UNIVERSIDAD ESTATAL

PENÍNSULA DE SANTA ELENA

FACULTAD DE SISTEMAS Y TELECOMUNICACIONES

CARRERA DE TECNOLOGÍAS DE INFORMACIÓN Y COMUNICACIÓN

EXAMEN COMPLEXIVO

Componente Práctico, previo a la obtención del Título de:

INGENIERO EN TECNOLOGÍAS DE LA INFORMACIÓN

Análisis de Vulnerabilidades del Portal Web utilizando Metodologías de Hacking Ético para un GAD Municipal de la Provincia de Santa Elena

Autor:

Luis Arturo Rodríguez Matías

LA LIBERTAD – ECUADOR

2021

APROBACIÒN DEL TUTOR

En mi calidad de tutor del trabajo de componente práctico del examen de carácter complexivo: "**Análisis de Vulnerabilidades del Portal Web utilizando Metodologías de Hacking Ético para un GAD Municipal de la Provincia de Santa Elena**", elaborado por el sr Rodriguez Matías Luis Arturo, de la carrera de Tecnología de la Información de la Universidad Estatal Península de Santa Elena, me permito declarar que luego de haber orientado, estudiado y revisado, la apruebo en todas sus partes.

La libertad, 7 de marzo 2021

……..

Ing. Iván Coronel Suárez, MSIA.

DECLARACIÓN

El contenido del presente componente práctico del examen de carácter complexivo es de mi responsabilidad; el patrimonio intelectual del mismo pertenece a la Universidad Estatal Península de Santa Elena

Luis Arturo Rodriguez Matias

AGRADECIMIENTO

Es indispensable agradecer primero a Dios, por darme las fuerzas para seguir en esta lucha de alcanzar mi meta, también agradecido a la Universidad Estatal Península de Santa Elena, por haberme brindado los conocimientos necesarios para el desarrollo de este proyecto.

A los docentes que nos brindaron su sabiduría en varios campos del conocimiento y formaron parte de esta etapa, unos fueron solo docentes y otros se convirtieron en verdaderos guías e incluso amigos. Gracias por la paciencia y dedicación en la formación de profesionales.

Quiero agradecer a mi madre que, aunque no esté conmigo sé que me guio y cuido todo el tiempo, a mi hermano y hermanas por creer en mí y apoyarme. Gracias por los valores fomentados y darme el aliento necesario para seguir adelante

Luis Arturo Rodriguez Matías

DEDICATORIA

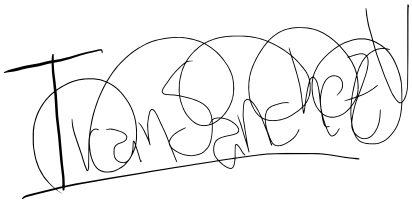
Principalmente dedico este trabajo a mi madre que, aunque se encuentre en un lugar mejor, sé que ha estado presente en todo momento, me ha cuidado, me ha guiado y me ha sabido guiar por el camino correcto. Como quisiera que estuvieras aquí y que puedas observar que cumplí mi promesa de ser un profesional, Dios lo quiso así y yo acepto eso. Sin embargo, esa promesa que te hice fue la motivación principal para continuar y seguir preparándome y en donde quieras que estés, sé que estarás orgullosa de mi porque esta es una de muchas metas que pretendo cumplir.

También dedico este trabajo a mi hermano y hermanas que son mi pilar fundamental para cada día seguir adelante gracias por su apoyo y comprensión en todo sentido.

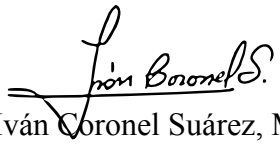
Luis Arturo Rodriguez Matías

TRIBUNAL DE GRADO

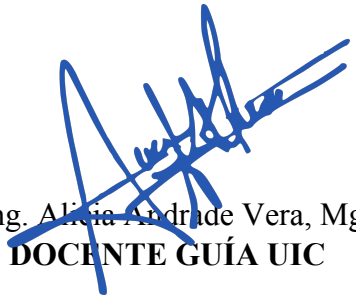
Ing. Samuel Bustos Gaibor, Mgt.
DIRECTOR DE LA CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN



Ing. Iván Sánchez Vera, MIST
DOCENTE ESPECIALISTA



Ing. Iván Coronel Suárez, MSIA.
DOCENTE TUTOR



Ing. Alicia Andrade Vera, Mgt.
DOCENTE GUÍA UIC

RESUMEN

La presente propuesta tecnológica estableció como finalidad realizar un análisis de vulnerabilidades para un Gad Municipal de la Provincia de Santa Elena, debido a que la información se ha convertido en uno de los activos intangibles más importantes del mundo, y por tal motivo la falta de realización de este tipo de análisis pone en riesgo la integridad de la información.

Para lograr este objetivo se utilizó varias herramientas gratuitas, mediante un análisis comparativo entre ellas, se determinó cual es la mejor y la más factible para la realización del respectivo análisis.

También se consideró utilizar la metodología de hacking ético, para recrear ambientes virtuales y poder analizar las brechas de seguridad desde una perspectiva diferente, permitiendo recolectar información, realizar un escaneo y una enumeración de resultados. Además, se utilizó la metodología Open Information System Security Group (ISSAF), puesto que está basada en la planificación - preparación, evaluación y reportes, la cual permitió examinar y emitir informe final detallado.

Para culminar la investigación se brindará un informe detallado de las vulnerabilidades encontradas y las acciones que se deberían tomar para mitigar la explotación de estas, con el objetivo principal de que sean insumos válidos y confiables para la toma de decisiones en el aspecto de mitigación y remediación de daños ante un ataque informático.

TABLA DE CONTENIDO

INDICE DE FIGURAS

INDICE DE TABLAS

CAPÍTULO 1

1. FUNDAMENTACIÓN

1.1 ANTECEDENTES

En los últimos años muchas empresas, instituciones y organizaciones gubernamentales manejan su información en la web y utilizan el Internet como parte de su estrategia en el mercado global, obteniendo así mejores ventajas de acceso y disponibilidad a la información, publicidad, ventas directas, noticias, teletrabajo, email y mensajería. En el mundo de la web la mayor preocupación son los delincuentes cibernéticos, personas curiosas quienes constantemente buscan la manera de romper la seguridad, comprometiendo los niveles de disponibilidad, integridad y confidencialidad de los sistemas de información de portales web de grandes organizaciones, con el fin de sacar provecho económico y personal. [1]

Se trata de un fenómeno que ha sufrido un enorme crecimiento en los últimos años debido al surgimiento de la evolución tecnológica: ejemplo de ello son los grupos o actores como WikiLeaks o Anonymous, quienes han llevado a cabo ciberataques dirigidos a la consecución de una denegación de servicio, la destrucción de datos o la publicación de información confidencial a modo de protesta en contra de determinados gobiernos. [2]

En la actualidad y en América Latina el porcentaje de ataques informáticos es de un 30% según datos expuestos en la 7ma Cumbre Latinoamérica de Análisis de Seguridad. [3] en este informe estadístico muestra el nivel de seguridad de las aplicaciones vulnerables en Latinoamérica. La configuración inadecuada de los sistemas e incluso los errores de programación de algunas aplicaciones influyen mucho en los ataques informáticos, demostrando el estado urgente de realizar análisis de vulnerabilidades en portales web.

Según el ITRC Data Breach Reports el porcentaje de ataques por tipo de instituciones estuvo distribuido en el 2017 de la siguiente manera: A empresas privadas un 55.1%, instituciones de salud un 23.7%, instituciones de Educación un 8.0% e instituciones gubernamentales un 4.7% [4]. Lo cual nos da a conocer que las instituciones gubernamentales están dentro del grupo vulnerable para ataques informáticos.

El Security Report Latinoamérica (ESET), indica los niveles de implementación de prácticas de gestión para la seguridad. Según el estudio que realizó ESET en el 2019, por medio de encuestas a administradores de sistemas y ejecutivos de varias empresas, las organizaciones no saben cómo responder en el caso de que ocurra un incidente que pueda

poner en riesgo sus operaciones. Esto es clave no solo para tener una respuesta rápida y eficiente para la recuperación del incidente, sino también como medida de protección para identificar las vulnerabilidades y evitar que incidentes de este tipo vuelvan a presentarse. [5]

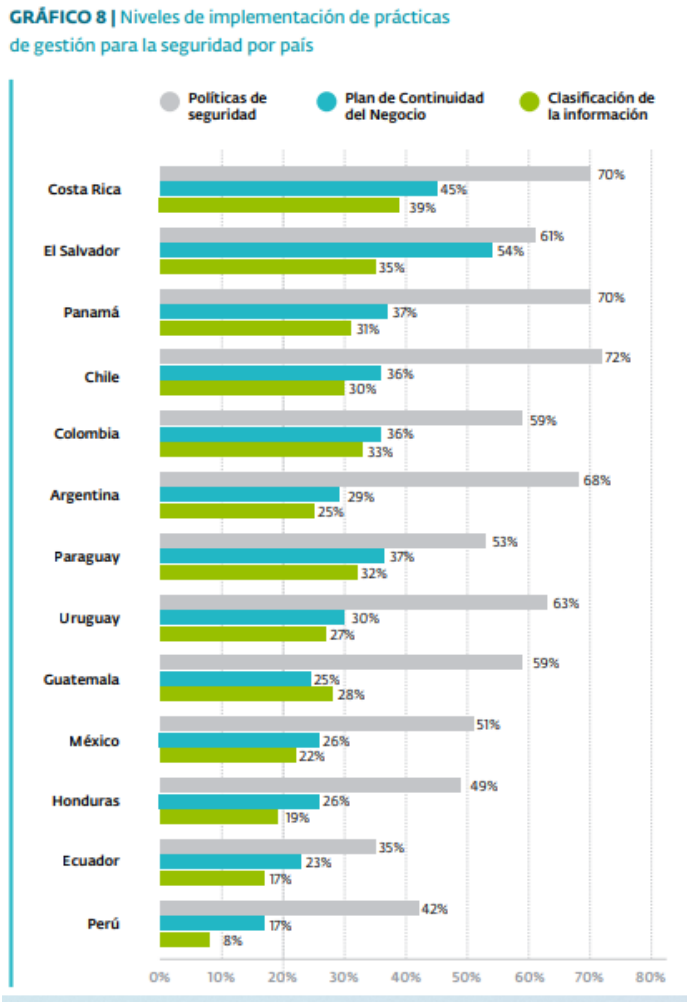


Ilustración 1 Niveles de Implementación de prácticas de gestión para la seguridad por país.

Casi todas las organizaciones públicas o privadas, al igual que las personas, dependen de alguna manera de la tecnología de la información como una herramienta esencial para lograr sus objetivos de negocio o para poder desarrollar actividades en su vida cotidiana; al mismo tiempo, todos tienen que enfrentarse con una amplia gama de amenazas y vulnerabilidades asociadas a los entornos informáticos de hoy. [6] sin embargo, no está de más preocuparse por la forma en que la información está siendo tratada y en las limitaciones gubernamentales que tienen las herramientas utilizadas.

Los gobiernos locales han empezado a estructurar sus gobiernos electrónicos aún a pesar de que no hay un modelo establecido. Estas instancias político-administrativas inician

con páginas Web, en las que insertan planes de trabajo, datos, rendiciones de cuentas, presupuestos, mapas, enlaces u otros, pero sin identificar la información actualizada. [7].

Estas sirven como herramientas para brindar atención a la ciudadanía y dar a conocer sus servicios y beneficios a nivel nacional e internacional, por tal motivo es indispensable precautelar la integridad, seguridad y disponibilidad de la información.

El presente trabajo propone realizar un análisis de las vulnerabilidades del portal web de un GAD Municipal de la Provincia de Santa Elena con la finalidad de dar a conocer las posibles vulnerabilidades y ayudar a mitigar la explotación de estas.

1.2 DESCRIPCIÓN DEL PROYECTO

Según el ITRC Data Breach Reports el porcentaje de ataques por tipo de instituciones estuvo distribuido en el 2017 de la siguiente manera: A empresas privadas un 55.1%, instituciones de salud un 23.7%, instituciones de Educación un 8.0% e instituciones gubernamentales un 4.7% [4]. Lo cual nos da a conocer que las instituciones gubernamentales están dentro del grupo vulnerable para ataques informáticos.

Es por esta razón que se desea realizar un análisis de vulnerabilidades que nos permita analizar las posibles causas de estas y brindar recomendaciones que ayuden a evitar que estas vulnerabilidades sean explotadas con fines maliciosos. Para este trabajo de investigación se realizarán las siguientes fases:

1.2.1 Fase de Recolección de Información

Se realizará un levantamiento de información del portal web del GAD Municipal donde se realizará el análisis de vulnerabilidades, este proceso se lo realizará mediante un reconocimiento pasivo. Este se consigue la información sin interacción directa con el objetivo mediante el uso de técnicas tales como la ingeniería social, búsquedas por internet o mediante el uso de aplicativos webs que nos puedan brindar información crucial para el desarrollo de la investigación.

Las herramientas para utilizar en esta fase son:

- Búsqueda de información en los buscadores como Google
- Buscar en la base de datos de Internet (Whois)
- Buscar en la base de datos de Internet (Shodan)
- Buscar nombres de dominios
- Buscar información de contacto

- Buscar toda la información que se pueda extraer de los DNS (Domain Name Server)

Una vez que se logre recolectar toda la información necesaria en la primera fase, esta se la clasificara de acuerdo con el tipo de información: Ip publica, nombre del servidor, sistema operativo, versión de las tecnologías utilizadas.

1.2.2 Fase de Virtualización de Sistemas

En esta fase se creará máquinas virtuales o Live USB con sus correspondientes sistemas operativos y sus configuraciones necesarias de las siguientes distribuciones que serán necesarias para el respectivo análisis:

- Distribución Linux (Kali Linux)
- Herramientas de Análisis de Vulnerabilidades

Kali Linux: Kali Linux es la plataforma de prueba de penetración más poderosa y popular del mundo, utilizada por profesionales de seguridad en una amplia gama de especializaciones, que incluyen pruebas de penetración, análisis forense, reversa ingeniería y evaluación de vulnerabilidad. Es la culminación de años de refinamiento y el resultado de una evolución continua de la plataforma, de WHoppiX a WHAX, a BackTrack, y ahora a un completo marco de pruebas de penetración que aprovecha muchas características de Debian GNU / Linux y la vibrante comunidad de código abierto en todo el mundo. [8]

Esta virtualización servirá para ofrecer un ambiente o escenario correspondiente para realizar las pruebas de escaneo y utilizar las diferentes herramientas de análisis de vulnerabilidades que nos ofrece Kali Linux.

1.2.3 Fase de Análisis de Vulnerabilidades

Para esta fase procederemos a utilizar las herramientas que vienen incluidas en el sistema Kali Linux las cuales son:

1.2.4 Herramientas de Escaneo:

Uniscan: Uniscan es una sencilla herramienta diseñada para ayudarnos a buscar vulnerabilidades en cualquier aplicación web muy fácilmente, vulnerabilidades como, por ejemplo, carga de archivos locales, ejecución de código remoto e incluso carga de archivos de forma remota en la herramienta. [9]

Nmap: Nmap ("mapeador de redes") es una herramienta de código abierto para exploración de red y auditoría de seguridad. Se diseñó para analizar rápidamente grandes redes, aunque funciona muy bien contra equipos individuales. [10]

Maltego: Maltego es una herramienta de reconocimiento de Kali Linux desarrollada por Paterva, empresa orientada a análisis de datos para la realización de auditorías informáticas creada en el año 2007. Se usa para la recolección de información abierta y pública de internet. También posee reconocimiento DNS, pero lo que le caracteriza es que despliega gráficas de análisis.

FOCA: (Fingerprinting Organizations with Collected Archives) es una herramienta utilizada principalmente para encontrar metadatos e información oculta en los documentos que examina. Estos documentos pueden estar en páginas web, y con FOCA se pueden descargar y analizar. [11]

Nessus: El escáner de vulnerabilidades más usado en el mundo. Esta herramienta de alto nivel detecta amenazas en tiempo real y gracias a su precisión, evita la generación de falsos positivos. El poderoso Nessus previene eficientemente los ataques de red identificando las debilidades y errores de configuración que pueden ser usados para permitir el ingreso de amenazas al sistema. [12]

Se utilizarán más de una herramienta en la fase de escaneo, para hacer un cuadro comparativo y poder visualizar que herramienta nos brinda la mejor información del escaneo de vulnerabilidades.

Estas herramientas nos brindan la posibilidad de obtener un análisis de manera más detallado, una vez identificadas las posibles vulnerabilidades presentes en el portal web las podremos clasificar de acuerdo con el nivel de riesgo que presenten.

1.2.5 Fase de Análisis de Resultados

La identificación de las vulnerabilidades obtenidas en la fase anterior se las clasificara de acuerdo con el nivel de riesgo, usualmente son tres: Alto, Medio y Bajo, esta se realiza de acuerdo con la versión del sistema operativo y de los servicios y aplicaciones detectados comparándolos contra una base de datos de vulnerabilidades que se actualiza frecuentemente conforme nuevos huecos de seguridad son descubiertos.

1.2.6 Informe final

En la última fase se realizará un informe final brindando toda la información detallada del análisis de vulnerabilidades realizado, utilizando las herramientas de la distribución de Kali Linux, el cual se brindará las recomendaciones necesarias para mitigar la explotación de las vulnerabilidades encontradas.

Este proyecto contribuirá a la línea de investigación Tecnologías y Gestión de la Información, debido a que la propuesta está relacionada con temas de seguridad de las tecnologías de la información que permitan generar información indispensable para la toma de decisiones [13]

1.3 Objetivos

1.3.1 Objetivo General

Analizar las vulnerabilidades existentes del portal web de un Gobierno Autónomo Descentralizado mediante metodologías de hacking ético para brindar recomendaciones que ayuden a mitigar los riesgos encontrados.

1.3.2 Objetivos Específicos

- Analizar el estado actual del portal web, utilizando la metodología ISSAF para la obtención de Información necesaria.
- Seleccionar las herramientas de análisis óptimas mediante un análisis comparativo para detectar las vulnerabilidades del portal web.
- Clasificar las vulnerabilidades encontradas de acuerdo con su nivel de riesgo.
- Redactar un informe final con los resultados obtenidos para mitigar ataques a futuro.

1.4 JUSTIFICACIÓN

La seguridad informática ha ganado popularidad en los últimos años y ha pasado de ser considerada un gasto, a ser vista como una inversión por parte de los directivos de las empresas y organizaciones a nivel mundial. En algunos países esto ha sucedido de forma acelerada, en otros el paso ha sido más lento; pero en última instancia todos han convergido en un mundo digital en el que la información es el activo intangible más

valioso; y por consiguiente debe ser protegido de posibles pérdidas, robos, mal uso, etc. [14]

La página web es un canal de comunicación, existen medios y recursos que permiten saber a las empresas quiénes visitan sus sitios web, cuánto tiempo permanecen en ella, la fidelidad de usuarios, de que países y el posicionamiento, por tanto, es una herramienta que permite medir el impacto y el beneficio para una organización. [15]

En el 2019 ESET realizó una encuesta en donde los datos finales determinaron que la cantidad de amenaza marcó un máximo histórico, cuando tanto el número de detecciones como la tasa de generación de nuevas variantes registraron un crecimiento exponencial Aquel año, un tercio de las detecciones se concentró en países de América Latina, donde Ecuador alcanzo 65% del índice de incidentes de seguridad por país. [5]

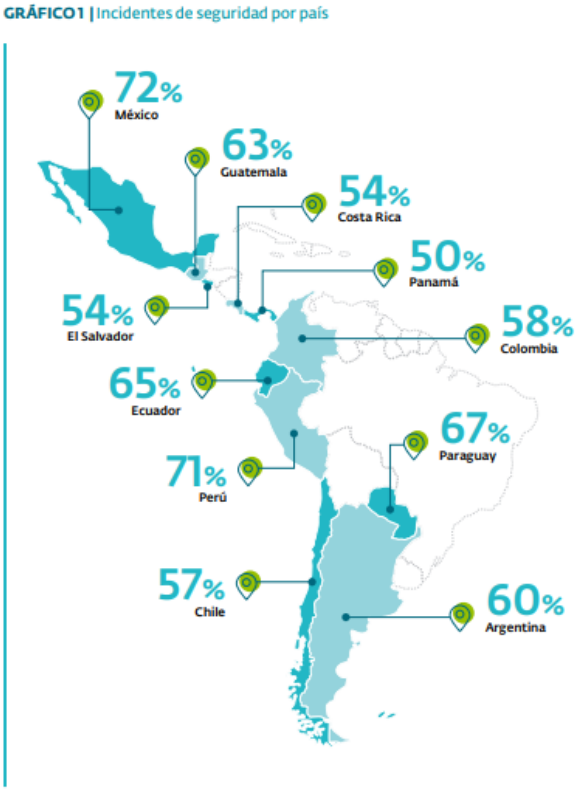


Ilustración 2 Incidentes de seguridad por país.

La comunidad científica ha investigado e implementado mecanismos que permitan disminuir y mitigar estos ataques de seguridad como hurto, modificación, espionaje, interrupción, falsificación, denegación de servicios, etc., empleando tecnologías de virtualización y metodologías de hacking ético, cuya aplicación permite disminuir el

riesgo a equipos y redes en producción, precautelando la información y servicios de las organizaciones. [14]

Utilizar metodologías de hacking ético, nos brinda la posibilidad de realizar un análisis de vulnerabilidades de la mejor manera posible, ya que existen metodologías a seguir, las cuales nos van a servir de guía para realizar varias actividades y así lograr los objetivos planteados.

Realizar una virtualización de sistemas nos permite crear un escenario la más parecido a la realidad, este se realiza con el objetivo de recrear todos los aspectos posibles para obtener una mejor recopilación de información, y utilizar las herramientas de escaneo en un ambiente totalmente controlado.

Utilizar varias herramientas de escaneo de vulnerabilidades nos ofrece la posibilidad de realizar un análisis comparativo, el cual nos ayudara a elegir cual es la mejor y cuál de ellas nos ofrece la información más adecuada para realizar el respectivo estudio.

La ventaja de la presente investigación es que mediante el resultado de los análisis se entregará un informe con recomendaciones de mitigación que ayude a reforzar los niveles de seguridad del portal web del Gobierno Autónomo Descentralizado y de esta manera aumentar la defensa contra los diferentes tipos de ataques informáticos, salvaguardando la confidencialidad, integridad y disponibilidad del portal web.

