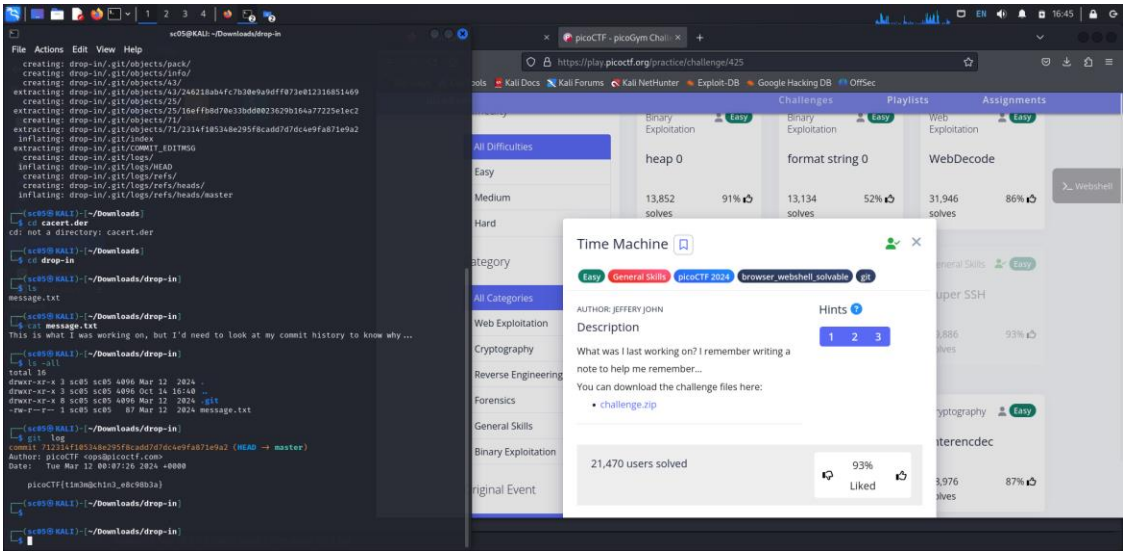
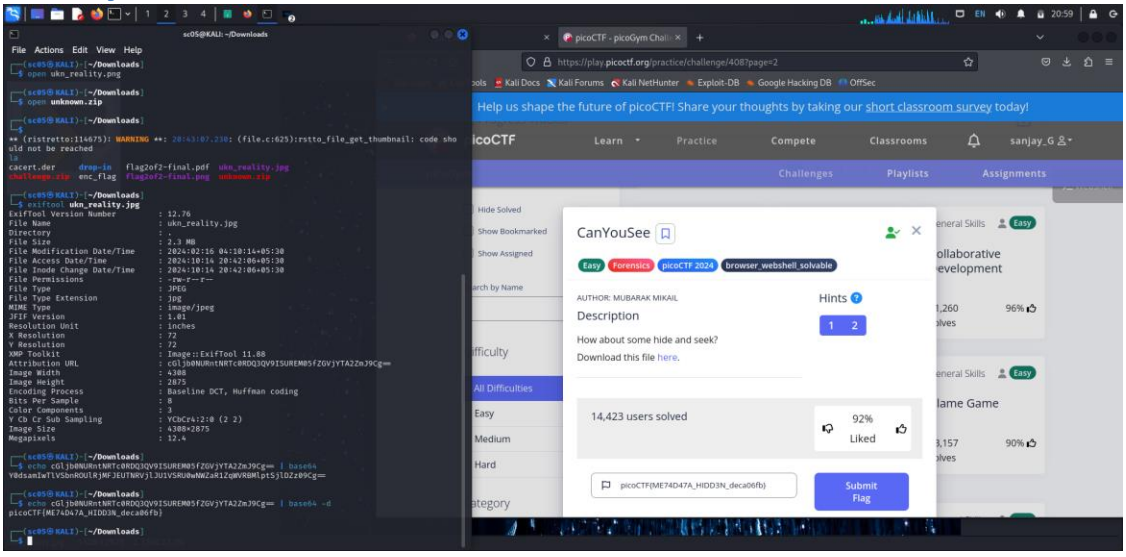
picoCTF-2021-All-Event

1.Time machine General skill



- Its a general skill task.It contains a message.txt file in a zip file which as the CTF.allenge. For this challenge it is only needed to use the `git log` command where prior commits could be seen to have the flag displayed as a message attached to the git commit.

