**CTF ( r00t@localhost ) Writeup**

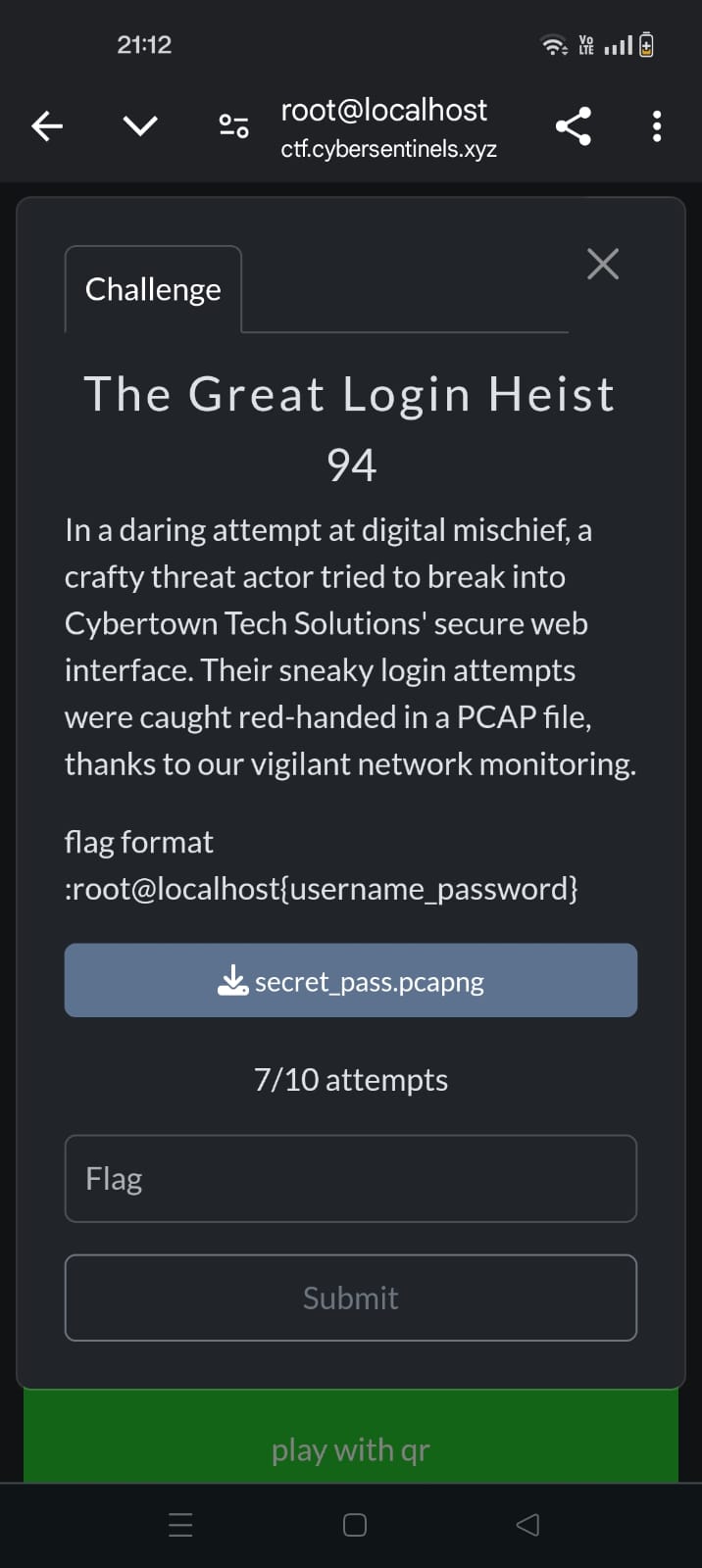
**This documentation describes the techniques, tools, and methodologies utilized to solve the challenges presented during the r00t@localhost CTF. The writeup aims to provide a detailed walkthrough of each challenge, explaining the thought process, strategies, and tools employed to achieve the solutions.**