El presente proyecto está direccionado al plan toda una vida, haciendo énfasis en el eje 2, el cual detalla lo siguiente:

Eje 2: Economía al servicio de la sociedad. [16]

Objetivo 5: Impulsar la productividad y competitividad para el crecimiento económico sostenible de manera redistributiva y solidaria. [16]

Política 5.6: Promover la investigación, la formación, la capacitación, el desarrollo y la transferencia tecnológica, la innovación y el emprendimiento, la protección de la propiedad intelectual, para impulsar el cambio de la matriz productiva mediante la vinculación entre el sector público, productivo y las universidades. [16]

1.5 ALCANCE

En el presente trabajo de investigación, se realizará un análisis de vulnerabilidades, mediante el uso del hacking ético, para encontrar posibles vulnerabilidades de seguridad

en el portal web de un Gobierno Autónomo Descentralizado de la Provincia de Santa Elena, con ayuda de herramientas preseleccionadas, con el fin de elaborar un informe final que brinde recomendaciones de mitigación frente a los resultados encontrados.

El presente proyecto abarcará las siguientes fases:

- Recolección de Información
- Virtualización de Sistemas
- Análisis de Vulnerabilidades
- Análisis de Resultados
- Informe final

Para esto se plantea realizar una investigación que permita incluir los puntos más importantes sobre el hacking ético en portales web, tales como: conceptos básicos, técnicas, metodologías, tipos de instrucciones, haciendo reseña a los ataques más frecuentes que actualmente se presentan y causen mayor impacto para estas instituciones, de esta forma tener una mejor dirección al análisis de vulnerabilidades que se va a realizar en el portal web. Se debe recalcar que solo se llegara a la fase 2 del hacking ético, la cual corresponde al escaneo y posterior análisis de vulnerabilidades.

Después de este estudio, se debe seleccionar de manera objetiva las herramientas de hacking ético que se van a usar en las respectivas pruebas, para detectar las vulnerabilidades de seguridad.

Una vez identificadas las herramientas a usarse, se va a recrear y definir un escenario de investigación, el cual nos ofrecerá realizar las pruebas necesarias en un ambiente totalmente controlado.

Como siguiente paso, se va a realizar la ejecución de las herramientas de análisis hacia el portal web, siguiendo el proceso de evaluación de seguridad de la información de la metodología ISSAF, la cual consta de: “planeación y preparación, evaluación y presentación de informes”.

Una vez ejecutadas las herramientas, se obtendrán los resultados de las pruebas de análisis al portal web, con la finalidad de identificar y analizar las principales vulnerabilidades que actualmente está expuesto el sitio, las cuales comprometen la confidencialidad, integridad y disponibilidad de la información.

Los beneficiarios de esta investigación serán todos los que conformen el departamento de TICS de la institución, ya que, por medio del informe final, tendrán una guía para minimizar el riesgo de ataques

CAPÍTULO 2

2 MARCO TEÓRICO Y METODOLOGÍA

2.1 MARCO TEÓRICO

2.1.1 La importancia de identificar, analizar y evaluar vulnerabilidades

De acuerdo con los resultados de la encuesta realizada por ESET Latinoamérica para el documento ESET Security Report 2017, la explotación de vulnerabilidades se ha convertido en la mayor preocupación de las empresas en materia de seguridad, seguida de otros incidentes como infección por malware, fraudes, phishing o ataques de denegación de servicio (DoS). [17]

De esta manera, estos resultados cobran relevancia para mitigar los acontecimientos relacionados con la explotación de estas y también como un método para la aplicación de elementos relacionados a la seguridad ofensiva, la cual se realiza mediante los escáneres de vulnerabilidades

2.1.2 Estudios realizados por Kaspersky

Casi el 30% de las compañías tardan varios días en detectar eventos de Seguridad informática y solo un 8% cuenta con herramientas que pueden generar alertas ante este tipo de eventualidades. El 14% logra detectar el incidente en unas horas, 20% en un día, 19% en varias semanas y 7% en varios meses, según un estudio realizado por Kaspersky. [18]

El costo promedio de la recuperación de un solo incidente de seguridad está estimado en $86.5 mil USD para las pequeñas y medianas empresas y $861 mil USD para las grandes. [18]

Mientras más tiempo tarden en notar la filtración, más le costará a una empresa en términos monetarios y de información. Incluso cuando las filtraciones se detectan casi instantáneamente, las Pymes estiman un costo de $28 mil USD, que sube a $105 mil USD, si pasa inadvertida por más de una semana. Para las empresas, cuando cuentan con un sistema de detección, el daño financiero estimado es de $393 mil USD, que puede crecer hasta un millón si permanece inadvertido a lo largo de una semana. [18]

Tomando los resultados de los estudios realizados por los laboratorios de Kaspersky, podemos comprender la importancia de mantener conocimiento sobre la seguridad informática, la información se ha convertido en uno de los activos intangibles más valiosos del mundo, por lo tanto realizar escáneres de vulnerabilidades nos brinda la posibilidad de mantenernos preparados ante estas eventualidades, mediante la aplicación de recomendaciones y configuraciones los cuales nos ayudan a minimizar la explotación de estas, y de esta manera ahorrarnos miles de dólares en daños por este tipo de ataques.

2.1.3 Ciberseguridad riesgos, avances y el camino a seguir en américa latina y el caribe

La crisis propiciada a principios de 2020 por la pandemia del COVID-19 ha puesto de relieve nuestra dependencia de una infraestructura vital que, para la gran mayoría de los ciudadanos, resulta invisible o su existencia pasa prácticamente desapercibida. Nuestra vida diaria gira alrededor de actividades cada vez más digitalizadas y, por consiguiente, más sensibles a amenazas cibernéticas. Cadenas de suministro de alimentos, transporte, pagos y transacciones financieras, actividades educativas, trámites gubernamentales, servicios de emergencia, y el suministro de agua y energía, entre un sinnúmero de actividades, operan en la actualidad a través de tecnologías digitales [19]. Según lo expresado por el PhD Moisés Schwartz gerente de desarrollo del BID, podemos recalcar que mas actividades de nuestra vida cotidiana se enlazaron directa o indirectamente con alguna de las tecnologías de información y comunicación.

La región de América Latina y el Caribe aún no está suficientemente preparada para enfrentar los ataques que se producen en el ciberespacio. Únicamente 7 países de los 32 analizados en este reporte cuentan con un plan de protección de su infraestructura crítica, y 20 han establecido algún tipo de grupo de respuesta a incidentes, llamado CERT o CSIRT, según sus siglas en inglés. Esto limita la capacidad de identificar ataques y responder oportunamente a los mismos. [19]. Según el estudio realizado América Latina no se encuentra preparada en la parte de infraestructura, ni en la respuesta de incidentes de este tipo, por tal motivo el presente trabajo de investigación es crucial en lo referente a la seguridad de la información.

Este año en particular, la pandemia global de COVID-19 ha destacado el papel vital y el uso de las tecnologías de la información y la comunicación (TIC) en la prestación de servicios esenciales y su profunda integración en nuestras sociedades. [19]. Según lo expresado por la PhD Farah Urrutia Secretaria de Seguridad Multidimensional de la OEA, se analiza la evolución que debieron tener las tecnologías de información para brindarnos soluciones tecnológicas en los tiempos de pandemia.

La pandemia de COVID-19 nos brinda la oportunidad de reflexionar sobre el progreso en la expansión de las TIC, la conectividad a Internet y la ciberseguridad en el hemisferio. Nuestra mayor dependencia del ciberespacio durante la crisis subraya la necesidad de extraer lecciones para lo que nos espera en la transformación continua de nuestras sociedades y economías, y en garantizar la ciberseguridad a nivel mundial. [19]. Cabe recalcar que por motivos de la situación actual que atraviesa el mundo debido a la pandemia de COVID19, muchas de las actividades cotidianas tuvieron que adaptarse y pasar de lo presencial a la manera virtual. Por lo tanto, esto llevo a la expansión de la TIC a varios sectores nuevos por ende la seguridad de la información es crucial en estos momentos donde todo es virtualizado.

2.1.4 Kali Linux

Es una distribución basada en Debian GNU/Linux diseñada principalmente para la auditoría y seguridad informática en general. Fue fundada y es mantenida por Offensive Security Ltd. Mati Aharoni and Devon Kearns, ambos pertenecientes al equipo de Offensive Security, desarrollaron la distribución a partir de la reescritura de BackTrack,que se podría denominar como la antecesora de Kali Linux. [20]

2.1.5 Hacking Ético

Los medios sensacionalistas se han encargado en darle un mal significado a la palabra hacker, un significado que equipara a los hackers con criminales. Un hacker puede ser bueno o malo al igual que un abogado, un médico, un profesor o el oficio que fuera. El termino ethical hacking o hacking ético nace por tal motivo, era necesario aclarar que no es un hacking malo sino bueno, ético, pero en definitiva es hacking. [21]

2.1.6 Tipos de Ataques Informáticos

Malware.- El termino malware es la abreviatura de software malicioso, este término conlleva a todos los códigos maliciosos y programas los cuales tienen como objetivo es dañar un 20 sistema o aprovechar sus vulnerabilidades para dañar archivos importantes del sistema operativo con el fin de causar un mal funcionamiento. [22]

Virus.- Los virus informáticos son programas con un fin malicioso que tiene como objetivo replicarse y dañar otros archivos del sistema operativo con la intención de causar un daño o modificación para su mal funcionamiento. Estos programas se encuentran dentro ejecutables de programas conocidos, al dar click sobre él se pueden replicar dentro del sistema operativo y ya que se han dañado archivos importantes del sistema este programa infecta unidades de almacenamiento con el fin expandirse por este medio a los demás ordenadores. [22]

Gusanos.- Son programas que al momento de ejecutarse dentro del computador se realizan copias de este y una de las principales características es la propagación mediante la red de una empresa dejando a su paso computadores colapsados impidiendo trabajar a los usuarios. Los gusanos a diferencia de los virus no atacan archivos sino específicamente el rendimiento de sus ordenadores. [22]

Troyanos.-Un troyano es catalogado como un virus, aunque no cumple con todas las características de uno de ellos, pero por su forma de propagación y de actuar dentro del sistema fue catalogado como un virus [22]

Spyware.- Un spyware es un programa espía que se mantiene en escucha para poder obtener todo tipo de información sobre una persona u organización sin que el usuario tenga conocimiento de que este programa se encuentra ejecutándose en segundo plano, estos programas espías tienen como objetivo ingresar en empresas publicitarias u organización especifica en busca de información con alto interés, este programa envía información periódica a un servidor en donde se aloja esta información para posterior el atacante pueda filtrar esta información y verificar que nivel de impacto tiene para la empresa [22]

AdWare.- Este programa malicioso no genera un daño al ordenador, pero genera una saturación de publicidad de productos y servicios de diferentes empresas, el objetivo es

poder obtener clientes mediante la publicidad en ventanas emergentes, o a través de herramientas que se incrustan dentro del navegador y al mantenerse ejecutando genera un número grande de publicidad mientras se navega por la web [22]

Ransomware .- Este código malicioso ingresa dentro de un ordenador en donde se encuentra información importante y vital para una organización y lo cifra, el atacante da una serie de instrucciones para que el usuario pueda recuperar su activo más importante la información. Para que el atacante pueda dar la contraseña para poder desencriptar el ordenador pide un rescate económico, de esta manera obtiene un redito económico por este tipo de ataque. [22]

Phishing.- El phishing es uno de los métodos más utilizados por los atacantes para poder obtener información por medio del correo electrónico utilizando ingeniería social, haciéndose pasar por empresas con renombre o personas de confianza, los atacantes utilizan todo tipo de maneras para que los usuarios caigan en su ataque, empezando con él envió de correos electrónicos hasta tener una llamada para poder obtener información personal de la víctima [22]

2.1.7 Seguridad Informática

El propósito de la seguridad en todos sus ámbitos de aplicación es reducir riesgos hasta un nivel que sea aceptable para los interesados en mitigar amenazas latentes. En un sentido amplio, por seguridad también se entienden todas aquellas actividades encaminadas a proteger de algún tipo de peligro [23]

2.1.8 Análisis de Vulnerabilidades

Las vulnerabilidades informáticas son consideradas como riesgos informáticos y son características de un activo de información. Las vulnerabilidades informáticas pueden ser detectadas con un análisis de vulnerabilidad y test de intrusión. Cuando se materializa un riesgo informático que pueda ser explotado, hay una posibilidad de ocurrencia de cualquier tipo de daño relacionado con la confidencialidad, integridad, disponibilidad y autenticidad de los datos empresariales. [24]

2.1.9 Análisis de Amenazas y Vulnerabilidades para Identificar Riesgos

Cuando se habla de ciberseguridad, el análisis de riesgos informáticos es la evaluación de los distintos peligros que afectan a nivel informático y que pueden producir situaciones de amenaza al negocio, como robos o intrusiones que comprometan los datos o ataques externos que impidan el funcionamiento de los sistemas propiciando periodos de inactividad empresarial. [25]

2.1.10 Herramientas de Análisis de Vulnerabilidades

¿Son realmente útiles y eficientes las herramientas para escanear vulnerabilidades?

Sí, son muy útiles y eficientes. No pueden ser reemplazadas por un pentest, pero pueden aportar varios puntos útiles a un entorno. Por ejemplo, permite el análisis constante del entorno en busca de vulnerabilidades recientemente descubiertas, indicando prioridad en relación con lo que debe tratarse en función de la criticidad de la vulnerabilidad, además de tener el enfoque que necesita la empresa, ya que al ser administrada por el responsable de la seguridad de la misma, quien interactúa constantemente con el ambiente, este puede definir en qué entorno se realizarán más pruebas. [26]

2.2 METODOLOGÍA DEL PROYECTO

2.2.1 Metodología de la Investigación

Los estudios exploratorios se efectúan cuando no se han realizado investigaciones previas o existe poca información acerca del objeto de estudio [27]. La presente investigación no se ha realizado en los Gobiernos Autónomos de la Provincia de Santa Elena, puntualmente en los portales web, ya que a pesar de que existe un departamento de Tics no se ha invertido en seguridad informática.

2.2.2 Modalidad Bibliográfica

El presente trabajo de investigación es basado en información que es posible encontrar en libros, informes, artículos, investigaciones similares, los cuales facilitarán información relevante para llevar a cabo la misma.

2.2.3 Recolección de la información

Para la recolección de información se ha utilizado la indagación con trabajos relacionados al tema de vulnerabilidades de portales web. Mediante investigaciones con información bibliográfica especializada.

2.2.4 Metodología de Desarrollo

Metodología Marco de Evaluación de Seguridad del Sistema de Información (ISSAF)

En la Investigación de la Universidad Tecnológica de Pereira se menciona, "la metodología de pruebas de intrusión ISSAF está diseñada para evaluar la red, sistemas y aplicaciones. Esta metodología se enfoca en 3 fases: Fase I - Planeación y preparación, Fase II – Evaluación y Fase III - Presentación de Informes" [28]. Basado en los lineamientos de la metodología, el proyecto se enfocará en tres fases descritas a continuación:

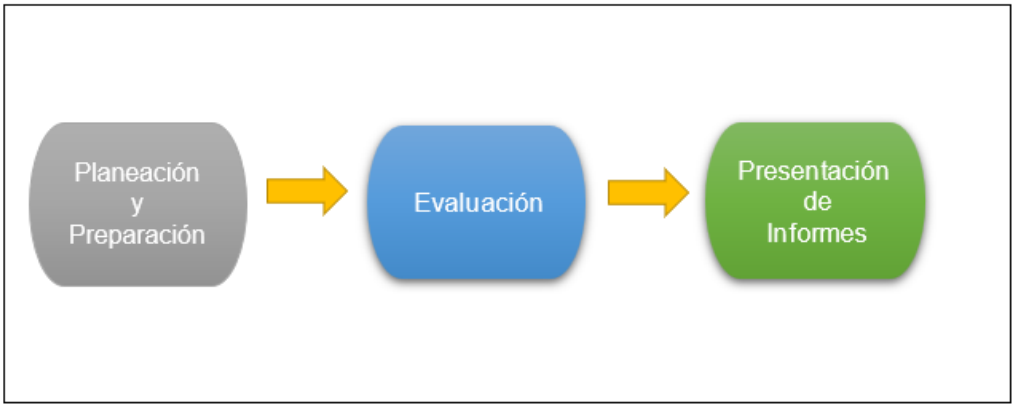


Ilustración 3 Fases de la Metodología ISSAF

2.2.5 FASE I: Planeación y Preparación

En esta primera fase procedemos con el intercambio de información inicial, planificar y preparar todo para los procesos de prueba. Antes de esto, se puede firmar un acuerdo formal entre ambas partes, (Empresa y auditor de seguridad), para proveer un mecanismo básico de protección legal en caso de ser necesario. También se debe especificar la persona a cargo, las fechas exactas, los tiempos de la prueba y otras evaluaciones.

2.2.6 Preparación de Escenario

Para realizar este escenario haremos uso del reconocimiento pasivo, ya que no conocemos la información básica del servidor, ni donde está alojado el portal web al que queremos realizar el análisis de las vulnerabilidades.

Para esto se utilizará una laptop con Windows 10 home, para realizar un levantamiento de información mediante el uso de un navegador web.

2.2.7 Recopilación de Información

a) Uso de un navegador web para verificar la existencia de información vulnerable

Existen elementos directos e indirectos que son obtenidos de manera fácil, mediante el uso de un navegador si estos no tienen una configuración correcta. Los elementos directos engloban los índices de búsqueda, mientras que los indirectos tiene que ver con la información sensible que puede ser usada para una explotación.

Objetivo

Entender el diseño y configuración del portal web, sus aplicaciones y si este tipo de información se encuentra expuesta directa o indirectamente. Se realiza una búsqueda

compuesta de site: “nombre de la página web” para encontrar las subpáginas que contiene.

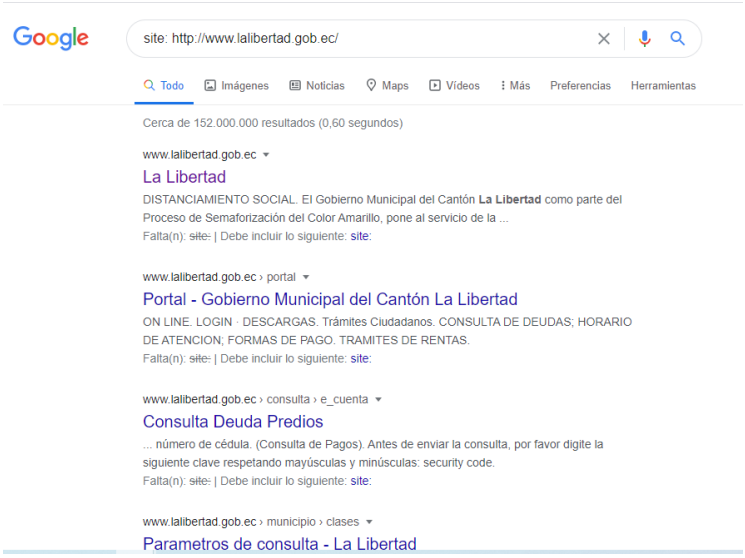


Ilustración 4 Resultados de la búsqueda del sitio web en Google

Entre los primeros datos se obtienen páginas web que pueden ser de utilidad para posteriores explotaciones como son

Ilustración 5 Sitio web interno para consultas

Se puede apreciar en la siguiente captura de pantalla una dirección que lleva hacia un ingreso con número de cédula para realizar una consulta.

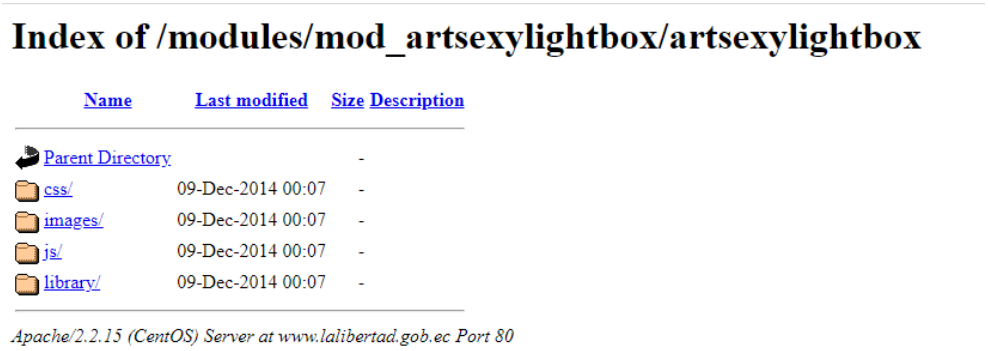


Ilustración 6 Url del sitio web con posible información sensible

Se encontró una dirección que puede resultar peligrosa al contener información sensible Con lo cual se puede concluir que existe un proceso el cual se debería corregir, dando una pauta a un atacante para focalizar su intrusión.

b) Análisis del sitio y servidor web para verificar nombres por defecto

El conocer el portal web que va a ser analizado, es una etapa crucial de un auditor informático, ya que en esta etapa se encuentra toda la información necesaria, así como el tipo de servidor en donde se encuentra alojado el portal, incluida la versión de este.

Objetivo

Conocer la IP del portal web que será analizado incluido el servidor que lo maneja y su versión.

En esta fase se hará uso de la herramienta Whois, la cual proporciona información básica del portal web a analizar de manera fácil mediante el uso de un navegador web.

