

CSAM 2021 CTF Solutions

Contents

CSAM 2021 CTF Solutions	2
Cryptography	2
50 The Swiss Army Knife	2
100 Corporate Caesar	2
200 C-XOR	3
300 RSA 256: Time to Decrypt	3
Network Capture	4
50 Baby Shark, doo doo doo	4
100 BBQ PIT	4
200 TFA Zip Files	5
300 HTTPS SSL	6
Not Cryptography	8
50 A Key Difference.....	8
100 2 Bits, 4 Bits, 8 Bits, a Dollar!	8
200 No Crying in Base64	8
300 Hash 4 Cash.....	9
OSINT	9
50 Google It.....	9
100 Lost Wallet	10
200 Mountain Hiking	10
300 Where in the world is Renee	12

Reverse Engineering	13
50 Three Letter Agency.....	13
100 String Theory.....	13
200 Matrix Mechanics	14
300 Quick Draw McGraw	14
Steganography	15
50 MITRE ATT^CK Matrix	15
100 Flag Officer	15
200 Clear Channel.....	16
300 Last Surviving Battlestar.....	18

CSAM 2021 CTF Solutions

Cryptography

50 The Swiss Army Knife

What intelligence agency runs the online "CyberChef" tool sometimes called the Cyber Swiss Army Knife - a web app for encryption, encoding, compression and data analysis

Find the intelligence agency which publishes **CyberChef**

The URL for CyberChef is <https://gchq.github.io/CyberChef>

Google "CyberChef gchq" will get you to the GCHQ CyberChefk github repository.

Looking at the github user "GCHQ" you can confirm this is the UK GCHQ intelligence agency.

FLAG: GCHQ

100 Corporate Caesar

Solve this caesar cipher:

PRUH KXPDQ WKDQ KXPDQ

Solve the Caesar cipher: **PRUH KXPDQ WKDQ KXPDQ**

There are online tools you can use to solve. If using CyberChef, you need to set to only upper case letters (note only upper case in the cipher text):

[https://gchq.github.io/CyberChef/#recipe=ROT13\(false,true,false,23\)&input=UFJVSCBLWFBEUSBXSORRIEtYUERR](https://gchq.github.io/CyberChef/#recipe=ROT13(false,true,false,23)&input=UFJVSCBLWFBEUSBXSORRIEtYUERR) (rotate 23)

Or you can use an 'auto solve' tool such as the one at <https://www.dcode.fr/caesar-cipher> (rotate 3)

FLAG{MORE_HUMAN_THAN_HUMAN}

200 C-XOR

Not Caesar, C-XOR

Find the key to decrypt this base64 encoded XOR encrypted message to see it.

BQ8CBDgNBhsWEG57HBEGEw8KAAINFz4=

The challenges asks you to find the key to decrypt this base64 encoded XOR encrypted message to see it.

BQ8CBDgNBhsWEG57HBEGEw8KAAINFz4=

There is no key provided, so you must brute force this or note that the name of the challenge provides the key which is the single char 'C'

You can solve this in Cyber Chef

1. Add the base64 string into the input panel
2. Add the 'from base64' decoder widget to the recipe panel
 - a. The output is mostly 'unprintable' dots
3. Add the "XOR Brute Force" widget to the recipe panel
 - a. Guess that the cipher text include 'FLAG' and enter that as the crib
4. Flag is seen in output

[https://gchq.github.io/CyberChef/#recipe=From_Base64\('A-Za-z0-9%2B/%3D',true\)XOR_Brute_Force\(1,100,0,'Standard',false,true,false,'FLAG'\)&input=QIE4Q0JEZ05CaHNXRUC1N0hCRUdFdzhLQUFJTkZ6ND0](https://gchq.github.io/CyberChef/#recipe=From_Base64('A-Za-z0-9%2B/%3D',true)XOR_Brute_Force(1,100,0,'Standard',false,true,false,'FLAG')&input=QIE4Q0JEZ05CaHNXRUC1N0hCRUdFdzhLQUFJTkZ6ND0)

FLAG{NEXUS-8_REPLICANT}

300 RSA 256: Time to Decrypt

Just as replicants have limited life spans, so do encryption methods.

Crack this public RSA 256 key to decrypt the flag in cipher.txt

The challenge asks you to crack a public RSA 256 key to decrypt the cipher text.

First you need to 'crack' the public key to create a private key by extracting the modulus.

