

Pentesting, ¿Porque es importante para las empresas?

Vanegas Romero, Alfonso Yucenid
Cadcalfonso13@hotmail.com
Universidad Piloto de Colombia

Resumen— Este artículo describe las técnicas, conceptos y metodologías asociadas al pentesting detallando cada una de sus fases. El desarrollo de los contenidos está basado en el proceso que se sigue para poder cumplir los objetivos del pentesting en donde se busca que las empresas entiendan su necesidad basándose en los riesgos que día a día están expuestas, veremos que los atacantes pueden sacar provecho inclusive de la más pequeña vulnerabilidad para hacer una intrusión en los sistemas, además veremos como con algunas herramientas se puede obtener información importante de la empresa con la cual se pueda comenzar un ataque o intrusión y también veremos finalmente como luego de los resultados de las pruebas realizadas las empresas entienden la importancia del pentesting y la necesidad de fortalecer sus políticas y procedimientos internos con el fin de salvaguardar su información.

Abstract— This article describes the techniques, concepts and methodologies associated with pentesting detailing each of its phases. The development of the content is based on the process followed to meet the objectives of the pentesting where it is sought that companies understand their need based on the risks that are exposed every day, we will see that attackers can take advantage even of the smallest vulnerability to make an intrusion in the systems, we will also see how with some tools you can obtain important information about the company with which you can start an attack or intrusion and we will also finally see how after the results of the tests carried out Companies understand the importance of pentesting and the need to strengthen their internal policies and procedures in order to safeguard their information.

Índice de Términos— Amenazas, Pentesting, Pruebas, Riesgo, Seguridad de Información, Vulnerabilidades.

I. INTRODUCCIÓN

La seguridad informática y/o la seguridad de la información va en auge y en todas las empresas u organizaciones que contengan algún sistema informático deben estar conscientes que las amenazas están por donde quiera y que la protección de su información debe ser prioridad para el cumplimiento de los objetivos de valor. Recordemos que la seguridad cuenta con tres pilares fundamentales que son: la confidencialidad, la integridad y la disponibilidad.

Día a día las empresas están sometidas a riesgos que ponen en riesgo los tres pilares antes mencionados, estos riesgos pueden ser externos e inclusive aún más peligrosos, pueden ser internos. Salvaguardar la información debe ser prioridad, pero entonces como lograr esto, como enfrentar los riesgos a los cuales están expuestas las empresas, para ello han surgido varias normas, leyes e inclusive metodologías y prácticas para que las empresas puedan crear sus propias políticas, procesos y procedimientos, pero un buen comienzo es saber que tan expuesto se está pues es bien sabido que no hay seguridad 100 % completa y por lo general no se sabe cuando se pueda llegar a tener uno de estos riesgos, simplemente cada empresa tendrá que estar preparada.

Las empresas deben conocer que impacto económico (aunque no solamente sería económico) generaría tener uno de estos riesgos informáticos tan de la mano, para ello podrían practicar ya sea de manera interna con el departamento de TI o el área encargada, unas serie de pruebas que deben hacerse a toda la infraestructura tecnológica y en general al entorno de TI, también podrían hacerse estas pruebas con empresas externas altamente calificadas y personal experto, con estas pruebas se busca que las empresas hagan una evaluación general y comiencen a priorizar sus objetivos de negocio, dicha evaluación se resume en encontrar las vulnerabilidades de riesgo bajo, alto y otras que puedan llevar a ser aprovechadas por algún atacante, ya sea externo o interno.

A esas pruebas realizadas sobre la infraestructura o sobre el entorno de TI se le conoce como pentesting o pruebas de penetración, donde se busca que las empresas sepan sus fallos de seguridad y las consecuencias que existen, además gracias al pentesting las empresas sabrán como minimizar los riesgos, como priorizarlos y como tener los controles pertinentes.

II. PENTESTING

Primero debemos definir o saber ¿Qué es el pentesting?, Podría definirse como una abreviatura entre las palabras de origen inglés “penetration” y “test” [1]. Pero una definición más completa es que el pentesting es una práctica y/o metodología realizada para descubrir vulnerabilidades y/o fallos de seguridad en un sistema informático, en una página web, en seguridad física o cualquier otro entorno.

Para las empresas es de gran utilidad pues pueden comprobar hasta qué punto su red interna o algún sistema informático es seguro ante algún ataque informático.

El pentesting está diseñado para clasificar y determinar los alcances y las repercusiones de los fallos de seguridad, dando resultados para las empresas en cuanto a poder identificar la información o entornos a los cuales se podrían alcanzar en un ataque, además de evaluar la eficiencia de la defensa con la que cuentan [2].

Con lo anterior debemos hablar de la clasificación del pentesting; estas son divididas en dos, ya sea por el origen de las pruebas o por el conocimiento del objetivo.

A. **ORIGEN DE LAS PRUEBAS**

- Interna
- Externas

Cualquiera de estos dos orígenes puede ser el objetivo final de los pentesting, por ejemplo, podríamos solamente fijar de forma interna un dispositivo de hardware o software y allí concentrar las pruebas, un ejemplo para una prueba externa podría fijarse en alguna página web, tienda virtual, etc.

B. **CONOCIMIENTO DEL OBJETIVO**

- Caja Negra
- Caja Gris
- Caja Blanca

Cuando ya conocemos nuestro objetivo se pueden practicar alguno de estos tres tipos de pruebas con el fin de averiguar vulnerabilidades, fallos, etc.

Con esta clasificación, las empresas pueden determinar las posibilidades de éxito de un ataque, también pueden identificar los fallos de seguridad que son provocados en consecuencia de las vulnerabilidades de menor riesgo y que son explotadas, otras vulnerabilidades de algún programa específico también pueden ser identificadas; finalmente, a las empresas les sirve comprobar la capacidad con la que el encargado de la seguridad puede actuar y enfrentar un ataque [3].