Metodo: FOOTPRINTING	**Observaciones: Descubrimiento Pasivo**
Herramientas: Google, Whois,	Visibilidad:Outsider

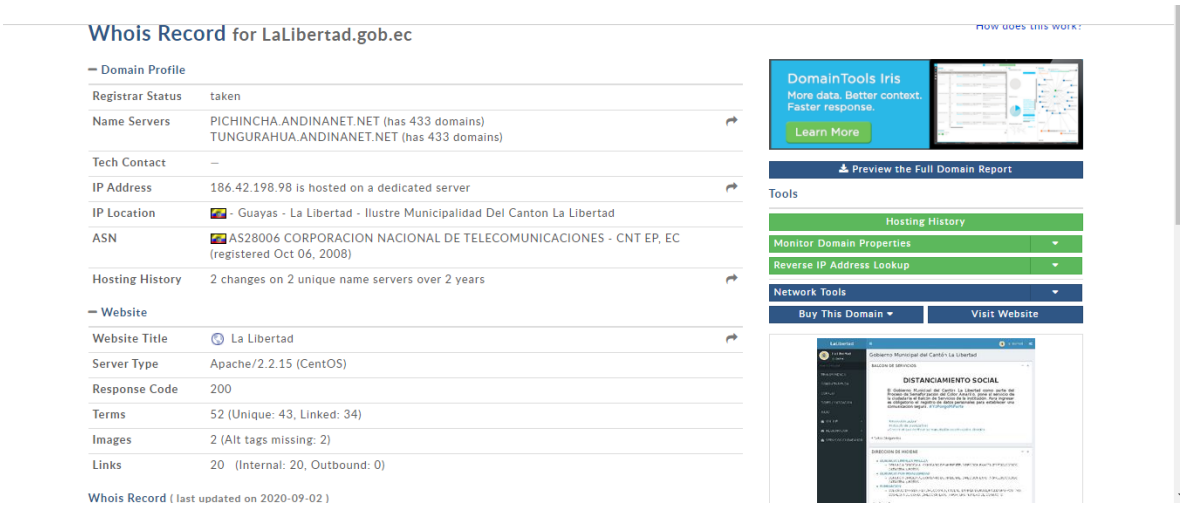


Ilustración 7 Información proporcionada por Whois

Resultados Obtenidos con Whois	
IP address:	186.42.198.98
IP Location:	Guayas – La Libertad
Proveedor:	CNT
Servidor:	Apache 2.2.15
Sistema Operativo:	CentOS

Tabla 1 Resultados Obtenidos con Whois

Al conocer la IP de la página o aplicación web específica se procede a realizar una consulta, mediante el comando Whois pero ahora en Kali Linux.

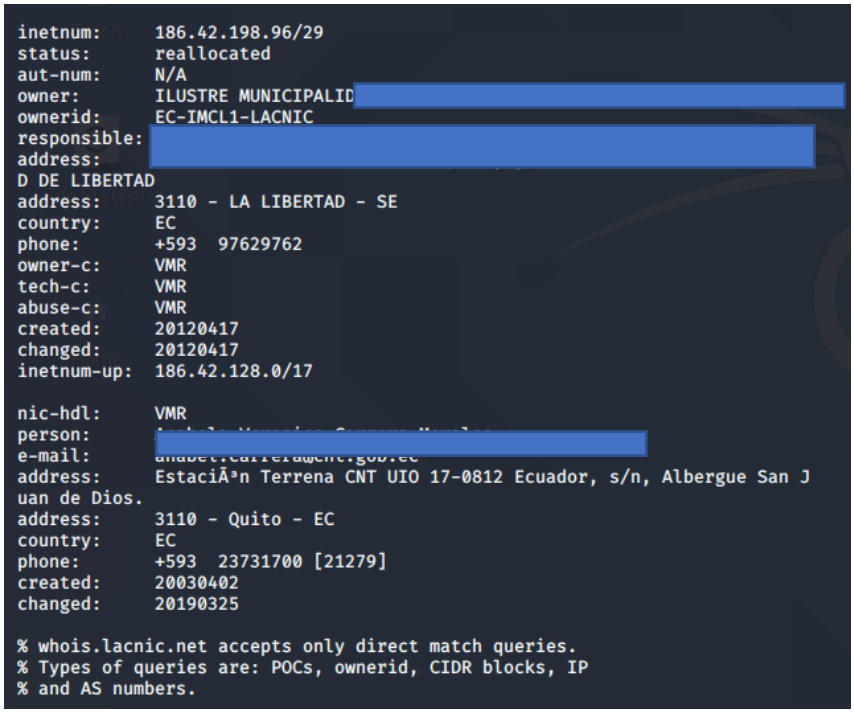


Ilustración 8 Respuesta de Whois en Kali Linux

Metodo: FOOTPRINTING	Observaciones: Descubrimiento Pasivo
Herramientas: Google, Shodan	Visibilidad:Outsider

Shodan tiene como objetivo el ubicar a todo tipo de dispositivos que estén conectados a Internet, es decir, desde routers, APs, dispositivos IoT hasta cámaras de seguridad. Con este buscador podremos obtener información de manera más profunda acerca de los servicios disponibles como por ejemplo el software del servidor, opciones de los servicios, puertos disponibles, información super importante en este aspecto para una posible intrusión futura.

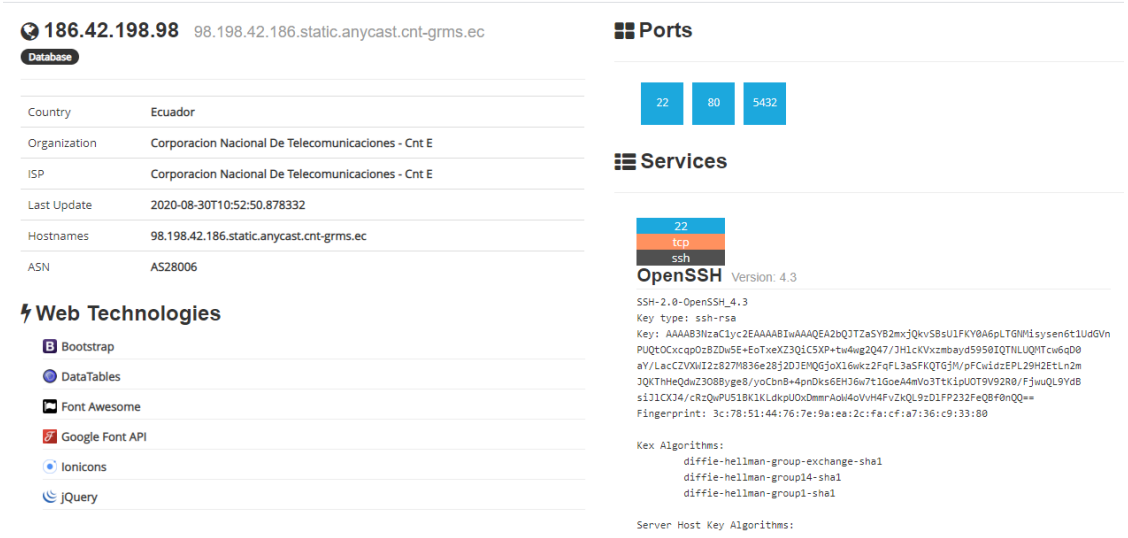


Ilustración 9 Resultados de la herramienta Shodan

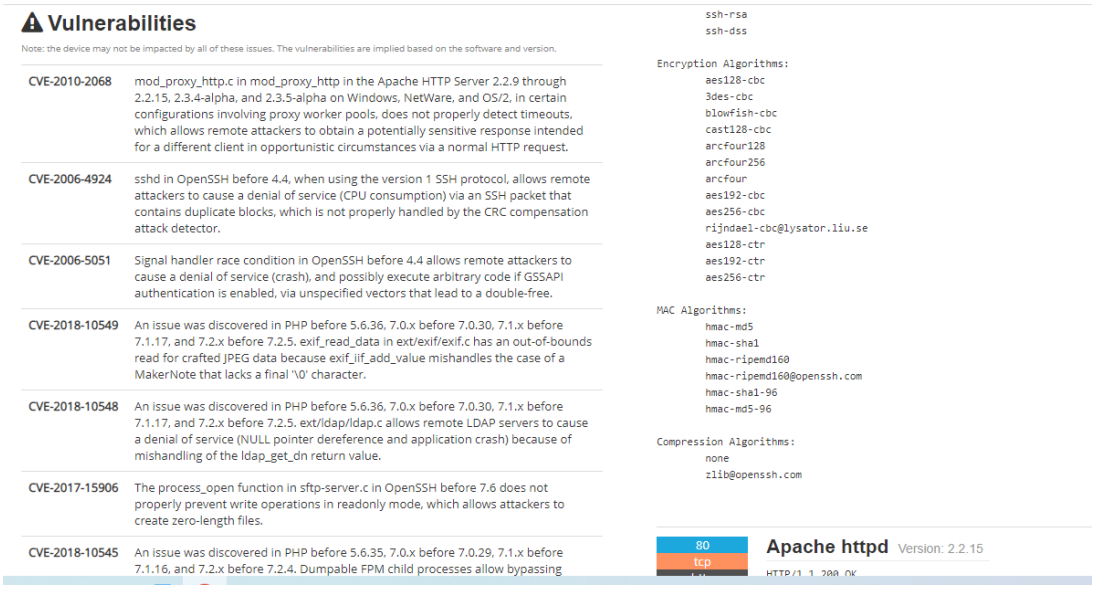


Ilustración 10 Resultados de la Herramienta Shodan

Ilustración 11 Resultados de la Herramienta Shodan

Resultados Obtenidos con Shodan	
IP address:	186.42.198.98
IP Location:	Guayas – La Libertad
Ultima actualización:	30/08/20
Hostnames:	98.198.42.186.static.anycast.cnt-grms.ec
Sistema Operativo:	CentOS
Servidor:	Apache http 2.2.15
Servidor de datos:	PostgreSQL 9.3.18
Servicios:	OpenSSH 4.3
Puertos:	22 – 80 - 5432
Tecnologías Web:	• Bootstrap • Data Tables • Font Awesome • Google Font Api • Ionicons • JQuery
Vulnerabilidades:	75

Tabla 2 Resultados Obtenidos con Shodan

Con las consultas antes realizadas se obtuvo información muy útil para los fines pertinentes a un análisis de vulnerabilidades, tales como:

- Direcciones IP de los servidores
- Direcciones de los proveedores
- Nombre del servidor
- Herramientas de funcionamiento y sus versiones
- Contactos de proveedores y administradores
- Sistema Operativo del servidor

c) Recolección de metadatos del portal web para la comprobación de existencia de información vulnerable

En este paso se analiza los archivos que puedan encontrarse en el portal web, así como carpetas existentes y así verificar sí existe alguna posibilidad de acceder a información sensible.

Objetivo: Analizar los metadatos obtenidos del portal web utilizando la Herramienta Foca

Metodo: FOOTPRINTING	**Observaciones: Descubrimiento Pasivo**
Herramientas: Foca	Visibilidad:Outsider

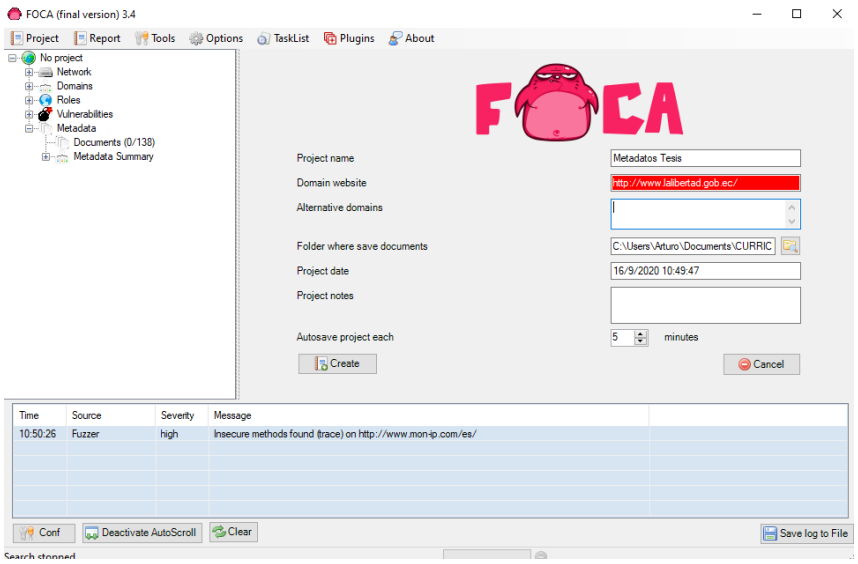


Ilustración 12 Inicio de Extracción de archivos del portal

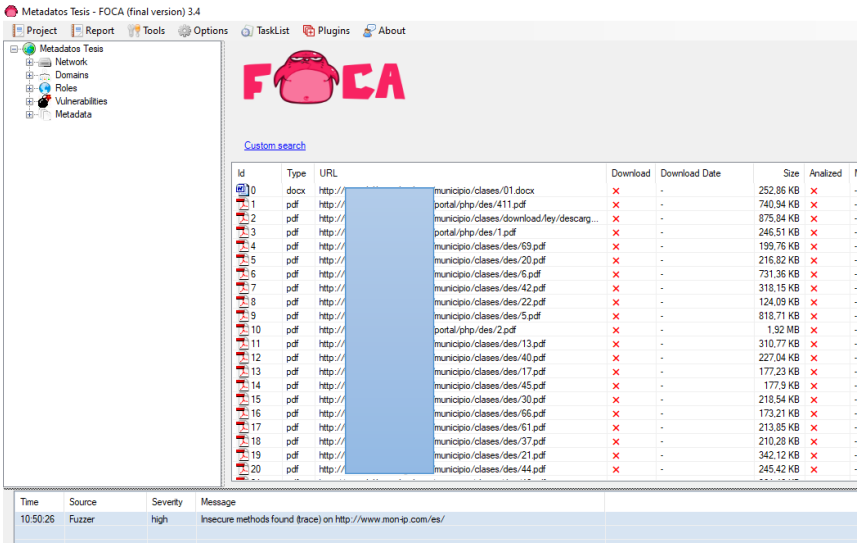


Ilustración 13 Archivos encontrados en el portal web

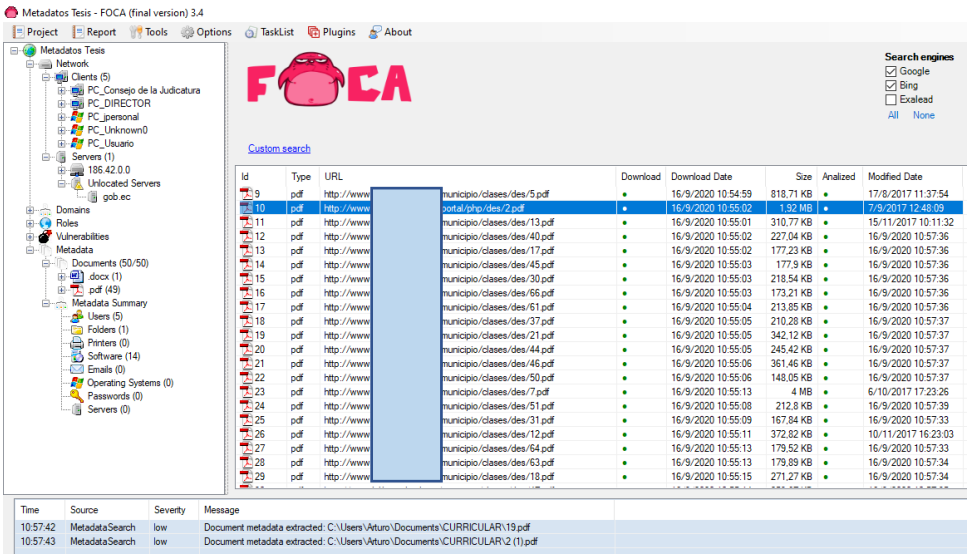


Ilustración 14 Extracción de metadatos

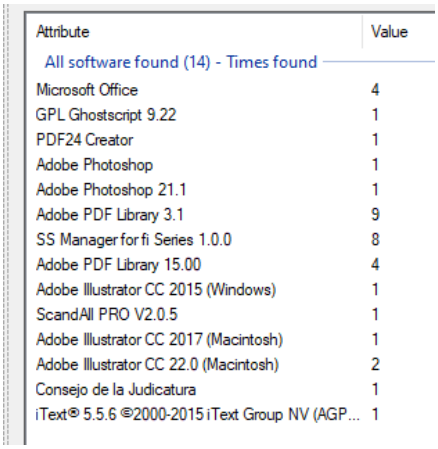


Ilustración 15 Software encontrados

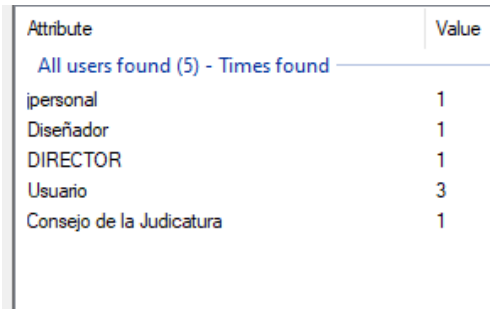


Ilustración 16 Nombres de usuarios

Resultados Obtenidos con Foca	
Archivos encontrados:	50
Archivos .doc:	1
Archivos pdf:	49
Servidores:	1
Usuarios:	5
Nombres de los usuarios:	• Ipersonal • Diseñador • Director • Usuario • Concejo de la judicatura
Softwares utilizados:	14
Nombres del software:	• Microsoft Office • GLP Ghooscript 9.22 • Pdf 24 Creator • Adobe Photoshop • Adobe Photoshop 21.1 • Adobe Pdf Library 3.1 • SS Manager • Adobe Pdf Library 15 • Adobe Illustrator cc 2015 (Windows) • ScandAll pro v2.0.5 • Adobe Illustrator cc 2017 (Macintosh) • Adobe Illustrator cc 22.0 (Macintosh) • Concejo de la Judicatura • Itext 5.5.6

Tabla 3 Resultados Obtenidos con Foca

2.2.8 Virtualización de Sistemas

Para este paso crearemos una máquina virtual con el sistema operativo Kali Linux y también instalaremos la herramienta Nessus en una maquina con Windows 10 home.

Objetivo: Instalar la herramienta Nessus en Windows 10 home de 64 bits

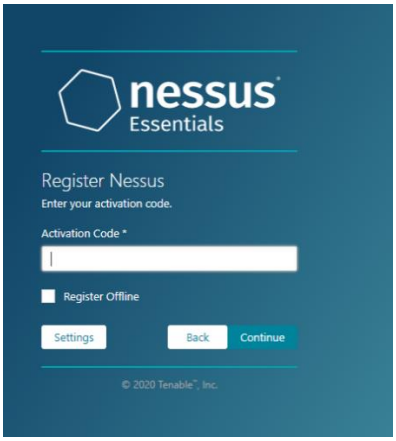


Ilustración 17 Licencia de la Herramienta

Después de registrarnos procedemos a obtener una licencia para el uso de la herramienta mediante el ingreso de nuestro coreo electrónico

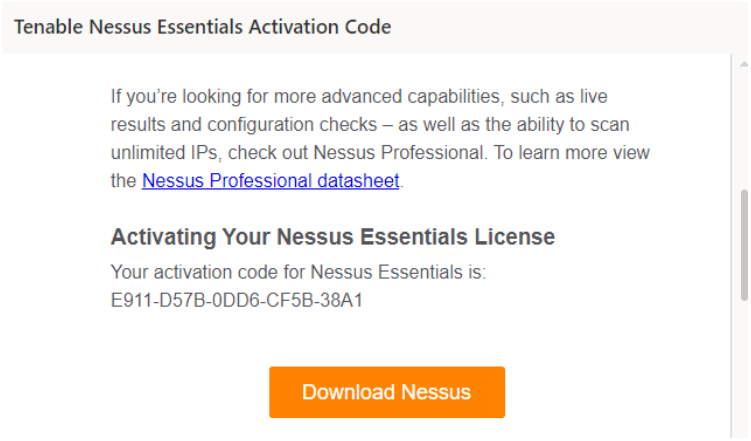


Ilustración 18 Licencia enviada por correo

Ilustración 19 Descargas de Plugin necesarios

Se procederá a descargar plugin necesarios para el correcto funcionamiento de la herramienta, en tal caso ocurra un error en la descarga se procede mediante comando utilizando el cmd de Windows.