2.Canyousee forensics



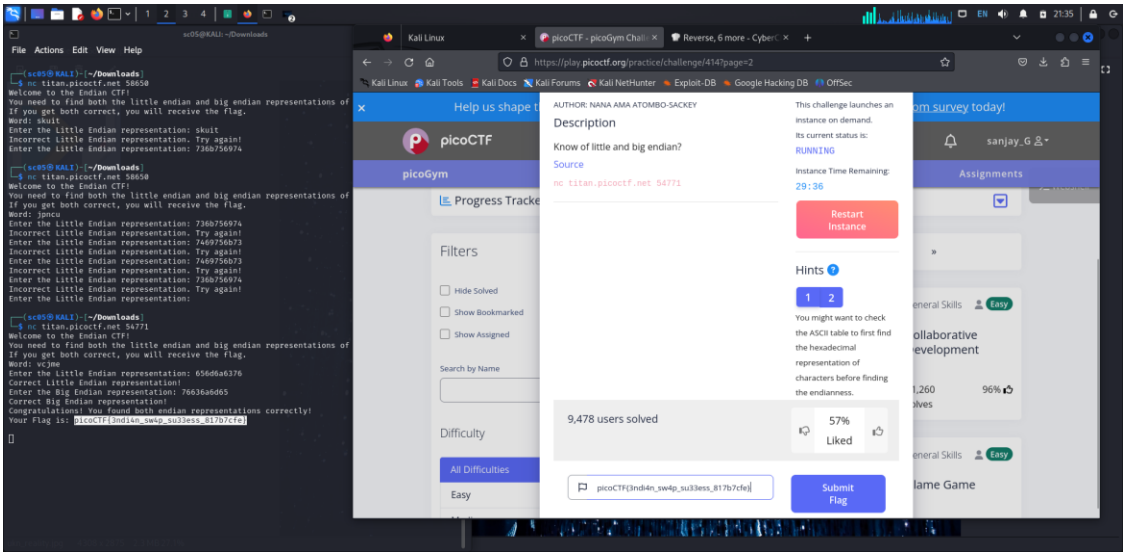
- This task comes under forensics. Actually forensics topic is intresting to solve .The flag is hiden in metadata.By running `exiftool ukn_reality.jpg` the Attribution

URL section looks like it has Base64 encoded text (cGljb0NURntNRTc0RDQ3QV9ISUREM05fYTZkZjhkYjh9Cg==). By putting that text into CyberChef with Base64 decoding the flag could be found.

- You could also get the flag in one command: `exiftool ukn_reality.jpg | grep At | cut -d ":" -f2 | tr -d " " | base64 -d`

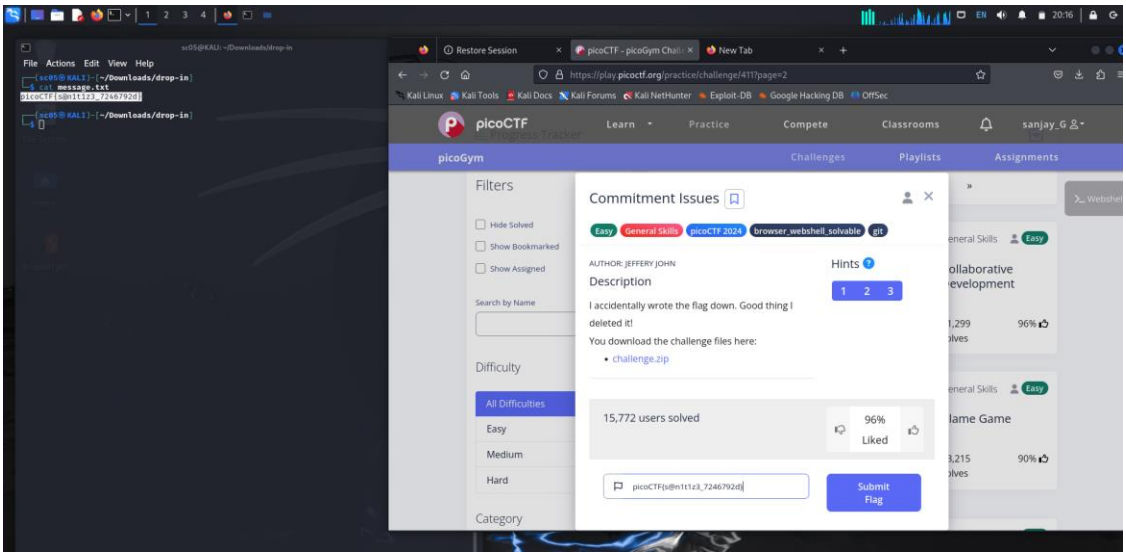
The first part is getting the line of the Attribution URL section. Then use the cut command with the delimiter of a colon (:) and get the second field. Using the tr command to trim the leading spaces. Lastly, use `base64 -d` to decode the output and get the flag.

