

CTF = CATCH THE FLAG  
DOUBLETRouble machine

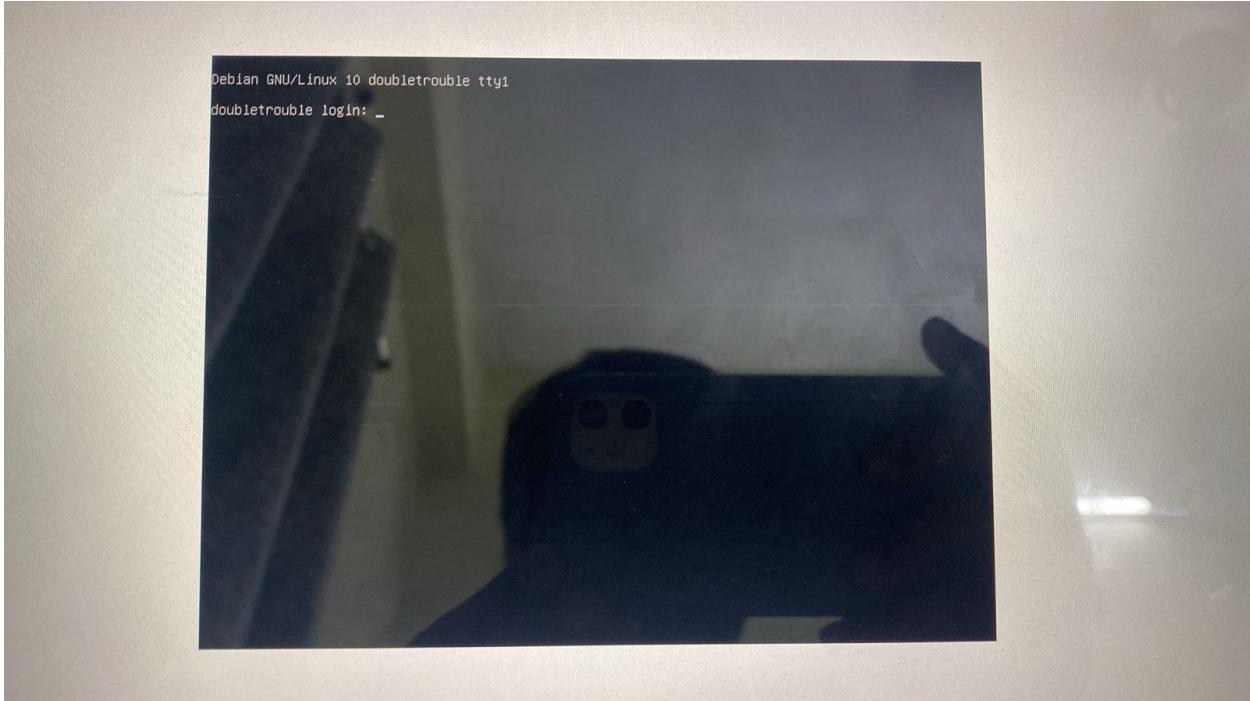
exploiting the doubletrouble machine

steps

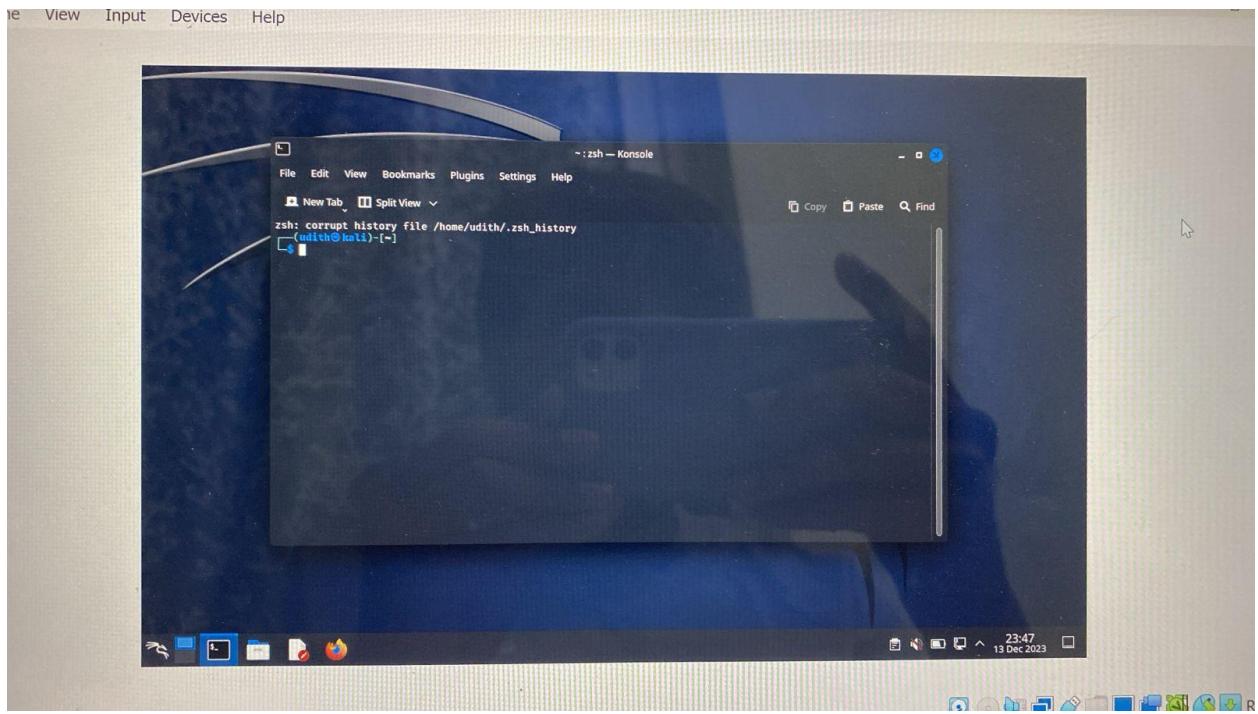
- 1: Getting the target machine IP address by using Netdiscover
- 2: Getting open port details by using the Nmap tool
- 3: Identifying vulnerabilities in running web applications
- 4: Enumerating applications with Drib utility
- 5: Cracking password with StegCracker

Download the Doubletrouble machine from the vulnhub.com and import in the virtual machine and the set the network to NAT and also check that the attacker machine also in NAT

After the importing start the virtual machine



open the terminal in kali

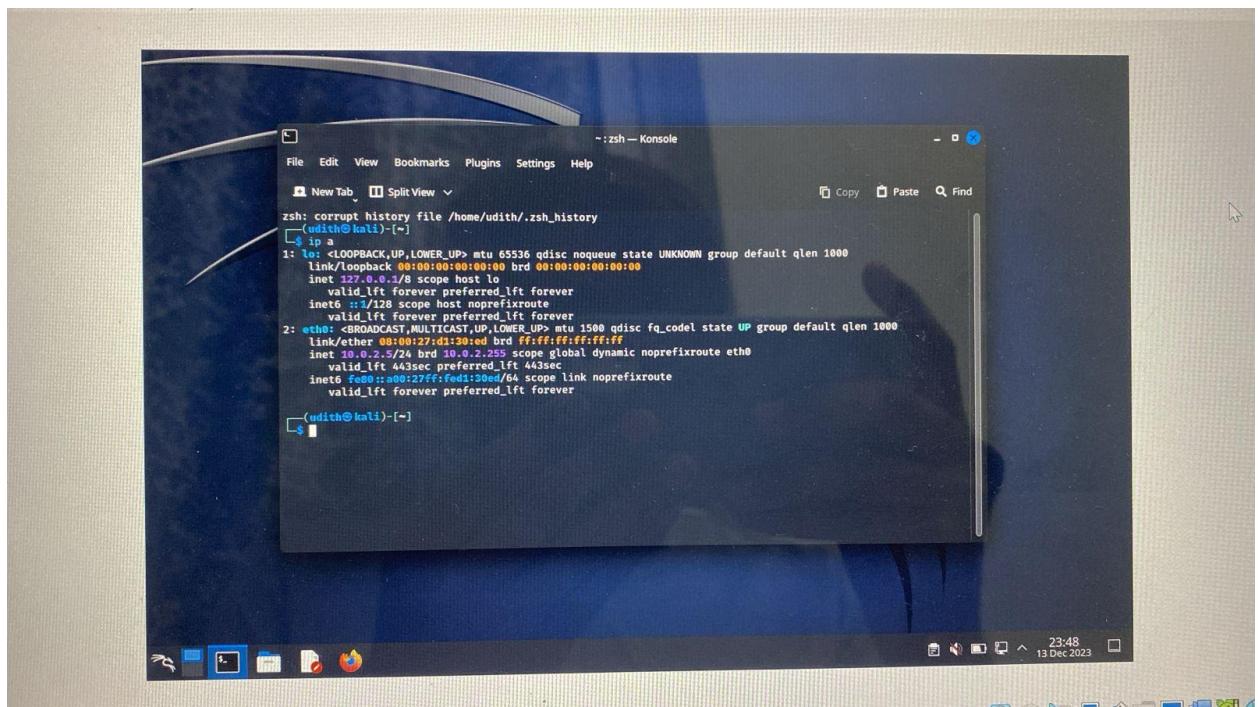


## Step 1

### GETTING THE TARGET MACHINE IP ADDRESS BY USING NETDISCOVER

Firstly we need to get the attacker IP address

Execute common "ip a"