III. TIPOS DE PENTESTING

Los tipos de pentesting que se pueden aplicar en una empresa normalmente pueden ser determinados en base a las condiciones de esta, pues dependerá del tipo de información que se tenga sobre algún sistema al que quieran aplicarlo.

A. **CAJA BLANCA**

Una de las pruebas más completas, parte de un análisis integral y son las más fáciles de practicar, pues la empresa suministra

toda la información posible como, por ejemplo: cantidad de equipos, tipos de sistemas, estructura de la red, servidores, contraseñas, código fuente, documentación, etc., se usa la mayor cantidad de datos posibles para detectar puntos de fallo o vulnerabilidades potenciales en lo que su tiempo de ejecución es mayor en comparación a los otros tipos de pruebas [4].

Sin embargo, esta prueba se centra más en los procesos del sistema, así como del software en sí, ejemplo, sus códigos fuentes y configuraciones. Se puede usar hardware de búsqueda para encontrar errores en códigos fuentes de las aplicaciones o malas configuraciones de software, así como también del mismo hardware.

Algunas ventajas de esta prueba es que son completamente minuciosas como se mencionó anteriormente, el resultado es más preciso ya que detecta amenazas de inmediato, así como defectos de configuración y construcción.

En su contra podríamos decir que al ser más completa requiere de muchos recursos y aunque se tenga toda la información interna o externa de la empresa no se logra simular un ataque o intrusión.

Las pruebas más comunes realizadas están la revisión del código fuente, se hace un examen del diseño y construcción del software y también se hacen auditorias de red.

Algunas de las técnicas usadas son: ataques con conocimiento de la arquitectura del sistema, vulnerabilidades de calidad del código, vulnerabilidades de los protocolos, vulnerabilidades criptográficas y vulnerabilidades en la gestión de contraseñas.

B. **CAJA NEGRA**

Es una prueba a ciegas, pues no se posee la información que se tenía en la anterior prueba de caja blanca, en otras palabras, no se tiene información de los sistemas de la infraestructura, redes, software, contraseñas, etc., esta prueba acerca a las empresas más a la realidad en la que se encuentran para con los atacantes y sirve para reconocer que tan frágiles o que tan fuerte se encuentran [4].

Se simula un ataque o intrusión real sobre los sistemas.

Proporciona una estimación real de amenazas, se obtiene información a través de información pública.

En contra, se podría decir que obtener la información tendría mucho esfuerzo, se pueden pasar por desapercibidas puertas traseras o vulnerabilidades parciales y se brindarían recomendaciones muy generales.

Las pruebas más comunes realizadas son, pruebas de penetración de infraestructura o de red, pruebas de penetración a aplicaciones y un ataque simulado completo.

Alguna de las técnicas usadas, vulnerabilidades de tipo deface, vulnerabilidades de tipo cross-site scripting, vulnerabilidades inyección de SQL, vulnerabilidades de desbordamiento de

memoria, vulnerabilidades basadas en secuestro de sesión, autenticación incompleta, entre otras.

C. CAJA GRIS

Es una mezcla de los dos anteriores pruebas, caja blanca y caja gris, consta en tener cierta información para realizar estas pruebas. Se usa comúnmente cuando se quiere identificar vulnerabilidades en sectores determinados, un ejemplo, podría ser suministrar la contraseña de administrador al tester y este podría determinar a qué información se puede acceder de forma superior [5].

Es más rentable y nos proporciona una estimación más real de amenazas.

No simula un ataque en tiempo real.

Las pruebas realizadas estaría la aplicación de pruebas de penetración (código parcial), test de infraestructura y de red.

IV. ETAPAS DEL PENTESTING

A. Descubrimiento y Enumeración.

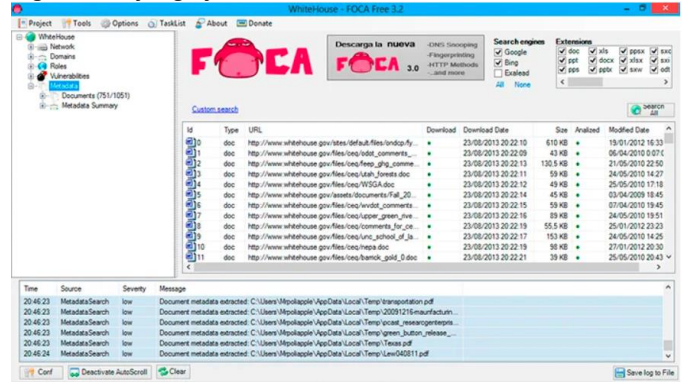
Es una de las etapas más importantes en un pentesting ya que es donde recopilamos toda la información necesaria sobre el objetivo, trataremos de saber que equipos de escritorio se tienen, dispositivos de red, servidores web, servidores de aplicación, impresoras, entre otros dispositivos en los cuales se podrían encontrar vulnerabilidades, también se puede recolectar otra información, por ejemplo, los subdominios de una página web para poder realizar ataques de tipo diccionario, en otras palabras podemos identificar la información que esta públicamente alojada en internet.

Para lo dicho anteriormente la información se puede recopilar de forma activa o de forma pasiva, una recopilación de forma pasiva es donde no se requiere que el atacante se comunique con el objetivo directamente y donde este se encuentra recopilando toda la información posible de internet y en la recopilación de forma activa, el atacante siempre estará en contacto con el objetivo tratando de reunir la información.

