Prueba de Penetración

HTB Labs (Pandora)

nico.sanchezsierra@hotmail.com, OSID: OS-014

2025-11-17

Contents

1 Reporte

1.1 Introducción

Este documento de Pruebas de Penetración está basado en máquinas simuladas en entornos controlados.

Las máquinas encontradas en esta serie de Documentos pueden ser encontradas en plataformas como Hack The Box, Try Hack Me, entre otras.

La máquina de hoy la podemos encontrar en Hack The Box con el nombre de Pandora. Dicho esto, continuemos.

1.2 Objetivo

La finalidad de estos Reportes de Penetración es demostrar mi capacidad a la hora de identificar y explotar vectores de ataque, además de demostrar mi capacidad a la hora de documentarlas. Se intenta buscar un formato lo más profesional posible teniendo en cuenta la experiencia en este campo. Por ese motivo se sigue un proceso riguroso y meticuloso, estructurado y siguiendo metodologías MITRE ATT&CK, CEH y OSCP.

Para los Reportes de Penetración se usan máquinas de HTB (Hack The Box) o THM (TryHackMe), de modo que sí se nos permite documentar y trabajar públicamente con ellas.

2 Resumen Alto Nivel

Se me encargó realizar una prueba de penetración interna hacia una máquina de Hack The Box. Una prueba de penetración interna es un ataque dedicado contra sistemas conectados internamente. El enfoque de esta prueba es identificar vulnerabilidades que supongan un riesgo al sistema y documentarlas. Mi objetivo era evaluar la red, identificar sistemas y explotar fallos mientras informamos de ello.

Al realizar la prueba interna, se hallaron varias vulnerabilidades preocupantes que fueron identificadas y reportadas. Durante las pruebas, pude obtener acceso a nivel administrativo sobre el sistema encargado.

A continuación se enumerarán las vulnerabilidades y fallas del sistema que suponen un riesgo al sistema. Serán clasificadas dependiendo la exposición, la facilidad y el impacto de las mismas.

Crítico	Alto	Medio	Bajo	Total
3	1	0	0	4

ID	Riesgo	CVE	Nombre Descriptivo
1	Crítico	N/A	Credenciales expuestas en SNMP
2	Crítico	CVE-2020-5844	Servidor Web vulnerable a RCE
3	Alto	N/A	Posibilidad de modificación de SSH keys
4	Crítico	N/A	Elevación de Privilegios de sistema

2.1 Recomendaciones

Recomiendo corregir las vulnerabilidades identificadas durante las pruebas para asegurar que un atacante no pueda explotar estos sistemas en el futuro. Una cosa a recordar es que estos sistemas requieren parches frecuentes y una vez parcheados, deberían mantenerse en un programa regular de parches para proteger las vulnerabilidades adiciones que se descubran más tarde.

3 Metodología

Utilicé un enfoque estándar de pruebas de penetración que incluye las fases de enumeración, explotación, post-explotación, persistencia y limpieza de pruebas. Este método es comúnmente empleado en entornos de certificación Offensive Security para evaluar la seguridad de sistemas y redes.

A continuación, se describen los pasos realizados para identificar y explotar las vulnerabilidades encontradas durante la prueba de penetración.

3.1 Recolección de Información

La parte de recopilación de información de una prueba de penetración se centra en identificar los límites y las tecnologías de nuestro objetivo. Durante esta prueba de penetración fui asignado la siguiente IP.

Redes disponibles

- 10.10.11.136

3.2 Penetración

Las partes de la prueba de penetración se centran en gran medida en obtener acceso a una variedad de sistemas. Durante la prueba de penetración, pude acceder con éxito al sistema encargado.

3.2.1 Sistema IP: 10.10.11.136

3.2.1.1 Enumeración de servicios

La parte de enumeración de servicios de una prueba de penetración se centra en recopilar información sobre qué servicios están activos en un sistema. Esto es valioso para un atacante, ya que proporciona información detallada sobre posibles vectores de ataque en un sistema. Entender qué aplicaciones

se están ejecutando en el sistema le brinda al atacante la información necesaria antes de realizar la prueba de penetración real.

