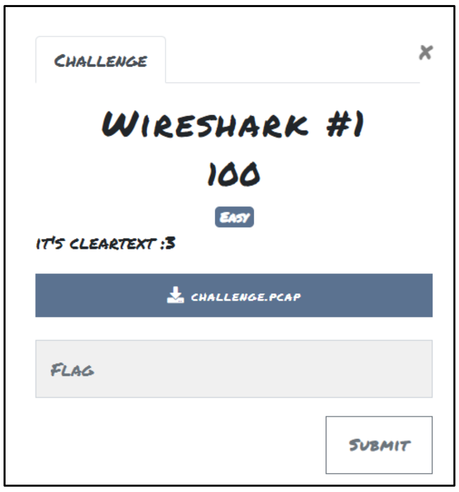
GCTF 2023 Writeups

Team	**:** $ /bin/cat
Members	**:** k07_ & latt3

Forensics

Wireshark #1


CHALLENGE

WIRESHARK #1

100

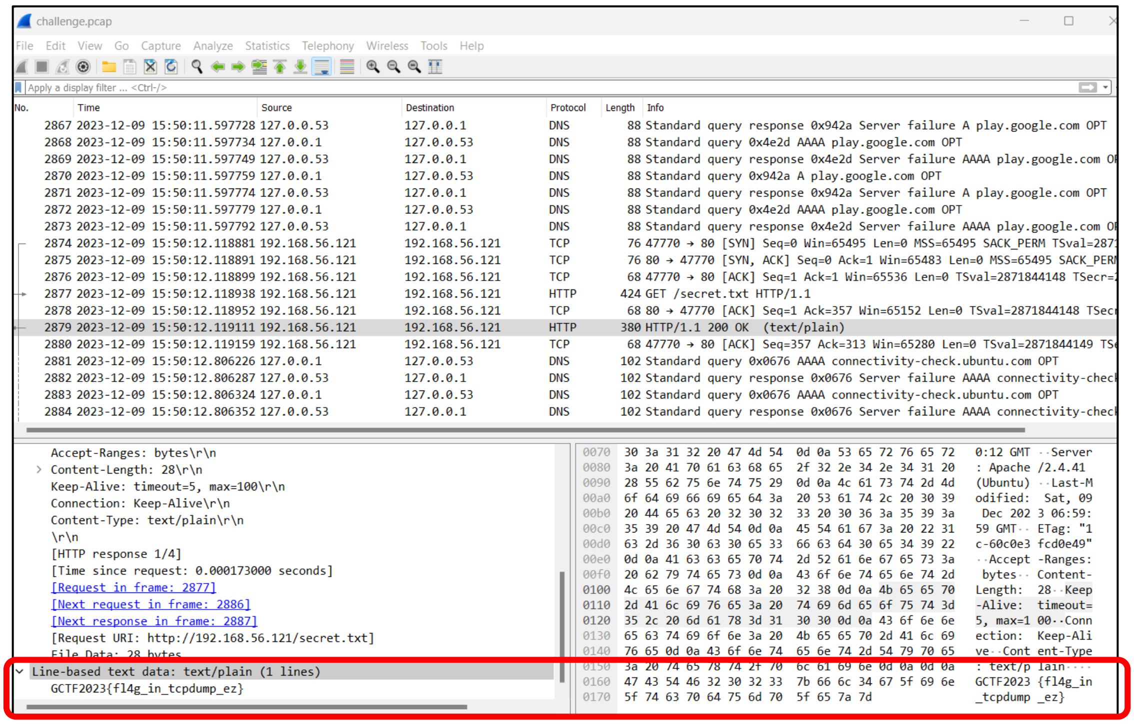
EASY

IT'S CLEARTEXT :3

Attachment: challenge.pcap

Solution:

At first, I use Wireshark to randomly scrolled and clicked to search for the flag. Luckily, I found it here:

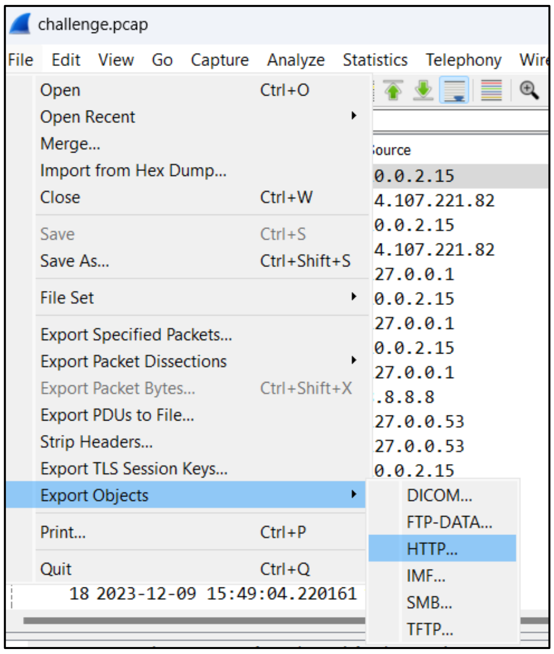

challenge.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

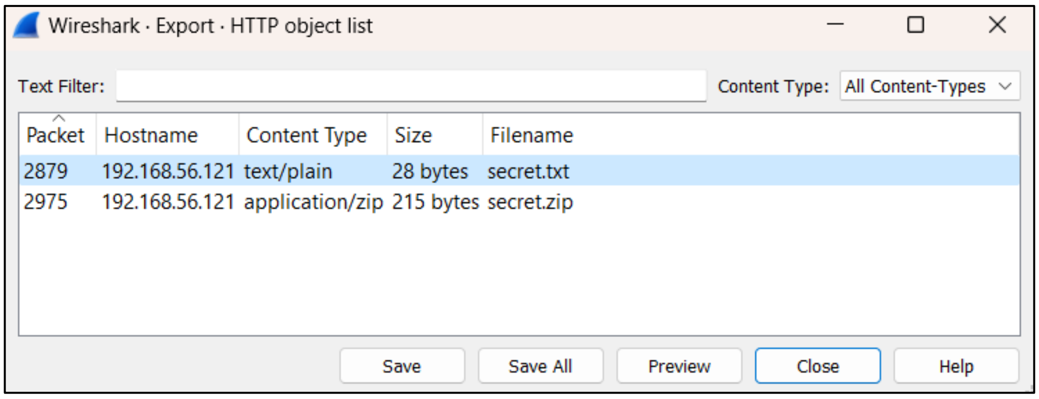
```
No.  Time                        Source          Destination     Protocol Length Info
2867 2023-12-09 15:50:11.597728 127.0.0.53      127.0.0.1       DNS       88 Standard query response 0x942a Server failure A play.google.com OPT
2868 2023-12-09 15:50:11.597734 127.0.0.1       127.0.0.53      DNS       88 Standard query 0x4e2d AAAA play.google.com OPT
2869 2023-12-09 15:50:11.597749 127.0.0.53      127.0.0.1       DNS       88 Standard query response 0x4e2d Server failure AAAA play.google.com O
2870 2023-12-09 15:50:11.597759 127.0.0.1       127.0.0.53      DNS       88 Standard query 0x942a A play.google.com OPT
2871 2023-12-09 15:50:11.597774 127.0.0.53      127.0.0.1       DNS       88 Standard query response 0x942a Server failure A play.google.com OPT
2872 2023-12-09 15:50:11.597779 127.0.0.1       127.0.0.53      DNS       88 Standard query 0x4e2d AAAA play.google.com OPT
2873 2023-12-09 15:50:11.597792 127.0.0.53      127.0.0.1       DNS       88 Standard query response 0x4e2d Server failure AAAA play.google.com O
2874 2023-12-09 15:50:12.118881 192.168.56.121  192.168.56.121  TCP       76 47770 → 80 [SYN] Seq=0 Win=65495 Len=0 MSS=65495 SACK_PERM TSval=287
2875 2023-12-09 15:50:12.118891 192.168.56.121  192.168.56.121  TCP       76 80 → 47770 [SYN, ACK] Seq=0 Ack=1 Win=65483 Len=0 MSS=65495 SACK_PER
2876 2023-12-09 15:50:12.118899 192.168.56.121  192.168.56.121  TCP       68 47770 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0 TSval=2871844148 TSecr=
2877 2023-12-09 15:50:12.118938 192.168.56.121  192.168.56.121  HTTP     424 GET /secret.txt HTTP/1.1
2878 2023-12-09 15:50:12.118952 192.168.56.121  192.168.56.121  TCP       68 80 → 47770 [ACK] Seq=1 Ack=357 Win=65152 Len=0 TSval=2871844148 TSec
2879 2023-12-09 15:50:12.119111 192.168.56.121  192.168.56.121  HTTP     380 HTTP/1.1 200 OK  (text/plain)
2880 2023-12-09 15:50:12.119159 192.168.56.121  192.168.56.121  TCP       68 47770 → 80 [ACK] Seq=357 Ack=313 Win=65280 Len=0 TSval=2871844149 TS
2881 2023-12-09 15:50:12.806226 127.0.0.1       127.0.0.53      DNS      102 Standard query 0x0676 AAAA connectivity-check.ubuntu.com OPT
2882 2023-12-09 15:50:12.806287 127.0.0.53      127.0.0.1       DNS      102 Standard query response 0x0676 Server failure AAAA connectivity-chec
2883 2023-12-09 15:50:12.806324 127.0.0.1       127.0.0.53      DNS      102 Standard query 0x0676 AAAA connectivity-check.ubuntu.com OPT
2884 2023-12-09 15:50:12.806352 127.0.0.53      127.0.0.1       DNS      102 Standard query response 0x0676 Server failure AAAA connectivity-chec
```

```
    Accept-Ranges: bytes\r\n
  > Content-Length: 28\r\n
    Keep-Alive: timeout=5, max=100\r\n
    Connection: Keep-Alive\r\n
    Content-Type: text/plain\r\n
    \r\n
    [HTTP response 1/4]
    [Time since request: 0.000173000 seconds]
    [Request in frame: 2877]
    [Next request in frame: 2886]
    [Next response in frame: 2887]
    [Request URI: http://192.168.56.121/secret.txt]
    File Data: 28 bytes
v Line-based text data: text/plain (1 lines)
    GCTF2023{fl4g_in_tcpdump_ez}
```

```
0070  30 3a 31 32 20 47 4d 54  0d 0a 53 65 72 76 65 72   0:12 GMT ··Server
0080  3a 20 41 70 61 63 68 65  2f 32 2e 34 2e 34 31 20   : Apache /2.4.41 
0090  28 55 62 75 6e 74 75 29  0d 0a 4c 61 73 74 2d 4d   (Ubuntu) ··Last-M
00a0  6f 64 69 66 69 65 64 3a  20 53 61 74 2c 20 30 39   odified:  Sat, 09
00b0  20 44 65 63 20 32 30 32  33 20 30 36 3a 35 39 3a    Dec 202 3 06:59:
00c0  35 39 20 47 4d 54 0d 0a  45 54 61 67 3a 20 22 31   59 GMT·· ETag: "1
00d0  63 2d 36 30 63 30 65 33  66 63 64 30 65 34 39 22   c-60c0e3 fcd0e49"
00e0  0d 0a 41 63 63 65 70 74  2d 52 61 6e 67 65 73 3a   ··Accept -Ranges:
00f0  20 62 79 74 65 73 0d 0a  43 6f 6e 74 65 6e 74 2d    bytes·· Content-
0100  4c 65 6e 67 74 68 3a 20  32 38 0d 0a 4b 65 65 70   Length:  28··Keep
0110  2d 41 6c 69 76 65 3a 20  74 69 6d 65 6f 75 74 3d   -Alive:  timeout=
0120  35 2c 20 6d 61 78 3d 31  30 30 0d 0a 43 6f 6e 6e   5, max=1 00··Conn
0130  65 63 74 69 6f 6e 3a 20  4b 65 65 70 2d 41 6c 69   ection:  Keep-Ali
0140  76 65 0d 0a 43 6f 6e 74  65 6e 74 2d 54 79 70 65   ve··Cont ent-Type
0150  3a 20 74 65 78 74 2f 70  6c 61 69 6e 0d 0a 0d 0a   : text/p lain····
0160  47 43 54 46 32 30 32 33  7b 66 6c 34 67 5f 69 6e   GCTF2023 {fl4g_in
0170  5f 74 63 70 64 75 6d 70  5f 65 7a 7d               _tcpdump _ez}
```

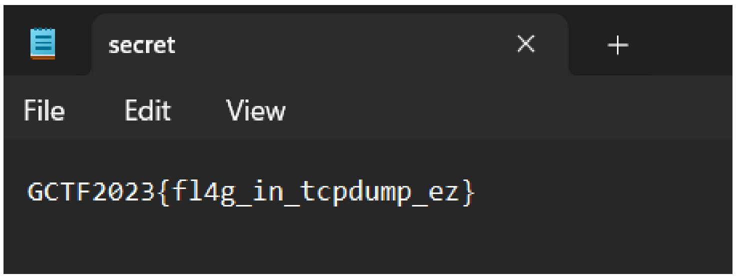
Afterward, I realized that we could also export the TXT file by clicking “File”, “Export Objects”, and then “HTTP”.




