[Project Sekai] GCSC 2022 Writeup

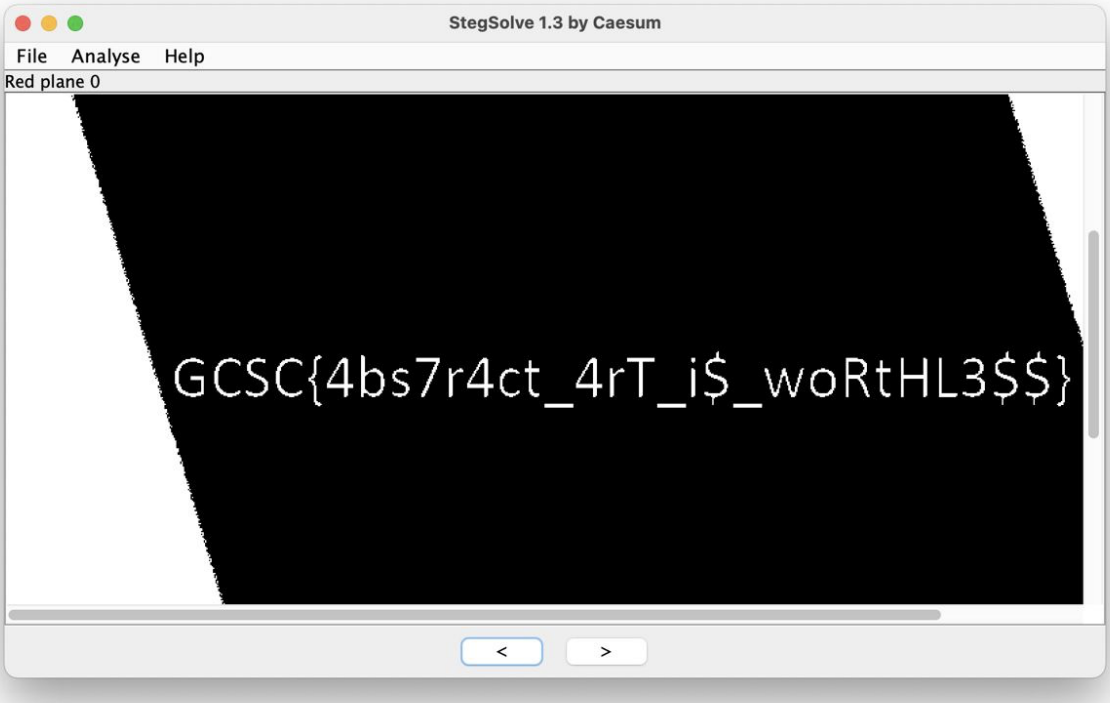
GCSC 2022 Write-ups

South Africa

- File attached is an SVG file.
- Inside of it we can find 2 PNG bitmaps
- The second bitmap contains a barcode.
- Scan the bar code to get flag.

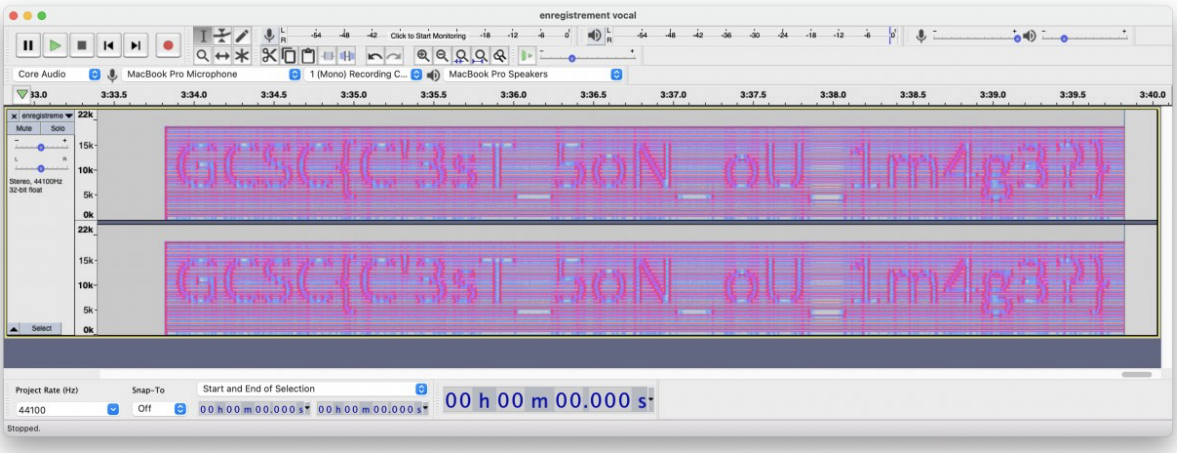
Cameroon

- Download the picture attached
- Inspect Bit 0 of Red channel to find the flag hidden in the square



Angola

- Download the music file attached
- View the spectrogram of the file. flag can be found at the end of the music



Russia

- Send request to /flag to find 405 Method Not Allowed and GCSC in the Accept header.
- Send the request with GCSC Method to get the flag.

```
$ http http://challenges.guinean-cybertaskforce.com:8001/flag
HTTP/1.0 405 METHOD NOT ALLOWED
Allow: OPTIONS, GCSC
Content-Length: 178
Content-Type: text/html; charset=utf-8
Date: Mon, 21 Feb 2022 00:35:46 GMT
Server: Werkzeug/2.0.3 Python/3.8.10

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<title>405 Method Not Allowed</title>
<h1>Method Not Allowed</h1>
<p>The method is not allowed for the requested URL.</p>

$ http GCSC http://challenges.guinean-cybertaskforce.com:8001/flag
HTTP/1.0 200 OK
Content-Length: 40
Content-Type: text/html; charset=utf-8
Date: Mon, 21 Feb 2022 00:35:42 GMT
Server: Werkzeug/2.0.3 Python/3.8.10

GCSC{p0s7_Et_g3t_ne_$ont_p4$_sp3c13lle5}
```