**Miscellaneous Challenge:**

1. **Welcome :**

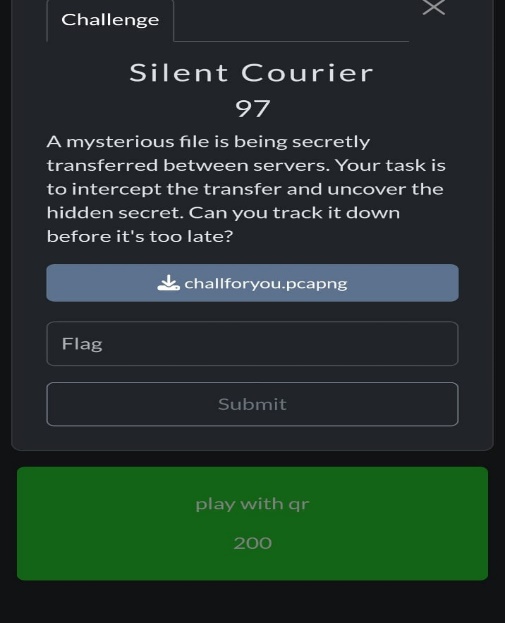
**This challenge was an introduction to this event. The flag was given at the announcement channel on discord . This challenge was just an initial boost to all the competitors.**

1. **The great Login heist :**

****

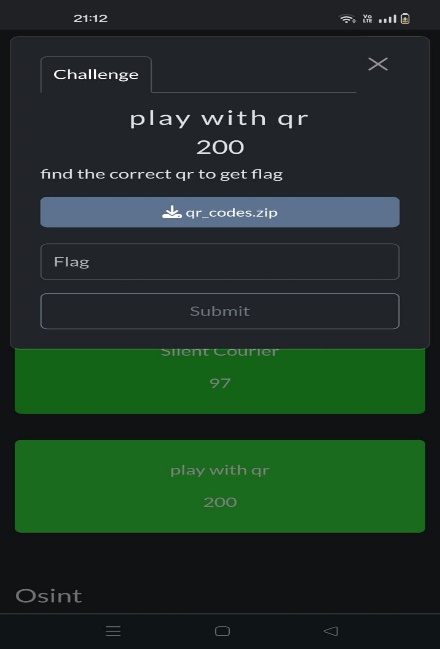
**In this challenge , a pcap file was given and the flag format is mentioned as username\_password , therefore we need to analyse the wireshark file and find the username and password . While following an HTTP request , the username and password was found , Liam\_24 and P%40ssw0rd!2024.**

1. **Silent Courier:**

****

**In this Challenge , the task is to find a secret file that was shared using the pcap file , while analysing the http requests in the pcap file , a protected.zip file was found . We then extracted the zip file and cracked the password using johntheripper tool.**

1. **Play with QR:**

****

**After extracting the zip file , there are 999 qr codes and one of them is the valid one , so I developed a python program to check for each qr code to be valid and found the flag.**

