**CTF name:** Quals : Syria and Nigeria National Cyber Security CTF 2020

**Challenge Name:** Life Protector

**Challenge Category:** General Information

**Challenge Description:** A network security system that monitors and controls the incoming and outgoing network traffic based on predetermined security rules

**Level:** Easy

**Answer = Firewall**

**Challenge Name:** Hash3rror

**Challenge Category:** Cryptography

**Challenge Description:** we got this corrupted hash password from a Pcap file with a note (password = sha-1(hash-result)).

**HASH:77be5d24ed2e3e590045e1d6o7e84i50d2799c19f48ede46804a8734e287df120f**

**Level:** Easy

So first off from the note we see that our solution is going to be the sha1 of the hash result but we need to fix the hash . I tried lots of things at first , I tried to identify the hash, I tried to use SHAtter attack, which didn’t work, then I observed the data and found non hexadecimal characters, I fixed the hash by trying to remove any character that is not between A-F & 0-9 after this the hash was fixed .

Our fixed hash was

**HASH:**77be5d24ed2e3e590045e1d67e8450d2799c19f48ede46804a8734e287df120f - Possible algorithms : SHA256

I tried to crack the hash using hashkiller.io.

I had the result : s3cr3tpasword.

After this we go back to the note that says (password = sha-1(hash-result)). Then we try to hash the s3cr3tpassword using sha1 we have

SHA1 : 83874343435092cb681c0d558a84bfeb389c32ed

**Answer = 83874343435092cb681c0d558a84bfeb389c32ed**

**Challenge Name:** Fence

**Challenge Category:** Cryptography

**Challenge Description:** I am lost in this cipher, can you help me?  - Flag format is flag{xxxxxxxxx}

**FIUAECLGSOBEKHFNEAYRTE**

**Challenge Level:** Easy

So for what we know this is a cipher encrypted text, we need to identify the cipher from the name fence, I guessed rail fence cipher, then I headed to ddoce.fr and I had the results:

**FLAGISYOUBREAKTHEFENCE**

**Challenge Name :** Attacker Fault

**Challenge Category:** Digital Forensics

**Challenge Description:** attacker uses IDE , can you find the way to him

We were given a link to download a zip file :

<https://hubchallenges.s3-eu-west-1.amazonaws.com/Forensics/the_key.exe.zip>

**Challenge Level:** Easy

After unzipping we have a executable for Microsoft Windows

since it’s a digital forensics challenge I used the strings command on linux , to find all strings in the executable

The command : strings <file\_name> | grep flag

that returns the following result

flag{\attacker\jack\0123456789phon\10-11-12ddthomraddre\payloadpdb}&'

