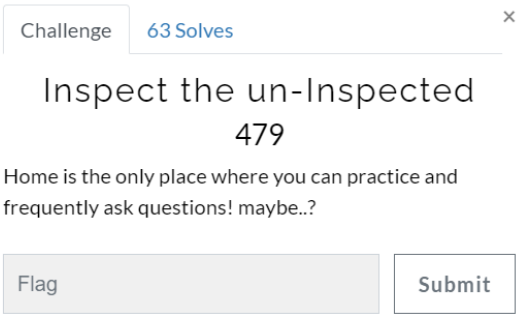
Team Kali Billi
(VishwaCTF 2021 Writeup)

WEB

Inspect the un-Inspected



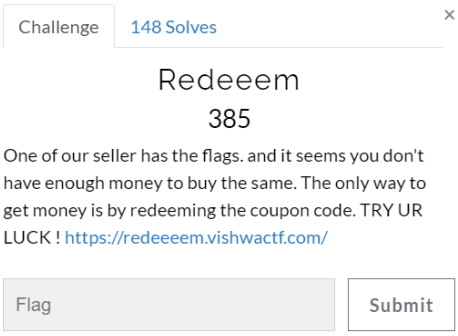
Steps:

- From questions I picked up word Home, Practice and FAQs.
- Flag was divided into three parts.
    - 1/3 flag : vishwaCTF{EvEry_ at Home (https://vishwactf.com/) source code.
    - 2/3 flag : C0iN_ha$ in Practice-Flag (https://play-vishwactf-mini.ml/flag) source code in hidden (black) format “ C0iN_ha$ ”
    - 3/3 flag : _3_s1Des} in GitHub repository FAQ file.

FLAG : **vishwaCTF{EvEry_C0iN_ha$__3_s1Des}**

By: -
Hardik Jain
Tejsi Sharma
Yogesh Khandekar

Redeeem

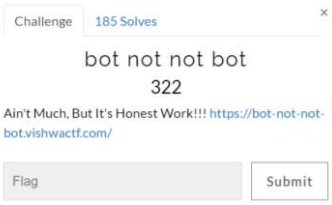


Steps:

- Opened BurpSuite and Intercepted the request.
- Changed the wallet money to 6969.
- It passed the request and showed the flag.

FLAG : **vishwaCTF{@DDed_T0_C@rT}**

bot not not bot



Steps:

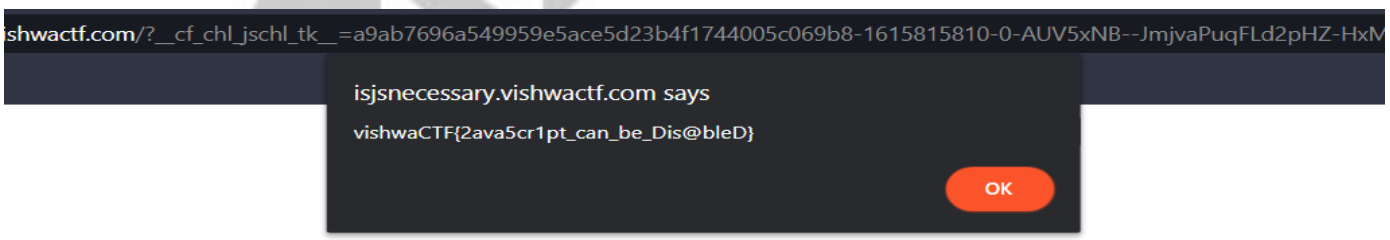
- Parts of flag were hidden in few links from the 500 different links which were present On this website https://bot-not-not-bot.vishwactf.com/
- Made a custom script for web scraping
- Then found the Flag

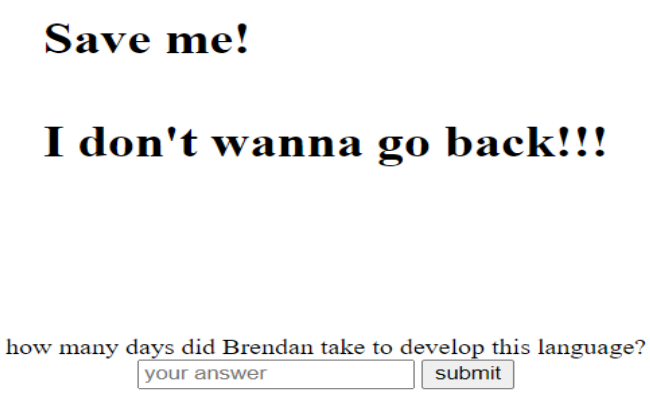
FLAG :- vishwaCTF{r0bot_15_t00_0P}

Is Js Necessary?

Steps:

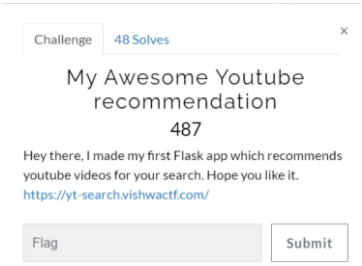
- Click the challenge a popup will come with link.
- After clicking the link it will redirect to google.com in fractions of seconds
- To check why this is happening click the link and trap the request in burpsuit and send it to burp repeater then you will see a JS function is causing this redirection in about 200 ms
- To stop this function being executed block the scripts from browser's setting
- First click the link then cloudfare will do some authentication for 5 sec after
- That 5 sec block the JS from browser setting after blocking the link will not redirect to google.com instead it will stay on original website
- When original website will not redirect then unblock scripts
- On original website one question was given after answering (answer was 10)the Question flag will be popup





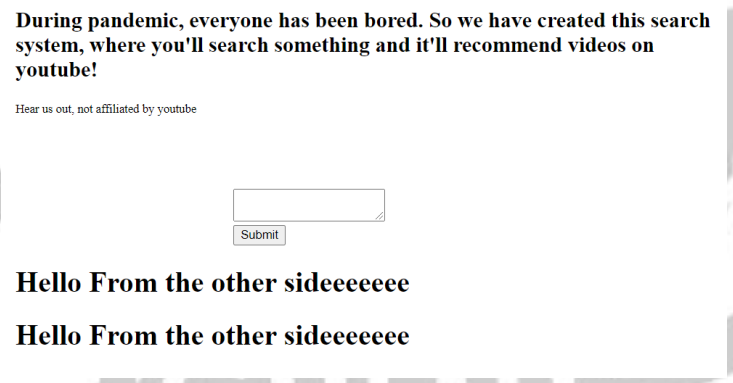
FLAG :- **vishwaCTF{2ava5cr1pt_can_be_Dis@bleD}**