The .pem and .txt file can be found on the slack as part of the bulk upload of files.

Network Capture

50 Baby Shark, doo doo doo

What is the most popular tool for examining network packet capture files and runs on both Windows and Linux?

The most common GUI network packet capture viewing tool is "Wireshark"

FLAG{WIRESHARK}

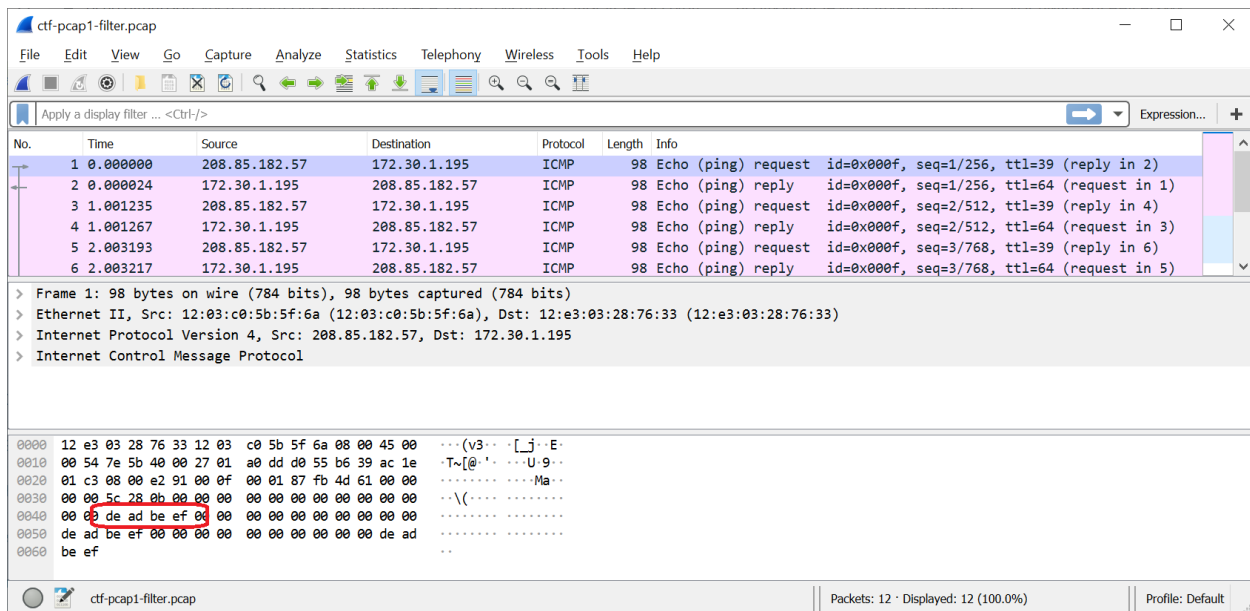
100 BBQ PIT

Find a flag in this PCAP capture.

Pcap file provided.

This flag is only a simple phrase without the "FLAG{}" bracket format and is tasty with BBQ sauce.