**Answer: flag{\attacker\jack\0123456789phon\10-11-12ddthomraddre\payloadpdb}**

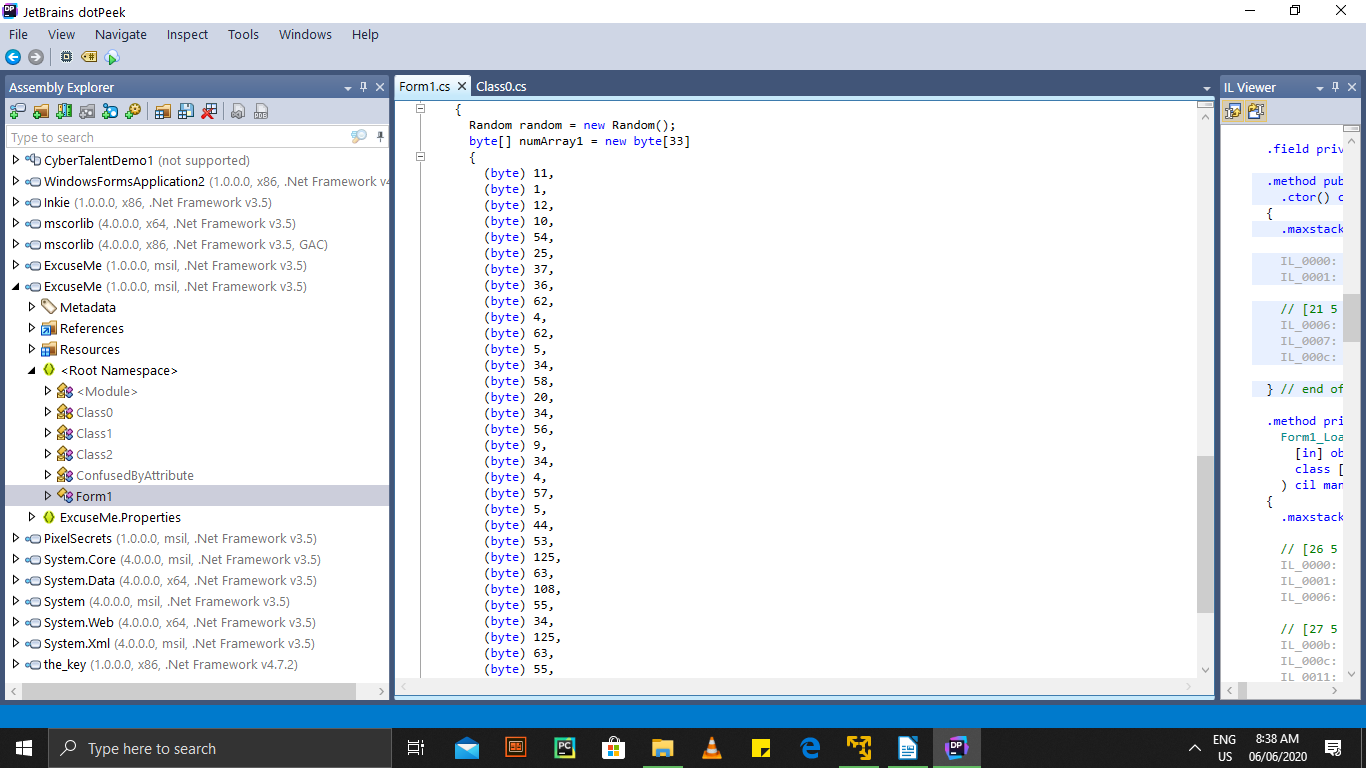
**Challenge Name:** Pardon

**Challenge Category:** Pardon Malware Reverse Engineering

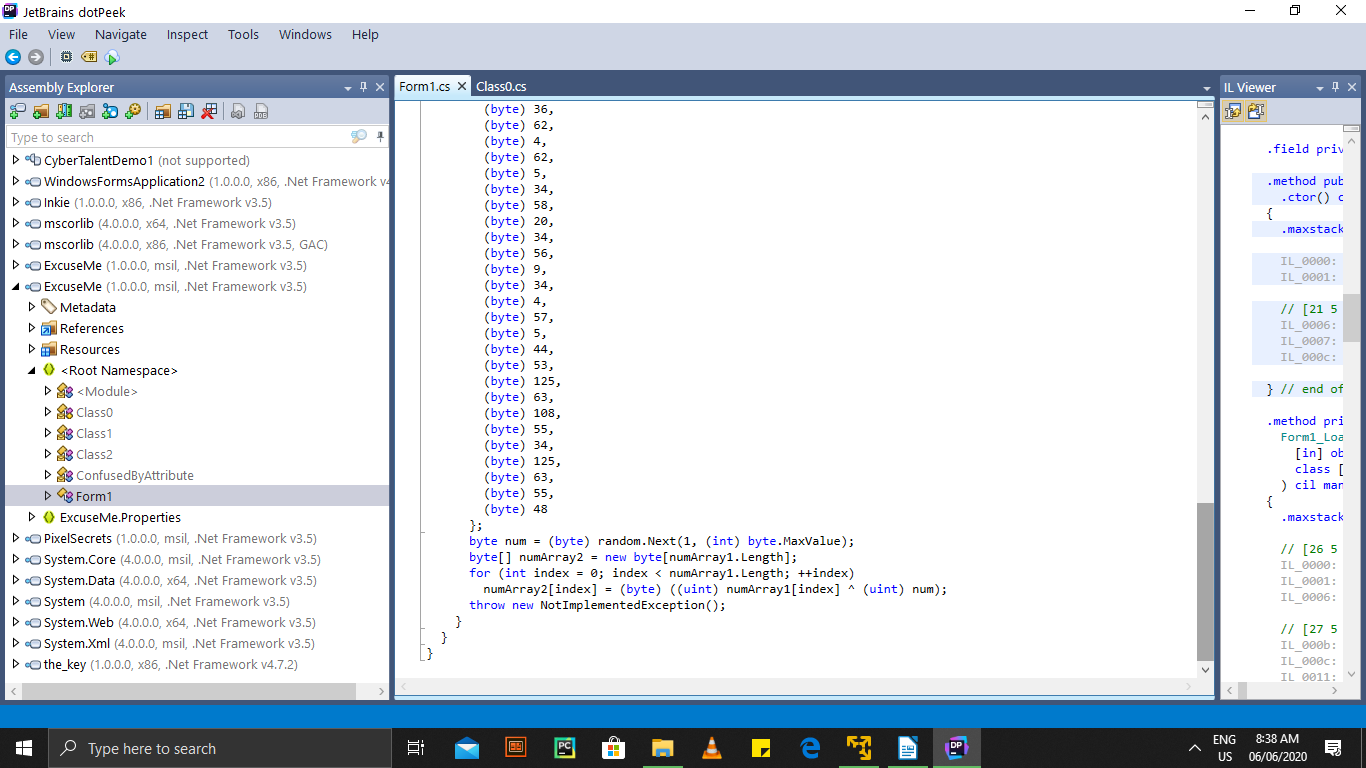
**Challenge Description:** A developer was too tired to remember where did he write his code, can you help him find it?

Here is the link to download the code : [**https://hubchallenges.s3-eu-west-1.amazonaws.com/Reverse/pardon.ex**](https://hubchallenges.s3-eu-west-1.amazonaws.com/Reverse/pardon.exe)e

I loaded the executable into dotpeek, then I navigated to the source code



We have a C# code an array of type bytes



We see that some logical operation is going on we try to reverse this operation see the link below for reference

<https://docs.microsoft.com/en-us/dotnet/csharp/language-reference/operators/boolean-logical-operators>

Based on the flag format we should expect that the format is flag{xxxxx}

### So picking the fifth element in the array and OR in it against all numbers form 1 – 256 and then convert the result to ascii and then plain text, any of the numbers that returns ‘ { ‘ will be used to OR all the elements in the array writing a simple C# code

Console.WriteLine(54^77)

and converting the result from decimal 123 we have the text ‘{‘ , repeating this process for all the elements in the array we have the result

**FLAG{ThisIsHowYouDoItHax0r!zo0rz}**

**Challenge Name:** Snake

**Challenge Category:** Web Security

**Challenge Description:** understand the snake

**Level:** Easy

http://ec2-18-184-241-234.eu-central-1.compute.amazonaws.com/snak3/

I started out by doing a directory brute force attack using dirb and word lists common.txt