3.endianness General skill



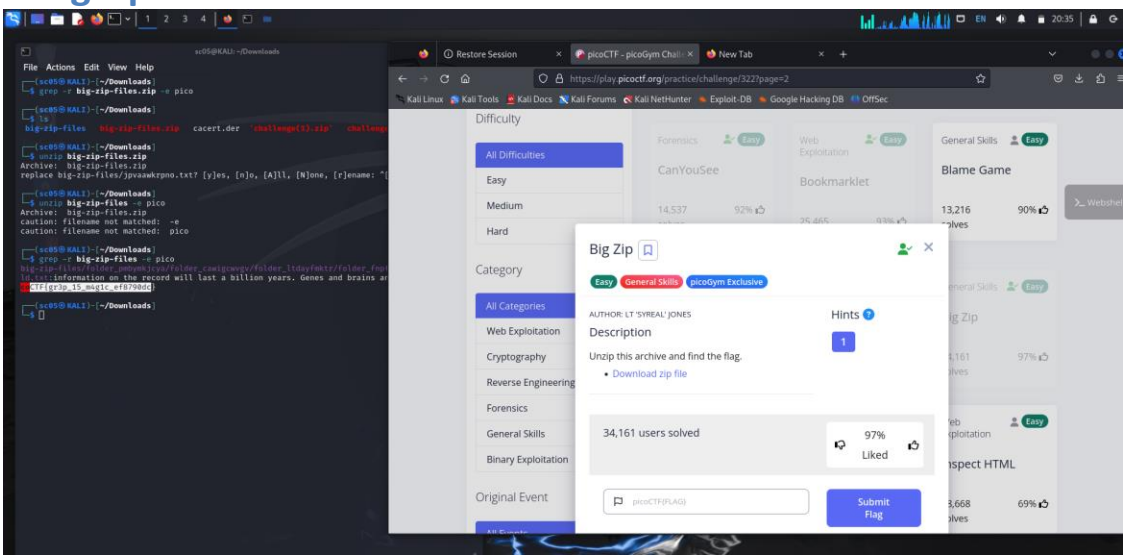
- By using CyberChef the file was put into the input section. Then converted to hex for the “Swap Endianness” function under a word length of 4. After this, the hex looks more like a JPG with the correct ÿØÿà□□JFIF□□ magic bytes start. After the endianness was swapped and it’s in the correct order it could be converted back from hex to get the data from the image. It could then be rendered in CyberChef to get the image that displays the flag.

4.commitement issues General skill



- first download the zip file Then unzip challenge.zip.
- Then cd drop-in/ and with ls -a the ".git" file can now be seen. Originally there was just a file called "message.txt" with file contents "TOP SECRET". You can use the git log command to see prior commits made

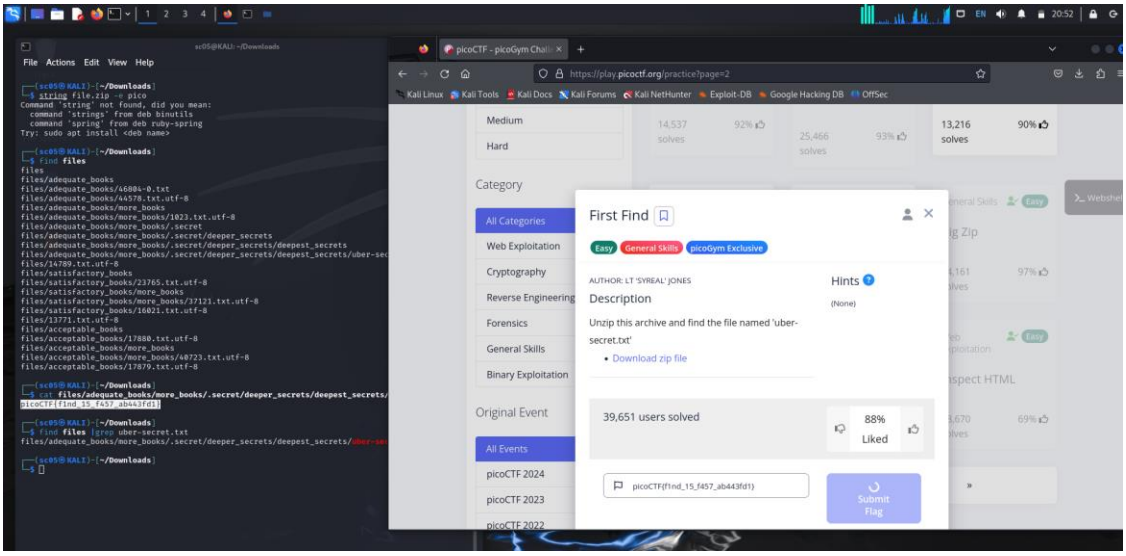
5.Big zip General skill



First we need to download and unzip this archive. To download and unzip the archive we will use wget a tool for downloading files online. Here is what I typed in my shell. * $ wget https://artifacts.picoctf.net/c/504/big-zip-files.zip $ unzip big-zip-files Now that we have downloaded and unzips the file we can go in and do some exploring using "ls" and "cd". Right now I'm keeping my eye out for any interesting named directories or files. I quickly realize that there is too much here to do this manually. I take a look at the hint. I find that