Entre alguna de las herramientas para la recolección pasiva podemos encontrar las siguientes:

- FOCA: “Es una herramienta utilizada principalmente para encontrar metadatos e información oculta en los documentos que examina. Estos documentos pueden estar en páginas web. Los documentos que es capaz de analizar son muy variados, siendo los más comunes los archivos de Microsoft Office, Open Office, o ficheros PDF, aunque también analiza ficheros de Adobe InDesign, o svg, por ejemplo. Estos documentos se buscan utilizando tres posibles buscadores que son Google, Bing y DuckDuckGo [6]”

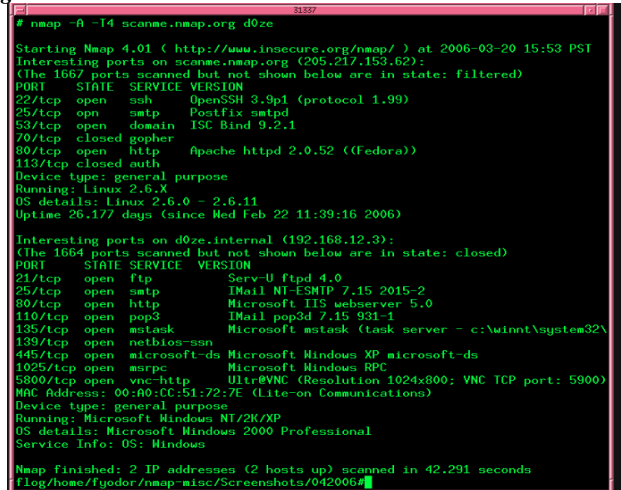
Fig. 1: Interfaz gráfica FOCA



Fuente: <https://rootear.com/seguridad/foca-metadatos-archivos>

- NMAP: “Es una aplicación multiplataforma usada para explorar redes y obtener información acerca de los servicios, sistemas operativos y vulnerabilidades derivadas de la conjunción de éstos [7].”

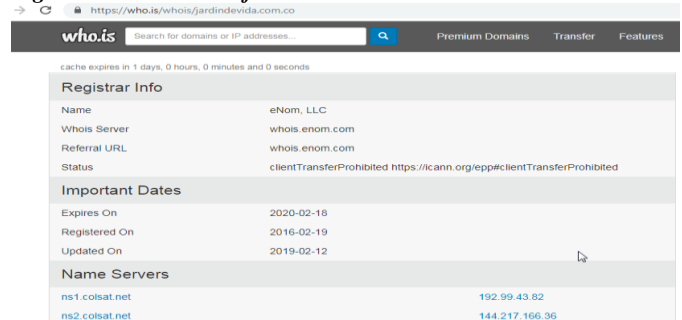
Fig. 2. Comandos NMAP



Fuente: <https://nmap.org/images/>

- Existen herramientas en internet para recopilar información como por ejemplo <https://who.is>, que nos sirve para ver la información acerca del dominio de la empresa la cual está siendo objetivo del pentest, podemos encontrar datos relevantes como direcciones de correo, dns, servidores web, entre otros.

Fig. 3. Descubriendo información en who.is



Fuente: Autor desde <https://who.is/>

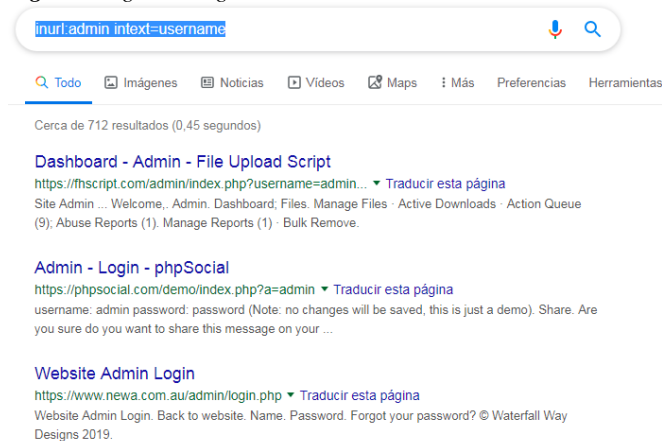
Otras de las herramientas de recolección pasiva es Google Hacking, es una técnica usada para recopilar información a través del buscador Google usando operadores lógicos y operadores avanzados.

Ejemplos de uso:

- Para construir una consulta se usa los operadores and, or y not, también se puede usar el símbolo “pipe” | para simplificar una búsqueda, ejemplo: username | userid | user
- Para consultas más avanzadas se usan operadores avanzados como: intitle, inurl, filetype, Site y Link, por ejemplo:

inurl:admin intext=username

Fig. 4: Google Hacking



Fuente: Autor desde google.com.co.

En la figura 4 vimos que el resultado arrojado por la consulta avanzada “inurl:admin intext=username” son una serie de sitios de login con username, así mismo podríamos buscar archivos de Log u otras páginas completas de login, así como archivos de configuración de servidores web y también archivos de configuración de bases de datos, entre otros.

Como el objetivo de esta etapa es recopilar la mayor cantidad de información posible de una empresa bien sea de forma externa o interna, es importante ayudarle a las empresas a ocultar cualquier información de sus tecnologías usadas, por ejemplo, si la empresa usa un servidor web basado en Windows para determinados servicios, estos fácilmente serán expuestos en internet y para un atacante será de gran ayuda obtener esta información, por lo que al realizar una ofuscación interna sería de gran ayuda, en nuestro ejemplo, este servidor web basado en Windows podría mostrarse como un servidor Web basado en Unix o en otras tecnologías que hagan que el atacante se demore o tome otros caminos para cumplir su objetivo y por qué no, se retracte de hacer alguna intrusión.

B. ANÁLISIS DE VULNERABILIDADES

En esta etapa se usa la información recopilada anteriormente y de acuerdo a los resultados obtenidos de buscar vulnerabilidades o brechas de seguridad.

El análisis de vulnerabilidades es el proceso de descubrir fallas en sistemas y aplicaciones que pueden ser aprovechadas por los atacantes. Estas fallas pueden variar desde la configuración incorrecta del host y el servicio o el diseño inseguro de la aplicación. El proceso usado para buscarlas fallas puede variar y depende del objetivo.

Se usan diferentes herramientas, por ejemplo, pueden usarse escáners de vulnerabilidades de red y escáners de vulnerabilidades web.