The machine is in the eth0 interface and the IP address is 10.0.2.5

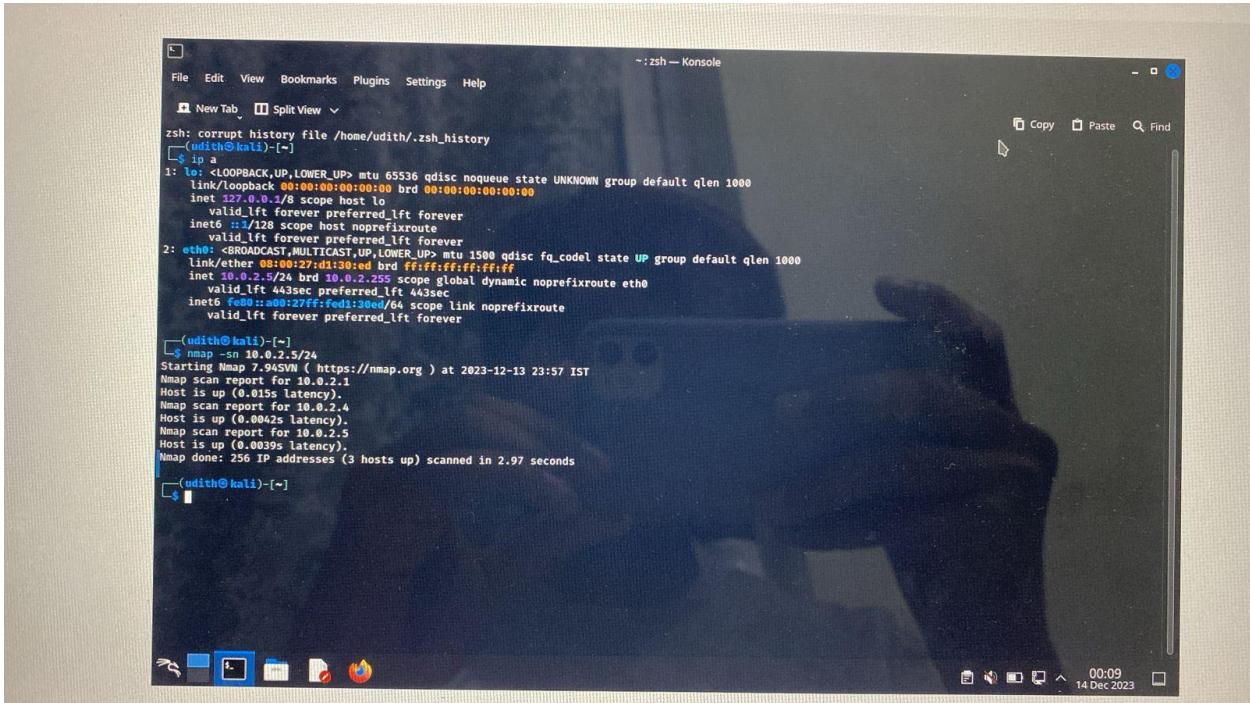
Next move is to find the target machine IP address by using the nmap and netdiscover we can get target machine IP address

Execute the command

"nmap -sn 10.0.2.5"

Where -sn is the ping scan flag

The result is



```
zsh: corrupt history file /home/udit@.zsh_history
(udit@kali)-[~]
$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever
            preferred_lft forever
inet6 ::1/128 scope host noprefixroute
    valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:00:00:27:d1:80 brd ff:ff:ff:ff:ff:ff
        inet 10.0.2.5/24 brd 10.0.2.255 scope global dynamic noprefixroute eth0
            valid_lft 443sec preferred_lft 443sec
        inet6 fe80::200:27ff:fed1:30e/64 scope link noprefixroute
            valid_lft forever preferred_lft forever
(udit@kali)-[~]
$ nmap -sn 10.0.2.5/24
Starting Nmap 7.84 ( https://nmap.org ) at 2023-12-13 23:57 IST
Nmap scan report for 10.0.2.1
Host is up (0.015s latency).
Nmap scan report for 10.0.2.4
Host is up (0.0042s latency).
Nmap scan report for 10.0.2.5
Host is up (0.0039s latency).
Nmap done: 256 IP addresses (3 hosts up) scanned in 2.97 seconds
(udit@kali)-[~]
```

And the another method in is the NETDISCOVER

Execute command "sudo netdiscover -i eth0 -r 10.0.2.5/24"

Flag -i is the interface flag

-r the attacker ip address

After the execution it will ask the password enter the kali password

We get the target machine ip address

```
-: sudo netdiscover --Konssole
File Edit View Bookmarks Plugins Settings Help
New Tab Split View
Currently scanning: Finished! | Screen View: Unique Hosts
4 Captured ARP Req/Rep packets, from 4 hosts. Total size: 240
IP At MAC Address Count Len MAC Vendor / Hostname
10.0.2.1 52:54:00:12:35:00 1 60 Unknown vendor
10.0.2.2 52:54:00:12:35:00 1 60 Unknown vendor
10.0.2.3 08:00:27:a9:e6:9e 1 60 PCS Systemtechnik GmbH
10.0.2.4 08:00:27:fd:30:3d 1 60 PCS Systemtechnik GmbH
```

The target machine ip address is 10.0.2.4

After getting the target machine ip address the next step is

## Step 2: GETTING THE OPEN PORT DETAILS BY USING THE NMAP TOOL

Getting the open port details on the machine

Execute the command

"nmap -p- -n -vvv -sCV 10.0.2.4"

Where flag -p- is for scan all ports

-n for do not resolve for dns(domain name system)