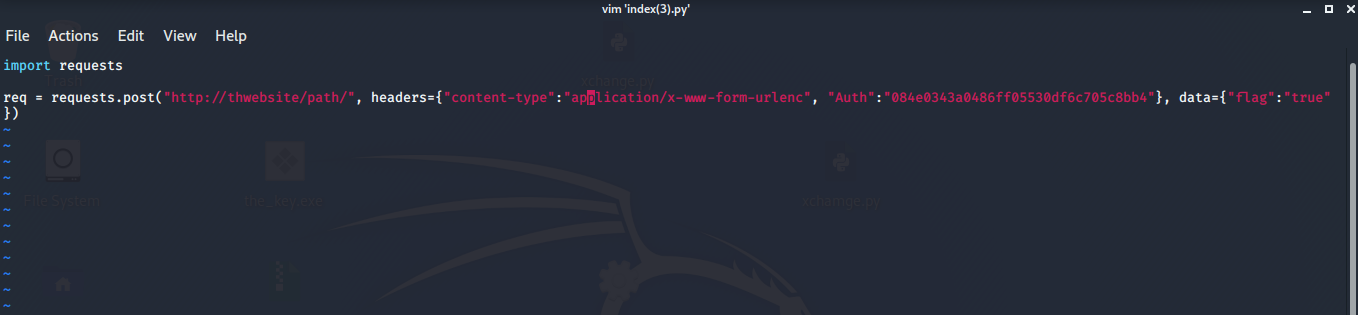
Command: dirb [**http://ec2-18-184-241-234.eu-central-1.compute.amazonaws.com/snak3/**](http://ec2-18-184-241-234.eu-central-1.compute.amazonaws.com/snak3/)

Found the directories: **1.** *snak3/robots.txt* **2.** snak3/index.php

At robots.txt I found that there was a index.py file so I tried to get the file by going to the directory *snak3*/*index.py* and downloaded the file

Opening the python file using vim you can use any text editor

Command: vim index.py



From the file I noticed that it was a script for a post request containing a authentication hash, I tried to crack the hash and it gave the result ‘guest’

Then I added the url of the web site to the post request , and tried to run it but it returned error 404: page not found so, I tried to change the authentication by hashing the word ‘admin’ to md5 and then tried to run the python script that gave the result:

FLAG{ReqUest\_heaDers\_&\_PyThon}the snake around you somewhere

**Answer : FLAG{ReqUest\_heaDers\_&\_PyThon}**

**Challenge Name:** 404

**Challenge Category**: Web Security

**Challenge Description:** Sometimes you should see all the request faces!

**Level:** Medium

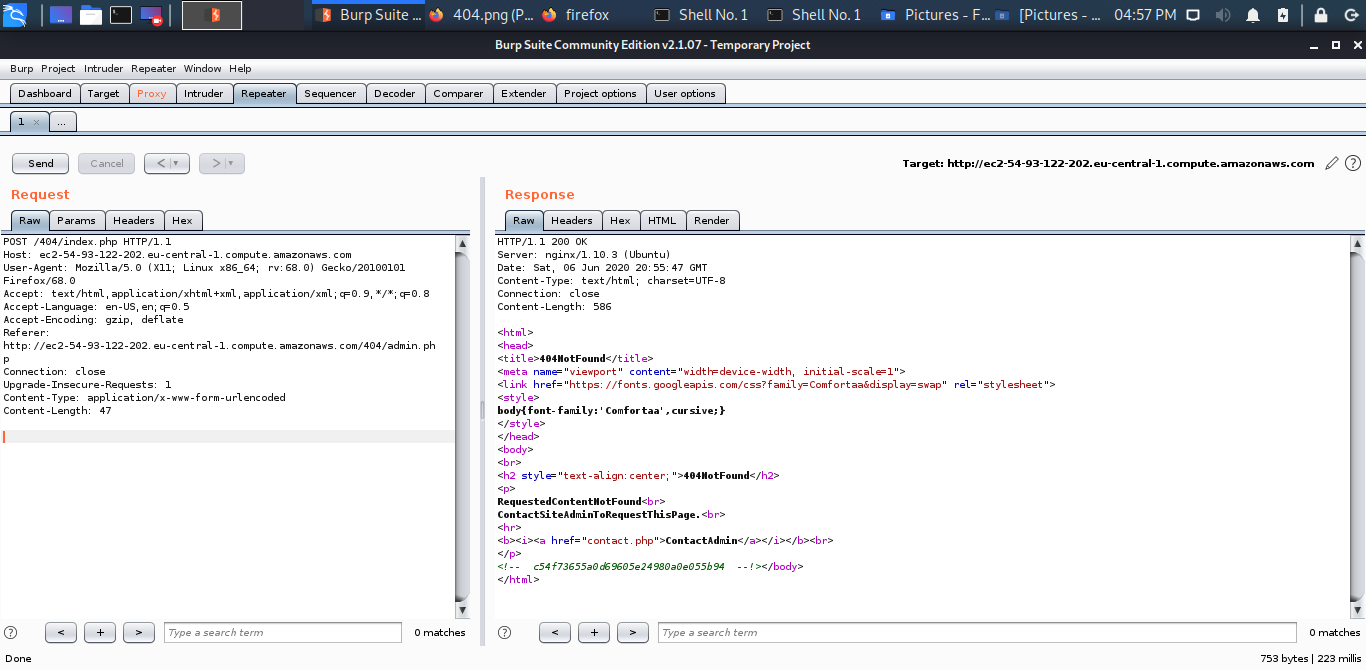
**Link:** http:// [**ec2-54-93-122-202.eu-central-1.compute.amazonaws.com/404/**](http://ec2-54-93-122-202.eu-central-1.compute.amazonaws.com/404/)

