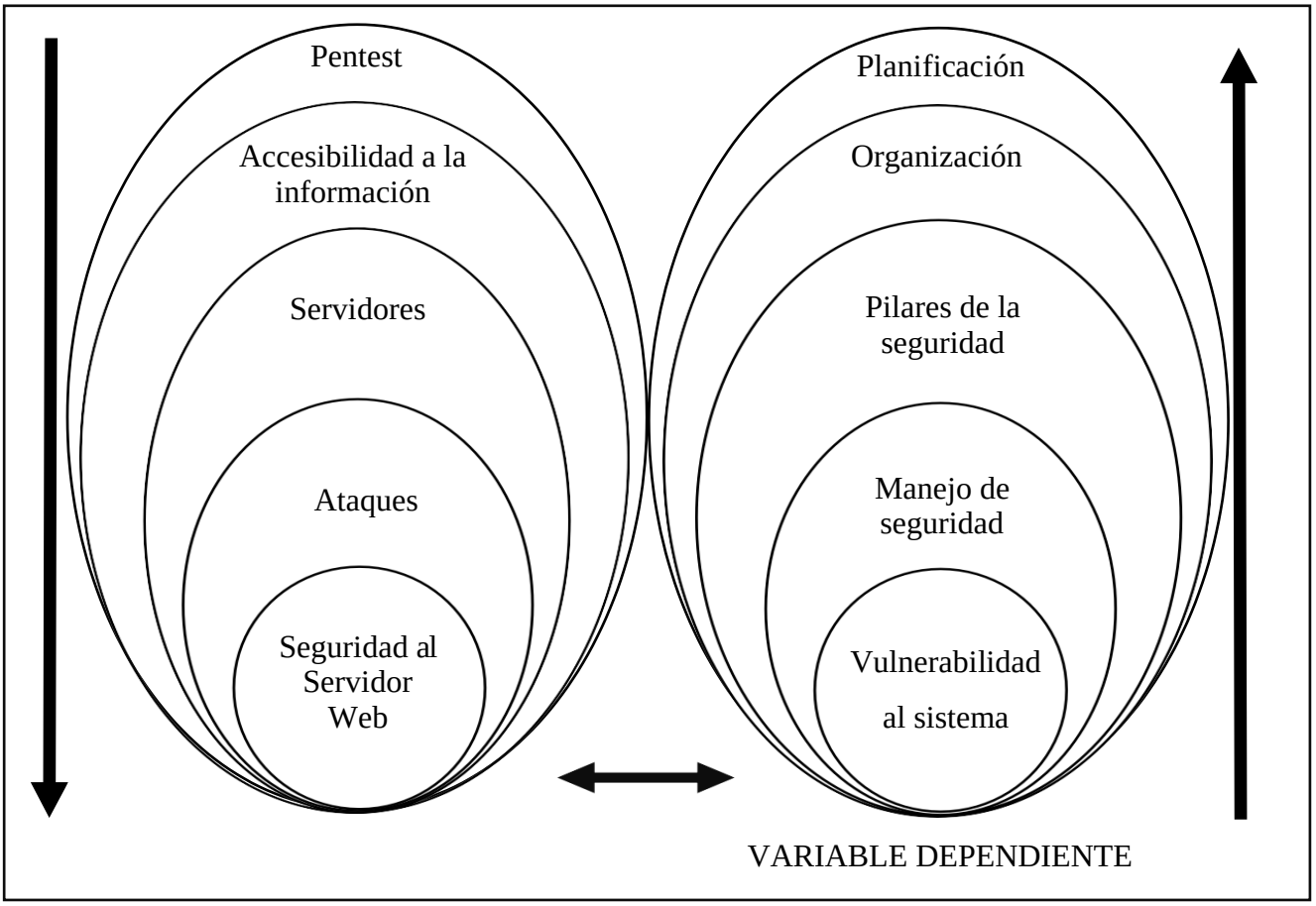


Figura 13. Variables dependientes e independientes

III. METODOLOGÍA

3.1. ENFOQUE METODOLÓGICO

En el enfoque cualitativo se utiliza para establecer e identificar estrategias, que acercan a la observación y evaluación del fenómeno de estudio. Con el propósito de brindar conocimientos con fundamentos en una base de análisis.

3.1.1. Enfoque cualitativo.

Para el marco metodológico el presente proyecto utilizó un enfoque cualitativo el cual permite analizar la realidad estudiada acerca del uso de pruebas de penetración (Pentest) para el análisis al laboratorio de ciberseguridad y determinar como la incidencia de esta afectan su ejecución. Además, la dependencia entre la investigación y el fenómeno estudiado se trata de una relación de interdependencia porque el investigador influye directamente en el desarrollo y ejecución de estos procesos de Pentest. En el proyecto realizado se manejó técnicas de recolección de información cualitativa como la entrevista al director del laboratorio, como también un análisis de vulnerabilidad al sitio de orientacionprofesionalupec con el fin de recopilar información relevante acerca de sus procesos.

3.1.2. Tipos de Investigación.

3.1.2.1. Investigación de Campo.

Se maneja esta investigación dado que la relación entre los problemas encontrados por el investigador y los tratados en el servidor web en el laboratorio de ciberseguridad es directa, para ello se llevará a cabo la utilización del método de observación directa el cual servirá para obtener un diagnóstico sobre los objetos de estudio. Se manejo este tipo de información ya que se trabajó juntamente con el laboratorio de ciberseguridad dentro de la Universidad Politécnica Estatal del Carchi, de esta manera se realizará las constantes configuraciones de los diferentes servidores y a su vez la comprobación del estado de rendimiento de cada uno de ellos.

3.1.2.2. Investigación Descriptiva.

Se llevará a cabo al describir el contexto al lugar donde se realizará la investigación, obteniendo características de los datos que maneja el laboratorio de ciberseguridad, permitiendo centralizar la información de mejor manera.

Se trabajo de esta forma ya que se puntualizó el análisis de las vulnerabilidades en los sitios web, como primera petición se pidió el respectivo permiso para la creación de los servidores y poder realizar el proyecto en el laboratorio de ciberseguridad, se hizo la entrevista con el director del área, para conocer los procesos manejados, las normativas y metodologías y de esta manera comenzar con la ejecución de las técnicas de vulnerabilidad, además se comparó los servidores que iban hacer analizados para evaluar su eficiencia al momento de instalar y configurar un servidor web, cumpliendo con criterios de evaluación de un total 8 criterios correspondientes a cada una de las fases propuestas por la metodología Owasp, de tal manera que se ejecute las pruebas oportunas y la documentación de todo el proceso.

3.1.2.3. Investigación Documental

Es de vital importancia tener un apoyo en distintos textos y medios de información como libros, revistas, artículos científicos, de igual manera medios digitales como internet, los cuales han facilitado la percepción de los fenómenos de estudio como también la visión sobre el análisis de vulnerabilidad y los procesos de pruebas de penetración en el laboratorio de ciberseguridad. Se manejo diversas fuentes bibliográficas como artículos científicos, tesis, libros, revistas, entre otros recursos que apoye la investigación que se realizó.

3.1.2.4. Investigación Exploratoria

La investigación exploratoria se empleó porque permitió analizar el fenómeno de estudio directamente con la realidad, específicamente en el laboratorio de ciberseguridad dentro de la Universidad Politécnica Estatal del Carchi en la ciudad de Tulcán con el objetivo de recolectar datos que permitan plantear una solución y generar nuevos procesos que pueden ser utilizados en futuros trabajos investigativos.
Se utilizo la investigación exploratoria en un nivel medio tratando de retroalimentar la investigación y el conocimiento obtenido previamente.

3.2. IDEA A DEFENDER

El uso de pruebas de penetración al servidor web en el laboratorio de ciberseguridad dentro de la Universidad Politécnica Estatal del Carchi en la ciudad de Tulcán integrará los procesos de seguridad.

3.3. DEFINICIÓN Y OPERACIONALIZACIÓN DE VARIABLES

3.3.1. Definición de Variables

- **Variable independiente:** Seguridad al servidor web
- **Variable dependiente:** Vulnerabilidad al sistema

Para esta investigación se han definido las siguientes variables: Seguridad al servidor web y vulnerabilidad al sistema; la seguridad al servidor web es una variable independiente cuantitativa continua y la variable vulnerabilidad al sistema es una variable dependiente cuantitativa continua. Para las variables se utilizarán las dimensiones e indicadores que se detallan en el cuadro de operacionalización de variables.

3.3.2. Operacionalización de variables

Variable independiente: Seguridad al servidor web

Tabla 15. Operacionalización de la variable independiente

	Variable	Definición	Dimensión	Indicador	Técnica	Instrumento
Variable independiente	Seguridad al servidor web	La seguridad al servidor web procede de la espera de peticiones y respuesta de contenido que el cliente solicita desde el desarrollo y la creación de diversos protocolos que se encargan de transferir datos incrustados, de tal manera que no pueden ser vulnerados, es así que muchos de los hackers tienden a invadir dichos servidores con métodos y técnicas de vulnerabilidad dejando sin protección al servidor (Saura, 2016).	Pentest	- Planificación y reconocimiento -Escaneo - Obtener acceso - Mantener el acceso - Análisis	Documentación	Ficha técnica Cuadro comparativo
			Accesibilidad a la información	- Nivel de amenazas - Nivel de seguridad - Nivel de ataques	Documentación	Ficha técnica Cuadro comparativo
			Servidores	-Número de información - Compatibilidad con las demás herramientas de seguridad - Uso de todos los recursos	Documentación	Ficha técnica Cuadro comparativo
			Ataques	- Número de entradas que se pueden se vulnerables a un análisis de paquetes - Nivel de vulnerabilidad destacada por descubrimiento de contraseñas, robo de información, etc.	Documentación	Ficha técnica Cuadro comparativo

Fuente: Autoría propia
La tabla muestra la variable independiente con sus respectivas definiciones, dimensiones, indicadores, técnica e instrumento

Variable dependiente: Vulnerabilidad al sistema

Tabla 16. Operacionalización de la variable dependiente

Variable dependiente	Vulnerabilidad al sistema	Es la impotencia de un sistema que deja a un atacante vulnerar la confidencialidad, integridad, disponibilidad, es decir a la seguridad del equipo a sus datos y aplicaciones (Romero et al., 2018).	Planificación	- Porcentaje de rendimiento de herramientas de seguridad -Porcentaje de utilización de las herramientas	Encuesta (prueba de vulnerabilidad al sistema)	Test Observacional
			Organización	- Número de procesos - Nivel de complejidad de procesos	Encuesta	Cuestionario
			Pilares de la seguridad	- Porcentaje de disponibilidad - Porcentaje de integridad - Porcentaje de confidencialidad	Encuesta	Cuestionario
			Manejo de la seguridad	- Numero de control de niveles de seguridad y consistencia del sistema - Nivel de cumplimiento - Metodología y estándares de seguridad	Encuesta	Test Observacional

Fuente: Autoría propia
La tabla muestra la variable dependiente con sus respectivas definiciones, dimensiones, indicadores, técnica e instrumento

3.4. MÉTODOS UTILIZADOS

3.4.1. Método deductivo

El método deductivo según Murillo (2020) permite obtener conclusiones personales a través de análisis de enunciados, definiendo así como un proceso racional que parte de lo general a lo específico, de esta manera las condiciones aplicadas a un grupo de individuos son aplicables a uno solo.

Este método permitió determinar el uso de pruebas de penetración en la investigación a través de un análisis macro partiendo de como el nivel de seguridad al servidor web tiende a ser vulnerable, además de la utilización de componentes de seguridad y un análisis micro con la utilización de herramientas, métodos específicos para la seguridad en el cuidado del servidor. Fue necesario comenzar por realizar comparativas entre las herramientas de pentesting a utilizar en los distintos servidores, y escoger los más competentes para la investigación.

3.4.2. Método analítico

El método analítico se descompone un todo en partes, para la observar su naturaleza en la que se desenvuelve y los fenómenos con su efectos. Este método puede declarar y comprender el fenómeno de estudio, formando nuevas teorías (Lopera et al., 2010).

La utilización de este método da lugar a describir los indicadores definidos en la operacionalización de variables y a través de estos se pudo identificar escalas de cumplimientos y riesgos que presentaron los servidores para la verificación de su mejora en un antes y después del proceso.

3.4.3. Método de Investigación Acción

El autor (Vargas, 2007) conceptualiza este método como una forma de desarrollar conocimiento teórico, el cual tiene como propósito la construcción de soluciones para la modificación de la realidad.

Este método permitió proponer el desarrollo de una solución tecnológica, creando una respuesta a la problemática planteada y de esta forma apoyando a la transformación de la realidad actual en el laboratorio de ciberseguridad.

3.5. TÉCNICAS E INSTRUMENTOS

3.5.1. Entrevista Semiestructurada

Las entrevistas semiestructuradas es una técnica cualitativa donde se basan en una conversación entre el entrevistador y el entrevistado, el entrevistador tiene la autonomía de introducir preguntas adicionales para aclarar conceptos u obtener más información, de esta manera el investigador puede agregar preguntas extras que no estuvieron analizadas (Hernández et al., 2014).

En la investigación se aplicó al encargado del área del laboratorio de ciberseguridad para obtener información correspondiente al manejo y la subutilización de recursos correspondientes al servidor web y se amplió la sesión de preguntas para conocer más a fondo las actividades que se desarrollan en el laboratorio.

3.5.2. Observación no estructurada

Esta técnica permitió hacer uso de un cuaderno, y una cámara fotográfica para evidenciar diferentes configuraciones en laboratorio y la obtención de varios documentos que fueron facilitados por el encargado del laboratorio de ciberseguridad, que fueron de gran utilidad para analizar detalladamente como se ejecutan cada uno de las configuración al servidor web, la realización de las herramientas de vulnerabilidad (pentest) y parámetros que son requeridos por parte del laboratorio de ciberseguridad.

3.6. RECURSOS

3.6.1. Humanos

Tabla 17. Recursos Humanos

Nombre	Función de Desempeña
Msc. Jairo Hidalgo	Tutor y responsable del área del laboratorio de ciberseguridad
Álvaro Castillo	Investigador

Fuente: Autoría propia
Personas que intervienen en el plan de titulación

3.6.2. Materiales

Tabla 18. Materiales usados

Recursos	Características
Hojas	Papel Bond, A4
Ordenadores	Máquinas virtuales, Servidor

Fuente: Autoría propia
Materiales usados en el plan de titulación

3.6.3. Tecnológicos

Tabla 19. Recursos tecnológicos

Recurso	Características
Computador portátil	Se utilizó para encontrar información importante para detectar el análisis y los riegos a un servidor web
Impresora	Para imprimir encuesta, informes y documentación relacionada al proyecto.
Celular	Fue indispensable para establecer horarios de tutorías y comprobar el funcionamiento de los servidores.
Internet	El uso del internet fue necesario para realizar consultas y recopilar información con el fin de tener un amplio criterio y fundamentación del proyecto accediendo a través del navegador.

Software	Sistema Operativo Windows 10, Máquinas virtuales, herramientas para las pruebas de penetración.
Servidor Web	Fue importante la instalación y configuración de los servidores tales como: Microsoft Azure, Apache y Microsoft IIS para las respectivas pruebas
Sistemas Operativos	Kali-Linux, Centos7, Linux, Microsoft IIS

Fuente: Autoría propia
Recursos tecnológicos utilizados en el desarrollo del plan de titulación

3.6.4. Recursos Económicos

Tabla 20. Recursos Económicos

Recursos	**Cantidad**	**Precio Unitario**	**Total**
Internet mensual	$ 30 x 14 meses	$30	$420
Esferos	$ 0,35 x 14 meses	$0,35	$4,90
Resma de papel bond	$ 0,25 x 4 unidades	$0,25	$3,50
Laptop Asus	$ 800 x 1 unidad	$800	$800
Empastado final	$ 15 x 1	$15	$15
Total			**$1.243,4**
Costo de improviso 5%			$150
Total del proyecto			**1.393,4**

Fuente: Autoría propia
Materiales usados en el plan de titulación

IV. RESULTADOS Y DISCUSIÓN

4.1. PROPUESTA

La propuesta se elaboró comenzando del análisis de los resultados de la investigación y de un primer acercamiento con el encargado de administrar el laboratorio de ciberseguridad, el cual se coordinó para el desarrollo de tres servidores web alojados en el laboratorio con el fin de identificar como se encuentra el nivel de seguridad, creados a partir de una configuración e instalación de los mismos, utilizando programas para su respectivos análisis y metodologías, en la que se elegio la metodología Owasp. Como segundo punto se manejó algunas de las herramientas de penetración (Pentest) que forman parte de Kali Linux que permitieron determinar la vulnerabilidad que se encontraban en los servidores, como también la ejecución de componentes de ataques como fuerza bruta y denegación de servicios, partiendo de ello se efectuó las fases correspondientes hasta culminar con la identificación de las vulnerabilidades y la mejora de la seguridad mediante el método Hardening a través de la comprobación de los indicadores de evaluación de riesgos, que dan como resultado que la ejecución de los objetivos haya sido correctamente aplicadas.

4.1.1. Alcance de la propuesta

Está orientado a las fases y procesos de seguridad, vulnerabilidad de riegos a servidores web. El aporte que se brindara en el laboratorio de ciberseguridad y lo que esperan del proyecto, es un proceso de análisis de vulnerabilidad utilizando herramientas que ayuden a detectar anomalías que se presentan en los servidores web y de igual alternativas para protegerlas, aprovechando los recursos existentes para las pruebas de penetración. Los procesos que le permitirá abordar todos los indicadores de riegos y conocer cuál es la realidad de su manejo en tiempo real.

Lo que se pretende mitigar, que la serie de estos procesos sean capaces de cumplir cabalmente los requerimientos establecidos a la hora de aplicar esta práctica de pentest.

4.1.2. Estudio de Factibilidad

- **Título:** “Pruebas de penetración para la seguridad informática al servidor web del laboratorio de ciberseguridad en la Universidad Politécnica Estatal del Carchi”
- **Institución Ejecutora:** Universidad Politécnica Estatal de Carchi. Facultad de Industrias Agropecuarias y Ciencias Ambientales.

- **Beneficiarios:** Laboratorio de ciberseguridad
- **Ubicación:** Provincia del Carchi, Cantón Tulcán.
- **Equipo técnico responsable:** El señor responsable de la investigación, Alvaro Castillo egresado de la Universidad Politécnica Estatal del Carchi.
- **Tiempo estimado para la ejecución:** 12 meses

4.1.3. Metodología OWASP

En este sentido se documenta y se evidencian los procesos identificados permitiendo tener un control más ordenado de resultados utilizando herramientas para pruebas técnicas tanto para la aplicación web como para los servidores web.

4.1.3.1 Recolección de Información

En esta fase se busca recolectar la cantidad total de información sobre el laboratorio de ciberseguridad y de los servidores web que fueron configurados como objeto del Pentest ya sean (archivos, IP, nombres, entre otros), todo esto sin dejar evidencias de IP del equipo atacante en los procesos hacia el objetivo, de igual forma se hará uso de técnicas como **Whois, Reverse IP**, **Nmap**, **OWASP ZAP. WebServer Stress Tools** y **Maltego** y utilización de los comandos como: **comand ab** que serán de utilidad para evaluar el rendimiento, carga y el estrés de los servidores.

Con la finalidad de tener un mayor entendimiento de las direcciones IP de cada uno de los servidores web que a continuación serán estudiadas. Se establecerá un nombre que determina el servidor del cual se está analizando.

Servidor Linux/Apache de Microsoft Azure

- **Serv1:** 191.237.251.161

Servidor Linux/Apache del laboratorio de ciberseguridad

- **Serv2:** 172.20.24.53

Servidor Windows Server 2017/Microsoft IIS del laboratorio de ciberseguridad

- **Serv3:** 172.20.24.12

4.1.3.1.1 Uso de motores de búsqueda para verificar la presencia de información vulnerable.

Se realiza una búsqueda al entorno de los servidores en los que se está trabajando como su sistema operativo, distribución, servidor, ip, bases de datos y memoria.

Tabla 21. Recolección de datos informativos

Sistema Operativo	Linux	Linux/Plataforma Azure	Microsoft IIS
Distribución	Centos Linux release 7.9.2009 (Core)	Centos Linux release 8.4 (Core)	Windows Server 2016 Standard Evolution 1607
Servidor	Apache 2.4.6 (Centos)	Apache 2.4.37 (Centos)	Windows Server 2016 IIS 10.0
IP	172.20.24.53	191.237.251.161	172.20.24.30
Php versión	7.2.24	7.2.24	7.1.5
Base de datos	5.6.51 MariaDB	10.3.28 MariaDB	Mysql Community 8.0
Memoria	150.00 GB	150.00 GB	200.0 GB

Mediante un checklist se verificará los cumplimientos y riegos que se encuentran presentes en el laboratorio de ciberseguridad, de la misma manera a los servidores configurados, con el fin de evaluar los aspectos que conforman cada uno de ellos.

Tabla 22. Checklist de verificación de seguridad informática

SEGURIDAD DE LOS DATOS						
		Cumple		Riesgo		
Indicador	Aspectos a evaluar	SI	NO	Bajo	Medio	Alto
1	Los servidores web tienen definidas políticas de seguridad		X			X
2	Estas políticas de seguridad son revisadas	X			X	

	periódicamente en los servidores web					
3	Dispone de alguna metodología de seguridad a los servidores web		X			X
4	Se monitoriza diariamente el estado de cada uno de los servidores		X		X	
5	Se tiene implementando dominios a los servidores web		X			X
6	Se tiene instalado antivirus licenciado en los servidores web		X		X	
7	Realiza pruebas para detectar vulnerabilidades en los servidores		X			X

SEGURIDAD DE LA INFRAESTRUCTURA A LOS SERVIDORES WEB

		Cumple		Riesgo		
Indicador	Aspectos a evaluar	SI	NO	Bajo	Medio	Alto
8	Dispone de firewall		X			X
9	Dispone de un sistema de protección anti-DDOS	X			X	
10	Dispone de comandos que ayuden al mejoramiento de la seguridad a los servidores web		X		X	

11	Dispone de certificación SSL		X	X		

CONTROLES DE ACCESO Y SEGURIDAD A LOS SERVIDORES WEB

		Cumple		Riesgo		
Indicador	Aspectos a evaluar	SI	NO	Bajo	Medio	Alto
12	Dispone de contraseñas seguras al momento de acceder a los paneles del administrador de los servidores web	X		X		
13	Dispone de proxy que mitiguen riegos de vulnerabilidad al servidor web		X		X	

HABITOS SEGUROS Y PREPARACIÓN A LOS SERVIDORES WEB

		Cumple		Riesgo		
Indicador	Aspectos a evaluar	SI	NO	Bajo	Medio	Alto
14	Desarrolla sistemas de capacitación al personal referente a la seguridad informática		X		X	
15	La actitud referente a la cuidado de normas de seguridad es positiva	X			X	

Fuente: Gordón, D y Pacheco, R. 2018. Análisis de estrategias de gestión de seguridad informática con base en la Metodología Open Source Security (artículo científico). Universidad de Especialidades Espíritu Santo.

Tabla 23. Escala de cumplimiento y riesgos

	Riego		
	Alto	Medio	Bajo
No cumplen: 11	5	5	1
Si cumplen: 4	-	3	1

Análisis.

De acuerdo con la información recopilada (tabla 23), los servidores web como objetos de investigación aplica políticas y procesos básicos de seguridad. Los datos obtenidos en esta tabla nos indica que el 73,33% de los indicadores no son cumplidos cabalmente y presentan un riesgo alto del 33,33%, medio del 33,33% y bajo del 6,66% que pueden ser elementos de vulnerabilidad. Para los indicadores que si cumplen con un 26,66% a los procesos de seguridad, el 20% corresponde a un riego medio y el 6,66% a un bajo de tal manera que la detección de las amenazas son consideradas leves. En este sentido existe una similitud de consecuencias con un estudio elaborado por ESET (2017) donde manifiesta que el 74% de las organizaciones en Latinoamérica, agregando a Ecuador, han ejecutado políticas de seguridad como antivirus, firewall, sistemas y controles de acceso, entre otros. Sin embargo se hace necesario mejorar y gestionar los procesos de seguridad informática al laboratorio de ciberseguridad.

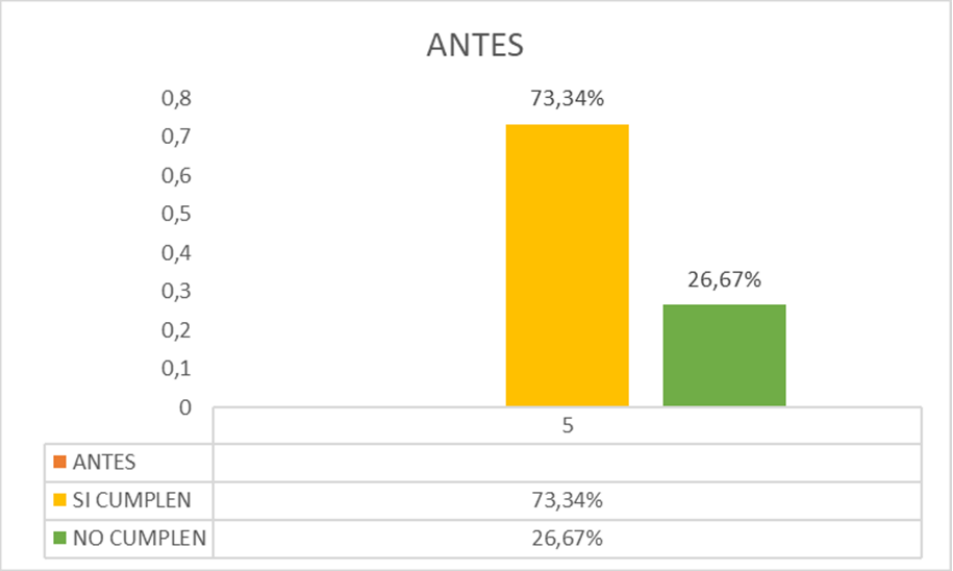
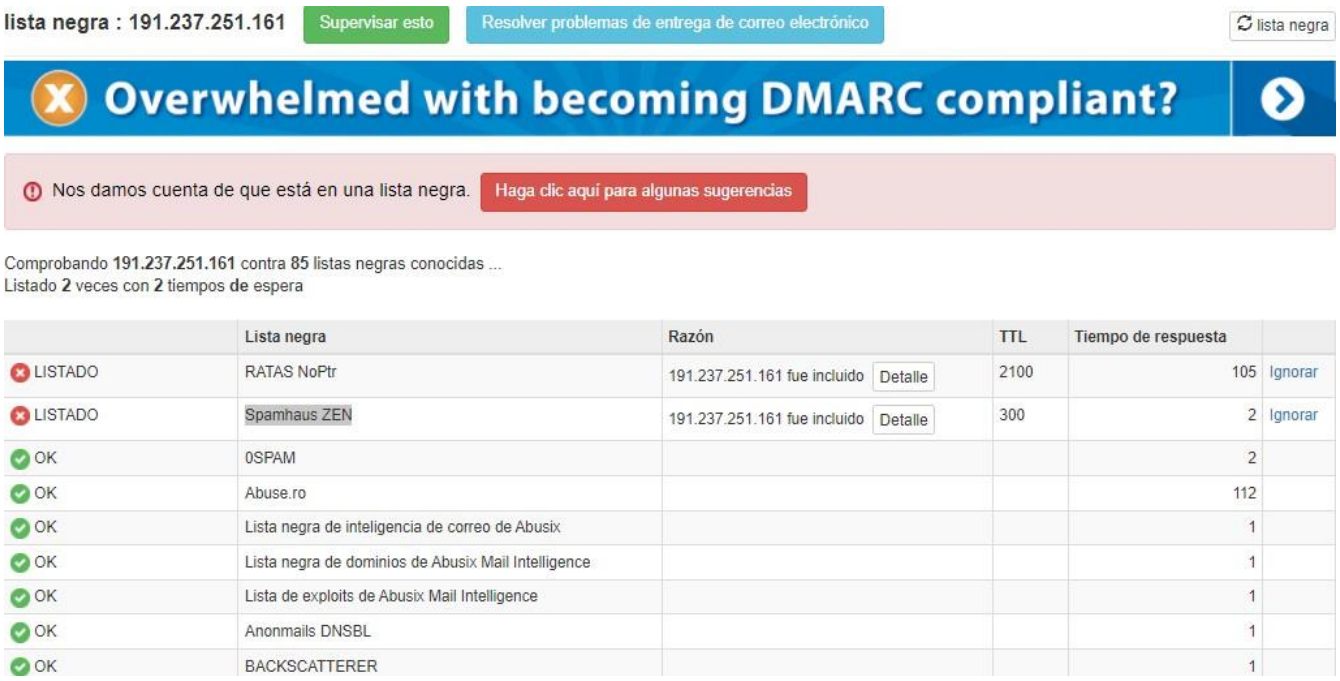


Figura 14. Resultado del cumplimiento de riesgos

- Mediante **Reserve Ip** verificaremos si se encuentra o no en una lista negra, es decir un blacklist es una lista en donde analiza IP que presentan anomalías generadas de forma voluntarias o involuntarias, dejando de funcionar servicios. https://mxtoolbox.com/blacklists.aspx



	Lista negra	Razón	TTL	Tiempo de respuesta	
LISTADO	RATAS NoPtr	191.237.251.161 fue incluido Detalle	2100	105	Ignorar
LISTADO	Spamhaus ZEN	191.237.251.161 fue incluido Detalle	300	2	Ignorar
OK	0SPAM			2	
OK	Abuse.ro			112	
OK	Lista negra de inteligencia de correo de Abusix			1	
OK	Lista negra de dominios de Abusix Mail Intelligence			1	
OK	Lista de exploits de Abusix Mail Intelligence			1	
OK	Anonmails DNSBL			1	
OK	BACKSCATTERER			1	

Figura 15. Lista negra: servidor 1

OK	BARRACUDA			15	
OK	BLOCKLIST.DE			1	
OK	CALIVENT			2	
OK	CBL			9	
OK	CYMRU BOGONS			1	
OK	DAN TOR			248	
OK	DAN TOREXIT			248	
OK	SERVICIOS DNS			1	
OK	DRMX			1	
OK	DRONE BL			15	
OK	FABELSOURCES			8	
OK	HIL			3	
OK	HIL2			2	
OK	Hostkarma Negro			1	
OK	Lista negra de DNS de IBM			2	
OK	ICMFORBIDDEN			112	
OK	SPAM IMP			1	
OK	GUSANO IMP			2	
OK	ivmSIP			1	
OK	ivmSIP24			1	

Figura 16. Lista negra: servidor 1

La dirección ip 191.237.151.161 la agrega en la lista negra, debido a que presenta dos anomalías (figura 17).

Comprobando **191.237.251.161** contra **85** listas negras conocidas ...
Listado **2** veces con **2** tiempos **de** espera

	Lista negra	Razón	TTL	Tiempo de respuesta	
LISTADO	RATAS NoPtr	191.237.251.161 fue incluido Detalle	2100	105	Ignorar
LISTADO	Spamhaus ZEN	191.237.251.161 fue incluido Detalle	300	2	Ignorar

Figura 17. Anomalías al servidor 1

Estas dos anomalías son:

RATAS NoPtr: son aquellas direcciones que se encuentran enviando cantidad excesiva de conexiones a usuarios inválidos, como también el hecho de no contar con un DNS inversa, es decir un nombre de domino o de un host determinado para la dirección Ip. En este sentido es marcado en rojo por la razón de contar con un dominio provocando la entrada de troyanos, gusano o bots. Cabe mencionar que las direcciones tipo C son más propensas de formar parte de la lista de SpamRats, ya que estos rangos normalmente son utilizado para ataques contra spam u otras formas de ataques.

Spamhaus ZEN: Este inconveniente está marcado por la dirección ip no cuenta con políticas o autorización para enviar correos electrónicos SMTP (Protocolo de comunicación para envió de correos electrónicos) directamente.

- Mediante **Maltego** se realizará la exploración de la información tanto de infraestructura como de individuos, independientemente de los datos que sean recogidos en el software, arrojará información como direcciones de email, teléfonos, sitios web entre otros.

Como se observa (figura 18) a través de una transformación al servidor se logró evidenciar información importante, entre los cuales se encontró dominios, direcciones de servidores, direcciones de correos que fueron utilizados a la hora de configurar el servidor, el gestor de contenido donde fue configurado, la Ipv4 asignado por la plataforma de Microsoft Azure y el nombre del sitio como fue creado. A demás de presentar el puerto en el que se encuentra abierto nuestro sitio web.

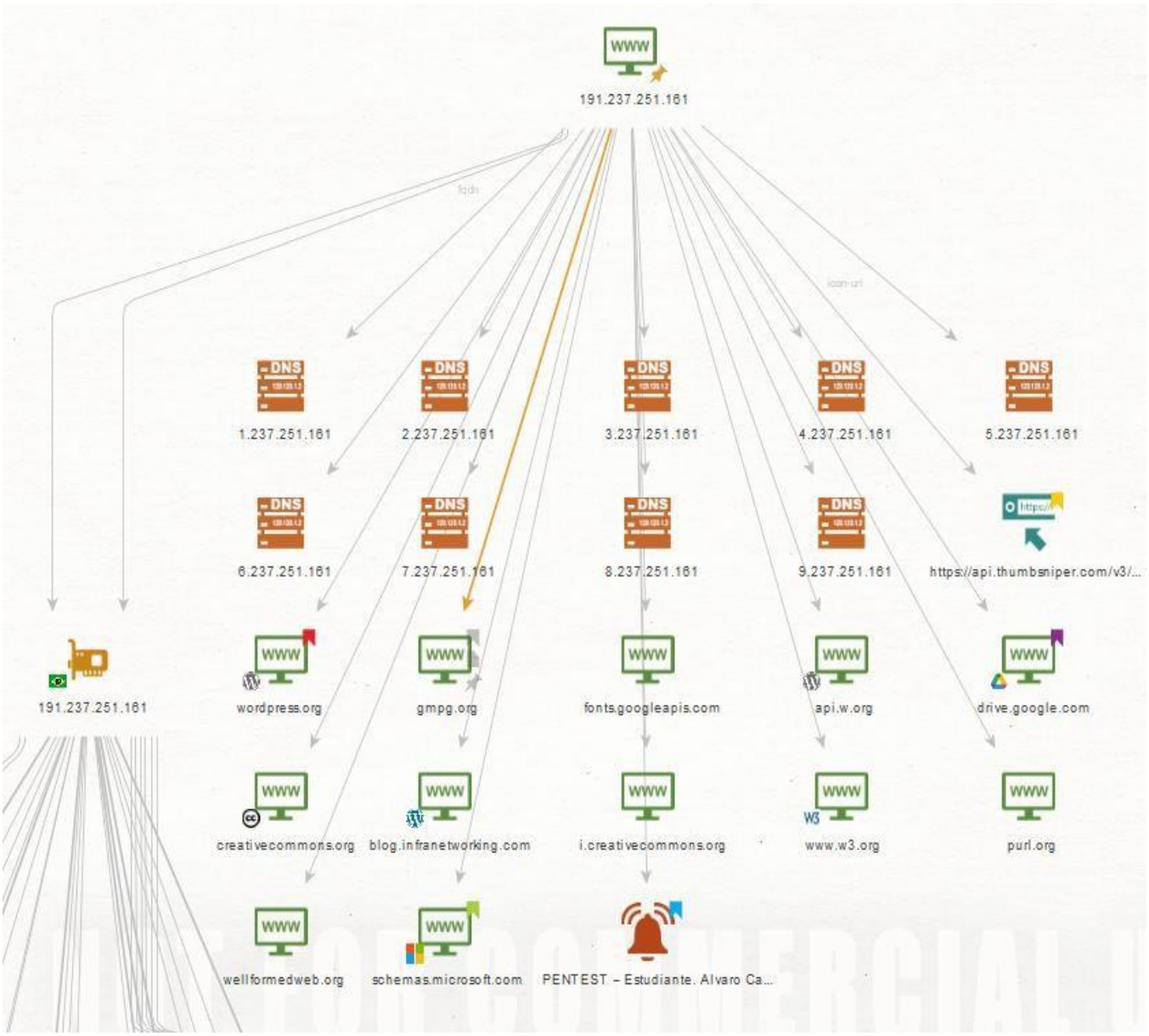


Figura 18. Recopilación de información Maltego servidor 1

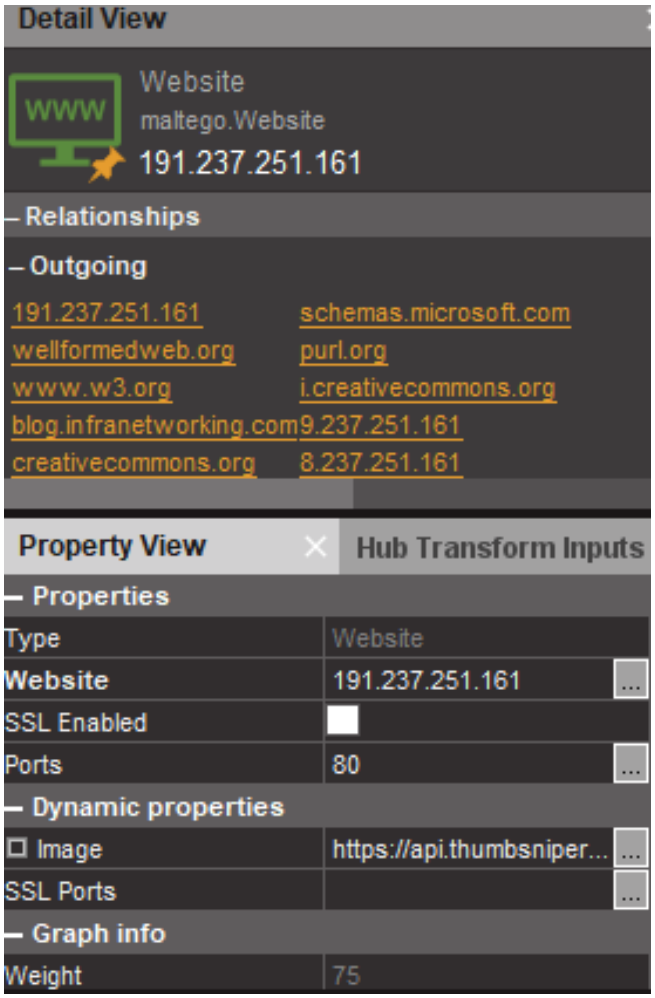


Figura 19. Puerto 80 abierto servidor 1

Una vez realizada la búsqueda, **Maltego** nos permite encontrar información más específica, de esta manera hubo la posibilidad de navegar desde la dirección original proporcionada, proveniente de Brasil, como también alguno de sus correos, números de teléfonos, ubicaciones y nombres de usuarios.

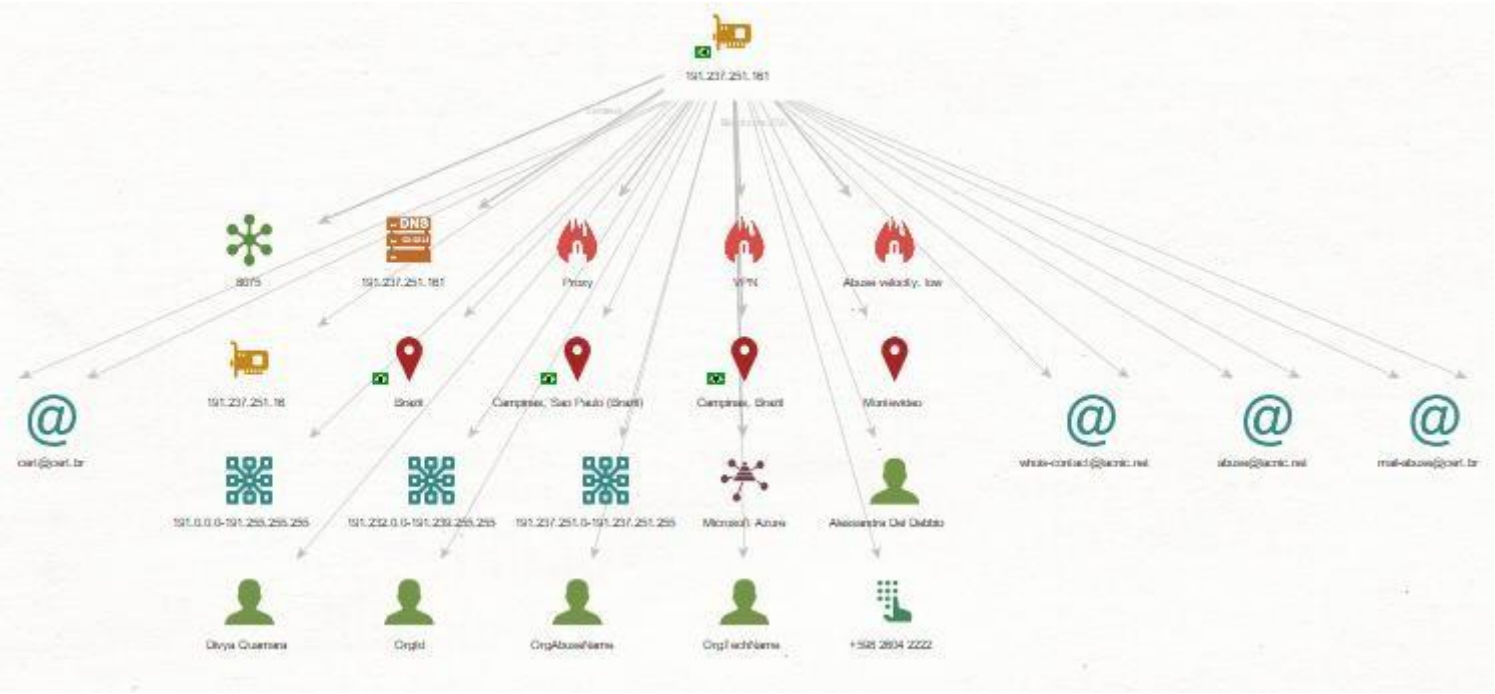


Figura 20. Recolección de información Maltego servidor 1

Se recolecta información de la ubicación, usuario y número de teléfono, en este caso el código indica que está registrado en Uruguay (figura 21).