Canada

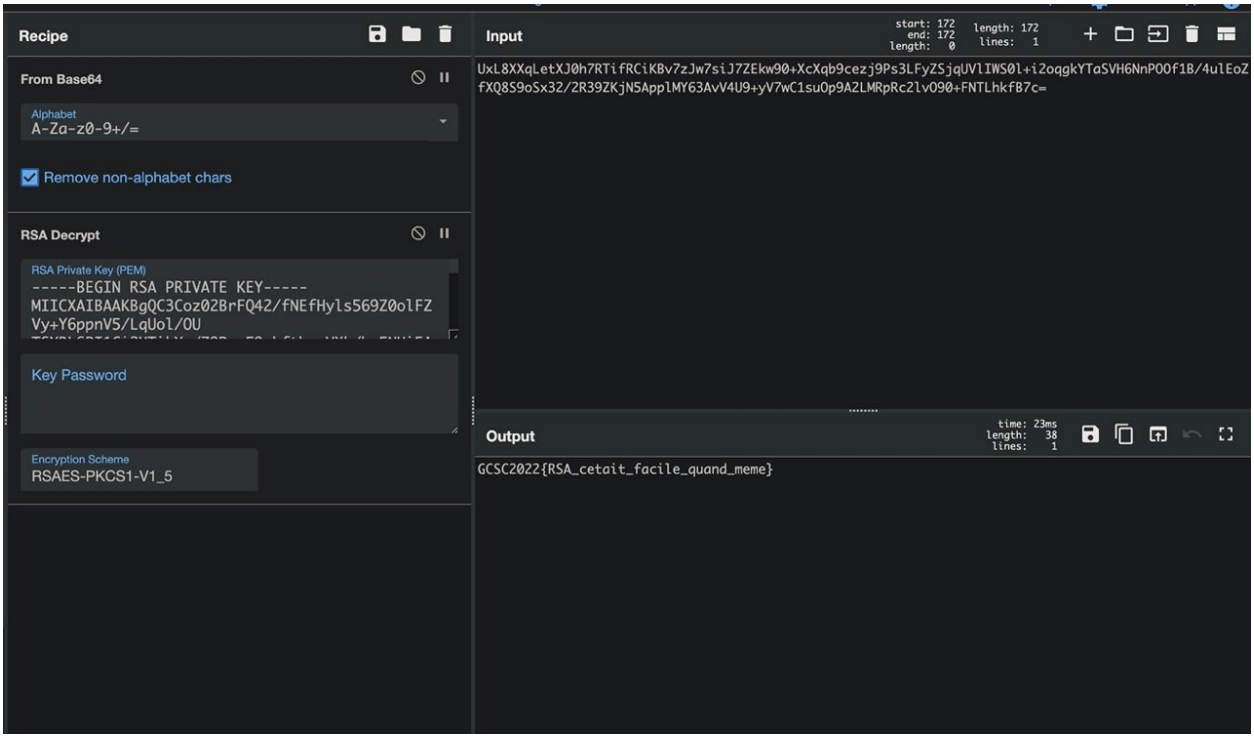
- Register with any username, password, email, and put personal website as `https://example.com`
- Submit the form and see an exception traceback of Flask
- Flag can be found in the exception message

http://challenges.guinean-cybertaskforce.com:5000/register?user=user&passwd=password&email=email&url=https://example.com

> requests.exceptions.ConnectionError: HTTPConnectionPool(host='https', port=80): Max retries exceeded with url: //example.com/**GCSC%7BUt1llisseZ_t0ujoUr$_c0ll4boRat0R%7D** (Caused by NewConnectionError('<urllib3.connection.HTTPConnection object at 0x7f74df7fc9a0>: Failed to establish a new connection: [Errno -2] Name or service not known'))

US

- Decode the cipher text from Base64
- Decrypt the cipher text with the given PGP private key in RSAES-PKCS1-V1_5 to get the flag



Greenland

- Brute force the entire ASCII printable non-whitespace characters one by one to get the flag.
- The initial part of the text is an extract of *the Declaration of Independence* in Leet. Optimization made to the brute force script to accelerate.

> We hold these truths to be self-evident, that all men are created equal, that they are endowed by their Creator with certain unalienable Rights, that among these are Life, Liberty and the pursuit of Happiness.

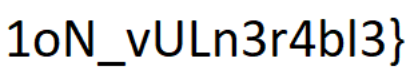
```
from pwn import *
context.update(log_level="error")
text = b""
flag = b"Flag:" + text
basis = "We hold these truths to be self-evident, that all men are created equal, that they are endowed by their Creator with certain unali
pred = {
    b"e": [b"3"],
    b"a": [b"4"],
    b"s": [b"$", b'5'],
    b"I": [b"1"],
    b"i": [b"1"],
    b"t": [b"7"],
    b"o": [b"0"],
    b"E": [b"3"],
    b"A": [b"4"],
    b"S": [b"$", b'5'],
    b"T": [b"7"],
    b"O": [b"0"],
    b" ": [b"_"],
    b",": [b","],
    b".": [b"_"],
    b"-": [b"-"],
}
```

```
for l, u in zip("abcdefghijklmnopqrstuvwxyz", "ABCDEFGHIJKLMNOPQRSTUVWXYZ"):
    l = l.encode()
    u = u.encode()
    if l not in pred: pred[l] = []
    if u not in pred: pred[u] = []
    pred[l].append(l)
    pred[l].append(u)
    pred[u].append(u)
    pred[u].append(l)

def query(text):
    io = remote('159.65.242.247', 9090)
    io.send(text + b'\n')
    return io.recv(500)

changed = True
while changed:
    changed = False
    if len(text) < len(basis):
        bas = basis[len(text)].encode()
        cand = pred[bas]
        for char in cand:
            res = query(text + char)
            if res.startswith(flag) and len(res) > len(flag):
                changed = True
                flag += char
                text += char
                print(text)
                break
    if not changed:
        order = b"01347abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ_{}25689$[\\]^`:;<=>?@|~!\"#%&\'()*+,-./"
        for i in order:
            res = query(text + char)
            if res.startswith(flag) and len(res) > len(flag):
                changed = True
                flag += char
                text += char
                print(text)
                break
```

Côte d'Ivoire

- Inspect the APK and 3 activities are found
  - `com.example.gcsc.MainActivity` (Main activity)
  - `com.example.gcsc.s3cr3t` (Exported activity)
  - `com.example.gcsc.ApPL1c4t`
- Launching the `s3cr3t` activity, we can find a picture with a partial flag.

`adb shell am start -n com.example.gcsc/.s3cr3t`



- Joining this with `GCSC2022{` and `ApPL1c4t` found in the third activity ID to get the flag.

Argentina

- Free flag in the hint.

China

- We are given a poem and following Python script.

```
#
#Trouve lerreur dans un script et exécute-le pour avoir le flag
#
```

```
#! /usr/local/bin/python  -*- coding: UTF-8 -*-
poeme = open('poem-14février.txt')

flag=""
for vers in poeme:
    vers=vers.strip()
    lettre = line[0:1]
    flag=flag+lettre

print flag
```

- Running the script on that poem gives the flag `FLAGCMPHDDSQNUCCPNNSOQACJOOP`

Iran

- We are given the following quizzes to complete.
  1. Which group of hackers mainly targets telecommunications companies: (Expected response: Group name - ID on Att&ck)
     - Searching on Att&ck, we need to look for clues given in this challenge: The group mainly targets `telecommunication` and is related to `Iran`. This gives us the result - `APT39 - G0087`
  2. What is the country of origin of this famous group? (Expected answer: Country in French)
     - The country is simply `Iran`.
  3. The group uses a special technique to exfiltrate data. What is the technique name and ID (Expected response: Name - ID)
     - The only data exfiltration listed on group page is `Exfiltration Over C2 Channel - T1041`.
  4. To mitigate the previous technique, a means is used. What is the name of this mitigation and its ID?
     - There are 2 mitigations for `T1041` (https://attack.mitre.org/techniques/T1041/): `M1057` and `M1031`. `M1057` is Data Loss Prevention and is not the mitigation here since some C2 use encrypted channel to transfer data. Therefore answer is `Network Intrusion Prevention - M1031`
  5. What is the most common sub-technique used by the hacker group to target corporate victims?
     - `Spearphishing Attachment - T1566.001`
  6. What is the command interpreter and scripts used in the previous sub-technique to run commands on victim machines?
     - Apparently this will be related to scripting, looking on Att&ck we found https://attack.mitre.org/techniques/T1059/ to be the only possible technique. `APT39` uses `Visual Basic - T1059.005` and this is the solution.

