Análisis de vulnerabilidades técnicas

https://cdn.pixabay.com/photo/2016/04/04/14/12/monitor-1307227_960_720.jpg

Los *frameworks* son marcos de referencia que permiten tener un esquema conceptual que simplifica el desarrollo de una actividad o tarea. Para el análisis de vulnerabilidades existen varios *frameworks* de referencia que son importantes de conocer; a continuación, se detallan:

- **OWASP Vulnerability Management Guide - OVMG**

  Consiste en una guía de mejores prácticas que puede ser utilizada para realizar una adecuada gestión de vulnerabilidades, proporcionando un esquema de ciclo de vida de gestión de la vulnerabilidad, el cual se estructura en tres ciclos *1. Detección, 2. Reporte, 3. Remediación.*

  La guía se encuentra en idioma inglés, pero puede ser fácilmente traducida. La misma puede ser consultada en la web de *OWASP Foundation,* por medio del siguiente enlace:

  https://owasp.org/www-project-vulnerability-management-guide/

- **NIST SP800-115. Technical Guide to Information Security Testing and Assessment**

  "La guía proporciona recomendaciones prácticas para diseñar, implementar y mantener procesos y procedimientos de prueba y examen de seguridad de la información técnica. Estos pueden usarse para varios propósitos, como encontrar vulnerabilidades en un sistema o red y verificar el cumplimiento de una política u otros requisitos. La guía no pretende presentar un programa integral de pruebas y exámenes de seguridad de la información, sino más bien una descripción general de los elementos clave de las pruebas y exámenes de seguridad técnica, con énfasis en técnicas específicas, los beneficios y limitaciones de cada una, y recomendaciones para su uso."
  Scarfone, Souppaya, Cody y Orebaugh (2008).

La guía se encuentra en idioma inglés, pero puede ser fácilmente traducida. La misma puede ser consultada en la *web* del *National Institute of Standards and Technology (NIST)* de los Estados Unidos, por medio del siguiente enlace:

https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-115.pdf

1. Metodologías

Ponemos en su conocimiento, las siguientes metodologías para el análisis de vulnerabilidades técnicas:

- **OWASP Top 10**

  Es un documento estándar que busca concientizar sobre la seguridad en aplicaciones *web*, especialmente a los desarrolladores e implementadores de aplicaciones, servicios y sistemas web. Aborda diez aspectos de riesgo claves para las aplicaciones *web*, los cuales se listan a continuación.

  *"**A1: Inyección:** las fallas de inyección, como SQL, NoSQL, OS y LDAP, ocurren cuando se envían datos que no son de confianza a un intérprete como parte de un comando o consulta. Los datos hostiles del atacante pueden engañar al intérprete para que ejecute comandos no deseados o acceda a los datos sin la debida autorización.*

  ***A2: Autenticación rota:** las funciones de la aplicación relacionadas con la autenticación y la administración de sesiones a menudo se implementan de manera incorrecta, lo que permite a los atacantes comprometer contraseñas, claves o tokens de sesión, o explotar otras fallas de implementación para asumir las identidades de otros usuarios de forma temporal o permanente.*

  ***A3: Exposición de datos confidenciales:** muchas aplicaciones web y API no protegen adecuadamente los datos confidenciales, como los financieros, la atención médica y la PII. Los atacantes pueden robar o modificar esos datos débilmente protegidos para cometer fraude con tarjetas de crédito, robo de identidad u otros delitos. Los datos confidenciales pueden verse comprometidos sin protección adicional, como el cifrado en reposo o en tránsito, y requieren precauciones especiales cuando se intercambian con el navegador.*

  ***A4: Entidades externas XML (XXE):** muchos procesadores XML más antiguos o mal configurados evalúan las referencias de entidades externas dentro de los documentos XML. Las entidades externas se pueden utilizar para divulgar archivos internos mediante el controlador de archivos URI, recursos compartidos de archivos internos, escaneo de puertos internos, ejecución remota de código y ataques de denegación de servicio.*

  ***A5: Control de acceso roto:** las restricciones sobre lo que los usuarios autenticados pueden hacer a menudo no se aplican correctamente. Los atacantes pueden aprovechar estas fallas para acceder a funciones y / o datos no autorizados, como acceder a las cuentas de otros usuarios, ver archivos confidenciales, modificar los datos de otros usuarios, cambiar los derechos de acceso, etc.*

A6: Configuración incorrecta de seguridad: *la configuración incorrecta de seguridad es el problema más común. Esto suele ser el resultado de configuraciones predeterminadas inseguras, configuraciones incompletas o ad hoc, almacenamiento en la nube abierta, encabezados HTTP mal configurados y mensajes de error detallados que contienen información confidencial. No solo todos los sistemas operativos, marcos, bibliotecas y aplicaciones deben estar configurados de manera segura, sino que también deben ser parcheados / actualizados de manera oportuna.*