Wireshark · Export · HTTP object list
Text Filter:
Content Type: All Content-Types
Save
Save All
Preview
Close
Help

Packet	Hostname	Content Type	Size	Filename
2879	192.168.56.121	text/plain	28 bytes	secret.txt
2975	192.168.56.121	application/zip	215 bytes	secret.zip

Click on packet 2879, and then click “Preview”. We can see the flag in the secret.txt file.

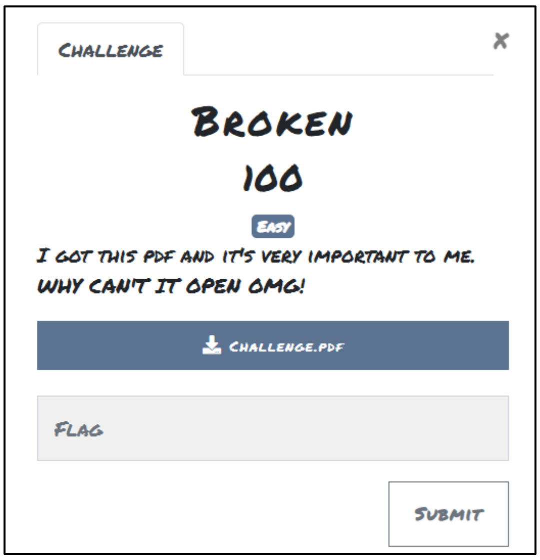


```
GCTF2023{fl4g_in_tcpdump_ez}
```

Tool: Wireshark

Flag: GCTF2023{fl4g_in_tcpdump_ez}

Broken


Challenge

Broken

100

Easy

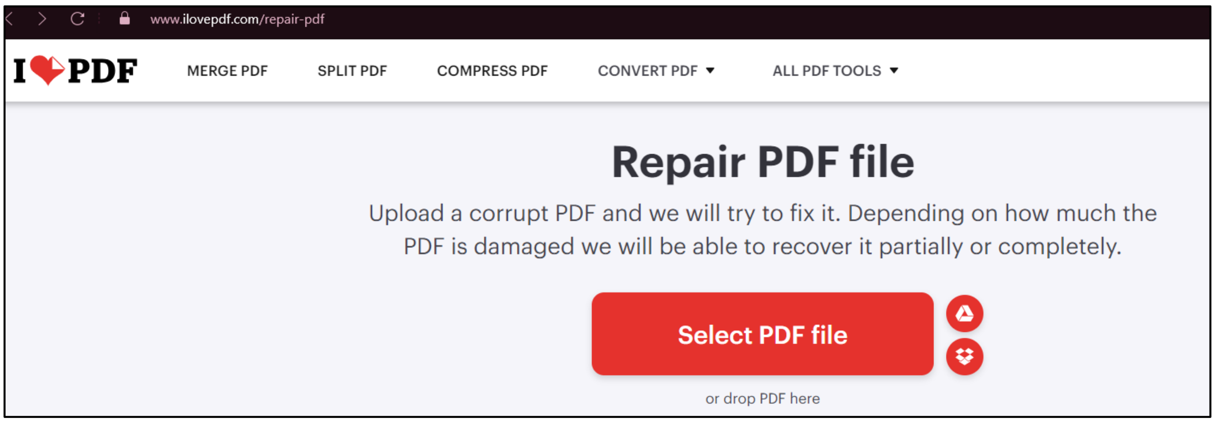
I got this PDF and it's very important to me. Why can't it open OMG!

Attachment: Challenge.pdf

Solution:

The provided PDF file is broken, and we can’t open it.

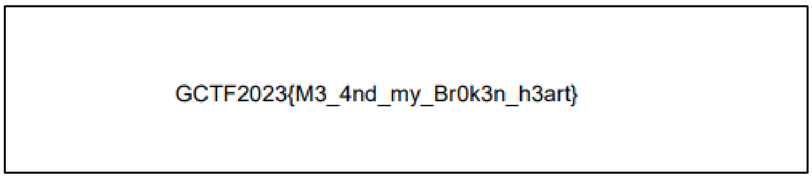
Use the online tool https://www.ilovepdf.com/repair-pdf to fix the PDF file.


www.ilovepdf.com/repair-pdf

Repair PDF file

Upload a corrupt PDF and we will try to fix it. Depending on how much the PDF is damaged we will be able to recover it partially or completely.

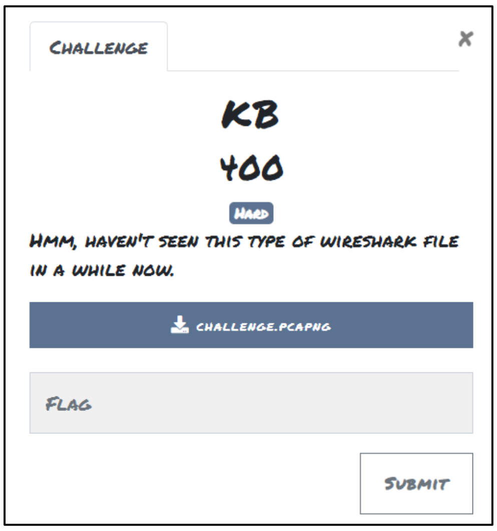
Now, we can open the PDF file successfully, and the flag is in the repaired PDF file.


GCTF2023{M3_4nd_my_Br0k3n_h3art}

Tool: iLovePDF - https://www.ilovepdf.com/repair-pdf

Flag: GCTF2023{M3_4nd_my_Br0k3n_h3art}

KB



CHALLENGE

KB

400

HARD

HMM, HAVEN'T SEEN THIS TYPE OF WIRESHARK FILE IN A WHILE NOW.

CHALLENGE.PCAPNG

FLAG

SUBMIT

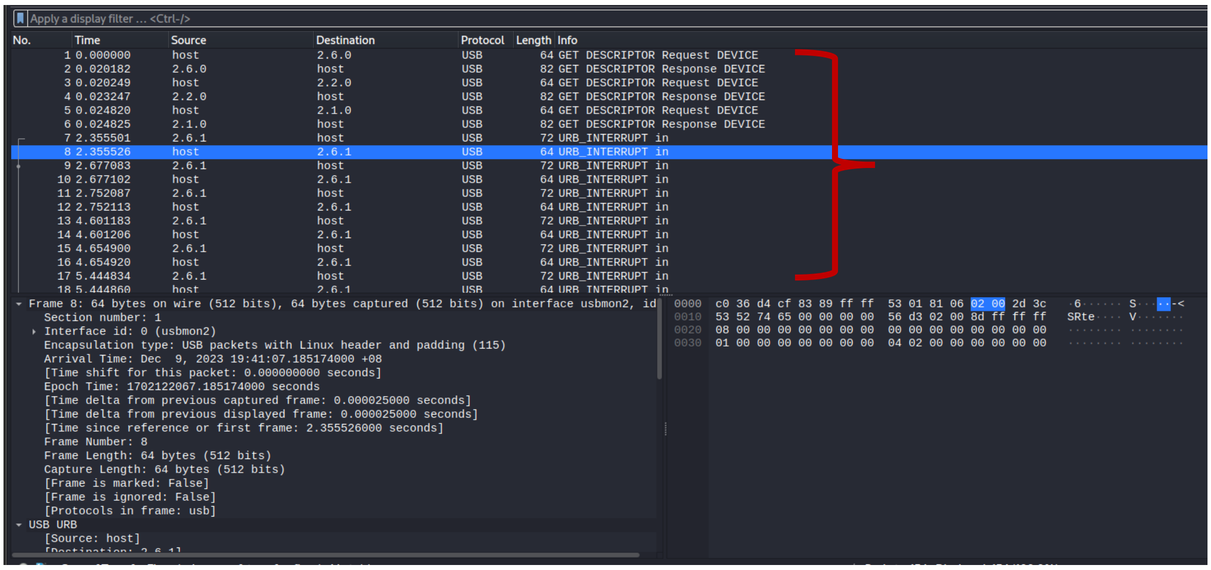
Attachment: challenge.pcapng

Solution:

Based on the image above, this challenge is with a PCAP file which means that requires to use of Wireshark or TShark.

- PCAP which means packet capture and this kind of file used to store the captured packet of data into a readable file.

First, open the PCAP file with wireshark ***$ wireshark challenge.pcapng*** and determine the information in the file.



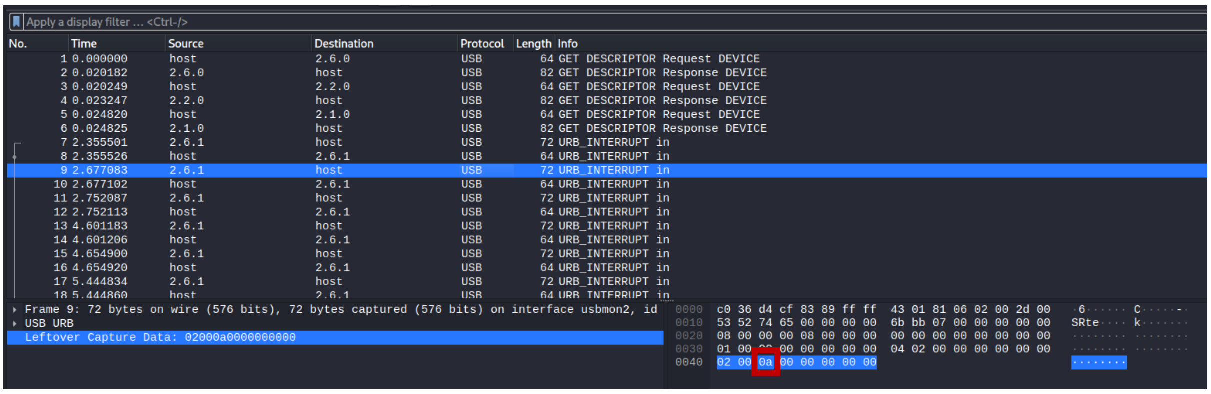
```
Apply a display filter ... <Ctrl-/>
No.   Time       Source   Destination  Protocol  Length Info
    1 0.000000   host     2.6.0        USB       64 GET DESCRIPTOR Request DEVICE
    2 0.020182   2.6.0    host         USB       82 GET DESCRIPTOR Response DEVICE
    3 0.020249   host     2.2.0        USB       64 GET DESCRIPTOR Request DEVICE
    4 0.023247   2.2.0    host         USB       82 GET DESCRIPTOR Response DEVICE
    5 0.024820   host     2.1.0        USB       64 GET DESCRIPTOR Request DEVICE
    6 0.024825   2.1.0    host         USB       82 GET DESCRIPTOR Response DEVICE
    7 2.355501   2.6.1    host         USB       72 URB_INTERRUPT in
    8 2.355526   host     2.6.1        USB       64 URB_INTERRUPT in
    9 2.677083   2.6.1    host         USB       72 URB_INTERRUPT in
   10 2.677102   host     2.6.1        USB       64 URB_INTERRUPT in
   11 2.752087   2.6.1    host         USB       72 URB_INTERRUPT in
   12 2.752113   host     2.6.1        USB       64 URB_INTERRUPT in
   13 4.601183   2.6.1    host         USB       72 URB_INTERRUPT in
   14 4.601206   host     2.6.1        USB       64 URB_INTERRUPT in
   15 4.654900   2.6.1    host         USB       72 URB_INTERRUPT in
   16 4.654920   host     2.6.1        USB       64 URB_INTERRUPT in
   17 5.444834   2.6.1    host         USB       72 URB_INTERRUPT in
   18 5.444860   host     2.6.1        USB       64 URB_INTERRUPT in

Frame 8: 64 bytes on wire (512 bits), 64 bytes captured (512 bits) on interface usbmon2, id
    Section number: 1
  Interface id: 0 (usbmon2)
    Encapsulation type: USB packets with Linux header and padding (115)
    Arrival Time: Dec  9, 2023 19:41:07.185174000 +08
    [Time shift for this packet: 0.000000000 seconds]
    Epoch Time: 1702122067.185174000 seconds
    [Time delta from previous captured frame: 0.000025000 seconds]
    [Time delta from previous displayed frame: 0.000025000 seconds]
    [Time since reference or first frame: 2.355526000 seconds]
    Frame Number: 8
    Frame Length: 64 bytes (512 bits)
    Capture Length: 64 bytes (512 bits)
    [Frame is marked: False]
    [Frame is ignored: False]
    [Protocols in frame: usb]
USB URB
    [Source: host]

0000  c0 36 d4 cf 83 89 ff ff  53 01 81 06 02 00 2d 3c   ·6······ S·····-<
0010  53 52 74 65 00 00 00 00  56 d3 02 00 8d ff ff ff   SRte···· V·······
0020  08 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0030  01 00 00 00 00 00 00 00  04 02 00 00 00 00 00 00   ········ ········
```

As you can see, this PCAP file contains the communication through the USB. In the Info column noticed that there are two different info. The relevant packet of the info that you should looking for were **“URB_INTERRUPT in”** packet.

