UNIVERSIDAD AGRARIA DEL ECUADOR
FACULTAD DE CIENCIAS AGRARIAS
CARRERA DE TECNOLOGÍA EN COMPUTACIÓN E INFORMÁTICA

ANÁLISIS DESCRIPTIVO DE LAS VULNERABILIDADES EN LOS SISTEMAS OPERATIVOS WINDOWS (PENTESTING)

MONOGRAFÍA

LINEA DE INVESTIGACION

INGENIERÍA EN SOFTWARE

AUTOR

GONZALES ARIAS OMAR SET

NARANJAL – ECUADOR

2021

UNIVERSIDAD AGRARIA DEL ECUADOR
FACULTAD DE CIENCIAS AGRARIAS
CARRERA DE TECNOLOGÍA EN COMPUTACIÓN E INFORMÁTICA

TEMA:
ANÁLISIS DESCRIPTIVO DE LAS VULNERABILIDADES EN LOS SISTEMAS OPERATIVOS WINDOWS (PENTESTING)

MONOGRAFÍA

Trabajo de titulación presentado como requisito para la obtención del título de
TECNÓLOGO EN COMPUTACIÓN E INFORMÁTICA

AUTOR
GONZALES ARIAS OMAR SET

TUTOR
LIC. GINA LOOR CAICEDO, M.SG.

NARANJAL – ECUADOR

2021

UNIVERSIDAD AGRARIA DEL ECUADOR
FACULTAD DE CIENCIAS AGRARIAS
CARRERA DE TECNOLOGÍA EN COMPUTACIÓN E INFORMÁTICA

CERTIFICACIÓN DE ACEPTACIÓN DEL TUTOR

Yo, GINA LOOR CAICEDO, docente de la Universidad Agraria del Ecuador, en mi calidad de Tutor, certifico que el presente trabajo de titulación: ANÁLISIS DESCRIPTIVO DE LAS VULNERABILIDADES EN LOS SISTEMAS OPERATIVOS WINDOWS (PENTESTING), realizado por el estudiante GONZALES ARIAS OMAR SET; con cedula de identidad 080381261-9, de la carrera Tecnología en Computación e Informática, unidad académica Naranjal, ha sido orientado y revisado durante su ejecución; y cumple con los requisitos técnicos exigidos por la Universidad Agraria del Ecuador; por lo tanto se aprueba la presentación del mismo.

Considerando aprobado en su totalidad.

Atentamente

Lic. Gina Loor Caicedo, M.Sc.
Docente tutora

Guayaquil, 3 de Septiembre del 2021

UNIVERSIDAD AGRARIA DEL ECUADOR
FACULTAD DE CIENCIAS AGRARIAS
CARRERA DE TECNOLOGÍA EN COMPUTACIÓN E INFORMÁTICA

APROBACIÓN DEL TRIBUNAL DE SUSTENTACIÒN

Los abajo firmantes, docentes miembros del Tribunal de Sustentación, aprobamos la sustentación del trabajo de titulación: ANÁLISIS DESCRIPTIVO DE LAS VULNERABILIDADES EN LOS SISTEMAS OPERATIVOS WINDOWS (PENTESTING), realizado por el estudiante GONZALES ARIAS OMAR SET, el mismo que cumple con los requisitos exigidos por la Universidad Agraria del Ecuador.

Atentamente,

_____

Ing. Enrique Ferruzola Gómez

PRESIDENTE

_____	_____
Ing. Julio Alvarado Zabala	Ing.Patricia Chavez Granizo
EXAMINADOR PRINCIPAL	**EXAMINADOR SUPLENTE**

Guayaquil, 3 de Septiembre del 2021

Dedicatoria

Dedico esta Monografía principalmente a Dios por darme la fortaleza para día a día seguir superándome, por los triunfos y momentos difíciles que me han enseñado afrontar y valorar cada día más.

A mis padres y hermanos por demostrarme siempre su cariño, apoyo y amor incondicional, por inculcarme valor para ser una buena persona, quienes con sus consejos han sabido guiarme para culminar mi carrera profesional.

Agradecimiento

Agradezco a Dios por darme la vida y satisfacción de ser una persona motivada y sacrificada para alcanzar sus logros, por protegerme durante todo mi camino y darme fuerzas para superar obstáculos y dificultades a lo largo de toda mi vida.

A mis padres, hermanos y familiares quienes me han enseñado a no desfallecer ni rendirme ante nada y siempre perseverar ante sus sabios consejos.

AUTORIZACIÒN DE AUTORÍA INTELECTUAL

Yo GONZALES ARIAS OMAR SET en calidad de autora del proyecto realizado, sobre “ANÁLISIS DESCRIPTIVO DE LAS VULNERABILIDADES EN LOS SISTEMAS OPERATIVOS WINDOWS (PENTESTING)” para optar el título de TECNÓLOGA EN COMPUTACIÓN E INFORMÁTICA, por la presente autorizo a la UNIVERSIDAD AGRARIA DEL ECUADOR, hacer uso de todos los contenidos que me pertenecen o parte de los que contienen esta obra, con fines estrictamente académicos o de investigación.

Los derechos que como autora me correspondan, con excepción de la presente autorización, seguirán vigentes a mi favor, de conformidad con lo establecido en los artículos 5, 6, 8; 19 y demás pertinentes de la Ley de Propiedad Intelectual y su Reglamento.

Guayaquil, 3 de Septiembre 2021

GONZALES ARIAS OMAR SET
C.I. 080381261-9

Índice general

Índice de Figuras

Resumen

Una vulnerabilidad es una debilidad del sistema informático que puede ser utilizada para causar un daño. Las debilidades pueden aparecer en cualquiera de los elementos de una computadora, tanto en el hardware, el sistema operativo, cómo en el software. Entre las miles de líneas de código que forman un programa, siempre hay algún fragmento que está mal diseñado o que es tan complejo que predice su comportamiento en todos los escenarios resulta una tarea casi imposible. Cuando no afecta al funcionamiento de la aplicación, ese fragmento de código puede quedarse sin detectar durante mucho tiempo, lo que obviamente retrasa su corrección. Incluso cuando el defecto se conoce, puede que no sea arreglado por falta de recursos. En el momento en que alguien malintencionado descubra el fallo, es posible que intente aprovecharlo con fines destructivos, convirtiéndolo en un auténtico agujero de seguridad, uno que pueda ser aprovechado mediante un ataque informático (*exploit*). Ningún programador diseña sus aplicaciones para que tengan agujeros de seguridad. Al contrario, la mayoría se esfuerza para que sus programas sean lo más seguros posible.

Palabras claves: Agujero de seguridad, exploit, fragmento, líneas de código, vulnerabilidad.

Abstract

A vulnerability is a weakness of the computer system that can be used to cause a damage. The weaknesses can appear in anyone of the elements of a computer, so much in the hardware, the operating system, how in the software. Among the thousands of code lines that form a program, there is always some fragment that is not well designed or that it is so complex that to predict their behavior in all the scenarios is an almost impossible task. When it doesn't affect to the operation of the application, that code fragment can be left without detecting during a lot of time, that that obviously retards its correction. Even when the defect is known, he/she can that it is not fixed by lack of resources. In the moment in that somebody bitchy he/she discovers the failure, it is possible that he/she tries to take advantage of it with destructive ends, transforming it into an authentic hole of security, one that can be taken advantage of by means of a computer attack (exploit). No programmer designs his applications so that they have holes of security. On the contrary, most makes an effort so that their programs are the surest possible.

Key words: Hole of security, exploit, fragment, code lines, vulnerability.

1. Introducción

1.1 Importancia o caracterización del tema

A finales de los años 60 aparecieron los ordenadores, los mismos que en la actualidad son de vital importancia. En el año 2000 se lanzó al mercado Windows 2000 y en el 2001 Microsoft Windows XP, estos sistemas fueron los que tuvieron mayores falencias de seguridad. Un sistema de información se considera seguro si se encuentra libre de todo riesgo y daño. Es imposible garantizar la seguridad absoluta de un sistema informático, por ello es preferible utilizar el término fiabilidad. Grupos de personas y organizaciones, algunas de tipo “underground” están en la búsqueda de fallos en sistemas operativos y aplicaciones informáticas, las lasitudes son reportadas por estas personas y a diario ellos exponen a grandes riesgos los sistemas afectados por esas amenazas, no importa el segmento de mercado a la que pertenezca la organización afectada. Los fallos son la piedra angular de la seguridad, puesto que suponen el origen del que derivan numerosos fallos de seguridad. Una extenuación en un programa informático o software es simplemente un error, un problema en su código o en su configuración, son difíciles de gestionar. Se descubren decenas día a día, y clasificarlas es una tarea compleja. El análisis de este problema se ha convertido en un requisito indispensable dentro del proceso de gestión de riesgos y es clave dentro del sistema de gestión de la seguridad de la información. Se define “Vulnerabilidad” como un error de software que puede usar directamente el intruso para ganar acceso a un sistema de información. Desde el desbordamiento de memoria hasta el consumo excesivo de memoria.

