First i enter my current IP address by following this command:

nmap -sV 10.8.130.0/24

Flag 1

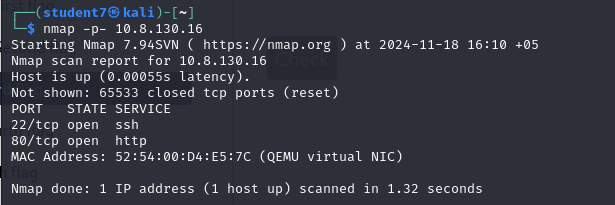
For scanning all ports i’ll enter nmap -p- 10.8.130.16

By

using “-p-”, nmap scans all 65535 ports.This tool is used to determine available

services, perform security checks, identify vulnerabilities, and analyze network

traffic.



Then in the browser i enter ip address 10.8.130.16 for capturing first flag

A screenshot of a computer

Description automatically generated

After i save it as “iitu.jpg”

For capturing flag i enter these commands:

Cd Downloads

Exiftool iitu.jpg

And finally i got flag:

A screen shot of a computer

Description automatically generated

I got our first flag,that was in the comment. It may not be anything

significant, but the metadata of the image itself may be significant.



Flag 2:

**Scan for Hidden Files and Vulnerabilities:** Use nikto to analyze the web server:

Nikto -host 10.8.130.16 port 80

-h: Specifies the target host.

nikto - open-source web vulnerability scanner designed to identify potential

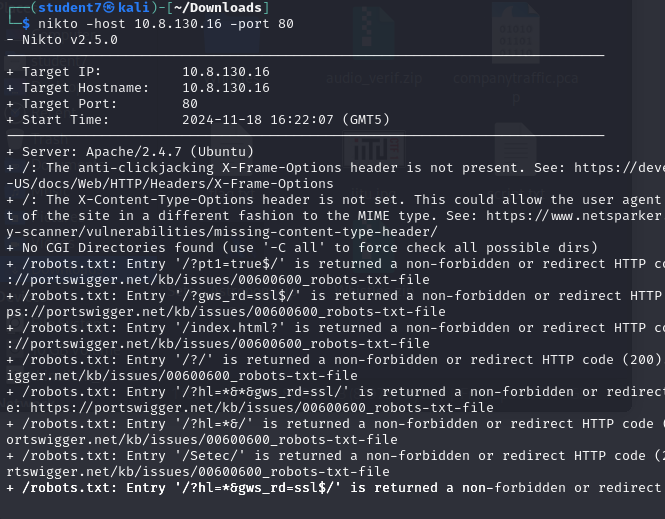
security issues in web servers. It&#39;s written in Perl and uses LibWhisker for fast

security checks. Nikto can identify a wide range of potential security issues, including

outdated software, server configuration problems, and common vulnerabilities like

SQL injection and cross-site scripting (XSS)

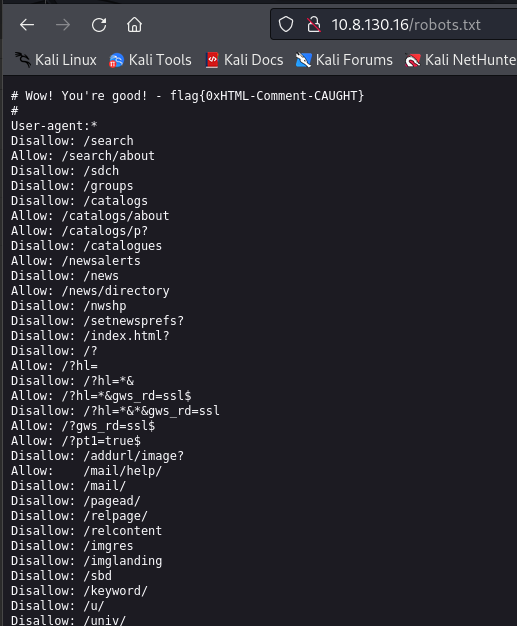




By executing it, now i can see other URLs available for us. I’ll

check out robots.txt URL to have more clues about next steps.

I’ll enter 10.8.130.16/robots.txt



As i can see, the first line of the robots.txt contains a flag, marked

as a comment. This could also be a simple flag, but sometimes in

production practice, web developers on the team may leave some

notes about the code and what might be missing.