Figura 21. Recolección de información Maltego servidor 1

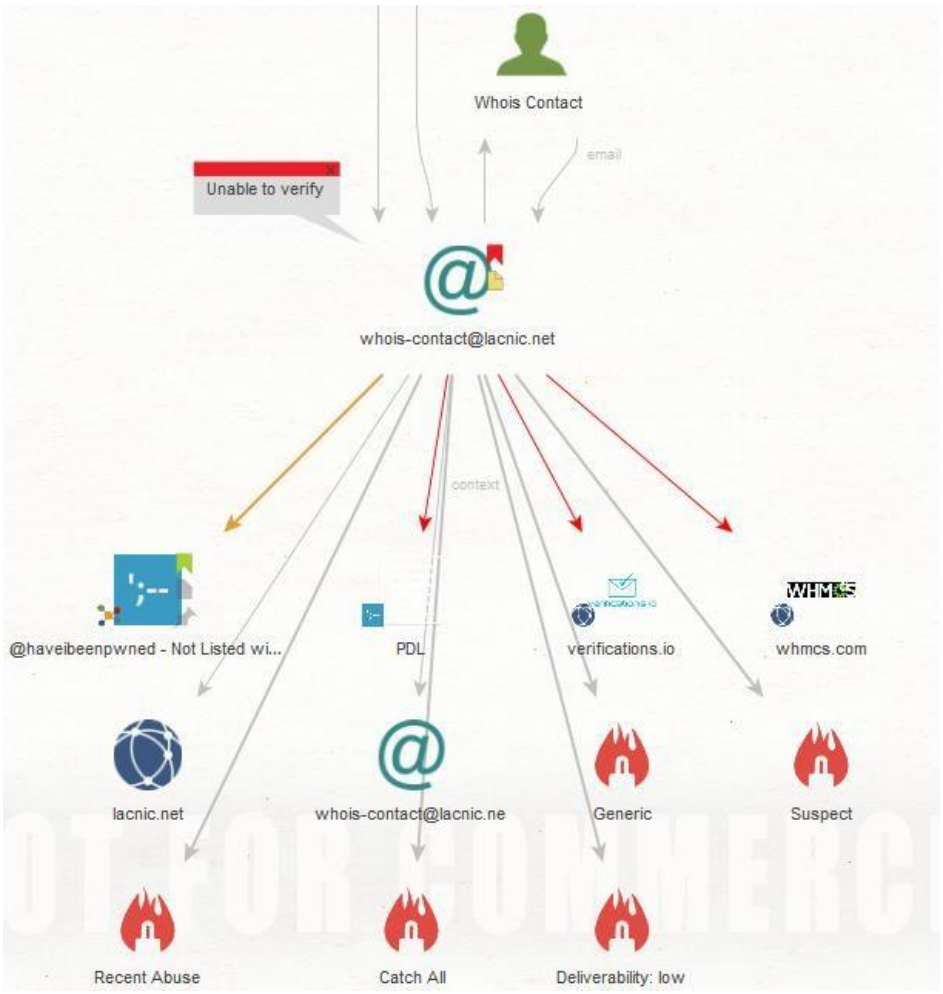


Figura 22. Información de correo electrónico servidor 1

Medición del rendimiento de nuestro servidor utilizando **comando ab**, elemento que sirve para indicar la respuesta que tiene el servidor.

ab -c 100 -c 5 -k http//191.237.251.161/

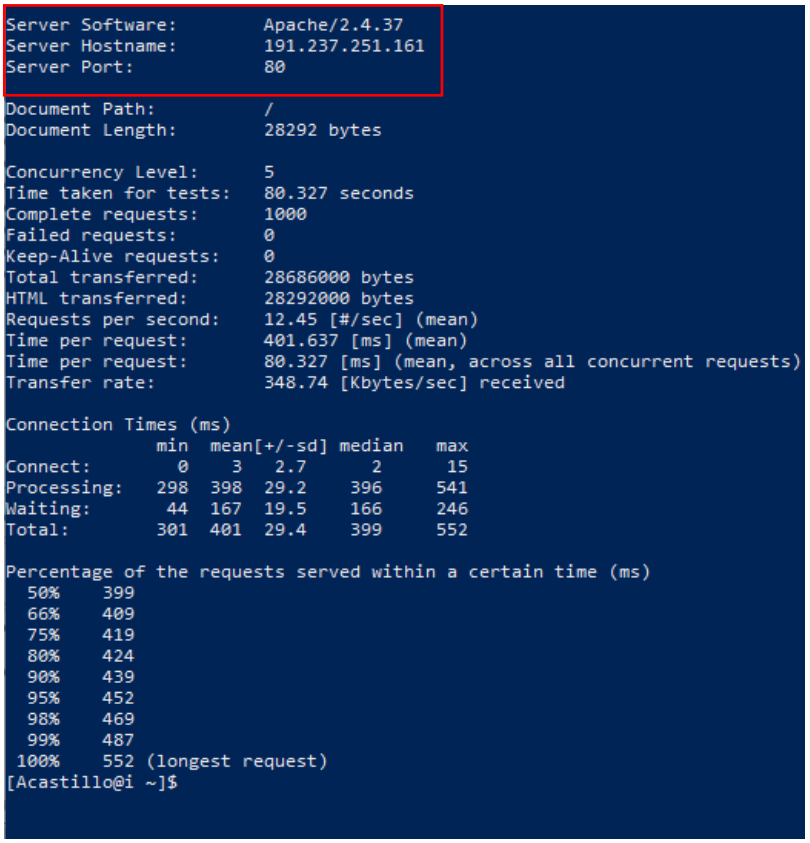


Figura 23. Respuesta de ejecución del comando ab 12#/seg

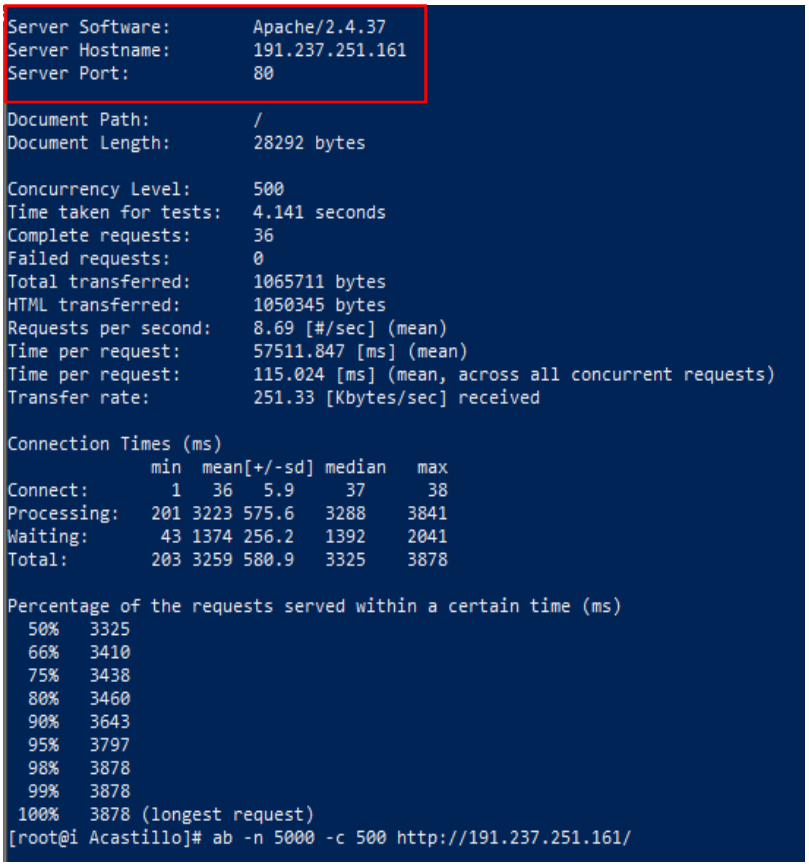


Figura 24. Respuesta de ejecución del comando ab 8#/seg

ab -n 1000 -c 5 -k http://172.20.24.53/

```
Server Software:        Apache/2.4.6
Server Hostname:        172.20.24.53
Server Port:            80

Document Path:          /
Document Length:        94489 bytes

Concurrency Level:      5
Time taken for tests:   164.781 seconds
Complete requests:      1000
Failed requests:        0
Write errors:           0
Keep-Alive requests:    0
Total transferred:      94915000 bytes
HTML transferred:       94489000 bytes
Requests per second:    6.07 [#/sec] (mean)
Time per request:       823.906 [ms] (mean)
Time per request:       164.781 [ms] (mean, across all concurrent requests)
Transfer rate:          562.51 [Kbytes/sec] received

Connection Times (ms)
              min  mean[+/-sd] median   max
Connect:        0    0   0.0      0       0
Processing:   604  815 202.1    777    2532
Waiting:      602  813 202.0    776    2530
Total:        604  815 202.1    777    2532

Percentage of the requests served within a certain time (ms)
  50%    777
  66%    840
  75%    876
  80%    904
  90%   1000
  95%   1072
  98%   1389
```

Figura 25. Respuesta de ejecución del comando ab 6#/seg

ab -n 1000 -c 5 -k http://172.20.24.12/

```
Server Software:        Microsoft-IIS/10.0
Server Hostname:        172.20.24.12
Server Port:            80

Document Path:          /
Document Length:        0 bytes

Concurrency Level:      5
Time taken for tests:   107.676 seconds
Complete requests:      1000
Failed requests:        0
Write errors:           0
Non-2xx responses:      1000
Keep-Alive requests:    1000
Total transferred:      263000 bytes
HTML transferred:       0 bytes
Requests per second:    9.29 [#/sec] (mean)
Time per request:       538.379 [ms] (mean)
Time per request:       107.676 [ms] (mean, across all concurrent requests)
Transfer rate:          2.39 [Kbytes/sec] received

Connection Times (ms)
              min  mean[+/-sd] median   max
Connect:        0    0   0.0      0       1
Processing:   410  532  59.2    523    1391
Waiting:      410  532  59.2    523    1391
Total:        410  532  59.2    523    1392

Percentage of the requests served within a certain time (ms)
  50%    523
  66%    545
  75%    559
  80%    569
  90%    599
  95%    631
```

Figura 26. Respuesta de ejecución del comando ab 9#/seg.

Entre estos parámetros de respuesta se puede observar la variedad de pedidos que el servidor realizo, entre ellos esta los más principales.

Tabla 24. Rendimiento del servidor Apache/Linux: servidor 1

Tasa de transferencia (Transfer rate)	Respuesta por servidor (Request per second)	Tiempo que tardo en realizar el test (Time taken for test)	Porcentaje de solicitudes atendidas según el tiempo que fue tomado.	
0.0 [Kbytes/sec]			50%	399
			66%	409
Puerto 80 y 443 abierto			75%	419
			80%	424
	685.41 [#/secc] (mean)	0.146 segundos	90%	439
			95%	452
			98%	469
			99%	487
			100%	552

Tabla 25. Rendimiento del servidor Apache/Linux: servidor 2

Tasa de transferencia (Transfer rate)	Respuesta por servidor (Request per second)	Tiempo que tardo en realizar el test (Time taken for test)	Porcentaje de solicitudes atendidas según el tiempo que fue tomado.	
562.51 [Kbytes/sec]			50%	777
			66%	840
Puerto 80 abierto			75%	876
			80%	904
	6.07 [#/secc] (mean)	164.781 segundos	90%	1000
			95%	1072
			98%	1389
			99%	1896
			100%	2532

Tabla 26. Rendimiento del servidor Microsoft IIS: servidor 3

Tasa de transferencia (Transfer rate)	Respuesta por servidor (Request per second)	Tiempo que tardo en realizar el test (Time taken for test)	Porcentaje de solicitudes atendidas según el tiempo que fue tomado.	
2.39 [Kbytes/sec]			50%	523
			66%	545
Puerto 80 abierto			75%	559
			80%	569
	9.29 [#/secc] (mean)	107.676 segundos	90%	599
			95%	631
			98%	668
			99%	689
			100%	1392

4.1.3.1.2 Análisis al servidor web para verificar peticiones por tiempo y nombres por defecto.

Mediante la utilización de la herramienta **Web Server Stress** podremos observar las peticiones por tiempo, es decir el número de usuarios que pude soportar nuestro servidor, en este ejemplo probaremos con 5 usuarios que se conectar al servidor al mismo tiempo. Como resultante (figura 27).

Logfiles | Results per User (Complete Test) | Results per URL (Complete Test)

User No.	Clicks	Hits	Errors	Avg. Click Time [ms]	Bytes	kbit/s	Cookies
1	4	3	0	800	86.010	286,69	
2	4	3	0	864	86.010	265,56	
3	4	3	0	669	86.010	343,02	
4	4	3	0	667	86.010	343,98	
5	4	3	0	1.009	86.010	227,25	
6	4	3	0	675	86.010	339,75	
7	4	3	0	728	86.010	315,10	
8	4	3	0	695	86.010	329,79	
9	4	3	0	663	86.010	345,89	
10	4	3	0	889	86.010	257,88	
11	4	3	0	713	86.010	321,74	
12	3	2	0	700	57.340	327,58	
13	3	2	0	666	57.340	344,24	
14	3	2	0	843	57.340	272,07	
15	3	2	0	693	57.340	330,98	
16	3	2	0	646	57.340	355,19	
17	3	2	0	1.200	57.340	191,17	
18	3	2	0	835	57.340	274,73	
19	3	2	0	648	57.340	353,93	
20	3	2	0	661	57.340	347,09	
21	3	2	0	817	57.340	280,67	
22	3	2	0	679	57.340	337,76	
23	3	2	0	713	57.340	321,69	
24	2	2	0	636	57.340	360,64	
25	2	1	0	627	28.670	365,83	

Figura 27. Resultados por usuario servidor 2

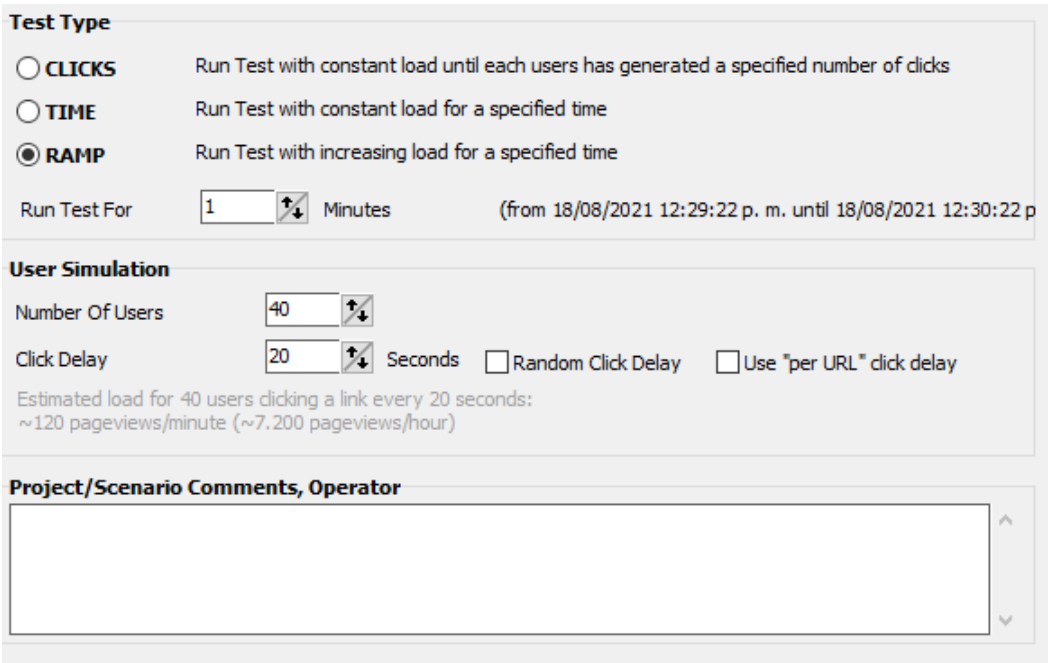


Figura 28. Configuración de tiempo para pruebas de carga servidor 2

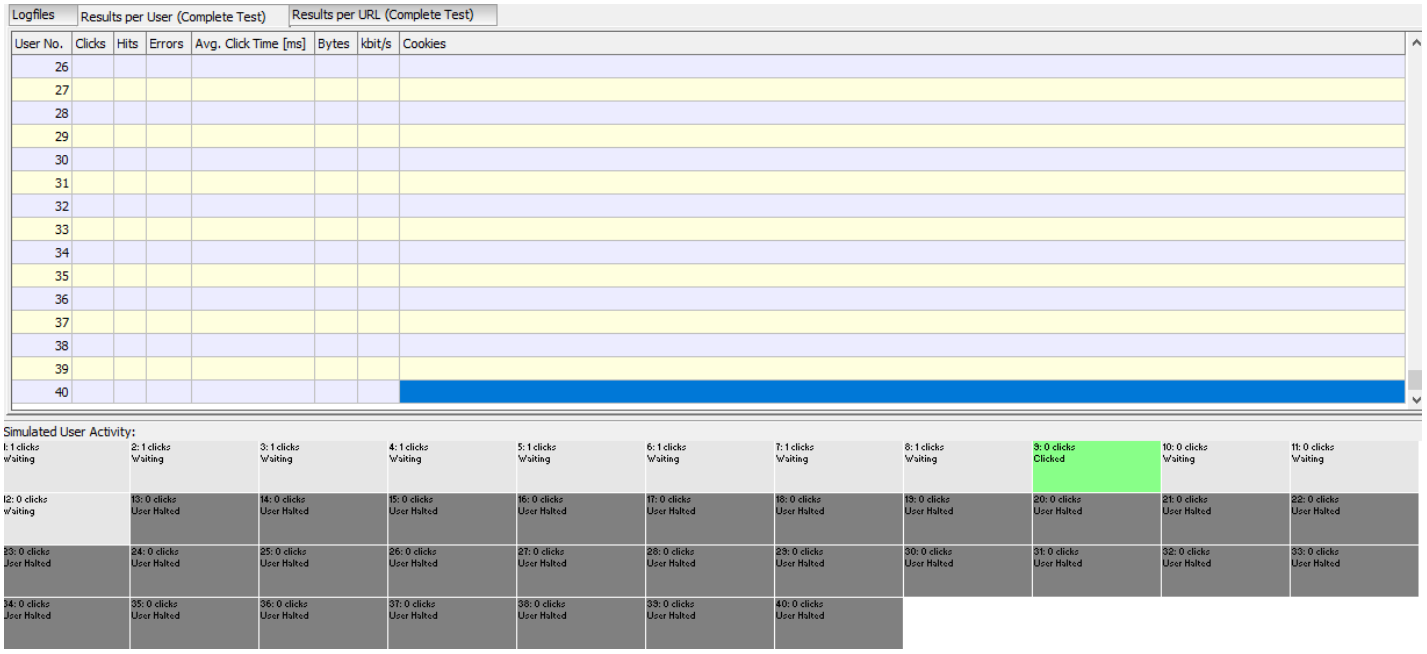


Figura 29. Valoración de 40 usuarios servidor 2

Indica que no existe error al proporcionar 40 usuarios (figura 40).

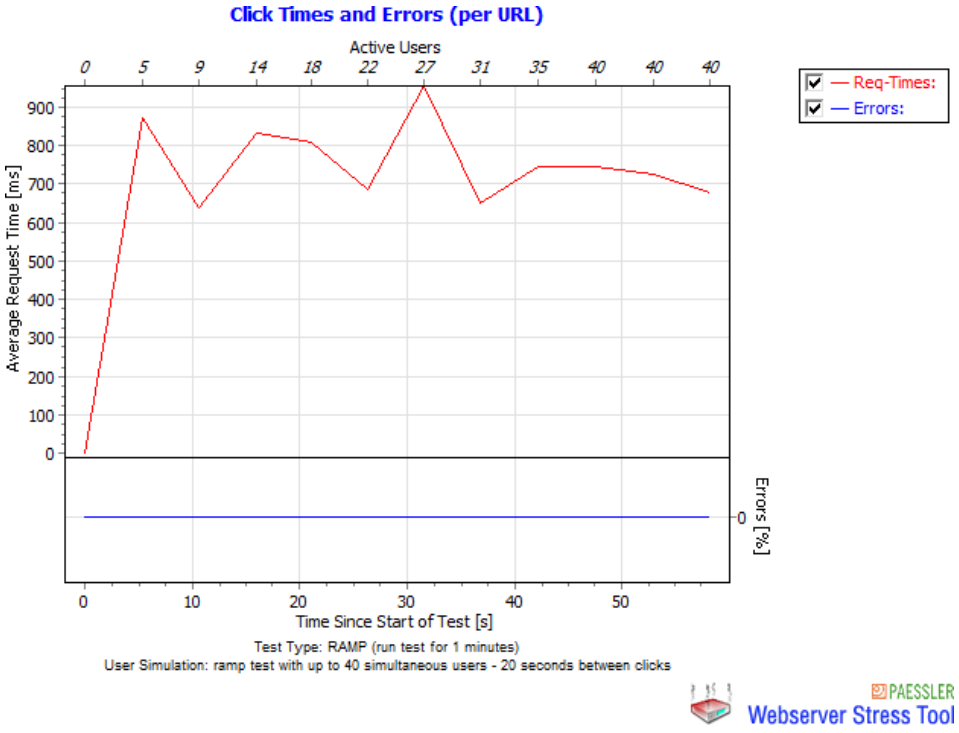


Figura 30. Resultado Gráfico de 40 usuarios servidor 2

Mientras que al ingresar 4000 usuarios, desde el usuario 2376 presenta un error de petición por tiempo, es decir un 48,2% de error.

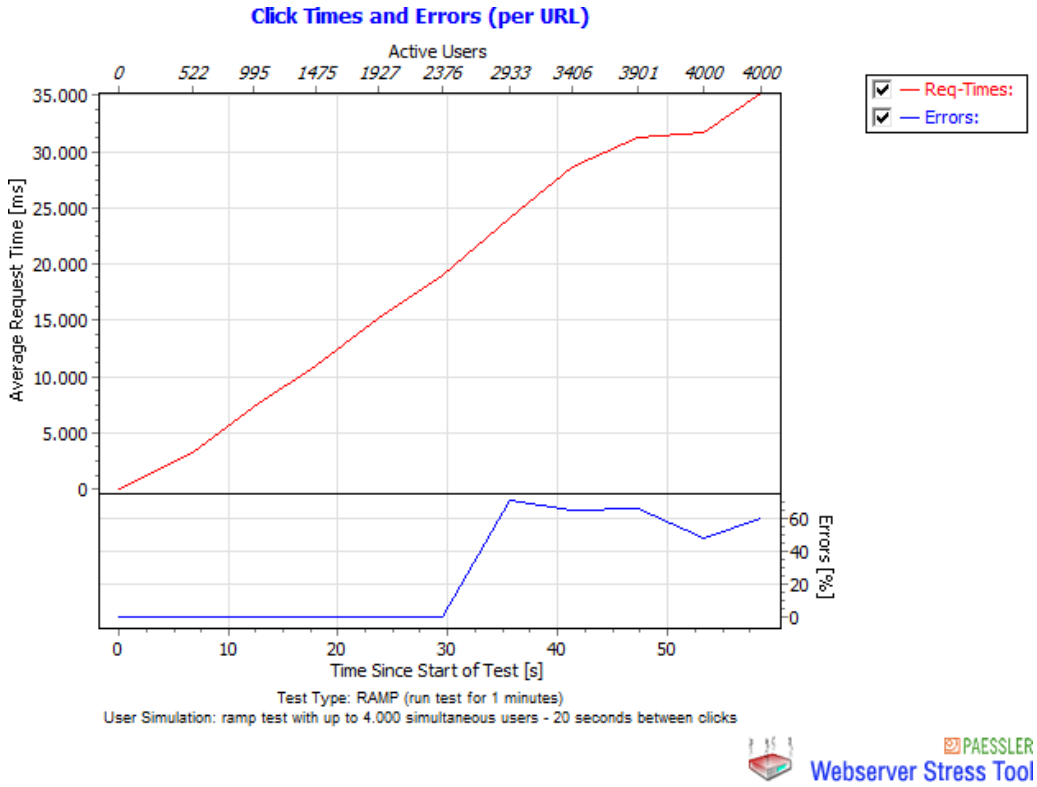


Figura 31. Resultado gráfico de 40000 usuarios servidor 1

Para el caso del servidor **Microsoft IIS**, al ingresar 500 usuarios no presentan ningún problema al momento de realizar la petición al servidor (figura 32).

Logfiles | Results per User (Complete Test) | Results per URL (Complete Test)

User No.	Clicks	Hits	Errors	Avg. Click Time [ms]	Bytes	kbit/s	Cookies
16	2	2	0	32.853	390.804	47,58	
17	2	2	0	33.210	390.804	47,07	
18	2	2	0	34.235	390.804	45,66	
19	2	2	0	35.543	390.804	43,98	
20	2	2	0	35.071	390.804	44,57	
21	2	2	0	35.708	390.804	43,78	
22	2	2	0	36.765	390.804	42,52	
23	2	2	0	37.808	390.804	41,35	
24	2	2	0	38.884	390.804	40,20	
25	2	2	0	38.098	390.804	41,03	
26	2	2	0	39.913	390.804	39,17	
27	2	2	0	41.281	390.804	37,87	
28	2	2	0	40.316	390.804	38,77	
29	2	2	0	42.754	390.804	36,56	
30	2	2	0	43.459	390.804	35,97	
31	2	2	0	42.039	390.804	37,19	
32	2	2	0	44.511	390.804	35,12	
33	2	2	0	45.607	390.804	34,28	
34	2	2	0	45.109	390.804	34,65	
35	2	2	0	45.958	390.804	34,01	
36	2	2	0	45.952	390.804	34,02	
37	2	2	0	46.487	390.804	33,63	
38	2	2	0	47.392	390.804	32,98	

Figura 32. Valoración por 500 usuarios servidor 3

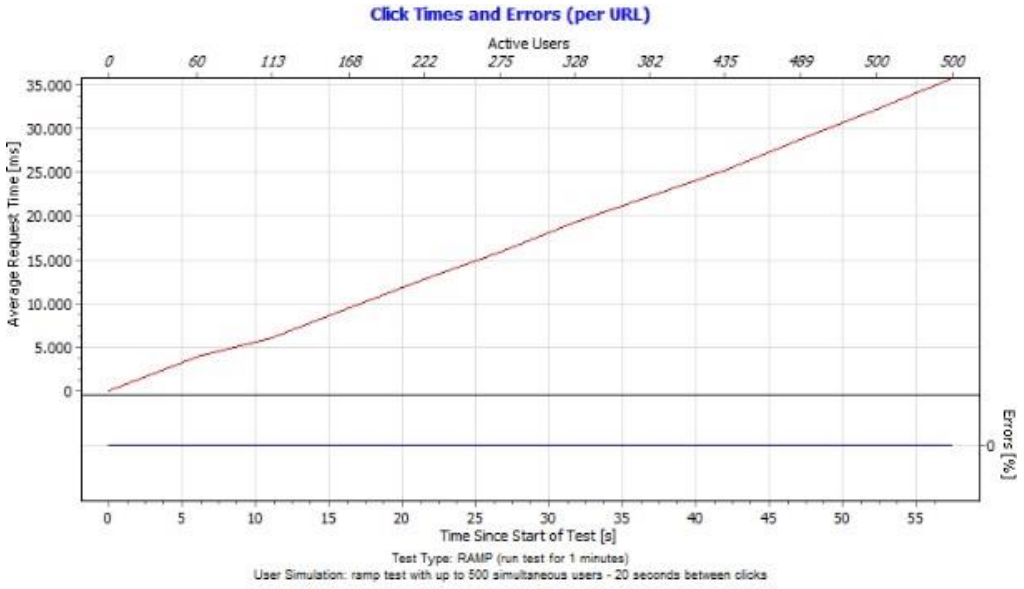


Figura 33. Resultado gráfico con 500 usuarios servidor 3

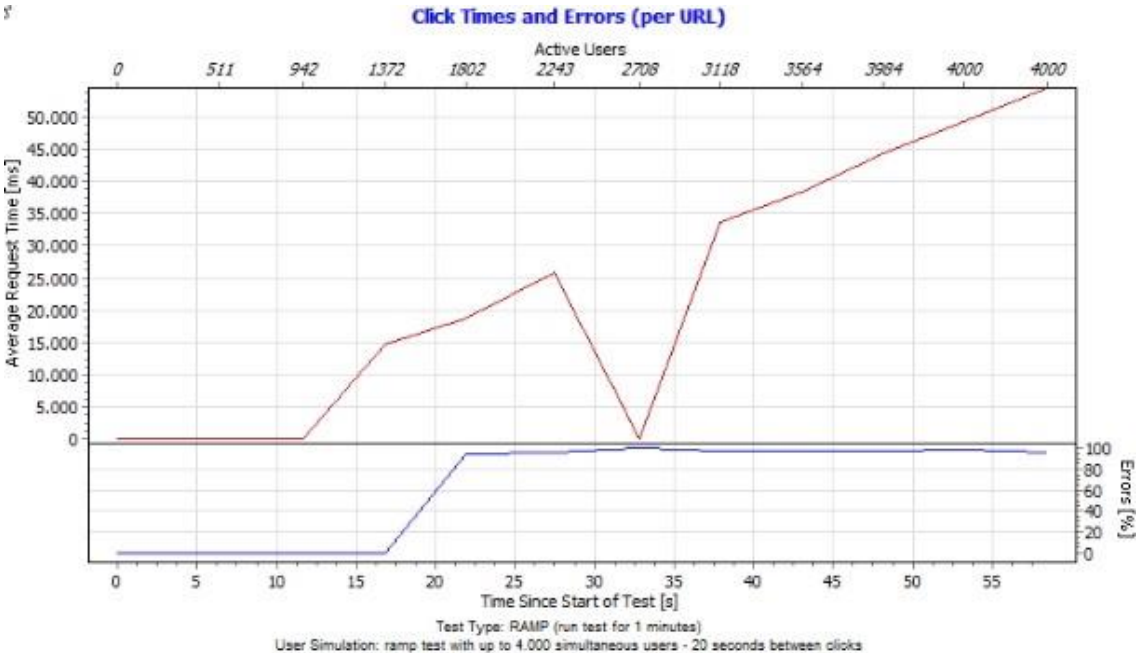


Figura 34. Resultado gráfico con 4000 usuarios servidor 3

En este sentido mientras que al ingresar 4000 usuarios de la misma manera que al otro servidor de Apache, desde el usuario 942 presenta un error de petición por tiempo de usuario, es decir un 96,56% de error.

- Mediante una búsqueda realizada por **whois** de la página oficial https://www.domaintools.com/ se obtendrá la siguiente información de cada uno de los servidores.

Información de IP para 191.237.251.161

— Estadísticas rápidas

Ubicación de IP	Brasil Campinas Microsoft Do Brasil Imp. E Com. Software E Video G
ASN	AS8075 MICROSOFT-CORP-MSN-AS-BLOCK, EE. UU. (Registrado el 31 de marzo de 1997)
Servidor Whois	whois.lacnic.net
Dirección IP	191.237.251.161

```
inetnum: 191.236.0.0/14
aut-num: AS8075
abuse-c: DIQUA12
propietario: Microsoft do Brasil Imp. e Com. Software e Video G
ownerid: 04.712.500 / 0001-07
responsable: Alessandra Del Debbio
país: BR
propietario-c: DIQUA12
tech-c: DIQUA12
inetrev: 191.237.248.0/21
nserver: ns2-201.azure-dns.net
nsstat : 20210812 AA
nslastaa: 20210812
nservidor: ns2prod.arpa-201.azuredns-prd.org
nsstat: 20210812 AA
nslastaa: 20210812
nserver: ns1-201.azure-dns.com
nsstat: 20210812 AA
nslastaa: 20210812
nserd. 201.azuredns-prd.info
nsstat: 20210812 AA
nslastaa: 20210812
creado: 20130911
modificado: 20210105

nic-hdl-br: DIQUA12
persona: Divya Quamara
correo electrónico:       país: BR creado: 20170615 modificado: 20170615
 iphostmaster@microsoft.com
```

Figura 35. Domaintools.com servidor 1

Como se puede observar el protocolo Whois ofrece información que pueden ser de mucha ayuda en la fase de recolección de información, de esta manera un hacker puede utilizar estos datos para realizar algún tipo ataque en los servidores.

- Mediante la búsqueda realizada con comando whois a la 191.237.251.161 seria así:

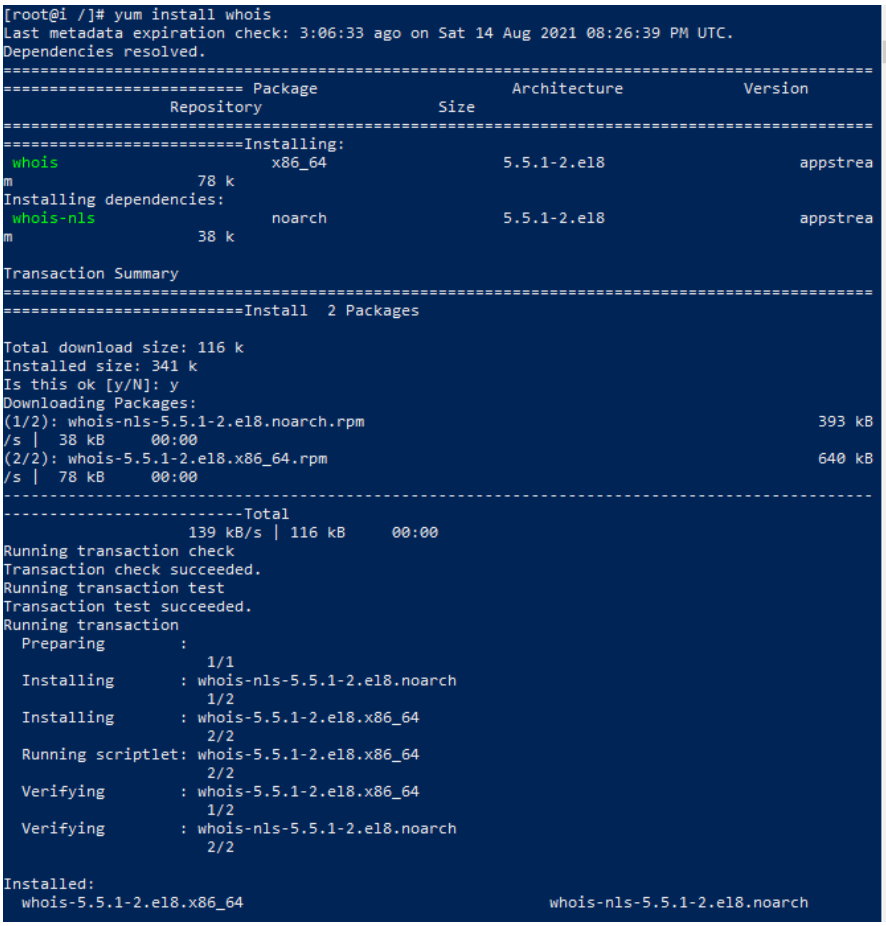


Figura 36. Instalación de Whois servidor 1

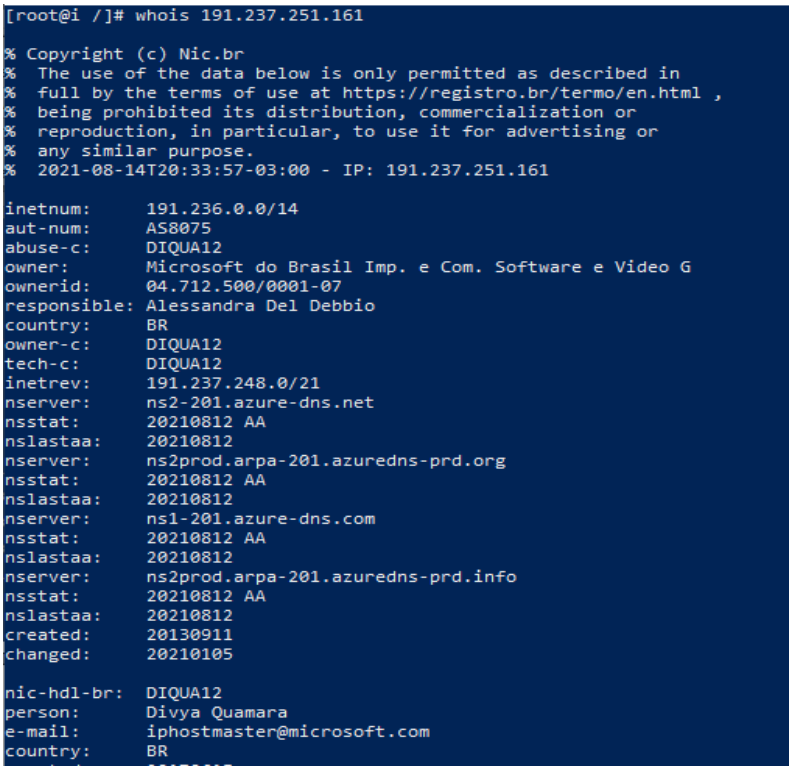


Figura 37. Información sobre el registro: servidor 1

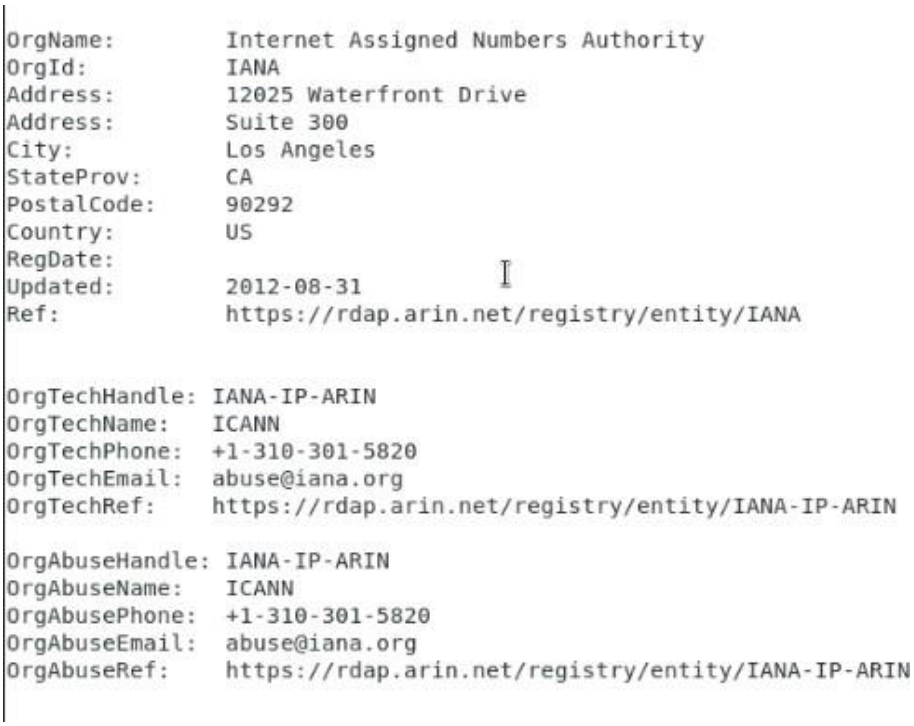


Figura 38. Información sobre el registro: servidor 1

Haciendo whois desde una línea de comando a las direcciones ip de cada servidor se encuentra información como la ubicación que se encuentra la ip, el propietario, el responsable, el país, la persona encargada del dominio (Tech-C), el bloque parcil o total (inetrev), el nombre del servidor (nserver), el correo electrónico, el contacto de la organización, información sobre servidores DNS, entre otros. De esta manera se obtiene evidencia útil de manera pública sin hacer un contacto directo con el objetivo.

4.1.3.1.3 Enumeración de las aplicaciones del servidor

Mediante la herramienta Nmap se realiza un escaneo de puertos con el fin encontrar aplicaciones que se encuentran dentro del servidor.