The flag can be found in every ICMP packet

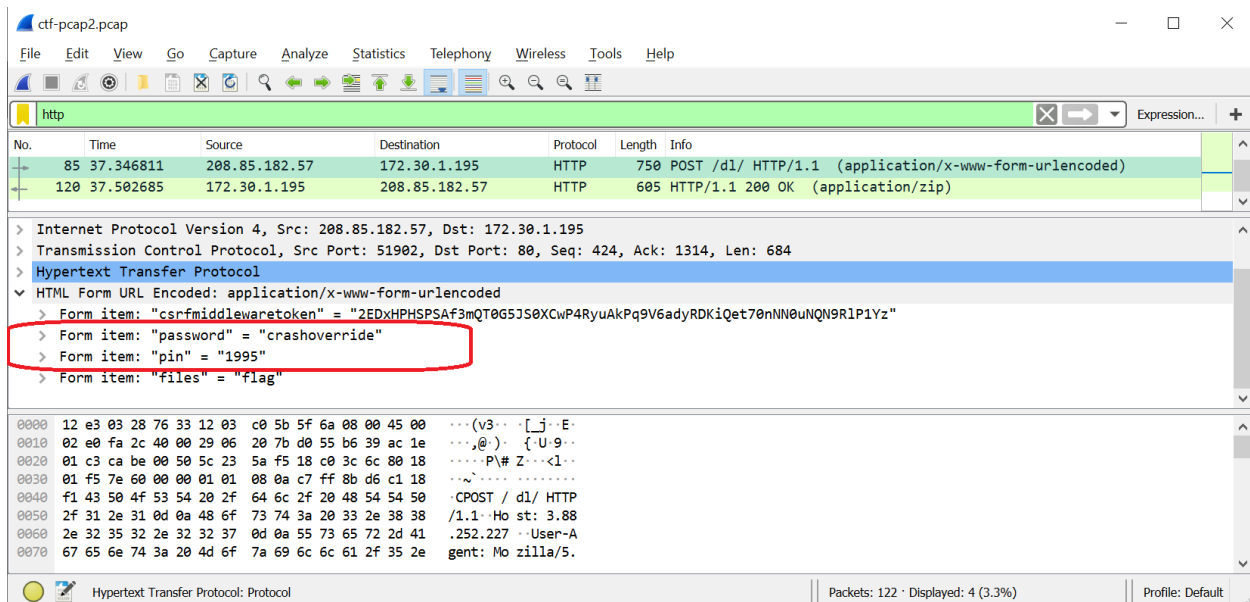


FLAG{deadbeef}

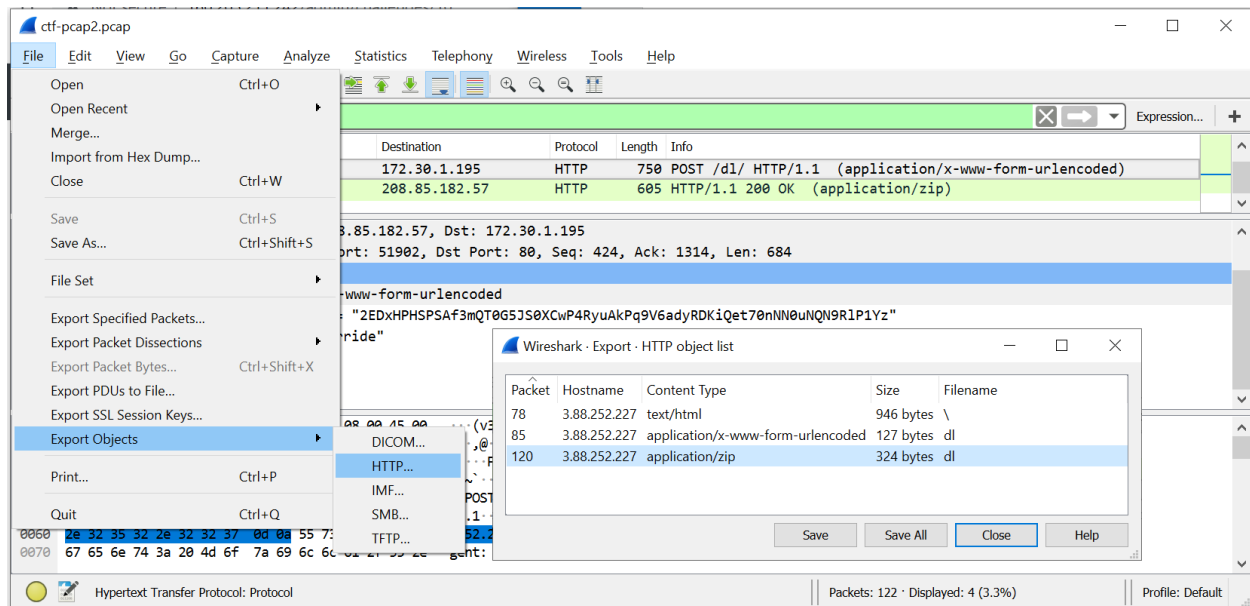
200 TFA Zip Files

This challenge requires the location of two passwords and the extraction of a zip file from the capture.

You can find the credentials in the form data of packet 84.



You can extract the zip file using the “File > Export objects > HTTP” menu. Select the “application/zip” data from packet 120 and save as “pcap2.zip”



When you double click the pcap2.zip, it prompts for a password. Enter the PIN ‘1995’

When you double click on ‘flag.txt’, it prompts for a password. Enter “**crashoverride**”

The flag is decrypted.

FLAG{WEYLAND-YUTANI}

300 R2D2 TLSv1.2

Our flag generator generates a unique flag every time you run it. There are 330 quadrillion possibilities. Give it a try!

docker pull rblm/ctf-ssl:1.0

> ****LEIA****: At least the information in Artoo is

> still intact.