Madagascar

- Flag is simply Discord invite link wrapped with `GCSC2022{}`.

Congo

- We are given a `secrets.kdbx` file. Using `keepass2johh`, we can obtain a hash file.

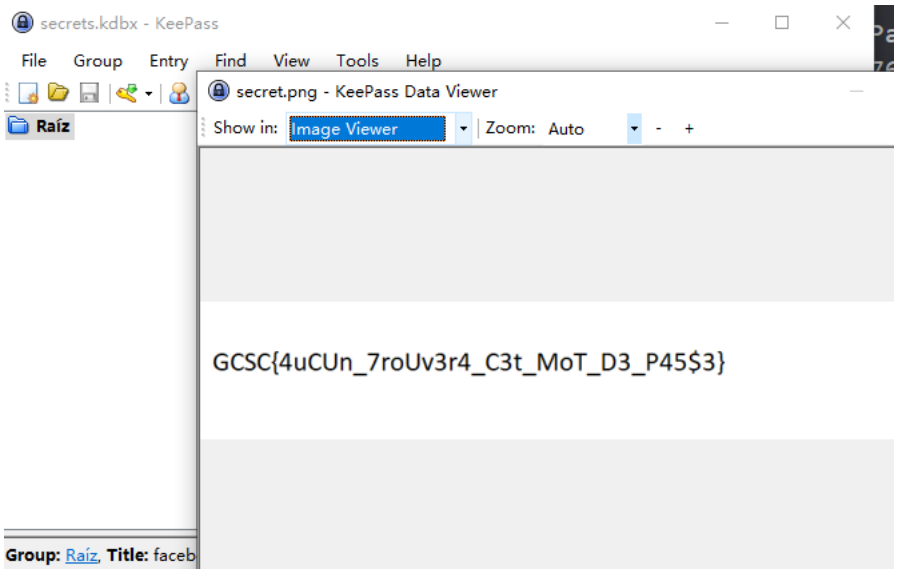
> secrets:$keepass$*230*222*baeaaca143ecac0a48c76109f1af109db2e1e2c27cc02e59a57317854ba71bfl

- Now we can use Hashcat on `rockyou.txt` to find matching hash.

```
Dictionary cache built:
* Filename..: wordlist\rockyou.txt
* Passwords.: 14344391
* Bytes.....: 139921497
* Keyspace..: 14344384
* Runtime...: 1 sec

$keepass$*2*30*222*baeaaca143ecac0a48c76109f1af109db2e1e2c27cc02e59a57317854ba71bfb*4b76a74e00b23ed3ffe5c3a10afcafc8bf0944500c32eda170cd900

Session..........: hashcat
Status...........: Cracked
Hash.Mode........: 13400 (KeePass 1 (AES/Twofish) and KeePass 2 (AES))
```

```
Hash.Target......: $keepass$*2*30*222*baeaaca143ecac0a48c76109f1af109d...ab350b
Time.Started.....: Sat Feb 19 16:31:46 2022 (1 sec)
Time.Estimated...: Sat Feb 19 16:31:47 2022 (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.......: File (wordlist\rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........:   852.5 kH/s (0.97ms) @ Accel:64 Loops:3 Thr:128 Vec:1
Recovered........: 1/1 (100.00%) Digests
Progress.........: 851968/14344384 (5.94%)
Rejected.........: 0/851968 (0.00%)
Restore.Point....: 745472/14344384 (5.20%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:27-30
Candidate.Engine.: Device Generator
Candidates.#1....: 007373 -> momo5390
Hardware.Mon.#1..: Temp: 44c Fan: 30% Util:  8% Core:1278MHz Mem:3004MHz Bus:16
```

- We now obtain the password `rellik000`. Logging into KeePass, in one of the account attachment we see `secret.png` which is the flag.

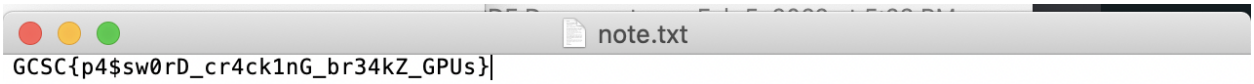


Chad

- We are given an encrypted zip file. Using `john` to obtain the zip hash.

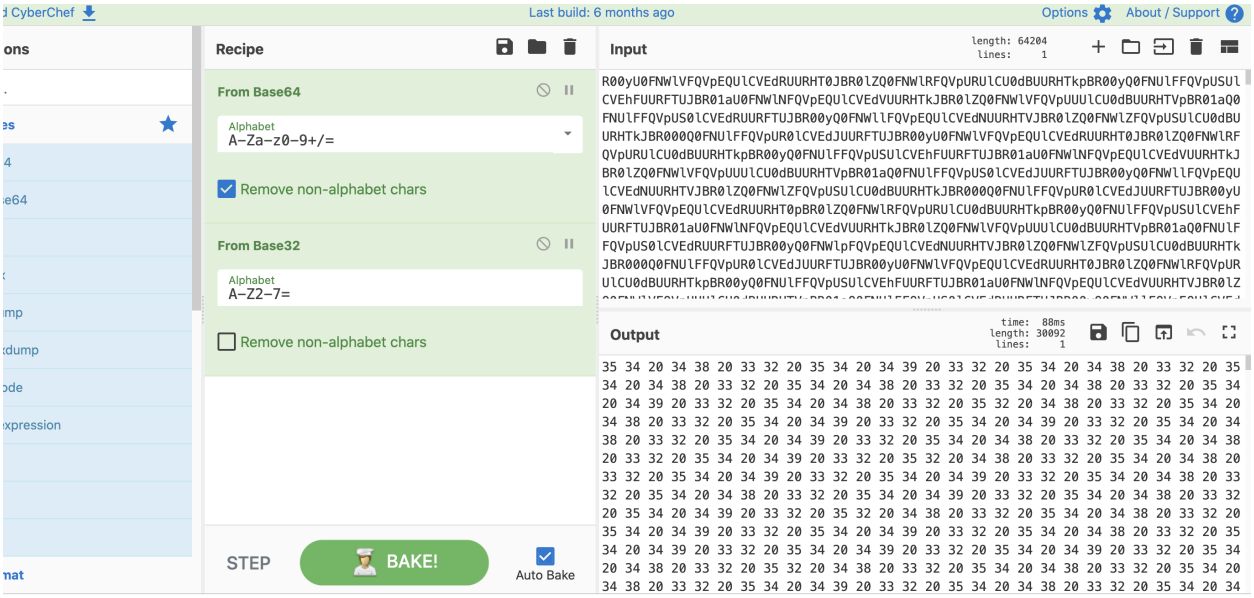
```
$pkzip2$11202f23d412461502602fd412745a0ff07062b8da65181bef4b45fc92eb57472f113ac911676100ab435a5d2f951791cd2aa52db3459b1524cce55b7e2f$/pkzip
```

- Again, we use Hashcat on `rockyou.txt` to find password corresponding to this `pkzip2` hash. This time we get `Catsandcows` as the password. Opening the txt inside zip gives the flag.