Dirección IP	Puertos Abiertos
10.10.11.136	22,80,161

Resultados del escaneo de Nmap (puertos TCP):

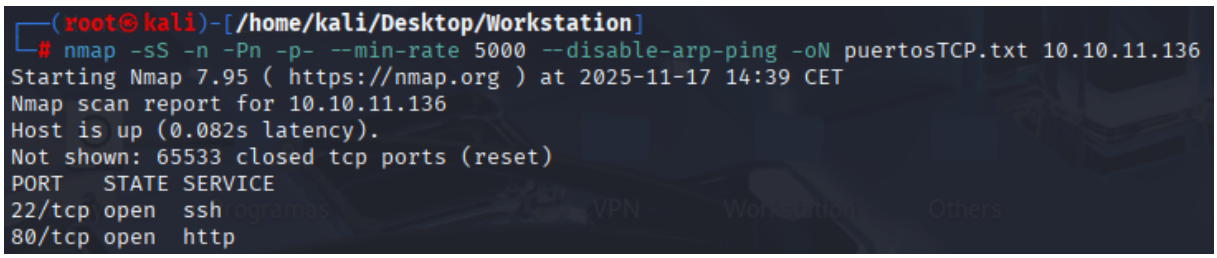

Figure 3.1: nmap -sS -n -Pn -p- –min-rate 5000 –disable-arp-ping 10.10.11.136

Resultados del escaneo de Nmap (puertos UDP):

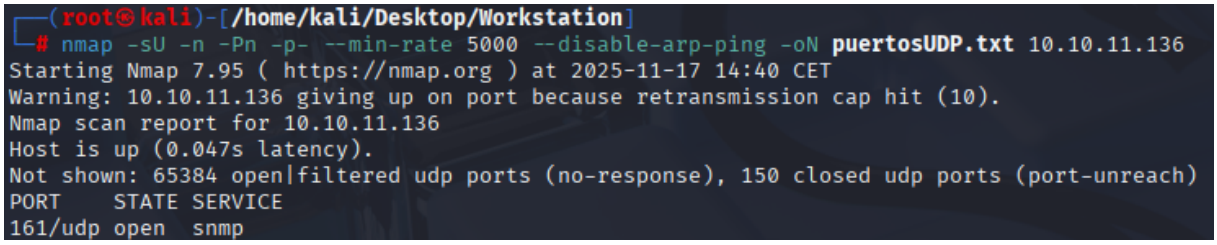

Figure 3.2: nmap -sU -n -Pn -p- –min-rate 5000 –disable-arp-ping 10.10.11.136

Resultado del escaneo de Nmap (versiones):

```
┌──(root㉿kali)-[/home/kali/Desktop/Workstation]
└─# nmap -sCV -O -p22,80,161 -oN versiones.txt 10.10.11.136
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-17 14:43 CET
Nmap scan report for 10.10.11.136
Host is up (0.10s latency).

PORT    STATE  SERVICE VERSION
22/tcp  open   ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 24:c2:95:a5:c3:0b:3f:f3:17:3c:68:d7:af:2b:53:38 (RSA)
|   256 b1:41:77:99:46:9a:6c:5d:d2:98:2f:c0:32:9a:ce:03 (ECDSA)
|_  256 e7:36:43:3b:a9:47:8a:19:01:58:b2:bc:89:f6:51:08 (ED25519)
80/tcp  open   http    Apache httpd 2.4.41 ((Ubuntu))
|_http-server-header: Apache/2.4.41 (Ubuntu)
|_http-title: Play | Landing
161/tcp closed snmp
Device type: general purpose
Running: Linux 5.X
OS CPE: cpe:/o:linux:linux_kernel:5
OS details: Linux 5.0 - 5.14
Network Distance: 2 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Figure 3.3: nmap -sCV -O -p22,80,161 10.10.11.136

3.2.1.2 Explotación y Post-Explotación de Vulnerabilidades

En este informe se ha decidido presentar la explotación y la post-explotación como una única sección. Esta organización facilita la verificación técnica: el revisor puede encontrar en un mismo bloque qué falla, cómo se explota y cuál fue el impacto real, mejorando la trazabilidad entre la evidencia y el hallazgo.

En una prueba de penetración la explotación se centra en explotar los vectores de ataque que anteriormente se enumeraron, ganando así acceso al sistema. Por otro lado, la post-explotación se basa en aumentar privilegios y obtener acceso administrativo.

A continuación he enumerado las fallas y vulnerabilidades que anotamos en el Resumen de Alto Nivel, pero más detalladamente. Además, se explica la vulnerabilidad, se muestra que es explotable y se presentan mitigaciones.

3.2.1.3 Vulnerabilidad (ID: 1, Credenciales expuestas en SNMP)

Riesgo: Crítico

CVE: N/A

Servicios Afectados: Servicio SNMP (puerto 161)

Explicación de la vulnerabilidad:

El sistema estaba configurado para permitir que cualquiera pudiera ver información interna sin necesidad de autenticar. Dentro de esa información se encontró un Usuario con Credenciales.

El servicio SNMP utiliza la community string public. Esto permite ejecutar consultas sin autenticación. Además, el servicio SNMP opera bajo SNMPv1, que no cifra los datos.

El impacto es crítico, pues hay filtración de usuarios e información sobre un servicio no seguro (sin autenticación ni cifrado de datos). Esto permite al atacante obtener información para atacar nuevos servicios o acceder al sistema.

Remedio de la vulnerabilidad:

- Deshabilitar SNMPv1/SNMPv2c y usar SNMPv3 (autenticación y cifrado)
- Evitar cadenas de comunidad como public y private
- Restringir/Limitar el acceso a administradores

Pruebas:

Con la herramienta “snmpwalk” se identificó un output con Usuario y Credenciales del sistema.