-vvv for verbosity it is used to get the open ports quickly

-sCV for default scripts and v for the version detection scan

The result is

```
(udith@kali)-[~]
└─$ nmap -p -n -vvv -SCV 10.0.2.4
Starting Nmap 7.94SVN ( https://nmap.org ) at 2023-12-14 00:20 IST
NSE: Loaded 156 scripts for scanning.
NSE: Script Pre-scanning.
NSE: Starting runlevel 1 (of 3) scan.
Initiating NSE at 00:20
Completed NSE at 00:20, 0.00s elapsed
NSE: Starting runlevel 2 (of 3) scan.
Initiating NSE at 00:20
Completed NSE at 00:20, 0.00s elapsed
NSE: Starting runlevel 3 (of 3) scan.
Initiating NSE at 00:20
Completed NSE at 00:20, 0.00s elapsed
NSE: Starting Connect Scan at 00:20
Scanning 10.0.2.4 [2 ports]
Completed Ping Scan at 00:20, 0.00s elapsed (1 total hosts)
Initiating Connect Scan at 00:20
Scanning 10.0.2.4 [65535 ports]
Discovered open port 80/tcp on 10.0.2.4
Discovered open port 22/tcp on 10.0.2.4
Completed Connect Scan at 00:20, 5.08s elapsed (65535 total ports)
Initiating Service scan at 00:20
Scanning 2 services on 10.0.2.4
Completed Service scan at 00:20, 7.31s elapsed (2 services on 1 host)
NSE: Script scanning 10.0.2.4.
NSE: Starting runlevel 1 (of 3) scan.
Initiating NSE at 00:21
Completed NSE at 00:21, 0.81s elapsed
NSE: Starting runlevel 2 (of 3) scan.
Initiating NSE at 00:21
Completed NSE at 00:21, 0.04s elapsed
NSE: Starting runlevel 3 (of 3) scan.
Initiating NSE at 00:21
Completed NSE at 00:21, 0.00s elapsed
Nmap scan report for 10.0.2.4
Host is up, received syn-ack (0.027s latency).
Scanned at 2023-12-14 00:20:47 IST for 13s
Not shown: 65533 closed tcp ports (conn-refused)
Not shown: 65533 closed tcp ports (conn-refused)
```

```
(udith@kali)-[~]
└─$ ssh -o StrictHostKeyChecking=no -p 22 10.0.2.4
Warning: Permanently added '10.0.2.4' (RSA) to the list of known hosts.
Last login: Sun Dec 14 00:20:47 2023 from 10.0.2.1
root@10.0.2.4:~#
```

```
Nmap map report for 10.0.2.4
Host is up, received syn-ack (0.027s latency).
Scanned at 2023-12-14 00:20:47 IST for 13s
Not shown: 65533 closed tcp ports (conn-refused)
PORT      STATE SERVICE REASON VERSION
22/tcp    open  ssh      syn-ack OpenSSH 7.9p1 Debian 10+deb10u2 (protocol 2.0)
|_ ssh-hostkey:
|   2048 6a:fe:d6:17:23:c8:90:79:2b:b1:2d:37:53:97:46:58 (RSA)
|   ssh-rsa AAAAB3NzaC1yc2EAAQDABAAAQc4uqqKnb1sYkzC7j1Mn80X4LqTf55m3n0lFxM6ID1r75V4JtHEGqnYs1wFGY0pwHJ80/pnc/Ehlnub7RCGyL5gxGKGhZPKYag6RDv0cJNgJHf5
|   otKJ0aFrhRZDXztGlfafCvWm0qgx3xweEVfU0GP2ac7jX80buja6bnxs7L0+DCBC1zxYwIqbRbkvKwpjQXNs/4HKLFz019db8C/GxEx4IekE98kZgcG20x/z08JMPXKHUCYKo1VxQCDmBgAn1ida
|   C7IBJMnC1VbVv7vNMrttaf/fftNDXb5Ydbqbbud2JsjmL0hHK3uWfHlEk0jBz23J
|   256 5b:c4:68:d1:89:59:d7:48:b6:96:f3:11:87:08:(EDDSA)
|   ecdsa-sha2-nistp256 AAAAE2VzZHMNLXNoYTtbmzdHAyNTYAAAAlbm1zGHAYNTYAAAABBB0kds8dHvtz2MxX2P71ej+q+QDe/MG80Gk7uYjNBTSK/T2R/QkD9FboGbq1+SpCox5qzIv08UQ+xvcE
|_ DPDv2:
|   256 61:39:66:88:1d:8f:f1:d0:40:61:1e:90:C5:1a:f4 (ED25519)
|   ssh-ed25519 AAAAC3NzaC1lZDI1NTEAAAAItokbb13ceQ01mFATBnU9sChiXF613cXEaY12Y2
80/tcp    open  http    syn-ack Apache httpd/2.4.38 ((Debian))
|_ http-title: qdPM | Login
|_ http-server-header: Apache/2.4.38 (Debian)
|_ http-methods:
|_ Supported Methods: GET HEAD POST OPTIONS
|_ http-favicon: Unknown favicon MD5: B0BD4BE57FD398C5D0A8E8F2CCC8D90D
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

NSE: Script Post-scanning.
NSE: Starting runlevel 1 (of 3) scan.
Initiating NSE at 00:21
Completed NSE at 00:21, 0.00s elapsed
NSE: Starting runlevel 2 (of 3) scan.
Initiating NSE at 00:21
Completed NSE at 00:21, 0.00s elapsed
NSE: Starting runlevel 3 (of 3) scan.
Initiating NSE at 00:21
Completed NSE at 00:21, 0.00s elapsed
Read data files from: /usr/bin/.../share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.80 seconds
```

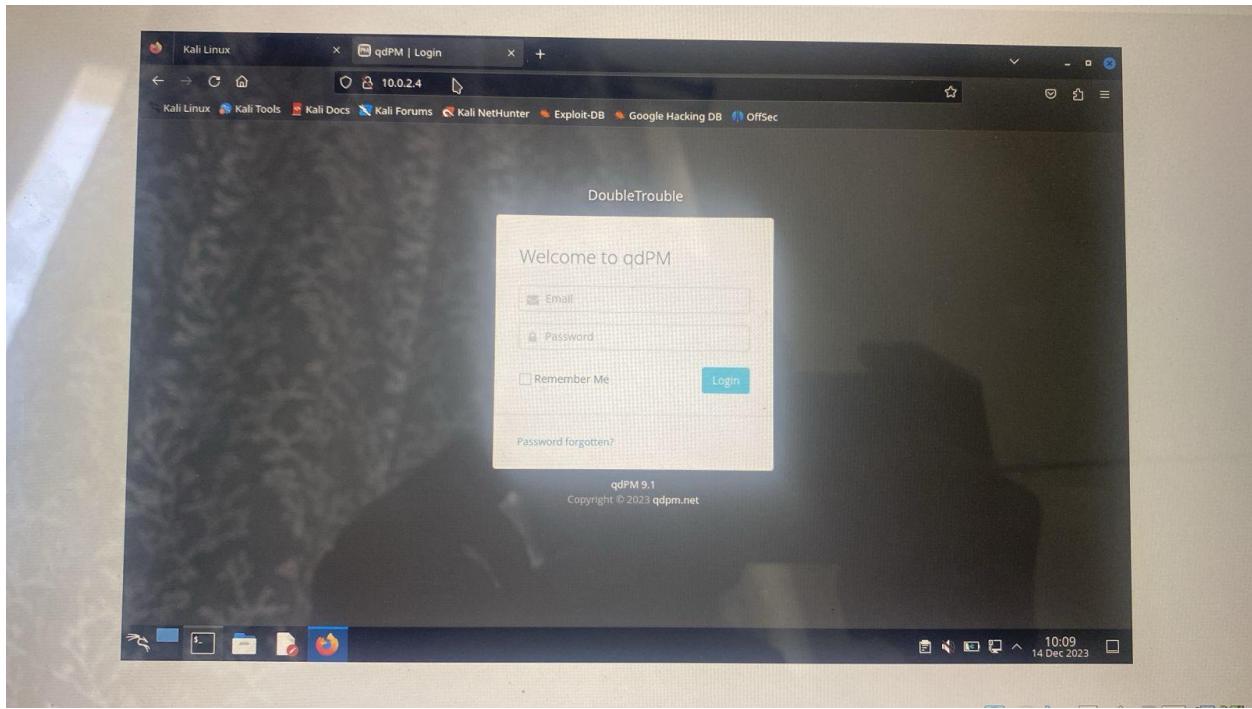
By this nmap port scan we can know that there are 2 open ports one is 80 which is a http port and second is 22 port which is ssh port which is a security port

Next step is

### Step 3: IDENTIFYING VULNERABILITIES IN RUNNING WEB PAGE

Where the port 80 is the http port so open browser in the kali and search about the ip address When i search the <http://10.0.2.4/> i get a welcome to qdpm login page

And it has the vulnerability that it is showing error in the username when we enter the wrong user