I’ll enter it imto vite and here is success:

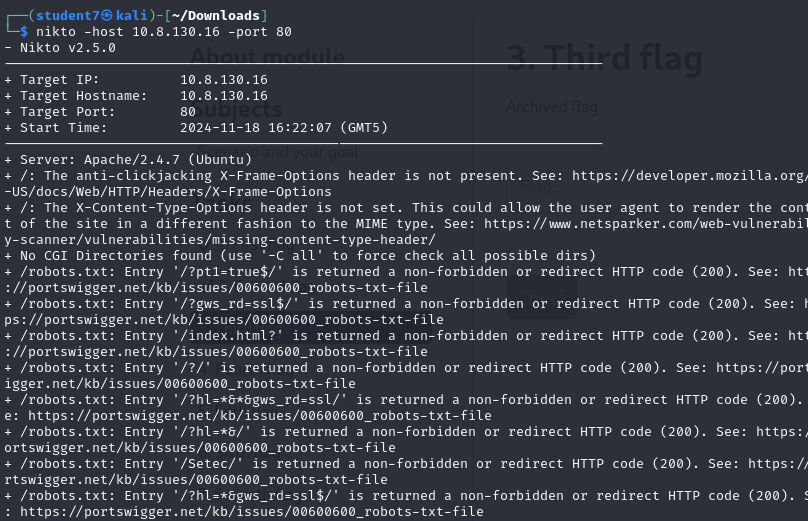
A screenshot of a computer

Description automatically generated

3. Flag 3

To get to the next step, we should remember what is the function of

“nikto” in Flag 2,

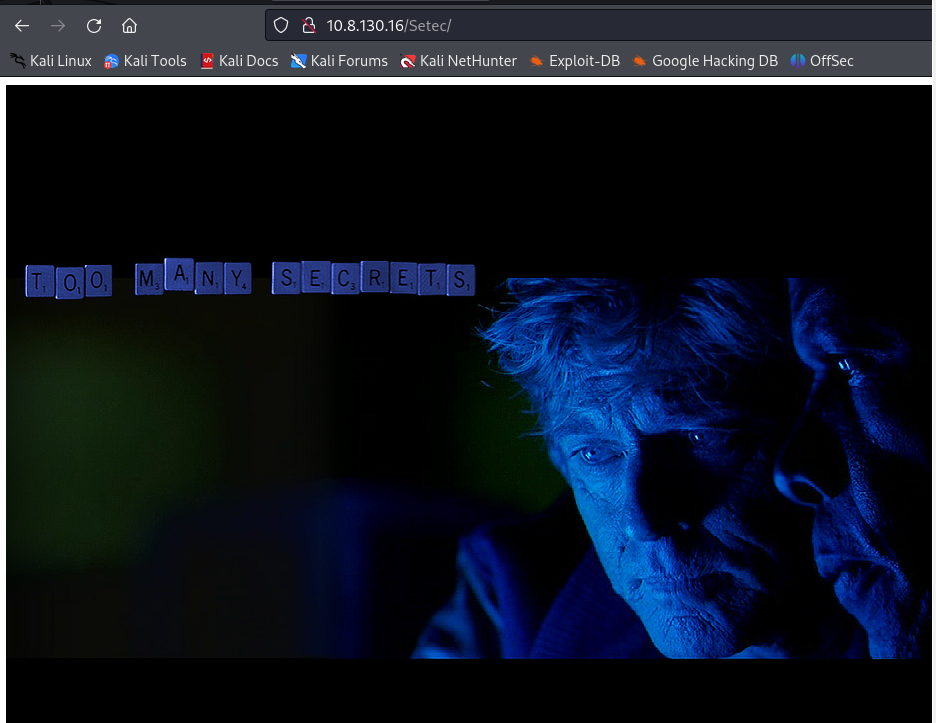


When Nikto scans the robots.txt file, it looks for entries that might

indicate sensitive or restricted areas of the website.

Now i should check every url for accessibility. Starting from Setec.

Then i’ll enter 10.8.130.16/setec/ and i saw:

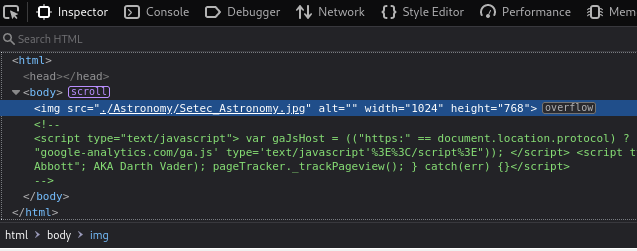


As i don;t know where to go, maybe we should check the directory

where it is saved. Press Q or Right Mouse Click (RMC) and then

Inspect. As downloading and executing “exiftool” didn’t give us any

useful information.



