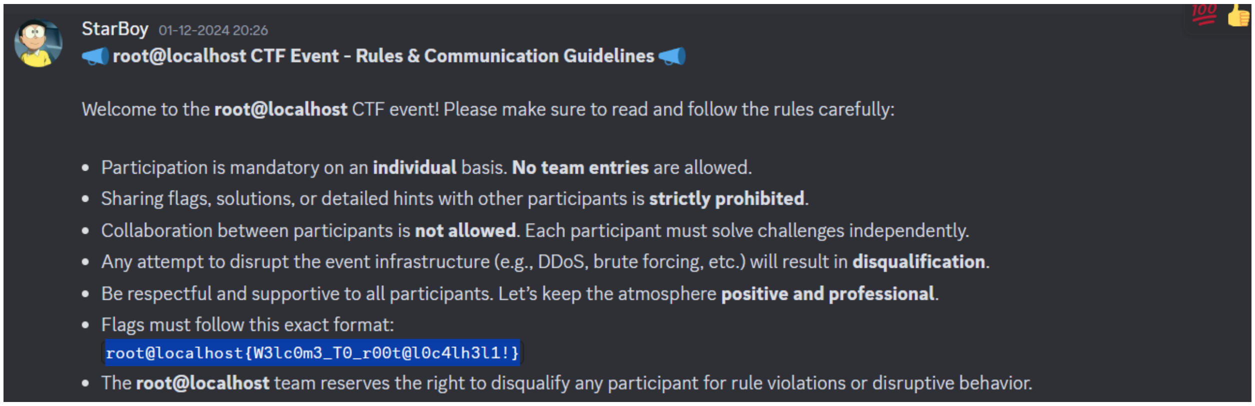
MISC

WELCOME

The very first challenge was this and gave a introduction to the ctf

- Firstly navigated to the announcement channel in the discord server
- Saw the very first pinned message which contained the flag.This made me understand to always check all the communication channels of the ctf.



StarBoy 01-12-2024 20:26

📣 **root@localhost CTF Event - Rules & Communication Guidelines** 📣

Welcome to the **root@localhost** CTF event! Please make sure to read and follow the rules carefully:

- Participation is mandatory on an **individual** basis. **No team entries** are allowed.
- Sharing flags, solutions, or detailed hints with other participants is **strictly prohibited**.
- Collaboration between participants is **not allowed**. Each participant must solve challenges independently.
- Any attempt to disrupt the event infrastructure (e.g., DDoS, brute forcing, etc.) will result in **disqualification**.
- Be respectful and supportive to all participants. Let's keep the atmosphere **positive and professional**.
- Flags must follow this exact format:
  `root@localhost{W3lc0m3_T0_r00t@l0c4lh3l1!}`
- The **root@localhost** team reserves the right to disqualify any participant for rule violations or disruptive behavior.

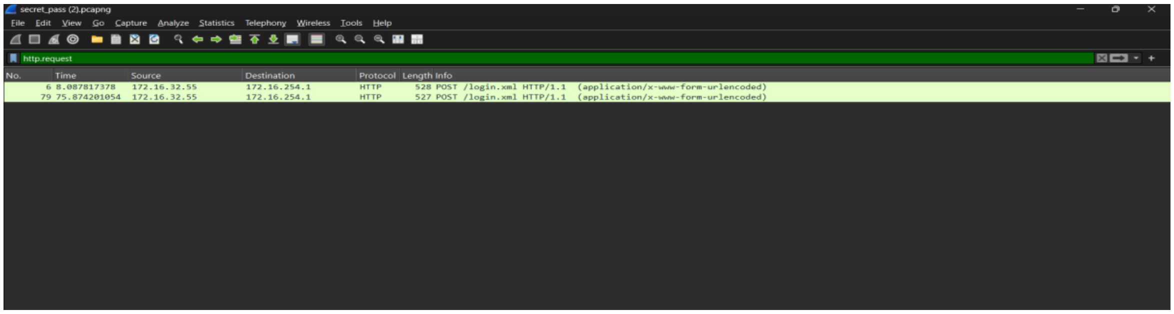
THE GREAT LOGIN HEIST

Description:

A PCAP file was given and asked to be analyzed and so was done by loading it in wireshark tool.