1.2 Actualidad del tema

En la actualidad tanto gobiernos como empresas se toman muy enserio la seguridad en sus sistemas, ya que a medida que avanza la tecnología, surgen nuevos ataques y defensas de parte de los gobiernos, hasta el punto de una guerra cibernética la cual se está palpando, siendo este el caso de China la cual quería lanzar unos misiles y estos supuestamente fueron deshabilitados mediante ciberataques.

Hoy en día se dan a conocer nuevos casos de robos de cuentas bancarias, estas se obtienen descubriendo el fallo de un sistema o bien de una página web, también mediante la ingeniería social la cual juega el mayor rol en un ataque.

La tendencia principal en los ciberataques modernos, eran los ataques conocidos como 0-day (del día cero, o zeroday). Este término hace referencia a los que aprovechan vulnerabilidades sin reparar para introducirse en un sistema.

1.3 Novedad científica

En vista de la innovación tecnológica y errores en los sistemas se han desarrollado programas o sistemas con herramientas de seguridad las cuales permiten tanto la protección como el ataque al mismo, sistema operativo Kali Linux cuenta con la mayoría de estas herramientas.

El aumento del número de programas maliciosos, así como delincuentes informáticos ha provocado una subida de la demanda de los profesionales de esta especialidad, las empresas quieren que sus servicios en la web y sus sistemas sean seguros.

1.4 Justificación del tema

Computadores, así como también la aparición de dispositivos móviles o tablets capaces de conectarse a la red no han hecho que aumentar aún más la

preocupación sobre este tema, debido a la gran cantidad de información personal que se puede llegar a guardar en estos dispositivos.

Hoy por hoy sino se toma la debida atención a la protección de nuestros datos estos se verán expuestos a riegos, con la posibilidad de que otra persona mal intencionada haga uso de ello.

1.5 Objetivos

1.5.1. Objetivo General

Analizar las amenazas y vulnerabilidades en los sistemas operativos Windows, mediante recopilación bibliográfica de Pentesting, para la protección integral de los datos e información.

1.5.2. Objetivo Específicos

- Analizar las amenazas de los sistemas operativos Windows.
- Dar a conocer los métodos de seguridad que emplea Windows.
- Identificar las herramientas y métodos para explotar las vulnerabilidades.

2. Aspectos metodológicos

2.1 Materiales

2.1.1 Recursos bibliográficos

Para la elaboración del trabajo monográfico se usó como fuente de información:

- Libros especializados sobre el tema del Centro de Información Agraria.
- Revistas.
- Artículos de diarios nacionales e internacionales.
- Otros trabajos de investigación que traten acerca de la misma temática como monografía y tesis obtenidas por medio del acceso al aula virtual de la Universidad Agraria del Ecuador en los diferentes repositorios o bibliografía virtual de Universidades o Institutos de investigación.
- Páginas y sitios de internet especializados en temas tecnológicos informáticos.

2.1.2 Materiales y equipos

- Internet.
- Computadora.
- Pendrive.
- Impresora.
- Hojas para impresión.

2.1.3 Recursos humanos

- El estudiante como postulante de la investigación.
- El docente guía asignado quien realizó las explicaciones y correcciones del trabajo.

2.2 Métodos

2.2.1 Modalidad y tipo de investigación

El presente proyecto de investigación es de modalidad monográfica, referente a recolección bibliográfica, respecto al tema “Análisis Descriptivo de las Vulnerabilidades en los sistemas operativos Windows (Pentesting)”, se analizó una presentación descriptiva de la información que hay al respecto, se dieron a conocer los diferentes puntos de vista, para luego poder realizar la opinión personal.

2.2.2 Tipos de métodos

2.2.2.1 *Método inductivo*

Es aquel método en el que se obtuvo conclusiones generales a partir de las premisas particulares. Con este método se pudo constatar todo lo relacionado con el análisis de un sistema operativo.

2.2.2.2 *Método analítico*

Conocido como el método de la descomposición de un todo en sus elementos constitutivos para proceder a su comprensión y estudio. Este método permitió mediante consultas de libros, revistas, web y fuentes de información conocer las características y funcionalidades detalladas que permitirán descubrir los fallos comunes que exponen a un equipo a riesgos.

2.2.2.3 *Método deductivo*

Es uno de los métodos más usados permitió conocer los distintos puntos de vista de una situación. Por lo cual este método permitió concretar cómo protegerse de un atacante mediante las vulnerabilidades comunes y peligrosas del sistema.

2.2.3 Técnicas

Con el propósito de analizar y dar a conocer la manera en que un sistema esta vulnerable, este trabajo monográfico se basó en el método analítico, la cual permitió conocer las ventajas de tener un pc actualizado, mediante el uso de libros especializados sobre el tema.

2.3 Marco legal

El Art. 1 de la Ley Orgánica de Comunicación tiene por objeto desarrollar, proteger, promover, garantizar, regular y fomentar, el ejercicio de los derechos a la comunicación establecidos en los instrumentos de derechos humanos y en la Constitución de la República del Ecuador.

El Art. 3 dispone que el contenido comunicacional para los efectos de esta ley, se entenderá por contenido todo tipo de información u opinión que se produzca, reciba, difunda e intercambie a través de los medios de comunicación social.

El Art. 4 menciona que la ley de contenidos personales en internet, no regula la información u opinión que de modo personal se emita a través de internet. Esta disposición no excluye las acciones penales o civiles a las que haya lugar por las infracciones a otras leyes que se cometan a través del internet (Lexis, 2019).

El Art. 18 de la Ley Orgánica de la Protección de Datos de acuerdo a la Seguridad de datos personales estipula que, los responsables y encargados de tratamiento de los datos personales deberán implementar todas las medidas de seguridad adecuadas y necesarias, sean estas técnicas, organizativas o de cualquier otra índole, para proteger los datos personales frente a cualquier riesgo, amenaza, vulnerabilidad, acceso no autorizados, pérdidas, alteraciones, destrucción o comunicación accidental o ilícita, atendiendo a la naturaleza de los datos de carácter personal, al ámbito y el contexto.

El Art. 55 de la Ley Orgánica de la Protección de Datos mediante la Notificación de vulneración de seguridad manifiesta que, el responsable del tratamiento deberá notificar la vulneración de la seguridad de datos personales a la Autoridad de Protección de Datos Personales, dentro del término de tres días a partir del conocimiento de dicha vulneración (Asambleanacional, 2019).

3. Análisis y revisión de literatura

3.1 Amenazas de los sistemas operativos Windows.

> La seguridad informática es una disciplina que se encarga de proteger la integridad y la privacidad de la información almacenada en un sistema informático, los programas dañinos que se instalan en la computadora del usuario por distintas vías, email, software ilegal creackeado y recompilado, parches para programas sin licencia, páginas web con código malicioso, etc (Dasit, 2018).

La seguridad informática es una estrategia o acción que nos permite navegar sin preocupación, al saber que se está protegiendo nuestra información.

> Dentro de los fabricantes más afectados por las vulnerabilidades aparece Microsoft en primer lugar. Se ha observado más de una docena de fallos que provocan corrupción en la memoria del navegador y una vulnerabilidad grave en el componente de Windows Structured Query, que permite la ejecución de código con los permisos del usuario registrado en el sistema (Itdigitalsecurity, 2018).

El administrador del sistema debe realizar constantemente la actualización y parcheo de sus servidores contra vulnerabilidades de Zero Day (o fallos sin reparar ante una actualización). Los parches no se limitan al sistema operativo, sino también deben incluir cualquier aplicación que se aloje en ellos.

> Las vulnerabilidades son fallos en un sistema que puede ser explotada por un atacante generando un riesgo para la organización o para el mismo sistema. Las vulnerabilidades físicas son las que van a afectar a la infraestructura de la organización de manera física, donde un usuario podría ingresar con una USB y copiar información, podría infectar la misma infraestructura; y las lógicas son las que van a afectar directamente la infraestructura y el desarrollo de la operación de estos, configuración, actualización, desarrollo (Romero, y otros, 2018).