My Awesome YouTube recommendation

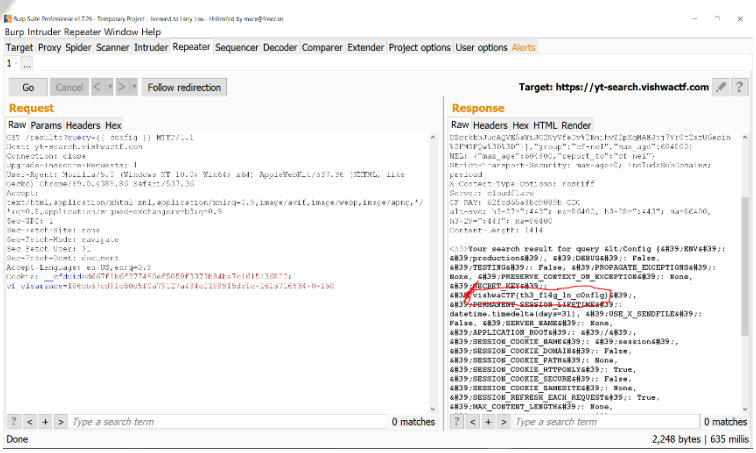


Steps:

- Open the challenge and go to given link in the challenge

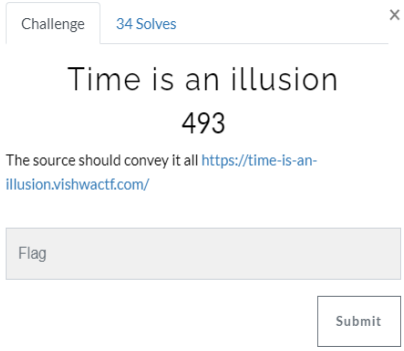


- Type vishwactf or anything in the input box and trap the req in burpSuite and then forward the req to burp repeater and change the Query of url to "{{ config }}" which is payload to get flag from the website
- After changing the query, you will be prompted to website where you will get the flag



FLAG :- **vishwaCTF{th3_f14g_ln_c0nflg}**

Time is an illusion



Steps:

- First let's visit to the source.
- Looking at the source it seems like we'll have to brute force.
- As it says that the key is 5 characters long including numbers and characters both capital and small, so we prepared a script to brute force it.
- After some time, the script gave us the output as "KuKa9".
- Entering it in the input box we get the flag.

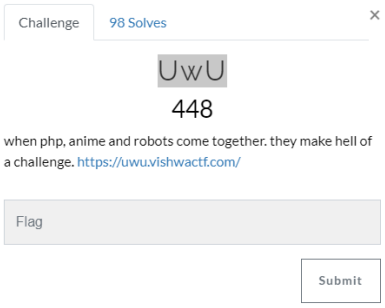
```php
<?php
include "env.php";
set_time_limit(0);
if(isset($_GET['key'])){
    $your_input = $_GET['key'];
    if(strlen($your_input)!=5){
    die("It's five characters long, and may or may not contain numbers or characters or small letters or
    }
    for($i=0; $i<strlen($your_input);$i++)
    {
        $let_check=$_ENV['our_input'];
        if($your_input[$i]!=$let_check[$i]){
            die("VERY VERY VERY INCORRECT");
        }
        usleep(1000000);
    }
    echo getenv('THE_THING_YOU_ARE_LOOKING_FOR');
}
?>
```

vishwaCTF{PhP_h@$_iTs_0wN_PErK$}

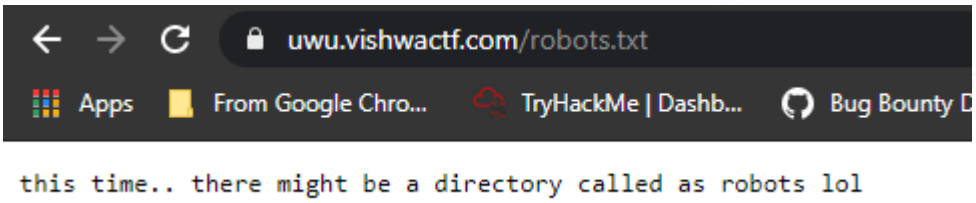
- Flag = **vishwaCTF{PhP_h@$_iTs_0wN_PErK$}**

UwU

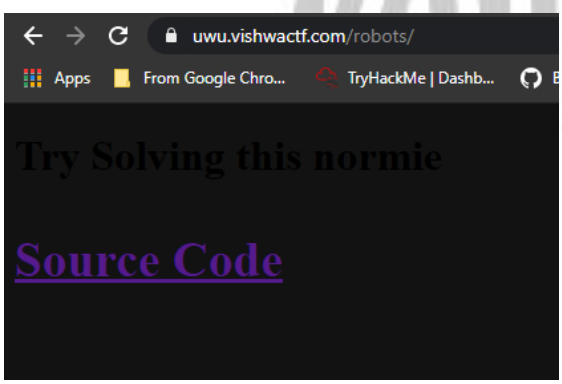
Steps:



- When you open the challenge link, there is some anime site where a video is playing. Nothing useful in that site, I guess.
- But then as the challenge suggest “robots” so that make me go to robots.txt.



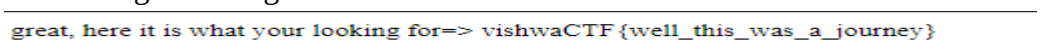
- So as robots.txt suggest, there is a directory named “robots”, which when visited looks something like this.
  Here, we find some source code which is in php with a preg_replace function in it.




```php
<!DOCTYPE HTML>
<?php
  require("cyberflagster.php");
  $try; //1
  $function; //5
  if (isset($_GET['showThem'])) {

    highlight_file(__FILE__);

    die();

  }
   $reach;//3
  if (isset($_GET['php_is_hard'])) {

    $you_enter = $_GET['php_is_hard'];

    $we_enter = 'suzuki_harumiya';

    $the_final_one = preg_replace(

      "/$we_enter/", '', $you_enter);

      if ($the_final_one === $we_enter) {

        open_up();
    }
  }
  $to; //2
  $open_up;//4
?>
```

- Read more about preg_replace over here.
- So using preg_replace function and a little brainstorming we edited the url of the challenge to this GET request

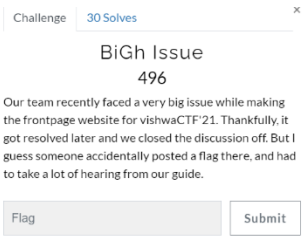
  https://uwu.vishwactf.com/robots/?php_is_hard=suzuki_suzuki_harumiyaharumiya
- There we got the flag.