Noted:

- Is it the Protocol is USB
- The Source and Destination which is different and is using the two-way communication

The info with **“URB_INTERRUPT in”** packet consists of two type of packets which different in Source, Destination and the length. From the length column you could notice that two different types of packets which have the range of 8 bytes. Those packtes that with extra 8 bytes will have the **“Leftover Capture Data”** field.



```
Apply a display filter ... <Ctrl-/>
No.   Time        Source   Destination  Protocol  Length Info
    1 0.000000    host     2.6.0        USB       64 GET DESCRIPTOR Request DEVICE
    2 0.020182    2.6.0    host         USB       82 GET DESCRIPTOR Response DEVICE
    3 0.020249    host     2.2.0        USB       64 GET DESCRIPTOR Request DEVICE
    4 0.023247    2.2.0    host         USB       82 GET DESCRIPTOR Response DEVICE
    5 0.024820    host     2.1.0        USB       64 GET DESCRIPTOR Request DEVICE
    6 0.024825    2.1.0    host         USB       82 GET DESCRIPTOR Response DEVICE
    7 2.355501    2.6.1    host         USB       72 URB_INTERRUPT in
    8 2.355526    host     2.6.1        USB       64 URB_INTERRUPT in
    9 2.677083    2.6.1    host         USB       72 URB_INTERRUPT in
   10 2.677102    host     2.6.1        USB       64 URB_INTERRUPT in
   11 2.752087    2.6.1    host         USB       72 URB_INTERRUPT in
   12 2.752113    host     2.6.1        USB       64 URB_INTERRUPT in
   13 4.601183    2.6.1    host         USB       72 URB_INTERRUPT in
   14 4.601206    host     2.6.1        USB       64 URB_INTERRUPT in
   15 4.654900    2.6.1    host         USB       72 URB_INTERRUPT in
   16 4.654920    host     2.6.1        USB       64 URB_INTERRUPT in
   17 5.444834    2.6.1    host         USB       72 URB_INTERRUPT in
   18 5.444860    host     2.6.1        USB       64 URB_INTERRUPT in

› Frame 9: 72 bytes on wire (576 bits), 72 bytes captured (576 bits) on interface usbmon2, id
› USB URB
  Leftover Capture Data: 02000a0000000000

0000  c0 36 d4 cf 83 89 ff ff  43 01 81 06 02 00 2d 00   ·6······ C·····-·
0010  53 52 74 65 00 00 00 00  6b bb 07 00 00 00 00 00   SRte···· k·······
0020  08 00 00 00 08 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0030  01 00 00 00 00 00 00 00  04 02 00 00 00 00 00 00   ········ ········
0040  02 00 0a 00 00 00 00 00                            ········
```

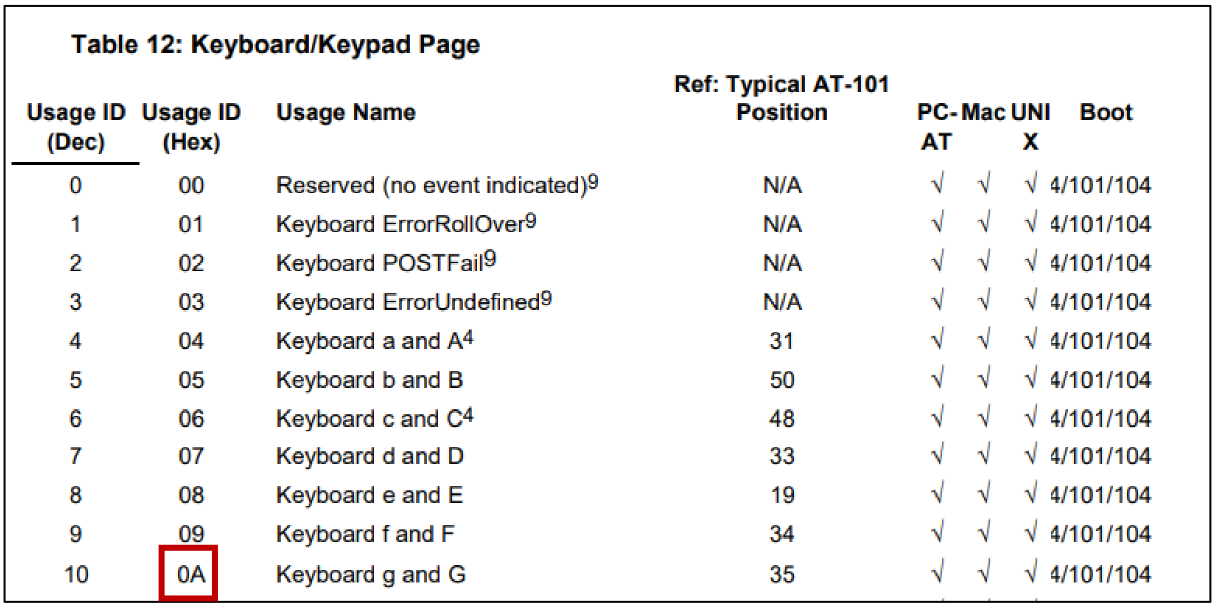
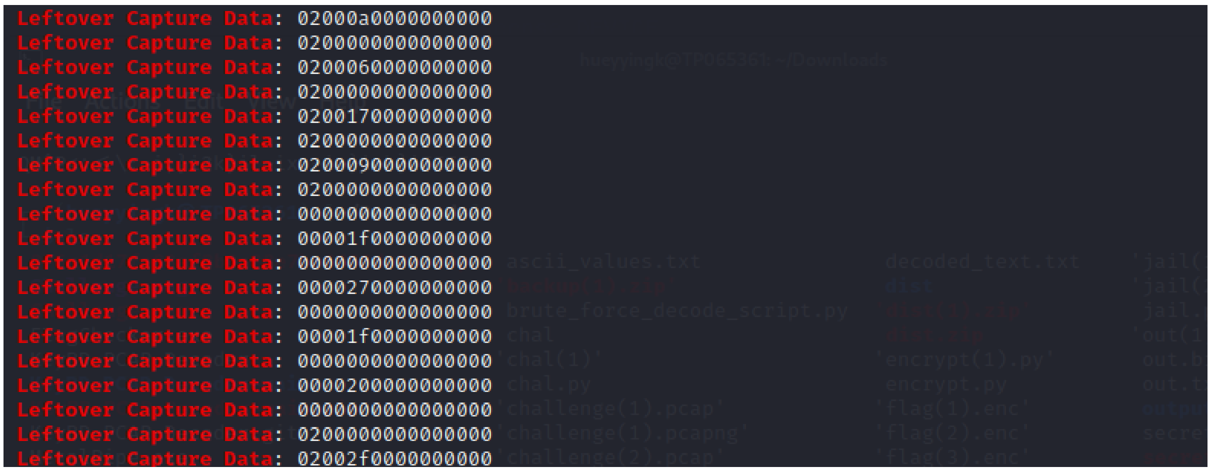


Table 12: Keyboard/Keypad Page

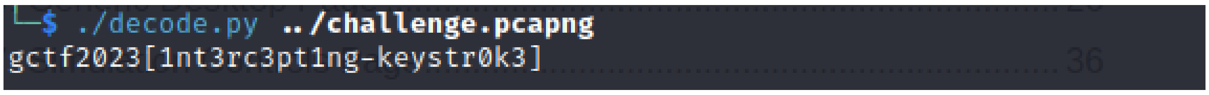
Usage ID (Dec)	Usage ID (Hex)	Usage Name	Ref: Typical AT-101 Position	PC-AT	Mac	UNIX	Boot
0	00	Reserved (no event indicated)[9]	N/A	√	√	√	4/101/104
1	01	Keyboard ErrorRollOver[9]	N/A	√	√	√	4/101/104
2	02	Keyboard POSTFail[9]	N/A	√	√	√	4/101/104
3	03	Keyboard ErrorUndefined[9]	N/A	√	√	√	4/101/104
4	04	Keyboard a and A[4]	31	√	√	√	4/101/104
5	05	Keyboard b and B	50	√	√	√	4/101/104
6	06	Keyboard c and C[4]	48	√	√	√	4/101/104
7	07	Keyboard d and D	33	√	√	√	4/101/104
8	08	Keyboard e and E	19	√	√	√	4/101/104
9	09	Keyboard f and F	34	√	√	√	4/101/104
10	0A	Keyboard g and G	35	√	√	√	4/101/104

From the data length that found in the **“Leftover Capture Data”** field can defined that it is keyboard packet as it consists of 8 bytes. Besides, for the keystroke is at the 3 bytes. Can determine the value by referring the table of key codes for the USB keyboard. The table can be found in https://www.usb.org/sites/default/files/documents/hut1_12v2.pdf.



```
Leftover Capture Data: 02000a0000000000
Leftover Capture Data: 0200000000000000
Leftover Capture Data: 0200060000000000
Leftover Capture Data: 0200000000000000
Leftover Capture Data: 0200170000000000
Leftover Capture Data: 0200000000000000
Leftover Capture Data: 0200090000000000
Leftover Capture Data: 0200000000000000
Leftover Capture Data: 0000000000000000
Leftover Capture Data: 00001f0000000000
Leftover Capture Data: 0000000000000000
Leftover Capture Data: 0000270000000000
Leftover Capture Data: 0000000000000000
Leftover Capture Data: 00001f0000000000
Leftover Capture Data: 0000000000000000
Leftover Capture Data: 0000200000000000
Leftover Capture Data: 0000000000000000
Leftover Capture Data: 0200000000000000
Leftover Capture Data: 02002f0000000000
```

To view all the **"Leftover Capture Data"** field, can use the command ***$ tshark –r filename.pcap -V | grep "Leftover Capture Data"*** in Kali Linux terminal which can filter out all the data.



```
└─$ ./decode.py ../challenge.pcapng
gctf2023[1nt3rc3pt1ng-keystr0k3]
```

To decode it in an easier and faster way, we found a pcap decode packages to be used. It can be found from here https://github.com/natesinger/KeyBD-PCAP-Decoder. After download this packages, you can used the command of ***./decode.py ../filename.pcap*** to decode the key code and find out the flag.

By following the format of the flag set, the flag that decode should be changed as

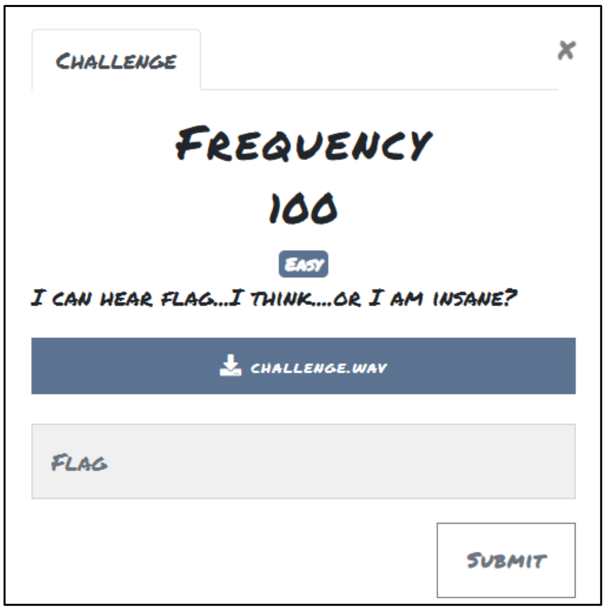
- "gctf" to "GCTF"
- "[ ]" to "{ }"
- "-" to "_"

Tool: Wireshark

Flag: GCTF2023{1nt3rc3pt1ng_keystr0k3}

MISC

Frequency



CHALLENGE

FREQUENCY

100

EASY

I CAN HEAR FLAG...I THINK....OR I AM INSANE?