Los administradores deben revisar periódicamente los equipos para verificar si no se encuentran anomalías como objetos sospechosos instalados en los equipos, los cuales no son autorizados por la empresa u organización.

- **Windows Server Update Services (WSUS).** Esta herramienta dispuesta por Microsoft de forma ideal para los productos Microsoft debe ser implementada en cualquier organización que tenga sistemas operativos Microsoft Windows. Ya que sus servicios a nivel de gestión de

> actualizaciones de productos Microsoft puede ser administrada de manera centralizada sin importar el tamaño de la organización o la cantidad de sedes o servidores que pueda tener, lo importantes es que estén conectadas entre sí de algún modo para que los servidores WSUS puedan sincronizarse (Coronado, 2018).

Proporciona un servicio de actualización de software para los sistemas operativos Microsoft Windows y otros softwares de Microsoft.

- **Microsoft Baseline Security Analyzer (MBSA).** Es una herramienta de auditoría fácil de usar, diseñada para determinar el estado de seguridad de conformidad con las recomendaciones de seguridad de Microsoft, ofreciendo orientación específica sobre la remediación. MBSA proporciona controles integrados que permiten determinar si las vulnerabilidades administrativas de Windows están presentes, si se están usando contraseñas débiles en las cuentas de Windows, la presencia de vulnerabilidades conocidas en IIS Server y SQL Server, y qué actualizaciones de seguridad se requieren en cada sistema individual. MBSA proporciona una evaluación dinámica de las actualizaciones de seguridad pendientes. MBSA puede escanear una o más servidores por dominio, rango de direcciones IP u otra agrupación. Una vez finalizado el análisis, MBSA proporciona un informe detallado y las instrucciones sobre la forma de ayudar a convertir el sistema en un entorno de trabajo más seguro (Tibocha, 2014).

Remite que el análisis de este problema se ha convertido en un requisito indispensable dentro del proceso de gestión de riesgos y es clave dentro del sistema de gestión de la seguridad de la información.

> A fines de marzo de 2018, investigadores de ESET identificaron una interesante muestra maliciosa en formato PDF que, al ser analizada, reveló que explotaba dos vulnerabilidades inéditas: una que permitía la ejecución remota de código en Adobe Reader y otra que permitía escalar privilegios en Microsoft Windows (Cherepanov, 2018).

Hace referencia a como un atacante puede incrustar una carga maliciosa aun programa, componente o documento de Windows, para luego llegar a tener control total del equipo mediante este método de ataque.

> Dentro de un mismo programa, una vulnerabilidad se localiza habitualmente en un componente o módulo. Los programas suelen componerse de varios módulos que interactúan entre sí. Una vulnerabilidad puede encontrarse en un módulo concreto del programa o bien por utilizar una configuración concreta. Por

> ejemplo, puede existir una vulnerabilidad en el módulo de interpretación de ficheros en formato RTF en Microsoft Word sin afectar al módulo que procesa otro tipo de ficheros. O en el módulo de procesado de ficheros MP3 en el programa de reproducción Winamp. Es posible que la vulnerabilidad no pueda ser aprovechada si este módulo no se encuentra activo. Por ejemplo, el módulo de procesamiento de JavaScript en documentos PDF no se encuentra activo por defecto en Adobe Reader (Telledo, 2017).

Las consecuencias técnicas de estos errores suelen ser diferentes. Desde el desbordamiento de memoria hasta el consumo excesivo de memoria. Estas consecuencias suelen ser la mayoría de las veces las mismas derivadas de iguales descuidos, y es lo que buscarán los cazadores de vulnerabilidades.

> Se refiere a que los sistemas de información de las organizaciones tienen multitud de recursos vulnerables ante ataques de seguridad. Por ello, es necesario que desarrollen estrategias y herramientas que sean capaces de identificar y valorar estos recursos y que, a su vez, puedan dar información sobre los ataques y daños que pueden afectarle (Chicano, 2014).

Hace referencia que la seguridad de nuestro sistema no debe depender de un solo mecanismo por muy fuerte que éste sea, sino que es necesario establecer varios mecanismos sucesivos. De este modo cualquier atacante tendrá que superar varias barreras para acceder a nuestro sistema.

> En la primera actualización principal de Windows 10, Microsoft sacó una nueva funcionalidad de seguridad para Edge: se trata de una opción que protege el navegador de la inyección binaria. Ahora, para cargar correctamente un archivo DLL en un navegador, la biblioteca debe estar firmada con un certificado digital de Microsoft o debe estar aprobada y certificada por Microsoft Windows Hardware Quality Lab (WHQL). La medida de seguridad bloquea la carga de todas las demás bibliotecas en el contexto de los procesos del navegador que no cumplan con dichas condiciones (Eset, 2015)

El objetivo es que cualquier problema que se produzca dentro del entorno controlado no afecte a los procesos fuera de ese entorno; ejemplo, que el desbordamiento de búfer de un proceso interno no implique un desbordamiento del sistema (Tejada, 2014).

Se refiera a que para mitigar un problema hay que tener en cuenta, primero, los derechos de las aplicaciones que se ejecutarán y, segundo, fortalecer lo máximo posible el software que gestiona ese entorno controlado.

3.1.1 Naturaleza de las amenazas

- ***Modificación:*** También llamados “web defacement”; Esta técnica propone un algoritmo para la detección de modificaciones de sitios web implementando un navegador web con técnicas de detección de Defacement incorpo-radas (Pulido, 2016).
- ***Fabricación:*** comprometen la integridad del sistema por ejemplo al insertar un nuevo usuario en el sistema operativo.
- ***Interceptación:*** Busca comprometer la confidencialidad del sistema, un ejemplo son los key loggers o spyware y los Sniffers.
- ***Interrupción:*** puede provocar que un objeto del sistema se pierda, quede no utilizable o no disponible, un ejemplo serían los ataques de denegación de servicios o DoS (Aguirre, 2016).

3.1.2 Propiedades de un sistema seguro

- Confidencialidad: consiste en asegurar que sólo el personal autorizado accede a la información que le corresponde, de este modo cada sistema automático o individuo solo podrá usar los recursos que necesita para ejercer sus tareas, para garantizar la confidencialidad se recurre principalmente a tres recursos: Autenticación de usuario, Gestión de privilegios, Cifrado de información; un ataque contra la confidencialidad es el uso de la herramienta sniffers.

- Integridad: Los recursos del sistema solo pueden ser modificados o alterados por los elementos autorizados, un ataque a la integridad es crakear o descompilar un programa.

Este principio garantiza la autenticidad y exactitud de la información en cualquier momento que se solicitó o se envía de un entorno tecnológico en que los datos no han sido alterados o destruidos de forma no autorizada (Avenía, 2017).

- Disponibilidad: Los recursos del sistema deben permanecer accesibles a los elementos autorizados, un posible ataque es la denegación de servicios (Macmillan, 2019)**.**

Significa que un equipo está protegido si cuenta como mínimo con la; confidencialidad, integridad y disponibilidad, para que los datos e información esten protegidos.

3.1.3 Principales fallos que afectaron a Windows

- **CVE-2017-0199:** Permiten a los atacantes remotos ejecutar código arbitrario a través de un documento elaborado, también conocido como "Microsoft Vulnerabilidad de ejecución remota de código de Office / WordPad con API de Windows". La vulnerabilidad podría permitir la divulgación de información si un atacante envía una solicitud especialmente diseñada a un servidor ADFS, lo que le permite leer información confidencial sobre el sistema de destino (Incibe-Cert, 2017).

- **CVE-2016-3298:** Presente en Internet Explorer, permite la obtención de información y permite a los atacantes remotos determinar la existencia de archivos arbitrarios a través de un sitio web diseñado, también conocido como 'Divulgación de información de Internet Explorer Vulnerabilidad' (Nvd, 2016).

- **CVE-2016-7189:** esta vulnerabilidad que se soluciona con el boletín MS16-119; esta actualización de seguridad resuelve vulnerabilidades en Microsoft Edge. La más grave de las vulnerabilidades podría permitir la ejecución remota de código si un usuario visita una página web especialmente diseñada con Microsoft Edge. Un atacante que explotara con éxito las vulnerabilidades podría obtener los mismos derechos de usuario que el usuario actual. Los clientes cuyas cuentas están configuradas para tener menos derechos de usuario en el sistema podrían verse menos afectados que los usuarios con derechos de usuarios administrativos (Microsoft, 2016).