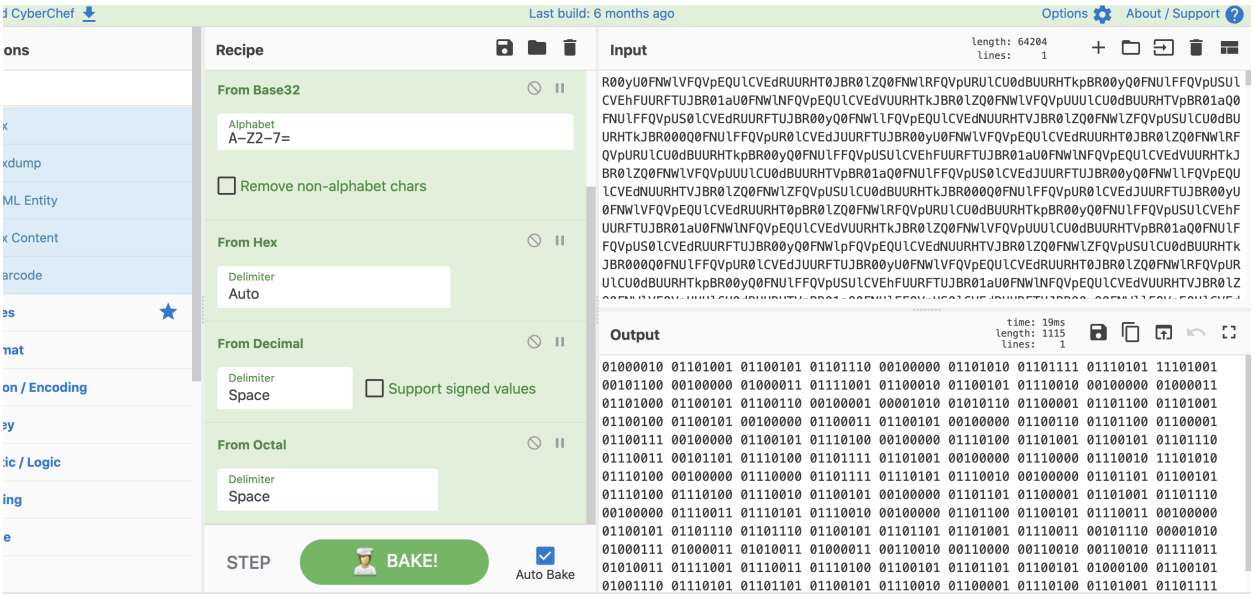


Senegal

- We are given a base64 string, after using magic 2 times in CyberChef we get the following:



- This seems to be some hex numbers, converting again with magic:



- Now it seems to be some ASCII numbers in their binary format, so we convert from binary and get the flag.

> Bien joué, Cyber Chef!
> Valide ce flag et tiens-toi prêt pour mettre main sur les ennemis.
> GCSC2022{SystemeDeNumeration!!!}

Zambia

- In this network analysis challenge we are given a pcap file. Opening it in WireShark and doing some analysis, we can see username and password from an html file.

```
username=GCSC&password=%7BUt1l1%243z_70uJoUR%24_H77P%24%21%7D&login=
```

- The flag is just the password - `GCSC2022{Ut1l1$3z_70uJoUR$_H77P$!}`

Nigeria

- We are given a `.db` file. Using sqlite3 to load the table, we can see the only table is `users` with the following entries:

```
[('John', 'ODI3MDQ0MGMwZTk0MTExZDExZmFmMzg1MTI1Nzc4ZTc='), ('Marie', 'NDFkNTM4YTcxYjJhYTYwMDRkNTM0MjM3NzEwNDFjOWMK'), ('Luc', 'NGVmY2RhYzg1YzZjNDk0YjNjZWY4NzgxY2M2Mzk4MDUK')]
```

- The string values are base64, converting gives

> John - 8270440c0e94111d11faf385125778e7
> Marie - 41d538a71b2aa6004d53423771041c9c
> Luc - 4efcdac85c6c494b3cef8781cc639805

- Those are md5 hashes and if we google them, we can get they correspond to `p1ntap`, `fd1972` and `AdDiCtIoN`. Concat them gives the flag.

Kenya

- Challenge:

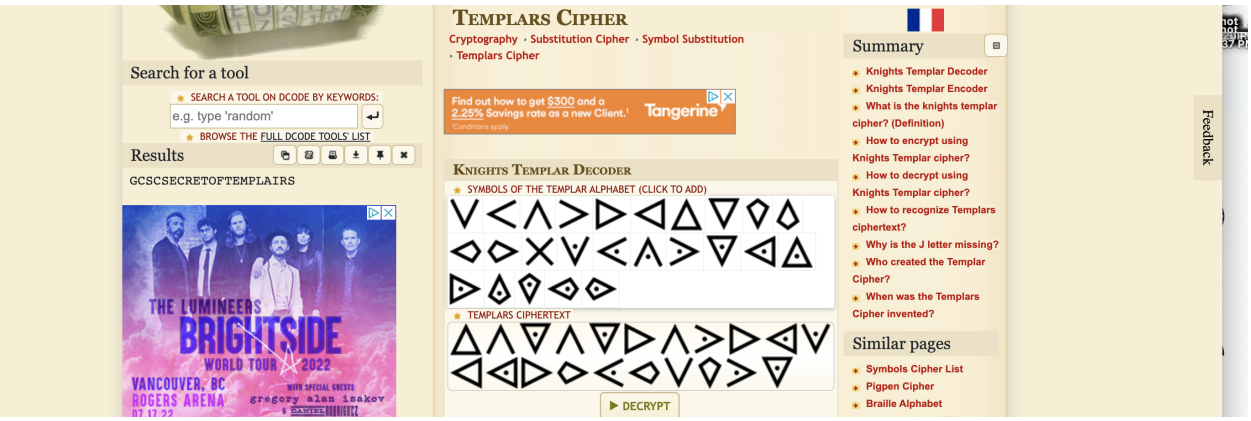
> We have identified an account and send you some information to guess its password.
> - pseudonym: CyberBadCorp
> - email: cyberbadcorp@gmail.com
> - registration token: 20022022

- The hint is that this account is similar to how participants login. From our login creds, we guessed that the password would be in format `GCSC2022_CyberBadCorp20022022cyberbadcorp@gmail.com20022022`.