**import cv2**

**import os**

**from pyzbar.pyzbar import decode**

**import requests**

**def is\_valid\_url(url):**

**"""Check if a URL is valid by making a request."""**

**try:**

**response = requests.get(url, timeout=5)**

**return response.status\_code == 200**

**except requests.RequestException:**

**return False**

**def check\_qr\_code(file\_path):**

**"""Decode the QR code and check if it contains a valid URL."""**

**image = cv2.imread(file\_path)**

**decoded\_objects = decode(image)**

**for obj in decoded\_objects:**

**data = obj.data.decode('utf-8')**

**if data.startswith('http://') or data.startswith('https://'):**

**print(f"[INFO] Found URL: {data}")**

**if is\_valid\_url(data):**

**print(f"[VALID] QR Code in '{file\_path}' contains a valid UR>**

**return True**

**else:**

**print(f"[INVALID] URL in '{file\_path}' is not accessible: {d>**

**return False**

**else:**

**print(f"[INVALID] QR Code in '{file\_path}' does not contain a UR>**

**return False**

**print(f"[ERROR] No QR code found in '{file\_path}'")**

**return False**

**def process\_qr\_codes(directory):**

**"""Process multiple QR codes in a directory."""**

**for file\_name in os.listdir(directory):**

**if file\_name.endswith(".png") or file\_name.endswith(".jpg") or file\_>**

**file\_path = os.path.join(directory, file\_name)**

**print(f"\nProcessing '{file\_path}'...")**

**check\_qr\_code(file\_path)**

**if \_\_name\_\_ == "\_\_main\_\_":**

**qr\_codes\_directory = "/home/shufflezz/Downloads/qrcodes/qr\_codes" # Upd>**

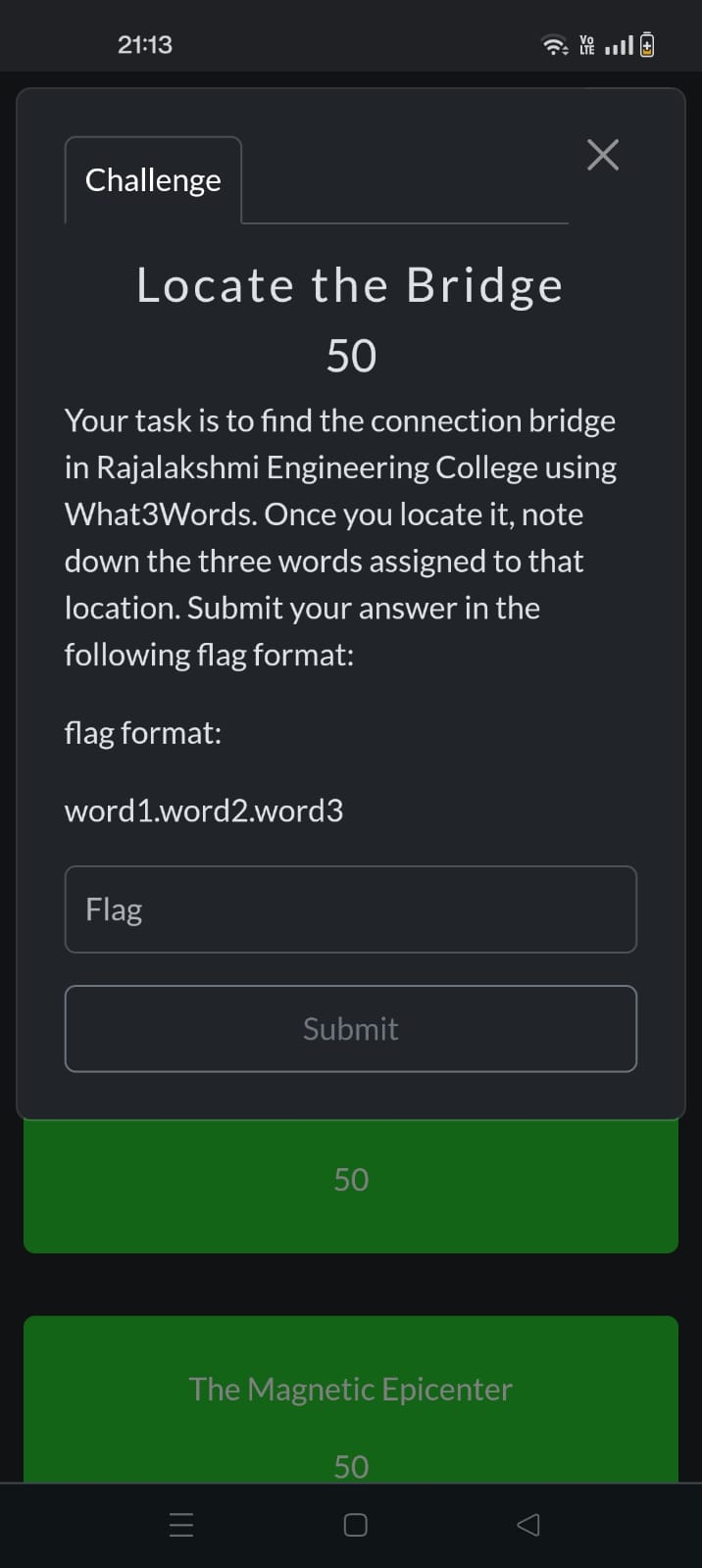
**process\_qr\_codes(qr\_codes\_directory)**