- **CVE-2016-3393:** en este caso, la vulnerabilidad se encuentra en Windows Graphics Component y permitía la explotación a través de la web o mediante un archivo malicioso. La misma es solucionada por el boletín MS16-120 (Cve, 2016).

Microsoft publica boletines informativos acerca de sus productos, para corregir las vulnerabilidades o fallos de seguridad que se detectan en ellos.

3.1.3.1 Métodos de ataque

- **Scanning:** La gestión o escaneo de vulnerabilidades: desinfección de entornos de código mediante revisión de código y pentest. Por lo general, se realiza después de actualizar una aplicación (Adidas, 2019).

Con este método se descubren todos los puertos que están al escucha, es decir aquellos puertos que están abiertos a la espera de una conexión.

- **TCP Connect:** Identifica puertos TCP que estén escuchando, no requiere privilegios de root para ejecutarse (Gómez, s.f).

Esta es la forma básica del escaneo de puertos TCP. Si el puerto está escuchando, devolverá una respuesta de éxito; cualquier otro caso significará que el puerto no está abierto o que no se puede establecer conexión con él.

- **Fuerza Bruta:** Es la forma de recuperar una clave probando todas las combinaciones posibles hasta encontrar aquella que permite el acceso. Consiste en generar el diccionario (hash) de todas las posibles combinaciones y compararlas con el patrón (hash) que permita el acceso.

> Técnicamente, el término Hash se refiere a una función o método para generar claves que representen de manera casi unívoca a un documento, registro, archivo, etc. El objetivo de este ataque es ingresar al sistema de la víctima con credenciales (nombre de usuario y contraseña) y haciendo uso de una conexión remota (i.e., ssh,telnet, etc.) (Fuertes, Zapata, Ayala, y Mejía, 2015).

Para la generación de ataques a contraseña se emplearon dos herramientas, Medusa y John The Ripper, que tienen como objetivo común obtener usuarios y contraseñas inseguras dentro de un servidor.

3.1.4 Medidas de seguridad

Las medidas de seguridad se clasifican en cuatro niveles:

1. **Físico**: Se refiere a toda medida orientada a la protección de instalaciones, equipos, soportes o sistemas de datos para la prevención de riesgos por caso fortuito o causas de fuerza mayor.
2. **Lógico**: Se refiere a las medidas de protección que permiten la identificación y autentificación de las personas o usuarios autorizados para el tratamiento de los datos personales de acuerdo con su función.
3. **Desarrollo y aplicaciones:** Autorizaciones con las que deberá contar la creación o tratamiento de sistemas de datos personales para garantizar el adecuado desarrollo y uso de los datos.
4. **Cifrado**: Es implementación de algoritmos, claves, contraseñas, así como dispositivos concretos de protección que garanticen la integralidad y confidencialidad de la información (Carrales, 2016).

Estas medidas de seguridad le permiten al sistema navegar por la red de manera segura, impidiendo que alguien no autorizado capte sus datos o información.

3.2 **Métodos de seguridad que emplea Windows**

3.2.1 Método de cifrado en Windows

3.2.1.1 Cifrado de datos con EFS

> Se llama EFS (*Encrypted File System*) Está diseñado como el cifrado del sistema de archivos NTFS y está enfocado en el cifrado solo de datos individuales. Al cifrar un archivo con EFS, la encriptación de este elemento está vinculada de forma directa al usuario de Windows 10 que ha realizado el proceso, de modo que si un usuario diferente está conectado al usuario que cifró los archivos, esos archivos permanecerán inaccesibles para este aumentado así la privacidad de estos (Solvetic, 2017).

Con este método la información está cifrada con mayor seguridad, así mismo a los pendrive que tiene el formato FAT32 se los puede transformar a NTFS para cuenten con esa seguridad.

3.2.1.2 **Cifrado de datos con *BitLocker***

Permite proteger los datos en el dispositivo para que solo puedan acceder a ellos personas con autorización. Si el cifrado de dispositivos no está disponible en el dispositivo, probablemente puedas activar el cifrado de BitLocker estándar en su lugar (Microsoft, 2019).

Windows cuenta con BitLocker, una aplicación que tiene como objetivo cifrar cualquier unidad de disco que le especifiquemos, incluido los sistemas de Windows necesarios para el inicio del equipo y sesiones contenidos en el disco de arranque del sistema (Eset, 2014).

> En la actualidad Microsoft permite cifrar el disco duro completo de un equipo con BitLocker. La clave de cifrado está almacenada en un chip TPM o en una memoria USB. Los datos no se pueden modificar offline. Pero todavía es posible arrancar el sistema y atacar los servicios expuestos. Para paliar esto, Microsoft permite añadir un código PIN que no se guarda en el equipo. El usuario puede definirlo y sin este código el sistema no arranca. En apariencia, es un buen sistema, pero hace falta que el usuario sea administrador para poder cambiarlo, lo que para nosotros es una pena. Crearemos una aplicación que, una vez instalada, permitirá a un usuario sin privilegios de administrador cambiar el código PIN de BitLocker (Kapfer, 2018).

Este método permite poner contraseñas a una partición de memoria; Ejemplo (C:, D:, E:, entre otros), para que la información contenida en esa unidad solo la pueda ver el que tiene la contraseña.

3.2.2 Criptografía

> La palabra criptografía se explica cómo cuando se trata de hablar de códigos y lenguajes secretos. La criptografía ha sido usada a través de los años para mandar mensajes confidenciales cuyo propósito es que sólo las personas autorizadas puedan entender el mensaje (Díaz, Alzórriz, Sancritóbal, y Castro, 2014).

Cualquier persona que quiere mandar información confidencial aplica técnicas criptográficas para poder “ocultar” el mensaje (o encriptar), manda el mensaje por una línea de comunicación que se supone insegura y después solo el receptor autorizado pueda leer el mensaje “escondido” (lo llamamos descifrar o descencriptar).

La criptografía es la creación de técnicas para el cifrado de datos. Teniendo como objetivo conseguir la confidencialidad de los mensajes (Universidad Privada Alfonso X El Sabio, s.f).

La criptografía actual se inicia en la segunda mitad de la década de los años 70. No es hasta la invención del sistema conocido como DES (Data Encryption Standard) en 1976 que se da a conocer más ampliamente, principalmente en el mundo industrial y comercial.

3.2.2.1 Problemas que resuelve la criptografía

- **Privacidad**

> Todos los sistemas de cifrado están basados en el concepto de clave. Una clave es la base de una transformación, normalmente matemática, de un mensaje ordinario en un mensaje ilegible. En la comunicación por Internet es muy difícil estar seguros de la privacidad de los datos e información, ya que no se tiene control de la línea de comunicación. Por lo tanto, al cifrar (esconder) la

información cualquier intercepción no autorizada no podrá, desvelar la información (IBM, 2014).

Se refiere a que la información va cifrada y sólo pueda ser leída por personas autorizadas, es decir a quien se les envía cierta información.

- **Integridad**

Una función hash es una función aplicada a un mensaje de tamaño variable genera una representación de tamaño fijo del propio mensaje**.** Una función hash unidireccional es una función hash H de modo que para cualquier mensaje es difícil encontrar un mensaje m tal (Lorge, 2015).

Entendida como la cualidad de un documento para estar completo y sin alteraciones, con la cual se asegura que el contenido y atributos están protegidos a lo largo del tiempo. Es uno de los componentes que conforman la confianza del documento (Rangel, 2017).

Se basa en que la información no pueda ser alterada en el transcurso de ser enviada.

- **Autenticidad**

Definir que la información requerida es válida y utilizable en tiempo, forma y distribución (Moran, 2016).

Se trata de que la información enviada llegue a su destino sin alteración alguna.

3.2.2.2 Tipos de criptografía

- **Simétrica**

Consiste en la distribución de una misma clave para la comunicación entre el emisor y el receptor, la clave asignada se utiliza para encriptar y desencriptar el mensaje. La idea general es aplicar diferentes funciones al mensaje que se quiere cifrar de tal modo que solo conociendo una clave pueda aplicarse de forma inversa para poder así descifrar (Velasco, Jiménez, y Chafla, 2016).

La criptografía simétrica es un método criptográfico mono-clave, esto quiere decir que se usa la misma clave para cifrar y descifrar. Esto supone un grave problema a la hora de realizar el intercambio entre el emisor y el receptor, dado

que si una tercera persona estuviese escuchando el canal podría hacerse con la clave, siendo inútil el cifrado (Corrales, Cilleruelo, y Cuevas, 2014).