```
┌──(root㉿kali)-[/home/kali/Desktop/Workstation]
└─# snmpwalk -c public -v1 10.10.11.136 iso.3.6.1.2.1.25.4.2.1.5.1118
iso.3.6.1.2.1.25.4.2.1.5.1118 = STRING: "-u daniel -p HotelBabylon23"
```

Figure 3.4: snmpwalk -v1 -c public 10.10.11.136 iso.3.6.1.2.1.25.4.2.1.5.1118

3.2.1.4 Vulnerabilidad (ID: 2, Servidor Web vulnerable a RCE)

Riesgo: Crítico

CVE: CVE-2020-5844

Servicios Afectados: Servidor Web HTTP (Pandora FMS)

Explicación de la vulnerabilidad:

El sistema tenía un fallo en su página web interna que permitía que un atacante ejecutara comandos directamente en el servidor.

El servidor web utiliza Pandora FMS versión 7.0NG.742, la cual es vulnerable a SQLi no autenticado. Un atacante puede manipular las consultas SQL para alterar su comportamiento y cargar código dentro de la base de datos. Además se combina con una falla del Software que permite ejecutar comandos del sistema almacenados en la base de datos SQL, resultando en un RCE.

El impacto es crítico, ya que la vulnerabilidad concede capacidad de ejecución remota de código. Eso significa que un atacante puede acceder al sistema interno a través del servidor web. Puede suponer pérdida de información, robo de información, manipulación de datos y acceso directo al sistema.

Remedio de la vulnerabilidad:

- Actualizar Pandora FMS a una versión corregida
- Implementar reglas WAF para detectar/prevenir SQLi y peticiones maliciosas
- Deshabilitar cualquier funcionalidad que permita ejecutar comandos en el sistema

Pruebas:

Primero se creó un túnel SSH para acceder al Servidor Web desde el interior.

Figure 3.5: ssh daniel@10.10.11.136 -L 8080:localhost:80

Se identificó una versión vulnerable de Pandora FMS (v7.0NG.742) susceptible a CVE-2020-5844.

Figure 3.6: http://localhost:8080

Se identificó un ejecutable python en “https://github.com/shyam0904a/Pandora_v7.0NG.742_exploit_unauthenticated para explotar la vulnerabilidad sin autenticación previa.

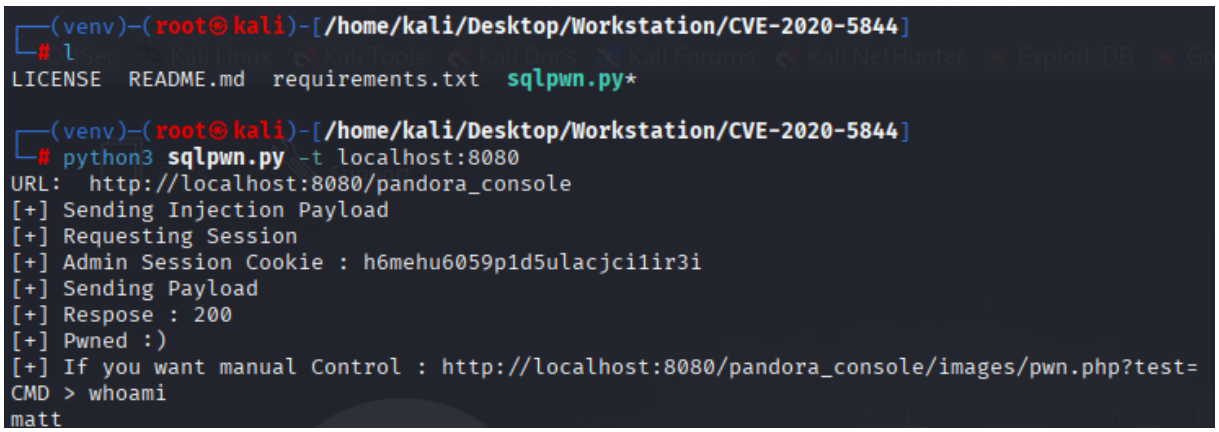

Figure 3.7: python3 sqlpwn.py -t localhost:8080

3.2.1.5 Vulnerabilidad (ID: 3, Posibilidad de modificación de SSH keys)

Riesgo: Alto

CVE: N/A

Servicios Afectados: Usuarios del sistema y Servicio SSH

Explicación de la vulnerabilidad:

Dentro del sistema comprometido se encontró que era posible modificar las claves de los usuarios para acceder por SSH sin necesidad de las credenciales.

Tras obtener acceso, se identificó que los usuarios podían modificar el directorio "/.ssh/" y específicamente "authorized_keys", sin restricciones.

Este impacto se considera alto, pues permite que un atacante acceda a las cuentas de usuarios sin necesidad de saber sus credenciales. Además permite que el atacante use una clave creada por él mismo para acceder a las cuentas.

Remedio de la vulnerabilidad:

- Restringir permisos de los directorios SSH
- Añadir mecanismos de auditoría y alertas

Pruebas:

Se insertó un String para autorizar nuestra máquina a autenticar con Clave (id_rsa) sin conocimiento de las credenciales.