Approach:

- Firstly the file was downloaded and opened it in the wireshark tool
- Then filtered the traffic and displayed by http.request

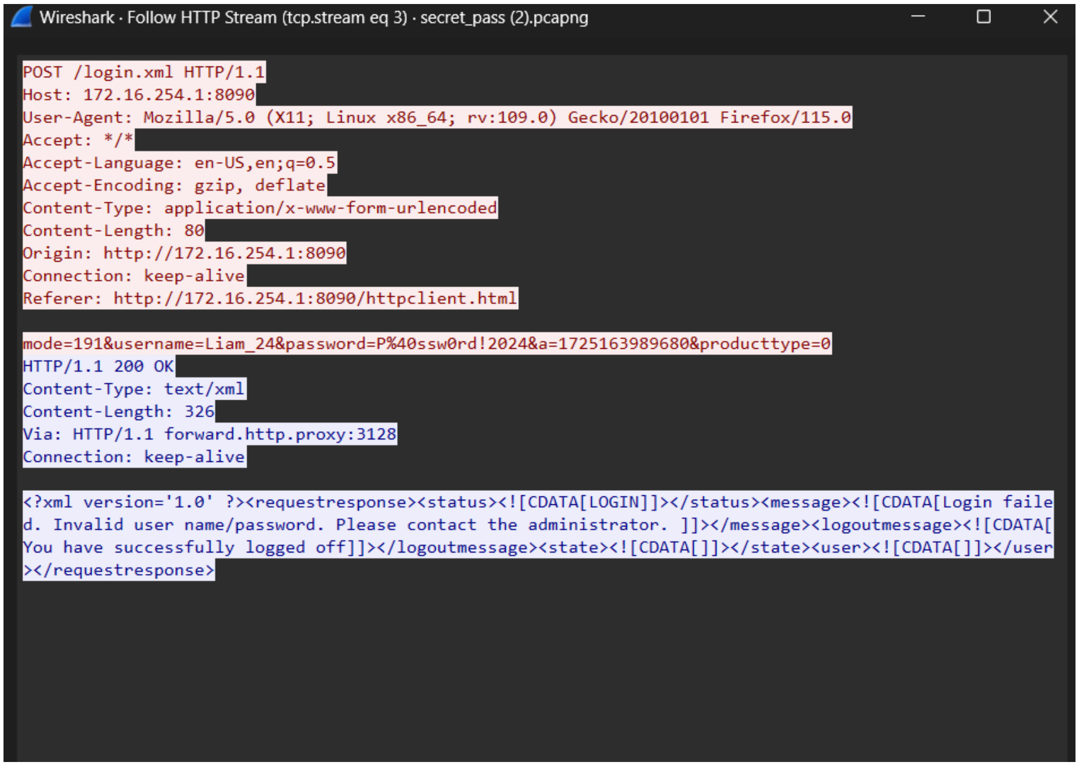


secret_pass (2).pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
http.request

```
No.   Time          Source          Destination     Protocol Length Info
    6 8.087817378   172.16.32.55    172.16.254.1    HTTP       528 POST /login.xml HTTP/1.1  (application/x-www-form-urlencoded)
   79 75.874201054  172.16.32.55    172.16.254.1    HTTP       527 POST /login.xml HTTP/1.1  (application/x-www-form-urlencoded)
```