Se refiere al conjunto de métodos que permiten tener comunicación segura entre las partes siempre y cuando anteriormente se hayan intercambiado la clave correspondiente que llamaremos clave simétrica.

- **Asimétrica**

Es por definición aquella que utiliza dos claves diferentes para cada usuario, una para cifrar que se le llama clave pública y otra para descifrar que es la clave privada. El nacimiento de la criptografía asimétrica se dio al estar buscando un modo más práctico de intercambiar las llaves simétricas. Diffie y Hellman, proponen una forma para hacer esto, sin embargo no fue hasta que el popular método de Rivest Shamir y Adleman RSA publicado en 1978, cuando toma forma la criptografía asimétrica, su funcionamiento está basado en la imposibilidad computacional de factorizar números enteros grandes (Jiménez, Velasco, y Chafla, 2016).

Éste tipo de criptografía genera un par de claves complementarias en donde una cifra el mensaje y sólo otra lo puede descifrar, esas claves son: privada la cual sólo ha de conocer el propietario de la misma y una clave pública que está disponible para todos los usuarios del sistema (García, 2016).

- ***Longitud de la clave***

Existe una gran discusión, sobre este aspecto pero sin duda en la actualidad se acepta que es recomendable usar claves de longitud 768 (231 dígitos) para actividades personales, 1024 bits (308 dígitos) para corporaciones y 2048 (616 dígitos) para actividades de alto riesgo. La longitud de las claves tiene que ver con la seguridad del sistema si el número *n* pudiese ser factorizado entonces sin mucha dificultad puede calcular a d a partir de e, p, y q por lo tanto descifrar cualquier mensaje (Chafla, Jiménez, y Velasco, 2016).

Dependiendo de la longitud de la clave será difícil que se descifre un logaritmo o en ingreso a un sistema.

3.3 Herramientas y métodos para explotar vulnerabiladades

3.3.1 Contextualización

Al emplear sistemas de cómputo en tareas críticas de actividades con impacto cada vez mayor, se necesita asegurar que estos sistemas operan de manera adecuada para evitar que perjudique a las mismas actividades o a otras que estén asociadas (Llaven, 2015).

Esto no quiere decir que uno es menor que otro, quiere decir que cada uno tiene características deseables para hacer una tarea específica, cada sistema es especializado para ciertos servicios.

3.3.2 Metasploit

> El Proyecto Metasploit es utilizado para manejar la base de datos de Metasploit, manejar las sesiones, además de configurar y ejecutar los módulos de Metasploit; representa un conjunto de herramientas que ayuda a los profesionales de seguridad y hacker a llevar a cabo ataques informáticos de manera sistematizada y automatizada. Esta herramienta permite conectarse al objetivo de tal manera que se puedan ejecutar los exploits contra este (ver figura 1) (Caballero, 2015).
>
> Su más conocido sub-proyecto es el marco de código abierto Metasploit, una herramienta para el desarrollo y ejecución de código de explotación en contra de un equipo o sistema de información destino remoto. Otros importantes sub-proyectos son la base de datos Opcode, archivo shellcode, e investigaciones de seguridad. La consola de metasploit framework permite ejecutar pruebas de penetración, recolección de información sobre un objetivo determinado, explotación de vulnerabilidades concretas (Hacking, 2014).
>
> Un Payload, es un programa que acompaña a un exploit para realizar funciones específicas una vez el sistema objetivo es comprometido, la elección de un buen payload es una decisión muy importante a la hora de aprovechar y mantener el nivel de acceso obtenido en un sistema (Backtrackacademy, 2016).

Esta herramienta nos permite verificar las vulnerabilidades de ciertos sistemas operativos, para luego ser explotados.

3.3.2.1 Automatización de tareas en Metasploit

En Metasploit se puede crear scripts que automaticen las tareas que se desee llevar a cabo, y luego llamarlos directamente desde la consola con el comando “resource filename.rc” o directamente desde la Shell de Linux con “msfconsole -r filename.rc” (ver figura 3) (Mollar, s.f).

Metasploit provee un entorno de trabajo verdaderamente impresionante. El MSF es mucho más que una colección de exploits, es una infraestructura donde se puede utilizar para necesidades específicas. Esto permite conectarse en un único entorno (ver figura 4) (Ayllón, 2014).

3.3.3 Nessus

> Es un programa de escaneo de vulnerabilidades en diversos sistemas operativos. Consiste en un daemon, nessusd, que realiza el escaneo en el sistema objetivo, y nessus, el cliente (basado en consola o gráfico) que muestra el avance e informa sobre el estado de los escaneos (ver figura 2) (Creadpag, 2018).

Nessus es un programa que busca cualquier tipo de vulnerabilidad en la red de Windows, Linux, ofrece un análisis completo de su nivel de seguridad. Su eficacia supera los límites de otras herramientas similares.

3.3.3.1 Ventajas de Nessus

- Escaneo de alta precisión con bajo número de falsos positivos.
- Capacidades y características de escaneo integrales.
- Escalabilidad a cientos de miles de sistemas.
- Implementación y mantenimiento sencillos.
- Bajo costo de administración y operación (Tenable, 2017).

3.3.4 Rompiendo claves con John The Ripper

> Este es uno de los crackers más populares de contraseñas. Esta clase de información debe ser accesible a cualquier persona en forma pública pues esto permite que la humanidad tenga conciencia de que nuestros sistemas críticos

del negocio están siendo accedidos por usuarios con contraseñas débiles que así mismo debilitan nuestra seguridad de la información (Gutiérrez, 2014).

Estas herramientas nos permiten a los administradores del sistema comprobar la solidez de las contraseñas para disminuir ataques por fuerza bruta, y ataques de diccionarios. Es decir, permiten la comprobación proactiva de las contraseñas. El conocer nuestras debilidades nos permite mejorar las políticas de seguridad.

La principal finalidad de este tipo de programas es detectar passwords débiles que vulneren la seguridad del sistema. Ahora cualquier elemento puede ser utilizado para fines buenos o malos, solo nosotros decidimos el uso que le damos (Espino, 2014).

3.3.4.1 Tipos de ataques de contraseñas

- ***Por diccionario:***

  Los ciberdelincuentes operan una lista de palabras con la expectativa y el optimismo de que la contraseña se puede obtener de las visitas previas al sitio web. Estos ataques también se consideran óptimos para las contraseñas que se basan en palabras fáciles. Los administradores necesitan usar una contraseña compleja que no sea la fecha de nacimiento o el número de teléfono del usuario, pero la combinación de fecha de nacimiento y número de teléfono es posible.

- ***Ataque por fuerza bruta***

  Se prueban todas las combinaciones posibles, sean palabras o no. Éste es el sistema más lento, y usado sólo en casos concretos, dado que los sistemas anteriores (el ataque por diccionario) ya permiten descubrir muy rápidamente las contraseñas débiles.

- ***Ataque mixto***

  Prueba con variaciones de las palabras del diccionario: les añade números, signos, mayúsculas y minúsculas, cambia letras, combina palabras, etc. Luego de esto el funcionamiento es relativamente simple, a través de los módulos internos se cifra la palabra del diccionario o la combinación actual en un ataque de fuerza bruta, se cifra y se compara con la clave a crackear (Enviralizate, 2019).

Los Diccionarios son archivos con millones de palabras, las cuales pueden ser passwords utilizadas por los usuarios. Este archivo es utilizado para descubrir dicha password en pruebas de fuerza bruta.

3.3.5 Ingeniería social como método de ataque

Es un conjunto de técnicas que se basan en el aprovechamiento de los errores y/o faltas de atención y precaución de los usuarios para acceder a información confidencial, privada de los mismos y luego, poder utilizarla a conveniencia (Ba-csirt, 2018).

Network Security está pendiente de los errores del sistema para luego que llegue una actualización corregir aquellos errores.

3.3.6 Meterpreter

Meterpreter es bastante robusto a la hora de manipular la memoria de una víctima y los procesos cargados en ella, este nivel de potencia es alcanzado gracias a la definición de scripts meterpreter escritos en Ruby, ya que le permite al desarrollador crearlos y desplegarlos en metasploit o utilizar algunos existentes para diversos fines. Se han incluido herramientas como Volatility FrameWork y PMDump (Offensive-Security, 2017).

Permite manipular la memoria del equipo al cual se está atacando para luego obtener el control completo del sistema.

3.3.7 Netcat

Permite realizar cualquier tipo de conexión a un servicio a un puerto determinado. Esto convierte a netcat en una verdadera navaja suiza, pudiéndose usar como escáner de puertos para conocer su estado, como servidor proxy o para transferir

archivos, además permite crear fácilmente puertas traseras, los *backdoors* (Olivares y Oncins, 2015).