**OSINT:**

1. **Weak:**

**The question is to find the weakest password that is common and insecure , The answer is 12345678.**

1. **Locate the Bridge:**

****

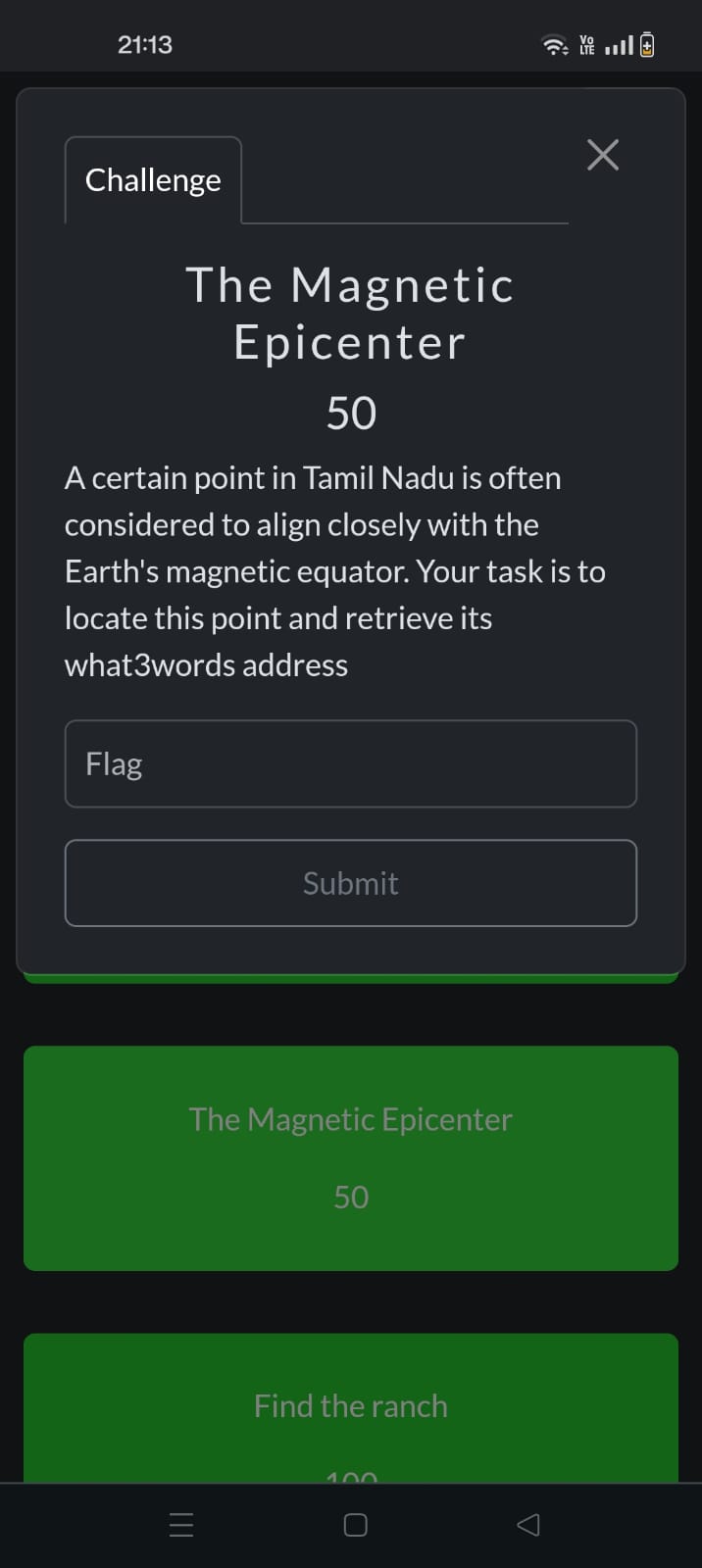
**To find the flag , Went to what3words.com , found the connection bridge in Rajalakshmi Engineering College in the website , the corresponding three words were retrieved as the flag.**

1. **Find The Lab:**

****

**To find the flag , I went to what3words.com , found the Idea lab in Rajalakshmi Engineering College , Retrieved the 3 words as the flag.**