- Then right lick the login packet - navigated to follow - http stream

- Then inspected the packet details


Wireshark · Follow HTTP Stream (tcp.stream eq 3) · secret_pass (2).pcapng

```
POST /login.xml HTTP/1.1
Host: 172.16.254.1:8090
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 80
Origin: http://172.16.254.1:8090
Connection: keep-alive
Referer: http://172.16.254.1:8090/httpclient.html

mode=191&username=Liam_24&password=P%40ssw0rd!2024&a=1725163989680&producttype=0
HTTP/1.1 200 OK
Content-Type: text/xml
Content-Length: 326
Via: HTTP/1.1 forward.http.proxy:3128
Connection: keep-alive

<?xml version='1.0' ?><requestresponse><status><![CDATA[LOGIN]]></status><message><![CDATA[Login faile
d. Invalid user name/password. Please contact the administrator. ]]></message><logoutmessage><![CDATA[
You have successfully logged off]]></logoutmessage><state><![CDATA[]]></state><user><![CDATA[]]></user
></requestresponse>
```

Then got the username and password from this
username=Liam_24
password=P%40ssw0rd!2024
Then formatted the flag with the credentials and got the flag for this challenge ; root@localhost{Liam_24_P%40ssw0rd!2024}

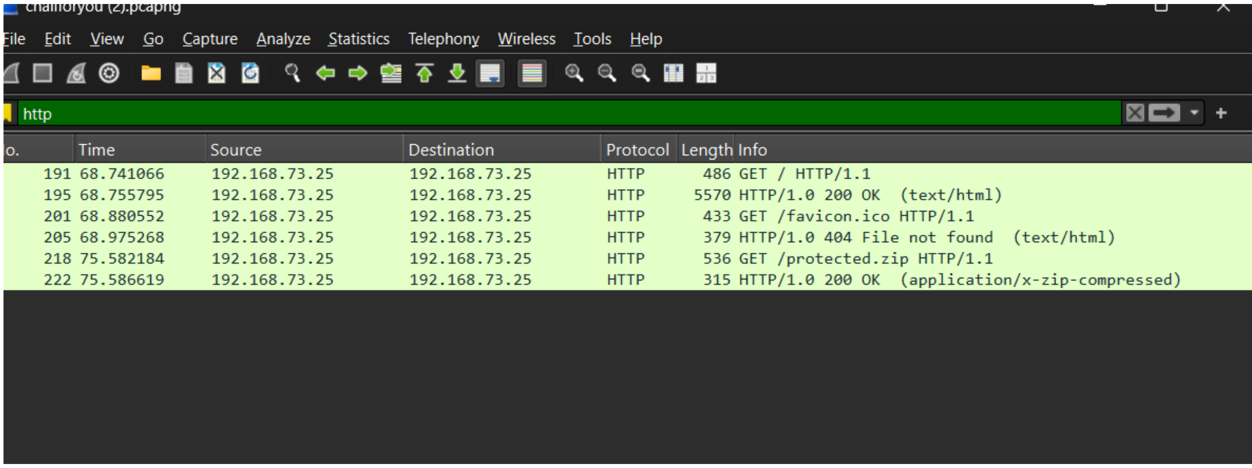
SILENT COURIER

Description:

To analyse the .pcap file and intercept the transfer and uncover the hidden secret of this mysterious file.

Approach and steps:

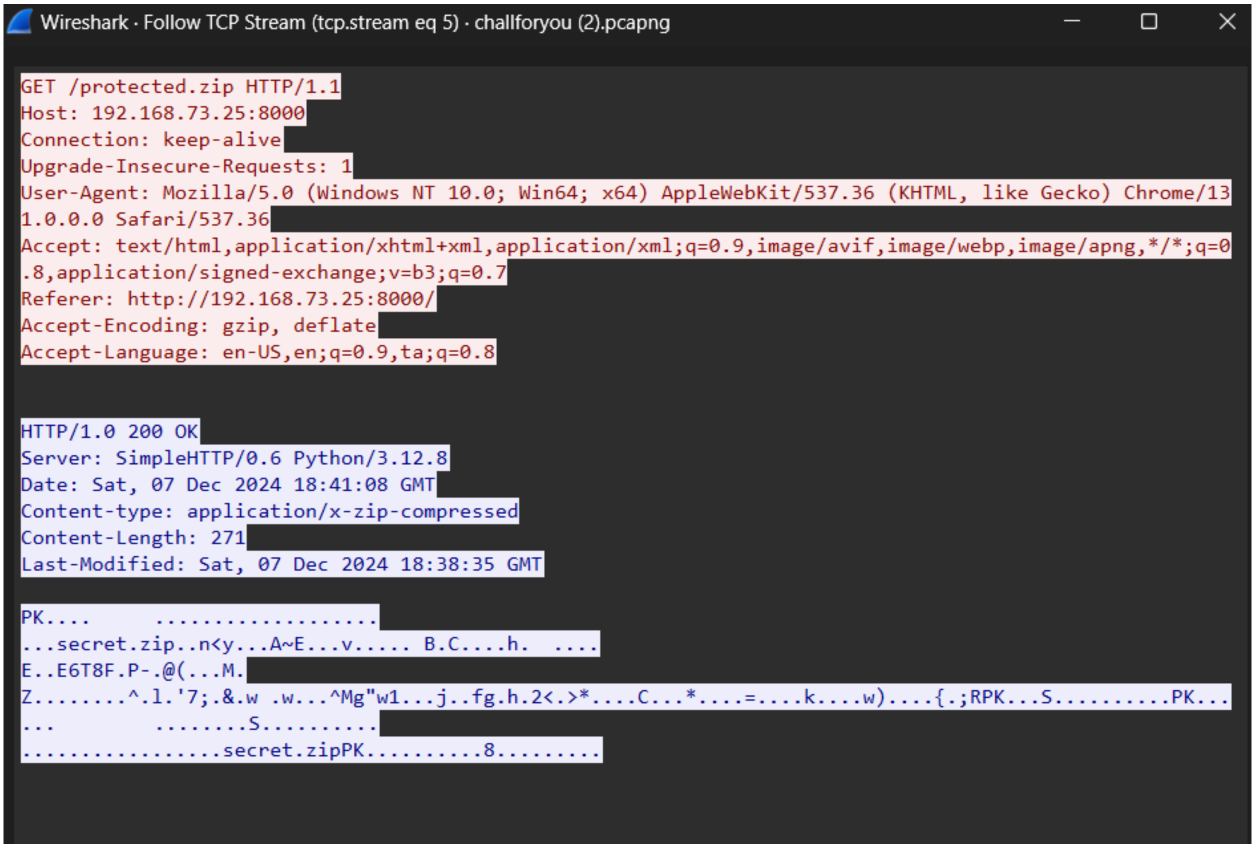
- Downloaded the file and loaded in the wirewshark tool
- Done the same steps as the previous one; filtered and displayed the traffic with http.



challforyou (2).pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
http

```
o.      Time       Source          Destination     Protocol Length Info
    191 68.741066  192.168.73.25   192.168.73.25   HTTP       486 GET / HTTP/1.1
    195 68.755795  192.168.73.25   192.168.73.25   HTTP      5570 HTTP/1.0 200 OK  (text/html)
    201 68.880552  192.168.73.25   192.168.73.25   HTTP       433 GET /favicon.ico HTTP/1.1
    205 68.975268  192.168.73.25   192.168.73.25   HTTP       379 HTTP/1.0 404 File not found  (text/html)
    218 75.582184  192.168.73.25   192.168.73.25   HTTP       536 GET /protected.zip HTTP/1.1
    222 75.586619  192.168.73.25   192.168.73.25   HTTP       315 HTTP/1.0 200 OK  (application/x-zip-compressed)
```