A7: 2017-Cross-Site Scripting XSS: *las fallas XSS ocurren cuando una aplicación incluye datos que no son de confianza en una nueva página web sin la validación o el escape adecuados, o actualiza una página web existente con datos proporcionados por el usuario utilizando una API de navegador que puede crear HTML o JavaScript. XSS permite a los atacantes ejecutar scripts en el navegador de la víctima que pueden secuestrar sesiones de usuario, desfigurar sitios web o redirigir al usuario a sitios maliciosos.*

A8: Deserialización insegura de 2017: *la deserialización insegura a menudo conduce a la ejecución remota de código. Incluso si las fallas de deserialización no dan como resultado la ejecución remota de código, se pueden usar para realizar ataques, incluidos ataques de reproducción, ataques de inyección y ataques de escalada de privilegios.*

A9: Uso de componentes con vulnerabilidades conocidas: *los componentes, como bibliotecas, marcos y otros módulos de software, se ejecutan con los mismos privilegios que la aplicación. Si se explota un componente vulnerable, dicho ataque puede facilitar la pérdida de datos o la toma de control del servidor. Las aplicaciones y API que utilizan componentes con vulnerabilidades conocidas pueden socavar las defensas de las aplicaciones y permitir varios ataques e impactos.*

A10: Registro y monitoreo insuficientes: *El registro y el monitoreo insuficientes, junto con una integración faltante o ineficaz con la respuesta a incidentes, permiten a los atacantes atacar aún más los sistemas, mantener la persistencia, cambiar a más sistemas y manipular, extraer o destruir datos. La mayoría de los estudios de infracciones muestran que el tiempo para detectar una infracción es de más de 200 días, generalmente detectados por partes externas en lugar de procesos internos o monitoreo"*.
OWASP Foundation (2017)

El *OWASP Top Ten* busca minimizar las vulnerabilidades en el desarrollo e implementación de aplicaciones.

- **Guía metodológica de pruebas de efectividad del MINTIC**

"Esta guía tiene como finalidad, indicar los procedimientos de seguridad que pueden generarse durante el proceso de evaluación en los avances en la implementación de la seguridad. La metodología de pruebas de efectividad es una serie de actividades, que tienen por finalidad comprobar o medir la eficiencia de la implementación del modelo de seguridad en las entidades".
MINTIC (2016)

Esta guía estructura un procedimiento de ejecución de pruebas de efectividad con un enfoque de *Ethical Hacking*, en donde se realizan las etapas de contextualización, reconocimiento del objetivo, modelo de amenazas, **análisis de vulnerabilidades**, explotación y reporte.

Se puede consultar la guía en la página web del MinTIC por medio del siguiente enlace:

https://www.mintic.gov.co/gestionti/615/articles-5482_G1_Metodologia_pruebas_efectividad.pdf

2. Conceptos

Para poder realizar un análisis de vulnerabilidades técnicas es importante conocer conceptos básicos de seguridad de la información, entendiendo que en seguridad siempre se busca proteger la confidencialidad, integridad y disponibilidad de los datos, por medio de actividades que permitan cerrar brechas y vulnerabilidades que puedan ser aprovechadas por posibles amenazas. También es importante conocer conceptos básicos como vulnerabilidad, amenaza, entre otros.

- **Elementos vulnerables**

  En ciberseguridad, son todos los componentes de *Hardware y Software*, que son parte de una red local o pública que están expuestos a posibles amenazas y riesgos que los hacen vulnerables. Entre ellos están: equipos, dispositivos móviles e inalámbricos, servidores, *switches, routers, software* de sistemas, aplicaciones y servicios en red.

- **Vulnerabilidad**

  En informática, se define como una debilidad o fallo de seguridad que se presenta en un sistema de información, que puede estar compuesto por *software*, *hardware* y otros componentes y servicios tecnológicos, generando riesgos de seguridad de la información.

- **Amenaza**

  Se define como toda aquella acción o serie de acciones que aprovechan las vulnerabilidades para romper la seguridad de los sistemas.

- **Escáner de vulnerabilidades**

  Herramienta *Software* que busca y analiza las debilidades o fallas de los elementos o dispositivos que componen una red.

- **Falsos positivos**

  Se refiere a falsas vulnerabilidades reportadas por un escáner de vulnerabilidades. El escáner reporta la existencia de una vulnerabilidad, pero en el proceso de comprobación se detecta que tal vulnerabilidad no existe o está bajo control, de tal forma que no es explotable.

- **Falsos negativos**

  También denominados fugas en ciberseguridad, son vulnerabilidades existentes en el *Software* (Aplicaciones, sistemas, servicios), que los escáneres de búsqueda automática de vulnerabilidades no logran detectar en las actividades de evaluación de la seguridad digital.