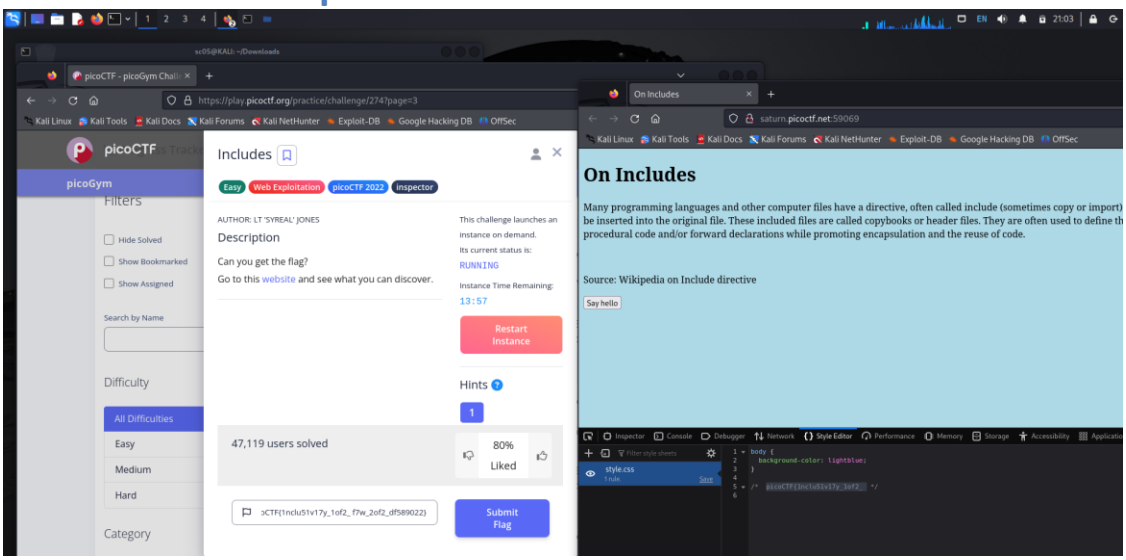
the -r parameter tells grep to search all the files in a directory. I run the following grep command. I'm telling grep to search all the files in big-zip-files for the pattern "pico" since the flags are structured as picoCTF{flag}

6.first find General skill



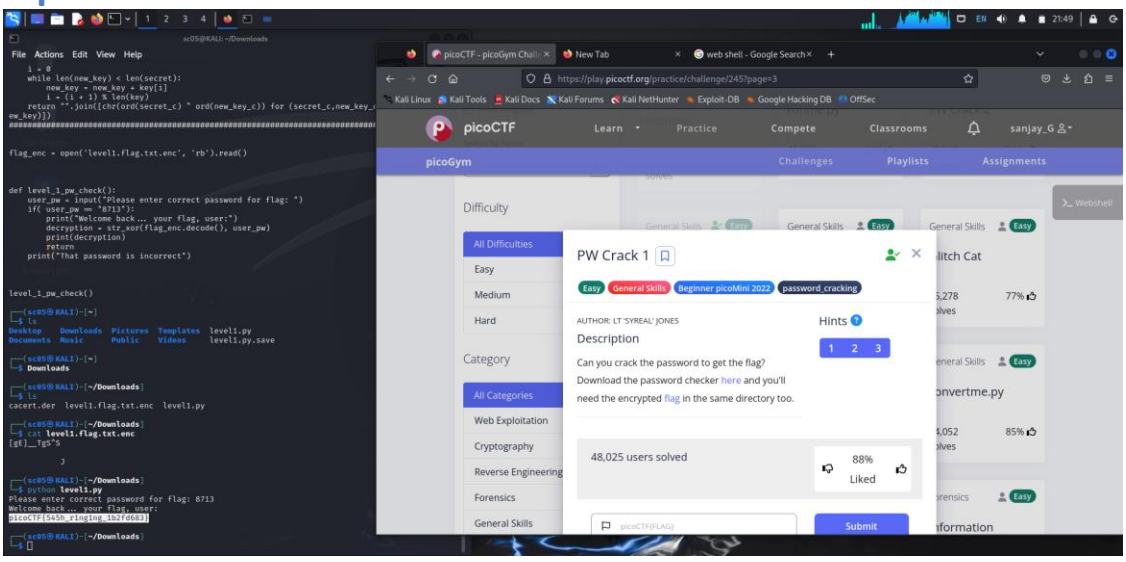
- When we simply `cat` this file in the shell to find the flag, we are met with a lot of random noise. Here,we can use a `grep` command. What `grep` does is it filters for a specific expression in a plain-text. We know that picoCTF flags are all in the format of picoCTF{...}, so we can grep for the expression `picoCTF`. Specifically, we would do `cat file | grep picoCTF`.

7.includes web exploitation



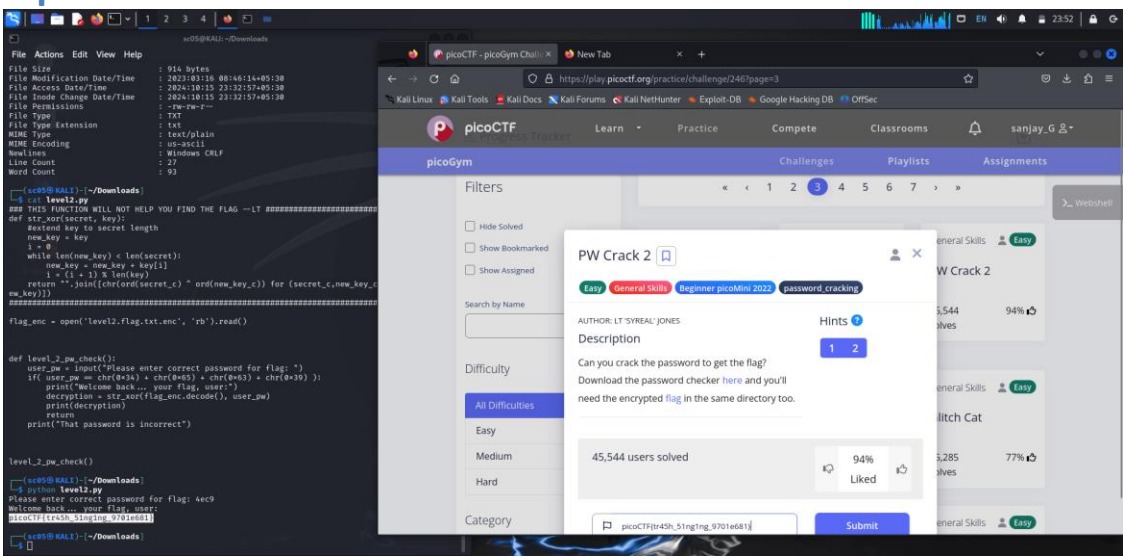
- Looking at the text on the website discussing importing files, when can make the educated guess that the file containing the flag is imported in the source code. Looking at the HTML of the website, we can see that there are two files imported, style.css and script.js. If we open these files, we can see that each contains half of the flag