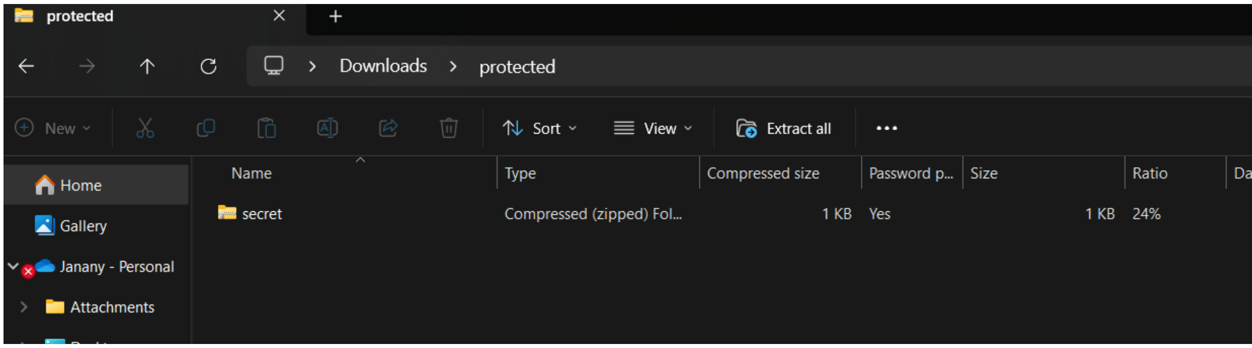
- Chose the packet which contained the zip file i.e **protected.zip**



Wireshark · Follow TCP Stream (tcp.stream eq 5) · challforyou (2).pcapng

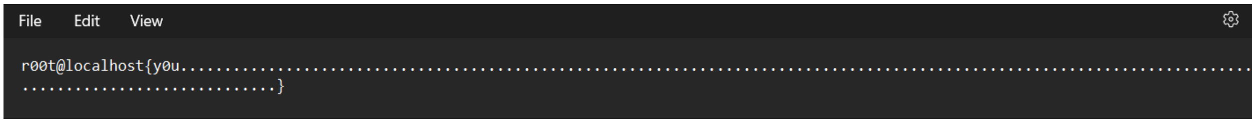
```
GET /protected.zip HTTP/1.1
Host: 192.168.73.25:8000
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/13
1.0.0.0 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0
.8,application/signed-exchange;v=b3;q=0.7
Referer: http://192.168.73.25:8000/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9,ta;q=0.8


HTTP/1.0 200 OK
Server: SimpleHTTP/0.6 Python/3.12.8
Date: Sat, 07 Dec 2024 18:41:08 GMT
Content-type: application/x-zip-compressed
Content-Length: 271
Last-Modified: Sat, 07 Dec 2024 18:38:35 GMT

PK....     ...................
...secret.zip..n<y...A~E...v..... B.C....h.  ....
E..E6T8F.P-.@(...M.
Z.......^.l.'7;.&.w .w...^Mg"w1...j..fg.h.2<.>*....C...*....=....k....w)....{.;RPK...S.........PK...
...        ........S.........
.................secret.zipPK.........8........
```

- Analysed the packets and saved it in the folder as zip file
- When extracted the zip file, it led to another zip file called as secret.zip



Name	Type	Compressed size	Password p...	Size	Ratio	Da
secret	Compressed (zipped) Fol...	1 KB	Yes	1 KB	24%	

- It was protected with password which was cracked using john the ripper tool
- And got the flag which rested inside the zip file as a text file



```
r00t@localhost{y0u.......................................................................................................................
.............................}
```