1. **Magnetic Epicentre:**



**Using the clue given in question , I researched and Used Artificial Intelligence to find the place Chidambaram Natraj Temple in tamilNadu . Visited this Place in what3words and retrieved the flag.**

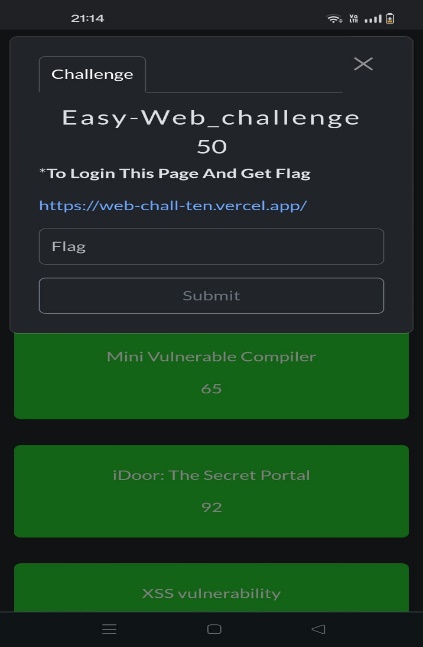
1. **Find the Ranch:**

**The task was to find the location of the image given , Searched the image through the internet and found the flag.**

1. **Cyber Sentinels Hunt:**

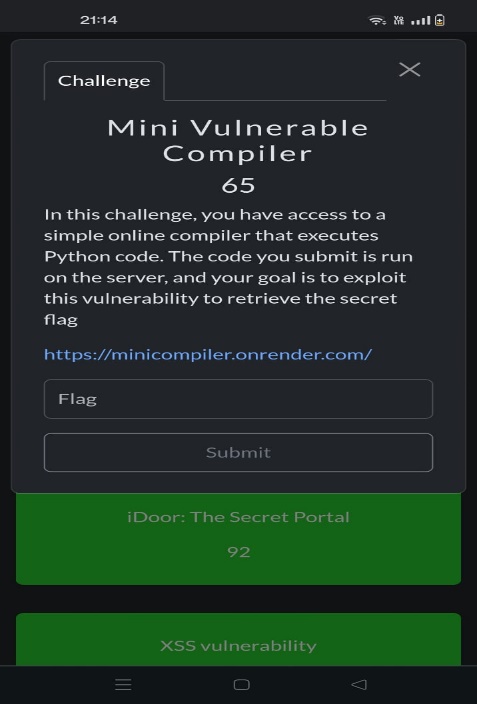
**The clue given to find the flag was to search through the Instagram , discord and linkedin pages of the cyber sentinels team , Found three different encoded texts and decoded them to reveal the flag.**