- Flag = **vishwaCTF{well_this_was_a_journey}**

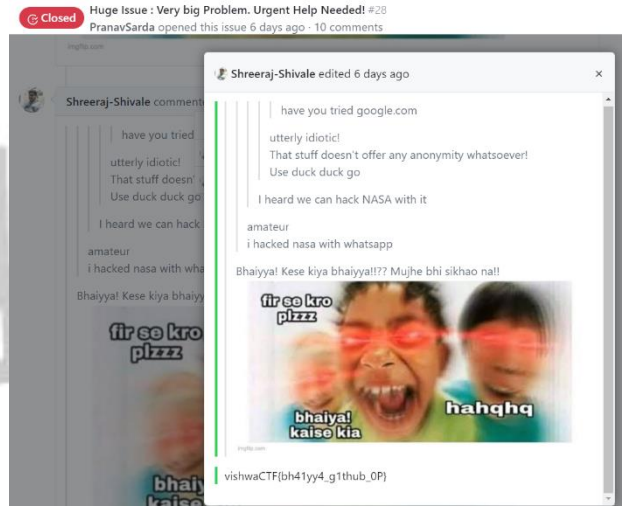
General

BiGh Issue



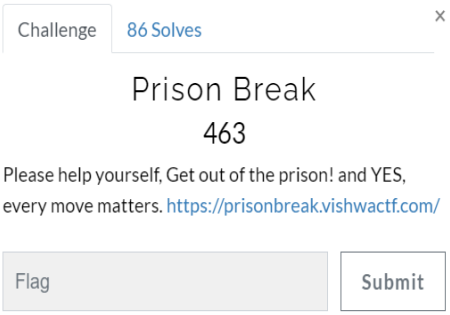
Steps:

- Found the GitHub repository of VishwaCTF2021 (https://github.com/CybercellVIIT/vishwaCTFWebsite21/issues/28)
- Found "Huge Issue : Very big Problem. Urgent Help Needed! " closed issues in insights.
- There people were talking about the "Big Issue"
- In last there was a post that was edited
- Found bug by seeing the edited history of that post
-



FLAG : **vishwaCTF{bh41yy4_g1thub_0P}**

Prison Break

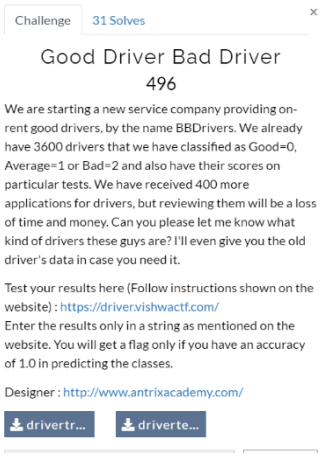


Step:

- I followed the story line and found the flag.

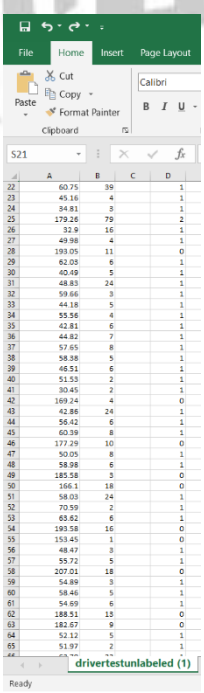
- Flag = **vishwaCTF{G@mE_Of_DeC1$ions}**

Good Driver Bad Driver



Steps:

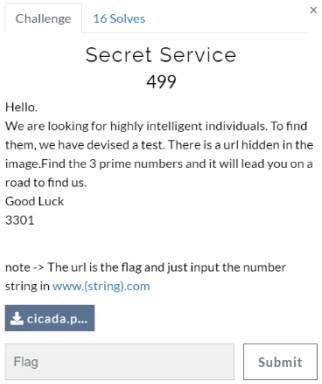
- I figured out the pattern of grading.
- Found out
    - If Distance_feature < 100 ; Rating = Average =1
    - If Distance_feature > =100
        - Speeding_Feature <50 ; Rating = Good=0
        - Speeding_Feature >=50 ; Rating = Bad=2
- Sorted out list of drivertestunlabbled and ranked them according to the above. Later I used MS Excel functions to unsort.



- Made a string of the rankings and submitted.

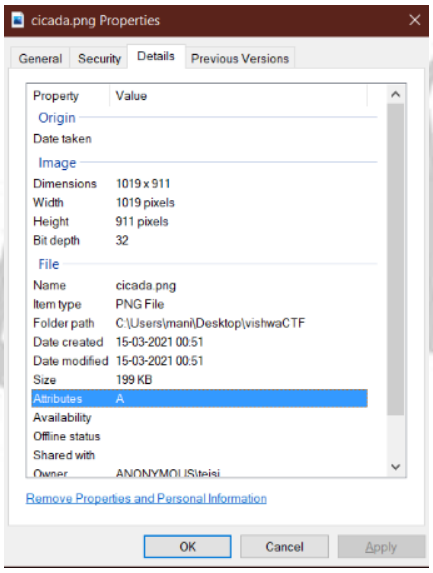
FLAG : **vishwaCTF{d4t4_5c13nc3_15_n3c3554ry}**

Secret Service



Steps:

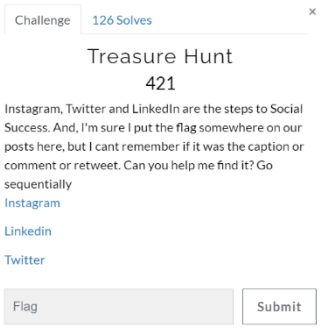
- I opened the description of the image and found 2 prime number in description (width-1019 and breadth-911) and third prime number was in question – 3301



- I multiplied all these three prime numbers which came out-3064348009

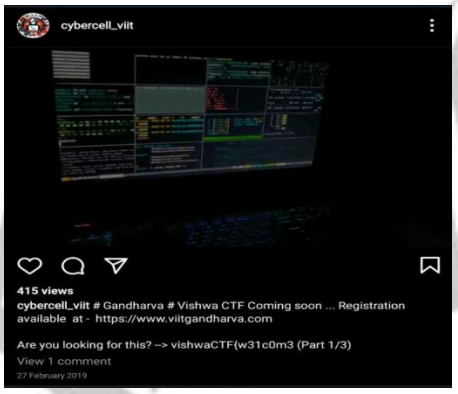
FLAG : **vishwaCTF{www.3064348009.com}**

Treasure Hunt

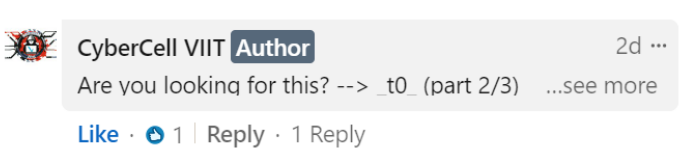


Steps:

- On 1st post of Instagram (https://www.instagram.com/p/BuY2bhkhboO/?igshid=1bxmneomcaevr) I found 1st part of flag.



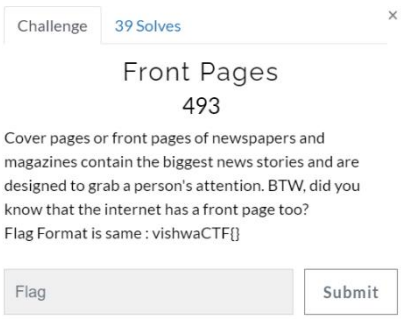
- On LinkedIn post (https://www.linkedin.com/posts/cybercell-viit_cybersecurity-security-cyber-activity-6705436186833387521-aghB) i found 2nd part of flag



- On twitter post (https://twitter.com/cybercellviit/status/1299671113039970305) I found 3rd part of the flag.