Los backdoors son pequeños programas pasan desapercibidos en el sistema y que permite al hacker volver a conectarse al sistema siempre que lo desee.

4. Conclusiones

Mediante esta investigación se determinó la importancia de tener un sistema actualizado y seguro. Actualmente, la tecnología está cambiando aceleradamente y de forma incontrolable, convirtiéndose en el medio más rápido para transferir datos e información, sean estos mediante; correos, cuentas bancarias, etc.

En vista que Windows se lo encuentra como software de fácil acceso, es el sistema más atacado en todo el mundo, no es el más vulnerable, sino que es el más utilizado, de allí que los atacantes desarrollan sus virus para un sistema en común (Windows).

Existen herramientas con la que podemos exponer la seguridad de nuestro sistema, comprobando la seguridad de un programa antes de instalar, un correo y el tráfico de la red.

5. Recomendaciones

Es recomendable el uso de programas antivirus debidamente actualizados; tenga en cuenta que cada día aparecen nuevos virus los cuales pueden estar enfocados en dejar inutilizable cualquier antivirus, para luego tener el control total de un sistema.

De forma periódica, los fabricantes de los sistemas operativos y de los programas publican actualizaciones o “parches” de sus productos. Estas actualizaciones solucionan desde pequeños defectos de funcionamiento hasta graves brechas de seguridad.

El sistema operativo provee un conjunto de herramientas con la que podemos hacer pentesting o un test de seguridad más eficiente y seguro, siendo este el más utilizado y recomendado por los expertos en ciberseguridad.

6. Bibliografía citada

Adidas, W. (2019). Gestión de Vulnerabilidades. En *Lo esencial del Hackeo* (pág. 32). Antioch, Estados Unidos: I Editorial.

Aguirre, E. (2016). Naturaleza de las amenazas. *Vulneración en la información en sistema administrativo del Ministerio de Transportes y Comunicaciones*, 16-17. Lima, Perú: Universidad Nacional de San Marcos.

Asambleanacional. (2019). Seguridad de Datos Personales. *Proyecto de Ley Orgánica de Protección de Datos Personales*, 16 - 25. Quito, Ecuador: Secretaria Nacional.

Avenía, C. A. (2017). Principios de la seguridad informática. *Fundamentos de seguridad informática*, 30 - 33. Bogotá, Colombia: Areandino.

Ayllón, J. (2014). Conectar Metasploit. *Curso Metasploit*, 9-10. Madrid, España: Universidad Privada del Norte.

Backtrackacademy. (2016). *Msfconsole*. Obtenido de Metasploit: Atacando a Windows: https://backtrackacademy.com/articulo/metasploit-atacando-a-windows

Ba-csirt. (2018). Ciberataques: Las estratégias delictivas del mundo digital. *Ataques Cibernéticos*, 1-4. Buenos Aires, Argentina: Bacsirt.

Caballero, A. (2015). Metasploit Framework. *Hacking con Kali Linux*, 45-64. Lima, Perú: ENI.

Carrales, J. (2016). Medidas de seguridad. *Guía de medidas de seguridad*, 1-5. Veracruz, México: Ivai.

Chafla, G., Jiménez, M., y Velasco, P. (2016). Longuitud de la clave. *Análisis de los mecanismos de encriptación para la seguridad de la información en redes de comunicaciones*, 98-99. Guayaquil, Ecuador: SATHIRI.

Cherepanov, A. (2018). *Dos vulnerabilidades zero-day en una misma muestra que afecta Adobe Reader y Windows*. Obtenido de Welivesecurity : https://www.welivesecurity.com/la-es/2018/05/21/vulnerabilidades-zero-day-misma-muestra-afecta-adobe-reader-windows/

Chicano, E. (2014). Análisis de riesgos de los sistemas de información. En *Auditoría de seguridad informática (MF0487_3)* (Primera ed., págs. 97-314). Málaga, España: IC Editorial.

Coronado, C. (2018). Selección de una herramienta de distribución de actualizaciónes de seguridad. *Metodología de aseguramiento a Sistemas Operativos Server*, 51. Bogotá, Colombia: Universisdad Nacional Abierta y a Distancia UNAD.

Corrales, H., Cilleruelo, C., y Cuevas, A. (2014). Criptografía simétrica:. *Criptografía y Métodos de Cifrado*, 7-10. Madrid, España: Universidad de Alcalá.

Creadpag. (2018). *¿Que es Nessus?* Obtenido de Buscar vulneravilidades en Kali Liux con Nessus: https://www.creadpag.com/2018/05/buscar-vulnerabilidades-en-kali-linux.html

Cve. (2016). *National Vulnerability Database (NVD)*. Obtenido de CVE - CVE-2016-3393: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2016-3393

Dasit. (2018). *¿Qué es seguridad informática?* Obtenido de ¿Qué es la seguridad informática? | Dasit: https://www.dasit.es/que-es-la-seguridad-informatica/

Díaz, G., Alzórriz, I., Sancritóbal, E., y Castro, M. (2014). Conocimiento y Competencias adquiridas. En *Procesos y herramientas para la seguridad de redes* (pág. 5). Madrid, España: Universidad Nacional de Educación a Distancia.

Enviralizate. (2019). *Tipos de ataques*. Obtenido de Diferencia entre fuerza bruta y ataque de diccionario: https://www.enviralizate.com/diferencia-entre-fuerza-bruta-y-ataque-de-diccionario/

Eset. (2014). Cifrado en Windows. *Guía de cifrado*, 8-9. California, Estados Unidos: Enjoy Safer Technology.

Eset. (2015). Explotación de Vulnerabilidades en Windows Durante 2015. *Explotación de Vulnerabilidades en Windows - WeLiveSecurity*, 10. Mexico, Mexico: Enjoy Safer Technology.

Espino, M. (2014). John The Ripper. En *Fundamentos de auditoría* (págs. 217-297). San Juan, México: Grupo Editorial Patria.

Fuertes, W., Zapata, P., Ayala, L., y Mejía, M. (2015). Tipos de ataques reales de redes ip evaluados. En *Plataforma de experimentaciñon de ataques reales a redes ip* (pág. 3). Sangolquí, Ecuador: SATHIRI.

García, R. (2016). Criptografía Asimétrica. *Firmas digitales basadas en funciones Hash y un algoritmo criptográfico híbrido*, 17-18. México, CIDETEC, México.

Gómez, R. (s.f). Tcp Connect. *La herramienta nmap*. Monterrey, México: Tecnológico de Monterrey.

Gutiérrez, F. (2014). John The Ripper. *Laboratorio de seguridad Informática con Kali Linux*, 52-59. Segovia, España: Universidad de Valladolid.

Hacking. (2014). *Metasploit*. Obtenido de Curso de Metasploit v0.3 AT4SEC: http://www.hacking.land/2014/07/curso-de-metasploit-v03-at4sec-espanol.html

IBM. (2014). *Privacidad de los datos*. Obtenido de Criptografía de clave pública: https://www.ibm.com/support/knowledgecenter/es/SSMKHH_9.0.0/com.ibm.etools.mft.doc/ac55940_.htm

Incibe-Cert. (2017). *Vulnerabilidad en Microsoft Office y múltiples versiones de Windows (CVE-2017-0199)*. Obtenido de CVE-2017-0199 | INCIBE-CERT: https://www.incibe-cert.es/alerta-temprana/vulnerabilidades/cve-2017-0199

Itdigitalsecurity. (2018). *Los fallos de seguridad en sistemas operativos, aplicaciones o en el propio firmware de los dispositivos*. Obtenido de Las vulnerabilidades en sistemas operativos y aplicaciones, entradas favoritas de los delincuentes: https://www.itdigitalsecurity.es/vulnerabilidades/2018/03/las-vulnerabilidades-en-sistemas-operativos-y-aplicaciones-entradas-favoritas-de-los-delincuentes

Jiménez, M., Velasco, P., y Chafla, G. (2016). Criptografía Asimétrica. *Análisis de los mecanismos de encriptación para la seguridad de la Información en redes de comunicaciones*, 93-94. Guayaquil, Ecuador: SATHIRI.

Kapfer, P. (2018). Cambiar el código PIN BitLocker con permisos de usuario. En *Internal Hacking y contramedidas en entorno Windows: Pirateo interno,*

medidas de protección, desarrollo de herramientas (Segunda ed., pág. 570). Barcelona, España: ENI.