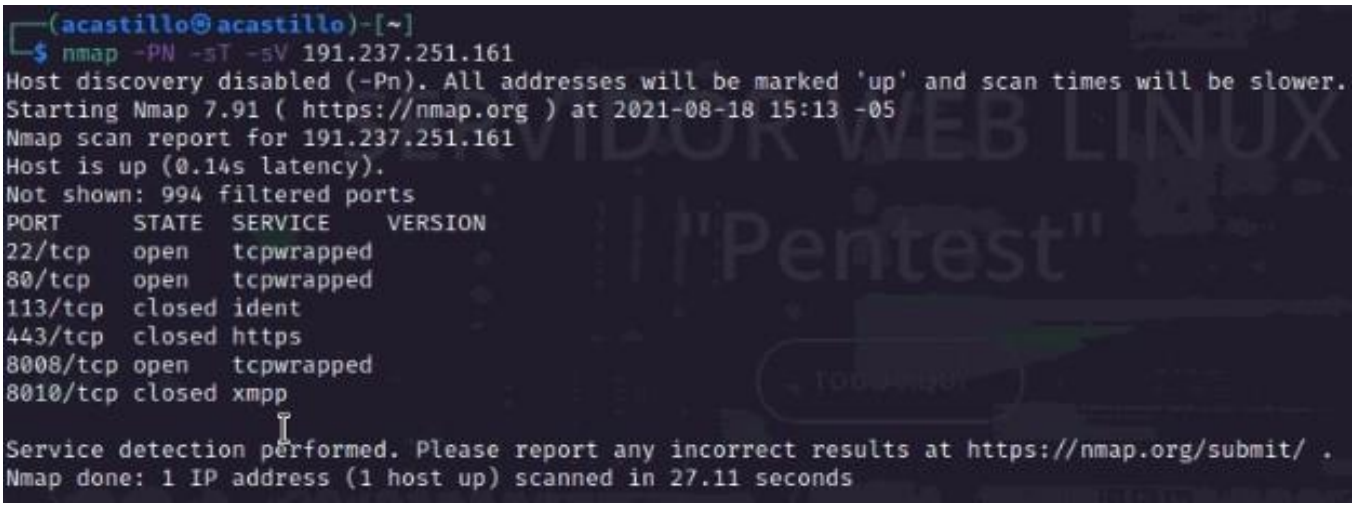


Figura 39. Escaneo de puertos : Servidor 1

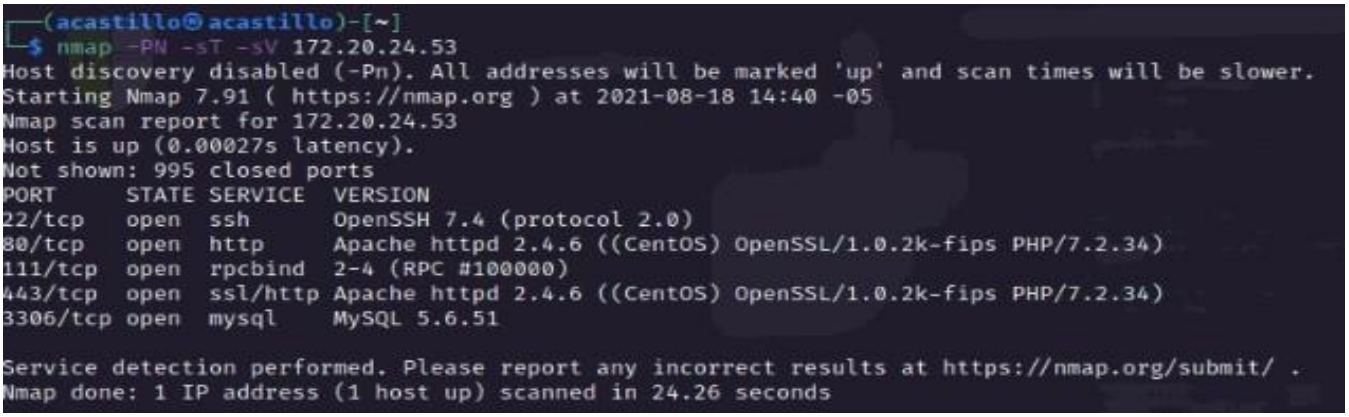


Figura 40. Escaneo de puertos a la : Servidor 2

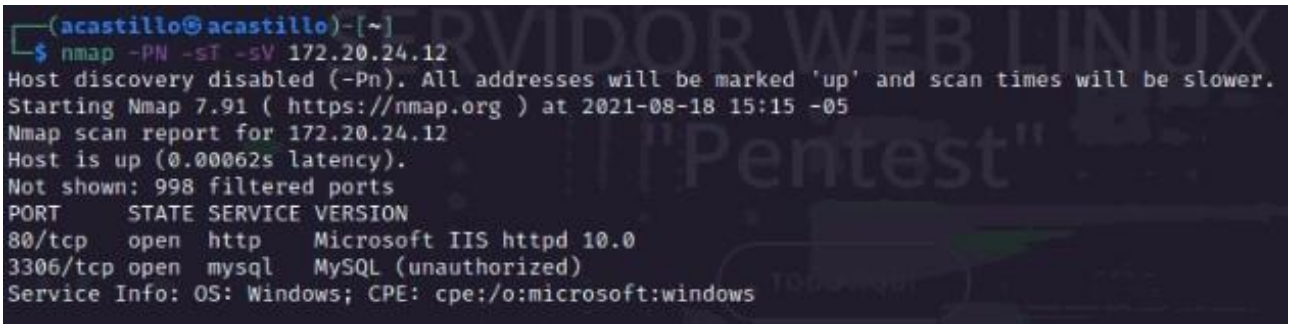


Figura 41. Escaneo de puertos : Servidor 3

Para evidenciar si existe cierto tipo de filtrado firewall.

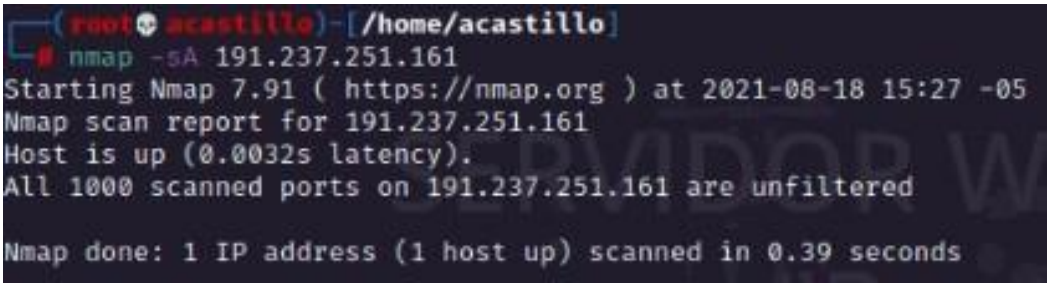


Figura 42. Escaneo de filtrado de firewall servidor 1

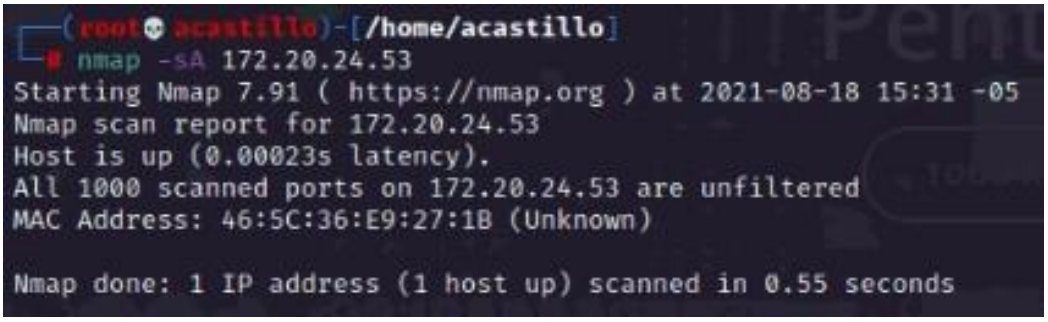


Figura 43. Escaneo de filtrado de firewall servidor 2

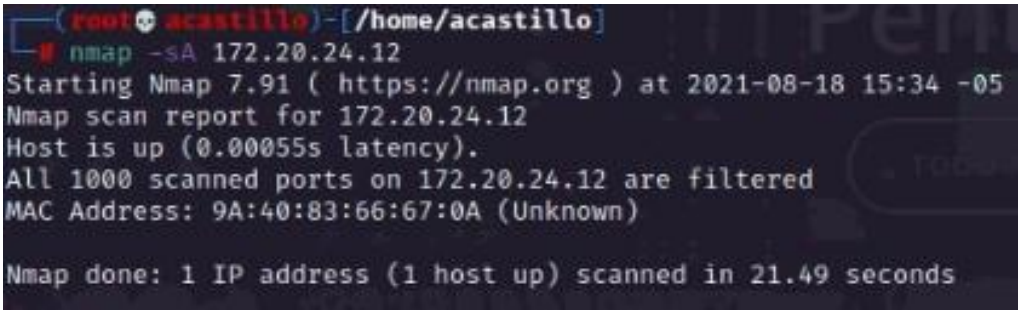


Figura 44. Escaneo de filtrado de firewall servidor 3

Para comprobar si existe puertos no viables externamente.

Figura 45. Escaneo de puertos desde red interna servidor 1

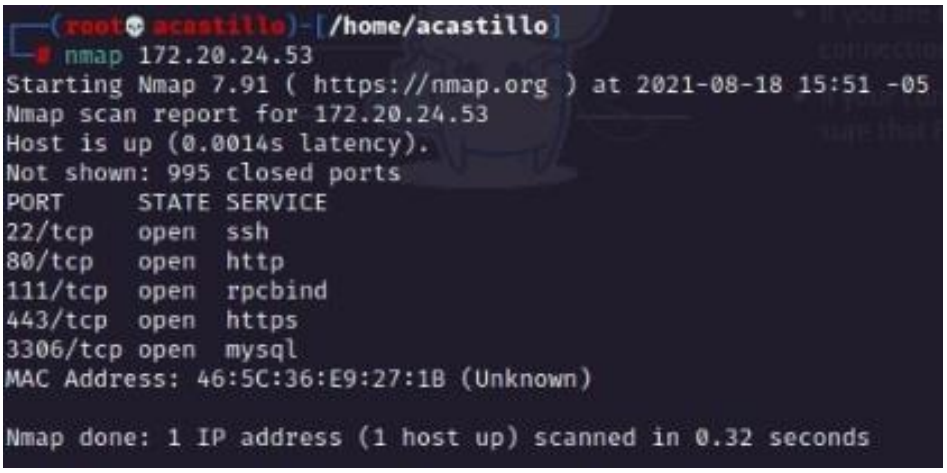


Figura 46. Escaneo de puertos desde red interna servidor 2

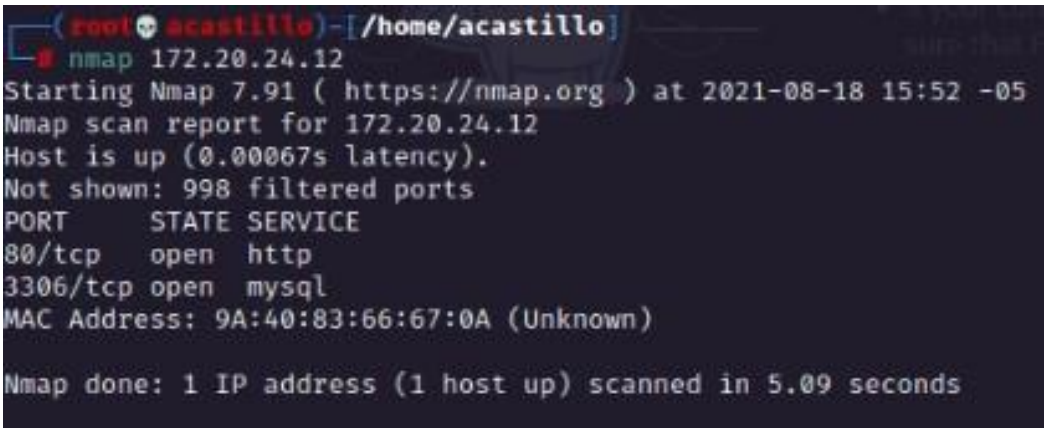


Figura 47. Escaneo de puertos desde red interna servidor 3

Se puede analizar mediante las figuras que en el caso del servidor Linux alojado en la plataforma de Microsoft Azure figura 45, que existen puertos abiertos como es el caso del protocolo 22, 80 y 8080. Seguido del servidor Linux del laboratorio de ciberseguridad indica que todos los protocolos se encuentran abiertos al igual que el servidor Microsoft IIS, en este sentido se manifiesta que su seguridad puede ser atacada mediante los protocolos abiertos.

4.1.3.1.4 Revisión de comentarios hacia sitio web para verificar la presencia de información vulnerable.

Se realiza un análisis al código de la página principal, buscando comentarios que poseen información importante.

```
<!DOCTYPE html>
<html lang="en-US">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="profile" href="http://gmpg.org/xfn/11">


<title>PENTEST &#8211; Estudiante. Alvaro Castillo</title>
<meta name='robots' content='max-image-preview:large' />
<link rel='dns-prefetch' href='//fonts.googleapis.com' />
<link rel='dns-prefetch' href='//s.w.org' />
<link rel="alternate" type="application/rss+xml" title="PENTEST &raquo; Feed" href="http://191.237.251.161/index.php/feed/" />
<link rel="alternate" type="application/rss+xml" title="PENTEST &raquo; Comments Feed" href="http://191.237.251.161/index.php/comments/feed/" />
        <script type="text/javascript">
            window._wpemojiSettings = {"baseUrl":"https:\/\/s.w.org\/images\/core\/emoji\/13.0.1\/72x72\/","ext":".png","svgUrl":"https:\/\/s.w.org\/images\/core\/emoji'
            !function(e,a,t){var n,r,o,i=a.createElement("canvas"),p=i.getContext&&i.getContext("2d");function s(e,t){var a=String.fromCharCode;p.clearRect(0,0,i.width,
        </script>
        <style type="text/css">
img.wp-smiley,
img.emoji {
    display: inline !important;
    border: none !important;
    box-shadow: none !important;
    height: 1em !important;
    width: 1em !important;
    margin: 0 .07em !important;
    vertical-align: -0.1em !important;
    background: none !important;
    padding: 0 !important;
}
</style>
    <link rel='stylesheet' id='dashicons-css'  href='http://191.237.251.161/wp-includes/css/dashicons.min.css?ver=5.7.2' type='text/css' media='all' />
<link rel='stylesheet' id='admin-bar-css'  href='http://191.237.251.161/wp-includes/css/admin-bar.min.css?ver=5.7.2' type='text/css' media='all' />
<link rel='stylesheet' id='moesia-bootstrap-css'  href='http://191.237.251.161/wp-content/themes/moesia/css/bootstrap/bootstrap.min.css?ver=1' type='text/css' media='al
<link rel='stylesheet' id='elementor-icons-css'  href='http://191.237.251.161/wp-content/plugins/elementor/assets/lib/eicons/css/elementor-icons.min.css?ver=5.11.0' type
<link rel='stylesheet' id='elementor-common-css'  href='http://191.237.251.161/wp-content/plugins/elementor/assets/css/common.min.css?ver=3.2.5' type='text/css' media='
<link rel='stylesheet' id='wp-block-library-css'  href='http://191.237.251.161/wp-includes/css/dist/block-library/style.min.css?ver=5.7.2' type='text/css' media='all' />
<link rel='stylesheet' id='moesia-style-css'  href='http://191.237.251.161/wp-content/themes/moesia/style.css?ver=5.7.2' type='text/css' media='all' />
<style id='moesia-style-inline-css' type='text/css'>
.services-area { background-color:  !important; }
.services-area .widget-title { color: ; }
.services-area .widget-title:after { border-color: ; }
.service-icon { background-color: ; }
.service-title, .service-title a { color: ; }
.service-desc { color: ; }
.employees-area { background-color:  !important; }
.employees-area .widget-title { color: ; }
.employees-area .widget-title:after { border-color: ; }
.employee-name { color: ; }
.employee-position, .employee-social a { color: ; }
.employee-desc { color: ; }
.testimonials-area { background-color:  !important; }
.testimonials-area .widget-title { color: ; }
```

Figura 48. Código HTML de la página servidor 1

```
<!DOCTYPE html>
<html lang="en-US">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="profile" href="http://gmpg.org/xfn/11">
    <title>Servidor Web Linux &#8211; Just another WordPress site</title>
<meta name='robots' content='max-image-preview:large' />
<link rel='dns-prefetch' href='//fonts.googleapis.com' />
<link rel='dns-prefetch' href='//s.w.org' />
<link rel="alternate" type="application/rss+xml" title="Servidor Web Linux &raquo; Feed" href="http://172.20.24.53/index.php/feed/" />
<link rel="alternate" type="application/rss+xml" title="Servidor Web Linux &raquo; Comments Feed" href="http://172.20.24.53/index.php/
        <script type="text/javascript">
            window._wpemojiSettings = {"baseUrl":"https:\/\/s.w.org\/images\/core\/emoji\/13.1.0\/72x72\/","ext":".png","svgUrl":"http
            !function(e,a,t){var n,r,o,i=a.createElement("canvas"),p=i.getContext&&i.getContext("2d");function s(e,t){var a=String.fro
        </script>
        <style type="text/css">
img.wp-smiley,
img.emoji {
    display: inline !important;
    border: none !important;
    box-shadow: none !important;
    height: 1em !important;
    width: 1em !important;
    margin: 0 .07em !important;
    vertical-align: -0.1em !important;
    background: none !important;
    padding: 0 !important;
}
</style>
    <link rel='stylesheet' id='extend-builder-css-css'  href='http://172.20.24.53/wp-content/plugins/colibri-page-builder/extend-build
<style id='extend-builder-css-inline-css' type='text/css'>
/* page css */
/* part css : theme-shapes */
```

Figura 49. Código HTML de la página servidor 2

```
<!DOCTYPE html>
<html lang="es">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="profile" href="http://gmpg.org/xfn/11">
    <title>Servidor IIS &#8211; Otro sitio realizado con WordPress</title>
<meta name='robots' content='max-image-preview:large' />
<link rel='dns-prefetch' href='//fonts.googleapis.com' />
<link rel='dns-prefetch' href='//s.w.org' />
<link rel="alternate" type="application/rss+xml" title="Servidor IIS &raquo; Feed" href="http://localhost/feed/" />
<link rel="alternate" type="application/rss+xml" title="Servidor IIS &raquo; Feed de los comentarios" href="http://local
        <script type="text/javascript">
            window._wpemojiSettings = {"baseUrl":"https:\/\/s.w.org\/images\/core\/emoji\/13.1.0\/72x72\/","ext":".png",
            !function(e,a,t){var n,r,o,i=a.createElement("canvas"),p=i.getContext&&i.getContext("2d");function s(e,t){va
        </script>
        <style type="text/css">
img.wp-smiley,
img.emoji {
    display: inline !important;
    border: none !important;
    box-shadow: none !important;
    height: 1em !important;
    width: 1em !important;
    margin: 0 .07em !important;
    vertical-align: -0.1em !important;
    background: none !important;
    padding: 0 !important;
}
</style>
    <link rel='stylesheet' id='extend-builder-css-css'  href='http://localhost/wp-content/plugins/colibri-page-builder/e
<style id='extend-builder-css-inline-css' type='text/css'>
/* page css */
/* part css : theme-shapes */
.colibri-shape-circles {
background-image:url('http://localhost/wp-content/themes/colibri-wp/resources/images/header-shapes/circles.png')
}
.colibri-shape-10degree-stripes {
background-image:url('http://localhost/wp-content/themes/colibri-wp/resources/images/header-shapes/10degree-stripes.png'
}
.colibri-shape-rounded-squares-blue {
```

Figura 50. Código HTML de la página servidor 3

Al momento de revisar el código de la página web mediante el navegador Chorme no hubo presencia de comentarios que sean objeto de vulnerabilidad.

4.1.3.1.5. Identificación de puntos de entrada a la aplicación.

Mediante el uso de la herramienta OWASP ZAP se obtiene información de métodos GET y POST dando como resultado lo siguiente:

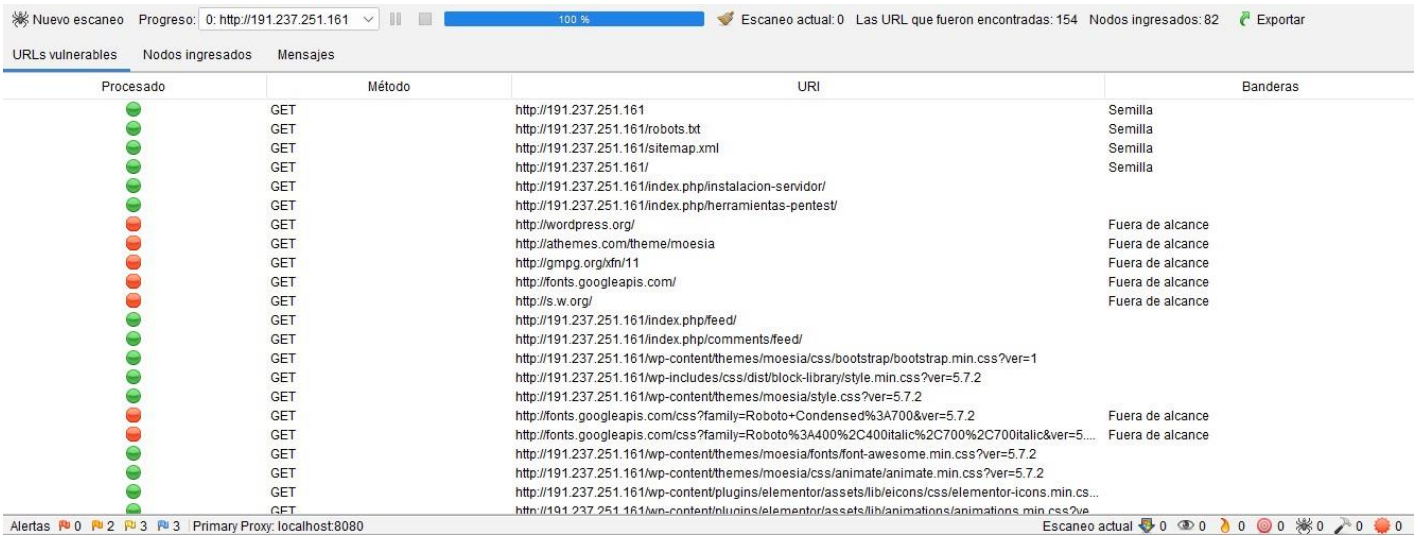


Figura 51. Métodos Get existentes en la dirección: servidor 1

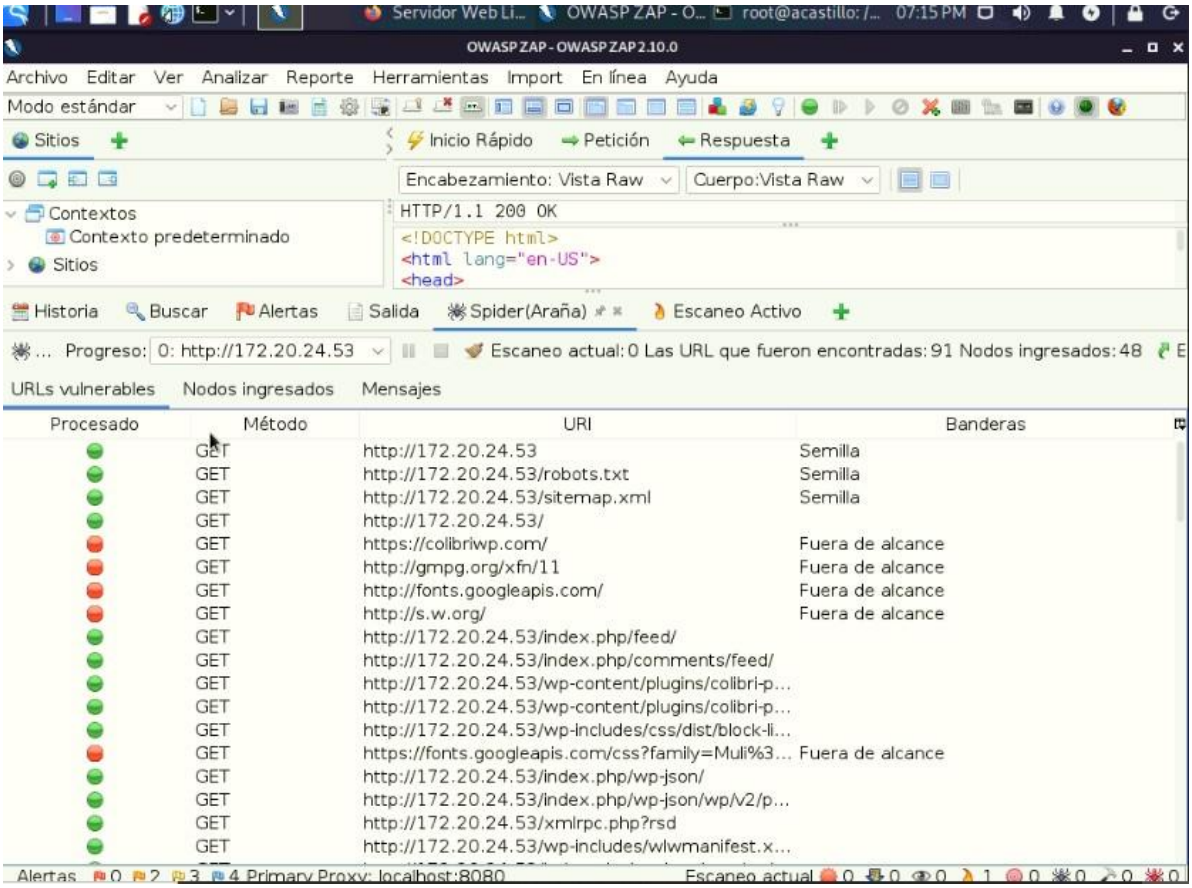


Figura 52.. Métodos Get existentes en la dirección: servidor 2

Procesado	Método	URI	Banderas
	GET	http://purl.org/dc/elements/1.1/	Fuera de alcance
	GET	http://www.w3.org/2005/Atom	Fuera de alcance
	GET	http://purl.org/rss/1.0/modules/syndication/	Fuera de alcance
	GET	http://purl.org/rss/1.0/modules/slash/	Fuera de alcance
	GET	https://wordpress.org/?v=5.7.2	Fuera de alcance
	GET	http://getbootstrap.com/	Fuera de alcance
	GET	https://github.com/twbs/bootstrap/blob/master/LICENSE	Fuera de alcance
	GET	http://www.w3.org/2000/svg	Fuera de alcance
	GET	https://athemes.com/theme/moesia/	Fuera de alcance
	GET	https://athemes.com/	Fuera de alcance
	GET	https://www.gnu.org/licenses/gpl-2.0.html	Fuera de alcance
	GET	http://underscores.me/,	Fuera de alcance
	GET	http://fontawesome.io/	Fuera de alcance
	GET	http://fontawesome.io/license	Fuera de alcance
	GET	http://daneden.me/animate	Fuera de alcance
	GET	http://opensource.org/licenses/MIT	Fuera de alcance
	GET	http://www.w3.org/1999/xlink	Fuera de alcance
	GET	http://archipelago.phrasewise.com/rsd	Fuera de alcance
	GET	https://wordpress.org/	Fuera de alcance
	GET	http://191.237.251.161/xmlrpc.php	
	GET	http://schemas.microsoft.com/wlw/manifest/weblog	Fuera de alcance

Figura 53. Métodos Get existentes en la dirección: servidor 3

Se identifico métodos POST en la sección principal del sitio, es decir los datos enviados al Http Request son visibles y pueden ser de información valiosa.

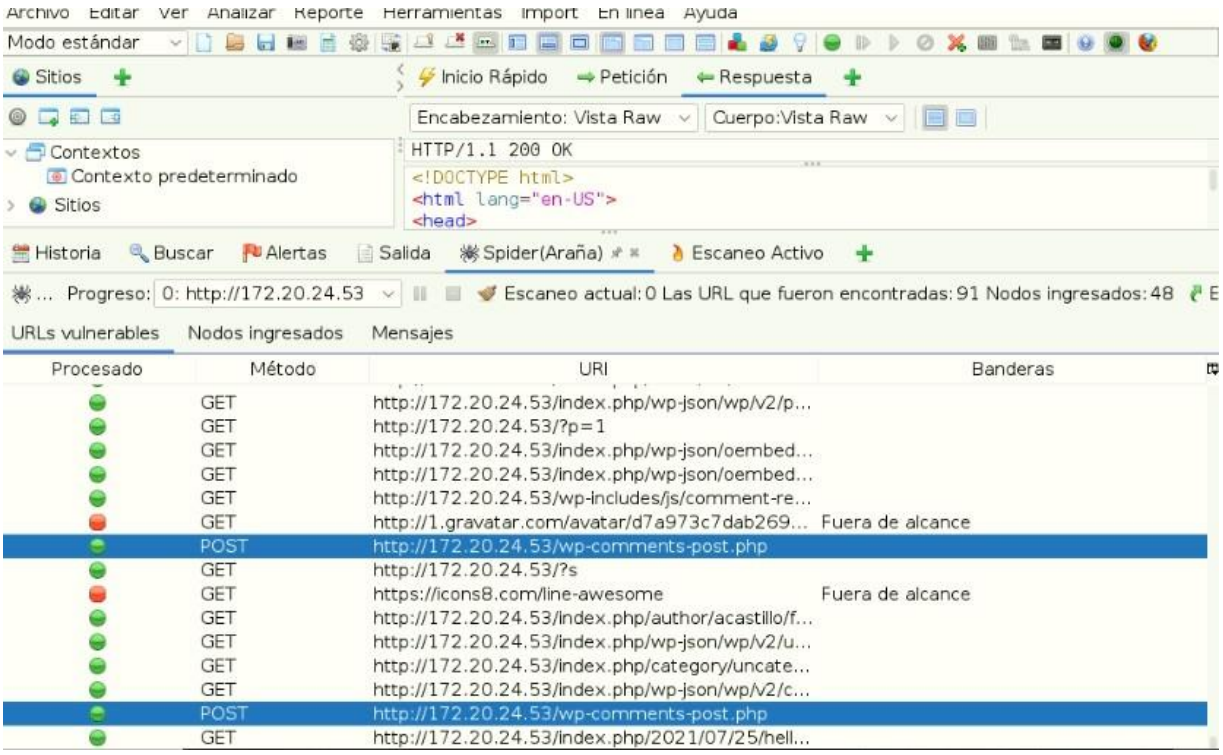


Figura 54. Método Post existente en la dirección: servidor 2

Son componentes que contienen URL de información valiosa e importante, que puede ser perjudicial para el servidor al conocer parte de su estructura.

4.1.3.1.6 Análisis al entorno del sitio web.

Mediante el comando **whatweb** podremos visualizar las configuración, versiones y tipos de estructura que se componen en un servidor web, de esta manera se podremos realizar distintas pruebas con el fin de analizar si el entorno es antiguo o no dispone de parches actualizados.

Con esta búsqueda se pudo observar que ha sido rechazada la conexión debido a que la dirección es local, no está asignada una.

Figura 55. Resultado del comando whatweb a la dirección servidor 1

Con esta búsqueda se pudo observar el lenguaje que fue desarrollado, su versión y el gestor de contenido que es instalado, en este caso Wordpress.

Figura 56. Resultado del comando whatweb a la dirección servidor 2

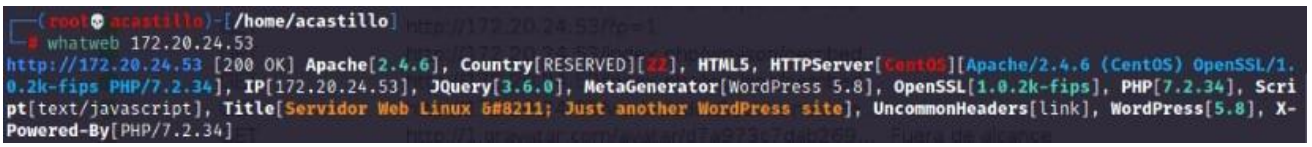


Figura 57. Resultado del comando whatweb a la dirección servidor 3

4.1.3.1.7 Alertas y análisis de la arquitectura de la aplicación

Mediante la herramienta de OWASP ZAP nos brinda un análisis profundo de la distribución de la página web, pilar para verificar errores detectar las alertas de amenazas en los servidores.

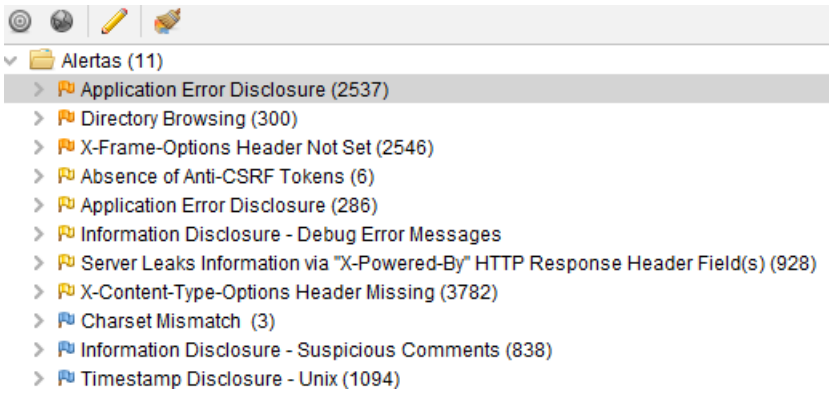


Figura 58. Alertas de vulnerabilidades servidor 1

Mediante la (tabla 22 y 23) se puede apreciar el tipo de vulnerabilidad y el riego en el que se encuentra cada uno de los servidores, indicando la alerta de cada una de ellas, por lo cual se a coloca mediante un indicar de riesgos.

Hay que tomar en cuenta los falsos positivos que arroja OWAS ZAP ya que son errores que no pueden dañar al sistema y tampoco que sea de un riesgo mayor, en este caso son riegos de nivel bajo, que pueden ser corregidos y solucionables.

Tabla 27. Vulnerabilidades al servidor web: servidor 1

Alerta	Riesgo	URL
Application Error Disclosure	Media	http://191.237.251.161/wp-content/plugins/elementor/
Directory Browsing	Media	http://191.237.251.161/wp-content/plugins/elementor/
X-Frame-Options Header Not Set	Media	http://191.237.251.161/
Absence of Anti-CSRF Tokens	Baja	http://191.237.251.161/index.php/author/acastillo/
Application Error Disclosure	Baja	http://191.237.251.161/wp-content/themes/moesia/
Information Disclosure - Debug Error Messages	Baja	http://191.237.251.161/wp-content/plugins/elementor/readme.txt
Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)	Baja	http://191.237.251.161/
X-Content-Type-Options Header Missing	Baja	http://191.237.251.161/

Charset Mismatch	Baja	http://191.237.251.161/index.php/wp-json/oembed/1.0/embed?format=xml&url=http%3A%2F%2F191.237.251.161%2F
Information Disclosure - Suspicious Comments	Baja	http://191.237.251.161/wp-includes/js/jquery/jquery-migrate.min.js?ver=3.3.2
Timestamp Disclosure - Unix	Baja	http://191.237.251.161/

Tabla 28. Total de riesgos

Riesgos	Alertas
Media	3
Baja	8
Total	11

En lo que respecta de las alertas para la dirección 172.20.24.53 servidor 2 (figura 59).

Figura 59. Alertas de vulnerabilidades servidor 2

Tabla 29. Vulnerabilidades al servidor web: servidor 2

Alerta	Riesgo	URL
Directory Browsing	Media	http:// 172.20.24.53 /wp-content/plugins/elementor/
X-Frame-Options Header Not Set	Media	http:// 172.20.24.53 /wp-content/plugins/elementor/
Absence of Anti-CSRF Tokens	Baja	http:// 172.20.24.53 /
Application Error Disclosure	Baja	http:// 172.20.24.53 /index.php/author/acastillo/
Private IP Disclosure	Baja	http:// 172.20.24.53 /wp-content/themes/moesia/
Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)	Baja	http:// 172.20.24.53 /wp-content/plugins/elementor/readme.txt
X-Content-Type-Options Header Missing	Baja	http:/ 172.20.24.53 /
Charset Mismatch (2)	Info	http:// 172.20.24.53/
Content-Type Header Missing	Info	http://172.20.24.53/index.php/wp-json/oembed/1.0/embed?format=xml&url=http%3A%2F%2F191.237.251.161%2F

Information Disclosure - Suspicious Comments	Info	http://172.20.24.53/wp-includes/js/jquery/jquery-migrate.min.js?ver=3.3.2
Timestamp Disclosure – Unix	Info	http:// 172.20.24.53 /
WSDL file passive scanner	Info	
Application error Disclosure (1694)	Media	

Tabla 30. Total de riesgos

Riesgos	Alertas
Media	3
Baja	5
Info	5
Total	13

En lo que respecta a la estructura de la página mediante la herramienta Owasp Zap tanto para el servidor Apache como Microsoft IIS.

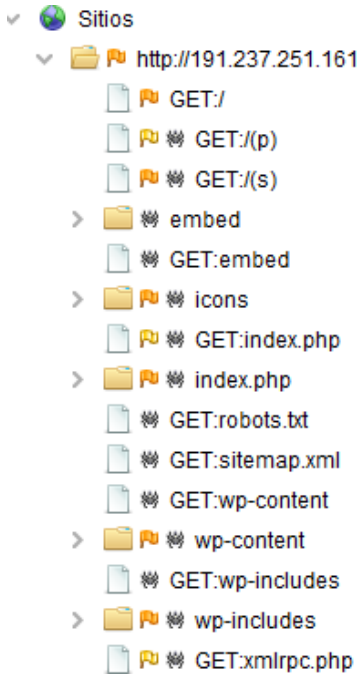


Figura 60. Estructura de la página web servidor 1

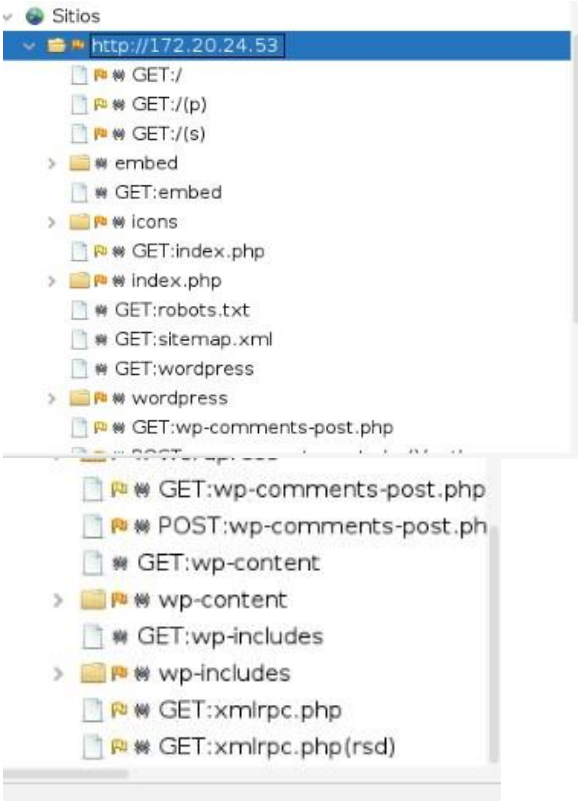


Figura 61. Estructura de la página web servidor 2

4.1.3.2. Test de manejo de configuración y desarrollo

4.1.3.2.1. Test de configuración e infraestructura

En esta fase es necesario establece las configuraciones adecuadas para cada uno de los servidores, ya que en muchas ocasiones un mínimo, puede ocasionar riegos de seguridad considerable.

Antes de utilizar las herramientas que detecten las vulnerabilidades en la página se procede analizar el acceso de administración de cada uno de los gestores de contenidos creados, en este caso se trabajará con Wordpress. Como todos los servidores fueron configurados en Wordpress la interfaz viene hacer la misma (figura 62).

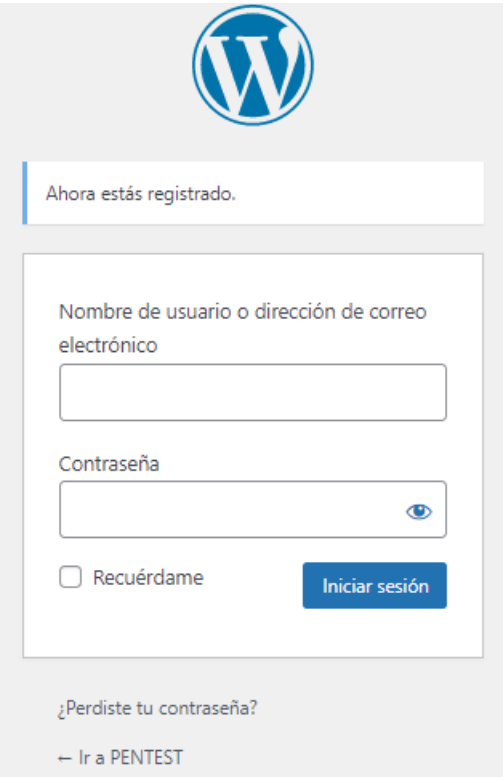


Figura 62. Inicio de acceso al panel de administración servidor 1

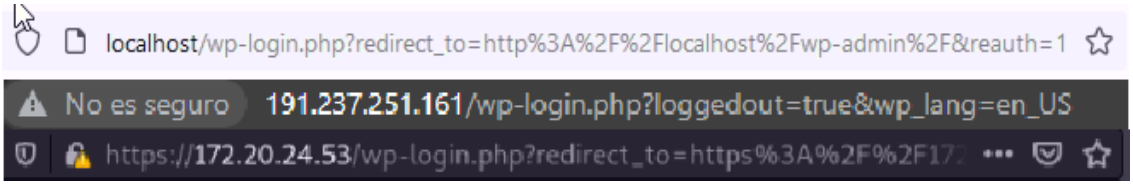


Figura 63. Login de las páginas de acceso servidor 1

Una vez identificado cada una de las páginas de acceso se realiza el análisis exhaustivo a las páginas web mediante la herramientas WPScan.