Like the last challenge we need to find all directories so we use dirb

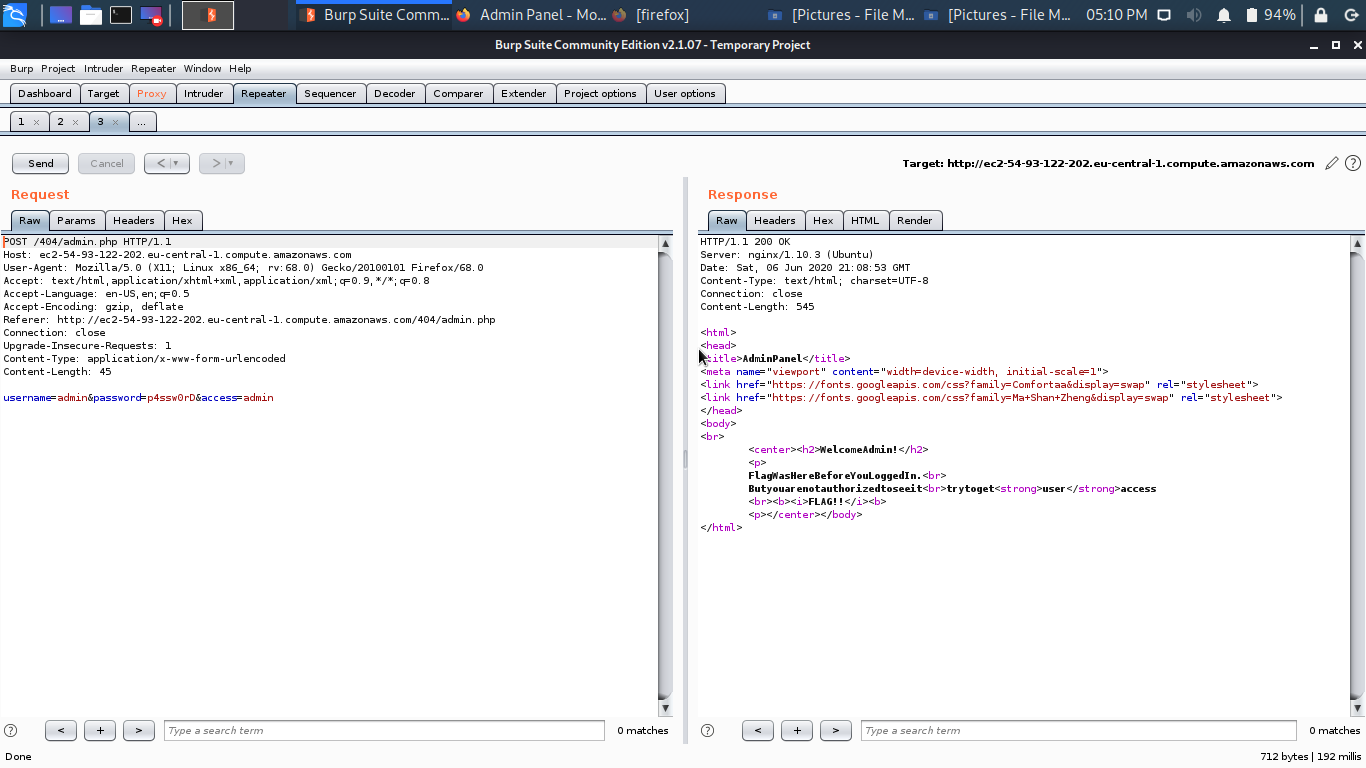
Command: dirb [**http://ec2-54-93-122-202.eu-central-1.compute.amazonaws.com/404/**](http://ec2-54-93-122-202.eu-central-1.compute.amazonaws.com/404/)

We find the directories: *404/admin.php, 404/index.php*

Sending a post request to the landing page I found md5sum of admin password cracking the hash we have the password p4ssw0rD



That is asking for the admin login and password having configured burp proxy we enter username admin and the password we got from the hash



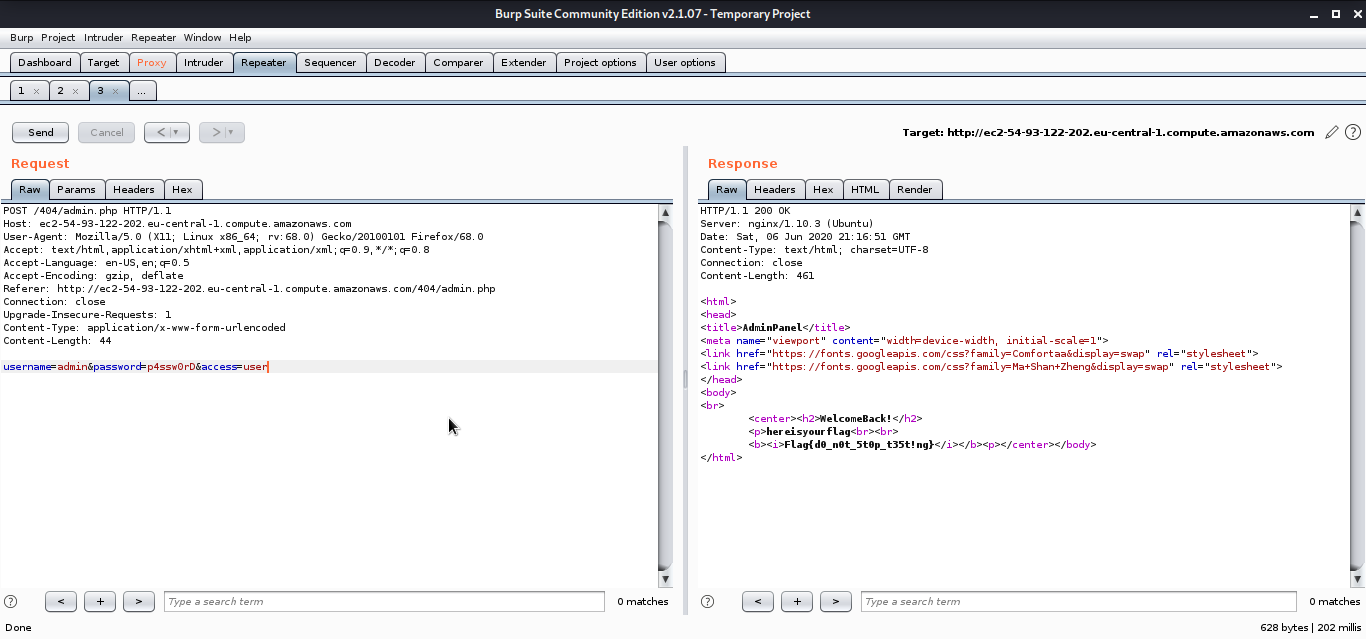
It we see from the HTML result see this line of code