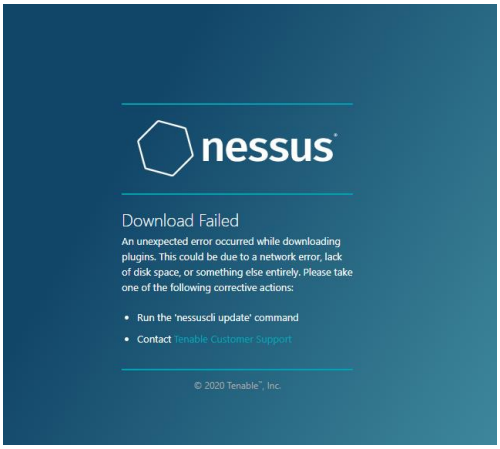


Ilustración 20 Error en la descarga

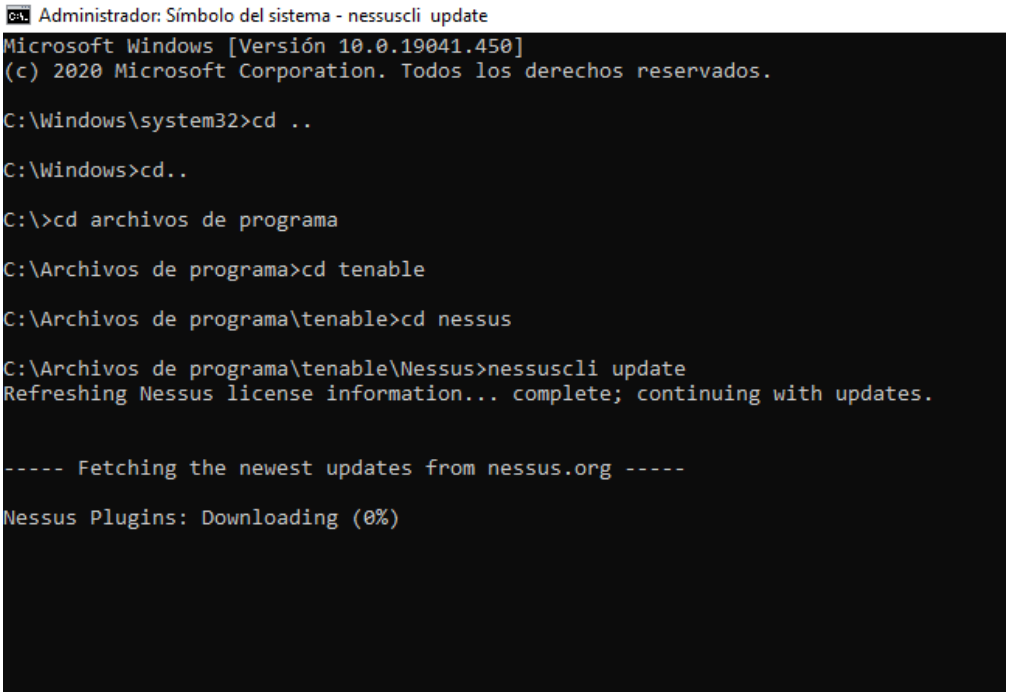


Ilustración 21 Descarga de Plugin mediante Cmd

Para este paso debemos ejecutar el cmd de Windows como administrador, navegamos hasta la raíz donde el programa fue instalado y digitamos el siguiente comando "*nessuscli update*"

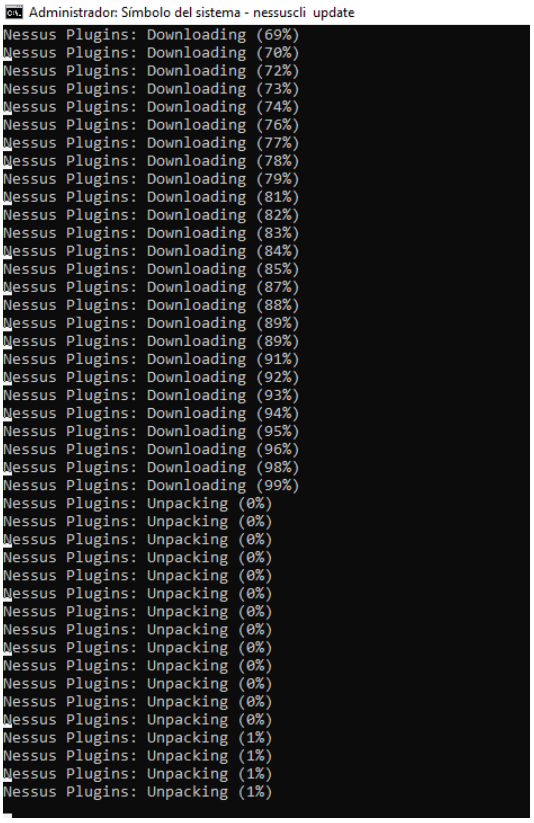


Ilustración 22 Proceso de descarga

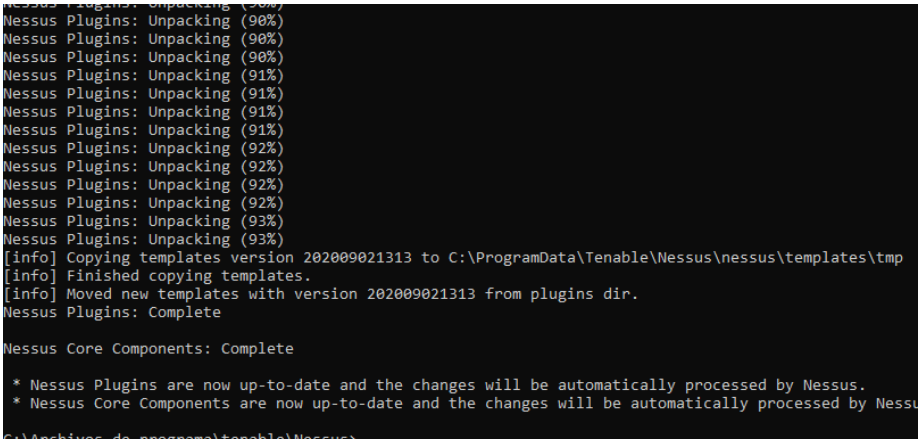


Ilustración 23 Descarga terminada

Con la descarga de estos complementos tendremos la herramienta lista para poder utilizarla en Windows 10

Virtualización de Kali Linux

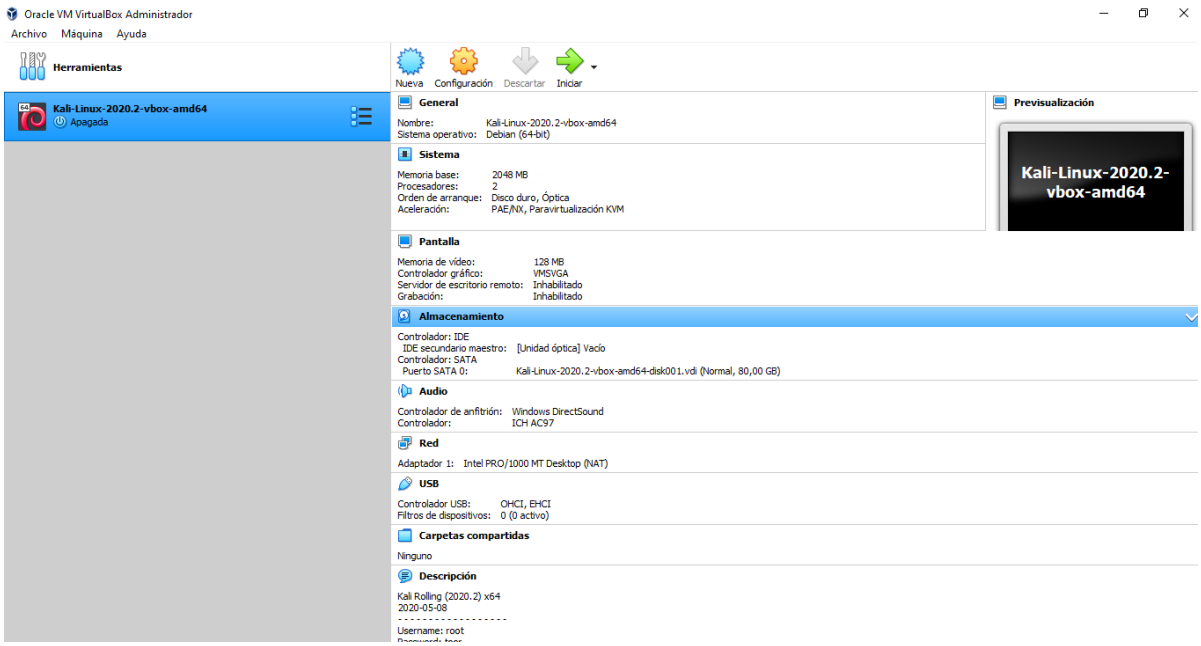


Ilustración 24 Virtualización de Kali Linux en VirtualBox

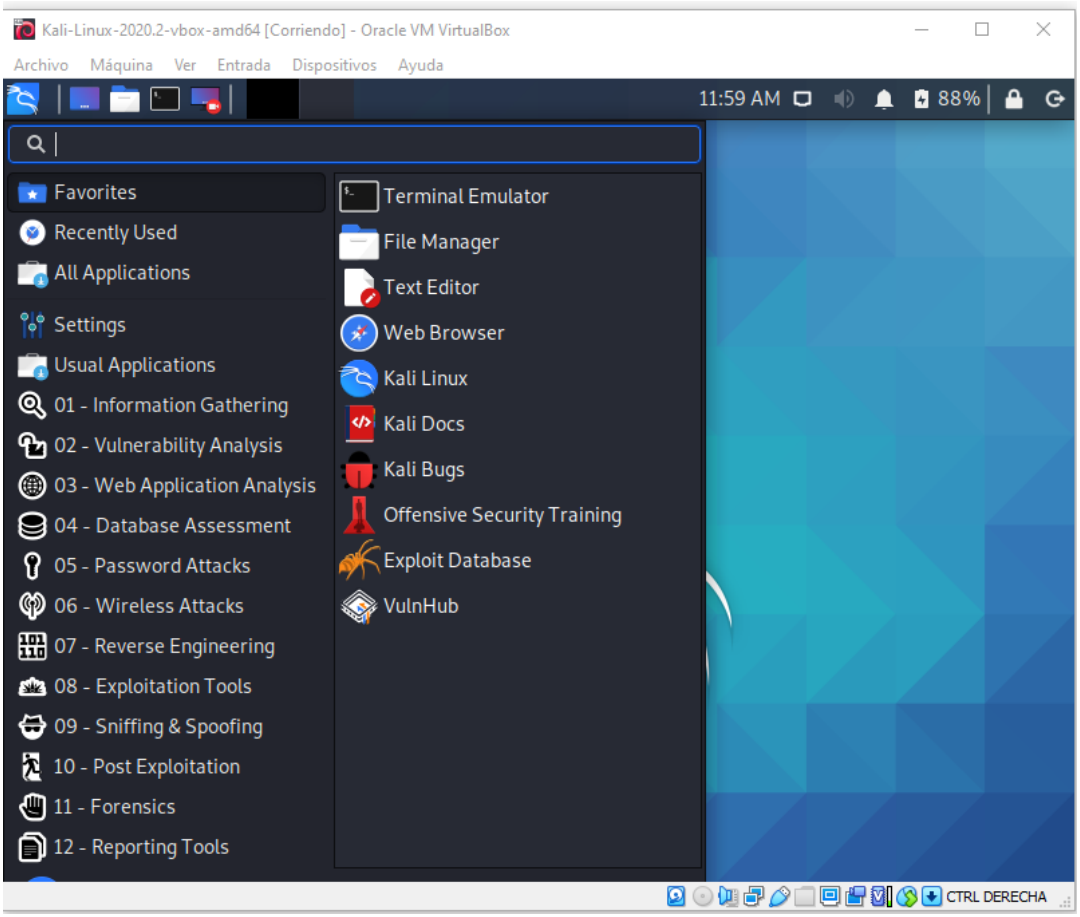


Ilustración 25 Virtualización de Kali Linux

2.2.9 FASE 2: Evaluación

En esta fase es en realidad en la que se va a llevar a cabo las pruebas pertinentes. Esta fase aplica un enfoque por capas, en donde cada capa representa un mayor nivel de acceso a los activos de la información. Las capas son las siguientes: Recopilación de Información, Mapeo de la Red, Identificación de vulnerabilidades, Intrusión, Ganando acceso y escalando privilegios, Enumeración adicional, Comprometiendo usuarios/sitios remotos, Manteniendo acceso y Cubriendo rastros [28]. Solo se llegará a la parte de identificación de vulnerabilidades para después realizar el respectivo informe detallado.

2.2.10 Análisis de Vulnerabilidades

Ejecución de herramientas

Nessus

Se utilizará la herramienta Nessus, un potente analizador de vulnerabilidades que será ejecutado en una maquina con Windows 10 home.

Objetivo: Realizar el análisis de vulnerabilidades con la herramienta Nessus

Ilustración 26 Inicio del escaneo con Nessus

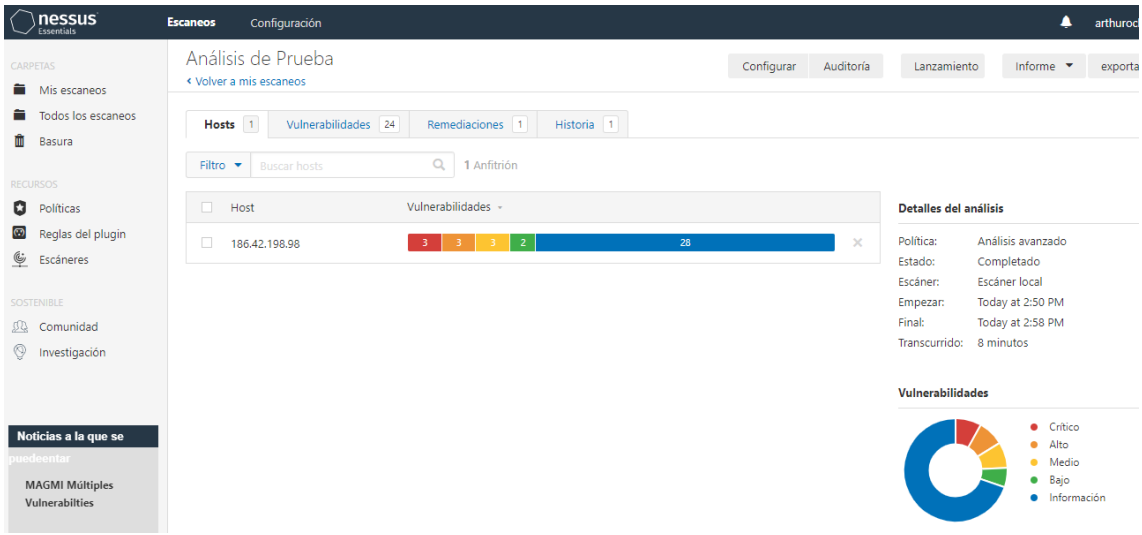


Ilustración 27 Resultados de la herramienta Nessus

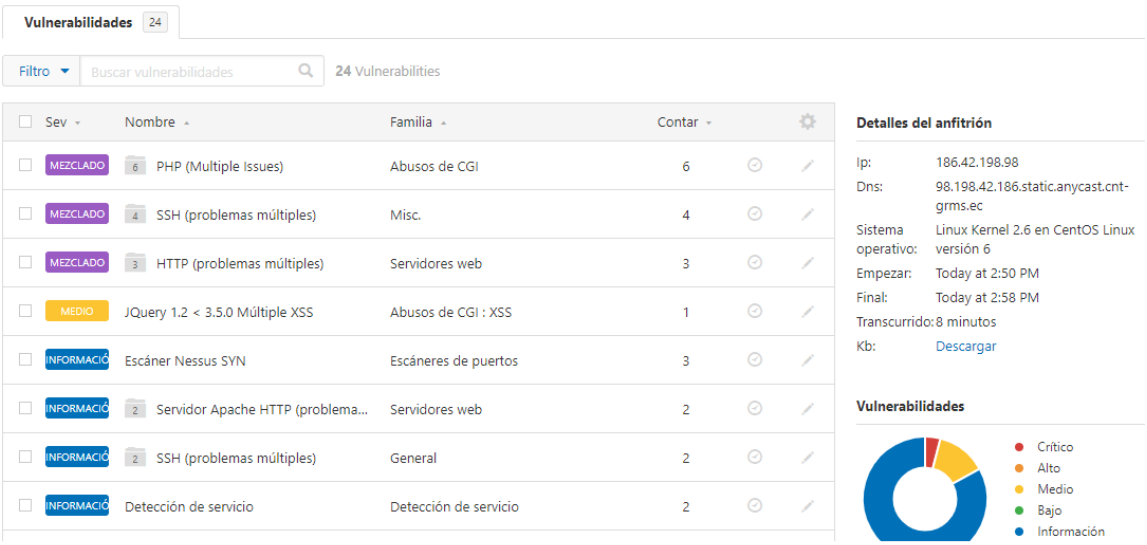


Ilustración 28 Vulnerabilidades encontradas

Resultados Obtenidos con Nessus	
Vulnerabilidades encontradas:	11
Información adicional:	13
Total:	23
Tipo de Análisis:	Análisis Avanzado
Tiempo de inicio:	14:50 Pm
Tiempo final:	14:58 Pm
Duración total:	8 minutos
Vulnerabilidades Criticas	3
Vulnerabilidades Altas	3
Vulnerabilidades Medias	3
Vulnerabilidades Bajas	2

Tabla 4 Resultados Obtenidos con Nessus

Detalles de Vulnerabilidades Criticas

Detalle de Vulnerabilidades	
Fecha de Publicación:	25/06/2015
Fecha de Modificación:	25/11/2019
Nombre:	CVE-2015-2325 - CVE-2015-2326 - CVE-2015-3414 - CVE-2015-3415 - CVE-2015-3416
Importancia:	Critica
Recursos Afectados:	Apache HTTP Server, versiones 5.4 – 5.4.42
Detalle:	• Un atacante remoto puede aprovechar estas condiciones para provocar un desbordamiento de búfer basado en montón, lo que resulta en una condición de denegación de servicio o la ejecución de código arbitrario. • Existe una vulnerabilidad de denegación de servicio en el componente SQLite incluido debido al manejo incorrecto de las cotizaciones en los nombres de secuencia de intercalación. • Existe una vulnerabilidad de inyección de comandos arbitraria debido a un defecto en la función php_escape_shell_arg() en exec.c.
Recomendación:	Actualice a PHP versión 5.4.42 o posterior.

Tabla 5 Detalle de Vulnerabilidades 1

Detalle de Vulnerabilidades	
Fecha de Publicación:	10/07/2015
Fecha de Modificación:	27/03/2019
Nombre:	CVE-2015-3152 - CVE-2015-5589 - CVE-2015-5590 - CVE-2015-8838
Importancia:	Critica
Recursos Afectados:	Apache HTTP Server, versiones 5.4 – 5.4.42
Detalle:	

	• Un atacante man-in-the-middle puede explotar este defecto para obligar al cliente a degradar a una conexión sin cifrar, lo que permite al atacante revelar datos de la base de datos o manipular consultas de base de datos. • Un atacante puede explotar esto para bloquear una aplicación PHP, lo que resulta en una condición de denegación de servicio. • Un atacante remoto puede aprovechar esto para des referenciar la memoria ya liberada, lo que podría dar lugar a la ejecución de código arbitrario
Recomendación:	Actualice a PHP versión 5.4.43 o posterior.

Tabla 6 Detalle de Vulnerabilidades 2

Detalle de Vulnerabilidades	
Fecha de Publicación:	4/05/2012
Fecha de Modificación:	25/08/2020
Importancia:	Critica
Recursos Afectados:	Apache HTTP Server, versiones 5.4 – 5.4.42
Detalle:	• Según su versión, la instalación de PHP en el host remoto ya no es compatible. • La falta de soporte implica que el proveedor no lanzará nuevos parches de seguridad para el producto. • Como resultado, es probable que contenga vulnerabilidades de seguridad.
Recomendación:	Actualice a una versión de PHP que sea compatible actualmente.

Tabla 7 Detalle de Vulnerabilidades 3

Detalles de Vulnerabilidades Altas

Detalle de Vulnerabilidades	
Fecha de Publicación:	18/05/2015
Fecha de Modificación:	27/03/2019
Nombre:	CVE-2015-2325 - CVE-2015-2326 - CVE-2015-4021 - CVE-2015-4022 - CVE-2006-7243 - CVE-2015-4025 - CVE-2015-4024
Importancia:	Alta
Recursos Afectados:	Apache HTTP Server, versiones 5.4 – 5.4.41
Detalle:	• Un defecto en la función de phar_parse_tarfile en ext/phar/tar.c podría permitir una denegación de servicio a través de una entrada hecha a mano en un archivo tar.

	• Un atacante remoto puede aprovechar esto para provocar un desbordamiento de búfer basado en montón, lo que resulta en una condición de denegación de servicio o una posible ejecución remota de código • Un atacante remoto puede explotar estos defectos, mediante la combinación del carácter '-0' con una extensión de archivo segura, para eludir las restricciones de acceso
Recomendación:	Actualice a PHP versión 5.4.41 o posterior.

Tabla 8 Detalle de Vulnerabilidades 4

Detalle de Vulnerabilidades	
Fecha de Publicación:	11/08/2015
Fecha de Modificación:	22/11/2019
Nombre:	CVE-2015-6831 - CVE-2015-6832 - CVE-2015-6833 - CVE-2015-8867
Importancia:	Alta
Recursos Afectados:	Apache HTTP Server, versiones 5.4
Detalle:	• Existe una vulnerabilidad de uso después de la liberación en ext/spl/spl_array.c debido al manejo incorrecto de un dato serializado especialmente diseñado. • Existe una vulnerabilidad de recorrido de directorio en la clase PharData, debido a la implementación incorrecta de la función exctractTo. • Un atacante remoto no autenticado puede explotar esto a través de una entrada de archivo ZIP diseñada para escribir en archivos arbitrarios.
Recomendación:	Actualice a PHP versión 5.4.44 o posterior.

Tabla 9 Detalle de Vulnerabilidades 5

Detalle de Vulnerabilidades	
Fecha de Publicación:	10/09/2015
Fecha de Modificación:	22/11/2019
Nombre:	CVE-2014-9767 - CVE-2015-6834 - CVE-2015-6835 - CVE-2015-6836 - CVE-2015-6837, CVE-2015-6838
Importancia:	Alta
Recursos Afectados:	Apache HTTP Server, versiones 5.4
Detalle:	• Una vulnerabilidad de recorrido de directorio en la función ZipArchive::extractTo en ext/zip/php_zip.c podría permitir a un atacante remoto crear directorios vacíos arbitrarios a través de un archivo ZIP creado • Existen varios errores de memoria de uso después de la liberación relacionados con la función unserialize().

	Un atacante remoto puede explotar estos errores para ejecutar código arbitrario. • Existen varios defectos en la clase XSLTProcessor debido a la validación incorrecta de la entrada de la biblioteca libxslt. • Un atacante remoto puede explotar estos defectos para tener un impacto no especificado
Recomendación:	Actualice a PHP versión 5.4.45 o posterior.

Tabla 10 Detalle de Vulnerabilidades 6

Detalles de Vulnerabilidad Media

Detalle de Vulnerabilidades	
Fecha de Publicación:	23/01/2003
Fecha de Modificación:	12/06/2020
Nombre:	CVE-2004-2320
Importancia:	Media
Recursos Afectados:	HTTP TRACE / TRACK Methods Allowed
Detalle:	• Las funciones de depuración están habilitadas en el servidor web remoto • El servidor web remoto admite los métodos TRACE y/o TRACK. TRACE y TRACK son métodos HTTP que se utilizan para depurar conexiones de servidor web.
Recomendación:	Deshabilite estos métodos HTTP. Consulte la salida del plugin para obtener más información.

Tabla 11 Detalle de Vulnerabilidades 7

Detalle de Vulnerabilidades	
Fecha de Publicación:	28/05/2020
Fecha de Modificación:	08/09/2020
Nombre:	CVE-2020-11022
Importancia:	Media
Recursos Afectados:	JQuery 1.2 < 3.5.0 Múltiple XSS
Detalle:	• El servidor web remoto se ve afectado por la vulnerabilidad de scripting entre sitios múltiple • Según la versión auto informada en el script, la versión de JQuery alojada en el servidor web remoto es mayor o igual que 1.2 y anterior a 3.5.0. Por lo tanto, se ve afectado por múltiples vulnerabilidades de scripting entre sitios.
Recomendación:	Actualice a JQuery versión 3.5.0 o posterior.

Tabla 12 Detalle de Vulnerabilidades 8

Detalle de Vulnerabilidades	
Fecha de Publicación:	04/04/2016
Fecha de Modificación:	14/12/2016
Nombre:	Algoritmos débiles SSH soportados
Importancia:	Media
Recursos Afectados:	Algoritmos débiles SSH soportados
Detalle:	• El servidor SSH remoto está configurado para permitir algoritmos de cifrado débiles o ningún algoritmo en absoluto.
Recomendación:	Póngase en contacto con el proveedor o consulte la documentación del producto para eliminar los cifrados débiles.