FLAG : **vishwaCTF{w31c0m3_t0_v1shw4ctf}**

Front Pages



Steps:

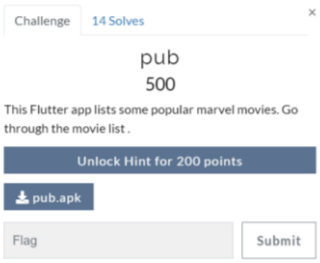
- Reddit is the front page of internet so searched VishwaCTF on reddit and found a post but something was deleted from it.
- So checked it through “Wayback Machine” and got this.

> *Well, so I'm just gonna drop the flag here, and I certainly hope no hackers get access to this.*
>
> *So here goes:*
>
> *vishwaCTF{0$dVl_1z_kFV3g_0a3mT0graD}*
>
> *But hang on, this text doesn't make sense, does it?*
>
> *Oh well, it is what it is I guess.*
>
> *Good luck for rest of the challenes!*
>
> *P.S. 18th century French scholars deserve more recognition!*

- Using vignere cipher decoder we encoded flag (i.e 0$dVl_1z_kFV3g_0a3mT0graD). In “text to decode” block and enter “VISHWACTF” as key in Key block and got the flag in the output.

Flag : **vishwaCTF{0$int_1s_oft3n_0v3rl0oked}**

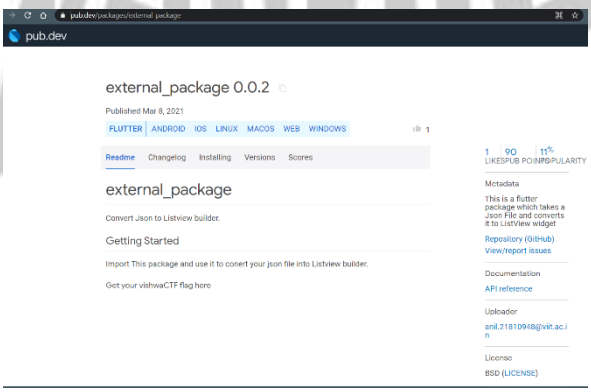
Pub



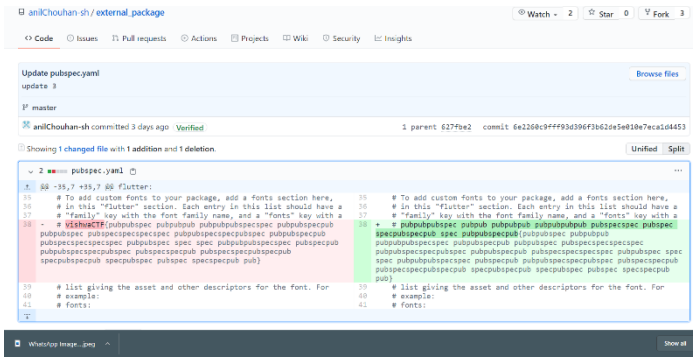
Steps:

1. Downloaded the given “pub.apk” and searched the contents of file.
2. It had a long list of Marvel Movies but one odd file named “external_package”

3. I searched for “pub flutter” and got a website (https://pub.dev/) where I searched for this “external_package” which made me land on (https://pub.dev/packages/external_package)

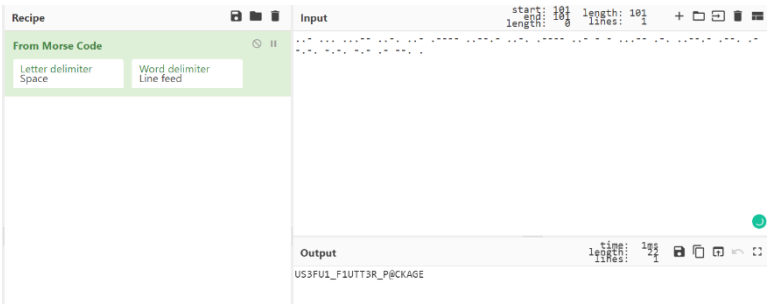


4. It had a GitHub repo link on right. On navigating there (https://github.com/anilChouhan-sh/external_package) I found a file " pubspec.yaml” which had editing history and encoded flag.



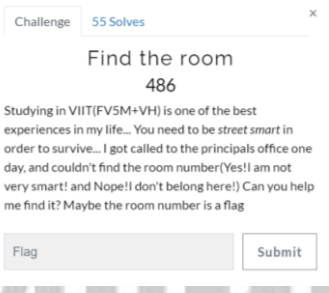
5. After analysing these edits we figured out **pub = .** and **spec = -** using which we converted whole flag to morse code.

6. Then using the online converter we converted the morse code and got the flag



FLAG : **vishwaCTF{US3FU1_F1UTT3R_P@CKAGE}**

Find the room

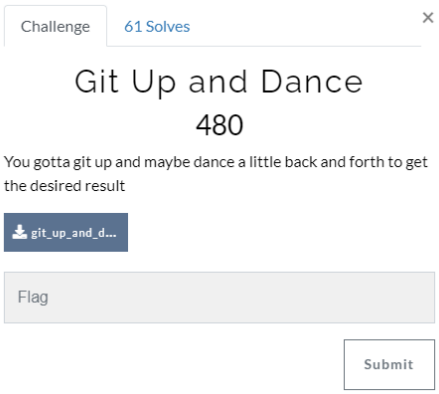


Steps:

1. Googled “FV5M+VH,pune” and zoomed google map
2. Changed to street view and found the room

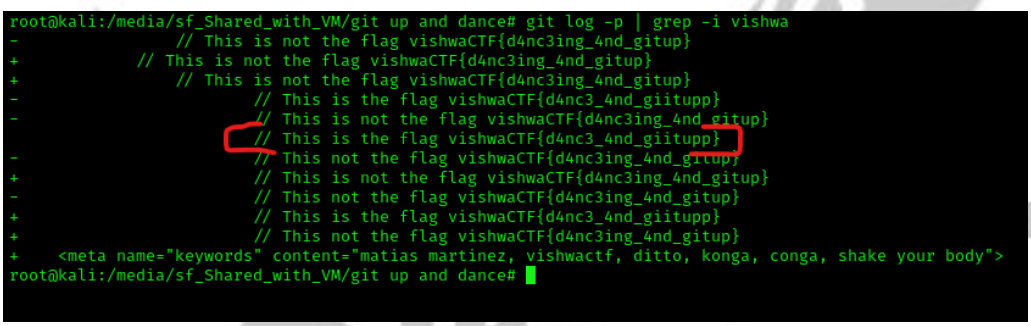
FLAG : **vishwaCTF{A 003}**

Git Up and Dance



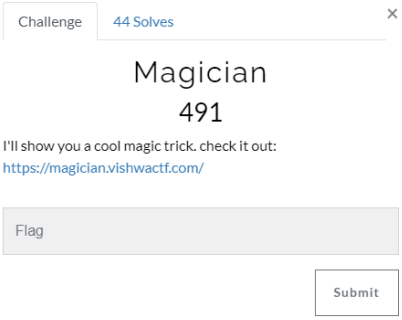
Steps:

- Extract the zip file from the question.
- You can see that this folder contains many files and an index file.
- We use “git log -p | grep -i vishwa” to see the log and find the flag from it.