‘Flag Was Here Before You Logged In’

Looking at the result we also see “try to get user access”

So changing the access to user from post request

We get our flag:

**Flag{d0\_n0t\_5t0p\_t35t!ng}**

**Challenge Name:** Force Me

**Challenge Category:** Web Security

**Challenge Description:** Write your own script

**Level:** Medium

**Link:** [http://ec2-54-93-122-2](http://ec2-54-93-122-2/) [02.eu-central-1.compute.amazonaws.com/Force-me/](http://ec2-54-93-122-202.eu-central-1.compute.amazonaws.com/Force-me/)

I started by doing a directory brute force attack as seen in the previous challenges we get a directory /*Force-me/Readme.txt*

Opening the directory we see this

*flag\_$$$$$.html*

# this $ refers to a number, u can brute force it just write a script :P.

I now know that our flag is at a directory *flag\_$$$$$.html* where $ represents a digit, we need 5 digits so I wrote a script to print out all the possible combinations of all 5 digits from 0-9 and then added the strings flag\_ to it , and then created a text file of the results to serve as a wordlist , I then used go buster to bruteforce the directories

we find a directory flag\_23097.html

at the directory we get the flag

**Answer : FLAG{I\_L0v3\_Scr1pt1nG}**

**Challenge Name:** Hidden Space

**Challenge Category:** Digital Forensics

**Challenge Description:** NASA website was hacked, but no defacement was found, can you find anything suspicious?

**Level:** Medium

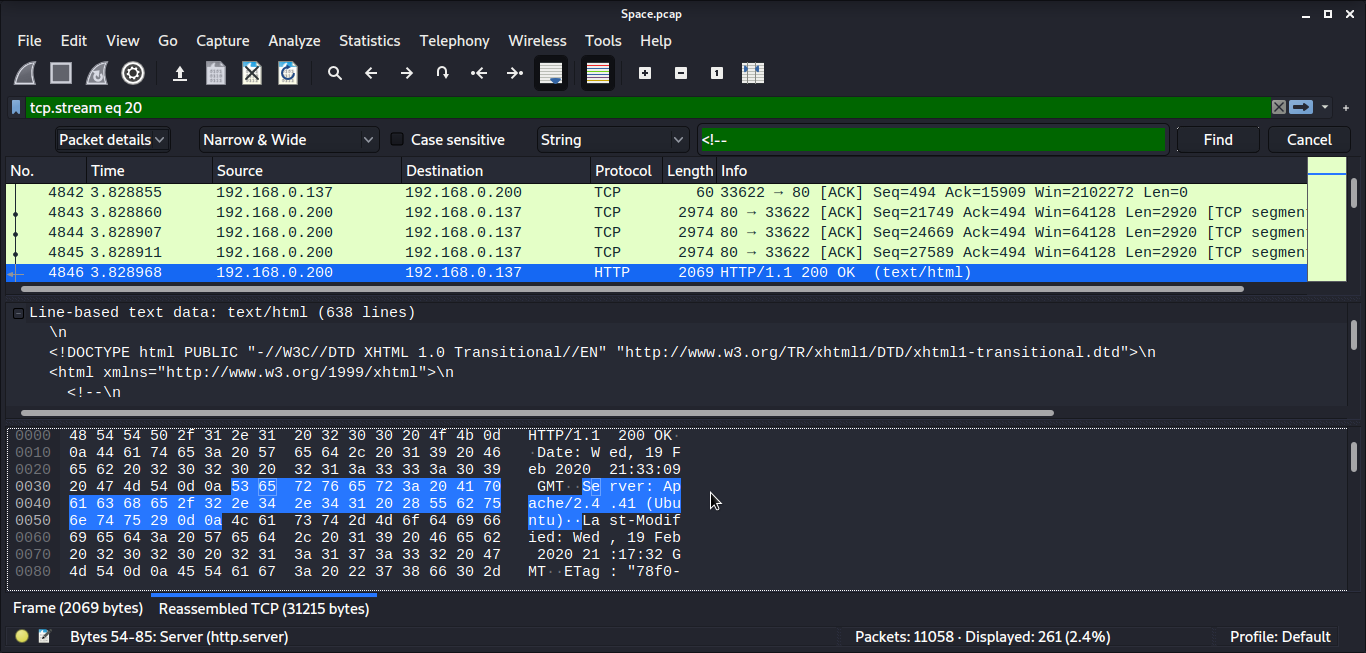
We are given a pcap file, since it’s a digital forensics challenge I tried to find hidden files, data from the file I started by using binwalk , foremost that led to the same results, I found that the pcap file contains a lot of HTML documents, then I tried to use the strings command