Tabla 13 Detalle de Vulnerabilidades 9

Detalle de Vulnerabilidad Baja

Detalle de Vulnerabilidades	
Fecha de Publicación:	28/10/2013
Fecha de Modificación:	30/07/2018
Nombre:	Cifrados de modo CBC del servidor SSH habilitados
Importancia:	Bajo
Recursos Afectados:	servidor SSH
Detalle:	• El servidor SSH está configurado para utilizar Cipher Block Chaining • Esto puede permitir a un atacante recuperar el mensaje de texto no cifrado del texto cifrado
Recomendación:	Póngase en contacto con el proveedor o consulte la documentación del producto para deshabilitar el cifrado de cifrado en modo CBC y habilite el cifrado de modo de cifrado CTR o GCM.

Tabla 14 Detalle de Vulnerabilidades 10

Detalle de Vulnerabilidades	
Fecha de Publicación:	22/11/2013
Fecha de Modificación:	14/12/2016
Nombre:	SSH Débiles algoritmos MAC habilitados
Importancia:	Bajo
Recursos Afectados:	SSH -MAC
Detalle:	• El servidor SSH remoto está configurado para permitir algoritmos MD5 y MAC de 96 bits. • El servidor SSH remoto se configura para permitir los algoritmos MD5 o MAC de 96 bits, ambos de los cuales se consideran débiles

Recomendación:	Póngase en contacto con el proveedor o consulte la documentación del producto para deshabilitar los algoritmos MD5 y MAC de 96 bitsCTR o GCM.

Tabla 15 Detalle de Vulnerabilidades 11

Uniscan

Objetivo: Utilizar la herramienta Uniscan en Kali Linux para la detección de vulnerabilidades

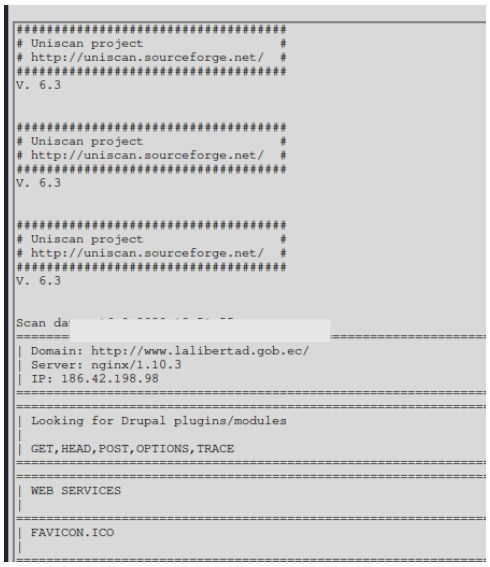


Ilustración 29 Escaneo mediante Uniscan

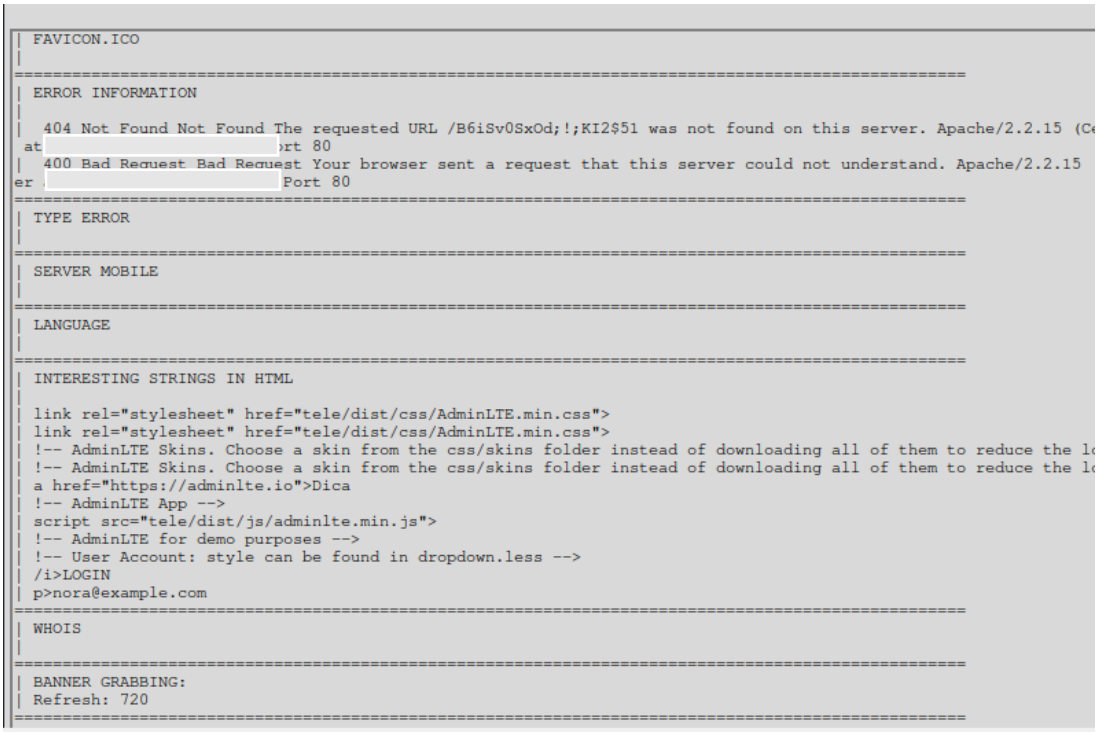


Ilustración 30 Resultados de Uniscan

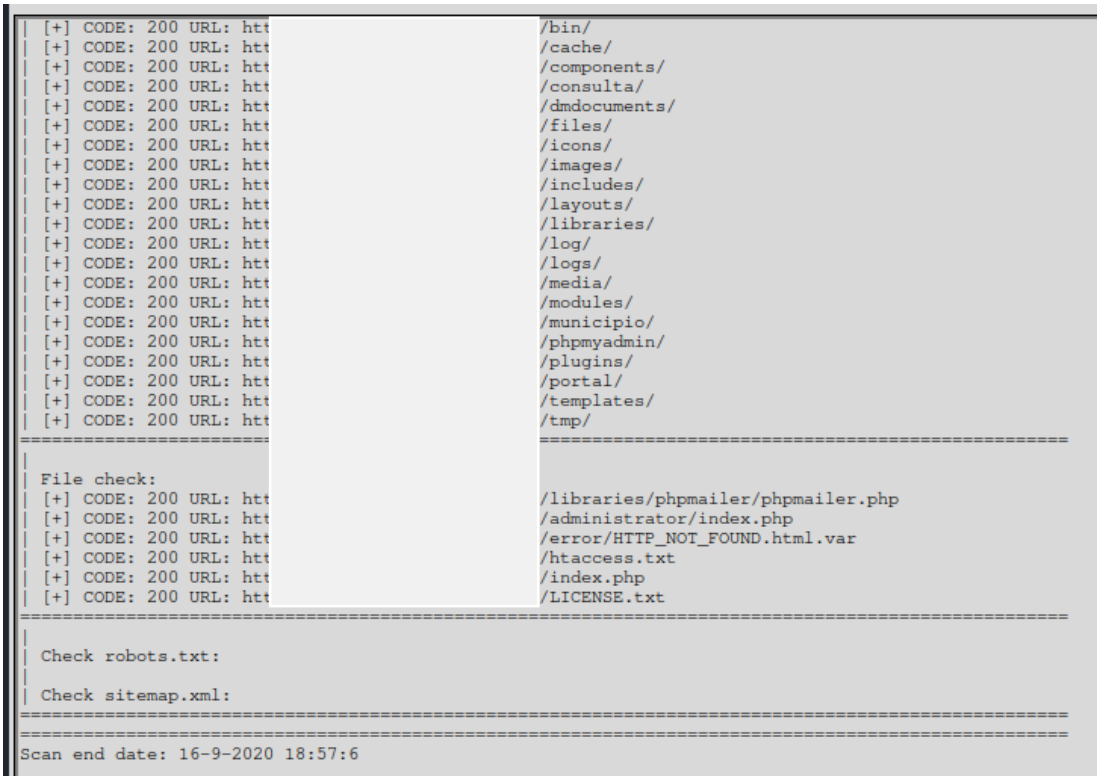


Ilustración 31 Resultados de Uniscan

Resultados Obtenidos con Uniscan	
IP address:	186.42.198.98
Inicio:	18:51:55
Final:	18:57:06
Servidor:	Apache 2.2.15
Directorios:	• /bin • /caches • /component • /consulta • /dmdocument • /files • /icons • /images • /includes • /libraries • /logs • /media • /modules • /municipio • /phpmyadmin • /plugin • /portal • /templates • /temp
Archivos:	• Phpmailer.php • Index.php

	• htaccess.txt • license.txt

Tabla 16 Resultados Obtenidos con Uniscan

Maltego

Ilustración 32 Pantalla de inicio de Maltego

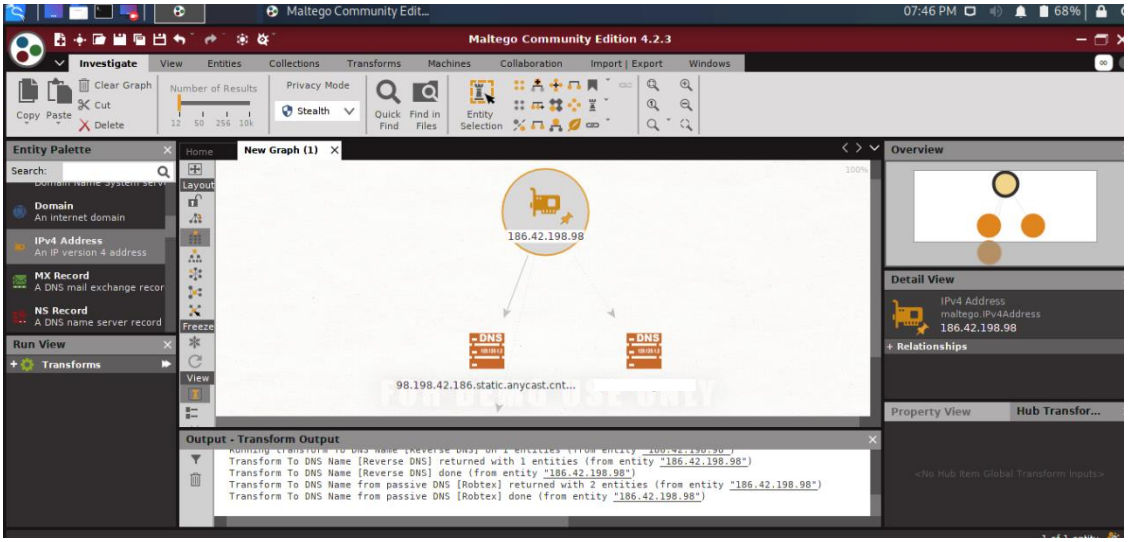


Ilustración 33 Inicio del análisis

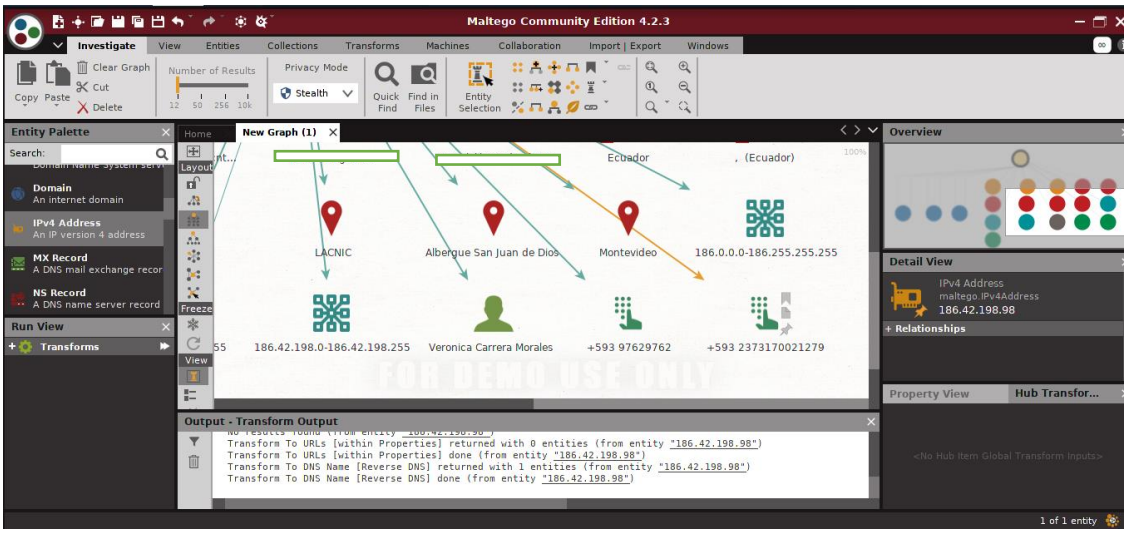


Ilustración 34 Resultados de las trasformaciones

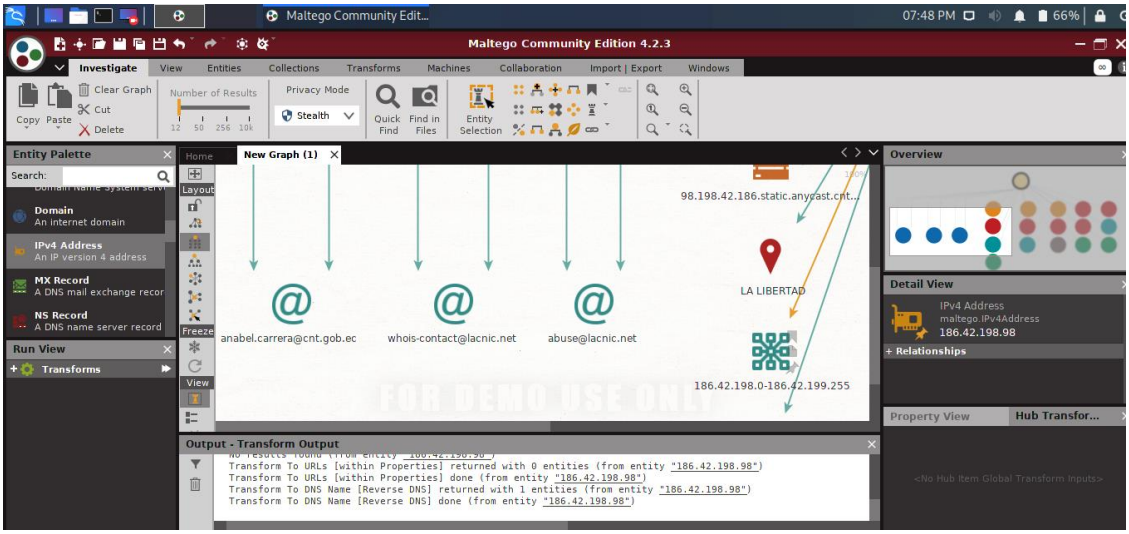


Ilustración 35 Resultados de las transformaciones

Resultados Obtenidos con Maltego	
IP address:	186.42.198.98
Inicio:	19:46
Final:	19:48
Servidor:	Apache 2.2.15
Contactos:	Verónica Carrera Morales
Teléfonos:	0997629762
Correos:	Anabel.carrera@cnt.gob.ec

Tabla 17 Resultados Obtenidos con Maltego

OwaspZap

Objetivo: Realizar el respectivo análisis de vulnerabilidades con la herramienta OwaspZap en un pc con Windows 10 home.

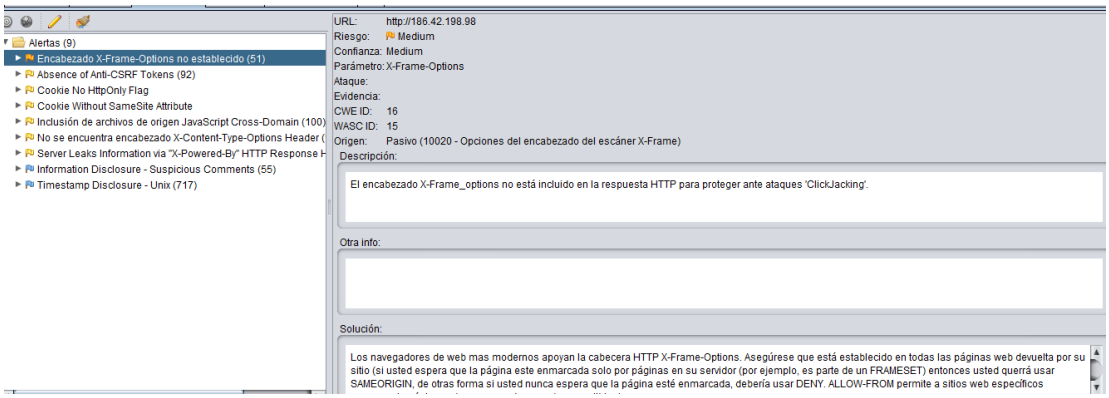


Ilustración 36 Resultado de OwaspZap

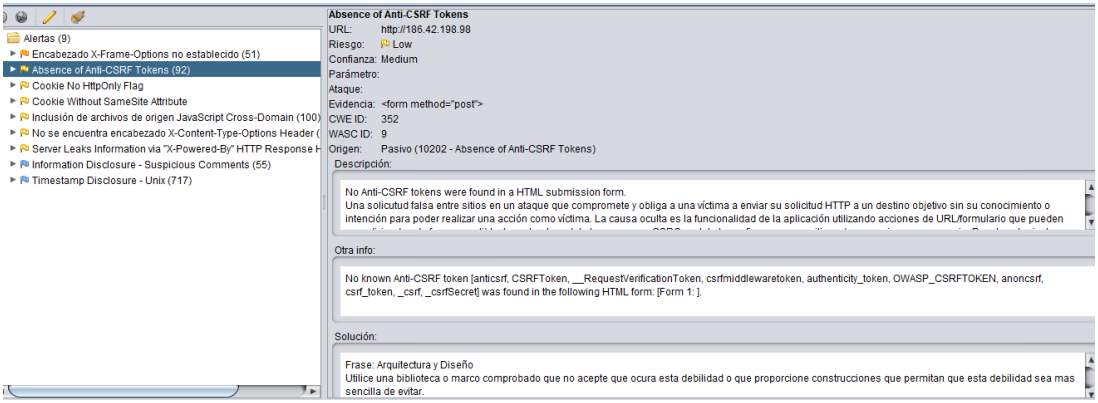


Ilustración 37 Resultados de OwaspZap

Resultados Obtenidos con OwaspZap	
Vulnerabilidades encontradas:	9
Tipo de Análisis:	Análisis Avanzado
Tiempo de inicio:	18:30 Pm
Tiempo final:	03:10 Am
Duración total:	9:50 minutos
Vulnerabilidades Criticas	0
Vulnerabilidades Altas	1
Vulnerabilidades Medias	6
Vulnerabilidades bajas	2

Tabla 18 Resultados Obtenidos con OwaspZap

Nikto

Objetivo: Realizar el uso de la herramienta Nikto en una máquina virtual con Kali Linux para la detección de vulnerabilidades.

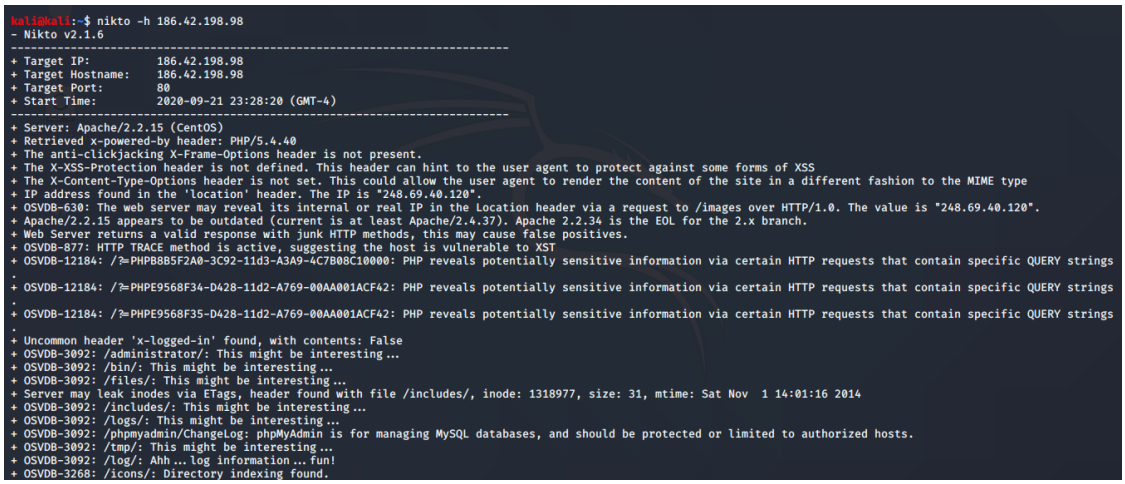


Ilustración 38 Inicio del escaneo con Nikto en Kali Linux

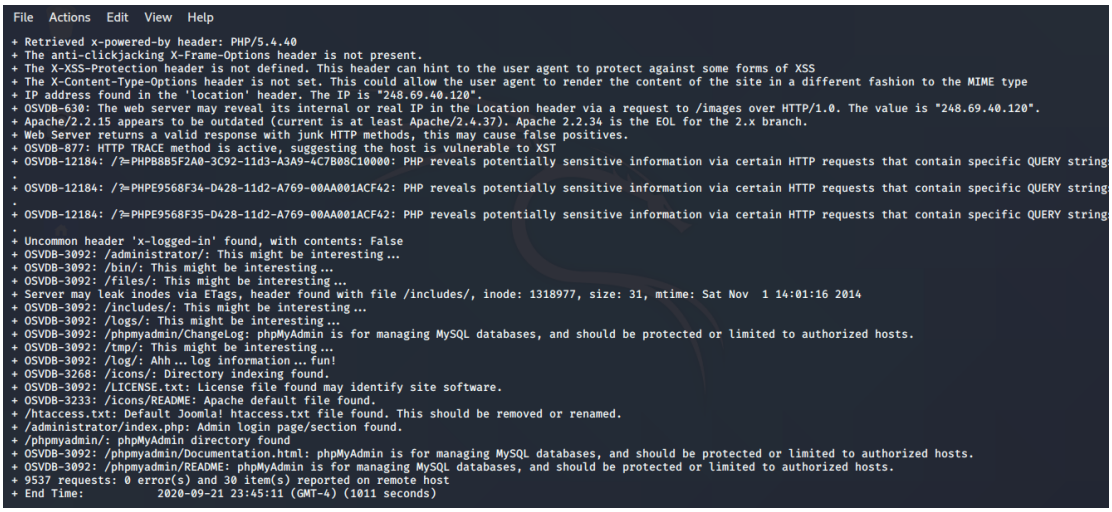


Ilustración 39 Resultado final de la herramienta Nikto

Resultados Obtenidos con Nikto	
Vulnerabilidades encontradas:	30
Tipo de Análisis:	Análisis Avanzado
Tiempo de inicio:	23:28:20 PM
Tiempo final:	23:45:11 PM
Duración total:	17 minutos
Servidor:	Apache/2.2.15
Vulnerabilidades Criticas:	-
Vulnerabilidades Altas	-
Vulnerabilidades Medias	-
Vulnerabilidades bajas	-

Tabla 19 Resultados Obtenidos con Nikto

2.2.11 Comparativa de herramientas

Se propone un enfoque para la detección de vulnerabilidades del portal web, con el fin de determinar la efectividad de dicho enfoque se utilizaron las siguientes herramientas en

cuestión para la detección de vulnerabilidades, finalmente se establece una comparativa entre los resultados obtenidos por estas herramientas.