8.pw crack 1General skill



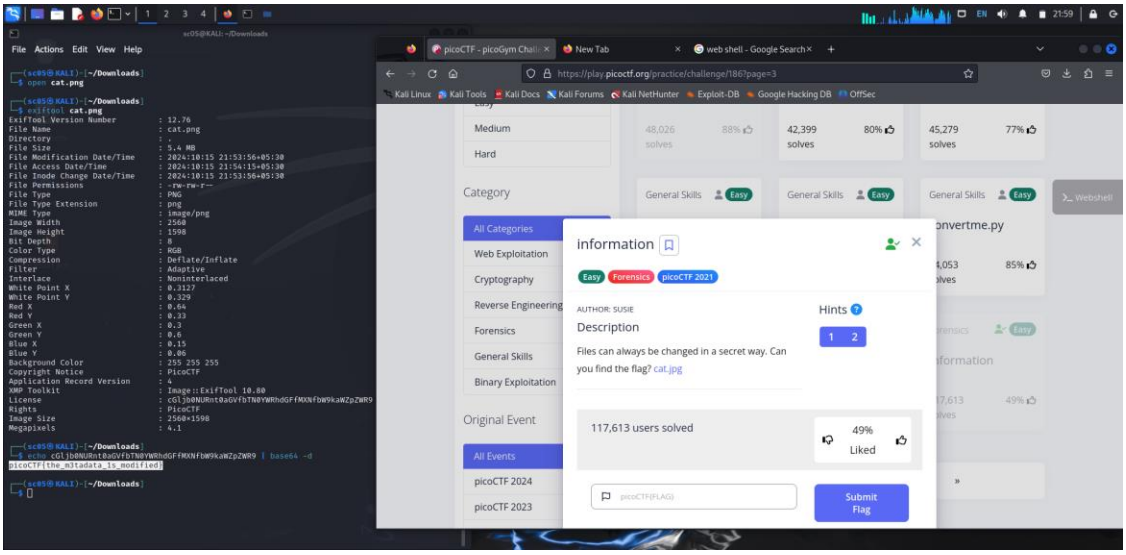
- I just saw the Pytohn file and found the password that was to be entered on running the file.
- Then just type in the password and you will get the flag.

9.pw crack 2 General skill



- Now here when I opened the code, I found some ASCIIs codes to get the password. I used the Table of ASCII to get the values

10.information forensics



- The task provides an image called “cat.jpg”, which we can download to look up a bit better. After downloading it we can run the commands file, hexdump and binwakto actually verify that is a jpg image.
- When we have to work with images, it’s always a good idea to open it and look at its metadata, to lookup authors, properties and other interesting data. We just need to open the image with an image viewer
- The licence field is filled with an interesting string, which looks a bit off for a real licence standard.

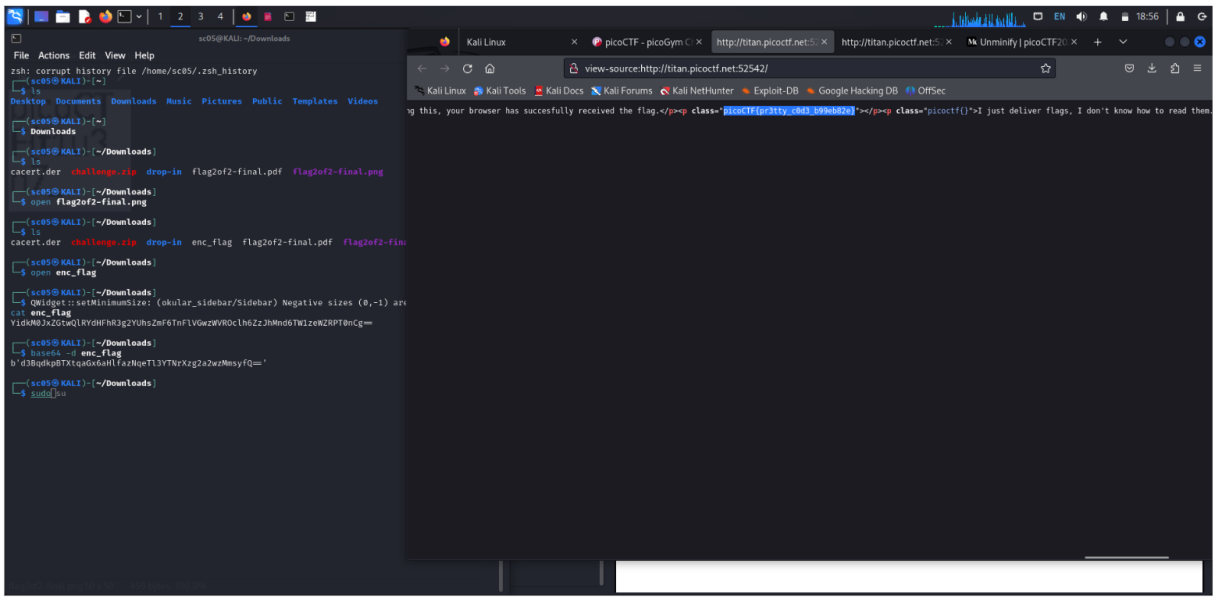
- That looks like a hash of base64.verify that by opening our terminal and run the command base64 -d A faster way to decode the string is toecho the copied text and pipe it to base64 -d, like this: echo cGljb0NURnt0aGVfbTN0YWRhdGFfMXNfbW9kaWZpZWR9 | base64 -d
- The flag will be displayed