Libya

- We are given the following image

- Searching “Cipher with triangle and dot” reveals this cipher: `Knights Templar Code Cipher`
- Using a tool to decode the flag.



Guinea

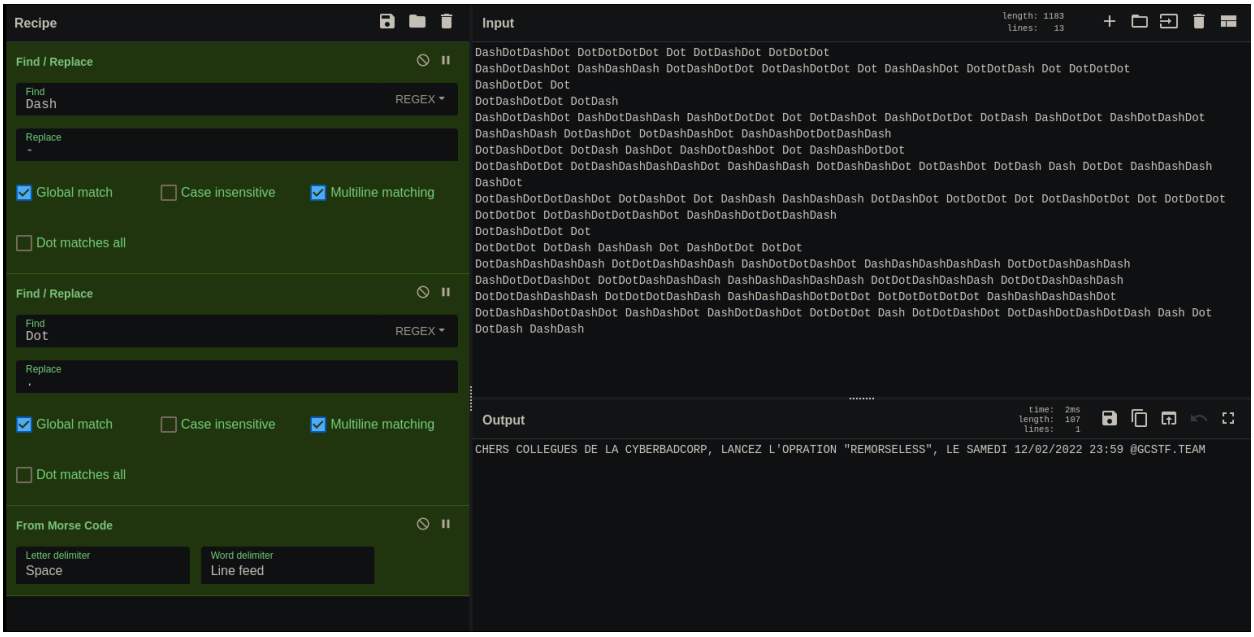
1. What is Guinea's ccTLD?
    - .gn
2. What is the codification of the law relating to Cybersecurity and the Protection of Personal Data in Guinea?
    - LOI L-2016-037-AN
3. What is the time zone of Guinea?
    - GMT+0

Mali

1. What are the three elements that make up the CIA triad?
    - Confidentiality-Integrity-Availability
2. We are interested in the telecommunications standard operating on the 2.4 GHz frequency whose name is inspired by the king of the Vikings. Give the name followed by its latest version.
    - Bluetooth v5.2
3. Among the following technologies, which is the most appropriate to secure your Wifi access: WEP, WPA, WPS, WPA2? The answer is its IEEE standard.
    - It is WPA2 so answer is IEEE 802.11i
4. What does each of the following acronyms mean: AV, FW, IDS, IPS, SIMS, SOC and VPN?
    - Antivirus - Firewall - Intrusion Detection System - Intrusion Prevention System - Security Information Management System - Security Operation Center - Virtual Private Network
5. I use a MacBook laptop for my daily activities. I use Linux to browse questionable sites. I use the LinkedIn social network. In this case, I cannot be hacked at all. True or false?
    - Obviously False, but we need to input French answer so Faux.

Algeria

- We are given a link to a file
- Using Cyber-Chef we can replace `Dot` with `.` and `Dash` with `-` to get Morse code
- Converting the Morse code, we get the message



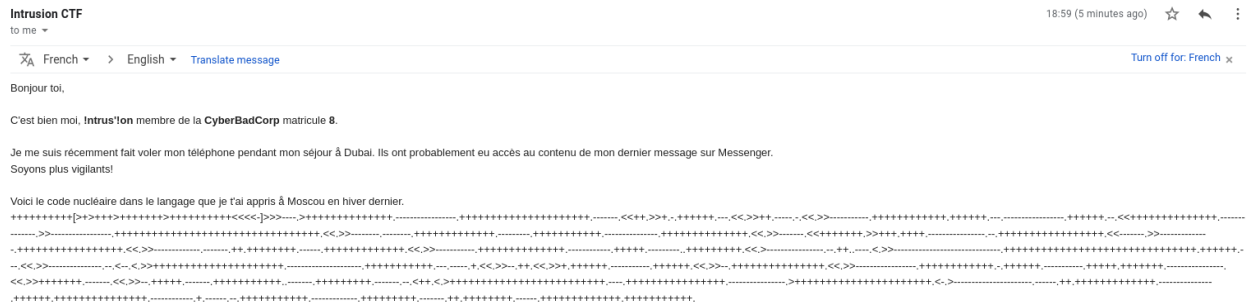
Morocco

- This challenge had to be solved after solving the `Algeria` challenge
- We get the following message by solving the Algeria challenge:

```
CHERS COLLEGUES DE LA CYBERBADCORP, LANCEZ L'OPRATION "REMORSELESS", LE SAMEDI 12/02/2022 23:59 @GCSTF.TEAM
```

- After doing an username enumeration of `GCSTF.TEAM`, we can find the profile on Facebook
- By sending them a message, we get this:

- After sending them a number between 0 - 10, we eventually get an email address: `am.intrusion@gmail.com`
- If we send an email to that id, we get this reply:

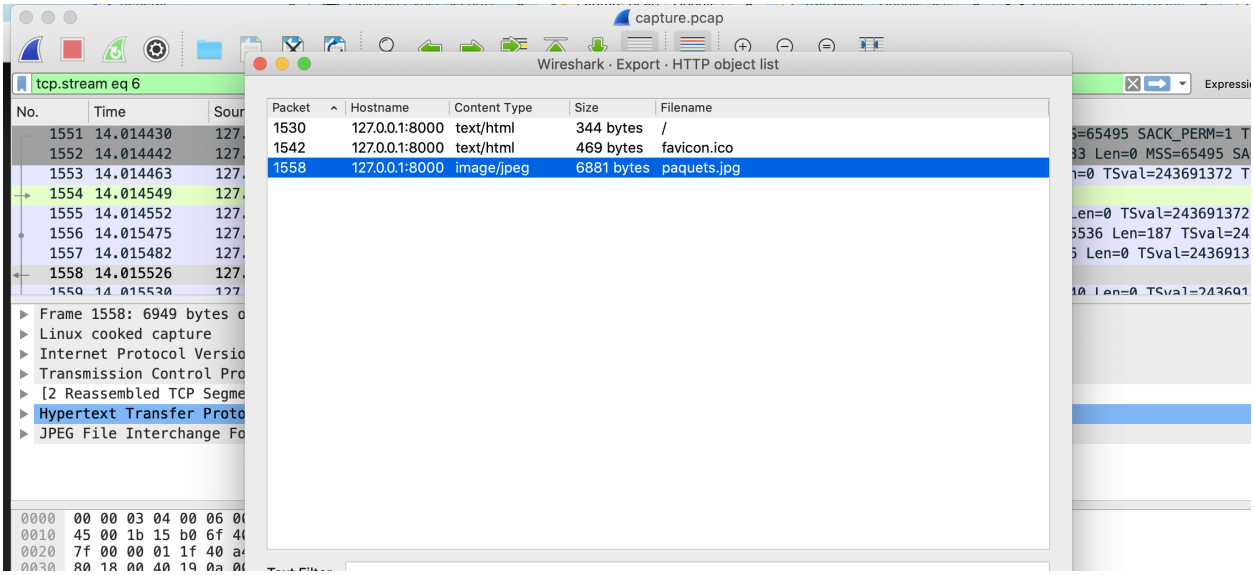


- And finally decoding the `brainfuck` code we get this message with the flag.

```
Bravo pour ton courage. Tu merites l'oscar du hacker guineen 2022. Pour ca, valide ce flag et continue le challenge.
GCSC{Be_an_ethical_hacker}
```

Somalia

- In this challenge we got a pcap file. Opening in WireShark, we first filter `http` and look for HTTP-related files (`File - Export object - HTTP`):



- The `paquest.jpg` looks suspicious, and indeed, it contains the flag.

India

- For this challenge we are asked to guess the admin's email and there was hint saying that he contacted us before with it so we just remembered that in Morocco challenge we got an email before so we just submitted that: `GCSC2022{am.intrusion@gmail.com}`

Brazil

- We are given a drive link `https://drive.google.com/drive/folders/1x39L7UX9djNf_cWXHfs952ZnaKyp0M-D?usp=sharing` which contains a zip file so let's download it. Looking at the task, we have 5 questions to answer and concatenating answers to all questions gives us the flag.
    - What is the name of the hacker?
    - What is the comment left by the hacker?
    - What is the password to extract the file exfiltrated by the hacker?
    - What is the original name and password of this previously extracted archived file? -Expected format: file.ext, password
    - What is the proof of the intrusion (flag)?
- Extracting that zip file we get a jpeg named: `mamadi-doumbouya.jpg`
- Question 1: With `exiftool` we can check the metadata of the image.

```
ExifTool Version Number         : 12.16
File Name                       : mamadi-doumbouya.jpg
Directory                       : .
File Size                       : 79 KiB
File Modification Date/Time     : 2022:02:20 08:14:46-05:00
File Access Date/Time           : 2022:02:20 14:38:13-05:00
File Inode Change Date/Time     : 2022:02:21 02:14:02-05:00
File Permissions                : rw-r--r--
File Type                       : JPEG
File Type Extension             : jpg
MIME Type                       : image/jpeg
JFIF Version                    : 1.01
Resolution Unit                 : inches
X Resolution                    : 96
Y Resolution                    : 96
Exif Byte Order                 : Big-endian (Motorola, MM)
Image Description               : GCSC
Make                            : Iphone 15
Camera Model Name               : Iphone 15
Artist                          : John TRUMP
XP Title                        : GCSC
XP Comment                      : 55 6e 33 5f 31 6d 34 67 65 5f 70 33 75 74 5f 63 34 63 68 33 72 5f 34 75 74 72 33 5f 63 68 30 73 33
XP Author                       : John TRUMP
XP Subject                      : GCSC
Padding                         : (Binary data 2060 bytes, use -b option to extract)
About                           : uuid:faf5bdd5-ba3d-11da-ad31-d33d75182f1b
Creator                         : John TRUMP
Title                           : GCSC
Description                     : GCSC
Image Width                     : 768
Image Height                    : 694
Encoding Process                : Baseline DCT, Huffman coding
Bits Per Sample                 : 8
Color Components                : 3
Y Cb Cr Sub Sampling            : YCbCr4:2:0 (2 2)
Image Size                      : 768x694
Megapixels                      : 0.533
```

- From this we can find `Answer1 -> John TRUMP`
- Question 2: now checking the XP comment we can see there are bunch of hex-decimal characters so let's unhex it.

```
$ python -c 'print(bytes.fromhex("55 6e 33 5f 31 6d 34 67 65 5f 70 33 75 74 5f 63 34 63 68 33 72 5f 34 75 74 72 33 5f 63 68 30 73 33".repla
-> b'Un3_1m4ge_p3ut_c4ch3r_4utr3_ch0s3'
```

- `Answer2 -> Un3_1m4ge_p3ut_c4ch3r_4utr3_ch0s3`
- Question 3: now there looks like a file hiding inside the image so let's check for that. I tried `steghide` with no password, binwalk, foremost but no output, so then I decided to brute it with `stegseek`.

```
$ stegseek mamadi-doumbouya.jpg /usr/share/wordlists/rockyou.txt
StegSeek 0.6 - https://github.com/RickdeJager/StegSeek
```

```
[i] Found passphrase: "ilikelinux"
[i] Original filename: "secret.zip".
[i] Extracting to "mamadi-doumbouya.jpg.out".
```

- We have the file called `secret.zip` which can be extacted by password `ilikelinux`. `Answer3 -> ilikelinux`
- Question 4: file name is `secret.zip`. Part2 -> password so let's find password. Let's extract secret.zip first. (I know we have it as `mamadi-doumbouya.jpg.out` but to avoid confusion I am extracting it again)

```
$ steghide extract -sf mamadi-doumbouya.jpg
Enter passphrase:
wrote extracted data to "secret.zip".
```

- Now let's crack the password for the protected zip file. We use `john` to generate has of the file and `rockyou` to get the password `love2linux`.
- Question 5: so now let's extract the zip file and see its content.

```
$ 7z x secret.zip

7-Zip [64] 16.02 : Copyright (c) 1999-2016 Igor Pavlov : 2016-05-21
p7zip Version 16.02 (locale=en_US.UTF-8,Utf16=on,HugeFiles=on,64 bits,2 CPUs Intel(R) Pentium(R) CPU  N3700  @ 1.60GHz (406C3),ASM,AES-NI)

Scanning the drive for archives:
1 file, 282 bytes (1 KiB)

Extracting archive: secret.zip
--
Path = secret.zip
Type = zip
Physical Size = 282


Enter password (will not be echoed):
Everything is Ok

Size:       88
Compressed: 282

$ ls
Doumbouya.zip  file  flag.txt  hash  mamadi-doumbouya.jpg  mamadi-doumbouya.jpg.out  secret.zip
```

- And we have a file called `flag.txt` with base64 `TDRfc3QzZzRuMGdyNHBoMTNfcDNybTN0X2QzX2M0Y2gzUl9kM3NfbTNzczRnZXMh`. Converting it gives flag `L4_st3g4n0gr4ph13_p3rm3t_d3_c4ch3R_d3s_m3ss4ges!`
- Combining all the above parts we get the final flag.