>

> ****HAN****: What's so important? What's he

> carrying?

>

> ****LEIA****: The technical readouts of that battle
> station. I only hope that when the
> data is analyzed, a weakness can be
> found. It's not over yet!

Can you find the unique flag generated for this challenge?

> ****WEDGE****: That's impossible even for a computer.
>
> ****LUKE****: It's not impossible. I used to bull's-
> eye womp rats in my T-sixteen back
> home. They're not much bigger than
> two meters.

This challenge requires you to decrypt the TLSv1.2 traffic. To do so you need either a session key or the server key. Neither is provided.

The challenge text includes a bit encouraging you to try the random flag generator yourself and provides the docker command "*docker pull rblm/ctf-ssl:1.0*".

There are two ways to obtain the server key "ssl.key"

1. Docker.
 - a. Docker pull rblm/ctf-ssl:1.0
 - b. Docker run --name ctf rblm/ctf-ssl:1.0
 - c. Docker exec -it ctf /bin/bash
 - d. Hunt around and find /opt/app/certs/ssl.key
2. Github.
 - a. Visit Docker Hub for the container
 - i. <https://hub.docker.com/r/rblm/ctf-ssl>
 - b. Note the github url
 - i. <https://github.com/rblm/ctf-ssl>
 - c. Visit the github site, explore, and find
 - i. <https://github.com/rblm/ctf-ssl/blob/main/app/cert/ssl.key>

pcap file included in slack bulk upload.

Not Cryptography

50 A Key Difference

One difference between encryption and encoding is that encryption requires a __ to decrypt the message while an encoded message can be decoded without one.

The key difference between encryption and encoding is that encryption requires a key to decrypt while encoding can be decoded without a key.

FLAG{KEY}

100 2 Bits, 4 Bits, 8 Bits, a Dollar!

Decode this hexadecimal encoded ascii message:

464c41477b494e464f524d4154494f4e5f4f5645524c4f41447d

Each ASCII character can be encoded as a 2 character hex value. For example, the ascii character 'A' is encoded as the decimal value of 65, or the hex value of 41 or the binary value of '01000001'. All of which mean 'A' in ASCII encoding. Online converters are available.

464c41477b494e464f524d4154494f4e5f4f5645524c4f41447d

Using CyberChef,

1. Enter the hex string into the input panel
2. Add the 'From hex' widget into the recipe panel
3. The flag is displayed in the output

FLAG{INFORMATION_OVERLOAD}

200 No Crying in Base64

Decode this base64 encoded message from Johnny Mnemonic

SSBJYW4gY2FycnkqbmVhcmx5IGVpZ2h0eSBnaWdzIG9mIGRhGEgaW4gbXkgaGVhZC4gRkxBR3tFSUdIVFlfR0I
HU19PRI9EQVRBfQ==

Base64 encoding is commonly used to transfer information that might include non-printable characters or characters that are 'illegal' in certain contexts such as HTTP URLs or web based forms. Sometimes, images are stored as base64 encoded strings.

SSBjYW4gY2FycnkgbmVhcmx5IGVpZ2h0eSBnaWdzIG9mIGRhdGEgaW4gbXkgaGVhZC4gRkxBR3tFSUdIVFlfR0I
HU19PRI9EQVRBfQ==

1. Enter the base64 encoded string into the input panel
2. Add the 'From Base64' widget into the recipe panel
3. The flag is displayed in the output

FLAG{EIGHTY_GIGS_OF_DATA}

300 Hash 4 Cash

We are hacking into Johnny Mnemonic's data implant and need a code but all we have found is the following hash. You typically can't reverse a hash, but you can sometimes find a match by hashing word lists and creating look-up tables. We suspect that this hash is based off some word in the Johnny Mnemonic movie transcript. Happy Hunting!

0641dafe4fa984daf519573ab7f8e8a6

Find a word in the Johnny Mnemonic script that matches the hash: 0641dafe4fa984daf519573ab7f8e8a6

We need to know what sort of hash this is. One online site is <https://www.tunnelsup.com/hash-analyzer/>

This site suggests MD5 or MD4. Since MD5 is much more common, assume that.