3. Características

Las características del análisis de vulnerabilidades se centran en que los análisis permiten el descubrimiento de activos en una red.

- **Escalas de seguridad**

  Los sistemas de análisis de vulnerabilidades en el momento de análisis, clasifican y ordenan las vulnerabilidades según niveles o criticidad de seguridad de las mismas. Estas clasificaciones pueden variar, pero por lo general se establecen de la siguiente manera:

  - Informativas: a*grupan las vulnerabilidades que no representan un riesgo de impacto.*
  - Bajas: a*grupan las vulnerabilidades que representan un riesgo bajo de impacto.*
  - Medias: a*grupan las vulnerabilidades que representan un riesgo medio de impacto.*
  - Altas: agrupan las vulnerabilidades que representan un riesgo alto de impacto.

- **Características para la enumeración de vulnerabilidades**

  Existen varios sistemas y metodologías de enumeración de vulnerabilidades (registro, bases de datos y fuentes de consulta) tales como *Common Vulnerability Scoring System (CVSS) y el Common Weakness Scoring System (CWSS™)*, que buscan llevar un registro de las vulnerabilidades de manera ordenada y clasificando o puntuando, teniendo en cuenta características particulares, entre ellas las siguientes.

  - Impacto de la confidencialidad.
  - Impacto en la integridad parcial.
  - Impacto en la disponibilidad.
  - Complejidad de acceso.
  - Autenticación.
  - Acceso obtenido.
  - Tipo (s) de vulnerabilidad.
  - Identificadores CVSS, CWSS.

4. *Software*

Existen muchas herramientas o soluciones *software* para realizar análisis de vulnerabilidades. Se encuentran soluciones comerciales o de pago, y también herramientas *open source* o en versiones denominadas *Community*.

En la tabla que se muestra a continuación, se mencionan algunas de ellas:

Tabla 1

Herramientas de análisis de vulnerabilidades

Nombre / Enlace	Dueño	Licencia	Plataformas	Nota
Abbey Scan	MisterScanner	Comercial	SaaS	
Acunetix	Acunetix	Comercial	Windows, Linux, MacOS	Gratis (capacidad limitada)
Escáner de aplicaciones	Trustwave	Comercial	Ventanas	
AppCheck Ltd.	AppCheck Ltd.	Comercial	SaaS	Escaneo de prueba gratuito disponible
AppScan	Software HCL	Comercial	Ventanas	
AppScan en la nube	Software HCL	Comercial	SaaS	
Arachni	Arachni	Gratis	La mayoría de las plataformas son compatibles	Gratis para la mayoría de los casos de uso
Suite de seguridad Astra	Seguridad Astra	Gratis	SaaS	Opción de pago disponible
Prueba de seguridad de aplicaciones dinámicas BREACHLOCK	BLOQUEO	Comercial	SaaS	
Seguridad Beagle	Seguridad Beagle	Comercial	SaaS	Gratis (capacidad limitada)
Detección de BC BlueClosure	BlueClosure	Comercial	La mayoría de las plataformas son compatibles	Prueba de 2 semanas
Suite Burp	PortSwiger	Comercial	La mayoría de las plataformas son compatibles	Gratis (capacidad limitada)
Contraste	Seguridad de contraste	Comercial	SaaS o en las instalaciones	Gratis (con todas las funciones para 1 aplicación)
Seguridad Crashtest	Seguridad Crashtest	Comercial	SaaS o en las instalaciones	
Jefe cibernético	Audacix	Comercial	SaaS o en las instalaciones	
Detectar	Detectar	Comercial	SaaS	
Edgescan	Edgescan	Comercial	SaaS	
Ladrón	*Romain Gaucher*	Fuente abierta	Python 2.4, BeautifulSoup y PyXML	
Escaneo de Grendel	*David Byrne*	Fuente abierta	Windows, Linux y Macintosh	