Sudan

- We are given a link to drive where it has system hive and sam hive of a windows machine. I downloaded them and dumped hashes.

```
$ 7z x some.zip

7-Zip [64] 16.02 : Copyright (c) 1999-2016 Igor Pavlov : 2016-05-21
p7zip Version 16.02 (locale=en_US.UTF-8,Utf16=on,HugeFiles=on,64 bits,2 CPUs Intel(R) Pentium(R) CPU  N3700  @ 1.60GHz (406C3),ASM,AES-N
I)

Scanning the drive for archives:
1 file, 2149923 bytes (2100 KiB)

Extracting archive: some.zip
--
Path = some.zip
Type = zip
Physical Size = 2149923

Everything is Ok
```

```
Files: 2
Size:       11386880
Compressed: 2149923

$ ls
 some.zip  'SYS et SAM'
```

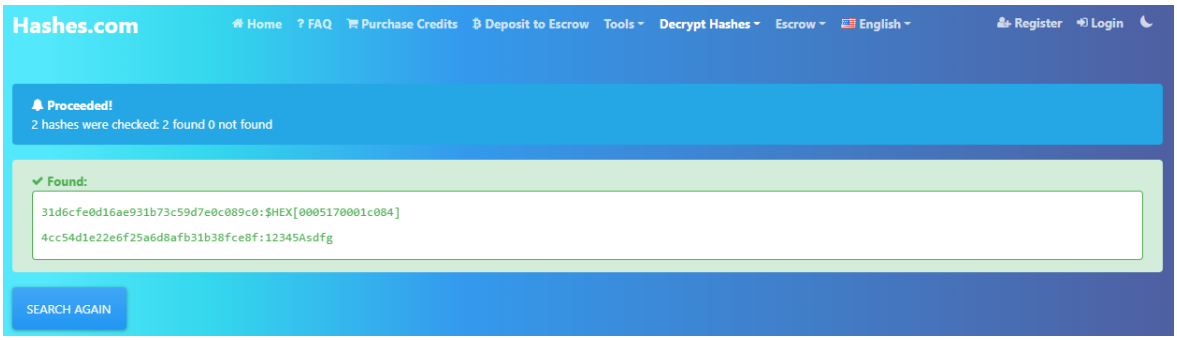
- Looks like we have a folder let's get into it and check it's content.

```
➜  Writeup cd SYS\\ et\\ SAM
➜  SYS et SAM ls
sam  system
```

- We can now dump the hashes:

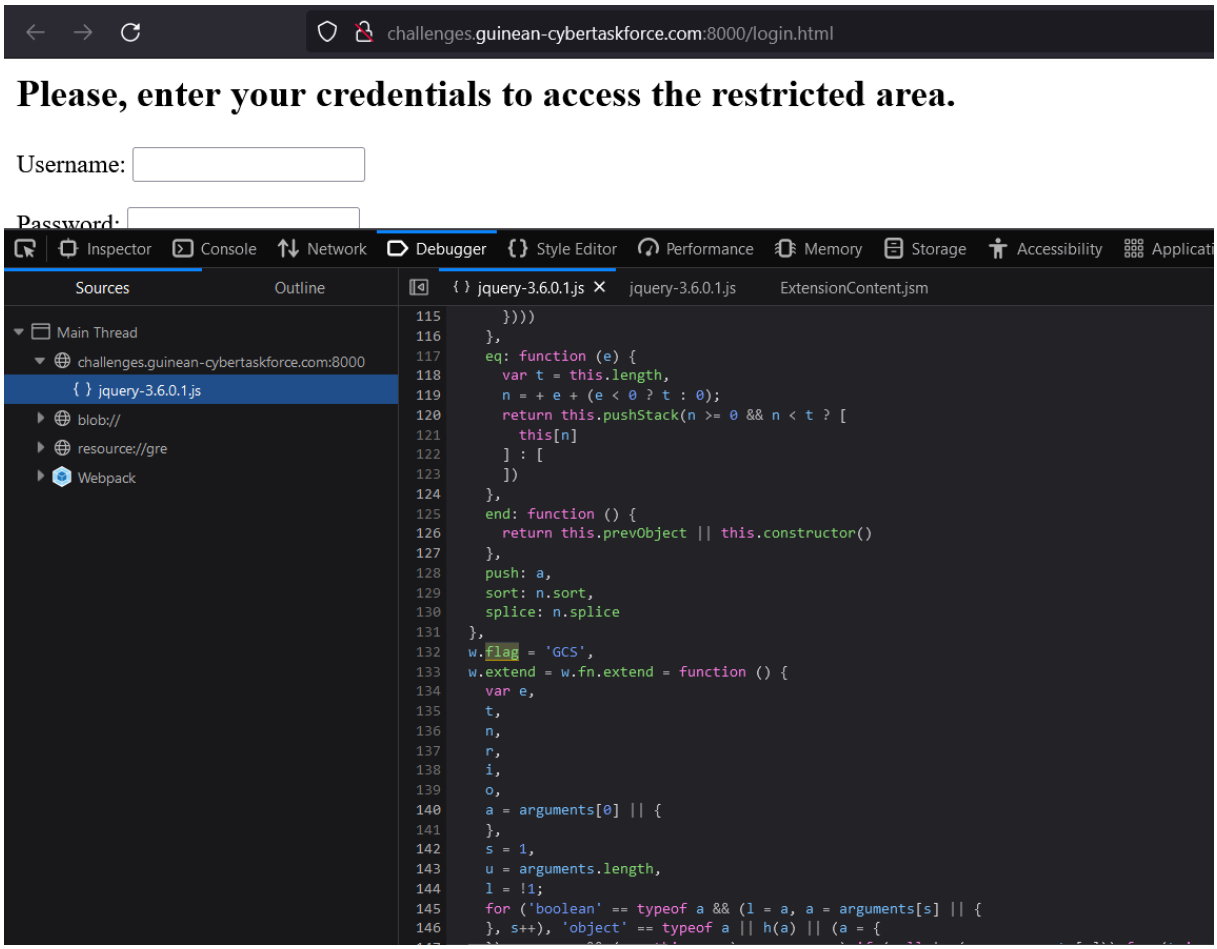
```
➜  SYS et SAM secretsdump.py -system system -sam sam local
Impacket v0.9.24.dev1+20210827.162957.5aa97fa7 - Copyright 2021 SecureAuth Corporation

[*] Target system bootKey: 0x586e3243d96945fe3f65221f2a56be38
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
Administrator:500:aad3b435b51404eeaad3b435b51404ee:4cc54d1e22e6f25a6d8afb31b38fce8f:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
[*] Cleaning up...
➜  SYS et SAM
```