Herramientas Utilizadas para la Comparativa	
1	Nessus (Windows)
2	Nikto (Kali Linux)
3	OwaspZap(Windows)

Tabla 20 Herramientas Utilizadas para la Comparativa

Los resultados han sido divididos en dos grupos, en primer lugar, se presentan aquellos resultados relacionados exclusivamente con la detección de vulnerabilidades haciendo énfasis en el análisis de vulnerabilidades de severidad alta; luego se dan a conocer aquellos relacionados con la comparación realizada entre las características de cada herramienta de detección.

Las siguientes graficas presentan el primer grupo de resultados. En cada una se presentan los resultados obtenidos por las 3 herramientas de detección de vulnerabilidades. Todas estas graficas contienen una gráfica de dos dimensiones en las que el eje Y representa el número de vulnerabilidades encontradas en la evaluación y el eje X el nivel de severidad determinado por cada herramienta. Es importante precisar en este punto que se ha establecido un mapeo entre el nivel de seguridad arrojado por cada herramienta y una escala definida por los autores con el fin de facilitar la comparación entre las herramientas, este se presenta en la siguiente tabla.

Nessus	Nikto	OwaspZap	Nueva Escala
Crítico, Alto	No definido	Alto	Alto
Medio	No definido	Medio	Medio
Bajo, Info	No definido	Bajo, Info	Bajo

Tabla 21 Calificación de herramientas

Mapeo entre las escalas de severidad utilizadas por las herramientas de evaluación y la escala definida por los autores

La tabla anterior indica entre otras cosas que la severidad Critico - Alto para Nessus, equivale a la Alta para OwaspZap, así como también en el apartado bajo – Info de Nessus es similar el criterio de calificación con OwaspZap. Mientras tanto Nikto no ofrece alguna clasificación, pero si brinda información de las vulnerabilidades de una manera general.

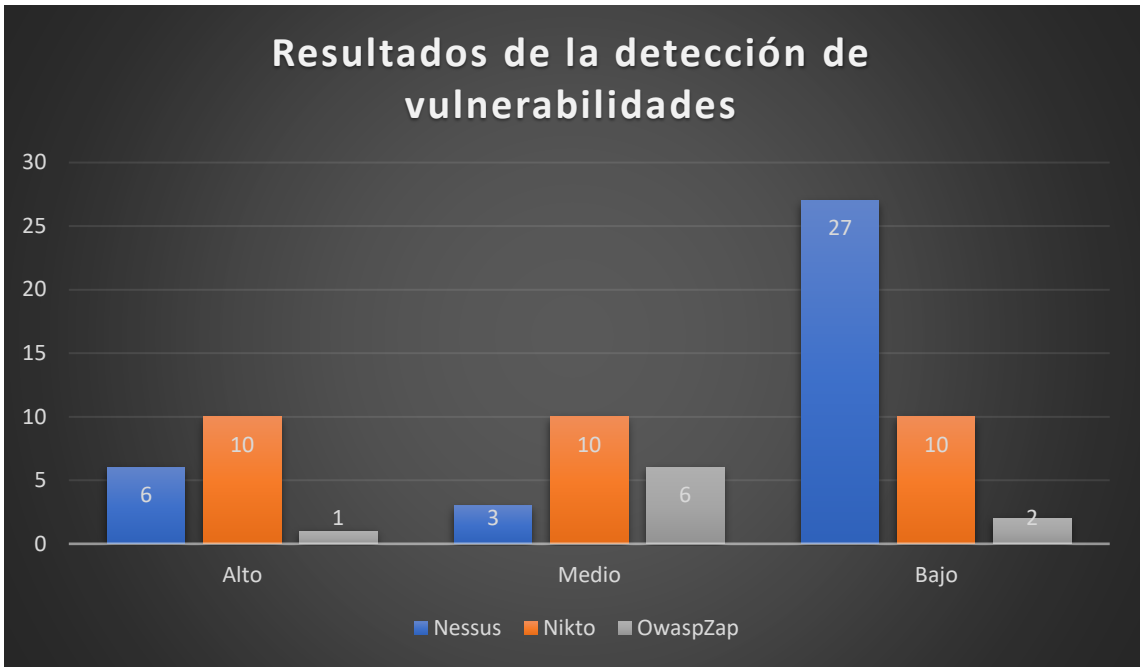


Ilustración 40 Comparativa de resultados entre las herramientas

Como se puede observar en la gráfica anterior, la herramienta que detecto más vulnerabilidades fue Nessus con un total de 36, seguido tenemos a Nikto con un total de 30 y por último tenemos a OwaspZap con 9. La herramienta OwaspZap fue la que presento un bajo número de vulnerabilidades, se comprobó que Nessus fue la que mayor número de vulnerabilidades detectó y estos resultados fueron detectados en su mayoría en los servicios de OpenSSH y Apache, con fallas a nivel de ejecución de código remoto y desbordamiento de memoria, mientras que Nikto también reporto vulnerabilidades que involucran al servicio OpenSSH, pero relacionados con la denegación de servicios y ejecución de código remoto. Los resultados de OwaspZap se enfocaron más en los servicios de PHP y los servicios de base de datos de MySQL.

Herrameinta / Parametro	Nessus	Nikto	Owasp Zap
Facilidad de Uso	Alta	Alta	Alta
Requerimiento del sistema	Bajo	Bajo	Medio
Facilidad de instalación	Medio	Alta	Bajo
Interfaz de Usuario	Alta	Baja	Medio
Complejidad	Baja	Media	Baja
N° de vulnerabilidades encontradas	36	30	9
Legibilidad del reporte final	Alta	Baja	Alta
Contenido del reporte	Alta	Medio	Alta
Velocidad de detección	Alta	Alta	Baja

Tabla 22 Indicadores para Comparativa

Comparación entre características de las herramientas de detección de vulnerabilidades

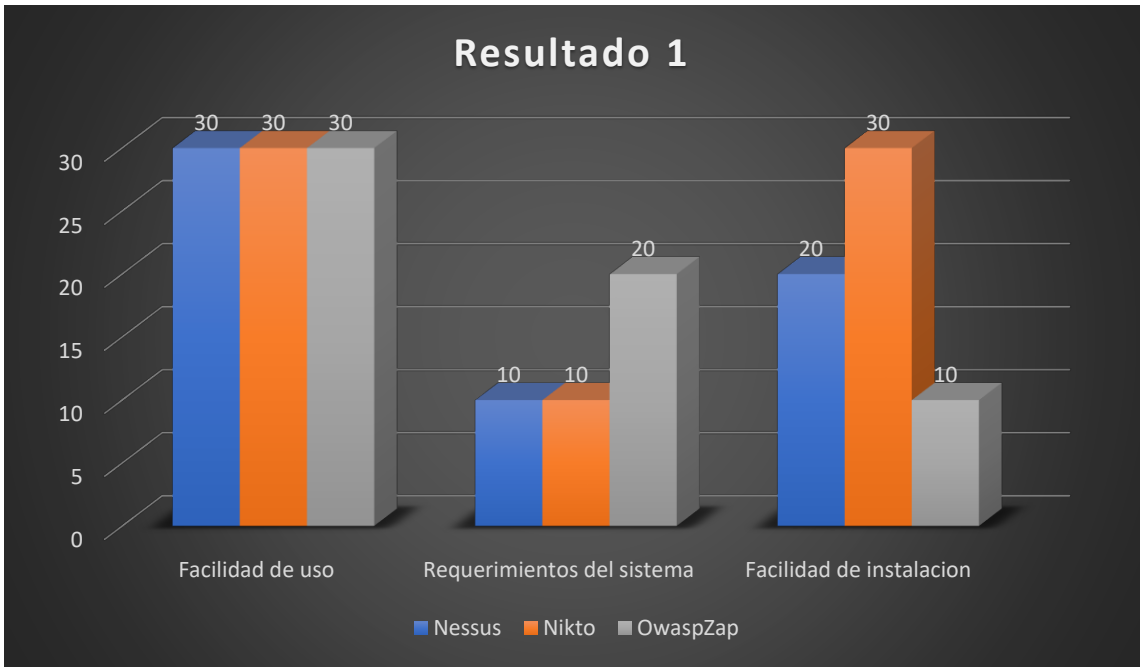


Ilustración 41 Resultados Comparativo 1

Como se puede apreciar en la gráfica anterior en lo que se refiere a facilidad de uso las 3 herramientas están en el mismo nivel, en requerimientos del sistema OwaspZap requiere un mejor hardware para su correcto funcionamiento, mientras que las dos herramientas restantes no tienen ningún inconveniente en su ejecución con un hardware básico. En el aspecto de instalación Nikto por estar ejecutado en Kali Linux si presenta alguna dificultad por que se necesitan algunas librerías y repositorios para su correcta ejecución.

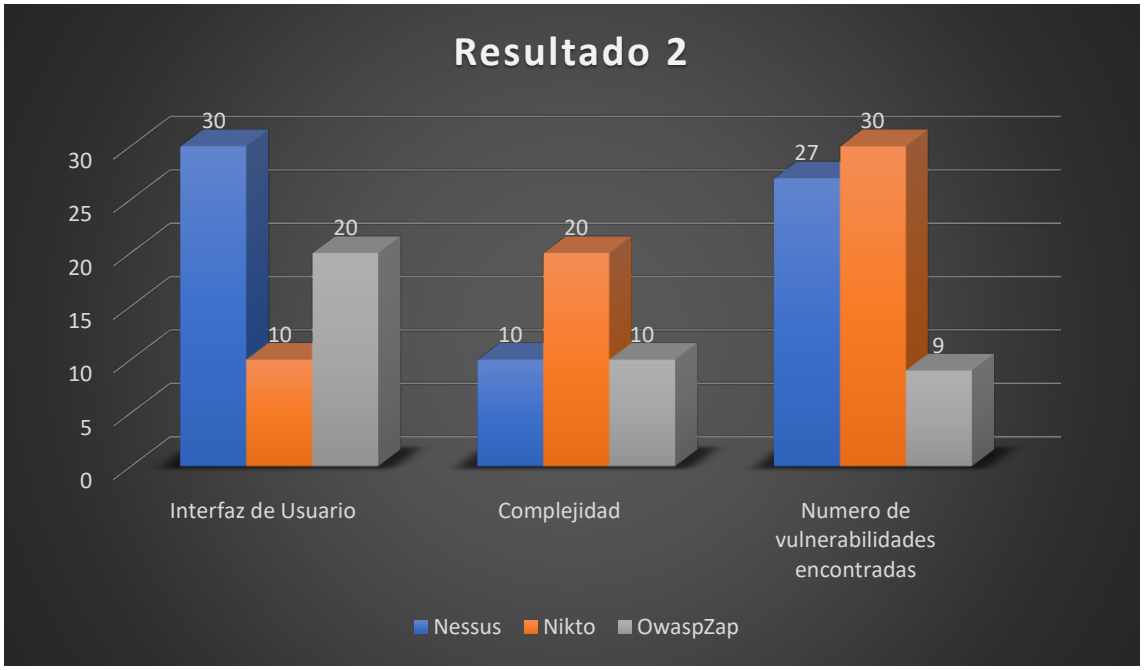


Ilustración 42 Resultados Comparativo 2

En el segundo resultado comparativo, en el primer aspecto obtenemos que Nessus tiene una mejor interfaz al usuario, quedando en último lugar Nikto debido a que su interfaz es mediante consola. En el aspecto de complejidad Nikto se lleva la delantera por el mismo aspecto que se mencionó anteriormente que es mediante consola, mientras que las dos herramientas restantes se mantienen en una complejidad baja. Por último, el número de vulnerabilidades encontradas se obtiene que Nessus está en primer lugar por sus 36 resultados, le sigue Nikto con 30 resultados y por último OwaspZap con 9 resultados obtenidos.

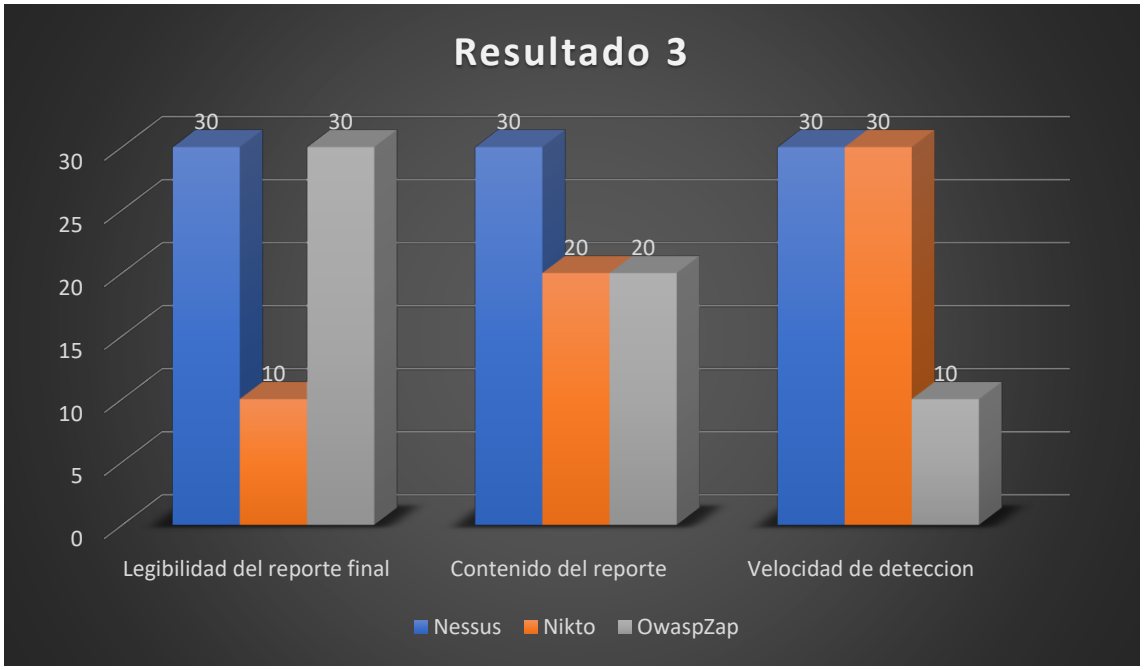


Ilustración 43 Resultado Comparativo 3

En el siguiente grafico se puede obtener que en el aspecto de Legibilidad del reporte final queda en último lugar Nikto debido a que sus resultados fueron de manera general sin clasificación alguna, mientras que Nessus queda en primer lugar por su excelente información y OwaspZap está en el mismo nivel que Nessus. En el aspecto de contenido del reporte se lleva la delantera Nessus por que ofrece una gran cantidad de información y detalles de las vulnerabilidades encontradas, en segundo lugar, queda OwaspZap debido a que también ofrece detalles sin llegar al nivel de Nessus, por último, queda Nikto que ofrece los resultados, pero no ofrece el nivel de detalles que las herramientas antes mencionadas. En el último aspecto al que se refiere a la velocidad del escaneo queda en primer lugar Nessus con aproximadamente 8 minutos, en segundo lugar, esta Nikto con 17 minutos y por último esta OwaspZap con aproximadamente 9 horas (depende del hardware necesario), las 3 herramientas fueron ejecutadas con el mismo hardware.

2.2.12 Resultados Obtenidos

Se obtuvo como resultado final de la comparativa, que la herramienta Nessus es la que lleva la delantera con respecto a tiempo de análisis, interfaz de usuario y contenido de reporte final y legibilidad de este, por lo tanto, sería la herramienta óptima para la detección de vulnerabilidades del portal web. Sin embargo, Nikto por su cantidad de resultados obtenidos sería la segunda opción con respecto a esta investigación, aunque su usabilidad podría ser un poco compleja.

2.2.13 Clasificación de los Resultados

Ilustración 44 Base de datos Incibe-cert

Con el objetivo de informar, advertir y ayudar a los profesionales sobre las ultimas vulnerabilidades de seguridad en sistemas tecnológicos, ponemos a disposición de los usuarios interesados en esta información una base de datos con información en castellano sobre cada una de las ultimas vulnerabilidades documentadas y conocidas. [29]

Este repositorio con más de 75.000 registros está basado en la información de NVD (http://nvd.nist.gov/) (National Vulnerability Database) – en función de un acuerdo de colaboración – por el cual desde INCIBE realizamos la traducción al castellano de la información incluida. En ocasiones este listado mostrará vulnerabilidades que aún no han sido traducidas debido a que se recogen en el transcurso del tiempo en el que el equipo de INCIBE realiza el proceso de traducción. [29]

La herramienta Nessus nos brindó resultados con la escala obtenida por los propios autores de la herramienta, las cuales fueron: Critico – Alto – Medio – Bajo e Información, esta información fue analizada y comparada con la base de datos incibe-cert para así poder clasificarla en 3 niveles: Alto, Medio y Bajo, los detalles de la clasificación se encuentran detallada en el Informe Final.

2.2.14 FASE 3: Presentación de Informes

Informe Verbal: "si en el transcurso de las pruebas de intrusión se encuentra una vulnerabilidad en el sistema, se debe informar inmediatamente a la organización para que sea consciente del problema" [28].

Informe Final: (Ascencio Mendoza & Moreno Patiño, 2011) menciona que "tras la finalización de todos los casos de prueba definidos en el alcance del trabajo, se debe hacer un informe escrito que describa los resultados de las pruebas con las recomendaciones de mejora respectivas". [28] .Ver Informe final de vulnerabilidades.

CAPÍTULO 3

3 PROPUESTA

En el siguiente trabajo de investigación se propuso realizar un análisis de vulnerabilidades, utilizando metodologías de hacking ético para un GAD municipal de la provincia de Santa Elena. Los resultados se muestran en el siguiente detalle:

3.1 Resultado de la comparativa de Herramientas

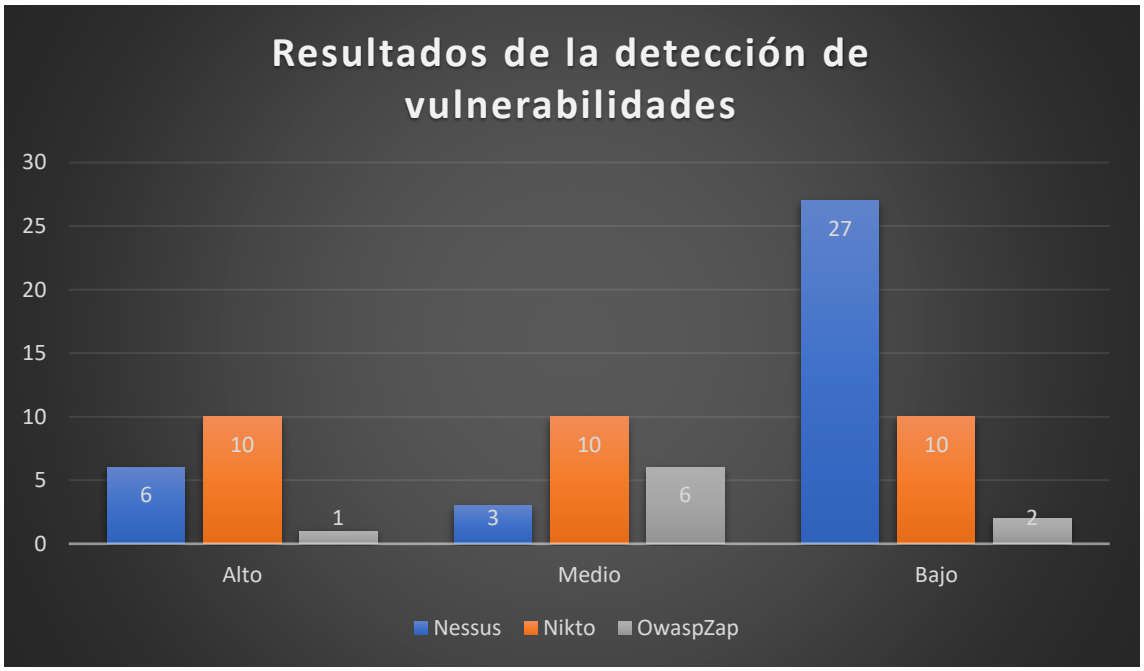


Ilustración 45 Comparativa de resultados entre las herramientas

Se obtuvo como resultado final de la comparativa, que la herramienta Nessus es la que lleva la delantera con respecto a tiempo de análisis, interfaz de usuario y contenido de reporte final y legibilidad de este, por lo tanto, sería la herramienta óptima para la detección de vulnerabilidades del portal web.