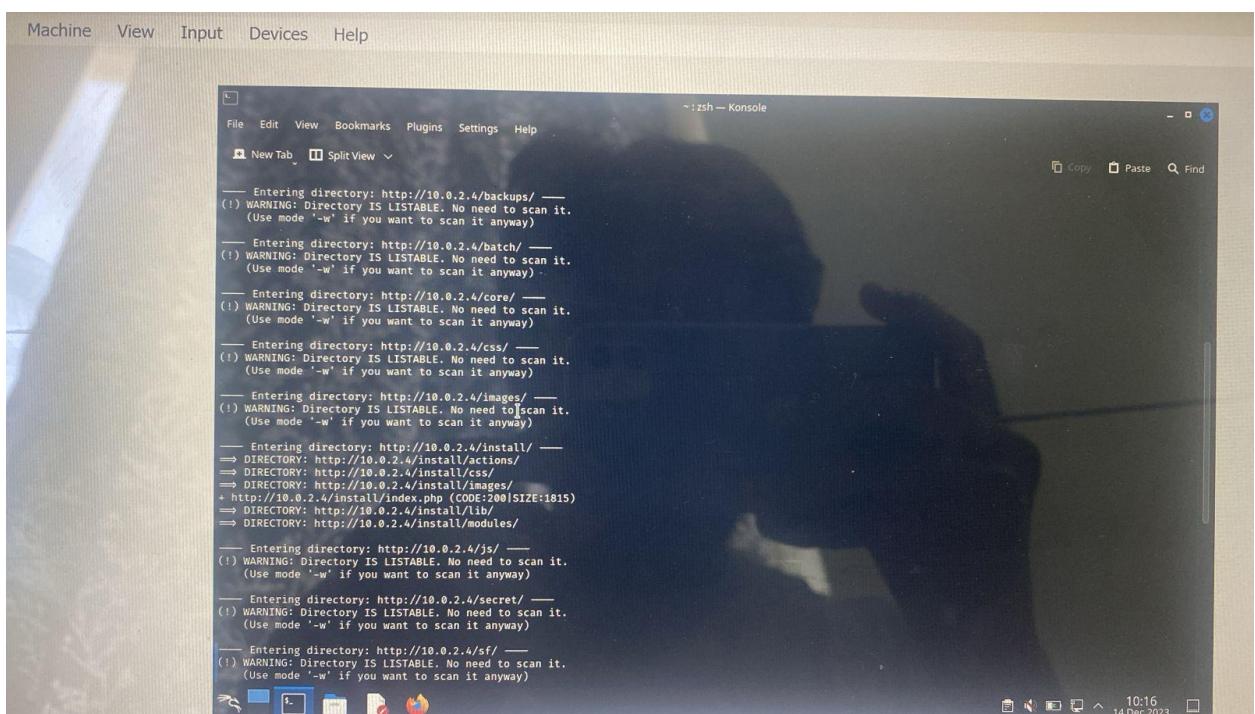
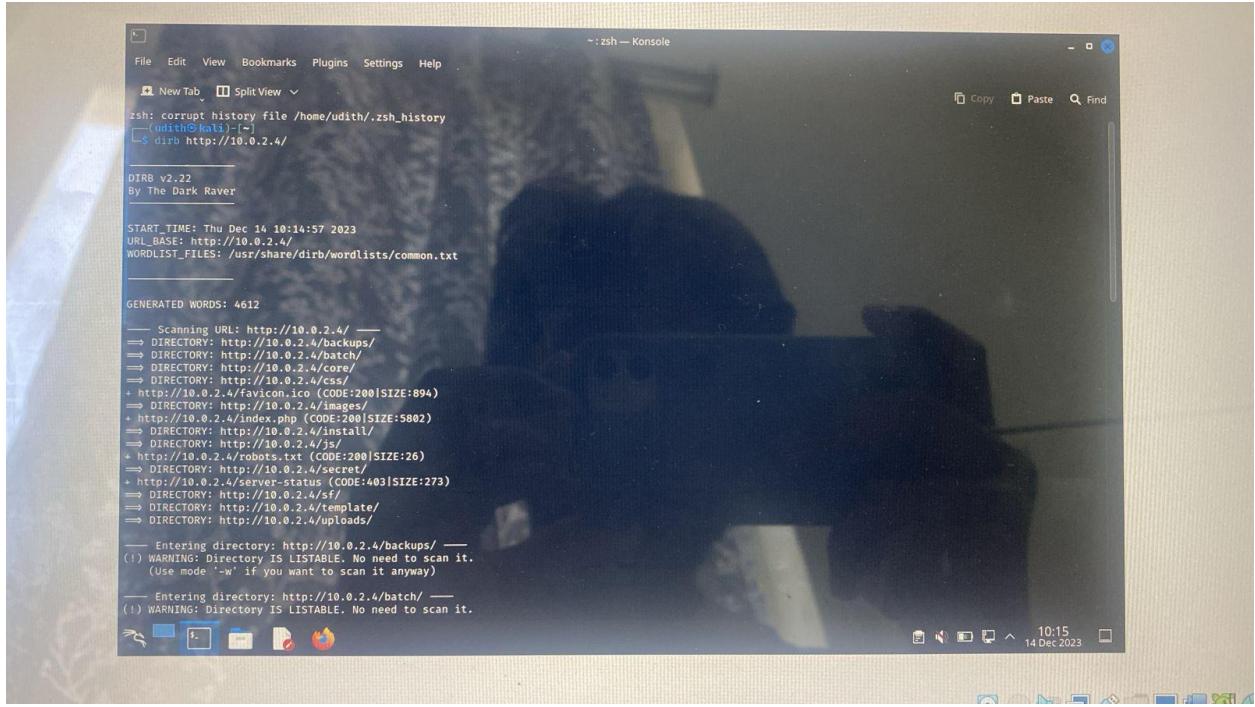


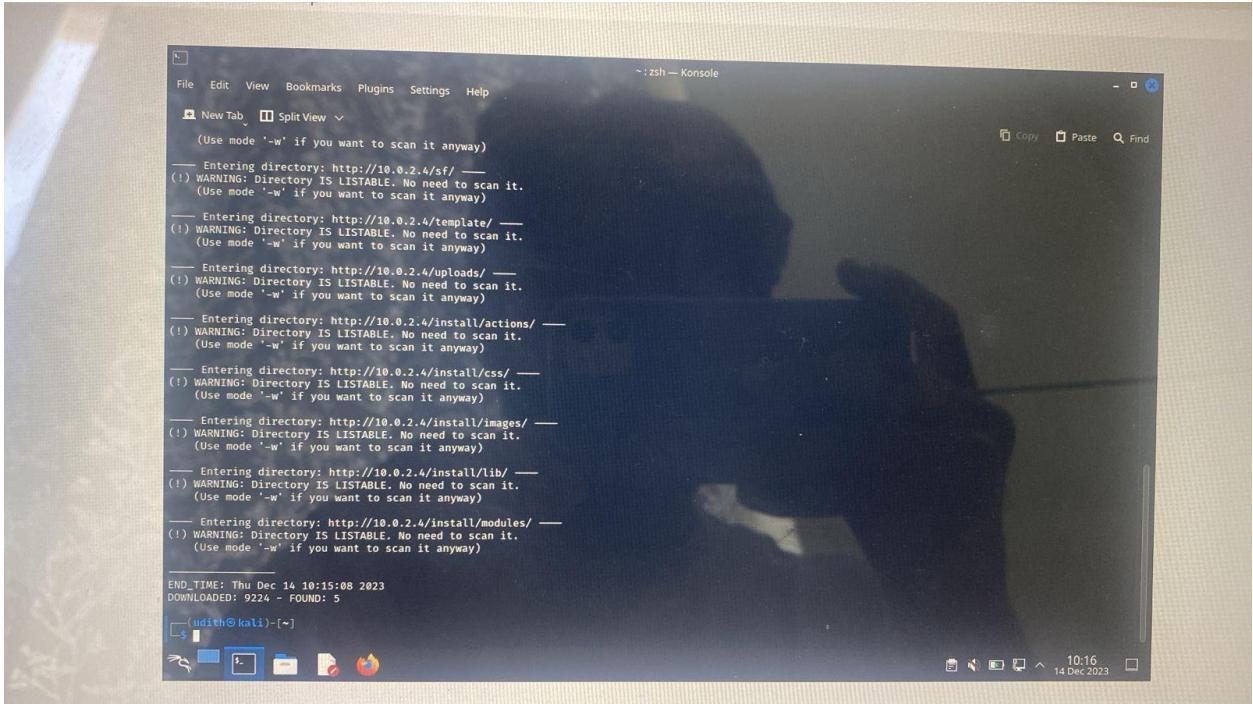
And the next step is

#### Step 4: ENUMERATING APPLICATION WITH DIRB UTILITY

It is directory brute force method which helps to get the find hidden or unprotected directories and files on web server

Execute the command “dirb <http://10.0.2.4/>”