Now i can see the directory check that path where could be

contained more files that we could abuse.

A screenshot of a computer

Description automatically generated

In this directory we have zip files that i have to download.

files, not only image of “too many

secrets”: some zip and text files.Download all of them and try to crack a

zip file, using “fcrackzip” and a big wordlist “rockyou”.

**Note 1: nmap**

nmap (Network Mapper) is a powerful open-source tool used for network exploration and security auditing. It allows scanning of networks to identify active hosts, open ports, and running services.

**Key Flags:**

-sV: Enables service version detection to identify software and versions.

-p-: Scans all 65535 ports instead of just common ones.

-A: Enables OS detection, version detection, script scanning, and traceroute.

Widely used in penetration testing and vulnerability assessments to detect misconfigurations or potential entry points in a network.

**Note 2: exiftool**

exiftool is a command-line tool designed for reading, writing, and editing metadata embedded in various file types, such as images, videos, and documents.

**Common Use Cases:**

Extract hidden comments or information stored in metadata.

Analyze file origins or editing history.

It supports a wide range of formats and is frequently used in digital forensics, CTFs, and penetration testing to uncover concealed information.

**Note 3: nikto**

nikto is an open-source web server scanner that identifies potential security issues in web applications. It scans for:

Outdated server software or plugins.

Default files and directories, such as robots.txt or configuration backups.

Known vulnerabilities like SQL Injection and Cross-Site Scripting (XSS).

**Key Features:**

Written in Perl for fast scanning using LibWhisker.

Displays categorized results (e.g., green for safe, red for critical).  
nikto is often used for quick vulnerability assessments in web security.

**Note 4: base64**

base64 is a command-line utility used for encoding and decoding data in Base64 format.

**Common Applications:**

Decode text hidden in files or scripts.

Handle encoded data transmitted over HTTP or other protocols.

A screen shot of a computer

Description automatically generated

That tool i found the password from this dictionary. Now i can extract

it and see more hidden files

A screenshot of a computer

Description automatically generated

Our first file “flag.txt” contains our 3-rd flag and “SomeInfo.txt” is an

unknown hint for us, as our network is not accessible to the ethernet.

A screen shot of a computer

Description automatically generated

Here is 3rd flag

A screenshot of a computer

Description automatically generated

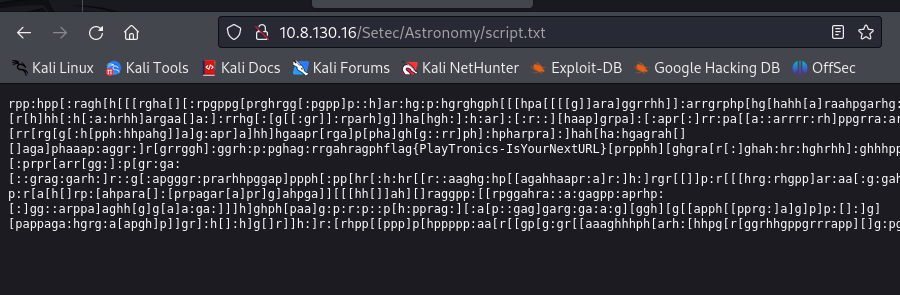
A screenshot of a computer

Description automatically generated

Flag 4:

http://10.8.130.16/Setec/Astronomy/script.txt

Here is easiest step. In same directory we have “script.txt” it contains messy text, here i found flag 4 and hint for the next flag:



Here is flag

A screenshot of a computer

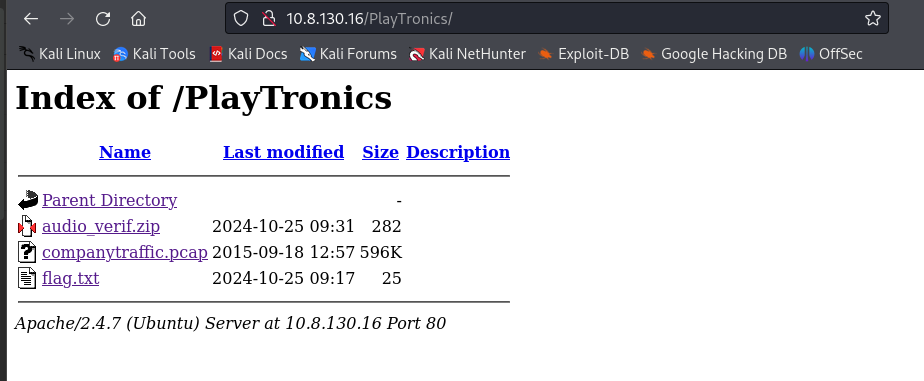
Description automatically generated

**Flag 5**

**Navigate to the URL from Flag 4:**

Use the URL provided in the previous step: http://10.8.130.16/P\*\*\*T\*\*\*\*\*\*.

This directory contains a flag.txt file.



**Retrieve Flag 5:**

Download and open flag.txt:

curl http://10.8.130.16/P\*\*\*T\*\*\*\*\*\*/flag.txt

cat flag.txt

Flag 5 is revealed in this file.

A screenshot of a computer

Description automatically generated

So here is flag

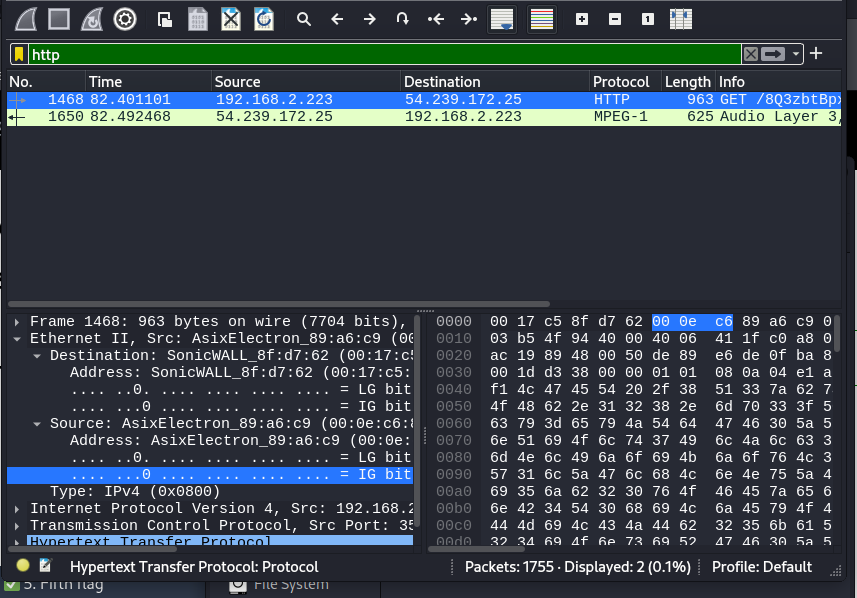
A screenshot of a computer

Description automatically generated

Flag 6

I will download companytrffic.pcap and open it.

I am searching for a specific packet, type in search “http” for any clues.



It’s very suspicious to have only two packets for “http”, but as “MPEG-

1” protocol is an audio format, we could use some parameters as a

password for a secured “audio\_verif.zip” archive.

And i archived it: as a password i took source ip address:

54.239.172.25

A screenshot of a computer

Description automatically generated

Finally i have a flag:

A screenshot of a computer

Description automatically generated

Extract files from “audio\_verif.zip” and observe the content of our

extracted file. As it hints, we can use flag 5 and 6 as credentials for

accessing account

Flag 7

I’ll connect to the same VM using flag 5 as a password and flag 6 as login

Password: myuniqekeystudent

A computer screen shot of a computer code

Description automatically generated

By typing “ls” we can see a list of files in our directory. And by issuing

“cat flag.txt” we are getting our 7-th flag.