3.2 Resultados del Analisis de Vulnerabilidades

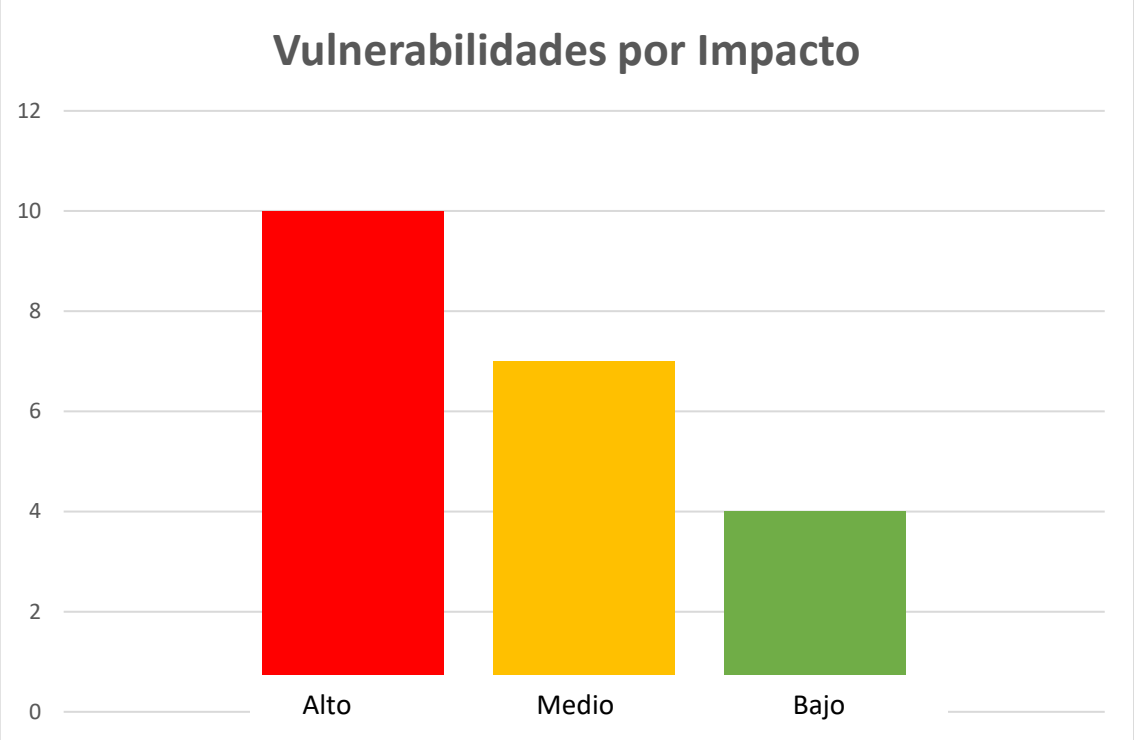
Durante el análisis de vulnerabilidades se obtuvieron los siguientes resultados: 6 de nivel alto , 3 de nivel medio y 27 de nivel bajo. En el siguiente detalle se muestra las vulnerabilidades por impacto que causan con su respectiva calificación.

3.3 Calificación de Gravedad

Severidad	CVSS	Descripción
Alto	7.0-10.0	La explotación es sencilla y generalmente da como resultado un compromiso a nivel del sistema. Se recomienda formar un plan de acción y parchear inmediatamente
Medio	4.0-6.9	La explotación es más difícil, pero podría causar privilegios elevados y potencialmente una pérdida de datos o tiempo de inactividad. Se aconseja formar un plan de acción y parchear lo antes posible.
Bajo	1.0-3.9	Existen vulnerabilidades, pero no se pueden explotar o requieren pasos adicionales, como la ingeniería social. Se recomienda elaborar un plan de acción y un parche después de que se hayan resuelto los problemas de alta prioridad.

Tabla 23 Calificaciones de Gravedad

3.4 Vulnerabilidades por Impacto



Las vulnerabilidades de nivel alto son las que se van a mostrar con sus respectivas acciones en el siguiente detalle:

Referencias:	CVE-2015-2325 - CVE-2015-2326 - CVE-2015-3414 - CVE-2015-3415 - CVE-2015-3416
Fecha de Publicación:	24/04/2015
Impacto:	Alto
Recursos Afectados:	Apache HTTP Server, versiones 5.4 – 5.4.42 – SQLite 3.8.9
Descripción:	• Existe una vulnerabilidad de denegación de servicio en el componente SQLite incluido debido al manejo incorrecto de las cotizaciones en los nombres de secuencia de intercalación • Existe una vulnerabilidad de inyección de comandos arbitraria debido a un defecto en la función php_escape_shell_arg(). • Permite a atacantes dependientes de contexto causar una denegación de servicio (desbordamiento de enteros y desbordamiento de buffer basado en pila)
Tipo:	Remoto
Acción:	• Actualice a PHP versión 5.4.42 o posterior. • Actualizar a una versión de SQLite superior a 3.8.9 • La directiva RequestReadTimeout permite limitar el tiempo que un cliente tarda en enviar una petición. • Deberían comprobarse también los valores de las directivas de timeout facilitadas por otros módulos. • El valor de la directiva TimeOut debería reducirse en sitios que son objeto de ataques DoS. Configurar ésta a unos segundos podría ser apropiado. • También es aconsejable comprobar la longitud de los datos para descartar posibles técnicas de inyección SQL, ya que, si por ejemplo estamos esperando un nombre, una cadena extremadamente larga puede suponer que estén intentando atacarnos por este método

Tabla 24 Vulnerabilidad Critica 1

Referencias:	CVE-2015-3152 - CVE-2015-5589 - CVE-2015-5590 - CVE-2015-8838
Fecha de Publicación:	16/05/2016
Impacto:	Alto
Recursos afectados:	Apache HTTP Server, versiones 5.4 – 5.4.42
Descripción:	• Un atacante man-in-the-middle puede explotar este defecto para obligar al cliente a degradar a una conexión sin cifrar, lo que permite al atacante revelar datos de la base de datos o manipular consultas de base de datos. • Un atacante puede explotar esto para bloquear una aplicación PHP, lo que resulta en una condición de denegación de servicio. • Un atacante remoto puede aprovechar esto para des referenciar la memoria ya liberada, lo que podría dar lugar a la ejecución de código arbitrario

Tipo:	Remoto
Acción:	• Actualice a PHP versión 5.4.43 o posterior. • Protege siempre los datos de tus clientes con un certificado SSL actualizado de una autoridad fiable en páginas web con acceso para clientes • Ofrece a tus usuarios métodos adicionales para que puedan iniciar sesión de forma segura. Por ejemplo, con una autenticación multifactor a través del correo. • Haz saber a los clientes que, en principio, nunca vas a pedir los datos de acceso a través del email y evita los enlaces en los correos que les envíes.

Fecha de Publicación:	4/05/2012
Impacto:	Alto 
Recursos Afectados:	Apache HTTP Server, versiones 5.4 – 5.4.42
Detalle:	• Según su versión, la instalación de PHP en el host remoto ya no es compatible. • La falta de soporte implica que el proveedor no lanzará nuevos parches de seguridad para el producto. • Como resultado, es probable que contenga vulnerabilidades de seguridad.
Tipo:	Remoto
Acción:	Actualice a una versión de PHP que sea compatible actualmente.

Referencias:	CVE-2015-6831 - CVE-2015-6832 - CVE-2015-6833 - CVE-2015-8867
Fecha de Publicación:	26/05/2016
Impacto:	Alto 
Recursos Afectados:	Apache HTTP Server, versiones 5.4
Descripción:	• Existe una vulnerabilidad de uso después de la liberación en ext/spl/spl_array.c debido al manejo incorrecto de un dato serializado especialmente diseñado. • Existe una vulnerabilidad de recorrido de directorio en la clase PharData, debido a la implementación incorrecta de la función exctractTo. • Un atacante remoto no autenticado puede explotar esto a través de una entrada de archivo ZIP diseñada para escribir en archivos

	arbitrarios. • Hace que sea más fácil para los atacantes remotos vencer a los mecanismos de protección de cifrado a través de vectores no especificados.
Tipo:	Remoto
Acción:	•Actualice a PHP versión 5.4.44 o posterior. •Otra cosa que se puede hacer (especialmente si un administrador quiere ir más allá del cumplimiento del deber) es realmente probar si su aplicación es vulnerable al recorrido del directorio. Es bastante fácil intentar estos procedimientos manualmente, pero existen herramientas que pueden automatizar fácilmente la mayoría de las pruebas como DirBuster, ZAP y DotDotPwn. •Garantizar que se utiliza la validación de entrada correcta por parte del usuario. De hecho, si se puede evitar, es mejor omitir por completo la entrada del usuario cuando se trata de operaciones del sistema de archivos.

Referencias:	CVE-2014-9767 - CVE-2015-6834 - CVE-2015-6835 - CVE-2015-6836 - CVE-2015-6837, CVE-2015-6838
Fecha de Publicación:	16/05/2016
Impacto:	Alto
Recursos Afectados:	Apache HTTP Server, versiones 5.4
Detalle:	• Una vulnerabilidad de recorrido de directorio en la función ZipArchive: extractTo en ext/zip/php_zip.c podría permitir a un atacante remoto crear directorios vacíos arbitrarios a través de un archivo ZIP creado • Existen varios errores de memoria de uso después de la liberación relacionados con la función unserialize (). Un atacante remoto puede explotar estos errores para ejecutar código arbitrario. • Existen varios defectos en la clase XSLTProcessor debido a la validación incorrecta de la entrada de la biblioteca libxslt. • Un atacante remoto puede explotar estos defectos para tener un impacto no especificado
Tipo:	Remoto
Recomendación:	• Actualice a PHP versión 5.4.45 o posterior. • Procesar las peticiones de URI para que no resulten en una petición de un fichero. • Cuando se realiza una petición URI por un fichero/directorio, se debe construir el path completo del fichero/directorio (sólo si este

	existe), y normalizar todos los caracteres (ej, 20% convertido a espacios). • Se asume que un documento raíz está normalizado, su directorio es conocido y tiene un tamaño N. Además, ningún fichero de este directorio puede ser servido. • Garantizar que todos los archivos y carpetas en el servidor tengan los controles de acceso adecuados. Además, los desarrolladores deben evitar almacenar datos confidenciales en la raíz web de la aplicación.

3.5 Requerimientos

Los requerimientos de esta investigación serán clasificados en el siguiente detalle de acuerdo con las fases de la investigación

- Recolección de Información
- Virtualización de Sistemas
- Análisis de Vulnerabilidades
- Análisis de Resultados
- Informe final

N°	Requerimientos de la Fase de Recolección de Información
1	Se utilizará la metodología **Marco de Evaluación de Seguridad del Sistema de Información (ISSAF)** para el levantamiento de información
2	Se utilizará un reconocimiento pasivo para la fase 1 de la investigación, el cual consiste en recolectar información sin la intervención directa de la victima
3	Se obtendrá información utilizando herramientas gratuitas en línea (Whois y Shodan), la información recolectada se detalla a continuación: • Direcciones IP de los servidores • Direcciones de los proveedores • Nombre del servidor • Herramientas de funcionamiento y sus versiones • Contactos de proveedores y administradores • Sistema Operativo del servidor

4	Se recopilará metadatos de archivos disponibles del portal web utilizando la herramienta foca, la información obtenida se detalla a continuación: • Archivos .doc • Archivos pdf • Servidores • Usuarios • Nombre de usuarios • Softwares utilizados • Nombre del software
	Requerimientos de la Fase de Virtualización de Sistemas
5	Para el desarrollo de la investigación se realizará la virtualización del sistema Kali Linux, con la finalidad de realizar todas las pruebas necesarias en un ambiente controlado. Características de la máquina virtual: • RAM: 2GB mínimo • Disco Duro: 80 GB
	Requerimientos de la Fase de Análisis de Vulnerabilidades
6	El análisis de vulnerabilidades se lo realizará con las siguientes herramientas: • Nessus (Windows) • Uniscan (Linux) • Maltego (Linux) • OwaspZap (Windows) • Nikto (Linux)
7	Se seleccionará la mejor herramienta realizando una comparativa entre ellas tomando en consideración los siguientes indicadores: • Facilidad de Uso • Requerimiento del sistema • Facilidad de instalación • Interfaz de Usuario • Complejidad • N° de vulnerabilidades encontradas • Legibilidad del reporte final

	• Contenido del reporte • Velocidad de detección
	Requerimientos de la Fase de Análisis de Resultados
8	Los resultados serán clasificados de acuerdo con el nivel de riesgo, los niveles se detallan a continuación: • Alto • Medio • Bajo
9	La clasificación de los resultados se la realizara mediante una consulta a la base de datos de la página web incibe-cert. Este repositorio con más de 75.000 registros está basado en la información de NVD (http://nvd.nist.gov/) (National Vulnerability Database)
10	Los resultados obtenidos por la herramienta más optima será mostrado en una tabla detallada que contará con los siguientes datos: • Fecha de Publicación • Nombre o Referencia • Importancia • Recursos Afectados • Detalle • Tipo • Acción

4 CONCLUSIONES

Los ataques informáticos a equipos y sistemas de información son prácticamente inevitables. Sin embargo, se pueden mitigar el daño causados por estos. En esta investigación se identificó varias falencias a nivel de seguridad de servicios, al no contar con políticas de seguridad adecuadas para salvaguardar la integridad de la información y las políticas existentes no están cumpliéndose de manera correcta.

Mediante la implementación del método propuesto, conocido como Hacking Ético, nos facilita el uso de herramientas y técnicas, las cuales es recomendable utilizarlas periódicamente en toda institución u organización, con la finalidad de que se permita mantener en constante vigilancia sobre las posibles vulnerabilidades que existan en nuestros sistemas, las cuales comprometan la seguridad de la información.

Todas las aplicaciones y herramientas en algún punto son vulnerables a diferentes tipos de ataques, realizar este tipo de análisis nos permite mantener la capacidad de respuesta ante vulnerabilidades encontradas desde el punto de vista de negocio, es decir enfrentar el costo de arreglar las cosas contra el costo de no hacer nada al respecto.

Es importante conocer las vulnerabilidades y riesgos, que se encuentran en los diferentes activos de información dentro de una empresa, para poder determinar si los activos son seguros o no, y así implementar medidas que permitan disminuir el impacto de cualquier evento.

5 RECOMENDACIONES

Se recomienda seguir implementando la metodología de hacking éticos en diversos escenarios, esto permitirá conocer más a fondo las brechas de seguridad que pueden comprometer la integridad de la información.

Realizar este tipo de análisis de forma periódica brinda la capacidad de mantener a los sistemas preparados ante posibles ataques a futuro, o en cierta manera minimizar los daños que puedan causar.

La información se ha convertido en el activo intangible más valioso del mundo por tal motivo se debe dar la importancia necesaria para mantener a las instituciones y organizaciones preparadas antes eventos adversos como los ataques informáticos.

Hoy en día el aumento del uso de las tecnologías web ha permitido un desarrollo de manera paralela a los ataques informáticos. Para poder mitigar y prevenir el riesgo de algunos de ellos y así evitar que se vea comprometida la seguridad de la información del portal web se recomienda:

- Analisis periódicos de las tecnologías y herramientas en la cual se almacena la información del portal web.
- Actualizar las herramientas y software que se utilicen en el portal web a su versión mas reciente, debido a que cada nueva versión nos brinda parches de seguridad.
- Una configuración correcta de las herramientas utilizadas permite minimizar cualquier explotación de vulnerabilidades

ANEXOS

Informe de Analisis de Vulnerabilidades

Confidencial

Autor:

Luis Arturo Rodriguez Matías

Tutor:

Ing. Iván Coronel

Fecha: Mar 11th, 2021

Versión 1.0

Declaración de Confidencialidad

Este documento es propiedad exclusiva de la Universidad Estatal Península de Santa Elena y Gobierno Autónomo Descentralizado del Cantón La Libertad. Este informe contiene información confidencial y de propiedad exclusiva. La duplicación, redistribución o uso, total o parcial, en cualquier forma, requiere el consentimiento de la UPSE y GADCL.

La UPSE puede compartir este documento con auditores en virtud de acuerdos de confidencialidad para demostrar el cumplimiento de los requisitos del análisis de vulnerabilidades.

Disclaimer

Un análisis de vulnerabilidades realizado en un periodo de tiempo determinado brinda los hallazgos y recomendaciones, reflejan la información recopilada durante la evaluación y no los cambios o modificaciones realizados fuera de ese período.

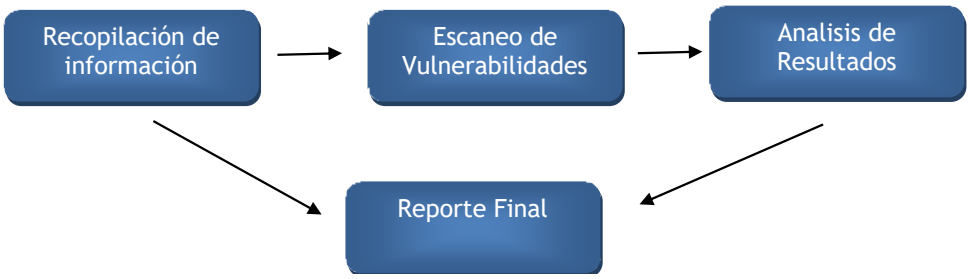
Los compromisos de tiempo limitado no permiten una evaluación completa de todos los controles de seguridad. Se dio prioridad a la evaluación para identificar los controles de seguridad más débiles que un atacante podría aprovechar. Se recomienda realizar evaluaciones similares anualmente por parte de evaluadores internos o externos para asegurar el éxito continuo de los controles.

Resumen de la evaluación

Desde el 21 de septiembre de 2020 hasta el 21 de enero de 2021, se ejecutó varias pruebas para evaluar la postura de seguridad de su infraestructura en comparación con las mejores prácticas actuales de la industria que incluían un análisis de vulnerabilidades. Todas las pruebas realizadas se basan en la Guía técnica para pruebas y evaluación de seguridad de la información.

Las fases de las actividades del análisis de vulnerabilidades incluyen las siguientes:

- Planificación: se recopilan los objetivos del cliente y se obtienen las reglas de participación.
- Escaneo: realice análisis y enumeración para identificar posibles vulnerabilidades, áreas débiles y exploits
- Analisis: confirme las vulnerabilidades potenciales mediante el escaneo se realiza la clasificación de estas.
- Informes: Se documenta todas las vulnerabilidades y exploits encontrados, los intentos fallidos y las fortalezas y debilidades de la empresa



Componentes de evaluación

Prueba de escaneo externo

Una prueba de escaneo externa emula el papel de un atacante que intenta obtener acceso a un sistema interno sin recursos o conocimiento internos. Un ingeniero de Tecnologías de Información intenta recopilar información confidencial a través de la inteligencia de código abierto (OSINT), ingeniería social, incluida la información obtenida de internet y más, que se puede aprovechar contra sistemas externos para obtener acceso a la red interna o algún sistema en específico. También se realiza análisis y enumeración para identificar posibles vulnerabilidades con la esperanza de que se exploten.

Encontrando calificaciones de gravedad

En el siguiente detalle se define los niveles de gravedad y el rango de puntuación CVSS correspondiente que se utilizan en todo el documento para evaluar la vulnerabilidad y el impacto del riesgo.

Severidad	CVSS	Descripción
Alto	7.0-10.0	La explotación es sencilla y generalmente da como resultado un compromiso a nivel del sistema. Se recomienda formar un plan de acción y parchear inmediatamente
Medio	4.0-6.9	La explotación es más difícil, pero podría causar privilegios elevados y potencialmente una pérdida de datos o tiempo de inactividad. Se aconseja formar un plan de acción y parchear lo antes posible.
Bajo	1.0-3.9	Existen vulnerabilidades, pero no se pueden explotar o requieren pasos adicionales, como la ingeniería social. Se recomienda elaborar un plan de acción y un parche después de que se hayan resuelto los problemas de alta prioridad.

Alcance

Titulo	Detalle
Prueba de Escaneo Externo	186.42.198.98

Durante el desarrollo del análisis de vulnerabilidades, se recopilará información de manera pasiva, esto quiere decir sin la intervención directa del usuario, estos datos serán de suma importancia para el respectivo escaneo de vulnerabilidades.

Exclusiones de alcance

No se realizará ningún ataque de denegación de servicio durante las pruebas, ni de ningún otro tipo

Resumen Ejecutivo

Se evaluó la postura de seguridad externa de un GADL a través de un análisis de vulnerabilidades desde el 21 de septiembre de 2020 hasta el 21 de enero de 2021. Se encontró vulnerabilidades de nivel crítico que permitirán acceso a sistemas y a otra información clasificada. Se recomienda encarecidamente que el GADML aborde estas vulnerabilidades lo antes posible, ya que las vulnerabilidades se encuentran fácilmente a través de un reconocimiento básico y se pueden explotar sin mucho esfuerzo

Resumen de Pruebas

la siguiente tabla describe las fases del análisis de seguridad, paso a paso:

N	Fase	Descripción
1	Recopilación de la Información	Este proceso se lo realizará mediante un reconocimiento pasivo. Este se consigue la información sin interacción directa con el objetivo mediante el uso de técnicas tales como la ingeniería social, búsquedas por internet o mediante el uso de aplicativos webs que nos puedan brindar información crucial para el desarrollo de la investigación.
2	Virtualización de Sistemas	En esta fase se creará máquinas virtuales o Live USB con sus correspondientes sistemas operativos y sus configuraciones necesarias de las siguientes distribuciones que serán necesarias para el respectivo análisis

3	Análisis de Vulnerabilidades	Las herramientas nos brindan la posibilidad de obtener un análisis de manera más detallado, una vez identificadas las posibles vulnerabilidades presentes en el portal web las podremos clasificar de acuerdo con el nivel de riesgo que presenten
4	Analisis de Resultados	La identificación de las vulnerabilidades obtenidas en la fase anterior se las clasificara de acuerdo con el nivel de riesgo, usualmente son tres: Alto, Medio y Bajo, esta se realiza de acuerdo con la versión del sistema operativo y de los servicios y aplicaciones detectados comparándolos contra una base de datos de vulnerabilidades que se actualiza frecuentemente conforme nuevas amenazas de seguridad son descubiertos

Fortalezas de seguridad

No encontradas

Debilidades de seguridad

Intentos de inicio de sesión sin restricciones

Durante la evaluación, se realizó varios intentos de ingreso a los formularios de inicio de sesión que se encuentran en la página. Para todos los inicios de sesión, se permitieron intentos ilimitados, lo que permite que un atacante pueda obtener acceso.