Lexis. (2019). Ley Orgánica de Comunicacion. *Ley Orgánica de Cominucación - Consejo de Regulación*, 3 - 4. Quito, Ecuador: Asamblea Nacional.

Llaven, D. (2015). Panorama de seguridad informática. En *Sistemas operativos: panorama para la ingeniería en computación e informática* (págs. 273-326). Guadalajara, México: Grupo Editorial Patria.

Lorge, F. (2015). Funciones Hash. En *Introducción a la seguridad en redes de datos* (págs. 35-36). Buenos Aires, Argentina: Anaya Multimedia.

Macmillan. (2019). Principios de seguridad informática. *Introducción a la seguridadinformática*, 13-15. Madrid, España: Macmillan Education.

Microsoft. (2016). *Cumulative Security Update for Microsoft Edge (3192890)*. Obtenido de Microsoft Security Bulletin MS16-119 - Critical: https://docs.microsoft.com/en-us/security-updates/securitybulletins/2016/ms16-119

Microsoft. (2019). *Activar el cifrado de dispositivo*. Obtenido de Cifrado de datos con BitLocker: https://support.microsoft.com/es-us/help/4028713/windows-10-turn-on-device-encryption

Mollar, I. (s.f). Automatización de tareas en Metasploit. . *Curso de Metasploit*, 30-31. Castellón, España: Advanced Technologies for Security .

Moran, F. (2016). Elementos. *Seguridad Informática*. México, México: Dohub.

Nvd. (2016). *CVE-2016-3298 Detail*. Obtenido de National Vulnerability Database: https://nvd.nist.gov/vuln/detail/CVE-2016-3298

Offensive-Security. (2017). *Using Meterpreter Commands*. Obtenido de Meterpreter Basic Commands: https://www.offensive-security.com/metasploit-unleashed/meterpreter-basics/

Olivares, J., y Oncins, A. (2015). Utiliar buenas herramientas. En *Seguridad informática Hacking Ético* (págs. 43-45). Barcelona, España: ENI.

Pulido, O. (2016). Implementar un navegador web con técnicas de detección de Defacement. *Web Defacement*, 26-27. Cartagena, Colombia: Universidad Tecnológica de Bolívar.

Rangel, E. (2017). Integridad. *Guía para la gestión de documentos y expedientes electrónicos*, 21-23. Bogotá, Colombia: MINTIC.

Romero, M., Figueroa, G., Vera, D., Álava, J., Parrales, G., Álava, C., ... Castillo, M. (2018). Las Vulnerabilidades. *Introducción a la seguridad informática y el análisis de vulnerabilidades*, 41-45. Manabí, Ecuador: Área de Innovación y Desarrollo.

Solvetic. (2017). *Que es EFS*. Obtenido de Cómo encriptar archivos o carpetas con EFS en Windows 10: https://www.solvetic.com/tutoriales/article/4252-como-encriptar-archivos-o-carpetas-efs-windows-10/

Tejada, E. (2014). Control de código malicioso. En *Gestión de incidentes de seguridad informática* (pág. 312). Málaga, España: IC Editorial.

Telledo, J. V. (2017). Vulnerabilidades en programas y parches. En *MF0221_2 - Instalación y configuración de aplicaciones informáticas* (págs. 126-127). Madrid, España: Paraninfo.

Tenable. (2017). La ventaja Nessus. *Nessus Manager extiende el poder de Nessus a los equipos de auditoría y seguridad*. Columbia, Estados Unidos: Inc.

Tibocha, S. (2014). Actualizaciones de Windows. *Guía de aseguramiento de servidores Microsoft Windows Server 2008 R2*, 6-9. Bogota, Colombia: Agencia Nacional de Hidrocarburos.

Universidad Privada Alfonso X El Sabio. (s.f). Criptografía. *Criptografía: Técnica y aplicaciones.* Madrid, España: UAX.

Velasco, P., Jiménez, M., y Chafla, G. (2016). Criptografia Simétrica. *Análisis de los mecanismos de encriptación para la seguridad de la información en redes de comunicaciones*, 92-93. Guayaquil, Ecuador: SATHIRI.

7. Glosario

ActiveX: Un sistema de tecnologías creadas por Microsoft para permitir contenido interactivo en sitios web.

ADFS: Active Directory Federation Services. Permite es establecer un único inicio de sesión para el acceso a todos los servicios de la red de forma que el usuario no deba estar haciendo un login para acceder a diferentes servicios.

Crackear: En términos informáticos se refiere a explotar o modificar un software, haciendo que este funcione de otra manera.

Defacement: Consiste en la modificación de un sitio web sin la autorización del administrador de la misma.

EFS: El Encrypting File System (EFS) es un sistema de archivos que, trabajando sobre NTFS, permite cifrado de archivos a nivel de sistema.

Exploit: Es el nombre con el que se identifica un programa informático malicioso, o parte del programa, que trata de forzar alguna deficiencia o vulnerabilidad (bug) del sistema.

Fragmento: Se entiende por fragmento a toda aquella parte que compone un elemento superior y que fue voluntaria o involuntariamente separada del resto por determinada razón.

Kali Linux: Sistema operativo diseñado y utilizado por expertos informáticos para la auditoria de seguridad.

Key loggers: Es un tipo de software o un dispositivo hardware específico que se encarga de registrar las pulsaciones que se realizan en el teclado, para posteriormente memorizarlas en un fichero o enviarlas a través de internet**.**

Open source: Se refiere a los programas informáticos que permiten el acceso a su código de programación, lo que facilita modificaciones por parte de otros programadores ajenos a los creadores originales del software en cuestión.

Pentesting: Es la práctica de atacar diversos entornos con la intención de descubrir fallos, vulnerabilidades u otros fallos de seguridad, para así poder prevenir ataques externos hacia esos equipos o sistemas.

Parches de seguridad: Conjunto de ficheros adicionales al software original de una herramienta o programa informático, que sirve para solucionar sus posibles carencias, vulnerabilidades, o defectos de funcionamiento.

Plugins: Es aquella aplicación que, en un programa informático, añade una funcionalidad adicional o una nueva característica al software.

Sniffers: Es un programa informático que registra la información que envían los periféricos, así como la actividad realizada en un determinado ordenador.

Spyware: Es un software que recopila información de un ordenador y después transmite esta información a una entidad externa sin el conocimiento o el consentimiento del propietario del ordenador.

Scripts: Es un documento que contiene instrucciones, escritas en códigos de programación. Es un programa usualmente simple, que por lo regular se almacena en un archivo de texto plano.

Underground: Se trata de movimientos contraculturales que se encuentran fuera de las tendencias de moda.

Vulnerabilidad: Es una debilidad del sistema informático que puede ser utilizada para causar un daño.

8. Anexos

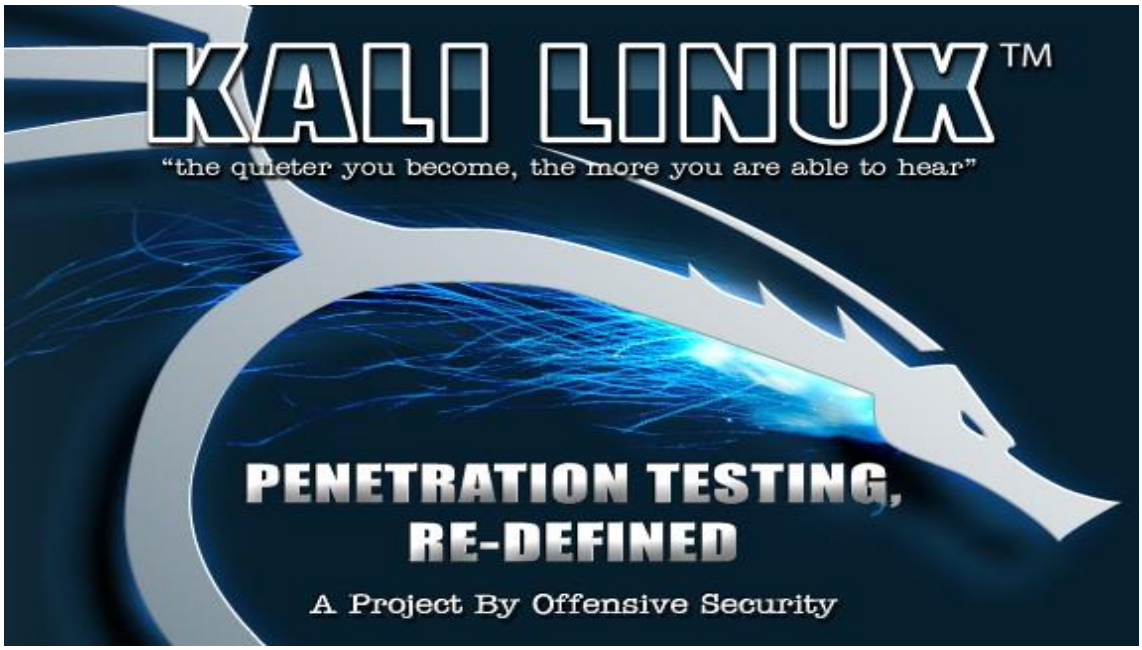

Figura 1. Sistema operativo para realizar auditorías de seguridad.
(Caballero, 2015).