OSINT

50 Google It

Hackers search Google with specially crafted queries which are sometimes called Google _ _ _ _ _

To solve, copy the question into Google and search for an answer. We accept either DORKS or HACKS

FLAG{DORKS}

FLAG{HACKS}

100 Lost Wallet



Ryan was walking his pet cat through the streets one night when he realized he dropped his wallet. Needing this to pay for his cat food, he walked back and forth on the path trying to locate the wallet. Unable to locate it, he snaps a picture of a local business to see if perhaps they found it. The only problem is, he doesn't know who owns the building he took a picture of. This is where you come in :

Who owns this building? (The main company that operates in the building)

Download the file OSINT-challenge-1.jpg.

Upload the file into an image search engine such as TinEye <https://tineye.com>

There are a couple of responses returned. One is labeled 'lockheedmartin.jpg' and is linked to

<https://www.huntsvillescoop.com/companiesspacedefense.html>

Click on the URL and you can see this building is listed as a Huntsville, Alabama company.

You can search for "Lockheed Martin Huntsville Alabama" in google maps and view photos of the three sites LM has there. It is quickly apparent that this is the 4800 Bradford Dr site.

<https://goo.gl/maps/G15GsQwVDpHCJdPg6> (view from nearby street)

200 Mountain Hiking



Oliver is what some would call a Mountain Man, others would just call him crazy. Not too long ago, Oliver went on a crazy hike that took him through the Imogene Pass. He claimed it was part of a rocky challenge or something out in Telluride. It has been some time since he has last checked in and his daughter Janice would like to know where he is. Using this picture, can you determine what mountain he hiked on?

Mt. Sneffels, near Ouray Colorado.

300 Where in the world is Renee



After Renee spent her night in this new place, she wanted to leave and find the garden view. She heard it on good authority it was near the garden. Walking down a long street, avoiding the student traffic of a nearby school, she was unable to find it. Although she took this picture, she still isn't quite sure where she is. However, she was assured the area she was looking for was outback. On what street was this picture taken?

Reverse Engineering

50 Three Letter Agency

The NSA released this free and open source reverse engineering tool in 2019.

NSA released the reverse engineering tool (disassembler) Ghidra as free and open source.

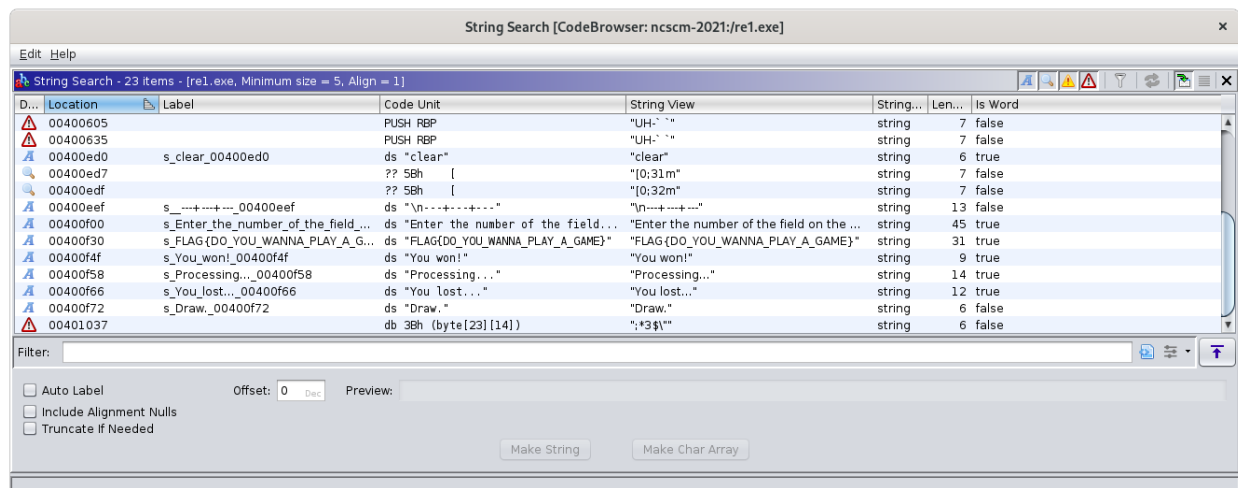
FLAG{GHIDRA}

100 String Theory

Find the flag in this Linux binary executable.

Compiled for CentOS 7

You could run the 'strings' command on this binary from a Linux host and find the flag. Here we see it in as Ghidra displays it when searching the binary for strings.