CHALLENGE.WAV

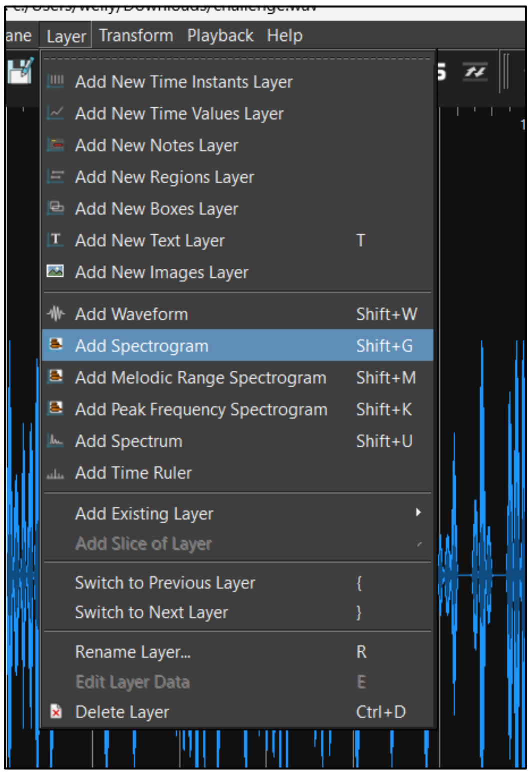
FLAG

SUBMIT

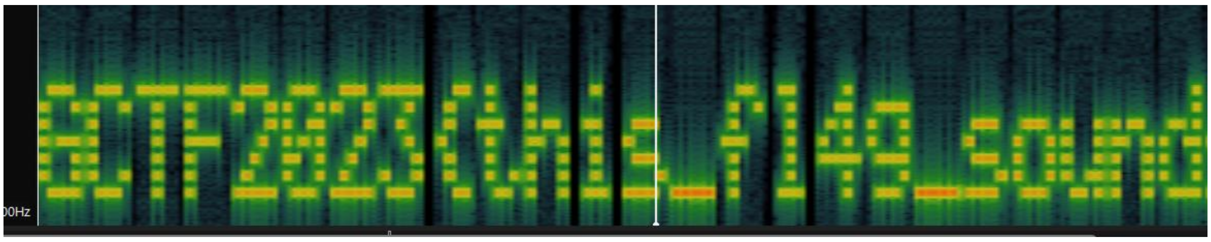
Attachment: challenge.wav

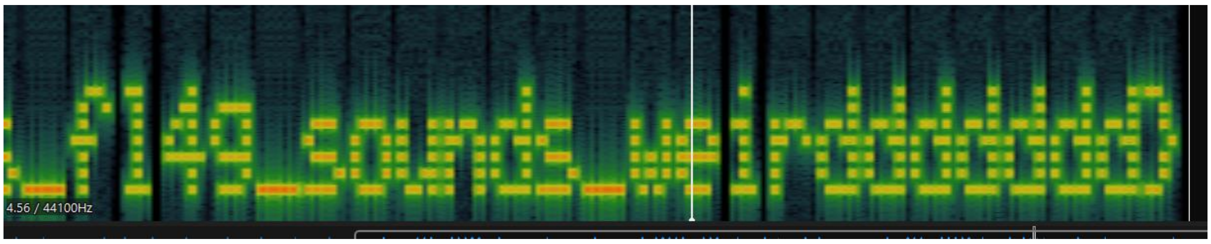
Solution:

Open the audio file in Sonic Visualiser. Then, add a spectrogram.



The flag is displayed.

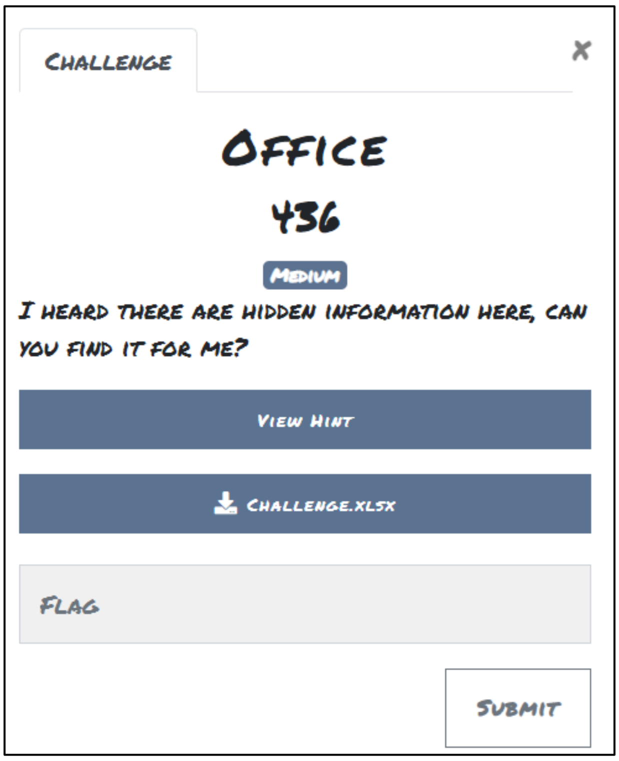




Tool: Sonic Visualiser

Flag: GCTF2023{this_fl4g_sounds_weirddddddd}

Office



CHALLENGE

OFFICE

436

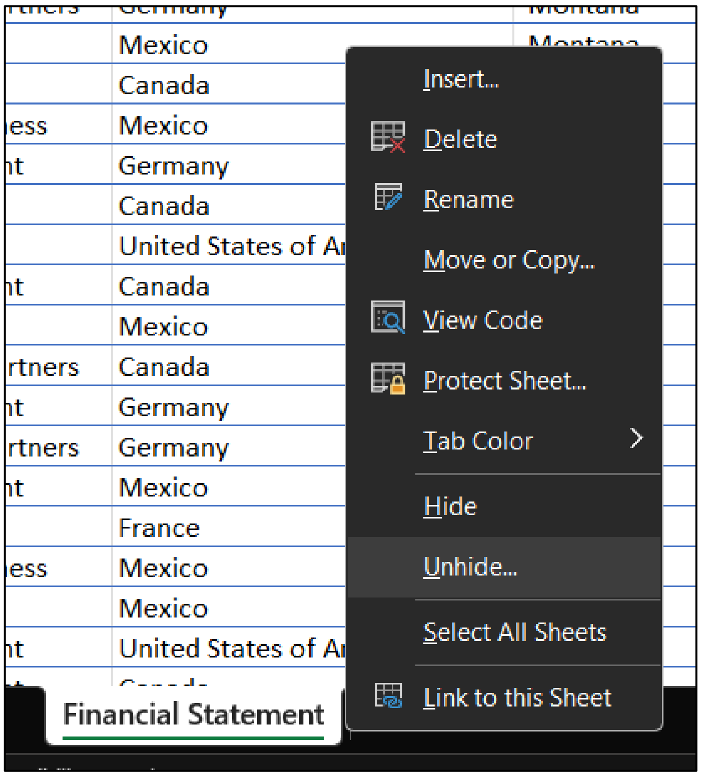
MEDIUM

I HEARD THERE ARE HIDDEN INFORMATION HERE, CAN YOU FIND IT FOR ME?

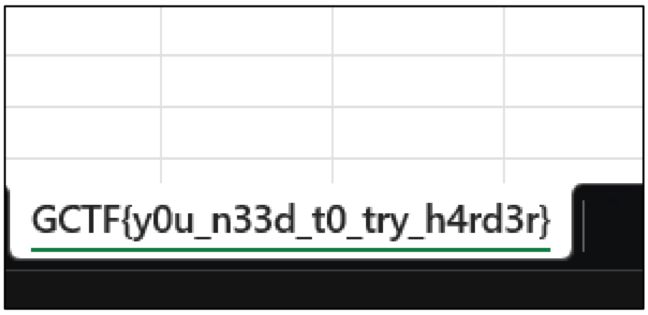
Attachment: Challenge.xlsx

Solution:

Unhide the sheets in the Excel workbook.

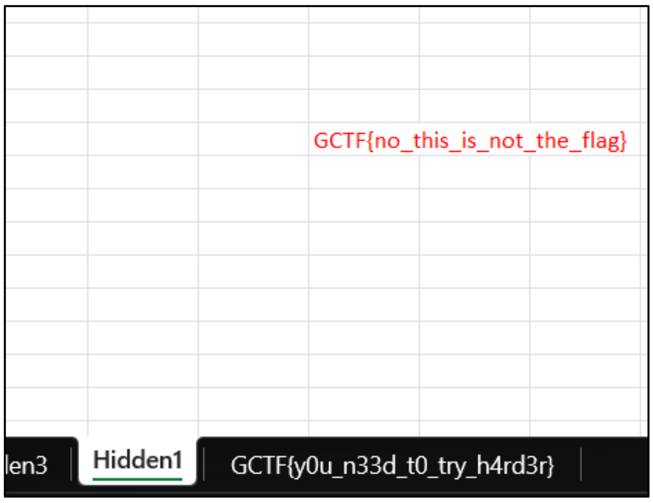


This tab name is not the correct flag.



GCTF{y0u_n33d_t0_try_h4rd3r}

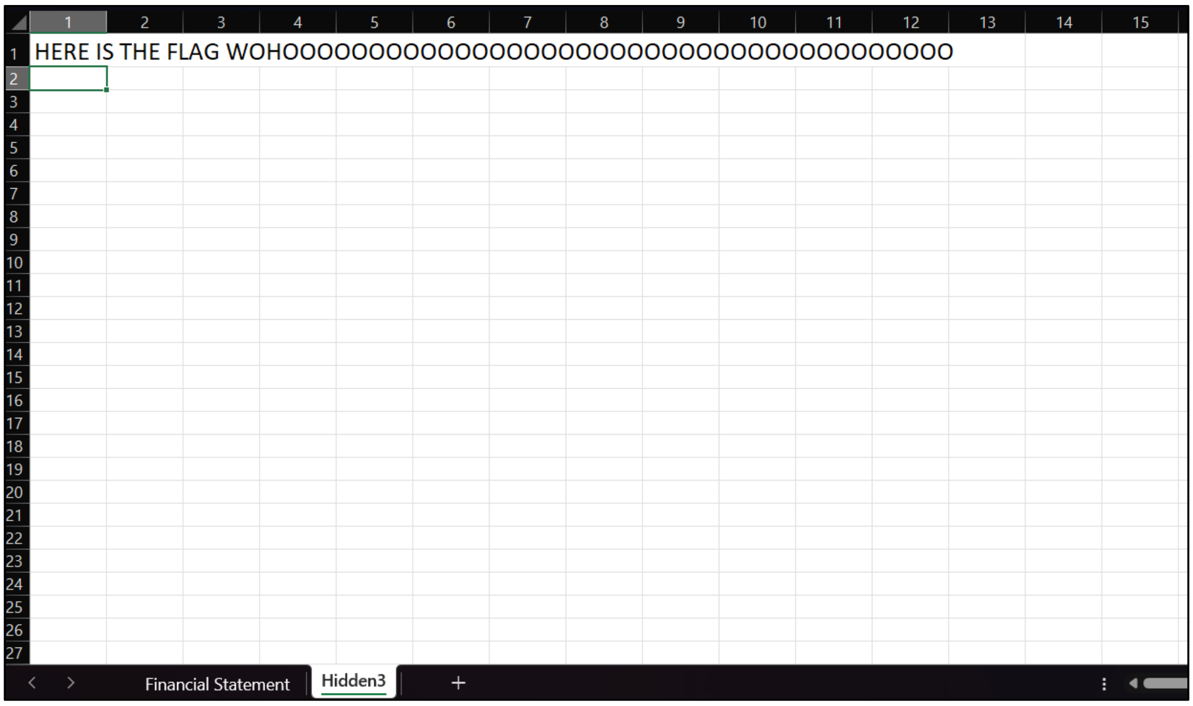
The correct flag is not on this "Hidden 1" tab either.



GCTF{no_this_is_not_the_flag}

len3 Hidden1 GCTF{y0u_n33d_t0_try_h4rd3r}

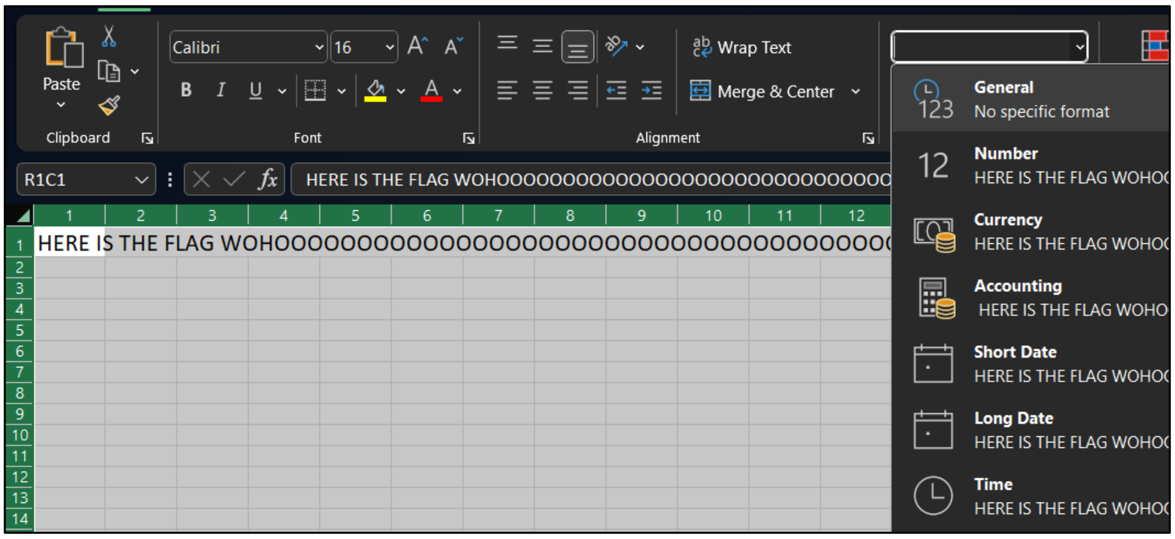
Click on the "Hidden3" tab.