- Flag = **vishwaCTF{d4nc3_4nd_giitupp}**

Magician



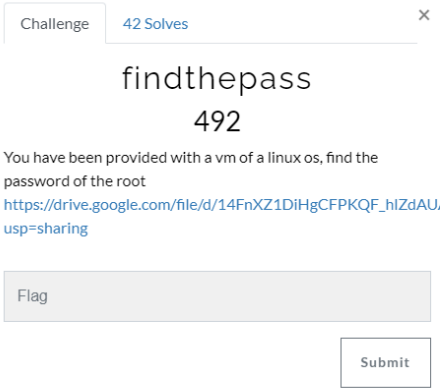
Steps:

- The site gives us one letter of a flag at a time and it takes 20-25 minutes for the next letter but do we have time?
- We automate the task using a script.

```
import requests, re
flag = []
for i in range(40):
    flag.append('')
while(True):
    s = requests.Session()

    html = s.get('https://magician.vishwactf.com/%27).text

    symbol = re.search("<body>\s*<h1>\s([a-zA-Z0-9_])\s</h1>",html).group(1)
    pos = re.search("(\d+)+\s</h1>\s<h3>", html).group(1)
    flag[int(pos)] = symbol
    print(''.join(flag))
```

- Now we get the flag as **vishwaCTF{cr0nj0bs_m4k3_l1f3_s1mp13}**

Findthepass

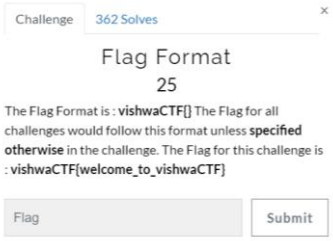


Steps:

- Install the vm given in the file.
- There's a dump of with 10 passwords.
- Convert the hash and try each with admin you will the right password that is "password"
- Flag = **vishwaCTF{password}**

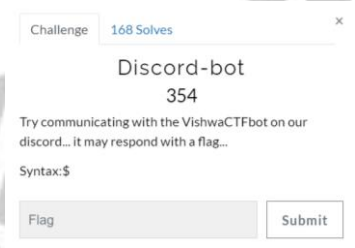
Warmup

Flag Format



Flag : **vishwaCTF{welcome_to_vishwaCTF}**

Discord-bot



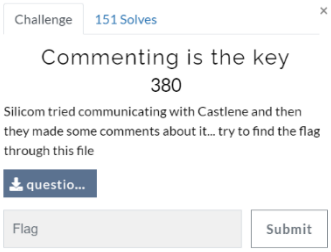
Steps:

- messaged “$flag” to VISHWACTFBOT and it gave the flag.

Flag: **vishwaCTF{d15c0rd_5p1ll3d_th3_b34n5}**

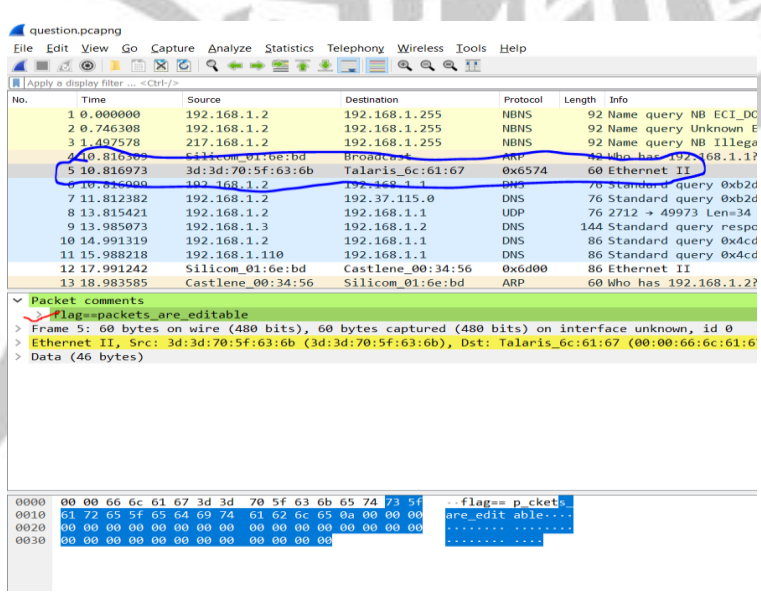
Networking

Commenting is the key



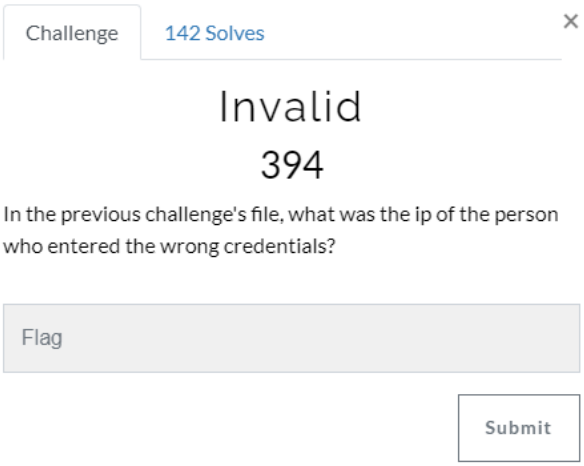
Steps:

- open the challenge and download the file given in challenge
- Open the file in wireshark then go through each packets
- Flag was hidden in the packets comments of 5th packet



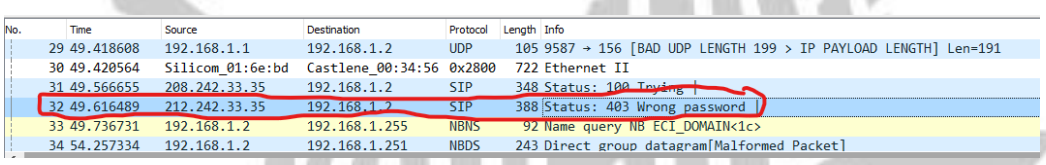
Flag : **vishwaCTF{packets_are_editable}**

Invalid



Steps:

- Open the pcap file in wireshark
- Search for SIP protocol and you will find the IP address of the user who input the wrong password.