11.Binary search General skill

Welcome to the Binary Search Game! * I'm thinking of a number between 1 and 1000.

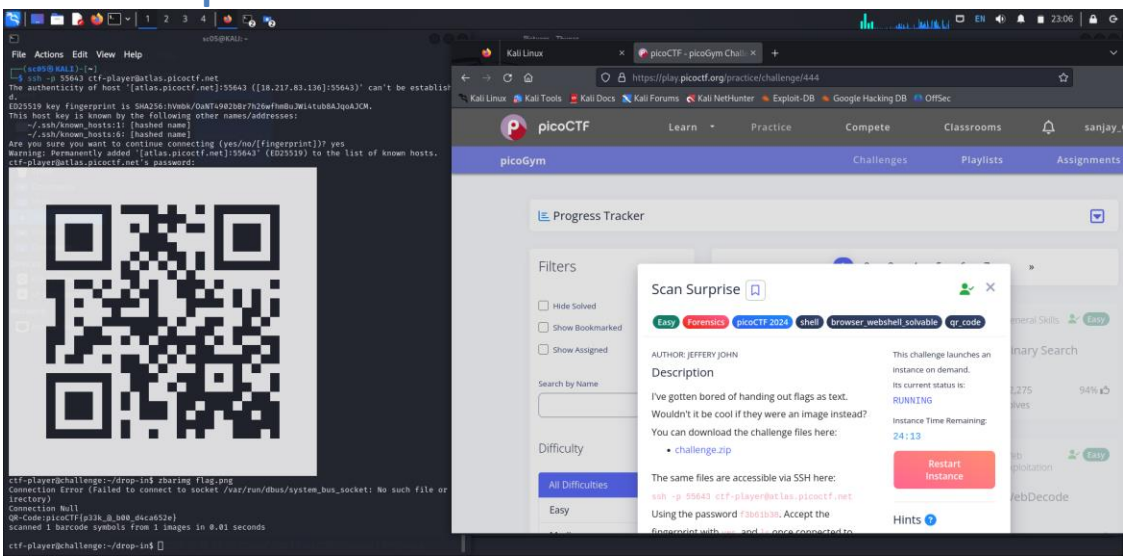
- The first guess is going to always be 500 because it is the middle of 1 and 1000.
- For each guess after 500 the known lower bound is taken and the known higher bound is taken and divided by two for the next guess.

12.Inspect html web exploitation



- we have to enter to the wesite and right click .Then select inspect.go to html code and search for flag displayed there.