HERE IS THE FLAG WOHOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOO

Financial Statement Hidden3

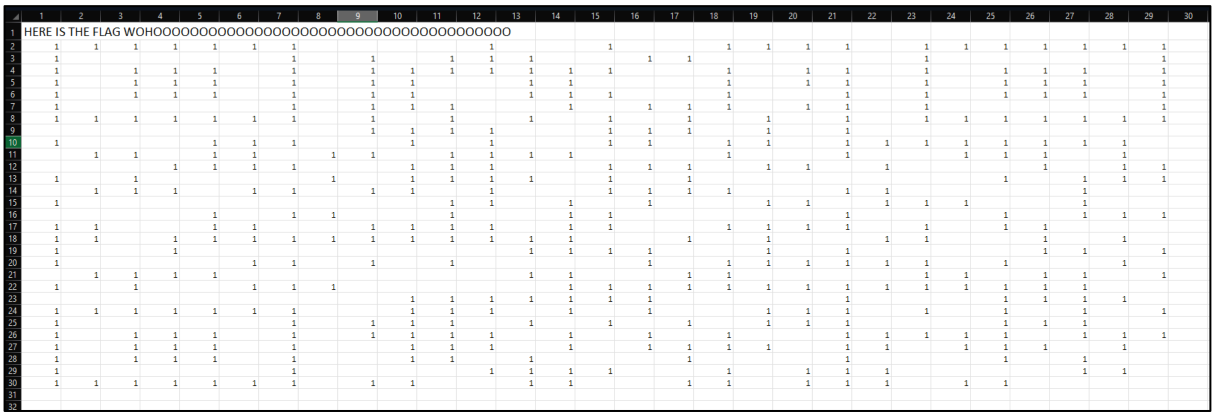
Select all cells and choose the number format as “General”.



HERE IS THE FLAG WOHOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOO

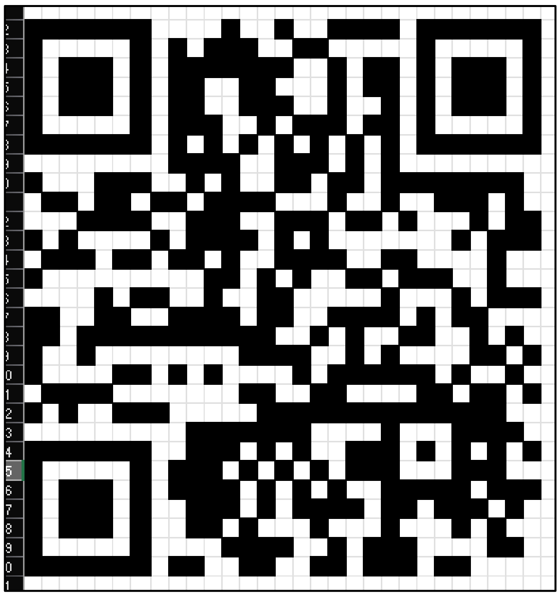
General
No specific format
Number
HERE IS THE FLAG WOHO
Currency
HERE IS THE FLAG WOHO
Accounting
HERE IS THE FLAG WOHO
Short Date
HERE IS THE FLAG WOHO
Long Date
HERE IS THE FLAG WOHO
Time
HERE IS THE FLAG WOHO

We can notice that there are many cells with the number “1”.



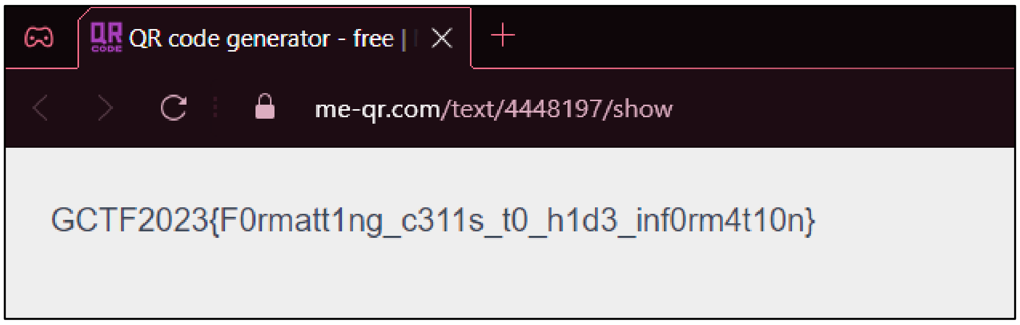
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	HERE IS THE FLAG WOHOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOO																													
2	1	1	1	1	1	1	1					1			1			1	1	1	1		1	1	1	1	1	1	1	
3	1						1		1		1	1	1			1	1						1						1	
4	1		1	1	1		1		1	1	1	1	1	1	1			1		1	1		1		1	1	1		1	
5	1		1	1	1		1		1	1			1	1				1		1	1		1		1	1	1		1	
6	1		1	1	1		1		1	1			1	1	1			1			1		1		1	1	1		1	
7	1						1		1	1	1			1		1	1	1		1	1		1						1	
8	1	1	1	1	1	1	1		1		1		1		1		1		1		1		1	1	1	1	1	1	1	
9									1	1	1	1			1	1	1		1		1									
10	1				1	1	1			1		1			1	1		1	1		1	1	1	1	1	1	1	1		
11		1	1		1	1		1	1		1	1	1	1				1			1			1	1	1		1		
12				1	1	1	1			1	1	1			1	1	1		1	1		1				1		1	1	
13	1		1					1		1	1	1	1		1		1								1		1	1	1	
14		1	1	1		1	1		1	1		1			1	1	1	1			1	1					1			
15	1										1	1		1		1			1	1		1	1	1			1			
16					1		1	1			1			1	1						1				1		1	1	1	
17	1	1			1	1			1	1	1	1		1	1			1	1	1	1		1		1	1				
18	1	1		1	1	1	1	1	1	1	1	1	1	1			1		1			1	1			1		1		
19	1			1									1	1	1	1			1		1					1	1		1	
20	1					1	1		1		1					1		1	1	1	1	1	1		1			1		
21		1	1	1	1								1	1			1	1					1	1		1	1		1	
22	1		1			1	1	1						1	1	1	1	1	1	1	1	1	1	1	1	1	1			
23										1	1	1	1	1	1	1					1				1	1	1	1		
24	1	1	1	1	1	1	1			1	1	1		1		1			1	1	1		1		1		1		1	
25	1						1		1	1	1		1		1		1		1	1	1				1	1	1			
26	1		1	1	1		1		1	1	1	1		1		1		1			1	1	1	1	1		1	1	1	
27	1		1	1	1		1			1	1	1		1		1	1	1	1		1	1		1	1	1		1		
28	1		1	1	1		1			1	1		1				1				1				1		1			
29	1						1					1	1	1	1			1		1	1	1					1	1		
30	1	1	1	1	1	1	1		1	1			1	1			1	1		1	1	1		1	1					
31																														
32																														

Fill the cells containing “1” with black color to reveal a QR Code.



Scan the QR Code to retrieve the flag.

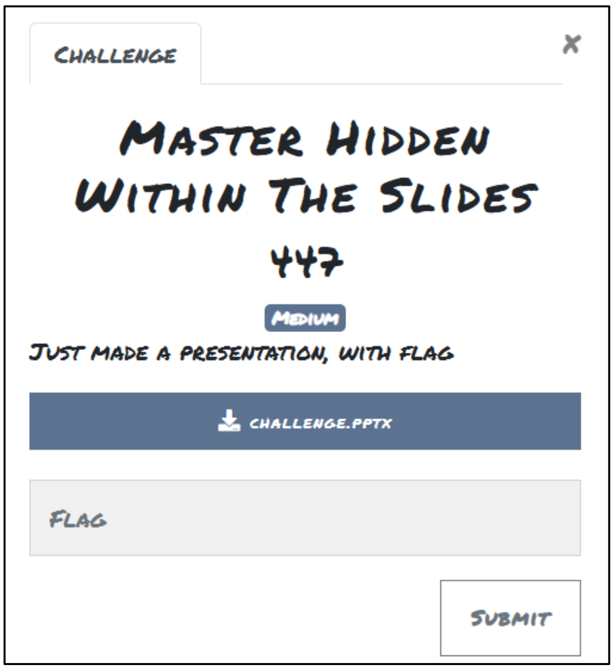
https://me-qr.com/text/4448197/show



GCTF2023{F0rmatt1ng_c311s_t0_h1d3_inf0rm4t10n}

Flag: GCTF2023{F0rmatt1ng_c311s_t0_h1d3_inf0rm4t10n}

Master Hidden Within The Slides


Challenge

Master Hidden Within The Slides

447

Medium

Just made a presentation, with flag

challenge.pptx

Flag

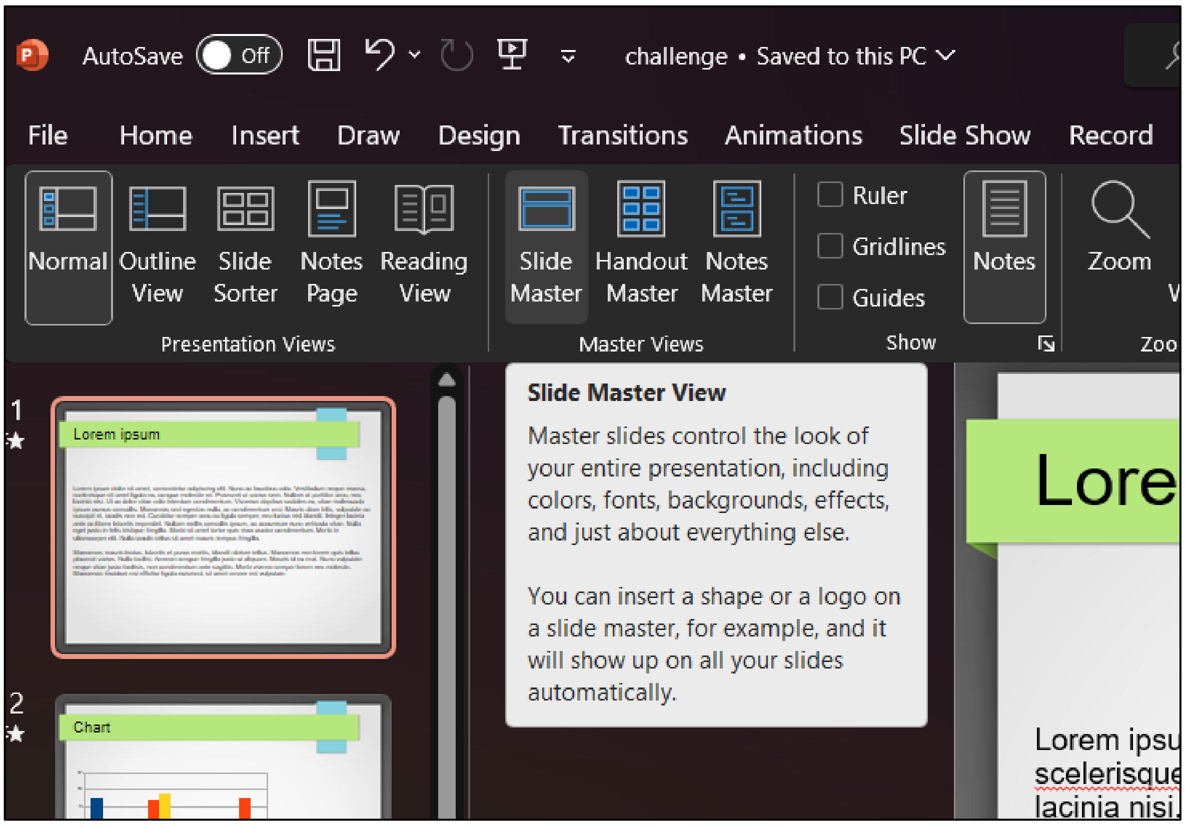
Submit

Attachment: challenge.pptx

Solution:

The clue is within the challenge title, which is "Master".

Open the PowerPoint file and enter "Slide Master" view.