PLAY WITH QR

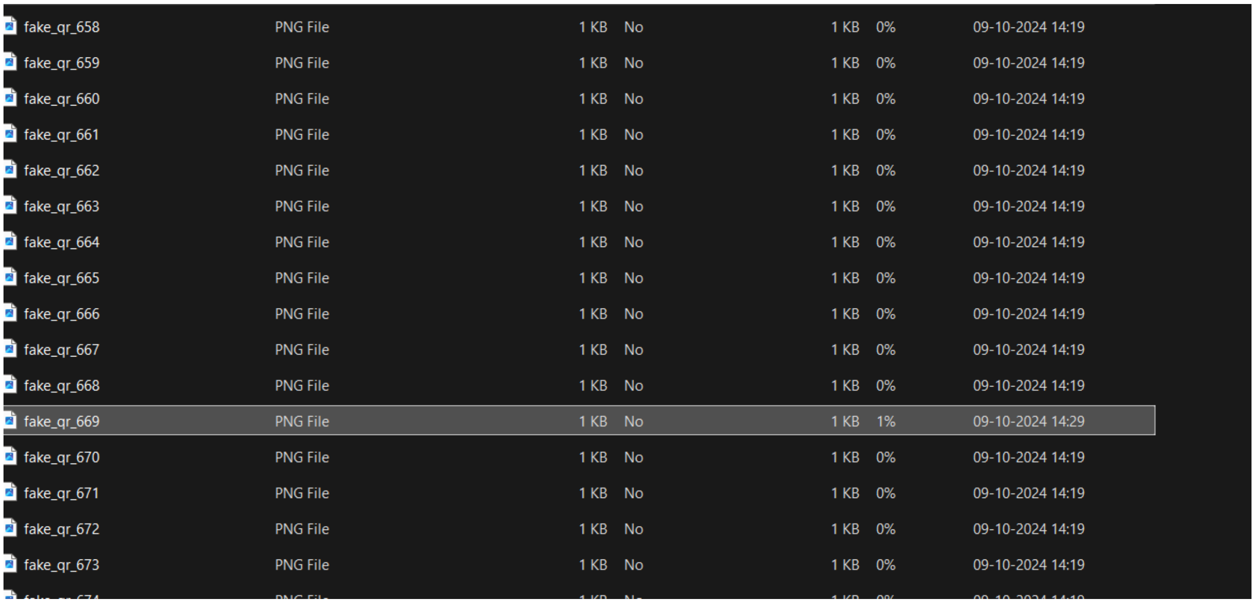
Description:

- A set of qr codes (999) was given in the folder
- One of the qr code contained the real scanner which gave the flag

Approach:

This could have been done using many ways but i just glanced through all the files in my folder and noticed that only qr had file ratio 1% and the rest as 0%

Scanned that particular qr (669)and got the flag luckily



fake_qr_658	PNG File	1 KB	No	1 KB	0%	09-10-2024 14:19
fake_qr_659	PNG File	1 KB	No	1 KB	0%	09-10-2024 14:19
fake_qr_660	PNG File	1 KB	No	1 KB	0%	09-10-2024 14:19
fake_qr_661	PNG File	1 KB	No	1 KB	0%	09-10-2024 14:19
fake_qr_662	PNG File	1 KB	No	1 KB	0%	09-10-2024 14:19
fake_qr_663	PNG File	1 KB	No	1 KB	0%	09-10-2024 14:19
fake_qr_664	PNG File	1 KB	No	1 KB	0%	09-10-2024 14:19
fake_qr_665	PNG File	1 KB	No	1 KB	0%	09-10-2024 14:19
fake_qr_666	PNG File	1 KB	No	1 KB	0%	09-10-2024 14:19
fake_qr_667	PNG File	1 KB	No	1 KB	0%	09-10-2024 14:19
fake_qr_668	PNG File	1 KB	No	1 KB	0%	09-10-2024 14:19
fake_qr_669	PNG File	1 KB	No	1 KB	1%	09-10-2024 14:29
fake_qr_670	PNG File	1 KB	No	1 KB	0%	09-10-2024 14:19
fake_qr_671	PNG File	1 KB	No	1 KB	0%	09-10-2024 14:19
fake_qr_672	PNG File	1 KB	No	1 KB	0%	09-10-2024 14:19
fake_qr_673	PNG File	1 KB	No	1 KB	0%	09-10-2024 14:19
fake_qr_674	PNG File	1 KB	No	1 KB	0%	09-10-2024 14:19

Scanned this qr and got the flag; root@localhost{7h3_q6_!s_fun}