A screenshot of a Kali Linux desktop environment showing a terminal window titled "zsh — Konsole". The terminal displays the output of a directory scan command, likely "dirbuster" or "/dirbuster", against a target at "http://10.0.2.4". The output shows the program entering various directories and issuing warnings about them being listable. At the bottom of the terminal, it shows the end time as "Thu Dec 14 10:15:08 2023" and the total download count as "DOWNLOADED: 9224 - FOUND: 5". The terminal prompt ends with a question mark "?".

```
(Use mode '-w' if you want to scan it anyway)
--- Entering directory: http://10.0.2.4/sf/
(!) WARNING: Directory IS LISTABLE. No need to scan it.
(Use mode '-w' if you want to scan it anyway)

--- Entering directory: http://10.0.2.4/template/
(!) WARNING: Directory IS LISTABLE. No need to scan it.
(Use mode '-w' if you want to scan it anyway)

--- Entering directory: http://10.0.2.4/upload/
(!) WARNING: Directory IS LISTABLE. No need to scan it.
(Use mode '-w' if you want to scan it anyway)

--- Entering directory: http://10.0.2.4/install/actions/
(!) WARNING: Directory IS LISTABLE. No need to scan it.
(Use mode '-w' if you want to scan it anyway)

--- Entering directory: http://10.0.2.4/install/css/
(!) WARNING: Directory IS LISTABLE. No need to scan it.
(Use mode '-w' if you want to scan it anyway)

--- Entering directory: http://10.0.2.4/install/images/
(!) WARNING: Directory IS LISTABLE. No need to scan it.
(Use mode '-w' if you want to scan it anyway)

--- Entering directory: http://10.0.2.4/install/lib/
(!) WARNING: Directory IS LISTABLE. No need to scan it.
(Use mode '-w' if you want to scan it anyway)

--- Entering directory: http://10.0.2.4/install/modules/
(!) WARNING: Directory IS LISTABLE. No need to scan it.
(Use mode '-w' if you want to scan it anyway)

END_TIME: Thu Dec 14 10:15:08 2023
DOWNLOADED: 9224 - FOUND: 5
[udith@kali:~] $
```

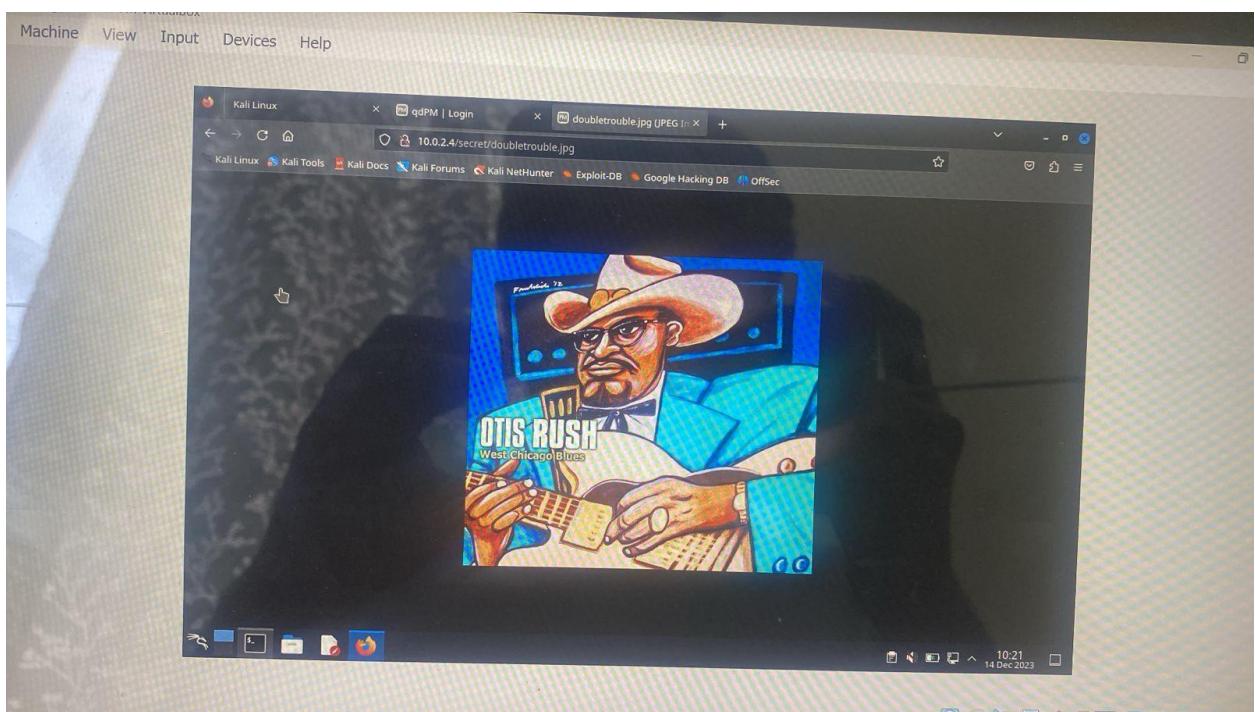
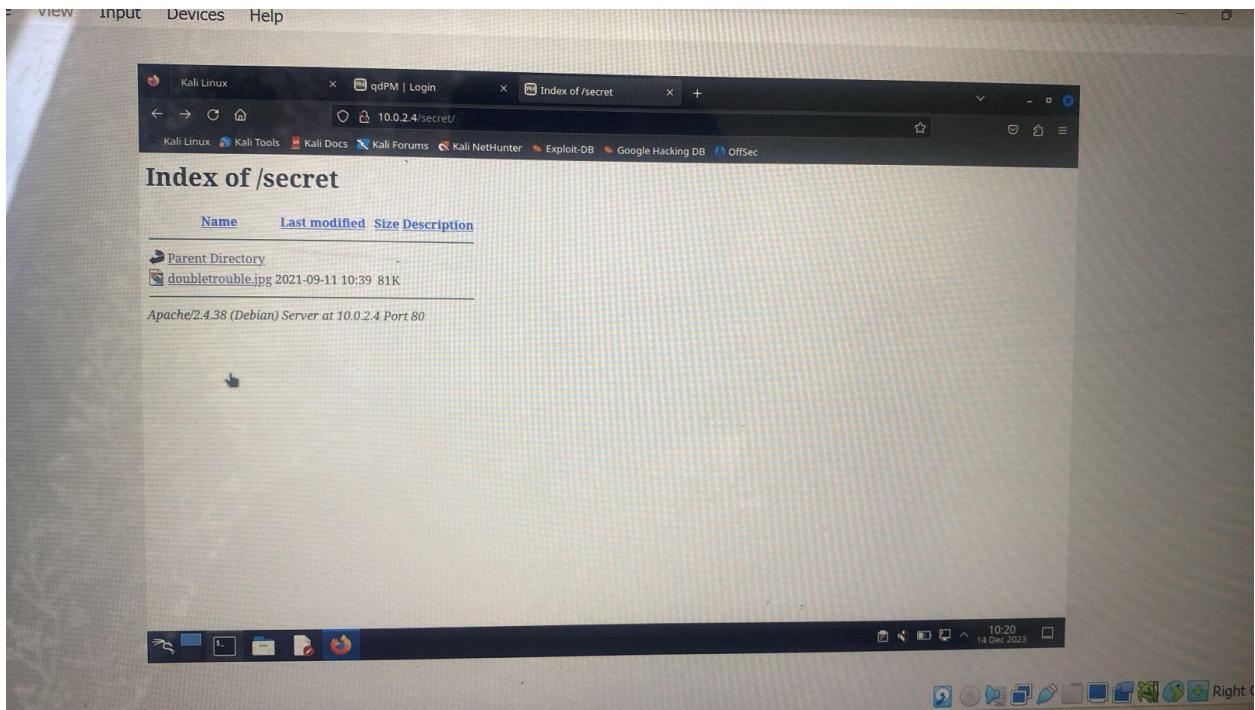
After getting the hidden or unprotected directories

We explore the website and directories in the browser and find the important information which will help to login