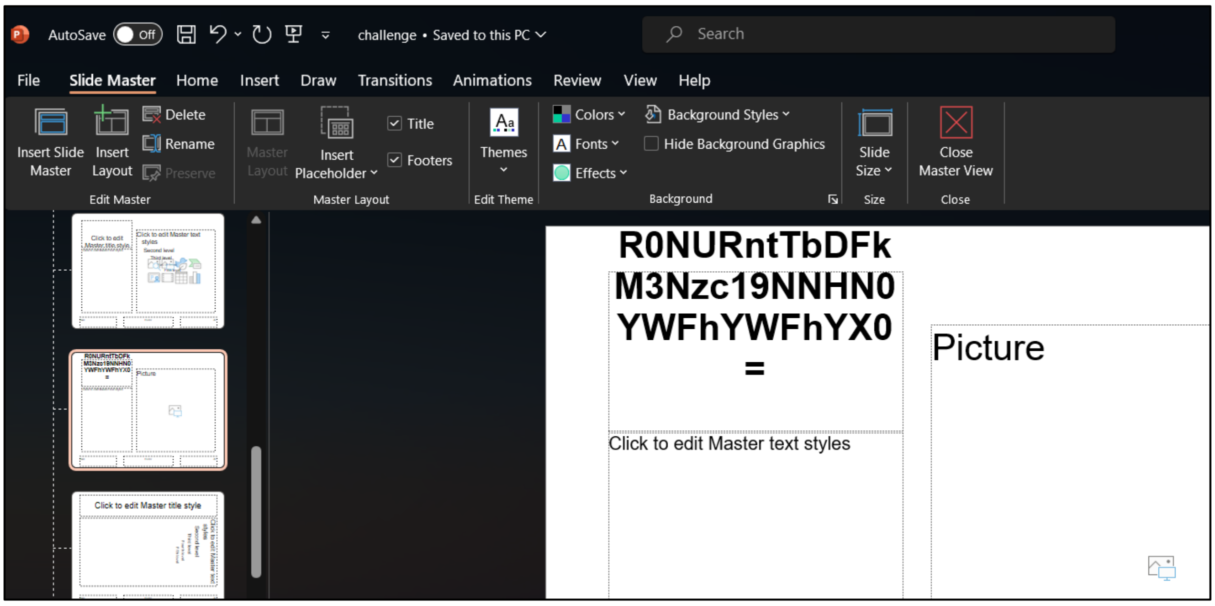

AutoSave Off
challenge • Saved to this PC
File Home Insert Draw Design Transitions Animations Slide Show Record
Normal Outline View Slide Sorter Notes Page Reading View
Presentation Views
Slide Master Handout Master Notes Master
Master Views
Ruler Gridlines Guides
Show
Notes
Zoom

Slide Master View

Master slides control the look of your entire presentation, including colors, fonts, backgrounds, effects, and just about everything else.

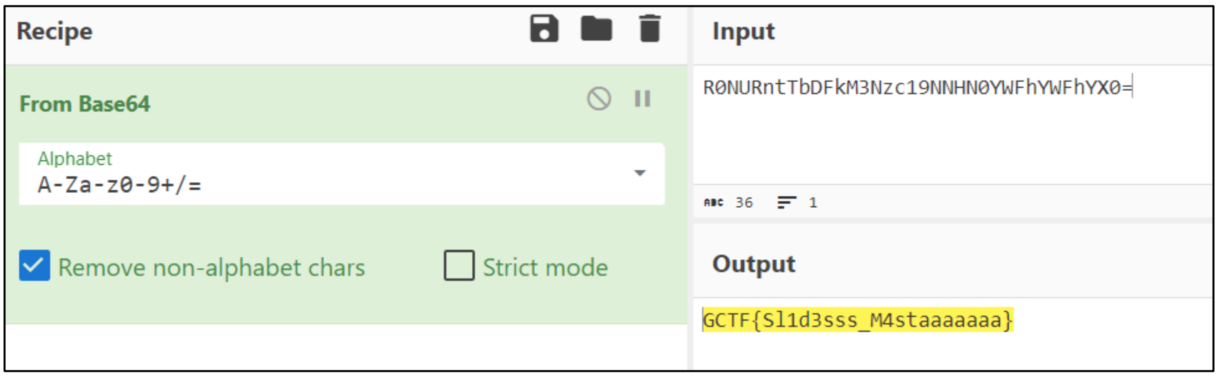
You can insert a shape or a logo on a slide master, for example, and it will show up on all your slides automatically.

The encoded flag is in one of the slides in the Slide Master view.



R0NURntTbDFkM3Nzc19NNHN0YWFhYWFhYX0=

Decode the flag from Base64.



Alphabet: A-Za-z0-9+/=

Remove non-alphabet chars [✓]

Strict mode

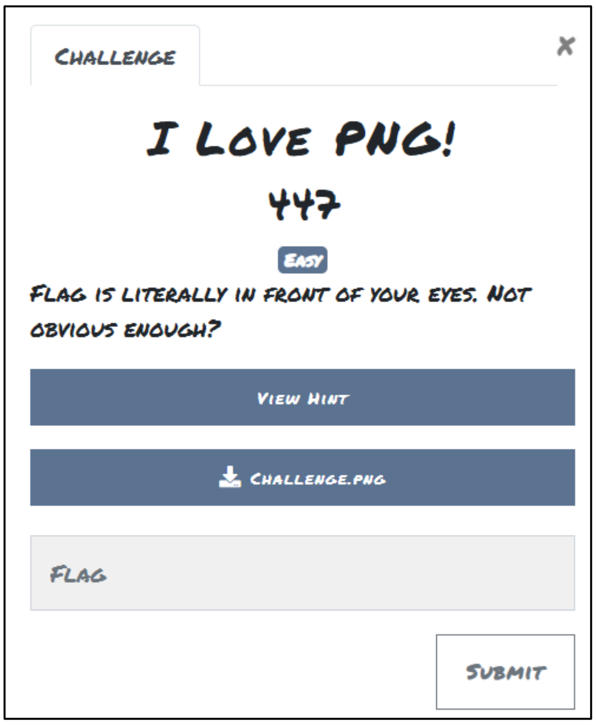
```
R0NURntTbDFkM3Nzc19NNHN0YWFhYWFhYX0=
```

```
GCTF{Sl1d3sss_M4staaaaaaa}
```

Tool: CyberChef - https://gchq.github.io/CyberChef/

Flag: GCTF{Sl1d3sss_M4staaaaaaa}

I Love PNG!


CHALLENGE

I LOVE PNG!

447

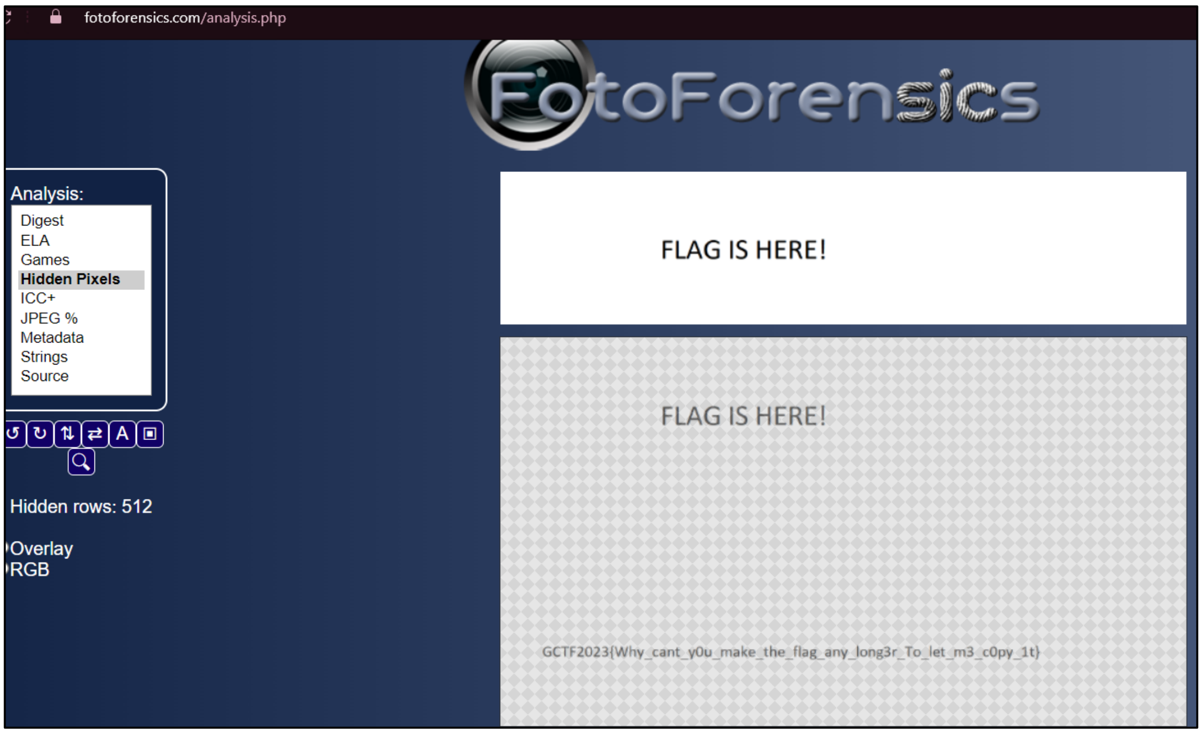
EASY

FLAG IS LITERALLY IN FRONT OF YOUR EYES. NOT OBVIOUS ENOUGH?

Attachment: Challenge.png

Solution:

Upload the PNG file to the FotoForensics website. Click on "Hidden Pixels" under Analysis, and the flag will be revealed.



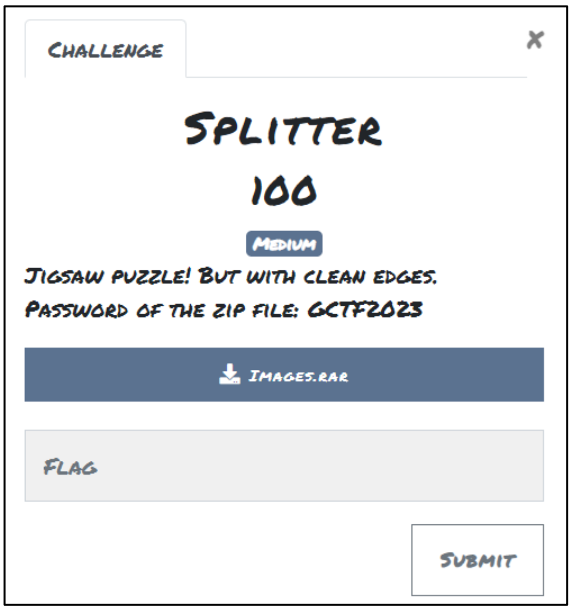
Hidden rows: 512

GCTF2023{Why_cant_y0u_make_the_flag_any_long3r_To_let_m3_c0py_1t}

Tool: FotoForensics - https://fotoforensics.com

Flag: GCTF2023{Why_cant_y0u_make_the_flag_any_long3r_To_let_m3_c0py_1t}

Splitter



Challenge

Splitter

100

Medium

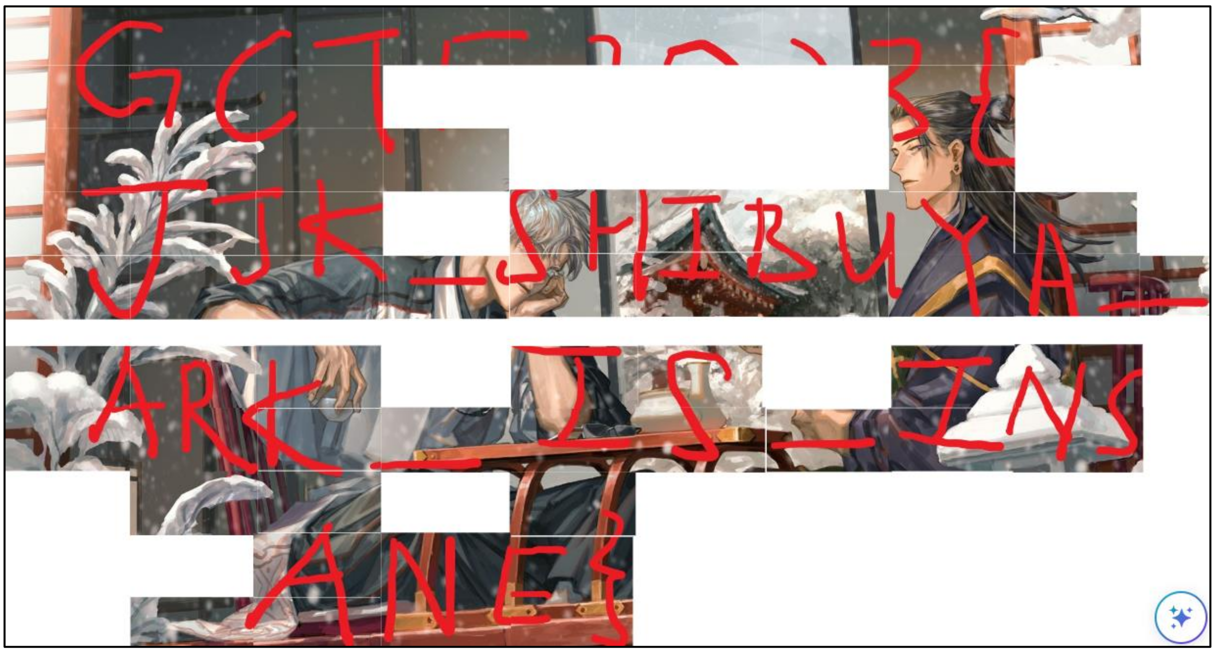
Jigsaw puzzle! But with clean edges.

Password of the zip file: GCTF2023

Attachment: Images.rar

Solution:

Extract the RAR file. In the 'Images' folder, I found 120 JPG files. These pictures are parts of a larger image. As I combine them in Canva, the flag will be revealed in red words.