- We have two hashes, one for guest and one for administrator. From experience I can say it for sure that guest hash is just empty hash as that is ntlm of a blank. For admin, we can try two ways:
  1. Crack the hash locally from that salt in description (which would be extensive for CPU)
  2. Use online cracking tools, which is preferred. Website → https://hashes.com/en/decrypt/hash and we have the password and as assumed the first one is just the blank hash.
     `FLAG -> GCSC2022{12345Asdfg}`

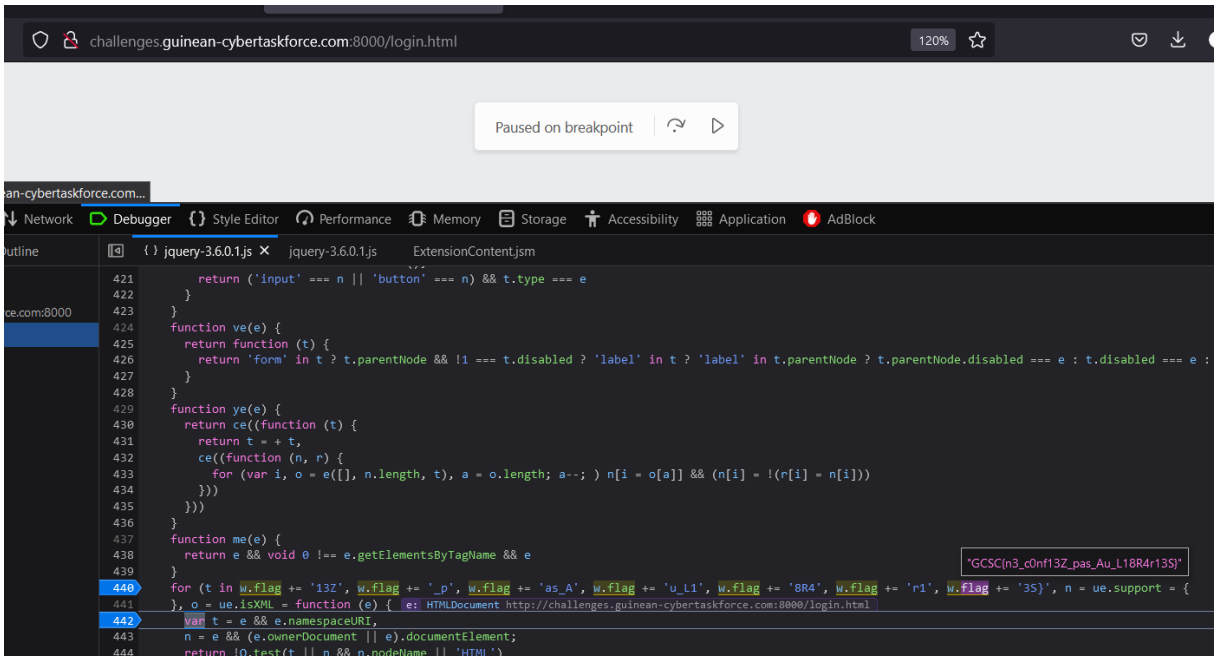


France

- This is a web challenge we start and get a 403 page. After several tries and errors we run a fuzzer with several wordlists and end up finding /login.html and /login.php. Once opened we can see the hint referring to jQuery so our first thought was checking the jQuery JS file that was loading in the page and we can notice a `w.flag` as shown below:



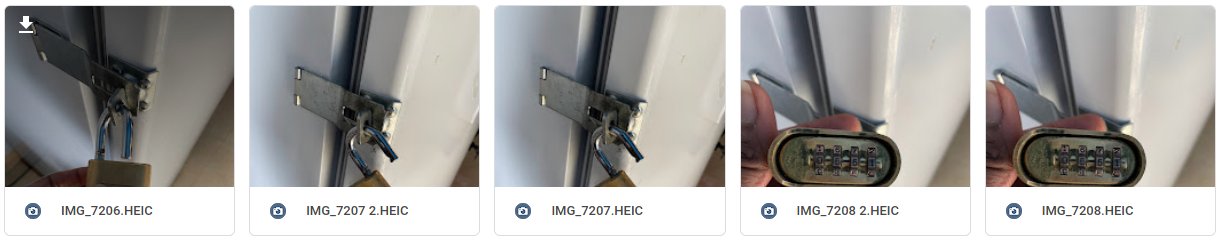
So I tried following and building up the flag by using a nice debugger trick. I just added 2 breakpoints after the last call of `w.flag` and just refresh the page and we can see the flag through the debugger:



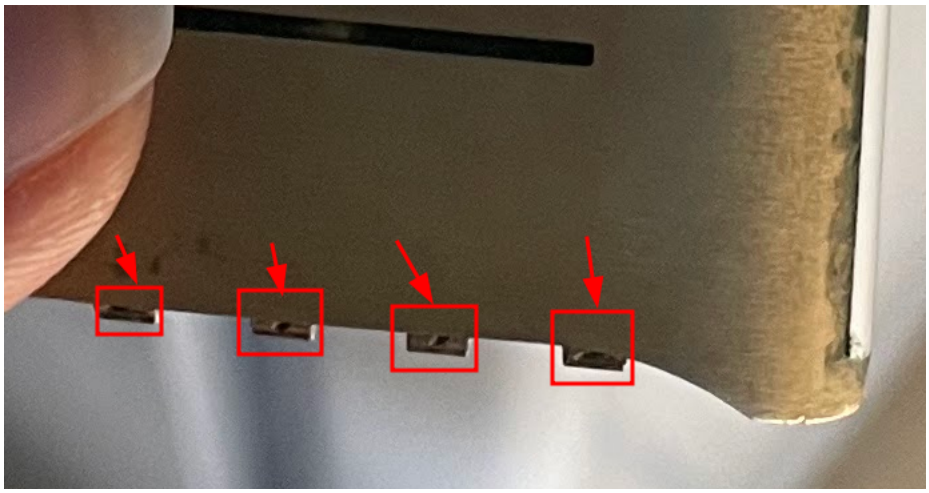
We submit the flag based on the format `GCSC2022{n3_c0nf13Z_pas_Au_L18R4r13S}`

Niger

- We are given a lot of photos and we have to find the pin that unlocks the lock. Going to the link and checking photos we just see 3 photos that has lock unlocked.



- Out of those 3, 2 are useless but 1 of them (`IMG_7206.heic`) does flash just a little bit of the padlock. Checking that photo we can see some numbers that would be one more than the real pin. We zoom into photo and can see the numbers clearly thanks to high resolution of photo.



- From this photo we can see that it is 2772. So the pin should be 1661. `FLAG -> GCSC2022{1661}`