Nombre / Enlace	Dueño	Licencia	Plataformas	Nota
HostedScan.com	*HostedScan*	Comercial	SaaS	Siempre libre
IKare	Confío	Comercial	N / A	
IOTHREAT	IOTHREAT	Comercial	SaaS	Gratis (Ver resultados parciales). Informe completo (PRO): 50% de descuento para la comunidad OWASP con 'OWASP50'.
ImmuniWeb	Puente de alta tecnología	Comercial	SaaS	Gratis (capacidad limitada)
Escaneo de aplicaciones web de Indusface	*Indusface*	Comercial	SaaS	Prueba gratuita disponible
InsightVM	*Rapid7*	Comercial	SaaS	Prueba gratuita disponible
Intruso	*Intruder Ltd.*	Comercial		
Plataforma de seguridad K2	*Seguridad cibernética de K2*	Comercial	SaaS / local	Prueba gratuita disponible
N-Stealth	*N-Stalker*	Comercial	Ventanas	
Nessus	Sostenible	Comercial	Ventanas	
Netsparker	*Netsparker*	Comercial	Ventanas	
Nikto	*CIRT*	Fuente abierta	Unix / Linux	
Colecciones de herramientas Nmmapper	*Nmmapper*	Comercial	Hablar con descaro a	Gran colección de Kali Tool alojada en línea
Núcleos	*ProyectoDiscovery*	Fuente abierta	Windows, Unix / Linux y Macintosh	Escáner de vulnerabilidades rápido y personalizable basado en DSL simple basado en YAML.
Probely	*Probely*	Comercial	SaaS	Gratis (capacidad limitada)
QualysGuard	*Qualys*	Comercial	N / A	
ReconwithMe	*Nassec*	Comercial	SaaS	Opción de pago disponible
Retina	BeyondTrust	Comercial	Ventanas	
Paseo (fuzzer de carga útil REST JSON)	Adobe, Inc.	Fuente abierta	Linux / Mac / Windows	Apache 2
SOATest	Parasoft	Comercial	Windows / Linux / Solaris	
Escáner de vulnerabilidades ScanTitan	ScanTitan	Comercial	SaaS	Gratis (capacidad limitada)
Sec-helpers	VWT Digital	Código abierto o gratuito	N / A	
Penetrador SecPoint	SecPoint	Comercial	N / A	
Seguridad para todos	Seguridad para todos	Comercial	SaaS	Gratis (capacidad limitada)

Nombre / Enlace	Dueño	Licencia	Plataformas	Nota
Centinela	Seguridad WhiteHat	Comercial	N / A	
Seguridad de hojalata	Sinopsis	Comercial	SaaS o en las instalaciones	Gratis (capacidad limitada)
Escáner de Trustkeeper	Trustwave SpiderLabs	Comercial	SaaS	
Vega	Subgrafo	Fuente abierta	Windows, Linux y Macintosh	
Vejar	*UBsecure*	Comercial	Ventanas	
Wapiti	Informática Gesfor	Fuente abierta	Windows, Unix / Linux y Macintosh	
WebApp360	*TripWire*	Comercial	Ventanas	
WebInspect	Enfoque micro	Comercial	Ventanas	
Comprobación de seguridad del sitio web	*CyberAnt*	Comercial	SaaS	20% de descuento con OWASP20
Proxy de ataque Zed	OWASP	Fuente abierta	Windows, Unix / Linux y Macintosh	Apache-2.0
beSECURE (anteriormente AVDS)	Más allá de la seguridad	Comercial	SaaS	Gratis (capacidad limitada)
equipo púrpura	OWASP	Fuente abierta	CLI y SaaS	GNU-AGPL v3
w3af	W3af	Fuente abierta	Linux y Mac	GPLv2.0

Nota: Tomado y traducido de OWASP Foundation (2021) https://owasp.org/www-community/Vulnerability_Scanning_Tools

5. Aplicación

La aplicación de análisis de vulnerabilidades se puede realizar en varios escenarios según el contexto y los requerimientos y objetivos que se pretendan alcanzar.

Realizar un análisis de vulnerabilidades puede estar sujeto a la necesidad de fortalecer las buenas prácticas de configuración en dispositivos de red, monitorear la funcionalidad de controles, y verificar el cumplimiento de los estándares de seguridad.

- **Análisis de vulnerabilidades para *hardening***

  Este tipo de análisis se enfoca en descubrir vulnerabilidades que demuestren malas configuraciones en los dispositivos y servicios de red (Equipos, servidores, *firewalls*, sistemas, aplicaciones, servicios, etc.) y no aplicación de parches de seguridad. De esta manera los equipos de seguridad podrán tomar las acciones correctivas para endurecer la seguridad de los dispositivos y servicios de red.

- **Análisis de vulnerabilidades para monitoreo**

  Se centra en descubrir nuevas vulnerabilidades en los sistemas de protección, de tal forma que pueda monitorearse que los controles no están siendo eficaces, de esta manera los equipos de seguridad pueden realizar los ajustes pertinentes en los controles de ciberseguridad.

- **Análisis de vulnerabilidades para auditoría**

  Este tipo de análisis se puede dar como parte de un ejercicio para la obtención de una certificación especial de ciberseguridad o dentro de un proceso de auditoría interna para la evaluación del cumplimiento de los procesos y equipos de seguridad, conforme a los requisitos de negocio.

Los análisis de vulnerabilidades son un paso previo a las pruebas de intrusión dentro del proceso de *Ethical Hacking* y por sí solos no determinan el panorama de ciberseguridad de los sistemas de información en una organización.