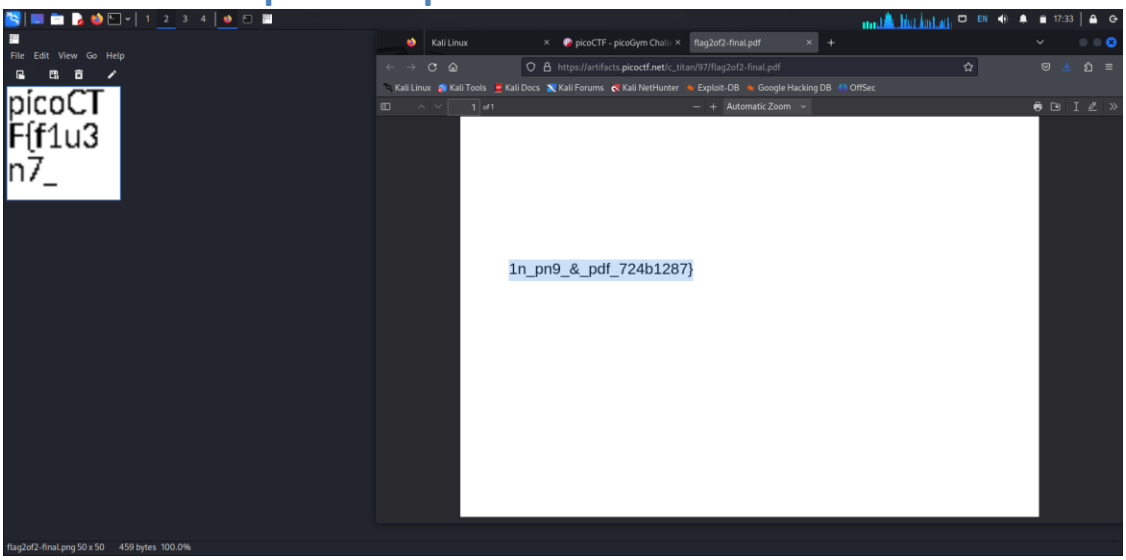
13.scan surpriseforensics



- To get the file: `wget https://artifacts.picoctf.net/c_atlas/3/challenge.zip`, then `unzip challenge.zip`. Note: The files are also accessible with the provided ssh in the description. Use `cd home/ctf-player/drop-in` to get to `flag.png`.
- Once there you can open the image and use a phone to scan the QR code and get the flag. Although it could also be done in Linux with zbar-tools.

- First install the package with `sudo apt install zbar-tools` then to use it run this command: `zbarimg flag.png`

14.Intro to burp web exploitation



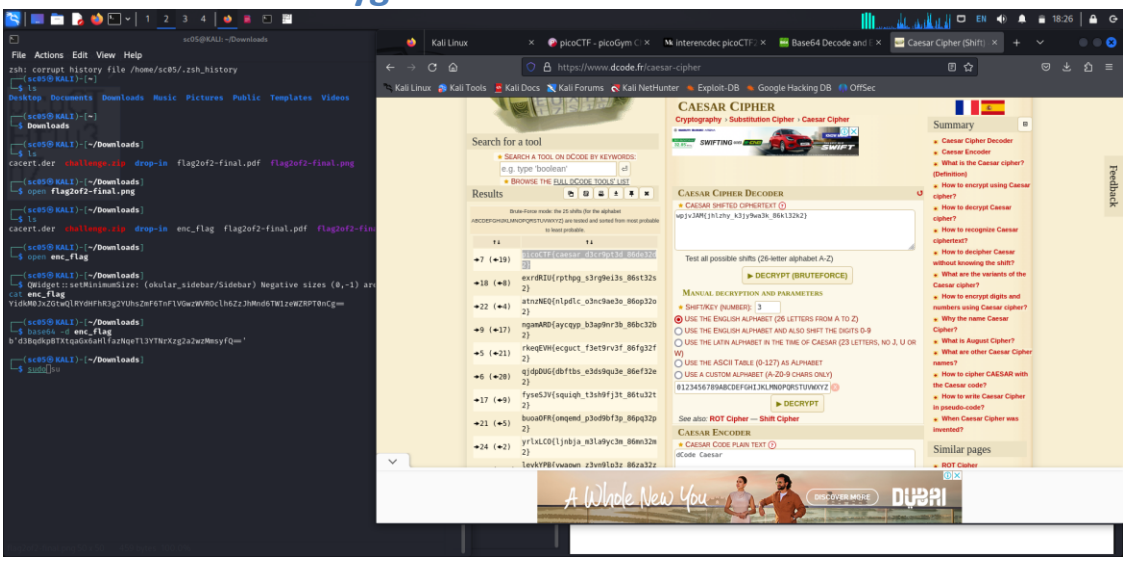
- Open BurpSuite and the proxy web browser with the link provided in the challenge description: http://titan.picoctf.net:49297/

For the first page, it doesn't matter the data you put in. You could put all values to anything and then click "Register". Now on the OTP page turn the "Intercept" function to on in BurpSuite.

Doesn't matter what is put for OTP. In the intercept now remove the text on line "otp=" but don't remove any spaces/lines just the text from the otp.

- When forwarding this request the website gives the flag.