Escaneo de vulnerabilidad del tema figura (63): wpscan --url http://172.20.2.12 --enumerate vt

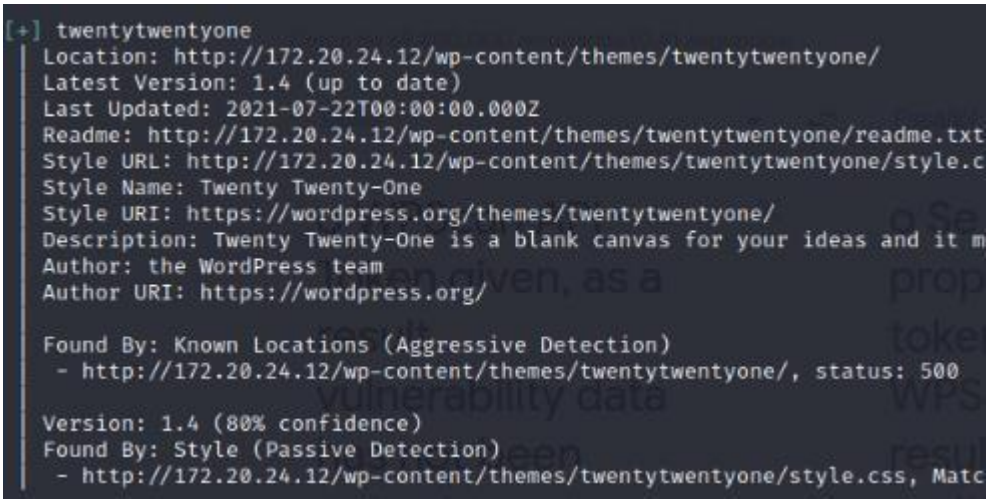


Figura 64. Vulnerabilidades del tema servidor 3

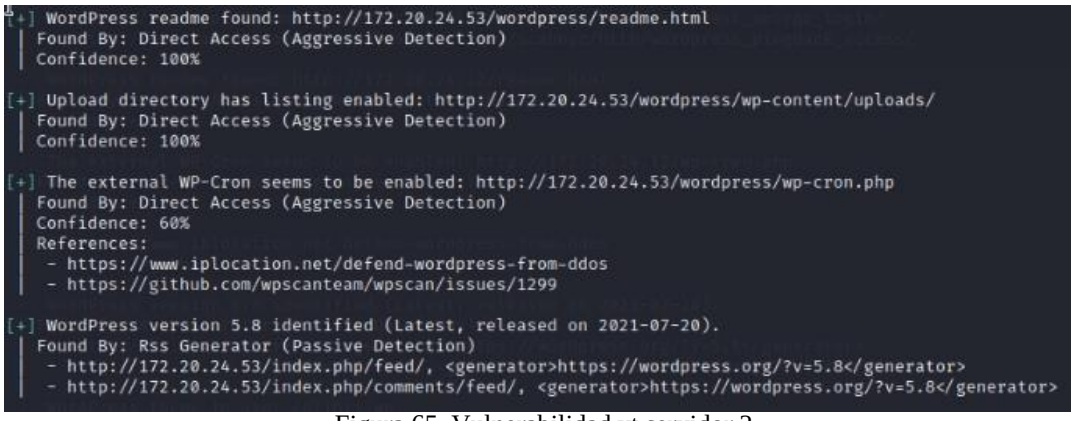


Figura 65. Vulnerabilidad vt servidor 2

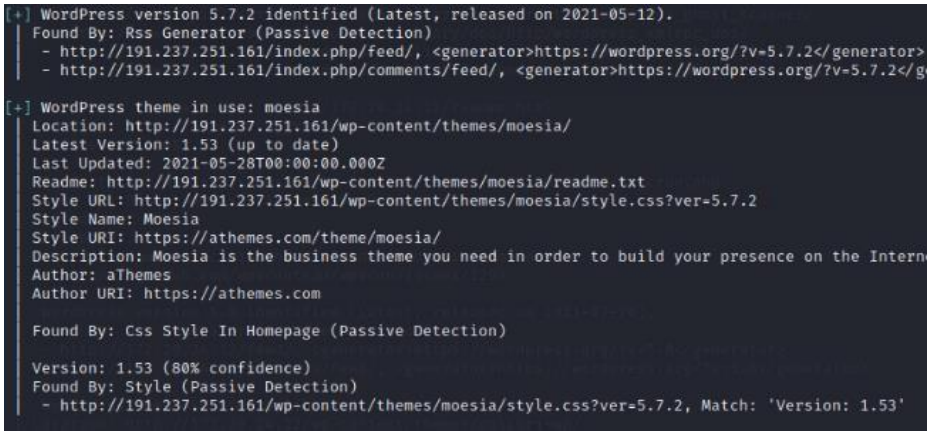


Figura 66. Vulnerabilidad vt servidor 1

Mediante el comando wpscan --url http://172.20.24.53:8080/wordpress --enumerate u, escanearemos los usuarios que se localizan en el sitio web.

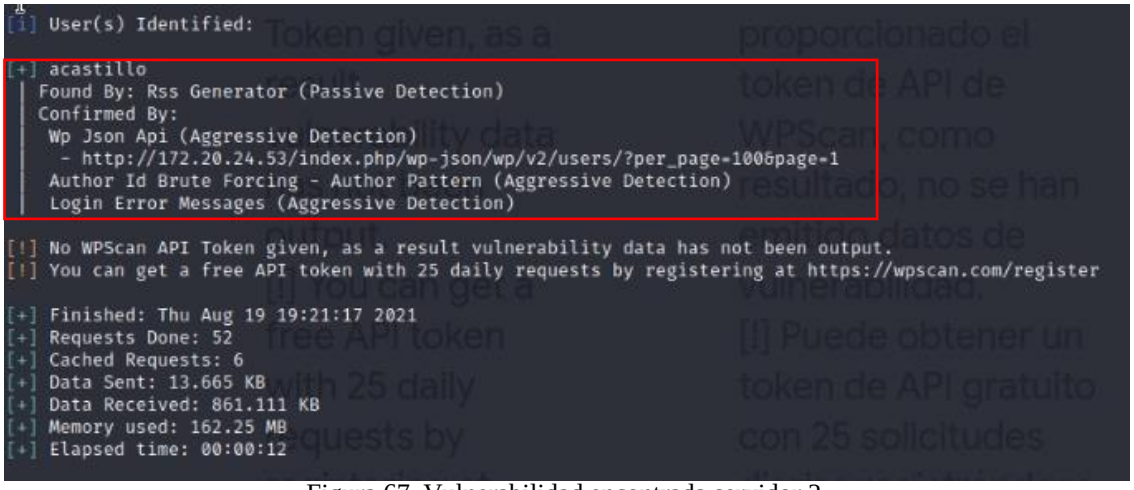


Figura 67. Vulnerabilidad encontrada servidor 2

Identifico un usuario con nombre: acastillo, logrando detectar el usuario (figura 67).

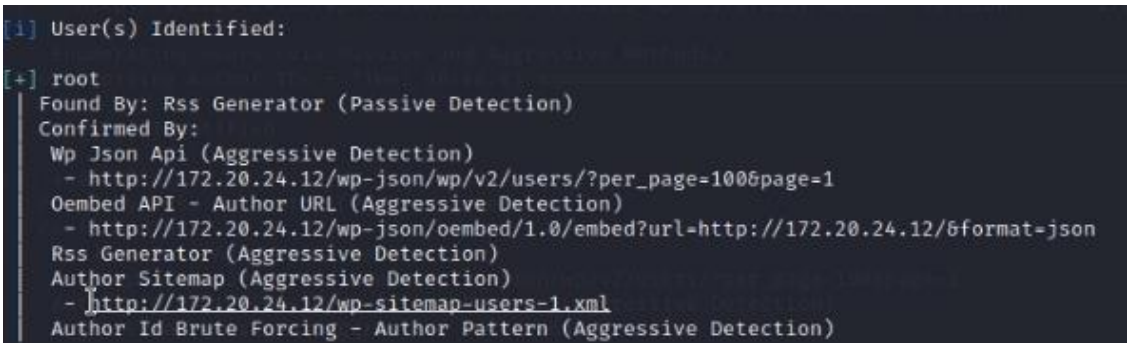


Figura 68. Vulnerabilidad encontrada servidor 3

Identifico un usuario con nombre: root, logrando detectar el usuario (figura 68).

Figura 69. Vulnerabilidad encontrada servidor 1

En este sentido se puede realizar un escaneo de fuerza bruta para identificar mediante un directorio que contenga una lista de contraseñas que pueda detectarla rápidamente mediante este comando. wpscan --url http://172.20.24.53/wordpress -P /usr/share/wordlist/numbers-as-words.txt -U acastillo.

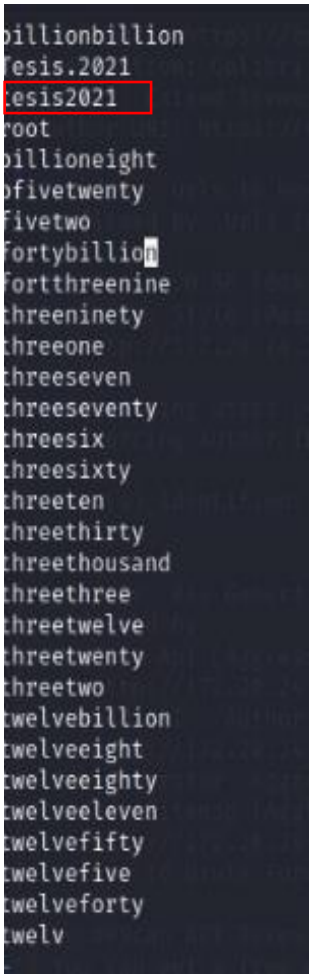


Figura 70. Contraseñas generadas

Como se puede apreciar (figura 93) la herramienta WPScan genera contraseñas encontrado la correcta (marca de rojo), una vez encontrada la clave se prosigue acceder al administrado de WordPress 172.20.24.53.

Figura 71. Acceso de fuerza bruta servidor 2

4.1.3.2.2. Test extensiones de archivos que manejan información sensible

En esta fase la extensión de los archivos que son manejados en los servidores web para comprobar que plugins, lenguajes o tecnologías son empleados para realizar dichas peticiones. De esta manera brindan información necesaria acerca del desarrollo interno de un sitio web.

Figura 72. Archivos encontrados en el sitio web servidor 2

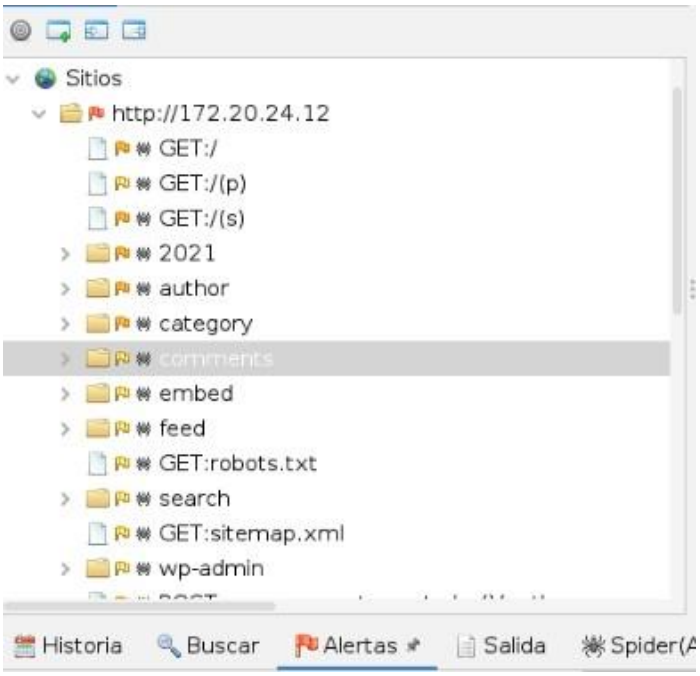


Figura 73. Archivos encontrados en el sitio web servidor 3

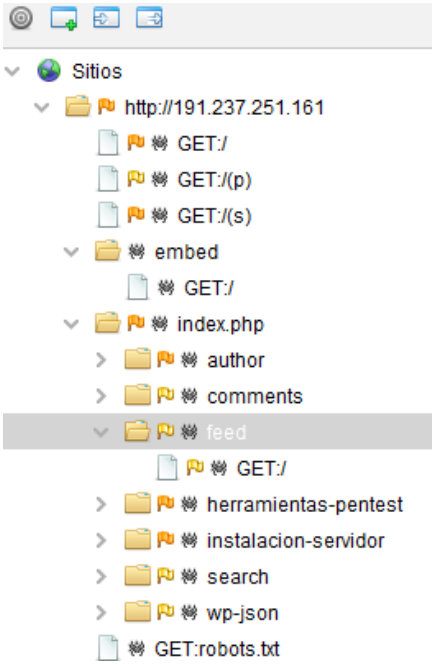


Figura 74. Archivos encontrados en el sitio web servidor 1

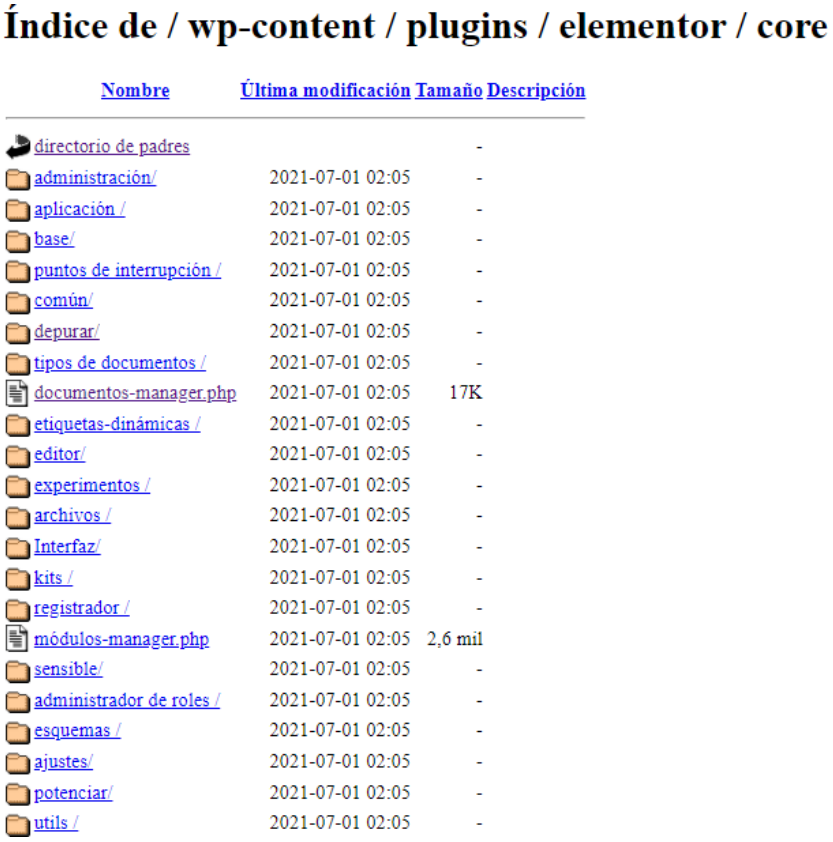
En el caso de los 3 sitios (figura 72,73,74) indican que tienen una buna configuración interna, ya que los archivos encontrados son empleados para peticiones y diseño. Caso contrario si existiesen registros con extensión .asa o inc habría un grave problema de seguridad, ya que son configurados para almacenar base de datos, es decir información sensible.

4.1.3.2.3. Revisión de archivos, backup para verificación de información sensible.

Mediante las alertas presentadas anteriormente se extrae el riego que tiene mayor tendencia a ser vulnerada mediante la herramienta OWASP ZAP.

http://191.237.251.161/wp-content/plugins/elementor/core/

Todos estos archivos dan pautas para iniciar un ataque tales como credenciales, clave, configuraciones, entre otros.



Índice de / wp-content / plugins / elementor / core

Nombre	Última modificación	Tamaño	Descripción
directorio de padres		-	
administración/	2021-07-01 02:05	-	
aplicación /	2021-07-01 02:05	-	
base/	2021-07-01 02:05	-	
puntos de interrupción /	2021-07-01 02:05	-	
común/	2021-07-01 02:05	-	
depurar/	2021-07-01 02:05	-	
tipos de documentos /	2021-07-01 02:05	-	
documentos-manager.php	2021-07-01 02:05	17K	
etiquetas-dinámicas /	2021-07-01 02:05	-	
editor/	2021-07-01 02:05	-	
experimentos /	2021-07-01 02:05	-	
archivos /	2021-07-01 02:05	-	
Interfaz/	2021-07-01 02:05	-	
kits /	2021-07-01 02:05	-	
registrador /	2021-07-01 02:05	-	
módulos-manager.php	2021-07-01 02:05	2,6 mil	
sensible/	2021-07-01 02:05	-	
administrador de roles /	2021-07-01 02:05	-	
esquemas /	2021-07-01 02:05	-	
ajustes/	2021-07-01 02:05	-	
potenciar/	2021-07-01 02:05	-	
utils /	2021-07-01 02:05	-	

Figura 75. Archivos obtenidos del sitios web servidor 1

En el caso de la figura anterior la dirección 172.20.24.12 al momento de realizar un análisis Nmap, presento una vulnerabilidad al ser detectado el phpMyAdmin, poniéndolo en un riesgo alto, ya con fuerza bruta puede acceder a la base de datos.

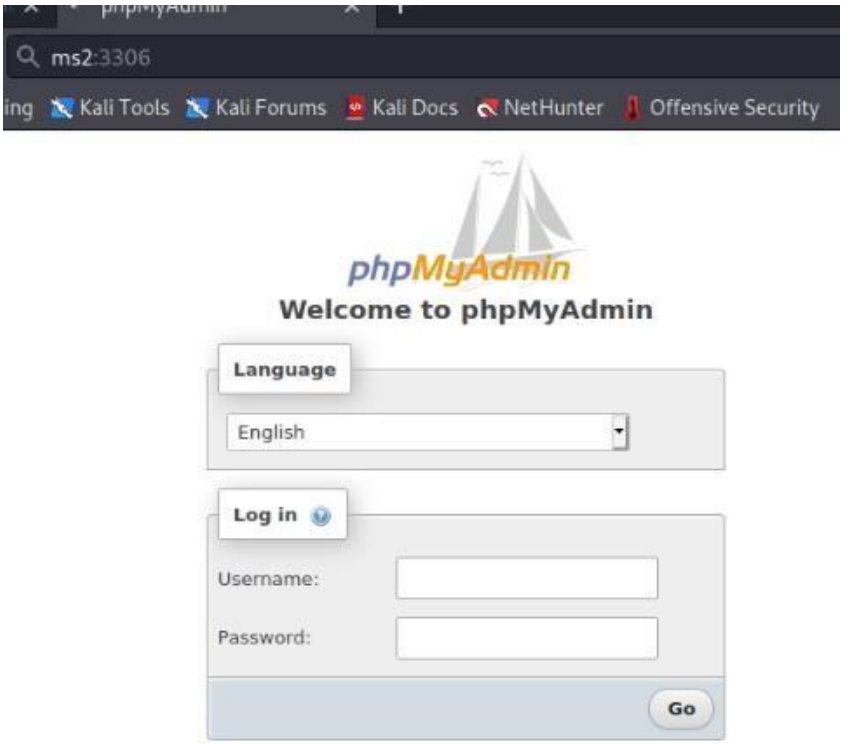


Figura 76. PhpMyAdmin vulnerado servidor 3

Mediante Nmap se pudo identificar los puertos abiertos y encontrar archivos y vulnerabilidades.

```
┌──(root💀acastillo)-[/home/acastillo]
└─# nmap -sV -oN nampto1000.txt 172.20.24.53
Starting Nmap 7.91 ( https://nmap.org ) at 2021-08-19 23:27 -05
Nmap scan report for 172.20.24.53
Host is up (0.000057s latency).
Not shown: 995 closed ports
PORT     STATE SERVICE  VERSION
22/tcp   open  ssh      OpenSSH 7.4 (protocol 2.0)
80/tcp   open  http     Apache httpd 2.4.6 ((CentOS) OpenSSL/1.0.2k-fips PHP/7.2.34)
111/tcp  open  rpcbind  2-4 (RPC #100000)
443/tcp  open  ssl/http Apache httpd 2.4.6 ((CentOS) OpenSSL/1.0.2k-fips PHP/7.2.34)
3306/tcp open  mysql    MySQL 5.6.51
MAC Address: 46:5C:36:E9:27:1B (Unknown)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 15.87 seconds

┌──(root💀acastillo)-[/home/acastillo]
└─# ls
Descargas  Documentos  Escritorio  Imágenes  Música  nampto1000.txt  Plantillas  Público  Videos

┌──(root💀acastillo)-[/home/acastillo]
└─# cat nampto1000.txt
# Nmap 7.91 scan initiated Thu Aug 19 23:27:55 2021 as: nmap -sV -oN nampto1000.txt 172.20.24.53
Nmap scan report for 172.20.24.53
Host is up (0.000057s latency).
Not shown: 995 closed ports
PORT     STATE SERVICE  VERSION
22/tcp   open  ssh      OpenSSH 7.4 (protocol 2.0)
80/tcp   open  http     Apache httpd 2.4.6 ((CentOS) OpenSSL/1.0.2k-fips PHP/7.2.34)
111/tcp  open  rpcbind  2-4 (RPC #100000)
443/tcp  open  ssl/http Apache httpd 2.4.6 ((CentOS) OpenSSL/1.0.2k-fips PHP/7.2.34)
3306/tcp open  mysql    MySQL 5.6.51
MAC Address: 46:5C:36:E9:27:1B (Unknown)
```

Figura 77. Puerto activados mediante Nmap servidor 2

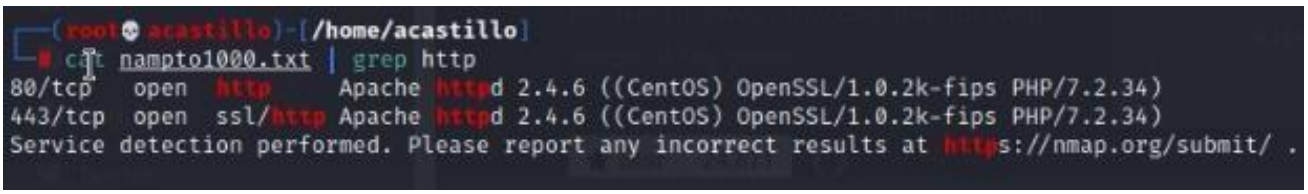


Figura 78. Puertos Http abiertos servidor 1

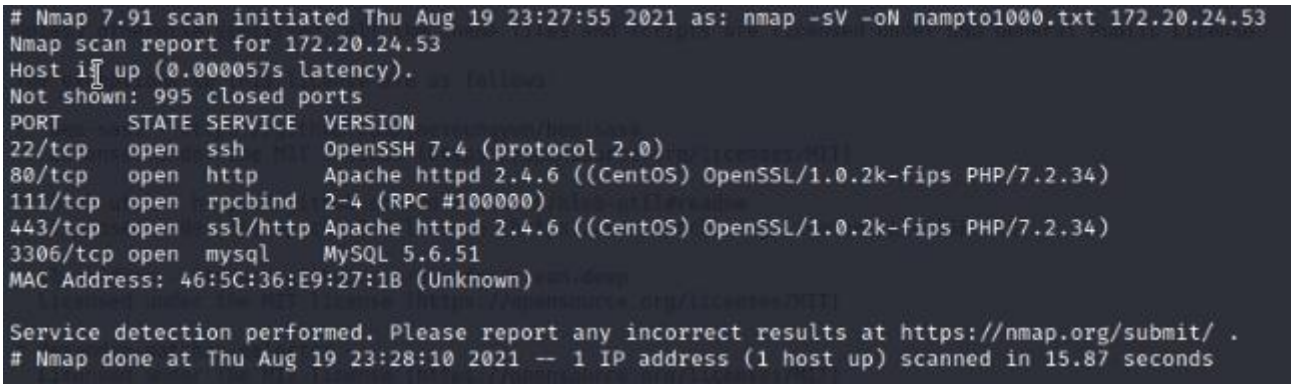


Figura 79. Puertos abiertos servidor 2

Figura 80. Puertos abiertos servidor 3

4.1.3.2.4. Test de método HTTP

En esta fase mediante el análisis de cabeceras del sitio web, podremos editar y buscar información, utilizando la herramienta OWAPZAP se consiguió tener la siguiente información.

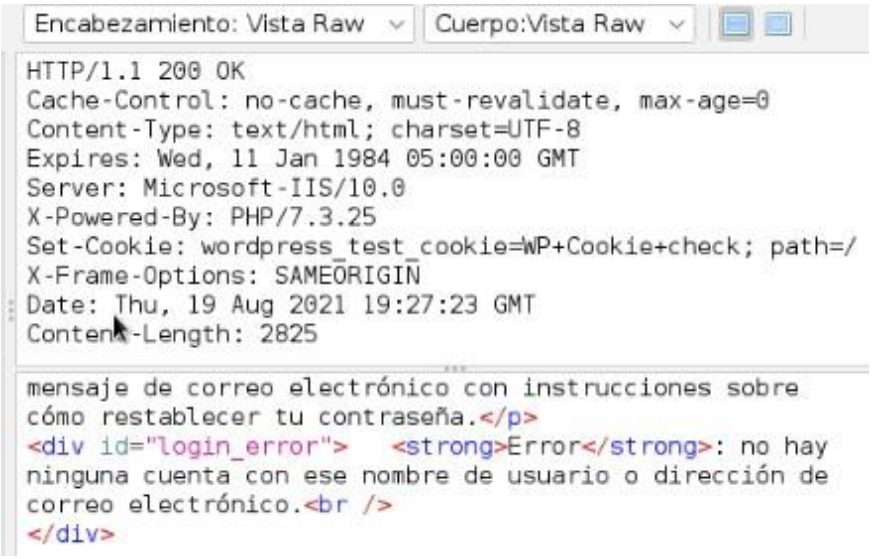


Figura 81. Consulta realizada a la página servidor 3

```
HTTP/1.1 200 OK
Date: Thu, 19 Aug 2021 21:23:07 GMT
Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/7.2
.34
X-Powered-By: PHP/7.2.34
Expires: Wed, 11 Jan 1984 05:00:00 GMT
Cache-Control: no-cache, must-revalidate, max-age=0
Content-Length: 2688
Content-Type: text/html; charset=UTF-8
```

Figura 82. Respuesta a la consulta Get servidor 3

4.1.3.2.5. Test de seguridad estricto HSTS

El transporte de seguridad estricto o HSTS es la manera en la que le servidor se comunica con el navegador de forma cifrada.

A través de un comando se observa que si pose o no política de HSTS (Seguridad para la protección contra ataques).

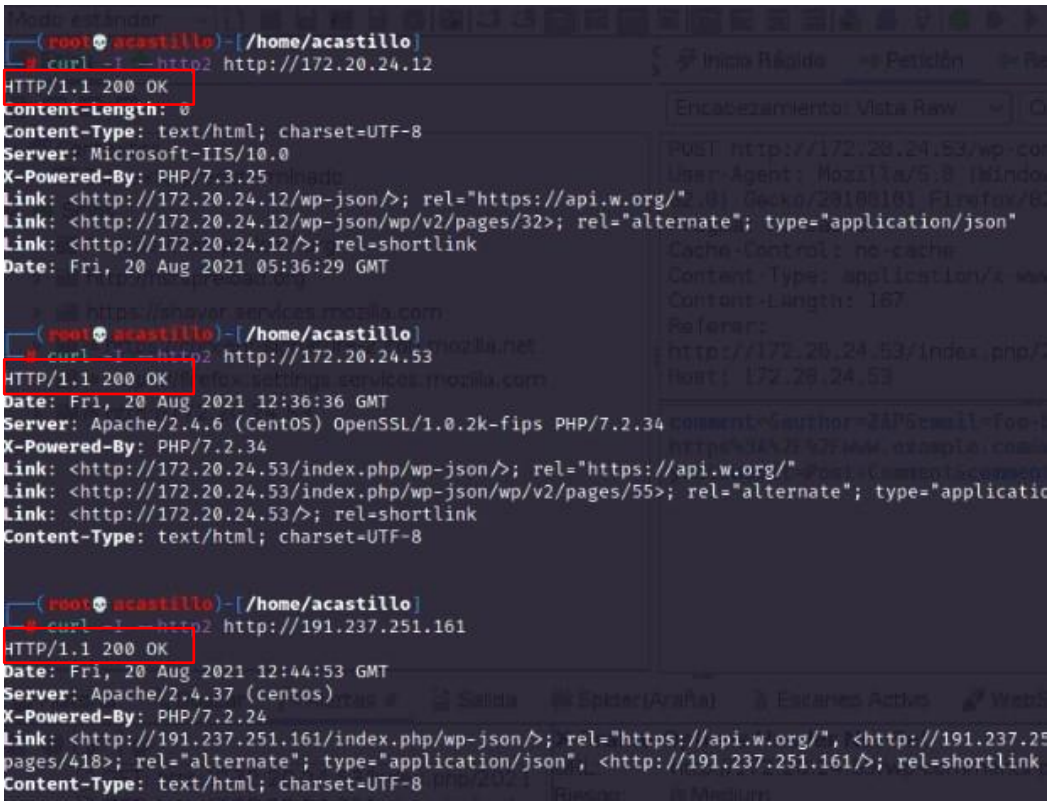


Figura 83. Análisis de existencia de HSTS

Figura 84. Análisis sin HSTS

Se puede observar que (figura 107) no presentan ningún tipo de seguridad HSTS, es decir se encuentra con un servicio HTTP, sin ningún cifrado de encriptación de seguridad.

Mientras que el resultado esperado es (figura 108)

Figura 85. Resultado esperado con HSTS

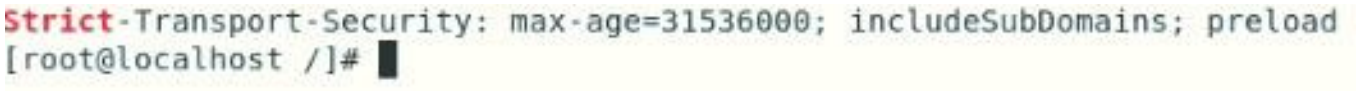


Figura 86. Resultado esperado con HSTS 2

De esta forma se garantiza que toda la información proporcionada desde el navegador hacia el sitio web sea cifrada.

El max-age = 3153600 permite generar un control de caché, el número constituye el valor de milisegundos que una petición se considera fresca, caso contrario expira y vuelve a cargarse.

4.2.3.5. Test de encabezas de seguridad

Esta fase mediante técnicas de seguridad aplicadas a los sitios web, mejoraremos la seguridad obteniendo una certificación Https a los servidores.

Figura 87. Certificado Https servidor 2

Como se observa (figura 110) la seguridad https se encuentra en color amarrillo debido a que no está verificado por una compañía. Sin embargo no quiere decir que no se encuentre seguro el sitio web.

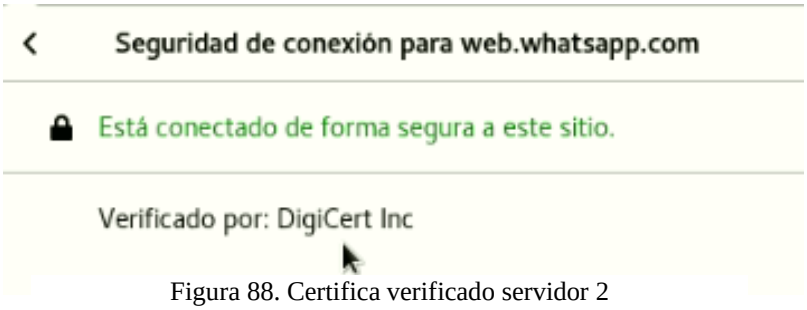


Figura 88. Certifica verificado servidor 2

En el caso (figura 88) la seguridad https no está amarrillo debido a que la compañía que lo verifica se llama DigiCert Inc.

Mediante este comando en la ubicación que se encuentra configurado el certificado SSL se cambia server-chain.crt por la compañía que lo apruebe.

```
#SSLCertificateChainFile /etc/pki/tls/certs/server-chain.crt
```

Figura 89. Comando de certificación SSL

Figura 90. Certificado crt y key

4.1.3.3. Test de manejo de identidad

4.1.3.3.1 Test de debilidades de mecanismo de cierre

En esta fase la metodología OWASP presenta una serie de pasos para la verificar el correcto sistema de cierre.

- Ingresar al sistema con identificaciones incorrectos (3 veces)
- Ir normalmente al sistema, demostrar que no se ha arroja un mecanismo de cierre.
- Ingresa al sistema con datos incorrecto (4 veces)
- Ir normalmente al sistema, demostrar que no se ha arroja ningún cierre.
- Ingresar al sistema con datos incorrectos (5 veces)
- Si existe mecanismo de cierre se bloqueará la cuenta.
- Ingresar al sistema correctamente después de (5 minutos), si se consigue el mecanismo de cierre se desactiva a los 5 minutos de ser bloqueado
- Ir de igual manera después de (10 min), si consigue que el mecanismo se desactiva a los 10 min del bloqueo
- Probar con 15 min después, si logra el mecanismo se desactiva a los 15min.

Después de haber intentado 25 veces el ingreso de la contraseña incorrecta, la cuenta no se bloqueó.

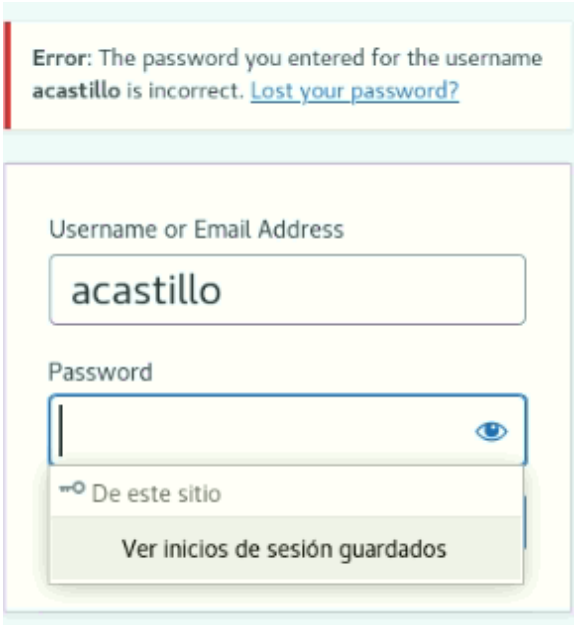


Figura 91. Intento #25 para ingreso de contraseña incorrecta

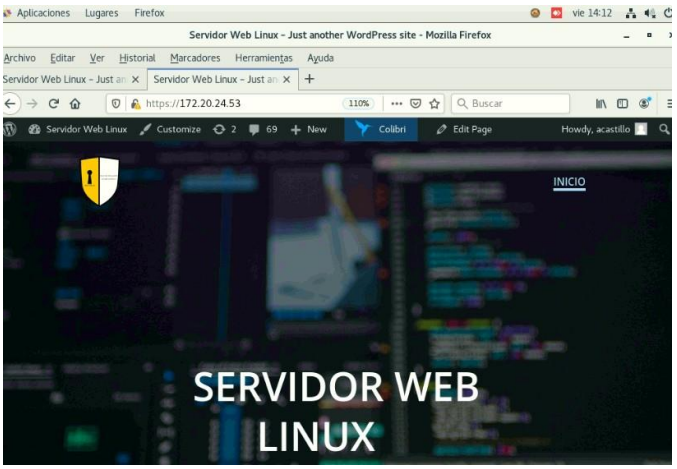


Figura 92. Ingreso exitoso servidor 1

4.1.3.3.2. Test de tiempo de espera de sesión

Mediante esta fase se analiza cuanto tiempo una sesión puede permanecer activa mientras no se realiza ninguna actividad.

Figura 93. Tiempo de espera Wordpress servidor 3

Dentro del WordPress no presenta un cierre automático. Al finalizar la sesión se debe tomar en cuenta que los tokens de sesión quedan inhabilitados, ya que el servidor se encarga de verificar el estado de sesión y es el encargado de realizar las acciones pertinentes.

4.1.3.4. Test de validación de entradas

En esta fase demuestra las vulnerabilidades que pueden ocasionar riesgos muy altos en el servidor, tales como inyecciones de código SQL, ataques a documentos y sobrecarga de buffer.

4.1.3.4.1. Test de Cross Site Scripting

Este método de ataque permite inyectar código malicioso al sitio web, esperando dar una respuesta HTTP. Mediante la utilización de Acunetix y comandos de Cross Site Scripting el sitio web detecto una alerta de XXS (Cross Site Scripting).

Método Boolean

<script>alert('XSS')</script>

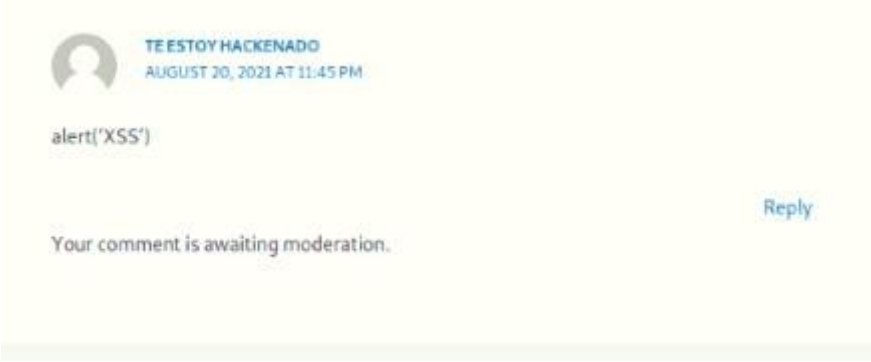


Figura 94. Inyección XXS servidor 2

De esta manera podemos analizar la vulnerabilidad que se encuentra la página al ser detectada por Cross Site Scripting.

4.1.3.4.2. Test Inyección SQL

Este método de ataque permite vulnerar desde la ip de la dirección atacante, rompiendo las medidas de seguridad con el fin de realizar operaciones sobre la bases de datos, de esta manera (figura 95) la seguridad a la base de datos ha sido vulnerada.

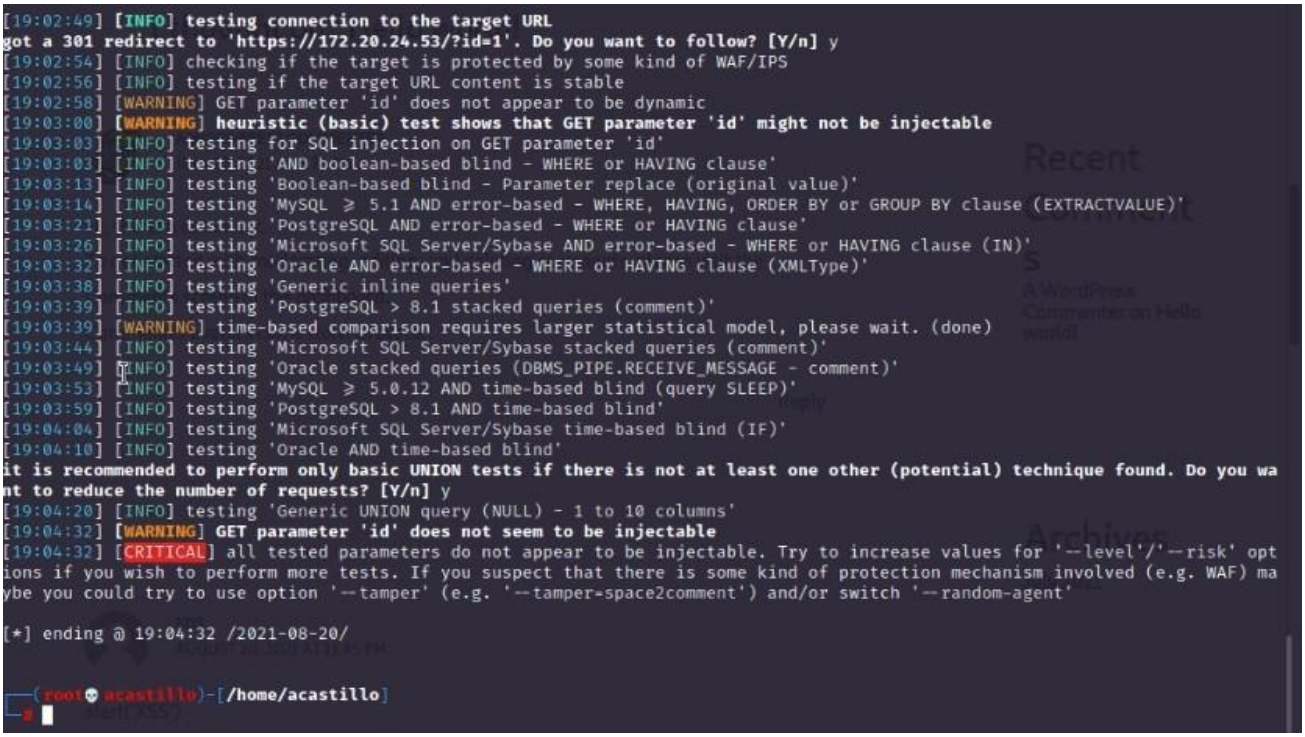


Figura 95. Inyección SQL servidor 2

4.1.3.4.3. Test Ataques (DOS)

En este ataque de negación de servicios, es provocado mediante el comando slowhttptest en la que provoca el truncamiento a la página web, durante unos segundos, eso es de acuerdo con el número de conexiones que le establece el atacante.