```
CMD > echo 'ssh%2Drsa%20AAAAB3NzaC1yc2EAAAADAQABAAACAQDR6FaT3JsYV5SpDeZb7KB8gUHzf840OK59Dxh4F48Rr6rZ4mksY%2BZL%2BxkBZrQLx3TnFXFnyE
KVQlCBDdvW5zr000p3kr2XmdUtLIKb%2Bgh90dYEESiH2VI6N6BjljIpYCTuY46pUHff0OXfOVT3zD05UYCe1OU4xZDfMYPLS6%2F0k63u6cApP6P1IjVPxz8na4n95XTk
ZllHrZcDkF8RZN4FdgR4Zc1i9Mk9%2BFsVTVpCW7fsw5KKVYfiNHuEF4Uyn4EGhfdeqbj3S97C88gYqdDGmry5wpGJXPCCrdgqSzwvkdElrGvTjGKFXm68u14g%2B8rmk1
WkQCy86Bh%2B0K2vVjI8jPV5Fxo7ShAC21GZe7lmDnfs9vkWA9RSkkPzeOk7m7hDnvPkLxc6KvDr69DxqQAf7Ho0Gk1wHxDB4mr8cDgsMF776hug%2BCzeR8%2BR%2B6dX
9C4BaYdUr305l3pxdUs3lfdUFRfyWKT%2FX2WQVMLduORDT1c9czUlVk5afdvwxRg1AVrvTIGxQjYybslCFvQQJLuyntqzFoCc7IltH7jf%2FQ6p35%2FjSLNlN5rvAod3
3E%2F6OEXVgaQ81EADkfGw%2BHc2zzIoSlOhKeCtmTWq9ueIFuMuX8JGvULEdupqzy0kL6eZEqwTCfS3sqRo6oP%2Fbt727iJNWLKXrCMPcmq97aSClS%2FFUQ%3D%3D%2
0root%40kali' > /home/matt/.ssh/authorized_keys
```

Figure 3.8: Injección a través de la vulnerabilidad CVE-2020-5844

Permitiendo así el acceso al usuario “matt” sin usar sus credenciales.

```
┌──(root㉿kali)-[/home/kali/Desktop/Workstation]
└─# ssh -i id_rsa matt@panda.htb
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.4.0-91-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Mon 17 Nov 15:39:35 UTC 2025

  System load:  0.0               Processes:             235
  Usage of /:   63.2% of 4.87GB   Users logged in:       1
  Memory usage: 8%                IPv4 address for eth0: 10.10.11.136
  Swap usage:   0%

  ⇒ /boot is using 91.8% of 219MB


0 updates can be applied immediately.