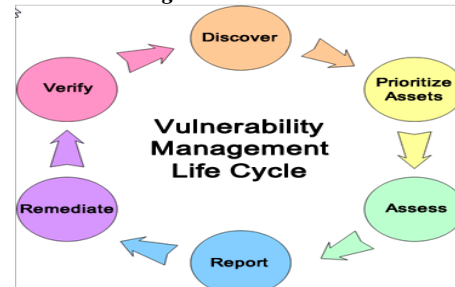
Dentro de los escáners de vulnerabilidades de red encontramos un escaneador de puertos donde se destaca la herramienta anteriormente mencionada NMAP, recordemos que esta herramienta nos ayuda a ver que puertos o servicios se encuentran abiertos en una red, el pentest debe efectuar el análisis en otras herramientas para así determinar las vulnerabilidades en general. Por otro lado, existen otros escaneadores como Nessus, OpenVas y Nexpose en donde este último por ejemplo nos busca vulnerabilidades dentro de una base de datos.

Ampliando lo mencionado anteriormente en cuanto al escaneo de puertos y servicios, debemos mencionar que el escaneo de puertos busca o involucra los protocolos usados por IP (TCP, UDP, ICMP, etc.) y que estos pueden tener dos estados, abierto o cerrado. Para el escaneo de servicios se hace más específico pues por ejemplo se puede buscar si en la red a escanear se tiene el servicio FTP o SSH habilitado para así detectar o buscar las más recientes vulnerabilidades.

Para los escaneos de aplicaciones web se usan herramientas como acunetix, esta herramienta es capaz de buscar fallos de seguridad a través de los protocolos http y https, también cuenta con herramientas internas para realizar otro tipo de pruebas independientes.

Debemos tener en cuenta el ciclo de gestión de las vulnerabilidades, que no es más que un paso a paso para cuando vamos a hacer una evaluación completa en la empresa en la cual priorizaremos las amenazas.

Fig. 5: Ciclo de vida de la gestión de vulnerabilidades.



Fuente: <https://www.cdc.gov/cancer/npcr/tools/security/vmlc.htm>

- **Discover:** Se debe inventariar todos los activos de la red e identificar detalles del host, como el sistema operativo y algunos servicios abiertos para identificar vulnerabilidades. Desarrollar una línea base de la Red. Identificar las vulnerabilidades de seguridad en un horario habitual [8].
- **Prioritize Assets:** Se deben categorizar los activos en grupos o unidades de negocio y asignar un valor comercial a estos grupos de activos en función de su importancia crítica en el negocio [8].
- **Assess:** Determinar una línea base de perfil de riesgo para poder eliminar los riesgos en función de la criticidad de los activos, la amenaza de la vulnerabilidad y la clasificación de los activos [8].
- **Report:** Medir el nivel de riesgo del negocio asociado de acuerdo con sus políticas de seguridad, monitorear y describir vulnerabilidades conocidas [8].
- **Remediate:** Priorizar y corregir las vulnerabilidades de acuerdo con el riesgo del negocio. Establecer controles y demostrar un progreso [8].
- **Verify:** Verificar que las amenazas se hayan eliminado mediante auditorias de seguimiento [8].

Entendiendo un poco este ciclo de vida de la gestión de vulnerabilidades debemos decir que es un modelo útil para comprender como las vulnerabilidades en los sistemas y aplicaciones se convierten en los puntos de entrada para los atacantes, cuando sus riesgos son mayores y como defenderse de la manera más adecuada [9]. Dentro del análisis que hacemos debemos saber que existen unas calificaciones o métricas y un diccionario público de vulnerabilidades, los cuales describiré a continuación:

Comenzaré con el diccionario público de vulnerabilidades o CVE(Common Vulnerabilities and Exposures), que no es más que una serie de listas de entradas, cada una con un número de identificación, una descripción, y al menos una referencia publica para las vulnerabilidades de seguridad públicamente conocidas [10].

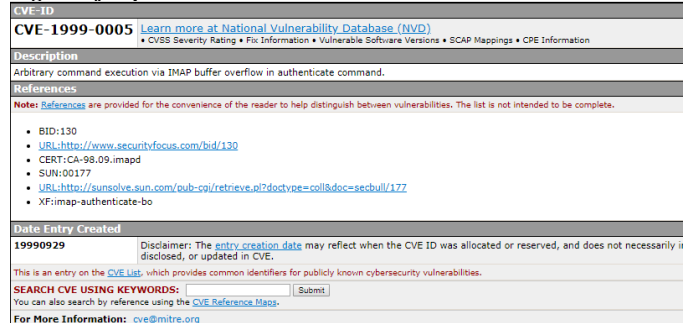
Recordando algo de la historia de CVE, se lanzó en 1999 cuando la mayoría de las herramientas de ciberseguridad utilizaban sus propias bases de datos con sus propios nombres para vulnerabilidades de seguridad En ese momento, había una variación significativa entre los productos y no era una forma fácil de determinar cuándo las diferentes bases de datos se referían al mismo problema. Las consecuencias fueron brechas potenciales en la cobertura de seguridad y no una interoperabilidad efectiva entre las diferentes bases de datos y herramientas. Además, cada proveedor de herramientas usó diferentes métricas para establecer el número de vulnerabilidades o exposiciones que detectaron, lo que significaba que no había una base estandarizada para la evaluación entre las herramientas [10].

CVE es ahora el estándar de la industria para los identificadores de vulnerabilidad y exposición. Las entradas de CVE, también llamadas "CVE", "ID de CVE" y "números de CVE" de la comunidad, proporcionan puntos de referencia para el

intercambio de datos para que los productos y servicios de ciberseguridad puedan comunicarse entre sí. CVE también proporciona una línea de base para evaluar la cobertura de herramientas y servicios para que los usuarios puedan determinar qué herramientas son las más efectivas y adecuadas para las necesidades de su organización. En resumen, los productos y servicios compatibles con CVE ofrecen una mejor cobertura, una interoperabilidad más sencilla y una mayor seguridad [10].