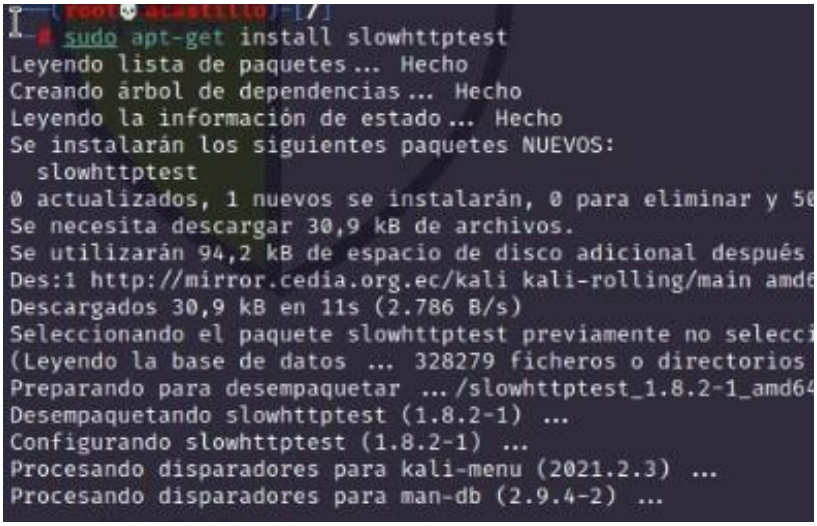


Figura 96. Instalación slowhttptest servidor 2

Figura 97. Comando de ataque DoS servidor 2

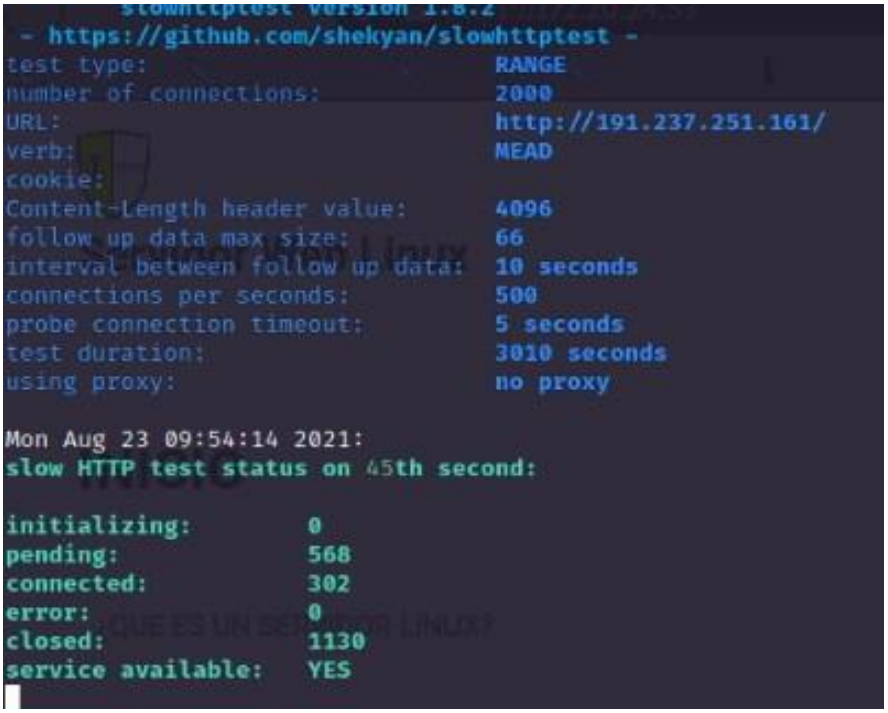


Figura 98. Truncamiento al sitio web servidor 1

4.1.3.5. Test para proteger la seguridad al servidor web: Hardening

Proceso para asegurar al servidor web a reducir el nivel de vulnerabilidad, logrando eliminar software, servicios, usuarios, entre otros.

Una vez analizado las vulnerabilidades existentes al servidor web se procedió a mitigar cada una de ellas mediante una serie de ejecuciones al servidor impidiendo que se convierta en amenaza. De esta manera el atacante no podre vulnerar el sistema ni tampoco tener visibilidad de los datos anteriormente arrojados.

4.1.3.5.1. Http Trace

Tener esta función habilitada permite al atacante un rastreo de sitios cruzado dando para al robo de información.

Al tener deshabilito evita una visualización a los protocolos con los que se configuro, permitiendo al atacante no tener acceso de dicha información.

Comprobación en el punto 4.2.3.2.2.5 (figura 99).

```
[serv-pro-cedia@localhost /]$ curl -i -X TRACE http://172.20.24.53
HTTP/1.1 405 Method Not Allowed
Date: Sat, 21 Aug 2021 17:39:12 GMT
Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/7.2.34
Allow:
Content-Length: 223
Content-Type: text/html; charset=iso-8859-1

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>405 Method Not Allowed</title>
</head><body>
<h1>Method Not Allowed</h1>
<p>The requested method TRACE is not allowed for the URL /.</p>
</body></html>
[serv-pro-cedia@localhost /]$
```

Figura 99. Seguridad a Http Request

Con esta configuración proporciona al servidores web una mejor seguridad al momento de navegar por el sitio web que es instalado y de igual manera permanece segura la información, datos y archivos que pueden ser útiles para un atacante. Esta son algunos de los comando que se utiliza para mantener seguro un sitio web ya sea información sensible de un servidor y Headers del navegador.

- TraceEnbale Off
- ServerSignature Off
- SererTokerns Pro[uctOnly]: Desactiva los Headers
- ServerTokens Major
- ServerTokens OS
- ServerTokens Full

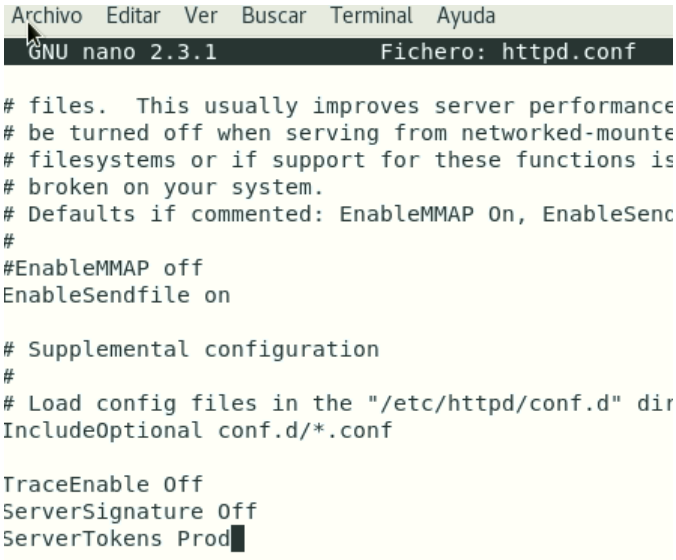


Figura 100. Seguridad a directorios servidor 2

Desactivar listado de directorios cambio de none por all para evitar la visibilidad (figura 101)

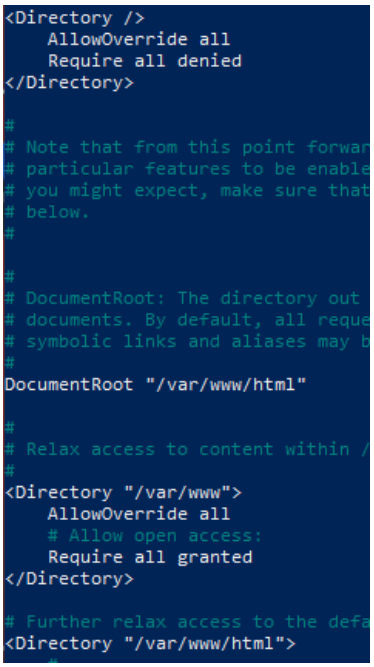


Figura 101. Mejora de seguridad a directorios servidor 1

Aun así se seguirá mirando los directorios, de esta manera se coloca los Index, es decir ocultar los archivos que se visualizan (figura 101), en la carpeta /var/www/html# nano .htaccess.

4.1.3.5.2. Eliminación de ETAG

Esta eliminación permitirá al servidor web almacenar la memoria cache con el fin de ser más eficiente y ahorra ancho de bando. Estos tag también son visibles para el atacante ya sea utilizando herramientas de vulnerabilidad como en el navegador mismo en la parte de los Headers.

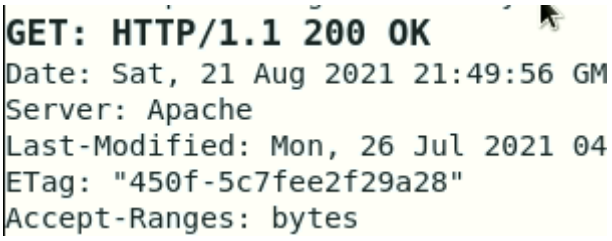


Figura 102. Etags visibles para el usuario servidor 2

Figura 103. Etags ocultos para el usuario servidor 1

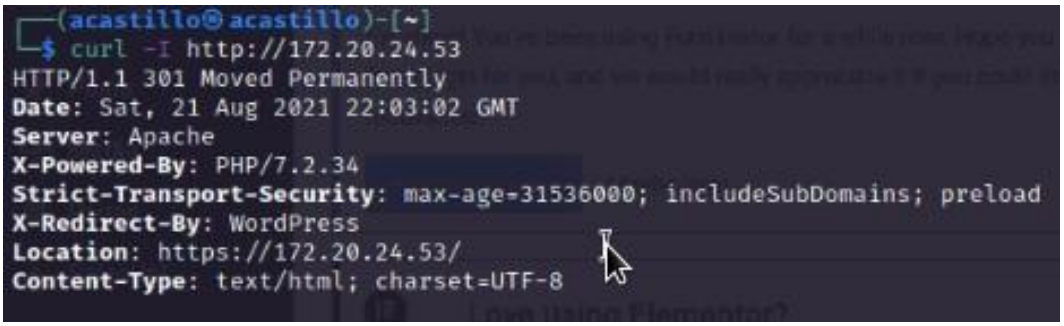


Figura 104. Etags ocultos servidor 2

4.1.3.5.3. Clickjacking Attack

Este ataque engaña al usuario para que presione click en un elemento de una página que nos es propia, haciéndose pasar por una página totalmente idéntica a la original. Provocando así que los usuarios descarguen malware, proporcionen información confidencial, acceso de credenciales o realicen compras de productos sin darse cuenta de este engaño. Este engaño por lo general se realiza mostrando un elemento html, dentro de un iframe en la parte de arriba o debajo de la página original, el usuario cree que realiza el clic en la página original, pero de hecho realiza el clic en el elemento invisible de la página maliciosa.

Existen 3 maneras para poder mitigar este error.

Deny: la página no se podrá visualizar en un frame/iframe.
Sameorgin: permite ser mostrado el frame/iframe desde un dominio propio.
Allow-From url: Solo se podrá mostrar el frame/iframe desde las url's indicadas

```
Header always append X-Frame-Options SAMEORGIN
```

Figura 105. Comando Clickjacking attack

```
[serv-pro-cedia@localhost /]$ curl -I http://172.20.24.53
HTTP/1.1 301 Moved Permanently
Date: Sat, 21 Aug 2021 22:26:12 GMT
Server: Apache
X-Frame-Options: SAMEORGIN
X-Powered-By: PHP/7.2.34
Strict-Transport-Security: max-age=31536000; includeSubDomains; preload
X-Redirect-By: WordPress
Location: https://172.20.24.53/
Content-Type: text/html; charset=UTF-8
```

Figura 106. Seguridad de Clickjacking attack servidor 2

4.1.3.5.4. Bloqueo de inyección XXS

Para evitar la inyección a la página web mediante código JavaScript o lenguaje similar, se utilizará un Mode = block que permitirá decirle al navegador que bloque las respuestas al detectar que es enviado por un atacante en lugar de desinfectar el script.

```
Header always set X-XSS-Protection "1; mode=block"
```

Figura 107. Seguridad contra la inyección XXS

4.1.3.5.5. X-Content-Type-Options

Mediante Content-Type evitara la imitación del contenido de una respuesta fuera de la que está siendo enviado por el servidor. Evitará la exposición de descargas ocultas, datos que son sensibles, entre otros.

```
Header always set X-Content-Type-Options "nosniff"
```

Figura 108. Seguridad contra imitación de contenido

4.1.3.5.6. Resumen de mejora del servidor web mediante Hardening

Como se puede apreciar en la (figura 109,110) existió un notable cambio al configurar Header en el servidor web mejorando políticas de seguridad, Opciones x-frame, protecciones de XSS, y X-Content-Type-Options. Página de escaneo https://securityheaders.com/

Figura 109. Seguridad en Headers nivel F

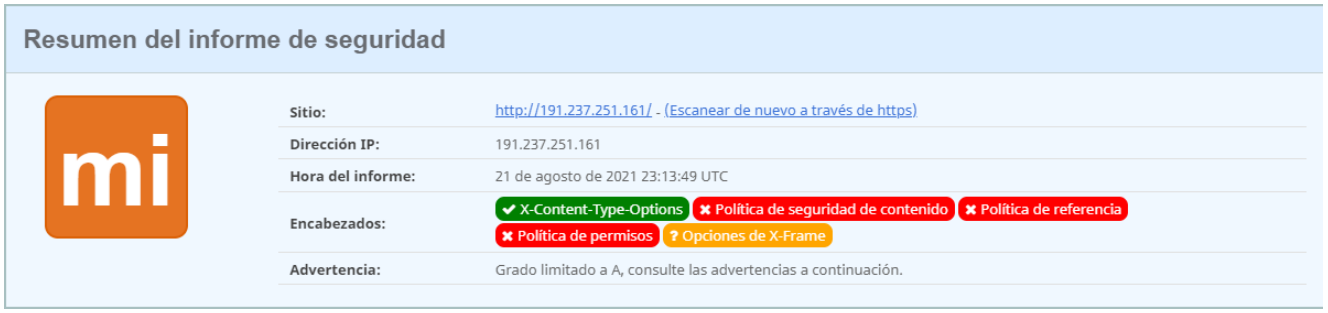


Figura 110. Seguridad en Headers nivel Mi

4.1.3.5.7 Seguridad en tiempo real para evitar ataques de fuerza bruta.

En este punto realizaremos un baneo de la página web utilizando fail2ban, cuando el servidor detecte una anomalía de fuerza bruta al sitio, de esta manera se ejecutará el proceso de baneo el tiempo que se requiera, de esta manera el atacante no podrá tener acceso al sitio web, impidiendo el robo de información.

```
[root@localhost /]# yum install fail2ban
Complementos cargados:fastestmirror, langpacks
Loading mirror speeds from cached hostfile
epel/x86_64/metalink                                    |  43 kB     00:00
 * base: mirror.uepg.br
 * epel: ewr.edge.kernel.org
 * extras: centos.itsbrasil.net
 * remi-php72: fr2.rpmfind.net
 * remi-safe: fr2.rpmfind.net
 * updates: mirror.uepg.br
base                                                    | 3.6 kB     00:00
cloudera-manager                                        | 2.9 kB     00:00
epel                                                    | 4.7 kB     00:00
extras                                                  | 2.9 kB     00:00
mysql-connectors-community                              | 2.6 kB     00:00
mysql-tools-community                                   | 2.6 kB     00:00
mysql56-community                                       | 2.6 kB     00:00
remi-php72                                              | 3.0 kB     00:00
remi-safe                                               | 3.0 kB     00:00
updates                                                 | 2.9 kB     00:00
(1/2): epel/x86_64/primary_db                             | 6.9 MB   00:07
(2/2): epel/x86_64/updateinfo                             | 1.0 MB   00:11
Resolviendo dependencias
--> Ejecutando prueba de transacción
---> Paquete fail2ban.noarch 0:0.11.1-10.el7 debe ser instalado
--> Procesando dependencias: fail2ban-firewalld = 0.11.1-10.el7 para el paquete: fail2ban-0.
--> Procesando dependencias: fail2ban-sendmail = 0.11.1-10.el7 para el paquete: fail2ban-0.1
--> Procesando dependencias: fail2ban-server = 0.11.1-10.el7 para el paquete: fail2ban-0.11.
--> Ejecutando prueba de transacción
---> Paquete fail2ban-firewalld.noarch 0:0.11.1-10.el7 debe ser instalado
---> Paquete fail2ban-sendmail.noarch 0:0.11.1-10.el7 debe ser instalado
```

Figura 111. Instalación de Fail2ban servidor 2

Como se muestra en la (figura 111) el tiempo de baneo que se lo otorgará será de 30 segundos, es decir el tiempo que la pagina se encontrará bloqueada, al igual que el tiempo de inactividad de ese parámetro y los tiempos de intento del baneo sera 5, ya sea para los shh o las jaulas a nivel local.

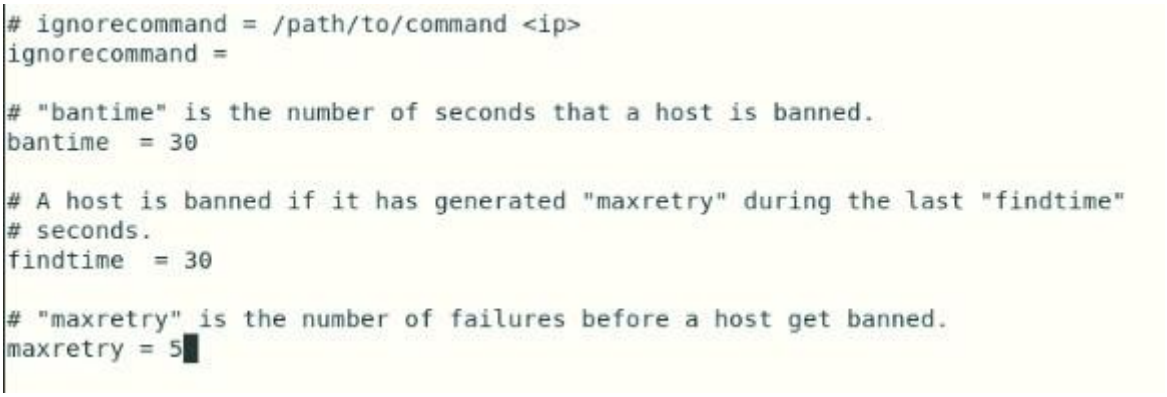


Figura 112. Tiempo de duración del Baneo servidor 2

A través de la instalación de fail2ban podremos utilizar jaulas que es el elemento más importante para activar y desactivar la configuraciones para proteger algunos de nuestros servicios, de esta manera podemos decir que la jail o jaulas nos ayudan a controlar intentos de entradas de amenazas.

En la jaula de HTTP server colocaremos lo siguiente, que indicará el baneo asignado al sitio web.

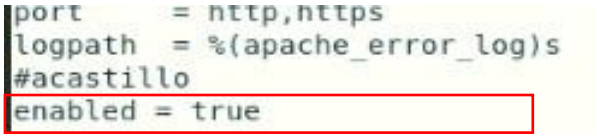


Figura 113. Activación de jails

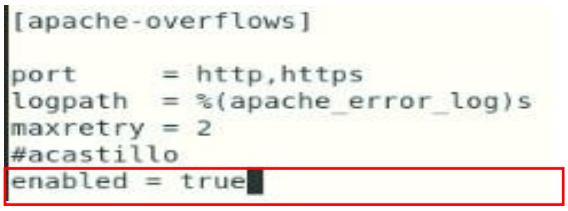


Figura 114. Activación de jaulas 2

Se desactivan el apache-nohome y apache-botsearchg

Figura 115. Apache-nohome y botsearch servidor 2

Finalmente se modifica en la aplicación la protección del servidor web conocida como **mod_security** que analiza el posible ataque en tiempo real que no se queda como almacenamiento de cache.

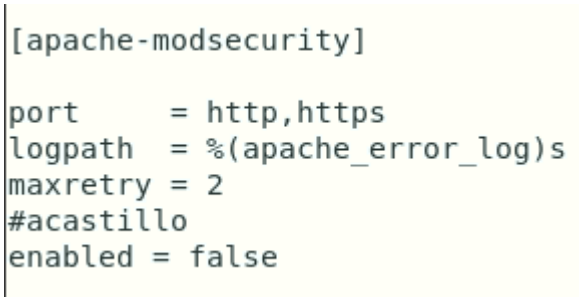


Figura 116. Apache-modsecurity servidor 2

Una vez configurado el mod_security nos arrojará la información en tiempo real de como la página se baneará con las indicaciones anteriormente configuradas.

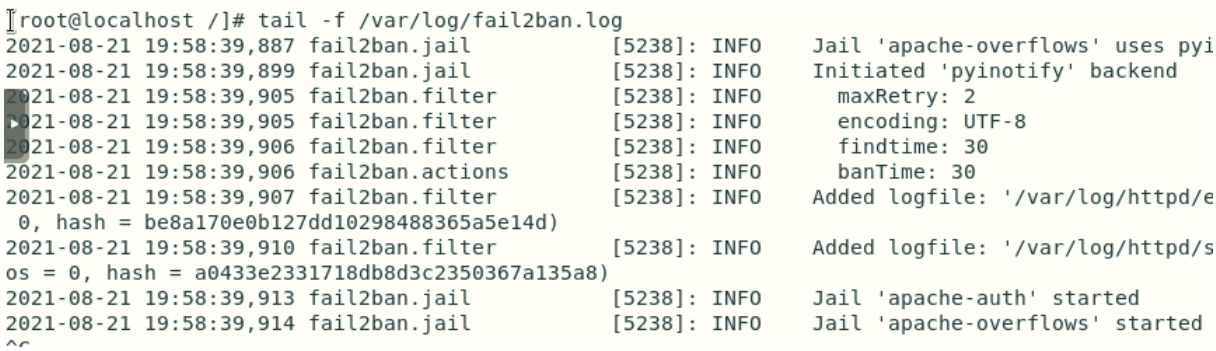


Figura 117. Mod_Security servidor 2

Finalmente para darle valor agregado se instaló un servicio en donde los resultados que arrojan en tiempo real son enviados al correo correspondiente.

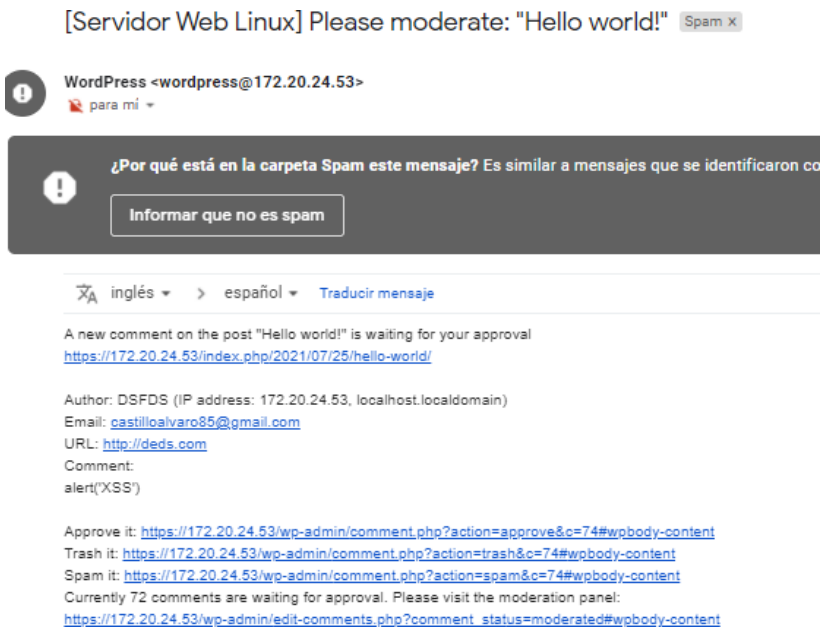


Figura 118. Test enviado por el servidor Linux

Figura 119. Confirmación de la validez del test Fil2ban

4.1.3.6. Resultados de los servidores web

4.1.3.6.1. Instalación y configuración de los servidores web.

Una vez configurado e instalado los tres servidores web elaborados para el cumplimiento referente al análisis de vulnerabilidad y al nivel de seguridad que cada uno de ellos presentan, se hará una breve explicación de su instalación y configuración, mencionando su importancia, ventajas y desventajas que presentan, finalmente obteniendo un cuadro comparativo que indique cuál de los servidores web cumplen con requerimientos óptimos tanto en seguridad,

alojamiento, rendimiento entre otros, con el fin de demostrar que servidores web son más factible a la hora levantar un sitio web.

4.1.3.6.2. Como primer servidor a analizar esta Apache en la plataforma de Microsoft Azure.

El servidor de Microsoft Azure, es un servidor compartido que forma parte de Microsoft, contando con característica y servicios que la plataforma ofrece, este servidor se aloja en una en un framework de nube publica, existen diferentes maneras de suscripción: la primera es de forma gratuita para estudiantes que tiene una vigencia de 1 a 3 meses y la pagada que cuenta con pago por uso para cada servicio que se desee utilizar. Además, el servidor de Microsoft Azure cuenta con un panel de información mediante una máquina virtual que soporta sistemas operativos Linux, y Windows como también brinda una **ip publica** que se le asigna para la utilización de su sitio web. Finalmente, el servidor web Microsoft Azure, brinda dentro de sus servicios una protección del sitio mediante Azure firewall como también protección y certificación de datos pero con un costo adicional para su funcionamiento.
Microsoft Azure está encaminado en modelos de nubes híbridas permitiendo una integración muy sencilla con otras herramientas propias de Microsoft, a la hora de instalar el servidor se procede mediante ssh siguiente del nombre de usuario, como se indica a continuación.

Análisis

De esta manera se procede a la configuración del servidor web, que está montada en una máquina virtual alojada en la plataforma de Microsoft Azure. Se instalo el sistema operativo **Linux (Centos 8.4.2105)** con un tamaño estandar de 81ms (1vcpu, 2 gb de memoria), y respectivamente su configuración LAMP, es decir servidor web Apache, como base de datos maría DB y Php con su versión actual y como gestor de contenido conocido como **“Wordpress”.**

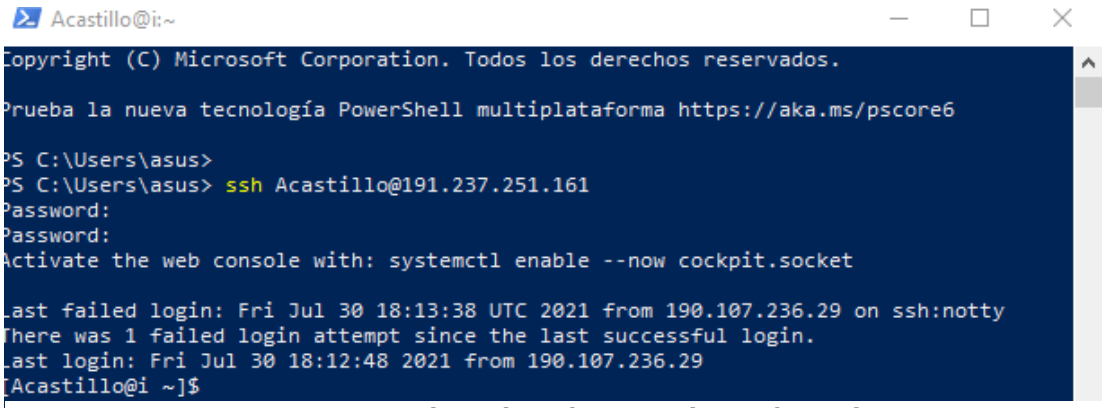
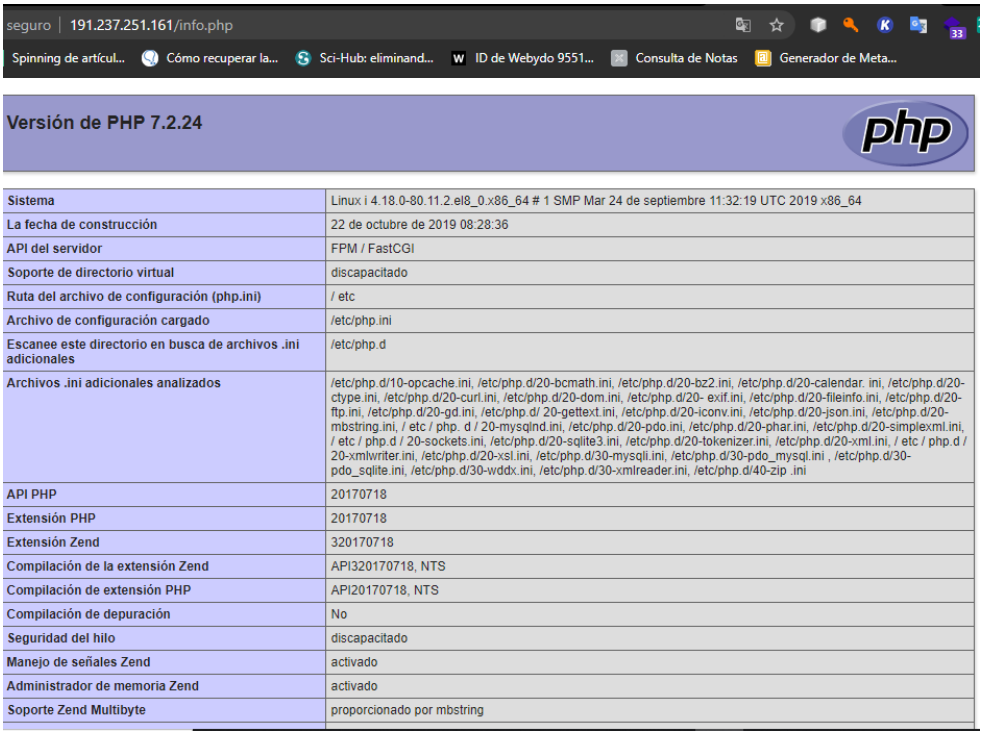


Figura 120. Acceso al servidor web Azur mediante ssh servidor 1

Con la siguiente ruta: **191.237.251.161/php.info**



Sistema	Linux i 4.18.0-80.11.2.el8_0.x86_64 # 1 SMP Mar 24 de septiembre 11:32:19 UTC 2019 x86_64
La fecha de construcción	22 de octubre de 2019 08:28:36
API del servidor	FPM / FastCGI
Soporte de directorio virtual	discapacitado
Ruta del archivo de configuración (php.ini)	/ etc
Archivo de configuración cargado	/etc/php.ini
Escanee este directorio en busca de archivos .ini adicionales	/etc/php.d
Archivos .ini adicionales analizados	/etc/php.d/10-opcache.ini, /etc/php.d/20-bcmath.ini, /etc/php.d/20-bz2.ini, /etc/php.d/20-calendar. ini, /etc/php.d/20-ctype.ini, /etc/php.d/20-curl.ini, /etc/php.d/20-dom.ini, /etc/php.d/20- exif.ini, /etc/php.d/20-fileinfo.ini, /etc/php.d/20-ftp.ini, /etc/php.d/20-gd.ini, /etc/php.d/ 20-gettext.ini, /etc/php.d/20-iconv.ini, /etc/php.d/20-json.ini, /etc/php.d/20-mbstring.ini, / etc / php. d / 20-mysqlnd.ini, /etc/php.d/20-pdo.ini, /etc/php.d/20-phar.ini, /etc/php.d/20-simplexml.ini, / etc / php.d / 20-sockets.ini, /etc/php.d/20-sqlite3.ini, /etc/php.d/20-tokenizer.ini, /etc/php.d/20-xml.ini, / etc / php.d / 20-xmlwriter.ini, /etc/php.d/20-xsl.ini, /etc/php.d/30-mysqli.ini, /etc/php.d/30-pdo_mysql.ini , /etc/php.d/30-pdo_sqlite.ini, /etc/php.d/30-wddx.ini, /etc/php.d/30-xmlreader.ini, /etc/php.d/40-zip .ini
API PHP	20170718
Extensión PHP	20170718
Extensión Zend	320170718
Compilación de la extensión Zend	API320170718, NTS
Compilación de extensión PHP	API20170718, NTS
Compilación de depuración	No
Seguridad del hilo	discapacitado
Manejo de señales Zend	activado
Administrador de memoria Zend	activado
Soporte Zend Multibyte	proporcionado por mbstring

Figura 121. Configuración de php versión 7.2.24.

Con la siguiente ruta: **191.237.251.161**

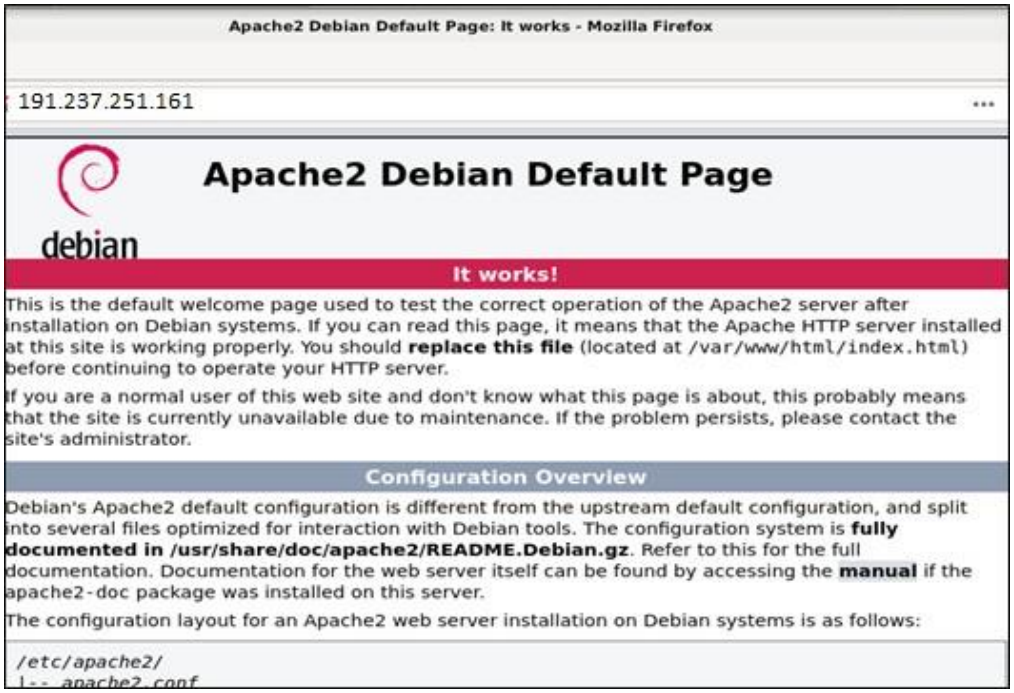


Figura 122. Configuración de Apache

En esta imagen se puede observar la base de datos que fue configurada e instalada.

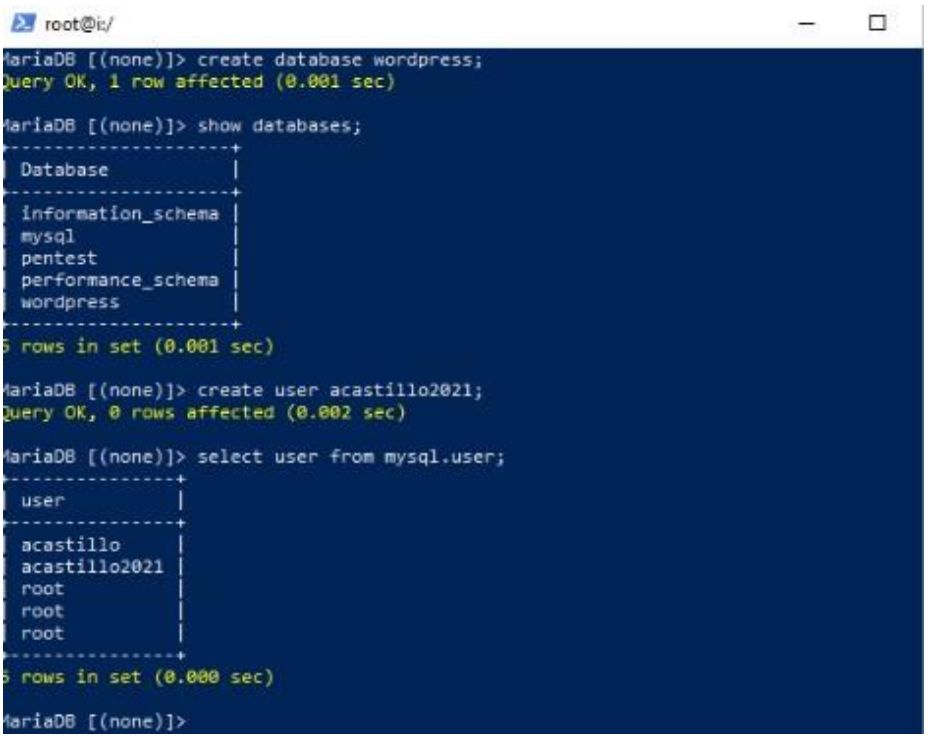


Figura 123. Configuración MariaDB.

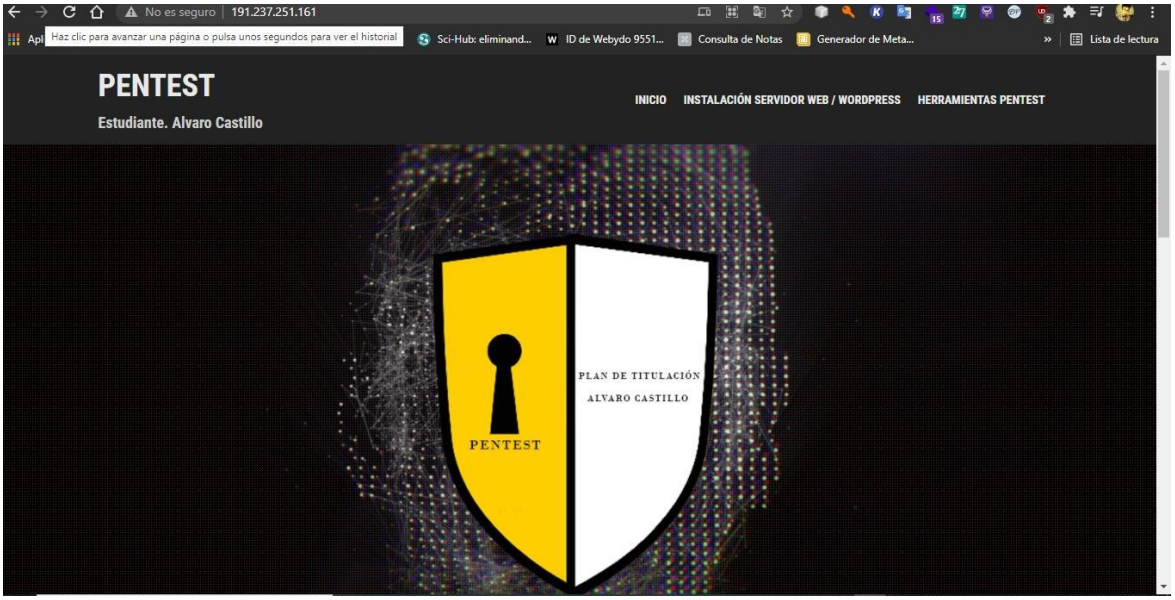


Figura 124. Primer Sitio Web con servidor Azure/Linux servidor 1

4.1.3.6.3. Métricas clave de Microsoft Azure

En la parte de métricas claves se puede observar el rendimiento del CPU, el tráfico de red y los bytes del disco duro, que se encuentra alojado en la plataforma de Microsoft Azure como otro servicio para la ejecución del servidor web en este caso de Apache-Linux. En este sentido muestra los datos del último periodo, tanto por horas o días.