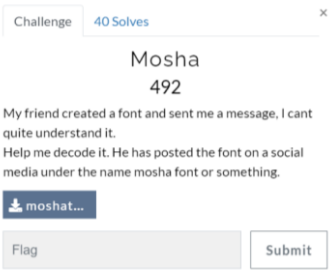


No.	Time	Source	Destination	Protocol	Length	Info
29	49.418608	192.168.1.1	192.168.1.2	UDP	105	9587 → 156 [BAD UDP LENGTH 199 > IP PAYLOAD LENGTH] Len=191
30	49.420564	Silicom_01:6e:bd	Castlene_00:34:56	0x2800	722	Ethernet II
31	49.566655	208.242.33.35	192.168.1.2	SIP	348	Status: 100 Trying |
32	49.616489	212.242.33.35	192.168.1.2	SIP	388	Status: 403 Wrong password |
33	49.736731	192.168.1.2	192.168.1.255	NBNS	92	Name query NB ECI_DOMAIN<1c>
34	54.257334	192.168.1.2	192.168.1.251	NBDS	243	Direct group datagram[Malformed Packet]

- Flag = **212.242.33.35**

Cryptography

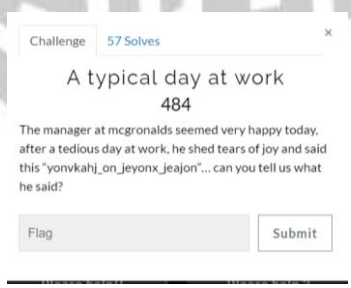
Mosha



- The language given in the image was weird but we had to find it on some social media platform to decode it.
- Found the Mosha font on Instagram with username *@mosha_font* and it had 2 posts with the font script.
- Compared the font with the image and got the flag

Flag :- **vishwaCTF{Y0u4rem05hAnoW}**

A typical day at work

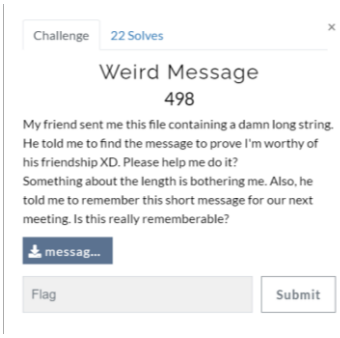


Steps:

1. We searched "mcgronalds" and found anime reference of 'hataraku'.
2. Figured out the "Entean language" is used in anime which is a substitution cypher with the help of this blog (https://hataraku.fandom.com/f/p/3009718557606080307)
3. Decoded the statement "yonvkahj_on_jeyonx_jeajon"

FLAG: **vishwaCTF{congrats_on_second_season}**

Weird Message



Steps:

1. Opened the file given in notepad and zoomed out.
2. It seemed to us like something is drawn.
3. Figured out it was a binary image
4. Used https://www.dcode.fr/binary-image to decode it and got the flag.

FLAG: **vishwaCTF{pr1m35_4r3_w31rd}**

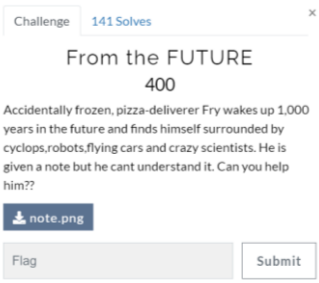
Can you see??

Steps:

- Used Braille Notation to decode the content of the given file, were 1=upper dots and 0=lower dots

FLAG: **vishwaCTF{vvho n33ds 3y3s 7o 5ee}**

From the FUTURE



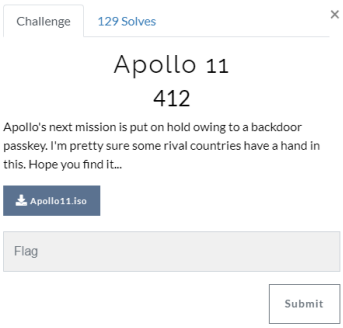
Steps:

- We searched termed from question like “frozen”,”pizza-dilivery Fry” by which we got to know about Futurama.
- Decoded the given file using Futurama Decoder (https://www.dcode.fr/futurama-alien-alphabet) and got the flag.

FLAG: **vishwaCTF{WEARENOTALONE}**

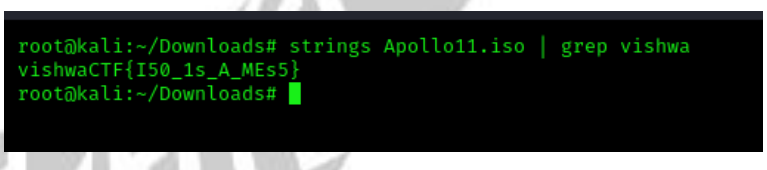
Reverse Engineering

Apollo 11

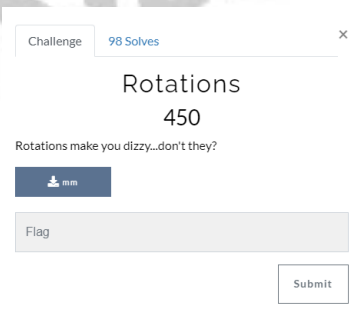


Steps:

- Strings Apollo11.iso to find the flag.
- Use "*strings Apollo11.iso | grep Vishwa"*
- Flag = **vishwaCTF{I50_1s_A_MEs5}**

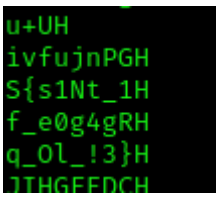


Rotations



Steps:

- Check file type by "file mm". It is a ELF file.
- "strings mm" and we see some format which is similar to our flag format. ivfujnPGS{s1Nt_1f_e0g4gRq_Ol_!3}.
- Not the challenge name hints about it being ROT encoded.
- We decoded it using ROT13 and got the flag.
- Flag= **vishwaCTF{f1Ag_1s_r0t4tEd_By_!3}**



Misleading Steps

Steps:

- Used strings to find it lol, it was a mislead vishwaCTF{1_0ft3n_M1sl3ad_pPl}.
- Now used Ghidra and saw the function main.
- There was some hex code with it "766973687761435446 7b556d4d5f77336952446f6f6f305f416d5f7468335f7233346c5f306e337d"
- Decoded it and got the flag.

```
root@kali:/media/sf_Shared_with_VM# echo 766973687761435446 7b556d4d5f77336952446f6f6f305f416d5f7468335f7233346c5f306e337d | xxd -p -r
vishwaCTF{UmM_w3iRDooo0_Am_th3_r34l_0n3}root@kali:/media/sf_Shared_with_VM#
```

- Flag = **vishwaCTF{UmM_w3iRDooo0_Am_th3_r34l_0n3}**

Facile

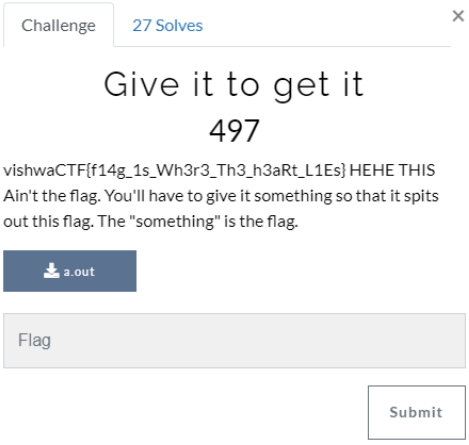
Steps:

- Use binwalk to extract the file "binwalk -e s1mple.gzf"
- There is a file names "FOLDER_ITEM".
- Strings it and you get the flag.

```
root@kali:/media/sf_Shared_with_VM/_s1mple.gzf-0.extracted# strings FOLDER_ITEM | grep -i vishwa
vishwaCTF{r3v_1t_1s5s5s}
root@kali:/media/sf_Shared_with_VM/_s1mple.gzf-0.extracted#
```

- Flag = **vishwaCTF{r3v_1t_1s5s5s}**

Give it to get it

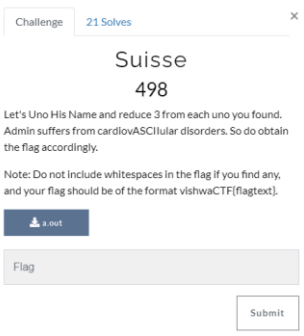


Steps:

```
a = 'vishwaCTF{f14g_1s_Wh3r3_Th3_h3aRt_L1Es}'
a = [hex(ord(i))[2:] for i in a]
print(''.join(a))
```

- This code will take the text from the flag given in the question of the challenge and will give us a hex output.
- Now we have a hex number and that is the flag.
- 766973687761435446 7b663134675f31735f5768337233 5f5468335f68336152745f4c3145737d0
- This hex is the flag.