**Easy Web Challenge:**

****

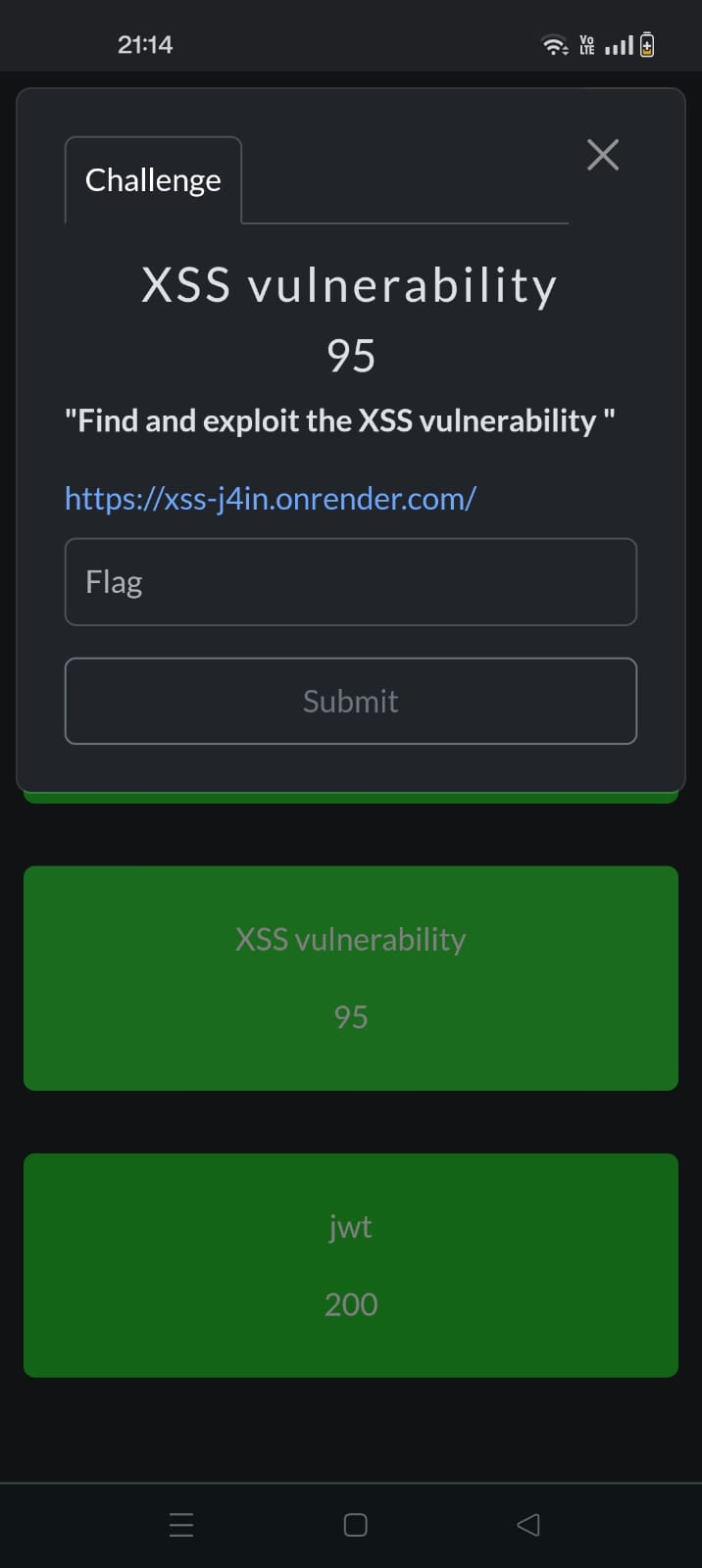
**In this challenge , we visit the website and inspect the page source , there we find a scipt.js file out of place, inside the file , there’s a base64 encoded message, decode it to get the answer.**

**Mini Vulnerable Compiler:**

****

**In this challenge , we visit a complier website which compiles python code , I wrote a command to list all the directories of that web guessing that it is open, Found all directories , found flag.txt in the comp directory.**