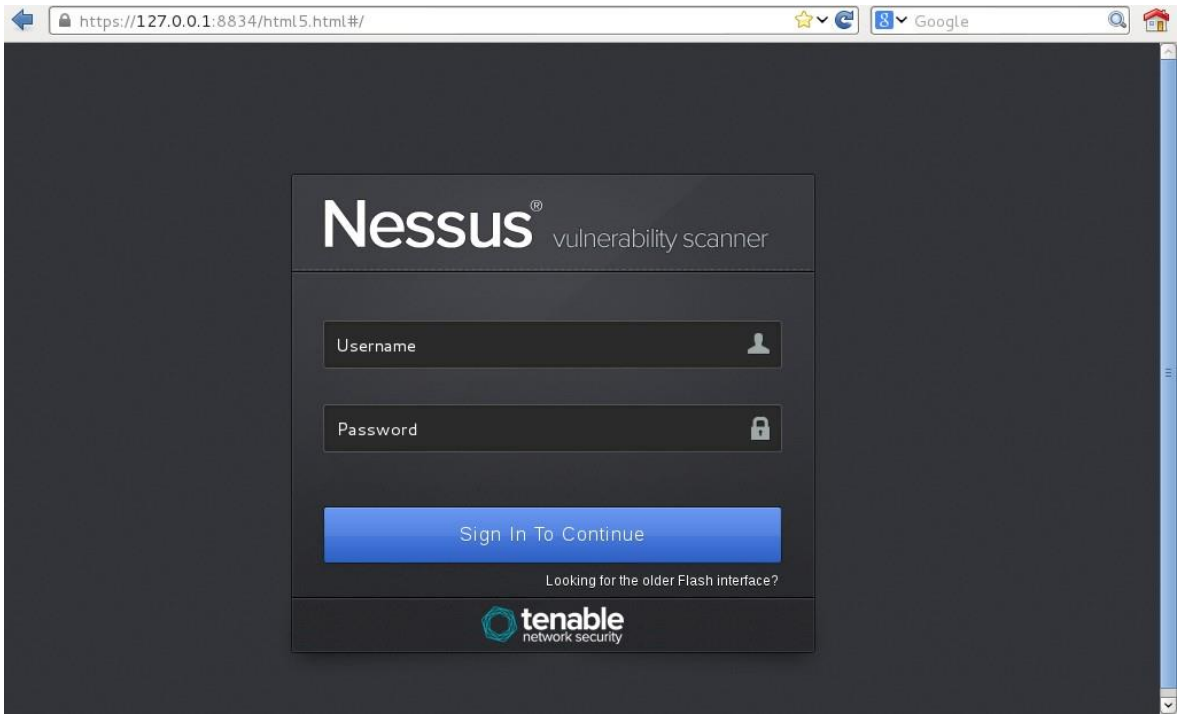

Figura 2. Escaneando vulnerabilidades con Nessus.
(Creadpag, 2018).

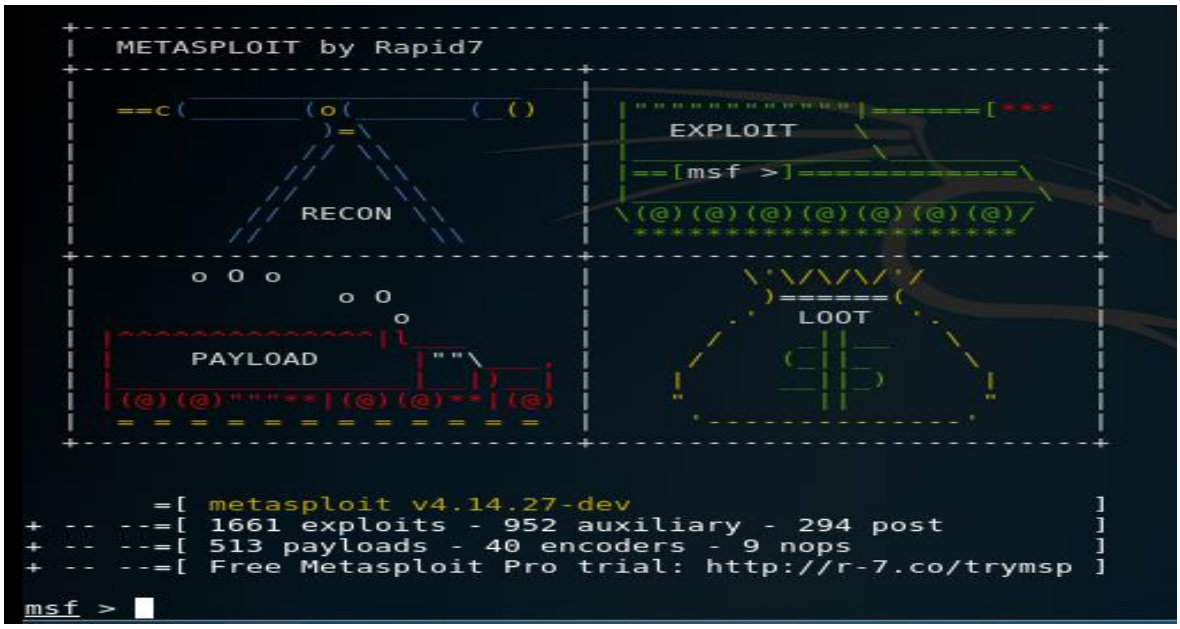

Figura 3. Consola Metasploit donde se crea el código para la explotación de un sistema.
(Mollar, s.f)**.**

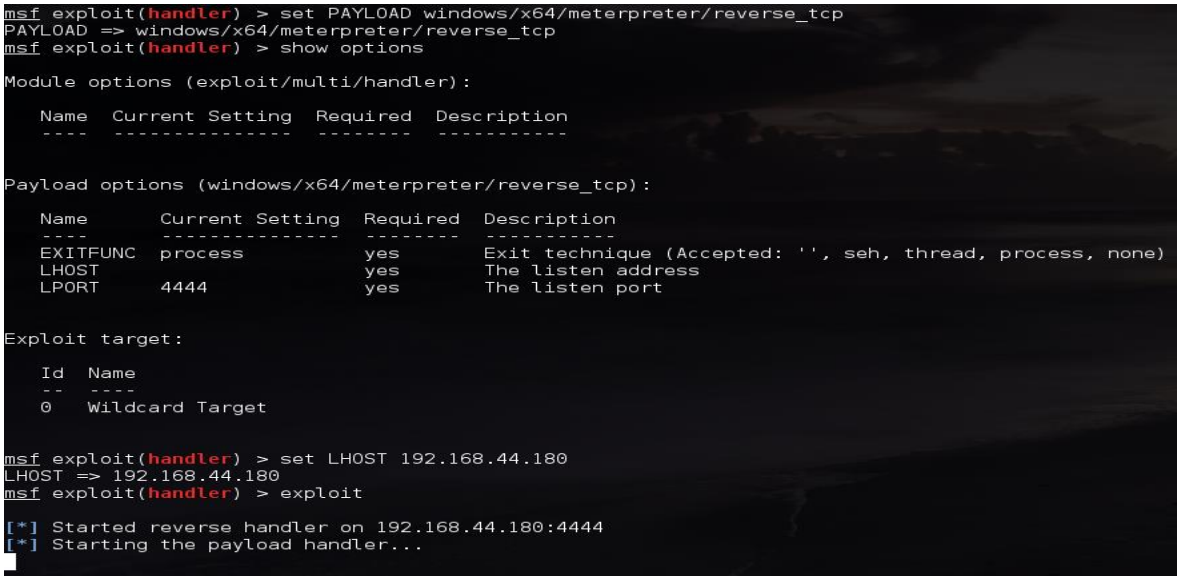

Figura 4. Explotando sistema con el Payload Windows x64.
(Ayllón, 2014).