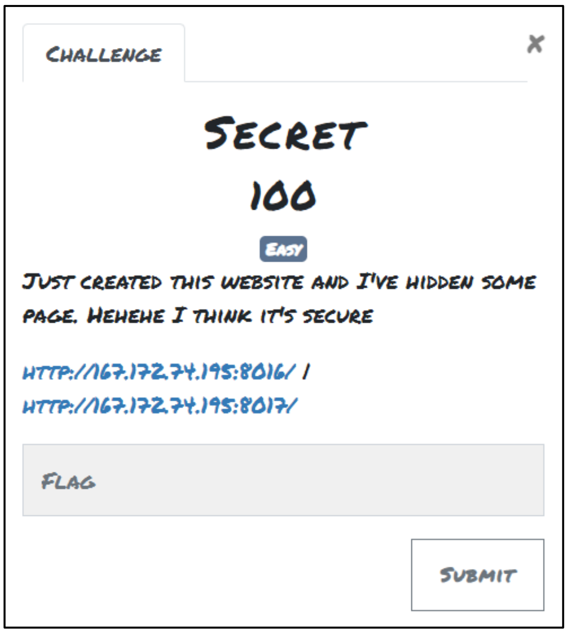


Tool: Canva - https://www.canva.com

Flag: GCTF2023{JJK_SHIBUYA_ARK_IS_INSANE}

Web

Secret

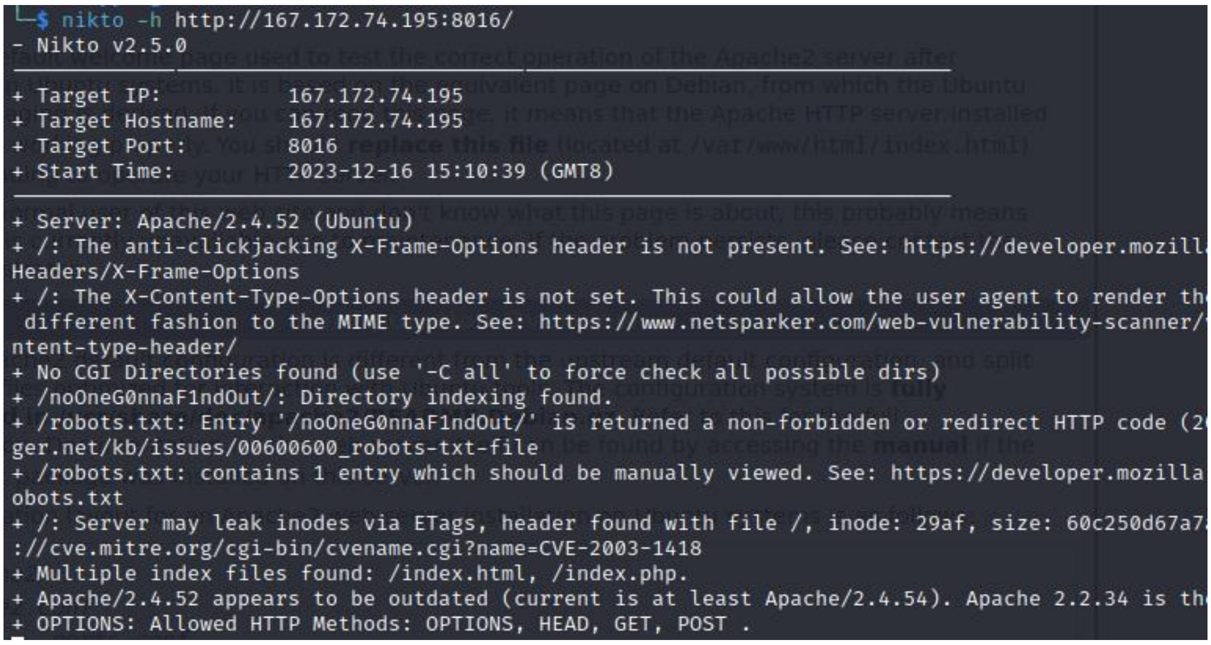


Just created this website and I've hidden some page. Hehehe I think it's secure

http://167.172.74.195:8016/ / http://167.172.74.195:8017/

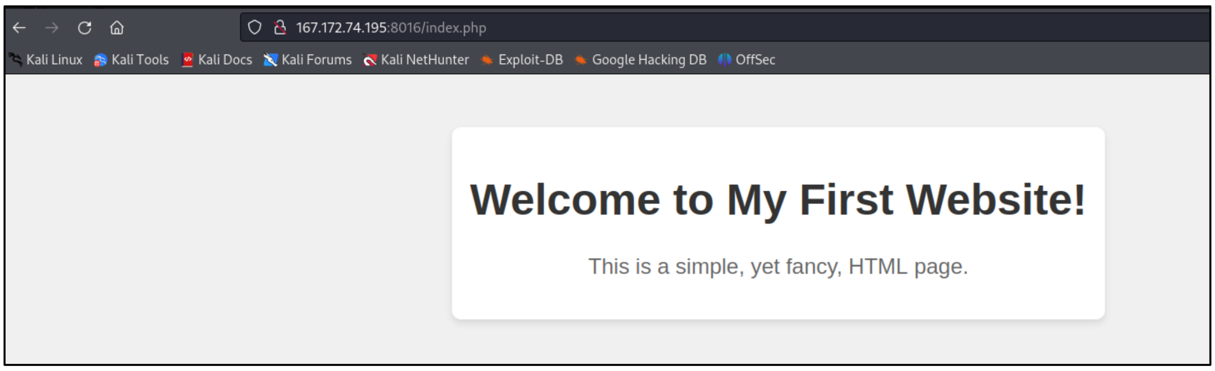
Solution:

As the question mentioned that hidden pages. To determine that can use the nikto which is an open-source web server scanner. It functions as examines a website and reports back vulnerabilities. By using the command ***# nikto -h $webserverurl***, the output provided the page that is hidden.



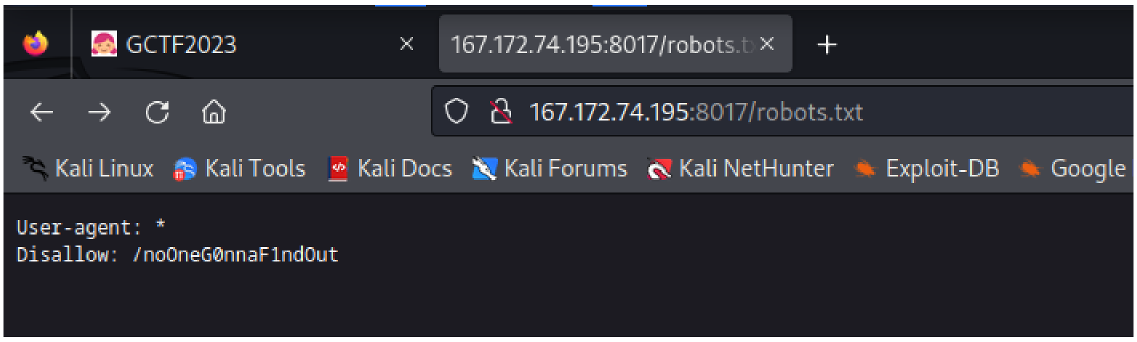
```
└─$ nikto -h http://167.172.74.195:8016/
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          167.172.74.195
+ Target Hostname:    167.172.74.195
+ Target Port:        8016
+ Start Time:         2023-12-16 15:10:39 (GMT8)
---------------------------------------------------------------------------
+ Server: Apache/2.4.52 (Ubuntu)
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozill
Headers/X-Frame-Options
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render th
 different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/
ntent-type-header/
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ /noOneG0nnaF1ndOut/: Directory indexing found.
+ /robots.txt: Entry '/noOneG0nnaF1ndOut/' is returned a non-forbidden or redirect HTTP code (2
ger.net/kb/issues/00600600_robots-txt-file
+ /robots.txt: contains 1 entry which should be manually viewed. See: https://developer.mozilla
obots.txt
+ /: Server may leak inodes via ETags, header found with file /, inode: 29af, size: 60c250d67a7
://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2003-1418
+ Multiple index files found: /index.html, /index.php.
+ Apache/2.4.52 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is th
+ OPTIONS: Allowed HTTP Methods: OPTIONS, HEAD, GET, POST .
```

From the above image, you can determine that there are multiple index files which are/index.html and /index.php. The /index.php is the hidden page. Added /index.php at the end of the link. Besides that, there is also a robots.txt which is probably next page that looking for.


167.172.74.195:8016/index.php
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

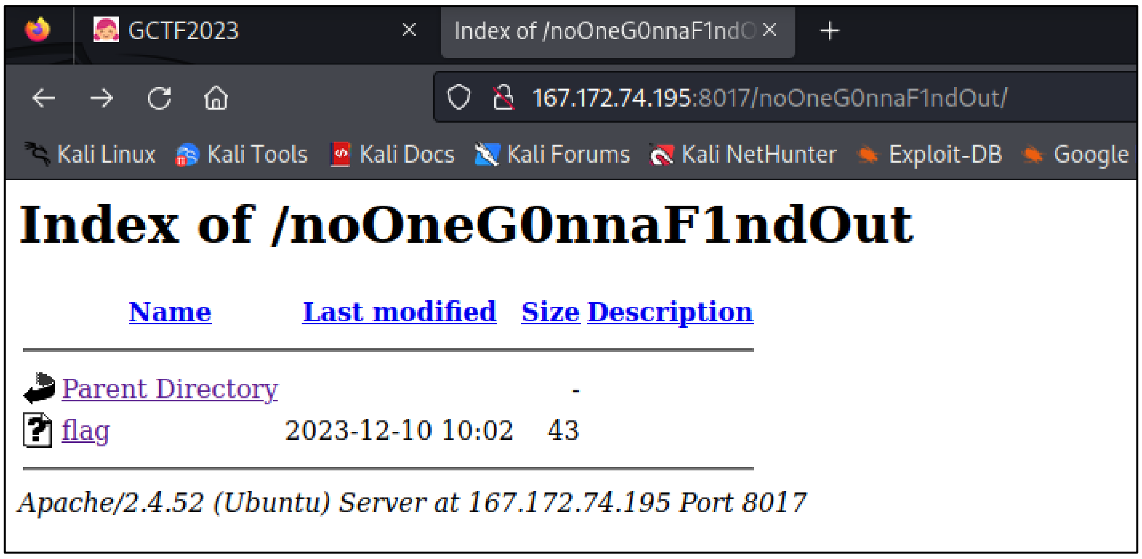
Welcome to My First Website!

This is a simple, yet fancy, HTML page.


GCTF2023
167.172.74.195:8017/robots.t
167.172.74.195:8017/robots.txt
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google

```
User-agent: *
Disallow: /noOneG0nnaF1ndOut
```

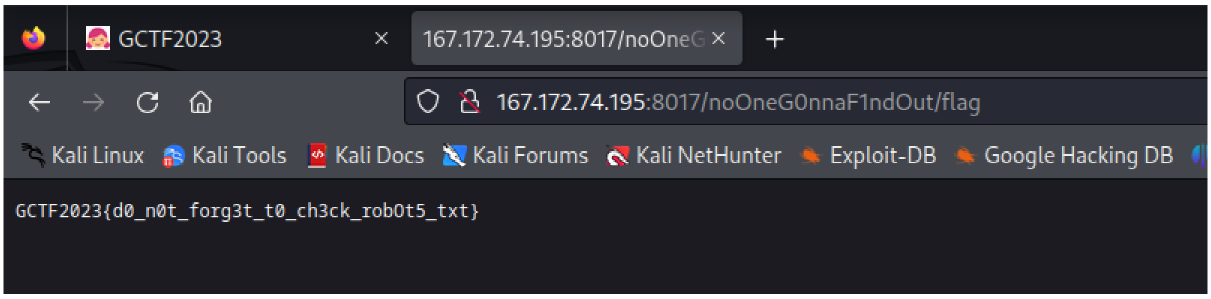
The page with /robots.txt will show all the page that does not allow someone to see it. In the picture, we know that the page that disallows to see is /noOneG0nnaF1ndOut and that page might have the flag.


GCTF2023
Index of /noOneG0nnaF1ndO
167.172.74.195:8017/noOneG0nnaF1ndOut/
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google

Index of /noOneG0nnaF1ndOut

```
Name              Last modified     Size Description
Parent Directory                      -
flag              2023-12-10 10:02   43
Apache/2.4.52 (Ubuntu) Server at 167.172.74.195 Port 8017
```

We found a hidden page named the flag and it is probably the last page of the website to find out.

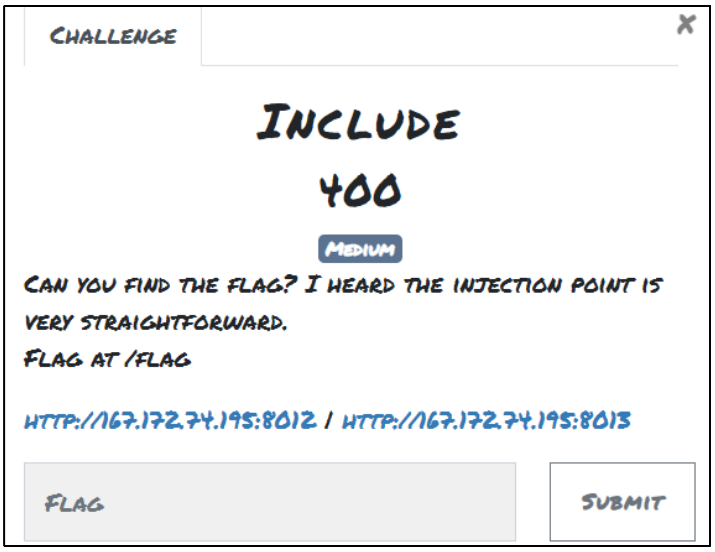


```
GCTF2023{d0_n0t_forg3t_t0_ch3ck_rob0t5_txt}
```

Here you go for the flag.