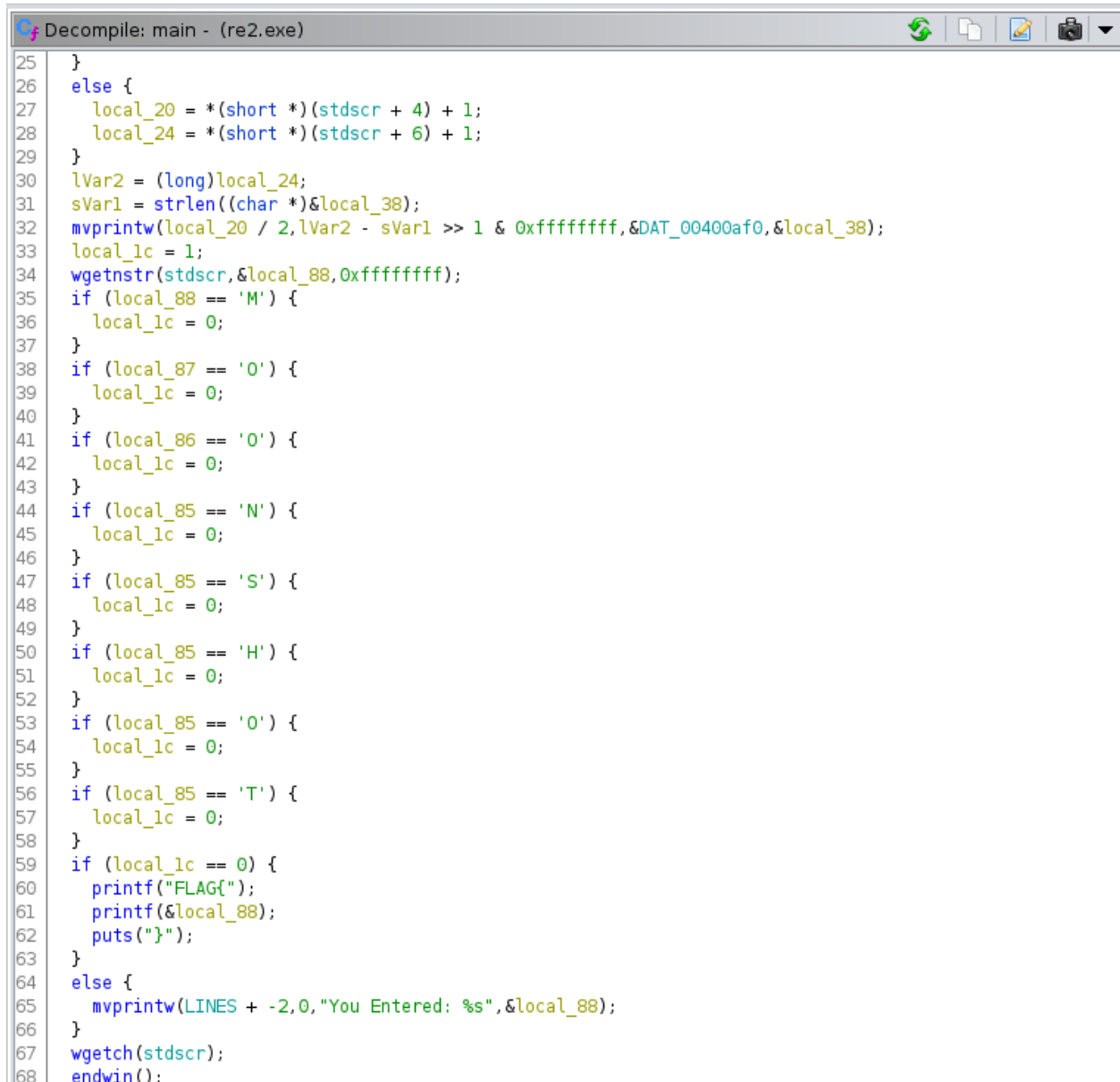
FLAG{DO_YOU_WANNA_PLAY_A_GAME}

200 Matrix Mechanics

Find the flag in this Linux binary executable.

Compiled for CentOS 7.

To thwart simple string searches, this binary defines the array one character at a time. Here we see it as displayed in the Ghidra decompiler window.