The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Che



The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

matt@pandora:~$ whoami
matt
```

Figure 3.9: ssh -i id_rsa matt@10.10.11.136

3.2.1.6 Vulnerabilidad (ID: 4, Elevación de Privilegios de sistema)

Riesgo: Crítico

CVE: N/A

Servicios Afectados: Usuario Root y ejecutable “pandora_backup”

Explicación de la vulnerabilidad:

Se encontró que un usuario del sistema podía ejecutar un programa específico con privilegios elevados.

El usuario “matt” tenía permisos especiales sobre “pandora_backup”, el cual utiliza el comando “tar” sin usar caminos absolutos. Esto permite realizar PATH Hijacking al comando “tar” lanzar un comando con privilegios elevados.

El impacto es crítico, ya que permite al atacante ejecutar comandos con privilegios elevados, habiendo la posibilidad de obtener una sesión de administrador del sistema.

Remedio de la vulnerabilidad:

- Modificar el ejecutable para usar rutas absolutas
- Eliminar permisos SUID si no son necesarios
- Validar y limpiar variables de entornos PATH antes de ejecutar comandos con privilegios elevados

Pruebas:

Se identificó que el usuario “matt” podía ejecutar “pandora_backup”.

```
matt@pandora:~$ find / 2>/dev/null -perm -u=s |  grep pandora
/usr/bin/pandora_backup
```

Figure 3.10: find / 2>/dev/null -perm -u=s | grep pandora_backup

Se examinó y se identificó que el comando “tar” no usa un camino absoluto en “pandora_backup”.

Figure 3.11: strings pandora_backup.sh

Se procedió a modificar el PATH de “matt” para crear una shell elevada al usar el ejecutable “pandora_backup”.

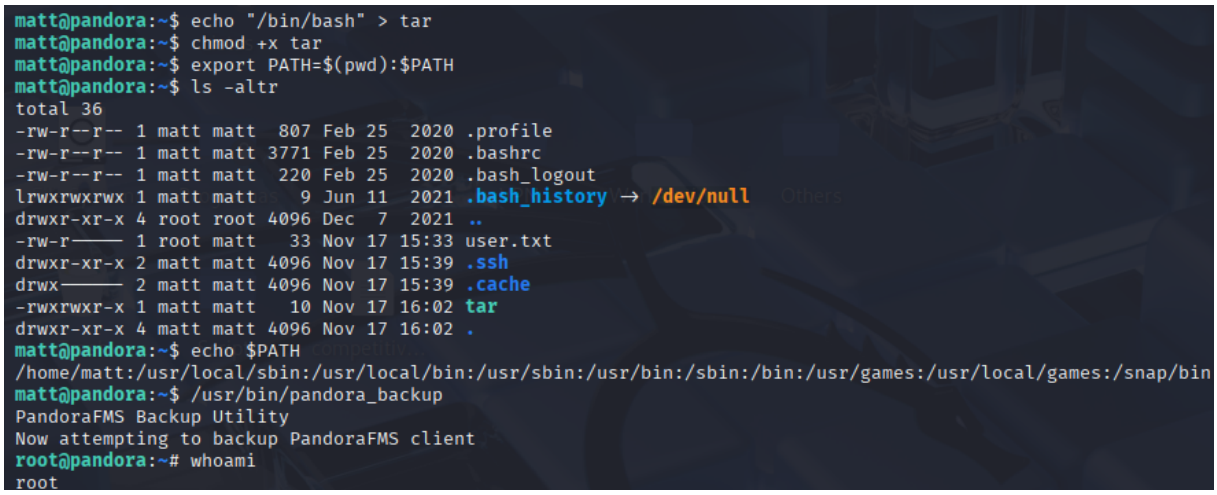

Figure 3.12: PATH injection + Elevated Privilege

3.3 Mantener Acceso

Mantener el acceso en un sistema es importante para nosotros como atacantes, asegurando que podamos volver a entrar en un sistema después de haber sido explotado.

La fase de mantenimiento de acceso de la prueba de penetración se centra en garantizar que una vez el ataque ha ocurrido, podamos volver a tener acceso administrativo fácilmente. Muchos exploits puedes ser ejecutados solo una vez y puede que nunca podamos volver a entrar en un sistema después de haber realizado la explotación.

Pruebas: Durante la prueba de penetración logramos tener acceso a los usuarios sin necesidad de credenciales. Con el usuario “matt” se logró obtener privilegios elevados de forma fácil y repetible.

Si no se toman medidas, el atacante podría seguir teniendo acceso a la máquina.

3.4 Limpieza de Pruebas

La parte de limpieza de pruebas nos garantiza que los restos ejecutados y creados durante la prueba de penetración estén completamente eliminados.

A menudo, fragmentos de herramientas o cuentas de usuario quedan en el sistema, lo que puede causar problemas de seguridad en un futuro.

Asegurarse de que somos meticulosos y que no quedan restos de nuestra prueba de penetración es importante.

Pruebas: Se borraron las claves SSH de los usuarios “daniel” y “matt”. Se borró el PATH Injection del usuario “matt”. Se borró la Web Shell interactiva al explotar la vulnerabilidad CVE-2020-5844.