Flag: GCTF2023{d0_n0t_forg3t_t0_ch3ck_robOt5_txt}

Include


CHALLENGE

INCLUDE

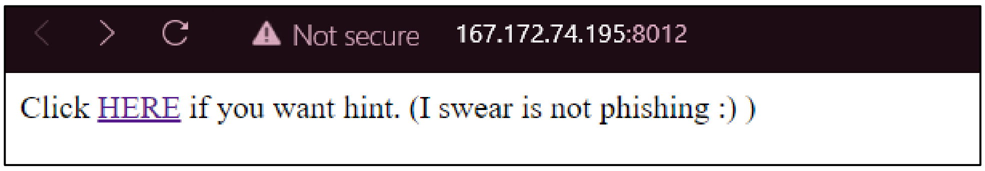
400

MEDIUM

CAN YOU FIND THE FLAG? I HEARD THE INJECTION POINT IS VERY STRAIGHTFORWARD.
FLAG AT /FLAG

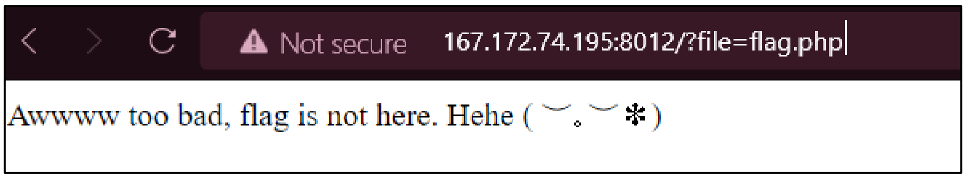
HTTP://167.172.74.195:8012 / HTTP://167.172.74.195:8013

Solution:



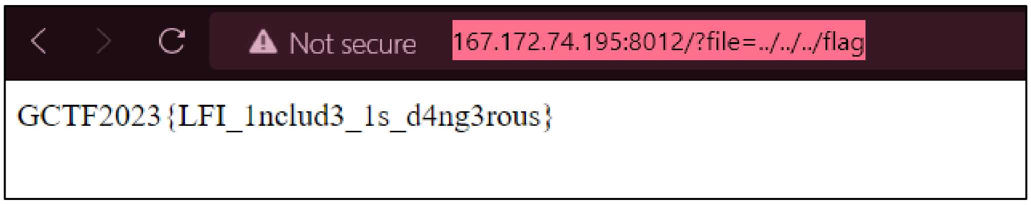
Click HERE if you want hint. (I swear is not phishing :))

After clicking "HERE" on the given link, it will direct us to another link that shows the flag is not there.



Awwww too bad, flag is not here. Hehe (︶｡︶✽)

I'm not familiar with web challenges, but I think it involves Local File Inclusion (LFI) because the challenge title mentions "Include". The description also states that "flag at /flag". So, I added the path to /flag. I tried to navigate to the correct directory by adding a few "../" in front of "flag" in the link until I reached the correct location.
The link that displays the flag is http://167.172.74.195:8012/?file=../../../flag.

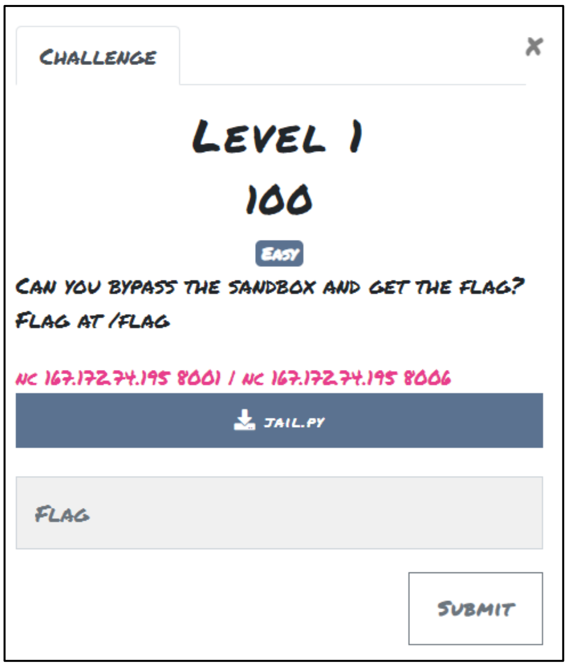


GCTF2023{LFI_1nclud3_1s_d4ng3rous}

Flag: GCTF2023{LFI_1nclud3_1s_d4ng3rous}

PWN

Level 1



Challenge

Level 1

100

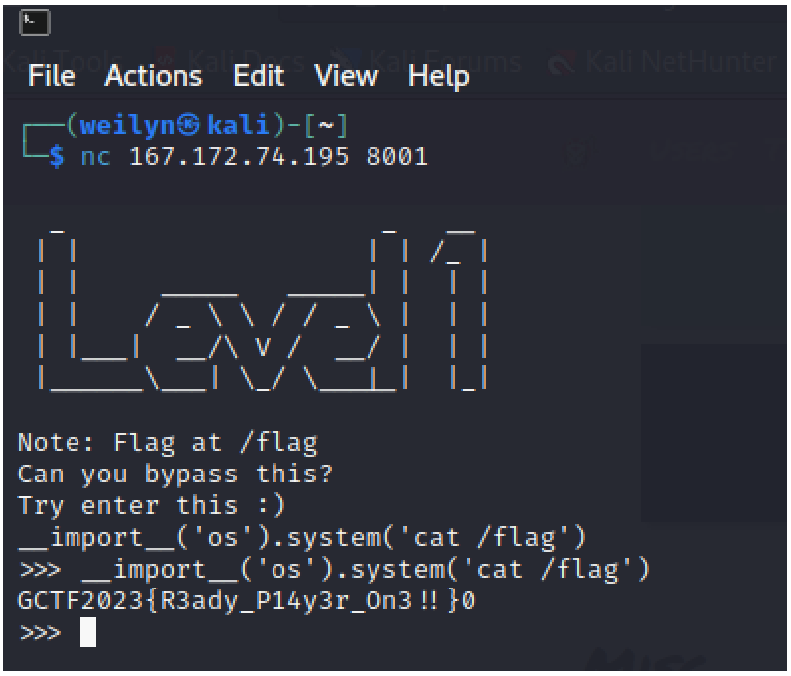
Easy

Can you bypass the sandbox and get the flag?
Flag at /flag

nc 167.172.74.195 8001 / nc 167.172.74.195 8006

Solution:

Type command “nc 167.172.74.195 8001” in terminal. Then, enter command “__import__(‘os’).system(‘cat /flag’)”. The flag is displayed.

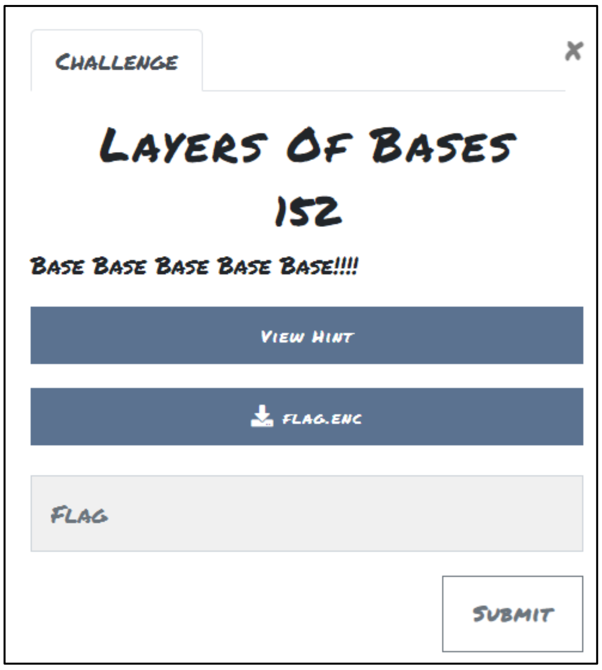


```
┌──(weilyn㉿kali)-[~]
└─$ nc 167.172.74.195 8001

 _                    _  __
| |                  | |/_ |
| |     ___ __   ___ | | | |
| |    / _ \ \ / / _ \| | | |
| |___|  __/\ V /  __/| | | |
|______\___| \_/ \___||_| |_|

Note: Flag at /flag
Can you bypass this?
Try enter this :)
__import__('os').system('cat /flag')
>>> __import__('os').system('cat /flag')
GCTF2023{R3ady_P14y3r_On3!! }0
>>> 
```

Flag: GCTF2023{Ready_P14y3r_0n3 !! }

Crypto

Layers Of Bases



CHALLENGE

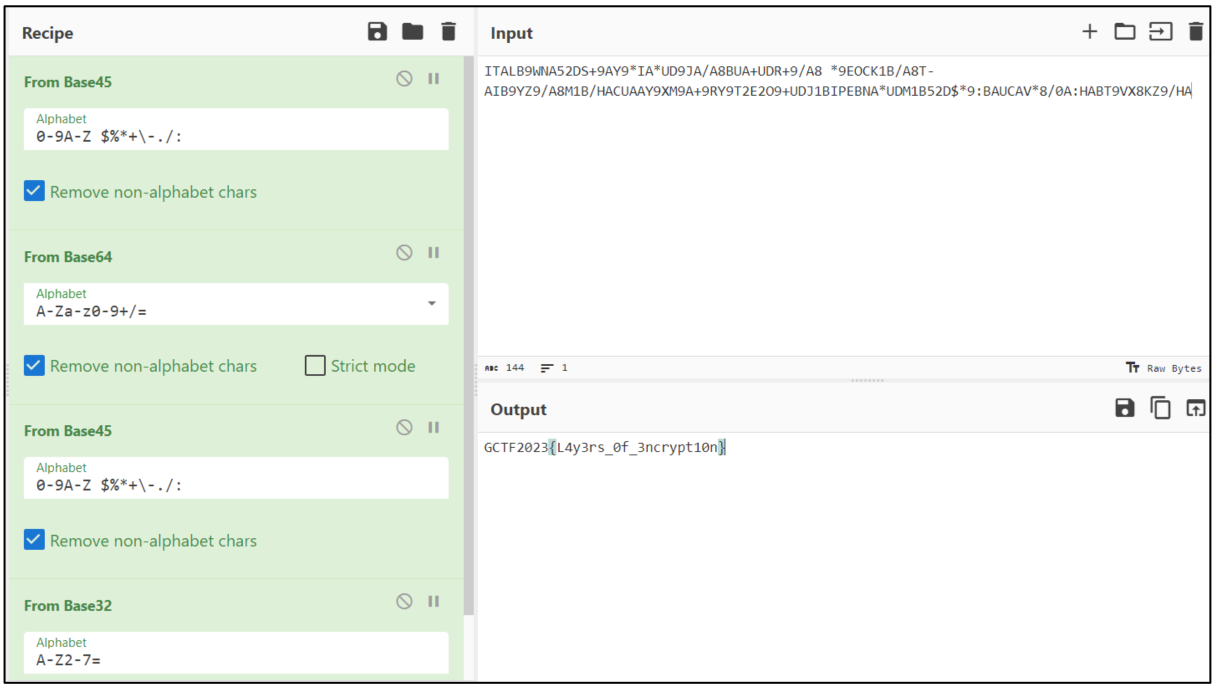
Layers Of Bases

152

Base Base Base Base Base!!!!

Solution:

I randomly inputting several recipes in CyberChef for a while and managed to obtain the flag. The flag is decoded using Base45 twice, followed by Base64 and Base32.



Recipe

From Base45
Alphabet: 0-9A-Z $%*+\-./:
☑ Remove non-alphabet chars

From Base64
Alphabet: A-Za-z0-9+/=
☑ Remove non-alphabet chars ☐ Strict mode

From Base45
Alphabet: 0-9A-Z $%*+\-./:
☑ Remove non-alphabet chars

From Base32
Alphabet: A-Z2-7=

Input

```
ITALB9WNA52DS+9AY9*IA*UD9JA/A8BUA+UDR+9/A8 *9EOCK1B/A8T-
AIB9YZ9/A8M1B/HACUAAY9XM9A+9RY9T2E2O9+UDJ1BIPEBNA*UDM1B52D$*9:BAUCAV*8/0A:HABT9VX8KZ9/HA
```

Output

```
GCTF2023{L4y3rs_0f_3ncrypt10n}
```

Tool: CyberChef - https://gchq.github.io/CyberChef/

Flag: GCTF2023{L4y3rs_0f_3ncrypt10n}