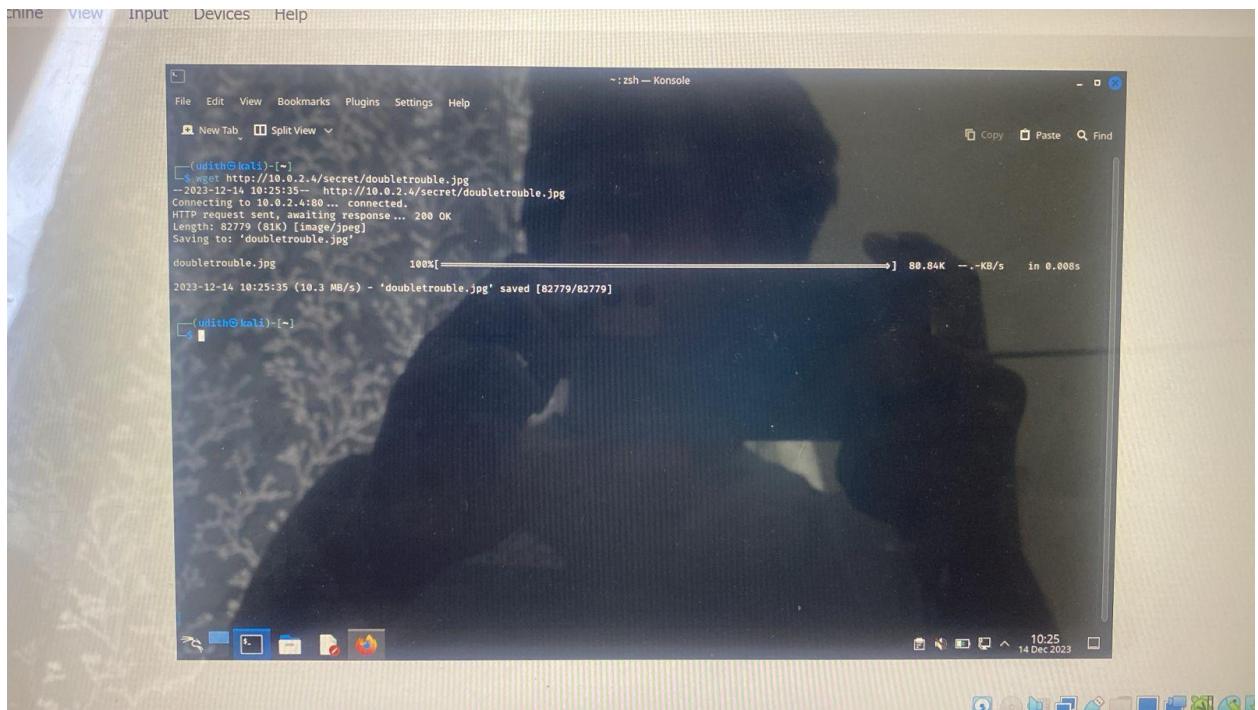
There is a website interesting directory that <http://10.0.2.4/secret/>

Open this directory in the browser

In this directory we found a interesting image which is named with the machine name called doubletrouble.jpg



Download the jpg in the kali by executing the “wget <http://10.0.2.4/secret/doubletrouble.jpg>”



Downloaded in the kali  
And the next step is to  
For extracting the information from the file we use  
Execute command “steghide –extract -sf doubletrouble.jpg”  
Steghide is a steganography program that is able to hide data in various kinds of  
image- and audio-files.  
After executing the command  
It is asking passphrase

```
(uid@kali)-[~]
└─$ curl http://10.0.2.4/secret/doubletrouble.jpg
--2023-12-14 10:25:39-- https://10.0.2.4/secret/doubletrouble.jpg
Connecting to 10.0.2.4:443... connected.
HTTP request sent, awaiting response ... 200 OK
Length: 82779 (81K) [image/jpeg]
Saving to: 'doubletrouble.jpg'

doubletrouble.jpg          100%[=====] 80.84K   --.-KB/s   in 0.008s

2023-12-14 10:25:39 (10.3 MB/s) - 'doubletrouble.jpg' saved [82779/82779]

(uid@kali)-[~]
└─$ steghide --extract -sf doubletrouble.jpg
Enter passphrase: [REDACTED]
```

The next step is to crack the passphrase  
We use stegcracker tool to get the passphrase

**stegcracker** is steganography brute-force utility to uncover hidden data inside files  
Install the stegcracker by using the command “stegcracker –help”  
After installing it will show like this

```
(uid@kali)-[~]
└─$ stegcracker --help
StegCracker 2.1.0 - (https://github.com/Paradoxis/StegCracker)
Copyright (c) 2023 - Luke Paris (Paradoxis)

StegCracker has been retired following the release of StegSeek, which
will blast through the rockyou.txt wordlist within 1.9 second as opposed
to StegCracker which takes ~5 hours.

StegSeek can be found at: https://github.com/RickdeJager/stegseek
usage: stegcracker <file> [<wordlist>]

Steganography brute-force utility to uncover hidden data inside files

positional arguments:
  file
    Input file you think contains hidden information and wish to crack. Note: Stegcracker only accepts the following file types: jpg, jpeg, bmp, wav,
    au

  wordlist
    Wordlist containing the one or more passwords (one password per line). If no password list is supplied, this will default to the rockyou.txt
    wordlist on Kali Linux.

options:
  -h, --help
    Show this help message and exit

  -o OUTPUT, --output OUTPUT
    Output file location, this will be the file the data will be written to on a successful cracked password. If no output location is specified, the
    default location will be the same filename with ".out" appended to the name.

  -t THREADS, --threads THREADS
    Number of concurrent threads used to crack passwords with, increasing this number might lead to better performance. Default: 16

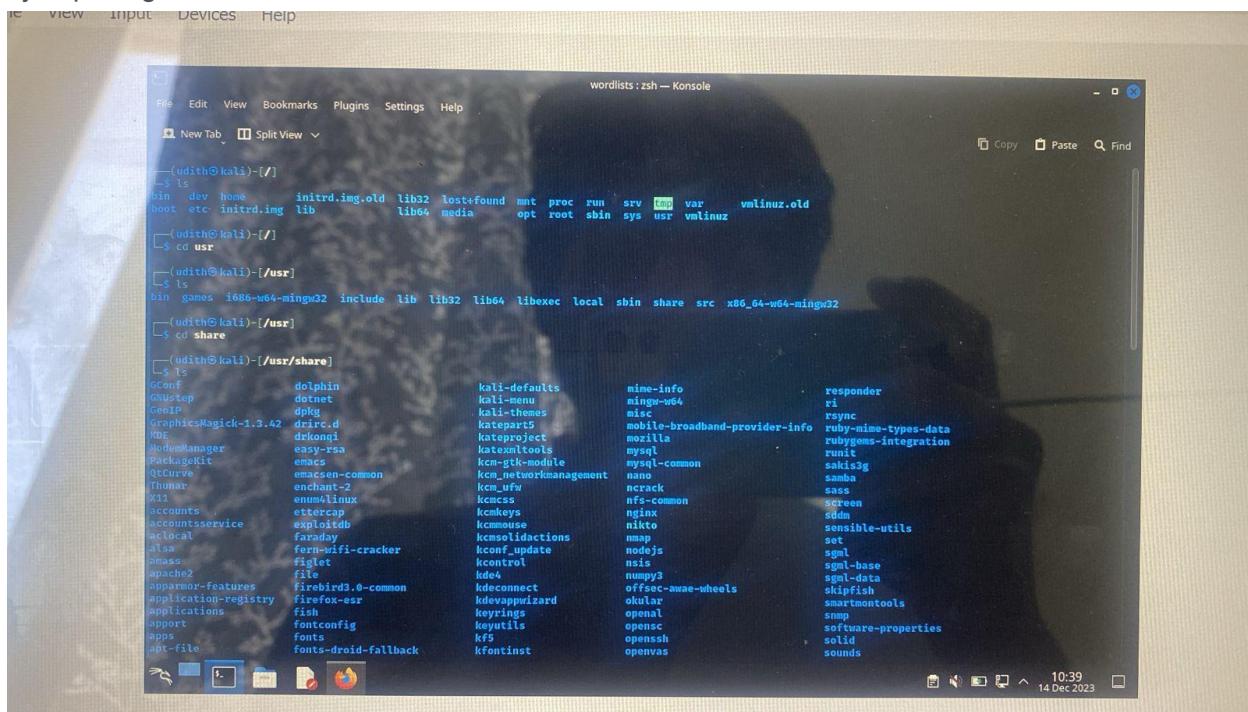
  -c CHUNK_SIZE, --chunk-size CHUNK_SIZE
    Number of passwords loaded into memory per thread cycle. After each password of the chunk has been depleted a status update will be printed to the
    console with the attempted password. Default: 64
```