Suisse



Steps:

- Using Ghidra to open a.out file, you will find a hex code in main function “111 88 107 81 113 93 52 118 56 104 102 88 85 104” and covert it to string “oXkQq]4v8hfXUh”

- Now Convert this in ASCII Number

```
111 88 107 81 113 93 52 118 56 104 102 88 85 104
```

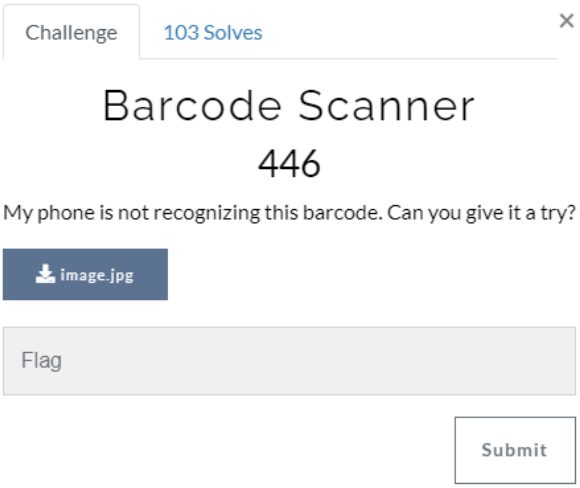
- And then reduce 3 from each number.

```
108 85 104 78 110 90 49 115 53 101 99 85 82 101
```

- Then convert this again to text you will get this “lUhNnZ1s5ecURe”
- Flag = **vishwaCTF{ lUhNnZ1s5ecURe}**

Forensics

Barcode Scanner

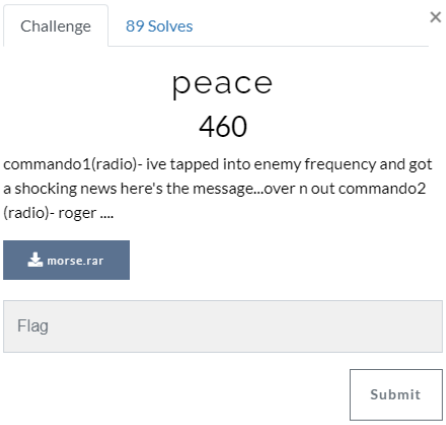


Steps:

- This was the given barcode.

- Download any barcode scanner app from Google play store/Apple store and scan it
- Flag = **vishwaCTF{5oo_3asY}**

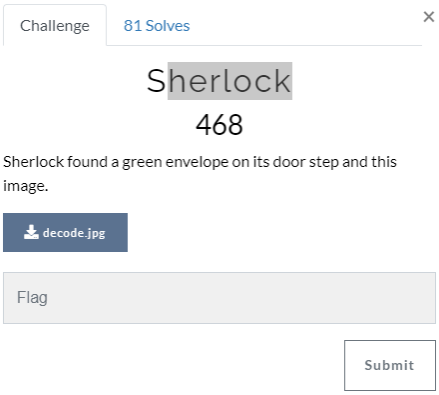
Peace



Steps:

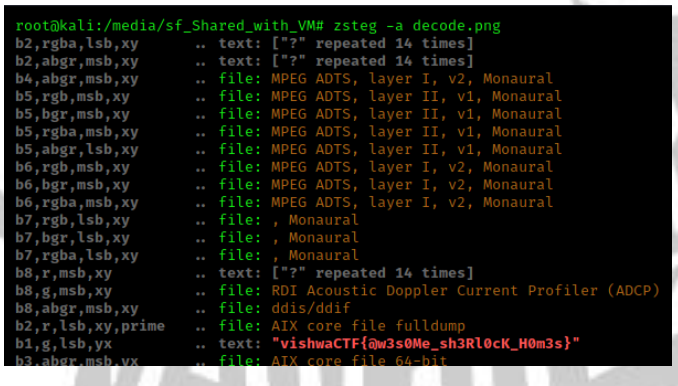
- We have a .rar file here which is password protected. What's the password? We don't have any hint on that.
- Here we could do 2 things, to use an online rar password cracker or run rockyou.txt to bruteforce the password.
- https://www.lostmypass.com/file-types/rar/ Used this site to crack the password of morse.rar and the password came out to be "india".
- After we unrar the morse.rar file we get an audio file which has some morse code in it.
- Let's again use this online morse decoder and upload the file here. https://morsecode.world/international/decoder/audio-decoder-adaptive.html
- We get some hex values
  " 76 69 73 68 77 61 63 74 66 7B 37 68 33 79 5F 34 72 45 5F 46 30 72 33 66 65 37 31 6E 67 7D "
- After converting it to text we get the flag as **vishwactf{7h3y_4rE_F0r3fe71ng}.**

Sherlock



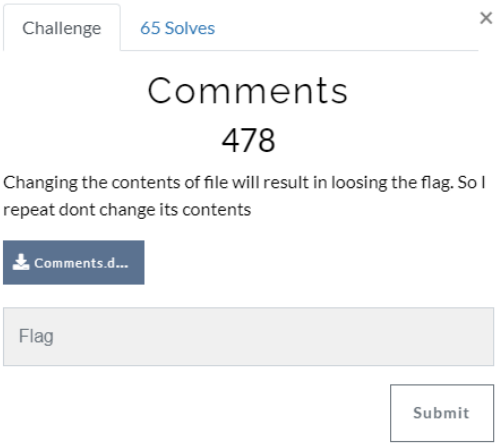
Steps:

- Since it is a png image, we used zsteg to find whats inside it and found the flag.