**XSS Vulnerability:**

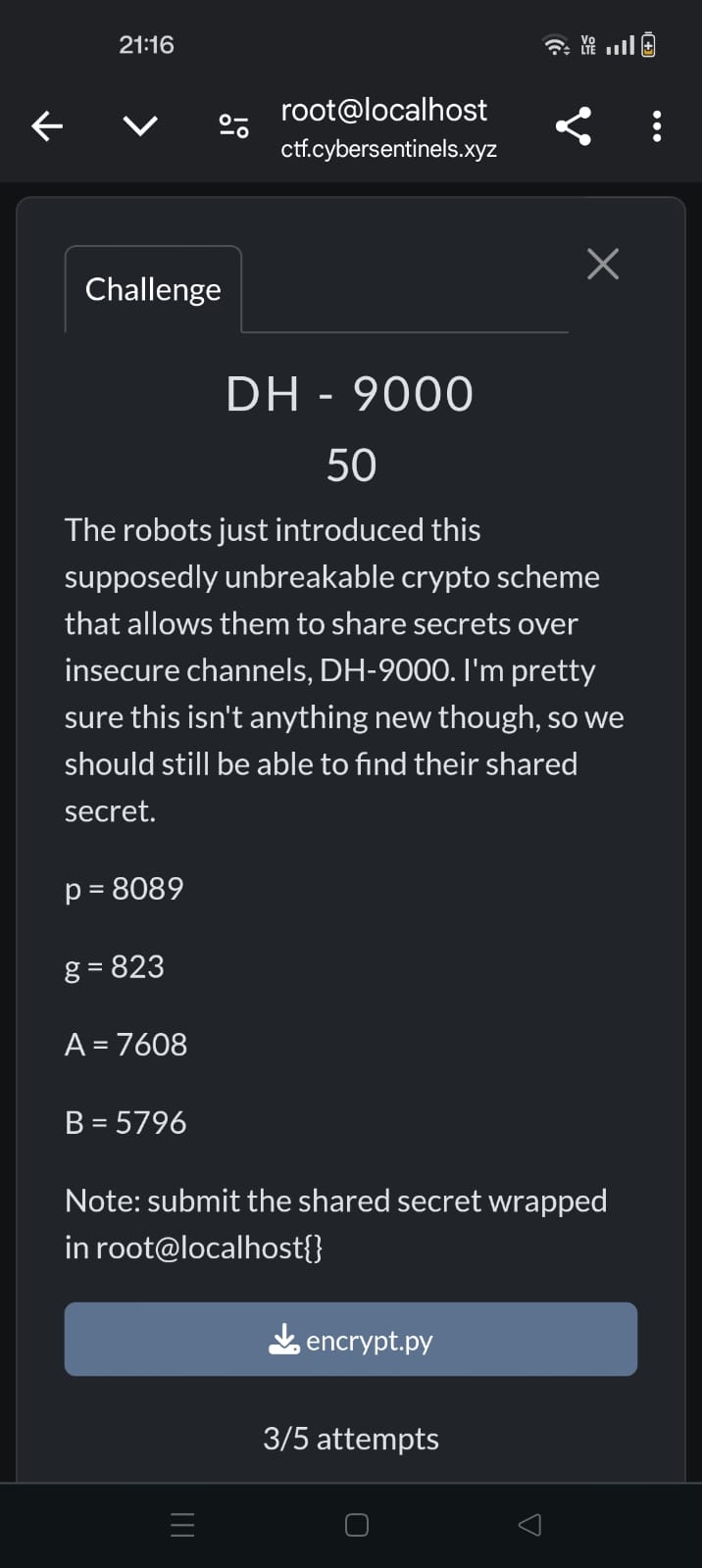
****

**To solve this challenge , We got to the website , it basically converts any text we give to lowercase, we need to give a simple xss malicious code to the website and it reveals the flag.**

**Decode The Hex value:**

**For this challenge , we need to convert the binary hex to text using any online compiler ,and we get the final answer.**

**DH – 9000:**

****

**In this challenge , we need to find the shared secret key of A and B in the Diffie Hellman key exchange process. We have the mod and the random number , as well as the public key of A and B. Using this , we can calculate the private key a and b. using these values , we can give these values in the python program that they have input to find the final value.**