The next step is

STEP 5:cracking the password with stegcracker

Find the default wordlists in the kali

By exploring the files in the kali

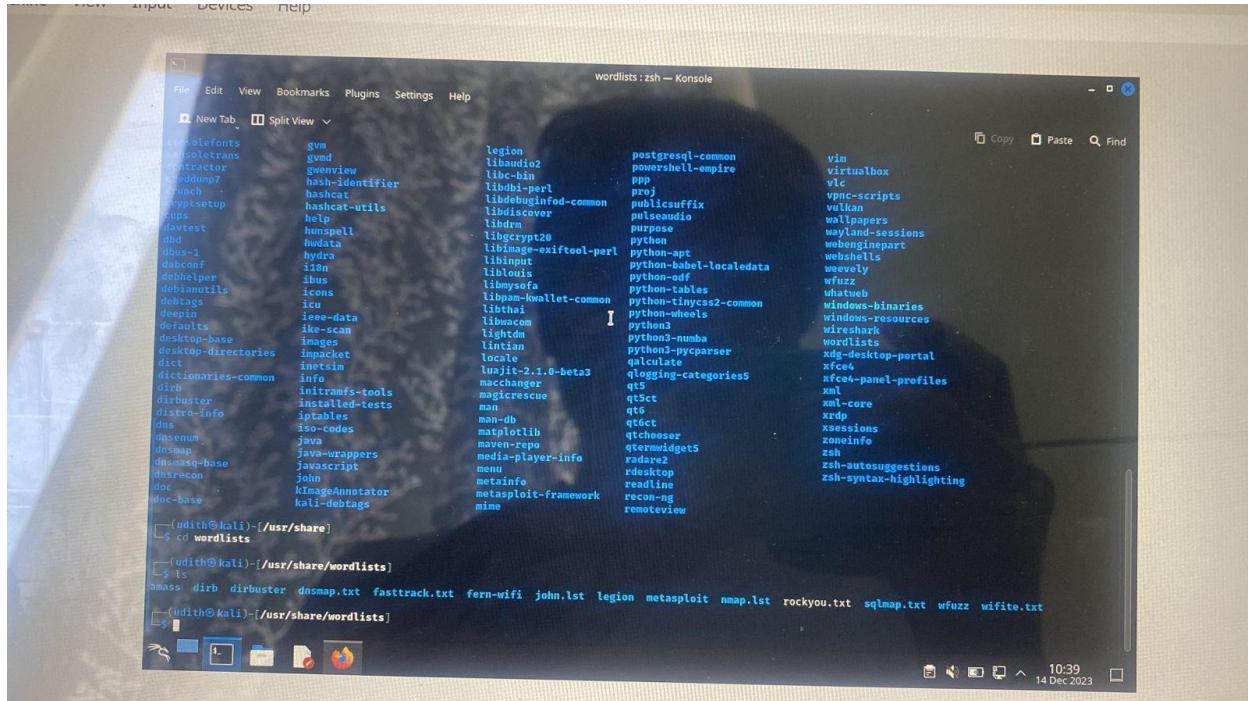


The terminal window shows the user navigating through the file system to find wordlists. The command history includes:

```
--(udith㉿kali)-[~]
$ ls
bin dev home  initrd.img.old  lib32  lost+found  mnt  proc  run  srv  tmp  var  vmlinuz.old
/boot etc  initrd.img  lib  lib64  media  opt  root  sbin  sys  usr  vmlinuz.old
[~] $ cd usr
[~] $ ls
bin games i686-w64-mingw32  include  lib  lib32  lib64  libexec  local  sbin  share  src  x86_64-w64-mingw32
[~] $ cd share
[~] $ ls
[~] $ ls /usr/share
[~] $ ls /usr/share/wordlists
```

The terminal window shows the user navigating through the file system to find wordlists. The command history includes:

```
--(udith㉿kali)-[~]
$ ls
bin dev home  initrd.img.old  lib32  lost+found  mnt  proc  run  srv  tmp  var  vmlinuz.old
/boot etc  initrd.img  lib  lib64  media  opt  root  sbin  sys  usr  vmlinuz.old
[~] $ cd usr
[~] $ ls
bin games i686-w64-mingw32  include  lib  lib32  lib64  libexec  local  sbin  share  src  x86_64-w64-mingw32
[~] $ cd share
[~] $ ls
[~] $ ls /usr/share
[~] $ ls /usr/share/wordlists
```



The terminal window shows the user navigating through the file system to find wordlists. The command history includes:

```
--(udith㉿kali)-[~]
$ ls
bin dev home  initrd.img.old  lib32  lost+found  mnt  proc  run  srv  tmp  var  vmlinuz.old
/boot etc  initrd.img  lib  lib64  media  opt  root  sbin  sys  usr  vmlinuz.old
[~] $ cd usr
[~] $ ls
bin games i686-w64-mingw32  include  lib  lib32  lib64  libexec  local  sbin  share  src  x86_64-w64-mingw32
[~] $ cd share
[~] $ ls
[~] $ ls /usr/share
[~] $ ls /usr/share/wordlists
```

The terminal window shows the user navigating through the file system to find wordlists. The command history includes:

```
--(udith㉿kali)-[~]
$ ls
bin dev home  initrd.img.old  lib32  lost+found  mnt  proc  run  srv  tmp  var  vmlinuz.old
/boot etc  initrd.img  lib  lib64  media  opt  root  sbin  sys  usr  vmlinuz.old
[~] $ cd usr
[~] $ ls
bin games i686-w64-mingw32  include  lib  lib32  lib64  libexec  local  sbin  share  src  x86_64-w64-mingw32
[~] $ cd share
[~] $ ls
[~] $ ls /usr/share
[~] $ ls /usr/share/wordlists
```

We find a default wordlist which is rockyou.txt

Check the words in the file by executing "wc -l rockyou.txt"

A screenshot of a Kali Linux desktop environment. The terminal window shows a list of files from the wordlists directory, including various tools like hydra, nmap, and john. The desktop interface includes a taskbar with icons for file explorer, terminal, and browser, and a system tray with network and battery status.

```
root@kali:~# cd /usr/share/wordlists
root@kali:/usr/share/wordlists# ls
amass  dirb  dirbuster  dnsmap.txt  fasttrack.txt  fern-wifi  john.lst  legion  metasploit  nmap.lst  rockyou.txt  sqlmap.txt  wfuzz  wifite.txt
root@kali:/usr/share/wordlists# wc -l rockyou.txt
14344392 rockyou.txt
root@kali:/usr/share/wordlists#
```