A screenshot of a computer

Description automatically generated

A black background with white text

Description automatically generated

So flag was captured

A screenshot of a computer

Description automatically generated

Flag 8

Type “cd / ” to have more directories to observe then type again “ls”

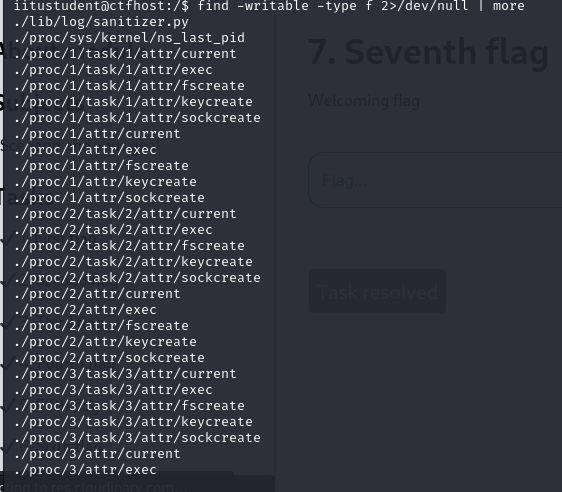
to have a list of folders and files.

A blue text on a black background

Description automatically generated

Issue “find -writable -type f 2&gt;/dev/null | more” to find which files are

writable.



Type “cat lib/log/sanitizer.py” to see what our python code can do. As

it does all for us, we don’t have any need to change that file. Our last

step - execute python code.

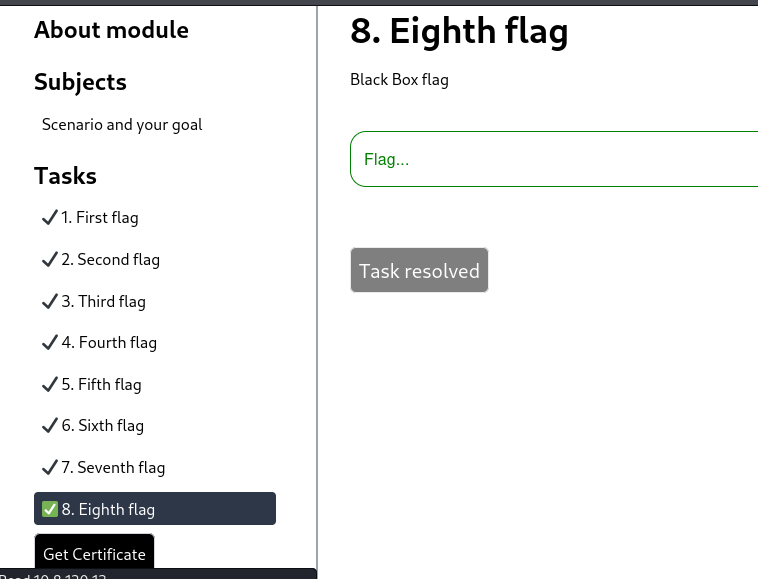
A screen shot of a computer

Description automatically generated

A screenshot of a computer code

Description automatically generated

Here is our flag



A screenshot of a certificate

Description automatically generated

**Conclusion for CTF Part-1 and Part-2**

The completion of CTF Part-1 and Part-2 provided valuable hands-on experience in identifying and exploiting vulnerabilities in a controlled environment. These exercises covered multiple stages of a penetration testing workflow, including reconnaissance, enumeration, exploitation, and post-exploitation.

**Key Learning Points:**

1. **Network Scanning and Enumeration:**

Tools like nmap and nikto were used to discover open ports, services, and web server vulnerabilities, emphasizing the importance of proper service configurations and patch management.

1. **Metadata Analysis and Hidden Files:**

Techniques such as analyzing metadata with exiftool and exploring misconfigured files like robots.txt highlighted how overlooked or improperly secured data can lead to information leakage.

1. **Password Cracking and Decryption:**

Using tools like fcrackzip and wordlists such as rockyou.txt demonstrated the vulnerabilities of weak password policies and the risks of using easily guessable passwords.

1. **Network Traffic Analysis:**

Analyzing .pcap files in Wireshark and identifying source IPs showcased the importance of monitoring and securing network communications.

1. **Post-Exploitation Techniques:**

Exploiting writable files, analyzing Python scripts, and navigating a vulnerable system emphasized the risks of privilege escalation and insecure coding practices.

**Takeaways:**

These exercises reinforced the criticality of proactive cybersecurity measures, such as:

Employing robust password policies and regularly rotating them.

Implementing strict access controls for sensitive directories and files.

Regularly monitoring and encrypting network traffic to prevent data leakage.

Ensuring software and services are updated and secured against known vulnerabilities.

By applying these tools and techniques, participants gained practical knowledge in identifying real-world security threats, understanding attacker methodologies, and improving system defenses. This comprehensive learning approach is invaluable for professionals in cybersecurity and ethical hacking.