Veamos un ejemplo de diccionario de CVE:

Fig. 6: Ejemplo CVE



Fuente: <http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-1999-0005>

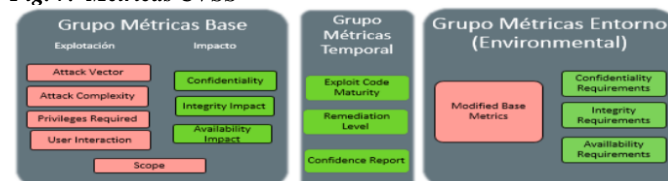
Para la calificación o métricas de las vulnerabilidades existe el CVSS (Common Vulnerability Scoring System), este es un sistema de puntuación que proporciona una manera de capturar las características principales de una vulnerabilidad y representarlo en un valor numérico que refleje su gravedad, la puntuación numérica se puede representar en valores cualitativos (como baja, media, alta y crítica) esto con el fin de ayudar a las empresas a evaluar y priorizar adecuadamente los procesos de gestión de las vulnerabilidades [11].

CVSS se compone de tres grupos principales de métricas, Base, Temporal y Entorno:

- **Grupo Base**, engloba las cualidades intrínsecas de una vulnerabilidad y que son independientes del tiempo y el entorno.
- **Grupo Temporal**, Características de la vulnerabilidad que cambian en el tiempo.
- **Grupo de Entorno**, Las características de la vulnerabilidad relacionadas con el entorno del usuario.

Las métricas evaluadas las podemos ver en la siguiente figura:

Fig. 7. Métricas CVSS



Fuente: <https://www.incibe-cert.es/blog/cvss3-0>

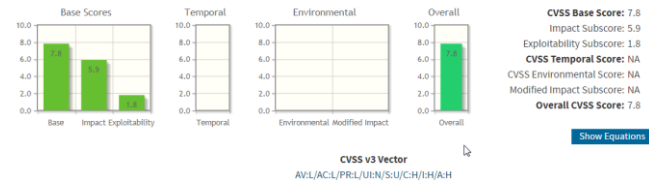
Luego de asignar los valores de cada métrica se aplicarán unas fórmulas recogidas en las especificaciones del CVSS y que resultarán en un valor numérico entre 0.0 y 10.0 para cada grupo. Este resultado numérico total puntúa y determina cuantitativamente el impacto final de una vulnerabilidad [11].

Actualmente la puntuación esta automatizada con una herramienta que implementa las fórmulas, esta hace parte del NIST y la podemos encontrar en el siguiente link:

<https://nvd.nist.gov/vuln-metrics/cvss/v2-calculator?calculator&adv&version=2%20>

y en la siguiente figura podemos ver un ejemplo de la puntuación obtenida para una vulnerabilidad usando la anterior herramienta del NIST donde previamente se evaluaron las métricas para cada grupo.

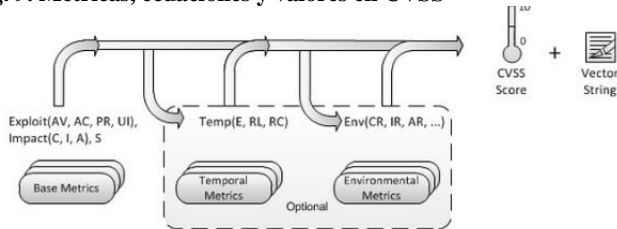
Fig. 8. Score Vulnerabilidad



Fuente: <https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?name=CVE-2016-0051>

La evaluación de la vulnerabilidad además del valor numérico CVSS se acompaña con una cadena de texto denominada “vector string” que contiene la representación textual de cada valor asignado a cada métrica contemplada.

Fig. 9. Métricas, ecuaciones y valores en CVSS



Grupo de Métricas	Métrica	Valores Posibles	Obligatorio
Base	Attack Vector, AV	[N,A,L,P]	Si
	Attack Complexity, AC	[L,H]	Si
	Privileges Required, PR	[N,L,H]	Si
	User Interaction, UI	[N,R]	Si
	Scope, S	[C,U]	Si
	Confidentiality, C	[H,L,N]	Si
Temporal	Integrity, I	[H,L,N]	Si
	Availability, A	[H,L,N]	Si
	Exploit Code Maturity, E	[X,H,F,P,U]	No
Environmental	Remediation Level, RL	[X,U,W,T,O]	No
	Report Confidence, RC	[X,C,R,U]	No
	Confidentiality Req., CR	[X,H,M,L]	No
	Integrity Req., IR	[X,H,M,L]	No
	Availability Req., AR	[X,H,M,L]	No
	Modified Attack Vector, MAV	[X,N,A,L,P]	No
	Modified Attack Complexity, MAC	[X,L,H]	No
	Modified Privileges Required, MPR	[X,N,L,H]	No
	Modified User Interaction, MUI	[X,N,R]	No
	Modified Scope, MS	[X,U,C]	No
	Modified Confidentiality, MC	[X,N,L,H]	No
	Modified Integrity, MI	[X,N,L,H]	No
Modified Availability, MA	[X,N,L,H]	No	

Fuente: <https://www.incibe-cert.es/blog/cvss3-0>

La siguiente figura nos muestra los valores cualitativos:

Fig. 10. Valores cualitativos

Puntuación	Severidad
0	Nula
0.1-3.9	Baja
4.0-6.9	Media
7.0-8.9	Alta
9.0-10.0	Crítica

Fuente: <https://www.incibe-cert.es/blog/cvss3-0>

C. EXPLOTACIÓN

Esta es la etapa donde se realiza por parte del pentest la explotación de alguna de las vulnerabilidades encontradas en el punto anterior, aquí se saca provecho de la vulnerabilidad para intentar comprometer el sistema o aplicaciones.

Pero ¿cómo se ejecuta un ataque para explotar una vulnerabilidad?, la respuesta es que esto se logra con un programa o herramienta llamada Exploit.

¿Qué es un exploit?