Figura 125. Métricas clave de Microsoft Azure-Apache servidor 1

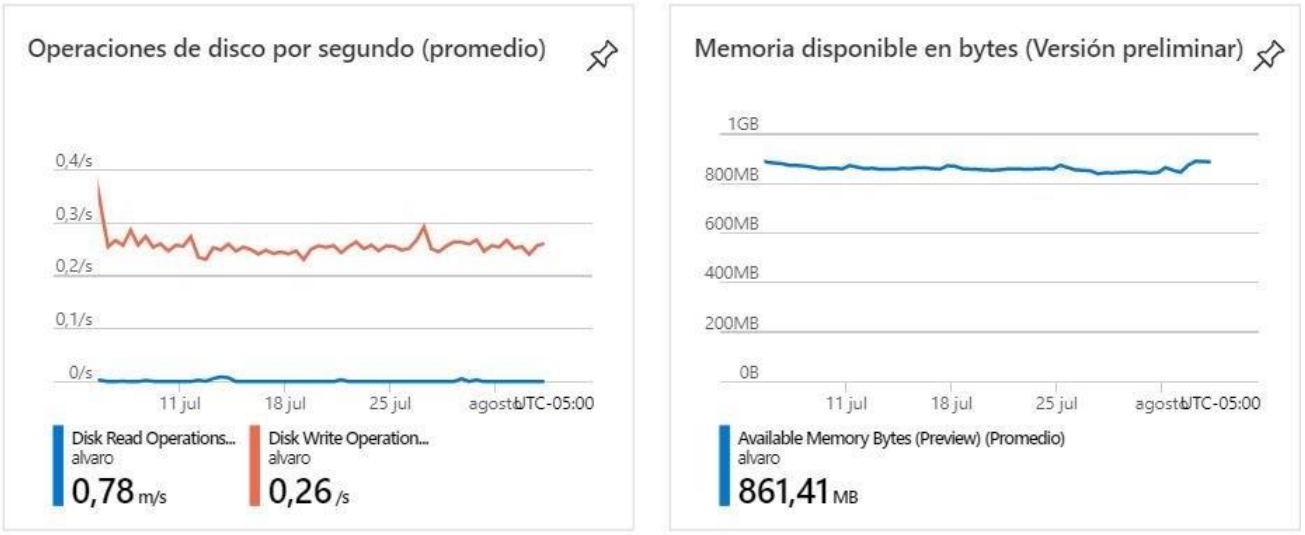


Figura 126. Disco y memoria de Microsoft Azure-Apache servidor 1

En las figuras 125 y 126 indica la manera en la que está trabajando el servidor Apache en la plataforma de Microsoft Azure mientras está en funcionamiento o activado.

- El porcentaje del uso de la CPU fue del 3% debido que se encontró constantemente trabajando.
- El tráfico de red y la cantidad de almacenamiento que usa, dando con valor superior a 120MB.
- Los bytes del disco se encuentran ocupando más de los 400MB.
- Las operaciones del disco por segundo, el disco de escritura corresponde a un valor superior al 0,3/s, es decir que el nivel las operaciones que mide el disco esta normal.
- Finalmente, la memoria disponible en bytes supera los 800MB de 1GB.

4.1.3.6.4. Como segundo servidor a analizar esta Microsoft IIS “Internet Information Service”

Este segundo servidor también conocido como Microsoft IIS es uno de los servidores más utilizados por parte de Microsoft, para la creación de este servidor web se requirió del servidor del laboratorio de ciberseguridad la cual nos habilito 3 servidores en una virtualización para servidores tanto para Linux y Microsoft y para la utilización de pruebas de penetración el sistema operativo Kali Linux. Los beneficios que se encontró al momento de su configuración

e instalación fue que al ser un entorno Windows el interfaz no cambiaba mucho a la hora de navegar por las opciones, de igual manera el administrador del servidor y la instancia de IIS para su configuración de roles y características del servidor. En este sentido Microsoft ISS no presento ninguna dificultad a la hora de su configuración, haciendo que sea rápida y sencilla.

Microsoft IIS utiliza un proceso netamente único, es decir que un solo proceso utiliza todas sus peticiones a diferencia de Apache y Nginx que dividen los procesos en diversos subprocesos. Los requisitos que se tomaron en cuenta para su versión Windows Server 2016 fueron de un procesador de 1,4 Ghz de 64bits compatible con instrucciones de x64 con una cantidad mínima de almacenamiento de 512 MB, finalmente dando soporte a varios protocolos tales como: HTTP, HTTPS, FTP, SMPT Y NNTP nativamente soportando ASP.NET. En el aspecto de seguridad utiliza capas de seguridad, es decir métodos de autenticación para la validación de usuarios.

Análisis

De esta manera se procedió a la instalación de Windows Server 2016, con el fin de configurar el servidor web y sus demás instancias, este servidor está montado en un servidor local **172.20.24.251** que forma parte del laboratorio de ciberseguridad, se instaló php en su versión actual, mysql como bases de datos, phpMyAdmin para administrar la bases de datos y un gestor de contenido **"Wordpress"**, cada uno de forma separada descargadas de los sitios oficiales de cada uno de los programas como se mira a continuación los procesos de su instalación.

Máquina de virtualización de todos los servidores configurados en el laboratorio de ciberseguridad.

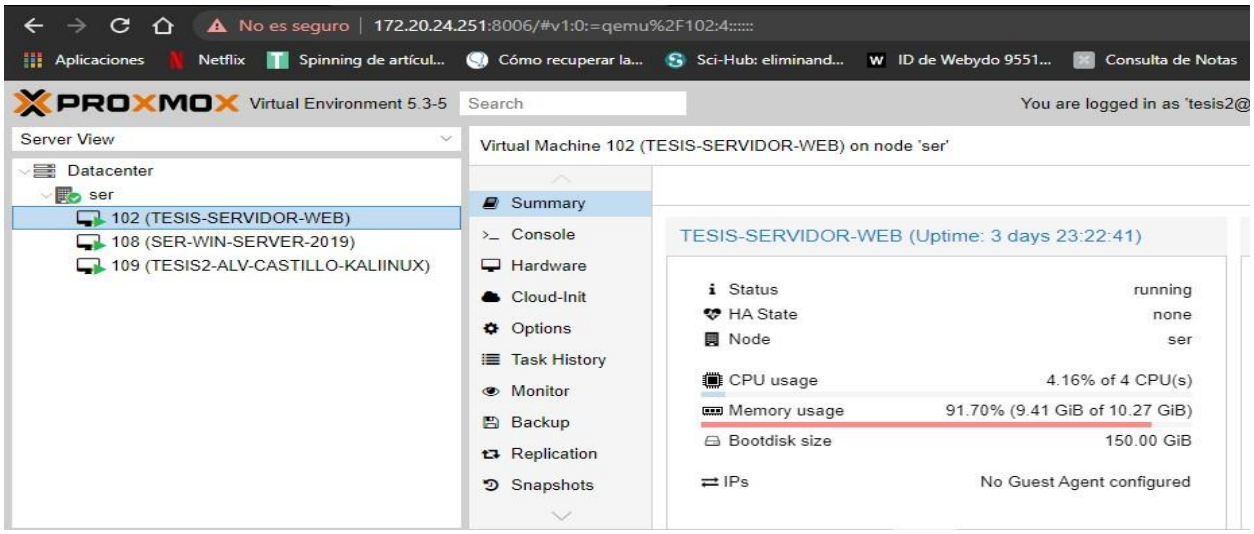


Figura 127. Máquina de virtualización

Instalación de Windows server 2016 desde la visualización de servidores del laboratorio de ciberseguridad.

Figura 128. Instalación de Microsoft IIS servidor 3

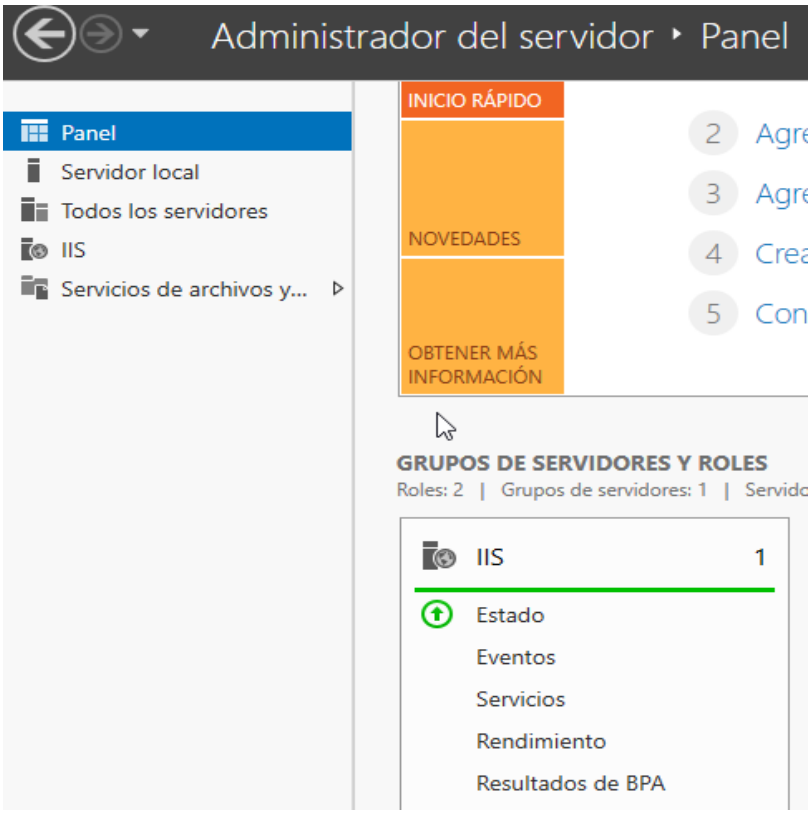


Figura 129. Panel del administrador del servidor 3

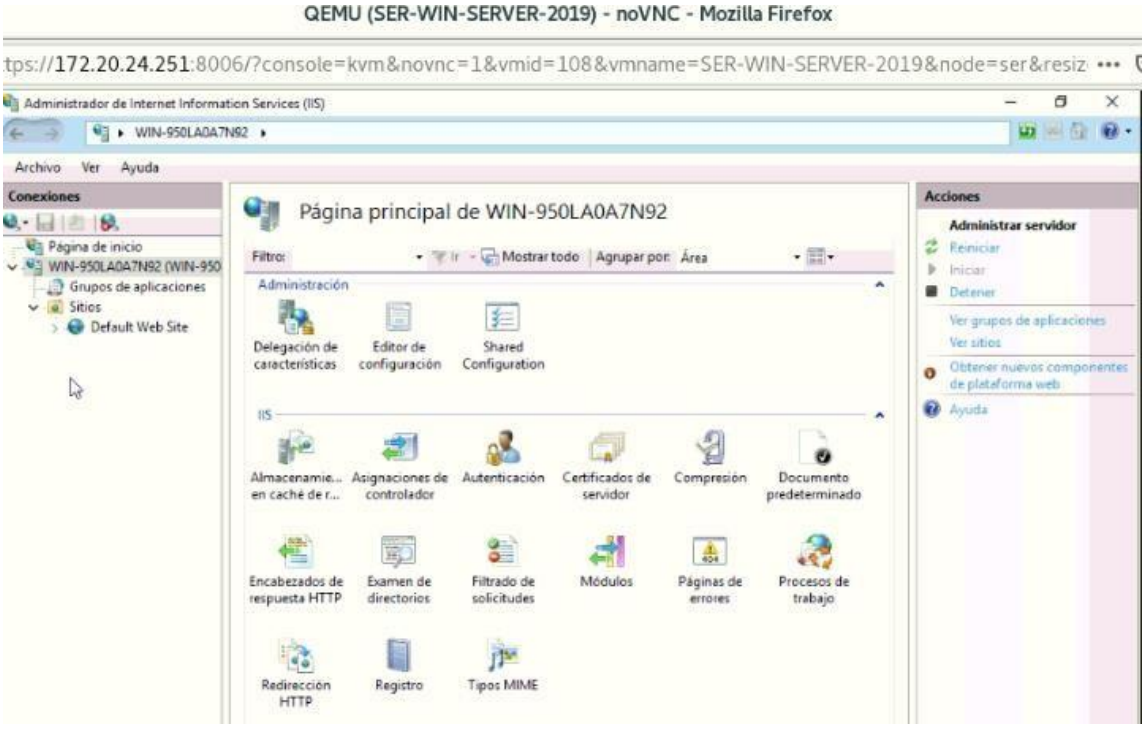


Figura 131. Página principal del servidor web

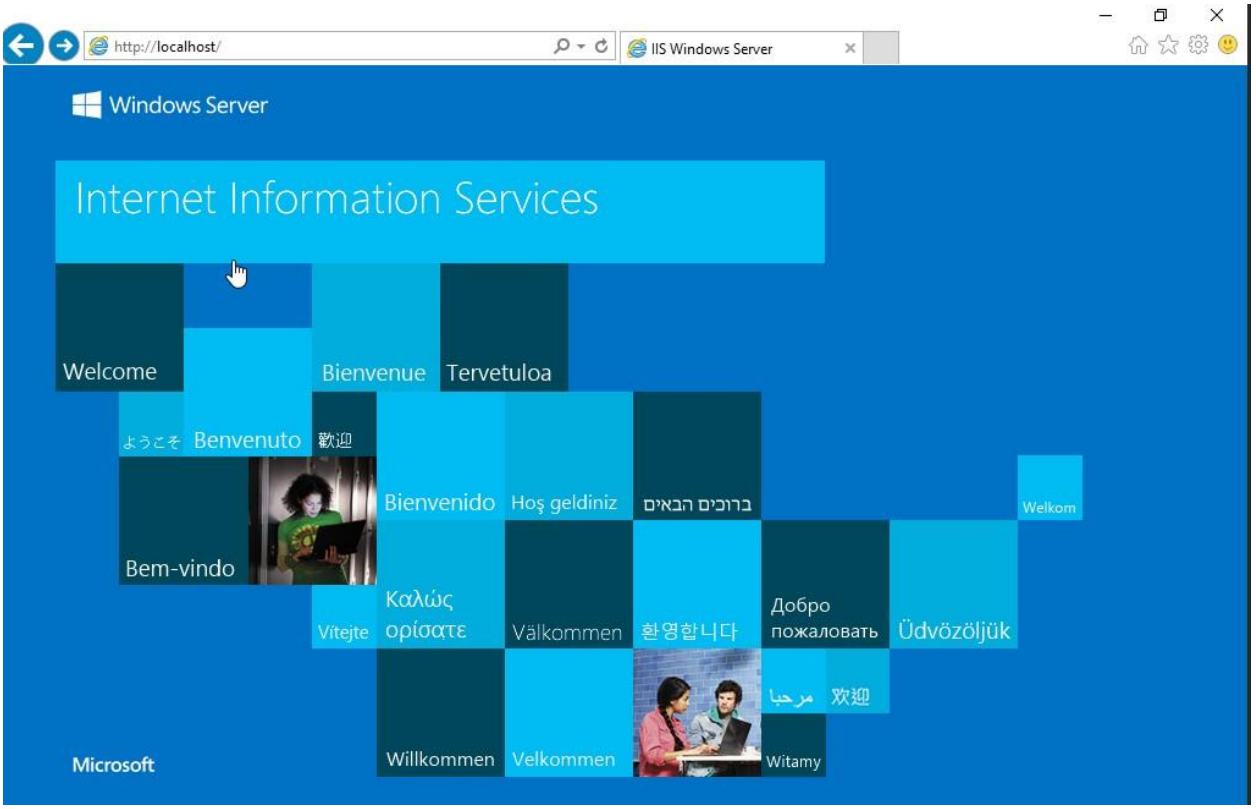


Figura 130. Localhost del servidor Microsoft IIS

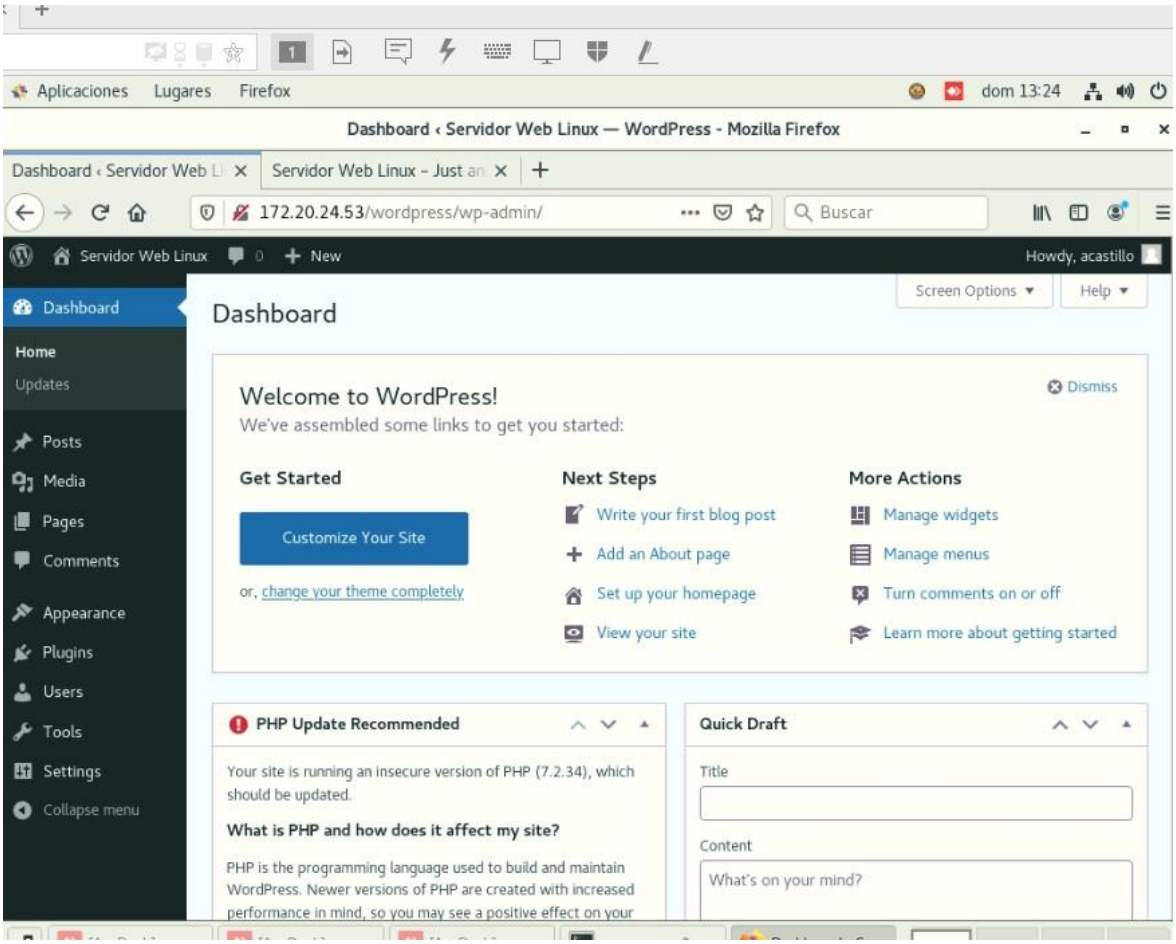


Figura 132. Gestor de contenido Wordpress

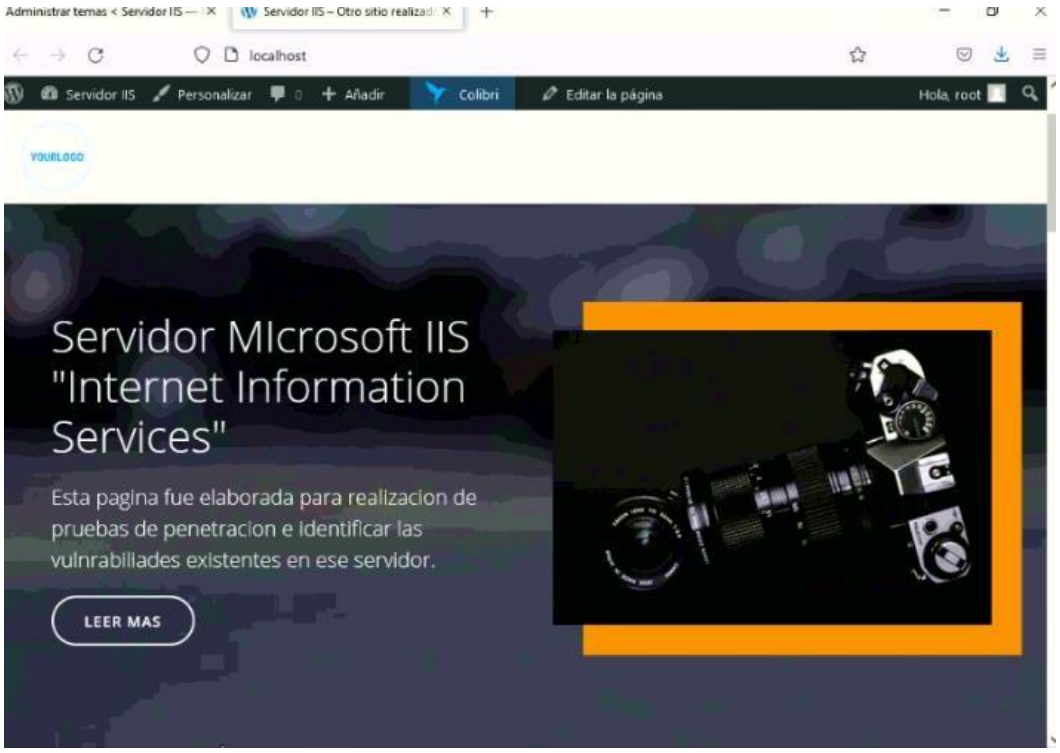


Figura 133. Sitio Web con servidor Microsoft IIS

4.1.3.7. Hardware e historial de tarea de Microsoft IIS.

Para ver el estado y rendimiento del hardware tanto del servidor ISS se tomó referencia de la máquina de virtualización Proxmox. Dada de la siguiente manera:

Figura 134. Hardware de Microsoft ISS

Figura 135. Estado del servidor Microsoft IIS

En la (figura 136) podemos apreciar el funcionamiento o un resumen de la forma que está trabajando el servidor mientras se encuentra en funcionamiento, en este sentido se puede analizar con los demás servidores y medir sus niveles de uso de CPU, tráfico de red, memoria y el disco tanto de escritura como de lectura.

- En la primera figura indica el nivel de uso de la CPU por día en donde es detallado por horas, es representado del 0 al 50, considerando el 50 como el nivel de uso más alto.
- En la segunda figura indica el nivel de tráfico, el indicador azul manifiesta la salida de tráfico, está representado del 500k al 2M, considerado el 2M como el nivel de tráfico más, en este sentido entre más alto sea el tráfico más paquetes o datos están siendo procesados en el servidor.
- En la tercera figura indica el uso de la memoria, en este caso está representado con azul el espacio de la memoria ram que está siendo ocupada, de esta manera se mira que está ocupado 9,26gb de memoria dejando un total de 2,74 de memoria disponible.
- Como ultima figura indica la entrada y salida de disco, de igual manera representada en color azul el disco de escritura y en verde el disco en lectura.

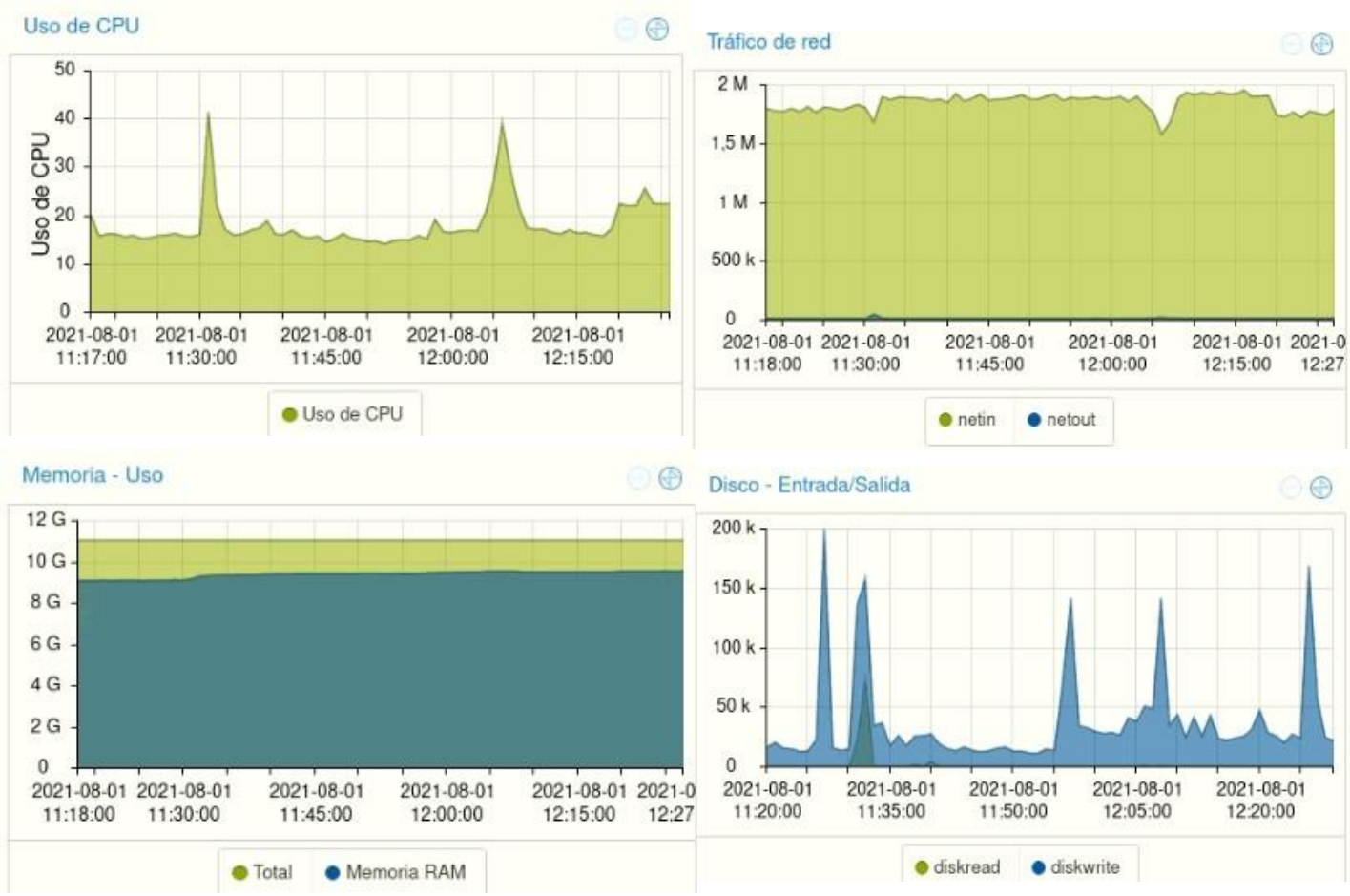


Figura 136. Funcionamiento del servidor Microsoft IIS

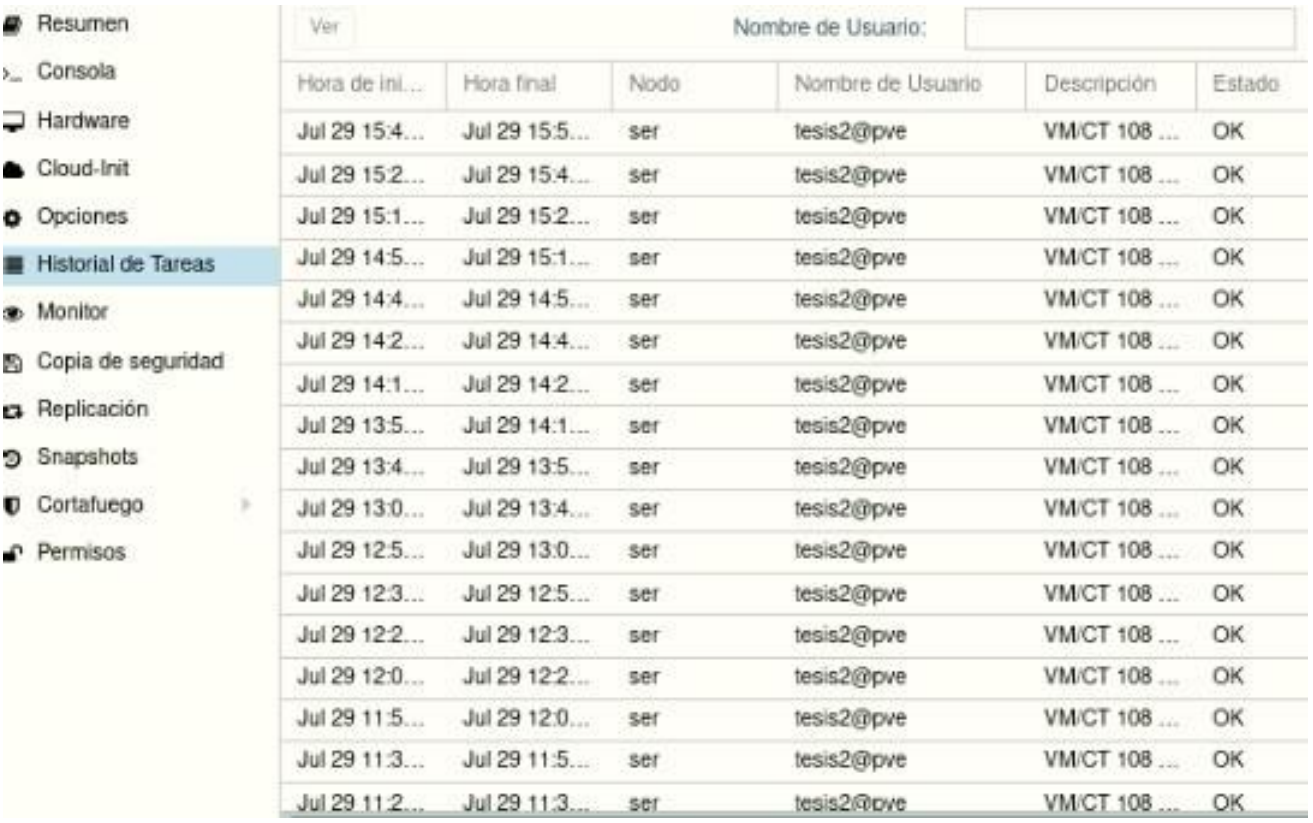


Figura 137. Funcionamiento del servidor Microsoft IIS

4.1.3.7.1. Como tercer servidor a analizar esta Apache en Linux en el laboratorio de ciberseguridad.

Este tercer servidor web configurado en el sistema operativo Linux en el laboratorio de ciberseguridad está alojado de igual manera en una máquina de virtualización, se procedió a instalar el servidor Apache para Linux utilizando comandos en el terminal para su configuración, de esta manera hace el entorno de trabajo sea más complejo a diferencia del entorno gráfico de Microsoft sin embargo la utilización de comandos hace más seguro y menos vulnerable a la hora posibles amenazas. Se manejo un proceso de instalación LAMP de modo que sea sencillo al momento tener configurado el sitio web, como parte de la investigación se evitó hacer cualquier tipo de configuración de seguridad para poder comprobar las vulnerabilidades que se tenía al no contar con dichas protecciones.

Análisis

Para el proceso de configuración e instalación se realizó de la misma manera que el punto 4.2.3.7.1, donde se configuro mediante LAMP, la única diferencia es que no se cuenta con una plataforma de servicio como fue el caso de Microsoft Azure, si no que fue alojado en el

laboratorio de ciberseguridad con una ip local **172.20.24.53**. al no contar con una plataforma que ofrece servicios de seguridad, hace que este servidor creado sea más propenso a las vulnerabilidades y amenazas.

A continuación, se indicará el estado del servidor y demás configuración que se aplicó a la hora de su instalación.

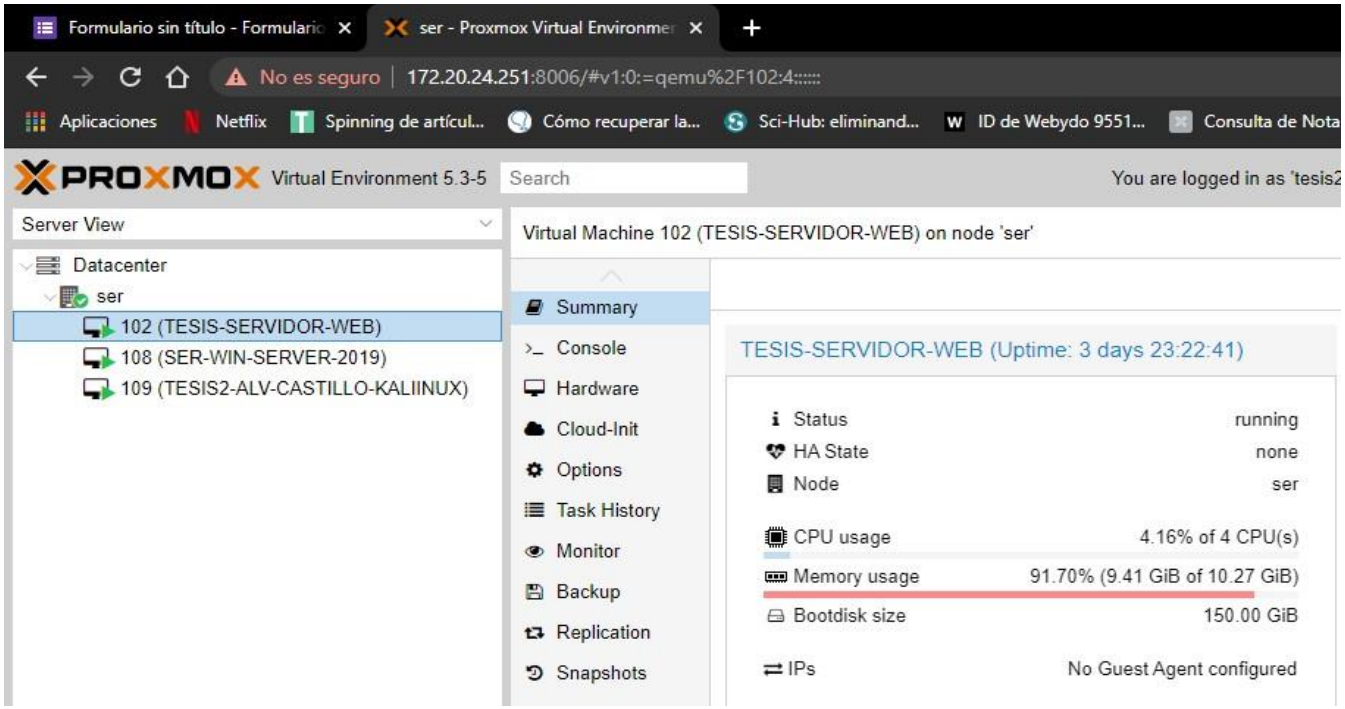


Figura 138. Estado del servidor web Linux.

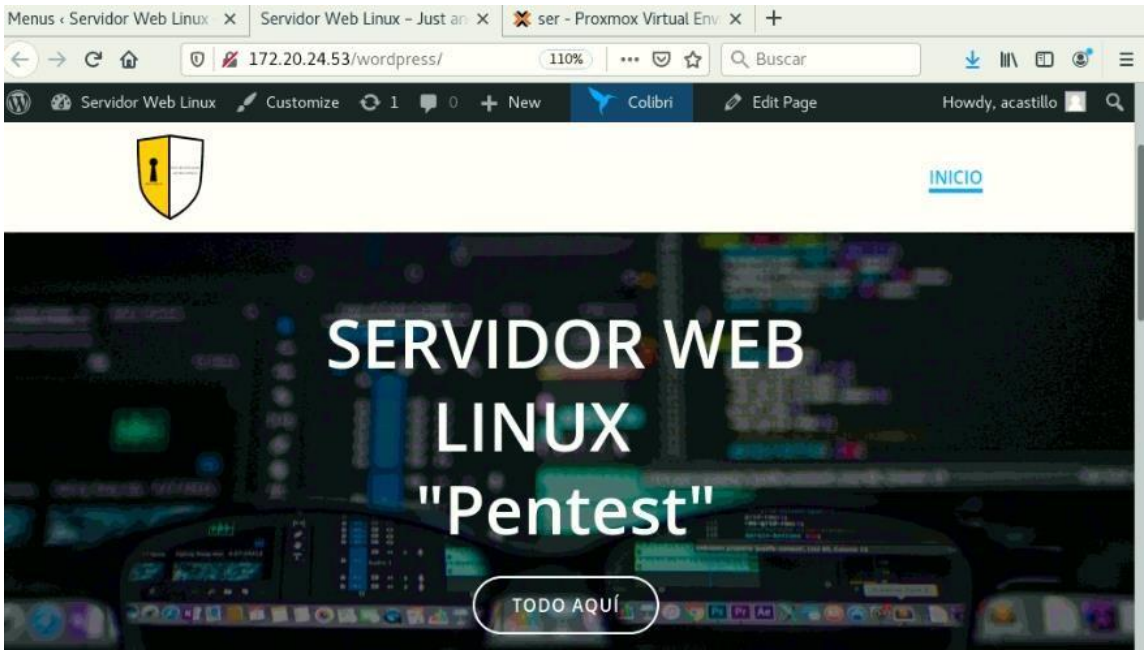


Figura 139. Sitio Web Linux- Laboratorio de ciberseguridad

4.1.3.8. Hardware e historial de tarea de Linux.

Para ver el estado y rendimiento del hardware tanto del servidor Linux se tomó referencia de la máquina de virtualización Proxmox. Dada de la siguiente manera:

Figura 140. Estado del servidor Linux.

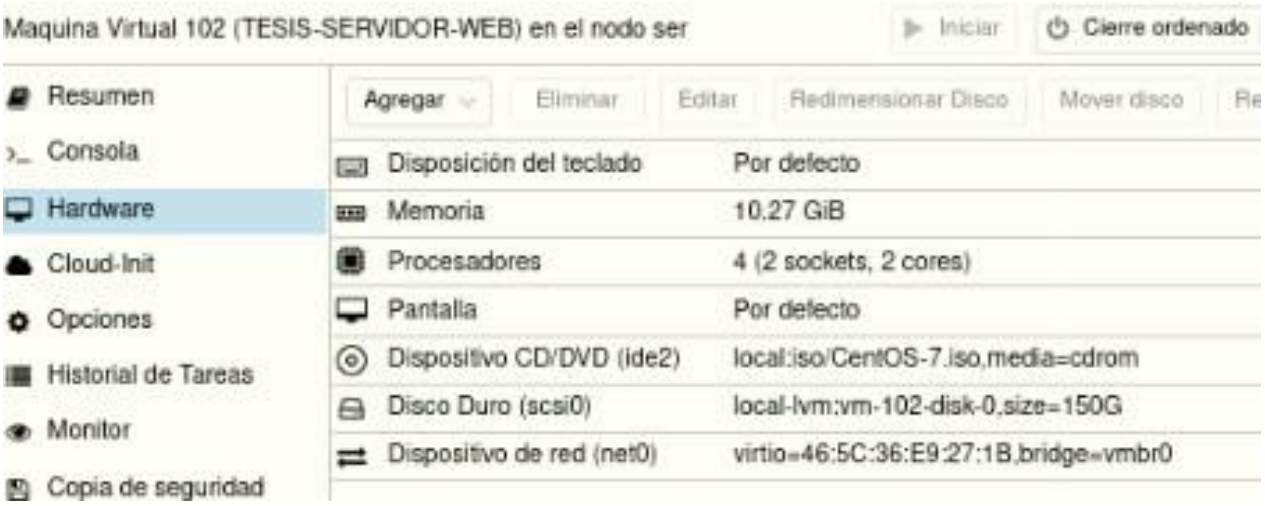


Figura 141. Hardware de Linux.

En la (figura 141) apreciar el funcionamiento o un resumen de la forma que está trabajando el servidor Apache en el sistema operativo Linux mientras se encuentra en funcionamiento.

- En la primera figura indica el nivel de uso de la CPU por día en donde se detalla por horas, de igual manera está representado del 0 al 50, dado en este caso un porcentaje de 15 en el uso de CPU en el servidor Linux.
- En la segunda figura indica el nivel de memoria, de igual manera está representado en color azul el espacio de la memoria ram, en esta figura ocupa 10gb de memoria dejando un total de 2 de memoria disponible.
- En la tercera figura indica el nivel de tráfico de red, el indicador verde manifiesta la entrada de tráfico, de esta manera es representado por 500k al 2M tomando en cuenta que el 2M es el más alto. En esta figura indica el 17 de entrada de tráfico de red.
- Como ultima figura indica la entrada y salida de disco, la última lectura de escritura muestra 200kps la velocidad en la que el tráfico recorre.

Figura 142. Funcionamiento del servidor Linux.

Figura 143. Historial de tareas Linux

Conclusión:

En lo respecta después de haber realizado las configuraciones e instalaciones de cada uno de los servidores tanto Apache como IIS dentro de los sistemas operativos de Linux y Microsoft Server 2016, hasta plataformas virtuales en la nube como es el caso de Microsoft Azure, se llega a la conclusión desde el punto de vista de ingeniería de software, que la diferencia que existía con apache ha desaparecido debido al rediseño que tuvo IIS a su versión 7.0, ahora ambos son modulares escalables y extensibles. La diferencia que aún siguen vigentes es que IIS requiere de un hardware de rango alto, a comparación de Apache que puede comenzar con un hardware muy reducido e ir creciendo conforme la organización lo necesite.

Referente a la seguridad, ambos servidores tienen una alta gama de configuraciones para hacer de sus servidores muy seguros como la empresa u organización lo requiera, pero hay que recalcar que ambos casos esta configuración segura no es parte de la instalación que se tiene por defecto, si no que requiere de conocimientos específicos para conseguir un alto nivel de seguridad.

Finalmente, la elección que se tome a la hora de ver cual servidor es mejor, se tomara en cuenta que función se ejecutaran en el servidor web. En este sentido Apache puede ser una buena opción si el contenido son principalmente páginas estáticas y con una expectativa de número de usuarios alto, de lo contrario IIS es una excelente opción si el servidor ya cuenta con versión de Microsoft Windows server, con una expectativa de número de usuarios medio o si solicitará aplicaciones .NET.