There are 14344392 words

To crack the passphrase we use stegcracker to crack

Go to the directory of the doubletrouble.jpg and

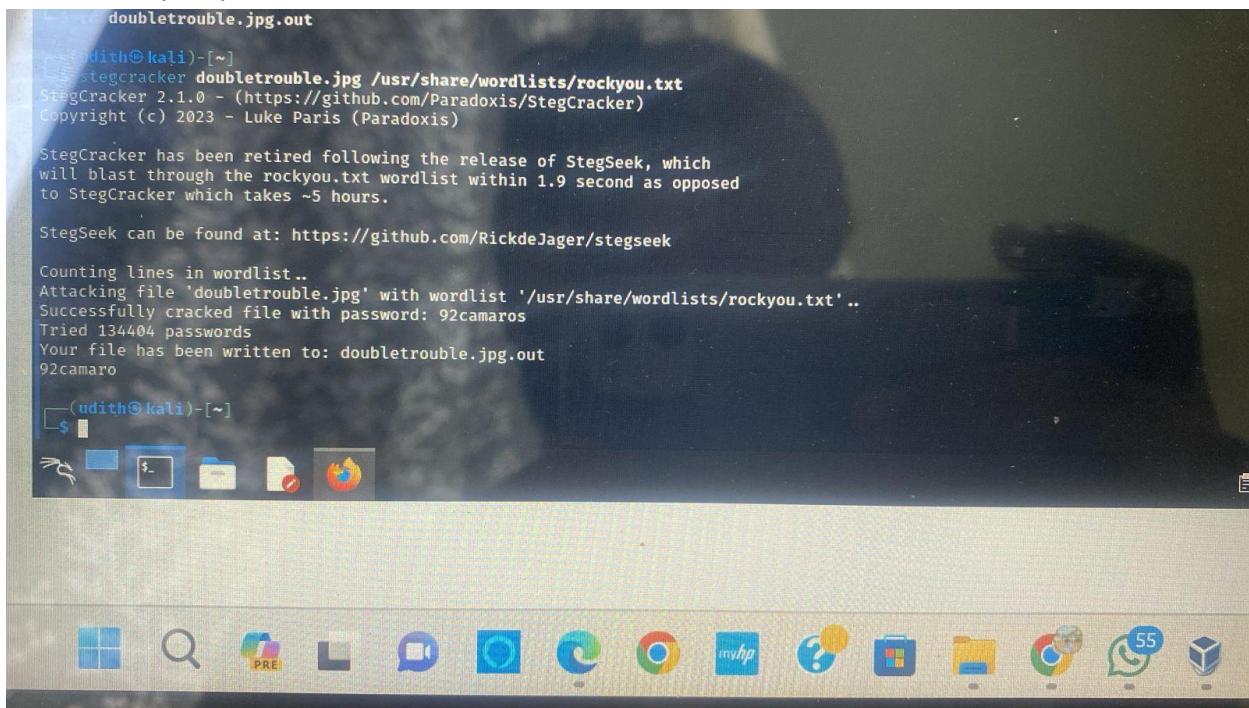
A screenshot of a Kali Linux desktop environment. The terminal window shows the user navigating through the file system, starting from the root directory and moving into the home directory of the user 'udith'. The desktop interface includes a taskbar with icons for file explorer, terminal, and browser, and a system tray with network and battery status.

```
root@kali:~# cd ..
root@kali:/# cd ..
root@kali:/# cd ..
root@kali:/# ls
bin  dev  home  initrd.img.old  lib32  lost+found  mnt  proc  run  srv  tmp  var  vmlinuz.old
boot  etc  initrd.img  lib  lib64  media  opt  root  sbin  sys  usr  vmlinuz
root@kali:/# cd ..
root@kali:/# ls
bin  dev  home  initrd.img.old  lib32  lost+found  mnt  proc  run  srv  tmp  var  vmlinuz.old
boot  etc  initrd.img  lib  lib64  media  opt  root  sbin  sys  usr  vmlinuz
root@kali:/# cd ..
root@kali:/# cd home
root@kali:/home# ls
udith
root@kali:/home# cd udith
root@kali:/home/udith# ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos  doubletrouble.jpg  doubletrouble.jpg.out  readme.txt  rockyou.txt
root@kali:/home/udith#
```

After reaching to the doubletrouble.jpg directory

Execute the command “stegcracker doubletrouble.jpg /usr/share/wordlists/rockyou.txt”

To crack the passphrase



The screenshot shows a Kali Linux desktop environment. At the top, there is a terminal window with the following text:

```
doubletrouble.jpg.out
[udith@kali)-[~]
$ stegcracker doubletrouble.jpg /usr/share/wordlists/rockyou.txt
StegCracker 2.1.0 - (https://github.com/Paradoxis/StegCracker)
Copyright (c) 2023 - Luke Paris (Paradoxis)

StegCracker has been retired following the release of StegSeek, which
will blast through the rockyou.txt wordlist within 1.9 second as opposed
to StegCracker which takes ~5 hours.

StegSeek can be found at: https://github.com/RickdeJager/stegseek

Counting lines in wordlist..
Attacking file 'doubletrouble.jpg' with wordlist '/usr/share/wordlists/rockyou.txt' ..
Successfully cracked file with password: 92camaro
Tried 134404 passwords
Your file has been written to: doubletrouble.jpg.out
92camaro

[udith@kali)-[~]
$
```

Below the terminal, the desktop environment includes a taskbar with various icons such as File Explorer, Task View, Start button, Search, Microsoft Edge, Google Chrome, File Explorer, and others.

after the cracking we get the passphrase as 92camaro

Now extract the data from the image by again executing the command “**steghide –extract -sf doubletrouble.jpg**”

It will ask the passphrase

Enter the passphrase

```
(udith㉿kali)-[~]
$ stegcracker doubletrouble.jpg /usr/share/wordlists/rockyou.txt
StegCracker 2.1.0 - (https://github.com/Paradoxis/StegCracker)
Copyright (c) 2023 - Luke Paris (Paradoxis)

StegCracker has been retired following the release of StegSeek, which
will blast through the rockyou.txt wordlist within 1.9 second as opposed
to StegCracker which takes ~5 hours.

StegSeek can be found at: https://github.com/RickdeJager/stegseek

Counting lines in wordlist..
Attacking file 'doubletrouble.jpg' with wordlist '/usr/share/wordlists/rockyou.txt'..
Successfully cracked file with password: 92camaros
Tried 134404 passwords
Your file has been written to: doubletrouble.jpg.out
92camaro

(udith㉿kali)-[~]
$ steghide --extract -sf doubletrouble.jpg
Enter passphrase:
wrote extracted data to "creds.txt".

(udith㉿kali)-[~]
$ cat creds.txt
```

It save the extracted file ass creds.txt

Open the file by "cat creds.txt"

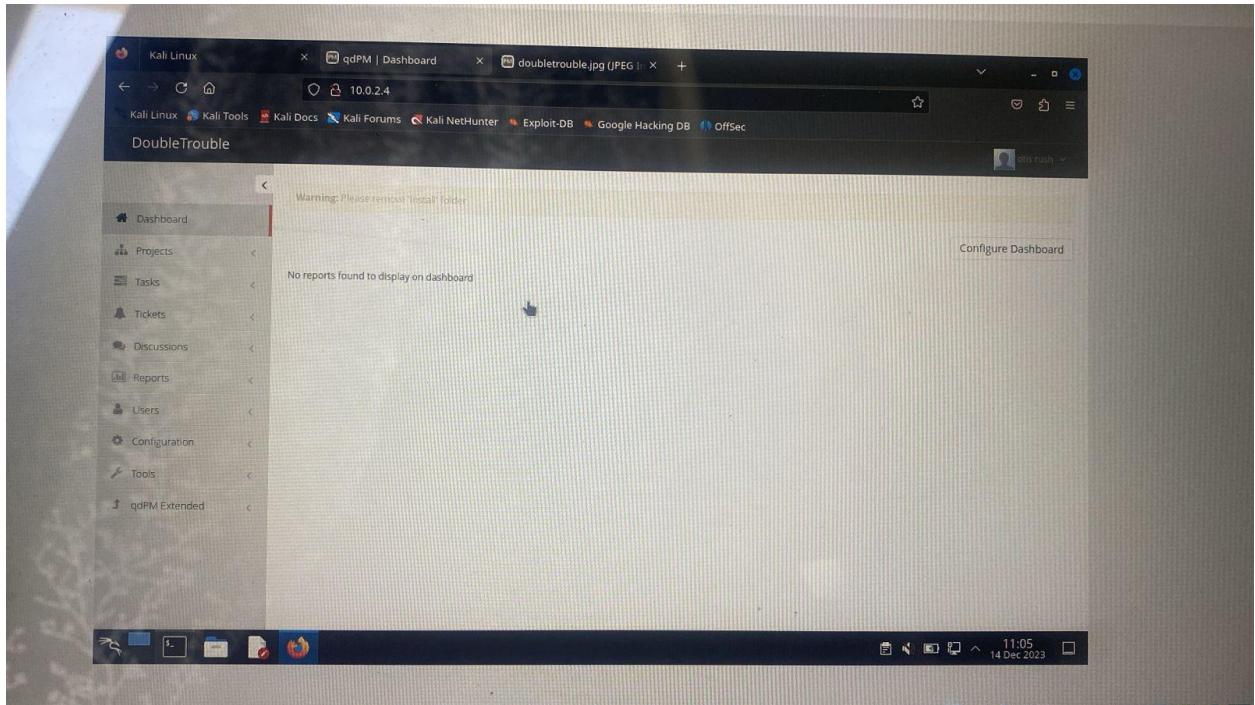
```
(udith㉿kali)-[~]
$ steghide --extract -sf doubletrouble.jpg
Enter passphrase:
wrote extracted data to "creds.txt".

(udith㉿kali)-[~]
$ cat creds.txt
otisrush@localhost.com
otis666

(udith㉿kali)-[~]
$
```

After reading the creds.txt file we get the credentials for the qdpm login

Enter the credentials in the login



By this step we logged in to the qdpm

The credentials are

User [gmail:otisrush@localhost.com](mailto:otisrush@localhost.com)

Password:otis666

By this we enter the machine and we can reshell the program and get the connectivity also  
THIS ARE THE STEPS FOR CTF OF DOUBLETROUBLE MACHINE