Veamos una definición de exploit, según welivesecurity de la empresa eset, “Es un programa o código que se aprovecha de un agujero de seguridad (vulnerabilidad) en una aplicación o sistema, de forma que un atacante podría usarla en su beneficio [12]”

Existen 3 tipos de exploits:

- **Exploit remotos**, actúa sin necesidad de estar físicamente en el equipo objetivo, puede realizarse desde la propia red local pero también se puede ejecutar el ataque con el solo hecho de que el objetivo tenga acceso a internet.
- **Exploit local**, actúa estando físicamente en el equipo a atacar, puede llegar a este de forma remota.
- **Exploit cliente**, es el más común ya que requiere alguna acción del usuario para ejecutarse, por ejemplo, un enlace enviado desde un correo o un archivo a descargar.

Con lo anterior, los exploits usan entornos de trabajo llamados frameworks que son los que permiten finalmente la ejecución del código interno de los exploits, uno de estos frameworks más conocidos es Metasploit, del cual hablaré un poco a continuación, pues como lo mencionaba, es por muchos pentesters la herramienta más usada en esta fase de explotación.

METASPLOIT

Metasploit es una plataforma de pruebas de penetración que le permite encontrar, explotar y validar vulnerabilidades. La plataforma incluye Metasploit Framework y sus contrapartes comerciales, como Metasploit Pro.

Metasploit Framework

Metasploit Framework es la base sobre la cual se construyen los productos comerciales. Es un proyecto de código abierto que proporciona la infraestructura, el contenido y las herramientas

para realizar pruebas de penetración y una extensa auditoría de seguridad. Gracias a la comunidad de código abierto y al propio equipo de contenido de Rapid7, se agregan nuevos módulos de forma regular, lo que significa que el último exploit está disponible para usted tan pronto como se publique [13].

Comandos Básicos:

- **Help:** Con este comando podremos desplegar la lista de comandos disponibles en la consola de Metasploit, también podemos especificar el comando seguido de -h, o poner help seguido del comando, para obtener información detallada acerca del comando específico [14].
- **search:** search, seguido de una característica, nos muestra los módulos que contienen dicha característica. Un pequeño truco para comprobar si está actualizado sería buscar una característica reciente para ver si disponemos de ella o no [14].
- **info:** Muestra detalles del módulo (opciones, objetivos y descripción). Si estamos usando ese módulo (hemos usado -use sobre él), nos basta con poner el comando, de lo contrario debemos especificar la ruta del módulo [14].
- **show:** Muestra en pantalla las opciones del módulo. Puede ir seguido de actions, advanced, all, auxiliary, en función de las opciones específicas que queramos ver [14].
- **use:** Selecciona el módulo que especifiquemos. El comando back sirve para quitar la selección [14].
- **set:** configurar parámetros de un módulo en concreto. Unset sería para borrar la configuración [14].
- **connect:** Sirve para conectarnos a otras máquinas especificando la ip + el puerto, de este modo podríamos disponer de módulos y configuraciones en otra máquina y acceder desde máquinas remotas. Esta utilidad se vale del famoso netcat, así que disponemos de opciones parecidas a las de netcat [14].
- **exploit:** Lanzar el módulo. Disponemos de las siguientes opciones:
 - j: Lo lanza en segundo plano (background).
 - z: Para que tras una explotación exitosa no se interactúe con la sesión.
 - e: Se lanza el payload con una codificación realizada previamente con un payload.

En esta etapa las empresas pueden darle mas importancia al pentesting, aquí como se mencionó anteriormente el pentest explota una de las vulnerabilidades encontradas en la anterior etapa, supongamos que el pentest exploto una vulnerabilidad y logro una intrusión en la base de datos principal, al estar dentro se pueden realizar muchas maniobras como, por ejemplo, cambiar el password de los usuarios principales, sacar información de ventas, eliminar datos, entre otras acciones. Esto para una empresa generaría un gran impacto negativo no solo por la indisponibilidad de los servicios sino también

porque la información más importante se vio comprometida y también se pueden tener impactos económicos. A demás de esto, las empresas sabrán que no estaban completamente seguras, que tenían debilidades en su infraestructura tecnológica y cuáles son sus activos de mayor riesgo.

D. INFORME

En esta fase se hace toda la documentación para la presentación de un informe con los resultados obtenidos durante las diferentes fases ejecutadas, su finalidad es darle una visibilidad completa en donde se detallan los riesgos de todas las vulnerabilidades encontradas, se indica en que puntos la seguridad hasta ahora implementada esta correcta y también en que puntos la seguridad es deficiente y deban ser corregidos, se debe entregar un informe técnico, que estaría dirigido al personal TI y un informe gerencial el cual sería más específico con graficas estadísticas por ejemplo, numero de las vulnerabilidades encontradas, números de equipos comprometidos, facilidad de explotación y con un lenguaje no tan técnico con el fin de ser entendido destacando todo a nivel del riesgo y consecuencias para la organización.

V. NORMAS Y LEYES

Existen una serie de normativas para la realización de pentesting, cada empresa debería seguir alguna de estas según la reglamentación por país o según la metodología usada para el sistema de gestión de seguridad de la información o según su actividad.

Según ISO 27001, para realizar un pentesting conviene implementar un ISMS (Information Security Management System) para beneficio de las pruebas en tres formas [15]:

1. Como parte del proceso de evaluación de riesgos, el pentesting identificará vulnerabilidades en aplicaciones web, dispositivos internos, direcciones IP expuestas a internet y los relacionará con las amenazas identificables.
2. Como parte del proceso de gestión de riesgos, el pentesting garantizará que todo funcione como debe ser.
3. Como parte del proceso de mejora continua, este tipo de pruebas de fortaleza asegura que los controles siguen siendo válidos y continúan proporcionando información sobre nuevas vulnerabilidades [15].