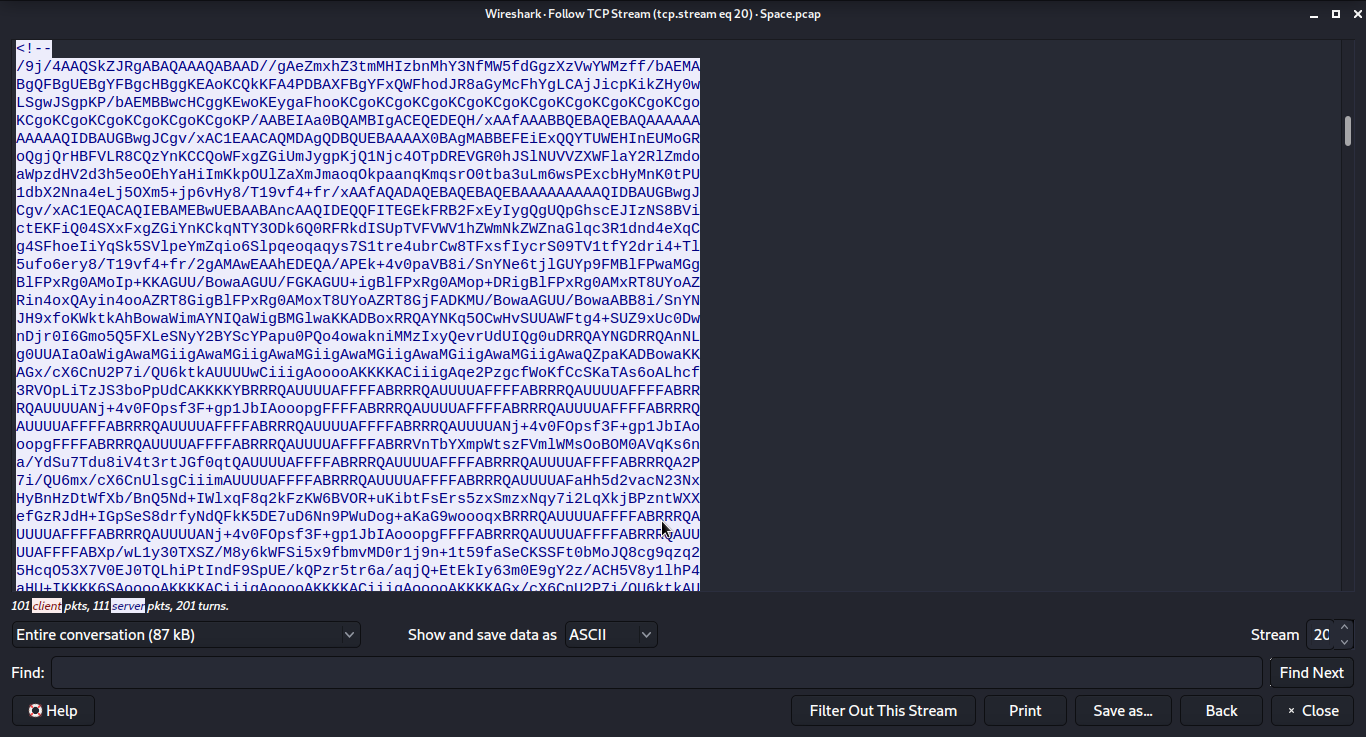
command: strings file\_name

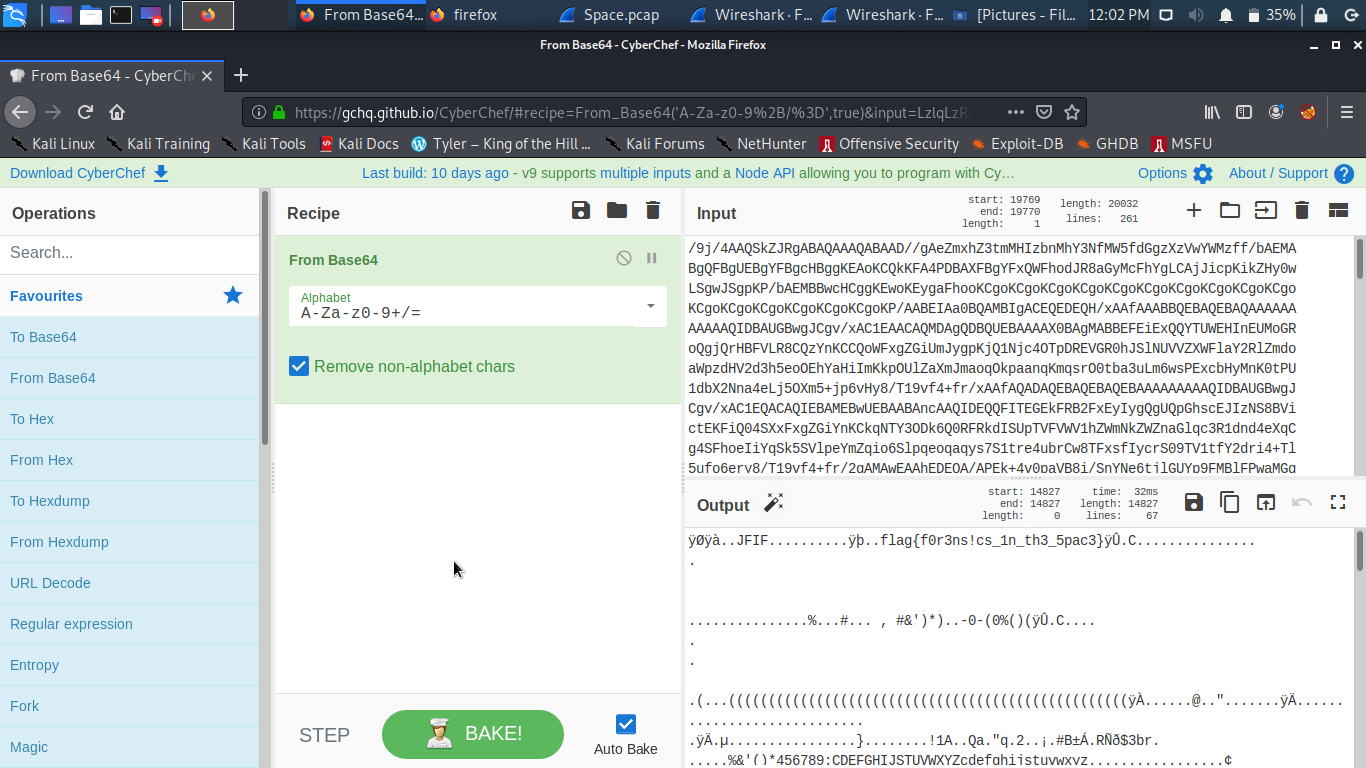
I noticed that a conversation is occurring between the server and the user