Aplicaciones con versiones desactualizadas

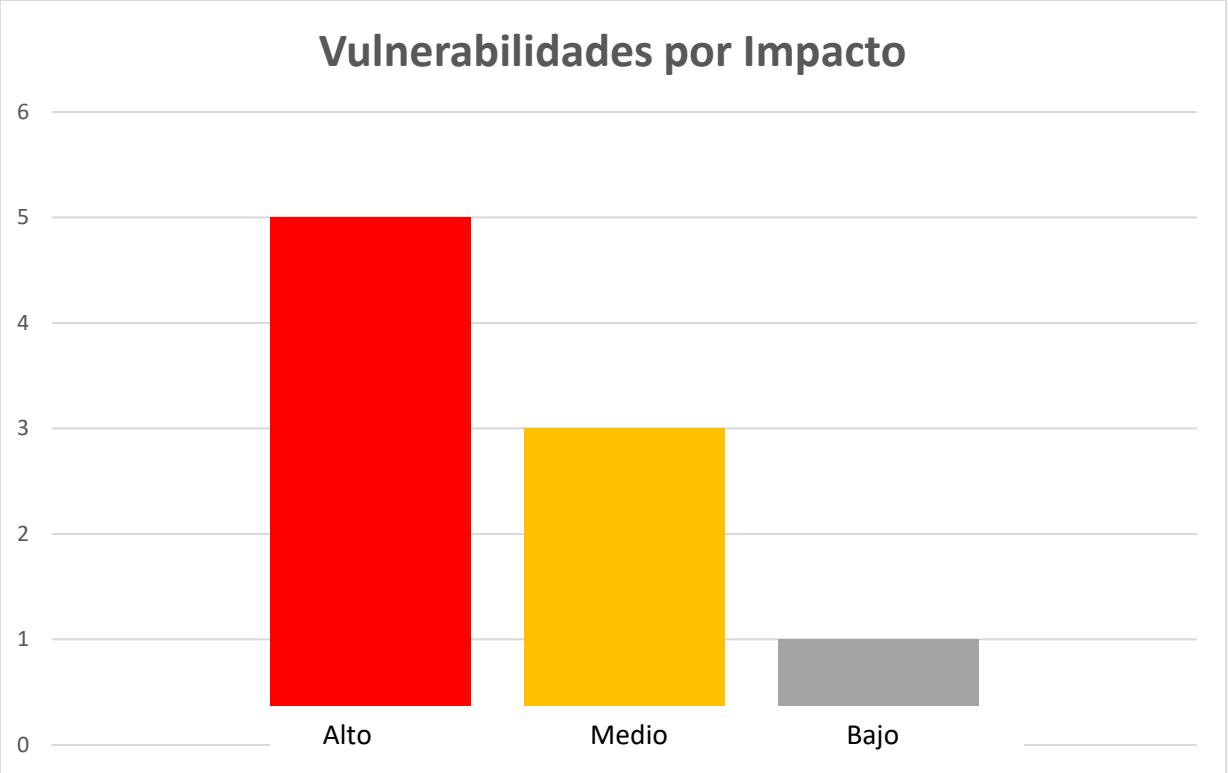
Durante la evaluación, se logró apreciar que varias aplicaciones utilizadas en el portal web están siendo ejecutadas con una versión muy antigua, lo que permite que se encuentren vulnerabilidades fáciles de explotar y facilitaría la intrusión de un atacante.

Configuración básica de aplicaciones

Durante la evaluación, se detectó que se ha dejado la configuración básica de herramientas (configuración por defecto), esto se atribuye algunas vulnerabilidades encontradas durante el análisis, una correcta configuración, minimiza los riesgos y hace que sea más complicado la explotación de una vulnerabilidad.

Vulnerabilidades por Impacto

El siguiente cuadro ilustra las vulnerabilidades encontradas por impacto:



Resultados de la prueba de escaneo externa

Vulnerabilidades de Nivel Alto

Referencias:	CVE-2015-2325 - CVE-2015-2326 - CVE-2015-3414 - CVE-2015-3415 - CVE-2015-3416
Fecha de Publicación:	24/04/2015
Impacto:	Alto
Recursos Afectados:	Apache HTTP Server, versiones 5.4 – 5.4.42 – SQLite 3.8.9
Descripción:	• Existe una vulnerabilidad de denegación de servicio en el componente SQLite incluido debido al manejo incorrecto de las cotizaciones en los nombres de secuencia de intercalación • Existe una vulnerabilidad de inyección de comandos arbitraria debido a un defecto en la función php_escape_shell_arg(). • Permite a atacantes dependientes de contexto causar una denegación de servicio (desbordamiento de enteros y desbordamiento de buffer basado en pila)
Tipo:	Remoto
Acción:	• Actualice a PHP versión 5.4.42 o posterior. • Actualizar a una versión de SQLite superior a 3.8.9

	• La directiva RequestReadTimeout permite limitar el tiempo que un cliente tarda en enviar una petición. • Deberían comprobarse también los valores de las directivas de timeout facilitadas por otros módulos. • El valor de la directiva TimeOut debería reducirse en sitios que son objeto de ataques DoS. Configurar ésta a unos segundos podría ser apropiado. • También es aconsejable comprobar la longitud de los datos para descartar posibles técnicas de inyección SQL, ya que, si por ejemplo estamos esperando un nombre, una cadena extremadamente larga puede suponer que estén intentando atacarnos por este método

Referencias:	CVE-2015-3152 - CVE-2015-5589 - CVE-2015-5590 - CVE-2015-8838
Fecha de Publicación:	16/05/2016
Impacto:	Alto
Recursos afectados:	Apache HTTP Server, versiones 5.4 – 5.4.42
Descripción:	• Un atacante man-in-the-middle puede explotar este defecto para obligar al cliente a degradar a una conexión sin cifrar, lo que permite al atacante revelar datos de la base de datos o manipular consultas de base de datos. • Un atacante puede explotar esto para bloquear una aplicación PHP, lo que resulta en una condición de denegación de servicio. • Un atacante remoto puede aprovechar esto para des referenciar la memoria ya liberada, lo que podría dar lugar a la ejecución de código arbitrario
Tipo:	Remoto
Acción:	• Actualice a PHP versión 5.4.43 o posterior. • Protege siempre los datos de tus clientes con un certificado SSL actualizado de una autoridad fiable en páginas web con acceso para clientes • Ofrece a tus usuarios métodos adicionales para que puedan iniciar sesión de forma segura. Por ejemplo, con una autenticación multifactor a través del correo. • Haz saber a los clientes que, en principio, nunca vas a pedir los datos de acceso a través del email y evita los enlaces en los correos que les envíes.

Fecha de Publicación:	4/05/2012
Impacto:	Alto
Recursos Afectados:	Apache HTTP Server, versiones 5.4 – 5.4.42
Detalle:	• Según su versión, la instalación de PHP en el host remoto ya no es compatible. • La falta de soporte implica que el proveedor no lanzará nuevos parches de seguridad para el producto. • Como resultado, es probable que contenga vulnerabilidades de seguridad.
Tipo:	Remoto
Acción:	Actualice a una versión de PHP que sea compatible actualmente.

Referencias:	CVE-2015-6831 - CVE-2015-6832 - CVE-2015-6833 - CVE-2015-8867
Fecha de Publicación:	26/05/2016
Impacto:	Alto
Recursos Afectados:	Apache HTTP Server, versiones 5.4
Descripción:	• Existe una vulnerabilidad de uso después de la liberación en ext/spl/spl_array.c debido al manejo incorrecto de un dato serializado especialmente diseñado. • Existe una vulnerabilidad de recorrido de directorio en la clase PharData, debido a la implementación incorrecta de la función exctractTo. • Un atacante remoto no autenticado puede explotar esto a través de una entrada de archivo ZIP diseñada para escribir en archivos arbitrarios. • Hace que sea más fácil para los atacantes remotos vencer a los mecanismos de protección de cifrado a través de vectores no especificados.
Tipo:	Remoto
Acción:	•Actualice a PHP versión 5.4.44 o posterior. •Otra cosa que se puede hacer (especialmente si un administrador quiere ir más allá del cumplimiento del deber) es realmente probar si su aplicación es vulnerable al recorrido del directorio. Es bastante fácil intentar estos procedimientos manualmente, pero existen herramientas

	que pueden automatizar fácilmente la mayoría de las pruebas como DirBuster, ZAP y DotDotPwn. •Garantizar que se utiliza la validación de entrada correcta por parte del usuario. De hecho, si se puede evitar, es mejor omitir por completo la entrada del usuario cuando se trata de operaciones del sistema de archivos.

Referencias:	CVE-2014-9767 - CVE-2015-6834 - CVE-2015-6835 - CVE-2015-6836 - CVE-2015-6837, CVE-2015-6838
Fecha de Publicación:	16/05/2016
Impacto:	Alto
Recursos Afectados:	Apache HTTP Server, versiones 5.4
Detalle:	• Una vulnerabilidad de recorrido de directorio en la función ZipArchive: extractTo en ext/zip/php_zip.c podría permitir a un atacante remoto crear directorios vacíos arbitrarios a través de un archivo ZIP creado • Existen varios errores de memoria de uso después de la liberación relacionados con la función unserialize (). Un atacante remoto puede explotar estos errores para ejecutar código arbitrario. • Existen varios defectos en la clase XSLTProcessor debido a la validación incorrecta de la entrada de la biblioteca libxslt. • Un atacante remoto puede explotar estos defectos para tener un impacto no especificado
Tipo:	Remoto
Acción:	• Actualice a PHP versión 5.4.45 o posterior. • Procesar las peticiones de URI para que no resulten en una petición de un fichero. • Cuando se realiza una petición URI por un fichero/directorio, se debe construir el path completo del fichero/directorio (sólo si este existe), y normalizar todos los caracteres (ej, 20% convertido a espacios). • Se asume que un documento raíz está normalizado, su directorio es conocido y tiene un tamaño N. Además, ningún fichero de este directorio puede ser servido. • Garantizar que todos los archivos y carpetas en el servidor tengan los controles de acceso adecuados. Además, los desarrolladores deben evitar almacenar datos confidenciales en la raíz web de la aplicación.

Vulnerabilidades de Nivel Medio

Referencias:	CVE-2015-2325 - CVE-2015-2326 - CVE-2015-4021 - CVE-2015-4022 - CVE-2006-7243 - CVE-2015-4025 - CVE-2015-4024
Fecha de Publicación:	14/01/2020
Impacto:	Media
Recursos Afectados:	Apache HTTP Server, versiones 5.4 – 5.4.41
Detalle:	• Un defecto en la función de phar_parse_tarfile en ext/phar/tar.c podría permitir una denegación de servicio a través de una entrada hecha a mano en un archivo tar. • Un atacante remoto puede aprovechar esto para provocar un desbordamiento de búfer basado en montón, lo que resulta en una condición de denegación de servicio o una posible ejecución remota de código • Un atacante remoto puede explotar estos defectos, mediante la combinación del carácter '-0' con una extensión de archivo segura, para eludir las restricciones de acceso
Tipo:	Remoto
Acción:	• Actualice a PHP versión 5.4.41 o posterior. • Seguir estándares de desarrollo de código seguro • Optar por lenguajes de programación que, además de ser eficientes en cuanto al uso de memoria, sean seguros. • Usar ejecutables de tipo position-independent. ¿Esto para qué serviría? Para limitar los impactos que pudiese causar un desbordamiento de búfer

Referencias:	CVE-2004-2320
Fecha de Publicación:	10/07/2017
Impacto:	Media
Recursos Afectados:	HTTP TRACE / TRACK Methods Allowed
Detalle:	• Las funciones de depuración están habilitadas en el servidor web remoto • El servidor web remoto admite los métodos TRACE y/o TRACK. TRACE y TRACK son métodos HTTP que se utilizan para depurar conexiones de servidor web.
Tipo:	Remoto

Acción:	• Deshabilite estos métodos HTTP. Consulte la salida del plugin para obtener más información.
	• Si la vulnerabilidad se presenta en APACHE, se debe agregar la siguiente linea al final del archivo httpd.conf: `<code>TraceEnable Off</code>`
	• **Para Linux**
	• **Ingresar por ssh a la maquina a intervenir** `ssh usuario@x.x.x.x`
	2. Ingresamos a la ruta /etc/httpd/conf con el comando cd `cd /etc/httpd/conf`
	3. Dentro de la carpeta se procede a hacer copia de seguridad del archivo httpd.conf el cual se va a editar `cp httpd.conf httpd.conf.copy`
	4. **Se procede a editar el archivo** httpd.conf **con el comando** vi, **se pueden utilizar otros editores como** vim o nano `vi httpd.conf`
	5. **Dentro del archivo** httpd.conf**, se presiona la tecla** i **para ingresar a modo inserción lo cual permita editar el texto, se debe ingresar la siguiente linea** TraceEnable off, **como se muestra en la imagen**
	`# you will save yourself a lot of trouble.`
	`#`
	`# Do NOT add a slash at the end of the directory path.`
	`#`
	`ServerRoot "/etc/httpd"`
	`TraceEnable off`
	`#`
	1.

Referencias:	CVE-2015-2325 - CVE-2015-2326 - CVE-2015-4021 - CVE-2015-4022 - CVE-2006-7243 - CVE-2015-4025 - CVE-2015-4024
Fecha de Publicación:	CVE-2020-11022
Impacto:	Media
Recursos Afectados:	JQuery 1.2 < 3.5.0 Múltiple XSS
Detalle:	• El servidor web remoto se ve afectado por la vulnerabilidad de scripting entre sitios múltiple • Según la versión auto informada en el script, la versión de JQuery alojada en el servidor web remoto es mayor o igual que 1.2 y anterior a 3.5.0. Por lo tanto, se ve afectado por múltiples vulnerabilidades de scripting entre sitios
Tipo:	Remoto
Acción:	• Actualice a JQuery versión 3.5.0 o posterior. • Estar siempre atento a dónde ingresas: siempre es bueno mirar la dirección URL a la que se está accediendo. Ya hemos visto ejemplos de esto, como por ejemplo las redirecciones de Facebook, que pueden ser aprovechadas por atacantes • Existen complementos para los navegadores que se encarga de bloquear estos scripts en sitios web. Uno de ellos es NoScript, que permite realizar configuraciones personalizadas • Si se trata de una redirección a algún sitio de phishing, se cuenta con la protección del antivirus y el bloqueo proactivo por parte de los navegadores

Vulnerabilidades de Nivel Bajo

Referencias:	Algoritmos débiles SSH soportados - CVE-2016-2183
Fecha de Publicación:	14/12/2016
Impacto:	Bajo
Recursos Afectados:	Algoritmos débiles SSH soportados
Detalle:	• El servidor SSH remoto está configurado para permitir algoritmos de cifrado débiles o ningún algoritmo en absoluto. • • El host remoto se ve afectado por una vulnerabilidad de divulgación de información de tipo man-in-the-middle (MitM) debido a un error en la implementación de cipher-suites que utilizan Arcfour debido a que este algoritmo de cifrado es débil.
Tipo:	Remoto
Acción:	• Reconfigure el servicio para quitar la compatibilidad con los conjuntos de cifrado arcfour256, arcfour128, Arcfour debido a que son débiles. • Elimine el cifrado de flujo de Arcfour a través de SSH utilizando PuTTY. • Inicie sesión en el sistema operativo SUSE Linux o Solaris como usuario de ossuser a través de SSH mediante PuTTY. • 2.Ejecute el siguiente comando para acceder al directorio /opt/oss/server/base_service/sysguard/resource. • $ cd /opt/oss/server/base_service/sysguard/resource • 3.Ejecute el siguiente comando para modificar la lista de algoritmos de seguridad débiles: • $ python modifyWeakCipherAlgList.pyc

Nombre o Referencia	Cifrados de modo CBC del servidor SSH habilitados – CVE-2020-5408
Fecha de Publicación:	14/05/2020
Impacto:	Bajo
Recursos Afectados:	Servidor SSH
Detalle:	• El servidor SSH está configurado para utilizar Cipher Block Chaining • Esto puede permitir a un atacante recuperar el mensaje de texto no cifrado del texto cifrado
Acción:	• Póngase en contacto con el proveedor o consulte la documentación del producto para deshabilitar el cifrado de cifrado en modo CBC y habilite el cifrado de modo de cifrado CTR o GCM.

Nombre o Referencia	SSH Débiles algoritmos MAC habilitados - CVE-2016-2183
Fecha de Publicación:	14/12/2016
Impacto:	Bajo
Recursos Afectados:	SSH -MAC
Detalle:	• El servidor SSH remoto está configurado para permitir algoritmos MD5 y MAC de 96 bits. • El servidor SSH remoto se configura para permitir los algoritmos MD5 o MAC de 96 bits, ambos de los cuales se consideran débiles
Acción:	• Reconfigure el servicio para quitar la compatibilidad con los conjuntos de cifrado CBC con HMAC-SHA1 o HMAC-SHA256. • SISTEMA OPERATIVO LINUX: • Para dar solución al problema /etc/sshd_config : • # default is aes128-ctr,aes192-ctr,aes256-ctr,arcfour256,arcfour128, • # aes128-cbc,3des-cbc,blowfish-cbc,cast128-cbc,aes192-cbc, • # aes256-cbc,arcfour • Ciphers aes128-ctr,aes192-ctr,aes256-ctr, <– agregar esta linea para eliminar los algoritmos cbc vulnerables • # default is hmac-md5,hmac-sha1,hmac-ripemd160,hmac-sha1-96,hmac-md5-96 • # you can remove the hmac-md5 MACs with MACs hmac-sha1,hmac-ripemd160 • MACs hmac-sha1,umac-64@openssh.com,hmac-ripemd160 –agregar esta linea para eliminar los algoritmos MAC vulnerables

Bibliografía

[1] M. V. B. SÁNCHEZ, «Universidad Politecnica Salesiana,» Julio 2018. [En línea]. Available: https://dspace.ups.edu.ec/bitstream/123456789/15860/1/UPS-ST003656.pdf.

[2] M. G. Nieva Machín, «LA CIBERSEGURIDAD COMO FACTOR CRÍTICO EN LA UNION EUROPEA,» *Revista UNISCI,* p. 22, Octubre 2016.

[3] K. Lab, «Brasil, Mexico y Colombia lideran incidentes de secuestros digitales en America Latina,» 2017. [En línea]. Available: https://latam.kaspersky.com/about/press-releases/2017_kaspersky-lab-incidents-of-digital-kidnappings-in-latin-america.

[4] I. T. R. Center, «Identity Theft Resource Center,» 2017. [En línea]. Available: https://www.idtheftcenter.org/2017-data-breaches/.

[5] W. ESET, «WeLiveSegurity,» 2019. [En línea]. Available: https://www.welivesecurity.com/wp-content/uploads/2019/07/ESET-security-report-LATAM-2019.pdf.

[6] C. H. T. T, «AMENAZAS INFORMÁTICAS Y SEGURIDAD DE LA INFORMACIÓN,» *Universidad Externado de Colombia,* 2016.

[7] L. C.-J. Patricio Barrazueta Molina, «Orientación al ciudadano en el "gobierno electrónico" de los municipios del Ecuador,» *Teknokultura,* p. 15, 2018.

[8] J. O. M. A. Raphaël Hertzog, Kali Linux Revealed, USA: Offsec Press, 2017.

[9] R. Velasco, «RedesZone,» 2019. [En línea]. Available: https://www.redeszone.net/2019/02/23/uniscan-buscar-vulnerabilidades/.

[10] Nmap, «Nmap.org,» 2017. [En línea]. Available: https://nmap.org/man/es/index.html.

[11] E. Paths, «Eleven Paths,» 2020. [En línea]. Available: https://www.elevenpaths.com/es/labstools/foca-2/index.html.

[12] G. Advisor, «G.B Advisor,» 2020. [En línea]. Available: https://www.gb-advisors.com/es/gestion-de-vulnerabilidades/nessus-escaner-vulnerabilidad/.

[13] FACSISTEL, «Universidad Estatal Península de Santa Elena,» 2020. [En línea]. Available: http://facsistel.upse.edu.ec/index.php?option=com_content&view=article&id=58&Itemid=463.

[14] D. J. R. R. Normandi Rocío Tirado Ríos, «Seguridad Informática, un mecanismo para salvaguardar la Información de las Empresas,» *Publicando,* 2017.

[15] C. J. Melania González, «Características y uso de páginas web en las empresas de la zona sur, Ecuador,» *Publicando,* 2017.

[16] S. T. P. Ecuador, «Secretaria Tecnica Planifica Ecuador,» 2017. [En línea]. Available: https://www.planificacion.gob.ec/plan-nacional-de-desarrollo-2017-2021-toda-una-vida/.

[17] ESET, «Welivesecurity,» Noviembre 2014. [En línea]. Available: https://www.welivesecurity.com/la-es/2014/11/12/identificar-analizar-evaluar-vulnerabilidades/.

[18] CEROUNO, «CEROUNO,» Agosto 2017. [En línea]. Available: https://cerounosoftware.com.mx/2017/08/02/por-qu%C3%A9-es-importante-detectar-las-vulnerabilidades/.

[19] O. BID, «Banco Interamericano de Desarrollo,» 2020. [En línea]. Available: https://publications.iadb.org/es/publications/spanish/document/Reporte-Ciberseguridad-2020-riesgos-avances-y-el-camino-a-seguir-en-America-Latina-y-el-Caribe.pdf.

[20] EcuRed, «EcuRed,» 2013. [En línea]. Available: https://www.ecured.cu/Kali_linux.

[21] C. Tori, Hacking Ético, Argentina: Mastroianni Impresiones, 2008.

[22] F. X. A. REINOSO, Artist, *ANÁLISIS Y DISEÑO DE UNA PROPUESTA PARA MITIGAR ATAQUES.* [Art]. UNIVERSIDAD POLITÉCNICA SALESIANA, 2019.

[23] Welivesecurity, «Welivesecurity,» 2015. [En línea]. Available: https://www.welivesecurity.com/la-es/2015/06/16/ciberseguridad-seguridad-informacion-diferencia/.

[24] I. I. D. S. CIBERNÉTICA, «INSTITUTO INTERNACIONAL DE SEGURIDAD CIBERNÉTICA,» 2020. [En línea]. Available: https://www.iicybersecurity.com/analisis-de-vulnerabilidad-informatica.html.

[25] Ambit, «Ambit,» 2020. [En línea]. Available: https://www.ambit-bst.com/blog/an%C3%A1lisis-de-riesgos-inform%C3%A1ticos-y-ciberseguridad.

[26] D. C. Barbosa, «WeliveSecurity,» 2020. [En línea]. Available: https://www.welivesecurity.com/la-es/2020/06/03/nexpose-herramienta-analisis-vulnerabilidad/.

[27] C. F. P. B. Roberto Hernandez, Metologia de la Investigación, 2010.

[28] M. Ascencio y M. Patiño, «Universidas Tecnologica de Pereira,» 2011. [En línea]. Available: repositorio.utp.edu.co/dspace/bitstream/handle/11059/2511/0058A811.pdf?sequence=1&isAllowed=y.

[29] Incibe_cert, «Incibe_cert,» 2020. [En línea]. Available: https://www.incibe-cert.es/alerta-temprana/vulnerabilidades.