Con el auge de las transacciones o pagos en línea con tarjetas débito o crédito se creó por parte de algunas empresas como American Express, Visa, MasterCard, entre otras, el PCI DSS (Payment Card Industry Security Standards Council) un estándar de seguridad y orientado a la definición de controles para la protección de los datos del titular y/o datos confidenciales de autenticación durante su procesamiento, almacenamiento y/o transmisión [16]. Para esta normativa textualmente en cuanto al pentesting nos indica “Componentes de sistema, procesos y software personalizado deben ser

puestos a prueba frecuentemente para asegurar que los controles siguen reflejando un panorama evolutivo”, es decir, se debería aplicar un pentesting cuando una o varias de las siguientes situaciones ocurra:

- Se han aplicado parches de seguridad.
- Se ha modificado de forma considerable la infraestructura o red.
- Se ha añadido una aplicación web o elemento de infraestructura nuevos.
- Se ha cambiado de ubicación nuestra oficina o se ha añadido una sede a la red corporativa [15].

Para Colombia, por ejemplo, existe para las entidades financieras la circular 042 de 2012, en donde se informa a gerentes, miembros de juntas directivas, representantes legales, entre otros, que se establecen una serie de medidas encaminadas a fortalecer la seguridad y la calidad en el manejo de la información de los clientes y usuarios de las entidades vigiladas por la Superintendencia Financiera de Colombia [17].

A razón de la actividad del negocio de las empresas se torna más importante seguir las normativas o leyes vigentes con el fin de no incurrir en multas por incumplimiento, sin embargo, estas multas no deberían ser el punto de partida para que las empresas realicen un pentesting, sino por el contrario, optar por realizar un pentesting ya sea con personal interno o con personal externo a la infraestructura tecnológica periódicamente ayudara a mantener la información segura.

VI. METODOLOGIAS

Teniendo en cuenta el auge de la seguridad informática y/o de la seguridad de la información algunas empresas han elaborado estándares y metodologías para el proceso del pentesting, todo esto con el fin de seguir buenas prácticas y buenos estándares de calidad dando cubrimiento a los principios básicos de la seguridad que son confidencialidad, integridad y disponibilidad, algunas de estas metodologías son:

A. OSSTMM (Open-Source Security Testing Methodology Manual).

Metodología que propone un proceso de evaluación de una serie de áreas que refleja de manera fiel los niveles de seguridad presentes en la infraestructura que va a ser auditada, a estos niveles de seguridad se le denominan comúnmente “Dimensiones de Seguridad” y normalmente consiste en analizar los siguientes factores [19]:

- Visibilidad
- Acceso
- Confianza
- Autenticación
- Confidencialidad
- Privacidad
- Autorización
- Integridad
- Seguridad

- Alarma

Como parte de un trabajo secuencial la metodología OSSTMM consta de 6 ítems los cuales comprenden todo sistema actual, estos son:

- 1) *Seguridad de la Información*
- 2) *Seguridad de los Procesos*
- 3) *Seguridad en las tecnologías de Internet*
- 4) *Seguridad en las comunicaciones*
- 5) *Seguridad inalámbrica*
- 6) *Seguridad Física*

B. ISSAF (Information Systems Security Assessment Framework).

Marco metodológico de trabajo desarrollado por la OISSG que permite clasificar la información de la evaluación de seguridad en diversos dominios usando diferentes criterios de prueba. Algunas de las características más representativas de ISSAF son:

Brinda medidas que permiten reflejar las condiciones de escenarios reales para las evaluaciones de seguridad.

Esta metodología se encuentra principalmente enfocada en cubrir los procesos de seguridad y la evaluación de estos para así obtener un panorama completo de las vulnerabilidades existentes. Permite el desarrollo de matriz de riesgo para verificar la efectividad en la implementación de controles [18].

C. OWASP (Open Web Application Security Project).

Metodología de pruebas enfocada en la seguridad de aplicaciones, El marco de trabajo descrito en este documento pretende alentar a las personas a evaluar y tomar una medida de la seguridad a través de todo el proceso de desarrollo. Así, pueden relacionar los costes de un software inseguro al impacto que tiene en su negocio, y de este modo gestionar decisiones de negocio apropiadas (recursos) para la gestión del riesgo, algunas de las características más representativas de OWASP son [18]:

- Pruebas de firma digital de aplicaciones Web.
- Comprobaciones del sistema de autenticación.
- Pruebas de Cross Site Scripting.
- Inyección XML
- Inyección SOAP
- HTTP Smuggling
- Sql Injection
- LDAP Injection
- Polución de Parámetros
- Cookie Hijacking
- Cross Site Request Forgery

VII. CONCLUSIONES

Las empresas tienen como medio y a su disposición el pentesting, donde normalmente este es considerado como detección de vulnerabilidades, sin embargo, este pentesting puede ser un aliado importante para las empresas ya que todas deben cumplir con regulaciones dadas por cada país y están sujetas al estricto cumplimiento. El pentesting puede ser realizado por equipos internos ad-hoc, que periódicamente evalúan la resistencia de los sistemas y también puede ser contratado con un tercero, que como vimos durante este artículo no solamente busca conocer el estado de resistencia de los sistemas, sino que también busca información relevante para poder hacer intrusiones.

El pentesting está dado por ciertas regulaciones de la industria a la cual pertenece la empresa (sector salud, sector financiero, sector educación, entre otras), por ejemplo, PCI-DSS exigen pentesting anuales y pentesting periódicos después de cualquier cambio en el sistema. SOX e HIPPA indican pentesting anuales por un tercero, inclusive si la empresa ya cuenta con un sistema de gestión para la seguridad de la información, dentro de este y en el marco de la norma ISO 27001 se recomienda hacer pentesting periódicos a la infraestructura crítica y a las aplicaciones, todo con el fin de ayudar a encontrar o descubrir vulnerabilidades y probar los controles de seguridad. Con todo esto las empresas evitan arriesgarse a ser multadas por no cumplir las regulaciones y hacer pentesting ayudan a que se cumpla.