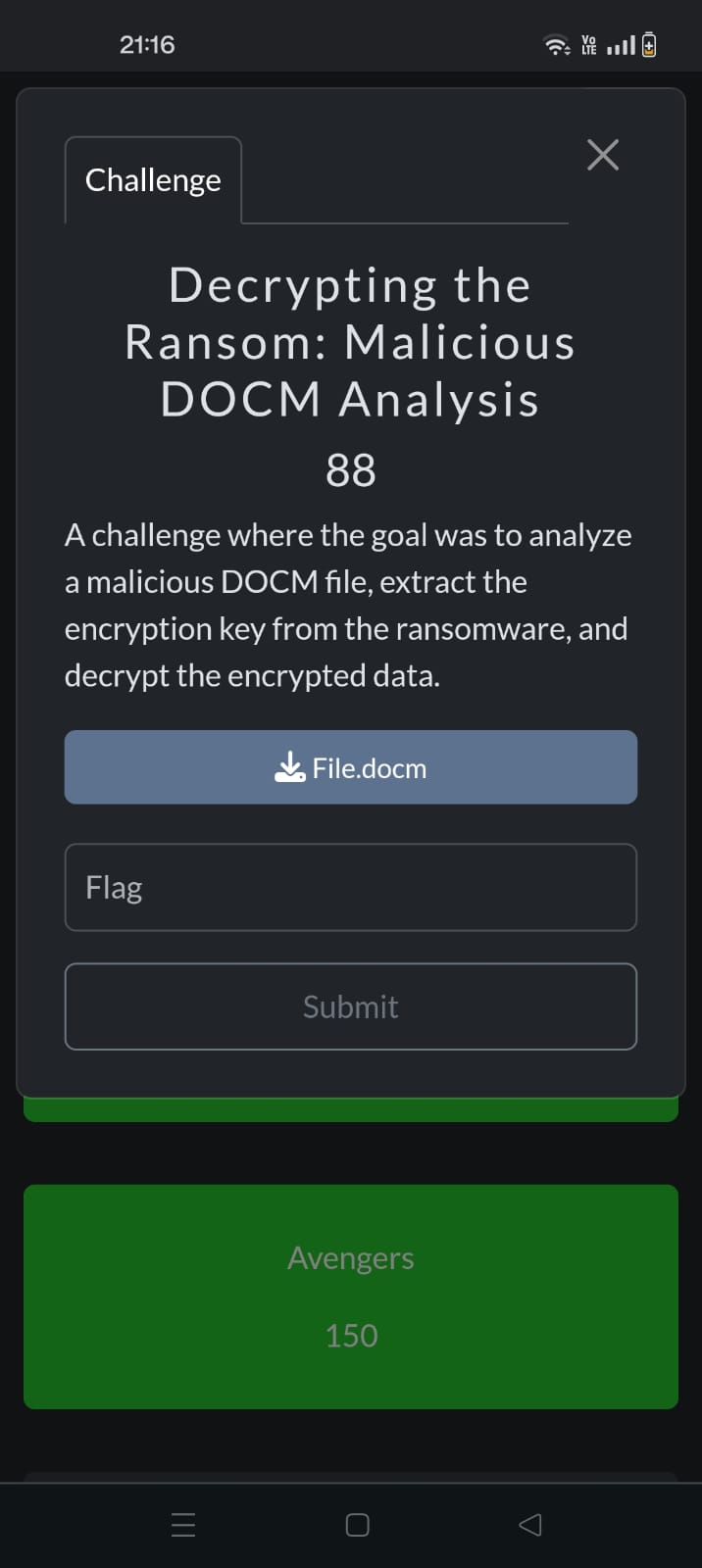
**Route-47:**

**In this challenge , we need to decrypt a ciphertext to reveal the flag , the encryption is pretty obvious as the title , it is ROT47 , using an online complier , decrypted the ROT47 encoded ciphertext to reveal the flag.**

**The Rail Conductor’s Secret:**

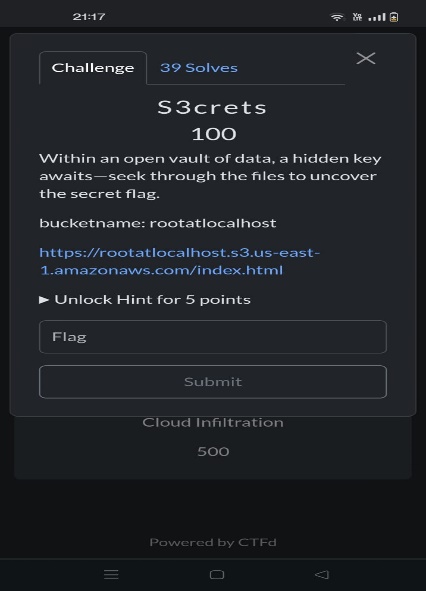
**In this challenge , we need to decrypt the ciphertext for the flag , the ciphertext is in the form of the flag, but the letters are different. This denotes that it is Caesar cipher . Using the clue in the question , and with the help of online compiler , we can retrieve the flag.**

**Decrypting the Ransom , Malicious DOCM Analysis:**



**In this challenge , we need to decrypt this DOCM file and retrieve the flag , using olevba command , we found a base64 encoded text , decoded it and found the flag.**

**S3crets:**



**In this challenge , we visit the website , given the clue in the question, It is an open vault , Therefore changing the URL By adding /flag.txt , we retrieved the flag.**