following the conversation, I noticed that the user is trying a directory bruteforce attack, then I opened up the pcap file with wireshark and then

searched for the strings ‘<!--’ in the packet details ,then I followed the tcp stream and found some type of data



At the end of the data i find a ‘ ==’ sign that means this data might be bas64 encoded,



Then converting the data from base64 to plain text we have our flag **Answer: flag{f0r3ns!cs\_1n\_th3\_5pac3}**

**Challenge Name:** Frenzy

**Challenge Category:** Malware Reverse Engineering

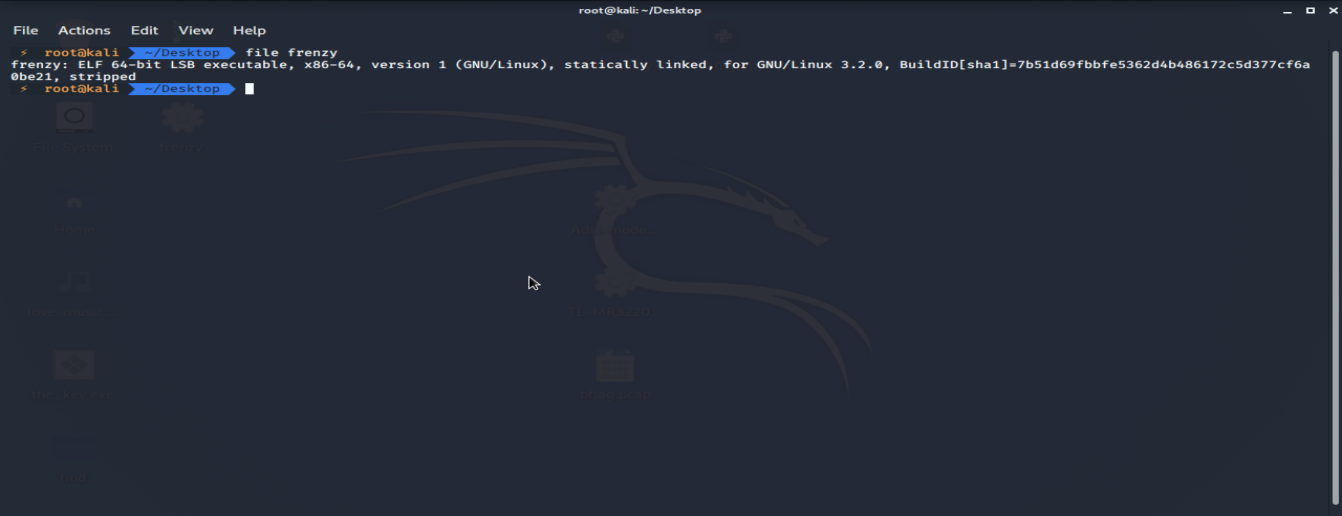
**Challenge Description:**

**Level:** Medium

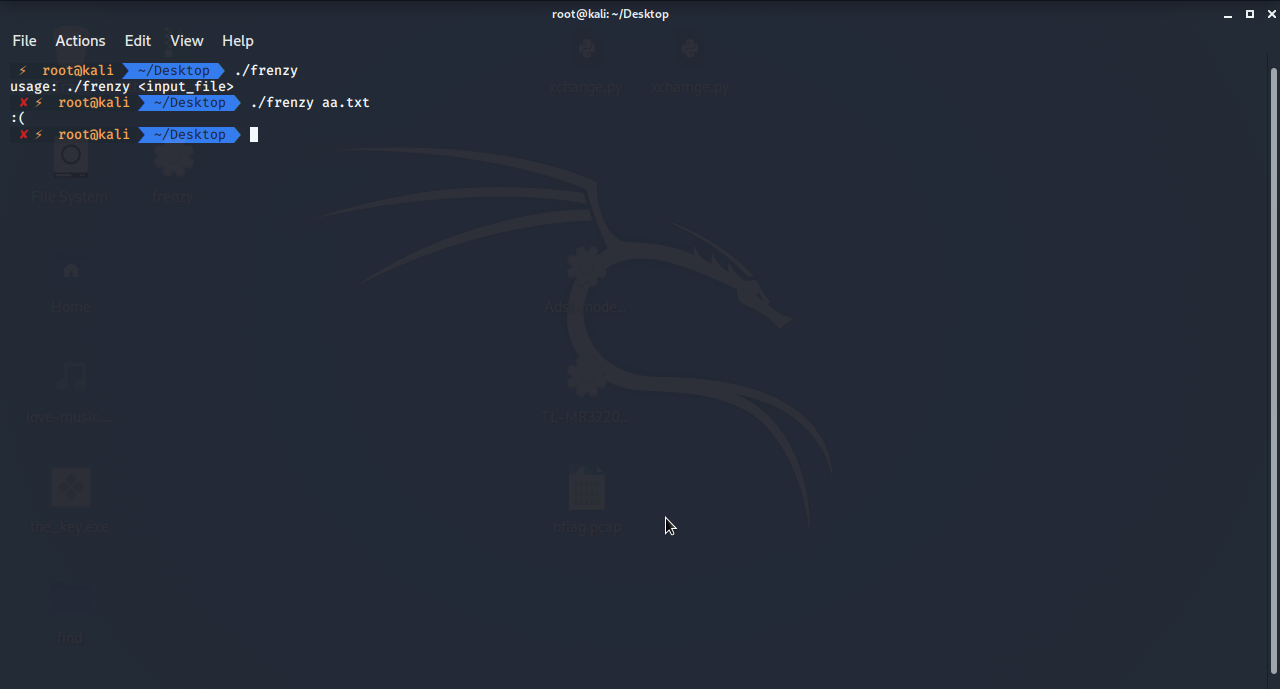