The image shows a screenshot of the Ghidra decompiler window. The title bar reads "Decompile: main - (re2.exe)". The code is written in C and is decompiled from assembly. It shows a series of conditional checks for characters 'M', 'O', 'O', 'N', 'S', 'H', 'O', 'T' in local_88. If all characters are found, it prints "FLAG{" followed by the contents of local_88 and a closing brace. Otherwise, it prints "You Entered: %s" with local_88. The code is as follows:

```
25 }
26 else {
27     local_20 = *(short *) (stdscr + 4) + 1;
28     local_24 = *(short *) (stdscr + 6) + 1;
29 }
30 lVar2 = (long) local_24;
31 sVar1 = strlen((char *) &local_38);
32 mvprintw(local_20 / 2, lVar2 - sVar1 >> 1 & 0xffffffff, &DAT_00400af0, &local_38);
33 local_1c = 1;
34 wgetnstr(stdscr, &local_88, 0xffffffff);
35 if (local_88 == 'M') {
36     local_1c = 0;
37 }
38 if (local_87 == 'O') {
39     local_1c = 0;
40 }
41 if (local_86 == 'O') {
42     local_1c = 0;
43 }
44 if (local_85 == 'N') {
45     local_1c = 0;
46 }
47 if (local_85 == 'S') {
48     local_1c = 0;
49 }
50 if (local_85 == 'H') {
51     local_1c = 0;
52 }
53 if (local_85 == 'O') {
54     local_1c = 0;
55 }
56 if (local_85 == 'T') {
57     local_1c = 0;
58 }
59 if (local_1c == 0) {
60     printf("FLAG(");
61     printf(&local_88);
62     puts("}");
63 }
64 else {
65     mvprintw(LINES + -2, 0, "You Entered: %s", &local_88);
66 }
67 wgetch(stdscr);
68 endwin();
```

FLAG{MOONSHOT}

300 Quick Draw McGraw

Can you beat McGraw to the draw in this Linux binary executable?

Compiled for CentOS 7.

One way to solve this is to recompile with a game cheat. The only way to get the flag is to draw 'faster' than the computer by hitting the 'return' key after a timer. But the best you can do is 'draw' at the same time (rounded off at seconds). So how to win the flag? Cheat. Change the code so you get the flag if you 'draw' slower or equal to the computer.

Examine the code in Ghidra.

Then figure it out!

Steganography

50 MITRE ATT^CK Matrix

Under what MITRE ATT&CK techniques is Steganography listed?

(there are two; either one will solve the challenge)

There are two techniques which include steganography as a sub-technique. Use the search bar to find them. Either flag will solve the challenge.

<https://attack.mitre.org/techniques/T1001/002/>

<https://attack.mitre.org/techniques/T1027/003/>

FLAG{Data Obfuscation}

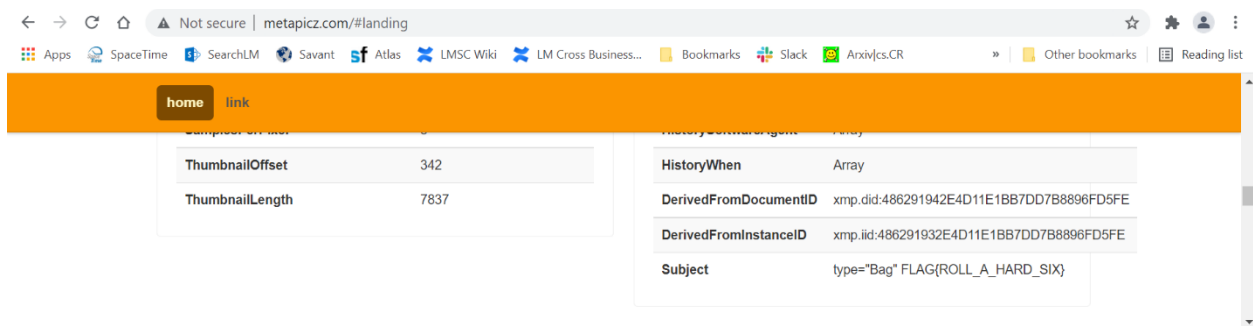
FLAG{Obfuscated Files or Information}

100 Flag Officer

Find the hidden flag in this image file of Admiral Adama of the Colonial Fleet.