4.1.3.8.1. Cuadro comparativo de los servidores Apache y Microsoft IIS

Es importante conocer las diferencias entre estos dos servidores para analizar cuál de ellos es más eficiente al hora de desarrollar un servidor web (Tabla 31).

Tabla 31. Linux vs Windows

	Windows	Linux
Costos	Costo de licencia	No presenta costo de licencia, sus precios de asistencia dependen de sus distribuciones
Uso estándar	Dispone de interfaz gráfica para el usuario	Uso líneas de comando
Acceso remoto	El cliente tiene que instalarse y configurarse	Recurso integrado (terminal y shell)
Software y características	Sobrelleva programas habituales; posibilidad de utilizar aplicaciones de Microsoft	No ofrece portabilidad, pero si con gran variedad de aplicaciones disponibles
Soporte de hardware	Diseñado solo para sistemas Windows	Pueden ser utilizados los controladores de hardware para Linux
Seguridad	Tiene un elevado potencial de errores de usuario; interfaz integrada con posibles puntos de ataque	Los usuarios no tienen acceso para realizar ajustes básicos del sistema, las vulnerabilidades encontradas se solucionan con rapidez.
Asistencia	Asistencia para todas sus versiones	Su asistencia varía en función de la versión
Documentación	El sistema y sus aplicaciones son bien estructuradas, algo suspende de componentes de la API y de formatos de datos	Se da a conocer el código fuente completo, las API, las bibliotecas y aplicaciones, la mayoría de su información está en ingles

Ventajas	-Confiable -Seguro -Administrable - Buen soporte técnico	-Altamente configurable -Estabilidad -Independencia de la plataforma -Código abierto
Desventajas	No es multiplataforma y solamente funciona con Windows	Su complejidad puede resultad complejo incluso para tareas básicas si no se tiene conocimiento con líneas de comando.

Fuente. Adaptado de (Ionos, 2021). Digital Guide Ionos

4.1.3.8.2. Evaluación de las características

El resultado de esta características se realizará con la asignación de pesos de la siguiente forma:

Pesos

Si = 1 No= 0

Alta =Si Media y Bajo= No

Tabla 32. Evaluación de las características

Característica	Apache	Puntuación	IIS	Puntuación
Sistema Operativo	Multiplataforma (Windows, Unix, Linux, Solaris)	Si	Microsoft Windows Server 2012	No
Portabilidad	Alta (funciona en gran variedad de procesadores/sistemas operativos en todas las gamas)	Si	Baja (servidores de gama media y alta con Microsoft Server 2016	No
Modularidad	Alta (82 módulos oficiales	Si	Media (poco más de 40 módulos disponibles	No

Escalabilidad	Alta (se puede migrar a plataformas con mejores prestaciones)	Si	Media (se puede migrar a servidores con mejores prestaciones pero solo con Windows Server 2016)	No
Seguridad	Alta (si se instalan y configurar de manera correcta los módulos de seguridad)	Si	Alta (si se instalan y configuran correctamente los módulos de seguridad	Si
Soporte	Medio (comunidad de usuarios y red no oficial de empresas)	No	Alta (directa del desarrollador y red oficial de asociados	Si
Costo	Ninguno (descargable gratica de internet)	Si	Ninguno (incluido como parte del sistema operativo)	Si
Soporte 2	Medio (2 de los 5 EMS más utilizados)	No	Medio (1 de los 5 más utilizados)	No
Herramientas para el desarrollo de contenidos	Alta (Python, php, perl, ruby, Rex, .net parcial y mono)	Si	Medio (.Net, php y java)	No
Código Abierto		Si		No
	Porcentaje	8	**Porcentaje**	3

Fuente. Adaptado de (Morales, Sánchez y Barrera, s.f.). Análisis comparativo de servidores Apache vs Microsoft IIS. Universidad Autónoma del Carmen.

10 características -> 100 %

10 x = 100%

8 x = 80%

10 -> 80%

10 x = 100%

3 x = 30%

10-> 30%

Tabla 33. Valor alcanzado

Valor Alcanzado	
Apache	80%
IIS	30%

4.1.3.9. Test de resultados

Tabla 34. Parámetros de evaluación explotación

Parámetros de evaluación explotación	
Difícil	1
Media	2
Fácil	3

Tabla 35. Parámetros de evaluación Prevalencia

Parámetros de evaluación Prevalencia	
Poco común	1
Común	2
Difundida	3
Muy difundida	4

Tabla 36. Parámetros de evaluación detección

Parámetros de evaluación detección	
Difícil	1
Media	2
Fácil	3

Tabla 37. Parámetros de evaluación impacto

Parámetros de evaluación impacto	
Menos	1
Moderado	2
Grave	3

4.1.3.9.1. Resultado final test de manejo de configuración y desarrollo N°1

En esta vulnerabilidad presenta una explotación fácil ya que mediante la recolección de información y herramientas como Owasp Zap, Nmap, Acunetix, Curl, y plataformas web como whois, web server tolos, entre otros, se pudo detectar con facilidad vulnerabilidades de todo tipo tanto GET Y POST. Como prevalencia común, porque es aprovechado de herramientas para que atacante pueda predecir todos los detalles para iniciar su ataque, y ese sentido el impacto es moderado ya que al contar con un buen manejo de hackeo podría provocar grandes daños al servidor web como al sitio.

Tabla 38. Valoración de la vulnerabilidad de manejo de configuración y desarrollo.

Manejo de configuración y desarrollo		
Explotación	Fácil	3
Prevalencia	Común	2
Detección	Fácil	3
Impacto	Moderado	2

4.1.3.9.2. Resultado final test de manejo de identidad N°2

En esta vulnerabilidad la explotación es fácil ya que la metodología Owasp propone una serie de paso en la que verifica el correcto uso de seguridad, en ese sentido la detección la convierte en fácil ya que mediante los pasos puede saber si existe o no vulnerabilidad. Cabe recalcar que el impacto es grave ya que al cumplir con esta serie de pasos y detectar los errores, el atacante podría aprovechar de esos servicios que son mal utilizados, configurados y sobre todo el desconocimiento provocaría el acceso directo de un individuo externo que quiera hacer daño a la identidad de este.

Tabla 39. Valoración de la vulnerabilidad: Manejo de identidad

Manejo de identidad		
Explotación	Fácil	3
Prevalencia	Común	2
Detección	Fácil	3
Impacto	Grave	3

4.1.3.9.3. Resultado final test de fuerza bruta N°3

Esta vulnerabilidad presenta explotación media porque la violación a la seguridad al servidor web no fue tan fácil, ya que se tuvo que esperar alrededor de 6 horas para encontrar la contraseña de acceso a panel de administración de WordPress, como prevalencia difundida ya que al momento de generar un directorio de contraseñas son alrededor un millón de contraseñas que genera el propio sistemas, de esa manera lo convierte complicado y único para el sistema. El impacto se considera grave ya que una vez localizado las credenciales tendría el total acceso a la información y al panel de administración, es decir robo y perdida de información sensible.

Tabla 40. Valoración de la vulnerabilidad de fuerza bruta

Fuerza bruta		
Explotación	Media	2
Prevalencia	Difundida	3
Detección	Media	2
Impacto	Grave	3

4.1.3.9.4. Resultado final test en Owasp/ vulnerabilidad a directorios N°4

En esta vulnerabilidad la explotación es fácil ya que el acceso a los directorios median la herramienta como Nmap y Owasp son muy fáciles a la hora de localizar puertos abiertos y direcciones que son dirigidos directamente a la vulnerabilidad, como prevalencia común ya que son fácil de descubrir mediante herramientas de vulnerabilidad, al igual que la detección son puntos que se detectan fácilmente, como impacto moderado ya que dicha vulnerabilidad puede comprometer la información de cada uno de los directorios, al menos de que las información se escasa resultando ineficiente la búsqueda para el atacante.

Tabla 41. Valoración de las vulnerabilidades Owasp/directorios

Owasp/ vulnerabilidad a directorios		
Explotación	Fácil	3
Prevalencia	Común	2
Detección	Media	2
Impacto	Moderado	2

4.1.3.9.5. Resultado final test de Cross Site Scripting N°5

Esta vulnerabilidad tiene la explotación fácil porque el atacante envía códigos maliciosos a un buscador de la página para indicar la existencia de una alerta XXS, las fallas de inyección tienen una prevalencia común ya que son sencillas de examinarlas ya sea con herramientas de pentest ayudando así a descubrir las fallas, finalmente el impacto es grave ya que por media de una inyección XXS podría provocar una pérdida de información, falta de integridad, confidencialidad y disponibilidad de los datos, llevando así a un control completo del servidor que se encuentra alojada el sitio web.

Tabla 42. Valoración de la vulnerabilidad de Cross Site Scripting

Cross Site Scripting		
Explotación	Fácil	3
Prevalencia	Común	2
Detección	Media	2
Impacto	Grave	3

4.1.3.9.6. Resultado final test de Inyección SQL N°6

Esta vulnerabilidad tiene la explotación difícil porque normalmente el atacante utiliza bastante está prueba de inyección SQL con el fin de acceder directamente a la base de datos del servidor web, su prevalencia es común debido a su mala configuración de seguridad provocando la visibilidad de estas vulnerabilidades, presenta detección media ya que al ser una vulnerabilidad común el navegador presenta esas debilidades y son fáciles de detectar, pero el impacto es grave ya que pone en riesgo todos los datos que se encuentran protegidos en las bases de datos.

Tabla 43. Valoración de la vulnerabilidad de Inyección SQL

Inyección SQL		
Explotación	Difícil	1
Prevalencia	Común	2
Detección	Media	2
Impacto	Grave	3

4.1.3.9.7. Resultado final test de ataques (DOS) N°7

En esta vulnerabilidad presenta una explotación muy difícil ya que al momento y interceptar con el sitio se debe tener un conocimiento de la aplicación de buffer que otorgaran el acceso a la denegación de servicios a la página web y por ende dejarla sin funcionamiento, la prevalencia poco común ya que al momento de ejecutar Dos se debe conocer en detalle algunos de los parámetros de vulnerabilidad que fueron detectados como es el caso de nombres de servidor, dominios, contraseñas, entre otros, finalmente impacto grave ya que al ser ejecutado con éxito el servidor se dañará dejando de funcionar servicios, provocando un cierre del sitios web.

Tabla 44. Valoración de la vulnerabilidad de DoS

DoS		
Explotación	Difícil	1
Prevalencia	Común	2
Detección	Media	2
Impacto	Grave	3

4.1.3.9.8. Resultado final test método Http N°8

En esta vulnerabilidad la explotación al igual que la detección son fáciles ya que mediante Owasp y comando curl se pudieron vulnerar cabeceras del sitio web poniendo en riesgo la información a la instalación del servidor web, de la misma manera prevalencia común, ya que son fáciles de descubrir si no cuentan con una configuración profunda en el servidor web.

Tabla 45. Valoración de la vulnerabilidad de Http

Método Http		
Explotación	Fácil	3
Prevalencia	Común	2
Detección	Fácil	3
Impacto	Grave	3

4.1.4. Resultado de los riesgos de la comparación de la vulnerabidades.

Tabla 46. Resultado final de las vulnerabilidades

Resultado Final de las vulnerabilidades																
CRITERIOS	Test 1		Test 2		Test 3		Test 4		Test 5		Test 6		Test 7		Test 8	
	Resultado	Valor	Resultado	Valor	Resultado	Valor	Resultado	Valor	Resultado	Valor	Resultado	Valor	Resultado	Valor	Resultado	Valor
Explotación	Fácil	3	Fácil	3	Media	2	Fácil	3	Fácil	3	Difícil	1	Difícil	1	Fácil	3
Prevalencia	Común	2	Común	2	Difundida	3	Común	2	Común	2	Común	2	Común	2	Común	2
Detección	Fácil	3	Fácil	3	Media	2	Media	2	Media	2	Media	2	Media	2	Fácil	3
Impacto	Moderado	2	Grave	3	Grave	3	Moderado	2	Grave	3	Grave	3	Grave	3	Grave	3
		5,33		8,00		7,00		4,67		7,00		5,00		5,00		8,00

Formula : Valor de Riesgo = Promedio (explotabilidad + prevalencia + detección) * impacto

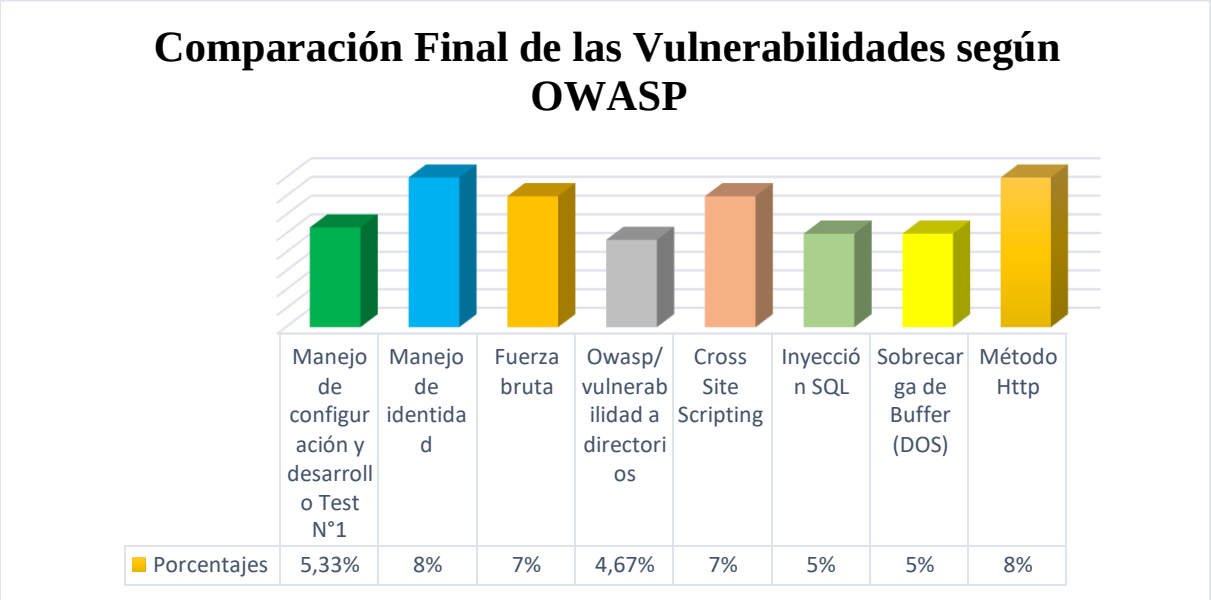


Figura 144. Resultado final de las vulnerabilidades

De esta manera en base a los indicadores realizados y de acuerdo con los parámetros planteados se puede visualizar que las vulnerabilidades de: manejo de identidad, fuerza bruta, Cross Site Script y los métodos Https tienen un índice superior al 7% que representan mayor riesgo al tratarse de vulnerar los servidores web que fueron indagados.

De esta manera se procede a evaluar los aspectos de seguridad mediante estos indicadores cumpliendo con cada uno de los niveles de riesgo.

Tabla 47. Checklist de verificación de seguridad informática

SEGURIDAD DE LOS DATOS						
		Cumple		Riesgo		
Indicador	Aspectos a evaluar	SI	NO	Bajo	Medio	Alto
1	Los servidores web tienen definidas políticas de seguridad		X		X	
2	Estas políticas de seguridad son revisadas	X			X	

	periódicamente en los servidores web					
3	Dispone de alguna metodología de seguridad a los servidores web	X		X		
4	Se monitoriza diariamente el estadode cada uno de los servidores	X		X		
5	Se tiene implementando dominios a los servidores web		X		X	
6	Se tiene instalado antivirus licenciado en los servidores web		X		X	
7	Realiza pruebas para detectar vulnerabilidades en los servidores	X		X		

SEGURIDAD DE LA INFRAESTRUCTURA A LOS SERVIDORES WEB

		Cumple		Riesgo		
Indicador	Aspectos a evaluar	SI	NO	Bajo	Medio	Alto
8	Dispone de firewall		X			X
9	Dispone de un sistema de protección anti-DDOS	X		X		
10	Dispone de comandos que ayuden al mejoramiento de la seguridad a los servidores web	X		X		

11	Dispone de certificación SSL	X		X		

CONTROLES DE ACCESO Y SEGURIDAD A LOS SERVIDORES WEB

		Cumple		Riesgo		
Indicador	Aspectos a evaluar	SI	NO	Bajo	Medio	Alto
12	Dispone de contraseñas seguras al momento de acceder a los paneles del administrador de los servidores web	X			X	
13	Dispone de proxy que mitiguen riegos de vulnerabilidad al servidor web		X		X	

HABITOS SEGUROS Y PREPARACIÓN A LOS SERVIDORES WEB

		Cumple		Riesgo		
Indicador	Aspectos a evaluar	SI	NO	Bajo	Medio	Alto
14	Desarrolla sistemas de capacitación al personal referente a la seguridad informática		X		X	
15	La actitud referente a la cuidado de normas de seguridad es positiva	X			X	

Tabla 48. Escala de cumplimiento de riesgos

	Riego		
	Alto	Medio	Bajo
No cumplen: 6	1	5	-
Si cumplen: 9	-	3	6

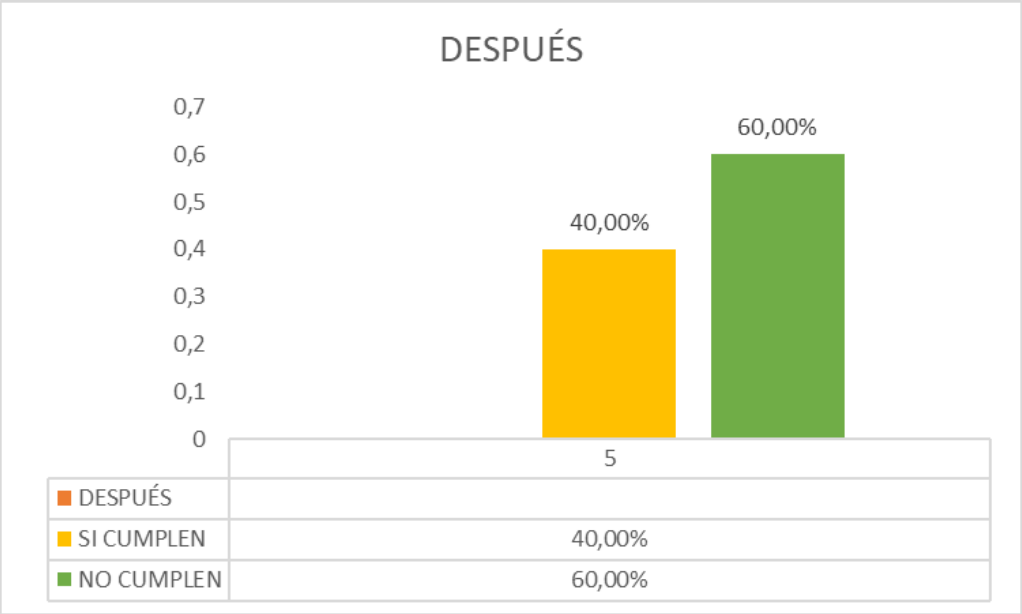


Figura 145. Resultado del cumplimiento de riesgos

De esta manera se puede observar el aumento del 60% en lo respecta a la seguridad al servidor web cumplimento con las fases y procesos óptimos que se vinieron realizando durante esta investigación.

4.1.5. Herramientas para la detección de vulnerabilidades

Tabla 49. Herramientas Pentest

Herramienta	Descripción	Sitio Oficial
Nmap	Herramienta propia de Kali Linux para la exploración de la red.	https://nmap.org/
Acunetix	Herramienta para la realización de pruebas de seguridad, verifica vulnerabilidades como inyecciones SQL	https://www.acunetix.com/
Hardening	Es una medida de seguridad que disminuye el nivel de vulnerabilidad al servidor web	
Fail2ban	Es una aplicación que previene el paso de intrusos a sitios y servidores web.	
Owasp Zap	Es una herramienta propia del proyecto Owasp, realiza pruebas de pentest aplicaciones web.	https://owasp.org/www-project-zap/

Maltego	Es un framework que encuentra información sobre personas, direcciones, empresas, números telefónicos, entre otros.	https://www.maltego.com/
Whois	Es un protocolo utilizado en los comandos de Kali Linux para ejecutar consultas en las base de datos identificando nombres de dominio, dirección ip, entre otros.	
Nessus	Es un programa que realiza escaneo de vulnerabilidades, muestra informes sobre el estado de vulnerabilidad.	https://es-la.tenable.com/products/nessus
WebServer Stress Tool	Herramienta para pruebas de cliente/servidor http, para verificar problemas críticos de rendimiento al servidor web	https://www.paessler.com/tools/webstress
WPScan	Es un escanear de vulnerabilidades propiamente de Wordpress, para comprobar las vulnerabilidades plugins, archivos, temas, entre otros.	https://wpscan.com/wordpress-security-scanner
Nikto	Es un escanear de vulnerabilidades hacia líneas de comando de los servidores web.	https://cirt.net/Nikto2
Burpsuite	Es una programa que permite analizar cómo se encuentra la seguridad en las aplicaciones web, escaneó de intrusos al servidor, entre otros.	https://portswigger.net/burp

4.1.6. Propuesta que responda los levantamientos de procesos de seguridad del servidor web.

Tomando en cuenta las diferentes estrategias, procesos, fases, normas, leyes, herramientas, metodologías y varios conceptos que existen hoy en día para mitigar los procesos de seguridad al servidor web. A continuación se dará a conocer una serie de procesos que brinden a los servicios de aplicaciones web una mejora en la seguridad.

- Como primera petición y de manera objetiva se debe realizar los respectivos cambios a las configuraciones que se encuentran en el servidor por predeterminado, con el fin de tener un control de los servicios, Hsts, Header, http y la información que debe ir visible para el usuarios y otras no.
- Manejar balances de carga independiente para distribuir las peticiones de que se hacen hacia las aplicaciones de una manera más eficiente, ya que al hacerlo directamente generaría un proceso innecesario para el servidor físico.
- Instalar y configurar correctamente SSL que encripten la comunicación entre el servidor y navegador.
- Actualizar de manera periódica prefijos por defecto en las bases de datos, es decir un cambio de al menos 3 veces al mes de contraseñas de acceso.
- Muy importante configurar los permisos de CHOMD a las carpetas y archivos que se encuentra almacenados en el servidor, con el fin de evitar robos a información sensible del sistema.
- Realizar backups periódicamente de al menos 1 vez a la semana, como también manejar versiones actuales de módulos. De esta manera poder contar con certificación de seguridad al sitio web.

4.2. RESULTADOS

En este capítulo se realiza los resultado de la investigación, cumpliendo con los objetivos y las variables que fueron estudiadas.

4.2.1. Resultado de la Variable Independiente

La variable independiente hace referencia a la seguridad al servidor web cumpliendo con los objetivos propuestos, la fundamentación teórica fue esencial para determinar las vulnerabilidades que contiene un servidor web y está definida en el marco teórico del proyecto. Otro resultado obtenido de la variable independiente fue el análisis de las herramientas de pentest. El cual se realizó una comparativa de los servidores web y el rendimiento de cada una ellas siendo los principales instrumentos para el desarrollo de la investigación.

Elección de servicios para la ejecución de las vulnerabilidades detectadas dentro del laboratorio de ciberseguridad.

Uno de los servicios necesarios para la ejecución de las pruebas de penetración fueron los servidores web, posteriormente de un análisis e información se seleccionó al servidor Apache y Microsoft IIS, ya que de los servidores más utilizados y competitivos a nivel global fueron de vital utilidad para la realización de configuraciones, análisis y detección de amenazas. Además de contar con la adaptación de sistemas operativos Linux y Microsoft Server por ser escalables y eficientes en sus procesos.

Para la elección de los servidores se optó por utilizar e instalar software libre para evitar realizar un gasto innecesario, además de que estos servidores cuentan con muchas funcionalidades y configuraciones, actualmente son utilizados por su estabilidad, seguridad, rendimiento, entre otros componentes. Durante el proceso de análisis de los servidores se llegó a establecer de esta manera una comparativa de sus características quedando como mejor servidor "Apache" el cual obtuvo la mejor calificación en las pruebas realizadas en cuanto a su escalabilidad, seguridad, soporte, entre otros.

El servicio que se utilizó como gestor de contenido fue Wordpress para la ejecución del diagnóstico de vulnerabilidades, ya que me permitió hacer uso de la herramienta de WPScan para la detección de amenazas y encontrar información que me permita vulnerar su sistema.

Las herramientas utilizadas para analizar los problemas de seguridad en el servidores web son Owasp Zap, Nmap, Nikto, Maltego, Nessus, WPScan y Nikto ya que permitieron encontrar todo tipo de vulnerabilidad alojada en el sitio web y de esta manera mediante Hardening proteger y evitar ataques a los servidores.

Las herramientas seleccionadas para el desarrollo de la investigación fueron escogidas por su efectividad a la hora de detectar las vulnerabilidades, su alcance, su rendimiento y su adaptabilidad con cualquier tipo de aplicaciones web alojadas en cualquier servidor web.

A continuación se describirá porque han sido escogidas dichas herramientas y técnicas.

Tabla 50. Herramientas seleccionadas para la investigación

Nmap	Nmap permitió a la investigación verificar los puertos que se encontraban abiertos y cerrados independientemente de cada servicio que presenta la aplicación web. Uno de los comando más necesarios fue: **nmap -PN -sT -Sv + dirección sitio web** -sP (sondeo de ping): establece cuantos dispositivos se encuentran activos -P0 (No realiza ping): realiza un escaneo de puertos -Ps (lista de puertos): envía un paquete logrando establecer conexión con la máquina objetivo -PU (lista de puertos): permite observar que dispositivos se encuentra online u offline -PR (ping ARP): realiza este escaneo cuando se detecta una red local
Hardening Fail2ban	Este método permitió a la investigación configurar y desarrollar técnicas para mejorar la seguridad en el servidor web. Los métodos necesarios que se utilizaron fue: Http Trace, Eliminación de ETAG, Clickjacking attack, bloqueo de inyección XXS y X-Content-Type-Options -Mediante técnicas, herramientas y buenas prácticas brindan reducir vulnerabilidades eliminando líneas de ataques. -Función del servidor mejorado: modifica configuraciones incorrectas e incompatibles. -Mejor seguridad: Permite la reducción de amenazas impidiendo la entrada de filtración de datos, ingreso no autorizado y acceso de malware. Esta herramienta permitió en la investigación realizar procesos de baneo al sitio web, como también mediante jaulas poder mitigarlas. En las jaulas de http se desactivan componentes como: apache-nohome y apache-botsearch para evitar baneo o detención al sitio web, como también la configuración en mod_security para la protección al servidor web. -Flexible -Eficaz

	-Previene ejecuciones de bots, scripts, entre otros atacas de servidores -Bloque direcciones IP temporalmente ingresos maliciosos Con esta herramienta fue de total importancia para el desarrollo del proyecto ya que permitió una profunda comprobación de las vulnerabidades existente como también la realización de ataques para verificar el nivel de riesgo que presenta cada una de las amenazas. Alertas encontradas tales como: HTTP Response Header Field(s), X-Content-Type-Options Header Missing, Charset Mismatch, Information Disclosure - Suspicious Comments, Timestamp Disclosure – Unix entre otras más.
Owasp zap	-Gratuito -Código abierto -Realiza pruebas de penetración como SQL, XSS, descubrimiento de ficheros -Ataques de fuerza bruta -Organización sin fines de lucro -Apoyan a la investigación referente a la seguridad -Owasp Top 10 - Inyección -Perdida de autenticación y gestión de sesiones -Secuencia de comando (XXS) -Referencia directa a objetos -Falsificación de peticiones (CSRF) Este software permitió al desarrollo de la investigación para cumplir con le fase de recolección de información requerida por la metodología Owasp.
Maltego	Informacion como: emails, usuarios, servicios, teléfonos, nombres, entre otros datos más. -Extrae informacion de los recursos -Permite formar una conexión con el nombre, correo electrónico, organizaciones, dominios, archivos, entre otros. -Recopila la mayor cantidad de información ante un objetivo

	-Realiza reconocimiento personal -Ayuda a establecer un reconocimiento total del objetivo
Nessus	Esta herramienta alojada en el sistema operativo Kali Linux permitió a mi investigación encontrar vulnerabilidades de fallas en las configuraciones, servidores no actualizados, fallas en los puertos TCP/IP, malware, DoS entre otras más. Permitió escanear vulnerabilidades como son alertas dentro del servidor, como también menciona soluciones de las amenazas encontradas.
Whatweb	Mediante esta herramientas aporte a mi investigación para visualizar las configuración, versiones y tipos de estructura que se componen en un servidor web, como también el lenguaje de desarrollo, su versión y gestor de contenido instalado, con el fin de identificar si presentan o no vulnerabilidades.
WPScan	Mediante esta herramienta permitió a la investigación a escanear la página con el gesto de contenido: Wordpress. Cumpliendo de manera óptima con el escaneo de temas, plugins, usuarios, y hasta contraseñas generadas para la obtención del ingreso a la base de datos. Comando utilizados como: wpscan --url http://172.20.24.53:8080/wordpress --enumerate u wpscan --url http://172.20.2.12 --enumerate vt wpscan --url http://172.20.24.53/wordpress -P /usr/share/wordlist/numbers-as-words.txt -U acastillo.
Nikto	La utilización de Nikto permitió a mi investigación escanear fiches peligrosos en los sitios web, es decir ficheros y directorios que se encontraban visibles para el atacante.
WebServer Stress Tool	Este software permitió a mi investigación comparar con el servidor Apache y el servidor Microsoft IIS la cantidad de usuarios que pueden soportar si ingresan al mismo tiempo.

4.2.2. Resultados de la Variable Dependiente

Para la variable dependiente definida como vulnerabilidad al sistema se tomó en cuenta el objetivo de cumplir cabalmente la metodología Owasp que mide aspectos basados en la detección de vulnerabilidades, identificación de amenazas con riesgos altos y el mejoramiento de la seguridad con técnicas de configuración al servidor web. Para la medición se utilizó la entrevista (Anexo 3) y el resultado de los riesgos encontrados (tabla 41). A continuación, se detalla los resultados.

Detección de vulnerabilidades

Una vez realizadas las pruebas a través del sistema operativo Kali Linux se determinó, la existencia de vulnerabilidades tanto de nivel alto, medio y bajo mediante la herramienta Owasp Zap.

Para el servidor 1

Application Error Disclosure
Directory Browsing
X-Frame-Options Header Not Set
Absence of Anti-CSRF Tokens
Application Error Disclosure
Information Disclosure - Debug Error Messages
Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)
X-Content-Type-Options Header Missing
Charset Mismatch
Information Disclosure - Suspicious Comments
Timestamp Disclosure - Unix

Para el servidor 2

Directory Browsing
X-Frame-Options Header Not Set
Absence of Anti-CSRF Tokens
Application Error Disclosure
Private IP Disclosure

Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)
X-Content-Type-Options Header Missing
Charset Mismatch (2)
Content-Type Header Missing
Information Disclosure - Suspicious Comments
Timestamp Disclosure – Unix
WSDL file passive scanner
Application error Disclosure (1694)

Se comprueba que el uso de herramientas de pentest permite encontrar alertas que pueden ser corregidas a tiempo y evitar que atacantes incidan en ello.

Identificar amenazas con riesgos altos

Terminadas la pruebas de pentest se procedió a realizar un test de resultados para detectar los riesgo más alto en los servidores mediante parámetros de evaluación, además de elaborar un cuadro final de resultados de las vulnerabilidades encontradas con criterios de explotación, prevalencia, detección e impacto.

Se comprobó que el uso de la metodología Owasp ayudo a encontrar los riesgos más altos vigentes en el servidor web cumplimiento matemáticamente el valor de riesgo establecer sus porcentajes.

Mejorar la seguridad con técnicas de configuración

Durante la investigación, nos encontramos en la necesidad de corregir la seguridad en el servidor y la aplicación web, de esta manera se hizo usó de Hardening para la reducción de vulnerabilidades siendo comprobada mediante plataformas securityheaders.com y webpagetest.org, que se encargan de medir la seguridad que se encuentra el servidor web y sus encabezados de protección.
Se identifico las siguiente vulnerabilidades, las cuales se las soluciono, reduciendo el riesgo de amenazas de atacantes que puedan alterar la información.

- Encabezado X-Frame-Options no determinado
- XSS-Protección de Web no habilitado.
- Cabecera de opciones de Tipo X-Content faltante.

De esta manera, se comprobó que hubo una mejora en los encabezados de protección y un aumento de seguridad del 60% de acuerdo con los indicadores evaluados en la escala de riesgo, cumplimento con el objetivo general y específicos de las variables, en este sentido se puedo concluir de manera óptima la satisfacción por el documento que garantiza dicha información.

4.3. DISCUSIÓN

Para la discusión se estableció como base el objetivo de la investigación que es el diagnosticar los problemas de seguridad al servidor web del laboratorio de ciberseguridad, a través de pruebas de penetración, disminuyendo la vulnerabilidad al sistema en la Universidad Politécnica Estatal del Carchi partiendo de la recolección de información para formar un marco teórico y metodológico que sea aprovechado como referencia para el diagnóstico y análisis de vulnerabilidad que presentaron los servidores web y de igual manera identificarlos y mitigarlos mediante herramientas y procesos que ayudaron a mejorar su seguridad.

La aplicación de un método cualitativo permitió realizar una entrevista al director y encargado del laboratorio de ciberseguridad, con su ejecución se pudo identificar el manejo a la seguridad, metodología, vulnerabilidades, representantes, procesos de análisis de vulnerabilidad, el tiempo en el que realizan un proceso de seguridad, indicadores de riegos o amenazas, por otro lado, se analizó los inconvenientes presentados a la hora de ejecutar herramientas de seguridad, alternativas de mejora y técnicas que mejoren el nivel de seguridad. De igual manera se aplicó observación no estructurada de donde se consiguió varias fotográficas a los servidores proporcionados, las vulnerabilidades capturadas por programas que ayudaron a encontrar riegos y amenazas, estos datos fueron de gran utilidad en el cumplimiento del objetivo del proyecto.

La meta principal de esta investigación fue el diagnosticar los problemas de seguridad presentados hacia los servidores web que se logró con la aplicación de la metodología Owasp (Proyecto de seguridad de aplicaciones web abiertas), que permitió recolectar los información con la utilización de pruebas y herramientas tales como: Owasp Zap, Nmap, SQLmap, Nessus, Acunetix, Hardening, Fail2ban, Nikto y WPScan, a partir de los cuales se manejó la configuración y desarrollo, manejo de identidad, validación de entradas, conjunto de actividades para reforzar la seguridad al servidor (Hardening) concluyendo con una fase final de los resultados conseguidos en el análisis de vulnerabilidad, de esta manera se compararon los riesgo con mayor impacto de las amenazas presentes los servidores web, estas vulnerabilidades son las siguientes:

- Con el porcentaje superior a 7% son: fuerza bruta, Cross Site Scripting, Método Http y manejo de identidad.
- Con el porcentaje inferior al 6% son: Manejo de configuración y desarrollo, Owasp/directorios visibles, inyección SQL y Denegación de servicios DoS.

Todo este proceso dio como resultado la identificación de amenazas latentes en los servidores, tomando en cuenta que los que se encuentra mayor al 8% son vulnerabilidades que deben ser corregida a tiempo para evitar daños, robos y ataques al sistema. Por otra parte una vez concluido y obtenido los resultados de las vulnerabilidades se procedió a realizar una escala de cumplimiento de riesgos con el fin de comparar y evaluar mediante un checklist el nivel de seguridad que se obtuvo antes y después de los procesos realizados en la investigación, en este sentido se logró aumentar el nivel de seguridad a los servidores web en un 60%. Finalmente se establece una propuesta que responda los levantamientos de seguridad y brinden a los servicios de aplicaciones y servidores web una mejora de seguridad en sus sistemas informáticos.

Los resultados obtenidos por una investigación de la Universidad de Guayaquil presenta un análisis de vulnerabilidad a los sistemas de información web mediante pruebas de penetración utilizando técnicas de Owasp en la que realiza la identificación de vulnerabilidades con varias herramientas de apoyo que según el autor ayudaron a identificar las vulnerabilidades existentes en el sitio de la carrera de ingeniería en sistemas, por otro lado el presente proyecto se basó en un enfoque más innovador llevando el diagnostico de vulnerabilidades hacia técnicas y herramientas que protegieron el nivel de seguridad al servidor web como también la realización de criterio de vulnerabilidad para identificar que amenazas se encuentran en un impacto muy grave para el servidor web, y de manera establecer propuesta que ayuden a evitar estos tipos de amenazas.

En la comparativa anterior es clave evaluar los riesgos desde los más bajos hasta los más altos, ya que atacantes ejecutan sus técnicas de hacking desde contraseñas débiles, hasta malas configuraciones en el servidor, inexistentes certificaciones crt, cst, Sll, procesos de Header mal configurados, tail2ban si realizar, entre otros. Sin embargo el objetivo de diagnosticar los problemas de seguridad al servidor web recae sobre los parámetros de evaluación de riesgos, lo que da un punto a favor para detectar la vulnerabilidad que se encuentra más presente en el servidor web y de esta manera aplicar técnicas para disminuir estas amenazas.

Otra investigación realizada en la universidad de Chimborazo con el tema “Análisis de vulnerabilidades de software para mejorar la seguridad en los sistemas en los sistemas informáticos” que está dirigido al análisis de vulnerabilidades a los sistemas informáticos y el desarrollo de un sitio web seguro mediante herramientas de pentest y Owasp para mejor los niveles de seguridad facilitando la fluidez al momento de navegar por el sitio web, este enfoque llevado al presente proyecto se traduce al diagnosticar los problemas de seguridad al servidor web, adaptando los criterios de selección de vulnerabilidades y parámetros de vulnerabilidad, con estos dos escenarios planteado se puede argumentar que el uso de herramientas de pentest a pesar de estar encaminados en diferentes campos de aplicación son adaptables a diferentes tipos de necesidades.

Con los resultados expuestos se ha formado una referencia para trabajos futuros que pueden tomar como base el diagnostico de los problemas de seguridad en los servidores para analizar el posible impacto de su implementación en el área de estudio o en otros departamentos, organizaciones afines que estén relacionados con los procesos de seguridad estudiados.

V. CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

- La información recolectada mediante la fundamentación teórica permitió construir una referencia sólida acerca de los problemas de seguridad a los servidores web y el nivel en el que se encontraban, además ayudó a comprender bases teóricas de los procesos de seguridad y su importancia dentro del laboratorio de ciberseguridad.

- A través de la aplicación de instrumentos de recolección de datos se logró establecer una relación al manejo y subutilización de recursos como también los niveles de seguridad alto, medio y bajo existentes en los servidores web.

- Se realizó un análisis de vulnerabilidad mediante el uso de herramientas: Owasp Zap, Nmap, SQLmap, Nessus, Acunetix, Fail2ban, Nikto y WPScan que permitieron la identificación de vulnerabilidades y posterior ejecución de procesos de seguridad mediante el método Hardening para la reducción de riesgos de amenazas en el servidor web.

- El uso de la "metodología Owasp (Open web Application Security Project)" permitió el cumplimiento de procesos de seguridad y resultados de las pruebas, las cuales se identificaron cuatro vulnerabilidades: manejo de identidad, fuerza bruta, Cross Site Script y los métodos Https con un porcentaje superior al 7% considerando los riesgos más peligrosos presentados en los servidores web.

- El uso de las escalas de cumplimento de riesgo permitieron evaluar mediante indicadores de seguridad el antes y después (figura 14 y 145) de los procesos realizados, obteniendo como resultado el 60% en el aumento de seguridad en los servidores web.

- El uso de herramientas gratuitas en el desarrollo de la seguridad al servidor web permitieron obtener los resultados esperados, con procesos óptimos impidiendo el paso de atacantes a alterar u obtener información valiosa.

5.2. RECOMENDACIONES

- La investigación está encaminada en el desarrollo de un propuesta, de esta manera se recomienda ampliar el proceso investigativo tomando como referencia la documentación generada en este proyecto, estudiando la posible implementación en otras organizaciones como entidades bancarias con el fin de aumentar al 100% la seguridad y medir el impacto real que puede causar los atacantes cibernéticos.

- Dar a conocer mediante conferencias pautas de seguridad, donde se recomiende a los analista de aplicaciones y servidores utilizar herramientas como Owasp Zap, para resolver las vulnerabilidades y alcanzar soluciones optimas.

- Dentro de la metodología Owasp es recomendable mantener el análisis de vulnerabilidades de forma constante para comprobar de manera más efectiva los procesos que se ejecuten cumpliendo con los parámetros y criterios de evaluación.

- Es recomendable basarse en indicadores de riesgos y herramientas Open Source para tener diferentes tipos de procesos de seguridad en calidad de los sistemas informáticos.