15.Secret of the Polyglot forensics

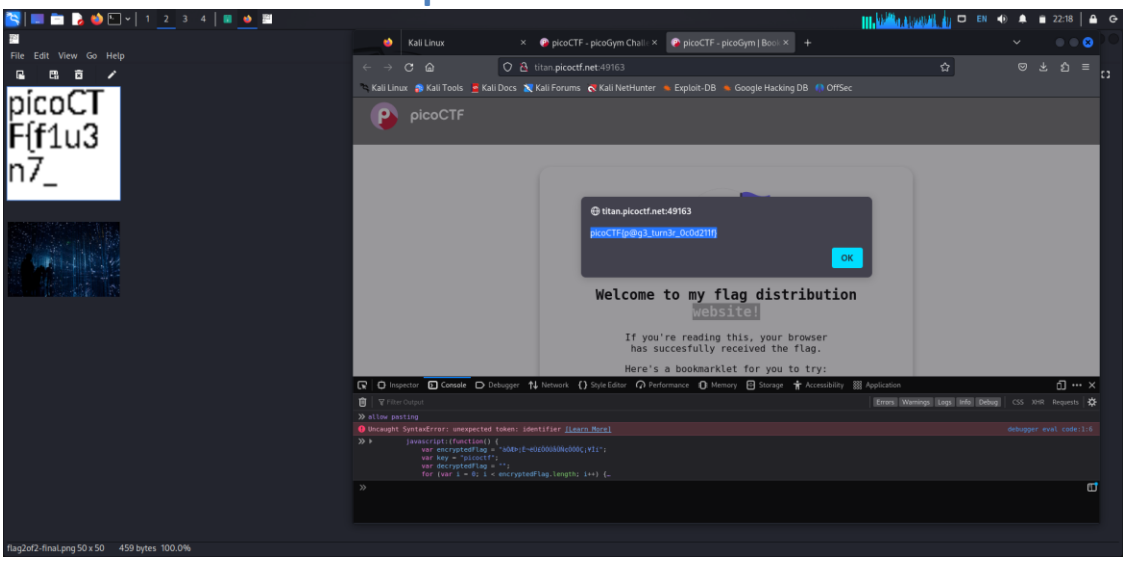


- To get the file: `wget https://artifacts.picoctf.net/c_titan/9/flag2of2-final.pdf`
- First, open it as a pdf to get the 2nd part of the flag. Through the command line, it could be done with `pdftotext` command.
- First to install use, `sudo apt install poppler-utils`, then to run the command: `pdftotext flag2of2-final.pdf`
- Then to get the flag use, `cat flag2of2-final.txt`, to get this: `1n_pn9_&_pdf_7f9...}`

When looking at the file with `cat flag2of2-final.pdf`, looking through the hex, or running the file command with `file flag2of2-final.pdf` it could be seen that the magic bytes show the file as a png. By changing the name with this command, `mv flag2of2-final.pdf flag2of2-final.png`, the file could be opened as a png and the first part of the flag could be read.

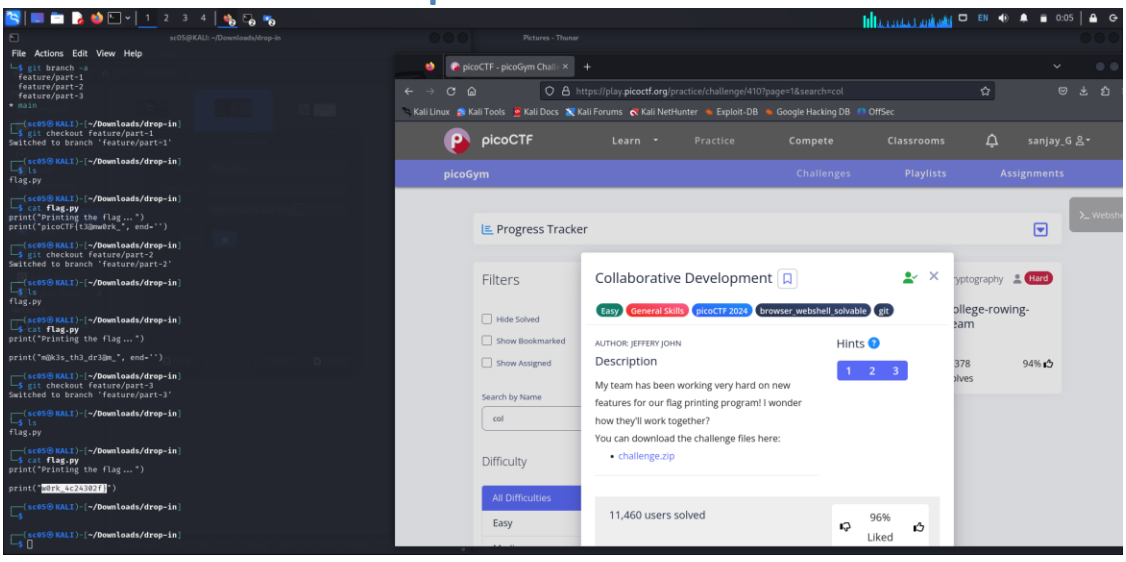
Doing it through the command line Optical Character Recognition (ocr) tools could be used. To download a well-known one, `sudo apt install gocr`, then `gocr flag2of2-final.png | tr -d "\n"` to remove the new lines and paste the contents. This gives `piconF{f1u3n7_` which is mostly right other than it regonizing an n instead of CT

16.bookmarklet web exploitation



- When you go to the site JavaScript code is shown and displayed as a “bookmarklet for you to try”. There are multiple ways to approach this.
- A bookmarklet could be created by bookmarking/starring the page and then editing the bookmark to put the JavaScript code in place of the URL field. Then when loading the bookmark it will give you a pop-up with the flag. Note that in Chrome to edit the bookmark you have to click the bookmark to edit it and under “Folder” go to “Choose another folder…” to see these options.

17.Collaborative DevelopmentGeneral skill



- To get the file: `wget https://artifacts.picoctf.net/c_titan/71/challenge.zip`. Then `unzip challenge.zip` and `cd drop-in/`.
- With `git branch -a` all the current branches could be seen. There are three feature branches and each one has a part of the flag. You could go to each one and retrieve the flags or you could merge them all to main and * deal with the merge conflicts. This is a command that prints all feature branches at once:

```
git checkout feature/part-1 && cat flag.py && git checkout feature/part-2 &&
cat flag.py && git checkout feature/part-3 && cat flag.py
```