- Flag = **vishwaCTF{@w3s0Me_sh3Rl0cK_H0m3s}**

Comments

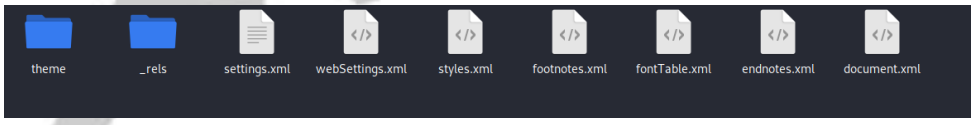


Steps:

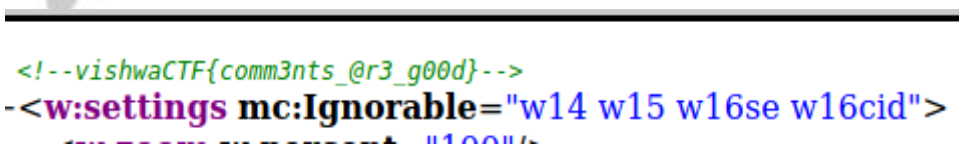
- There was a word file named Comments.docx which at first was not suspicious at all
- When opened it contained this false flag.
-
- Now we did some basic recon o the .docx file by using “file Comments.docx” and it

  vishwaCTF{f@ls3_fl@g}

  came out that this is a OOXML file which is a zip file containing various files in it.
- Extract it using “unzip Comments.docx” in Linux.
- Now you get a bunch of files in extracted



- When you open “settigs.xml” file you will find the flag.

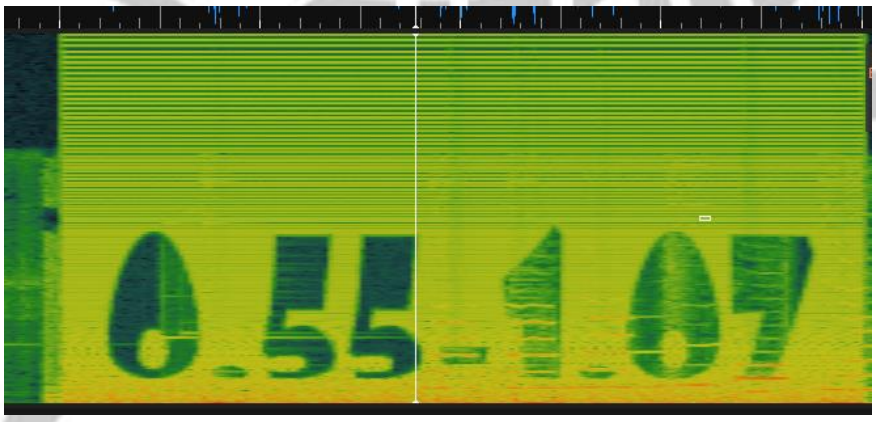


- Flag = **vishwaCTF{comm3nts_@r3_g00d}**

Bubblegum

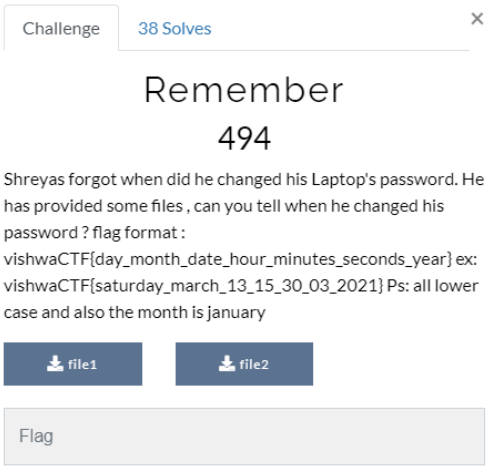
Steps:

- When you listen to bkk.wav in any music player at a certain point of time you will find some distortion in the music.
- Evaluate it in Sonic Visualizer and enable the spectrogram view there we find something written there.



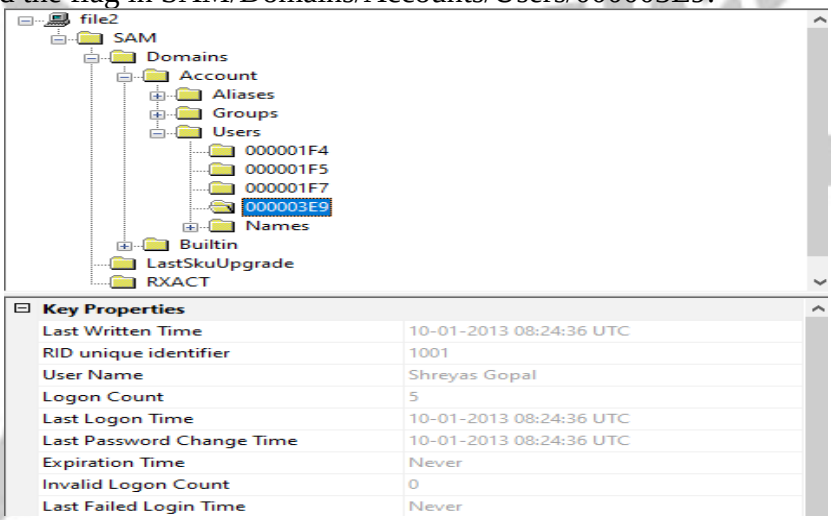
- (0.55 – 1.07) What could this be? The question said that the guy need the lyrics then this can be the timestamp of the lyrics in this music but we can't find lyrics in this audio.
- Simple OSINT on Bubblegum music on YouTube gives us various options but then the name of audio file suggests "bkk.wav" which means bubblegum by kk.
- Found the video there (https://www.youtube.com/watch?v=5x441jo1-sg). It had lyrics and at 0.55 to 1.07 we found the flag which was needed.
- Flag = **vishwaCTF{oh bubble gum dear im yours forever i would never let them take your bubblegum away}**

Remember



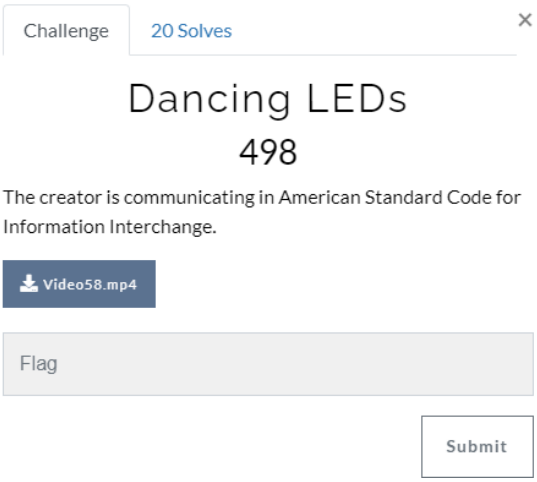
Steps:

- 2 files were given to analyse.
- Using Registry Viewer from access data to analyse the files.
- File2 has a SAM directory we open it.
- We find the flag in SAM/Domains/Accounts/Users/000003E9.



- Flag = **vishwaCTF{thursday_january_10_08_24_36_2013}**

Dancing LEDs



Steps:

- Video with 7 blinking LEDs was given named “Video58.mp4”
- LED’s blinking patterns were Binary Codes where one can see ON LED = 1 and OFF LED = 0
- 0110100 1101001 1110000 1011010 1001010 1001000 1111010 1111000 0110100 0110001 were the binary numbers the LEDs showed us.
- Decoded this to ASCII and got “4ipZJHzx41”. Now this is Base58 as the video name also suggested us “58”. Decode this in CyberChef and we get the flag.
- Flag = **vishwaCTF{b1!nk3r}**