El pentesting ayuda a las empresas a comprender cuales son los puntos débiles de la infraestructura y del entorno de TI, así mismo a identificar también que soluciones deben aplicarse, en otras palabras, las empresas pueden entender la necesidad de fortalecer sus defensas, pero esto no se logra con una simple evaluación de vulnerabilidades hecha con alguna de las herramientas usadas, sino que el pentesting agrega el ingenio humano a las medidas de descubrimiento y hace que sea invaluable evaluar las necesidades de las empresas con respecto a la seguridad.

El pentesting ayuda a las empresas a evaluar si se están siguiendo las políticas y procedimientos por parte de sus empleados, dando una medida para reforzar la capacitación de estos en seguridad de TI. Dentro del pentesting, los pentest emplean técnicas de ingeniería social para así explotar las debilidades humanas, por ejemplo, recreando escenarios para que el empleado acceda a enlaces que aparentan ser seguros y este caiga en un intento de phishing o descarga de documentos seguros que en realidad es descarga de software malicioso. Ver como los empleados responden a estas situaciones de amenaza en donde ellos creen que están seguros hace que las empresas estimulen el cumplimiento de los programas de concientización y hacen que estos programas estén adaptados a cada una de las necesidades.

El pentesting ayuda también a entender las prioridades para las empresas; una vez se simula una intrusión y se identifican las posibles consecuencias, es necesario dar un paso adelante.

Además, el pentesting ayuda a evaluar también los peores escenarios y cuáles son los activos de mayor riesgo, pero sobre todo, ayuda a que las empresas se centren en esas áreas que requieran mayores esfuerzos para la protección de los datos y sin duda esto puede ser útil para tener una visión realista de un incidente que aunque sea simulado pueda ser evaluado para que las empresas sepan que tanto están dispuestas a asumir el riesgo y así mismo que tanto pueden soportar consecuencias catastróficas.

Detallando los riesgos y al explorar los impactos que una posible intrusión podría tener en una empresa podemos deducir con el pentesting que ayuda a verificar si los tiempos de respuesta del personal que se encuentra disponible es el adecuado, adicional se valida si el tiempo promedio para la restauración de los sistemas es el óptimo y también se evidencia que reacciones tienen los empleados con respecto a una amenaza, pero sin duda el más importante para las empresas es saber si están siendo efectivos los procedimientos y si están listos para aplicarlos.

REFERENCIAS

- [1] Esaú A, “Que es el pentesting”, Oct 2018, [Online]. Available: <https://openwebinars.net/blog/que-es-el-pentesting/>.
- [2] Panda Security, “Pentesting: una herramienta muy valiosa para tu empresa”, Feb 2018, [Online]. Available: <https://www.pandasecurity.com/spain/mediacenter/seguridad/pentesting-herramienta-empresa/>
- [3] “Pentesting” [Online]. Available: <https://cyberseguridad.net/index.php/pentesting>
- [4] “Pentest: ¿qué es y cuáles son los principales tipos?” [Online]. Available: <https://ostec.blog/es/seguridad-perimetral/pentest-concepto-tipos>
- [5] Esaú A, “Tutorial hacking: Razones para realizar un Pentesting a nuestra empresa”, Sep 2015, [Online]. Available: <https://openwebinars.net/blog/Tutorial-hacking-razones-para-realizar-un-pentesting-a-nuestra-empresa/>
- [6] “Foca” [Online]. Available: <https://www.elevenpaths.com/es/labtools/foca-2/index.html>
- [7] “¿Qué es Nmap?”, Jun 2017 [Online]. Available: <https://seguinfo.wordpress.com/2007/06/27/%C2%BFque-es-nmap/>
- [8] “Vulnerability Management Life Cycle”, Oct 2018, [Online]. Available: <https://www.cdc.gov/cancer/npcr/tools/security/vmlc.htm>
- [9] “Defense throughout the vulnerability life cycle” [Online]. Available: <https://www.alertlogic.com/assets/whitepapers/Defense-Throughout-the-Vulnerability-Life-Cycle-3.pdf>
- [10] “CVE, Common Vulnerabilities and Exposures” [Online]. Available: <http://cve.mitre.org/>
- [11] “CVSS, Common Vulnerability Scoring System SIG” [Online]. Available: <https://www.first.org/cvss/>
- [12] “¿Sabes qué es un exploit y cómo funciona?”, Oct 2014 [Online]. Available: <https://www.welivesecurity.com/la-es/2014/10/09/exploits-que-son-como-funcionan/>

- [13] “Metasploit”, Oct 2014, [Online]. Available: <https://metasploit.help.rapid7.com/docs>
- [14] “Comandos básicos Metasploit”, [Online]. Available: <https://lifkablog.wordpress.com/2014/07/26/comandos-basicos-de-metasploit/>
- [15] “¿Cada cuánto tiempo debería realizar un test de penetración?”, Ene 2017, [Online]. Available: <https://protegermipc.net/2017/01/31/cada-cuanto-tiempo-deberia-realizar-un-test-de-penetracion/>
- [16] “¿Qué es PCI DSS?”, May 2019, [Online]. Available: <https://www.pcihispano.com/que-es-pci-dss/>
- [17] “Circular 042 de 2012”, [Online]. Available: <https://contadormmc.files.wordpress.com/2016/08/sintesis-circular-ext-042-de-2012-superfinanciera.pdf>
- [18] “Metodologías y Herramientas de Ethical Hacking”, Feb 2013, [Online]. Available: <https://seguridadinformaticahoy.blogspot.com/2013/02/metodologias-y-herramientas-de-ethical.html>

Autor. Vanegas Romero Alfonso Yucenid, Nació en la ciudad de Bogotá, el 06 de noviembre de 1986, se graduó de ingeniero de sistemas en el año de 2013 de la universidad Los Libertadores, cuenta con experiencia en áreas de sistemas de la información por más de 8 años y actualmente cursa una