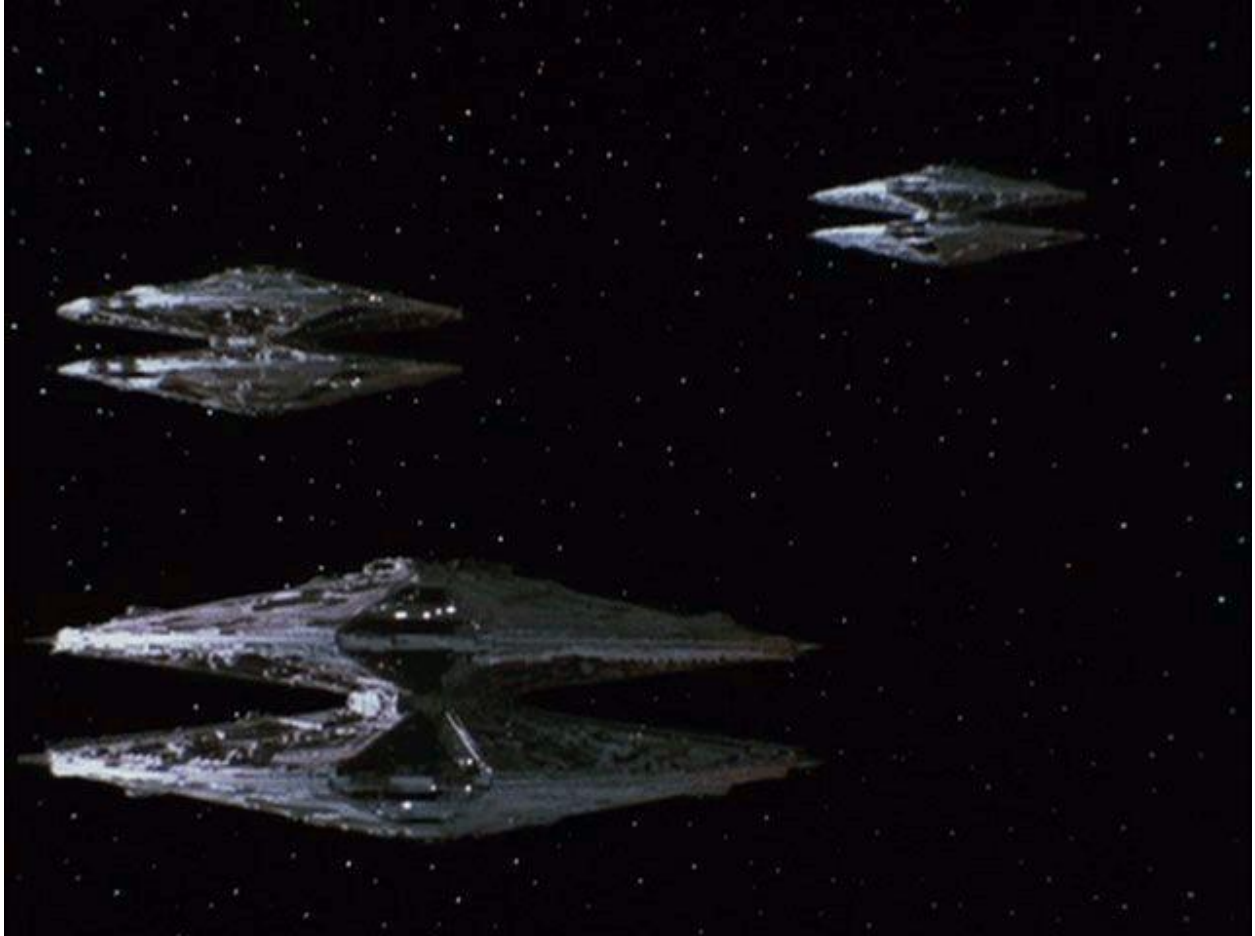
Text data can be stored in a file as metadata. Tools can read this metadata. On Linux, exiftool is one such program. There are also online exif data readers. <https://metapicz.com> is one such. Beware, that such sites could be scraping data including personal data. But nothing personal or proprietary is included in this file.



FLAG{ROLL_A_HARD_SIX}

200 Animating The Toasters

Send a message to the frakkin toasters!



There is a hint in the title. This image is a GIF animation. Use a photo editor such as Photoshop or GIMP to open the image. Note that each frame is a layer in the animation. One layer is rated as '0ms' display time. Examine it closely (after disabling visibility on the other images). You can find the text on the image. Various tools can enhance the readability of needed (ie, shift color levels to towards the dark end of spectrum).



FLAG{ALL_YOUR_BASE_ARE_BELONG_TO_US}

300 Last Surviving Battlestar

A picture is worth a 1000 words or, as in this image, 27 words. Can you find the meaning within this image of Galactica reaching Earth?



There is a hint in the initials of the title of this challenge: Last Surviving Battlestar -> LSB

LSB is also the initials of "Least Significant Bit" which is a well known steganography technique.