VI. REFERENCIAS BIBLIOGRÁFICAS

Akamai. (2021). *Visualización de ataques web*. Recuperado de https://www.akamai.com/es/es/resources/our-thinking/state-of-the-internet-report/web-attack-visualization.jsp?theme=dark

Álvarez, K. (2018). *Propuesta de una metodología de pruebas de penetración orientada a riegos* (tesis de maestría). Universidad Espíritu Santo, Guayaquil, Ecuador http://201.159.223.2/bitstream/123456789/2525/1/alvarez%20intriago%20vilma%20karina.pdf

Alvear, R. (2019). *Análisis y diseño de una propuesta para mitigar ataques cibernéticos a correos electrónicos utilizando técnicas de hacking ético* (tesis de grado). Universidad Politécnica Salesiana, Quito, Ecuador https://dspace.ups.edu.ec/bitstream/123456789/17035/1/ups-st004012.pdf

Arenas, E., y López, D. (2021). *Ventajas y desventajas como mecanismo para la prevención de intrusos informáticos*. Universidad Piloto de Colombia. http://polux.unipiloto.edu.co:8080/00000846.pdf

Bautista, J. (2019). *Ataques DDoS con IoT. Análisis y prevención de riesgos* (tesis de grado). Universidad Carlos III de Madrid. España https://core.ac.uk/download/pdf/288502095.pdf

Bermejo, J. (2017). Ataques DoS en aplicaciones web. *Eazel Owasp*. https://owasp.org/www-pdf-archive/Conferencia_OWASP.pdf

Blog, T. (2018). *El hackeo ético al servicio de la detección de vulnerabidades*. Recuperado de https://www.gb-advisors.com/es/pentesting/

Briones, G., y Hernández, E. (2018). *Auditoría de Seguridad del Servidor Web de la Empresa Publinext S.A. Utilizando Mecanismos Basados en OWASP* (tesis de grado). Universidad de Guayaquil. Ecuador http://repositorio.ug.edu.ec/bitstream/redug/26837/1/b-cint-ptg-n.249%20briones%20pincay%20gerson%20hamner.%20hern%c3%a1ndez%20pe%c3%b1a%20herrera%20erika%20bel%c3%a9n.pdf

Burbano, C. (2019). *Propuesta Metodológica para realizar pruebas de penetración en ambientes virtuales* (tesis de grado). Universidad de las fuerzas armadas. Sangolquí, Ecuador https://181.39.85.171/bitstream/123456789/1890/1/burbano%20angulo%20carolina%20alexandra.pdf

Cabezas, N. (2020). *Configuración del firewall de aplicaciones web modsecurity para prevenir diversos ataques hacia aplicaciones web alojados en servidores open source* (tesis de

grado). Pontificia Universidad Católica del Ecuador Sede Esmeraldas, Ecuador https://181.39.85.171/bitstream/123456789/2232/1/cabezas%20cede%c3%91o%20natalie%20karine.pdf

Cañola, J. (2020). *Sistema preventivo contra ataques de denegación de servicio web utilizando Deep Learning* (tesis de grado). Institución Universitaria Acreditada de Alta Calidad. Medellín, Colombia https://repositorio.itm.edu.co/bitstream/handle/20.500.12622/4674/juancanola_2021.pdf?sequence=1&isallowed=y

Carrasco, S. (2020). *Evaluación de mecanismos de seguridad basados en resultados de pentesting para mitigar riegos de intrusión en servidores* (tesis de grado). Escuela Superior Politécnica de Chimborazo, Riobamba, Ecuador http://dspace.espoch.edu.ec/handle/123456789/14345

Centro Criptológico Nacional. (2018). Implementación de seguridad en internet information Service 10 sobre Microsoft Windows server 2016. *Revista CCN-STIC(575).* España

Carvajal, J., Vega, E., & García, R. (2021). *Diseño de un plan de seguridad informática para el sistema de información* (tesis de grado). Universidad Cooperativa de Colombia https://repository.ucc.edu.co/bitstream/20.500.12494/33277/2/2021_dise%c3%b1o_plan_seguridad.pdf

Cautín, C. (2019). *Análisis de vulnerabilidades mediante pruebas de penetración avanzada pentesting al sitio web oficial de la alcaldía del municipio de Quibdó* (tesis de grado). Universidad Nacional Abierta y a Distancia "UNAD". Quibdó, Chocó https://repository.unad.edu.co/bitstream/handle/10596/26950/%20%09cacouting.pdf?sequence=1&isallowed=y

Chavarría, B., & Gudiño, E. (2017). *Implementación de un servidor web y un diseño de una página utilizando herramientas de software libre para el dispensario "Sagrada Familia."* (tesis de grado). Universidad Politécnica Salesiana Sede Guayaquil. Ecuador https://dspace.ups.edu.ec/bitstream/123456789/14162/1/gt001840.pdf

Chiquito, P. (2016). *Análisis de vulnerabilidad de un sistema de información web mediante prueba de penetración utilizando la técnica de OWASP con la finalidad de comprometer el almacenamiento de información de la base de datos* (tesis de grado). Universidad de Guayaquil. Ecuador http://repositorio.ug.edu.ec/handle/redug/11426

Deep Knowledge Group. (2020). *Covid-19 Regional Safety Assessment:* Deep Knowledge Group

Díaz, E. (2018). *Análisis de metodologías para pruebas de penetración mediante Ethical Hacking (*tesis de grado). Universidad Nacional Abierta y a Distancia "UNAD". Yopal https://repository.unad.edu.co/bitstream/handle/10596/27647/erdiazb.pdf?sequence=1&isallowed=y

Enríquez, J. (2015). Los delitos informáticos y su penalización en el código orgánico integral penal Ecuatoriano. *Revista Sathiri*. *CITT(8)* https://revistasdigitales.upec.edu.ec/index.php/sathiri/article/view/404/438

Esteban, B. (2018). *Tipo de Servidores Web*, *Intranetworking.* Recuperado de https://blog.infranetworking.com/tipos-de-servidores-web/

Gado, I. (2015). *Sistema de Detección de Ataques DDoS en Tor* (tesis de grado). Universidad Complutense de Madrid. España https://eprints.ucm.es/id/eprint/33415/1/memoria_tfg.pdf

Garcés, S. (2015). *Seguridad Informática para la red de datos en la Cooperativa de Ahorros y Crédito Unión Popular LTDA* (tesis de grado). Universidad Técnica de Ambato. Ecuador https://repositorio.uta.edu.ec/bitstream/123456789/8654/1/tesis_t975si.pdf

García, D. (2014). *Pruebas de penetración al entorno virtual De-ICE nivel 1* (tesis de grado). Universidad Central "Marta Abreu" de las Villas. Santa Clara https://dspace.uclv.edu.cu/bitstream/handle/123456789/1043/dayana%20garcia%20morell.pdf?sequence=1&isallowed=y

García, J. (2021). *Exploración de red con Nmap y Nessus.* FUOC http://www.sw-computacion.f2s.com/linux/012.3-aspectos_avanzados_en_seguridad_en_redes_apendice.pdf

Gavira, R., Cárdenas, J., y Supelano, J. (2015). *Guía práctica para pruebas de pentest basada en la metodología OSSTMM V2.1 y la guía OWASP V3.0*. Pereira https://repository.unilibre.edu.co/bitstream/handle/10901/17296/gu%c3%8da%20pr%c3%81ctica%20para%20pruebas.pdf?sequence=1&isallowed=y

Gonzales, M. (2018). *Aplicabilidad de controles de seguridad informática que garantice la eficiencia de la administración del servicio de red wifi de la cooperativa UltraHuilca* (tesis de grado). Universidad Nacional Abierta y Distancia UNAD, Neiva. Huila https://repository.unad.edu.co/bitstream/handle/10596/20692/1075224766.pdf?sequence=3&isallowed=y

Gordón, D., y Pacheco, R. (2018). Análisis de estrategias de gestión de seguridad informática con base en la metodología open source security testing Methodology manual (OSSTMM) para la intranet de una institución de Educación Superior. *ReCIBE(1).* Universidad de

Especialidades Espíritu Santo. Guayaquil. Ecuador http://recibe.cucei.udg.mx/index.php/recibe/article/view/90/84

Guirado, R. (2017). *Pentration Testing*-Conceptos generales y situación actual. *Revista ISACA, CGEIT* https://docuri.com/download/penetration-testing-conceptos-generales-y-situacin-actual_59c1e3eef581710b286af888_pdf

Hernández, S., Fernández, C., y Baptista, L. (2014). *Metodología de la investigación* . *Mcgrawhill*. https://www.uca.ac.cr/wp-content/uploads/2017/10/investigacion.pdf

Hernández, S., Zapata, O., y Mendoza, P. (2020). *Metodología de la investigación*.

IC3. Internet Crime Complaint Center IC3

Hidalgo, J. Diseño de una red Wi-Fi para proporcionar servicios de una ciudad digital para Tulcán (tesis de grado). Pontificia Universidad Católica del Ecuador, Quito. Ecuador http://repositorio.puce.edu.ec/handle/22000/7661

Imperva. (2021). *Pruebas de penetración*. Recuperado de https://www.imperva.com/learn/application-security/penetration-testing/

Ionos. (2021). *Linux vs Windows cuadro comparativo-Diguital Guide*. Recuperado https://www.ionos.es/digitalguide/servidores/know-how/linux-vs-windows-el-gran-cuadro-comparativo/

Idiaquez, P. (2018). Cuadro comparativo de servidores. Instituto Tecnológico *Superior de San Luis Potosí.*

Jaramillo Castillo, C. M., & Riofrío Herrera, J. C. (2015). *Metodología para realizar la evaluación, detección de riegos, vulnerabilidades y contramedidas en el diseño e implementación de la infraestructura de la red de la editorial Don Bosco, mediante un test de instrucción de caja blanca (*tesis de grado). Universidad Politécnica Salesiana, Cuenca, Ecuador https://dspace.ups.edu.ec/handle/123456789/7910?mode=full

León, S. (2013). *Manual de Hacking Ético para Pymes (*tesis de grado). Universidad de Azuay. Ecuador http://dspace.uazuay.edu.ec/handle/datos/3584

León, Alvaro (2019). Servidor LiteSpeed: ¿Qué es? Características y Ventajas. Recuperado de https://blog.infranetworking.com/servidor-litespeed/

López, J., y Romero, M. Diseño de la página web de estadías profesionales para la división de administración accesible para personas con discapacidad. *Revista de medios y educación.* Ciudad de México.

Lopera, J., Ramírez, C., Zuluaga, M., y Ortiz, J. (2010). El método analítico como método natural. *Revista Crítica de Ciencias Sociales y Jurídicas*. https://www.redalyc.org/pdf/181/18112179017.pdf

Mateu, C. (2004.). *Desarrollo de aplicaciones web*. Universidad Oberta de Catalunya https://libros.metabiblioteca.org/bitstream/001/591/1/004%20desarrollo%20de%20aplicaciones%20web.pdf

Mendaño, L., y Hurtado, M. (2016). *Implementación de técnicas de hacking ético para el descubrimiento y evaluación de vulnerabilidades de la red de una cartera de estado* (tesis de grado) Escuela Politécnica Nacional. Quito. Ecuador https://bibdigital.epn.edu.ec/handle/15000/16836

Mendoza, M. (2015). *De la identificación y análisis a la gestión de riesgos de seguridad*. *Welivesecurity.* Escuela Europea de Excelencia. Recuperado de https://www.escuelaeuropeaexcelencia.com/2016/07/gestion-de-riesgos-identificacion-analisis/

Metso, J. (2019). Pruebas de penetración. *Artículo científico OANMK.* file:///c:/users/asus/desktop/metso_janne.pdf

Miranda, J. (2016). *Implementación y configuración de un servidor basado en Linux para el laboratorio de desarrollo de software (*tesis de grado). Universidad Técnica de Cotopaxi, la Mana, Ecuador http://repositorio.utc.edu.ec/bitstream/27000/3437/1/t-utc-00714.pdf

Mohammed, K., y Mahmud Daniel. (2021). Mdxpoit: An automated port and scanner. *Revista Mauritius*. https://www.researchgate.net/profile/kashim-kyari-mohammed/publication/350819163_mdxploit_an_automated_port_and_vulnerability_scanner/links/60745bbf458515e7aed403e0/mdxploit-an-automated-port-and-vulnerability-scanner.pdf

Morales, E., Sánchez, F., & Barrera, R. (s.f.). Análisis comparativo de servidores web: Apache vs Microsoft IIS. *Articulo Acalán.* Universidad Autónoma Del Carmen. http://www.repositorio.unacar.mx/jspui/bitstream/1030620191/438/1/portada%2081%20web-1.pdf

Muñoz, A., & Pérez, S. (2017). Pentesting sobre aplicaciones web basado en la metodología OWASP utilizando SBC de bajo costo. *Revista Ibérica de sistemas y tecnologías de información.*

Murillo, M. (2020). *Propuesta de implementación de la metodología 5s para mejorar la gestión de inventarios en una empresa de servicios multiétnicos de energía (*tesis de grado). Universidad Norbert Wiener, Lima, Perú http://repositorio.uwiener.edu.pe/bitstream/handle/123456789/3927/t061_71426088_t.pdf?sequence=1&isallowed=y

Narváez, J. (2019). *Aplicación de metodología OSSTMM para la seguridad de la red inalámbrica* (tesis de grado). Universidad Técnica del Norte, Ibarra, Ecuador http://repositorio.utn.edu.ec/bitstream/123456789/9357/2/04%20isc%20519%20trabajo%20grado.pdf

Netcraft. (2021). *Encuesta sobre servidores web, Netcraft.* Recuperado de https://news.netcraft.com/archives/2021/02/26/february-2021-web-server-survey.html

Nmap.org. (2021). *Escáner de seguridad de Nmap*. Nmap.Org. Recuperado de https://nmap.org/book/inst-source.html

Ochoa, F. (2018). *Estudio de seguridad en las bases de datos, mediante metodologías de pentest, ethical hacking en la secretaria de hacienda Municipal de los Patios (*tesis de grado). Universidad Nacional Abierta a Distancia, San José de Cúcuta https://repository.unad.edu.co/bitstream/handle/10596/21194/13506570.pdf?sequence=1&isallowed=y

OffSec Services Limited. (2021). *Que es Kali Linux-OffSec Services Limited*. Recuperado de https://www.kali.org/docs/introduction/should-i-use-kali-linux/

Open Web Application Security Project. (2017). *Metodología OWASP Top 10 - 2017*. https://wiki.owasp.org/images/5/5e/owasp-top-10-2017-es.pdf

Pérez, C., y Quiñones, J. (2017). *Uso de herramientas de pentesting para el análisis de vulnerabilidades en las comunicaciones móviles de las operadoras ubicadas en la ciudad de Guayaquil (*tesis de grado). Universidad de Guayaquil. Ecuador http://repositorio.ug.edu.ec/bitstream/redug/22444/1/B-CINT-PTG-n.190.p%c3%a9rez%20falcon%c3%ad%20carolina%20victoria.qui%c3%b1ones%20monta%c3%b1o%20jairo%20alexander.pdf

Pilar, A. (2019). *Pentesting para web* (tesis de grado). Universidad Nacional Abierta y a Distancia-UNAD, Bogotá. Colombia https://repository.unad.edu.co/bitstream/handle/10596/25188/adlopezmo.pdf?sequence=1&isAllowed=y

Rodríguez, A., y Santibáñez, A. (2018). Ataques en redes- Redes de computadores 1. *ELO322* http://profesores.elo.utfsm.cl/~agv/elo322/1s18/projects/reports/ataques_en_redes.pdf

Romero, M., Figueroa, G., Vera, D., Álava, J., Parrales, G., Álava, C., Murillo, Á., y Castillo, M. (2018). Introducción a la seguridad informática y el análisis de vulnerabilidades. *3Ciencias. Editorial Área de innovación y desarrollo* http://dx.doi.org/10.17993/ingytec.2018.46

Sánchez, J. (2017). *Análisis de vulnerabilidades y diseño de procesos correctivos de la página web de la Dirección de Educación a Distancia y Virtual de la Universidad Técnica de Ambato (*tesis de grado). Universidad Técnica de Ambato. Ecuador https://repositorio.uta.edu.ec/bitstream/123456789/25531/1/tesis_t1232si.pdf

Sánchez, M., y Santander, M. (2016). Herramientas DNP3 pentesting para redes de infraestructura critica . *Revista* USBMed, 1–8, Vol7 http://revistas.usbbog.edu.co/index.php/ingusbmed/article/view/2487/2248

Sarmiento, W., y Rodríguez, E. (2019). *Definición de una metodología personalizada de hacking ético para empresas públicas de Cundinamarca S.A.E.S.P y ejecución de una prueba a la página web y a los servidores de la entidad, soportando sobre la metodología defina (*tesis de grado). Universidad Católica de Colombia. Bogotá, Colombia https://repository.ucatolica.edu.co/bitstream/10983/23377/1/trabajo%20de%20grado%20seg.%20de%20la%20informacion%20final.pdf

Saura, M. (2016). *Implantación de seguridad en entornos web (*tesis de grado). Universidad Politécnica de Cartagena. Colombia https://repositorio.upct.es/bitstream/handle/10317/233/pfc1918.pdf?sequence=2&isallowed=y

Silva, A., Rentería, D., y Duque, F. (2011). *Análisis Comparativo de las principales técnicas de Hacking Empresarial* (tesis de grado). Universidad Tecnológica de Pereira. Colombia http://repositorio.utp.edu.co/dspace/bitstream/handle/11059/2518/0058S586.pdf?sequence=1&isAllowed=y

Troein, C., y Acayo, G. (2020). ITU Global Cybersecurity Index Overview. *ITU*, 1–25 https://www.wto.org/english/res_e/reser_e/caroline_troein_and_grace_acoyo.pdf

Valderrama, J. (2017). *Pentesting "Prueba de Penetración" para la identificación de vulnerabilidades en la red de computadoras en la alcaldía del Municipio de Cantón del San Pablo, departamento del Chocó (*tesis de grado). Universidad Nacional Abierta y a Distancia UNADA, Quibdó, Chocó https://repository.unad.edu.co/bitstream/handle/10596/18049/1077436201.pdf?sequence=1&isallowed=y

Vargas, X. (2007). *Como hacer investigación cualitativa*. Ciudad de México: Etxeta. http://www.paginaspersonales.unam.mx/files/981/94805617-xavier-vargas-b-como-hacer-investiga.pdf

Verdesoto, A. (2007). *Utilización de hacking ético para diagnosticar, analizar y mejorar la seguridad informática en la intranet de vía celular comunicaciones y representaciones*

*(*tesis de grado). Escuela Politécnica Nacional. Quito https://bibdigital.epn.edu.ec/bitstream/15000/548/1/cd-1053.pdf

Yacchirema, A., Alulema, D., y Aguilar, D. (s.f.). *Análisis de los sistemas de ataque y protección en red inalámbricas Wi Fi, bajo el sistema operativo Linux.* Recuperado de https://repositorio.espe.edu.ec/bitstream/21000/8435/1/ac-red-espe-047801.pdf

Zabala, V. (2016). Desarrollo de una aplicación web utilizando el servidor NGIX en la compañía "Group Tektron" (tesis de grado). Escuela Superior Politécnica de Chimborazo, Riobamba, Ecuador http://dspace.espoch.edu.ec/bitstream/123456789/6256/1/18T00658.pdf

Zaragoza, J. (s.f). Manual Básico Metasploit. *TheJez*. Recuperado de http://index-of.co.uk/infosec/metasploit.pdf

Zetina, E. (2014). *Auditoría de seguridad informática utilizando smartphone (*tesis de grado). Universidad de Quintana Roo. México http://risisbi.uqroo.mx/bitstream/handle/20.500.12249/2109/qa76.9.z57.2014-377.pdf?sequence=1

VII. ANEXOS

Anexo 1: Certificado del acta de sustentación de pre defensa del informe de investigación

UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI
FACULTAD DE INDUSTRIAS AGROPECUARIAS Y CIENCIAS AMBIENTALES
CARRERA DE INGENIERIA EN INFORMATICA

ACTA

DE LA SUSTENTACIÓN DE PREDEFENSA DEL INFORME DE INVESTIGACIÓN DE:

NOMBRE: Castillo Enríquez Alvaro Steebe **CÉDULA DE IDENTIDAD:** 0402039614
NIVEL/PARALELO: 0 **PERIODO ACADÉMICO:** JUN 2020 - SEP 2021

TEMA DE INVESTIGACIÓN: "Pruebas de penetración para la seguridad informática al servidor web del laboratorio de ciberseguridad en la Universidad Politécnica Estatal del Carchi"

Tribunal designado por la dirección de esta Carrera, conformado por:
PRESIDENTE: MSC. CARLITOS ALBERTO GUANO CÁRDENAS
LECTOR: MSC. MILTON GABRIEL DEL HIERRO MOSQUERA
ASESOR: MSC. SAMUEL BENJAMIN LASCANO RIVERA

De acuerdo al artículo 21: Una vez entregados los requisitos para la realización de la pre-defensa el Director de Carrera integrará el Tribunal de Pre-defensa del informe de investigación, fijando lugar, fecha y hora para la realización de este acto:

EDIFICIO DE AULAS: 0 **AULA:** VIRTUAL
FECHA: lunes, 6 de septiembre de 2021
HORA: 15H00
Obteniendo las siguientes notas:
1) Sustentación de la predefensa: 5,53
2) Trabajo escrito 2,32
Nota final de PRE DEFENSA **7,85**
Por lo tanto: **APRUEBA CON OBSERVACIONES** ; debiendo acatar el siguiente artículo:

Art. 24.- De los estudiantes que aprueban el Plan de Investigación con observaciones. - El estudiante tendrá el plazo de 10 días laborables para proceder a corregir su informe de investigación de conformidad a las observaciones y recomendaciones realizadas por los miembros Tribunal de sustentación de la pre-defensa.

Para constancia del presente, firman en la ciudad de Tulcán el lunes, 6 de septiembre de 2021

Firmado digitalmente por CARLITOS ALBERTO GUANO CARDENAS
MSC. CARLITOS ALBERTO GUANO CÁRDENAS
PRESIDENTE

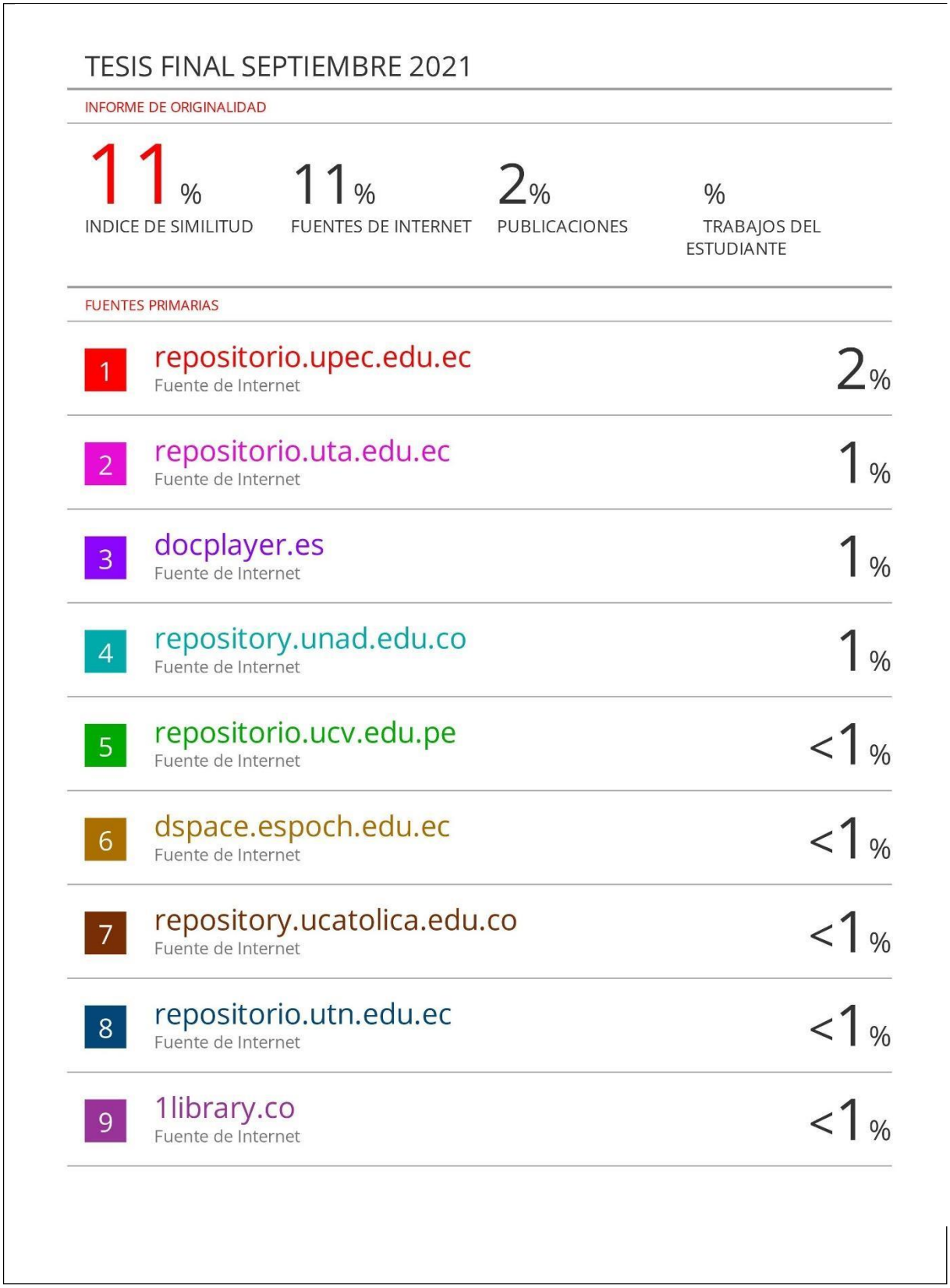
1802590222 SAMUEL BENJAMIN LASCANO RIVERA
MSC. SAMUEL BENJAMIN LASCANO RIVERA
TUTOR

MILTON GABRIEL DEL HIERRO MOSQUERA
Firmado digitalmente por MILTON GABRIEL DEL HIERRO MOSQUERA Fecha: 2021.09.06 16:48:07 -05'00'
MSC. MILTON GABRIEL DEL HIERRO MOSQUERA
LECTOR

Adj.: Observaciones y recomendaciones

Figura 146. Acta sustentación de pre defensa del informe de investigación

Anexo 2: Certificado Turnitin



TESIS FINAL SEPTIEMBRE 2021

INFORME DE ORIGINALIDAD

11%	11%	2%	%
INDICE DE SIMILITUD	FUENTES DE INTERNET	PUBLICACIONES	TRABAJOS DEL ESTUDIANTE

FUENTES PRIMARIAS

1	repositorio.upec.edu.ec Fuente de Internet	2%
2	repositorio.uta.edu.ec Fuente de Internet	1%
3	docplayer.es Fuente de Internet	1%
4	repository.unad.edu.co Fuente de Internet	1%
5	repositorio.ucv.edu.pe Fuente de Internet	<1%
6	dspace.espoch.edu.ec Fuente de Internet	<1%
7	repository.ucatolica.edu.co Fuente de Internet	<1%
8	repositorio.utn.edu.ec Fuente de Internet	<1%
9	1library.co Fuente de Internet	<1%

Anexo 3: Certificado del abstract por parte de idiomas

UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI
FOREIGN AND NATIVE LANGUAGE CENTER

ABSTRACT- EVALUATION SHEET

NAME: Alvaro Steebe Castillo Enríquez **DATE:** 17 de septiembre de 2021

TOPIC: "Pruebas de penetración para la seguridad informática al servidor web del laboratorio de ciberseguridad en la Universidad Politécnica Estatal del Carchi"

MARKS AWARDED **QUANTITATIVE AND QUALITATIVE**

VOCABULARY AND WORD USE	Use new learnt vocabulary and precise words related to the topic	Use a little new vocabulary and some appropriate words related to the topic	Use basic vocabulary and simplistic words related to the topic	Limited vocabulary and inadequate words related to the topic
	EXCELLENT: 2	GOOD: 1,5	AVERAGE: 1	LIMITED: 0,5
WRITING COHESION	Clear and logical progression of ideas and supporting paragraphs.	Adequate progression of ideas and supporting paragraphs.	Some progression of ideas and supporting paragraphs.	Inadequate ideas and supporting paragraphs.
	EXCELLENT: 2	GOOD: 1,5	AVERAGE: 1	LIMITED: 0,5
ARGUMENT	The message has been communicated very well and identify the type of text	The message has been communicated appropriately and identify the type of text	Some of the message has been communicated and the type of text is little confusing	The message hasn't been communicated and the type of text is inadequate
	EXCELLENT: 2	GOOD: 1,5	AVERAGE: 1	LIMITED: 0,5
CREATIVITY	Outstanding flow of ideas and events	Good flow of ideas and events	Average flow of ideas and events	Poor flow of ideas and events
	EXCELLENT: 2	GOOD: 1,5	AVERAGE: 1	LIMITED: 0,5
SCIENTIFIC SUSTAINABILITY	Reasonable, specific and supportable opinion or thesis statement	Minor errors when supporting the thesis statement	Some errors when supporting the thesis statement	Lots of errors when supporting the thesis statement
	EXCELLENT: 2	GOOD: 1,5	AVERAGE: 1	LIMITED: 0,5
TOTAL/AVERAGE	9 - 10: EXCELLENT 7 - 8,9: GOOD 5 - 6,9: AVERAGE 0 - 4,9: LIMITED	TOTAL 9		

1

Figura 147. Certificado rúbrica del abstract por parte de idiomas

UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI FOREIGN AND NATIVE LANGUAGE CENTER

Informe sobre el Abstract de Artículo Científico o Investigación.

Autor: Alvaro Steebe Castillo Enríquez

Fecha de recepción del abstract: 17 de septiembre de 2021
Fecha de entrega del informe: 17 de septiembre de 2021

El presente informe validará la traducción del idioma español al inglés si alcanza un porcentaje de: 9 – 10 Excelente.

Si la traducción no está dentro de los parámetros de 9 – 10, el autor deberá realizar las observaciones presentadas en el ABSTRACT, para su posterior presentación y aprobación.

Observaciones:
Después de realizar la revisión del presente abstract, éste presenta una apropiada traducción sobre el tema planteado en el idioma Inglés. Según los rubrics de evaluación de la traducción en Inglés, ésta alcanza un valor de 9, por lo cual se validad dicho trabajo.

Atentamente

Firmado electrónicamente por:
EDISON BOANERGES PENAFIEL ARCOS

Ing. Edison Peñafiel Arcos MSc
Coordinador del CIDEN

2

Figura 148. Certificado del abstract por parte de idiomas

Anexo 4: Entrevista para la extracción de la información

Entrevista al director del laboratorio de ciberseguridad de la Universidad Politécnica Estatal del Carchi en la ciudad de Tulcán.

La entrevista tiene como finalidad recolectar datos relacionados a los indicadores de las variables dependientes e independientes. La información recolectada hace referencia a los procesos de seguridad y el manejo empleado a los servidores web del laboratorio de ciberseguridad, esto con la finalidad de facilitar la respuesta del entrevistado.

1) **Indique como maneja actualmente la seguridad en los servidores que tiene en el laboratorio de ciberseguridad.**

Prácticamente el tema del laboratorio que tiene en este momento la institución esta específicamente habilitado para realizar pruebas de investigación por las configuraciones o servicios que se quieran desarrollar en el lugar. Actualmente el laboratorio cuenta con equipamiento e infraestructura que va a permitir a los estudiantes y docentes realizar pruebas de investigación sobre algún tema en específico que lo desea realizar.

Análisis

La respuesta por parte del entrevistado indica que el laboratorio netamente está habilitado para encuentros prácticos con los estudiantes y docentes en donde poseen un espacio el cual puedan realizar sus pruebas de investigación contando equipamiento para la elaboración.

2) **En que metodología se basa para ejecutar los procesos de seguridad y evitar vulnerabilidades.**

Actualmente nosotros no contamos con un proceso o una metodología para el tema de la seguridad sino estamos habilitando diferentes servicios como por ejemplo el trabajo de titulación como es la incorporación de un centro respuesta, un CSIRT para la Universidad y sobre eso enmarcarnos para la implementación de seguridades e implementación de políticas que permitan administrar esos servicios, pero el centro del laboratorio no cuenta con un organigrama funcional de profesionales en vista de que se encuentran dispersos en diferentes áreas. Por cada trabajo de investigación que va generando, implementando, desarrollando y sobre eso se va aplicando técnicas o metodologías que vaya a permitir el fortalecimiento de la seguridad en ese espacio e igual manera el servicio que se quiera brindar a la sociedad.

Análisis

El entrevistado señala que no cuenta con un proceso o metodología, es decir no cuentan con un organigrama funcional en donde profesionales se encarguen de estar en constante monitoreo con respecta a la seguridad, sino que habilita espacios o servicios como es la incorporación de un centro de respuesta CSIRT para la Universidad y mediante esto implementar las seguridad, políticas y servicios, al mismo tiempo esta respuesta da cumplimiento para medir la variable independiente porque se precisa de forma clara los objetivos, por ende debe existir metodologías procesos que ayuden y mitiguen cualquier riesgo o amenaza a los servidores.

3) **¿Cuáles son los objetivos del laboratorio de ciberseguridad y como se relacionan con los procesos que usted realiza?**

Dentro de los objetivos principalmente es contar con un espacio que va a permitir no solamente a los estudiantes o docentes sino que a la comunidad, a nuestro medio a contar con un espacio físico en donde podamos nosotros solventar inquietudes, inconvenientes y problemas que se presentan día a día en cuento al tema de la seguridad de los sistemas y de la información. Este espacio está constituido para que los profesionales, estudiantes levante y realicen investigaciones y trabajos de carrera y puedan ellos tener un espacio físico donde puedan contar con diferentes elementos para poder aplicar toda la parte técnica y tecnológica para el desarrollo de sus actividades.

Análisis

Según lo manifiesta el encargado del laboratorio de ciberseguridad que los objetivos del laboratorio se relacionan con un ambiente tecnológico que da cumplimiento a temas de ciberseguridad como también problemas que se dan día a día con lo que respecta a la seguridad, brindando este ambiente a los estudiantes, docentes y la comunidad. En este sentido se puede tener claro cuáles son las actividades y metas que se cumplen.

4) **Cuantos representantes conforman el laboratorio de ciberseguridad.**

Los representantes principalmente parte de un proyecto de investigación que participaron docentes de la cerrera, docentes de otras instituciones como Universidad Indoamérica y de igual manera profesionales de la Universidad de Colombia, Este proyecto tuvo una vigencia de 3 años y dentro de la implementación nosotros continuamos con la parte de la administración y de manera personal. Sin embargo no cuenta con una estructura que este

constituida por profesionales que respondan por los trabajos que se desarrollen en ese lugar.

Análisis

A través de la respuesta se concluye que este proyecto tuvo una participación de docentes de otras carreras e instituciones de otras Universidad. Sin embargo se mantuvo 3 años en constante observación, después de eso solamente se mantuvo por parte de la administración y de manera personal, es decir que al no contar con una estructura constituida por profesionales difícilmente que los trabajos que se desarrollen en ese lugar sean implementados de manera continua.

5) **Aproximadamente cuanto tiempo se demora en realizar un proceso de análisis de vulnerabilidad o pentest.**

Depende el trabajo que se lo vaya a realizar, el trabajo de titulación que lo vienen desarrollando los estudiantes ellos cuentan con su planificación del desarrollo de su trabajo, con sus tiempos establecidos para ejecutar sus trabajos de investigación y va dependiendo no específicamente en un tiempo determinado sino más bien se constituyen en las prácticas y pruebas que se vayan generando, como también el tiempo que ellos vayan considerando los resultados que obtengan.

Análisis

Se puede inferir que el tiempo que se demore en aplicar un proceso de analices de vulnerabilidad sera dependiendo del trabajo exhaustivo que se desea aplicar, tomando en cuenta los resultados y el límite de tiempo que se les ha asignado en un proyecto de investigación.

6) **Para realizar un proceso referente a indicadores de riegos o amenazas que documentos o recursos son utilizados?**

En este sentido nosotros estamos trabajando con un trabajo de titulación a nivel de lo que es el Gobierno municipal de Tulcán y este trabajo nos ha permitido que se hago un levantamiento base de la información que actualmente disponen en ese lugar y sobre eso proponer o trabajar con sistemas de gestión de seguridad de la información que sería lo más importante. La institución en donde el departamento de tecnología tenga este material este documento permita solventar o minimizar los riesgos que la empresa lo requiera.

Análisis

Con la respuesta facilitada por el entrevistador se menciona que se debe realizar un levantamiento base de la información con el fin de proponer proceso o metodologías que sirvan a la mejora de la seguridad y de igual manera que permitan disminuir la *vulnerabilidad a los sistemas* correspondiente al indicador de la variable dependiente.

7) **¿Qué inconveniente percibió dentro de la seguridad en el laboratorio de ciberseguridad?**

En este proceso prácticamente necesitamos nosotros habilitar una infraestructura independiente a lo constituyen la infraestructura de red de la Universidad. Actualmente contamos con un enlace de comunicación directa al centro de datos de la Universidad que van a ser en algún momento un poco de riesgo para las aplicaciones que ellos disponen, sin embargo nos encontramos en diferentes segmentos de red en otras Vlans que la institución nos ha habilitado y en base a eso nosotros habilitamos y configuramos los servicios para poder realizar las prácticas y los trabajos que necesitamos realizar.

Análisis

El resultado de la pregunta permitió realizar el siguiente diagrama que describen el acceso de una infraestructura independiente contando con una comunicación directa al centro de datos de la Universidad y de igual manera proporcionar mediante diferentes segmentos de red otras Vlans que son otorgados por la institución, con el fin de poder ahí realizar las configuraciones, prácticas y los trabajos que se deseen ejecutar.

8) **¿Qué alternativas daría para mejorar la seguridad en el laboratorio de ciberseguridad?**

El tema de la seguridad es un tema bastante amplio que pasa por el conocimiento y por la implementación de equipos tecnológicos y de igual manera por la participación del recurso humano. En este sentido se está generando campañas de concientización acerca de phishing que se está evitando a través de cuentas de correo electrónico y esto va a permitir de alguna manera concientizar a las personas que trabajan dentro de la institución para que ellos también sean una parte fundamental de la seguridad de la información, porque una amenaza no únicamente llega por ataques a la infraestructura sino que llega también por vulnerabilidades a los usuarios que de alguna manera sin hacerlo de manera intencional abren o descargan aplicaciones que pueden comprometer a los servicios que tienen las instituciones.

Análisis

Se concluye que el laboratorio cuenta con equipos tecnológicos, servicios y espacios que ayudan a la comunidad en general, aportando con conocimientos mediante los trabajos de investigación con el fin de concientizar una calidez de seguridad adecuada, ya que actualmente se ha presenciado muchas amenazas y vulnerabilidades afectado la confidencialidad, integridad y disponibilidad de la información. Es por ello que es necesario y optimo proponer procesos, metodologías y herramientas que ayuden a proteger a cada parte que conforma una infraestructura de red. En este sentido se requiere de una concientización profunda a los usuarios que forman parte de cada entidad para ser partícipes y puedan comprometerse a la mejora continua de la seguridad.

Anexo 5: Certificado de conformidad

MEMORANDO Nro. UPEC-CIC-2021-199
Tulcán, 31 de Agosto de 2021

PARA: Msc.
Georgina Arcos
DIRECTORA CARRERA DE COMPUTACIÓN
UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI

ASUNTO: INFORME DE CONFORMIDAD

De mi consideración,

Saludos cordiales, por medio del presente, y en respuesta al oficio Nro. 001 emitido por el Sr. Estudiante Álvaro Stebee Castillo Enríquez, me permito informar, y ***CERTIFICAR*** *que se realizaron las configuraciones y pruebas en el servidor alojado en el Laboratorio de Cyberseguridad de la UPEC, del proyecto de culminación de carrera cuyo tema es:* ***"Pruebas de penetración para la seguridad informática al servidor web del laboratorio de ciberseguridad en la Universidad Politécnica Estatal del Carchi".***

Particular que me permito poner en su conocimiento, para los fines académicos correspondientes.

Atentamente,

Firmado electrónicamente por:
JAIRO VLADIMIR
HIDALGO
GUIJARRO

Ing. Jairo V. Hidalgo G. MSc.
DOCENTE INVESTIGADOR_UPEC

Anexo 6: Entrega del artículo científico

MEMORANDO Nro. UPEC-CIC-2021-224
Tulcán, 22 de Septiembre de 2021

PARA: PhD.
Duván Ávalos
RESPONSABLE DE LA UNIDAD DE PUBLICACIONES
UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI

ASUNTO: ENTREGA DE ARTÍCULO CIENTÍFICO

De mi consideración,

Saludos cordiales, por medio del presente, me permito hacer la entrega del artículo científico para su revisión y proceso de publicación en la revisa ***SATHIRI;*** *trabajo de investigación desarrollado como resultado del Trabajo Final de Carrera.*

TEMA: ***"PRUEBAS DE PENETRACIÓN PARA LA SEGURIDAD INFORMÁTICA AL SERVIDOR WEB DEL LABORATORIO DE CIBERSEGURIDAD EN LA UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI, 2021".***

Particular que me permito poner en su conocimiento, para los fines académicos correspondientes.

Atentamente,

Firmado electrónicamente por:
JAIRO VLADIMIR
HIDALGO
GUIJARRO

Ing. Jairo V. Hidalgo G. MSc.
DOCENTE INVESTIGADOR_UPEC