First off, we need to gather information about the file by using the file command

command: file file\_name

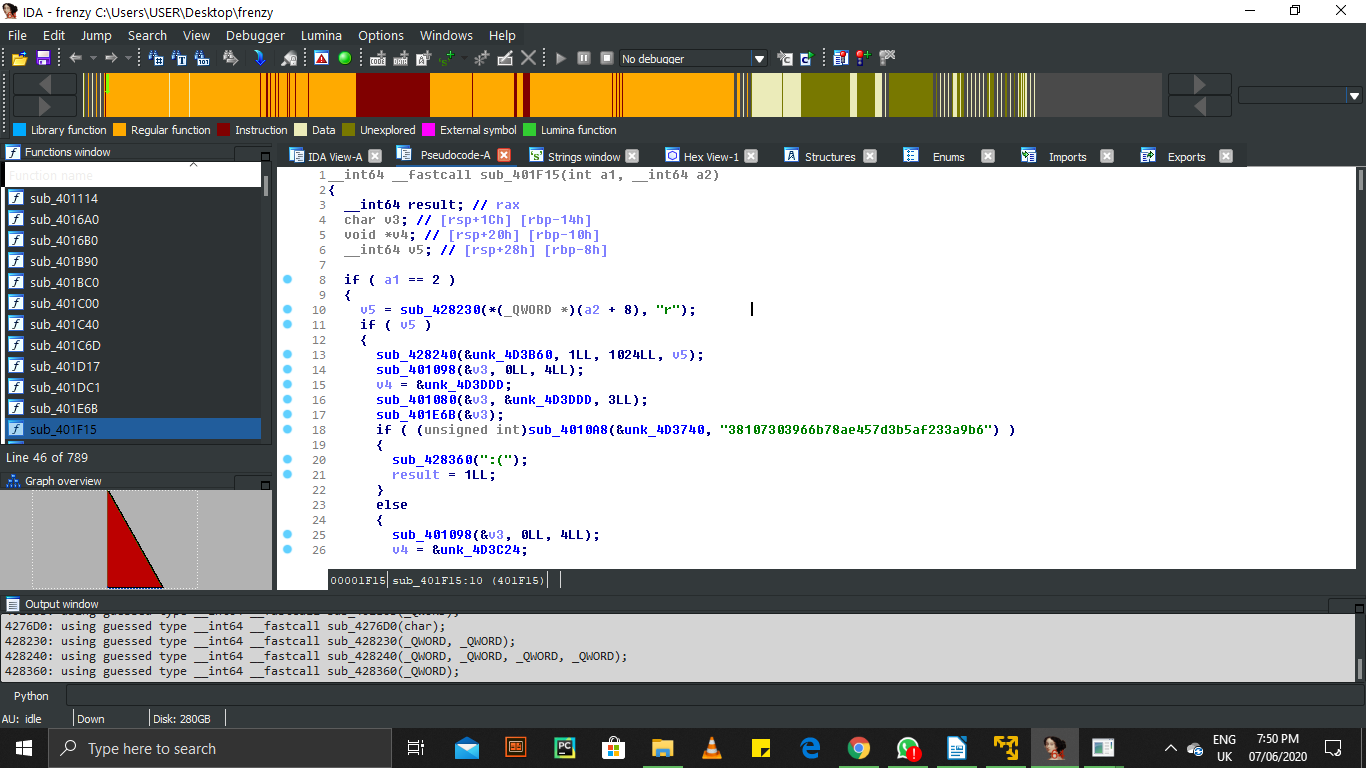
As we can see we have a elf executable so I tried to execute it.



The executable asked for an input file, I tried using a random file we see that the elf returns ‘:(‘



Now we go to IDA to disassemble the file, we see a whole lot of functions.

I try to find the main function so I searched the strings window and found the string <input\_file> using the xrefs option, we get the main function, and then generate the pseudocode, to try and understand the instructions